A Probabilistic Method to Predict Classifier Accuracy on Larger Datasets given Small Pilot Data

Ethan Harvey[1] ETHAN.HARVEY@TUFTS.EDU
Wansu Chen[2] WANSU.CHEN@KP.ORG
David M. Kent[3] DAVID.KENT@TUFTSMEDICINE.ORG
Michael C. Hughes[1] MICHAEL.HUGHES@TUFTS.EDU
[1] *Department of Computer Science, Tufts University, Medford, MA, USA*
[2] *Department of Research and Evaluation, Kaiser Permanente Southern California, Pasadena, CA, USA*
[3] *Predictive Analytics and Comparative Effectiveness Center, Tufts Medical Center, Boston, MA, USA*

Abstract

Practitioners building classifiers often start with a smaller pilot dataset and plan to grow to larger data in the near future. Such projects need a toolkit for extrapolating how much classifier accuracy may improve from a 2x, 10x, or 50x increase in data size. While existing work has focused on finding a single "best-fit" curve using various functional forms like power laws, we argue that modeling and assessing the *uncertainty* of predictions is critical yet has seen less attention. In this paper, we propose a Gaussian process model to obtain probabilistic extrapolations of accuracy or similar performance metrics as dataset size increases. We evaluate our approach in terms of error, likelihood, and coverage across six datasets. Though we focus on medical tasks and image modalities, our open source approach[1] generalizes to any kind of classifier.

Keywords: Learning curve; Gaussian process

1. Introduction

Consider the development of a medical image classifier for a new diagnostic task. In this and other applications of supervised machine learning, the biggest key to success is often the size of the available labeled training set. When a large dataset of labeled images is not available, research projects often have a common trajectory: (1) gather a small "pilot" dataset of images and corresponding class labels, (2) train classifiers using this available data, and then (3) plan to collect an even larger dataset to further improve performance. When gathering more labeled data is expensive, practitioners face a key decision in step 3: *given that the classifier's accuracy is y% at the current size x, how much better might the model do at* $2x$, $10x$, *or* $50x$ *images?*

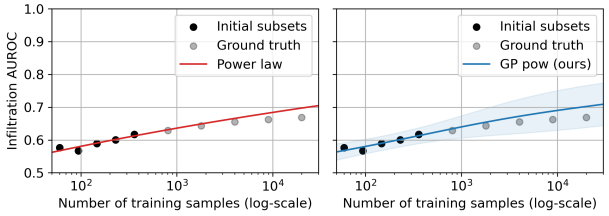

Figure 1: Example learning curves for predicting infiltration from chest x-rays assessed using area under the receiver operating characteristic (AUROC). Left: Single "best-fit" using power law (Rosenfeld et al., 2020). Right: Our probabilistic Gaussian process with a power law mean function and 95% confidence interval for uncertainty.

Despite decades of research, practitioners lack standardized tools to help answer this question of how to extrapolate classifier performance to larger datasets. We argue that tools that help *manage uncertainty* are especially needed. As illustrated in the left panel of Fig. 1, recent approaches have focused almost entirely on estimating one single "best-fit" curve, using power laws (Hestness et al., 2017; Rosenfeld et al., 2020), piecewise power laws (Jain et al., 2023), or other functional forms (Mahmood et al., 2022a). In practice, however, no single curve fit to a limited set of size-accuracy pairs can extrapolate perfectly. Probabilistic methods that can model a *range* of plausible curves, as illustrated in Fig. 1 (right), are thus a more natural solution. Surprisingly, however, most existing methods focus on deterministic rather than probabilistic modeling (see Tab. 1). The few methods that do explain how to provide probabilistic intervals for their

1. We open source our code at https://github.com/tufts-ml/extrapolating-classifier-accuracy-to-larger-datasets

predictions lack careful evaluation of their ability to quantify uncertainty.

In this work, we provide a portable modeling toolkit to help practitioners extrapolate classifier performance *probabilistically*. We pursue a Bayesian approach via an underlying Gaussian process (GP) model (Rasmussen et al., 2006). The mean function of our GP can match common curve forms like power law or arctan, but is adjusted to encode desirable inductive biases (e.g., more data implies better performance) as well as common sense (e.g., accuracy or AUROC can never exceed 100%). We further use prior distributions over model hyperparameters to encode domain-specific knowledge (e.g., where might accuracy saturate for this task given enough data). The whole pipeline remains learnable from a handful of size-accuracy exemplars gathered on a small pilot dataset.

To summarize, our contributions are: (1) a reusable GP-based accuracy probabilistic extrapolator (APEx-GP) that can match existing curve-fitting approaches in terms of error while providing additional uncertainty estimates, and (2) a careful assessment of our proposed probabilistic extrapolations compared to ground truth on larger datasets across six medical classification tasks involving both 2D and 3D images across diverse modalities (x-ray, ultrasound, and CT) with various sample sizes. While we focus on image analysis tasks here, nothing about our methodology is specialized to images; our pipeline could be repurposed for tabular data, genomics, text, time series, or even heterogeneous data from many domains. Ultimately, we hope our methods help research teams and sponsoring funding agencies assess data adequacy for proposed research studies.

2. Background

We wish to build a classifier for a task of interest using a bespoke dataset. We assume the largest available labeled dataset $\mathcal{D}$ for our task has *limited size*, which we operationalize as roughly 500-20000 total images with corresponding labels. We partition $\mathcal{D}$ into non-overlapping training and test sets. Given a chosen classifier and a specific data partition with x training images, we fit the classifier (including hyperparameter search) then evaluate on that partition's test set, obtaining a performance value $y \in [0.0, 1.0]$. We'll assume throughout that higher y implies a better model; we will informally refer to y as "accuracy" for convenience, though y could measure any common classifier metric like AUROC or balanced accuracy where 1.0 means "perfect".

To estimate how y changes with dataset size using available data, we construct a handful of nested *subsets*, following Mahmood et al. (2022a); Jain et al. (2023). First, pick R desired train-set sizes $\{x_r\}_{r=1}^R$ in increasing order. Next, stochastically sample training sets $\mathcal{S}_1 \subset \mathcal{S}_2 \subset \ldots \subset \mathcal{S}_R$, such that $|\mathcal{S}_r| = x_r$ for each index r. Also sample a non-overlapping test set. Finally, fit the classifier to each train set $\mathcal{S}_r$ and record the performance on the test set as y_r. Averaging over y_r from multiple random partitions of $\mathcal{D}$ into train and test sets can obtain smoother estimates of heldout performance at each data size x_r.

Problem statement. We now present two possible formulations of our extrapolation problem.

(i) Point estimate extrapolation. Given a small dataset of R size-accuracy pairs $\{x_r, y_r\}_{r=1}^R$, fit a function $f_\theta(x)$ so that we can extrapolate classifier accuracy on larger datasets with size $x_* > x_R$.

(ii) Probabilistic extrapolation. Given a small dataset of R size-accuracy pairs $\{x_r, y_r\}_{r=1}^R$, fit a probability density function $p_\theta(Y|x)$ treating accuracy Y as the random variable so that we can extrapolate classifier accuracy on larger datasets with appropriate uncertainty.

Evaluation metrics. Given a heldout set of size-accuracy pairs x_*, y_*, we consider several evaluation metrics. The first applies to both deterministic (i) and probabilistic (ii) methods. The latter two are only for probabilistic methods.

- Error (i or ii). For each heldout pair x_*, y_*, assess the error between y_* and $f_\theta(x_*)$, via *root mean squared error* or *mean absolute error*.
- Quantized likelihood (type ii only). For each pair x_*, y_*, we compute the probability mass assigned to a narrow interval around the true observation $p(Y \in (y_* - \Delta, y_* + \Delta)|x_*)$, with $\Delta = 0.01$. Assessing an interval ensures this metric is always between 0 and 100% (higher is better).
- Coverage (type ii only) (Dodge, 2003, p. 93). Here we assume that for each x_* we observe many replicates of y_* (via re-sampling train-test splits). From the model's probability density function (PDF) $p(Y|x_*)$ we obtain an interval y_a, y_b corresponding to the P% high-density interval. We then measure the fraction of times the measured replicates of y_* falls in that interval: this empirical fraction should match P% if the model is well-calibrated.

Table 1: Related work focused on predicting model performance.

Related work	Models uncertainty?	Evaluates uncertainty?	Validated on medical data?
Power law (Rosenfeld et al., 2020)	No	N/A	No
Arctan + other functions (Mahmood et al., 2022a)	No	N/A	No
Learn-Optimize-Collect (Mahmood et al., 2022b)	Post-hoc PDF over x not y	No	No
Piecewise power law (Jain et al., 2023)	Yes, but PDF $p(y\|x)$ via asymptotic formulas	No	No
APEx-GP (ours)	Yes, via direct model	Yes	Yes

3. Related Work

Data scaling. Prior works have empirically validated that generalization in deep learning scales with dataset size according to a power law function in both computer vision (Sun et al., 2017; Bahri et al., 2021; Hoiem et al., 2021; Zhai et al., 2022) and natural language processing (Hestness et al., 2017; Kaplan et al., 2020). Sun et al. (2017) find that image classification accuracy increases logarithmically based on the training dataset size. Hestness et al. (2017) show test loss decreases according to a power law as training dataset size increases in machine translation, language modeling, image processing, and speech recognition. Zhai et al. (2022) scale vision transformer (ViT) (Dosovitskiy et al., 2021) models and data, both up and down, and find the power law characterizes the relationships between error rate, data, and compute.

Predicting model performance. Other works look at predicting model performance at larger dataset sizes (Cortes et al., 1993; Frey and Fisher, 1999; Johnson et al., 2018; Rosenfeld et al., 2020; Jain et al., 2023) and estimating data requirements given performance targets (Mahmood et al., 2022a,b) (see Tab. 1). Rosenfeld et al. (2020) develop a model for predicting performance given a specified model. They find that errors are larger when extrapolating from smaller dataset sizes. Jain et al. (2023) propose a piecewise power law that models performance as a quadratic curve in the few-shot setting and a linear curve in the high-shot setting. They estimate confidence intervals using a formula from Gavin (2019) inspired by estimators of covariance matrices for parameters fit by maximum likelihood (Murphy, 2022). However, such estimators are only justified *asymptotically* as sample size increases; use when estimating from only a few data points seems questionable.

Mahmood et al. (2022a) consider a broad class of computer vision tasks and systematically investigate a family of functions that generalize the power law function to allow for better estimation of data requirements. They focus on estimating target data requirements given an approximate relationship between data size and model performance; such as a power law function. Mahmood et al. (2022b) propose a new paradigm for modeling the data collection workflow as a formal optimal data collection problem that allows designers to specify performance targets, collections costs, a time horizon, and penalties for failing to meet the targets. They estimate the distribution of x that achieves the target performance y by bootstrapping size-accuracy pairs, estimating the dataset size, and fitting a density estimation model; in contrast, we directly model uncertainty in y.

Sample size estimation. Loosely related to our work are traditional sample size estimation calulations. Riley et al. (2020) provide practical guidance for calculating the sample size required for the development of clinical prediction models. These include calculations that might identify datasets that are too small (for example, if overall outcome risk cannot be estimated precisely).

4. Probabilistic Model

We now develop our approach to modeling a probability density function $p(y|x)$ that can estimate the distribution in accuracy y at any specific training set size x.

4.1. GP extrapolation model

For each possible dataset size x, we imagine there is an unobservable random variable f representing *true* classifier performance, as well as an observable random variable y representing *realized* classifier performance on a finite test set. To achieve a flexible model for function $f(x)$, we turn to a Gaussian process

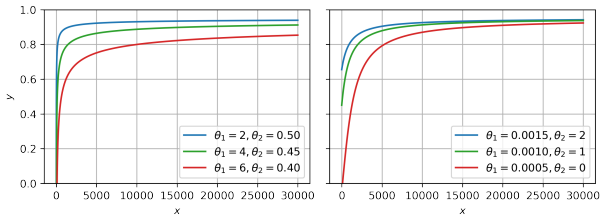

Figure 2: Example varying parameters for the power law (left) and arctan function (right) with $\varepsilon = 0.05$.

prior with mean function m and covariance function k. Given true accuracy f, we then model each observable accuracy y as a perturbation of true accuracy f by independant and identically distributed (IID) Gaussian noise with scalar variance τ^2. When we condition on a finite set of data size inputs $\mathbf{x} = \{x_r\}_{r=1}^R$ of interest, our model's joint distribution $p(\mathbf{y}, \mathbf{f}|\mathbf{x})$ factorizes as

$$p(\mathbf{f} \mid \mathbf{x}) = \mathcal{N}(\mathbf{f} \mid m(\mathbf{x}), k(\mathbf{x}, \mathbf{x})) \quad (1)$$
$$p(\mathbf{y}|\mathbf{f}, \mathbf{x}) = \prod_{r=1}^R \mathcal{N}(y_r|f_r, \tau^2)$$

where $\mathbf{f}, \mathbf{y}$, and $m(\mathbf{x})$ are each R-dimensional vectors whose entry at index r corresponds to the provided data size input x_r. Similarly, $k(\mathbf{x}, \mathbf{x})$ is an $R \times R$ covariance matrix, with entry s, t equal to $k(x_s, x_t)$.

Below, we provide recommended options for both mean m and covariance function k. We pay particular attention to the mean, offering two concrete choices, a power law and an arctan, inspired by the best performing methods from prior work on point estimation of "best-fit" curves for size-accuracy extrapolation. Fig. 2 shows both possible mean functions across a range of parameters θ while saturating at a maximum accuracy of $1 - \varepsilon$. Other uses of GPs in practice often assume a constant mean of zero; we select forms that deliberately allow accuracy to grow as more data is added.

Power law mean. Our power law function is

$$m^{\text{pow}}(x) = (1 - \varepsilon) - \theta_1 x^{\theta_2}, \quad (2)$$

with two trainable parameters $\theta_1 \geq 0$ and $\theta_2 \in [-1, 0]$. Similar power law forms have been previously recommended (Cortes et al., 1993; Frey and Fisher, 1999; Johnson et al., 2018; Rosenfeld et al., 2020). Our version guarantees that $\lim_{x\to\infty} m(x) = 1 - \varepsilon$.

Arctan mean. Our arctan mean function is

$$m^{\text{arc}}(x) = \frac{2}{\pi} \arctan\left(\theta_1 \frac{\pi}{2} x + \theta_2\right) - \varepsilon \quad (3)$$

with two trainable parameters: $\theta_1 \geq 0$ and $\theta_2 \geq 0$. Similar forms were recommended by recent work (Mahmood et al., 2022a). Our modified version guarantees that $\lim_{x\to\infty} m(x) = 1 - \varepsilon$.

Encoding saturation limits via ε. Our definition of both power law and arctan means above deliberately includes an ε term to allow domain experts to define how the function should saturate as the training set size grows $x \to \infty$. Setting $\varepsilon = 0$ allows a "perfect classifier" with $m(x) = 1$, while setting $\varepsilon = 0.05$ reflects a lower ceiling that may be more appropriate. Even with an infinite training set, we might not be able to build a perfect classifier (e.g., due to image noise, label noise, insufficiency of images alone for the diagnostic task, and interrater reliability issues).

Covariance function. The classic radial basis function (RBF) kernel (Rasmussen et al., 2006) applied to *logarithms* of input sizes defines our covariance:

$$k(x, x') = \sigma^2 \exp\left(-\frac{(\log(x) - \log(x'))^2}{2\lambda^2}\right) \quad (4)$$

where both the output-scale $\sigma > 0$ and length-scale $\lambda > 0$ are trainable parameters. This log-RBF form implies that $f(x)$ and $f(x')$ values have high covariance at similar sizes x and x' , while at very different sizes (numerator $\gg 2\lambda^2$) the f values have near-zero covariance. We select the log-RBF because it tends to produce *smooth* functions $f(x)$, and we expect the idealized classifier accuracy as a function of data size to be smooth (Mahmood et al., 2022a,b). We expect selecting other stationary kernels like Matern or avoiding the log would have relatively minor impact on overall model quality, as long as output scale and length scale are trainable.

4.2. Prior control of extrapolation uncertainty

Our GP model has two kinds of parameters. Let $\eta = \{\tau, \sigma, \lambda, \varepsilon\}$ denote the parameters that control *uncertainty* (in the likelihood or the GP covariance function) or *asymptotic behavior* (e.g., ε sets the saturation value of $m(x)$). We argue that domain knowledge can and should guide learning of η. In this section, we develop prior distributions for each parameter in η. In contrast, parameters $\theta = \{\theta_1, \theta_2\}$ define the shape of the mean function and thus are more straightforward to estimate even given a small training set of R size-accuracy pairs. We do not define priors for θ.

Likelihood scale τ. The scalar $\tau > 0$ represents the standard deviation of each realized accuracy y given ideal accuracy f. We expect *a priori* that realized accuracy y does not vary too much around f; any deviation of more than 0.03 seems undesirable.

We operationalize our desiderata (details in App. A.2) with a truncated normal prior (Fisher, 1931).

Kernel output-scale σ. Scalar $\sigma > 0$ controls the magnitude of variance for random variable f. Along with τ, it also controls the variance of accuracy y if f is marginalized away (as we do in extrapolation). Given only a small dataset of size-accuracy pairs, we *a priori* should have considerable uncertainty about realized accuracy $y(x_*)$ at sizes x_* much larger than seen in the pilot set. This matches past empirical evidence (Rosenfeld et al., 2020; Mahmood et al., 2022a; Jain et al., 2023). We operationalize the prior as a truncated normal and do a numerical grid search for mean and variance hyperparameters such that a desired wide 20-80 percentile range is satisfied (details in App. A.2).

Kernel length-scale λ. Scalar $\lambda > 0$ controls the rate at which the correlation between $f(x)$ and $f(x')$ decreases as the distance between $\log(x)$ and $\log(x')$ increases. A reasonable *a priori* belief is that strong correlations may only exist when the distance between x and x' is less than $1.5x$; at larger distances we should not expect strong correlation as the dataset would have nearly doubled in size. We chose a truncated normal prior that which gives roughly the desired behavior (details in App. A.2).

Saturation limit ε. Scalar $\varepsilon \geq 0$ represents one minus the maximum accuracy we expect as dataset size gets asymptotically large, using domain knowledge. Given plausible lower and upper bounds from domain experts (see App. A.1 and A.2), we form a uniform prior over ε.

4.3. Fitting to data via MAP estimation

Given a training set of R size-accuracy pairs, represented by sizes $\mathbf{x}$ and accuracies $\mathbf{y}$, fitting the model means estimating the parameters θ and η, defined early in Sec. 4.2. We can take advantage of our model's conjugacy to integrate away our latent variable $\mathbf{f}$. This leaves the marginal likelihood of the observable training set as:

$$p_{\theta,\eta}(\mathbf{y}|\mathbf{x}) = \int_{\mathbf{f}} p_\eta(\mathbf{y}|\mathbf{f},\mathbf{X}) p_{\theta,\eta}(\mathbf{f}|\mathbf{X}) d\mathbf{f}. \tag{5}$$

We then optimize the following maximum a-posteriori (MAP) objective to obtain point estimates of θ and η:

$$\hat{\theta},\hat{\eta} = \underset{\theta,\eta}{\operatorname{argmax}} \ \log p_{\theta,\eta}(\mathbf{y}|\mathbf{x}) + \log p(\eta). \tag{6}$$

The objective works in log space for numerical stability, where the log of the marginal likelihood has the following closed-form

$$\begin{aligned}\log p(\mathbf{y}|\mathbf{x}) = &-\frac{R}{2}\log 2\pi - \frac{1}{2}\log|\mathbf{K}+\tau^2 I_R| \\ &-\frac{1}{2}(\mathbf{y}-\mathbf{m})^T(\mathbf{K}+\tau^2 I_R)^{-1}(\mathbf{y}-\mathbf{m}).\end{aligned} \tag{7}$$

Here, $\mathbf{K}$ is a $R \times R$ matrix defined as $k(\mathbf{x},\mathbf{x})$ and depends on $\sigma, \lambda \in \eta$. Vector $\mathbf{m}$ is defined as $m(\mathbf{x})$ and depends on θ and ε. We suppressed the explicit dependence on θ, η in this notation for simplicity.

4.4. Extrapolation via the posterior predictive

Given a fit model via estimates $\hat{\theta},\hat{\eta}$, our target use for our model is to extrapolate probabilistically. Conditioning on a pilot training set of R size-accuracy pairs $\mathbf{x},\mathbf{y}$, we wish to estimate the posterior over accuracies y_* at a set of Q larger dataset sizes x_*. Again using well-known properties of GPs, specifically the joint-to-conditional transformation of Gaussian variables (details in App. A.3), we can directly compute the PDF of the posterior predictive

$$\begin{aligned}p(\mathbf{y}_*|\mathbf{x}_*,\mathbf{y},\mathbf{x}) = \ &\mathcal{N}(\mathbf{y}_*|\mu,\mathbf{\Sigma}), \\ &\mu = \mathbf{m}_* + \mathbf{K}_*^T(\mathbf{K}+\tau^2 I_R)^{-1}(\mathbf{y}-\mathbf{m}) \\ &\mathbf{\Sigma} = \mathbf{K}_{**} + \tau^2 I_Q - \mathbf{K}_*^T(\mathbf{K}+\tau^2 I_R)^{-1}\mathbf{K}_*.\end{aligned} \tag{8}$$

Here K_* is a $R \times Q$ matrix and K_{**} is a $Q \times Q$ matrix, where each entry is a call to covariance function k with appropriate inputs from the train or test sets. All calls to m or k use estimated parameters $\hat{\theta},\hat{\eta}$.

Constraining y to [0,1]. A careful reader will note that in our application the "accuracy" y must be a positive real confined to the interval [0.0, 1.0]. In contrast, throughout our modeling derivation we allow y broader support over the whole real line. We chose this broader support for the computational convenience it provides at training time: our training objective in Eq. (6) and our posterior predictive in Eq. (8) are both computable in closed-form. To avoid extraneous predictions, for any scalar invocation of Eq. (8) after forming the univariate posterior prediction, we truncate the predictive density to the unit interval [0.0, 1.0]. This support broadening has statistical justification (Wojnowicz et al., 2023).

5. Experimental Procedures

We now describe how we gather the size-accuracy pairs used to train and evaluate models from various 2D and 3D medical imaging datasets. We then outline

procedures for assessing predictions for both short-range (up to $2x$ the train set size) and long-range (up to $50x$) settings.

5.1. Datasets and Classifier Procedures

Our chosen 2D datasets all use 224x224 resolution and span x-ray and ultrasound modalities, including ChestX-ray14 (Wang et al., 2017), Kermany et al. (2018)'s Chest X-Ray Pneumonia dataset, the Breast Ultrasound Image dataset (BUSI) (Al-Dhabyani et al., 2020), and the Tufts Medical Echocardiogram Dataset (TMED-2) (Huang et al., 2021, 2022). We also study two 3D datasets of head CT scans: the Open Access Series of Imaging Studies (OASIS-3) dataset (LaMontagne et al., 2019) and a proprietary pilot neuroimaging dataset. See App. B for detailed dataset descriptions and preprocessing steps.

From each dataset, we form one or more separate binary classification tasks, using only labels with sufficient data (at least 10% prevalence). When raw labels are multiclass, we transform to several one-vs-rest binary tasks. This choice allows using the same pipeline and same evaluation metrics in all analyses, substantially simplifying the work and presentation.

All datasets contain fully deidentified images gathered during routine care. Only the last one is not public. The use of these deidentified images for research has been approved by our Institutional Review Board (Tufts Health Science IRB #11953).

Classifiers. For all tasks, we fine-tune the classification head of a ViT pretrained on ImageNet (Deng et al., 2009). For 2D tasks, the pretrained ViT processes each 224x224 image and produces an image-specific embedding $h_i \in \mathbb{R}^D$, where $D = 768$. We then model the binary label of interest C_i given h_i as

$$C_i | h_i \sim \text{Bern}(C_i \mid \sigma(w^T h_i)), \tag{9}$$

where $w \in \mathbb{R}^D$ are learnable weights and σ is the sigmoid function.

For 3D tasks, we feed each 224x224 2D slice into the pretrained ViT, apply a linear per-slice classifier, and then aggregate via mean pooling to produce a scan-level prediction. Let $h_{i,n} \in \mathbb{R}^D$ denote ViT embedding of the n-th slice (out of N) of the i-th image, where $D = 768$. We model binary label C_i given all embeddings $\mathbf{h}_{i,1:N}$ as

$$C_i | \mathbf{h}_{i,1:N} \sim \text{Bern}(C_i \mid \tfrac{1}{N} \textstyle\sum_{n=1}^{N} \sigma(w^T h_n)), \tag{10}$$

where again $w \in \mathbb{R}^D$ are learnable weights and σ is the sigmoid function. While other flexible 3D architectures are possible, we chose this path for simplicity.

Training details. For all tasks, we fit weights via MAP estimation, maximizing the above Bernoulli likelihoods plus a Gaussian prior on w (also known as weight decay). We fit using L-BFGS for 2D and SGD with momentum for 3D. For grayscale images, we replicate to 3 channels to feed into the 3-channel pretrained ViT. For 3D images, we reduce computational costs by subsampling at most 50 slices.

5.2. Experimental protocol

For each dataset, we randomly assign images at an 8:1:1 ratio into training, validation, and testing sets, ensuring each patient's data belongs to exactly one split and stratifying by class to ensure comparable class frequencies. We repeat this process with three data-split random seeds; each seed selects a different train, validation, and test partition.

Given a split's particular training set, we form $R = 5$ log-spaced subsets with 60 to 360 training samples: $\{60, 94, 147, 230, 360\}$. We select these subset sizes because small pilot datasets used for demonstrating feasibility typically have only a few hundred samples. Log-spacing captures macro trends rather than micro fluctuations. We then evaluate approaches for *short-range extrapolation* on five log-spaced subsets between 360 and 720 training samples $\{414, 475, 546, 627, 720\}$, and *long-range extrapolation* on five log-spaced subsets between 360 and 20000 training samples $\{804, 1796, 4010, 8955, 20000\}$. For both short-range and long-range, we omit any values beyond total available training set size.

When fitting each model on each train-set size, we tune hyperparameters including weight initialization seed, weight decay, and number of epochs (to approximate early stopping, see App. C), selecting the configuration that maximizes validation performance.

At each train-set size x, we record as "accuracy" y the average AUROC. We average across 3 data-split seeds for 2D (15 for 3D) to mitigate high-variance estimates from small test sets. We can then finally fit extrapolation models and assess error (RMSE), quantized likelihood, and coverage (see Sec. 2) using these x, y pairs.

Dense Coverage evaluations. For the two largest 2D datasets, ChestX-ray14 and TMED-2, we evaluate coverage on 100 replicates of the above protocol using *long-range coverage* train-set sizes of $\{5\text{k}, 10\text{k}, 20\text{k}\}$ training samples. Each accuracy value y_* still represents the mean test performance across three distinct data-split seeds. Having 100 replicates at each train-

Table 2: Quantized likelihood evaluations at heldout x, y pairs for short-range and long-range extrapolations, with $\Delta = 0.01$ (see Sec. 2). Standard deviations generated from 500 bootstrapping rounds to select data-split seeds to average across. We bold values with non-overlapping intervals. As a baseline, we form a uniform distribution from the minimum accuracy observed in training up to the task-specific maximum accuracy (Sec. A.1).

			Short-range extrapolation		Long-range extrapolation	
Dataset	**Label**	**Baseline**	**GP pow**	**GP arc**	**GP pow**	**GP arc**
ChestX-ray14	Atelectasis	$6.1 \pm 0.0\%$	$45.2 \pm 4.5\%$	$44.3 \pm 4.8\%$	$\mathbf{29.2 \pm 2.6\%}$	$22.1 \pm 2.6\%$
	Effusion	$6.2 \pm 0.0\%$	$37.7 \pm 4.8\%$	$38.3 \pm 4.4\%$	$15.3 \pm 1.9\%$	$15.2 \pm 2.0\%$
	Infiltration	$4.6 \pm 0.0\%$	$\mathbf{44.7 \pm 3.8\%}$	$24.0 \pm 5.4\%$	$\mathbf{25.2 \pm 2.4\%}$	$1.1 \pm 1.4\%$
Chest X-Ray	Bacterial	$11.3 \pm 0.0\%$	$42.3 \pm 8.0\%$	$42.5 \pm 7.7\%$	$38.2 \pm 10.5\%$	$43.5 \pm 8.6\%$
	Viral	$6.4 \pm 0.0\%$	$39.8 \pm 6.2\%$	$38.9 \pm 6.6\%$	$12.3 \pm 5.2\%$	$\mathbf{24.6 \pm 6.9\%}$
BUSI	Normal	$20.3 \pm 0.0\%$	$48.8 \pm 9.4\%$	$48.8 \pm 9.1\%$	—	—
	Benign	$8.5 \pm 0.0\%$	$27.9 \pm 10.8\%$	$28.7 \pm 11.2\%$	—	—
	Malignant	$15.1 \pm 0.0\%$	$27.0 \pm 13.3\%$	$27.5 \pm 12.5\%$	—	—
TMED-2	PLAX	$20.8 \pm 0.0\%$	$65.6 \pm 1.5\%$	$64.5 \pm 1.9\%$	$\mathbf{64.8 \pm 1.0\%}$	$39.0 \pm 1.6\%$
	PSAX	$9.6 \pm 0.0\%$	$62.6 \pm 1.5\%$	$62.6 \pm 1.5\%$	$\mathbf{63.2 \pm 1.1\%}$	$58.4 \pm 1.4\%$
	A4C	$14.3 \pm 0.0\%$	$\mathbf{62.8 \pm 2.4\%}$	$56.1 \pm 3.4\%$	$\mathbf{58.2 \pm 3.2\%}$	$24.5 \pm 3.8\%$
	A2C	$8.5 \pm 0.0\%$	$18.0 \pm 2.5\%$	$\mathbf{61.3 \pm 0.7\%}$	$24.9 \pm 2.8\%$	$20.8 \pm 3.2\%$
OASIS-3	Alzheimer's	$4.6 \pm 0.0\%$	$22.5 \pm 12.5\%$	$23.6 \pm 12.3\%$	—	—
Pilot neuro-	WMD	$6.0 \pm 0.0\%$	$29.3 \pm 14.0\%$	$27.8 \pm 14.6\%$	—	—
imaging dataset	CBI	$5.9 \pm 0.0\%$	$29.6 \pm 14.8\%$	$27.4 \pm 14.4\%$	—	—

set size x_* allows better estimation of the coverage percentage $P\%$. We include single point coverage evaluations for each dataset in App. H.

6. Results

Using the procedures described above, we performed extensive experiments designed to answer several key research questions. First, "which mean function performs best?" Second, "in terms of predictive error, is there a substantial difference between our probablistic approach and previous deterministic approaches?" Finally, "is the coverage obtained by our probabilistic approach compelling?"

Our major findings are highlighted below.

For short-range extrapolation (up to 2x train size), both mean functions seem competitive. Tab. 2 reports quantized likelihoods (higher is better). Looking at short-range results, both mean functions perform similarly. The difference between power law and arctan in 12 of 15 tasks is less than 2%.

For long-range extrapolation (from 2x-50x train size), the power law mean function seems best. Looking at the long-range results in Tab. 2, power law wins clearly in 5 of 6 cases and essentially ties in the other cases. In the other case where arctan wins, power law still clearly outperformas a simpler baseline (arctan can be worse than this baseline sometimes). Power law's superiority in long-range settings is also supported by coverage results in Tab. 3, and RMSE results in App. D). Unlike the power law, the arctan function seems to produce learning curves that asymptote quickly. This results in minimal change in predicted performance after 5000 samples and in some cases overestimates of performance (see curve for infiltration on ChestX-ray14 in Fig. 3).

Error from our GP models is competitive with deterministic models. By design, our GP models use mean functions shown by past work to be effective deterministic predictors. We therefore intend that in terms of pointwise error metrics like root mean squared error, our GP approach is indistinguishable from the non-probabilistic "best-fit" curve approach of past works. RMSE results in App. D suggest that on 13 of 15 short-range tasks and 7 of 9 long-range tasks, our GP power law model is either clearly better or within 0.04 RMSE of deterministic power law.

Our chosen priors improve long-range coverage over no priors. In Tab. 3, we include coverage at 20k training samples from our GP power law model without priors. When performance is low and there is room for variation in performance as dataset size grows, coverage with priors is significantly better than without. However, when performance is high at small

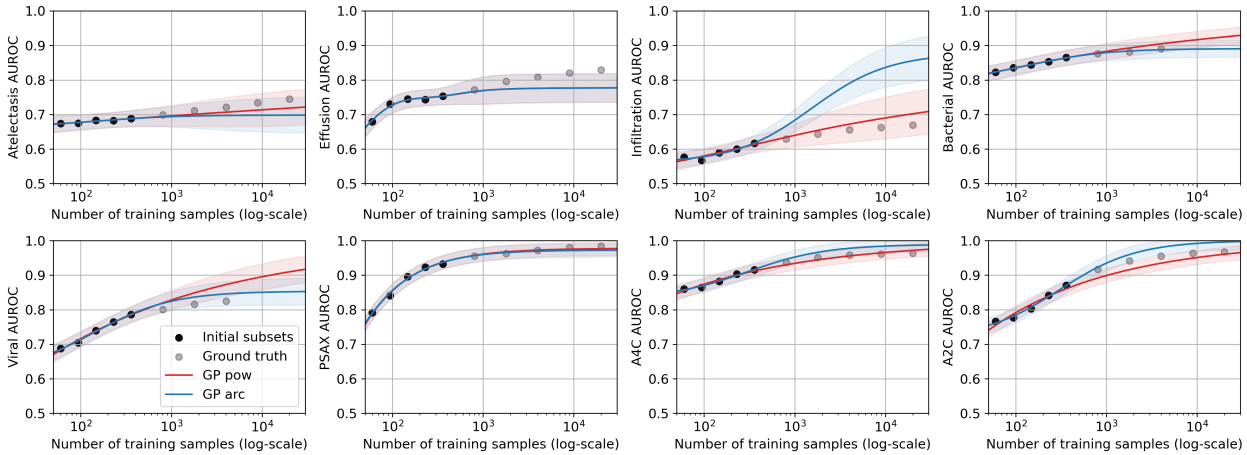

Figure 3: Long-range extrapolation results for atelectasis, effusion, and infiltration from the ChestX-ray14 dataset; bacterial and viral pneumonia from the Chest X-Ray dataset; and PSAX, A4C, and A2C from the TMED-2 dataset.

Table 3: Coverage rates for long-range extrapolations, using a target interval of 95% from our GP model and 100 replicates. Standard deviations generated from 500 bootstrapping rounds to select data-split seeds to average across.

		5k training samples		10k training samples		20k training samples		
Dataset	**Label**	**GP pow**	**GP arc**	**GP pow**	**GP arc**	**GP pow w/o priors**	**GP pow**	**GP arc**
ChestX-ray14	Atelectasis	$100.0 \pm 0.0\%$	$100.0 \pm 0.0\%$	$100.0 \pm 0.0\%$	$100.0 \pm 0.1\%$	$55.4 \pm 2.8\%$	$100.0 \pm 0.0\%$	$99.7 \pm 0.5\%$
	Effusion	$100.0 \pm 0.2\%$	$99.9 \pm 0.3\%$	$94.4 \pm 1.9\%$	$91.0 \pm 2.1\%$	$0.0 \pm 0.0\%$	$48.9 \pm 3.0\%$	$39.6 \pm 2.8\%$
	Infiltration	$100.0 \pm 0.0\%$	$0.0 \pm 0.0\%$	$100.0 \pm 0.0\%$	$0.0 \pm 0.0\%$	$14.5 \pm 2.6\%$	$100.0 \pm 0.0\%$	$0.0 \pm 0.0\%$
TMED-2	PLAX	$100.0 \pm 0.0\%$	$98.5 \pm 1.1\%$	$100.0 \pm 0.0\%$	$83.2 \pm 2.6\%$	$100.0 \pm 0.0\%$	$100.0 \pm 0.0\%$	$45.0 \pm 2.9\%$
	PSAX	$100.0 \pm 0.0\%$	$100.0 \pm 0.0\%$	$100.0 \pm 0.0\%$	$100.0 \pm 0.0\%$	$100.0 \pm 0.0\%$	$100.0 \pm 0.0\%$	$100.0 \pm 0.0\%$
	A4C	$100.0 \pm 0.0\%$	$27.3 \pm 2.8\%$	$100.0 \pm 0.0\%$	$20.5 \pm 2.7\%$	$100.0 \pm 0.1\%$	$100.0 \pm 0.1\%$	$20.7 \pm 2.7\%$
	A2C	$98.5 \pm 1.1\%$	$0.2 \pm 0.4\%$	$100.0 \pm 0.0\%$	$0.1 \pm 0.3\%$	$100.0 \pm 0.0\%$	$100.0 \pm 0.0\%$	$0.2 \pm 0.4\%$

dataset sizes coverage with priors is just as good as without (see coverage for TMED-2 dataset).

Our GP power law model tends to have decent coverage, but over-estimates the intervals a bit. Looking at coverage in Tab. 3, our GP power law model consistently achieves 100% coverage, over-estimating a well calibrated interval. Although wider intervals are preferred to narrow ones that miss the truth, we emphasize that our goal in Tab. 3 is to achieve *95*% coverage.

7. Discussion and Conclusion

We introduced a portable GP-based probabilistic modeling pipeline for classifier performance extrapolation that can match existing curve-fitting approaches in terms of error while providing additional uncertainty estimates. We compared our probabilistic extrapolations to ground truth on 2-50x larger datasets across six medical classification tasks involving both 2D and 3D images across diverse modalities (x-ray, ultrasound, and CT). We recommend the power law mean function based off its superior long-range error, quantized likelihood, and coverage.

Limitations. We acknowledge that APEx-GP is not universally preferred over previous deterministic alternatives; on some tasks (effusion on ChestX-ray14) our error was worse than the power law baseline. More work is needed to understand if our model would be effective beyond the data modalities, train set sizes, classifier architectures, and AUROC metric used here. However, we designed our approach to be effective out-of-the-box for other "accuracy"-like metrics that satisfy two properties: higher is better and 1.0 is a "perfect" score.

Outlook. We hope our approach provides a useful tool for practitioners in medical imaging and beyond to manage uncertainty when assessing data adequacy.

Acknowledgments

This work was supported by NIH grant R01-NS102233, recently renewed as 2 RF1 NS102233-05, as well as a grant from the Alzheimer's Drug Discovery Foundation.

References

Walid Al-Dhabyani, Mohammed Gomaa, Hussien Khaled, and Aly Fahmy. Dataset of breast ultrasound images. *Data in Brief*, 28:104863, 2020. URL https://doi.org/10.1016/j.dib.2019.104863.

Yasaman Bahri, Ethan Dyer, Jared Kaplan, Jaehoon Lee, and Utkarsh Sharma. Explaining Neural Scaling Laws. *arXiv preprint arXiv:2102.06701*, 2021.

Corinna Cortes, Lawrence D Jackel, Sara Solla, Vladimir Vapnik, and John Denker. Learning Curves: Asymptotic Values and Rate of Convergence. In *Advances in Neural Information Processing Systems (NeurIPS)*, 1993.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

Yadolah Dodge. *The Oxford Dictionary of Statistical Terms*. Oxford University Press, 2003.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations (ICLR)*, 2021.

Ronald Fisher. The Truncated Normal Distribution. *British Association for the Advancement of Science*, 5:xxxiii–xxxiv, 1931.

Lewis J Frey and Douglas H Fisher. Modeling Decision Tree Performance with the Power Law. In *Seventh International Workshop on Artificial Intelligence and Statistics (AISTATS)*, 1999.

Henri P Gavin. The Levenberg-Marquardt algorithm for nonlinear least squares curve-fitting problems. *Department of Civil and Environmental Engineering, Duke University*, 19, 2019.

Joel Hestness, Sharan Narang, Newsha Ardalani, Gregory Diamos, Heewoo Jun, Hassan Kianinejad, Md Patwary, Mostofa Ali, Yang Yang, and Yanqi Zhou. Deep Learning Scaling is Predictable, Empirically. *arXiv preprint arXiv:1712.00409*, 2017.

Derek Hoiem, Tanmay Gupta, Zhizhong Li, and Michal Shlapentokh-Rothman. Learning Curves for Analysis of Deep Networks. In *International Conference on Machine Learning (ICML)*, 2021.

Zhe Huang, Gary Long, Benjamin Wessler, and Michael C Hughes. A New Semi-supervised Learning Benchmark for Classifying View and Diagnosing Aortic Stenosis from Echocardiograms. In *Proceedings of the 6th Machine Learning for Healthcare Conference (MLHC)*, 2021.

Zhe Huang, Gary Long, Benjamin Wessler, and Michael C. Hughes. TMED 2: A Dataset for Semi-Supervised Classification of Echocardiograms. In *DataPerf workshop at International Conference on Machine Learning (ICML)*, 2022.

Achin Jain, Gurumurthy Swaminathan, Paolo Favaro, Hao Yang, Avinash Ravichandran, Hrayr Harutyunyan, Alessandro Achille, Onkar Dabeer, Bernt Schiele, Ashwin Swaminathan, et al. A meta-learning approach to predicting performance and data requirements. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Mark Johnson, Peter Anderson, Mark Dras, and Mark Steedman. Predicting accuracy on large datasets from smaller pilot data. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2018.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling Laws for Neural Language Models. *arXiv preprint arXiv:2001.08361*, 2020.

Daniel S Kermany, Kang Zhang, and Michael Goldbaum. Labeled Optical Coherence Tomography (OCT) and Chest X-Ray Images for Classification. 2018. URL https://doi.org/10.17632/rscbjbr9sj.2.

Pamela J LaMontagne, Tammie LS Benzinger, John C Morris, Sarah Keefe, Russ Hornbeck, Chengjie Xiong, Elizabeth Grant, Jason Hassenstab, Krista

Moulder, Andrei G Vlassenko, et al. OASIS-3: Longitudinal Neuroimaging, Clinical, and Cognitive Dataset for Normal Aging and Alzheimer Disease. *MedRxiv*, pages 2019–12, 2019. URL https://doi.org/10.1101/2019.12.13.19014902.

Rafid Mahmood, James Lucas, David Acuna, Daiqing Li, Jonah Philion, Jose M. Alvarez, Zhiding Yu, Sanja Fidler, and Marc T. Law. How Much More Data Do I Need? Estimating Requirements for Downstream Tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022a.

Rafid Mahmood, James Lucas, Jose M Alvarez, Sanja Fidler, and Marc Law. Optimizing Data Collection for Machine Learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022b.

Kevin S. Murphy. *Probabilistic Machine Learning: An Introduction*, chapter 4.7.2 Gaussian approximation of the sampling distribution of the MLE. MIT Press, 2022.

John Muschelli. Recommendations for processing head CT data. *Frontiers in Neuroinformatics*, 13: 61, 2019.

Carl Edward Rasmussen, Christopher KI Williams, et al. *Gaussian Processes for Machine Learning*, chapter 2.2 Function-space View. Springer, 2006.

Richard D Riley, Joie Ensor, Kym IE Snell, Frank E Harrell, Glen P Martin, Johannes B Reitsma, Karel GM Moons, Gary Collins, and Maarten Van Smeden. Calculating the sample size required for developing a clinical prediction model. *BMJ*, 368, 2020. URL https://doi.org/10.1136/bmj.m441.

Jonathan S Rosenfeld, Amir Rosenfeld, Yonatan Belinkov, and Nir Shavit. A Constructive Prediction of the Generalization Error Across Scales. In *International Conference on Learning Representations (ICLR)*, 2020.

Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting Unreasonable Effectiveness of Data in Deep Learning Era. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Michael Wojnowicz, Martin D Buck, and Michael C Hughes. Approximate inference by broadening the support of the likelihood. In *Fifth Symposium on Advances in Approximate Bayesian Inference*, 2023. URL https://openreview.net/forum?id=wSZrMV2akW.

Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling Vision Transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Appendix Contents

Appendix A. GP Model Details

A.1. Plausible Upper and Lower Bounds

Given plausible lower and upper bounds from domain experts, we form a uniform prior over ε. We can elicit a plausible upper bound for ε as $1 - y'$, where y' is the maximum accuracy observed in the pilot set. We can elicit a plausible lower bound by talking with task experts, which we define as ε_{min}. Based off plausible upper bounds from domain experts, we use $\varepsilon_{\text{min}} = 0.05$ for 3D datasets of head CT scans. For all other datasets we use $\varepsilon_{\text{min}} = 0.0$.

A.2. Chosen Priors

However, the variation we model for $y(x_*)$ should be *at most* the width W of interval $[y', 1.0]$, where y' is the largest accuracy observed in pilot set. We typically expect less deviation than W: we suggest that the marginal of $y(x_*)$, whose variance is approximately $s^2 = \tau^2 + \sigma^2$ if $x_* \gg x_R$, should have a 3-standard-deviation window w whose 20-80 percentile range is between $\frac{W}{2}$ and $\frac{3W}{4}$. We therefore seek a prior on σ such that if we draw many samples of σ as well as many samples of τ from its prior above, the implied window has the desired properties:

$$\textsc{Prctile}(w, 20) \approx \tfrac{W}{2}, \textsc{Prctile}(w, 80) \approx \tfrac{3W}{4}. \quad (11)$$
$$\text{where } w \leftarrow 6s, s \leftarrow \sqrt{\tau^2 + \sigma^2}, \tau \sim p(\tau), \sigma \sim p(\sigma)$$

We operationalize the prior as a truncated normal $p(\sigma) = \mathcal{N}_{[0,\infty)}(\mu_\sigma, \nu_\sigma)$, and do a numerical grid search for hyperparameters μ_σ and ν_σ such that our desired 20-80 range is satisfied. Figure A.1 shows the implied distribution on w (the two-sided 3-std.-dev. window for y_*) given our chosen prior $p(\sigma)$.

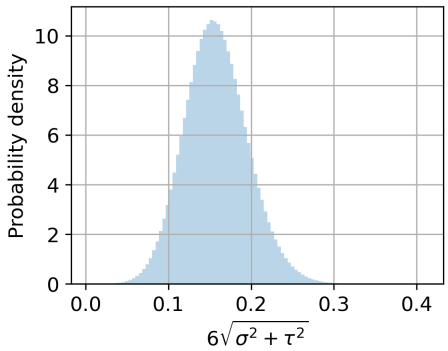

Figure A.1: PDF for the margin of 95% of the posterior distribution based off the prior on τ, a clinically informed maximum performance of 0.95, a test performance of 0.7 for the small initial dataset, and the 20$^{\text{th}}$ and 80$^{\text{th}}$ percentile of the margin of the majority of the posterior density to be a fourth and a half of the maximum performance minus the performance of the small initial dataset. The 20$^{\text{th}}$ and 80$^{\text{th}}$ percentile of the distribution are 0.0625 and 0.125, as desired. The histogram was generated with 10 million samples from the prior on τ and σ.

Kernel length-scale λ. Scalar $\lambda > 0$ controls the rate at which the correlation between $f(x)$ and $f(x')$ decreases as the distance between $\log(x)$ and $\log(x')$ increases. Setting $x' = rx$ for $r \geq 1$, the distance becomes $\log(x') - \log(x) = \log(r)$. A reasonable *a priori* belief is that strong correlations may only exist when r is less than 1.5; at larger distances we should not expect strong correlation as the dataset would have nearly doubled in size. We thus seek a prior $p(\lambda)$ whose 10th percentile is around $\lambda = 0.13$ (implying a low covariance $k(r) = 0.01\sigma^2$) and 90th percentile around $\lambda = 2.86$ (implying a high covariance of $k(r) = 0.99\sigma^2$). These desired λ values were obtained by solving for $\lambda > 0$ in the relation

$$k(r) = \sigma^2 \exp\Big(-\frac{1}{2\lambda^2}\log(r)^2\Big), \quad (12)$$

with $r = 1.5$. Concretely, we chose a truncated normal prior $p(\lambda) = \mathcal{N}_{[0,\infty)}(-1.23, 2.14^2)$, which gives roughly the desired behavior.

Saturation limit ε. Scalar $\varepsilon \geq 0$ represents one minus the maximum accuracy we expect as dataset size gets asymptotically large, using domain knowledge. We can elicit a plausible upper bound for ε as $1 - y'$, where y' is the maximum accuracy observed in the pilot set. We can elicit a plausible lower bound by talking with task experts, which we define as ε_{min}. For example, we may not expect to beat accuracy of 0.95 on some task due to inherent interrater reliability issues. Given these two bounds, we form a uniform prior over ε.

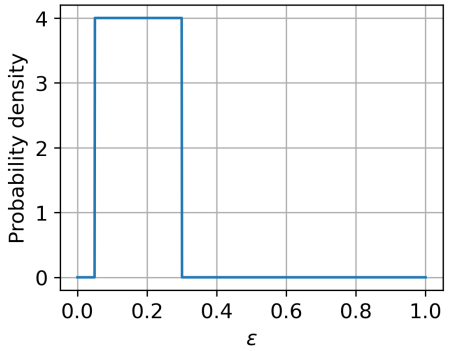

Figure A.2: Example PDF for ε prior with $\varepsilon_{\text{min}} = 0.05$ and $y' = 0.7$.

A.3. Conditional Gaussian Math

Given a marginal Gaussian distribution for $\mathbf{f}$ and a conditional Gaussian distribution for $\mathbf{y}$ given $\mathbf{f}$

$$\begin{bmatrix}\mathbf{f}\\ \mathbf{f}_*\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\mathbf{m}\\ \mathbf{m}_*\end{bmatrix}, \begin{bmatrix}\mathbf{K} & \mathbf{K}_*\\ \mathbf{K}_*^T & \mathbf{K}_{**}\end{bmatrix}\right)$$
$$\begin{bmatrix}\mathbf{y}\\ \mathbf{y}_*\end{bmatrix} \mid \begin{bmatrix}\mathbf{f}\\ \mathbf{f}_*\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\mathbf{f}\\ \mathbf{f}_*\end{bmatrix}, \begin{bmatrix}\tau^2 I_{\mathcal{R}} & \mathbf{0}\\ \mathbf{0} & \tau^2 I_{\mathcal{Q}}\end{bmatrix}\right)$$

the marginal distribution of $\mathbf{y}$ is given by

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{y}_* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{m} \\ \mathbf{m}_* \end{bmatrix}, \begin{bmatrix} \mathbf{K} + \tau^2 I_{\mathcal{R}} & \mathbf{K}_* \\ \mathbf{K}_*^T & \mathbf{K}_{**} + \tau^2 I_{\mathcal{Q}} \end{bmatrix} \right).$$

Appendix B. Dataset Details

ChestX-ray14 is an open access dataset comprised of 112,120 de-identified frontal view X-ray images of 30,805 unique patients with fourteen text-mined disease image labels. For preprocessing we resize the images to 224×224 pixels, rescale pixel values to $[0.0, 1.0]$, and normalize pixel values with the source mean and standard deviation.

Chest X-Ray is an open access dataset comprised of 5,856 de-identified pediatric chest X-Ray images. For preprocessing we center-crop the images using a window size equal to the length of the shorter edge, resize them to 224×224 pixels, rescale pixel values to $[0.0, 1.0]$, and normalize pixel values with the source mean and standard deviation. We evaluate our Gaussian process' extrapolation performance for the classification of bacterial and viral pneumonia.

BUSI is an open access dataset comprised of 780 de-identified breast ultrasound images from 600 female patients with an average image size of 500×500 pixels. For preprocessing we center-crop the images using a window size equal to the length of the shorter edge, resize them to 224×224 pixels, rescale pixel values to $[0.0, 1.0]$, and normalize pixel values with the source mean and standard deviation. The images are categorized into three classes, which are normal, benign, and malignant.

OASIS-3 is a open access project aimed at making neuroimaging datasets freely available to the scientific community. The dataset includes 895 de-identified CT scans from 610 patients where the patient has a diagnosis at least 80 days before or up to 365 days after the CT scan was taken. We use these diagnoses for binary classification. Each 3D scan contains a variable number (74-148) of 512×512 transverse slices. The images are provided in Hounsfield Units (HU). For preprocessing we skull strip images (only including -100 to 300 HU) (Muschelli, 2019), resize each 3D scan to $N \times 224 \times 224$ voxels (where N is the number of transverse slices), rescale pixel values to $[0.0, 1.0]$, and normalize pixel values with the source mean and standard deviation. We do not correct gantry-tilt since each image's degree of gantry-tilt is not include in the header.

TMED-2 is a clinically-motivated benchmark dataset for computer vision and machine learning from limited labeled data. The dataset includes 24964 de-identified echocardiogram images with view labels from 1280 patients. We use these view labels for binary classification. For preprocessing we resize the images to 224×224 pixels, rescale pixel values to $[0.0, 1.0]$, and normalize pixel values with the source mean and standard deviation.

Pilot neuroimaging dataset is a sample of de-identified CT scans from 600 patients. 500 scans were randomly sampled from the cohort of patients 50+ years of age who reveived MRI in 2009-2019 and 100 were randomly sampled from the cohort members who had covert brain infarction (CBI) and/or white matter disease (WMD). This yielded a total sample that included 142 CBI cases and 156 WMD cases. The dataset includes scans from multiple planes for each patient in the Digital Imaging and Communications in Medicine (DICOM) CT format. To simplify the input of our model, we use the largest scan from the axial plane for each patient. Each 3D scan contains a variable number (23-373) of 512×512 transverse slices. For preprocessing we correct gantry-tilt, convert images into HU using each image's rescale slope and intercept, skull strip images, resize each 3D scan to $N \times 224 \times 224$ voxels (where N is the number of transverse slices), rescale pixel values to $[0.0, 1.0]$, and normalize pixel values with the source mean and standard deviation.

Appendix C. Classifier Details

2D datasets. We tune hyperparameters including weight initialization seed, weight decay, and number of epochs to maximize validation AUROC. We select the weight initialization seed from 5 different seeds and weight decay from 11 logarithmically spaced values between $1e^5$ to $1e^{-5}$.

3D datasets. We tune hyperparameters including learning rate, weight initialization seed, weight decay, and number of epochs to maximize validation AUROC. We select the learning rate from 0.05 and 0.01, weight initialization seed from 5 different seeds, and weight decay from 6 logarithmically spaced values between $1e^0$ to $1e^{-5}$, as well as without weight decay.

Appendix D. Results: RMSE

Table D.1: AUROC RMSE for short-range extrapolations.

Dataset	Label	Power law	GP pow (ours)	Arctan	GP arc (ours)
ChestX-ray14	Atelectasis	0.344	0.329	0.397	0.396
	Effusion	0.673	0.859	0.671	0.811
	Infiltration	0.320	0.433	2.264	2.192
Chest X-Ray	Bacterial	0.212	0.225	0.164	0.169
	Viral	1.012	1.046	1.095	1.095
BUSI	Normal	0.705	0.705	0.705	0.705
	Benign	1.539	1.544	1.543	1.547
	Malignant	1.003	1.003	1.003	1.003
TMED-2	PLAX	0.124	0.126	0.261	0.261
	PSAX	0.447	0.447	0.450	0.451
	A4C	0.408	0.408	0.721	0.721
	A2C	2.177	2.174	0.081	0.082
OASIS-3	Alzheimer's	1.561	1.563	1.039	1.046
Pilot neuro-	WMD	1.457	1.427	1.472	1.442
imaging dataset	CBI	1.309	1.338	1.303	1.323

Table D.2: AUROC RMSE for long-range extrapolations.

Dataset	Label	Power law	GP pow (ours)	Arctan	GP arc (ours)
ChestX-ray14	Atelectasis	1.855	1.722	2.949	2.947
	Effusion	3.971	3.506	3.966	3.513
	Infiltration	1.616	2.079	15.350	13.470
Chest X-Ray	Bacterial	1.067	1.089	0.208	0.210
	Viral	3.484	3.540	2.001	1.995
TMED-2	PLAX	0.401	0.403	1.312	1.312
	PSAX	0.450	0.450	0.679	0.679
	A4C	0.616	0.616	1.863	1.863
	A2C	1.974	1.971	2.285	2.286

Appendix E. Results: Curves for Short-Range

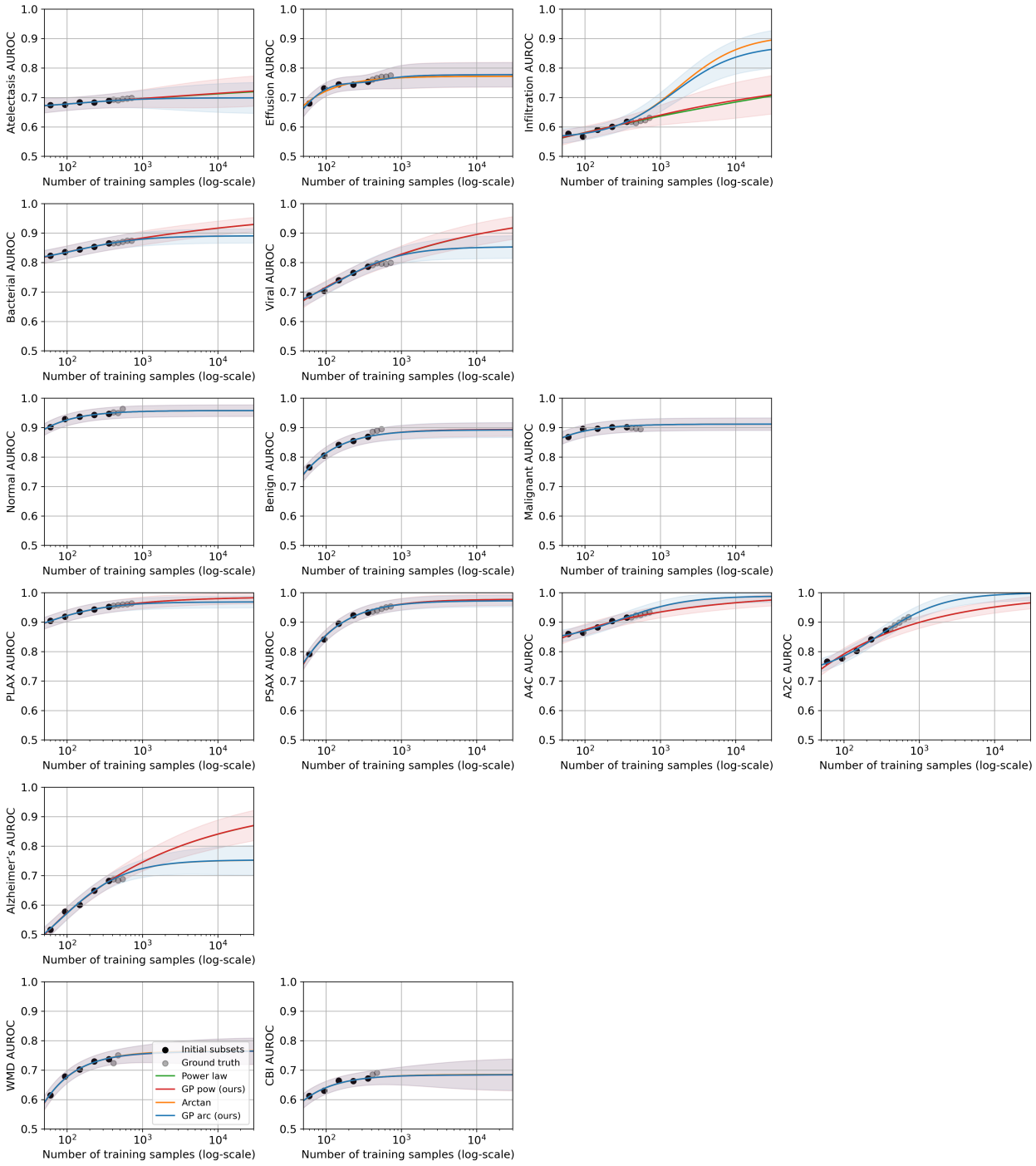

Figure E.1: Short-range extrapolations results from ChestX-ray14, Chest X-Ray, BUSI, TMED-2, OASIS-3, and Pilot neuroimaging dataset (top to bottom).

Appendix F. Results: Curves for Long-Range

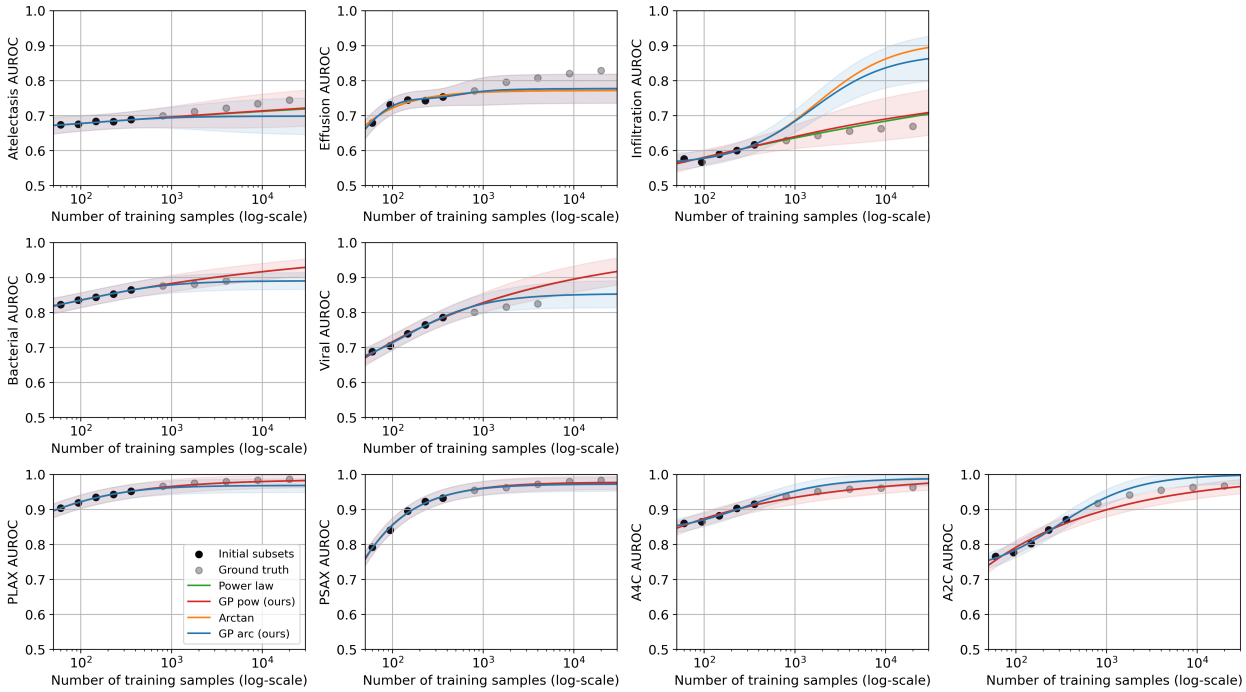

Figure F.1: Long-range extrapolations results from ChestX-ray14, Chest X-Ray, and TMED-2 (top to bottom).

Appendix G. Results: Dense Coverage

Table G.1: Coverage rates for long-range extrapolations, using a target interval of 80% from our GP model and 100 replicates. Standard deviations generated from 500 bootstrapping rounds to select data-split seeds to average across.

		5k training samples		10k training samples		20k training samples		
Dataset	**Label**	**GP pow**	**GP arc**	**GP pow**	**GP arc**	**GP pow w/o priors**	**GP pow**	**GP arc**
ChestX-ray14	Atelectasis	$100.0 \pm 0.0\%$	$97.9 \pm 1.2\%$	$100.0 \pm 0.1\%$	$77.7 \pm 2.7\%$	$12.1 \pm 2.4\%$	$100.0 \pm 0.2\%$	$29.6 \pm 2.9\%$
	Effusion	$67.5 \pm 2.7\%$	$61.9 \pm 3.0\%$	$9.0 \pm 2.1\%$	$6.9 \pm 1.9\%$	$0.0 \pm 0.0\%$	$0.1 \pm 0.3\%$	$0.1 \pm 0.2\%$
	Infiltration	$99.4 \pm 0.8\%$	$0.0 \pm 0.0\%$	$99.1 \pm 0.9\%$	$0.0 \pm 0.0\%$	$1.0 \pm 1.0\%$	$97.8 \pm 1.3\%$	$0.0 \pm 0.0\%$
TMED-2	PLAX	$100.0 \pm 0.1\%$	$3.9 \pm 1.6\%$	$100.0 \pm 0.0\%$	$0.1 \pm 0.2\%$	$100.0 \pm 0.0\%$	$100.0 \pm 0.0\%$	$0.0 \pm 0.0\%$
	PSAX	$100.0 \pm 0.0\%$	$100.0 \pm 0.2\%$	$100.0 \pm 0.0\%$	$99.2 \pm 0.8\%$	$100.0 \pm 0.2\%$	$100.0 \pm 0.2\%$	$89.5 \pm 2.4\%$
	A4C	$99.9 \pm 0.2\%$	$0.3 \pm 0.6\%$	$99.4 \pm 0.7\%$	$0.1 \pm 0.2\%$	$94.0 \pm 1.8\%$	$94.2 \pm 1.8\%$	$0.0 \pm 0.2\%$
	A2C	$62.2 \pm 2.8\%$	$0.0 \pm 0.0\%$	$96.6 \pm 1.6\%$	$0.0 \pm 0.0\%$	$100.0 \pm 0.2\%$	$100.0 \pm 0.2\%$	$0.0 \pm 0.0\%$

Appendix H. Results: Single Point Coverage

Table H.1: Single point coverage evaluations for 95% confidence interval for short-range and long-range extrapolations.

		Short-range extrapolation		Long-range extrapolation	
Dataset	**Label**	**GP pow**	**GP arc**	**GP pow**	**GP arc**
ChestX-ray14	Atelectasis	100.0%	100.0%	100.0%	100.0%
	Effusion	100.0%	100.0%	60.0%	60.0%
	Infiltration	100.0%	100.0%	100.0%	0.0%
Chest X-Ray	Bacterial	100.0%	100.0%	100.0%	100.0%
	Viral	100.0%	100.0%	33.3%	100.0%
BUSI	Normal	100.0%	100.0%	—	—
	Benign	100.0%	100.0%	—	—
	Malignant	100.0%	100.0%	—	—
TMED-2	PLAX	100.0%	100.0%	100.0%	100.0%
	PSAX	100.0%	100.0%	100.0%	100.0%
	A4C	100.0%	100.0%	100.0%	40.0%
	A2C	40.0%	100.0%	60.0%	40.0%
OASIS-3	Alzheimer's	100.0%	100.0%	—	—
Pilot neuro-	WMD	100.0%	100.0%	—	—
imaging dataset	CBI	100.0%	100.0%	—	—