Feature Collapsing for Gaussian process variable ranking

Isaac Sebenius
University of Cambridge

Topi Paananen
Aalto University

Aki Vehtari
Aalto University

Abstract

At present, there is no consensus on the most effective way to establish feature relevance for Gaussian process models. The most common heuristic, Automatic Relevance Determination, has several downsides; many alternate methods incur unacceptable computational costs. Existing methods based on sensitivity analysis of the posterior predictive distribution are promising, but are biased and show room for improvement. This paper proposes Feature Collapsing as a novel method for performing GP feature relevance determination in an effective, consistent, unbiased, and computationally-inexpensive manner compared to existing algorithms.

1 INTRODUCTION

Recent years have seen major advances in the predictive capability of machine learning algorithms, but efforts to increase model interpretability have lagged behind (Lipton, 2018). Gaussian processes (GP) are no exception to this trend. GPs have seen widespread, successful application in domains ranging from climate prediction to biomedical data, and as a probabilistic framework, can offer invaluable confidence estimates of their own predictions (Salter and Williamson, 2016; Clifton et al., 2013). Yet, it remains unclear how to reliably answer the simple question: *Which input features are most influential to a GP's prediction?*

In real-world applications, identifying feature relevance is essential for developing useful models. For example, a patient deemed at critical risk for heart disease must know the most important factors in this assessment. The most common method for determining GP feature importance is Automatic Relevance Determination (ARD), a heuristic of inferring feature relevance directly from the optimized kernel hyperparameters. ARD is convenient in that it requires no additional computation after model training (Paananen et al., 2019), though the method suffers from multiple drawbacks as discussed in section 2.1.

Several methods have been developed to more effectively rank GP features, but often incur immense computational costs (Paananen et al., 2019). For example, Savitsky et al. (2011) proposed a feature-selection method for Gaussian processes using a sparsity-inducing spike-and-slab prior. Under this framework, variable relevance can be determined by the estimated posterior probability of inclusion of each feature following MCMC sampling. While effective, the sampling involved in this method is notoriously slow (Park et al., 2020). Similarly, Piironen and Vehtari (2016) proposed a feature-selection method that finds the features included in the optimal "submodel projection," using a subset of variables, that most closely mimics the behavior of the full model. Yet, with a complexity of $\mathcal{O}\left(p^2 n^3\right)$, where p is the number of features and n the number of observations, the computational cost of this method is high for nearly all real-world applications (Paananen et al., 2019).

More recently, promising work using sensitivity analysis of a GP's posterior predictive distribution has sought to combine the computational feasibility of ARD with the efficacy of methods such as submodel projection. Paananen et al. (2019) proposed two such methods, the 'KL' and 'VAR' relevance scores. Yet, these two methods show room for improvement; for example, like ARD, these feature selection methods are biased towards features with nonlinear effects (Paananen et al., 2019).

In this paper, we propose a novel method for determining GP feature relevance, henceforth referred to as the Feature Collapsing (FC) method. The FC method builds on the KL relevance method proposed by Paananen et al. (2019). Extending a previously-described toy synthetic dataset, we demonstrate that, unlike ARD and the methods proposed by Paananen et al. (2019), the FC method is almost completely unbiased

Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS) 2022, Valencia, Spain. PMLR: Volume 151.

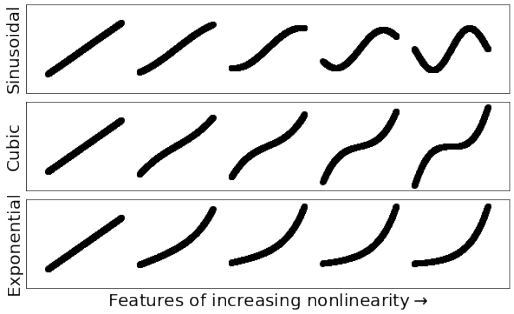


Figure 1: Visualization of the transformation between linear and nonlinear latent functions used in toy datasets. Each latent function is equally relevant to the target variable with respect to both mean and variance, while displaying increasing nonlinearity corresponding to sinusoidal, cubic, or exponential patterns.

towards features with nonlinear behavior. We then discuss the relationship between pointwise FC relevance scores, KL relevance scores, and the latent mean of a studied variable. Finally, we rigorously compare the performance of FC, KL, and ARD rankings on eight widely-studied real-world datasets. We observe that the FC method ranks input features as or more effectively than the ARD and KL algorithms across all eight datasets and generally demonstrates higher consistency in its relevance rankings. These results suggests that Feature Collapsing presents a straightforward and promising new method for performing feature selection using Gaussian processes.

Details on code availability and implementation can be found in the Supplementary Materials.

2 THEORETICAL BACKGROUND

2.1 Automatic Relevance Determination

One of the most popular covariance functions is the Exponentiated Quadratic (EQ) kernel. When performing Automatic Relevance Determination (ARD), a modified version of the EQ kernel is used wherein a different length-scale hyperparameter ℓ_i is optimized for each input dimension. The resulting ARD-EQ kernel takes the following form:

$$k_{\text{ARD}-\text{EQ}}(\mathbf{x}, \mathbf{x}') = \sigma^2 \exp\left(-\sum_{i=1}^{p} \frac{(x_i - x_i')^2}{2\ell_i^2}\right).$$

After the GP is trained, the ARD relevance of each input feature i is defined as $\frac{1}{\ell_i}$. Intuitively, this definition is built on the assumption that if a dimension has a small value of ℓ_i, small changes in the feature would lead to relatively large responses in the target.

While convenient, ARD has several drawbacks. For instance, Piironen and Vehtari (2016) demonstrate that ARD is heavily biased towards nonlinear effects; in other words, ARD will not rank two equally-relevant features as such if one exhibits more nonlinear behavior than the other. Moreover, ARD does not naturally extend to GPs where alternative kernels are needed (e.g. linear, composite, periodic, etc.), which limits its general applicability. However, ARD remains the most commonly used method for ranking feature relevance using Gaussian processes.

2.2 Sensitivity Analysis of the Posterior Predictive

Paananen et al. (2019) propose a novel approach to feature ranking for Gaussian Processes using sensitivity analysis of the posterior predictive distribution. In essence, the approach seeks to determine which features have the largest impact on the predictive distribution when evaluated at points near the training observations.

This approach has several advantages. Unlike ARD, ranking methods that use sensitivity analysis are applicable to Gaussian processes with any kernel function – not just the ARD-EQ kernel described above. Moreover, they can provide *pointwise* estimates of feature relevance in addition to aggregate measures, which offer a sense of how important each feature is within each local region of its domain (Paananen et al., 2019). Finally, in contrast to methods such as spike-and-slab priors (Savitsky et al., 2011) and submodel projection (Piironen and Vehtari, 2016), sensitivity analysis is computationally cheap.

Paananen et al. (2019) propose two variants – the VAR and the KL methods. The complexities of the two methods are $\mathcal{O}(pn^2)$ for VAR and $\mathcal{O}(pn^3)$ for KL, which are reasonable costs given the $\mathcal{O}(n^3)$ complexity of GP inference (Paananen et al., 2019).[1] The VAR method appears to offer no appreciable gain in performance over KL, and is much less intuitively defined. Thus in this paper, we focus primarily on the proposed KL relevance measure, only including the VAR score to recreate existing results on the toy dataset (Section 3).

The intuition behind the KL relevance score is as follows: after a GP is trained, the predictive distribution at each training point is determined. Then, for each training observation, a small perturbation is applied to

[1] The use of approximation methods to reduce the complexity of GP inference would also reduce the cost of feature ranking methods that rely on sensitivity analysis.

each dimension separately, and the distance between the new predictive distribution and the unperturbed version is calculated. A high KL relevance indicates that a small perturbation in the observation at dimension j leads to a large change in the predictive distribution at that point.

More formally, Paananen et al. (2019) define a distance measure as follows based on Kullback-Leibler (KL) divergence, taking the square root to more easily detect and linearly approximate minuscule changes:

$$d(p\|q) = \sqrt{2\mathcal{D}_{\mathrm{KL}}(p\|q)}.$$

Then, for a given point $\mathbf{x}^{(i)}$, the KL relevance of dimension j is calculated as

$$r(i, j, \Delta) = \frac{d\left(p\left(y_* \mid \mathbf{x}^{(i)}, \mathbf{y}\right) \| p\left(y_* \mid \mathbf{x}^{(i)} + \Delta_j, \mathbf{y}\right)\right)}{\Delta},$$

where Δ_j is a vector of zeros except at dimension j, where it takes the small value Δ. Paananen et al. (2019) use $\Delta = 0.0001$, but also show that the KL is relatively insensitive to changes in the magnitude of Δ.

Finally, the total KL relevance of dimension j is defined as the mean of all pointwise estimates.

2.3 Feature Collapsing

The Feature Collapsing (FC) method proposed in this paper uses an adapted perturbation scheme to extend the KL relevance method defined above. Rather than applying a perturbation of a fixed Δ to all inputs at dimension j, the FC method instead sets all training points to a constant value (zero) at j. By thus 'collapsing' all training observations to the same value at j, this process leads to varying levels of perturbation for different training observations. Following this collapsing process, a similar process of sensitivity analysis is used to rank feature importance. The properties and advantages of this collapsing scheme are investigated in later sections.

Formally, similar to Paananen et al. (2019), we define a distance measure as

$$d(p\|q) = \sqrt{\mathcal{D}(p\|q)}.$$

We then define the relevance of feature j at observation $\mathbf{x}^{(i)}$ to be the distance between the predictive distribution before and after feature collapsing has been applied:

$$r(i, j, \delta) = d\left(p\left(y_* \mid \mathbf{x}^{(i)}, \mathbf{y}\right) \| p\left(y_* \mid \mathbf{x}^{(i)}(\mathbf{1} - \delta[j]), \mathbf{y}\right)\right).$$

In the above, $(\mathbf{1} - \delta[j])$ is a vector of ones except at position j, where it takes the value of zero. Finally, we define the FC relevance of feature j to be the average over all training observations:

$$\mathrm{FC}_j = \frac{1}{n}\sum_{i=1}^{n} r(i, j, \delta).$$

For the distance function $\mathcal{D}(p\|q)$, we use the Bhattacharyya distance between two normal distributions (see, e.g., Nagino and Shozakai, 2006). We also considered using KL divergence as the distance function which led to qualitatively identical results, available in the Supplementary Materials.

Naturally, all input variables must be standardized in order to have comparable FC relevance scores.

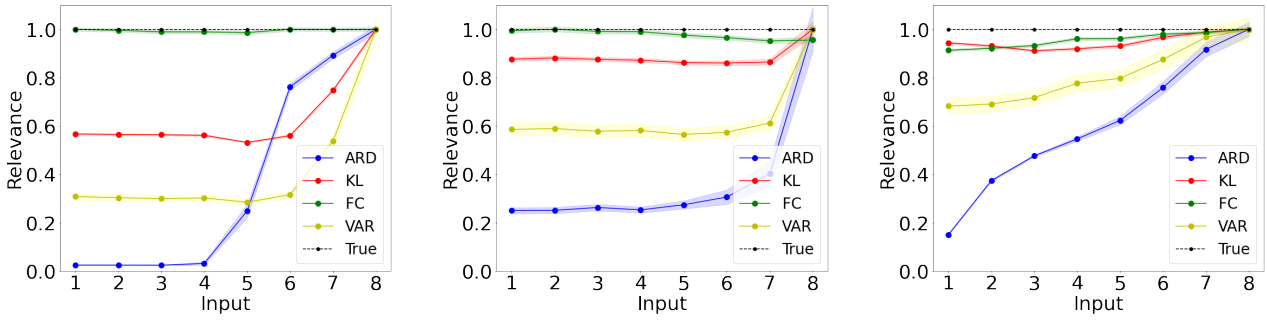
Why Collapse to Zero? The theoretical justification for choosing to collapse features to zero (the mean after standardization) is that this is the constant that allows the perturbed data to most closely mirror the true data by minimizing the expected squared perturbation distance. This is somewhat intuitive, but to see it mathematically, we are trying to find the value of c that minimizes $\mathrm{E}\left[(X - (\mathrm{E}[X] + c))^2\right]$, or more simply when $\mathrm{E}[X] = 0$, $\mathrm{E}\left[(X - c)^2\right] = \mathrm{E}\left[X^2\right] + c^2$. As $\mathrm{E}\left[X^2\right]$, is always positive, it is clear that the minimum occurs when c=0. Thus, collapsing to zero corresponds to the FC perturbation scheme that most closely mirrors the true data.

3 TOY DATASET

Previous work has illustrated the severe nonlinear bias of ARD using an elegantly constructed toy dataset (Piironen and Vehtari, 2016; Paananen et al., 2019). In this dataset, the target variable y is calculated as the sum of the latent functions of p variables, $f_1(x_1) + \cdots + f_p(x_p)$, plus random noise. All latent functions have equal mean and variance, and thus are equally relevant to the target variable in the L2 sense (which is the definition of relevance implicitly used in this paper). However, they each exhibit a varying amount of nonlinearity, as illustrated in Figure 1. A full specification of the dataset construction is offered in the Supplementary Materials as well as in the original publication by Paananen et al. (2019).

In this work, we extended the toy dataset to include increasing nonlinearity according to sinusoidal, cubic, and exponential patterns (see Figure 1) in order to demonstrate how different functions affect nonlinear bias. We included eight features with all inputs $x_1 \ldots x_8 \overset{i.i.d}{\sim} \mathcal{U}(-1, 1)$.

Figure 2 shows the relevance scores for FC, KL, VAR, and ARD for each of the inputs and type of nonlinearity, averaged over 150 runs. The maximum relevance score



(a) Sinusoidal (b) Cubic (c) Exponential

Figure 2: Results on the toy dataset, where the eight input features demonstrate increasing levels of a) sinusoidal, b) cubic, or c) exponential nonlinearity while remaining equally relevant to the target. As shown, ARD, VAR, and KL are heavily biased towards nonlinear variables, with the degree of bias greatly affected by the type of nonlinearity. The FC ranking method shows little to no bias across all three studied function families. Shading represents one standard error of the mean.

was scaled to one.[2] In the plot of increasing sinusoidal nonlinearity shown in Figure 2(a), the KL, VAR, and ARD lines recreate the findings reported by Paananen et al. (2019), who show that the KL and VAR feature selection methods are less biased towards nonlinear effects than ARD. Indeed, in this plot, ARD is clearly most biased towards variables with nonlinear effect; however, both KL and VAR are heavily biased as well, if only to a slightly lesser extent. By contrast, FC shows essentially no bias towards nonlinearity. These results were replicated when the inputs $x_1 \dots x_8$ were distributed $\overset{i.i.d}{\sim} \mathcal{N}(0, 0.3^2)$.

Interestingly, the type of the nonlinear latent function has a large effect on the performance of KL, VAR, and ARD. All three are most biased when the nonlinearity is sinusoidal (Figure 2(a)), followed by cubic (Figure 2(b)) and exponential (Figure 2(c)). An explanation for this behavior is offered in Section 3.1. Importantly, the FC methods report similarly unbiased feature rankings across all studied nonlinear function types.

Piironen and Vehtari (2016) show that their submodel projection approach is unbiased, but as discussed above, this method can be very slow on real datasets. Thus, Feature Collapsing (with the same complexity as the KL method) presents a promising alternative to overcome the nonlinear biases present in prior computationally-feasible GP variable ranking methods.

[2]This toy example, alongside the KL/VAR relevance methods, was implemented using a mildly adapted version of the code developed by Paananen et al. (2019) at `https://github.com/topipa/gp-varsel-kl-var`.

3.1 Relationship to latent function

Why might the Feature Collapsing perturbation scheme mitigate the nonlinear bias present in the conceptually-similar KL method? One potential answer lies in examining the relationship between point-wise relevance estimates for the two methods and the latent mean of a given variable.

Paananen et al. (2019) demonstrate how the KL relevance measure is directly related to the partial derivative of the latent mean of a given variable. This relationship is illustrated in Figure 3, where pointwise KL estimates are clearly related to the absolute value of the derivative of the latent mean for inputs x_1 and x_8 from the toy experiment using sinusoidal nonlinearity. Intuitively, this makes sense; small, fixed perturbations will have a larger effect if the latent mean has a steeper slope in a given local region. This is an interesting property of the KL, but is a source of bias towards nonlinearity. In the toy example above, for example, one explanation for the bias in the KL method shown in Figure 2(a) is that the more nonlinear functions have derivatives with much larger absolute value on average. This also explains why different latent functions give rise to different amounts of bias. For example, the cubic and exponential functions used in Figures 2(b) and 2(c) are monotonic and the average absolute derivative of the latent means do not vary as much, leading to less-biased KL ranking estimates.

By contrast, pointwise FC estimates do not estimate the derivative of a variable's latent mean, but are rather related to its absolute value as shown in Figure 3. This property results from the altered perturbation scheme, where all points are collapsed to a constant value; the

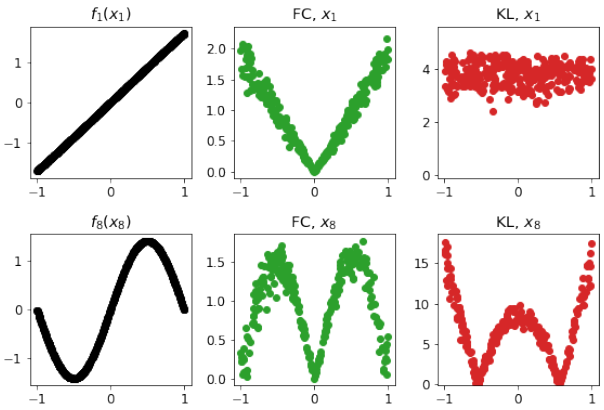


Figure 3: Left column: latent function for all observations at features x_1 and x_8 with sinusoidal nonlinearity. Middle column: pointwise relevance scores for FC. Right column: corresponding pointwise KL relevance scores.

Table 1: Summary of datasets used. p indicates the number of variables.

Dataset	p	n_{total}	n_{train}
Automobile	38	192	172
Banknote	4	1372	400
Boston Housing	13	506	300
Crime	102	1992	700
Concrete	8	1030	700
Liver	5	345	305
Pima	8	768	500
Wine	11	1599	500

latent function can thus be estimated in relation to a common reference point. By estimating the effect of the latent function, FC is able to largely avoid the bias arising from the increased values of the derivative of nonlinear relationships.

4 RESULTS

We next rigorously compared the efficacy and consistency of FC, KL, and ARD feature-ranking methods on eight benchmark datasets from the UCI dataset repository: the Automobile, Banknote Authentication, Boston Housing, Concrete Compressive Strength, Crime, Liver Disorders Pima Indians, and (Red) Wine datasets (Dua and Graff, 2017). Several of these widely-used datasets have been shown to be especially useful in work on GP feature selection, (Piironen and Vehtari, 2016; Savitsky et al., 2011; Paananen et al., 2019), and they include all five original datasets used to establish the efficacy of the KL method by Paananen et al. (2019).

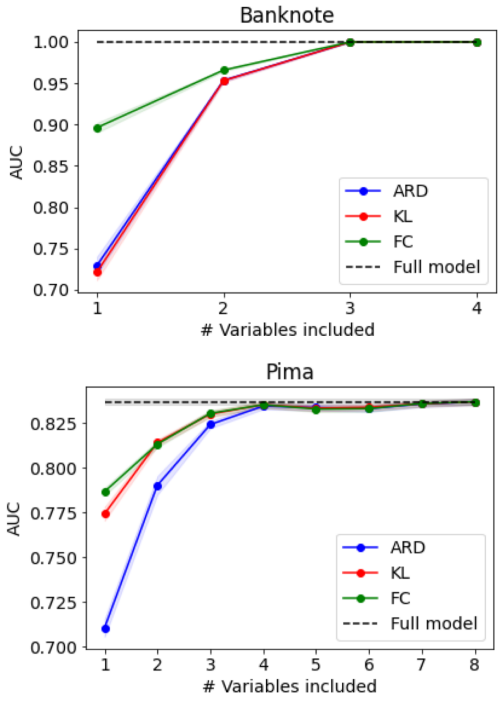


Figure 4: Results on classification datasets: Banknote and Pima Indians. ROC AUC is shown using an increasing number of variables, ranked by each method. Shaded regions are one standard error of the mean.

Details on the exact tasks considered for each as well as preprocessing steps are provided in the Supplementary Materials.

4.1 Performance

Following the experimental setup from Paananen et al. (2019), we investigated how effectively FC, KL, and ARD could rank the features in each dataset, measured by the predictive power of models trained on small subsets of highly-ranked variables from each method. First, we fit a GP to predict the target variables on the complete datasets, including all input features. For all models, we used a sum of an ARD-EQ kernel and a constant kernel (equivalent to a bias term), and we implemented all models using scikit-learn (Buitinck et al., 2013) and GPy (GPy authors, 2012). To prevent overfitting, we set a $\mathcal{U}(0.0001, 15)$ prior on the length-scale hyperparameters. Moreover, all features were standardized before training.

Next, we determined the feature importance rankings using FC, KL, and ARD methods. For each method, we fit new models using an increasing number of variables until all features were used or the results plateaued. We incorporated new features based on descending relevance scores (i.e. the first variable included being

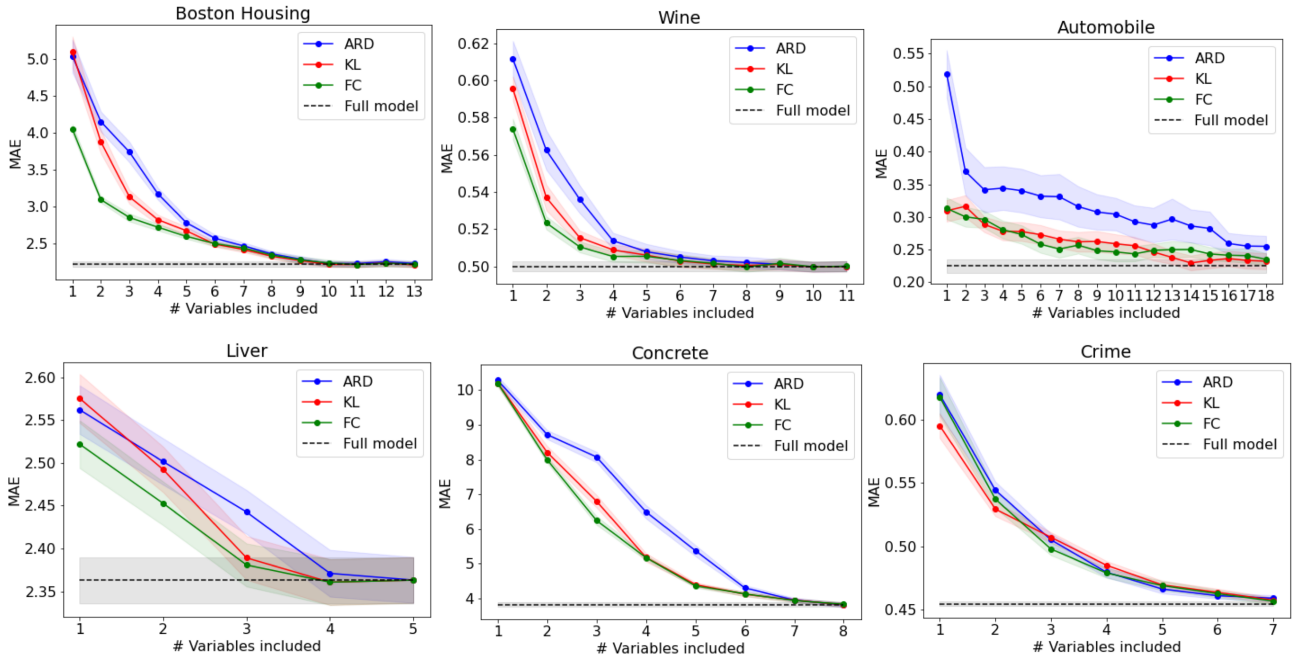


Figure 5: Results on regression datasets: Boston Housing, Red Wine Quality, Automobile, Liver, Concrete, and Crime. Mean absolute error is shown using an increasing number of variables, ranked by each method. Shaded regions are one standard error of the mean.

the most important). We measured the mean absolute error (MAE) between the predicted and true values of an independent test dataset (or ROC AUC for classification tasks) for each model over 20 random splits for each dataset.[3]

The results of the two classification tasks are shown in Figure 4, and those from the six regression tasks are shown in Figure 5; we observe that FC matches or outperforms KL and ARD across all eight studied datasets. For six datasets, FC is able to identify the most relevant inputs notably more effectively than either KL or ARD, indicated by the higher performance in terms of lower MAE or higher AUC at fewer numbers of included variables. On the Automobile dataset, KL and FC show similarly impressive improvements over ARD, and on the Crime dataset, all feature selection methods obtain comparable performance.

Interestingly, the results on the well-studied Boston Housing dataset – in addition to the Banknote and Liver datasets – showed a particularly clear advantage of FC over the KL and ARD methods. As discussed by Savistky et al. (2011), certain features in the Boston Housing dataset exhibit interesting nonlinear behaviors that make it particularly useful for studying feature selection. For example, the fifth variable x_5 represents a neighborhood's levels of nitrogen oxide, a compound emitted by factories and cars. At low levels, x_5 has a positive correlation with median housing price, and a negative correlation at high levels (Savitsky et al., 2011). The lack of bias towards nonlinear effects exhibited by FC methods, therefore, was likely a key factor in the improvement over KL and ARD on this dataset.

4.2 Consistency

In addition to identifying important variables *effectively*, it is vital for a feature-ranking algorithm to determine relevance *consistently* given variation in the training set due to different random data splits. Figure 6 offers a sense of the consistency of each of the feature ranking scores applied to the three studied datasets. The figure displays the mean pairwise cosine distance between the relevance scores for each method across all experimental runs. The FC method achieved the highest consistency in four out of eight studied datasets (Concrete, Boston Housing, Wine, Liver, and Pima), was tied with KL for most consistent in two datasets (Automobile and Concrete), was second most consistent in one (Crime), and had the lower consistency in one (Banknote).

Interestingly, while FC was least consistent on the Banknote dataset, this was also perhaps the dataset where the method's improvement in performance was most marked. Taken as a whole, these results suggest

[3]For the Liver dataset, 100 random splits were used.

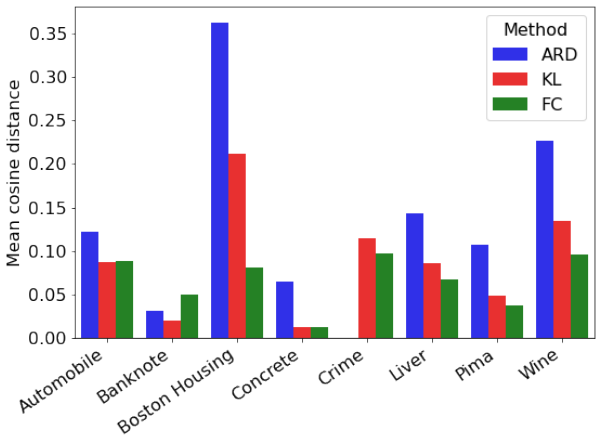


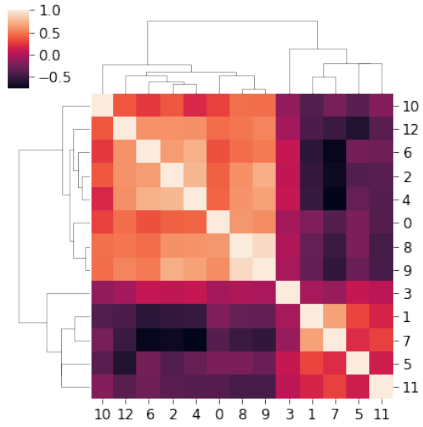
Figure 6: Consistency of each method, measured by the mean cosine distance between relevance vectors obtained on different runs.

that FC is highly consistent relative to the other studied metric, and that any loss of consistency may be due to FC sporadically identifying important patterns that the other methods miss entirely.

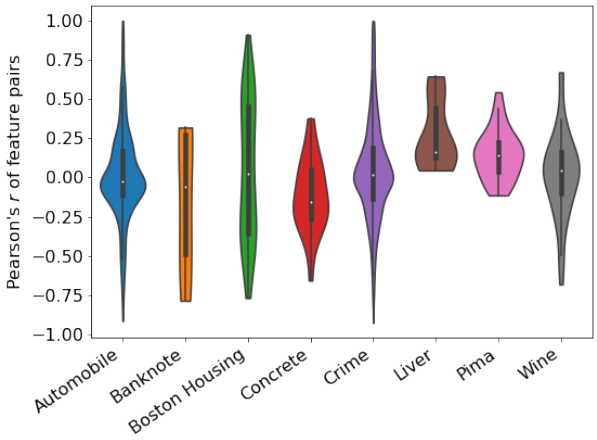
4.3 Correlated features

Despite the advantages of FC ranking demonstrated so far, the method contains one important theoretical risk: in the case where features are very highly correlated, setting one dimension to zero while leaving the others unchanged may introduce unrealistic virtual observations that do not reflect real world data. By contrast, local perturbation methods, such as KL, would largely maintain the correlation structure of the perturbed data since altered features would not stray far beyond what is actually observed.

We visualized the correlation structure of the studied datasets to ensure that the features contained within demonstrated reasonable levels of collinearity. Figure 7(a) shows the particularly rich correlation matrix for the Boston Housing dataset, which contains near-perfectly correlated feature pairs and two clear feature clusters. However, the improved performance of FC on this dataset suggests that possibly using unrealistic virtual observations does not lead to degraded performance. One possible reason for the good performance is that the exponentiated quadratic kernel used in the experiments is a local kernel, meaning that it only captures short-range structure (Bengio et al., 2006; Duvenaud et al., 2011). Because of this, its predictions tend to zero outside the observed data and behave predictably. For this reason, one should be more careful when using the FC method with other kernels or models that model long-range effects.



(a) Feature correlations in the Boston Housing dataset.



(b) Violinplot of off-diagonal entries in feature correlation matrices for all studied datasets.

Figure 7: Visualization of the correlation structure in the studied datasets, showing a) example of rich correlation structure in the Boston Housing dataset and b) large correlations present in the other studied datasets as well.

Figure 7(b) indicates that all other datasets contained varying distributions of pairwise feature correlations, suggesting that the FC method remains robust across a wide range of correlation structures. This observation bodes well for FC's generalizability; still, the potential drawback of correlated features should be kept in mind when FC is applied to new domains.

5 CONCLUSION

In this paper, we introduce the Feature Collapsing method as a novel way to perform feature relevance determination for Gaussian processes using sensitivity analysis of the posterior predictive distribution. Us-

ing a toy dataset, we demonstrate how FC rankings are essentially unbiased to features with nonlinear effects, in contrast to both ARD and the related methods proposed by Paananen et al. (2019), and propose an explanation for this behavior by relating FC relevance scores to features' latent means. On eight real-world datasets, we demonstrate that the FC methods can identify relevant features as or more effectively compared to the KL and ARD methods. Moreover, FC rankings were generally more consistent than other methods. Interestingly – if unsurprisingly – despite being the most commonly-used method, ARD showed the worst results in terms of bias, predictive performance and consistency.

It is important to note that the FC method is unlikely to outperform feature selection methods such as the submodel projection method developed by Piironen and Vehtari (2016). However, such methods can be very slow in real applications. For reference, Piironen and Vehtari (2016) report that the submodel projection method took over four hours to run on the Automobile and Crime datasets; by contrast, the FC method took less than one second to run on consumer hardware for these datasets using the same number of training points.

This paper suggests that the FC method could provide a useful direction to guide future research in simple, effective methods for feature relevance determination in Gaussian processes. The theoretical risk of highly correlated variables should be kept in mind, especially if applying FC to different models. However, at least when using exponentiated quadratic Gaussian processes, the results of this work show a clear gain in performance in multiple real-world datasets. This suggests that FC can be safely used even in the presence of correlated variables.

Furthermore, it would be fruitful to more deeply investigate how best to estimate local feature relevance. One of the advantages of feature relevance using sensitivity analysis of the posterior predictive, noted by Paananen et al. (2019), is that it can offer pointwise relevance estimates, and therefore a sense of the local relevance of a feature at particular regions in its domain. This information could be invaluable – for example, in a clinical setting, it would be hugely valuable to know if a measurement such as weight is globally correlated with higher risk for a certain disease, or only becomes a risk factor beyond a certain threshold. However, the KL and FC pointwise relevance estimates offer very different estimates of local relevance. Thus, future research is needed for real-world conclusions about local feature relevance to become a reliable part of the GP feature relevance determination pipeline.

References

Zachary C. Lipton. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57, June 2018. ISSN 1542-7730. doi: 10.1145/3236386.3241340. URL https://doi.org/10.1145/3236386.3241340.

James M. Salter and Daniel Williamson. A comparison of statistical emulation methodologies for multi-wave calibration of environmental models. *Environmetrics*, 27(8):507–523, 2016. doi: https://doi.org/10.1002/env.2405. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/env.2405.

L. Clifton, D. A. Clifton, M. A. F. Pimentel, P. J. Watkinson, and L. Tarassenko. Gaussian processes for personalized e-health monitoring with wearable sensors. *IEEE Transactions on Biomedical Engineering*, 60(1):193–197, 2013. doi: 10.1109/TBME.2012.2208459.

Topi Paananen, Juho Piironen, Michael Riis Andersen, and Aki Vehtari. Variable selection for Gaussian processes via sensitivity analysis of the posterior predictive distribution. In Kamalika Chaudhuri and Masashi Sugiyama, editors, *Proceedings of Machine Learning Research*, volume 89 of *Proceedings of Machine Learning Research*, pages 1743–1752. PMLR, 16–18 Apr 2019. URL http://proceedings.mlr.press/v89/paananen19a.html.

Terrance D. Savitsky, M. Vannucci, and N. Sha. Variable selection for nonparametric Gaussian process priors: Models and computational strategies. *Statistical science : a review journal of the Institute of Mathematical Statistics*, 26 1:130–149, 2011.

Chiwoo Park, David J. Borth, Nicholas S. Wilson, and Chad N. Hunter. Variable selection for Gaussian process regression through a sparse projection, 2020.

Juho Piironen and Aki Vehtari. Projection predictive model selection for Gaussian processes. *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, Sep 2016. doi: 10.1109/mlsp.2016.7738829. URL http://dx.doi.org/10.1109/MLSP.2016.7738829.

Goshu Nagino and Makoto Shozakai. Distance measure between Gaussian distributions for discriminating speaking styles. In *Ninth International Conference on Spoken Language Processing*, volume 2, 01 2006.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort,

Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.

GPy authors. GPy: A Gaussian process framework in python. http://github.com/SheffieldML/GPy, 2012.

Yoshua Bengio, Olivier Delalleau, and Nicolas Le Roux. The curse of highly variable functions for local kernel machines. *Advances in neural information processing systems*, 18:107, 2006.

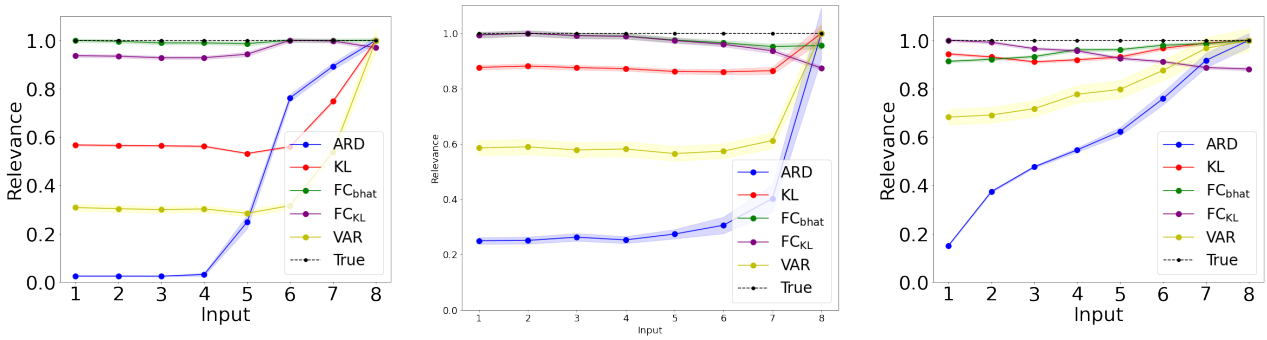
David Duvenaud, Hannes Nickisch, and Carl Edward Rasmussen. Additive Gaussian processes. *arXiv preprint arXiv:1112.4394*, 2011.

Supplementary Materials for Paper "Feature Collapsing for Gaussian process variable ranking"

1 SUPPLEMENTAL RESULTS

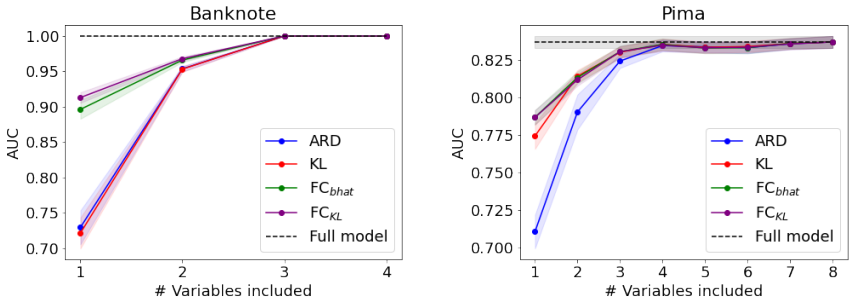
1.1 Results comparing KL divergence as distance metric

In this section, we replicate all results in the paper (on the toy dataset and all eight UCI datasets, as well as the plot of consistency corresponding to Figure 6) using KL divergence in addition to Bhattacharyya distance as the distance metric in the FC algorithm. Nearly all results are qualitatively identical. The two versions of FC are labeled FC_{KL} and FC_{bhat}, respectively. This section uses a GP with the ARD-EQ kernel discussed in the main body.



(a) Sinusoidal (b) Cubic (c) Exponential

Figure 1: Results on the toy dataset, where the eight input features demonstrate increasing levels of a) sinusoidal, b) cubic, or c) exponential nonlinearity while remaining equally relevant to the target.



(a) (b)

Figure 2: Results on classification datasets: Banknote and Pima Indians. ROC AUC is shown using an increasing number of variables, ranked by each method. Shaded regions are one standard error of the mean.

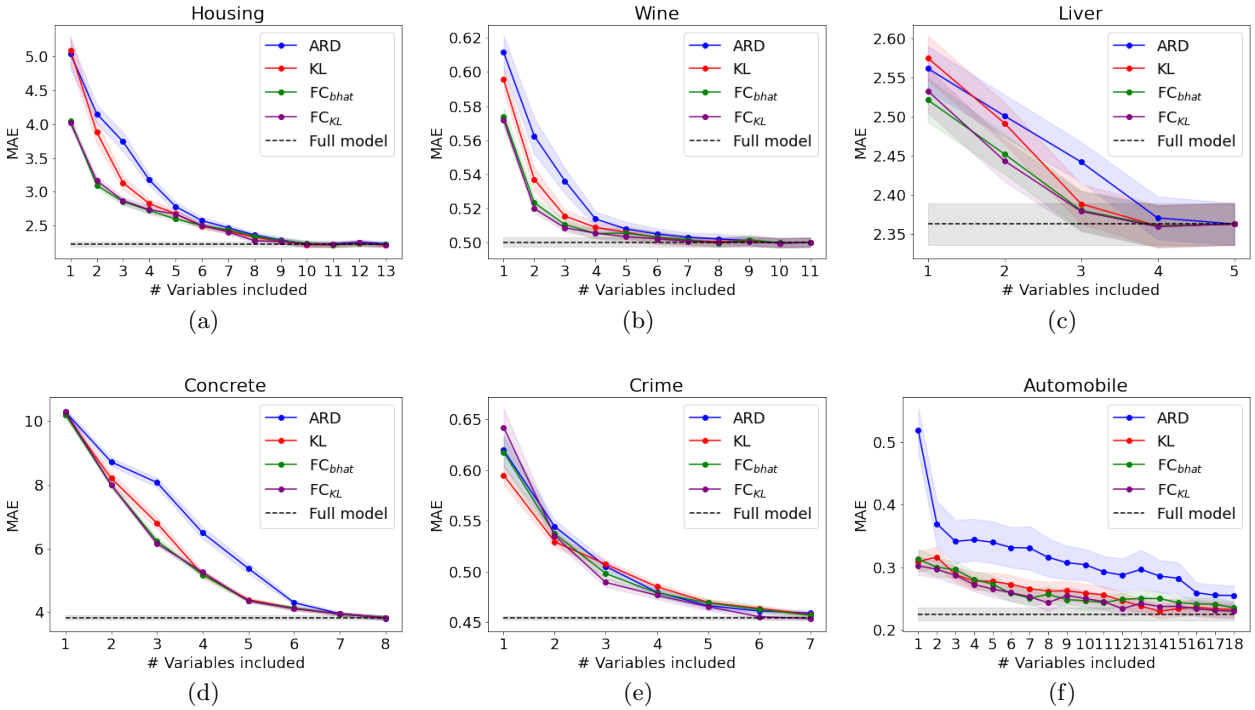


Figure 3: Results on regression datasets: Boston Housing, Red Wine Quality, Liver, Concrete, Crime, and Automobile. Mean absolute error is shown using an increasing number of variables, ranked by each method. Shaded regions are one standard error of the mean.

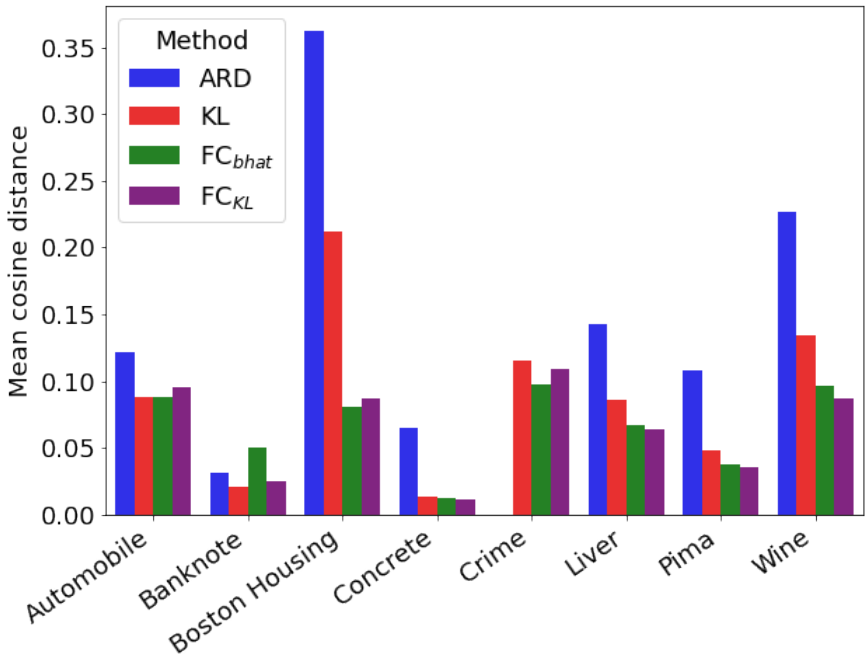


Figure 4: Consistency of each method, measured by the mean cosine distance between relevance vectors obtained on different runs.

1.2 Results using Matérn kernel

The following section reproduces the major results from the paper (results on toy and UCI datasets) when GPs were trained using a ARD-Matérn 5/2 kernel (see https://gpy.readthedocs.io/en/deploy/GPy.kern.src.html for details) instead of the ARD-EQ kernel used in the main body. In the toy dataset, the reduction in bias towards nonlinear features remains clear under this change in kernel. The results on the UCI datasets also demonstrate a highly similar pattern to the results discussed in the main text.

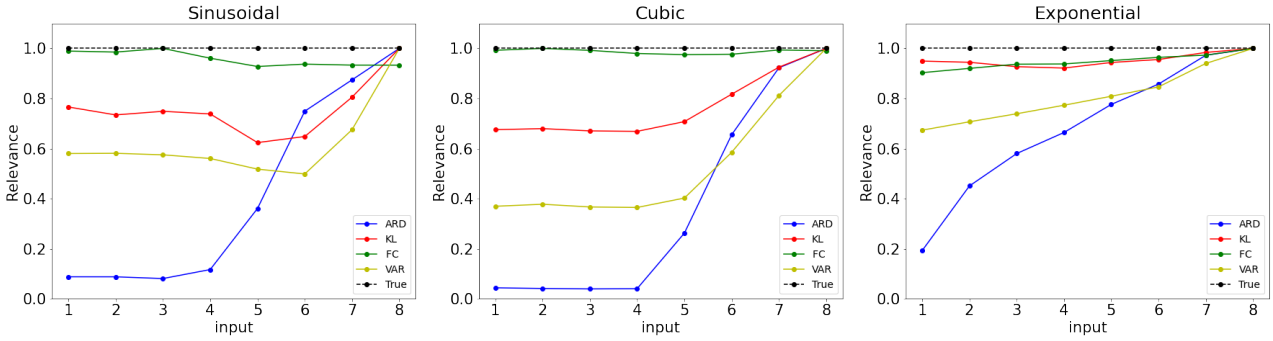
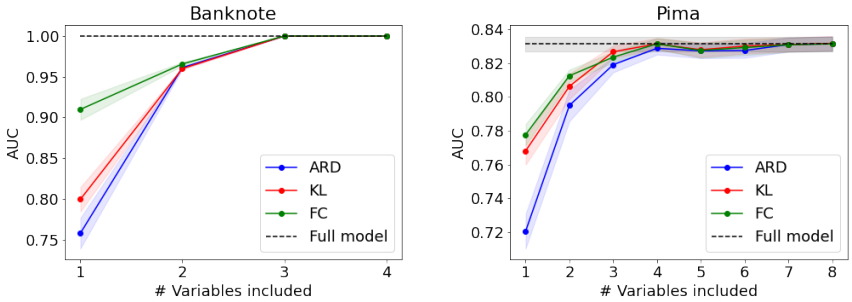


Figure 5: Results on the toy dataset with ARD-Matérn kernel.



(a) (b)

Figure 6: Results using ARD-Matérn kernel on classification datasets.

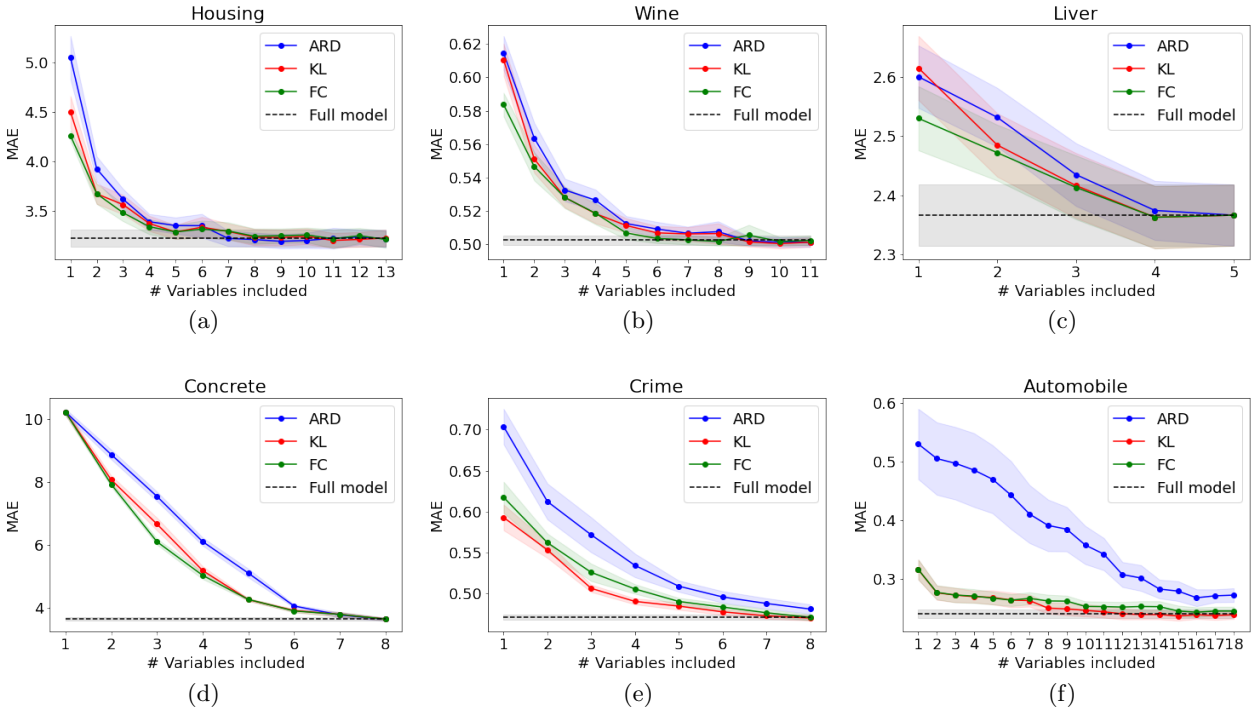


Figure 7: Results on regression datasets with ARD-Matérn kernel.

2 TOY DATASET CONSTRUCTION

We provide the details for how the toy dataset was constructed. Following the notation in Paananen et al. (2019), we set the target variable to be the sum of functions of 8 features $x_1 \dots x_8$ of increasing nonlinearity as follows:

$$y = f_1(x_1) + \ldots + f_8(x_8) + \varepsilon$$
$$\varepsilon \sim \mathcal{N}(0, 0.3^2).$$

Under the condition where the increasing nonlinearity is sinusoidal, we let $f_j(x_j) = A_j \sin(\rho_j x_j)$, where ρ_j are equally spaced between $\pi/10$ and π (Paananen et al., 2019). A_j is a scaling coefficient calculated so that the variance of each $f_j(x_j)$ is one, thereby ensuring equal relevance in the L2 sense for all features. We set all $x_j \sim \mathcal{U}(-1, 1)$. For this sinusoidal case,

$$A_j = \sqrt{4\left(2 - \frac{\sin(2\rho_j)}{\rho_j}\right)^{-1}}$$

In the cubic case, $f_j(x_j) = A_j\left(\alpha_j x_j + (1-\alpha_j)\, x_j^3\right)$, where α_j is evenly spaced between 1 and 0. Here, the scaling coefficient comes out to the following:

$$A_j = \left(\sqrt{\frac{\alpha_j^2}{3} + \frac{(1-\alpha_j)^2}{7} + \frac{2}{5}\alpha_j(1-\alpha_j)}\right)^{-1}$$

Finally, in the exponential example, we let $f_j(x_j) = A_j\left(\alpha_j x_j + (1-\alpha_j)\, e^{2x_j}\right)$. In this example, the analytical scaling coefficient is as follows:

$$A_j = \left(\sqrt{\frac{\alpha_j^2}{3} + (1-\alpha_j)^2\omega + 2\gamma\alpha_j(1-\alpha_j)}\right)^{-1},$$

$$\omega = \frac{e^4 - e^{-4}}{8} - \frac{(e^2 - e^{-2})^2}{16}$$

$$\gamma = \frac{1}{2}\left(\cosh(2) - \frac{1}{2}\sinh(2)\right).$$

For all toy dataset experiments, we used 300 data points.

3 DATASET PREPROCESSING

In this section we provide details about the tasks considered for each of the eight UCI datasets, any preprocessing steps considered, and links to where they can be accessed.

Automobile. Dataset was acquired from the original authors of Paananen et al. (2019); preprocessing details are available in the original publication and include using log-price as the target and one-hot encoding categorical variables. Based on the dataset available at https://archive.ics.uci.edu/ml/datasets/automobile.

Banknote. No pre-processing was applied, and the outcome features was an indicator variable representing forged or real banknotes. Accessible at https://archive.ics.uci.edu/ml/datasets/banknote+authentication.

Boston Housing. No pre-processing was applied, and the outcome feature was median housing value. Accessible at https://github.com/selva86/datasets/blob/master/BostonHousing.csv.

Concrete. No pre-processing was applied, and the outcome feature was concrete compressive strength. Accessible at http://archive.ics.uci.edu/ml/datasets/concrete+compressive+strength.

Crime. Dataset was acquired from the original authors of Paananen et al. (2019); preprocessing details are available in the original publication. Based on the dataset available at http://archive.ics.uci.edu/ml/datasets/communities+and+crime.

Liver Disorders. Features 2-5 were log transformed. The target variable used was the number of drinks (equivalent to half-pints) consumed per day. Accessible at https://archive.ics.uci.edu/ml/datasets/liver+disorders.

Pima. No pre-processing was applied. The target variable used was an indicator outcome variable representing the presence or absense of diabetes per subject. Accessible at https://www.kaggle.com/uciml/pima-indians-diabetes-database?select=diabetes.csv.

Wine. No pre-processing was applied, and the outcome feature was wine quality (ranked 1-10). Only the red wine dataset was used. Accessible at https://archive.ics.uci.edu/ml/datasets/wine+quality.

4 CODE AND DATA AVAILABILITY

Code associated with this paper can be accessed at https://github.com/isebenius/FeatureCollapsing, and includes an implementation of the feature collapsing method using GPy (GPy authors, 2012).

References

Topi Paananen, Juho Piironen, Michael Riis Andersen, and Aki Vehtari. Variable selection for Gaussian processes via sensitivity analysis of the posterior predictive distribution. In Kamalika Chaudhuri and Masashi Sugiyama, editors, *Proceedings of Machine Learning Research*, volume 89 of *Proceedings of Machine Learning Research*, pages 1743–1752. PMLR, 16–18 Apr 2019. URL http://proceedings.mlr.press/v89/paananen19a.html.

GPy authors. GPy: A Gaussian process framework in python. http://github.com/SheffieldML/GPy, 2012.