
Metrics reloaded: recommendations for image analysis validation

In the format provided by the authors and unedited

Supplementary Notes – Metrics Reloaded

Content of Supplementary Notes

Our metric recommendations are detailed in the following sections.



SUPL. METHODS

Delphi process, expert consortium, reference implementation, web-based tool



Step 1 - Problem Fingerprinting (SUPL. NOTE 1)

General instructions (Suppl. Note 1.1), problem category mapping (Suppl. Note 1.2), generation of the problem fingerprint (Suppl. Note 1.3)



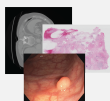
Step 2 - Metric Selection (SUPL. NOTE 2)

Metrics Reloaded pool of reference-based metrics (Suppl. Note 2.1), recommendations for metric selection (Suppl. Notes 2.2-2.6), decision guides (Suppl. Note 2.7)



Step 3 - Metric Application (SUPL. NOTE 3)

Metric cheat sheets (Suppl. Note 3.1)



Recommendations for selected use cases (SUPL. NOTE 4)



Terminology and Notation (SUPL. NOTE 5)

Symbols (Suppl. Note 5.1), expected formats (Suppl. Note 5.2), acronyms (Suppl. Note 5.3), glossary (Suppl. Note 5.4)

Overview of relevant content for Image-level Classification (ImLC), Semantic Segmentation (SemS), Object Detection (ObD) and Instance Segmentation (InS) problems.

	ImLC	SemS	ObD	InS
Problem fingerprint (SUPL. NOTE 1)	Figs. SN 1.7 - SN 1.9	Figs. SN 1.10 - SN 1.11	Figs. SN 1.12 - SN 1.14	Figs. SN 1.15 - SN 1.17
Recommendations for metric selection (SUPL. NOTE 2)	Suppl. Note 2.2; Calibration: Suppl. Note 2.6	Suppl. Note 2.3	Suppl. Note 2.4	Suppl. Note 2.5
Metric cheat sheets (SUPL. NOTE 3)	Suppl. Note 3.1	Suppl. Note 3.1	Suppl. Note 3.1	Suppl. Note 3.1
Instantiation for common use cases (SUPL. NOTE 4)	Fig. SN 4.1	Fig. SN 4.2	Fig. SN 4.3	Fig. SN 4.4

SUPPLEMENTARY METHODS

Delphi process

We compiled the recommendations provided by the *Metrics Reloaded* framework by assembling an international expert consortium which then underwent a multi-stage Delphi process. A Delphi process is a structured group communication process that serves to gather opinions from an expert panel via a series of individual interrogations, usually in the form of questionnaires, interspersed with feedback from the respondents [18]. The technique is widely used for establishing consensus among experts in medicine, particularly in the development of best practices in areas where evidence may be limited, conflicting, or absent [77]. The initial panel participating in our Delphi process comprised 30 international biomedical image analysis experts representing 25 institutions. Member selection was initially based on membership in one of the three initiatives that triggered this research, namely the Biomedical Image Analysis Challenges (BIAS) initiative, the Medical Open Network for Artificial Intelligence (MONAI) Working Group for Evaluation, Reproducibility and Benchmarks, and the Medical Image Computing and Computer Assisted Interventions (MICCAI) Special Interest Group for Challenges (previously MICCAI board working group). To reflect as broad a range of imaging domains as possible and expand the available expertise, the number of consortium members was gradually increased from the initial 30 to a final number of 73. The members provided a wide range of expertise ranging from biology, medicine, epidemiology and biomedical image analysis all the way to statistics, mathematics and computer science. Furthermore, leading members of major standardization initiatives were included, such as the Enhancing the QUALity and Transparency Of health Research (EQUATOR) network, from which imaging and clinical guidelines have originated, including CONSORT/CONSORT-AI [30, 91], TRIPOD/TRIPOD-AI [28, 73], STARD/STARD-AI [16, 95], and others.

Overall, the process comprised six distinct stages and encompassed five workshops and nine surveys before a final Delphi consensus voting was performed. Each survey was developed by the Metrics Reloaded core team and taken by the remaining members of the consortium; in other words, the researchers that designed the surveys did not take part in them. Upon completion, the core team then analyzed the results, discussed them with team members where necessary, and integrated the feedback, thus iteratively refining the framework. The main stages of the compilation and consensus building process are detailed in the following:

- 1. Initialization.* A kickoff workshop was held in December 2020 with the primary goal of deciding on the concrete scope of the recommendation framework. Prior to the workshop, an initial survey had been conducted with a focus on gathering relevant literature as well as theoretical and practical failure cases of metrics in the broader scope of classification, segmentation and detection. Based on the discussions at the workshop, a series of three surveys was issued whose responses resulted in (1) a joint terminology (see Suppl. Note 5), (2) inclusion criteria for the paper, namely the decision to cover classification tasks at image/object and pixel level (Fig. 4), (3) a shortlist of relevant metrics for each category (whose subsequent refinement resulted in Tab. SN 2.1), and (4) an initial set of fingerprint items, whose refinement resulted in the final fingerprints presented in Suppl. Note 1.3. It was further decided by the consortium that the choice of problem category should be part of the recommendation itself (covered in Subprocess S1, Extended Data Fig. 1).

- 2. Compilation of first draft of recommendations in expert groups.* The primary purpose of the second Delphi workshop in June 2021 was the formation of expert groups that should coordinate individual task forces. Five expert groups were initially formed; three dedicated to the problem

categories addressed in the framework (one for image-level classification, one for semantic segmentation and one for object detection and instance segmentation), plus a biomedical expert group and a cross-topic expert group. The task of the expert groups corresponding to the problem categories was to develop recommendations for their respective categories that address the pitfalls compiled in the sister publication [86] (now captured in Fig. 2 and the nine subprocesses in Extended Data Figs. 1-9). The task of the cross-topic group was to identify and tackle metric-related issues going beyond pure metric selection, such as metric aggregation, reporting and implementation, statistical considerations, rankings, and biases (now captured in Extended Data Tab. 1). The task of the biomedical expert group was to ensure that the recommendation framework would satisfy the needs of domain experts, such as clinicians, and to identify relevant biomedical scenarios (now captured in Suppl. Note 4). Group-specific surveys were issued to support the work of the task forces. To give the experts enough freedom, no specific restrictions were imposed with respect to how the individual groups would arrive at their recommendations. In the following third Delphi workshop, held in October 2021, the expert group leaders discussed their preliminary results with the entire core team.

3. Consolidation of recommendations by Metrics Reloaded core team. Once the expert groups had finalized their initial recommendation drafts for the problem categories, the *Metrics Reloaded* core team consolidated and harmonized the recommendations in close collaboration with the groups. In the fourth Delphi workshop in January 2022, the resulting decision trees capturing the core recommendations (Fig. 2; S1-S4, S6-S9) were presented and discussed.

4. Revision by Metrics Reloaded consortium. The decision trees were then subjected to internal tests by members of the consortium and their teams. The *Metrics Reloaded* core team incorporated the survey-based feedback in close collaboration with the expert groups. The first draft of the entire framework was then presented and discussed at the fifth Delphi workshop in March 2022.

5. Crowdsourcing of feedback. Finally, community feedback was obtained via a social media campaign. The recommendation framework was released on arXiv [66], and a survey link was sent by the *Metrics Reloaded* core team to various mailing lists, as well as posted on the social media platforms LinkedIn, with the hashtag #imageanalysis, and Twitter, where we tagged various relevant accounts, e.g., @MICCAIStudents, @WomenInMICCAI, @midl_conference, @ELLISforEurope, @ProjectMONAI, and @naturemethods. The original tweet received more than 42,000 impressions. All co-authors were asked to distribute the survey among their colleagues and societies. Furthermore, the survey link was added to the released arXiv version. The survey was initially opened at the beginning of June 2022 and closed at the end of July 2022. The community had the choice between submitting one-click feedback or detailed feedback by answering questions on the comprehensiveness and usefulness of our approach, the specific mappings, as well as voicing concerns or questions. In addition, we specifically asked for which biomedical use cases the framework should be instantiated. All contributors were given the choice to be included in the acknowledgements. A total of 186 researchers participated in the survey. Of those, 82 provided feedback in the form of free text answers. 58 participants chose to give detailed feedback rather than one-click feedback. A total of 46 researchers wished to be mentioned in the acknowledgements and provided their names. Contributors who provided substantial feedback were invited into the consortium (seven in total). The social media survey was used as a basis to select biomedical use cases for which the framework was instantiated. Based on the feedback, we designed the metric cheat sheets (see Suppl. Note 4). The implementation of the web toolkit was highly encouraged by several survey participants. Moreover, in response to the feedback, an additional expert group on the topic of

calibration was established with newly recruited consortium members, which led to the generation of the calibration recommendations captured in S5 (Extended Data Fig. 5). A revised framework, with the community feedback integrated (e.g., including new classification metrics, such as the EC), was presented to the consortium in another survey, based on which the Metrics Reloaded core team compiled the final recommendations (captured in Fig. 2 and S1-S9) that served as basis for the final Delphi-based consensus building.

6. Final Delphi consensus building. In the final stage, an accelerated final Delphi process was initiated to vote for the ten core components of the recommendation framework (Fig. 2 and Subprocesses S1-S9). In response to the consortium’s comments, final modifications to the calibration recommendations were made. After two rounds of revisions to S5, the final recommendation received strong support (only one member disagreed). For all other nine components, the first round had already resulted in a very strong consensus (disagreement 0%-7%). Minor modifications, primarily concerning formatting and style, were communicated to the entire consortium whose members were then given the opportunity to veto any of the changes, which none of the consortium made use of.

Expert consortium

The expert consortium consisted of a total of 73 researchers (73% male, 27% female) from a total of 65 institutions. The majority of experts (52%) were professors, followed by postdoctoral researchers (37%). The median h-index of the consortium was 34 (mean: 27; minimum: 6; maximum: 113) and the median academic age was 18 years (mean: 19; minimum: 3; max: 42). Experts were from 18 countries and 5 continents. 66% of experts had a technical, 7% a clinical, 3% a biological, and 24% a mixed background. From the 65 institutions, we could identify the number of employees for 88% of institutions. From those, the majority of institutions had a size between 1,000 and 10,000 employees (58%), followed by even larger institutions between 10,000 and 100,000 employees (25%), and smaller institutions below 1,000 employees (16%). Only a small portion of institutions were above 100,000 employees (2%).

Reference implementations

To overcome pitfalls related to metric implementation [86], we provide reference implementations for all *Metrics Reloaded* metrics within the MONAI open-source framework. They are accessible at <https://github.com/Project-MONAI/MetricsReloaded>.

Web-based tool

The recommendation framework was implemented as a web-based tool, which guides the users through the entire recommendation processes of Fig. 2. The core advantage of the tool compared to the decision trees depicted in S1-S9 is the fact that the tool automatically restricts the visualization only to the relevant information that is required in each specific step and for the specific use case. It further provides comprehensive profiles of all metrics contained in the *Metric Reloaded* pool.

The *Metric Reloaded* tool is available at <https://metrics-reloaded.dkfz.de>.

SUPPL. NOTE 1 STEP 1 - PROBLEM FINGERPRINTING

The first step in the framework *Step 1: Problem fingerprinting* (Fig. 2) requires the user to read the general instructions provided in Suppl. Note 1.1, perform the problem category mapping according to Suppl. Note 1.2, and generate the corresponding problem fingerprint as detailed in Suppl. Note 1.3.

1.1 General Instructions

Users of the *Metrics Reloaded* framework should read the following instructions prior to metric selection.

Inclusion criteria. The *Metrics Reloaded* framework currently considers problems in which categorical target variables are to be predicted based on a given n -dimensional input image. Hence, it covers a broad range of imaging modalities from classical 2D/3D modalities, such as fluorescence, computed tomography (CT) or X-ray imaging, to novel, for example spectral, imaging modalities that yield high-dimensional output per pixel [24]. Classification can occur at pixel, object or image level, resulting in the four problem categories covered by the framework and depicted in Fig. 4:

Image-level classification refers to the assignment of one or multiple category labels to the entire image or fixed regions/predefined locations within an image.

Semantic segmentation refers to the assignment of one or multiple category labels to each pixel. For many segmentation problems, object boundaries are generated in addition to the pixel-wise classification images, which enables the computation of distance-based metrics, such as the Normalized Surface Distance (NSD).

Object detection refers to the localization and categorization of an unknown number of structures.

Instance segmentation refers to the localization and delineation of each distinct structure of a particular class. It can be regarded as delivering the tasks of object detection and semantic segmentation at the same time. In contrast to object detection, instance segmentation also involves the accurate marking of the structure boundary. In contrast to semantic segmentation, it distinguishes different structures of the same class.

Notably, the four different categories are mathematically closely related (Fig. 4) as they typically rely on the generation of confusion matrices as a foundation of metric computation. Application examples for all categories can be found in Fig. 5. Importantly, *Metrics Reloaded* does not require an entire image to be provided as input for the validation. For example, the classification of a Region of Interest (ROI) within a medical image may be required. In this example, the framework would proceed with the ROI as input as if it was an entire image. Furthermore, the shape of the image/input does not need to be rectangular. Finally, context information may be provided along with the input. For example, medical images may be processed along with clinical data to arrive at a diagnosis; video frames may be processed along with preceding video snippets. Ultimately, only the algorithm *output* must correspond to an n -dimensional image.

Phrasing of the biomedical task. The recommendation framework has been designed in a way to support the metric selection and application process for one specific driving biomedical question. In practice, multiple questions are often addressed with one given data set. For example, a clinician may have the ultimate interest of diagnosing brain cancer in a patient based on a given magnetic resonance imaging (MRI) data set. While this would be phrased as an image-level classification

task, an interesting *surrogate task* could be that of segmentation to assess the quality of tumor delineation. In the case of multiple different driving biomedical questions, a recommendation is generated separately for each question. This specifically holds true for multi-label problems, in which multiple labels can simultaneously be assigned to the same image/object/pixel (e.g., multiple sclerosis and brain tumor both assigned to the same magnetic resonance image). In such a case, the problem should be converted to multiple binary problems, for which the framework is traversed individually.

Matching reference annotations. The metric selection process begins with the step of mapping a given problem with all its intrinsic and data set-related properties to the corresponding problem category via the *category mapping* shown in Extended Data Fig. 1. Our framework assumes that the reference annotations of the given data set meet the requirements of the identified problem. Expected formats for both the reference annotations and the algorithm output are provided in Suppl. Note 5.2.

Model-agnostic metric recommendation. Metrics should be chosen based solely on the driving biomedical problem and not be affected by algorithm design choices. For example, the error functions applied in common neural network architectures do not justify the use of corresponding metrics (e.g., validating with Dice Similarity Coefficient (DSC) to match the Dice loss used for training a neural network). Instead, the domain interest should guide the choice of metric, which, in turn, can guide the choice of the loss term.

Dealing with multiple classes. Multi-class metrics, such as Accuracy or Matthews Correlation Coefficient (MCC), have the unique advantage that they capture the performance of an algorithm for all classes in a single score without the need for customized class-aggregation schemes. On the other hand, they do not allow for detailed class-specific analyses. *Metrics Reloaded* therefore generally recommends performing a per-class validation for all classes (in addition to potential multi-class validation). Specifically in segmentation problems, problem properties may differ from class to class (e.g. the size or size variability of target structures). In these rare cases, the problem fingerprint needs to be generated separately for each class and several subprocesses (denoted by the \boxplus -symbol in the framework overview shown in Fig. 2) need to be traversed separately for each class. Although not common in current validation practice, this may - in theory - lead to different validation metrics for different classes. We speak of *class-specific metric pools* in this case, which are generated in addition to the multi-class metric pool.

Primary and secondary metrics. In general, biomedical interest cannot be captured with a single metric. The framework has therefore been designed to recommend multiple complementary metrics for a given task. We assume two main use cases for our framework. In **comparative benchmarking studies** (e.g., competitive challenges), multiple algorithms or algorithm variants are compared on identical data sets. This requires the ranking of the competing algorithms according to performance. Typically, multiple complementary validation metrics are applied in this use case, resulting in either multiple rankings or a merged ranking that takes all or several metric values into account. We refer to the metrics that contribute to the (primary) ranking(s) as *primary metrics*. While our framework focuses on the recommendation of primary metrics, users are invited to complement them with *secondary metrics* according to their specific needs. Secondary metrics can additionally be applied for comprehensive reporting, for example because they reflect complementary properties of interest (e.g., compute time, carbon footprint), or for providing performance measures that are comparable across publications. The computer vision community, for instance, typically reports

the Intersection over Union (IoU) rather than the DSC. The second use case of metrics addressed by our framework are **validation studies centered around a single algorithm** that focus on comprehensive diagnostics rather than comparative assessment. In this case, it is often desired to report as many complementary metrics as possible in order to comprehensively analyze the properties of an algorithm. Users interested in this second use case can ignore the discrimination between primary and secondary metrics.

Decision rule applied to predicted class scores. A classification system in practice operates by making decisions. Converting the raw continuous model outputs – the predicted class scores – into discrete decisions is achieved by determining an appropriate decision rule. Common options are detailed in Suppl. Note 1.3 (→ FP2.6: Decision rule applied to predicted class scores). While identifying the optimal decision rule for a classification system is beyond the scope of this work, it is important to know that the choice affects the selection of adequate validation metrics.

In binary tasks, defining an optimal decision rule boils down to determining a suitable cutoff (i.e., threshold) on predicted class scores (Fig. SN 1.1). In contrast, identifying an optimal decision rule for multi-class problems is generally more complex. A common, intuitive workaround for this challenge is to determine an individual decision rule for each predicted class score. However, this strategy implies that multiple decisions are made for the same input, thus fundamentally changing the task to multi-label classification (in this framework, multi-label classification is handled as separate binary tasks, as detailed in Suppl. Note 1.1 - inclusion criteria). Instead, in practice, a multi-class system requires a single global decision rule for all classes, which amounts to identifying optimal global 'decision regions'. The most common global decision rule is to simply select the class associated with the highest predicted class score, which is typically implemented as an 'argmax' operation and is also referred to as a 'maximum a posteriori' decision. Bayesian decision theory, however, shows that this argmax rule is only the optimal choice in case of equal severity of class confusions (FP2.5.2=False) and no compensation for class imbalances being requested (FP2.5.5=False). If one of these requirements is not fulfilled, a cost-dependent variation of the argmax-rule should be employed (see equation 44 in [40]). Further, the argmax decision rule assumes that predicted class scores are calibrated (see Section 2.6 for details on calibration). Fig. SN 1.1 showcases a hypothetical example of how argmax can be a suboptimal decision rule in combination with miscalibrated model outputs. While a variety of calibration metrics is discussed in Section 2.6, it should be noted that Expected Cost (EC) features a framework to directly validate the effect of a decision rule on the quality of associated decisions. Moreover, any measured negative effect can be associated with the miscalibration of scores, thus guiding users to enhance their decision making.

A further potential pitfall associated with the global decision rule of a classifier can occur when the validation of a multi-class problem is primarily based on multi-threshold metrics. This is because multi-threshold metrics, which do not rely on a decision rule, may conceal the fact that in practice, the optimal global decision rule will not be identified. Thus, the resulting metric scores may overestimate the decision-making performance of a model in practice.

Finally, an important consideration for identifying a decision rule of a classifier is that any data-based optimization or search must be performed on a separate data split different from the validation data. This consideration includes any configuration of re-calibration methods.

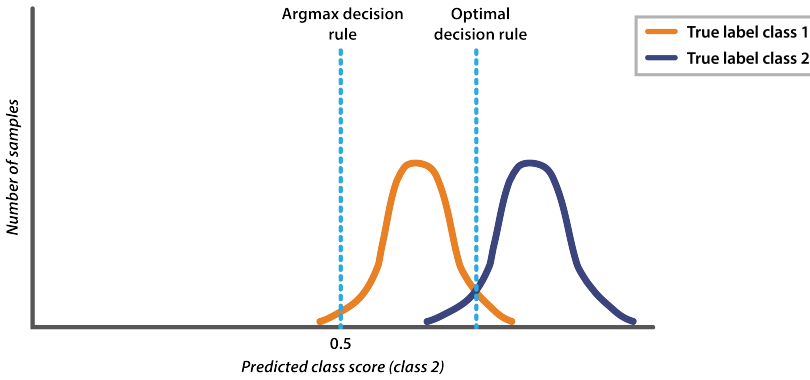


Fig. SN 1.1. Argmax decision rule for converting predicted class scores to a categorical label. Choosing the class with the highest predicted class score does not necessarily result in the best decision-making, as for example measured by Accuracy.

Notation. The notation for our recommendations has been based on Business Process Model and Notation (BPMN)¹. The individual components used in the recommendation diagrams are explained in Fig. SN 5.1. Please note that we do not strictly follow BPMN to improve clarity of presentation.

Terminology. Terminology may differ substantially across communities. For example, the statistics community prefers the term Positive Predictive Value (PPV) over *Precision*, as the latter can be confused with the mathematical precision (repeatability) term. In the medical domain, the term *validation* is used for an independent assessment (untouched test set) of an algorithm, while the machine learning community commonly uses a *validation set* for hyperparameter tuning. To avoid confusion resulting from unclear terminology, we follow the general terminology of [85] and have included a glossary in the Suppl. Note 5.4.

¹<https://www.omg.org/spec/BPMN/>

1.2 Problem Category Mapping

The problem fingerprinting (Step 1 in Fig. 2; see Sec. 1.3) begins with the step of mapping a given problem with all its intrinsic and data set-related properties to the corresponding problem category via the *category mapping* shown in Extended Data Fig. 1. This step is crucial for avoiding pitfalls related to the inappropriate choice of the problem category, as detailed in the sister publication of this work [86]. Specifically, when multiple instances of the same structure type can occur in an image, it is typically advisable to phrase the underlying problem as an object detection or instance segmentation problem rather than a semantic segmentation problem (Figs.1 and SN 1.2).

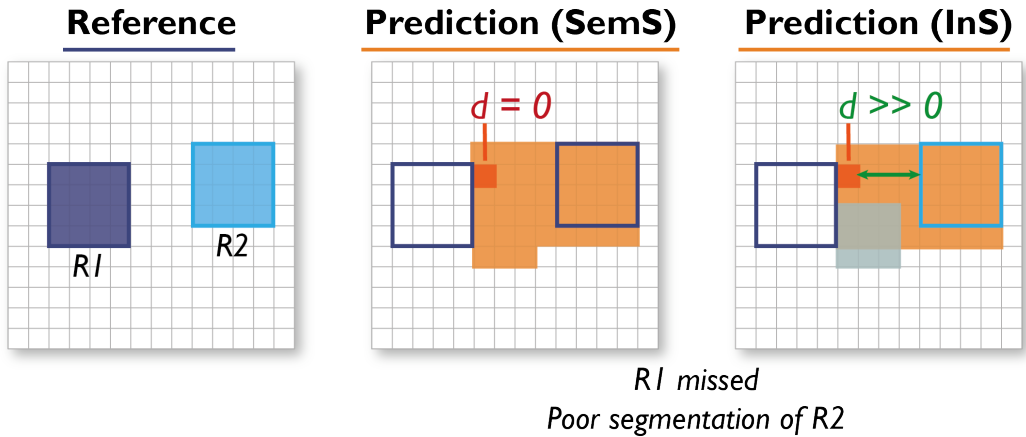


Fig. SN 1.2. **Boundary-based metrics in semantic/instance segmentation problems.** If multiple structures of the same type can be seen within the same image (here: reference objects $R1$ and $R2$), it is generally advisable to phrase the problem as instance segmentation (InS; right) rather than semantic segmentation (SemS; left). This way, issues with boundary-based metrics resulting from comparing a given structure boundary to the boundary of the wrong instance in the reference can be avoided. In the provided example, the distance of the red boundary pixel to the reference, as measured by a boundary-based metric in SemS problems, would be zero, because different instances of the same structure cannot be distinguished. This problem is overcome by phrasing the problem as InS. In this case, (only) the boundary of the matched instance (here: $R2$) is considered for distance computation.

1.3 Generation of the problem fingerprint

Metrics Reloaded is based on the novel concept of *problem fingerprinting* – the generation of a structured representation of the given problem that captures all aspects that are relevant for metric selection. Specifically, the fingerprint comprises a set of items, each of which represents a specific property of the problem, is either binary or categorical, and must be instantiated by the user. In the following, we will refer to all fingerprint items with the notation $FPX.Y$, where Y is a numerical identifier and the index X represents one of the following families: general, domain interest-related, target structure-related, data set-related, algorithm output-related.

Fingerprint generation begins with the aforementioned mapping of the underlying problem with its intrinsic and data set-related properties to the corresponding problem category via the *category mapping* (Subprocess S1) shown in Extended Data Fig. 1. Next, the user needs to instantiate the category-specific fingerprint items provided in Figs. SN 1.7-SN 1.9 (image-level classification), Figs. SN 1.10/SN 1.11 (semantic segmentation), Figs. SN 1.12-SN 1.14 (object detection), and Figs. SN 1.15-SN 1.17 (instance segmentation).

Instantiating fingerprint items may not always be straightforward due to their binary/categorical nature. Therefore, the *Metrics Reloaded* tool comprises a "Why are we asking this question?" button in each branch based on a fingerprint that may not be straightforward to instantiate. In case of ongoing doubt, the user may traverse all appropriate branches originating from the questions.

Importantly, some fingerprint items require particularly careful consideration and/or are not sufficiently self-explanatory. These are the following:

FP2.6: Decision rule applied to predicted class scores. Modern algorithms output (continuous) predicted class scores. To classify cases in an actual biomedical application (i.e., to make actual decisions), however, applying a decision rule to the scores is required; this amounts to setting a cutoff value in the binary classification case. The deciding factor for whether or not to apply a decision rule during validation should be how much focus is to be put on the quality of the actual decisions of a classification system versus the general quality of its continuous predictions. While some communities have converged to decision rule-based validation (e.g., cell instance segmentation [20]), recent clinical initiatives advocate for decision rule-agnostic validation, arguing that decision rules are often over-optimized on a specific data set, associated results are not transferable across study cohorts (e.g., with differing disease prevalence) and clinical applications (e.g., with differing cost-benefit trade-offs for patients), and continuous "risk scores" might be beneficial for communicating results with patients [10, 73, 111]. One study goes so far as to call out the common practice of imposing decision rules on continuous predictions as 'dichotomania' [117]. We handle this controversy in current practices by making validation with specific decision rules applied optional (for all tasks except semantic segmentation) and encoding user preferences in this fingerprint. The fingerprint offers the following decision rule strategies (Fig. SN 1.3):

Target-value based (for binary image-level classification problems) Sometimes, the underlying problem provides a specific target metric value to be reached (e.g., Sensitivity of 0.95), requiring a corresponding cutoff value. In this case, we use the notation $Metric@(TargetMetric = TargetValue)$, for example, $Specificity@(Sensitivity = 0.95)$, denoting the Specificity for a Sensitivity matching the target value (here: 0.95). Importantly, this cutoff needs to be configured on a separate and dedicated data split.

Optimization-based If no specific target value is provided, a data-based decision rule can also be identified by optimizing a primary metric (e.g., F_1 Score) using a dedicated data

set for decision rule configuration. Notably, simple (one-dimensional) cutoff scans are only possible in binary tasks, while identifying decision rules in multiple classes represents a computationally and technically complex process.

Argmax-based An alternative widely used strategy is to simply apply a decision rule based on the 'argmax' operation, which boils down to a cutoff of 0.5 in binary classification problems. The underlying hypothesis for this strategy is that the highest class score resembles the highest probability for the associated class being the true class. In Bayesian theory, this decision rule defines a Bayes classifier, and the theory further shows that the underlying hypothesis only holds for equal severity of class confusions (FP2.5.2=False) and when the class scores are calibrated. Detailed considerations for this decision rule strategy are provided in Sec. 1.1.

Cost-benefit-based If the predicted class scores are calibrated (see FP2.7), and task-related error costs or a risk cutoff (the latter only for binary classification tasks, e.g., "only treat patients with cancer risk >10%") are provided, one can apply this decision rule directly to the scores without data-driven optimization. Notably, in binary classification tasks, cost-benefit-based cutoffs often correspond to a cost ratio of True Positive (TP) versus False Positive (FP) (e.g., not more than 10FP per 1 TP should be treated), while for cost-based cutoffs the explicit costs for both errors FP and False Negative (FN) are defined (see DG3.2, Suppl. Note 2.7.2). Cost-based decision rules are further extendable to multi-class problems [40].

No decision rule applied A complementary strategy is to abstain from validating algorithms under a certain decision rule and exclusively report results on multi-threshold metrics (averaging over various cutoffs) instead.

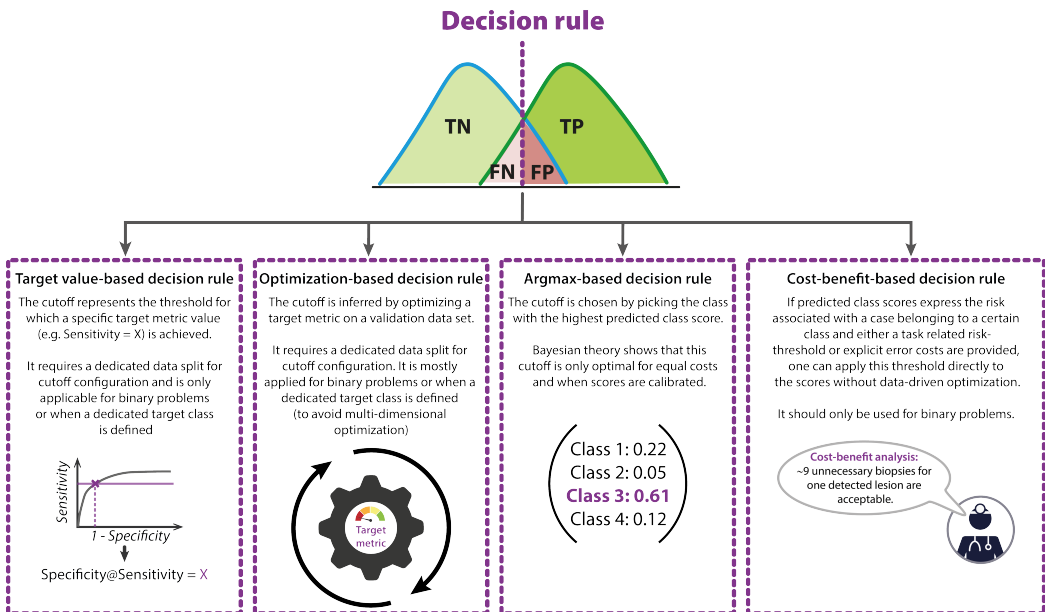


Fig. SN 1.3. Illustration of strategies for identifying a decision rule applied to predicted class scores.

FP2.4 Desired granularity of localization. Selecting a localization criterion operating on a lower/coarser resolution with regard to provided reference annotations effectively discards spatial information and should be well motivated by the given task (see Fig. SN 1.4). For instance, Box Intersection over Union (IoU) is sometimes employed despite access to pixel-mask annotations (FP4.4) because associated models (object detectors) are considered simpler approaches compared to instance segmentation models. Such simplification may cause problems if structures are not well-approximated by a box shape – especially for 3D shapes, boxes usually constitute poor approximations – or if structures can overlap (FP3.5), causing multi-component masks (see Fig. SN 1.5).

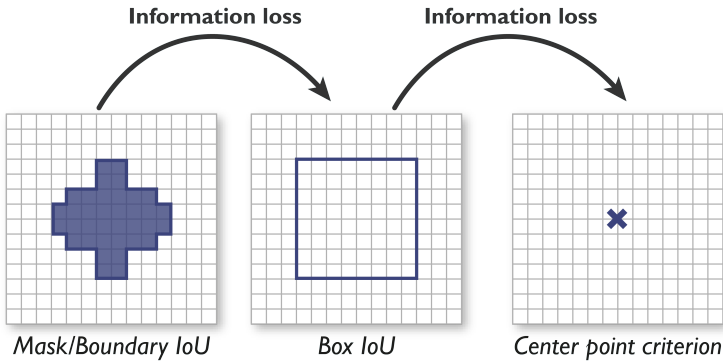


Fig. SN 1.4. Selection of a localization criterion that discards spatial information should be well motivated by the given task.

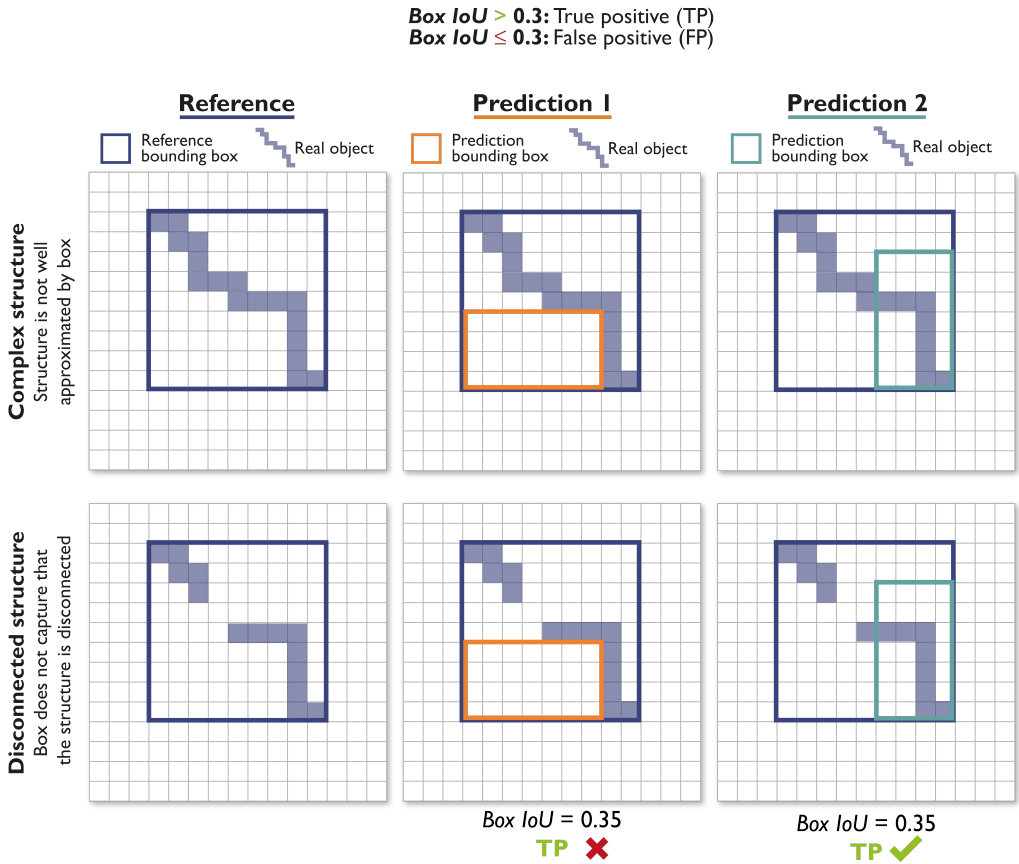


Fig. SN 1.5. Bounding boxes are not well-suited for representing complex (top) and disconnected (bottom) shapes. Specifically, they are not well-suited for capturing multi-component structures. *Predictions 1* and *2* both end up in a True Positive (TP) detection, as the Box Intersection over Union (IoU) is larger than the cutoff 0.3. However, *Prediction 1* does not hit the real objects at all.

FP2.5.5: Compensation for class imbalances. While Accuracy is the de facto standard metric in multi-class settings with balanced class frequencies and error costs, this metric is prone to several pitfalls when class imbalances are present. To give an example, consider the following confusion matrix for a binary classification task: $TP = 0$, $FP = 1$, $FN = 1$, $TN = 10,000$, which leads to an accuracy of ≈ 1 . Three pitfalls pertain to this metric score, which at the same time represent the three reasons why users may want to compensate for the underlying effects caused by the class imbalance:

Misleading metric values due to missing reference value for naive classifier: In the provided example, the near-perfect score hides the fact that the same performance could have been achieved by a naive system always predicting the dominant class. Generally, in balanced scenarios, the Accuracy of a naive classifier is known to be “1/number of classes”, which serves as an important anchor when interpreting the metric scores. However, when class imbalances are present, no such interpretation can be made and the naive reference depends on the class prevalences.

Misleading metric values due to unequal contribution of classes to the metric score: In the provided example, the near-perfect score hides the fact that all samples of the positive class (here: one sample) were misclassified. While all classes contribute similarly to the Accuracy metric in balanced scenarios, frequent classes dominate the performance value in imbalanced settings. While 0% (0/1) of the rare cases have been classified correctly, the metric achieves an almost perfect score due to the very good performance on the dominant class. Other prevalence-independent metrics, such as Balanced Accuracy (BA), are based on the equal contribution of each class irrespective of prevalence.

Misleading metric values due to missing consideration of predictive values: In the provided example, the near-perfect score hides the fact that the positive predictive value of this system is 0. Generally, in balanced scenarios, high accuracy scores imply high predictive values (Positive Predictive Value (PPV) and Negative Predictive Value (NPV)), which are important indicators of the utility of a classification system in practice. This is not necessarily the case in imbalanced scenarios, as seen in the provided example, where the PPV is 0 despite a high Accuracy. To compensate for this effect, alternative metrics such as Matthews Correlation Coefficient (MCC) can be considered, which explicitly assess the predictive performance of a classifier.

FP4.2 Class prevalences reflect the population of interest. Class prevalences and their differences across data sets are highly important, although this aspect is often ignored in common validation practice. This can best be explained with the example of diagnostic tests, for example image-based disease classification. While several metrics, such as Sensitivity and Specificity, are independent of class frequencies and measure the inherent properties of the test, other metrics, such as Accuracy, measure the test performance for the specific prevalence of the test set. This is not problematic if the class prevalences of the provided test set reflect the population of interest, but can lead to problems otherwise (see Fig. SN 1.6). This fingerprint should hence be set to true if either the validation interest is constrained to the data set at hand (no future comparison to data sets with different class prevalences is desired) or no variation of prevalences is expected in other cohorts and upon application of the method.

Inherent properties of a method: Sensitivity = 0.90, Specificity = 0.80

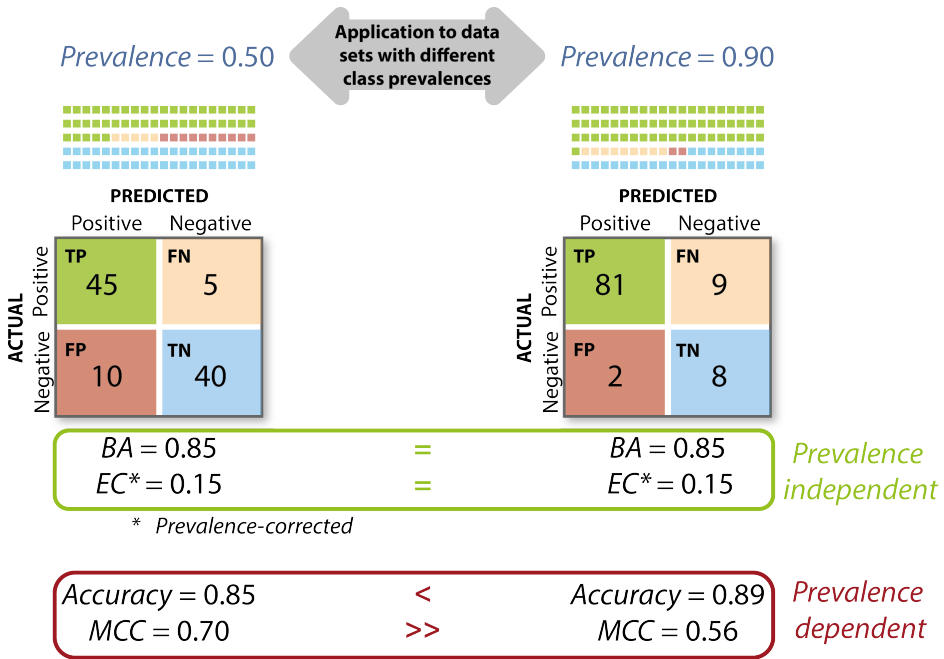


Fig. SN 1.6. Effect of prevalence dependency. An algorithm with specific inherent properties (here: Sensitivity of 0.9 and Specificity of 0.8) may perform completely differently on different data sets if the prevalences differ (here: 50% (left) and 90% (right)) and prevalence-dependent metrics are used for validation (here: Accuracy and Matthews Correlation Coefficient (MCC)). In contrast, prevalence-independent metrics (here: Balanced Accuracy (BA) and the prevalence-corrected Expected Cost (EC)) can be used to compare validation results across different data sets. Used abbreviations: True Positive (TP), False Negative (FN), False Positive (FP) and True Negative (TN).

IMAGE-LEVEL CLASSIFICATION (ImLC) PART 1		
Fingerprint ID and name	Fingerprint illustration	Fingerprint description
1.1 Image processing category identified by category mapping	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto;"> Class 1 Class 2 </div>	Image-level classification (ImLC): assignment of one or multiple category labels to the entire image. <i>Example: disease screening; deciding on the presence or absence of a certain condition/pathology without localizing the phenomenon.</i>
Domain interest-related properties (part 1)		
2.5 Penalization of errors	There may be a preference for certain types of errors from a domain perspective.	
2.5.1 Unequal interest across classes		<p>There is a preference for one or several of the classes. This has implications for both the metric selection and the metric aggregation. It is important to note that this fingerprint only considers "a priori interest" in classes that is irrespective of the class prevalences in the data. This distinction is necessary, because one can also think of the importance of a class in terms of how much it contributes to the final metric score. This latter concept, however, is based on the class prevalence at hand and thus considered via compensation for class imbalances (FP2.5.5) in our framework.</p> <p>Note that class interest in this context can be considered as costs for all cells of a confusion matrix related to one class as a whole. In contrast, "class confusions" (FP2.5.2) considers individual cells in the confusion matrix.</p> <p><i>Example 1: five-way classification on a heavily imbalanced dataset. One class dominates the other classes in terms of frequency, but the interest lies in the overall error rate of the system, implying the dominating class should contribute more to the final metric score.</i></p> <p><i>Example 2: In cell classification scenarios, it may be more important to correctly classify tumor cells compared to correctly classifying muscle cells or connective tissue.</i></p> <p><i>Example 3: in full surgical scene segmentation for autonomous robotics, critical structures, such as nerves or vessels, should be localized more accurately compared to fatty tissue.</i></p>
2.5.2 Unequal severity of class confusions		<p>Any class can be confused with another, but certain mismatches are more severe than others, from a domain point of view. This holds especially true (1) in screening tasks, in which FNs are typically more severe than FP, (2) in retrieval tasks, in which FP are typically more severe than FN and (3) in tasks with ordinal rating. Note that class confusions in this context can be considered as costs for individual cells in the confusion matrix, while "interest across classes" (FP2.5.1) would consider all matrix cells related to one class as a whole.</p> <p>It is important to note that this fingerprint only considers "a priori costs" of a task that is irrespective of the class prevalences in the data. This distinction is necessary, because one can also tweak the confusion costs in hindsight to compensate for certain imbalances in the data (not considered here).</p> <p><i>Example 1 (multi-class): Depending on the application, confusing different kinds of immune cells is more problematic compared to confusing an immune cell with a tumor epithelial cell.</i></p> <p><i>Example 2 (multi-class): lung tumor categorization T1-T5 depends largely on structure size, implying an ordinal scale of classes. Thus, penalization of class confusions should reflect this ordinal scale.</i></p>
2.5.3 Mismatch between class prevalences and class importance		<p>The class prevalences do not reflect the class importance. There are three scenarios in which this property should be set to TRUE.</p> <ol style="list-style-type: none"> 1. Class prevalences are balanced (FP4.1 = FALSE), but there is an unequal interest across classes (FP2.5.1 = TRUE). 2. Class imbalance is present (FP4.1 = TRUE), but there is an equal interest across classes (FP2.5.1 = FALSE). 3. Class imbalance is present (FP4.1 = TRUE) and there is an unequal interest across classes (FP2.5.1 = TRUE), but the way in which classes are imbalanced does not match the "imbalance of interest". <p>Importantly, while scenarios 1 and 2 can be expressed with other fingerprints, scenario 3 represents a new set of use cases.</p>
2.5.4 Costs for class confusions available		<p>In the case of an unequal severity of class confusions (FP2.5.2 = TRUE), these unequal severities might be explicitly defined in the form of cost values associated with each confusion. For example, a cost analysis may lead to the result that FN errors are 5 times more costly than FP errors. In case such costs are defined or can be estimated with adequate accuracy for the use case, it is possible to apply certain metrics which explicitly consider these costs in validation (e.g., WCK and EC). If costs are not provided and cannot be estimated, we recommend to proceed with validation separately for individual classes.</p>
2.5.5 Compensation for class imbalances requested		<p>Severe class imbalances might lower interpretability and impede objective assessment of method validation and for example, lead to overly optimistic conclusions. Specifically, we distinguish three pitfalls:</p> <ol style="list-style-type: none"> 1. Missing reference value for random performance 2. Neglect of equal importance of classes 3. Missing consideration of predictive values <p>The choice of counting metric(s) depends crucially on which of these pitfalls should be avoided.</p>

Fig. SN 1.7. **Fingerprint for image-level classification (Part 1)**. In the case of binary fingerprint items, the blue column shows examples for which the property is true while the red column shows counterexamples. Categorical fingerprint items are only shown in blue. Suppl. Note 1.3 provides more detailed explanations of selected fingerprint items. Used abbreviations: True Positive (TP), False Positive (FP), False Negative (FN), True Negative (TN).

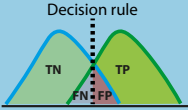



IMAGE-LEVEL CLASSIFICATION (ImLC) PART 2		
Fingerprint ID and name	Fingerprint illustration	Fingerprint description
Domain interest-related properties (part 2)		
<p>2.6 Decision rule applied to predicted class scores</p> <p>Options:</p> <ul style="list-style-type: none"> - Target value-based - Optimization-based - Argmax-based - Cost-benefit-based - No decision rule applied 		<p>Modern algorithms output continuous class scores. Making a classification decision requires identifying a decision rule applied to the scores, which amounts to setting a cutoff value in binary tasks. A product of this process is the (decision rule-specific) confusion matrix. This matrix enables the computation of popular single-threshold counting metrics, such as Sensitivity, PPV and F₁ Score. Depending on domain interest the decision rule can be set in multiple ways:</p> <p>Target value-based (only for binary tasks): The cutoff represents the threshold for which a specific target metric value (e.g., Sensitivity = 0.95) is achieved. Importantly, this threshold has to be determined on a separate data split. Other metric values (e.g., Specificity) are then reported for this specific threshold. We use the notation Metric@(TargetMetric = TargetValue) (e.g., Specificity@Sensitivity = 0.95) in this case. This cutoff strategy is limited to binary classification problems.</p> <p>Optimization-based: The decision rule is inferred by optimizing a target metric, such as the F₁ Score in the binary case, on a separate data split.</p> <p>Argmax-based: Especially in multi-class scenarios and if no target value is defined, no separate data split for optimization is available, or there are concerns w.r.t generalization of data-based decision rule optimization, a common option is to follow the principle of a Bayes classifier and pick the class with the highest predicted class score.</p> <p>Cost-benefit-based: In case the predicted class scores are calibrated (see FP2.7), and there is either a task related risk-cutoff (only for binary classification tasks, e.g. "only treat patients with cancer risk >10%"), or explicit costs for misclassification errors provided, one can apply this decision rule directly to the scores without data-driven optimization. Notably, in binary classification tasks provided risk cutoffs are often based on a cost ratio of TP versus FP (e.g., not more than 10 FPs per 1 TP should be treated). In most cases, no specific risk cutoff can be determined, thus model performance is plotted over a reasonable range of risk scores ("decision curve analysis").</p> <p>No decision rule applied: Examples for no interest in validating a method at a specific decision rule are 1) focus on general methodological performance across many tasks and data sets without application interest, 2) concerns regarding the comparability of results based on a single decision rule that is fixed across varying study cohorts (see also FP4.2), or 3) focus on the probabilistic predictions to obtain and communicate personalized risk factors of indi-</p>
<p>2.7 Calibration of predicted class scores</p>	<p>When validating classification methods - particularly those with applications that involve direct human read-out - it is often crucial for the predicted class scores themselves to be interpretable. A system is well-calibrated if the predicted class scores (i.e., the output of the model) reflect the true probabilities of the outcome (formal definition see App. C.6).</p>	
<p>2.7.1 Calibration assessment requested</p>		<p>This property should be set to TRUE if the predicted class scores should reflect the true probabilities of the outcome. An obvious requirement for this assessment is that predicted class scores are available (FP5.1 = TRUE). Methods subject to validation in this context are either classification models whose inherent calibration quality is assessed, or re-calibration methods, which are typically accuracy-preserving (bijective) transformations on the classifier outputs aiming to improve calibration quality.</p>
<p>2.7.2 Comparative calibration assessment requested</p> <p>Options:</p> <ul style="list-style-type: none"> - Comparison of re-calibration methods on the same classifier - Comparison of calibration performance across classifiers - Comparison of overall performance across classifiers - No comparative assessment 		<p>Comparison of re-calibration methods on the same classifier: The potential benefit of one or more re-calibration methods is to be assessed and compared. The desired validation output is a ranking of re-calibration methods (including the performance of "no re-calibration") from which the best method can be selected.</p> <p>Comparison of calibration performance across classifiers: This comparison of classification models potentially includes re-calibration methods applied on their outputs. The desired validation output is a ranking of methods according to calibration quality.</p> <p>Comparison of overall performance across classifiers: Overall performance refers to the joint assessment of discrimination performance and calibration quality. This comparison of classification models potentially includes re-calibration methods applied on their outputs. The desired validation output is a single ranking naturally weighting both aspects.</p> <p>No comparative assessment: If the interest lies in understanding the reliability of predicted class scores for one given model, no metrics for comparative assessment are required.</p>
<p>2.7.3 Assessment of interpretability of model outputs requested</p> <p>Options:</p> <ul style="list-style-type: none"> - Assessment of calibration error in isolation - Joint assessment of calibration and discrimination - No assessment of interpretability required 		<p>There is an interest in understanding the reliability of predicted class scores for a given model as a basis for interpreting and communicating results. The desired validation output is a single score that captures the interpretability of the output. We differentiate two notions of interpretability that correspond to different families of calibration-sensitive metrics.</p> <p>Assessment of calibration error in isolation: The user may be interested in assessing the calibration quality of a model "in isolation" (i.e., without associated discrimination power). Quantifying this property is often a desirable aspect of validating a model. Knowing that predicted class scores are well-calibrated allows making specific statements about individual output scores such as "80% of outputs with score 0.8 belong to the true class". Importantly, calibration metrics do not assess whether outputs match the true posterior probabilities. Hence, the "risk statements" depend on the model at hand (and its discrimination power) and thus cannot associate a true model-independent risk to a data sample (e.g., patient).</p> <p>Joint assessment of calibration and discrimination: An alternative approach to assessing the interpretability of outputs is to compare the predicted class scores directly to the reference and thus to quantify the overall performance of a model (discrimination and calibration) in one joint score. This assessment can also be interpreted as measuring whether scores match the true posteriors, e.g. the risks of individual patients. A disadvantage of this strategy is that the calibration error is conflated with discrimination performance, thus prohibiting statements about the reliability of particular scores such as "80% of outputs with score 0.8 belong to the true class".</p>

Fig. SN 1.8. **Fingerprint for image-level classification (Part 2).** In the case of binary fingerprint items, the blue column shows examples for which the property is true while the red column shows counterexamples. Categorical fingerprint items are only shown in blue. Suppl. Note 1.3 provides more detailed explanations of selected fingerprint items. Used abbreviations: True Positive (TP), False Positive (FP), False Negative (FN), True Negative (TN).

IMAGE-LEVEL CLASSIFICATION (ImLC) PART 3		
Fingerprint ID and name	Fingerprint illustration	Fingerprint description
Data set-related properties		
4.1 High class imbalance		The class prevalences differ substantially. <i>Example: In a screening application, the positive class (e.g., cancer) may occur extremely rarely. In this case, prevalence-dependent metrics, such as Accuracy, may be extremely misleading.</i>
4.2 Provided class prevalences reflect the population of interest		The class prevalences of the provided data set are representative of the prevalences to be expected in the population of interest. This property should be set to TRUE if either the validation interest is constrained to the data set at hand or no variation of prevalences is expected in other cohorts and upon application of the method. The property should be set to FALSE if variation of prevalences is expected to occur beyond the current data set and, at the same time, comparability across study cohorts or estimation of method performance upon future application are requested. In this case, only prevalence-independent metrics will be recommended.
4.5 Non-independence of test cases		The test cases are hierarchically structured, indicating non-independence of test cases. <i>Examples: multiple images of the same patient, hospital or video.</i>
Algorithm output-related properties		
5.1 Availability of predicted class scores		Modern algorithms in biomedical image classification output continuous class scores, which are often interpreted as predicted class probabilities. These scores contain relevant information about the performance of a model and are thus crucial for comprehensive and meaningful validation. If no predicted class probabilities are available, this property is set to false.
5.3 Possibility of invalid algorithm output (e.g., Prediction is NaN)		The files representing the algorithm output can contain invalid output. Note that an invalid prediction differs from an empty prediction.

Fig. SN 1.9. **Fingerprint for image-level classification (Part 3).** In the case of binary fingerprint items, the blue column shows examples for which the property is true while the red column shows counterexamples. Categorical fingerprint items are only shown in blue. Suppl. Note 1.3 provides more detailed explanations of selected fingerprint items. Used abbreviations: Reference (Ref), Prediction (Pred), True Positive (TP), False Positive (FP), False Negative (FN), True Negative (TN).

SEMANTIC SEGMENTATION (SemS) PART 1		
Fingerprint ID and name	Fingerprint illustration	Fingerprint description
1.1 Image processing category identified by category mapping		Semantic segmentation (SemS): assignment of one or multiple category labels to each pixel. <i>Example: surgical scene segmentation for autonomous robotics; assigning each pixel the corresponding structure/organ/pathology label.</i>
Domain interest-related properties		
2.1 Particular importance of structure boundaries		The biomedical application requires exact structure boundaries. <i>Example: segmentation for radiotherapy planning; knowledge of exact structure boundaries is crucial to destroy the tumor while sparing healthy tissue.</i> Important: Overlap-based metrics do not measure shape agreement. In the case of complex shapes (high boundary-to-volume ratio) it is therefore typically advisable to set this property to TRUE.
2.2 Particular importance of structure volume		The biomedical application requires accurate knowledge of structure volumes. <i>Example: liver segmentation as basis for remnant liver volume computation in surgical resection planning.</i>
2.3 Particular importance of structure center (e.g., in cells, vessels)		The biomedical application requires accurate knowledge of structure centers. <i>Example: cell centers are subsequently used for cell tracking and cell motion characterization, so false center movement should be suppressed.</i>
2.5 Penalization of errors	There may be a preference for certain types of errors from a domain perspective.	
2.5.1 Unequal interest across classes		There is a preference for one or several of the classes. This has implications for both the metric selection and the metric aggregation. It is important to note that this fingerprint only considers "a priori interest" in classes that is irrespective of the class prevalences in the data . This distinction is necessary, because one can also think of the importance of a class in terms of how much it contributes to the final metric score. This latter concept, however, is based on the class prevalence at hand and thus considered via compensation for class imbalances (FP2.5.5) in our framework. Note that class interest in this context can be considered as costs for all cells of a confusion matrix related to one class as a whole. In contrast, "class confusions" (FP2.5.2) considers individual cells in the confusion matrix. <i>Example: in full surgical scene segmentation for autonomous robotics, critical structures, such as nerves or vessels, should be localized more accurately compared to fatty tissue.</i>
2.5.2 Unequal severity of class confusions		Any class can be confused with another, but certain mismatches are more severe than others, from a domain point of view. This holds especially true (1) in screening tasks, in which FN are typically more severe than FP, (2) in retrieval tasks, in which FP are typically more severe than FN and (3) in tasks with ordinal rating. Note that class confusions in this context can be considered as costs for individual cells in the confusion matrix, while "interest across classes" (FP2.5.1) would consider all matrix cells related to one class as a whole. It is important to note that this fingerprint only considers "a priori costs" of a task that is irrespective of the class prevalences in the data . This distinction is necessary, because one can also tweak the confusion costs in hindsight to compensate for certain imbalances in the data. <i>Example (multi-class): lung tumor categorization T1-T5 depends largely on structure size, implying an ordinal scale of classes. Thus, penalization of class confusions should reflect this ordinal scale.</i>
2.5.6 Handling of spatial outliers		Spatial outliers are FP predictions that feature a large distance to the reference. They can be handled in three different ways: Distance-based penalization with outlier focus: Individual outliers should be heavily penalized as a function of the distance to the reference contour. Distance-based penalization with whole contour focus: Outliers should be penalized as a function of the distance to the reference, but the assessment should focus on the general contour agreement rather than individual outliers. Existence-based penalization: The existence of spatial outliers should be penalized irrespective of their distance to the reference contour. Note that distance-based penalization is not possible when either the reference or the prediction is empty. In applications in which many of such cases potentially occur, we therefore recommend an existence-based penalization.
2.5.7 Compensation for annotation imprecisions requested		The reference annotation is typically only an approximation of the (forever unknown) ground truth. It may be desirable to compensate for known uncertainties, such as intra-rater or inter-rater variability, by configuring the metric accordingly. This is only possible for some metrics.

Fig. SN 1.10. **Fingerprint for semantic segmentation (Part 1)**. In the case of binary fingerprint items, the blue column shows examples for which the property is true while the red column shows counterexamples. Categorical fingerprint items are only shown in blue. Suppl. Note 1.3 provides more detailed explanations of selected fingerprint items. Used abbreviations: Reference (Ref), Prediction (Pred), True Positive (TP), False Positive (FP), False Negative (FN), True Negative (TN).

SEMANTIC SEGMENTATION (SemS) PART 2		
Fingerprint ID and name	Fingerprint illustration	Fingerprint description
Target structure-related properties		
3.1 Small size of structures relative to pixel size		Structures of the provided class are only a few pixels in size. <i>Example: multiple sclerosis lesions in magnetic resonance imaging (MRI) scans.</i>
3.2 High variability of structure sizes (within an image and/or across images)		The target structures vary substantially in size, such that some structures are several times the size of others. <i>Example: polyps in colonoscopy screening, where some polyps are several times the size of others.</i> <i>Counterexample: large organs, such as the liver or the kidneys, which are relatively comparable in size across individuals.</i>
3.3 Target structures feature tubular shape		The target structures feature a tubular shape. <i>Examples: vessels, neurons, microtubules.</i>
3.4 Possibility of multiple labels per unit (pixel or image)		One pixel may be assigned to multiple reference categories. <i>Example: one pixel assigned to two labels 'tumor core' and 'tumor'.</i>
3.5 Possibility of overlapping or touching target structures (e.g., medical instruments or cells)		Different instances of a class can overlap or touch each other. <i>Examples: overlapping cells or organisms, such as BBBC010 (worms in a dish); overlapping medical instruments in laparoscopy.</i>
Data set-related properties		
4.1 Presence of class imbalance		The class prevalences differ substantially. <i>Example: In a screening application, the positive class (e.g., cancer) may occur extremely rarely. In this case, prevalence-dependent metrics, such as Accuracy, may be extremely misleading.</i>
4.3 Uncertainties in the reference	The reference is typically only an approximation of the (forever unknown) ground truth. Various sources and types of errors exist, which require special treatment in the context of metric selection.	
4.3.1 High inter-/intra-rater variability		The reference can be assumed to be noisy due to high inter-rater variability.
4.3.2 Possibility of spatial outliers in reference annotation		The reference may feature spatial outliers that are distant from the (unknown) ground truth.
4.5 Non-independence of test cases		The test cases are hierarchically structured, indicating non-independence of test cases. <i>Examples: multiple images of the same patient, hospital or video.</i>
4.6 Possibility of reference without target structure(s)		There are test cases in which the reference comprises only the background class.
Algorithm output-related properties		
5.2 Possibility of algorithm output not containing the target structure(s)		The algorithm may yield outputs in which not all classes are present.
5.3 Possibility of invalid algorithm output (e.g., Prediction is NaN)		The files representing the algorithm output can contain invalid output. Note that an invalid prediction differs from an empty prediction.

Fig. SN 1.11. **Fingerprint for semantic segmentation (Part 2).** In the case of binary fingerprint items, the blue column shows examples for which the property is true while the red column shows counterexamples. Categorical fingerprint items are only shown in blue. Suppl. Note 1.3 provides more detailed explanations of selected fingerprint items.

OBJECT DETECTION (ObD) PART 1		
Fingerprint ID and name	Fingerprint illustration	Fingerprint description
1.1 Image processing category identified by category mapping		Object detection (ObD): detection and localization of structures of one or multiple categories. <i>Example: detection and bounding box-based localization of polyps in colonoscopy sequences.</i>
Domain interest-related properties (part 1)		
2.3 Particular importance of structure center (e.g., in cells, vessels)		The biomedical application requires accurate knowledge of structure centers. <i>Example: cell centers are subsequently used for cell tracking and cell motion characterization, so false center movement should be suppressed.</i>
2.4 Desired granularity of localization Options: - Only position - Rough outline - Exact outline		The granularity of required localization can vary in object detection tasks. We distinguish two main categories: Only position: given an n-dimensional image, the object is represented by its position, encoded in n degrees of freedom (e.g. xy/xyz coordinates of center point). Rough outline: a rough outline of the object is provided, typically given by simple geometric approximations such as bounding boxes or ellipsoids. <i>It should be noted that if a substantial fraction of objects are tiny (FP3.1), any outline-based localization becomes very noisy. In such cases, users might want to consider alternative localization strategies, such as a center point-based localization.</i>
2.5 Penalization of errors	There may be a preference for certain types of errors from a domain perspective.	
2.5.1 Unequal interest across classes		There is a preference for one or several of the classes. This has implications for both the metric selection and the metric aggregation. It is important to note that this fingerprint only considers "a priori interest" in classes that is irrespective of the class prevalences in the data . This distinction is necessary, because one can also think of the importance of a class in terms of how much it contributes to the final metric score. This latter concept, however, is based on the class prevalence at hand and thus considered via compensation for class imbalances (FP2.5.5) in our framework. Note that class interest in this context can be considered as costs for all cells of a confusion matrix related to one class as a whole. In contrast, "class confusions" (FP2.5.2) considers individual cells in the confusion matrix. <i>Example 1: five-way classification on a heavily imbalanced dataset. One class dominates the other classes in terms of frequency, but the interest lies in the overall error rate of the system, implying the dominating class should contribute more to the final metric score.</i> <i>Example 2: In cell classification scenarios, it may be more important to correctly classify tumor cells compared to correctly classifying muscle cells or connective tissue.</i> <i>Example 3: in full surgical scene segmentation for autonomous robotics, critical structures, such as nerves or vessels, should be localized more accurately compared to fatty tissue.</i>
2.5.2 Unequal severity of class confusions		Any class can be confused with another, but certain mismatches are more severe than others, from a domain point of view. This holds especially true (1) in screening tasks, in which FN are typically more severe than FP, (2) in retrieval tasks, in which FP are typically more severe than FN and (3) in tasks with ordinal rating. Note that class confusions in this context can be considered as costs for individual cells in the confusion matrix, while "interest across classes" (FP2.5.1) would consider all matrix cells related to one class as a whole. It is important to note that this fingerprint only considers "a priori costs" of a task that is irrespective of the class prevalences in the data . This distinction is necessary, because one can also tweak the confusion costs in hindsight to compensate for certain imbalances in the data (not considered here). <i>Example 1 (binary): polyp detection; a FN (missed polyp) is clinically much more severe than a FP.</i> <i>Example 2 (multi-class): Depending on the application, confusing different kinds of immune cells is more problematic compared to confusing an immune cell with a tumor epithelial cell.</i> <i>Example 3 (multi-class): lung tumor categorization T1-T5 depends largely on structure size, implying an ordinal scale of classes. Thus, penalization of class confusions should reflect this ordinal scale.</i>
2.5.3 Mismatch between class prevalences and class importance		The class prevalences do not reflect the class importance. For metric selection it is crucial to understand whether the class prevalences match the target population (covered by FP4.2) and whether they match the class importance. There are three scenarios in which this property should be set to TRUE: 1. Class prevalences are balanced (FP4.1 = FALSE), but there is an unequal interest across classes (FP2.5.1 = TRUE). 2. Class imbalance is present (FP4.1 = TRUE), but there is an equal interest across classes (FP2.5.1 = FALSE). 3. Class imbalance is present (FP4.1 = TRUE) and there is an unequal interest across classes (FP2.5.1 = TRUE), but the way in which classes are imbalanced does not match the "imbalance of interest". Importantly, while scenarios 1 and 2 can be expressed with other fingerprints, scenario 3 represents a new set of use cases.
2.5.8 Penalization of multiple predictions assigned to the same reference object requested		Object detection algorithms involve the step of assigning predicted objects to reference objects. This may result in more than one prediction being assigned to the same reference. This fingerprint property should be set to TRUE if all but one prediction of such an assignment (i.e., "double assignments") should be penalized as FP, and set to false if these spare predictions should be ignored during validation. Note that in the inverse case of one prediction assigned to multiple reference objects, the convention is to not ignore the spare (non-matched) reference(s), but penalize them as FN (or via application-specific penalization such as "merge errors" or subsequent segmentation metrics).

Fig. SN 1.12. **Fingerprint for object detection (Part 1)**. In the case of binary fingerprint items, the blue column shows examples for which the property is true while the red column shows counterexamples. Categorical fingerprint items are only shown in blue. Suppl. Note 1.3 provides more detailed explanations of selected fingerprint items. Used abbreviations: True Positive (TP), False Positive (FP), False Negative (FN), True Negative (TN).

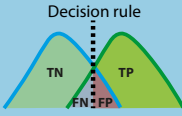
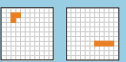

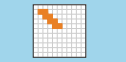
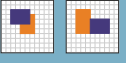
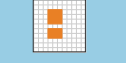
OBJECT DETECTION (ObD) PART 2		
Fingerprint ID and name	Fingerprint illustration	Fingerprint description
Domain interest-related properties (part 2)		
<p>2.6 Decision rule applied to predicted class scores</p> <p>Options:</p> <ul style="list-style-type: none"> - Target value-based - Optimization-based - Argmax-based - No decision rule applied 		<p>Modern algorithms output continuous class scores. Making a classification decision requires identifying a decision rule applied to the scores, which amounts to setting a cutoff value in binary tasks. A product of this process is the (decision rule-specific) confusion matrix. This matrix enables the computation of popular single-threshold counting metrics, such as Sensitivity, PPV and F_1 Score. Depending on domain interest the decision rule can be set in multiple ways:</p> <p>Target value-based (only for binary tasks): The cutoff represents the threshold for which a specific target metric value (e.g., Sensitivity = 0.95) is achieved. Importantly, this threshold has to be determined on a separate data split. Other metric values (e.g., Specificity) are then reported for this specific threshold. We use the notation Metric@(TargetMetric = TargetValue) (e.g., Specificity@Sensitivity = 0.95) in this case. This cutoff strategy is limited to binary classification problems.</p> <p>Optimization-based: The decision rule is inferred by optimizing a target metric, such as the F_1 Score in the binary case, on a separate data split.</p> <p>Argmax-based: Especially in multi-class scenarios and if no target value is defined, no separate data split for optimization is available, or there are concerns w.r.t generalization of data-based decision rule optimization, a common option is to follow the principle of a Bayes classifier and pick the class with the highest predicted class score.</p> <p>No decision rule applied: Examples for no interest in validating a method with a specific decision rule are 1) focus on general methodological performance across many tasks and data sets without application interest, 2) concerns regarding the comparability of results based on a single decision rule that is fixed across varying study cohorts, or 3) focus on the probabilistic predictions to obtain and communicate personalized risk factors of individual patients.</p>
Target structure-related properties		
<p>3.1 Small size of structures relative to pixel size</p>		<p>Structures of the provided class are only a few pixels in size. <i>Example: multiple sclerosis lesions in magnetic resonance imaging (MRI) scans.</i></p>
<p>3.2 High variability of structure sizes (within an image and/or across images)</p>		<p>The target structures vary substantially in size, such that some structures are several times the size of others. <i>Example: polyps in colonoscopy screening, where some polyps are several times the size of others.</i> <i>Counterexample: large organs, such as the liver or the kidneys, which are relatively comparable in size across individuals.</i></p>
<p>3.3 Target structures feature tubular shape</p>		<p>The target structures feature a tubular shape. <i>Examples: vessels, neurons, microtubules.</i></p>
<p>3.5 Possibility of overlapping or touching target structures (e.g., medical instruments or cells)</p>		<p>Different instances of a class can overlap or touch each other. <i>Examples: overlapping cells or organisms, such as BBBC010 (worms in a dish); overlapping medical instruments in laparoscopy.</i></p>
<p>3.6 Possibility of disconnected target structure(s)</p>		<p>A given structure appears disconnected in the given image. <i>Examples: neurons in 2D microscopy of a slice of tissue; single tomographic image slice depicting complex vessels.</i></p>

Fig. SN 1.13. **Fingerprint for object detection (Part 2).** In the case of binary fingerprint items, the blue column shows examples for which the property is true while the red column shows counterexamples. Categorical fingerprint items are only shown in blue. Suppl. Note 1.3 provides more detailed explanations of selected fingerprint items. Used abbreviations: True Positive (TP), False Positive (FP), False Negative (FN), True Negative (TN).

OBJECT DETECTION (OBD) PART 3		
Fingerprint ID and name	Fingerprint illustration	Fingerprint description
Data set-related properties		
4.1 Presence of class imbalance		The class prevalences differ substantially. <i>Example: In a screening application, the positive class (e.g., cancer) may occur extremely rarely. In this case, prevalence-dependent metrics, such as Accuracy, may be extremely misleading.</i>
4.3 Uncertainties in the reference	The reference is typically only an approximation of the (forever unknown) ground truth. Various sources and types of errors exist, which require special treatment in the context of metric selection.	
4.3.1 High inter-/intra-rater variability		The reference can be assumed to be noisy due to high inter-rater variability.
4.4 Granularity of provided reference annotations Options: - Only position - Rough outline - Exact outline		The granularity of the reference can vary in object detection problems. We distinguish three main categories: Only position: Given an n-dimensional image, the object is represented by its position, encoded in n degrees of freedom (e.g. xy/xyz coordinates of center point) Rough outline: A rough outline of the object is provided, typically given by simple geometric objects such as bounding boxes or ellipsoids. Exact outline: The object is outlined exactly.
4.5 Non-independence of test cases		The test cases are hierarchically structured, indicating non-independence of test cases. <i>Examples: multiple images of the same patient, hospital or video.</i>
4.6 Possibility of reference without target structure(s)		There are test cases in which the reference for at least one class is empty.
Algorithm output-related properties		
5.1 Availability of predicted class scores		Modern algorithms in biomedical image classification output continuous class scores, which are often interpreted as predicted class probabilities. These scores contain relevant information about the performance of a model and are thus crucial for comprehensive and meaningful validation. In object detection, predicted class probabilities are typically available for each detected object. If no predicted class probabilities are available, this property is set to false.
5.2 Possibility of algorithm output not containing the target structure(s)		The algorithm may yield outputs in which not all classes are present.
5.3 Possibility of invalid algorithm output (e.g., Prediction is NaN)		The files representing the algorithm output can contain invalid output. Note that an invalid prediction differs from an empty prediction.
5.4 Possibility of overlapping predictions		Predictions of the algorithm can potentially overlap.

Fig. SN 1.14. **Fingerprint for object detection (Part 3).** In the case of binary fingerprint items, the blue column shows examples for which the property is true while the red column shows counterexamples. Categorical fingerprint items are only shown in blue. Suppl. Note 1.3 provides more detailed explanations of selected fingerprint items. Used abbreviations: Reference (Ref), Prediction (Pred).

INSTANCE SEGMENTATION (InS) PART 1		
Fingerprint ID and name	Fingerprint illustration	Fingerprint description
1.1 Image processing category identified by category mapping		Instance segmentation (InS): detection and delineation of each distinct object of a particular class. It can be regarded as delivering the tasks of object detection and semantic segmentation at the same time. In contrast to object detection, instance segmentation also involves the accurate marking of the object boundary. In contrast to semantic segmentation, it distinguishes different instances of the same class. <i>Example: cell segmentation with a subsequent goal of measuring cell properties.</i>
Domain interest-related properties (part 1)		
2.1 Particular importance of structure boundaries		The biomedical application requires exact structure boundaries. <i>Example: segmentation for radiotherapy planning; knowledge of exact structure boundaries is crucial to destroy the tumor while sparing healthy tissue.</i> Important: Overlap-based metrics do not measure shape agreement. In the case of complex shapes (high boundary-to-volume ratio) it is therefore typically advisable to set this property to TRUE.
2.2 Particular importance of structure volume		The biomedical application requires accurate knowledge of structure volumes. <i>Example: liver segmentation as basis for remnant liver volume computation in surgical resection planning.</i>
2.3 Particular importance of structure center (e.g., in cells, vessels)		The biomedical application requires accurate knowledge of structure centers. <i>Example: cell centers are subsequently used for cell tracking and cell motion characterization, so false center movement should be suppressed.</i>
2.5 Penalization of errors	There may be a preference for certain types of errors from a domain perspective.	
2.5.1 Unequal interest across classes		There is a preference for one or several of the classes. This has implications for both the metric selection and the metric aggregation. It is important to note that this fingerprint only considers "a priori interest" in classes that is irrespective of the class prevalences in the data . This distinction is necessary, because one can also think of the importance of a class in terms of how much it contributes to the final metric score. This latter concept, however, is based on the class prevalence at hand and thus considered via compensation for class imbalances (FP2.5.5) in our framework. Note that class interest in this context can be considered as costs for all cells of a confusion matrix related to one class as a whole. In contrast, "class confusions" (FP2.5.2) considers individual cells in the confusion matrix. <i>Example: in full surgical scene segmentation for autonomous robotics; critical structures, such as nerves or vessels, should be localized more accurately compared to fatty tissue.</i>
2.5.2 Unequal severity of class confusions a) for detection b) for segmentation (per instance)		Any class can be confused with another, but certain mismatches are more severe than others, from a domain point of view. This holds especially true (1) in screening tasks, in which FN are typically more severe than FP, (2) in retrieval tasks, in which FP are typically more severe than FN and (3) in tasks with ordinal rating. Note that class confusions in this context can be considered as costs for individual cells in the confusion matrix, while "interest across classes" (FP2.5.1) would consider all matrix cells related to one class as a whole. It is important to note that this fingerprint only considers "a priori costs" of a task that is irrespective of the class prevalences in the data . This distinction is necessary, because one can also tweak the confusion costs in hindsight to compensate for certain imbalances in the data (not considered here). <i>Example 1 (multi-class): Depending on the application, confusing different kinds of immune cells is more problematic compared to confusing an immune cell with a tumor epithelial cell.</i> <i>Example 2 (multi-class): lung tumor categorization T1-T5 depends largely on structure size, implying an ordinal scale of classes. Thus, penalization of class confusions should reflect this ordinal scale.</i> Specifically in instance segmentation problems, the property needs to be set separately for the validation of the (a) detection (relevant decision guide: 3.5) and (b) segmentation performance (relevant subprocess: S6). At object level, FNs (missed instances) are sometimes more severe than FPs, while FNs (e.g. undersegmentation) and FPs (e.g. oversegmentation) may be equally important at pixel level.
2.5.3 Mismatch between class prevalences and class importance		The class prevalences do not reflect the class importance. For metric selection it is crucial to understand whether the class prevalences match the target population (covered by FP4.2) and whether they match the class importance. There are three scenarios in which this property should be set to TRUE: 1. Class prevalences are balanced (FP4.1 = FALSE), but there is an unequal interest across classes (FP2.5.1 = TRUE). 2. Class imbalance is present (FP4.1 = TRUE), but there is an equal interest across classes (FP2.5.1 = FALSE). 3. Class imbalance is present (FP4.1 = TRUE) and there is an unequal interest across classes (FP2.5.1 = TRUE), but the way in which classes are imbalanced does not match the "imbalance of interest". Importantly, while scenarios 1 and 2 can be expressed with other fingerprints, scenario 3 represents a new set of use cases.

Fig. SN 1.15. **Fingerprint for instance segmentation (Part 1)**. In the case of binary fingerprint items, the blue column shows examples for which the property is true while the red column shows counterexamples. Categorical fingerprint items are only shown in blue. Suppl. Note 1.3 provides more detailed explanations of selected fingerprint items. Used abbreviations: True Positive (TP), False Positive (FP), False Negative (FN), True Negative (TN).

INSTANCE SEGMENTATION (InS) PART 2		
Fingerprint ID and name	Fingerprint illustration	Fingerprint description
Domain interest-related properties (part 2)		
2.5.6 Handling of spatial outliers		<p>Spatial outliers are FP predictions that feature a large distance to the reference. They can be handled in three different ways:</p> <p>Distance-based penalization with outlier focus: Individual outliers should be heavily penalized as a function of the distance to the reference contour.</p> <p>Distance-based penalization with whole contour focus: Outliers should be penalized as a function of the distance to the reference, but the assessment should focus on the general contour agreement rather than individual outliers.</p> <p>Existence-based penalization: The existence of spatial outliers should be penalized irrespective of their distance to the reference contour.</p> <p>Note that distance-based penalization is not possible when either the reference or the prediction is empty. In applications in which many of such cases potentially occur, we therefore recommend an existence-based penalization.</p>
2.5.7 Compensation for annotation imprecisions requested		<p>The reference annotation is typically only an approximation of the (forever) uncertainties, such as intra-rater or inter-rater variability, by configuring the metric accordingly. This is only possible for some metrics.</p>
2.5.8 Penalization of multiple predictions assigned to the same reference object requested		<p>Instance segmentation algorithms often involve the step of assigning predicted objects to reference objects. This may result in more than one prediction being assigned to the same reference. This fingerprint property should be set to TRUE if all but one prediction of such an assignment (i.e., "double assignments") should be penalized as FP, and set to false if these spare predictions should be ignored during validation. Note that in the inverse case of one prediction assigned to multiple reference objects, the convention is to not ignore the spare (non-matched) reference(s), but penalize them as FN (or via application-specific penalization such as "merge errors" or subsequent segmentation metrics).</p>
<p>2.6 Decision rule applied to predicted class scores</p> <p>Options:</p> <ul style="list-style-type: none"> - Target value-based - Optimization-based - Argmax-based - No decision rule applied 		<p>Modern algorithms output continuous class scores. Making a classification decision requires identifying a decision rule applied to the scores, which amounts to setting a cutoff value in binary tasks. A product of this process is the (decision rule-specific) confusion matrix. This matrix enables the computation of popular single-threshold counting metrics, such as Sensitivity, PPV and F_1 Score.</p> <p>Depending on domain interest the decision rule can be set in multiple ways:</p> <p>Target value-based (only for binary tasks): The cutoff represents the threshold for which a specific target metric value (e.g., Sensitivity = 0.95) is achieved. Importantly, this threshold has to be determined on a separate data split. Other metric values (e.g., Specificity) are then reported for this specific threshold. We use the notation Metric@(TargetMetric = TargetValue) (e.g., Specificity@Sensitivity = 0.95) in this case. This cutoff strategy is limited to binary classification problems.</p> <p>Optimization-based: The decision rule is inferred by optimizing a target metric, such as the F_1 Score in the binary case on a separate data split.</p> <p>Argmax-based: Especially in multi-class scenarios and if no target value is defined, no separate data split for optimization is available, or there are concerns w.r.t generalization of data-based decision rule optimization, a common option is to follow the principle of a Bayes classifier and pick the class with the highest predicted class score.</p> <p>No decision rule applied: Examples for no interest in validating a method with a specific decision rule are 1) focus on general methodological performance across many tasks and data sets without application interest, 2) concerns regarding the comparability of results based on a single decision rule that is fixed across varying study cohorts, or 3) focus on the probabilistic predictions to obtain and communicate personalized risk factors of individual patients.</p>

Fig. SN 1.16. **Fingerprint for instance segmentation (Part 2).** In the case of binary fingerprint items, the blue column shows examples for which the property is true while the red column shows counterexamples. Categorical fingerprint items are only shown in blue. Suppl. Note 1.3 provides more detailed explanations of selected fingerprint items. Used abbreviations: Reference (Ref), True Positive (TP), False Positive (FP), False Negative (FN), True Negative (TN).

INSTANCE SEGMENTATION (INS) PART 3		
Fingerprint ID and name	Fingerprint illustration	Fingerprint description
Target structure-related properties		
3.1 Small size of structures relative to pixel size		Structures of the provided class are only a few pixels in size. <i>Example: multiple sclerosis lesions in magnetic resonance imaging (MRI) scans.</i>
3.2 High variability of structure sizes (within an image and/or across images)		The target structures vary substantially in size, such that some structures are several times the size of others. <i>Example: polyps in colonoscopy screening, where some polyps are several times the size of others.</i> <i>Counterexample: large organs, such as the liver or the kidneys, which are relatively comparable in size across individuals.</i>
3.3 Target structures feature tubular shape		The target structures feature a tubular shape. <i>Examples: vessels, neurons, microtubules.</i>
3.4 Possibility of multiple labels per unit (pixel or image)		One pixel may be assigned to multiple reference categories. <i>Example: one pixel assigned to two labels 'tumor core' and 'tumor.'</i>
3.5 Possibility of overlapping or touching target structures (e.g., medical instruments or cells)		Different instances of a class can overlap or touch each other. <i>Examples: overlapping cells or organisms, such as BBBC010 (worms in a dish); overlapping medical instruments in laparoscopy.</i>
3.6 Possibility of disconnected target structure(s)		A given structure appears disconnected in the given image. <i>Examples: neurons in 2D microscopy of a slice of tissue; single tomographic image slice depicting complex vessels.</i>
Data set-related properties		
4.1 Presence of class imbalance		The class prevalences differ substantially. <i>Example: In a screening application, the positive class (e.g., cancer) may occur extremely rarely. In this case, prevalence-dependent metrics, such as Accuracy, may be extremely misleading.</i>
4.3 Uncertainties in the reference	The reference is typically only an approximation of the (forever unknown) ground truth. Various sources and types of errors exist, which require special treatment in the context of metric selection.	
4.3.1 High inter-/intra-rater variability		The reference can be assumed to be noisy due to high inter-rater variability.
4.3.2 Possibility of spatial outliers in reference annotation		The reference may feature spatial outliers that are distant from the (unknown) ground truth.
4.5 Non-independence of test cases		The test cases are hierarchically structured, indicating non-independence of test cases. <i>Examples: multiple images of the same patient, hospital or video.</i>
4.6 Possibility of reference without target structure(s)		There are test cases in which the reference for at least one class is empty.
Algorithm output-related properties		
5.1 Availability of predicted class scores		Modern algorithms in biomedical image classification output continuous class scores, which are often interpreted as predicted class probabilities. These scores contain relevant information about the performance of a model and are thus crucial for comprehensive and meaningful validation. Instance segmentation problems in the biomedical domain are often approached by adding a post-processing step (e.g. connected component analysis) to a semantic segmentation algorithm. In this process, predicted class probabilities often get lost. If no predicted class probabilities are available, this property is set to false.
5.2 Possibility of algorithm output not containing the target structure(s)		The algorithm may yield outputs in which not all classes are present.
5.3 Possibility of invalid algorithm output (e.g., Prediction is NaN)		The files representing the algorithm output can contain invalid output. Note that an invalid prediction differs from an empty prediction.
5.4 Possibility of overlapping predictions		Predictions of the algorithm can potentially overlap.

Fig. SN 1.17. **Fingerprint for instance segmentation (Part 3).** In the case of binary fingerprint items, the blue column shows examples for which the property is true while the red column shows counterexamples. Categorical fingerprint items are only shown in blue. Suppl. Note 1.3 provides more detailed explanations of selected fingerprint items. Used abbreviations: Reference (Ref), Prediction (Pred).

SUPPL. NOTE 2 STEP 2 - METRIC SELECTION

As a foundation for the metric selection process, the *Metrics Reloaded* consortium compiled a set of common reference-based validation metrics (Suppl. Note 2.1). The framework leverages the problem fingerprints to guide the user through the process of selecting an appropriate set of category-specific reference-based validation metrics while being made aware of potential pitfalls related to individual choices. A bird's eye perspective of the process is shown in Fig. 2. A detailed explanation for the selection of reference-based metrics is provided separately for all four problem categories in Suppl. Notes 2.2-2.5. Details on selecting appropriate calibration metrics, if desired, are given in Suppl. Note 2.6. The corresponding formal decision trees (subprocesses) along with corresponding decision guides are shown in Extended Data Figs. 1-9 and Suppl. Note 2.7, respectively.

2.1 *Metrics Reloaded* pool of reference-based metrics

The *Metrics Reloaded* pool of common reference-based validation metrics is shown in Tab. SN 2.1. Most of these metrics are directly or indirectly based on the cardinalities of the *confusion matrix* (i.e., the true (T)/false (F) positives (P)/negatives (N) in binary problems). For the purpose of metric recommendation, we follow the terminology in the sister publication of this work [85] and classify the metrics into **counting metrics** that operate directly on a single fixed confusion matrix and express the metric value as a function of the cardinalities, **multi-threshold metrics** that operate on a dynamic confusion matrix, and **distance-based metrics**, designed to measure differences between boundaries, volumes, center (line)s or shapes [85]. In addition to these, our framework considers metrics designed to measure calibration capabilities of models.

Importantly, many popular counting metrics are closely related. In Fig. SN 2.1, we categorize the relationship as follows.

- (1) **Synonyms**: For some metrics, various terms exist. Popular examples are:
 - Recall = Sensitivity = True Positive Rate (TPR) = Hit rate
 - Positive Predictive Value (PPV) = Precision
 - Dice Similarity Coefficient (DSC) = F_1 Score (at pixel level)
- (2) **Mutually computable from each other**: Some metrics are directly computable from each other without further information. Popular examples are:
 - Accuracy = 1 - Error Rate (ER)
 - DSC = $(2 * \text{Intersection over Union (IoU)}) / (1 + \text{IoU})$
 - Balanced Accuracy = $(\text{Bookmaker Informedness (BM)} + 1) / 2$
- (3) **Generalization/Instantiation**: Some metrics are an instantiation of others. Popular examples are:
 - DSC is an instantiation of the F_β Score for $\beta = 1$
 - Accuracy is a specialization of Expected Cost (EC), where costs are chosen as "0-1-costs", meaning $c_{ii} = 0$ and $c_{ij} = 1$ otherwise.
 - Balanced Accuracy (BA) is a specialization of EC, where costs are chosen such that $c_{ii} = 0$ and $c_{ij} = \frac{1}{CP_i}$ with P_i reflecting the class prevalence of class i and C denoting the number of classes.
- (4) **Mutually computable under certain conditions**: Assuming a simple problem setup, additional metrics coincide. Popular examples are:
 - Assuming perfect class balance in a binary problem, BA = Accuracy = Cohen's Kappa (CK)
 - Assuming $\beta = 1$ allows to compute IoU and Jaccard index from F_β Score

- (5) **Other notable relationship**: Some metrics share another notable relationship. These are detailed in the metric cheat sheets (Suppl. Note 3.1)

Cheat sheets for all metrics, comprising basic information such as definition and links to reference implementations, relationships to other metrics, and *Metrics Reloaded* recommendations for their usage, can be found in Suppl. Note 3.1.

Table SN 2.1. **Overview of recommended reference-based metrics.** For each metric, name, acronym, synonyms, reference to the definition and illustration, range and corresponding problem categories are provided. The direction of the arrow in the 'range' column indicates whether higher (up) or lower scores (down) are better. A detailed introduction and discussion of all metrics can be found in the sister publication of this work [85]. ImLC: image-level classification; SemS: semantic segmentation; ObD: object detection; InS: instance segmentation.

Metric	Acronym	Synonyms	Definition	Range	Recommended for			
					ImLC	SemS	ObD	InS
Counting Metrics								
Accuracy			[42, 99]	[0, 1] ↑	x			
Balanced Accuracy	BA		[42, 99]	[0, 1] ↑	x			
Weighted Cohen's Kappa	WCK	Cohen's Kappa Coefficient, Kappa Statistic, Kappa Score	[27]	[-1, 1] ↑	x			
centerline Dice Similarity Coefficient	cDice		[93]	[0, 1] ↑		x		x
Dice Similarity Coefficient	DSC	Sørensen–Dice Coefficient, F ₁ Score, Balanced F Score	[35]	[0, 1] ↑		x		x
Expected Cost	EC		[15, 40]	(-∞, ∞) ↓	x			
F _β Score			[23]	[0, 1] ↑	x	x	x	x
False Positives per Image*	FPPI		[8, 108]	[0, ∞) ↓			x	x
Intersection over Union	IoU	Jaccard Index, Tanimoto Coefficient	[52]	[0, 1] ↑		x		x
Matthews Correlation Coefficient	MCC	Phi Coefficient	[69]	[-1, 1] ↑	x			
Panoptic Quality	PQ		[56]	[0, 1] ↑				x
Net Benefit	NB		[111]	(-∞, ∞) ↑	x			
Negative Predictive Value*			[14, 99]	[0, 1] ↑	x			
Positive Likelihood Ratio	LR+	Likelihood Ratio Positive, Likelihood Ratio for Positive Results	[6]	[0, ∞) ↑	x			
Positive Predictive Value*	PPV	Precision	[14, 42, 99]	[0, 1] ↑	x		x	x
Sensitivity*		Recall, Hit Rate, True Positive Rate (TPR)	[14, 42, 99]	[0, 1] ↑	x		x	x
Specificity*		Selectivity, True Negative Rate (TNR)	[14, 42, 99]	[0, 1] ↑	x			
Multi-threshold Metrics								
Area under the Receiver Operating Characteristic Curve	AUROC	Area under the curve (AUC), AUC Receiver Operating Characteristic (ROC), C-Index, C-Statistics	[47]	[0, 1] ↑	x			
Average Precision	AP		[64]	[0, 1] ↑	x		x	x
Free-Response Receiver Operating Characteristic Score	FRoC Score		[8, 108]	[0, 1] ↑			x	x
Distance-based Metrics								
Average Symmetric Surface Distance	ASSD		[119]	[0, ∞) ↓		x		x
Boundary Intersection over Union	Boundary IoU		[22]	[0, 1] ↑		x		x
Hausdorff Distance	HD	Hausdorff Metric, Pompeiu–Hausdorff Distance, Maximum Symmetric Surface Distance	[50]	[0, ∞) ↓		x		x
Mean Average Surface Distance	MASD		[11]	[0, ∞) ↓		x		x
Normalized Surface Distance	NSD	Normalized Surface Dice, Surface Distance, Surface Dice	[80]	[0, 1] ↑		x		x
X th Percentile Hausdorff Distance	X th Percentile HD		[50]	[0, ∞) ↓		x		x
Calibration Metrics								
Brier Score	BS		[17]	[0, 1] ↓	x		x	x
Class-wise Calibration Error	CWCE		[59, 60]	[0, 1] ↓	x		x	x
Expected Calibration Error	ECE		[44, 74]	[0, 1] ↓	x		x	x
Expected Calibration Error Kernel Density Estimate	ECE ^{KDE}		[83]	[0, 1] ↓	x		x	x
Kernel Calibration Error	KCE		[43, 115]	[0, 1] ↓	x		x	x
Negative Log Likelihood	NLL	Cross Entropy Loss	[31]	[0, ∞) ↓	x			
Root Brier Score	RBS		[43]	[0, 1] ↓	x		x	x

*: This metric is best used in combination with another metric using a predefined target value (see "Target value-based cutoff" in the definition of FP2.6: *Cutoff on predicted class scores* (Suppl. Note 1.2)).

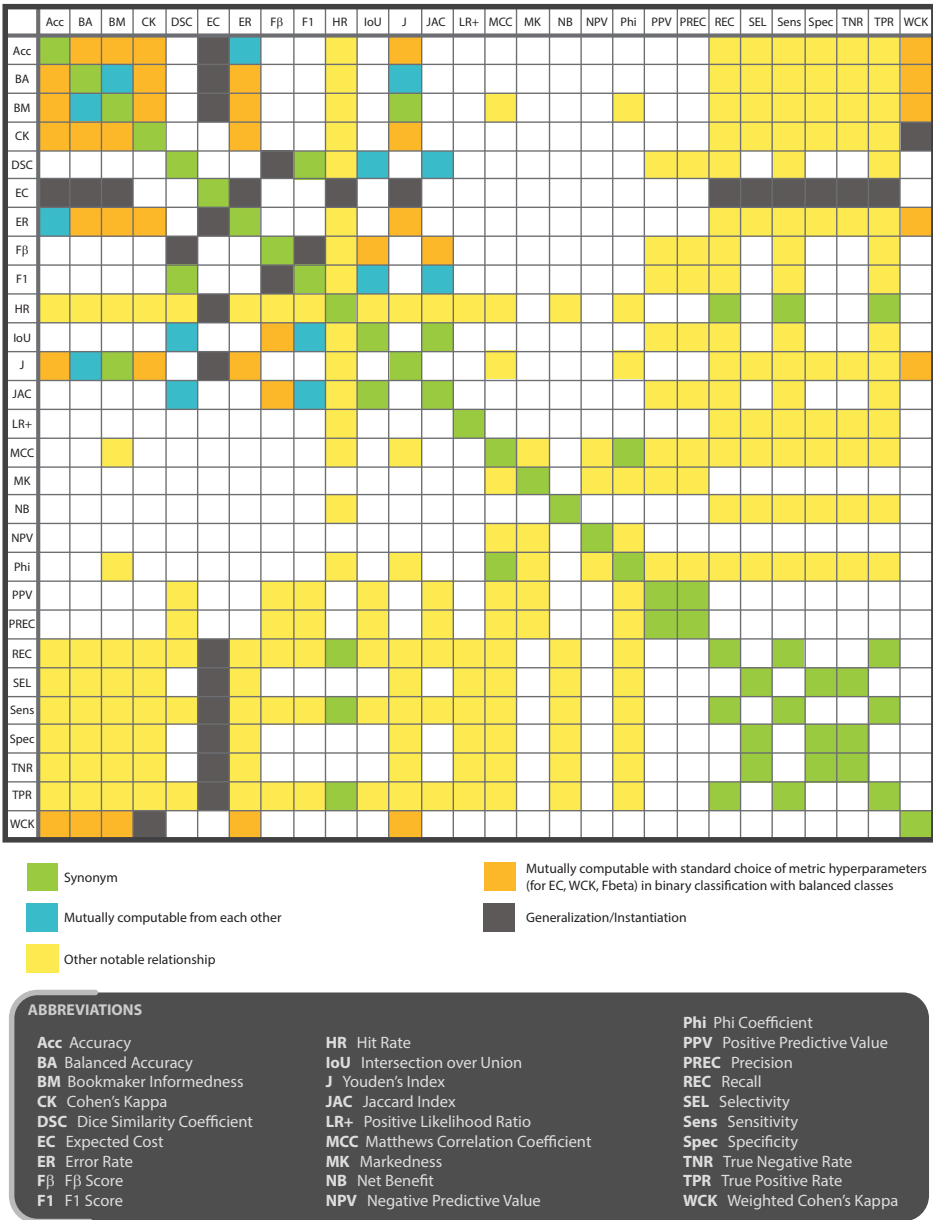


Fig. SN 2.1. Common counting metrics and their relation to each other. The depicted metrics comprise the counting metrics of the *Metrics Reloaded* pool (Tab. SN 2.1) as well as closely related metrics, namely Error Rate (ER), Bookmaker Informedness (BM), Markedness (MK), and Cohen’s Kappa (CK) with synonyms. Panoptic Quality (PQ), centerline Dice Similarity Coefficient (cIDice), and False Positives per Image (FPPI) have been excluded as they rely on more information than solely the confusion matrix.

2.2 Recommendations for Image-level Classification

Essentials

FPX.Y refers to a fingerprint item detailed in Figs. [SN 1.7-SN 1.9](#).

SX refers to a subprocess in Extended Data Figs. 2-5.

DGX.Y refers to a decision guide in Suppl. Notes [2.7.1-2.7.4](#).

This section provides recommendations for selecting *common reference-based metrics* for image-level classification problems. As depicted in Fig. 2, these common metrics can then be complemented by application-specific metrics as well as non-reference-based metrics (assessing run time or carbon footprint, for example).

Image-level classification refers to the process of assigning one or multiple labels (*classes*) to an image. Modern algorithms usually output **predicted class scores** between 0 and 1 for every image and class, which are often interpreted as the probability of the image belonging to a specific class. In binary classification, a threshold can be applied to convert the continuous scores to a classification decision (e.g. cancer = true for values above 0.5). In multi-class classification, the class associated with the highest predicted score is often selected as the final prediction ('argmax' operation). The most common strategies for converting predicted class scores into discrete decisions are captured in the fingerprint *FP2.6 Decision rule applied to predicted class scores* and are detailed in Suppl. Note [1.3](#).

Comparing the algorithm predictions with the reference labels enables the generation of a confusion matrix, which captures the number of correct class assignments on the diagonal for each class and the numbers for all possible class confusions in the remaining cells. In the binary case, these numbers, here referred to as the *cardinalities*, are simply the true/false positives/negatives arranged in a 2×2 matrix. **Counting metrics** operate on this matrix by relating the cardinalities of different matrix entries [85]. They can be classified into **multi-class counting metrics** that operate on the full, potentially multi-class confusion matrix, such as Accuracy, Matthews Correlation Coefficient (MCC) and Expected Cost (EC), and **per-class counting metrics** that validate the performance of a particular class of interest defined as the *positive class* (e.g. with a one-vs-rest comparison for multi-class scenarios), such as the F_β Score. Per-class validation is typically recommended (see below) to obtain an in-depth understanding of the performance of each individual class, as multi-class metrics may potentially hide poor performance of individual classes. All counting metrics differ exclusively in which cardinalities of the confusion matrix they use and how they are combined.

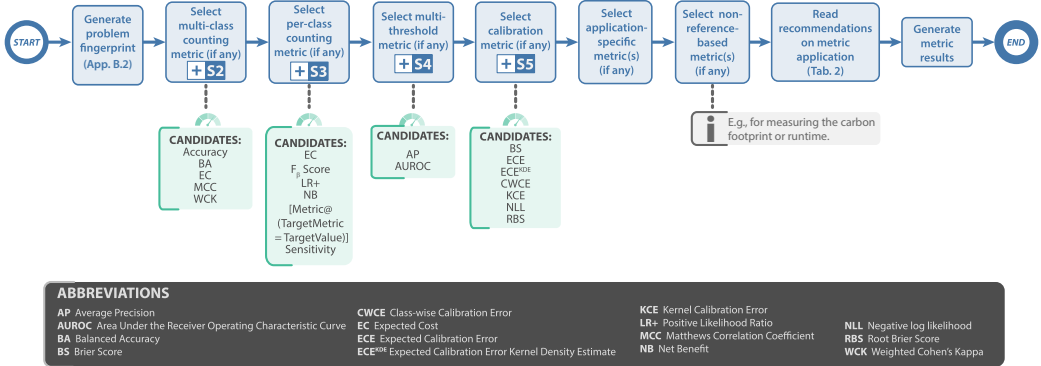


Fig. SN 2.2. *Metrics Reloaded* recommendation framework for image-level classification at a glance.

Counting metrics in general reflect the fact that systems in practice need to define a strategy for converting the predicted class scores (if available) into actual decisions. Choosing a decision rule for the generation of a confusion matrix, however, is not necessarily straightforward, and counting metrics may fail to capture the full capacities of a classifier by restricting performance analysis to a single working point on the decision curve [86] (Fig. SN 1.3). **Multi-threshold metrics** (Fig. SN 2.9) such as Area under the Receiver Operating Characteristic Curve (AUROC) overcome the limitation of a potentially arbitrary threshold by calculating metric scores based on a range of thresholds. They are commonly only defined for binary classification (again, one-versus-rest validation can be performed) and relate basic complementary properties, such as Sensitivity and Specificity in the case of AUROC, or Sensitivity and Positive Predictive Value (PPV) in the case of Average Precision (AP), to each other. The metric value is then obtained by computing the area under the resulting curve [86].

While both counting metrics and multi-threshold metrics measure the *discrimination* capabilities of a classifier, they do not assess whether the predicted class scores reflect the true probability of cases belonging to the predicted class. An orthogonal class of metrics has therefore been designed to assess the interpretability of classifier outputs. As detailed in Suppl. Note 2.6, these **calibration metrics** can roughly be categorized in metrics that assess discrimination and calibration quality together, such as the Brier Score (BS), and those that assess only calibration, such as the Expected Calibration Error (ECE).

Taking into account these considerations as well as the complementary strengths and weaknesses of classification metrics, we recommend the following process for selecting reference-based classification metrics (blue path in Fig. 2 and Fig. SN 2.2):

1: Select multi-class counting metric (if any): Multi-class counting metrics have the unique advantage that they capture the performance of an algorithm for all classes in a single value. With the ability to take into account all entries of the multi-class confusion matrix, they provide a holistic measure of performance without the need for customized class-aggregation schemes. We therefore recommend the selection of a multi-class counting metric based on Subprocess S2 (Extended Data Fig. 2) if a decision rule should be applied to the predicted class scores (FP2.6). In some use cases and especially in the presence of ordinal data, there may be an unequal severity of class confusions (FP2.5.2 = TRUE), implying that different costs to be applied to different errors reflected by the confusion matrix must be available (FP2.5.4 =

TRUE). In this case, the only viable options are Weighted Cohen’s Kappa (WCK) (Fig. SN 3.18) and EC (Fig. SN 3.6). While WCK is widely used, it comes with severe drawbacks (see Suppl. Note 2.7.1 for details), such as high prevalence dependency and ‘paradoxical results’ [113] for the most common variant based on quadratic weights. For this reason, the consortium recommends EC as the default choice for the described scenario. In the case of equal costs, Accuracy (Fig. SN 3.2) is the most widely used multi-class metric, but we recommend it in only one specific scenario: when the class prevalences in the data set reflect those in the target population (FP4.2) and potential class imbalances should not be compensated for. In the more general case, the decision boils down to either picking one of the prevalence-independent metrics EC or Balanced Accuracy (BA) (Fig. SN 3.3), which is specifically recommended if the class prevalences do *not* reflect the target population, or MCC (Fig. SN 3.10), which has the important property that it requires not only the class-specific Sensitivities (i.e. Sensitivity and Specificity in the binary case) but also the corresponding predictive values (PPV and Negative Predictive Value (NPV)) to be high (see Fig. SN 2.3). Irrespective of the metric choice, we recommend additionally reporting the whole confusion matrix in the case of a reasonable number of classes.

- 2: **Select per-class counting metric (if any):** As detailed class-specific analyses are not possible with multi-class counting metrics, which may potentially hide the poor performance of individual classes, we recommend an additional per-class validation with metrics selected according to Subprocess S3 (Extended Data Fig. 3). To this end, class-specific metric pools are generated. The choice of metric depends primarily on the decision rule applied to the predicted class scores (FP2.6; see Suppl. Note 1.3 for a detailed explanation). If a **target value-based** strategy is preferred, the decision rule applied to the predicted class scores is optimized such that a specific target value (e.g. Sensitivity = 0.95; see Fig. SN 3.16) is achieved (see Fig. SN 1.3). Complementary metrics, such as Specificity (Fig. SN 3.17), can then be reported for this fixed value of the target metric (see decision guide 3.1 in Suppl. Note 2.7.2). In this case, the target metric is only reported for the specified target class. If a **cost-benefit-based** strategy is chosen (only recommended for binary classification tasks), we recommend selecting either Net Benefit (NB) (explicit risk-centric view; Fig. SN 3.11) or EC (cost-centric view; Fig. SN 3.6) (see decision guide 3.2 in Suppl. Note 2.7.2). In contrast, in the case of **optimization-based** or **argmax-based** decision rules, the metric choice should be made between Sensitivity, Positive Likelihood Ratio (LR+) (Fig. SN 3.14), and $F\text{-}\beta$ Score (Fig. SN 3.7) (see decision guide 3.3 and 3.4 in Suppl. Note 2.7.2).
- 3: **Select multi-threshold metric:** To obtain a more comprehensive picture of the discrimination performance of a classifier, we always recommend the selection of a multi-threshold metric according to Subprocess S4 (Extended Data Fig. 4), irrespective of the decision rule. Multi-threshold metrics are again applied per class. A particular strength of AUROC (Fig. SN 3.19) is the fact that it is well-interpretable, as the value simply reflects the probability of a sample from the positive class being assigned a higher predicted class score compared to a sample from the negative class. Furthermore, it is prevalence-independent and therefore well-suited for comparison of performance across different data sets. AP (Fig. SN 3.20), on the other hand, is a prevalence-dependent metric, which comes with the advantage that predictive values are considered. This may be a crucial property in class-imbalanced scenarios where the focus is to be put on the rare class while AUROC scores are dominated by the frequent class and may lead to overly optimistic interpretation.
- 4: **Select calibration metric (if any):** If the calibration of a method should be assessed in addition to its discrimination capabilities (FP2.7.1), one or multiple calibration metrics should

be chosen based on Subprocess S5 (Extended Data Fig. 5). Details on this process are provided in Suppl. Note 2.6.

		PREDICTED	
		Positive	Negative
ACTUAL	Positive	TP 99	FN 1
	Negative	FP 100	TN 99900

Sensitivity = 0.99

ECN = 1.00

Specificity = 0.99

EC_{nb} = 0.01

PPV = 0.50

BA = 0.99

NPV = 0.99

F₁ Score = 0.66

MCC = 0.70

Fig. SN 2.3. Failure of prevalence-independent metrics in a screening scenario with high class imbalance. Intuitively, the system is substantially better than random guessing because almost all positive cases have been retrieved out of a large database. At the same time, it is not perfect because only about half of the retrieved cases are correctly classified. However, common popular metrics either indicate near-perfect performance (Sensitivity and Specificity close to 1) or random performance (normalized EC (ECN) about 1). Only the F_1 Score and Matthews Correlation Coefficient (MCC) reflect the intuitive scoring of “quite good, but not perfect” because they incorporate predictive values.

2.3 Recommendations for Semantic segmentation

Essentials

FPX.Y refers to a fingerprint item detailed in Figs. [SN 1.10-SN 1.11](#).

SX refers to a subprocess in Extended Data Figs. 6-7.

DGX.Y refers to a decision guide in Apps. [2.7.5-2.7.6](#).

This section provides recommendations for selecting *common reference-based metrics* for semantic segmentation problems. As depicted in Fig. 2, these common metrics can then be complemented by application-specific metrics as well as non-reference-based metrics (assessing run time or carbon footprint, for example).

Semantic segmentation is commonly defined as the process of partitioning an image into multiple segments/regions. To this end, one or multiple labels are assigned to each pixel such that pixels with the same label share certain characteristics. Semantic segmentation can therefore also be regarded as pixel-level classification. As in image-classification problems, predicted class probabilities are typically calculated for each pixel deciding on the class affiliation based on a threshold over the class scores [5]. In semantic segmentation problems, the pixel-level classification is typically followed by a post-processing step, in which boundary pixels are identified.

The most common semantic segmentation metrics (e.g. Dice Similarity Coefficient (DSC) and Intersection over Union (IoU)) are per-class counting metrics – here referred to as **overlap-based metrics**, which measure the overlap between the reference annotation and the prediction of the algorithm. They can be considered the *de facto* standard for assessing segmentation quality and are well-interpretable.

A key weakness of overlap-based metrics is their shape and contour unawareness [85]. A second class of metrics, the **distance-based metrics**, therefore explicitly assess certain spatial characteristics such as the quality of structure centers or boundaries. Note that in scenarios in which multiple structures of the same type are present within the same image (e.g., in multiple sclerosis (MS) lesion segmentation), a potential pitfall is related to comparing a given structure boundary to the boundary of the wrong instance in the reference (Fig. [SN 1.2](#)). Similar issues arise in the case of completely missed instances. In such scenarios, we explicitly recommend reconsideration to phrase the problem as an instance segmentation problem. If semantic segmentation remains the chosen category, we advise against the use of distance-based metrics, as these are not designed for cases where confusion of boundaries across different instances can occur.

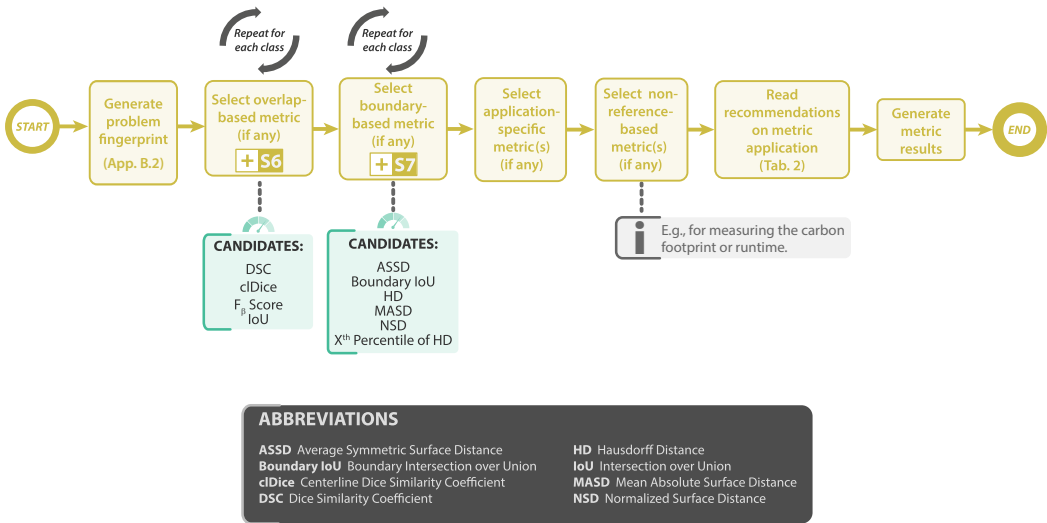


Fig. SN 2.4. *Metrics Reloaded* recommendation framework for semantic segmentation at a glance.

Based on the complementary strengths and weaknesses of common segmentation metrics [85], we recommend the following process for segmentation problems (orange path in Fig. 2 and Fig. SN 2.4):

- 1: Select overlap-based metric (if any):** We recommend selecting an overlap-based metric by default unless the target structures are consistently small (FP3.1) *and* the reference may be noisy (FP4.3.1). As detailed in Subprocess S6 for selecting overlap-based metrics (Extended Data Fig. 6), our default recommendation is the DSC (Fig. SN 3.5), which is almost identical to the IoU (Fig. SN 3.9). The F_β Score (Fig. SN 3.7), which can be seen as a generalization of the $DSC = F_1$ Score, is an alternative if there is a preference for either False Positive (FP) or False Negative (FN). In the specific case of tubular structures (FP3.3), the centerline Dice Similarity Coefficient (cDice) (Fig. SN 3.4) as an increasingly used variant of the DSC can also be applied (optionally in addition to the DSC).
- 2: Select boundary-based metric (if any):** To compensate for the weakness of overlap-based metrics, specifically their shape unawareness and limitations when dealing with small structures or high size variability [85], our general recommendation is to complement an overlap-based metric with a boundary-based metric according to Subprocess S7 (Extended Data Fig. 7). If annotation imprecisions should be compensated for, our default recommendation is the Normalized Surface Distance (NSD) (Fig. SN 3.26). Otherwise, the fundamental user preference guiding metric selection is whether errors should be penalized by existence or distance (FP2.5.6). In the case of existence-based penalization, Boundary IoU (Fig. SN 3.23) should be preferred over NSD if even slight contour errors can be seen as crucial inconsistencies that should be assessed. In the case of distance-based penalization, Mean Average Surface Distance (MASD) (Fig. SN 3.25) is our default recommendation, as it has mathematical advantages over Average Symmetric Surface Distance (ASSD) (Fig. SN 3.22; see decision guide 7.2 in Suppl. Note 2.7.6) and is not as sensitive to annotation outliers as Hausdorff Distance (HD) and its variants (Fig. SN 3.24).

While overlap- and distance-based metrics are the standard metrics used by the general computer vision community, biomedical applications sometimes have special domain-specific requirements.

To accommodate specific domain needs, the standard metrics can therefore be complemented by further 'application-specific' metrics as shown in Fig. 2. In medical imaging, for example, the actual volume of an object, for example a tumor, may be of particular interest (FP2.2). In this case, **volume metrics** such as the *Absolute* or *Relative Volume Error* and the *Symmetric Relative Volume Difference* can be computed [76]. Also, the cDice can be complemented by application-specific connectivity metrics, for instance in the case of tubular structures [32, 78]. Similarly, the explicit agreement of object centers (e.g., in cells) or shapes may be of interest. Note that the latter can often be addressed by choosing a boundary metric with a high tolerance. In other cases, shape agreement may be measured by comparing specific object properties, such as curvatures or principal components. Finally, compliance with prior knowledge, such as hierarchical label structure (FP3.4), can be measured with additional application-specific metrics.

Once a set of metrics has been selected, an appropriate **aggregation strategy** should be chosen. We recommend handling of 'Not a Number's (NaNs) by setting the corresponding metric value to the worst possible value (see Fig. SN 2.5). In the case of distance-based metrics such as the HD, the image diagonal can be chosen, for example (see Fig. SN 2.6). In a benchmarking setting, an alternative lies in using a "rank-then-aggregate" strategy [116]. A test case with a NaN value can then be assigned the worst rank for the given image.

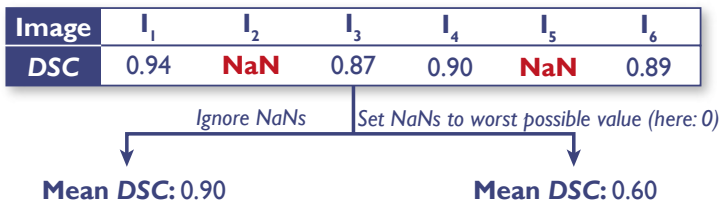


Fig. SN 2.5. Effect of missing values when aggregating metric values. In this example, ignoring missing values leads to a substantially higher Dice Similarity Coefficient (DSC) compared to setting missing values to the worst possible value (here: 0).

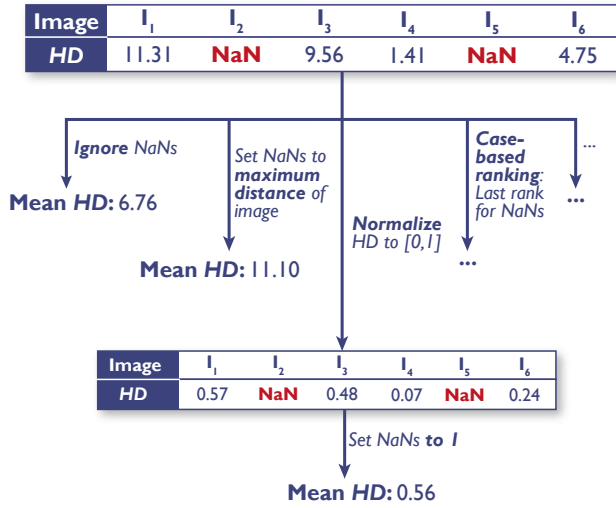


Fig. SN 2.6. Effect of missing values when aggregating metric values for metrics without fixed boundaries (here: Hausdorff Distance (HD)). In this example, ignoring or treating missing values in different ways leads to substantially different HD values.

In multi-class settings, the metric values for the individual classes can be combined in a single score. This can be done via *macro averaging* over class-specific scores, indicating equal importance for each class (FP2.5.1 = FALSE) and an interest to compensate for potential class imbalance (FP2.5.5 = TRUE). Alternatively, weighted averaging, which takes the unequal interest across classes and/or different class prevalences into account, may be performed (Fig. SN 2.7).

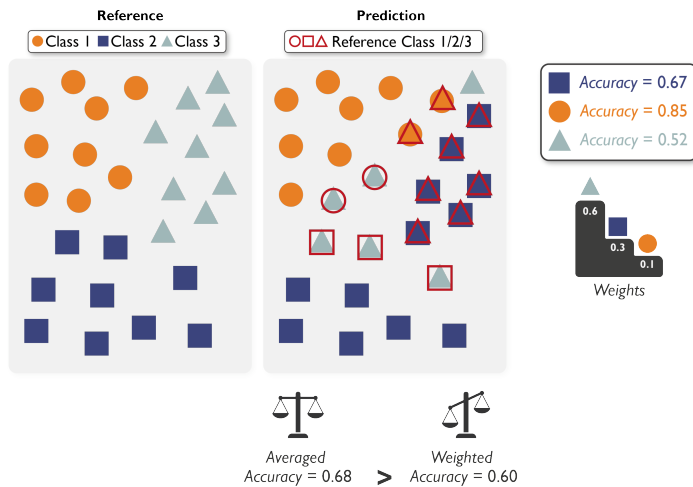


Fig. SN 2.7. Effect of unequal handling of classes. Simple averaging (macro-averaging) of the Accuracy ignores the unequal importance of classes, given by pre-defined weights of classes. Incorrect predictions are indicated by a red square.

2.4 Recommendations for Object detection

Essentials

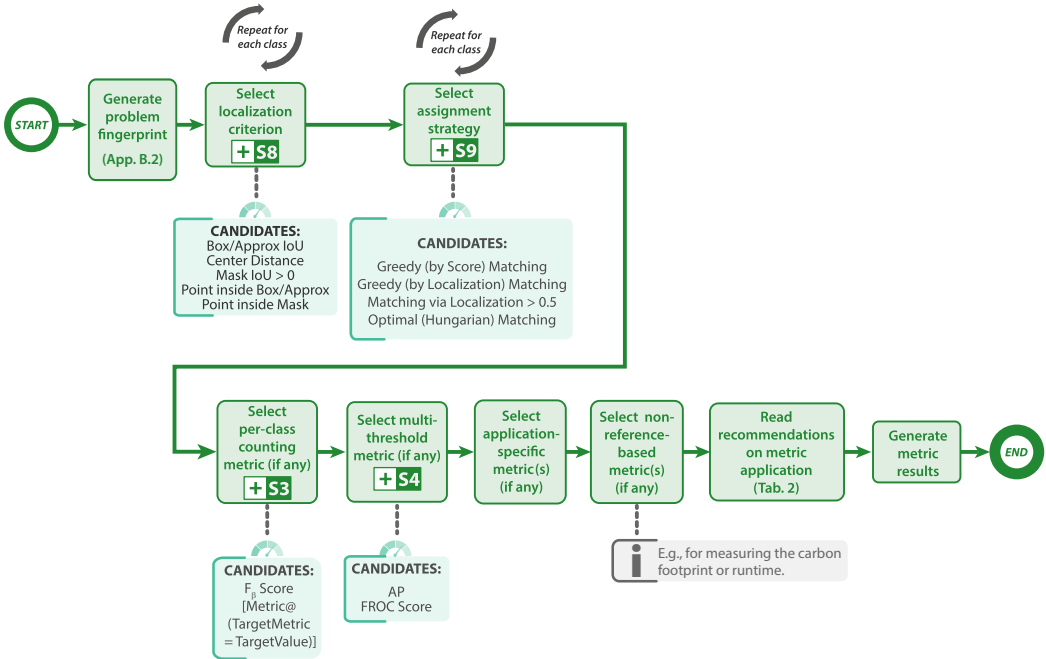
FPX.Y refers to a fingerprint item detailed in Figs. [SN 1.12-SN 1.14](#).

SX refers to a subprocess in Extended Data Figs. 3-4 and Extended Data Figs. 8-9.

DGX.Y refers to a decision guide in Suppl. Note [2.7.2 - 2.7.4, 2.7.7-2.7.8](#).

This section provides recommendations for selecting *common reference-based metrics* for object detection problems. As depicted in Fig. 2, these common metrics can then be complemented by application-specific metrics as well as non-reference-based metrics (assessing run time or carbon footprint, for example).

Object detection refers to the detection and localization of structures of one or multiple categories. A key feature of object detection algorithms is their ability to distinguish different instances of the same class, which may be of crucial domain interest (see Fig. 1). The confusion matrix is generated by comparing reference objects to predicted objects. Based on their matching (see below), True Positives (TPs) (prediction matched to reference object), False Positives (FPs) (prediction without assigned reference object) and False Negatives (FNs) (references without assigned predictions) can be computed. While the design choices in image-level classification are primarily related to the selection of discrimination and calibration metrics (see Suppl. Note [2.2](#)), additional design decisions must be made in object detection due to the object-centric validation. Specifically, a **localization criterion** must be chosen to determine whether a predicted object spatially corresponds to one of the reference objects and vice versa. To this end, an appropriate representation of objects must be chosen. Typical choices are bounding boxes or other approximating shapes. As the object localization step might lead to ambiguous matchings, such as two predictions being assigned to the same reference object, an **assignment strategy** needs to be picked as well. Overall, predictions in object detection have to fulfill the following requirements to be labeled TP: Firstly, the localization criterion must be fulfilled (spatial correspondence). Secondly, the predicted class must match the class of the reference object (always given in the binary case). Finally, the assignment strategy must yield a matching reference object for the given prediction. The recommended localization criteria are provided in Suppl. Note [3.1.3](#).



ABBREVIATIONS

AP	Average Precision	FROC Score	Free-Response Receiver Operating Characteristic Score	Mask IoU	Mask Intersection over Union
Box/Approx	Box/Approximation	IoU	Intersection over Union	NSD	Normalized Surface Distance
Box/Approx IoU	Box/Approximation Intersection over Union				

Fig. SN 2.8. *Metrics Reloaded* recommendation framework for object detection at a glance.

Based on the choice of localization criterion and assignment strategy, standard classification metrics can be computed on object level. Importantly from a mathematical perspective in this context, the absence of True Negatives (TNs) in object detection problems renders many popular classification metrics (e.g., Accuracy, Specificity, Area under the Receiver Operating Characteristic Curve (AUROC)) invalid. Based on these considerations and taking into account all the complementary strengths and weaknesses of existing metrics [85], we propose the following steps for object detection problems (green path in Fig. 2 and SN 2.8):

- 1: Select localization criterion:** The selection of the localization criterion should be performed according to Subprocess S8 (Extended Data Fig. 8). If a rough outline of objects is desired, rather than just obtaining the object position (FP2.4 Desired granularity of localization, see Suppl. Note 1.3 for details), our recommendation is the *Box/Approximation Intersection over Union (IoU)* (Fig. SN 3.38). If only the position of objects is relevant from a domain interest (e.g. for determining the location of cells), the *Center Distance* (Fig. SN 2.28) is often an attractive option, although *Mask IoU > 0* (Fig. SN 3.39) or *Point inside Mask/Box/Approximation* (Fig. SN 3.40) are viable alternatives in case of fine-granular reference annotations (FP4.4 Granularity of provided references).
- 2: Select assignment strategy:** The recommendations for the assignment strategy are provided in Subprocess S9 (Extended Data Fig. 9). In case of the availability of predicted class scores (FP5.1 = TRUE) *Greedy (by Score) Matching* (Fig. SN 3.41) is our default recommendation.

Otherwise, *Greedy (by Localization criterion) Matching* (Fig. SN 3.42), *Optimal (Hungarian) Matching* (Fig. SN 3.43) or *Matching via Overlap > 0.5* (Fig. SN 3.44) are viable options, as detailed in decision guide 9.1 in Suppl. Note 2.7.8. The user must also decide whether double assignments should be punished (FP2.5.8).

- 3: Select classification metric(s) (if any):** Once objects have been located and assigned to reference objects, generation of a confusion matrix (without TN) is possible. The final step therefore simply comprises choosing suitable classification metrics.
- a: Select counting metric (if any):** The selection of a per-class counting metric according to Subprocess S3 (Extended Data Fig. 3) is governed by the decision rule (FP2.6). If a target value for a specific target metric is provided (e.g. Sensitivity = 0.95; Fig. SN 3.16), complementary metrics such as Positive Predictive Value (PPV) (Fig. SN 3.15) can be assessed at the provided point on the decision curve (Fig. SN 2.9). Otherwise, we recommend the F_β Score (Fig. SN 3.7) as a counting metric.
- b: Select multi-threshold metric:** Several subfields of biomedical image analysis have converged to choosing solely a counting metric as the primary metric. This choice seems to be a historical artifact from when algorithms did not provide predicted class scores. We generally recommend not discarding the scores typically provided by current algorithms and disagree with the practice of basing performance assessment solely on a single, potentially suboptimal, decision rule applied to the predicted class scores. Instead, we primarily propose selecting a multi-threshold metric (Subprocess S4, Extended Data Fig. 4) to present a more holistic picture of performance. As multi-threshold metric, we recommend Average Precision (AP) (Fig. SN 3.20) or Free-Response Receiver Operating Characteristic (FROC) Score (Fig. SN 3.21), depending on whether an easy interpretation (FROC Score) or a standardized metric (AP) is preferred (see decision guide 4.2 in Suppl. Note 2.7.3).

Note that the previous description implicitly assumed single-class problems, but generalization to multi-class problems is straightforward by applying the validation per class.

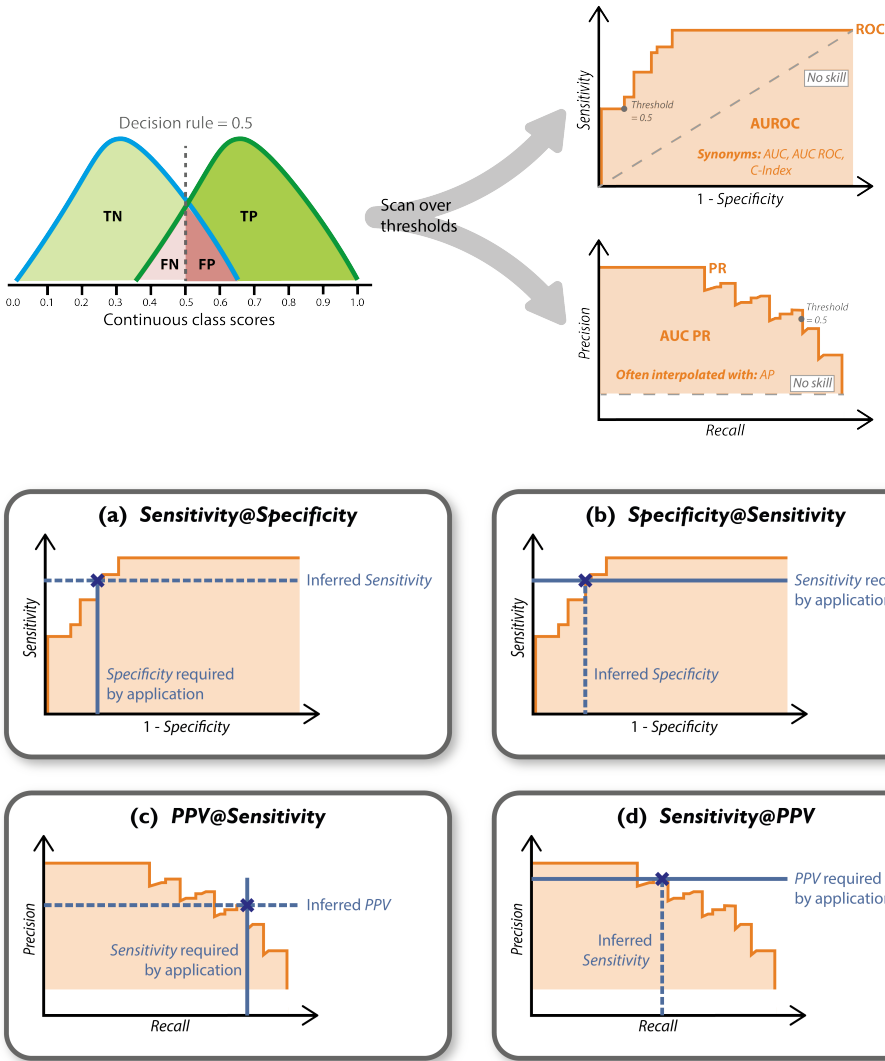


Fig. SN 2.9. Principle of multi-threshold metrics (top) and per-class counting metrics with application-driven thresholds (bottom). Rather than being based on a static threshold (e.g., for generating the confusion matrix), multi-threshold-based metrics integrate over a range of thresholds. Prominent examples are the Area under the Receiver Operating Characteristic Curve (AUROC) (also known as Area under the curve (AUC) or AUC Receiver Operating Characteristic (ROC)) and the Area under the Precision-Recall (PR) curve (AUC PR). Cardinalities, i.e., the true (T)/false (F) positives (P)/negatives (N), are computed based on a threshold (e.g., 0.5) of predicted class probabilities (left). Based on those values, Sensitivity (also known as Recall) and 1 - Specificity/Positive Predictive Value (PPV) are calculated and plotted against each other (right). The procedure is repeated for several thresholds, resulting in the ROC/PR curve. The area under the ROC/PR curve is referred to as AUROC/AUC PR. The latter is often interpolated by the Average Precision (AP) metric. The dashed gray lines refer to a classifier with no skill level (random guessing). In the case of an application-driven threshold (e.g., required Sensitivity of 0.95), the metrics Sensitivity@Specificity, Specificity@Sensitivity, PPV@Sensitivity and Sensitivity@PPV can be calculated on the basis of the ROC/PR curves. Please note that we use the synonyms Precision instead of PPV and Recall instead of Sensitivity for the PR curve, given their common use.

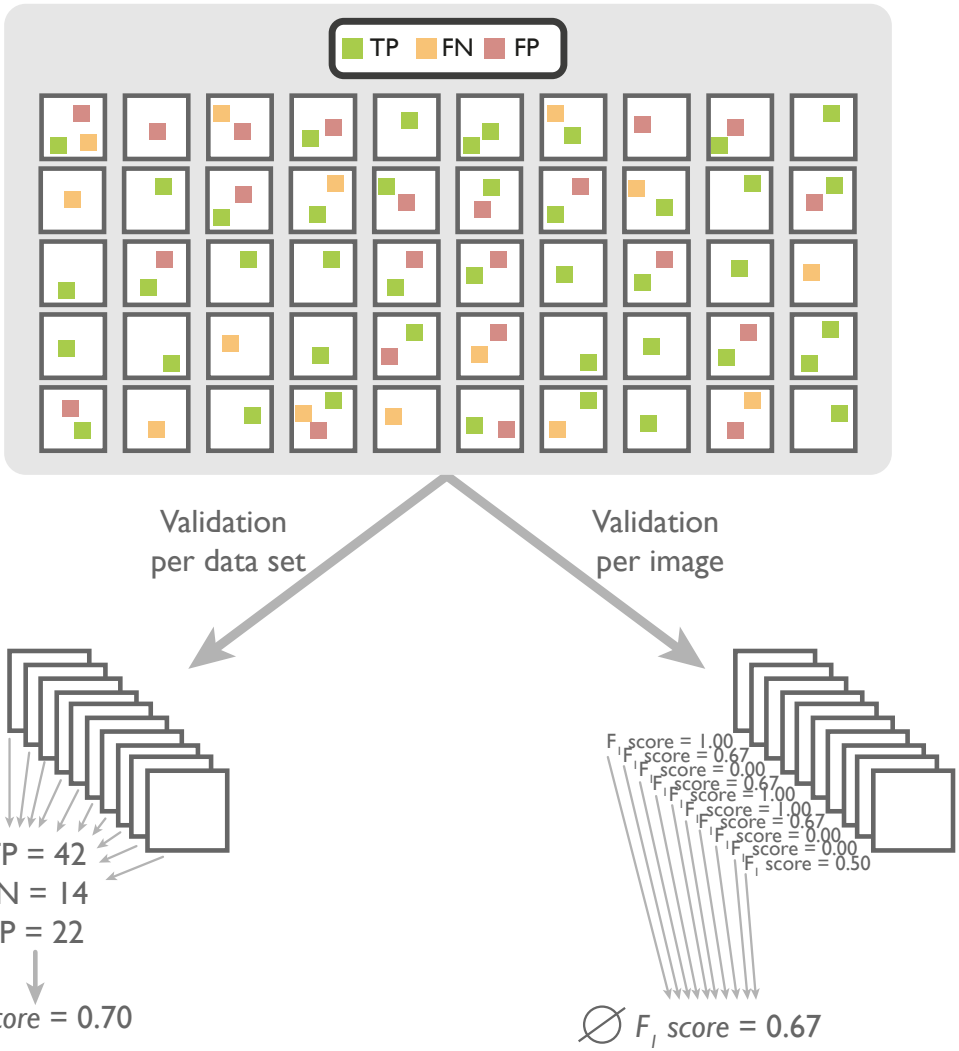


Fig. SN 2.10. Validation on object level can be performed per data set (left) or per image (right). For the per-data set validation of objects, the cardinalities are calculated over the whole data set. For the per-image validation of objects, metric scores are computed per image and aggregated afterward. \emptyset refers to the average F_1 Score.

It is further worth mentioning that metric *application* is not straightforward in object detection problems. One example is the fact that the number of objects in an image may be extremely small (even zero) compared to the number of pixels in an image. Special considerations with respect to **aggregation strategy** must therefore be made (Fig. SN 2.10). In fact, in the machine learning community, object detection tasks are typically validated by pooling all matched objects (i.e. TP, FP, and FN) over the entire data set and computing global metrics on the entire pool ('per-data set aggregation'). An alternative strategy is the 'per-image aggregation', where matched objects are aggregated per individual image to compute corresponding metrics (e.g., F_β Score). The per-image metrics are subsequently averaged over the data set. This alternative aggregation may be desirable for two reasons. Firstly, due to the hierarchical data structure (potentially multiple objects per image and/or multiple images per patient), a hierarchical aggregation of metric values, which compensates for the non-independence of images, is generally recommended. Secondly, from a domain interest, the expected metric value per image (rather than per entire data set) may be desirable. Importantly, the per-image aggregation strategy also changes the way multi-threshold metrics such as AP are computed: While the thresholds are still scanned over the scale of predicted class scores simultaneously for the entire data set, the precision and recall used to generate the PR-curve are now per-image scores averaged over the data set rather than the global per-data set scores. It should further be noted that validating an object-level problem per image rather than per data set comes with the problem that images containing no reference or prediction objects lead to division by zero for some metrics and thus to 'Not a Number' (NaN) as metric output. We therefore propose strategies for NaN handling in Fig. SN 2.11a. In summary, we recommend to exclude NaN cases from metric computation except when an empty prediction corresponds to an empty reference, in which case PPV, and in extension F_β Score, should be set to 1.

A further critical consideration for metric application in object detection is the fact that structure sizes may have a large effect on performance metrics [102]. We therefore recommend size stratification, i.e., the separate validation for different size ranges, if size variability is high (FP3.2 = TRUE).

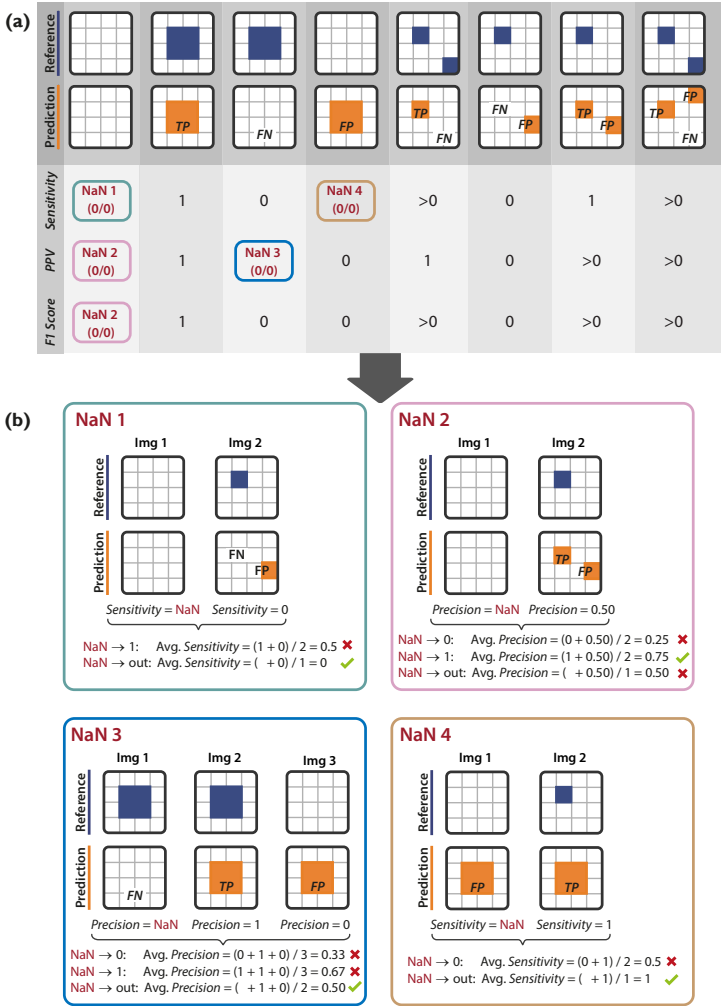


Fig. SN 2.11. Effect of handling a 'Not a Number' (NaN) occurring during metric computation, when object detection/instance segmentation tasks are validated per image. Specifically, NaN cases occur when an image features no target structures and/or no object predictions by the model, which causes division by zero errors in prevalent metrics. (a) Demonstration of how and when NaN can occur. Each column represents a potential scenario for per-image validation of objects, categorized by whether TPs, FNs, and FPs are present ($n > 0$) or not present ($n = 0$) after matching/assignment. The sketches on the top showcase each scenario when setting " $n > 0$ " to " $n = 1$ ". For each scenario, Sensitivity, Positive Predictive Value (PPV), and F_1 Score are calculated. (b) Effect of different NaN handling strategies based on different conventions for the aggregation across multiple images. Four examples are shown for the NaN scenarios from (a) (NaN 1-4). NaN 1 and 4: The intuitive penalization for FPs in "empty" images is already established by means of PPV scores (see NaN 4) and further penalization by means of Sensitivity is neither required nor appropriate. Instead, images without reference objects should be ignored when averaging Sensitivity scores over images. NaN 2: The intuitive penalization for FP in "empty" images is established when assigning a PPV (and F_1 Score) of 1. NaN 3: The intuitive penalization for FP is established when removing images with FN and no FP from the aggregation of PPV (and F_1) scores.

2.5 Recommendations for Instance segmentation

Essentials

FPX.Y refers to a fingerprint item detailed in Figs. [SN 1.15-SN 1.17](#).

SX refers to a subprocess in Extended Data Figs. 3-4 and Extended Data Figs. 6-9.

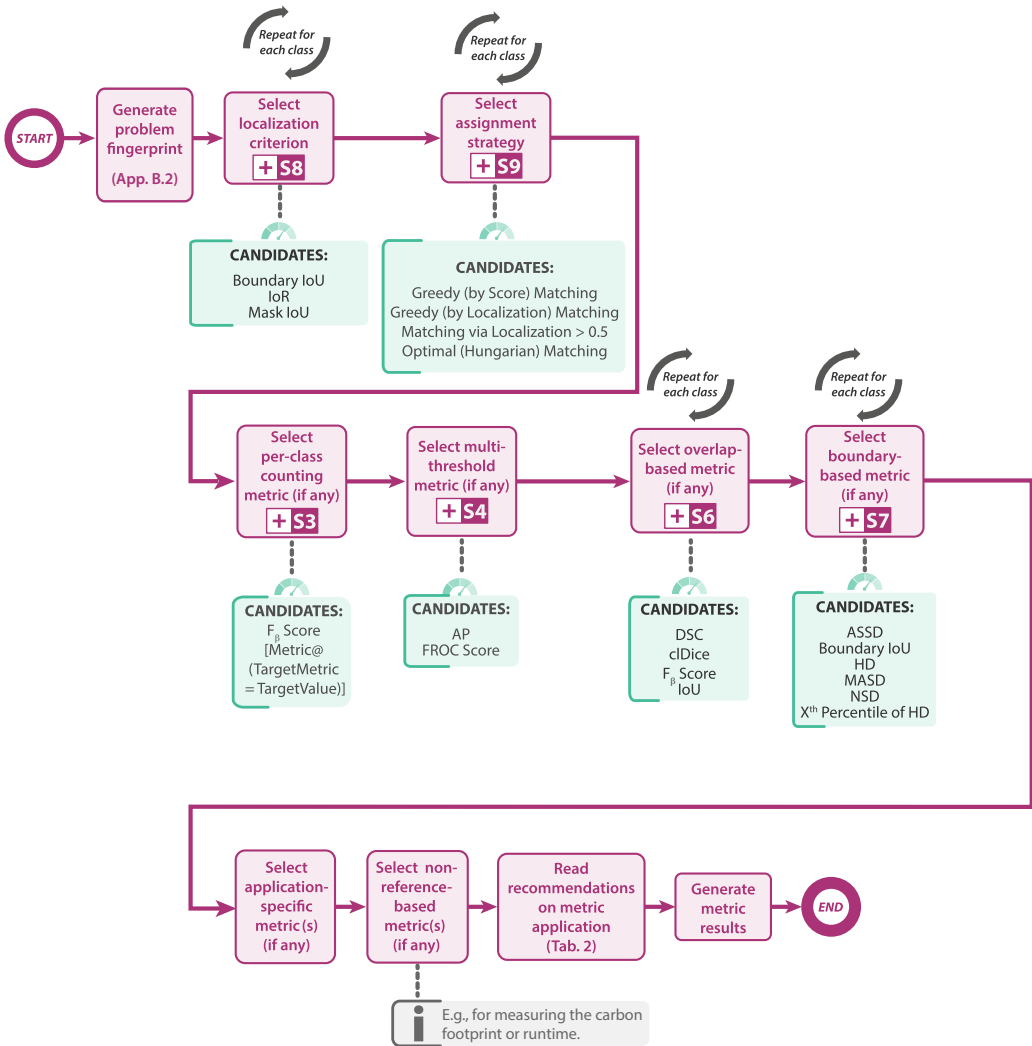
DGX.Y refers to a decision guide in Suppl. Notes [2.7.2-2.7.8](#).

Segmentation metrics are to be applied per instance.

This section provides recommendations for selecting *common reference-based metrics* for instance segmentation problems. As depicted in Fig. 2, these common metrics can then be complemented by application-specific metrics as well as non-reference-based metrics (assessing run time or carbon footprint, for example).

Instance segmentation can be regarded as delivering the tasks of object detection and semantic segmentation at the same time. In contrast to object detection, instance segmentation also involves the accurate marking of the object boundary. In contrast to semantic segmentation, it distinguishes different instances of the same class. The pitfalls and recommendations for instance segmentation problems are closely related to those for segmentation and object detection [85] and we recommend reading Suppl. Note [2.4](#) and Suppl. Note [2.3](#) as a foundation for this section.

Our recommendations for assessing instance segmentation quality can be summarized as follows (purple path in Fig. 2 and Fig. [SN 2.12](#)):



ABBREVIATIONS					
AP	Average Precision	DSC	Dice Similarity Coefficient	IoU	Intersection over Union
ASSD	Average Symmetric Surface Distance	FROC Score	Free-Response Receiver Operating Characteristic Score	Mask IoU	Mask Intersection over Union
Boundary IoU	Boundary Intersection over Union	HD	Hausdorff Distance	MASD	Mean Absolute Surface Distance
Box/Approx	Box/Approximation	IoR	Intersection over Reference	NSD	Normalized Surface Distance
Box/Approx IoU	Box/Approximation Intersection over Union			PQ	Panoptic Quality
cDice	Centerline Dice Similarity Coefficient				

Fig. SN 2.12. *Metrics Reloaded* recommendation framework for instance segmentation at a glance.

1: Select object detection metric(s): From a semantic segmentation perspective, overcoming problems related to instance unawareness (Fig. 1a (top left)) requires the selection of a set of detection metrics to explicitly measure detection performance. To this end, we follow the same general process as in the object detection recommendation by selecting a localization criterion, an assignment strategy, and suitable classification metrics. The specific recommendations for instance segmentation are:

- a: Select localization criterion:** Although not common in practice, we argue that for consistency it might be appropriate to base the localization criterion on the corresponding target segmentation metric (see step 2: "Select segmentation metric(s) (if any)" below). For example, if the target metric is Normalized Surface Distance (NSD), the localization criterion could be defined accordingly. This may not always be possible, for instance because the target metric has no fixed upper bound (e.g., Hausdorff Distance (HD)), rendering the setting of adequate cutoffs challenging. As an alternative strategy, we therefore recommend choosing the localization criterion according to common practice (see Subprocess S8, Extended Data Fig. 8). For this strategy, given the fine granularity of both the output and the reference annotation, we recommend selecting between Boundary Intersection over Union (IoU) (Fig. SN 3.35), Mask IoU (Fig. SN 3.38), and Intersection over Reference (IoR), (Fig. SN 3.37) using decision guide 8.1 in Suppl. Note 2.7.7.
- b: Select assignment strategy:** The recommendations for the assignment strategy are identical to those for object detection (Extended Data Fig. 9). In case of the availability of predicted class scores (FP5.1 = TRUE) Greedy (by Score) Matching is our default recommendation. Otherwise, Greedy (by Localization criterion) Matching, Optimal (Hungarian) Matching, or Matching via Overlap > 0.5 are viable options, as detailed in decision guide 9.1 in Suppl. Note 2.7.8. The user must also decide whether double assignments should be punished (FP2.5.8).
- c: Select classification metrics:** Our recommendations with respect to classification metrics are identical to those for object detection (Suppl. Note 2.4) with a single exception. As depicted in S3, Extended Data Fig. 3, we recommend the Panoptic Quality (PQ) (Fig. SN 3.13) [56] as an alternative to the F_β Score (Fig. SN 3.7). As illustrated in Fig. SN 2.13, this metric is especially suited for instance segmentation, as it combines the assessment of overall detection performance and segmentation quality of successfully matched (True Positive (TP)) instances in a single score.
- 2: Select segmentation metric(s) (if any):** In a second step, metrics for explicit assessment of the segmentation quality for the TP instances, i.e., successfully matched instances, may be selected. Here, we follow the exact same process as in semantic segmentation (Subprocesses S6, Extended Data Fig. 6 and S7, Extended Data Fig. 7). The primary difference is that the segmentation metrics are computed per-instance and subsequently averaged resulting, for example, in a 'Dice Similarity Coefficient (DSC) per instance' score.

While we have found our recommendations for instance segmentation to match the majority of biomedical problems, standard reference-based metrics are not well-suited for some applications. Specifically, standard metrics struggle in images with structures of extreme density and complex shapes, because overlap often fails as a criterion to establish unique correspondences between predicted and reference instances. In such cases, specialized metrics not relying on one-to-one correspondences may be required, such as pair-counting metrics or information theoretic-based metrics [97]. Another example that calls for application-specific metrics is cell nucleus segmentation, where splitting a reference object by two separate predictions is assessed by a dedicated 'split error', and the converse by a dedicated "merge error" [21]. These application-specific errors can either be used as stand-alone metrics or integrated into compound metrics such as F_1 Score.

Recommendations for aggregating object detection and instance segmentation metrics are provided in the respective appendices Suppl. Note 2.4 and Suppl. Note 2.5, respectively.

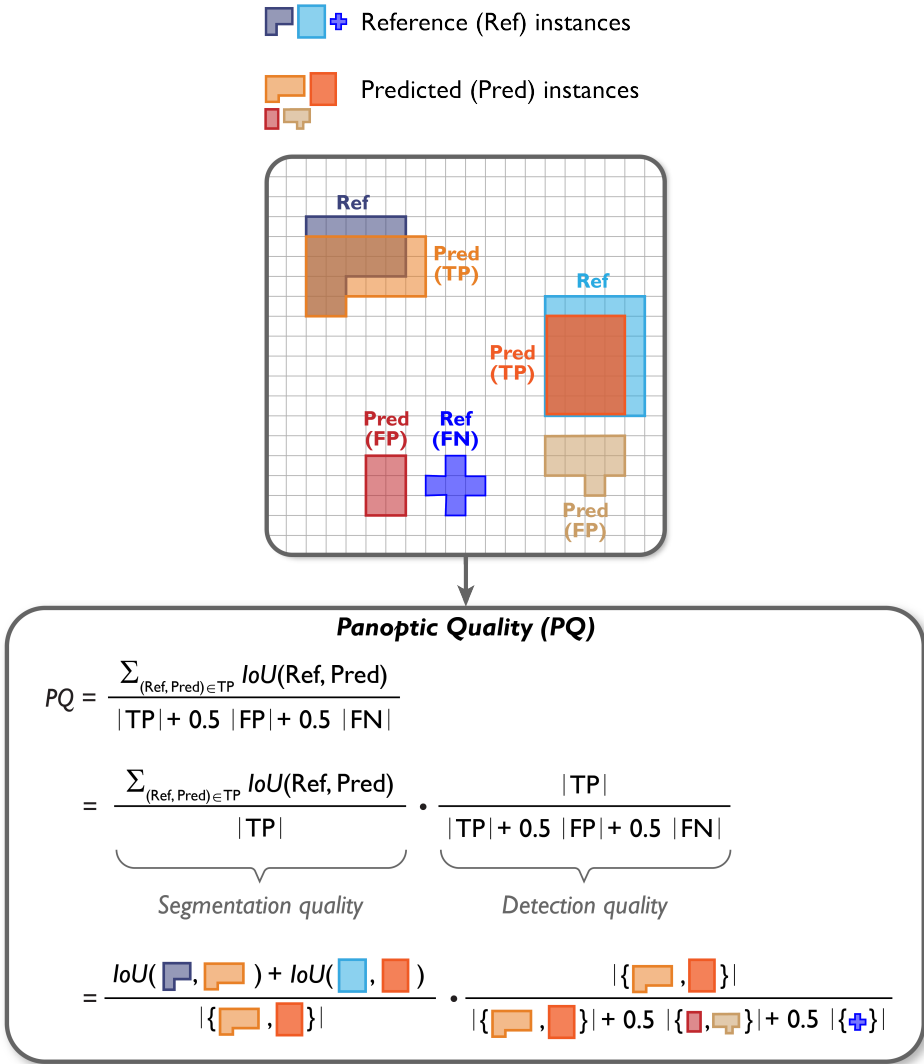


Fig. SN 2.13. The Panoptic Quality (PQ) measures the **segmentation and detection quality** of a prediction in one score. The metric simply averages the IoU scores for all True Positive (TP) instances and multiplies the result with the F₁ score. For perfect segmentation results, i.e., an average IoU of 1, the PQ would equal the F₁ Score.

2.6 Recommendations for Calibration of Predicted Class Scores

While most research in biomedical image analysis focuses on the discrimination capabilities of classifiers, a complementary property of relevance is the *calibration* of predicted class scores. Importantly, a large portion of the research in this field is comparatively young, and a variety of new calibration metrics are proposed every year. As it might be premature to call for rigid standardization in such a vibrant environment, the following recommendations are to be seen as general guidance through the current landscape of calibration metrics, which might be subject to updates in the following years.

Intuitively speaking, a system is well-calibrated if the predicted class scores (i.e., the output of the model) reflect the true probabilities of the outcome. In practice, this means that calibrated scores match the empirical success rate of associated predictions. For example, in a binary classification task, calibration implies that of all the data samples assigned a predicted score of 0.8 for the positive class, empirically, 80% belong to this class.

One common but critically important misconception about calibration is that the predicted class scores of a well-calibrated model express the true posterior probability $\mathbb{P}_{Y|X}$ of an input belonging to a certain class [82], e.g., that they express a patient’s risk for a certain condition based on an image. While this probability is commonly of interest in classification problems, common calibration metrics instead typically consider $\mathbb{P}_{Y|f(X)}$, i.e., the probability of a model’s output score belonging to a certain class. Conditioning on the output entirely ignores the mapping $f: \mathcal{X} \rightarrow \mathcal{P}$. Thus, while calibration allows making statements about the empirical class membership of predicted scores, such as in the example above, these statements are conditioned on the discrimination power of a model. This means that different models may predict different probabilities for the same input even though all of them are perfectly calibrated [82]. Going back to the clinical example, this implies that a classifier that always predicts the score 0.5 is considered perfectly calibrated on a balanced binary task, although another perfectly calibrated model with better discrimination ability could output completely different, practically more meaningful scores. Again, this discrepancy occurs because calibrated scores reflect the empirical success rate of predictions and not a patient-specific (model-agnostic) inherent risk. The clinical prediction modelling community therefore traditionally distinguishes different levels of calibration [106], where *level 4 strong calibration* implies correct posteriors ($\mathbb{P}_{Y|X}$). As level 4 is practically unfeasible to measure (the true individual posteriors are unknown), common research focuses on *level 3 moderate calibration*, which implies that the predicted scores match the empirical success rate.

For a more formal definition of (level 3) calibration, let the random variables X and Y correspond to the feature (e.g., an image) and target variables (encoding the outcome), respectively, with feature and target spaces \mathcal{X} and \mathcal{Y} . Let $f: \mathcal{X} \rightarrow \mathcal{P}$ denote a classifier with predicted class scores $f(X)$ and \mathcal{P} a set of distributions on \mathcal{Y} . We further use the notation $\mathbb{P}_Y, \mathbb{P}_{Y|f(X)} \in \mathcal{P}$, where \mathbb{P}_Y refers to the distribution of Y , and $\mathbb{P}_{Y|f(X)}$ to the conditional distribution of Y given $f(X)$.

In practice, three different variations of calibration conditions can be distinguished [105]:

- **Canonical calibration:** $f(X) = \mathbb{P}_{Y|f(X)}$. This condition implies pairwise matching of all entries across the two distributions (see also the top panel in Fig. SN 2.14). Although not the most commonly applied condition in practice, a common perception is that this condition is the appropriate perspective on calibration in many application scenarios as the weaker conditions (see below) are prone to underestimating miscalibration [40, 43, 83].
- **Class-wise calibration:** $f_k(X) = \mathbb{P}(Y = k | f_k(X))$ for all classes $k \in \mathcal{Y}$. This is a weaker condition, where not the joint, but the marginal distributions for each class are required

to match (see also the middle panel in Fig. SN 2.14). Assessing the calibration quality for individual classes provides crucial information, for example in scenarios where there is a mismatch between class prevalences and class importance (FP2.5.3=TRUE).

- **Top-label calibration:** $f_K(X) = \mathbb{P}(Y = K \mid f_K(X))$, where $K = \arg \max_k f_k(X)$ of a model $f: \mathcal{X} \rightarrow \mathcal{P}$. This is the weakest of the three conditions, where only the maximum entry (top label) of each predicted class vector is considered (see also the bottom panel in Fig. SN 2.14). This condition assesses only the highest class score, which is often used to determine the predicted class, and thus implies a strong focus on validating the reliability of a model's decisions.

While these three conditions are equivalent for binary classification problems, they may differ substantially in the broader multi-class setting, as illustrated in Fig. SN 2.14.

In practice, no model is perfectly calibrated. Calibration quality is captured by the Calibration Error (CE), which can be computed via a divergence, i.e., a distance function, between predictions $f(X)$ and either of the three conditions (canonical, class-wise, top-label). For instance, typical choices for quantifying the canonical CE are expected L_1 or L_2 errors [43, 60, 74]. These can be further generalized to the L_p CE: For $1 \leq p \in \mathbb{R}$, the canonical ℓ_p CE (CE_p) of model $f: \mathcal{X} \rightarrow \mathcal{P}$ is defined as:

$$CE_p(f) = \left(\mathbb{E} \left[\left\| f(X) - \mathbb{P}_{Y|f(X)} \right\|_p^p \right] \right)^{\frac{1}{p}}. \quad (1)$$

The relations of CE variants associated with the three conditions above intuitively translate to $CE_{canonical} \geq CE_{class-wise} \geq CE_{top-label}$. In the example provided in Fig. SN 2.14, the weaker conditions of top-label calibration and class-wise calibration are fulfilled (associated errors are zero), while the broader canonical condition for calibration is not met. The fact that the calibration quality of a classifier varies when assessed through the lens of different conditions causes common calibration measures to be characterized as *inconsistent* in multi-class settings [43].

The canonical ℓ_p CE can be generalized by replacing the ℓ_p norm as a distance measure between $f(X)$ and $\mathbb{P}_{Y|f(X)}$ with alternative distance functions. For example, [115] introduced a canonical CE based on matrix-valued kernels.

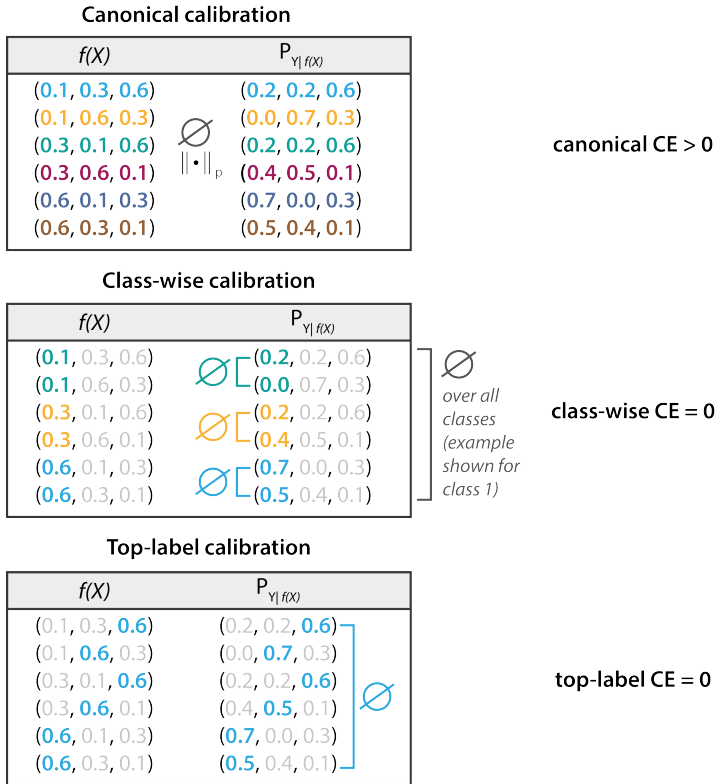


Fig. SN 2.14. Estimating the Calibration Error (CE) according to the three different conditions in multi-class settings yields inconsistent results. For the top-label calibration, only the maximum values of the predicted class scores $f(X)$ are considered, while all other values are neglected. For the computation of the CE, for each distinct output value of $f(X)$ (only 0.6 in this case), $\mathbb{P}_{Y|f(X)}$ is determined as the average over the empirical rates of this output (0.6, 0.7, 0.6, 0.5, 0.7, 0.5 in this case). The top-label calibration condition (i.e., matching the two scores) results in a perfect CE = 0 in this scenario. Similarly, for the class-wise calibration, the predicted class scores are compared per class, a requirement that is also fulfilled by the depicted system. Only the canonical calibration, which comes with the strict requirement that the model output must match the full probability distribution (implying the comparison of entire vectors rather than single values) indicates a miscalibrated system (CE > 0). This figure is inspired by [105].

Generally, measuring the CE is challenging, because $\mathbb{P}_{Y|f(X)}$ is unknown and needs to be estimated as the expected value on the available data. Returning to the simple example above, as we only have access to a small subset of all potential cases for which the model would predict a score of 0.8, we do not know whether the corresponding success rate of these cases is 80% in general; instead, our assessment relies on the estimated success rate based on the available samples. The fact that classifier outputs are generally continuous often reduces the number of available samples per prediction to one. Strategies for alleviating the sparse sampling problem include binning the continuous scale of $f(X)$ and estimating the CE per bin (such as done for Expected Calibration Error (ECE) (Fig. SN 3.30) and Class-wise Calibration Error (CWCE) (Fig. SN 3.29), as illustrated in Fig. SN 2.16), or using kernel density estimation methods (such as done for Expected Calibration Error Kernel Density Estimate (ECE^{KDE}), see Fig. SN 3.31). Despite these efforts, the most popular

calibration measures are generally biased estimators of the true error, which means their estimates depend on the number of samples (i.e., size of the validation data set). Gruber et al. [43] recently described this bias and how it leads to substantial under- and over-estimations of the true error. Popordanoska et al. [83] showed that straightforward estimators of ℓ_p calibration based on density estimation (such as done for ECE^{KDE}) have a generally lower bias compared to statistical estimators (such as binning) and presented means to additionally de-bias estimators. There are also ongoing efforts investigating canonical CEs that are not based on the ℓ_p norm, such as the Kernel Calibration Error (KCE), where ‘maximum mean discrepancy’ is used as a distance function instead (see Fig. SN 3.31). These efforts have resulted in fully unbiased estimators, which, however, do not allow for interpretable calibration assessment and further require nontrivial configuration of the kernels and associated hyperparameters.

An attractive alternative to estimate CEs are so-called Proper Scoring Rules (PSRs) (also referred to as *overall performance measures* [96]), which measure discrimination and calibration in a single score (e.g. Negative Log Likelihood (NLL), Brier Score (BS); Figs. SN 3.28, SN 3.33). An intuitive example for this metric category is the BS: For a model $f: \mathcal{X} \rightarrow \mathcal{P}$ the BS is defined as the expected value of the squared error between predictions and reference values as determined on the validation data:

$$\text{BS}(f) = \mathbb{E} \left[\|f(X) - Y'\|_2^2 \right], \quad (2)$$

where Y' is the one-hot-encoded version of the reference vector Y for each individual data sample. This equation illustrates the difference between overall performance measures and calibration metrics measuring the CE in Equation 1. While the CE measures whether the predicted class scores match the empirical success rate (see also SN 2.14), BS is defined as an expected value over every single prediction, thus posing a stronger requirement on the scores which can be interpreted as assessing the true posterior probabilities or individual risks. In theory, BS can be decomposed into explicit terms for discrimination and calibration assessment [34]. In practice, however, although overall performance measures do not suffer from the sampling problem, they conflate the true CE with the discrimination error and can thus not make calibration quality explicit. However, proper scores are still useful for comparative studies. Furthermore, it has been shown that the square root of the BS, referred to as the Root Brier Score (RBS) (Fig. SN 3.34), represents a robust estimator and upper bound of the canonical CE [43]. Such a guarantee can be particularly relevant in safety-critical scenarios where the error must not be underestimated.

The choice of which calibration condition to validate as well as which metric to use depends on the task interest. Methods subject to validation in this context are either classification models whose inherent calibration quality shall be assessed, or so-called ‘re-calibration methods’, i.e., transformations on the classifier outputs aiming to improve the calibration quality. In the most common scenarios, the driving interest may either be a comparative performance assessment, in which methods are ranked according to calibration quality, or an absolute performance assessment, in which an interpretable and communicable measure of calibration quality is desired. We identified four main use cases (U1-U4) which our framework addresses (Fig. SN 2.15).

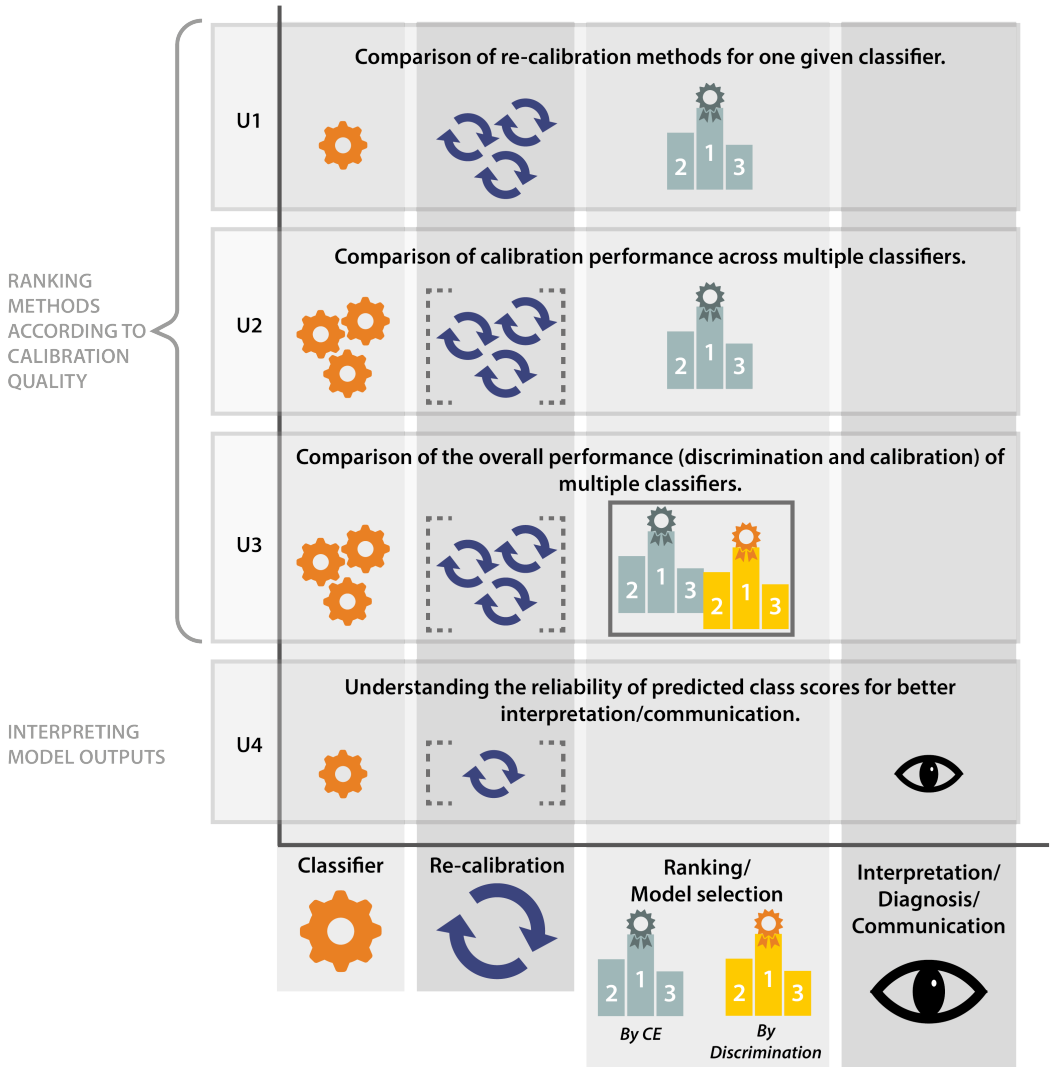


Fig. SN 2.15. Underlying interest related to the assessment of calibration quality. The user is interested in the comparative calibration assessment (U1-U3) and/or obtaining a reliable estimate of the Calibration Error (CE) for interpreting and communicating the algorithm output (U4). The use cases are detailed in Suppl. Note 2.6. The brackets around re-calibration methods denote that their application is optional in the corresponding use case.

- (1) **Ranking methods to determine calibration quality:** The following use cases focus on the comparative assessment of the calibration quality of one or multiple classifiers.
 - (a) **Use case 1 (U1):** comparing the effect of one or more re-calibration methods on the same (fixed) classifier. The desired validation output is a ranking of re-calibration methods (possibly including the performance of 'no re-calibration') from which the best method can be selected.
 - (b) **Use case 2 (U2):** comparing the calibration quality across multiple classifiers on the same task. The desired validation output is a ranking of classifiers according to calibration quality. In practice, such a ranking should be accompanied by a ranking according to discrimination performance, as it is not recommended to base model selection purely on calibration performance.
 - (c) **Use case 3 (U3):** comparing the 'overall performance' of classifiers (optionally including potential re-calibration methods), i.e., a joint assessment of discrimination performance and calibration quality. The desired validation output is a single ranking naturally weighing both aspects.
- (2) **Interpreting model outputs:** Complementary interest may lie in the analysis of the CE to the end of assessing the reliability of the predicted class scores of one or multiple classifiers.
 - (a) **Use case 4 (U4):** interest in understanding the reliability of predicted class scores for a given model as a basis for interpreting and communicating results. The desired validation output is a single score which provides insight into how well the model is calibrated. The reliability of model outputs is often considered crucial upon application, such as for clinical prediction models [36, 107, 118]. Importantly, U4 can be used in addition to U1, U2 or U3 as it is based on an orthogonal interest.

Because some decision rules assume calibrated model outputs, a further potential interest in calibration validation may lie in determining the quality of a decision rule applied to predicted class scores (see FP2.6), i.e., answering the question: "How much better could the classifier's decisions have been under this rule if predicted class scores were calibrated?". While such ablations of classifier design decisions are generally out of the scope of our framework, decision rule-related pitfalls and countermeasures are discussed in Sec. 1.1.

Based on all of the above considerations, we recommend selecting calibration metrics using Subprocess S5 (Extended Data Fig. 5) in case the assessment of calibration quality is desired (FP2.7.1 = TRUE):

1: Select metric for comparative calibration assessment (if any): This step selects an adequate metric in case a comparative assessment of calibration methods is desired (FP2.7.2). The fingerprint FP2.7.2 covers the presented use cases U1-U3 (Fig. SN 2.15). For U1 "Comparison of re-calibration methods for the same fixed classifier", one option is to select a metric that assesses the canonical CE, such as KCE as an unbiased estimator of a canonical CE based on an alternative distance function, or ECE^{KDE} as a well-interpretable estimator of canonical ℓ_p calibration. Alternatively, an overall performance measure such as the BS can be used (see DG5.2), because the classifier is fixed in this scenario, the conflation of the CE with discrimination errors is no disturbing factor, and the true CE is exposed for relative comparison of scores. For U2 "Comparison of calibration quality across classifiers on the same task", we recommend reporting the CE per class by using an estimator of marginal CE, such as CWCE, if there is an unequal interest across classes (FP2.5.1). Otherwise the canonical CE should be assessed, e.g. using KCE or ECE^{KDE} (see DG5.1). For U3 "Comparison of overall performance across classifiers", we recommend reporting a PSR (i.e., BS or NLL, see

DG5.3) as the joint assessment of calibration and discrimination is exactly what this category of metrics is designed for.

- 2: Select metric for assessing output interpretability (if any):** This step selects an adequate metric for assessing the interpretability of the model output (FP2.7.3), which corresponds to U4. The first decision to be made in FP2.7.3 is whether to assess the calibration quality in isolation, as measured by CE estimates, or jointly with discrimination as measured by overall performance measures. When deciding for calibration-only assessment, the core decision to be made is whether to measure top-label, marginal or canonical CE, as detailed in DG 5.4. If there is an unequal interest across classes (FP2.5.3), a well-interpretable estimator of the marginal CE, such as CWCE, is recommended. Otherwise, the default option is to select a well-interpretable estimator of the canonical CE (e.g., ECE^{KDE}) and a corresponding guaranteed upper bound (e.g., RBS), together with the a per-class estimator of marginal CE (e.g., CWCE). Top-label calibration (as measured by ECE) is only recommended in rare cases, as detailed in DG5.4.

Note that the selection of the same metric (e.g., CWCE) in both steps is a potential outcome of the mapping. Crucially, metrics involving calibration assessment are generally prevalence-dependent. Thus, comparative studies as described in U2 and U3 are generally restricted to one data set and, if the prevalence of the data set does not represent the population of interest (see FP4.2), the calibration quality of a classifier needs to be re-validated on each new study cohort (see Fig. SN 1.6).

Calibration is most commonly assessed for image-level classification tasks. Due to the comparatively sparse research basis in the other problem categories, no specific recommendations are provided in our framework at this time. There are however, a few recent studies employing calibration metrics in object detection [61, 79] and slightly more studies in semantic segmentation, especially in the biomedical domain [55, 63, 71, 90].

Nevertheless, in theory, Subprocess S5 may also be traversed for object detection, instance segmentation, and semantic segmentation. When traversing S5 for object-level tasks, the following considerations should be noted:

- **Calibration recommendations only apply to the classification part of object detectors:** As described in Suppl. Note 5.2, object detection and instance segmentation methods commonly provide outputs beyond the predicted class score vector such as bounding box coordinates or, depending on the method, 'intermediate objectness scores' [87]. Thus, it is important to note that when utilizing calibration recommendations in this framework for object-level methods, the recommendations only apply to the classification output.
- **Why considerations in image-level classification translate to object detection:** When validating *discrimination* performance, a fundamental conceptual difference between image level and object level is the fact that True Negatives (TNs) are not defined in the latter case. This difference does not translate to calibration, where only predictions of the model $f(X)$ are validated. As the background class is discarded from validation (see below), this means that only True Positive (TP) and False Positive (FP) predictions are relevant for calibration, i.e., non-matched predictions are considered while non-matched reference objects (False Negatives (FNs)) are not. A further conceptual difference between object-level classification and image-level classification is the former's additional requirement of localization to distinguish TPs and FPs. This aspect is inherently covered by calibration validation because non-matched predictions are simply considered as additional FP errors (mistaking the 'true' background

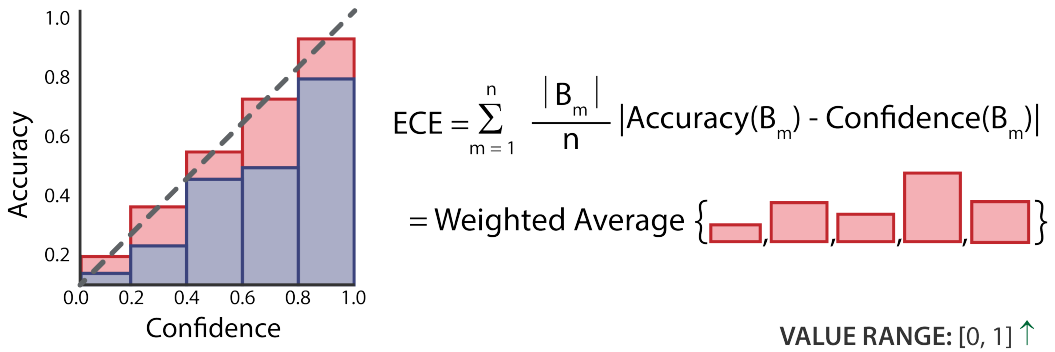


Fig. SN 2.16. Computation of the Expected Calibration Error (ECE) based on the binning of predicted class scores. The error is based on the discrepancy between the Accuracy per bin $Accuracy(B_m) = 1/|B_m| \sum_{i \in B_m} \mathbf{1}(\hat{y}_i = y_i)$ and the average over predicted class scores per bin $Confidence(B_m) = 1/|B_m| \sum_{i \in B_m} \hat{p}_i$. The final ECE score is obtained as the average over bin discrepancies weighted by the number of samples $|B_m|$ per bin. Here, n denotes the total number of samples, \hat{y}_i denotes the predicted class labels and y_i the true class labels, $\mathbf{1}$ is the indicator function (1 if $\hat{y}_i = y_i$, 0 otherwise), and \hat{p}_i refers to the predicted class scores. The dashed diagonal line acts as a visual reference for a perfectly calibrated system, where discrepancies between per-bin confidences and accuracies are zero.

class for one of the foreground classes), equivalently to the standard FP error case (mistaking two classes).

- Dealing with the background class:** When validating classification performance in object-level tasks, the model output predicting class scores for the background class is commonly discarded (see Suppl. Note 5.2 [39], because rewarding correct background predictions contradicts the task interest (there are no 'background objects') and would be easily exploitable (predicting high numbers of background objects). Further, penalization of background predictions is already ensured implicitly by considering them as FPs with respect to the true foreground class. Discarding the background class leads to class prediction vectors that do not sum to one, which is of no concern for validation as metrics do not rely on the probabilistic interpretation of scores. These considerations translate directly to calibration validation, which is equivalently exploitable by predicting high numbers of background objects. Here, the background class is discarded from outputs, and calibration of outputs only refers to the foreground class predictions (the actual 'object detection' outputs). Moreover, the calibration conditions introduced above describe the matching of single entries across two distributions and do not rely on their scores summing to one.
- Non-applicability of the NLL on object level:** There is one exception for object-level tasks where metric recommendations differ when traversing S5: The NLL is not applicable. This is because the NLL considers the predicted class score for the correct reference class ('true class probability'). For non-matched predictions, this 'true class' is the background class, which is discarded from validation as described above. In contrast, BS remains applicable and a meaningful measure of CE under the recommended protocol (i.e., when only considering foreground classes).

2.7 Decision Guides

While the problem fingerprint helps exclude common metrics that are not suitable for the driving problem, the final choice in each subprocess may not be unambiguous. In these cases, decision guides support the users in making an educated decision that best matches their preferences.

2.7.1 Decision guide S2.

DG2.1: Weighted Cohen's Kappa (WCK) versus Expected Cost (EC)

Summary of DG2.1: WCK versus EC

WCK

- ➖ Designed for symmetric situations (guesses of two raters)
- ➖ Limited interpretability
- ➕ Widely used
- ➖ Lack of framework to identify and validate the decision rule applied to class scores
- ➖ Possibility of paradoxical results

EC

- ➔ Designed for asymmetric situations
- ➕ Good interpretability with normalized variant
- ➔ Not widely used in biomedical image analysis
- ➕ Availability of framework to identify and validate the decision rule applied to class scores

Table SN 2.2. Comparison of Weighted Cohen's Kappa (WCK) to Expected Cost (EC) in the context of the decision guide DG2.1 for Subprocess S2. Context: unequal severity of class confusions (FP2.5.2 = TRUE), costs for class confusions available (FP2.5.3 = TRUE), and provided class prevalences reflect the population of interest (FP4.2 = TRUE).

Both WCK (Fig. SN 3.18) and EC (Fig. SN 3.6) are metrics that allow for incorporating task-specific penalties for confusions between individual pairs of classes. Common use cases for this property are tasks with ordinal classes or diagnostic decisions with errors of varying clinical severity. Importantly, however, Kappa statistics in general and WCK in particular were originally proposed to compare annotations/guesses of two raters, which is a symmetric problem by nature. Validation studies, on the other hand, involve the comparison of a prediction to a reference that approximates the truth (asymmetric setting). Hence, unlike EC, WCK does not conceptually match the intended comparison. For this reason and due to further favorable properties, we generally recommend the usage of EC rather than WCK. When deciding between the two metrics, the following further properties are of relevance:

- **Interpretability:** While both metrics can be interpreted as 'measures of (dis)agreement', the main difference is the fact that WCK provides this measure in reference to 'agreement by chance'. The equivalent concept for EC is its normalized variant normalized EC (ECN), where the disagreement measure is divided by a 'random performance' measure. Due to the conceptual similarity, it is more sensible to compare WCK to ECN. Both metrics are prevalence-dependent due to relating model performance to a random performance reference. Their main difference is the definition of the 'random reference': In ECN this reference is straightforward to interpret as the 'best possible naive classification system' which always predicts the most dominant class. The definition in WCK stems from its symmetric concept

to compare the predictions of two raters. The random reference in this case is the probability of both raters agreeing by chance. Using this definition in classification tasks results in random reference systems that can be weaker than the naive system of ECN. Thus, the random reference in WCK is less intuitive and arguably not useful in classification tasks (i.e., asymmetric settings).

- **Undesired behaviour in practice:** Using WCK with quadratic weights, often done for ordinal tasks, has been found to lead to 'paradoxical results', as detailed in [113].
- **Popularity:** WCK is widely used in the biomedical domain, whenever customized penalties for class confusion are required. EC, on the other hand, is currently mostly found either in statistical textbooks or in non-related domains such as speech recognition.
- **Theoretical foundation:** EC comes with a comprehensive theoretical foundation based on Bayesian decision theory [40]. As a consequence, it is possible to analytically derive the optimal decision rule applied to the predicted class scores (more generally: decision region for more than two classes) and empirically validate the quality of this decision rule by means of calibration assessment. This is an important property in this context because the standard argmax-based decision rule (i.e., predicting the class with the highest class score) is by definition not optimal in scenarios with unequal costs of misclassifications.

DG2.2: Balanced Accuracy (BA) versus Expected Cost (EC)

Summary of DG2.2: BA versus EC

BA

- ➔ Prevalence independence
- ➕ Widely used

EC

- ➕ Possibility of reflecting expected prevalences in the target population
- ➔ Not commonly known in biomedical image analysis

Table SN 2.3. Comparison of Balanced Accuracy (BA) to Expected Cost (EC) in the context of the decision guide DG2.2 for Subprocess S2. Context: Equal severity of class confusions (FP2.5.2 = FALSE), either (1) unequal interest across classes (FP2.5.1 = TRUE) and no mismatch between class prevalences and class importance (FP2.5.3 = FALSE), or (2) equal interest across classes (FP2.5.1 = FALSE) and provided class prevalences do not reflect the population of interest (FP4.2 = FALSE).

When deciding between BA and EC in the provided context, two primary scenarios should be distinguished:

Classes should contribute according to prevalence in the target application: Although the user may have an inherently equal interest in all classes (FP2.5.1 = FALSE), reporting a metric score to which all classes contribute equally may *not* necessarily be desired. Instead, the user may simply be interested in the overall performance on a given data set and thus want classes to contribute according to their prevalence in the target application. This is not straightforward in the provided scenario because the data set at hand does not match the prevalences of the target population (FP4.2 = FALSE). In this case, we recommend EC, because it offers a mechanism to explicit set (expected) class prevalences directly in the formula. This strategy allows getting a glimpse of a model's performance on the target application while validating on the data at hand. Application of EC in this way, however, is only possible if the prevalences can be specified upfront.

Each class should contribute equally to the metric: In this case, compensation for potential class imbalance is required in order to ensure equal contribution from each class. Here, we recommend BA as metric because it was designed for exactly this purpose. Although EC can be configured to be identical to BA (Suppl. Note 2.1), we favor BA due to its widespread use.

EC also offers a complementary way to address class imbalance as it allows for the exchange of the class priors directly in the formula: When the class priors upon application on a new data set are known, they can be incorporated in EC. This can be useful for identifying the optimal decision rule applied to predicted class scores on a new data set, as described in [40], essentially rendering the re-calibration of class scores unnecessary. However, one might argue that class priors being known upfront is quite a strong assumption for a new application.

DG2.3: Balanced Accuracy (BA) versus Matthews Correlation Coefficient (MCC) versus normalized EC (ECN)

Summary of DG2.3: BA versus MCC versus ECN

BA

- ⊕ Inherent interpretability with respect to naive classifier
- ⊖ Implication of equal class contribution
- ⊖ Insensitive to predictive values (Positive Predictive Value (PPV) and Negative Predictive Value (NPV))
- ⊕ Availability of framework to identify and validate decision rule applied to class scores
- ⊕ Good interpretability
- ⊕ Widely used

MCC

- ⊕ Inherent interpretability with respect to naive classifier
- ⊖ Implication of equal class contribution
- ⊖ High scores ensure high predictive values (PPV and NPV)
- ⊖ Lack of framework to identify and validate the decision rule applied to class scores
- ⊖ Limited interpretability
- ⊖ Fairly well-known but not much used

ECN

- ⊕ Inherent interpretability with respect to naive classifier
- ⊖ No establishment of equal class contribution
- ⊖ Limited sensitivity to predictive values (PPV and NPV)
- ⊕ Availability of framework to identify and validate the decision rule applied to class scores
- ⊕ Good interpretability
- ⊖ Not known or used in the biomedical imaging domain although based on well-studied statistical concepts

Table SN 2.4. Comparison of Balanced Accuracy (BA) to Matthews Correlation Coefficient (MCC) to normalized EC (ECN) in the context of the decision guide DG2.3 for Subprocess S2. Context: Equal severity of class confusions (FP2.5.2 = FALSE), either (1) unequal interest across classes (FP2.5.1 = TRUE) and no mismatch between class prevalences and class importance (FP2.5.3 = FALSE) or (2) equal interest across classes (FP2.5.1 = FALSE), provided class prevalences reflect the population of interest (FP4.2 = TRUE), presence of class imbalance (FP4.1 = TRUE) and compensation for class imbalances requested (FP2.5.5 = TRUE).

Three metrics are particularly attractive when class prevalences reflect the population of interest but compensation for class imbalance is desired (FP4.1 = TRUE and FP2.5.5 = TRUE). These are MCC, BA, and the normalized variant of EC, ECN. As described in Suppl. Note 1.3 (FP2.5.5 *Compensation for class imbalance requested*), there are three effects of class imbalance that can be compensated for.

- **Effect 1:** Misleading metric values due to missing reference value for naive classifier
- **Effect 2:** Misleading metric values due to unequal contribution of classes
- **Effect 3:** Misleading metric values due to missing consideration of predictive values

While the most common multi-class metric, Accuracy, is subject to all three pitfalls when used in imbalanced settings, this decision guide discusses the three aforementioned alternatives (BA, MCC, and ECN) that compensate for one or more of these effects. The following aspects are relevant when deciding between the three:

Compensating for Effect 1: All three metrics establish a fixed score for the performance of a naive classifier, i.e., one that always predicts the most frequent class – which is a more realistic baseline in class imbalanced scenarios – compared to an entirely random system. The corresponding scores are 0 for MCC, 1 for ECN, and $1/C$ for BA, where C is the number of classes. However, the nature of the different compensation methods is fundamentally different. Consider the following confusion matrix of a binary classification system, as shown in Tab. SN 2.5:

Table SN 2.5. Confusion matrix illustrating Effect 1.

		Prediction	
		Positive	Negative
Actual	Positive	TP = 100	FN = 1
	Negative	FP = 100	TN = 10,000

Respective metric values are BA: 0.99, MCC: 0.7, ECN: 1. Although all metrics feature fixed values for a random classifier, the same system can be assessed differently, as it is being considered 'near-perfect' by BA (0.99), 'fairly good' by MCC (0.7), and 'random'/'naive' by ECN (1). Intuitively, the BA assessment seems overly optimistic, which can be attributed to the fact that BA does not compensate for Effect 3, as described in more detail below. On the other hand, the ECN assessment appears overly strict, which can be attributed to the fact that ECN does not compensate for Effect 2 as described in more detail below.

Compensating for Effect 2: In balanced scenarios, all classes are weighted equally by common discrimination metrics. In contrast, in imbalanced scenarios, common metrics such as Accuracy are dominated by the frequent classes. Equal contribution of classes in this context would imply that each class can contribute equally to the final metric score, irrespective of prevalence. This is exactly what BA does by computing the average of individual class Sensitivities. An alternative way of thinking about this compensation is tweaking the costs of misclassification errors by assigning higher costs for errors in rare classes and vice versa. Hence, BA can be thought of as a cost instantiation of EC if the costs are set proportional to the inverse of class prevalences (see Suppl. Note 2.1). Importantly, the normalized variant of EC, ECN, does not generally compensate for Effect 2, but merely rescales metric scores in a way that the value of 1 corresponds to a naive classifier always predicting the most frequent class (see Effect 1). In other words, the rankings obtained for a set of test cases would be the same for EC and ECN. Analogously to EC, it is also possible to tweak the costs to compensate for Effect 2 in ECN, but the resulting metric would yield no advantages over BA. Importantly, the fact that ECN does not compensate for Effect 2 implies that if there is an unequal interest across classes ($FP_{2.5.1} = TRUE$), then ECN is the only correct choice. Analogously to BA, MCC establishes equal contribution of classes by assessing individual class sensitivities.

Compensating for Effect 3: The predictive values (PPV and NPV) are an important aspect of assessing the quality of a classification system. To showcase this importance, consider the following confusion matrix of a binary classification task, as shown in Tab. SN 2.6:

Table SN 2.6. Confusion matrix illustrating Effect 3.

		Prediction	
		Positive	Negative
Actual	Positive	TP = 10	FN = 1
	Negative	FP = 100	TN = 10,000

This system is assessed as 'near-perfect' by BA (0.95), 'better than random, but not really useful' by MCC (0.29), and 'much worse than random' by ECN (9.2).

This example shows that BA does not consider predictive values, thus yielding a near-perfect score despite a low PPV of 0.09. This assessment could be considered a pitfall in many scenarios, where the classification system would be fairly useless. Consider, for instance, a breast cancer screening program where, based on the provided system, > 90% of all biopsies (True Positives (TPs) + False Positives (FPs)) would be unnecessary (FPs).

In contrast, the MCC score could be considered intuitive for many scenarios such as the screening example. This is due to MCC explicitly considering all four basic rates True Positive Rate (TPR), True Negative Rate (TNR), PPV, and NPV. Thus, MCC poses further requirements compared to BA, which focuses only on Sensitivities. ECN also ensures high predictive values by design. In practice, however, it is not always a good indicator for predictive values because of the sometimes overly strict penalization of errors, as seen in the above example. In theory the weights in ECN could be adjusted to simulate the behavior of predictive value-sensitive metrics like MCC, but this implies a trial-and-error tuning process on each new task.

Identifying the optimal decision rule applied to predicted class scores: The different strategies for identifying a decision rule applied to predicted class scores are described in FP2.6 (see Suppl. Note 1.3). In the multi-class setting, argmax-based decision rules (i.e., predicting the class with the highest class score) are very common, but make arguably strong assumptions such as calibrated scores and equal penalization of all misclassifications.

It should be noted here that metrics that can be viewed as instantiations of EC (in this case BA and ECN) come with a theoretical framework on how to validate the decision rule, i.e., answering the question "how much better could the classification performance have been with an optimal decision rule?" [40]. MCC, on the other hand, lacks such a framework.

Interpretability: Arguably, BA features the most straightforward interpretation as the average over individual class Sensitivities, with bounded scores $[0, 1]$ and a fixed random reference at $1/C$. ECN scores are also fairly interpretable ("the EC of the system in relation to the EC of a naive system"), but scores are not bounded $[0, \infty)$. Furthermore, the random reference could be interpreted as 'too strict' for many scenarios such as the provided example. As for MCC, a random reference value is provided at 0 and the scores are bounded $[-1, 1]$, but all intermediate scores are arguably less intuitive. The general interpretation of MCC would be that it is a metric that depends on individual class Sensitivities and predictive values, i.e., a high MCC score guarantees all of these being high and a low MCC score indicates that at least one of them is low.

Popularity: BA is a widely used metric. MCC is fairly well-known but arguably not used as much. ECN is used prominently in the field of speaker verification but has not been introduced to the biomedical imaging or clinical community yet, although the statistical concepts it is based upon are long-standing in Bayesian decision theory.

2.7.2 Decision guide S3.

DG3.1: Metric@(TargetMetric = TargetValue)

If a target value for a specific metric (typically Sensitivity; Fig. SN 3.16) is provided, the decision rule applied to the predicted class scores is optimized such that the specific target value is reached on a validation data set (Suppl. Note 5.4). Other metrics, depending on the target application, can then be reported for that specific threshold. Example Specificity@(Sensitivity = 0.95): As illustrated in Fig. SN 2.9, the decision rule is set such that a Sensitivity of 0.95 is achieved. Other metric values (here Specificity; Fig. SN 3.17) can then be obtained from the corresponding fixed confusion matrix. In the example, this yields the Specificity at the predefined Sensitivity level. Possible candidates include Sensitivity (Fig. SN 3.16), Specificity (Fig. SN 3.17), PPV (Fig. SN 3.15), NPV, (Fig. SN 3.12) and False Positives per Image (FPPI) (Fig. SN 3.8).

DG3.2: Net Benefit (NB) versus Expected Cost (EC)

Summary of DG3.2: NB versus EC

NB

- ➔ Decisions can be defined directly based on predicted class scores, interpreted as risks
- ➔ Weighting of True Positive (TP) against False Positive (FP) in risk perspective
- ➔ Lack of framework to validate the decision rule applied to class scores
- ➕ Focus on reflectance of the (e.g., clinical) interest in the scores
- ➔ Popular metric in clinical studies but not common in image analysis

EC

- ➔ Decisions based on explicit definition of misclassification costs
- ➔ Weighting of False Positive (FP) against False Negative (FN) in cost perspective
- ➕ Availability of framework to validate the decision rule applied to class scores
- ➕ Inherent interpretability with respect to naive classifier
- ➔ Not known or used in the biomedical imaging domain although based on well-studied statistical concepts

Table SN 2.7. Comparison of Net Benefit (NB) and Expected Cost (EC) in the context of the decision guide DG3.2 for Subprocess S3. Context: FP2.6 = cost-benefit-based decision rule applied to predicted class scores requested.

This decision guide is embedded in the framework in Subprocess S3, which guides the selection of metrics that are reported separately for each class. In multi-class tasks (i.e., more than two classes present) this reporting amounts to a one-versus-rest validation scheme. However, this scheme is not intuitively applicable to a cost-benefit analysis (what are the costs and benefits of the 'rest' class?), which is the concept behind decision rules of both metrics in this decision guide. Thus, for

multi-class tasks we recommend to only proceed with the metrics selected in Subprocess S2 (e.g., EC or WCK) and not select any further metrics here to be reported in a one-versus-rest fashion, i.e., we recommend to skip the guide.

Both NB (Fig. SN 3.11) and EC (Fig. SN 3.6) are linked to cost-benefit analysis [81] and are well-suited when a cost-benefit-based approach for determining an appropriate decision rule applied to the predicted class scores is desired (FP2.6 = cost-benefit-based). To this end, both require the knowledge of task-dependent tradeoffs between benefits and costs, as detailed below. The following aspects are relevant when deciding between EC and NB (note that cost-based decision rule applied to predicted class scores is only considered for binary classification tasks in the scope of this work, thus referred to as a cutoff in this context):

Cost versus risk perspective: *Cost perspective:* For EC, explicit costs for both basic misclassifications (FP, FN) need to be defined or estimated. The optimal decision rule (i.e., cutoff on predicted class scores) that minimizes these costs can be analytically determined without data-based optimization. *Risk perspective:* In contrast, NB does not require the costs to be defined explicitly. Instead, predicted class scores are interpreted as probabilities or 'risks' of certain model output scores belonging to the positive class and the cutoff on the scores is defined directly on this scale based on task interest (e.g., "only treat patients with cancer risk >10%"). This can be interpreted as an implicit cost-benefit analysis resulting in a single intuitive risk score. However, it is also common for NB to make this cost-benefit analysis more explicit and define the risk as a relation of the benefit of TPs to the harms caused by FPs. A diagnostic test, for example, may lead to early identification and treatment of a disease, but typically the process will also cause some patients without disease being subjected to unnecessary further interventions. NB allows to consider such tradeoffs by putting the benefits and harms of the test on the same scale so that they can be directly compared. A physician may, for example, state that 10 FPs, resulting in unnecessary biopsies, are acceptable to find one more cancer case (TP).

Decision curves: In most scenarios it is not possible to precisely define the costs or risks associated with the task. For example, it is not straightforward to make an exact decision on how many FPs would be acceptable to obtain one more TP. To compensate for this uncertainty, it is common practice to plot NB over a "reasonable range of risk thresholds" resulting in so-called decision curves [111]. This analysis allows assessing and comparing methods according to their NB scores without relying on a single cutoff. Although not common practice, one could also generate such curves for EC when expressing cost ratios as a risk score (i.e., switching from the cost to the risk perspective).

Cutoff on predicted class scores: In NB, the cost-benefit-based cutoff, which is determined directly from provided knowledge about the task and does not require data-based optimization, is an explicit part of the metric computation. In contrast, EC allows to alternatively determine a data-based cutoff by taking into account the provided costs in the metric calculation and minimizing EC on a dedicated data split, if available. A further difference between the two metrics is the way prevalence dependency is handled: EC isolates the class priors from the predicted class scores and defines them as a parameter of the cutoff itself, such that all application dependent parameters (costs and class priors) are part of the cutoff [40]. Upon deployment of a model on a new data set, the threshold can simply be updated analytically. Note that this process only works under the arguably strong assumption that the class priors of the new data set are known. In contrast, NB considers risk scores that incorporate the class priors, implying that the threshold depends solely on the cost-benefit tradeoff. As a consequence, when the class priors shift on a new data set, the risk-cutoff in NB requires class scores to be re-calibrated. The latter might be a harder requirement

because it requires a labeled validation set for re-calibration as opposed to requiring merely the class priors of the new data set for a threshold update.

Interpretability: The following tradeoff exists between the two metrics regarding interpretability: EC allows reporting a normalized version (ECN), which makes the metric scores interpretable with regard to the performance of a random classifier. In contrast, in NB, the reference to a random classifier is typically done manually (by comparing the two scores), because NB itself allows for an interpretation as the 'proportion of net-TP', which would get lost by normalization.

Calibration: Both metrics rely on the fact that predicted class scores are well-calibrated with regard to a chosen cutoff. EC allows assessing this requirement by calculating the extra cost entailed by miscalibration (or the potential for reducing cost by calibrating scores) [40]. The calibration error here is measured as the increase of EC with the analytical, i.e., task interest-based, cutoff compared to an empirical cutoff optimized on the data. Compared to related calibration errors (see Suppl. Note 2.6), this technique assesses a weaker calibration condition, which is directly targeted to the decision process at hand. For instance, even when assessing the relatively weak top-label calibration condition by means of Expected Calibration Error (ECE) with two bins and the border at the determined cutoff value, the distribution inside the bins would be considered, while EC only focuses on how many more cases would have been on the 'correct side of the cutoff' if scores were calibrated, without considering score distributions on either side of the cutoff.

Popularity: Neither NB nor EC are widely used in the biomedical image analysis community. NB is a popular metric in clinical studies, while EC is currently not used but is part of a coherent framework of intuitive metric formulations (linked to Accuracy, BA, and extends to multi-class scenarios).

DG3.3: Positive Likelihood Ratio (LR+) versus Sensitivity

Summary of DG3.3: LR+ versus Sensitivity

LR+

- ➕ Straightforward application in the case of an optimization-based decision rule (FP2.6)
- ➕ Interpretation often reflecting interest in binary tasks

Sensitivity

- ➖ Challenging application in the case of an optimization-based decision rule (FP2.6)
- ➕ Good interpretability

Table SN 2.8. Comparison of LR+ and Sensitivity in the context of the decision guide DG3.3 for Subprocess S3. Context: FP2.6 = optimization- or argmax-based decision rule applied to predicted class scores requested and provided class prevalences do not reflect the population of interest (FP4.2 = FALSE).

This decision guide helps deciding between LR+ and Sensitivity in the context of per-class validation (Subprocess S3) with an optimization- or argmax-based decision rule applied to predicted class scores (FP2.6).

LR+ (Fig. SN 3.14) is the likelihood ratio of the positive class. In a clinical example where the quality of a diagnostic test is to be assessed, this could be interpreted as the ratio of the likelihood of a diseased patient receiving a positive test result versus a healthy patient receiving a positive

test result $(P(t + |d+)/P(t + |d-))$, where t/d denotes a positive(+)/negative(-) test/disease status). In other words: How much more likely is the occurrence of a positive test result for a diseased person compared to a healthy person? The formal calculation for this metric boils down to the following formula: $LR+ = TPR / (1-TNR)$, where TPR/TNR are the Sensitivities of the positive/negative class.

In the provided context of this decision guide, where metrics are reported individually per class, Sensitivity (Fig. SN 3.16) and LR+ convey similar information and there is no 'incorrect' choice. Thus, the choice between the two can generally be made as the metric that is easier to interpret in the given task: In binary classification tasks (e.g., the provided example), LR+ conveys Sensitivities of both classes in a single score. Due to its intuitive and meaningful interpretation, it is often reported in clinical studies. In multi-class settings (which, in this context, amount to a one-versus-rest validation scheme), Sensitivities are generally easier to interpret, while the direct interpretation of LR+ as a property of a (clinical) test does not apply.

In case the decision rule applied to predicted class scores is to be determined on the basis of optimization on the target class, one additional consideration is of importance (FP2.6 = optimization-based decision rule). When reporting Sensitivity per class, the decision rule can not be optimized based solely on the single Sensitivity at hand because this would always yield a cutoff value of 1. LR+ naturally overcomes this problem. Other possible workarounds include choosing a different decision rule (FP2.6) or optimizing a weighted average over Sensitivity for all classes instead. The latter option should only be considered if meaningful weights across classes can be defined (e.g., based on class importance).

DG3.4: Positive Likelihood Ratio (LR+) versus Sensitivity versus F_β Score

Summary of DG3.4: LR+ versus Sensitivity versus F_β Score

LR+	Sensitivity	F_β Score
<ul style="list-style-type: none"> + Meaningful interpretation in binary tasks + Inherent interpretability with respect to naive classifier - Insensitive to PPV 	<ul style="list-style-type: none"> + Generally good interpretability + Inherent interpretability with respect to naive classifier only when averaging over classes - Insensitive to PPV 	<ul style="list-style-type: none"> ➔ Limited interpretability - No interpretability with respect to naive classifier + High scores ensures high PPV

Table SN 2.9. Comparison of LR+, Sensitivity and F_β Score in the context of the decision guide DG3.4 for Subprocess S3. Context: FP2.6 = optimization- or argmax-based decision rule applied to predicted class scores requested and provided class prevalences reflecting the population of interest (FP4.2 = TRUE).

In the context of this decision guide, prevalence dependency is not an exclusion criterion (see FP4.2) and thus F_β Score (Fig. SN 3.7) can be considered as an alternative to Sensitivity-based

metrics (Sensitivity and LR+, Figs. [SN 3.16](#) and [SN 3.14](#)). Details for the decision between the latter are provided in DG3.3; the present guide focuses on the pros and cons of opting for F_β Score.

Per-class validation is commonly performed in a one-versus-rest fashion, naturally introducing class imbalance into the validation. Exceptions are binary scenarios with two balanced classes. For this exception, no compensation for class imbalance is needed (FP2.5.5 = FALSE) and the choice between F_β Score and Sensitivity-based metrics becomes less relevant, i.e., there are no obvious incorrect choices. Thus, the decision can be made on the basis of which metric is easier to interpret in a given task. For all other cases, the decision should be based on whether compensation for class imbalance is required (FP2.5.5 = TRUE).

Compensation for class imbalance: As described in FP2.5.5 (and explained in more detail in Suppl. Note [1.3](#), "Compensation for class imbalance"), there are three aspects of compensation for class imbalance:

- (1) **Establishing a reference value for random performance:** LR+ provides a fixed random reference value at $LR+ = 1$, while for Sensitivity the scores of individual classes can vary and only their average is fixed at "1/number of classes" (equivalent to BA). F_β Score does not provide a reference value for random performance.
- (2) **Establishing equal class contribution:** In the provided context (S3), the validation is performed per class, such that this aspect is irrelevant.
- (3) **Establishing consideration of predictive values:** This aspect is the main reason to opt for F_β Score in this decision guide, because it is the only metric of the three where high scores ensure a high PPV. In contrast, LR+ and Sensitivity are insensitive to PPV, which, depending on the task interest, can substantially diminish their utility. An exemplary pitfall related to this choice is the confusion matrix of a binary classification task, as shown in Tab. [SN 2.6](#). This classification system yields (for the two individual classes) Sensitivities of (90%, 99%) and LR+ of (90, 90), respectively. Resulting F_1 Scores are (0.165, 0.995), indicating a low PPV and thus unsatisfying performance for the rare positive class. This pitfall may be of practical relevance in class-imbalanced tasks where FPs shall not be neglected. For example, in breast cancer screening, the provided classifier would not be useful, because > 90% of all biopsies (TP+FP) would be unnecessary (FP).

Interpretability: Out of the three, Sensitivity is arguably the easiest-to-interpret metric (exceptions are binary tasks, where LR+ might be preferable as detailed in DG3.3, see Tab. [SN 2.8](#)). F_β Score can be interpreted as the harmonic mean of Sensitivity and PPV, which adds a layer of complexity to the interpretation compared to Sensitivity. Thus, if the aspects discussed in "compensation for class imbalance" are not relevant, F_β Score might not be the metric of choice.

DG3.5: How to determine β in F_β Score

Summary of DG3.5: β in F_β Score

$\beta < 1$

➔ Higher weighting of False Positive (FP) penalties (Positive Predictive Value (PPV))

$\beta = 1$

➔ Harmonic mean of PPV and Sensitivity

$\beta > 1$

➔ Higher weighting of False Negative (FN) penalties (Sensitivity)

Table SN 2.10. Determining the hyperparameter of the F_β Score in the context of the decision guide DG3.5 for Subprocess S3. Context: [Image-level classification (ImLC)]: FP2.6 = optimization- or argmax-based decision rule applied to predicted class scores requested and provided class prevalences reflecting the population of interest (FP4.2 = TRUE). [Object detection (ObD) or instance segmentation (InS)]: Either no predicted class scores available (FP5.1 = FALSE) or FP2.6 = optimization- or argmax-based decision rule applied to predicted class scores requested.

The F_β Score (Fig. SN 3.7) is defined as:

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{PPV} \cdot \text{Sensitivity}}{(\beta^2 \cdot \text{PPV}) + \text{Sensitivity}} = \frac{(1 + \beta^2) \cdot \text{TP}}{(1 + \beta^2) \cdot \text{TP} + \beta^2 \cdot \text{FN} + \text{FP}} \quad (3)$$

The most common choice is to set β to 1, resulting in equal weighting of FP and FN penalties. If unequal penalization of class confusions is desired (see FP2.5.2), higher values of β result in higher weights on FN penalties compared to FP penalties and thus imply a focus on Sensitivity compared to PPV.

DG3.6: F_β Score versus Panoptic Quality (PQ)

Summary of DG3.6: F_β Score versus PQ

F_β Score

➔ Pure detection metric

PQ

➔ Hybrid metric for assessing detection and segmentation quality

Table SN 2.11. Comparison of F_β Score and Panoptic Quality (PQ) in the context of the decision guide DG3.6 for Subprocess S3. Context: FP2.6 = optimization- or argmax-based decision rule applied to predicted class scores and FP1.1 = instance segmentation (InS).

The F_β Score (Fig. SN 3.7) is a pure detection metric counting TP, FP, and FN detections on instance level (specifically, it represents the harmonic mean of PPV (Fig. SN 3.15) and Sensitivity (Fig. SN 3.16; see also [85]). The “segmentation aspect” of instance segmentation is here only incorporated via a prior cutoff on the localization criterion operating on pixel level (e.g., “Intersection over Union (IoU) > 0.5”). In case shifting the focus of validation more towards the segmentation quality of successfully matched (TP) instances is desired, there are two options:

- (1) **Complementary segmentation metric:** One option is to select separate segmentation metrics in addition to object detection metrics such as F_β Score on a per-instance basis (e.g. “Dice Similarity Coefficient (DSC) per TP-instance”). This selection is naturally incorporated in the instance segmentation recommendation (Fig. 2) by the subroutines S6 (Extended Data Fig. 6) and S7 (Extended Data Fig. 7).
- (2) **Hybrid metric:** An alternative is to select PQ (Fig. SN 3.13) instead of F_β Score, which allows expressing both interests (detection performance and segmentation quality) in a single score. Essentially, PQ is a modified F_1 Score, where TP instances do not count as “1” in the calculation, but the “1” is replaced with the associated DSC score (range [0,1]) of the instance. While combining the two aspects in a single score might be desirable, e.g., for method benchmarking or ranking, on the downside, such combined metrics make it harder to trace back performance to individual aspects (in this case: object detection versus segmentation; see Fig. SN 2.17).

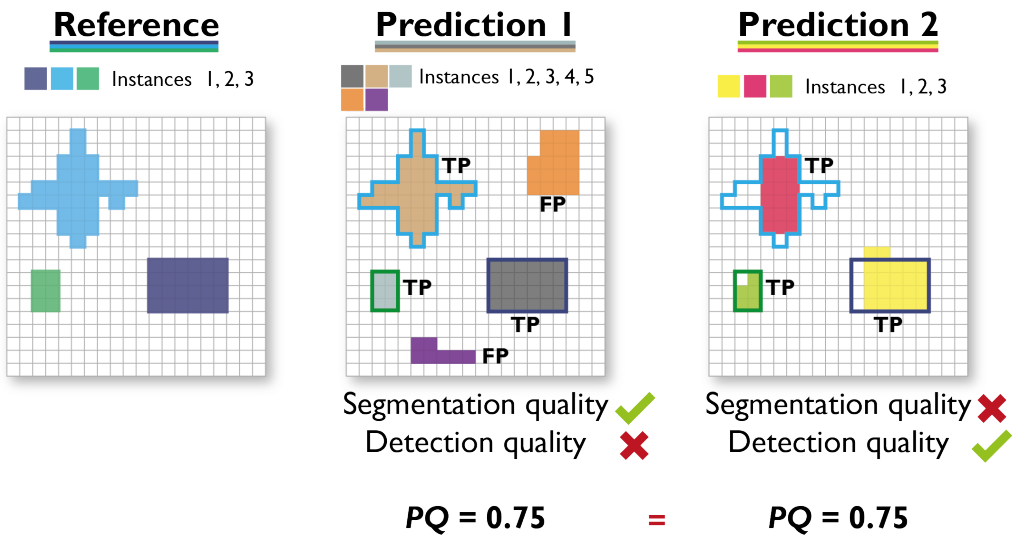


Fig. SN 2.17. Effect of assessing segmentation and detection quality in a single score. *Prediction 1* achieves a high segmentation but low detection quality (with several False Positive (FP) predictions); vice versa for *Prediction 2* (only predicting True Positive (TP) instances; no FP but low segmentation quality). However, both yield the same Panoptic Quality (PQ) score.

2.7.3 Decision guide S4.

DG4.1: Area under the Receiver Operating Characteristic Curve (AUROC) versus Average Precision (AP)

Summary of DG4.1: AUROC versus AP

AUROC

- ➔ Insensitive to Positive Predictive Value (PPV) under class imbalance
- ➕ Inherent interpretability with respect to naive classifier
- ➕ Straightforward interpretability

AP

- ➕ High scores ensure high PPV including under class imbalance
- ➔ Prevalence-dependent reference value for naive classifier
- ➔ Limited interpretability

Table SN 2.12. Comparison of Area under the Receiver Operating Characteristic Curve (AUROC) and Average Precision (AP) in the context of the decision guide DG4.1 for Subprocess S4. Context: availability of predicted class scores (FP5.1 = TRUE), FP1.1 = image-level classification (ImLC) and provided class prevalences reflecting the population of interest.

The comparison between the two concepts behind AUROC and AP, i.e., the comparison between Receiver Operating Characteristic (ROC) curves and Precision-Recall (PR) curves has been extensively studied [33]. In practice, the choice between the two metrics boils down to the following aspects (if no clear choice can be made, we recommend reporting both metrics):

Compensation for class imbalance effects: Of relevance in the context of this decision guide is pitfall 3 from FP2.5.5: "Missing consideration of predictive values" (more on this topic can be found in Suppl. Note 1.3, "Compensation for class imbalance"). AUROC is based on the Sensitivities of the two classes and does not consider predictive values. In class-imbalanced scenarios, this may lead to pitfalls such as depicted in Fig. SN 2.18, where near-perfect AUROC scores hide the fact that a system might have limited to no predictive utility. AP assesses the predictive value of the positive class (PPV) and thus compensates for the undesired effects caused by class imbalance: In the provided example, AP yields an intuitive score of 0.32, reflecting the low PPV and thus low utility of the system in the context of the task. A technical explanation is given by the fact that the high number of True Negatives (TNs) dominate and suppress the FPs in the calculation of the TNR, thus yielding high scores for AUROC. A practical example for this pitfall might be a breast screening program, where a high PPV is of great importance to prevent unnecessary biopsies (FPs). The focus of AP on the positive class further has the effect that the resulting scores differ depending on which of the two classes is defined as positive and negative. This is in contrast to AUROC, which yields the same scores irrespective of this definition. The general approach for AP-based assessment in class-imbalanced scenarios is to define the rare class as the positive class. The fact that AP focuses on the positive class reflects the task interest of not letting rare (important) events be dominated by frequent events in the metric score.

Interpretability: AUROC is easy to interpret as it simply represents the probability of a randomly sampled positive case having a higher predicted class score than a randomly sampled negative case. It further comes with a fixed reference value for the performance of a random classifier at 0.5. AP, on the other hand, is harder to interpret and features no fixed random reference value. Instead, the AP score of a random classifier is the prevalence of the positive class which varies on each data set.

Implementations: For reasons described in [33], the PR curve is more complex to interpolate compared to the ROC curve. This results in the existence of various implementations of AP, whereas no such heterogeneities exist for AUROC.

Popularity: Although AUROC is the common choice for multi-threshold metrics, AP is also widely known and used.

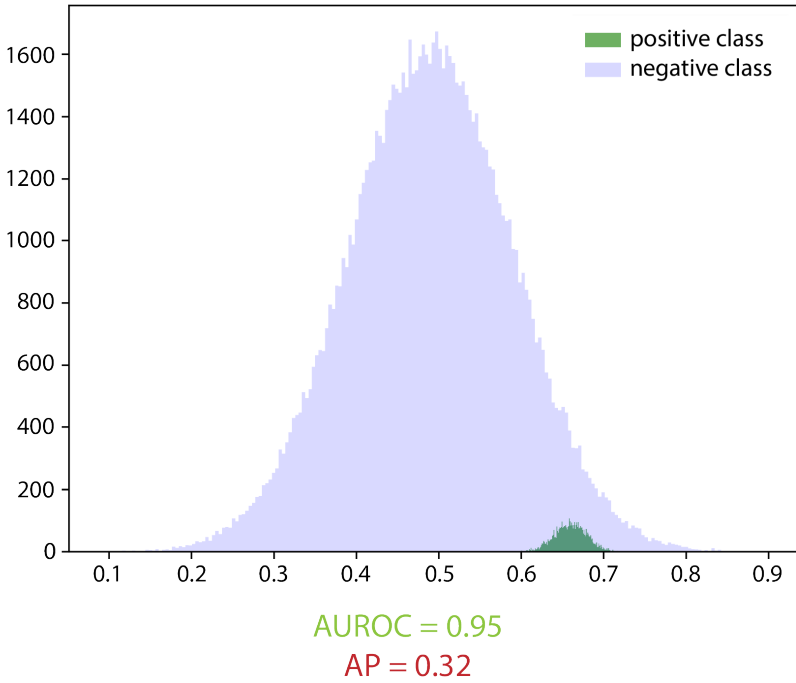


Fig. SN 2.18. Area under the Receiver Operating Characteristic Curve (AUROC) scores neglect the Positive Predictive Value (PPV) in class-imbalanced settings and might lead to misinterpretation of a model's discrimination quality. The figure shows the simulation outcome for a binary classification problem with a low prevalence for the positive class. A clinical example of this scenario are cancer screening programs, where most of the subjects are healthy. While AUROC is agnostic to the class prevalence and thus implies near-perfect discrimination with a score of 0.95, the prevalence-dependent Average Precision (AP) allows focusing on discrimination of the rare positive class by explicitly considering the PPV and yields an intuitive score of 0.32.

DG4.2: Average Precision (AP) versus Free-Response Receiver Operating Characteristic (FROC) Score

Summary of DG4.2: AP versus FROC Score

AP

- ➔ Standard metric in computer vision community
- ➔ Unawareness of data set sizes
- ➖ For filtering low confidence predictions, a cutoff on confidence scores is required
- ➕ Relatively good standardization of hyperparameters

FROC Score

- ➔ Preference in clinical context due to its domain-centered approach
- ➔ Consideration of data set sizes
- ➕ No consideration of low-confidence predictions
- ➖ Lack of standardization

Table SN 2.13. Comparison of Average Precision (AP) and Free-Response Receiver Operating Characteristic (FROC) Score in the context of the decision guide DG4.2 for Subprocess S4. Context: availability of predicted class scores (FP5.1 = TRUE) and FP1.1 = object detection (ObD) or instance segmentation (InS).

The following aspects should be taken into account when deciding between AP (Fig. SN 3.20) and FROC Score (Fig. SN 3.21):

- **Community preferences:** While AP constitutes the undisputed standard metric for object detection and instance segmentation in the computer vision community, the FROC Score is often favoured in the clinical context due to its easier interpretability despite its lack of standardization (employed FPPI Scores vary across studies [10, 54, 92]). Thus, the decision between the two metrics often boils down to a decision between a standardized and technical validation versus an interpretable and application-focused validation.
- **Data set size awareness:** In its default configuration, where AP is computed globally over the entire data set, the metric is insensitive to performance per individual images. In contrast, the FROC Score takes into account the total number of images in the data set (see also Fig. SN 2.19). For example, given a data set and an AP as well as FROC score computed for this data set, adding more 'empty' images (i.e., images with no reference objects and no model predictions) would lead to an improved FROC score, because FROC rewards the model for correctly not predicting structures on these images. In contrast, the AP score would not be affected, because globally no new TPs, FPs or FNs are added that would alter the metric score. This property does not affect relative method comparison and can be related to the underlying question "at which scale are matched objects (cardinalities) aggregated/counted?". As described in Suppl. Note 2.4, AP can alternatively be configured to aggregate scores per individual image, in which case the total number of images in the data set is considered equally to FROC. [85] demonstrates how to apply AP and other object detection metrics to, for example, clinical scenarios requiring per-image aggregation. FROC score is a hybrid metric in this context, where *Sensitivity* is computed per data set while FPs are averaged over single images (FPPI).

- Dealing with low-confidence predictions:** It is often desired to filter low confidence predictions (e.g., objects with high confidence of being background) prior to metric computation. For AP computation, this requires a cutoff on the confidence score or upper limits of considered predictions per image or per data set. For FROC, however, with typical values of FPPI, such low-confidence predictions naturally go unconsidered, thus allowing to avoid additional filtering measures.
- FPPI:** Different FPPI values are used in the field for computing the FROC Score, yielding non-standardized results (see Fig. SN 2.20). A potential default are the values 1/8, 1/4, 1/2, 1, 2, 4, 8, as used for multiple popular benchmarks [92, 108]. Here, lower FPPI values (smaller than one) are weighted equally to higher FPPI values (greater than 1; four values each). Deviation from this weighting might be appropriate depending on the application, but should be explained. In the biostatistics community, areas under the curve are sometimes computed constraining the FPPI range to [0,1] [88].

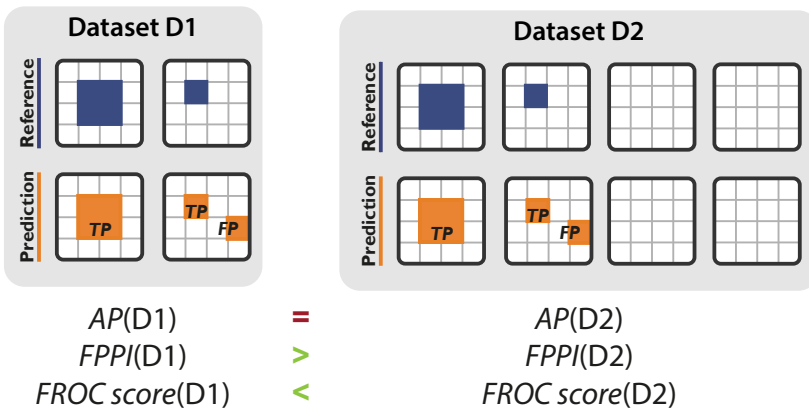


Fig. SN 2.19. Effect of the number of images per data set on the metric scores. The Average Precision (AP) metric does not take into account the total number of images, yielding the same score for data sets D1 and D2. The Free-Response Receiver Operating Characteristic (FROC) curve plots the average number of False Positives per Image (FPPI) against the Sensitivity, therefore accounting for the number of images. The FPPI is lower for D2, yielding a higher FROC score.

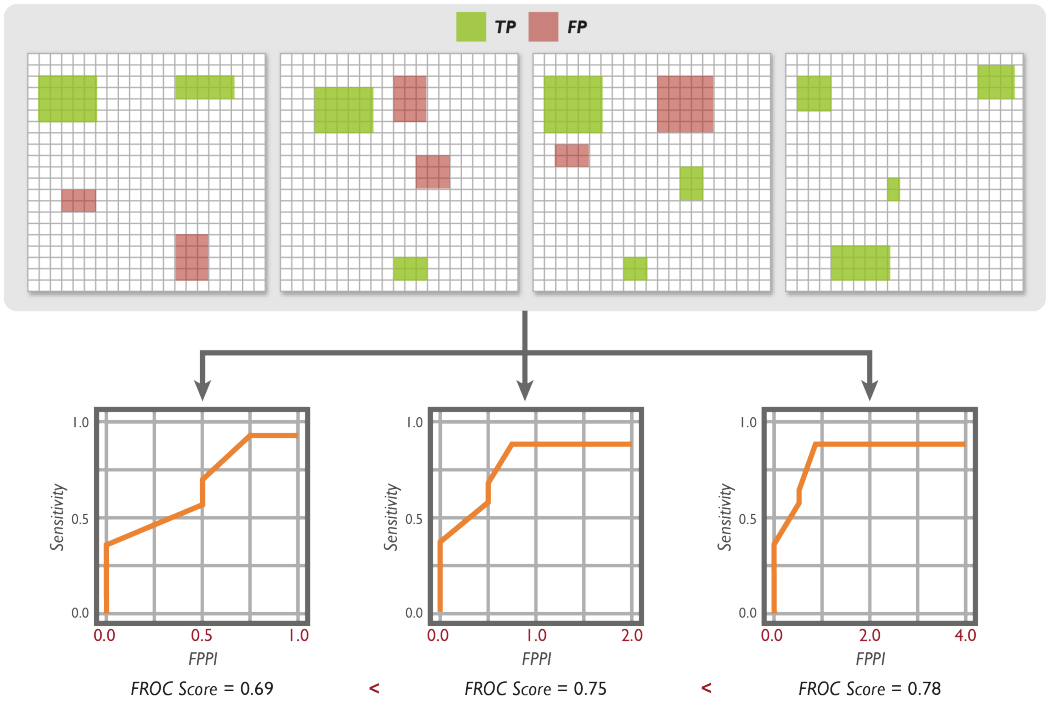


Fig. SN 2.20. Effect of defining different ranges for the False Positives per Image (FPPI) used to draw the Free-Response Receiver Operating Characteristic (FROC) curve for the same prediction (top). The resulting FROC Scores change for different boundaries of the x-axis.

2.7.4 Decision guide S5.

DG5.1: Kernel Calibration Error (KCE) versus Expected Calibration Error Kernel Density Estimate (ECE^{KDE})Summary of DG5.1: KCE versus ECE^{KDE}**KCE**

- ⊕ Capture of isolated calibration quality
- ⊖ Unbiased estimator of canonical calibration error based on an alternative distance function
- ⊖ Bad interpretability, also due to negative output values
- ⊖ Recent proposition, not widely used
- ⊖ Depends on nontrivial configuration choices of kernels and associated hyperparameters

ECE^{KDE}

- ⊕ Capture of isolated calibration quality
- ⊖ Potentially biased estimator of an ℓ_p canonical calibration error (bias might be rendered neglectable by future de-biasing schemes)
- ⊕ Straightforward interpretability of relative improvement
- ⊖ Recent proposition, not widely used

Table SN 2.14. Comparison of Kernel Calibration Error (KCE) and Expected Calibration Error Kernel Density Estimate (ECE^{KDE}) in the context of the decision guide DG5.1 for Subprocess S5. Context: FP2.7.2 = U2 - comparison of calibration performance across classifiers on the same task requested and no mismatch between class prevalences and class importance (F2.5.3 = FALSE).

The context for this decision guide between KCE (Fig. SN 3.32) and ECE^{KDE} (Fig. SN 3.31) is use case 2 (U2) in Fig. SN 2.15: "comparing the calibration quality across multiple classifiers on the same task."

General differences: Both KCE and ECE^{KDE} are estimators of a canonical calibration error, but measure this error based on different divergences, i.e., distance functions: ECE^{KDE} is based on the ℓ_p norm and thus straightforward to interpret and configure. In contrast, KCE is based on the "maximum mean discrepancy" and thus not interpretable (it may even take on negative values) and requires nontrivial configuration of kernels as well as associated hyperparameters. On the other hand, ℓ_p norm estimators such as ECE^{KDE} are inherently biased while KCE is an unbiased estimator. Arguably, in the context of this decision guide (U2), interpretability of the calibration error estimate is not required, since only a comparative, or relative assessment is requested rendering the unbiased KCE the intuitive choice. However, recent research on ℓ_p norm estimators presents effective de-biasing schemes [83], which might render the resulting bias neglectable in the near future and thus make ℓ_p estimators such as ECE^{KDE} a viable alternative for comparative calibration assessment.

Popularity: Calibration error estimates KCE and ECE^{KDE} are both recently proposed measures that are not widely known in the biomedical community.

DG5.2: Brier Score (BS) versus Kernel Calibration Error (KCE) versus Expected Calibration Error Kernel Density Estimate (ECE^{KDE})

Summary of DG5.2: BS versus KCE versus ECE^{KDE}

BS

➔ Capture of effects of (re-) calibration methods on discrimination performance in addition to calibration quality

➕ Unbiased measure of an ℓ_2 norm canonical calibration error

➕ Straightforward interpretability of relative improvement

➕ Established statistical concept with long history of applications in many fields of research

KCE

➔ Capture of isolated calibration quality

➔ Unbiased estimator of canonical calibration error based on an alternative distance function

➖ Bad interpretability, also due to negative output values

➔ Recent proposition, not widely used

➖ Depends on non-trivial configuration choices of kernels and associated hyper-parameters

ECE^{KDE}

➔ Capture of isolated calibration quality

➔ Potentially biased estimator of an ℓ_p canonical calibration error (bias might be rendered neglectable by future de-biasing schemes)

➕ Straightforward interpretability of relative improvement

➔ Recent proposition, not widely used

Table SN 2.15. Comparison of Brier Score (BS), Kernel Calibration Error (KCE) and Expected Calibration Error Kernel Density Estimate (ECE^{KDE}) in the context of the decision guide DG5.2 for Subprocess S5. Context: U1 - FP2.7.2 = comparison of re-calibration methods for the same classifier requested.

The context for this decision guide between BS (Fig. SN 3.28), ECE^{KDE} (Fig. SN 3.31), and KCE (Fig. SN 3.32) is use case 1 (U1) in Fig. SN 2.15: "comparing the effect of one or more re-calibration methods on the same (fixed) classifier."

General differences: BS can be decomposed into discrimination and calibration terms, where the calibration term exactly resembles the canonical calibration error (as defined in Suppl. Note 2.6). As the purpose of the metric in the provided context is to assess the performance of different re-calibration methods for the same classifier, a higher BS score also implies a better calibration in the case of *accuracy-preserving* calibration methods. As a major difference to BS, KCE estimates the

canonical calibration error directly. While this estimation is not biased (i.e., it is not dependent on the data set size), the resulting estimates are not interpretable, that is, they only allow for relative comparison on the same task (equivalently to BS). Further, KCE requires nontrivial configuration of kernels as well as associated hyperparameters. In contrast to KCE, current estimators of ℓ_p calibration error are biased, but are highly interpretable and straightforward to configure. Moreover, recent developments in this line of research present effective de-biasing schemes [83], which might render the resulting bias neglectable in the near future and thus make ℓ_p estimators such as ECE^{KDE} a viable alternative also for comparative calibration assessment.

Applicability: Generally, BS is attractive for ranking re-calibration methods that are guaranteed to be accuracy-preserving (such as the common temperature scaling [44]). Otherwise, the metric must be applied with care, because altered discrimination performance will dilute the focus on calibration quality in the ranking. Note that it may also be desirable to capture the effect of (non-accuracy-preserving) re-calibration methods on the discrimination performance. In such cases of comprehensive assessment of re-calibration methods, it is also appropriate to apply BS. In contrast to BS, calibration error estimators such as KCE and ECE^{KDE} are capable of comparing the calibration error of re-calibration while being agnostic to potential changes of discrimination performance caused by the transformations. For the provided use case, this property allows the ranking of non-accuracy-preserving transformations, such as recently proposed techniques employing spline interpolations [45] or Gaussian processes [114], purely according to their calibration error while ignoring their effects on the discrimination performance.

Interpretability: Defined as the root mean square error between predictions and references, BS is bounded between [0, 1] and therefore straightforward to interpret as an overall measure. However, as the calibration error is not isolated and scores are still conflated with the (same fixed) discrimination performance, only a relative comparison of calibration errors is possible. KCE is generally hard to interpret, also because it can yield negative values. ECE^{KDE} as an estimator of ℓ_p calibration error is straightforward to interpret.

Popularity: BS is a widely known metric for overall performance measures with a long history of usage. Calibration error estimates KCE and ECE^{KDE} are both recently proposed measures and not widely known in the biomedical community.

Reasons to not recommend Negative Log Likelihood (NLL) in this context: NLL essentially assesses a weighted version of the canonical calibration error as the logarithm leads to heavy penalization of tail probabilities. As the implications of this weighting on calibration assessment (as opposed to the overall performance measure) are not intuitive, we generally do not recommend NLL in this use case.

DG5.3: Brier Score (BS) versus Negative Log Likelihood (NLL)

Summary of DG5.3: BS versus NLL

BS

➖ Bounded penalization of errors leads to preference of naive systems in imbalanced settings

➕ Straightforward interpretability as the mean squared error

➕ Established statistical concept with long history of applications in many fields of research

NLL

➔ Heavy penalization of extreme scores (close to 0 or 1), thus ability to capture missing rare events. General preference of conservative models

➔ Difficult interpretability due to lack of upper bound

➕ Established statistical concept with long history of applications in many fields of research

Table SN 2.16. Comparison of Brier Score (BS) and Negative Log Likelihood (NLL) in the context of the decision guide DG5.3 for Subprocess S5. Context: FP2.7.2 = U3 - comparison of overall performance across classifiers requested.

The context for this decision guide between BS and NLL is use case 3 (U3) in Fig. SN 2.15: "overall performance measure requested." Both BS (Fig. SN 3.28) and NLL (Fig. SN 3.33) are overall performance measures, which capture discrimination and canonical calibration in a single score.

Penalization of errors: Like Accuracy, BS penalizes errors of all events equivalently irrespective of the class prevalence. This implies that scores may drastically change when the prevalence changes and thus renders BS a highly prevalence-dependent metric. For instance, in imbalanced scenarios, a naive system that simply predicts the dominant class can receive a high BS, similarly to a high Accuracy score. One strategy to cope with this is to divide the BS by the BS achieved with a naive system, resulting in the normalized variant Brier Skill Score (BSS). Equivalently to ECN, this transformation is a rescaling of scores to establish a 'naive baseline' and enhance interpretability, but errors are still penalized equivalently irrespective of class prevalence. In other words, equal importance of classes (FP2.5.1) is not reflected in the metric, and missing a frequent event is still as heavily penalized as missing a rare event although missing a rare event has a greater effect on the respective class sensitivity. This results in a strict interpretation where the total amount of errors has to be lower than the number of events in the rare class in order for a system to be considered 'better than random'.

Compared to squared error penalization in BS, the logarithm introduces a stronger penalization of tail probabilities [84]. In consequence, overconfident predictions (such as a score of 1, implying scores of 0 in the other classes) lead to higher losses. For example, predicting 0.001 rather than 0.01 (when the true class is '1') increases BS by $\approx 2\%$ and NLL by $\approx 230\%$ (for this single entry). A practical effect of this penalty is a naturally higher penalization of naive systems in class imbalance scenarios, addressing the pitfall of BS above. NLL is thus of potential interest in scenarios with high class imbalance, where missing rare events would be heavily penalized, compared to BS which is prone to favoring naive systems. Generally, the penalization effect can also be described as NLL favoring more conservative models that avoid predictions of extreme class scores.

Interpretability: BS is relatively straightforward to interpret as the mean squared error between predictions and the reference. The resulting scores are bounded ($[0, 1]$). NLL is arguably harder to interpret featuring logarithmic penalization of errors and thus no upper bound of the resulting score (bounds: $[0, \infty]$)

Popularity: Both metrics are common statistical concepts and come with a long history of usage in many fields of research.

DG5.4: Expected Calibration Error (ECE)/ Root Brier Score (RBS) versus Class-wise Calibration Error (CWCE) versus Expected Calibration Error Kernel Density Estimate (ECE^{KDE})/ Class-wise Calibration Error (CWCE)/ Root Brier Score (RBS)

The decision between the sets of metrics boils down to determining whether predicted class scores should be tested for top-label calibration (as measured by ECE, Fig. SN 3.30), marginal calibration (as measured by CWCE, Fig. SN 3.29), or canonical calibration (as measured by ECE^{KDE}, Fig. SN 3.31. If there is an unequal interest across classes (FP2.5.1), CWCE is the natural choice. In this case we recommend both per-class and weighted reporting (by class importance). Note that only aggregated reporting comes with the pitfall of unstable results, specifically in the case of few samples or many classes. In the case of equal interest across classes, the key question is whether the task interest is limited to the predicted scores that lead to the classification decision (top-label) or whether there are reasons to request all predicted scores to be calibrated.

Notably, in binary classification tasks, the two conditions are equivalent [105].

Reasons for and against focusing on top-label calibration (ECE): The task interest focuses on the decisions made by the classifier and only lies in the probabilities of the resulting decisions. In case the underlying biomedical research question has a dedicated focus on the decision process, top-label error might be the right choice, because it directly reflects this focus. Conflating the calibration of decisions with other probabilities might be interpreted as washing out the task focus in this case. Although it is common practice to assess calibration quality with ECE, this approach comes with various pitfalls. Importantly, it is often ignored that top-label calibration implies an argmax decision rule based on the predicted class scores, which is often not an optimal decision rule as discussed in Suppl. Note 1.1 ('decision rule on predicted class scores'; see Fig. SN 1.1). Caution should also be exercised if there is a mismatch between class prevalences and class importance (FP2.5.3) as the top-label calibration is highly biased towards the high-prevalence classes. Furthermore, ECE commonly relies on binning of class scores, which introduces a dependency of the resulting metric score on the specific binning scheme. The number of bins is a configuration parameter that should by no means be optimized on the final validation data. Note in this context that binning has been shown to result in a more biased estimation compared to density estimation methods [83].

Reasons to extend the focus to all predicted scores (ECE^{KDE} and CWCE): A common perception is that the canonical calibration condition, which is the strongest condition considering all predicted class scores, is the appropriate one in many application scenarios [40, 43, 83]. One reason lies in the limitations of top-label calibration and associated binning estimators described above. Another reason could be a broad task interest in all predictions beyond the classification decision. In the clinical context, for instance, the risk for all potential outcomes might be relevant for further treatment or shall be communicated to the patient. In such scenarios, calibration of all probabilities might be of interest. Consider, for instance, a multi-way classification of tumor categories, where one category is more aggressive than others. Even though the final prediction of the system is 'benign lesion', it might be of clinical interest to know (and communicate to the

patient) whether the probability for this outcome was 5% or 20%. While the primary calibration metric for such scenarios should be ECE^{KDE} as an estimate of the canonical calibration, it might be of interest to additionally report marginal calibration (as measured by CWCE) separately for each class. Notably, for these scenarios, alternatively splitting the problem into individual domain questions that result in separate traversals for each class of interest should be considered (see Suppl. Note 1.1).

Additional reporting of RBS (Fig. SN 3.34) as a guaranteed upper bound on the calibration error: In top-label and canonical calibration, we recommend the additional reporting of RBS as a guaranteed upper bound on the calibration error. As popular methods to assess calibration quality such as ECE or ECE^{KDE} are known to over- or underestimate the error [43], this guarantee provides additional information, especially in safety-critical applications where the calibration error must not be underestimated.

2.7.5 Decision guide S6.

DG6.1: Dice Similarity Coefficient (DSC) versus Intersection over Union (IoU)

Summary of DG6.1: DSC versus IoU

DSC

- ➔ Identical to F_1 Score
- ➔ Close relation to IoU (see Eq. 5)
- ➔ Preference in medical community

IoU

- ➔ Identical to Jaccard Index
- ➔ Close relation to DSC (see Eq. 4)
- ➔ Preference in computer vision community

Table SN 2.17. Comparison of Dice Similarity Coefficient (DSC) and Intersection over Union (IoU) in the context of the decision guide DG6.1 for Subprocess S6. Context: no exclusive interest in the center line of structures (FP2.3 = FALSE, FP3.3 = FALSE) and equal severity of class confusions (FP2.5.2 = FALSE).

The DSC (Fig. SN 3.5) is identical to the F_1 Score on pixel level and closely related to the IoU (Fig. SN 3.9), which, in turn, is identical to the Jaccard Index (see equations 4 and 5). The two metrics will yield the same ranking of aggregated metric values in most applications (theoretically, deviations are possible), such that there is no value in combining them. Commonly, the computer vision community prefers the IoU, while the medical image community favors the DSC.

$$IoU = \frac{DSC}{2 - DSC} \quad (4)$$

$$DSC = \frac{2IoU}{1 + IoU} \quad (5)$$

DG6.2: How to determine β in F_β Score

The F_β Score (Fig. SN 3.7) is defined as:

$$F_\beta = (1 + \beta^2) \cdot \frac{PPV \cdot \text{Sensitivity}}{(\beta^2 \cdot PPV) + \text{Sensitivity}} = \frac{(1 + \beta^2) \cdot TP}{(1 + \beta^2) \cdot TP + \beta^2 \cdot FN + FP} \quad (6)$$

The most common choice is to set β to 1, resulting in equal weighting of Sensitivity (Fig. SN 3.16) and PPV (Fig. SN 3.15). Higher values of β result in higher weights on FN penalties (undersegmentation in segmentation problems) compared to FP penalties (oversegmentation) and thus imply a focus on Sensitivity compared to PPV.

2.7.6 Decision guide S7.

DG7.1: Normalized Surface Distance (NSD) versus Boundary Intersection over Union (IoU)

Summary of DG7.1: NSD versus Boundary IoU

NSD

- ➔ Accounting for noisy images, limited resolution or imprecise reference annotations
- ➔ Influence of hyperparameter on scores: distances below tolerance threshold are considered TP

Boundary IoU

- ➔ Measurement of overlap between predicted and reference contours up to certain width
- ➔ Influence of hyperparameter on scores: distance parameter determines thickness of the considered boundary

Table SN 2.18. Comparison of Normalized Surface Distance (NSD) and Boundary Intersection over Union (IoU) in the context of the decision guide DG7.1 for Subprocess S7. Context: possibility of spatial outliers in the reference annotation (FP4.3.2 = TRUE) or, if FALSE, FP2.5.6 = existence-based penalization of outliers.

The following aspects should be considered when deciding between NSD (Fig. SN 3.26) and Boundary IoU (Fig. SN 3.23):

- **Different research questions:** Both metrics set the focus on the boundary/contour of structures, but fundamentally differ in what they measure: NSD measures the DSC score on the surface voxels (often interpreted as the ratio of correctly predicted contour), where the strictness for what constitutes a correct boundary is controlled by a tolerance parameter. This way, noise in the image, limited resolution, or imprecise reference annotations can be accounted for. Boundary IoU directly measures the overlap between predicted and reference contours (without tolerance) up to a certain width (which is controlled by a width parameter). Thus, NSD is preferable if a tolerance accounting for imprecise annotations is requested. Boundary IoU, on the other hand, is preferable if contour errors are thought of as crucial inconsistencies that should be assessed, or if a wider area around the contour line is of interest (dynamic transition to the classical IoU).
- **Setting the hyperparameter:** The NSD and Boundary IoU both require users to manually set a hyperparameter. **NSD:** Boundary distances below the tolerance threshold will be considered TP (deviations do not count as errors). This parameter can be set according to the inter-rater variability or, if not available, heuristics. **Boundary IoU:** The distance parameter determines the thickness of the considered boundary and thus also influences the sensitivity to contour errors (the smaller the distance, the higher the sensitivity). This parameter can also be set according to the inter-rater variability (here in order to capture potential inconsistencies, as opposed to disregarding noise as in NSD) or, if not available, heuristics.

DG7.2: Mean Average Surface Distance (MASD) versus Average Symmetric Surface Distance (ASSD)

Summary of DG7.2: MASD versus ASSD

<p>MASD</p> <ul style="list-style-type: none"> ➔ Equal contribution of reference and prediction boundaries to the metric score ➖ Possibility of misleading results in corner cases (e.g., tiny prediction closely located to the reference) <p>Table SN 2.19. Comparison of Mean Average Surface Distance (MASD) and Average Symmetric Surface Distance (ASSD) in the context of the decision guide DG7.2 for Subprocess S7. Context: FP2.5.6 = distance-based penalization of outliers with contour focus.</p>	<p>ASSD</p> <ul style="list-style-type: none"> ➖ Domination of the metric score by the larger contour
--	---

The ASSD (Fig. SN 3.22) puts all boundary distances (all distances from boundary A to boundary B and all distances from boundary B to boundary A) in a list, then takes the mean (Fig. SN 2.21). Thus, if one boundary is much larger than the other, this boundary will impact the mean much more. The MASD (Fig. SN 3.25) computes the sum of the mean distances from boundary A to boundary B and the mean distances from boundary B to boundary A. Therefore, the reference and prediction boundaries contribute equally (see Fig. SN 2.21). While there are corner cases in which MASD features disadvantages compared to ASSD as well (see Fig. SN 2.22), we generally recommend MASD because of the aforementioned advantage.

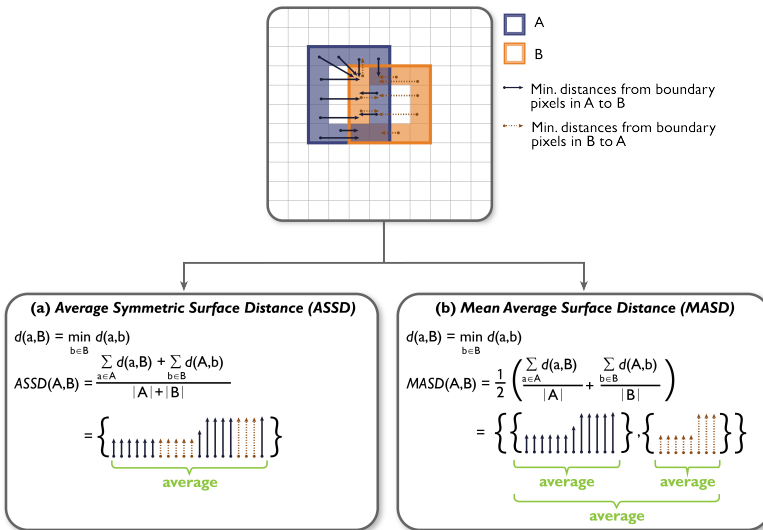


Fig. SN 2.21. Most commonly used distance-based segmentation metrics: (a) the Average Symmetric Surface Distance (ASSD) and (b) the Mean Average Surface Distance (MASD). The term $d(a, b)$ denotes the Euclidean distance between boundary pixels a and b . Only the True Positives (TPs) are considered.

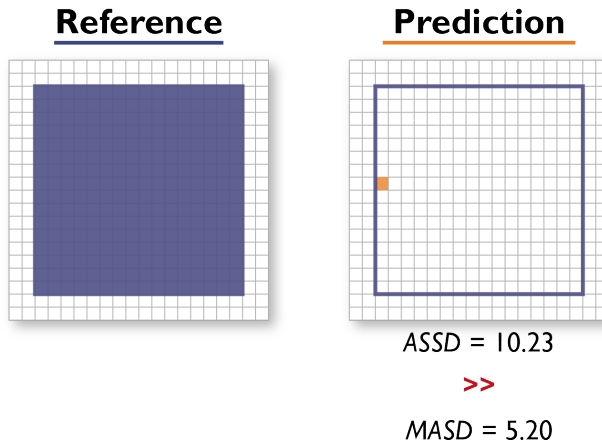


Fig. SN 2.22. Corner case in which Mean Average Surface Distance (MASD) yields an undesired result. If the *Prediction* is very small (here: one pixel) and located close to the reference boundary, the Mean Average Surface Distance (MASD) will be much lower compared to the Average Symmetric Surface Distance (ASSD).

DG7.3: Hausdorff Distance (HD) versus X^{th} Percentile Hausdorff Distance (X^{th} Percentile HD)

Summary of DG7.3: HD versus X^{th} Percentile HD

<p>HD</p> <ul style="list-style-type: none"> ➔ Sensitivity to spatial outliers 	<p>X^{th} Percentile HD</p> <ul style="list-style-type: none"> ➔ Compensation for spatial outliers
--	---

Table SN 2.20. Comparison of Hausdorff Distance (HD) and X^{th} Percentile Hausdorff Distance (X^{th} Percentile HD) in the context of the decision guide DG7.3 for Subprocess S7. Context: FP2.5.6 = distance-based penalization of outliers with outlier focus.

The HD (Fig. SN 3.24) calculates the maximum of all shortest distances for all points from one object boundary to the other, which is why it is also known as the Maximum Symmetric Surface Distance [120]. The X^{th} Percentile HD calculates the X^{th} percentile (e.g., 95% percentile, the Hausdorff Distance 95th Percentile (HD95), Fig. SN 3.27) instead of the maximum, and should therefore be used instead if spatial outliers should be disregarded (FP2.5.6, see Fig. SN 2.23).

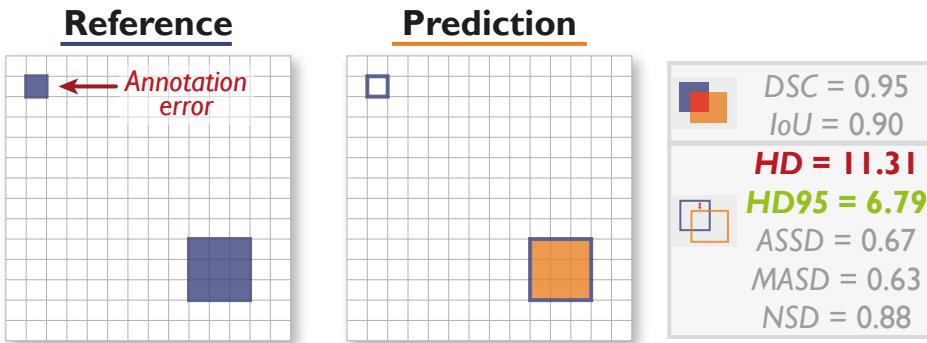


Fig. SN 2.23. Effect of annotation errors/noise. A single erroneously annotated pixel may lead to a large decrease in performance, especially in the case of the Hausdorff Distance (HD) when applied to small structures. The Hausdorff Distance 95th Percentile (HD95), on the other hand, was designed to deal with spatial outliers. Further abbreviations: Dice Similarity Coefficient (DSC), Intersection over Union (IoU), Average Symmetric Surface Distance (ASSD), Normalized Surface Distance (NSD).

2.7.7 Decision guide S8.

DG8.1: Mask Intersection over Union (IoU) versus Boundary IoU versus Intersection over Reference (IoR)

Summary of DG8.1: Mask IoU versus Boundary IoU versus IoR

Mask IoU

- ➔ Focus on overlap

- ➕ Widely used

- ➖ Possible heavy penalization of predictions in the case of a high ratio of touching reference objects

- ➖ Over-penalization of small structure sizes in tasks with high variability of sizes (FP3.2)

Boundary IoU

- ➔ Focus on structure boundaries

- ➔ Recently proposed alternative; not well known

- ➖ Possibility of yielding perfect value for imperfect predictions

- ➔ Additional hyperparameter, which can be determined based on inter-rater variability, for example

IoR

- ➔ Focus on overlap

- ➔ Not well known

- ➔ Preferability in the case of a high ratio of touching reference objects

- ➖ Can be deceived by large predictions

Table SN 2.21. Comparison of Mask Intersection over Union (IoU), Boundary IoU and Intersection over Reference (IoR) in the context of the decision guide DG8.1 for Subprocess S8. Context: FP1.1 = instance segmentation (InS).

In instance segmentation problems, it might be appropriate to base the localization criterion on the corresponding target segmentation metric (*custom criterion*). For example, if the target segmentation metric chosen in Subprocess S6 (Extended Data Fig. 6) is NSD, the localization criterion could be defined accordingly. This may not always be possible, for example because the target metric has no fixed upper bound (e.g., HD), rendering the setting of adequate localization cutoffs challenging. An alternative strategy is to choose one of the common object detection localization criteria.

The following aspects should be taken into account when deciding between Mask IoU (Fig. SN 3.9), Boundary IoU (Fig. SN 3.23), and IoR (Fig. SN 3.37) in instance segmentation problems. We will

first focus on the more subtle distinction between Mask IoU and Boundary IoU, and finally discuss scenarios for potential usage of IoR:

Boundary versus Mask IoU

- **Boundary focus:** While Mask IoU measures the overlap of structures in general, Boundary IoU allows to focus on the correctness of boundaries (FP2.1, see Fig. SN 2.24). Note that the focus on boundaries also comes with pitfalls. Boundary IoU can even be deceived to result in a perfect value of 1.0 despite an imperfect prediction (see Fig. SN 2.25).
- **Small structures:** Mask IoU over-penalizes small structures in tasks with high variability of structure sizes (FP3.2) because boundary pixels increase linearly (or quadratically) with size, while total pixels increase quadratically (or cubically) with size. Boundary IoU [22] addresses this issue by selecting only pixels with a maximum distance of “d” with regard to the boundary for validation (see Fig. SN 2.24).
- **Hyperparameters:** For the computation of Boundary IoU, the distance “d” constitutes an additional and sensitive hyperparameter to be determined. It can be determined based on inter-rater variability, for example.
- **Popularity:** While Mask IoU represents an established concept that is well-known to the community, Boundary IoU is a recently proposed modification [22] that might thus require specific introduction when used in validation.

IoR In the case of a high ratio of touching reference objects, ‘non-split errors’ (one prediction overlaps multiple reference objects) might occur frequently. While the IoU criterion can potentially heavily penalize this scenario resulting in FN and multiple FPs, a less severe penalization might be desired, e.g., in the form of the Intersection over Reference (IoR) [70]. IoR essentially considers the ratio of the area of a reference object that is covered by a prediction (see Fig. SN 2.26), allowing for multiple TP matches of the same prediction. Appropriate penalization in these cases is then ensured either by separating such errors as ‘merge errors’ [21], or by means of additional segmentation metrics. IoR shares the behaviour of Mask IoU regarding the above discussions on boundaries and small structures. As a major disadvantage, it can be deceived by large predictions. Widespread usage of IoR is currently limited to the field of cell segmentation, where images with high density of structures are present [70].

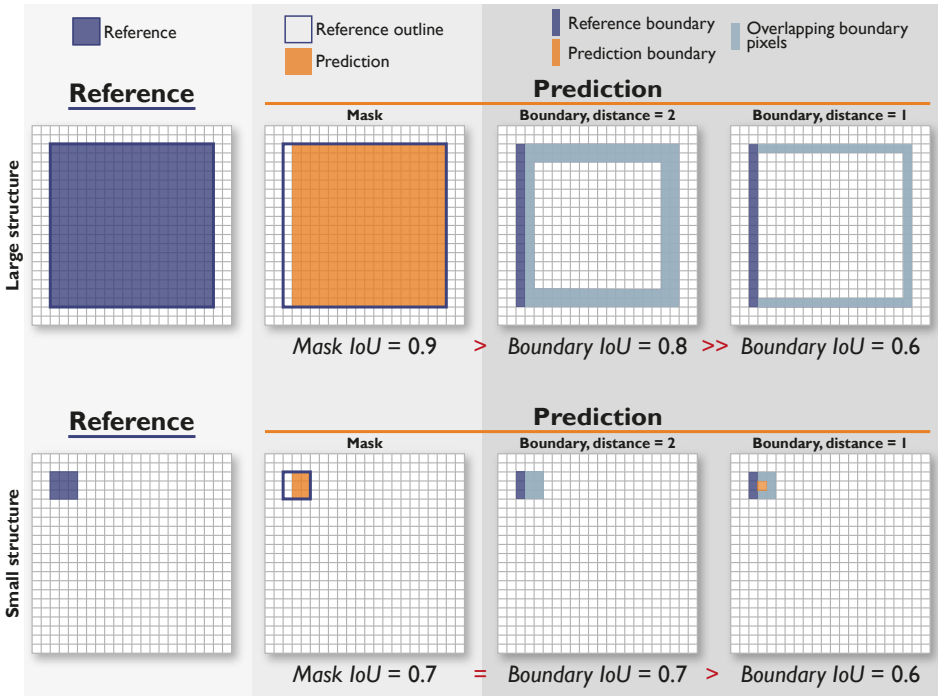


Fig. SN 2.24. Compared to the Mask Intersection over Union (IoU), the Boundary IoU (third and fourth column, representing two different thresholds) (1) specifically penalizes errors in the boundaries and (2) is more invariant to structure sizes (top: large; bottom: small).

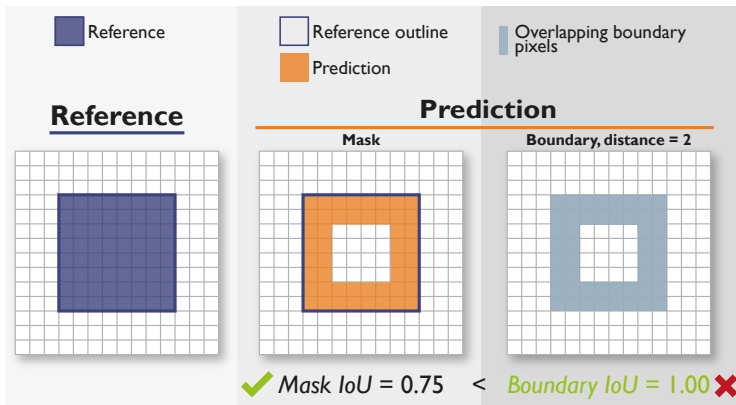


Fig. SN 2.25. Example of a perfect Boundary Intersection over Union (IoU) score for an imperfect prediction. Overlapping pixels from the reference and prediction are shown in light blue. For a prediction with a hole in the middle, the Boundary IoU may result in a score of 1.00 if the distance to border contains all mask pixels (here: distance = 2). However, the Mask IoU spots the problem and yields a lower score.

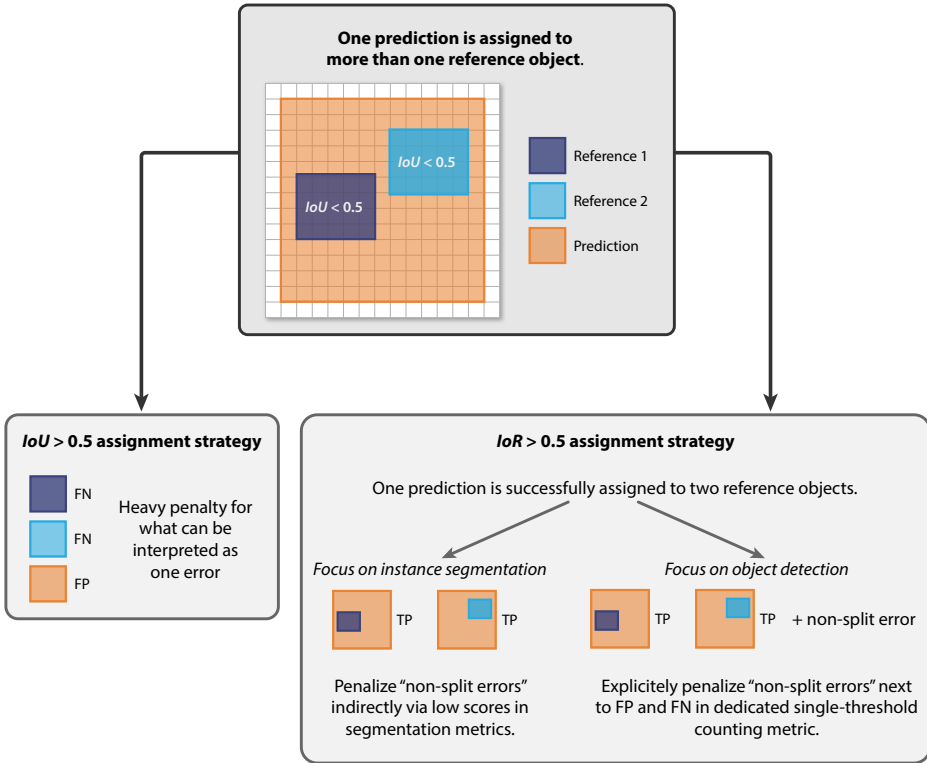


Fig. SN 2.26. In case of one prediction assigned to multiple reference objects, an assignment strategy needs to be chosen. This may be based, for example, on the *Intersection over Union* (IoU) > 0.5 strategy, which may result in a heavy penalty (two False Negatives (FN) and one False Positive (FP)). Another option is to use the *Intersection over Reference* (IoR) > 0.5 strategy, which examines whether the prediction was successfully assigned to the reference objects. In an additional step, the "non-split errors" will be penalized. Used abbreviations: False Negative (FN), False Positive (FP) and True Positive (TP).

DG8.2: Mask Intersection over Union (IoU) > 0 versus Center Distance versus Point inside Mask/Box/Approx

Summary of DG8.2: Mask IoU > 0 versus Center Distance versus Point inside Mask/Box/Approx

Mask IoU > 0

- ⊕ No hyperparameters (standardized)
- ⊖ Strictness of criterion cannot be varied
- ⊖ Potential large ambiguity of the predicted location as only few pixels overlap the reference

Center Distance

- ⊖ Distance threshold must be provided
- ⊕ Strictness of criterion can be varied
- ⊕ Good representation of the object center (FP2.3)

Point inside Mask/Box/Approx

- ⊕ No hyperparameters (standardized)
- ⊖ Strictness of criterion cannot be varied
- ⊕ Relatively good representation of tubular or disconnected structures

Table SN 2.22. Comparison of Mask Intersection over Union (IoU) > 0, Center Distance and Point inside Mask/Box/Approx localization criteria in the context of the decision guide DG8.2 for Subprocess S8. Context: FP1.1 = object detection (ObD) problems in the case of either (1) FP4.4 = reference annotations provided as exact outline and FP2.4 = a desired localization as only position or (2) FP4.4 = reference annotations provided as rough outline and FP2.4 = a desired localization as only position. Note that Mask IoU > 0 is only relevant for case (1).

When choosing a localization criterion for tasks where the mere existence of objects is of interest (as opposed to the outlining of objects), the following aspects should be considered:

- **Loose criterion:** (only recommended if the reference is provided as exact outline (FP2.4)) The intuitive choice of a very loose IoU criterion (e.g., “IoU > 0” or “at least one pixel overlap”, Fig. SN 3.9) comes with simplicity but implies the pitfall that the size of the predicted structure is in theory unbounded, i.e., the predicted location can be ambiguous (see Fig. SN 2.27).
- **Point-based criteria:** A preferable alternative for the case of pure localization (without interest in outlines) is to constrain the prediction to a single coordinate. A common criterion for this scenario is the distance to the center point of the structure (Fig. SN 3.36; which can also be of explicit interest, see FP2.3, Fig. SN 2.28). The center point², however, might not be a good reference for tubular structures (check FP3.3) or disconnected structures (check FP3.6). In such cases (and if annotations are provided in the form of masks), a binary Point inside Mask/Box/Approx criterion (Fig. SN 3.40) might be the better choice. On the other hand, the Point inside Mask/Box/Approx criterion does not allow for a variation of the criterion’s strictness (i.e., threshold). Application despite this shortcoming should be well-justified.

²Depending on what kind of information the center point is derived from, different definitions are possible as detailed in Fig. SN 3.36.

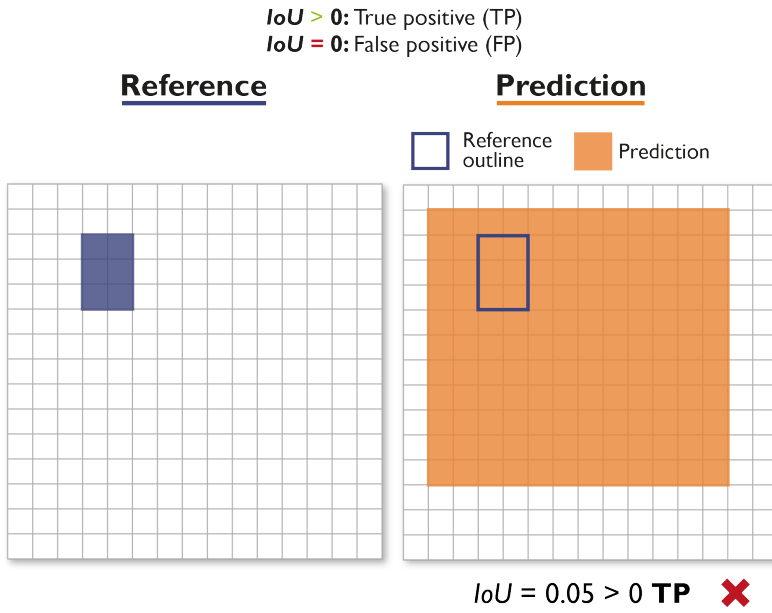


Fig. SN 2.27. Effect of a loose Intersection over Union (IoU) criterion. When defining a True Positive (TP) by an $IoU > 0$, the resulting localizations may be deceived by very large predictions.

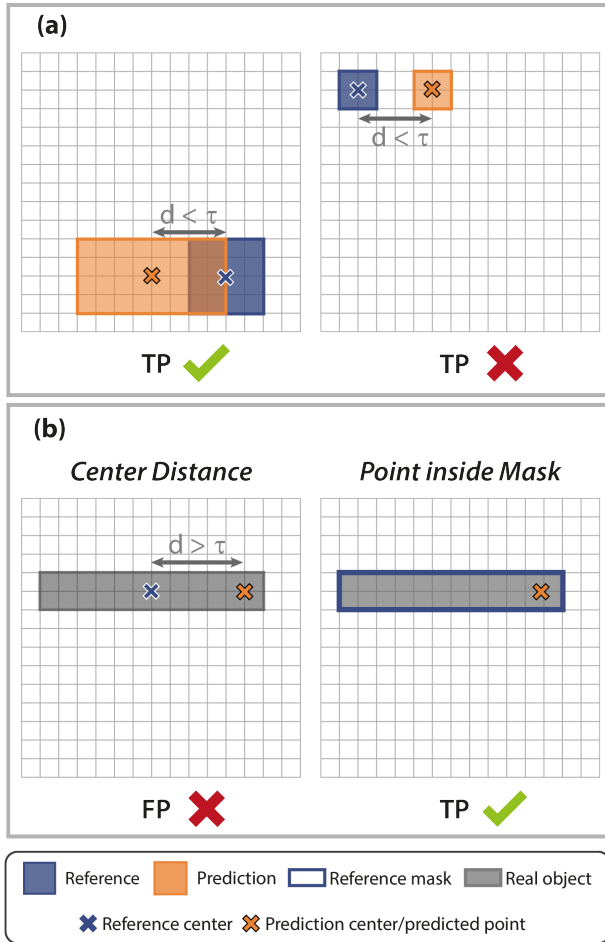


Fig. SN 2.28. Pitfalls of the Center Distance. **(a) Ignoring overlap between objects.** Both predictions have the same distance to their corresponding reference center. The Center Distance, which requires a threshold distance τ between center points not be exceeded, does not take into account the overlap between objects. However, the right prediction does not overlap the reference and should thus not be considered a True Positive (TP). **(b) Tubular structures.** The Center Distance is not an ideal criterion because it implies that the prediction shown would result in a False Positive (FP), although it hits the elongated structure. This could be overcome by a Point inside Mask criterion.

DG8.3: Choose localization threshold

Summary DG8.3: Choose localization threshold

Lower thresholds

- ➔ Interest in the existence of objects rather than their precise localization
- ➔ Small size of structures (FP3.1)
- ➔ High variability of structure sizes (FP3.2)
- ➔ 3D input images
- ➔ Uncertainties in the reference (FP4.3.1)

Higher thresholds

- ➔ Interest in precise localization
- ➔ Dense distribution of structures in images (FP3.5)

Table SN 2.23. Choosing the localization threshold in the context of the decision guide DG8.3 for Subprocess S8. This decision guide does not apply for Point inside Mask/Box/Approx criteria.

Note that most localization criteria require a threshold to be set (e.g., $\text{IoU} > 0.5$ counts as detected). However, such cutoff renders the validity of results limited to the specific threshold. To increase robustness of reported metrics, it is common practice in the computer vision community to average metrics over multiple cutoff values (default for IoU criteria: from 0.5 until 0.9 in steps of 0.05). On the other hand, certain properties of the underlying problem may limit the relevance of cutoff values to lower or higher values.

The following properties might warrant validation with lower thresholds: interest in the existence of objects rather than their precise localization, small size of structures (FP3.1), high variability of structure sizes (FP3.2), 3D input images (as volume increases cubically with size, the desired overlap ratio might require adaptation), uncertainties in the reference (FP4.3.1). Conversely, these properties typically warrant validation at higher thresholds: interest in precise localization, dense distribution of structures in images (FP3.5).

It should be noted that no threshold is needed for the Point inside Mask/Box/Approx and Mask $\text{IoU} > 0$ criteria.

2.7.8 Decision guide S9.

DG9.1: Assignment without predicted class probabilities on instance level

Summary DG9.1

"Localization criterion" > 0.5

- ➔ Inherent avoidance of assignment ambiguities
- ➔ Unfeasibility if overlapping predictions are possible

Greedy Matching

- ➔ No necessity of sophisticated strategies

Optimal (Hungarian) Matching

- ➔ Necessity of sophisticated strategies
- ➔ Optimistic interpretation/validation of ambiguous model outputs, but might not represent the most realistic approximation of model performance

Table SN 2.24. Comparison of assignment strategies in the context of the decision guide DG9.1 for Subprocess S9. Context: lack of predicted class scores (FP5.1 = FALSE).

The following aspects should be considered when selecting the assignment strategy:

- **Matching via Overlap > 0.5 (Fig. SN 3.44):** If overlapping predictions are not possible (FP5.4 = FALSE), sophisticated matching strategies are often avoided in the biomedical domain by setting the threshold for the localization criterion (Mask IoU, Boundary IoU, or IoR) to > 0.5. With this strategy, assignment ambiguities are inherently avoided. However, if either overlapping predictions are possible, a non-overlap based criterion is employed, or a criterion with a threshold above 0.5 is not appropriate, one of the following strategies should be chosen.
- **Greedy Matching (Figs. SN 3.41, SN 3.42):** A greedy approach can be taken, in which each reference is assigned to the best matching prediction. If predicted class scores are available (FP5.1 = TRUE) this is typically achieved based on the class score ("Greedy by Score Matching", Fig. SN 3.41). In the given scenario with FP5.1 = FALSE, an intuitive alternative is to rank predictions by the localization criterion score ("Greedy by localization criterion Matching", Fig. SN 3.42). Assignment is then achieved by stepping through the ranked list, matching the current prediction with the most overlapping reference object, and removing the reference object from the assignment process.
- **Optimal (Hungarian) Matching (Fig. SN 3.43):** The Hungarian algorithm optimizes the matching between predictions and reference objects while minimizing a given cost function, such as the average overlap for all matched pairs. Notably, this optimization generally leads to optimistic interpretation/validation of ambiguous model outputs, but might not represent the most realistic approximation of model performance upon application (see Fig. SN 2.29).

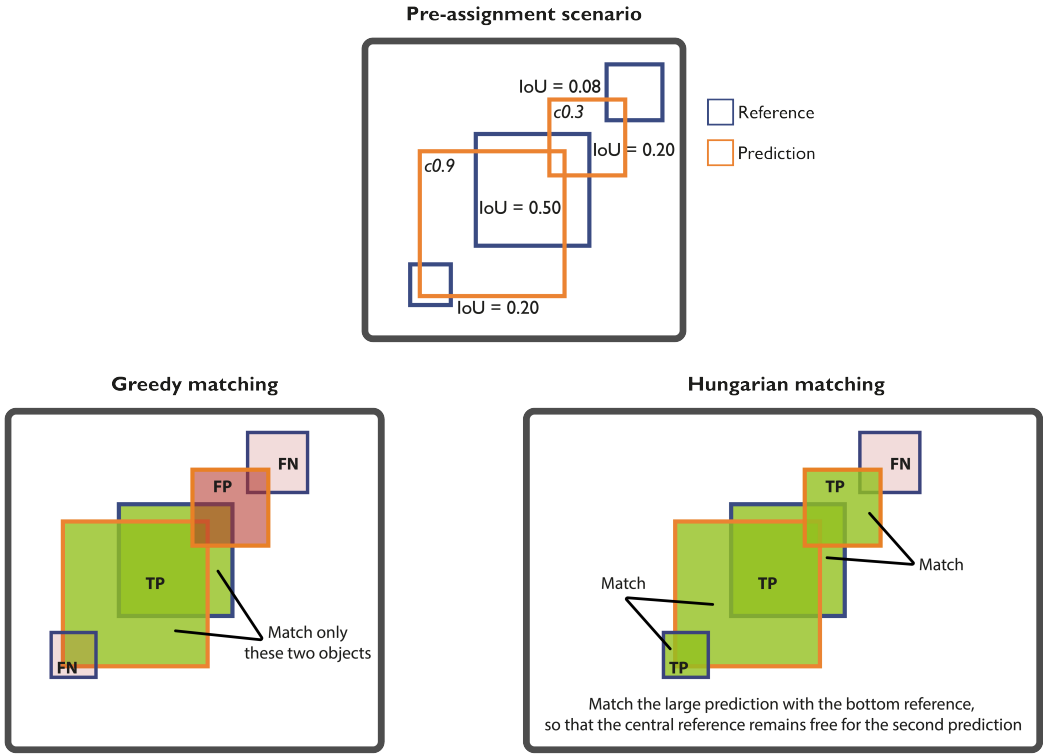


Fig. SN 2.29. Comparison of Greedy versus Hungarian matching assignment strategies.

SUPL. NOTE 3 STEP 3 - METRIC APPLICATION

Once a suitable metric pool has been generated, the chosen metrics must be applied to the given data set. We recommend beginning with the setting of the global decision threshold in case metrics based on a fixed cutoff on the predicted class scores (FP2.6; more generally: decision region for more than two classes) have been selected, which is generally the case. In order to avoid overestimation of algorithm performance, this threshold needs to be set globally for all classes and metrics, as detailed in Suppl. Note 1.1. Once raw metric values have been computed for all metrics, metric values are aggregated, potentially combined (for rankings) and reported according to the recommendations in Tab. 2. Importantly, we support the user by providing cheat sheets for the entire pool of *Metrics Reloaded* metrics that help find reference implementations and overcome metric-specific pitfalls (Suppl. Note 3.1).

3.1 Metrics Cheat Sheets

In this section, we present cheat sheets for the metrics deemed particularly relevant by the *Metrics Reloaded* consortium. We provide a description along with the formula as well the respective value range. For every metric, we indicate further important characteristics, such as the recommended problem categories or potential prevalence dependency. Finally, we highlight our recommendations. Many of the presented metrics rely on the confusion matrix, which is illustrated in Fig. SN 3.1.

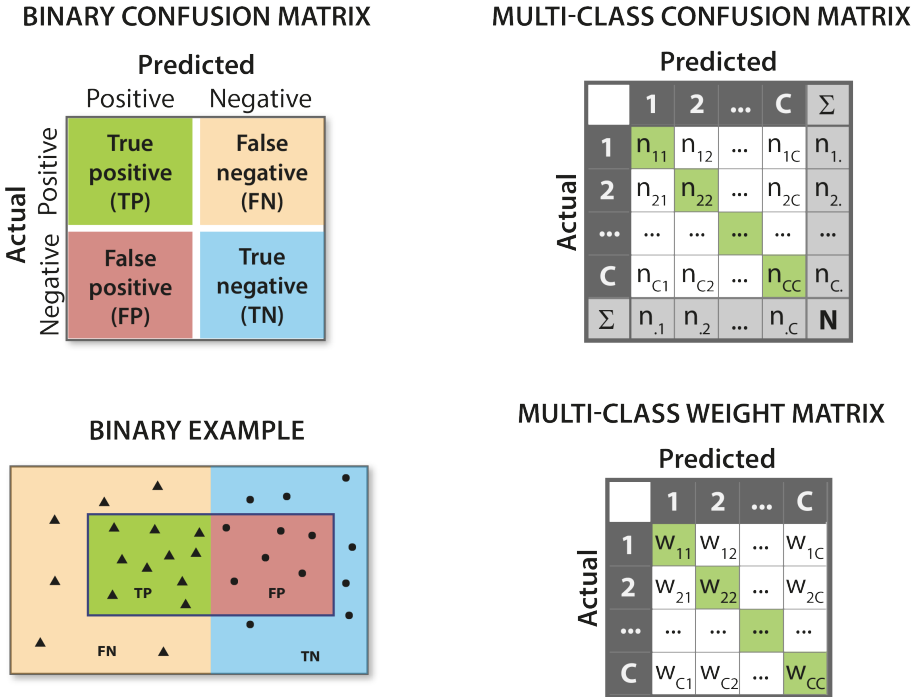


Fig. SN 3.1. Schematic example of the confusion matrix for two and for C classes. For the latter case, we also present a weight or cost matrix with weights $w_{ij} > 0$ without loss of generality. For the binary confusion matrix, we show an example illustrating the cardinalities for a prediction of triangles and circles.

3.1.1 Discrimination metrics.

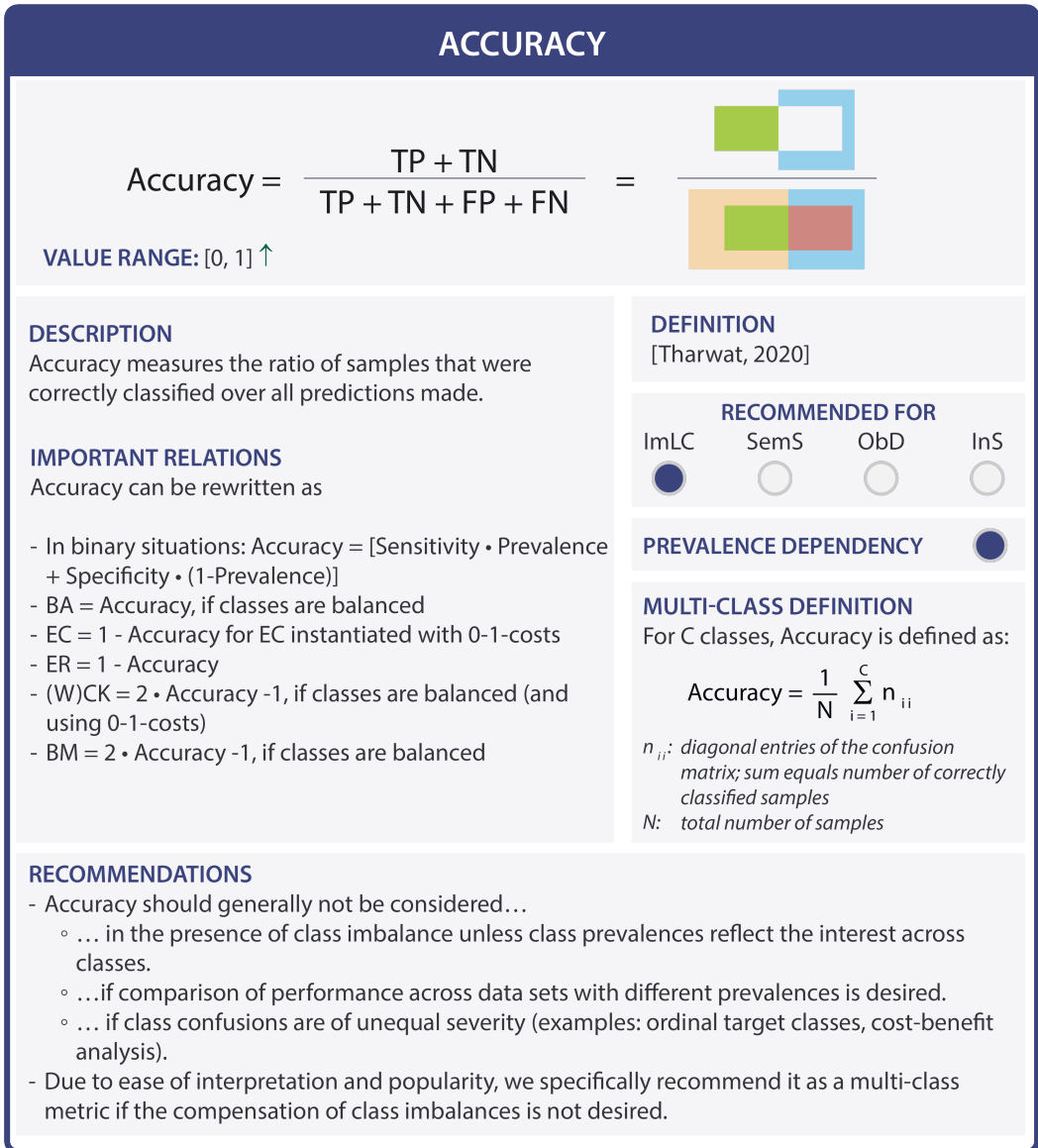
Counting metrics

Fig. SN 3.2. Cheat Sheet for the Accuracy. The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: Balanced Accuracy (BA), Bookmaker Informedness (BM), Cohen's Kappa (CK), Expected Cost (EC), Error Rate (ER), False Negative (FN), False Positive (FP), Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Semantic Segmentation (SemS), True Negative (TN), True Positive (TP), Weighted Cohen's Kappa (WCK). Reference used in the figure: Tharwat, 2020: [99]. We recommend Accuracy as a multi-class counting metric in Subprocess S2 (Extended Data Fig. 2).

BALANCED ACCURACY (BA)

$$BA = \frac{1}{2} (\text{Sensitivity} + \text{Specificity}) = \frac{1}{2} \left(\frac{\text{Green Box}}{\text{Orange Box}} + \frac{\text{Blue Box}}{\text{Red Box}} \right)$$

VALUE RANGE: [0, 1] ↑

DESCRIPTION
BA measures the arithmetic mean of Sensitivities for each class, i.e., for each class, it measures the fraction of actual positive samples that were predicted as such.

DEFINITION
[Tharwat, 2020]

IMPORTANT RELATIONS

- $J = 2BA - 1$
- $(W)CK = 2BA - 1$, if classes are balanced (and using 0-1-costs)
- Accuracy = BA, if classes are balanced
- $EC = 1 - BA$, if EC costs are chosen such that $w_{ii} = 0$ and $w_{ij} = 1/(C P_i)$, where w_{ij} are the costs for a sample of actual class i that was predicted as class j , C is number of classes and P_i is prevalence of class i .

RECOMMENDED FOR

ImLC	SemS	ObD	InS
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

PREVALENCE DEPENDENCY

MULTI-CLASS DEFINITION
For C classes, BA is defined as the arithmetic mean of Sensitivities per class:

$$BA = \frac{1}{C} \sum_{i=1}^C \text{Sensitivity}_i = \frac{1}{C} \sum_{i=1}^C \frac{n_{ii}}{n_{i.}}$$

n_{ii}: diagonal entries of the confusion matrix; sum equals number of correctly classified samples
n_{i.}: sum of entries of row i in the confusion matrix

RECOMMENDATIONS

- BA should not be applied if...
 - ... there is an unequal interest across classes.
 - ... predictive values should be assessed.
 - ... class confusions are of unequal severity (examples: ordinal target classes, cost-benefit analysis).
- Otherwise, it should specifically be considered...
 - ... in the presence of high class imbalance in case there is an equal interest across classes.
 - ... if a comparison of performance across data sets with different prevalences is desired.
- BA can be used to identify and validate the decision rule applied to predicted class scores.

Fig. SN 3.3. Cheat Sheet for the Balanced Accuracy (BA). The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Semantic Segmentation (SemS), Cohen’s Kappa (CK), Expected Cost (EC), Youden Index (J), Weighted Cohen’s Kappa (WCK). We recommend BA as a multi-class counting metric in Subprocess S2 (Extended Data Fig. 2).

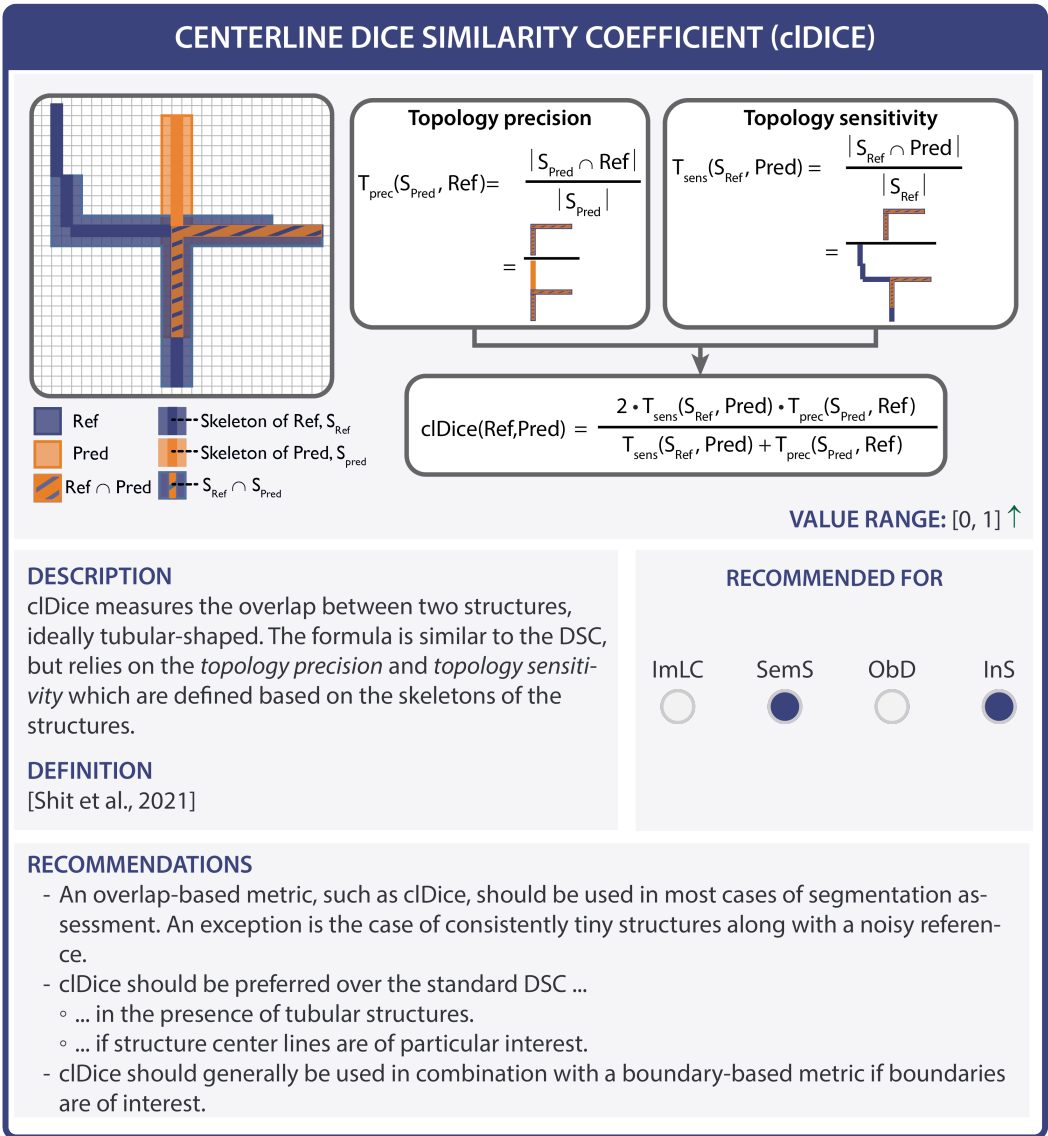
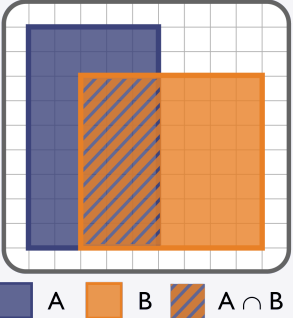


Fig. SN 3.4. Cheat Sheet for the centerline Dice Similarity Coefficient (cIDice). The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: Dice Similarity Coefficient (DSC), Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Semantic Segmentation (SemS). We recommend cIDice as an overlap-based metric in Subprocess S6 (Extended Data Fig. 6).

DICE SIMILARITY COEFFICIENT (DSC)

Synonyms: Dice, Dice Coefficient, Sørensen–Dice Coefficient, F_1 Score, Balanced F Score



■ A ■ B ■ $A \cap B$

$$DSC(A,B) = \frac{2 |A \cap B|}{|A| + |B|} = \frac{2 \text{ PPV} \cdot \text{Sensitivity}}{\text{PPV} + \text{Sensitivity}}$$

VALUE RANGE: [0, 1] ↑

DESCRIPTION
DSC measures the overlap between two structures.

DEFINITION
[Dice, 1945]

IMPORTANT RELATIONS

DSC is closely related to the IoU = Jaccard index:

$$DSC = \frac{2 \text{IoU}}{1 + \text{IoU}}$$

DSC is equal to the F_1 Score ($\beta = 1$ in F_β Score) at pixel level.

RECOMMENDED FOR

ImLC

SemS

ObD

InS

RECOMMENDATIONS

- An overlap-based metric (by default the DSC or IoU) should be used in most cases of segmentation assessment. An exception is the case of consistently tiny structures along with a noisy reference.
- DSC should generally be used in combination with a boundary-based metric if boundaries are of interest.
- DSC should generally not be considered if...
 - ... there is a high variability of structure sizes within an image or across images.
 - ... inter-rater variability is requested to be compensated.
 - ... over- and undersegmentation should be treated similarly.
- DSC should be considered as a metric in the medical community rather than in the computer vision and biology communities (where the almost identical IoU is preferred).

Fig. SN 3.5. Cheat Sheet for the Dice Similarity Coefficient (DSC). The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: False Negative (FN), False Positive (FP), Image-level Classification (ImLC), Instance Segmentation (InS), Intersection over Union (IoU), Object Detection (ObD), Positive Predictive Value (PPV), Semantic Segmentation (SemS), True Negative (TN), True Positive (TP). Reference used in the figure: Dice, 1945: [35]. We recommend DSC as an overlap-based metric in Subprocess S6 (Extended Data Fig. 6).

EXPECTED COST (EC)/NORMALIZED EC (ECN)

Synonyms: Expected prediction error, Expected loss

$$EC = w_{miss} \cdot \frac{FN}{TP + FN} \cdot \frac{TP + FN}{TP + TN + FP + FN} + w_{FA} \cdot \frac{FP}{TN + FP} \cdot 1 - \frac{TP + FN}{TP + TN + FP + FN}$$

$$= w_{miss} \cdot \frac{\text{[Diagram: Missed Positive]}}{\text{[Diagram: Total Positives]}} \cdot \frac{\text{[Diagram: True Positives]}}{\text{[Diagram: Total Positives]}} + w_{FA} \cdot \frac{\text{[Diagram: False Positives]}}{\text{[Diagram: Total Negatives]}} \cdot 1 - \frac{\text{[Diagram: True Positives]}}{\text{[Diagram: Total Positives]}}$$

P_{miss} : FN (miss) rate, P_{FA} : FP (false alarm) rate
 P_{tar} : prior probability (prevalence)
 w_{miss}/w_{FA} : (estimation of) costs of the respective errors; can be adjusted as a weighting of them.

VALUE RANGE: $[0, \infty)$ ↓
 EC can be assumed to be positive if costs are non-negative, which can be done without loss of generality.

DESCRIPTION

EC is a generalization of the probability of error (which is, in turn, 1 - Accuracy) for cases in which errors cannot all be considered to have equally severe consequences. It is defined as the expectation of the cost, where the cost incurred on a certain sample depends on the sample's class and the decision made for that sample. In practice, the expectation can be estimated as a simple average of the costs over the evaluation samples. EC describes the weighted sum of error rates. It can be used to measure discrimination and calibration in one score.

VARIANT

Normalized EC (ECN): normalizes EC by the EC of a naive system.

DEFINITION
 [Bishop and Nasrabadi, 2006; Hastie et al., 2009; Ferrer, 2022]

IMPORTANT RELATIONS

- $BA = 1 - EC$, if costs are chosen such that $w_{ij} = 0$ for all i and $w_{ij} = 1/(C \cdot P_i)$, where w_{ij} are the costs for a sample of actual class i that was predicted as class j , C is number of classes and P_i is prevalence of class i .
- Accuracy = $1 - EC$, if using 0-1-costs
- Sensitivity = EC , if costs are set $w_{ii} = 1$ for that single i and 0 otherwise
- Specificity = EC , if costs are set $w_{ij} = 1$ all $j \neq i$ and 0 otherwise

RECOMMENDED FOR

ImLC	SemS	ObD	InS
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

PREVALENCE DEPENDENCY?

Both options are possible depending on how the priors are set in the definition of the metric.

MULTI-CLASS DEFINITION

For C classes, EC is defined as:

$$EC = \sum_{i=1}^C \sum_{j=1}^C P_i \cdot w_{ij} \cdot \frac{n_{ij}}{n_i}$$

n_{ij} : entry of the confusion matrix for row i and column j , i.e., samples of actual class i that have been predicted as class j
 n_i : sum of entries of row i of the confusion matrix
 w_{ij} : costs for the entry of the confusion matrix for row i and column j , i.e., the cost for predicting a sample of actual class i that was predicted as class j
 P_i : prevalence of class i ;
 usually (n_i / N) , but in some cases one might want to plug in P_i directly from a target application

RECOMMENDATIONS

- EC is generally recommended as multi-class counting metric due to its flexibility in handling costs and prevalences.
- It is specifically recommended if problem-specific error costs are available, because it naturally integrates these in the metric score.
- If class confusions are of unequal severity (examples: ordinal target classes), it is our default recommendation as multi-class counting metric.
- In binary settings, it is well-suited as a per-class counting metric in case a cost-benefit-based decision rule is desired for converting predicted class scores to decisions.
- The prevalences P_i can be set according to the expected prevalences in the target population (if known) in case the class prevalences of a data set do not match the prevalences of the target population.
- EC can be used as the basis for decomposing the system performance into discrimination and calibration components.
- EC can be used to identify and validate the decision rule applied to predicted class scores.

Fig. SN 3.6. Cheat Sheet for the Expected Cost (EC)/normalized EC (ECN). The downward arrow in the value range indicates that lower values are better than higher values. Abbreviations used in the figure: Balanced Accuracy (BA), Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Semantic Segmentation (SemS). References used in the figure: Bishop and Nasrabadi, 2006: [15], Ferrer 2022: [40], Hastie et al., 2009: [48]. We recommend EC as a per-class counting metric in Subprocess S2 (Extended Data Fig. 2).

F_β SCORE

$$F_{\beta} \text{ Score} = (1+\beta^2) \frac{\text{PPV} \cdot \text{Sensitivity}}{\beta^2 \cdot \text{PPV} + \text{Sensitivity}}$$

$$= \frac{(1+\beta^2) \cdot \text{TP}}{(1+\beta^2) \cdot \text{TP} + \beta^2 \cdot \text{FN} + \text{FP}} = \frac{(1+\beta^2) \cdot \text{Green}}{(1+\beta^2) \cdot \text{Green} + \beta^2 \cdot \text{Orange} + \text{Red}}$$

VALUE RANGE: [0, 1] ↑

DESCRIPTION

The F_β Score weights PPV (FP) and Sensitivity (FN) with the parameter β.

The special case of β = 1 is the harmonic mean of PPV and Sensitivity and is a common metric in segmentation problems (here usually referred to as DSC). In segmentation problems, F_β Score weights the penalization of oversegmentation (FP) and undersegmentation (FN) with the parameter β.

RECOMMENDED FOR

ImLC	SemS	ObD	InS
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

PREVALENCE DEPENDENCY

DEFINITION
[Van Rijsbergen, 1979; Chinchor 1992]

IMPORTANT RELATIONS

DSC = F_β, if β = 1, IoU = F_β / (2 - F_β), if β = 1

RECOMMENDATIONS

- F_β Score should generally not be considered ...
 - ... in the presence of class imbalance unless class prevalences reflect the interest across classes (ImLC).
 - ... if comparison of performance across data sets with different prevalences is desired (ImLC).
 - ... in the case of a high variability of structure sizes within an image or across images (SemS, InS).
 - ... if inter-rater variability is requested to be compensated (SemS, InS).
 - ... if the comparability relative to a naive classifier should be provided.
- Otherwise, the F_β Score is specifically recommended as per-class counting metric if ...
 - ... the task is object detection because unlike most popular per-class counting metrics, F_β Score does not require TN.
 - ... an optimization or argmax-based decision rule should be applied.
 - ... no predicted class scores are available.
 - ... high metric values should imply a high PPV.
- In segmentation tasks, F_β Score with β ≠ 1 should be considered if over- and undersegmentation are of unequal severity (SemS, InS).
- In InS tasks, the object-level F_β Score should be considered if the detection quality should be assessed independently from the segmentation quality.

Fig. SN 3.7. Cheat Sheet for the F_β Score. The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: Dice Similarity Coefficient (DSC), False Negative (FN), False Positive (FP), Image-level Classification (ImLC), Instance Segmentation (InS), Intersection over Union (IoU), Object Detection (ObD), Positive Predictive Value (PPV), Semantic Segmentation (SemS), True Negative (TN), True Positive (TP). References used in the figure: Chinchor 1992: [23], Van Rijsbergen, 1979: [109]. We recommend F_β Score as a per-class counting metric in Subprocess S3 (Extended Data Fig. 3).

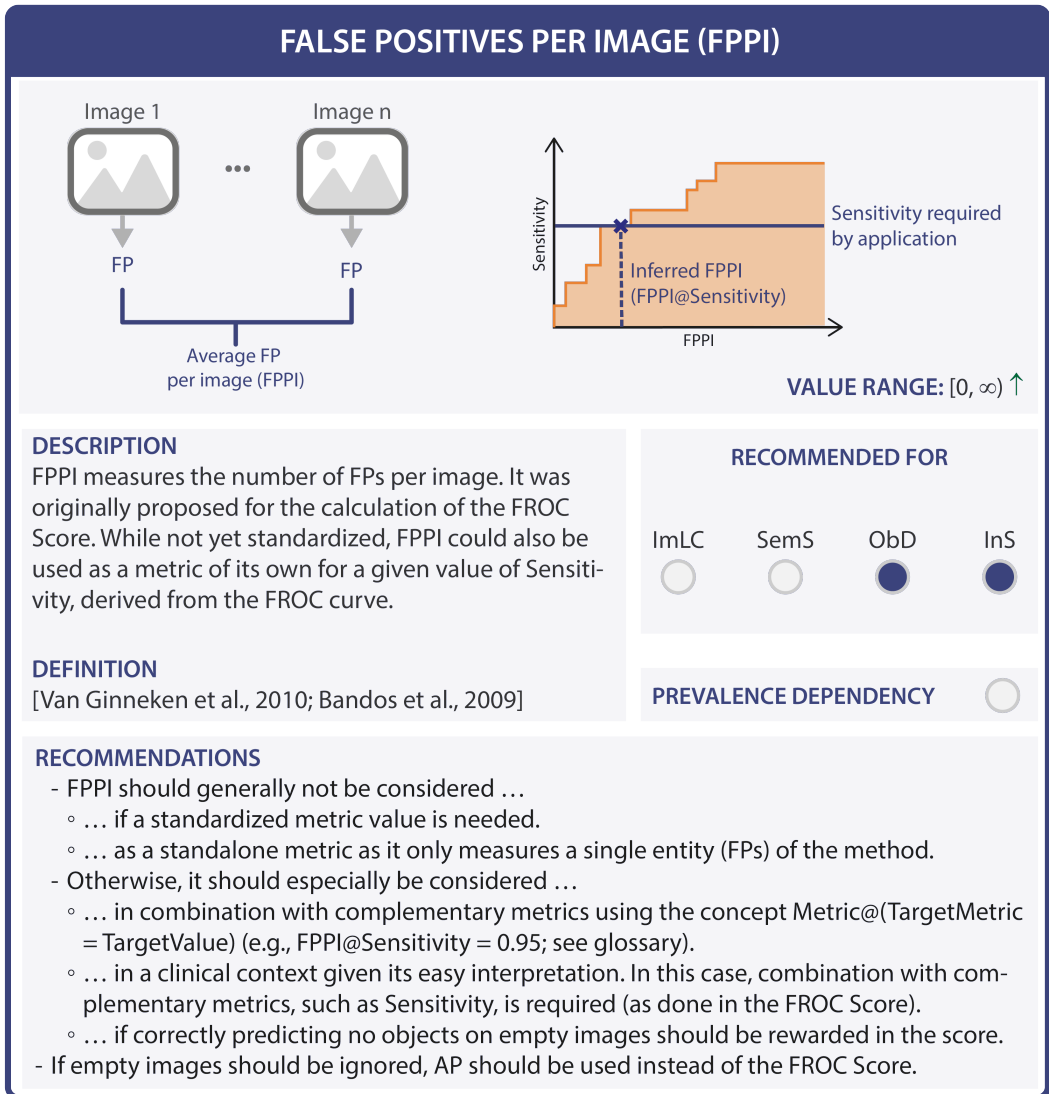
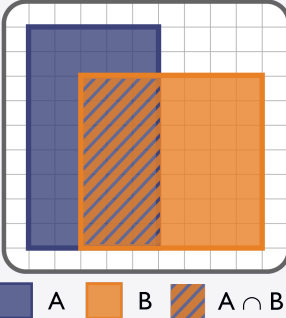


Fig. SN 3.8. Cheat Sheet for the False Positives per Image (FPPI) [8, 108]. The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: False Positive (FP), Free-Response Receiver Operating Characteristic (FROC), Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Semantic Segmentation (SemS). References used in the figure: Van Ginneken et al., 2010: [108], Bandos et al., 2009: [8]. We recommend FPPI as a per-class counting metric in Subprocess S3 (Extended Data Fig. 3) for target value-based optimization using the concept Metric@(TargetMetric = TargetValue) (e.g., FPPI@Sensitivity = 0.95; see glossary in Suppl. Note 5.4).

INTERSECTION OVER UNION (IoU)

Synonyms: Jaccard Index, Tanimoto Coefficient



$$IoU(A,B) = \frac{\text{Intersection}}{|A| + |B| - \text{Intersection}}$$

$$= \frac{|A \cap B|}{|A| + |B| - |A \cap B|} = \frac{|A \cap B|}{|A \cup B|}$$

$$= \frac{PPV \cdot Sensitivity}{PPV + Sensitivity - PPV \cdot Sensitivity}$$

VALUE RANGE: [0, 1] ↑

DESCRIPTION

IoU measures the overlap between two structures. It is often referred to as **Box IoU** when comparing bounding boxes, **Mask IoU** when comparing segmentation masks, or **Approx IoU** when comparing approximations of objects beyond bounding boxes.

DEFINITION
[Jaccard, 1912]

RECOMMENDED FOR

ImLC	SemS	ObD	InS
<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

IMPORTANT RELATIONS

$$IoU = \frac{DSC}{2 - DSC} \quad IoU = \frac{F_\beta}{2 - F_\beta}$$

for $\beta = 1$

RECOMMENDATIONS

- An overlap-based metric (by default DSC or IoU) should be used in most cases for segmentation assessment. An exception is the case of consistently tiny structures along with a noisy reference.
- IoU should generally be used in combination with a boundary-based metric if boundaries are of interest.
- IoU should generally not be considered if...
 - ... there is a high variability of structure sizes within an image or across images.
 - ... inter-rater variability is requested to be compensated.
 - ... over- and undersegmentation should be treated similarly.
- IoU should be considered as a metric in the computer vision and biology communities rather than in the medical community (which prefers the almost identical DSC).

Fig. SN 3.9. Cheat Sheet for the Intersection over Union (IoU). The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: Dice Similarity Coefficient (DSC), Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Positive Predictive Value (PPV), Semantic Segmentation (SemS). Reference used in the figure: Jaccard, 1912: [52]. We recommend IoU as an overlap-based metric in Subprocess S6 (Extended Data Fig. 6).

MATTHEWS CORRELATION COEFFICIENT (MCC)

Synonyms: Phi Coefficient

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} = \frac{\text{Green} \cdot \text{Blue} - \text{Red} \cdot \text{Orange}}{\sqrt{\text{Green} \cdot \text{Red} \cdot \text{Orange} \cdot \text{Blue} \cdot \text{Green} \cdot \text{Orange} \cdot \text{Red} \cdot \text{Blue}}}$$

VALUE RANGE: [-1, 1] ↑
A value of 0 refers to a prediction which is not better than random guessing.

DESCRIPTION
 MCC measures the correlation between the actual and the predicted class.

DEFINITION
 [Matthews, 1975]

IMPORTANT RELATIONS
 MCC can be rewritten as:

$$MCC = \sqrt{PPV \cdot \text{Sensitivity} \cdot \text{Specificity} \cdot NPV} - \sqrt{(1 - PPV) \cdot (1 - \text{Sensitivity}) \cdot (1 - \text{Specificity}) \cdot (1 - NPV)}$$

MCC is equivalent to the geometric mean of Markedness and Informedness.

MULTI-CLASS DEFINITION
 For C classes, MCC can be defined as:

$$MCC = \frac{\sum_{i=1}^C \sum_{j=1}^C \sum_{k=1}^C n_{ii} \cdot n_{jk} - n_{ij} \cdot n_{ki}}{\sqrt{\sum_{i=1}^C (\sum_{j=1}^C n_{ij}) (\sum_{r|r \neq i} \sum_{j'=1}^C n_{r'j'})} \sqrt{\sum_{i=1}^C (\sum_{j=1}^C n_{ji}) (\sum_{r|r \neq i} \sum_{j'=1}^C n_{j'r'})}}$$

n_{ij}: entry of the confusion matrix for row i and column j, i.e., samples of actual class i that were predicted as class j

RECOMMENDED FOR

ImLC	SemS	ObD	InS
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

PREVALENCE DEPENDENCY

RECOMMENDATIONS

- MCC should not be used/used with care if...
 - ... class confusions are of unequal severity (example: ordinal target classes).
 - ... the provided class prevalences do not reflect the population of interest.
 - ... there is a mismatch between class prevalences and class importance.
 - ... compensation for class imbalance is not requested.
- Otherwise, MCC should be used as a multi-class metric specifically if all basic error rates (Sensitivity, Specificity, PPV, NPV) should be captured in one score.
- MCC scores should be carefully interpreted in the presence of class imbalance as the distribution becomes skewed [Zhu 2020].

Fig. SN 3.10. Cheat Sheet for the Matthews Correlation Coefficient (MCC). The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: Image-level Classification (ImLC), Instance Segmentation (InS), Negative Predictive Value (NPV), Positive Predictive Value (PPV), Object Detection (ObD), Semantic Segmentation (SemS). References used in the figure: Matthews, 1975: [69], Zhu, 2020: [122]. We recommend MCC as a multi-class counting metric in Subprocess S2 (Extended Data Fig. 2).

NET BENEFIT (NB)

$$NB = \frac{TP}{TP + TN + FP + FN} - \frac{FP}{TP + TN + FP + FN} \cdot \left(\frac{p_t}{1 - p_t} \right)$$

VALUE RANGE: [-1, 1] ↑

DESCRIPTION
 NB validates the quality of a model intended to support a specific clinical decision. NB gives the 'net' proportion of TPs that results from a prediction. This is equivalent to the proportion of TPs in the absence of FPs. For its calculation, NB considers a task-related risk threshold (= exchange rate between the benefit of TPs and harm of FPs).

When varying the risk threshold over a 'reasonable range' of possible thresholds, plotting NB by risk threshold yields a decision curve. It is a strictly proper performance measure.

<p>DEFINITION [Vickers and Elkin, 2006]</p> <p>RELATIONS NB can be reformulated as: NB = Sensitivity • Prevalence - (1 - Specificity) • (1 - Prevalence) • "odds at the threshold"</p>	<p>RECOMMENDED FOR</p> <table style="width: 100%; text-align: center;"> <tr> <td>ImLC</td> <td>SemS</td> <td>ObD</td> <td>InS</td> </tr> <tr> <td><input checked="" type="radio"/></td> <td><input type="radio"/></td> <td><input type="radio"/></td> <td><input type="radio"/></td> </tr> </table> <p>PREVALENCE DEPENDENCY? <input checked="" type="radio"/></p>	ImLC	SemS	ObD	InS	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
ImLC	SemS	ObD	InS						
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>						

RECOMMENDATIONS

- NB should be considered if (1) the predicted class scores indicate the risk associated with a case belonging to a particular class and (2) there is a (range of reasonable) task-related risk threshold(s) or problem-specific penalties.
- If problem-specific penalties can be provided, these can be used to calculate the decision threshold, see [Pauker & Kassirer, 1975].

Fig. SN 3.11. Cheat Sheet for the Net Benefit (NB). The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: False Negative (FN), False Positive (FP), Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Semantic Segmentation (SemS), True Negative (TN). References used in the figure: Pauker and Kassirer, 1975: [81], Vickers and Elkin, 2006: [110], Vickers et al., 2016: [111]. We recommend NB as a per-class counting metric in Subprocess S3 (Extended Data Fig. 3).

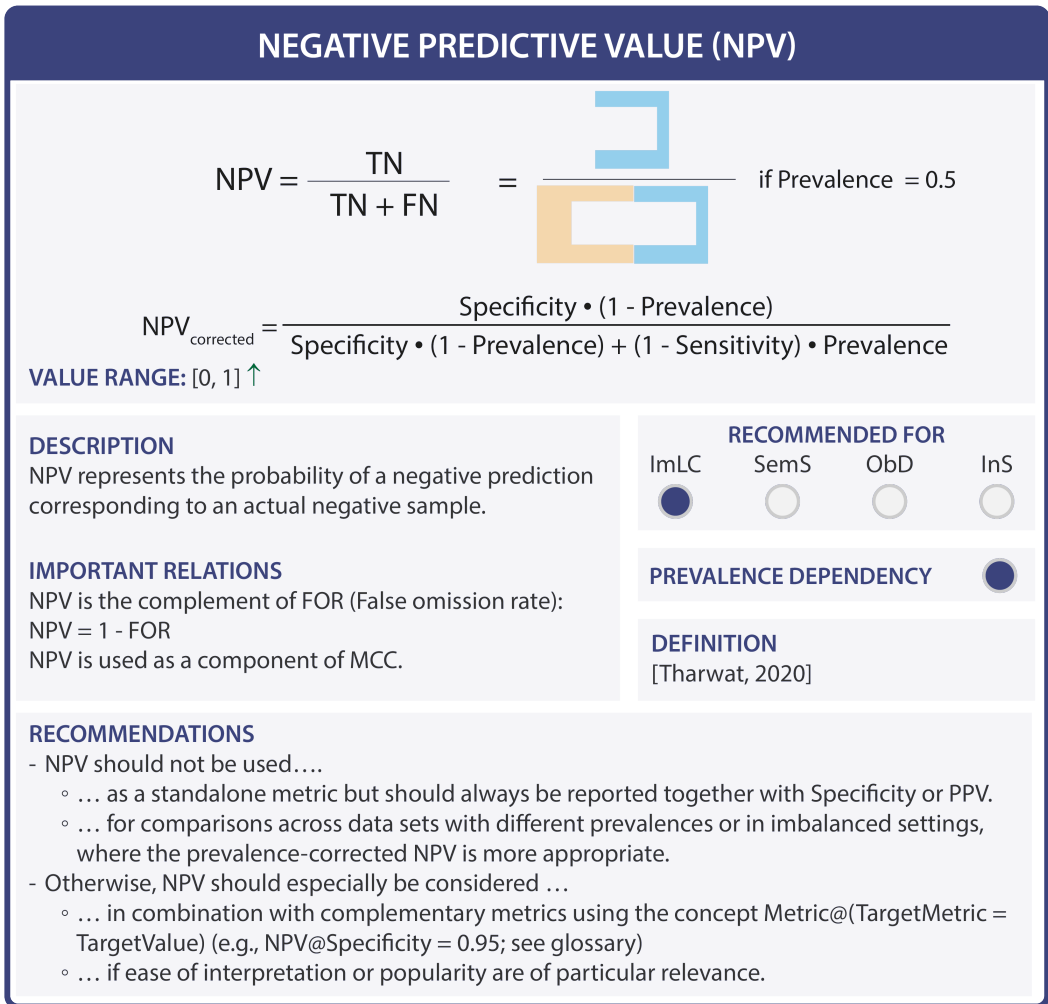


Fig. SN 3.12. Cheat Sheet for the Negative Predictive Value (NPV). The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: False Negative (FN), False Omission Rate (FOR), False Positive (FP), Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Positive Predictive Value (PPV), Semantic Segmentation (SemS), True Negative (TN). Reference used in the figure: Tharwat, 2020: [99]. We recommend NPV as a per-class counting metric in Sub-process S3 (Extended Data Fig. 3) for target value-based optimization using the concept Metric@(TargetMetric = TargetValue) (e.g., NPV@Sensitivity = 0.95; see glossary in Suppl. Note 5.4).

PANOPTIC QUALITY

■ + Reference (Ref) instances
■ Predicted (Pred) instances

VALUE RANGE: [0, 1] ↑

$$\begin{aligned}
 PQ &= \frac{\sum_{(\text{Ref}, \text{Pred}) \in \text{TP}} \text{IoU}(\text{Ref}, \text{Pred})}{|\text{TP}| + 0.5 |\text{FP}| + 0.5 |\text{FN}|} \\
 &= \underbrace{\frac{\sum_{(\text{Ref}, \text{Pred}) \in \text{TP}} \text{IoU}(\text{Ref}, \text{Pred})}{|\text{TP}|}}_{\text{Segmentation quality}} \cdot \underbrace{\frac{|\text{TP}|}{|\text{TP}| + 0.5 |\text{FP}| + 0.5 |\text{FN}|}}_{\text{Detection quality}} \\
 &= \frac{\text{IoU}(\{\text{blue square}, \text{orange square}\}) + \text{IoU}(\{\text{blue square}, \text{orange square}\})}{|\{\text{orange square}, \text{orange square}\}|} \\
 &\quad \cdot \frac{|\{\text{orange square}, \text{orange square}\}| + 0.5 |\{\text{red square}, \text{brown square}\}| + 0.5 |\{\text{blue cross}\}|}{|\{\text{orange square}, \text{orange square}\}| + 0.5 |\{\text{red square}, \text{brown square}\}| + 0.5 |\{\text{blue cross}\}|}
 \end{aligned}$$

DESCRIPTION

PQ assesses segmentation and detection quality in one metric. The segmentation quality is measured by averaging the IoU scores of all TP instances. The detection quality is measured by the F_1 Score. While in the F_1 Score, each TP counts as "1", PQ replaces this "1" score in the numerator with the IoU score of each TP.

The F_1 Score as a detection metric implies two cutoffs:

1. a prior cutoff on a localization criterion for matching and
2. a prior cutoff on object class scores to generate a confusion matrix.

In this context, PQ can be interpreted as making the localization quality in F_1 Score explicit (1) and thus only relying on the cutoff on class scores (2).

DEFINITION
[Kirillov et al., 2019]

RECOMMENDED FOR

ImLC	SemS	ObD	InS
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

RECOMMENDATIONS

PQ should be considered if the detection and segmentation quality should be assessed in a single score, e.g., when the overall best model needs to be selected, which can naturally not be done based on two separate metrics. For comprehensive validation, individual detection and segmentation performance should also be reported.

Fig. SN 3.13. Cheat Sheet for the Panoptic Quality (PQ). The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: Average Precision (AP), False Negative (FN), False Positive (FP), Free-Response Receiver Operating Characteristic (FROC), Image-level Classification (ImLC), Instance Segmentation (InS), Intersection over Union (IoU), Object Detection (ObD), Semantic Segmentation (SemS), True Positive (TP). Reference used in the figure: Kirillov et al., 2019: [56]. We recommend PQ as a per-class counting metric in Subprocess S3 (Extended Data Fig. 3).

POSITIVE LIKELIHOOD RATIO (LR+)

Synonyms: Likelihood ratio positive, Likelihood ratio for positive results

$$LR+ = \frac{\text{Sensitivity}}{1 - \text{Specificity}} = \frac{\text{Green Box}}{\text{Orange Box} - \text{Green Box}} \Bigg/ \left(1 - \frac{\text{Blue Box} - \text{Red Box}}{\text{Blue Box}} \right)$$

VALUE RANGE: $[0, \infty)$ ↑

DESCRIPTION	RECOMMENDED FOR								
LR+ indicates the factor by which a positive prediction occurs more frequently among actual positive samples than among actual negative samples. In a clinical example where the quality of a diagnostic test is to be assessed, this could be interpreted as how much more likely a positive test result is for a diseased person compared to a healthy person (the higher the better).	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px 5px;">ImLC</td> <td style="padding: 2px 5px;">SemS</td> <td style="padding: 2px 5px;">ObD</td> <td style="padding: 2px 5px;">InS</td> </tr> <tr> <td style="text-align: center;"><input checked="" type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> </tr> </table>	ImLC	SemS	ObD	InS	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
ImLC	SemS	ObD	InS						
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>						
	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px 5px;">PREVALENCE DEPENDENCY</td> <td style="text-align: center;"><input type="radio"/></td> </tr> </table>	PREVALENCE DEPENDENCY	<input type="radio"/>						
PREVALENCE DEPENDENCY	<input type="radio"/>								
	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px 5px;">DEFINITION [Attia, 2003]</td> </tr> </table>	DEFINITION [Attia, 2003]							
DEFINITION [Attia, 2003]									

RECOMMENDATIONS

- LR+ should not be considered if...
 - ... predictive values (PPV, NPV) are of interest.
 - ... a cost-benefit-based analysis is desired.
- Otherwise, we recommend it as a per-class counting metric ...
 - ... for a comparison across data sets given its prevalence independence.
 - ... in the presence of class imbalances.
 - ... if the comparison with respect to a naive classifier is desired.

Fig. SN 3.14. Cheat Sheet for the Positive Likelihood Ratio (LR+). The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Positive Predictive Value (PPV), Semantic Segmentation (SemS). Reference used in the figure: Attia, 2003: [6]. We recommend LR+ as a per-class counting metric in Subprocess S3 (Extended Data Fig. 3).

POSITIVE PREDICTIVE VALUE (PPV)

Synonym: Precision

$$PPV = \frac{TP}{TP + FP} = \frac{\text{Green Box}}{\text{Green Box} + \text{Red Box}} \quad \text{if Prevalence} = 0.5$$

$$PPV_{corrected} = \frac{\text{Sensitivity} \cdot \text{Prevalence}}{\text{Sensitivity} \cdot \text{Prevalence} + (1 - \text{Specificity}) \cdot (1 - \text{Prevalence})}$$

VALUE RANGE: [0, 1] ↑

<p>DESCRIPTION</p> <p>PPV represents the probability of a positive prediction corresponding to an actual positive sample.</p> <p>IMPORTANT RELATIONS</p> <p>PPV is the complement of FDR (False discovery rate): $PPV = 1 - FDR$</p> <p>PPV is used as part of many other metrics such as F_β Score and MCC.</p>	RECOMMENDED FOR			
	ImLC	SemS	ObD	InS
	<input checked="" type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
	PREVALENCE DEPENDENCY			<input type="radio"/>
	DEFINITION [Tharwat, 2020]			

RECOMMENDATIONS

- PPV should not be used...
 - ... as a standalone metric but should always be reported together with Sensitivity or NPV.
 - ... for comparisons across data sets with different prevalences or in imbalanced settings, where the prevalence-corrected PPV or alternatively the positive Likelihood ratio (LR+) are more appropriate.
 - ... at image level in case of many images with empty predictions (ObD, InS).
- Otherwise, PPV should especially be considered ...
 - ... in combination with complementary metrics using the concept Metric@(TargetMetric = TargetValue) (e.g., PPV@Sensitivity = 0.95; see glossary)
 - ... if ease of interpretation or popularity are of particular relevance.

Fig. SN 3.15. Cheat Sheet for the Positive Predictive Value (PPV). The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: False Discovery Rate (FDR), False Positive (FP), Image-level Classification (ImLC), Instance Segmentation (InS), Positive Likelihood Ratio (LR+), Matthews Correlation Coefficient (MCC), Negative Predictive Value (NPV), Object Detection (ObD), Semantic Segmentation (SemS), True Positive (TP). Reference used in the figure: Tharwat, 2020: [99]. We recommend PPV as a per-class counting metric in Subprocess S3 (Extended Data Fig. 3) for target value-based optimization using the concept Metric@(TargetMetric = TargetValue) (e.g., PPV@Sensitivity = 0.95; see glossary in Suppl. Note 5.4).

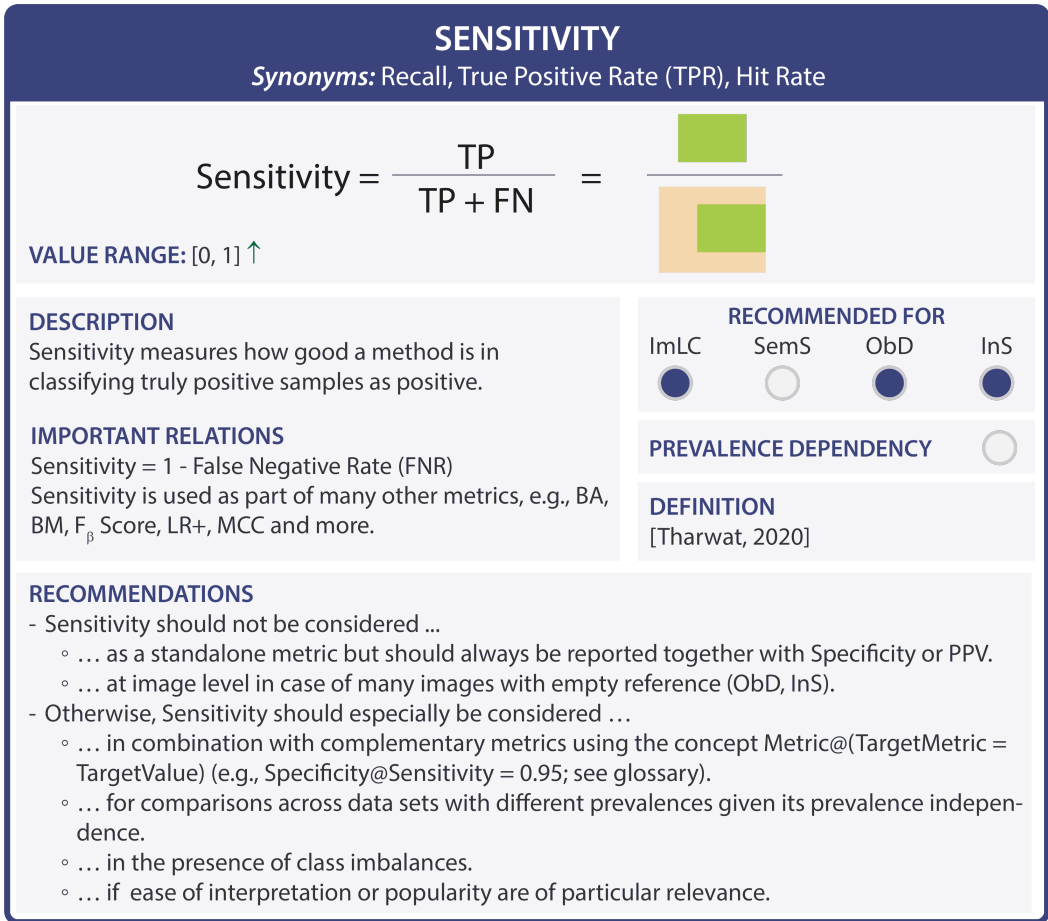


Fig. SN 3.16. Cheat Sheet for the Sensitivity. The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: Bookmaker Informedness (BM), False Negative (FN), Image-level Classification (ImLC), Instance Segmentation (InS), Positive Likelihood Ratio (LR+), Matthews Correlation Coefficient (MCC), Object Detection (ObD), Semantic Segmentation (SemS), True Positive (TP). Reference used in the figure: Tharwat, 2020: [99]. We recommend Sensitivity as a per-class counting metric in Subprocess S3 (Extended Data Fig. 3) for target value-based optimization using the concept Metric@(TargetMetric = TargetValue) (e.g., Specificity@Sensitivity = 0.95; see glossary in Suppl. Note 5.4).

SPECIFICITY

Synonyms: Selectivity, True Negative Rate (TNR)

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} = \frac{\text{[Blue L-shape]}}{\text{[Blue L-shape] + [Red Square]}}$$

VALUE RANGE: [0, 1] ↑

DESCRIPTION
Specificity measures how good a method is in classifying truly negative samples as negative.

IMPORTANT RELATIONS
Specificity = 1 - False Positive Rate
Specificity is used as part of many other metrics, e.g., BA, BM, LR+, MCC and more.

RECOMMENDATIONS

- Specificity should not be used...
 - ... as a standalone metric but always be reported together with Sensitivity or NPV.
 - ... at image level in case of many images with empty reference.
- Otherwise, Specificity should especially be considered ...
 - ... in combination with complementary metrics using the concept Metric@(TargetMetric = TargetValue) (e.g., Specificity@Sensitivity = 0.95; see glossary)
 - ... for comparisons across data sets with different prevalences given its prevalence independence.
 - ... in the presence of class imbalances.
 - ... if ease of interpretation or popularity are of particular relevance.

RECOMMENDED FOR

ImLC	SemS	ObD	InS
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

PREVALENCE DEPENDENCY

DEFINITION
[Tharwat, 2020]

Fig. SN 3.17. Cheat Sheet for the Specificity. The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: Bookmaker Informedness (BM), False Positive (FP), Image-level Classification (ImLC), Instance Segmentation (InS), Positive Likelihood Ratio (LR+), Matthews Correlation Coefficient (MCC), Object Detection (ObD), Semantic Segmentation (SemS), True Negative (TN). Reference used in the figure: Tharwat, 2020: [99]. We recommend Specificity as a per-class counting metric in Subprocess S3 (Extended Data Fig. 3) for target value-based optimization using the concept Metric@(TargetMetric = TargetValue) (e.g., Specificity@Sensitivity = 0.95; see glossary in Suppl. Note 5.4).

WEIGHTED COHEN'S KAPPA (WCK)

Synonyms: Weighted Cohen's Kappa Coefficient, Weighted Kappa Statistic, Weighted Kappa Score

$$WCK = \frac{p_o^w - p_e^w}{1 - p_e^w}, p_o^w = \frac{w_{TP}TP + w_{TN}TN + w_{FP}FP + w_{FN}FN}{TP + TN + FP + FN} = \frac{w_{TP} \cdot \text{green} + w_{TN} \cdot \text{blue} + w_{FP} \cdot \text{red} + w_{FN} \cdot \text{orange}}{\text{green} + \text{blue} + \text{red} + \text{orange}}$$

$$p_e^w = w_{TP} \frac{(TP + FP)(TP + FN)}{TP + TN + FP + FN} + w_{TN} \frac{(TN + FP)(TN + FN)}{TP + TN + FP + FN} + w_{FN} \frac{(FN + FP)(FN + TN)}{TP + TN + FP + FN} + w_{FP} \frac{(FP + TP)(FP + TN)}{TP + TN + FP + FN}$$

$$= w_{TP} \frac{\text{green} \cdot \text{green}}{\text{green} + \text{blue} + \text{red} + \text{orange}} + w_{TN} \frac{\text{blue} \cdot \text{blue}}{\text{green} + \text{blue} + \text{red} + \text{orange}} + w_{FN} \frac{\text{orange} \cdot \text{orange}}{\text{green} + \text{blue} + \text{red} + \text{orange}} + w_{FP} \frac{\text{red} \cdot \text{red}}{\text{green} + \text{blue} + \text{red} + \text{orange}}$$

VALUE RANGE: [-1, 1] ↑
A value of 0 refers to a prediction which is not better than random guessing.
w_{TP}/w_{TN}/w_{FP}/w_{FN}: (estimation of) costs of the respective cardinalities; can be adjusted as a weighting of them.

RECOMMENDED FOR			
ImLC	SemS	ObD	InS
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

DESCRIPTION
WCK calculates the degree of agreement between the reference and prediction while incorporating the agreement resulting from chance. WCK is a generalization of CK with 0-1 weights.

DEFINITION
[Cohen, 1960]

IMPORTANT RELATIONS

- WCK is a generalization of CK = [2 • Prevalence • (1-Prevalence) • (Sensitivity + Specificity - 1)] / [Prevalence² + (1-Prevalence)² + (1-2 • Prevalence) • (Prevalence • Sensitivity - (1-Prevalence) • Specificity)], and equal for 0-1-weights
- For a Prevalence of 50% and weights of 1, WCK = J = 2BA - 1
- Accuracy = (WCK + 1)/2, for 0-1-weights and balanced classes
- BA = (WCK + 1)/2, for 0-1-weights and balanced classes

MULTI-CLASS DEFINITION
For C classes, WCK can be defined as: $WCK = 1 - \frac{(\sum_{i=1}^C \sum_{j=1}^C w_{ij} \cdot n_{ij})}{(\sum_{i=1}^C \sum_{j=1}^C w_{ij} \cdot \frac{n_{i \cdot} \cdot n_{\cdot j}}{N^2})}$

n_{ij}: entry of the confusion matrix for row i and column j, i.e. samples of actual class i that were predicted as class j
n_{i·}: sum of entries of row i of the confusion matrix
n_{·j}: sum of entries of column j of the confusion matrix
w_{ij}: costs for the entry of the confusion matrix for row i and column j, i.e., the cost for samples of actual class i that were predicted as class j
N: total number of samples

RECOMMENDATIONS

- WCK has the rare advantage of being able to incorporate unequal costs for class confusions. However, we generally recommend EC over WCK for this use case due to the former's strong theoretical foundation and straightforward behaviour.
- WCK should generally not be considered in the presence of class imbalance unless class prevalences reflect the interest across classes.

Fig. SN 3.18. Cheat Sheet for the Weighted Cohen's Kappa (WCK). The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: Balanced Accuracy (BA), Cohen's Kappa (CK), False Negative (FN), False Positive (FP), Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Semantic Segmentation (SemS), True Negative (TN), True Positive (TP). Reference used in the figure: Cohen, 1960: [27]. We recommend WCK as a multi-class counting metric in Subprocess S2 (Extended Data Fig. 2).

Multi-threshold metrics

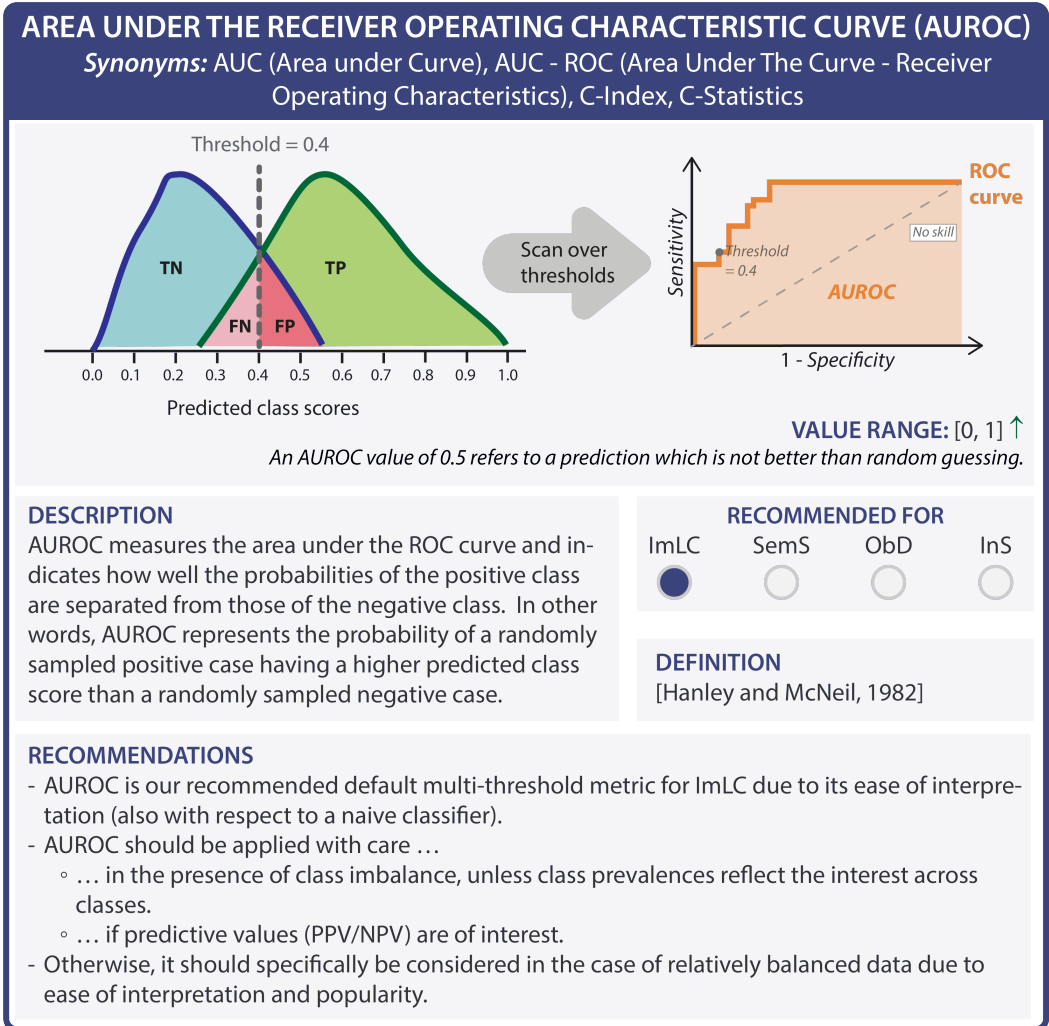


Fig. SN 3.19. Cheat Sheet for the Area under the Receiver Operating Characteristic Curve (AUROC). The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: False Negative (FN), False Positive (FP), Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Receiver Operating Characteristic (ROC), Semantic Segmentation (SemS), True Negative (TN), True Positive (TP). Reference used in the figure: Hanley and McNeil, 1982: [47]. We recommend AUROC as a multi-threshold metric in Subprocess S4 (Extended Data Fig. 4).

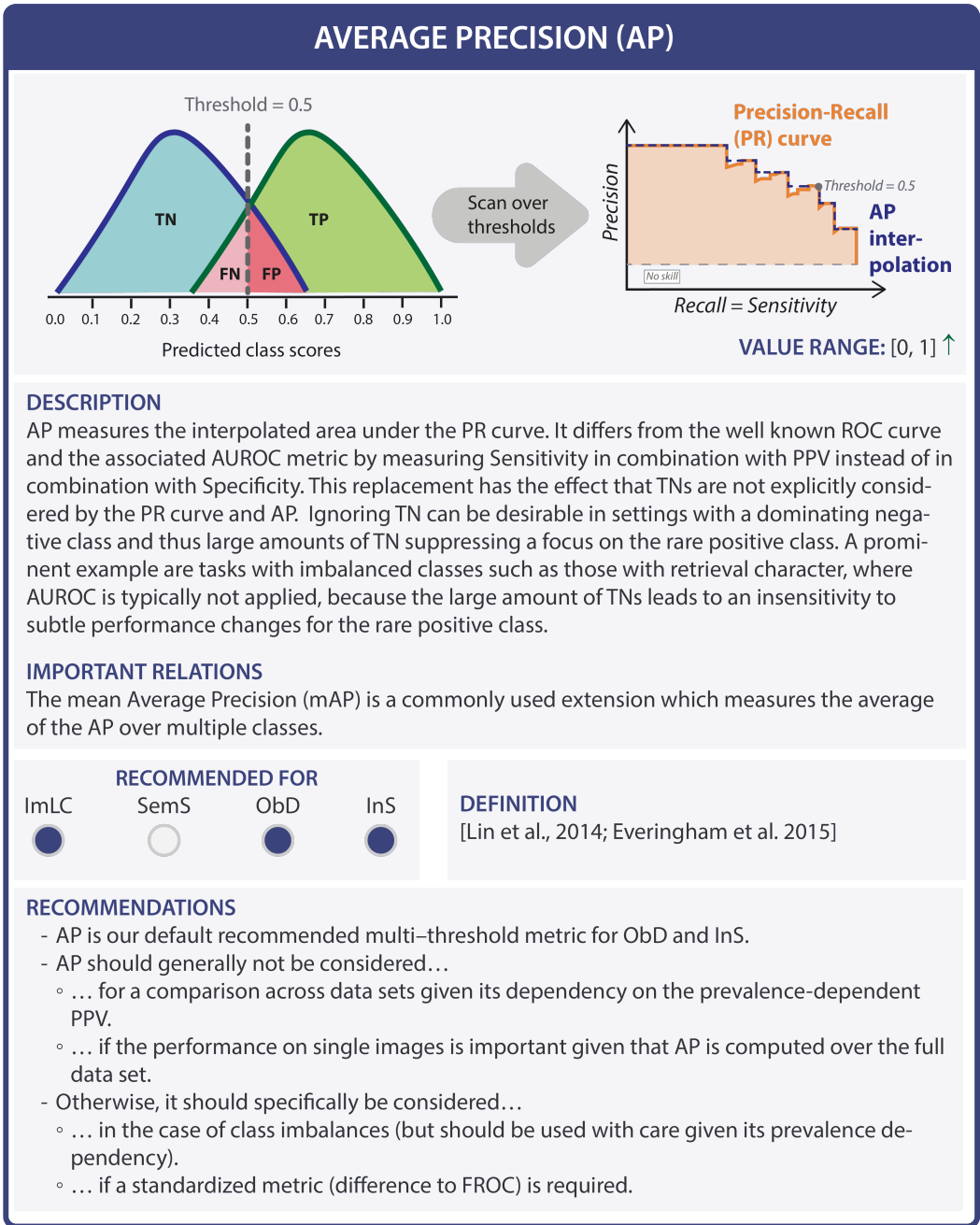


Fig. SN 3.20. Cheat Sheet for the Average Precision (AP). The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: False Negative (FN), False Positive (FP), Image-level Classification (ImLC), Instance Segmentation (InS), Mean Average Precision (mAP), Object Detection (ObD), Positive Predictive Value (PPV), Precision-Recall (PR), Semantic Segmentation (SemS), True Negative (TN), True Positive (TP). References used in the figure: Lin et al., 2014: [64], Everingham et al., 2015: [38]. We recommend AP as a multi-threshold metric in Subprocess S4 (Extended Data Fig. 4).

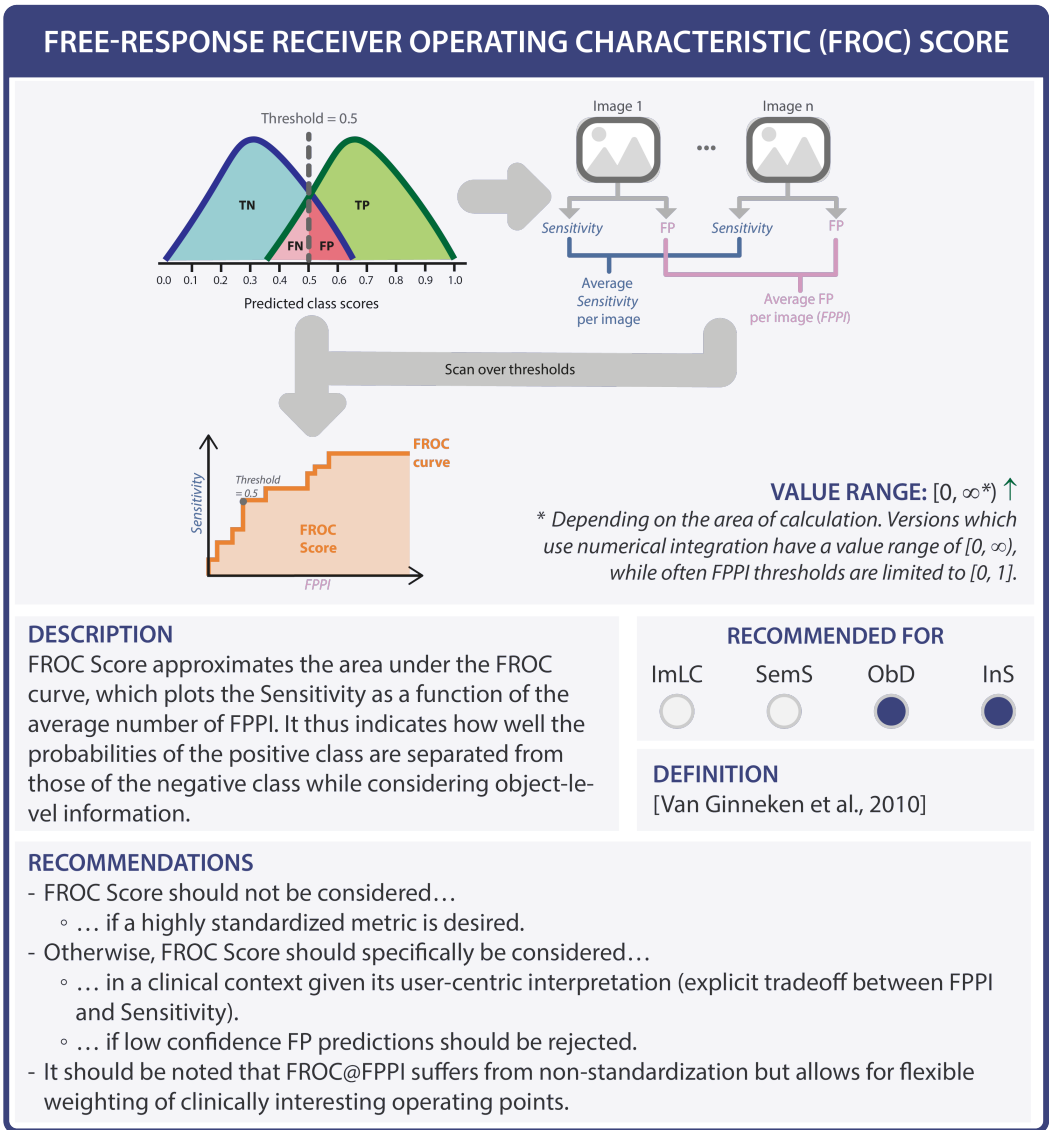
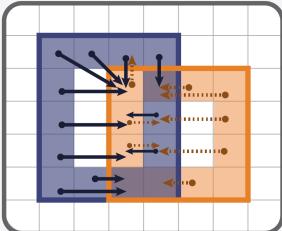


Fig. SN 3.21. Cheat Sheet for the Free-Response Receiver Operating Characteristic (FROC). The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: False Negative (FN), False Positive (FP), False Positives per Image (FPPI), Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Semantic Segmentation (SemS), True Negative (TN), True Positive (TP). Reference used in the figure: Van Ginneken et al., 2010: [108]. We recommend FROC as a multi-threshold metric in Subprocess S4 (Extended Data Fig. 4).

Boundary-based metrics

AVERAGE SYMMETRIC SURFACE DISTANCE (ASSD)

Synonym: Weighted bilateral mean contour distance




→ Min. distances from boundary pixels in A to B

→ Min. distances from boundary pixels in B to A

$$d(a,B) = \min_{b \in B} d(a,b)$$

$$ASSD(A,B) = \frac{\sum_{a \in A} d(a,B) + \sum_{b \in B} d(b,A)}{|A| + |B|}$$



average

VALUE RANGE: $[0, \infty)$ ↓

DESCRIPTION

ASSD measures the average of all shortest boundary distances between contour A to any point on contour B and vice versa, symmetrically.

DEFINITION

[Yeghiazaryan, Varduhi and Voiculescu, 2015]

RECOMMENDED FOR

ImLC	SemS	ObD	InS
<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

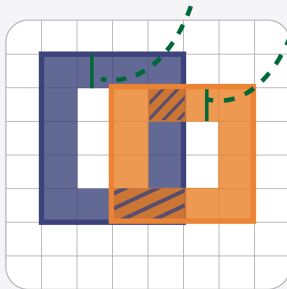
RECOMMENDATIONS

- Boundary-based metrics such as ASSD should generally be reported together with an overlap-based metric.
- We generally recommend MASD over ASSD for boundary-based penalization of spatial outliers with contour focus.
- ASSD should not be used ...
 - ... if the sizes of reference and predictions potentially vary a lot.
 - ... if inter-rater variability should be compensated.
 - ... to compare relationships between boundaries of many dense objects.
- For missing value handling, an advanced strategy should be defined, for example by setting the penalty equal to the largest distance within an image.

Fig. SN 3.22. Cheat Sheet for the Average Symmetric Surface Distance (ASSD). The downward arrow in the value range indicates that lower values are better than higher values. Abbreviations used in the figure: Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Semantic Segmentation (SemS). Reference used in the figure: Yeghiazaryan, Varduhi and Voiculescu, 2015: [119]. We recommend Average Symmetric Surface Distance (ASSD) as a boundary-based metric in Subprocess S7 (Extended Data Fig. 7).

BOUNDARY INTERSECTION OVER UNION (BOUNDARY IOU)

Boundary distance d



■ A_d : Pixels of structure A within width d from boundary

■ B_d : Pixels of structure B within width d from boundary

■ $A_d \cap B_d$

$$\text{Boundary IoU}(A,B) = \frac{|A_d \cap B_d|}{|A_d| + |B_d| - |A_d \cap B_d|} = \frac{|A_d \cap B_d|}{|A_d \cup B_d|}$$

VALUE RANGE: $[0, 1] \uparrow$

DESCRIPTION

Boundary IoU measures the overlap between the predicted and reference boundaries up to a predefined width d .

DEFINITION

[Cheng et al., 2021]

RECOMMENDED FOR

ImLC	SemS	ObD	InS
<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

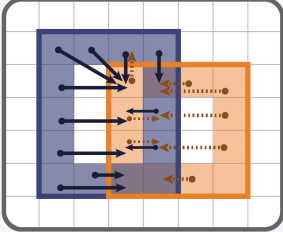
RECOMMENDATIONS

- Boundary-based metrics such as Boundary IoU should generally be reported together with an overlap-based metric.
- Boundary IoU should specifically be considered if...
 - ... structure boundaries are of particular interest.
 - ... boundary errors should be penalized as severe inconsistencies.
 - ... spatial outliers should be penalized by their existence rather than their distance.
- The hyperparameter d influences the Boundary IoU score and denotes the thickness of the considered boundary. It should be chosen according to inter-rater variability, for example. For sufficiently large d , Boundary IoU is equal to Mask IoU.

Fig. SN 3.23. Cheat Sheet for the Boundary Intersection over Union (IoU). The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: Image-level Classification (ImLC), Instance Segmentation (InS), Mean Average Surface Distance (MASD), Object Detection (ObD), Semantic Segmentation (SemS). Reference used in the figure: Cheng et al., 2021: [22]. We recommend Boundary IoU as a boundary-based metric in Subprocess S7 (Extended Data Fig. 7).

MEAN AVERAGE SURFACE DISTANCE (MASD)

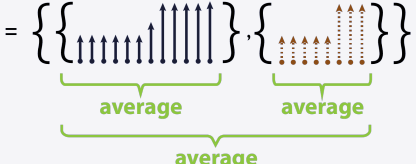
Synonym: Mean Surface Distance



→ Min. distances from boundary pixels in A to B
⋯→ Min. distances from boundary pixels in B to A

$$d(a,B) = \min_{b \in B} d(a,b)$$

$$\text{MASD}(A,B) = \frac{1}{2} \left(\frac{\sum_{a \in A} d(a,B)}{|A|} + \frac{\sum_{b \in B} d(b,A)}{|B|} \right)$$



VALUE RANGE: $[0, \infty) \downarrow$

DESCRIPTION
 MASD measures the mean of the averages over all shortest distances from all sampled points on one boundary to any other point on another boundary.

DEFINITION
 [Beneš and Zitová, 2015]

RECOMMENDED FOR

ImLC	SemS	ObD	InS
<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

RECOMMENDATIONS

- Boundary-based metrics such as MASD should generally be reported together with an overlap-based metric.
- We generally recommend MASD as a boundary-based penalization metric with contour focus over ASSD.
- MASD should not be used if inter-rater variability should be compensated.
- For missing value handling, an advanced strategy should be defined, for example by setting the penalty equal to the largest distance within an image.

Fig. SN 3.25. Cheat Sheet for the Mean Average Surface Distance (MASD). The downward arrow in the value range indicates that lower values are better than higher values. Abbreviations used in the figure: Average Symmetric Surface Distance (ASSD), Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Semantic Segmentation (SemS). Reference used in the figure: Beneš and Zitová, 2015: [11]. We recommend MASD as a boundary-based metric in Subprocess S7 (Extended Data Fig. 7).

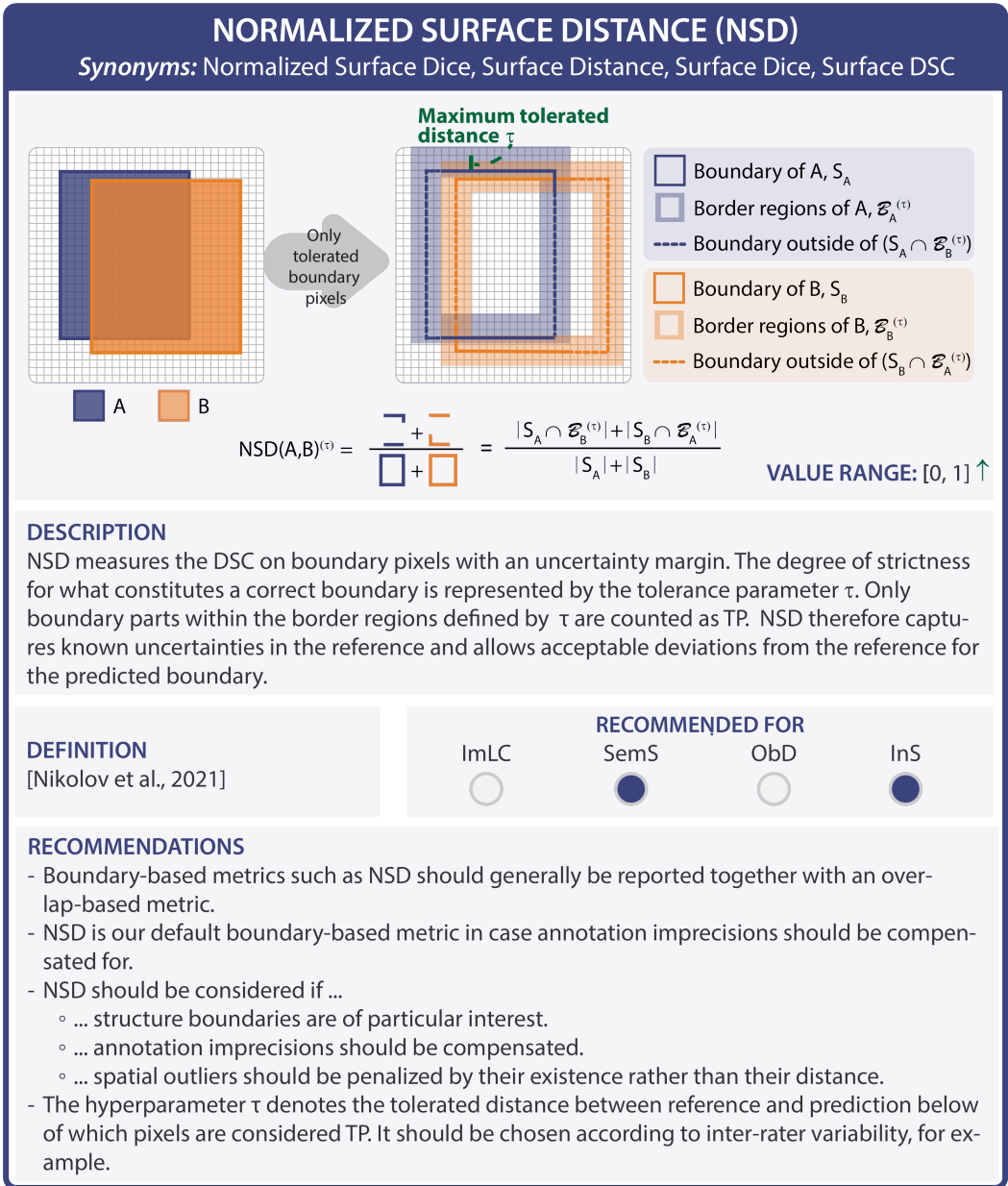


Fig. SN 3.26. Cheat Sheet for the Normalized Surface Distance (NSD). The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: Dice Similarity Coefficient (DSC), Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Semantic Segmentation (SemS), True Positive (TP). Reference used in the figure: Nikolov et al., 2021: [80]. We recommend NSD as a boundary-based metric in Subprocess S7 (Extended Data Fig. 7).

3.1.2 Calibration metrics.

BRIER SCORE (BS)/BRIER SKILL SCORE (BSS)

$$BS = \frac{1}{N} \sum_{i=1}^N \sum_{k=1}^C (p_{ik} - y_{ik})^2$$

VALUE RANGE: [0, 2] ↓

N: number of samples
C: number of classes

p_{ik}: predicted probability for sample *x_i* and class *k*
y_{ik}: outcome; *y_{ik}* = 1 if *y_i* is equal to *k* and 0 otherwise

DESCRIPTION
BS is the mean squared error of a predicted class score and the actual outcome, thus assessing discrimination and calibration in one joint score. It is a proper scoring rule.

VARIANT
Brier Skill Score (BSS): normalizes BS by the BS of a naive system.

DEFINITION
[Gneiting and Raftery, 2007]

RECOMMENDED FOR

ImLC	SemS	ObD	InS
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

TYPE OF CALIBRATION

Top-label	Marginal	Canonical
<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

RECOMMENDATIONS

- BS should be considered if ...
 - ... discrimination and calibration performance should be simultaneously assessed in one score.
 - ... the true posterior probabilities, i.e., the “risks” for individual cases are of interest.
 - ... a straightforward interpretation of relative improvement is desired.
- BS should carefully be used ...
 - ... in imbalanced settings as the bounded penalization of errors leads to preference of naive systems.
 - ... for classes with unequal severity of confusions (e.g. ordinal classes).

Fig. SN 3.28. Cheat Sheet for the Brier Score (BS)/Brier Skill Score (BSS). The downward arrow in the value range indicates that lower values are better than higher values. Abbreviations used in the figure: Brier Skill Score (BSS), Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Semantic Segmentation (SemS). An introduction to calibration and corresponding terminology can be found in Suppl. Note 2.6. Reference used in the figure: Gneiting and Raftery, 2007: [41]. We recommend BS/Brier Skill Score (BSS) as a calibration metric in Subprocess S5 (Extended Data Fig. 5).

CLASS-WISE CALIBRATION ERROR (CWCE)

$$CWCE = \frac{1}{C} \sum_{c=1}^C \sum_{m=1}^M \frac{|B_{c,m}|}{N} \|\text{Accuracy}_c(B_{c,m}) - \text{Confidence}_c(B_{c,m})\|_p^p$$

N: number of samples; *C*: number of classes
B_{c,m}: bin *m* for class *c*
p: determines which *L_p* calibration error is desired; typically *p* = 1

VALUE RANGE: [0, 1] ↓

DESCRIPTION

CWCE is an estimator of the marginal calibration error applying binning to estimate the observed probabilities corresponding to a confidence range. It can be reported per class or in an aggregated fashion with class-specific weights reflecting prevalence or importance of classes, for example.

DEFINITION

[Kull et al., 2019; Kumar et al., 2019]

RECOMMENDED FOR

ImLC	SemS	ObD	InS
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

TYPE OF CALIBRATION

Top-label	Marginal	Canonical
<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>

RECOMMENDATIONS

- CWCE should generally not be considered...
 - ... for small sample sizes (CWCE is dependent on the sample size).
 - ... if the canonical calibration error should be assessed.
- Otherwise, CWCE should be considered...
 - ... for a class-wise calibration assessment.
 - ... in the case of an unequal interest across classes.
- It is generally advisable to report the CWCE both per class and in an aggregated fashion.
- In the case of small sample sizes, the number of bins should be adjusted and additional metrics capturing canonical calibration (KCE or RBS) should be reported as well.

Fig. SN 3.29. Cheat Sheet for the Class-wise Calibration Error (CWCE). The downward arrow in the value range indicates that lower values are better than higher values. Abbreviations used in the figure: Image-level Classification (ImLC), Instance Segmentation (InS), Kernel Calibration Error (KCE), Object Detection (ObD), Root Brier Score (RBS), Semantic Segmentation (SemS). An introduction to calibration and corresponding terminology can be found in Suppl. Note 2.6. References used in the figure: Kumar et al., 2019: [60], Kull et al., 2019: [59]. We recommend CWCE as a calibration metric in Subprocess S5 (Extended Data Fig. 5).

EXPECTED CALIBRATION ERROR (ECE)

$$ECE = \sum_{m=1}^M \frac{|B_m|}{N} \|\text{Accuracy}(B_m) - \text{Confidence}(B_m)\|^p$$

= Weighted Average { }

N: number of samples, B_m : bin m
p: determines which L_p calibration error is desired; typically $p = 1$

VALUE RANGE: [0, 1] ↓

DESCRIPTION

ECE is an estimator for the L_p top-label calibration error. For a binned estimation, it is the weighted average of the absolute difference between the average predicted class score (Confidence) of the top label per bin B_m and the corresponding fraction of correct predictions (Accuracy).

VARIANT

The marginal variant of ECE is CWCE.

DEFINITION
[Naeini et al., 2015]

RECOMMENDED FOR

ImLC	SemS	ObD	InS
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

TYPE OF CALIBRATION

Top-label	Marginal	Canonical
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>

RECOMMENDATIONS

- ECE should generally not be considered...
 - ... for small sample sizes (ECE is dependent on the sample size).
 - ... if the canonical or marginal calibration error should be assessed.
- Otherwise, ECE should be considered...
 - ... if the top-label calibration error should be assessed (especially in binary settings, where it equals the marginal and canonical calibration error).
- ECE should be reported together with RBS, which gives an unbiased upper bound of the canonical calibration error.

Fig. SN 3.30. Cheat Sheet for the Expected Calibration Error (ECE). The downward arrow in the value range indicates that lower values are better than higher values. Abbreviations used in the figure: Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Semantic Segmentation (SemS). An introduction to calibration and corresponding terminology can be found in Suppl. Note 2.6. Reference used in the figure: Naeini et al., 2015: [74]. We recommend ECE as a calibration metric in Subprocess S5 (Extended Data Fig. 5).

EXPECTED CALIBRATION ERROR KERNEL DENSITY ESTIMATE (ECE^{KDE})

$$ECE^{KDE} = \frac{1}{N} \sum_{j=1}^N \left\| \frac{\sum_{i \neq j} k(f(x_j), f(x_i)) e_{y_i}}{\sum_{i \neq j} k(f(x_j), f(x_i))} - f(x_j) \right\|_p^p$$

N: number of samples
k: kernel, e.g. Dirichlet kernel [Popordanoska et al., 2022]
f(*x*): predicted probability vector, *y_i*: outcome (one-hot encoded)
e_y: *C*-dimensional vector with *y_i*-th entry being 1, else 0
p: determines which *L_p* calibration error is desired; typically *p* ∈ {1, 2}

VALUE RANGE: [0, 2] ↓

<p>DESCRIPTION</p> <p>ECE^{KDE} is an estimator for the canonical calibration error. It uses a kernel density estimate in contrast to the binning strategy applied by the standard ECE.</p> <p>DEFINITION</p> <p>[Popordanoska et al., 2022]</p>	<p style="text-align: center; font-weight: bold; color: #1a3d54;">RECOMMENDED FOR</p> <table style="width: 100%; text-align: center;"> <tr> <td>ImLC</td> <td>SemS</td> <td>ObD</td> <td>InS</td> </tr> <tr> <td><input checked="" type="radio"/></td> <td><input type="radio"/></td> <td><input type="radio"/></td> <td><input type="radio"/></td> </tr> </table> <p style="text-align: center; font-weight: bold; color: #1a3d54;">TYPE OF CALIBRATION</p> <table style="width: 100%; text-align: center;"> <tr> <td>Top-label</td> <td>Marginal</td> <td>Canonical</td> </tr> <tr> <td><input type="radio"/></td> <td><input type="radio"/></td> <td><input checked="" type="radio"/></td> </tr> </table>	ImLC	SemS	ObD	InS	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Top-label	Marginal	Canonical	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
ImLC	SemS	ObD	InS												
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>												
Top-label	Marginal	Canonical													
<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>													

RECOMMENDATIONS

- ECE^{KDE} should generally not be considered...
 - ... for small sample sizes (ECE^{KDE} is dependent on sample size).
 - ... for very large numbers of classes or unequal interest across classes. In such cases, CWCE should be considered instead.
- Otherwise, ECE^{KDE} should be considered...
 - ... for quantifying the canonical calibration error.
- ECE^{KDE} should be reported together with RBS, which gives an unbiased upper bound on the canonical calibration error.

Fig. SN 3.31. Cheat Sheet for the Expected Calibration Error Kernel Density Estimate (ECE^{KDE}). The downward arrow in the value range indicates that lower values are better than higher values. Abbreviations used in the figure: Class-wise Calibration Error (CWCE), Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Root Brier Score (RBS), Semantic Segmentation (SemS). An introduction to calibration and corresponding terminology can be found in Suppl. Note 2.6. Reference used in the figure: Popordanoska et al., 2022: [83]. We recommend ECE^{KDE} as a calibration metric in Subprocess S5 (Extended Data Fig. 5).

KERNEL CALIBRATION ERROR (KCE)

$$\text{KCE} = \left(\mathbb{E} \left((e_y - f(x))^T k(f(x), f(x')) (e_y - f(x')) \right) \right)^{1/2}$$

Example estimator: $\widehat{\text{KCE}} = \left(\binom{N}{2}^{-1} \sum_{i=1}^N \sum_{j=i+1}^N (e_{y_i} - f(x_i))^T k(f(x_i), f(x_j)) (e_{y_j} - f(x_j)) \right)^{1/2}$

N: number of samples; *k*: matrix-valued kernel; *f*(*x*): predicted probability vector;
y_i: outcome; *e_{y_i}*: *C*-dimensional vector with *y_i*-th entry being 1, else 0

VALUE RANGE: Kernel dependent; in expectation > 0 but estimator can be arbitrarily negative

DESCRIPTION

KCE measures a canonical calibration error based on an alternative distance function, the “maximum mean discrepancy” (MMD). It is based on a matrix-valued kernel *k*.

KCE is an unbiased estimator of the calibration error measured by MMD.

DEFINITION

[Widmann et al., 2019; Gruber and Buettner, 2022]

RECOMMENDED FOR

ImLC	SemS	ObD	InS
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

TYPE OF CALIBRATION

Top-label	Marginal	Canonical
<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

RECOMMENDATIONS

- KCE should generally not be considered...
 - ... for unequal interest across classes. In such a case, CWCE should be considered instead.
 - ... for interpreting calibration performance (absolute values) as it is relatively hard to interpret (for example compared to BS), also due to negative output values.
- KCE should be considered...
 - ... for a canonical calibration assessment if interpretability of absolute values is not of key interest. This holds especially true for comparative calibration assessment as KCE is an unbiased estimator of the calibration error measured by MMD.
 - ... in the presence of small sample sizes and a large number of classes as it is an unbiased estimator and therefore also well-suited.
- KCE should be configured carefully as it depends on nontrivial configuration choices of kernels and associated hyperparameters.

Fig. SN 3.32. Cheat Sheet for the Kernel Calibration Error (KCE). The downward arrow in the value range indicates that lower values are better than higher values. Abbreviations used in the figure: Brier Score (BS), Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Semantic Segmentation (SemS). An introduction to calibration and corresponding terminology can be found in Suppl. Note 2.6. References used in the figure: Gruber and Buettner, 2022: [43], Widmann et al., 2019: [115]. We recommend KCE as a calibration metric in Subprocess S5 (Extended Data Fig. 5).

ROOT BRIER SCORE (RBS)

$$RBS = \sqrt{\frac{1}{N} \sum_{i=1}^N \sum_{k=1}^C (p_{ik} - y_{ik})^2}$$

N: number of samples
C: number of classes

VALUE RANGE: $[0, \sqrt{2}] \downarrow$

DESCRIPTION

RBS is the square root of the mean squared error of a predicted class score and the actual outcome.

It represents a robust upper bound of the canonical calibration error.

RECOMMENDED FOR

ImLC	SemS	ObD	InS
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

DEFINITION

[Gruber and Buettner, 2022]

TYPE OF CALIBRATION

Top-label	Marginal	Canonical
<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

RECOMMENDATIONS

RBS should be considered as a guaranteed upper bound of the canonical calibration error and should be reported together with ECE/ECE^{KDE}.

Fig. SN 3.34. Cheat Sheet for the Root Brier Score (RBS). The downward arrow in the value range indicates that lower values are better than higher values. Abbreviations used in the figure: Expected Calibration Error (ECE), Expected Calibration Error Kernel Density Estimate (ECE^{KDE}), Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Semantic Segmentation (SemS). An introduction to calibration and corresponding terminology can be found in Suppl. Note 2.6. Reference used in the figure: Gruber and Buettner, 2022: [43]. We recommend RBS as a calibration metric in Subprocess S5 (Extended Data Fig. 5).

3.1.3 Localization criteria.

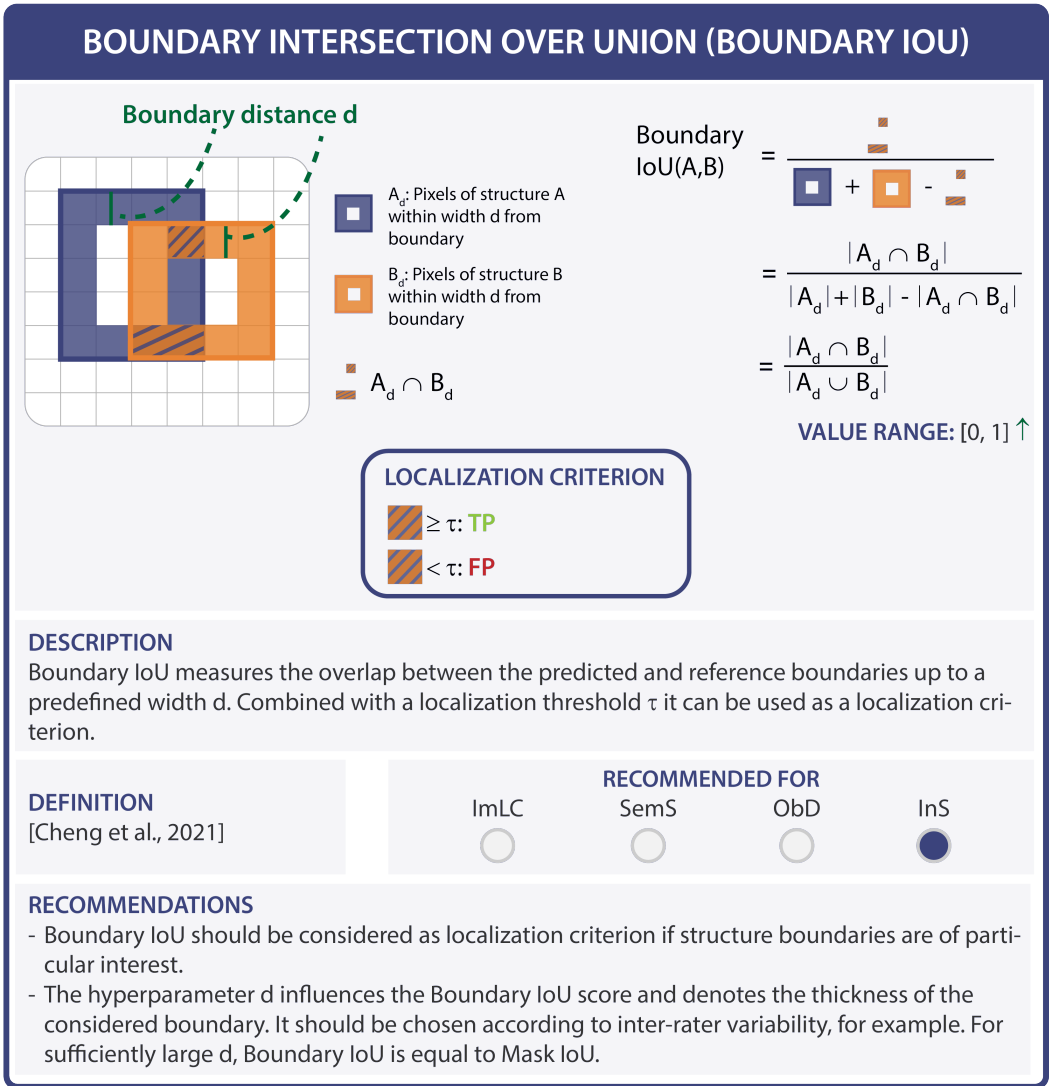


Fig. SN 3.35. Metric profile of the Boundary IoU (localization criterion). The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Semantic Segmentation (SemS). Reference used in the figure: Cheng et al., 2021: [22]. We recommend Boundary Intersection over Union (IoU) as a localization criterion in Subprocess S8 (Extended Data Fig. 8).

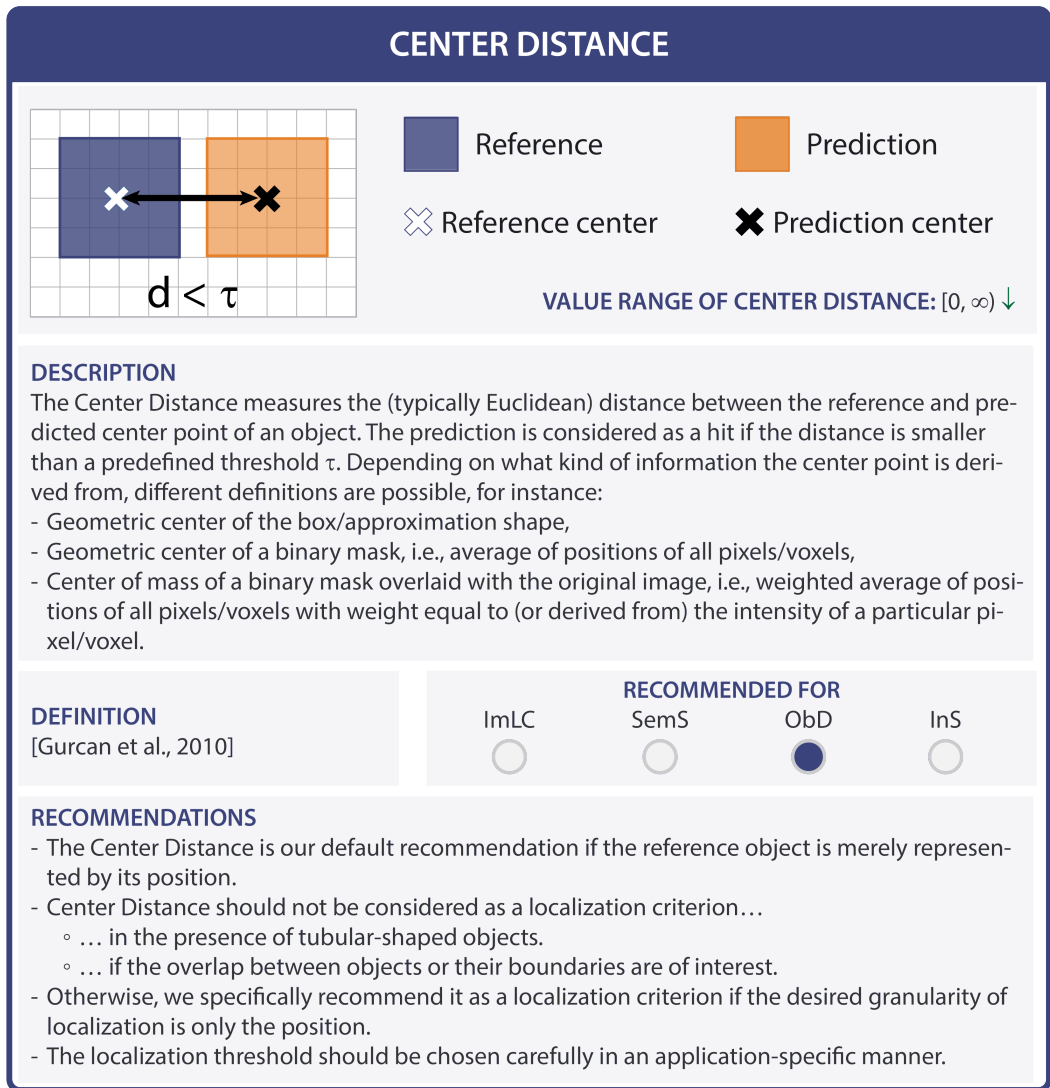
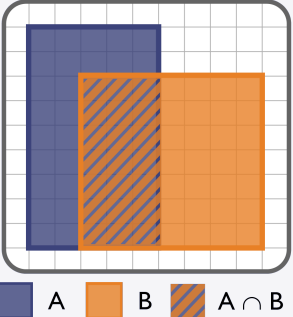


Fig. SN 3.36. Cheat Sheet for the Center Distance. The downward arrow in the value range indicates that lower values are better than higher values. Abbreviations used in the figure: False Negative (FN), False Positive (FP), Image-level Classification (ImLC), Intersection over Union (IoU), Instance Segmentation (InS), Object Detection (ObD), Semantic Segmentation (SemS). Reference used in the figure: Gurcan et al., 2010: [46]. We recommend Center Distance as a localization criterion in Subprocess S8 (Extended Data Fig. 8).

INTERSECTION OVER REFERENCE (IoR)

Synonyms: Pixel-level Sensitivity



$$\text{IoR}(A,B) = \frac{\text{diagonally striped}}{\text{blue}} = \frac{|A \cap B|}{|A|}$$

VALUE RANGE: [0, 1] ↑

LOCALIZATION CRITERION

$\geq \tau$: **TP**

$< \tau$: **FP**

DESCRIPTION

IoR measures the overlap between two structures. It is defined as the pixel-level Sensitivity and only considers the FN pixels (not the FPs). The metric is rather uncommon for segmentation assessment, but combined with a localization τ threshold it can be used as a localization criterion.

DEFINITION

[Maska et al., 2014]

RECOMMENDED FOR

ImLC	SemS	ObD	InS
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

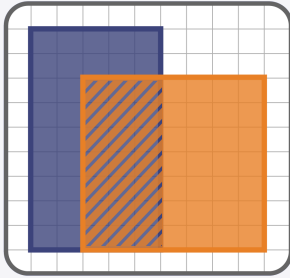
RECOMMENDATIONS

- IoR is a rather uncommon metric/localization criterion, which we do not generally recommend as it can yield extremely misleading results in the case of large predictions.
- IoR should be preferred over IoU as localization criterion in InS in the case of touching structures, in which one prediction may overlap multiple reference objects. In this case, assignment of one prediction to multiple reference objects must be enabled and a penalty for this type of algorithm error (“non-split error”) introduced (see Fig. SN 2.26).

Fig. SN 3.37. Cheat Sheet for the Intersection over Reference (IoR). The upward arrow in the value range indicates that higher values are better than lower values. Abbreviations used in the figure: Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Semantic Segmentation (SemS). Reference used in the figure: Maška et al., 2014: [68]. We recommend IoR as a localization criterion in Subprocess S8 (Extended Data Fig. 8).

MASK/BOX/APPROX INTERSECTION OVER UNION (MASK/BOX/APPROX IoU)

Synonyms: Jaccard Index, Tanimoto Coefficient



■ A ■ B ■ A ∩ B

$$\begin{aligned} \text{IoU}(A,B) &= \frac{\text{Area of } A \cap B}{\text{Area of } A + \text{Area of } B - \text{Area of } A \cap B} \\ &= \frac{|A \cap B|}{|A| + |B| - |A \cap B|} = \frac{|A \cap B|}{|A \cup B|} \\ &= \frac{\text{PPV} \cdot \text{Sensitivity}}{\text{PPV} + \text{Sensitivity} - \text{PPV} \cdot \text{Sensitivity}} \end{aligned}$$

LOCALIZATION CRITERION

- $\geq \tau$: TP
- $< \tau$: FP

DESCRIPTION

IoU measures the overlap between two structures (see above). Combined with a localization threshold, it is a common localization criterion. It is often referred to as **Box IoU** when comparing bounding boxes, **Mask IoU** when comparing segmentation masks, or **Approx IoU** when comparing approximations of objects beyond bounding boxes.

DEFINITION

[Jaccard, 1912]

RECOMMENDED FOR

ImLC	SemS	ObD	InS
<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>

IMPORTANT RELATIONS

$$\text{IoU} = \frac{\text{DSC}}{2 - \text{DSC}} \quad \text{IoU} = \frac{F_\beta}{2 - F_\beta} \quad \text{for } \beta = 1$$

RECOMMENDATIONS

- IoU is our default recommended localization criterion if a rough outline of the target structures is desired (in contrast to scenarios in which only the position is of importance).
- IoU should not be used ...
 - ... if contour agreement is important for deciding on a match between predicted and reference object.
 - ... to approximate disconnected or tubular structures as boxes (Box IoU).
- Otherwise, it is specifically well-suited if overlap is a meaningful measure of how well an object has been located.
- The localization threshold should be chosen carefully in an application-specific manner.

Fig. SN 3.38. Metric profile of the Mask/Box/Approx Intersection over Union (IoU) (localization criterion). Abbreviations used in the figure: Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Positive Predictive Value (PPV), Semantic Segmentation (SemS). Reference used in the figure: Jaccard, 1912: [52]. We recommend Mask/Box/Approx IoU as a localization criterion in Subprocess S8 (Extended Data Fig. 8).

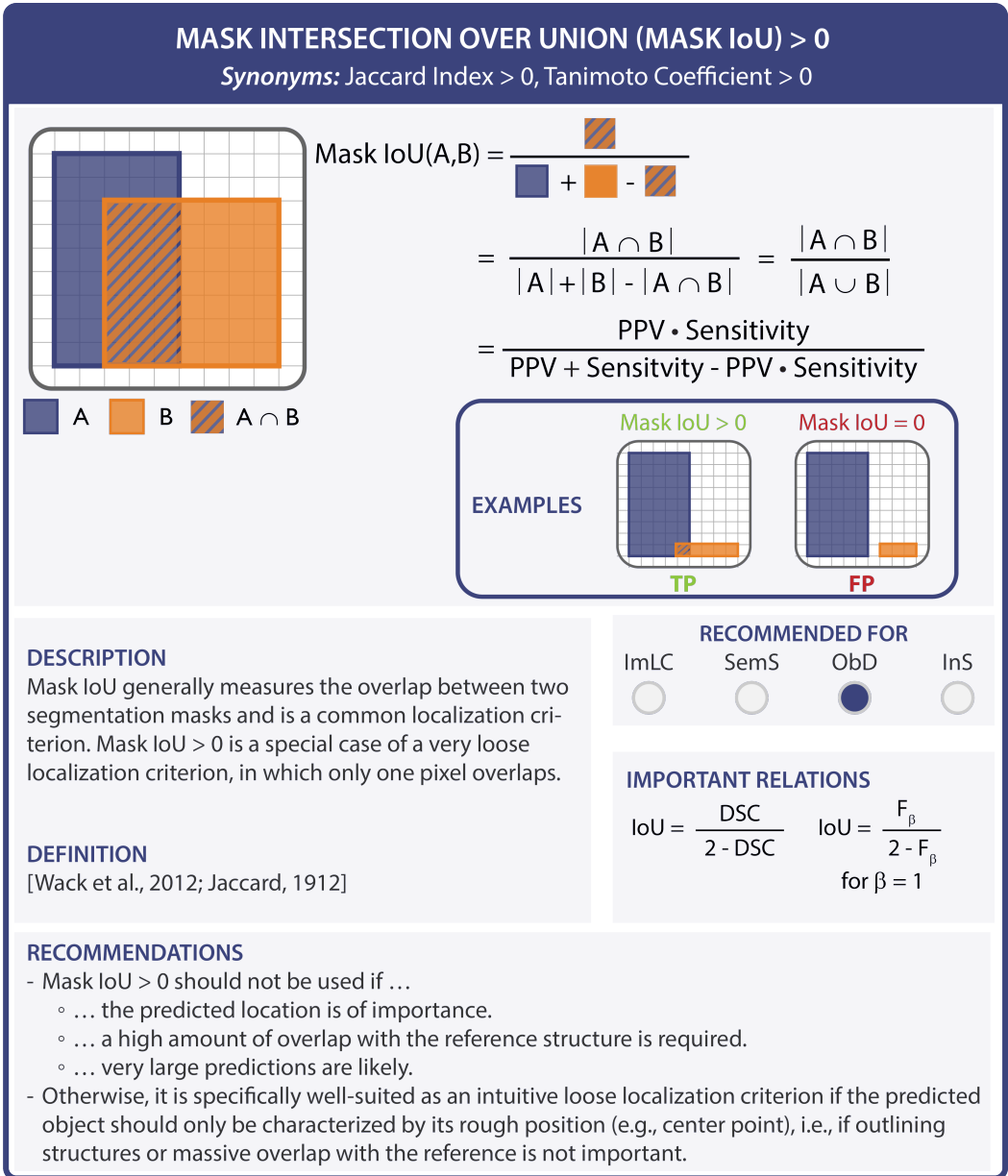
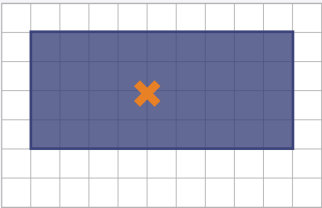


Fig. SN 3.39. Metric profile of the Mask Intersection over Union (IoU) > 0. Abbreviations used in the figure: False Positive (FP), Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Positive Predictive Value (PPV), Semantic Segmentation (SemS). References used in the figure: Jaccard, 1912: [52], Wack et al., 2012: [112]. We recommend Mask IoU > 0 as a localization criterion in Subprocess S8 (Extended Data Fig. 8).

POINT INSIDE MASK/BOX/APPROXIMATION



Reference

X

Predicted point

VALUE RANGE: {True, False}

DESCRIPTION

The Point inside Mask/Box/Approximation is a localization criterion that defines a point-based prediction as a hit as long as it is inside the reference object, which may be a mask, bounding box, or other approximation of a structure.

DEFINITION

<https://cada.grand-challenge.org/Assessment/>

RECOMMENDED FOR

ImLC	SemS	ObD	InS
<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>

RECOMMENDATIONS

- The Point inside Mask/Box/Approximation criterion should be considered as a localization criterion...
 - ... if only a rough estimate of the object location is required.
 - ... for complex shapes, e.g., tubular objects.
- It should be noted that the Point inside Mask/Box/Approximation criterion does not allow to adjust the localization strictness. If such a property is desired, a different localization criterion should be chosen, such as Box/Approx IoU.

Fig. SN 3.40. Cheat Sheet for the Point inside Mask/Box/Approximation. Abbreviations used in the figure: Dice Similarity Coefficient (DSC), Image-level Classification (ImLC), Instance Segmentation (InS), Object Detection (ObD), Positive Predictive Value (PPV), Semantic Segmentation (SemS). Reference used in the figure: <https://cada.grand-challenge.org/Assessment/>. We recommend Point inside Mask/Box/Approximation as a localization criterion in Subprocess S8 (Extended Data Fig. 8).

3.1.4 Assignment strategies.

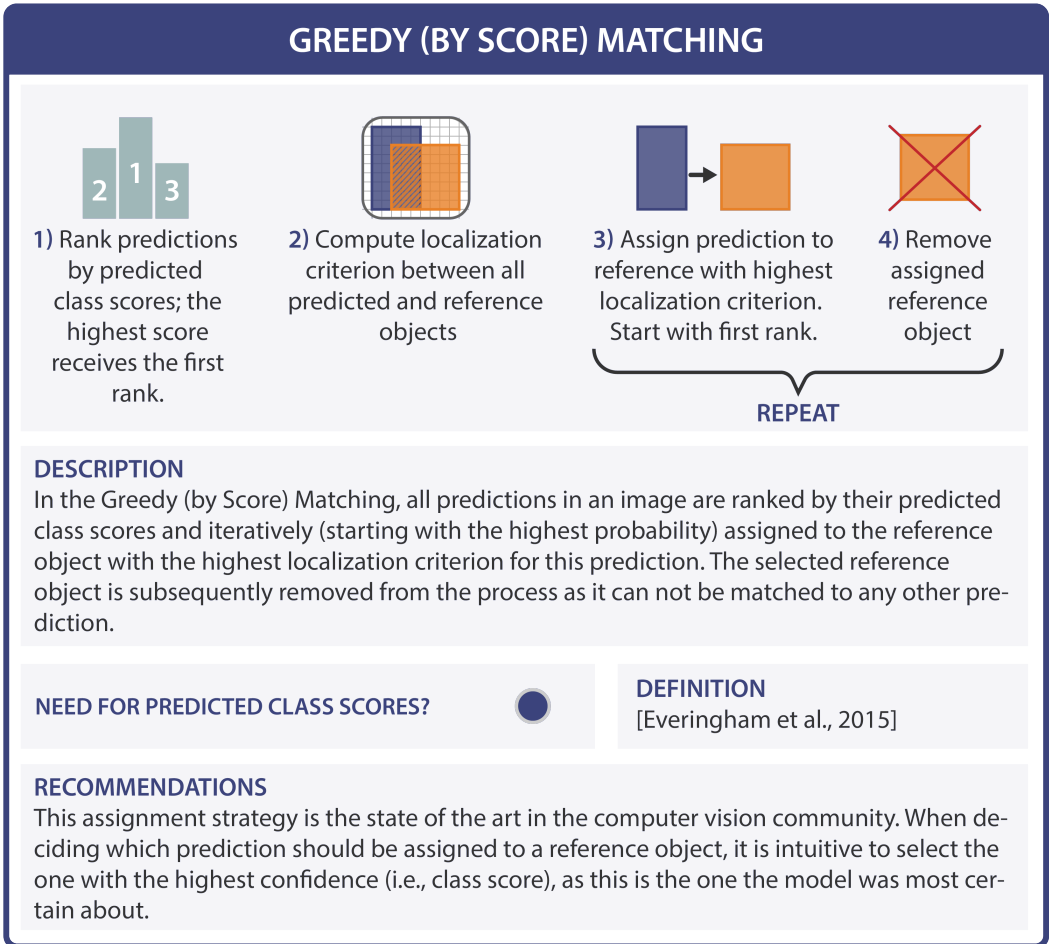


Fig. SN 3.41. Cheat Sheet for the Greedy (by Score) Matching. Reference used in the figure: Everingham et al., 2015: [39]. We recommend Greedy (by Score) Matching as an assignment strategy in Subprocess S9 (Extended Data Fig. 9).

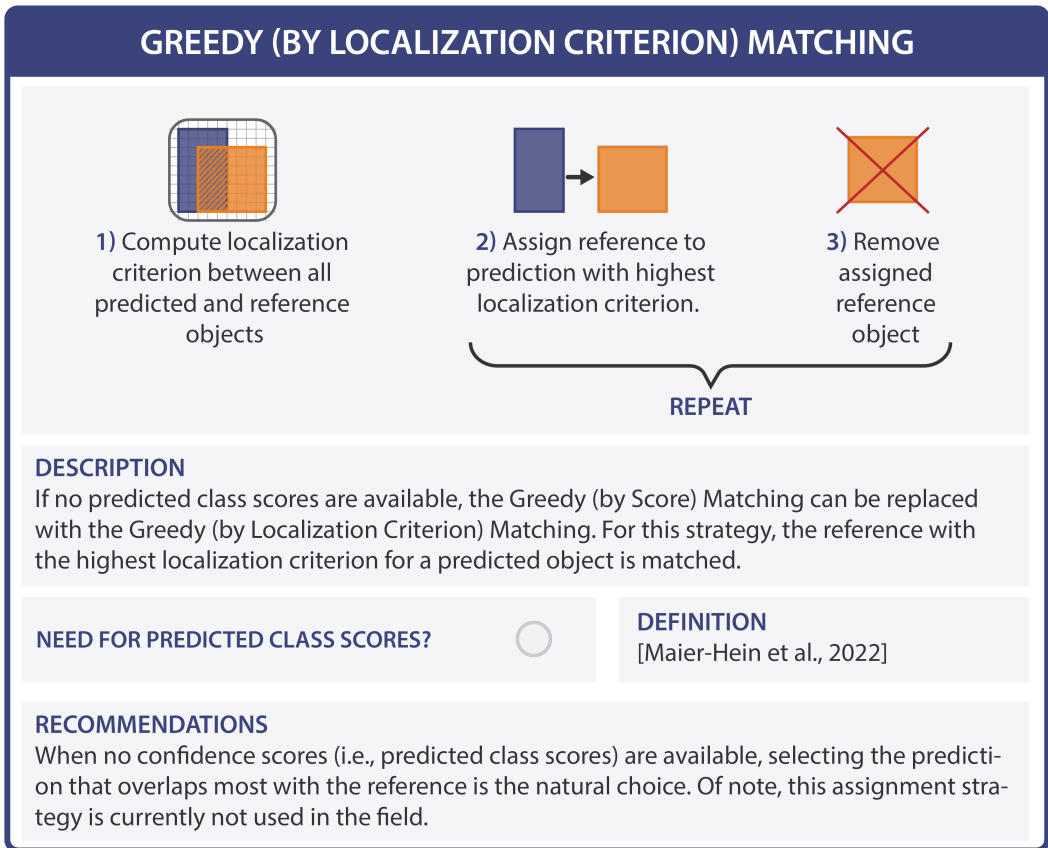

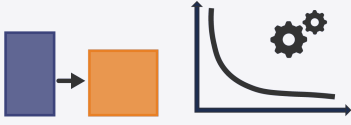


Fig. SN 3.42. Cheat Sheet for the Greedy (by Localization Criterion) Matching. Reference used in the figure: Maier-Hein et al., 2022: [66]. We recommend Greedy (by Localization Criterion) Matching as an assignment strategy in Subprocess S9 (Extended Data Fig. 9).

OPTIMAL (HUNGARIAN) MATCHING



1) Compute localization criterion between all predicted and reference objects



2) Use cost function to find the optimal assignment of predictions and references based on the localization criterion.

DESCRIPTION
The Optimal (Hungarian) Matching is associated with a cost function, usually depending on the localization criterion, which is minimized to find the optimal assignment of predictions and reference.

NEED FOR PREDICTED CLASS SCORES?


DEFINITION
[Kuhn, 1955]

RECOMMENDATIONS
If the application at hand provides a dedicated cost function as to which assignments should be penalized, this strategy is the natural choice. In most cases, however, an adequate cost function would need to be defined by the user which can often not be justified given the available, more intuitive assignment strategies.


Fig. SN 3.43. Cheat Sheet for the Optimal (Hungarian) Matching. Reference used in the figure: Kuhn et al., 1955: [58]. We recommend the Optimal (Hungarian) Matching as an assignment strategy in Subprocess S9 (Extended Data Fig. 9).

MATCHING VIA OVERLAP > 0.5


PREREQUISITE: Overlapping predictions are not possible.



1) Compute overlap-based localization criterion between all predicted and reference objects



2) If the overlap is greater than 0.5, assign prediction to the reference.



3) Remove assigned reference object

} REPEAT

DESCRIPTION

If there are no overlapping predictions, complex assignment strategies can be avoided by simply setting the localization criterion to $\text{IoU} > 0.5$. This strategy inherently avoids matching conflicts, because any secondary prediction would by definition have an overlap < 0.5 of the same reference object.

NEED FOR PREDICTED CLASS SCORES?

DEFINITION
[Everingham et al., 2006]

RECOMMENDATIONS

- This assignment strategy should not be applied if ...
 - ... overlapping predictions are possible.
 - ... a non-overlap based criterion is employed.
 - ... an IoU threshold lower than 0.5 is requested for localization.
- Otherwise, it represents a simple and intuitive localization criterion that inherently avoids matching conflicts and thus the need for a dedicated assignment strategy. This criterion is often used in the cell segmentation domain.

Fig. SN 3.44. Cheat Sheet for the Matching via Overlap > 0.5 . Reference used in the figure: Everingham et al., 2006: [37]. We recommend Matching via Overlap > 0.5 as an assignment strategy in Subprocess S9 (Extended Data Fig. 9).

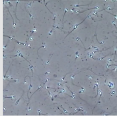

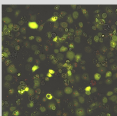

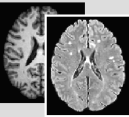

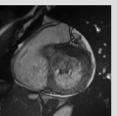
SUPPL. NOTE 4 RECOMMENDATIONS FOR SELECTED USE CASES

We instantiated the framework for several biological and medical image analysis use cases. The list of use cases with a link to the figures representing the recommendations is provided below:

4.1 Image-level classification

The following use cases have been instantiated for image-level classification problems. The resulting metric recommendations can be found in Fig. SN 4.1, while Figs. SN 4.5-SN 4.7 provide a detailed overview of the recommendations for the use cases in the metric selection Subprocesses S2-S5.

- ImLC-1** Frame-based sperm motility classification from microscopy time-lapse video of human spermatozoa [49]
- ImLC-2** Disease classification in dermoscopic images [26, 104]
- ImLC-3** Classification of the overall autophagy stage for a collection of cells [75, 121]
- ImLC-4** Diagnostic standard plane classification in ultrasound images [9]
- ImLC-5** Identification of new lesions in brain multi-modal magnetic resonance imaging (MRI) images of patients with multiple sclerosis (MS) [29, 57]
- ImLC-6** Breast cancer classification in mammography images [62]
- ImLC-7** Multi-class cardiac disease classification in MRI images [13]

IMAGE-LEVEL CLASSIFICATION				
ID	SCENARIO	SAMPLE INPUT IMAGE	POTENTIAL OUPUT	RECOMMENDATION
ImLC-1	Frame-based sperm motility classification from microscopy time-lapse video of human spermatozoa		Progressive motility: 0.5 Non-progressive motility: 0.4 Immotile: 0.1	Multi-class counting metric (S2): BA Per-class counting metric (S3): LR+ Multi-threshold metric (S4): AUROC Calibration metric (S5): ECE (top-label) and RBS
ImLC-2	Disease classification in dermoscopic images		Dermatofibroma: 0.6 Melanocytic nevus: 0.2 Melanoma: 0.1 Basal cell carcinoma: 0.0 Actinic keratosis: 0.0 Benign keratosis: 0.0 Vascular lesion: 0.1	
ImLC-3	Classification of overall autophagy stage for a collection of cells		Sequestration: 0.7 Transport to lysosomes: 0.2 Degradation: 0.1 Utilization of degradation products: 0.0	Multi-class counting metric (S2): MCC Per-class counting metric (S3): LR+ Multi-threshold metric (S4): AUROC No calibration metric (S5)
ImLC-4	Diagnostic standard plane classification in ultrasound images		Spine (sag.): 0.65 Background: 0.165, Femur: 0.01, 3VV: 0.01, Spine (cor): 0.05, RVOT: 0.05, LVOT: 0.05	No multi-class counting metric (S2) <i>(only per-class validation)</i> Multi-threshold metric (S3): AUROC Per-class counting metric (S4): Sensitivity@Specificity Calibration metric (S5) needed if used in interactive imaging guidance mode: CWCE
ImLC-5	Identification of new lesions in brain multi-modal MRI images of patients with MS		Lesion: 0.9 No lesion: 0.1	Multi-class counting metric (S2): EC Per-class counting metric (S3): F_{β} Score Multi-threshold metric (S4): AP Calibration metric (S5): BS (for comparative calibration assessment and assessment of interpretability of model outputs)
ImLC-6	Breast cancer classification in mammography images		Cancer: 0.6 No cancer: 0.4	No multi-class counting metric (S2) <i>(only per-class validation)</i> Per-class counting metric (S3): NB Multi-threshold metric (S4): AP Calibration metric (S5): BS (for comparative calibration assessment)
ImLC-7	Multi-class cardiac disease classification in MRI images		Normal case: 0.5, Heart failure with infarction: 0.3, Dilated cardiomyopathy: 0.0, Hypertrophic cardiomyopathy: 0.1, Abnormal right ventricle: 0.1	Multi-class counting metric (S2): Accuracy Per-class counting metric (S3): Sensitivity Multi-threshold metric (S4): AUROC No calibration metric (S5)

ABBREVIATIONS
AP Average Precision

AUROC Area Under the Receiver Operating Characteristic Curve

BA Balanced Accuracy

BS Brier Score

CWCE Class-wise Calibration Error

EC Expected Cost

ECE Expected Calibration Error

LR+ Positive Likelihood Ratio

MCC Matthews Correlation Coefficient

RBS Root Brier Score

Fig. SN 4.1. **Instantiation of the framework with recommendations for concrete biomedical image-level classification problems.** **(ImLC-1)** Frame-based sperm motility classification from microscopy time-lapse video of human spermatozoa [49]. **(ImLC-2)** Disease classification in dermoscopic images [26, 104]. **(ImLC-3)** Classification of the overall autophagy stage for a collection of cells [75, 121]. **(ImLC-4)** Diagnostic standard plane classification in ultrasound images [9]. **(ImLC-5)** Identification of new lesions in brain multi-modal magnetic resonance imaging (MRI) images of MS patients [29, 57]. **(ImLC-6)** Breast cancer classification in mammography images [62]. **(ImLC-7)** Multi-class cardiac disease classification in MRI images [13].

4.2 Semantic segmentation

The following use cases have been instantiated for semantic segmentation problems. The resulting metric recommendations can be found in Fig. SN 4.2, while Figs. SN 4.10-SN 4.11 provide a detailed overview of the recommendations for the use cases in the metric selection Subprocesses S6 and S7.

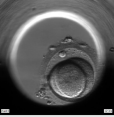
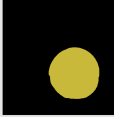
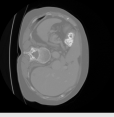

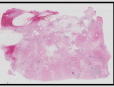
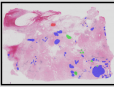
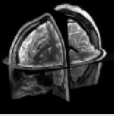
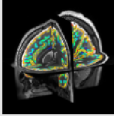
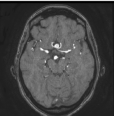
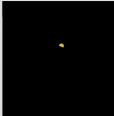
SemS-1 Embryo segmentation from microscopy images [98]

SemS-2 Liver segmentation in computed tomography (CT) images [1, 94]

SemS-3 Labeling of invasive/non-invasive/benign lesions in breast whole slide imaging (WSI) [2]

SemS-4 Cortical structure segmentation from 3D MRI images [19]

SemS-5 Aneurysm segmentation in time-of-flight magnetic resonance angiography (TOF-MRA) images [100]

SEMANTIC SEGMENTATION				
ID	SCENARIO	SAMPLE INPUT IMAGE	RECOMMENDED OUPUT IMAGE	RECOMMENDATION
SemS-1	Embryo segmentation from microscopy images			Overlap-based metric (S6): DSC Boundary-based metric (S7): NSD Specific property-related metric: Liver segmentation: Absolute Volume Difference
SemS-2	Liver segmentation in CT images			
SemS-3	Labeling of invasive/non-invasive/ benign lesions on breast WSIs			Overlap-based metric (S6): F_{β} Score No boundary-based metric (S7) recommended (possibility of overlapping or touching structures)
SemS-4	Cortical structure segmentation from 3D MRI images			Overlap-based metric (S6): cDice Boundary-based metric (S7): NSD Specific property-related metric: Local and average cortical thickness
SemS-5	Aneurysm segmentation in TOF-MRA images			Overlap-based metric (S6): DSC Boundary-based metric (S7): HD95

ABBREVIATIONS

cDice Centerline Dice Similarity Coefficient
 DSC Dice Similarity Coefficient
 HD95 95th Percentile Hausdorff Distance
 NSD Normalized Surface Distance

Fig. SN 4.2. **Instantiation of the framework with recommendations for concrete biomedical semantic segmentation problems.** (SemS-1) Embryo segmentation from microscopy images [98]. (SemS-2) Liver segmentation in computed tomography (CT) images [1, 94]. (SemS-3) Labeling of invasive/non-invasive/benign lesions in breast whole slide imaging (WSI) [2]. (SemS-4) Cortical structure segmentation from 3D magnetic resonance imaging (MRI) images [19]. (SemS-5) Aneurysm segmentation in time-of-flight magnetic resonance angiography (TOF-MRA) images [100].

4.3 Object detection

The following use cases have been instantiated for object detection problems. The resulting metric recommendations can be found in Fig. SN 4.3, while Figs. SN 4.6-SN 4.9 provide a detailed overview of the recommendations for the use cases in the metric selection Subprocesses S3 - S4, S8 - S9.

ObD-1 Cell detection and tracking during the autophagy process in time-lapse microscopy [75, 121]

ObD-2 MS lesion detection in multi-modal brain MRI images [29, 57]

ObD-3 Polyp detection in colonoscopy videos with predefined sensitivity of 0.95 [12, 89]

ObD-4 Mitosis detection in histopathology images [7]

ObD-5 Lung nodule detection in CT images [3, 4, 25]

OBJECT DETECTION				
ID	SCENARIO	SAMPLE INPUT IMAGE	RECOMMENDED OUPUT IMAGE	RECOMMENDATION
ObD-1	Cell detection and tracking during the autophagy process in time-lapse microscopy videos			<p>Per-class counting metric (S3): FPPV@Sensitivity = 0.95</p> <p>Multi-threshold metric (S4): FROC Score</p> <p>Localization criterion (S8): Box IoU</p> <p>Assignment strategy (S9): Greedy (by Score) Matching Set double assignments to FP</p> <p>¹ Polyp detection: the FP are determined per video (= patient), not per frame (= image level), reflecting clinical interest.</p>
ObD-2	MS lesion detection in multi-modal brain MRI images			
ObD-3	Polyp detection in colonoscopy videos with predefined sensitivity of 0.95			
ObD-4	Mitosis detection in histopathology images			<p>Per-class counting metric (S3): F_p Score</p> <p>No multi-threshold metric (S4) needed (predicted class scores not available)</p> <p>Localization criterion (S8): Center Distance</p> <p>Assignment strategy (S9): Greedy (by Center Distance) Matching Set double assignments to FP</p>
ObD-5	Lung nodule detection in CT images			<p>Per-class counting metric (S3): PPV@Sensitivity = 0.90</p> <p>Multi-threshold metric (S4): AP</p> <p>Localization criterion (S8): Box IoU</p> <p>Assignment strategy (S9): Greedy (by Score) Matching Set double assignments to FP</p>

ABBREVIATIONS

AP Average Precision
Box IoU Box Intersection over Union

FP False Positive
FROC Free-Response Receiver Operating Characteristic
PPV Positive Predictive Value

Fig. SN 4.3. **Instantiation of the framework with recommendations for concrete biomedical object detection problems.** (ObD-1) Cell detection and tracking during the autophagy process in time-lapse microscopy [75, 121]. (ObD-2) Multiple sclerosis (MS) lesion detection in multi-modal brain magnetic resonance imaging (MRI) images [29, 57]. (ObD-3) Polyp detection in colonoscopy videos with predefined sensitivity of 0.95 [12, 89]. (ObD-4) Mitosis detection in histopathology images [7]. (ObD-5) Lung nodule detection in computed tomography (CT) images [3, 4, 25].

4.4 Instance segmentation

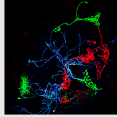
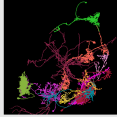


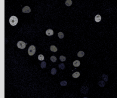

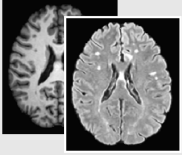
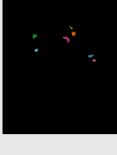
The following use cases have been instantiated for instance segmentation problems. The resulting metric recommendations can be found in Fig. SN 4.4, while Figs. SN 4.6-SN 4.11 provide a detailed overview of the recommendations for the use cases in the metric selection Subprocesses S3 - S4, S6 - S9.

InS-1 Instance segmentation of neurons from the fruit fly in 3D multi-color light microscopy images [67, 72, 101]

InS-2 Surgical instrument instance segmentation in colonoscopy videos [65]

InS-3 Cell nuclei instance segmentation in time-lapse light microscopy for cell tracking [103]

InS-4 MS lesion segmentation in multi-modal brain MRI images [29, 57]

INSTANCE SEGMENTATION				
ID	SCENARIO	SAMPLE INPUT IMAGE	RECOMMENDED OUPUT IMAGE	RECOMMENDATION
InS-1	Instance segmentation of neurons from the fruit fly in 3D multi-color light microscopy images			<ul style="list-style-type: none"> Per-class counting metric (S3): F_{β} Score Multi-threshold metric (S4): AP Overlap-based metric (S6): cDice Boundary-based metric (S7): NSD Localization criterion (S8): Neuron segmentation: Mask IoU Instrument segmentation: Boundary IoU Assignment strategy (S9): Greedy (by Score) Matching, set double assignments to FP
InS-2	Surgical instrument instance segmentation in colonoscopy videos			
InS-3	Cell nuclei instance segmentation in time-lapse light microscopy for cell tracking			<ul style="list-style-type: none"> Per-class counting metric (S3): F_{β} Score No multi-threshold metric (S4) needed (predicted class score not available) Overlap-based metric (S6): IoU No boundary-based metric (S7) needed (no interest in structure boundaries) Localization criterion (S8): IoR Assignment strategy (S9): Matching via IoR > 0.5 Set double assignments to FP
InS-4	MS lesion segmentation in multi-modal brain MRI images			<ul style="list-style-type: none"> Per-class counting metric (S3): FPPI@Sensitivity = 0.95 Multi-threshold metric (S4): FROC Score Overlap-based metric (S6): DSC Boundary-based metric (S7): NSD Localization criterion (S8): Boundary IoU Assignment strategy (S9): Greedy (by Score) Matching, set double assignments to FP

ABBREVIATIONS

<p>AP Average Precision</p> <p>Boundary IoU Boundary Intersection over Union</p> <p>cDice Centerline Dice Similarity Coefficient</p> <p>DSC Dice Similarity Coefficient</p> <p>FROC Free-Response Receiver Operating Characteristic</p> <p>FP False Positive</p>	<p>FPPI False Positives Per Image</p> <p>IoR Intersection over Reference</p> <p>IoU Intersection over Union</p> <p>Mask IoU Mask Intersection over Union</p> <p>NSD Normalized Surface Distance</p> <p>PSR Proper Scoring Rules</p>
--	---

Fig. SN 4.4. **Instantiation of the framework with recommendations for concrete biomedical instance segmentation problems.** (InS-1) Instance segmentation of neurons from the fruit fly in 3D multi-color light microscopy images [67, 72, 101]. (InS-2) Surgical instrument instance segmentation in colonoscopy videos [65]. (InS-3) Cell nuclei instance segmentation in time-lapse light microscopy for cell tracking [103]. (InS-4) Multiple sclerosis (MS) lesion segmentation in multi-modal brain magnetic resonance imaging (MRI) images [29, 57].

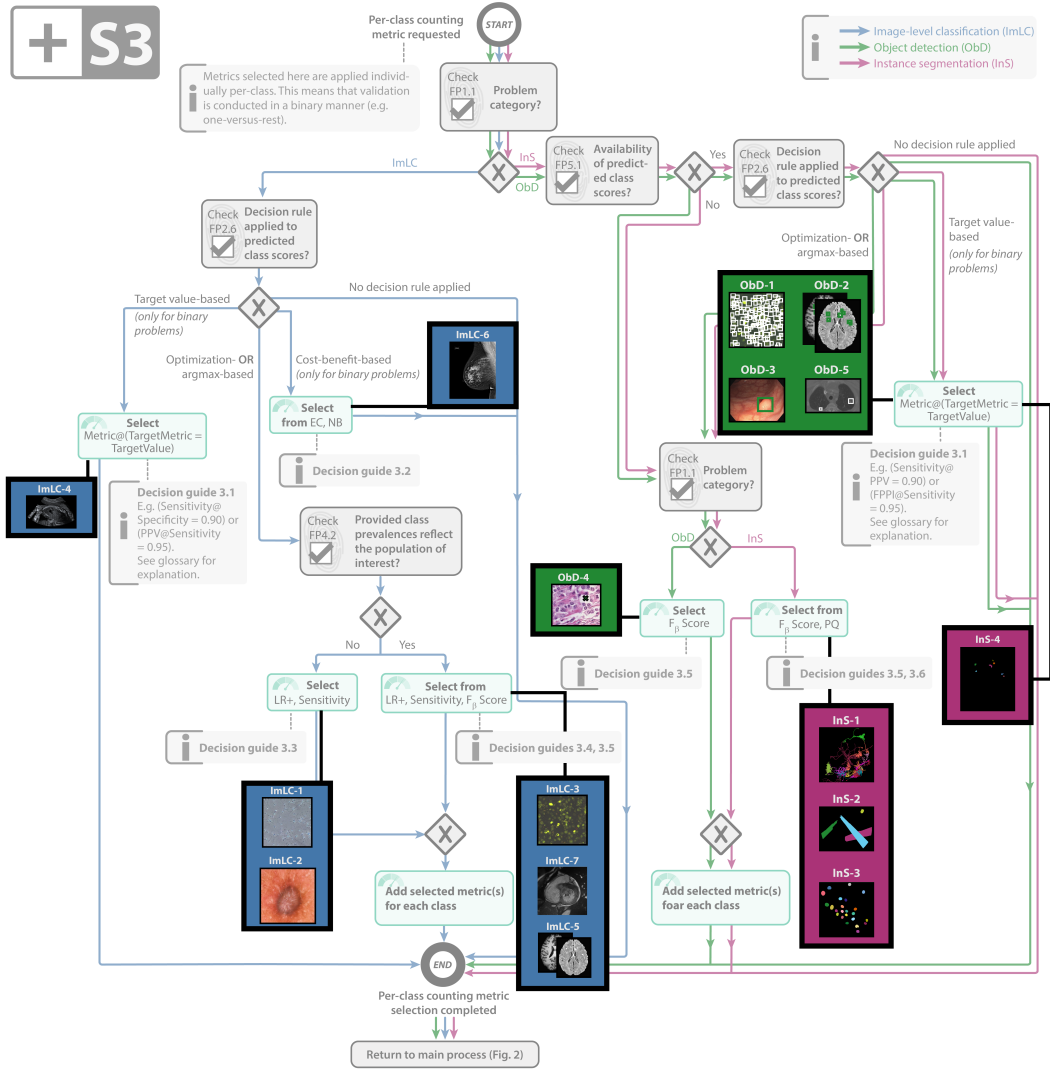


Fig. SN 4.6. **Instantiation of Subprocess S3 for the selection of per-class counting metrics with recommendations for concrete biomedical problems.** Included use cases: **(ImLC-1)** Frame-based sperm motility classification from microscopy time-lapse video of human spermatozoa [49]. **(ImLC-2)** Disease classification in dermoscopic images [26, 104]. **(ImLC-3)** Classification of the overall autophagy stage for a collection of cells [75, 121]. **(ImLC-4)** Diagnostic standard plane classification in ultrasound images [9]. **(ImLC-5)** Identification of new lesions in brain multi-modal magnetic resonance imaging (MRI) images of multiple sclerosis (MS) patients [29, 57]. **(ImLC-6)** Breast cancer classification in mammography images [62]. **(ImLC-7)** Multi-class cardiac disease classification in MRI images [13]. **(Obd-1)** Cell detection and tracking during the autophagy process in time-lapse microscopy [75, 121]. **(Obd-2)** MS lesion detection in multi-modal brain MRI images [29, 57]. **(Obd-3)** Polyp detection in colonoscopy videos with predefined sensitivity of 0.95 [12, 89]. **(Obd-4)** Mitosis detection in histopathology images [7]. **(Obd-5)** Lung nodule detection in computed tomography (CT) images [3, 4, 25]. **(InS-1)** Instance segmentation of neurons from the fruit fly in 3D multi-color light microscopy images [67, 72, 101]. **(InS-2)** Surgical instrument instance segmentation in colonoscopy videos [65]. **(InS-3)** Cell nuclei instance segmentation in time-lapse light microscopy with a subsequent goal of cell tracking [103]. **(InS-4)** MS Lesion segmentation in multi-modal brain MRI images [29, 57].

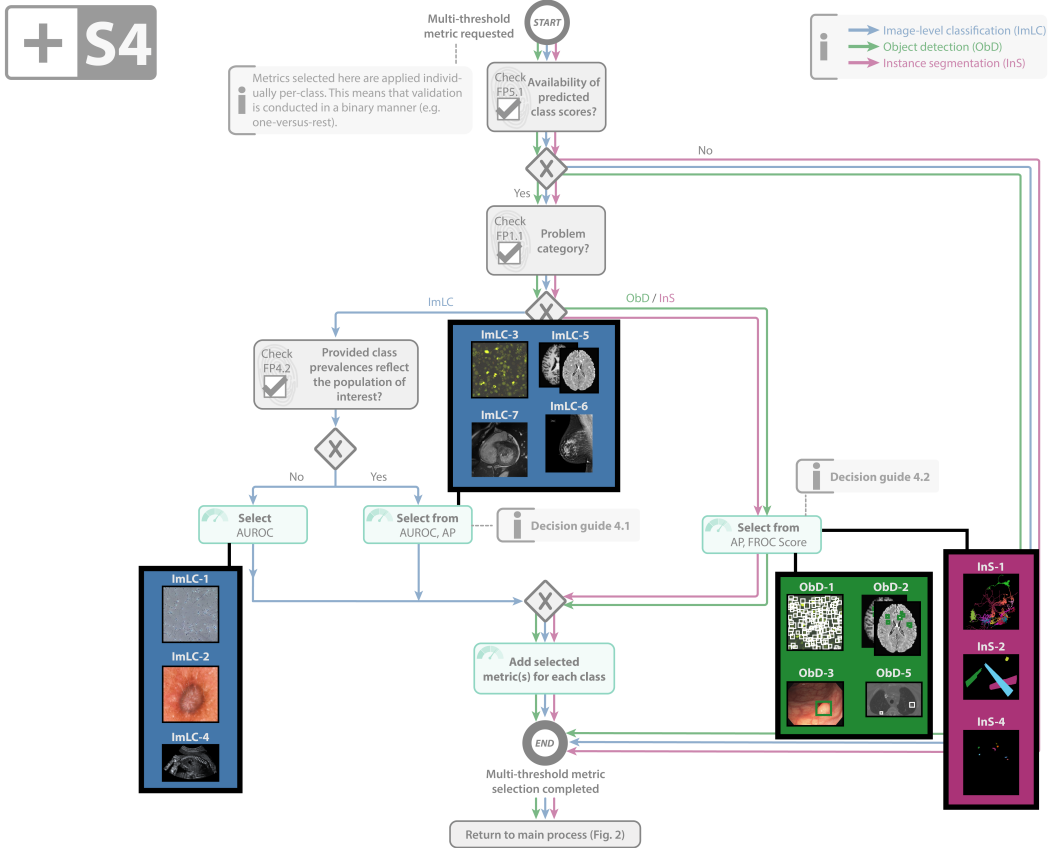


Fig. SN 4.7. **Instantiation of Subprocess S4 for the selection of multi-threshold metrics with recommendations for concrete biomedical problems.** Included use cases: **(ImLC-1)** Frame-based sperm motility classification from microscopy time-lapse video of human spermatozoa [49]. **(ImLC-2)** Disease classification in dermoscopic images [26, 104]. **(ImLC-3)** Classification of the overall autophagy stage for a collection of cells [75, 121]. **(ImLC-4)** Diagnostic standard plane classification in ultrasound images [9]. **(ImLC-5)** Identification of new lesions in brain multi-modal magnetic resonance imaging (MRI) images of multiple sclerosis (MS) patients [29, 57]. **(ImLC-6)** Breast cancer classification in mammography images [62]. **(ImLC-7)** Multi-class cardiac disease classification in MRI images [13]. **(ObD-1)** Cell detection and tracking during the autophagy process in time-lapse microscopy [75, 121]. **(ObD-2)** MS lesion detection in multi-modal brain magnetic resonance imaging (MRI) images [29, 57]. **(ObD-3)** Polyp detection in colonoscopy videos with predefined sensitivity of 0.95 [12, 89]. **(ObD-5)** Lung nodule detection in computed tomography (CT) images [3, 4, 25]. **(InS-1)** Instance segmentation of neurons from the fruit fly in 3D multi-color light microscopy images [67, 72, 101]. **(InS-2)** Surgical instrument instance segmentation in colonoscopy videos [65]. **(InS-4)** MS lesion segmentation in multi-modal brain MRI images [29, 57].

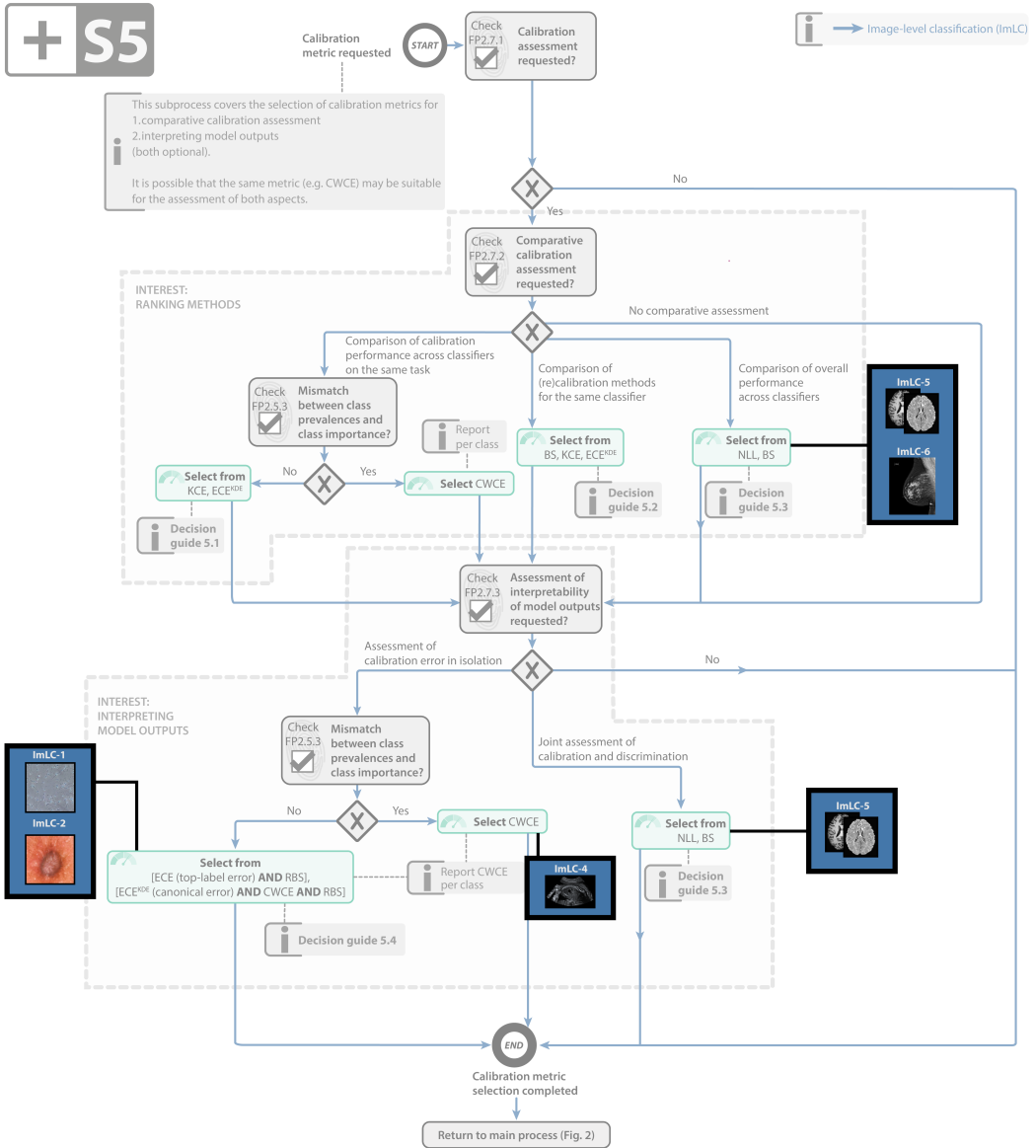


Fig. SN 4.8. Instantiation of Subprocess S5 for the selection of calibration metrics with recommendations for concrete biomedical problems. Included use cases: **(ImLC-1)** Frame-based sperm motility classification from microscopy time-lapse video of human spermatozoa [49]. **(ImLC-2)** Disease classification in dermoscopic images [26, 104]. **(ImLC-4)** Diagnostic standard plane classification in ultrasound images [9]. **(ImLC-5)** Identification of new lesions in brain multi-modal magnetic resonance imaging (MRI) images of multiple sclerosis (MS) patients [29, 57]. **(ImLC-6)** Breast cancer classification in mammography images [62].

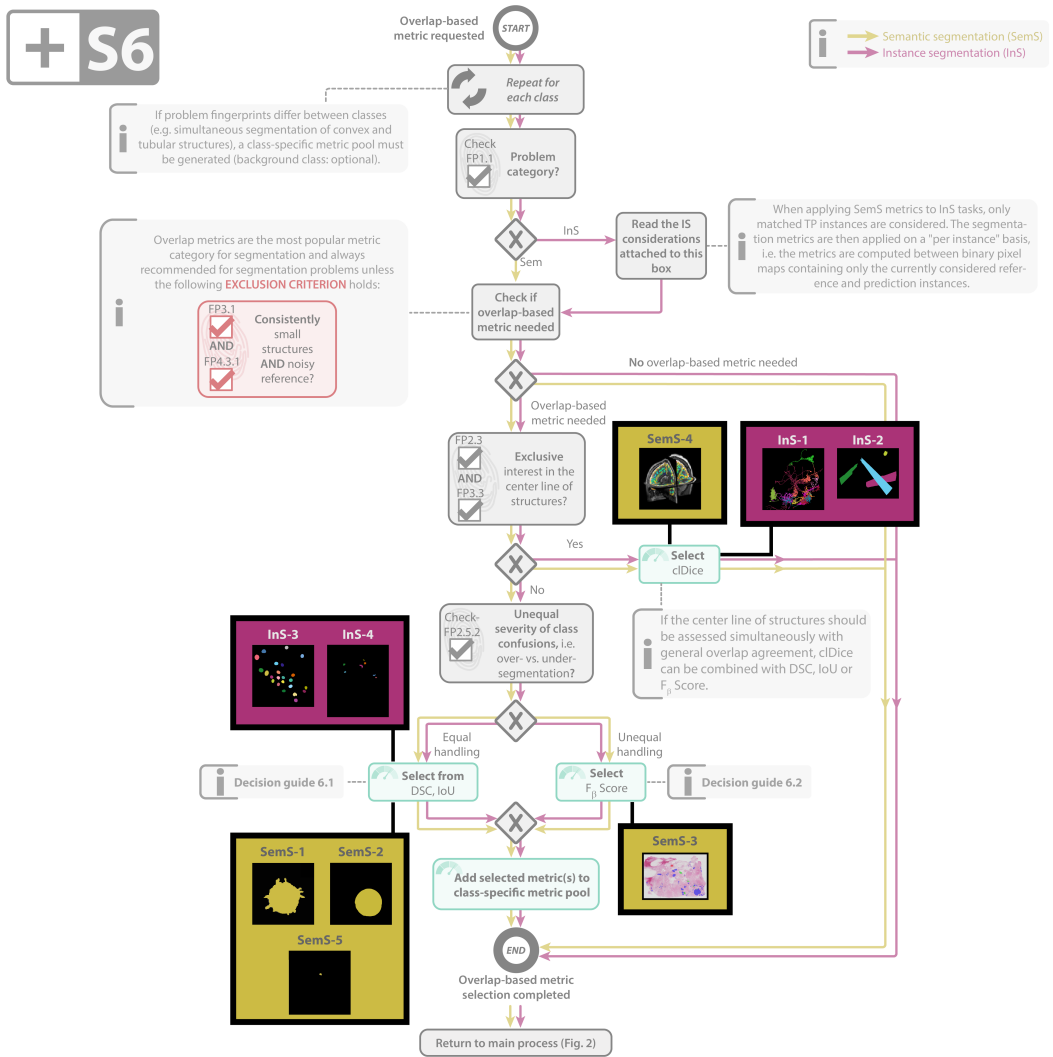


Fig. SN 4.9. **Instantiation of Subprocess S6 for the selection of overlap-based metrics with recommendations for concrete biomedical problems.** (**SemS-1**) Embryo cell segmentation from microscopy images [98]. (**SemS-2**) Liver segmentation in computed tomography (CT) images [1, 94]. (**SemS-3**) Labeling of invasive/ non-invasive/ benign lesions in breast whole slide imaging (WSI) [2]. (**SemS-4**) Cortical structure segmentation from 3D magnetic resonance imaging (MRI) images [19]. (**SemS-5**) Aneurysm segmentation in time-of-flight magnetic resonance angiography (TOF-MRA) images [100]. (**InS-1**) Instance segmentation of neurons from the fruit fly in 3D multi-color light microscopy images [67, 72, 101]. (**InS-2**) Surgical instrument segmentation in colonoscopy videos [65]. (**InS-3**) Cell nuclei instance segmentation in time-lapse light microscopy for cell tracking [103]. (**InS-4**) MS lesion segmentation in multi-modal brain MRI images [29, 57].

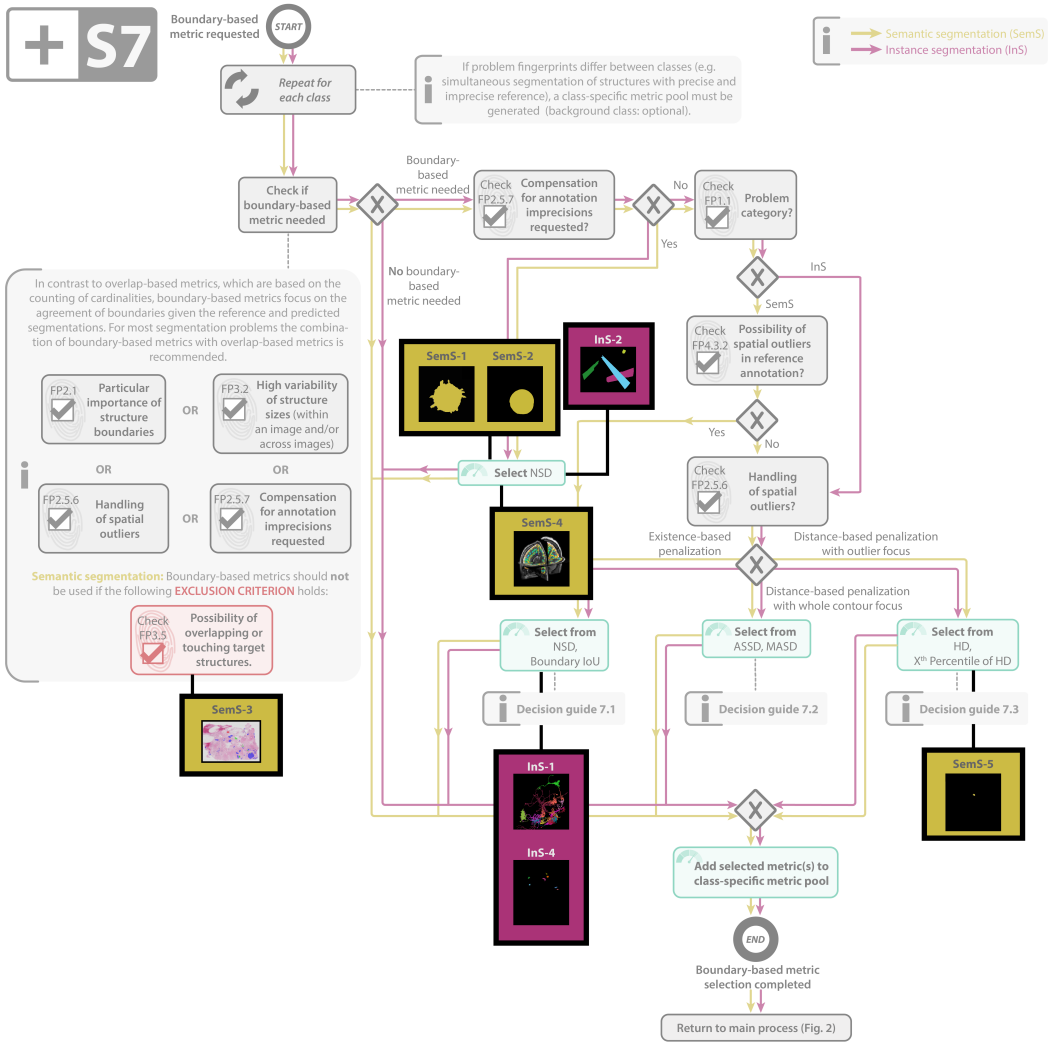


Fig. SN 4.10. **Instantiation of Subprocess S7 for the selection of boundary-based metrics with recommendations for concrete biomedical problems.** (**SemS-1**) Embryo segmentation from microscopy images [98]. (**SemS-2**) Liver segmentation in computed tomography (CT) images [1, 94]. (**SemS-3**) Labeling of invasive/non-invasive/benign lesions in breast whole slide imaging (WSI) [2]. (**SemS-4**) Cortical structure segmentation from 3D magnetic resonance imaging (MRI) images [19]. (**SemS-5**) Aneurysm segmentation in time-of-flight magnetic resonance angiography (TOF-MRA) images [100]. (**InS-1**) Instance segmentation of neurons from the fruit fly in 3D multi-color light microscopy images [67, 72, 101]. (**InS-2**) Surgical instrument instance segmentation in colonoscopy videos [65]. (**InS-3**) Cell nuclei instance segmentation in time-lapse light microscopy for cell tracking [103]. (**InS-4**) MS lesion segmentation in multi-modal brain MRI images [29, 57].

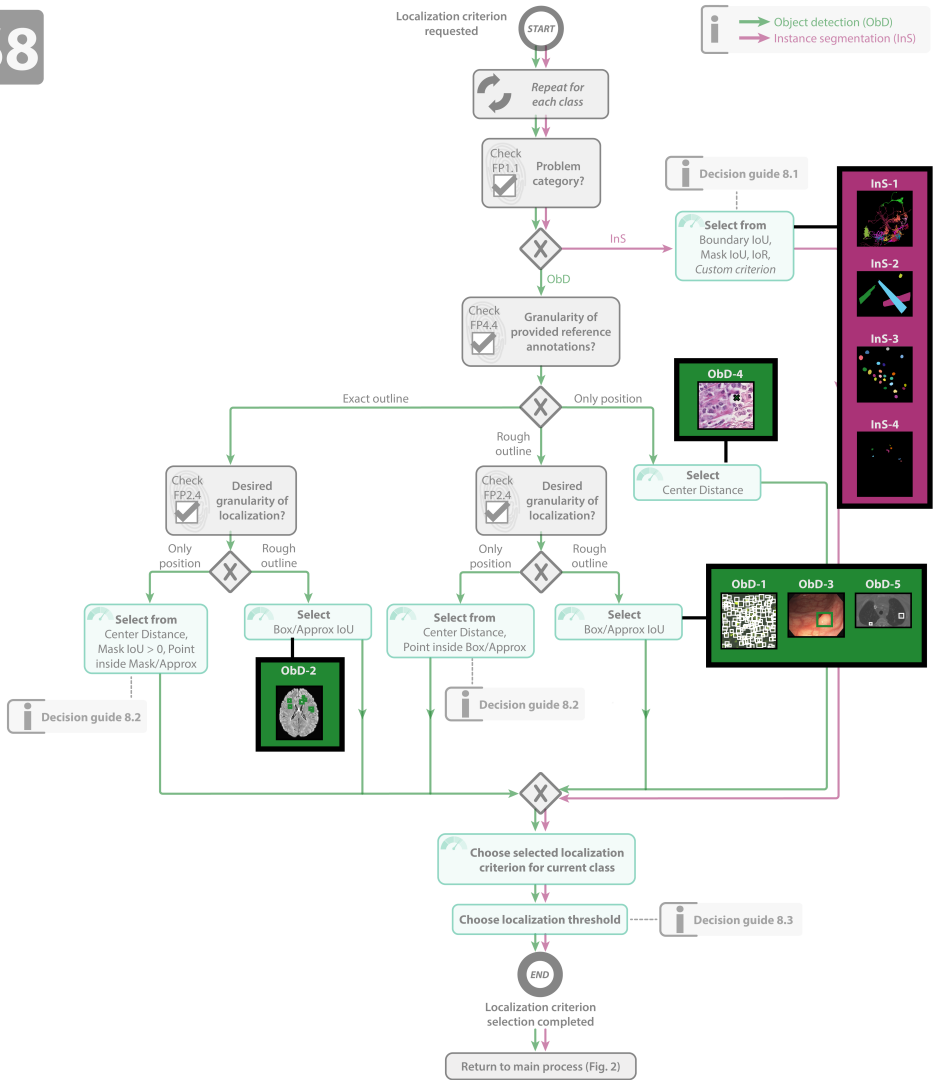


Fig. SN 4.11. **Instantiation of Subprocess S8 for the selection of localization criteria with recommendations for concrete biomedical problems.** (ObD-1) Cell detection and tracking during the autophagy process in time-lapse microscopy [75, 121]. (ObD-2) multiple sclerosis (MS) lesion detection in multi-modal brain magnetic resonance imaging (MRI) images [29, 57]. (ObD-3) Polyp detection in colonoscopy videos with predefined sensitivity of 0.9 [12, 89]. (ObD-4) Mitosis detection in histopathology images [7]. (ObD-5) Lung nodule detection in computed tomography (CT) images [3, 4, 25]. (InS-1) Instance segmentation of neurons from the fruit fly in 3D multi-color light microscopy images [67, 72, 101]. (InS-2) Surgical instrument instance segmentation in colonoscopy videos [65]. (InS-3) Cell nuclei instance segmentation in time-lapse light microscopy for cell tracking [103]. (InS-4) MS lesion segmentation in multi-modal brain MRI images [29, 57].

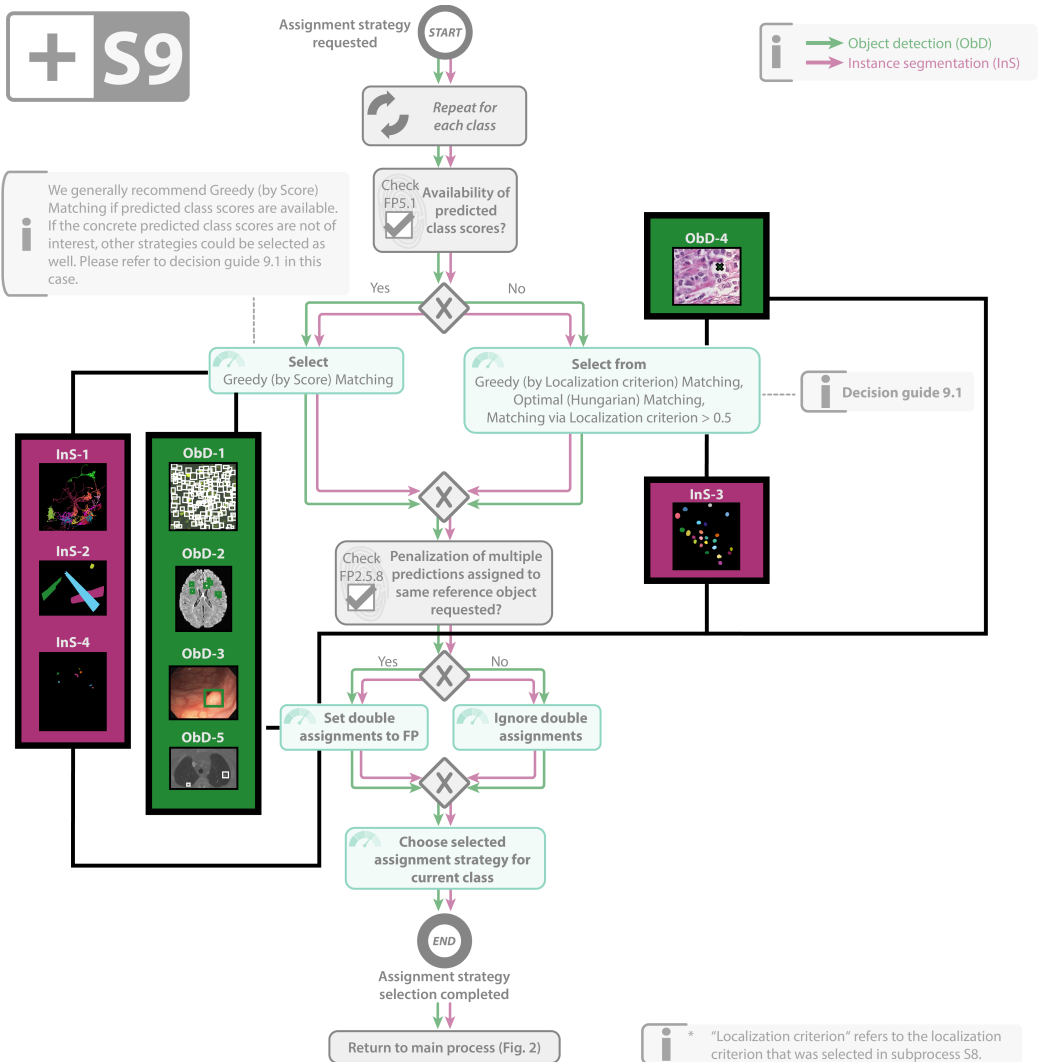


Fig. SN 4.12. **Instantiation of Subprocess S9 for the selection of assignment strategies with recommendations for concrete biomedical problems.** (ObD-1) Cell detection and tracking during the autophagy process in time-lapse microscopy [75, 121]. (ObD-2) multiple sclerosis (MS) Lesion detection in multi-modal brain magnetic resonance imaging (MRI) images [29, 57]. (ObD-3) Polyp detection in colonoscopy videos with predefined sensitivity of 0.95 [12, 89]. (ObD-4) Mitosis detection in histopathology images [7]. (ObD-5) Lung nodule detection in computed tomography (CT) images [3, 4, 25]. (InS-1) Instance segmentation of neurons from the fruit fly in 3D multi-color light microscopy images [67, 72, 101]. (InS-2) Surgical instrument instance segmentation in colonoscopy videos [65]. (InS-3) Cell nuclei instance segmentation in time-lapse light microscopy for cell tracking [103]. (InS-4) MS lesion segmentation in multi-modal brain MRI images [29, 57].

SUPPL. NOTE 5 TERMINOLOGY AND NOTATION

5.1 Symbol References



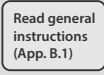
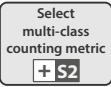

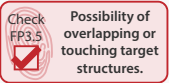
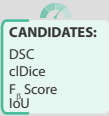
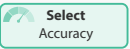


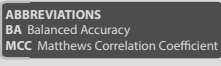
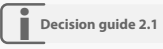

Symbol	Explanation
	Start of a process
	End of a process
	Task to be performed by the user
	Subprocess
	Task with reference to problem fingerprint
	Exclusion criterion
	Pool of options the user can choose from in the respective step
	A metric/criterion/strategy is selected
	Exclusive gateway: An exclusive gateway (or XOR-gateway) allows the user to make a decision. It can have multiple outgoing sequence flows. It is used when several conditions are mutually exclusive and only one selection is possible. An exclusive gateway is also used to join multiple incoming flows together and improve the readability of the diagram.
	Group
	Notes
	Further information
	The respective step needs to be repeated for each class

Fig. SN 5.1. **Overview of symbols used in the process diagrams.** The notation used in the process diagrams originates from Business Process Model and Notation (BPMN).

5.2 Expected formats of reference and algorithm output

Image-level Classification: The metric mapping expects the following format for image-level classification with C classes: For each image I there is a reference annotation y_I that either indicates the class for the image ($y_I \in \{1, \dots, C\}$), or, in the case of multi-label classification, indicates presence for each class ($y_I \in \{0, 1\}^C$). If the algorithm does not provide predicted class scores (FP5.1 = FALSE), the algorithm output should be provided in an identical format. Otherwise, for each image I , the continuous class scores for each of the classes ($\hat{y}_I \in [0, 1]^C$), indicating the predicted class probability, should be provided.

Semantic Segmentation We assume the reference annotation and the algorithm output to be in the same coordinate system with identical spacing. The metric mapping expects the following format for semantic segmentation with C classes: For each pixel P there is a reference annotation y_P that either assigns a single class to P ($y_P \in \{1, \dots, C\}$) or, in case of possible multiple labels per pixel, indicates assignment for each class ($y_P \in \{0, 1\}^C$). As for the algorithm output, for each pixel P there is expected to be either a single prediction ($\hat{y}_P \in \{1, \dots, C\}$) or, in case of multiple possible labels per pixel, a prediction for each class ($\hat{y}_P \in \{0, 1\}^C$). Some segmentation metrics require structure boundaries. For each class, boundaries are expected to be provided as a list of boundary pixels for both the reference and the prediction.

Object detection: The metric mapping expects the following format for object detection with C classes: For each object O the reference consists of a tuple (y_O, l_O) , where $y_O \in \{1, \dots, C\}$ indicates the class of the object and l_O is some location information (box, center point, radius, etc.). The algorithm output for an object prediction O is expected to comprise a tuple (\hat{y}_O, \hat{l}_O) as well, where \hat{y}_O indicates a single predicted class ($\hat{y}_O \in \{1, \dots, C\}$) optionally accompanied by an associated predicted class score ($\hat{c}_O \in [0, 1]$). Note that methods usually provide a predicted class score for the background class as well, but this score is typically discarded in validation as there are no "background objects" [39]. See FP5.1 in case no predicted class score is provided. \hat{l}_O is expected to provide location information about the prediction in a similar format as the reference (box, center point, radius, etc.). In case reference objects are represented by rough outlines (FP4.3) we assume that the chosen shapes (e.g., bounding box or ellipsoid) represent the underlying object adequately. Particular attention needs to be given to this aspect if objects feature a tubular shape (FP3.3) or can potentially appear disconnected (FP3.6).

Instance Segmentation The metric mapping expects the following format for instance segmentation with C classes: For each object O the reference consists of a tuple (y_O, m_O) , where $y_O \in \{1, \dots, C\}$ indicates the class of the object and $m_O \in \{0, 1\}^{H \times W}$ is a binary pixel map per instance matching the size of the image (height H and width W) and indicating pixel-wise location. The algorithm output for an object prediction O is expected to comprise a tuple (\hat{y}_O, \hat{m}_O) , where, similarly to object detection, \hat{y}_O indicates a single predicted class ($\hat{y}_O \in \{1, \dots, C\}$) optionally accompanied by an associated predicted class score ($\hat{c}_O \in [0, 1]$). \hat{m}_O denotes a binary pixel map per instance analogously to m_O . For both the reference and the predictions, structure boundaries should be provided as a list of boundary pixels separately for each instance. Note that annotations from semantic segmentation (not distinguishing instances of the same class) can be transformed to the instance segmentation format via connected component analysis (in case of purely non-touching and connected instances).

In case the provided reference annotations deviate from the expected format, matching can be achieved via various measures (e.g., aggregation of pixel-level reference to required image-level reference).

5.3 Acronyms

AI artificial intelligence
AP Average Precision
ASSD Average Symmetric Surface Distance
AUC Area under the curve
AUROC Area under the Receiver Operating Characteristic Curve
BA Balanced Accuracy
BM Bookmaker Informedness
BPMN Business Process Model and Notation
BS Brier Score
BSS Brier Skill Score
CE Calibration Error
CK Cohen’s Kappa
clDice centerline Dice Similarity Coefficient
CT computed tomography
CWCE Class-wise Calibration Error
DSC Dice Similarity Coefficient
EC Expected Cost
ECE Expected Calibration Error
ECE^{KDE} Expected Calibration Error Kernel Density Estimate
ECN normalized EC
EQUATOR Enhancing the QUALity and Transparency Of health Research
ER Error Rate
FN False Negative
FP False Positive
FPPI False Positives per Image
FDR False Discovery Rate
FOR False Omission Rate
FROC Free-Response Receiver Operating Characteristic
HD Hausdorff Distance
HD95 Hausdorff Distance 95th Percentile
ImLC Image-level Classification
InS Instance Segmentation
IoU Intersection over Union
IoR Intersection over Reference
J Youden Index
KCE Kernel Calibration Error
LR+ Positive Likelihood Ratio
mAP Mean Average Precision
MASD Mean Average Surface Distance
MCC Matthews Correlation Coefficient
MICCAI Medical Image Computing and Computer Assisted Interventions
MK Markedness

ML machine learning
MONAI Medical Open Network for Artificial Intelligence
MRI magnetic resonance imaging
MS multiple sclerosis
NaN 'Not a Number'
NB Net Benefit
NLL Negative Log Likelihood
NPV Negative Predictive Value
NSD Normalized Surface Distance
ObD Object Detection
O:E ratio Observed:Expected ratio
PPV Positive Predictive Value
PQ Panoptic Quality
PHI Protected Health Information
PR Precision-Recall
PSR Proper Scoring Rule
RBS Root Brier Score
RI Rand Index
ROC Receiver Operating Characteristic
ROI Region of Interest
SemS Semantic Segmentation
TN True Negative
TNR True Negative Rate
TOF-MRA time-of-flight magnetic resonance angiography
TP True Positive
TPR True Positive Rate
VoI Variation of Information
WCK Weighted Cohen's Kappa
WSI whole slide imaging
Xth Percentile HD Xth Percentile Hausdorff Distance

5.4 Glossary

- **Bounding box:** A bounding box is a rectangle, typically the smallest possible, drawn around and completely surrounding an object to be detected.
- **Calibration plot:** A calibration plot, also referred to as reliability diagram, is a visualization of the calibration ability of a model's outputs (see e.g., [44]). Specifically, the diagram allows to diagnose a model's general bias towards "overconfident" or "underconfident" predictions by visualizing the deviation from perfect calibration (diagonal line in the plot) for different output scores. The diagram also acts as the basis for further diagnostic measurements such as the calibration slope.
- **Challenge:** A challenge is an international competition, commonly hosted by individual researchers, an institute, or a professional society, that aims to comparatively assess the performance of competing algorithms on an identical data set, and thus serves to validate them. This validation is a crucial step towards the translation of an algorithm into practice.
- **Classification task:** A classification task is the task of giving categorical labels to an image or parts thereof. We distinguish classification at different scales, e.g., at image level, pixel level or object level.
- **Confidence:** See Predicted class scores.
- **Continuous class scores:** See Predicted class scores.
- **Decision rule:** A rule transforming continuous predicted class scores into discrete classification decisions. This rule amounts to setting a simple cutoff value in binary classification problems but is more complex to define in multi-class problems (for more information see Suppl. Note 1.1).
- **Evaluation:** See Validation.
- **Hierarchical structure of classes/data:** A hierarchical structure of classes/data is present when classes or data are dependent on each other or paired, e.g., when data have been derived from the same patient, or from the same center. It requires interpretation and statistical efforts different from those suitable for independent data.
- **Hyperparameter:** A hyperparameter is a parameter whose value is optimized to control the training of an algorithm. In contrast to other parameters, it is not derived through the training process itself, but rather set before the training procedure.
- **Inference:** In the context of ML, inference denotes the processing of data by an algorithm to produce the desired output.
- **Instance:** An instance refers to a dedicated object, structure or entity in an image, such as an individual cell, tumor or medical instrument.
- **Image-level classification:** Image-level classification is the assignment of one or multiple category labels to an entire image, as detailed in Suppl. Note 1.1.
- **Instance segmentation:** Detection and delineation of each distinct object of a particular class in an image, as detailed in Suppl. Note 1.1.
- **Instantiation:** Instantiation here refers to the act of creating a specific application case of a general principle/framework.
- **Macro/micro averaging:** Macro averaging is the process of computing a metric (e.g., Sensitivity) for each class and subsequently averaging the metric scores. Micro averaging is the process of aggregating an average metric score over all classes.
- **Meta-information:** Meta-information refers to data about an image that is not explicitly contained within the image, e.g., Protected Health Information (PHI) data about the patient in radiology images.

- **Metric:** Metrics are the measures according to which performance of algorithms is quantified and validated. Depending on the domain-specific validation goal and property of interest, we distinguish between different types of metrics, e.g., reference-based vs. non-reference based (see Reference/Reference-based metrics). Metrics can further be subdivided into different families based on their mathematical properties.
- **Metric@(TargetMetric = TargetValue):** (e.g., Specificity@(Sensitivity = 0.95)): Once a cutoff value for the predicted class probabilities has been set in such a way that the target metric value is achieved (here: target metric Sensitivity with a target value of 0.95), other metric values (here: Specificity) are obtained from the corresponding fixed confusion matrix. In the example, this yields the Specificity at the predefined Sensitivity level.
- **Object detection:** Detection and localization of structures of one or multiple categories in an image, as detailed in Suppl. Note 1.1.
- **(Output) Calibration:** In application scenarios that involve interpreting the raw algorithm output (specifically the predicted class scores), output calibration can be used to obtain a reliable measure of confidence associated with the decision (see description of FP2.7 in Suppl. Note 1.2).
- **Precision:** Precision is a term used differently in different scientific communities. In the medical community, for example, it commonly refers to the confidence of an output. Here, we use the term to denote the Positive Predictive Value (PPV).
- **Predicted class scores:** Modern neural network-based approaches usually output predicted class scores (also referred to as continuous class scores, confidence scores or pseudo-probabilities) between 0 and 1 for every image/object/pixel and class, indicating the probability of the image/object/pixel belonging to a specific class.
- **Prediction:** Prediction refers to the output of an algorithm. It is not used in the temporal sense in this paper.
- **Problem category:** Biomedical image analysis problems can be subdivided into problem categories according to the procedures performed. The category a problem falls into informs the appropriate choice of metrics. In this paper, we focus on four problem categories: Image-level classification, Semantic Segmentation, Object Detection, and Instance Segmentation.
- **Pseudo-probabilities:** See Predicted class scores.
- **Reference/Reference-based metrics:** We assume that the validation process is based on the comparison of the algorithm output and a **reference** (sometimes called **gold standard**), which is assumed to be close or equal to the correct result – the (often forever unknown) **ground truth**. In terms of metrics, we distinguish between *reference-based metrics* [53], which use the image-based reference, and *non-reference-based metrics* that assess complementary properties, such as runtime, memory consumption, or carbon footprint.
- **Reliability diagram:** See calibration plot.
- **Semantic segmentation:** Assignment of one or multiple category labels to each pixel in an image, as detailed in Suppl. Note 1.1.
- **Structure instance:** See Instance.
- **Training/Test case:** The data sets used in the process of algorithm development and validation comprise training/test cases. A case refers to the data (typically an n-dimensional image, possibly enhanced with clinical context information) that is required for an algorithm to produce one result (e.g., a segmentation or classification). A training case refers to a data set that includes reference annotations and is thus used for training an algorithm. A test case refers to a data set that is used for performance assessment

- **Type 1 and Type 2 error:** A type 1 error is a False Positive (FP) result, e.g., a false detection of something that is not present. A type 2 error is a False Negative (FN) result, e.g., a non-detection of something that is present.
- **Validation:** Validation is the process of assessing that the validated algorithm is effectively doing what it is expected to do and what it was developed for, for example that a segmentation method is actually segmenting. Evaluation is the process of assessing that the algorithm is valuable, i.e., that it brings quantifiable added value for the clinical user in a dedicated clinical context [51].

REFERENCES

- [1] Michela Antonelli, Annika Reinke, Spyridon Bakas, Keyvan Farahani, Annette Kopp-Schneider, Bennett A Landman, Geert Litjens, Bjoern Menze, Olaf Ronneberger, Ronald M Summers, et al. The medical segmentation decathlon. *Nature Communications*, 13(1):1–13, 2022.
- [2] Guilherme Aresta, Teresa Araújo, Scotty Kwok, Sai Saketh Chennamsetty, Mohammed Safwan, Varghese Alex, Bahram Marami, Marcel Prastawa, Monica Chan, Michael Donovan, et al. Bach: Grand challenge on breast cancer histology images. *Medical image analysis*, 56:122–139, 2019.
- [3] S. G. Armato III, G. McLennan, L. Bidaut, M. F. McNitt-Gray, C. R. Meyer, A. P. Reeves, B. Zhao, D. R. Aberle, C. I. Henschke, E. A. Hoffman, et al. Data from lidc-idri [data set]. *The Cancer Imaging Archive*, 2015.
- [4] Samuel G Armato III, Geoffrey McLennan, Luc Bidaut, Michael F McNitt-Gray, Charles R Meyer, Anthony P Reeves, Binsheng Zhao, Denise R Aberle, Claudia I Henschke, Eric A Hoffman, et al. The lung image database consortium (lidc) and image database resource initiative (idri): a completed reference database of lung nodules on ct scans. *Medical physics*, 38(2):915–931, 2011.
- [5] Saeid Asgari Taghanaki, Kumar Abhishek, Joseph Paul Cohen, Julien Cohen-Adad, and Ghassan Hamarneh. Deep semantic segmentation of natural and medical images: a review. *Artificial Intelligence Review*, 54(1):137–178, 2021.
- [6] John Attia. Moving beyond sensitivity and specificity: using likelihood ratios to help interpret diagnostic tests. *Australian prescriber*, 26(5):111–113, 2003.
- [7] Marc Aubreville, Nikolas Stathonikos, Christof A Bertram, Robert Klopfeisch, Natalie ter Hoeve, Francesco Ciompi, Frauke Wilm, Christian Marzahl, Taryn A Donovan, Andreas Maier, et al. Mitosis domain generalization in histopathology images—the midog challenge. *arXiv preprint arXiv:2204.03742*, 2022.
- [8] Andriy I Bandos, Howard E Rockette, Tao Song, and David Gur. Area under the free-response roc curve (froc) and a related summary index. *Biometrics*, 65(1):247–256, 2009.
- [9] Christian F Baumgartner, Konstantinos Kamnitsas, Jacqueline Matthew, Tara P Fletcher, Sandra Smith, Lisa M Koch, Bernhard Kainz, and Daniel Rueckert. Sononet: real-time detection and localisation of fetal standard scan planes in freehand ultrasound. *IEEE transactions on medical imaging*, 36(11):2204–2215, 2017.
- [10] Babak Ehteshami Bejnordi, Mitko Veta, Paul Johannes Van Diest, Bram Van Ginneken, Nico Karssemeijer, Geert Litjens, Jeroen AWM Van Der Laak, Meyke Hermesen, Quirine F Manson, Maschenka Balkenhol, et al. Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. *Jama*, 318(22):2199–2210, 2017.
- [11] Miroslav Beneš and Barbara Zitová. Performance evaluation of image segmentation algorithms on microscopic image data. *Journal of microscopy*, 257(1):65–85, 2015.
- [12] Jorge Bernal, Aymeric Histace, Marc Masana, Quentin Angermann, Cristina Sánchez-Montes, Cristina Rodríguez de Miguel, Maroua Hammami, Ana García-Rodríguez, Henry Córdova, Olivier Romain, et al. Gtcreator: a flexible annotation tool for image-based datasets. *International journal of computer assisted radiology and surgery*, 14(2):191–201, 2019.
- [13] Olivier Bernard, Alain Lalonde, Clement Zotti, Frederick Cervenansky, Xin Yang, Pheng-Ann Heng, Irem Cetin, Karim Lekadir, Oscar Camara, Miguel Angel Gonzalez Ballester, et al. Deep learning techniques for automatic mri cardiac multi-structures segmentation and diagnosis: is the problem solved? *IEEE transactions on medical imaging*, 37(11):2514–2525, 2018.
- [14] Sebastian Bickelhaupt, Paul Ferdinand Jaeger, Frederik Bernd Laun, Wolfgang Lederer, Heidi Daniel, Tristan Anselm Kuder, Lorenz Wuesthof, Daniel Paech, David Bonekamp, Alexander Radbruch, et al. Radiomics based on adapted diffusion kurtosis imaging helps to clarify most mammographic findings suspicious for cancer. *Radiology*, 287(3):761–770, 2018.
- [15] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.
- [16] Patrick M Bossuyt, Johannes B Reitsma, David E Bruns, Constantine A Gatsonis, Paul P Glasziou, Les M Irwig, Jeroen G Lijmer, David Moher, Drummond Rennie, Henrica CW De Vet, et al. Towards complete and accurate reporting of studies of diagnostic accuracy: the stard initiative. *Annals of internal medicine*, 138(1):40–44, 2003.
- [17] Glenn W Brier et al. Verification of forecasts expressed in terms of probability. *Monthly weather review*, 78(1):1–3, 1950.
- [18] Bernice B Brown. Delphi process: a methodology used for the elicitation of opinions of experts. Technical report, Rand Corp Santa Monica CA, 1968.
- [19] Samuel Budd, Prachi Patke, Ana Baburamani, Mary Rutherford, Emma C Robinson, and Bernhard Kainz. Surface agnostic metrics for cortical volume segmentation and regression. In *Machine Learning in Clinical Neuroimaging and Radiogenomics in Neuro-Oncology*, pages 3–12. Springer, 2020.
- [20] Juan C Caicedo, Allen Goodman, Kyle W Karhohs, Beth A Cimini, Jeanelle Ackerman, Marzieh Haghighi, CherKeng Heng, Tim Becker, Minh Doan, Claire McQuin, et al. Nucleus segmentation across imaging experiments: the 2018

- data science bowl. *Nature methods*, 16(12):1247–1253, 2019.
- [21] Juan C Caicedo, Jonathan Roth, Allen Goodman, Tim Becker, Kyle W Karhohs, Matthieu Broisin, Csaba Molnar, Claire McQuin, Shantanu Singh, Fabian J Theis, et al. Evaluation of deep learning strategies for nucleus segmentation in fluorescence images. *Cytometry Part A*, 95(9):952–965, 2019.
- [22] Bowen Cheng, Ross Girshick, Piotr Dollár, Alexander C Berg, and Alexander Kirillov. Boundary iou: Improving object-centric image segmentation evaluation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15334–15342, 2021.
- [23] Nancy Chinchor. Muc-4 evaluation metrics. In *Proceedings of the 4th Conference on Message Understanding, MUC4 '92*, page 22–29, USA, 1992. Association for Computational Linguistics. ISBN 1558602739. doi: 10.3115/1072064.1072067. URL <https://doi.org/10.3115/1072064.1072067>.
- [24] Neil T Clancy, Geoffrey Jones, Lena Maier-Hein, Daniel S Elson, and Danaïl Stoyanov. Surgical spectral imaging. *Medical image analysis*, 63:101699, 2020.
- [25] Kenneth Clark, Bruce Vendt, Kirk Smith, John Freymann, Justin Kirby, Paul Koppel, Stephen Moore, Stanley Phillips, David Maffitt, Michael Pringle, et al. The cancer imaging archive (tcia): maintaining and operating a public information repository. *Journal of digital imaging*, 26(6):1045–1057, 2013.
- [26] Noel Codella, Veronica Rotemberg, Philipp Tschandl, M Emre Celebi, Stephen Dusza, David Gutman, Brian Helba, Aadi Kalloo, Konstantinos Liopyris, Michael Marchetti, et al. Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic). *arXiv preprint arXiv:1902.03368*, 2019.
- [27] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.
- [28] Gary S Collins, Paula Dhiman, Constanza L Andaur Navarro, Jie Ma, Lotty Hooft, Johannes B Reitsma, Patricia Logullo, Andrew L Beam, Lily Peng, Ben Van Calster, et al. Protocol for development of a reporting guideline (tripod-ai) and risk of bias tool (probast-ai) for diagnostic and prognostic prediction model studies based on artificial intelligence. *BMJ open*, 11(7):e048008, 2021.
- [29] Olivier Commowick, Audrey Istace, Michael Kain, Baptiste Laurent, Florent Leray, Mathieu Simon, Sorina Camarasu Pop, Pascal Girard, Roxana Ameli, Jean-Christophe Ferré, et al. Objective evaluation of multiple sclerosis lesion segmentation using a data management and processing infrastructure. *Scientific reports*, 8(1):1–17, 2018.
- [30] CONSORT-AI and SPIRIT-AI Steering Group. Reporting guidelines for clinical trials evaluating artificial intelligence interventions are needed. *Nat. Med.*, 25(10):1467–1468, October 2019.
- [31] George Cybenko, Dianne P O’Leary, and Jorma Rissanen. *The Mathematics of Information Coding, Extraction and Distribution*, volume 107. Springer Science & Business Media, 1998.
- [32] Marc-Alexandre Côté, Gabriel Girard, Arnaud Boré, Eleftherios Garyfallidis, Jean-Christophe Houde, and Maxime Descoteaux. Tractometer: towards validation of tractography pipelines. *Medical Image Analysis*, 17(7):844–857, October 2013. ISSN 1361-8423. doi: 10.1016/j.media.2013.03.009.
- [33] Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd international conference on Machine learning*, pages 233–240, 2006.
- [34] Morris H DeGroot and Stephen E Fienberg. The comparison and evaluation of forecasters. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 32(1-2):12–22, 1983.
- [35] Lee R Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302, 1945.
- [36] James M Dolezal, Andrew Srisuwananukorn, Dmitry Karpeyev, Siddhi Ramesh, Sara Kochanny, Brittany Cody, Aaron S Mansfield, Sagar Rakshit, Radhika Bansal, Melanie C Bois, et al. Uncertainty-informed deep learning models enable high-confidence predictions for digital histopathology. *Nature communications*, 13(1):6572, 2022.
- [37] Mark Everingham, Andrew Zisserman, Christopher KI Williams, Luc Van Gool, Moray Allan, Christopher M Bishop, Olivier Chapelle, Navneet Dalal, Thomas Deselaers, Gyuri Dorkó, et al. The 2005 pascal visual object classes challenge. In *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment: First PASCAL Machine Learning Challenges Workshop, MLCW 2005, Southampton, UK, April 11-13, 2005, Revised Selected Papers*, pages 117–176. Springer, 2006.
- [38] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.
- [39] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111(1):98–136, 2015.
- [40] Luciana Ferrer. Analysis and comparison of classification metrics. *arXiv preprint arXiv:2209.05355*, 2022. The document discusses common performance metrics used in machine learning classification, and introduces the expected cost (EC) metric. It compares these metrics and argues that EC is superior due to its generality, simplicity, and intuitive nature. Additionally, it highlights the potential of EC in measuring calibration and optimal decision-making using class posteriors.

- [41] Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378, 2007.
- [42] Margherita Grandini, Enrico Bagli, and Giorgio Visani. Metrics for multi-class classification: an overview. *arXiv preprint arXiv:2008.05756*, 2020.
- [43] Sebastian Gregor Gruber and Florian Buettner. Better uncertainty calibration via proper scores for classification and beyond. In *Advances in Neural Information Processing Systems*, 2022.
- [44] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On Calibration of Modern Neural Networks. *ICML*, page 10, 2017.
- [45] Kartik Gupta, Amir Rahimi, Thalaiyasingham Ajanthan, Thomas Mensink, Cristian Sminchisescu, and Richard Hartley. Calibration of neural networks using splines. *arXiv preprint arXiv:2006.12800*, 2020.
- [46] Metin N Gurcan, Anant Madabhushi, and Nasir Rajpoot. Pattern recognition in histopathological images: An icpr 2010 contest. In *International Conference on Pattern Recognition*, pages 226–234. Springer, 2010.
- [47] James A Hanley and Barbara J McNeil. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36, 1982.
- [48] Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.
- [49] Trine B Haugen, Steven A Hicks, Jorunn M Andersen, Oliwia Witczak, Hugo L Hammer, Rune Borgli, Pål Halvorsen, and Michael Riegler. Visem: A multimodal video dataset of human spermatozoa. In *Proceedings of the 10th ACM Multimedia Systems Conference*, pages 261–266, 2019.
- [50] Daniel P Huttenlocher, Gregory A. Klanderman, and William J Rucklidge. Comparing images using the hausdorff distance. *IEEE Transactions on pattern analysis and machine intelligence*, 15(9):850–863, 1993.
- [51] BSEN ISO 9000. Quality management systems: Fundamentals and vocabulary. *London: British Standards Institution*, 2000.
- [52] Paul Jaccard. The distribution of the flora in the alpine zone. 1. *New phytologist*, 11(2):37–50, 1912.
- [53] Pierre Jannin, Christophe Grova, and Calvin R Maurer. Model for defining and reporting reference-based validation protocols in medical image processing. *International Journal of Computer Assisted Radiology and Surgery*, 1(2):63–73, 2006.
- [54] Liang Jin, Jiancheng Yang, Kaiming Kuang, Bingbing Ni, Yiyi Gao, Yingli Sun, Pan Gao, Weiling Ma, Mingyu Tan, Hui Kang, et al. Deep-learning-assisted detection and segmentation of rib fractures from ct scans: Development and validation of fracnet. *EBioMedicine*, 62:103106, 2020.
- [55] Thierry Judge, Olivier Bernard, Mihaela Porumb, Agisilaos Chartsias, Arian Beqiri, and Pierre-Marc Jodoin. -reliable uncertainty estimation for medical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 492–502. Springer, 2022.
- [56] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9404–9413, 2019.
- [57] Florian Kofler, Suprosanna Shit, Ivan Ezhov, Lucas Fidon, Rami Al-Maskari, Hongwei Li, Harsharan Bhatia, Timo Loehr, Marie Piraud, Ali Erturk, et al. blob loss: instance imbalance aware loss functions for semantic segmentation. *arXiv preprint arXiv:2205.08209*, 2022.
- [58] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.
- [59] Meelis Kull, Miquel Perello Nieto, Markus Kängsepp, Telmo Silva Filho, Hao Song, and Peter Flach. Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration. *Advances in neural information processing systems*, 32, 2019.
- [60] Ananya Kumar, Percy S Liang, and Tengyu Ma. Verified uncertainty calibration. *Advances in Neural Information Processing Systems*, 32, 2019.
- [61] Fabian Kuppens, Jan Kronenberger, Amirhossein Shantia, and Anselm Haselhoff. Multivariate confidence calibration for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 326–327, 2020.
- [62] EPV Le, Y Wang, Yuan Huang, Sarah Hickman, and FJ Gilbert. Artificial intelligence in breast imaging. *Clinical radiology*, 74(5):357–366, 2019.
- [63] Zeju Li, Konstantinos Kamnitsas, Mobarakol Islam, Chen Chen, and Ben Glocker. Estimating model performance under domain shifts with class-specific confidence scores. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 693–703. Springer, 2022.
- [64] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

- [65] Lena Maier-Hein, Martin Wagner, Tobias Ross, Annika Reinke, Sebastian Bodenstedt, Peter M Full, Hellena Hempe, Diana Mindroc-Filimon, Patrick Scholz, Thuy Nuong Tran, et al. Heidelberg colorectal data set for surgical data science in the sensor operating room. *Scientific data*, 8(1):1–11, 2021.
- [66] Lena Maier-Hein, Annika Reinke, Evangelia Christodoulou, Ben Glocker, Patrick Godau, Fabian Isensee, Jens Kleesiek, Michal Kozubek, Mauricio Reyes, Michael A Riegler, et al. Metrics reloaded: Pitfalls and recommendations for image analysis validation. *arXiv preprint arXiv:2206.01653*, 2022.
- [67] Lisa Mais, Peter Hirsch, and Dagmar Kainmueller. Patchperpix for instance segmentation. In *European Conference on Computer Vision*, pages 288–304. Springer, 2020.
- [68] Martin Maška, Vladimír Ulman, David Svoboda, Pavel Matula, Petr Matula, Cristina Ederra, Ainhoa Urbiola, Tomás España, Subramanian Venkatesan, Deepak MW Balak, et al. A benchmark for comparison of cell tracking algorithms. *Bioinformatics*, 30(11):1609–1617, 2014.
- [69] Brian W Matthews. Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *Biochimica et Biophysica Acta (BBA)-Protein Structure*, 405(2):442–451, 1975.
- [70] Pavel Matula, Martin Maška, Dmitry V Sorokin, Petr Matula, Carlos Ortiz-de Solórzano, and Michal Kozubek. Cell tracking accuracy measurement based on comparison of acyclic oriented graphs. *PLoS one*, 10(12):e0144959, 2015.
- [71] Alireza Mehrtash, William M Wells, Clare M Tempany, Purang Abolmaesumi, and Tina Kapur. Confidence calibration and predictive uncertainty estimation for deep medical image segmentation. *IEEE transactions on medical imaging*, 39(12):3868–3878, 2020.
- [72] G. Meissner, A. Nern, Z. Dorman, DePasquale G.M., K. Forster, T. Gibney, Hausenfluck J.H., Y. He, N. Iyer, J. Jeter, et al. A searchable image resource of drosophila gal4-driver expression patterns with single neuron resolution. *BioRxiv*, page 2020.05.29.080473, 2022.
- [73] Karel GM Moons, Douglas G Altman, Johannes B Reitsma, John PA Ioannidis, Petra Macaskill, Ewout W Steyerberg, Andrew J Vickers, David F Ransohoff, and Gary S Collins. Transparent reporting of a multivariable prediction model for individual prognosis or diagnosis (tripod): explanation and elaboration. *Annals of internal medicine*, 162(1):W1–W73, 2015.
- [74] Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using bayesian binning. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.
- [75] Yukiko Nagao, Mika Sakamoto, Takumi Chinen, Yasushi Okada, and Daisuke Takao. Robust classification of cell cycle phase and biological feature extraction by image-based deep learning. *Molecular biology of the cell*, 31(13):1346–1354, 2020.
- [76] Ying-Hwey Nai, Bernice W Teo, Nadya L Tan, Sophie O’Doherty, Mary C Stephenson, Yee Liang Thian, Edmund Chiong, and Anthonin Reilhac. Comparison of metrics for the evaluation of medical segmentations using prostate mri dataset. *Computers in Biology and Medicine*, 134:104497, 2021.
- [77] Prashant Nasa, Ravi Jain, and Deven Juneja. Delphi methodology in healthcare research: how to decide its appropriateness. *World Journal of Methodology*, 11(4):116, 2021.
- [78] Vishwesh Nath, Kurt G Schilling, Prasanna Parvathaneni, Yuankai Huo, Justin A Blaber, Allison E Hainline, Muhamed Barakovic, David Romascano, Jonathan Rafael-Patino, Matteo Frigo, et al. Tractography reproducibility challenge with empirical data (traced): the 2017 ismrm diffusion study group challenge. *Journal of Magnetic Resonance Imaging*, 51(1):234–249, 2020.
- [79] Lukas Neumann, Andrew Zisserman, and Andrea Vedaldi. Relaxed softmax: Efficient confidence auto-calibration for safe pedestrian detection. *2018 NIPS Workshop on Machine Learning for Intelligent Transportation Systems*, 2018.
- [80] Stanislav Nikolov, Sam Blackwell, Alexei Zverovitch, Ruheena Mendes, Michelle Livne, Jeffrey De Fauw, Yojan Patel, Clemens Meyer, Harry Askham, Bernadino Romera-Paredes, et al. Clinically applicable segmentation of head and neck anatomy for radiotherapy: deep learning algorithm development and validation study. *Journal of Medical Internet Research*, 23(7):e26151, 2021.
- [81] Stephen G Pauker and Jerome P Kassirer. Therapeutic decision making: a cost-benefit analysis. *New England Journal of Medicine*, 293(5):229–234, 1975.
- [82] Alexandre Perez-Lebel, Marine Le Morvan, and Gaël Varoquaux. Beyond calibration: estimating the grouping loss of modern neural networks. *International Conference on Learning Representations*, 2023.
- [83] Teodora Popordanoska, Raphael Sayer, and Matthew B Blaschko. A consistent and differentiable lp canonical calibration error estimator. In *Advances in Neural Information Processing Systems*, 2022.
- [84] Joaquin Quinonero-Candela, Carl Edward Rasmussen, Fabian Sinz, Olivier Bousquet, and Bernhard Schölkopf. Evaluating predictive uncertainty challenge. In *Machine Learning Challenges Workshop*, pages 1–27. Springer, 2005.
- [85] Annika Reinke, Matthias Eisenmann, Minu D Tizabi, Carole H Sudre, Tim Rädtsch, Michela Antonelli, Tal Arbel, Spyridon Bakas, M Jorge Cardoso, Veronika Cheplygina, et al. Common limitations of image processing metrics: A picture story. *arXiv preprint arXiv:2104.05642*, 2021.

- [86] Annika Reinke, Minu D. Tizabi, Michael Baumgartner, Matthias Eisenmann, Doreen Heckmann-Nötzel, Emre Kavur, Tim Rädtsch, Carole Sudre, et al. Understanding metric-related pitfalls in image analysis validation. *arXiv preprint arXiv:2302.01790; sister publication jointly submitted with this work*, 2023.
- [87] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.
- [88] Frank W Samuelson and Nicholas Petrick. Comparing image detection algorithms using resampling. In *3rd IEEE International Symposium on Biomedical Imaging: Nano to Macro*, 2006., pages 1312–1315. IEEE, 2006.
- [89] Cristina Sánchez-Montes, Francisco Javier Sánchez, Jorge Bernal, Henry Córdova, María López-Cerón, Miriam Cuatrecasas, Cristina Rodríguez De Miguel, Ana García-Rodríguez, Rodrigo Garcés-Durán, María Pellisé, et al. Computer-aided prediction of polyp histology on white light colonoscopy using surface pattern analysis. *Endoscopy*, 51(03):261–265, 2019.
- [90] Jörg Sander, Bob D de Vos, Jelmer M Wolterink, and Ivana Išgum. Towards increased trustworthiness of deep learning segmentation methods on cardiac mri. In *Medical imaging 2019: image Processing*, volume 10949, pages 324–330. SPIE, 2019.
- [91] Kenneth F Schulz, Douglas G Altman, David Moher, and CONSORT Group*. Consort 2010 statement: updated guidelines for reporting parallel group randomized trials. *Annals of internal medicine*, 152(11):726–732, 2010.
- [92] Arnaud Arindra Adiyoso Setio, Alberto Traverso, Thomas De Bel, Moira SN Berens, Cas Van Den Bogaard, Piergiorgio Cerello, Hao Chen, Qi Dou, Maria Evelina Fantacci, Bram Geurts, et al. Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: the luna16 challenge. *Medical image analysis*, 42:1–13, 2017.
- [93] Suprosanna Shit, Johannes C Paetzold, Anjany Sekuboyina, Ivan Ezhov, Alexander Unger, Andrey Zhylyka, Josien PW Pluim, Ulrich Bauer, and Bjoern H Menze. cldice-a novel topology-preserving loss function for tubular structure segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16560–16569, 2021.
- [94] Amber L Simpson, Michela Antonelli, Spyridon Bakas, Michel Bilello, Keyvan Farahani, Bram Van Ginneken, Annette Kopp-Schneider, Bennett A Landman, Geert Litjens, Bjoern Menze, et al. A large annotated medical image dataset for the development and evaluation of segmentation algorithms. *arXiv preprint arXiv:1902.09063*, 2019.
- [95] Viknesh Sounderajah, Hutan Ashrafian, Ravi Aggarwal, Jeffrey De Fauw, Alastair K Denniston, Felix Greaves, Alan Karthikesalingam, Dominic King, Xiaoxuan Liu, Sheraz R Markar, Matthew D F McInnes, Trishan Panch, Jonathan Pearson-Stuttard, Daniel S W Ting, Robert M Golub, David Moher, Patrick M Bossuyt, and Ara Darzi. Developing specific reporting guidelines for diagnostic accuracy studies assessing AI interventions: The STARD-AI steering group. *Nat. Med.*, 26(6):807–808, June 2020.
- [96] Ewout W Steyerberg, Andrew J Vickers, Nancy R Cook, Thomas Gerds, Mithat Gonen, Nancy Obuchowski, Michael J Pencina, and Michael W Kattan. Assessing the performance of prediction models: a framework for some traditional and novel measures. *Epidemiology (Cambridge, Mass.)*, 21(1):128, 2010.
- [97] Abdel Aziz Taha and Allan Hanbury. Metrics for evaluating 3d medical image segmentation: analysis, selection, and tool. *BMC medical imaging*, 15(1):1–28, 2015. The paper discusses the importance of effective metrics for evaluating the accuracy of 3D medical image segmentation algorithms. The authors analyze existing metrics, propose a selection methodology, and develop a tool to aid researchers in choosing appropriate evaluation metrics based on the specific characteristics of the segmentation task.
- [98] Anna Targosz, Piotr Przyszałka, Ryszard Wiaderekiewicz, and Grzegorz Mrugacz. Semantic segmentation of human oocyte images using deep neural networks. *BioMedical Engineering OnLine*, 20(1):40, 2021.
- [99] Alaa Tharwat. Classification assessment methods. *Applied Computing and Informatics*, 2020.
- [100] Kimberley M Timmins, Irene C van der Schaaf, Edwin Bennink, Ynte M Ruijgrok, Xingle An, Michael Baumgartner, Pascal Bourdon, Riccardo De Feo, Tommaso Di Noto, Florian Dubost, et al. Comparing methods of detecting and segmenting unruptured intracranial aneurysms on tof-mras: The adam challenge. *Neuroimage*, 238:118216, 2021.
- [101] Laszlo Tirian and Barry J Dickson. The vt gal4, lexa, and split-gal4 driver line collections for targeted expression in the drosophila nervous system. *BioRxiv*, page 198648, 2017.
- [102] Thuy N Tran, Tim Adler, Amine Yamlahi, Evangelia Christodoulou, Patrick Godau, Annika Reinke, Minu D Tizabi, Peter Sauer, Tillmann Persicke, Jörg G. Albert, and Lena Maier-Hein. Sources of performance variability in deep learning-based polyp detection. *arXiv preprint arXiv:2211.09708*, 2022.
- [103] Vladimír Ulman, Martin Maška, Klas EG Magnusson, Olaf Ronneberger, Carsten Haubold, Nathalie Harder, Pavel Matula, Petr Matula, David Svoboda, Miroslav Radojevic, et al. An objective comparison of cell-tracking algorithms. *Nature methods*, 14(12):1141–1152, 2017.
- [104] Richard Usatine and Rachel Mancini. Dermoscopedia, 2021. https://dermoscopedia.org/File:DF_chinese_dms.JPG.
- [105] Juozas Vaicenavicius, David Widmann, Carl Andersson, Fredrik Lindsten, Jacob Roll, and Thomas Schön. Evaluating model calibration in classification. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages

- 3459–3467. PMLR, 2019.
- [106] Ben Van Calster, Daan Nieboer, Yvonne Vergouwe, Bavo De Cock, Michael J Pencina, and Ewout W Steyerberg. A calibration hierarchy for risk models was defined: from utopia to empirical data. *Journal of clinical epidemiology*, 74: 167–176, 2016.
 - [107] Ben Van Calster, David J McLernon, Maarten Van Smeden, Laure Wynants, and Ewout W Steyerberg. Calibration: the achilles heel of predictive analytics. *BMC medicine*, 17(1):1–7, 2019.
 - [108] Bram Van Ginneken, Samuel G Armato III, Bartjan de Hoop, Saskia van Amelsvoort-van de Vorst, Thomas Duindam, Meindert Niemeijer, Keelin Murphy, Arnold Schilham, Alessandra Retico, Maria Evelina Fantacci, et al. Comparing and combining algorithms for computer-aided detection of pulmonary nodules in computed tomography scans: the anode09 study. *Medical image analysis*, 14(6):707–722, 2010.
 - [109] C Van Rijsbergen. Information retrieval: theory and practice. In *Proceedings of the Joint IBM/University of Newcastle upon Tyne Seminar on Data Base Systems*, volume 79, 1979.
 - [110] Andrew J Vickers and Elena B Elkin. Decision curve analysis: a novel method for evaluating prediction models. *Medical Decision Making*, 26(6):565–574, 2006.
 - [111] Andrew J Vickers, Ben Van Calster, and Ewout W Steyerberg. Net benefit approaches to the evaluation of prediction models, molecular markers, and diagnostic tests. *bmj*, 352, 2016.
 - [112] David S Wack, Michael G Dwyer, Niels Bergsland, Carol Di Perri, Laura Ranza, Sara Hussein, Deepa Ramasamy, Guy Poloni, and Robert Zivadinov. Improved assessment of multiple sclerosis lesion segmentation agreement via detection and outline error estimates. *BMC medical imaging*, 12(1):1–10, 2012.
 - [113] Matthijs J Warrens. Some paradoxical results for the quadratically weighted kappa. *Psychometrika*, 77(2):315–323, 2012.
 - [114] Jonathan Wenger, Hedvig Kjellström, and Rudolph Triebel. Non-parametric calibration for classification. In *International Conference on Artificial Intelligence and Statistics*, pages 178–190. PMLR, 2020.
 - [115] David Widmann, Fredrik Lindsten, and Dave Zachariah. Calibration tests in multi-class classification: A unifying framework. *Advances in Neural Information Processing Systems*, 32, 2019.
 - [116] Manuel Wiesenfarth, Annika Reinke, Bennett A Landman, Matthias Eisenmann, Laura Aguilera Saiz, M Jorge Cardoso, Lena Maier-Hein, and Annette Kopp-Schneider. Methods and open-source toolkit for analyzing and visualizing challenge results. *Scientific reports*, 11(1):1–15, 2021.
 - [117] Laure Wynants, Maarten Van Smeden, David J McLernon, Dirk Timmerman, Ewout W Steyerberg, and Ben Van Calster. Three myths about risk thresholds for prediction models. *BMC medicine*, 17(1):1–7, 2019.
 - [118] Yinchong Yang and Florian Buettner. Multi-output gaussian processes for uncertainty-aware recommender systems. In *Uncertainty in Artificial Intelligence*, pages 1505–1514. PMLR, 2021.
 - [119] Varduhi Yeghiazaryan and Irina Voiculescu. An overview of current evaluation methods used in medical image segmentation. *Department of Computer Science, University of Oxford*, 2015.
 - [120] Varduhi Yeghiazaryan and Irina D Voiculescu. Family of boundary overlap metrics for the evaluation of medical image segmentation. *Journal of Medical Imaging*, 5(1):015006, 2018.
 - [121] Ying Zhang, Yubin Xie, Wenzhong Liu, Wankun Deng, Di Peng, Chenwei Wang, Haodong Xu, Chen Ruan, Yongjie Deng, Yaping Guo, et al. Deepphagy: a deep learning framework for quantitatively measuring autophagy activity in *saccharomyces cerevisiae*. *Autophagy*, 16(4):626–640, 2020.
 - [122] Qiuming Zhu. On the performance of matthews correlation coefficient (mcc) for imbalanced dataset. *Pattern Recognit. Lett.*, 136:71–80, 2020.