On Feature Learning of Recursive Feature Machines and Automatic Relevance Determination

Daniel Gedon
Uppsala University
daniel.gedon@it.uu.se

Amirhesam Abedsoltan
UC San Diego
aabedsoltan@ucsd.edu

Thomas B. Schön
Uppsala University
thomas.schon@it.uu.se

Mikhail Belkin
UC San Diego
mbelkin@ucsd.edu

Abstract

Feature learning is a crucial element for the performance of machine learning models. Recently, the exploration of feature learning in the context of kernel methods has led to the introduction of Recursive Feature Machines (RFMs). In this work, we connect diagonal RFMs to Automatic Relevance Determination (ARD) from the Gaussian process literature. We demonstrate that diagonal RFMs, similar to ARD, serve as a weighted covariate selection technique. However, they are trained using different paradigms: RFMs use recursive iterations of the so-called Average Gradient Outer Product, while ARD employs maximum likelihood estimation. Our experiments show that while the learned features in both models correlate highly across various tabular datasets, this correlation is lower for other datasets. Furthermore, we demonstrate that the RFM effectively captures correlation between covariates, and we present instances where the RFM outperforms both ARD and diagonal RFM.

1 Introduction

Feature learning is pivotal in machine learning due to its impact on model performance, with much work on learning useful features (Bengio et al., 2013). While feature learning is not clearly defined, successful models such as deep neural networks seem to harness useful representations during training. Recently, Recursive Feature Machines (RFMs) were introduced as a feature-learning kernel machine (Radhakrishnan et al., 2022). The authors comprehensively compared feature matrices learnt from RFMs to those from fully connected neural networks. They highlighted the similarities, noting that both learning methods often produce highly correlated feature matrices. This observation was rigorously validated across a wide range of tabular datasets.

We establish a connection between diagonal RFMs and Automatic Relevance Determination (ARD) as presented in Neal (1996), from the Gaussian process (GP) literature (Rasmussen and Williams, 2006). Both diagonal RFM and ARD can be interpreted as techniques that assign a unique length scale parameter to each covariate, which is then learned during optimization. While RFM employs recursive iterations of the Average Gradient Outer Product (AGOP), ARD utilizes maximum likelihood estimation (MLE). Our experiments indicate that while the correlation of feature matrices learned in both models correlate highly across various datasets, we observe datasets with a low correlation of learnt features.

In our study, we integrate RFMs into GPs to evaluate their performance against ARD. Although both methods exhibit comparable performance on tabular datasets, we highlight that RFMs can be especially beneficial when dealing with correlated covariates, a scenario where ARD falls short in capturing these correlations.

Preprint.

2 Method

We contrast two paradigms to learn features in kernel-based predictive models: MLE as used in the Gaussian Process literature and AGOP as utilized in the recently proposed RFMs.

2.1 Kernel machines

Assume $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is a positive semi-definite symmetric kernel function (Aronszajn, 1950) with its corresponding (unique) Reproducing Kernel Hilbert space $\mathcal{H}$. Given training data $\mathcal{D} = (\boldsymbol{X}, \boldsymbol{y}) = \{\boldsymbol{x}_i \in \mathbb{R}^d, y_i \in \mathbb{R}\}_{i=1}^n$ the representer theorem (Kimeldorf and Wahba, 1970) states that the unique solution to the infinite-dimensional optimization problem $\arg\min f \in \mathcal{H} \sum_{i=1}^n (f(\boldsymbol{x}_i) - y_i)^2 + \lambda \|f\|_{\mathcal{H}}^2$ has the form $f(\boldsymbol{x}) = \sum_{i=1}^n \alpha_i k(\boldsymbol{x}, \boldsymbol{x}_i)$, where $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ is the unique solution to the linear system $(k(\boldsymbol{X}, \boldsymbol{X}) + \lambda \boldsymbol{I}_n)\, \boldsymbol{\alpha} = \boldsymbol{y}$.

2.2 Gaussian processes

To utilize the advancement of feature learning in the area of GPs, we extend kernel machines into a probabilistic framework. Thus, we can augment point estimates with reliable uncertainty quantification to obtain the predictive distribution $p(f(\boldsymbol{x}_*)|\boldsymbol{x}_*, \mathcal{D})$ for a new test data point $\boldsymbol{x}_*$.

Model structure We can define a distribution over the predictive function which yields a GP $f \sim \mathcal{GP}(m, k)$ specified by its mean function m and its covariance function. Because of its properties, we utilize the kernel function k as the covariance function in the GP. The posterior predictive distribution of the GP is then given by $p(f(\boldsymbol{x}_*)|\boldsymbol{x}_*, \mathcal{D}) = \mathcal{N}(f(\boldsymbol{x}_*), \mathbb{V}[f(\boldsymbol{x}_*)])$, with the mean $f(\boldsymbol{x}) = \sum_{i=1}^n \alpha_i k(\boldsymbol{x}, \boldsymbol{x}_i)$ and the covariance $\mathbb{V}[f(\boldsymbol{x}_*)] = k(\boldsymbol{x}_*, \boldsymbol{x}_*) - \boldsymbol{k}_*^\top (\boldsymbol{K} + \sigma^2 \boldsymbol{I})^{-1} \boldsymbol{k}_*$. We denote the kernel matrix as $\boldsymbol{K}$ with $K_{i,j} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$, $\boldsymbol{k}_* = k(\boldsymbol{X}, \boldsymbol{x}_*)$ and the measurement noise variance as σ^2. For the mean function, we choose $m = 0$. The choice of kernel encodes high-level assumptions about the resulting function. We consider an exponential kernel of the form $k(\boldsymbol{x}, \boldsymbol{z}) = \exp(g(\boldsymbol{x}, \boldsymbol{z}))$. When we define $g(\boldsymbol{x}, \boldsymbol{z}) = -\frac{1}{2\ell^2}\|\boldsymbol{x} - \boldsymbol{z}\|^2$, we arrive at the widely adopted Radial Basis Function (RBF) kernel. Conversely, $g(\boldsymbol{x}, \boldsymbol{z}) = -\frac{1}{\ell}\|\boldsymbol{x} - \boldsymbol{z}\|$ leads to the Laplace kernel.

Within the GP framework, explicit feature learning can be achieved through the extension with ARD (Neal, 1996). The RBF kernel is extended by using $g(\boldsymbol{x}, \boldsymbol{z}) = -\frac{1}{\ell^2}\|\boldsymbol{x} - \boldsymbol{z}\|_{\boldsymbol{M}}^2$ with $\boldsymbol{M}^{-1} = \text{diag}([\ell_1^2, \ldots, \ell_d^2])$ and similarly for the Laplace kernel. Here we utilize the Mahalanobis distance as $\|\boldsymbol{x} - \boldsymbol{z}\|_{\boldsymbol{M}} := \sqrt{(\boldsymbol{x} - \boldsymbol{z})^T \boldsymbol{M} (\boldsymbol{x} - \boldsymbol{z})}$. Note that in this setup the length scale ℓ acts globally on all covariates and could be incorporated into $\boldsymbol{M}$. Therefore, $\boldsymbol{M}$ is a diagonal matrix which re-weights the individual covariates. Thus, in this context feature learning amounts to re-weighting covariates through the diagonal $\boldsymbol{M}$ which is not possible without ARD.

Training The parameters $\boldsymbol{\theta}$ of the kernel include noise variance σ and length scale ℓ for the RBF and Laplace kernel. For the kernel with ARD, we have one length scale parameter ℓ_i for each covariate i. Optimization of the parameters is commonly performed using the MLE framework. Specifically, we can estimate these parameters in a Bayesian framework by minimizing the Negative Log Likelihood (NLL), defined as $-\log p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta})$.

2.3 Recursive Feature Machines

A fundamental limitation of kernel machines is their reliance on kernel functions that are not adaptive to data. As a result, for certain tasks, kernel machines can significantly underperform compared to neural networks. Recently, RFMs have emerged as a type of kernel machine that is capable of learning features, making them data-adaptive.

Model structure To develop kernel machines that can learn features, RFM integrates a positive semi-definite, symmetric matrix, $\boldsymbol{M}$, as a learnable parameter into the kernel function. Specifically, this is suited for kernel functions that depend on the distance between points, such as $k(\boldsymbol{x}, \boldsymbol{z}) = \phi(\|\boldsymbol{x} - \boldsymbol{z}\|^2)$ where $\phi : \mathbb{R} \to \mathbb{R}$ and $\boldsymbol{x}, \boldsymbol{z} \in \mathbb{R}^d$. We incorporate the learnable matrix $\boldsymbol{M}$ by using the Mahalanobis distance $\|\boldsymbol{x} - \boldsymbol{z}\|_{\boldsymbol{M}}$ as defined above.Therefore, the matrix $\boldsymbol{M}$ re-weights the individual covariates and can incorporate correlation between covariates, for which we call $\boldsymbol{M}$ the *feature matrix*. While

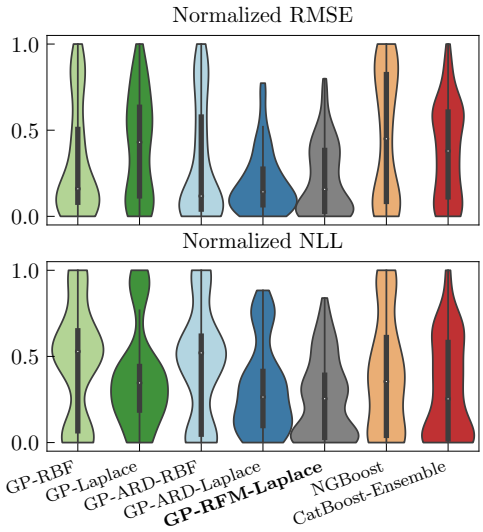

Figure 1: Violin plot including boxplot results on RMSE and NLL with the UCI benchmark.

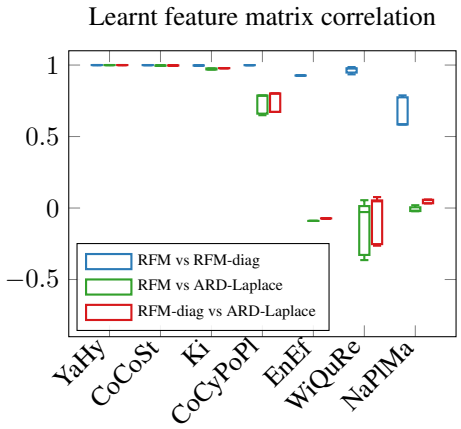

Figure 2: Boxplot of correlation between the diagonal of feature matrix in RFM, RFM-diag and ARD-Laplace for all UCI benchmark datasets.

any kernel function for ϕ can be used, we utilize the Laplace kernel based on the Mahalanobis distance $k_{\boldsymbol{M}}(\boldsymbol{x}, \boldsymbol{z}) := \exp\left(-\frac{1}{\ell}\left\|\boldsymbol{x}-\boldsymbol{z}\right\|_{\boldsymbol{M}}\right)$. The Laplace kernel is due to its tail behavior more robust to outliers compared to an RBF kernel also shown in its often superior empirical performance. The prediction function corresponding to this kernel is given by $f_{\boldsymbol{M}}(\boldsymbol{x}) = k_{\boldsymbol{M}}(\boldsymbol{x}, \boldsymbol{X})\boldsymbol{\alpha}$ with $\boldsymbol{\alpha} = k_{\boldsymbol{M}}(\boldsymbol{X}, \boldsymbol{X})^{-1}\boldsymbol{y}$.

Training To learn the feature matrix $\boldsymbol{M}$ we make use of the proposed idea of the AGOP from Radhakrishnan et al. (2022): We start by initializing $\boldsymbol{M}^{(0)} = \boldsymbol{I}_d$. Then, at each iteration step t we first solve for the kernel weights $\boldsymbol{\alpha}$ with fixed $\boldsymbol{M}$. Second, we update $\boldsymbol{M}$ using the AGOP defined as

$$\boldsymbol{M}^{(t+1)} = \frac{1}{n}\sum_{i=1}^{n} \nabla_{\boldsymbol{x}} f_{\boldsymbol{M}^{(t)}}(\boldsymbol{x}_i) \nabla_{\boldsymbol{x}} f_{\boldsymbol{M}^{(t)}}(\boldsymbol{x}_i)^T.$$

Intuitively, RFM prioritises the covariates that have the most impact on the prediction function. Thus, it learns the presentation most relevant to the underlying task.

A special case of RFMs is when we restrict the feature matrix $\boldsymbol{M}$ to be diagonal. This is equivalent to learning a separate length scale for each covariate. We denote this model as *RFM-diag*.

Probabilistic extension We include the RFM-based kernel in the GP framework. To disentangle the training of this *GP-RFM*, we first learn the feature matrix $\boldsymbol{M}$ using the recursive AGOP iteration. Second, we learn the GP-specific kernel parameters $\boldsymbol{\theta}$ by MLE optimization with fixed $\boldsymbol{M}$.

2.4 Recursive Feature Machines vs Automatic Relevance Determination

RFMs and GPs with ARD both address the challenge of adaptive feature learning in kernel machines. Both methods introduce adaptability through learnable parameters within the feature matrix $\boldsymbol{M}$. ARD extends the GP framework with a diagonal feature matrix, $\boldsymbol{M}$, enabling length scale adjustments for individual covariates. In contrast, RFMs are more general by the ability to utilise a non-diagonal feature matrix $\boldsymbol{M}$ which enables RFMs to capture the correlation between covariates for the task by off-diagonal elements in $\boldsymbol{M}$.

In terms of training, GPs with ARD on the one side are optimized through MLE with parameters θ including the diagonal elements of $\boldsymbol{M}$. On the other side, RFMs iteratively optimize the feature matrix $\boldsymbol{M}$ with the AGOP and the kernel weights through least squares solutions.

3 Results

We experiment with 7 datasets inspired by Duan et al. (2020) from the UCI benchmark (Asuncion and Newman, 2007). We follow the protocol in Duan et al. (2020) to hold out 10% of the data as a test set. The remaining data is split into a 70% training set and a 30% validation set in order to tune the hyperparameters. We use grid-search over all combinations of hyperparameters and select the best hyperparameters based on the NLL on the validation set. Finally, we train the model on the full training set and evaluate it on the test set. The process is repeated for 20 random seeds.

3.1 Performance of RFM and ARD

To highlight that RFM and kernels with ARD learn useful feature matrices, we compare them with a variety of probabilistic baseline methods. For GPs, we consider the *RBF* and *Laplace* kernel. For the kernels with ARD, we choose the *ARD-RBF* which is used in many settings and the *ARD-Laplace* kernel which is rarely used but is a natural extension of the Laplace kernel to ARD. Furthermore, we consider probabilistic extensions of the powerful boosting approaches. Specifically, *NGBoost* (Duan et al., 2020) and *CatBoost-Ensemble* (Prokhorenkova et al., 2018).

In Figure 1 we assess model performance across diverse datasets. Due to varying scales, we normalize metrics for cross-dataset comparisons. We achieve this by calculating the min and max values for each dataset across all methods and seeds, followed by normalizing the results to the range $[0, 1]$. The results indicate that both, the GP-RFM-Laplace and the GP-ARD-Laplace do not only outperform other models in predictive performance through low RMSE but also in terms of uncertainty quantification through NLL. Therefore, features learnt by both methods are meaningful and useful for the task.

3.2 Comparison of learnt features

Given the comparable performance of the GP-RFM-Laplace and the GP-ARD-Laplace, we compare the correlation between the diagonal of the feature matrix of the GP-RFM-Laplace and the GP-ARD-Laplace. Additionally, we train a GP-RFM-Laplace where we restrict the feature matrix $\boldsymbol{M}$ to be diagonal, i.e. we use the RFM-diag kernel.

Figure 2 shows the correlation between the diagonal of the feature matrix for all three methods. While there is a high correlation between all methods for some datasets, there are also datasets where the correlation is low—even between methods learning diagonal features, the RFM-diag and the GP-ARD-Laplace. This indicates that learning with AGOP in RFMs or with MLE for the kernels with ARD may result in the same features in some cases but is not guaranteed to do so. Further investigation is required to understand the differences between feature learning in the two methods.

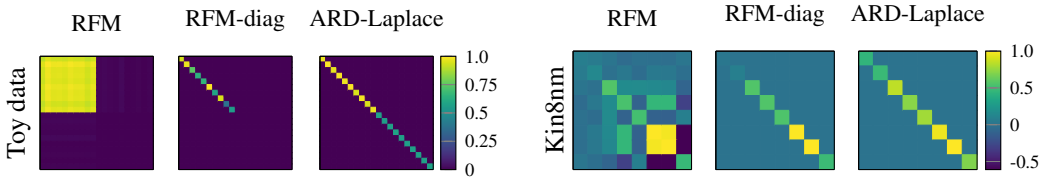

Figure 3: Normalized feature matrices $\boldsymbol{M}$ for toy dataset (left) and Kin8nm dataset (right).

3.3 Visualizing feature matrices

To highlight the difference between both kernels, we generate a toy dataset where the covariates $\boldsymbol{x} \sim \mathcal{N}(0, \boldsymbol{I}_d)$ are independent and the labels are the squared sum of the first 10 covariates $y = (\sum_{i=1}^{10} \boldsymbol{x}_{[i]})^2$. This dataset is challenging for the kernels with ARD as it requires learning the covariate correlation. We can compute the Jacobian of the labels with respect to the covariates to obtain the true feature matrix $\boldsymbol{M}$ which is a block matrix with a 10×10 block of $\frac{1}{n}\sum_{i=1}^{n}(\sum_{j=1}^{10} \boldsymbol{x}_{i[j]})^2$ and the remaining entries are zero, where $\boldsymbol{x}_{i[j]}$ denotes the jth dimension of the ith sample. Experimentally, in Figure 3 (left) as we expected the RFM learns relevant covariate correlation as indicated by

nonzero off-diagonal values of the feature matrix while the diagonal methods are unable to capture this relation.

In Figure 3 (right), we compare the three methods on the Kin8nm dataset, for which the diagonal between the methods correlates highly as seen in Figure 2. Here, we can qualitatively confirm this observation. Again, in this dataset, the RFM captures the non-zero covariate correlation. Therefore, RFMs when they are not restricted to the RFM-diag can learn more complex features than kernels with ARD, which allows for both of these datasets to be predicted more accurately as there is a necessary covariate correlation to be modelled.

4 Conclusion

We demonstrated that while the RFM is related to ARD, it has the potential to be more potent. Observations suggest that diagonal RFMs and ARD-based kernels frequently perform on par with or even surpass the RFM, possibly because of RFM's increased sample complexity. Future studies should (1) theoretically analyse the similarities and differences between MLE and AGOP-based optimization to identify the origin of the high feature correlation which we observe for some datasets and (2) examine the sample complexity disparities between RFM and RFM-diag or ARD.

Acknowledgments and Disclosure of Funding

A.A and M.B are grateful for the support from the National Science Foundation (NSF) and the Simons Foundation for the Collaboration on the Theoretical Foundations of Deep Learning (https://deepfoundations.ai/) through awards DMS-2031883 and #814639 and the TILOS institute (NSF CCF-2112665). D.G and T.S are partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. The computations were enabled by the Berzelius resource provided by the Knut and Alice Wallenberg Foundation at the Swedish National Supercomputer Centre.

References

Nachman Aronszajn. Theory of reproducing kernels. *Transactions of the American mathematical society*, 68(3):337–404, 1950.

Arthur Asuncion and David Newman. Uci machine learning repository, 2007.

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

Tony Duan, Avati Anand, Daisy Yi Ding, Khanh K Thai, Sanjay Basu, Andrew Ng, and Alejandro Schuler. Ngboost: Natural gradient boosting for probabilistic prediction. In *International conference on machine learning*, pages 2690–2700. PMLR, 2020.

George S Kimeldorf and Grace Wahba. A correspondence between bayesian estimation on stochastic processes and smoothing by splines. *The Annals of Mathematical Statistics*, 41(2):495–502, 1970.

Radford M Neal. Bayesian learning for neural networks, 1996.

Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. Catboost: unbiased boosting with categorical features. *Advances in neural information processing systems*, 31, 2018.

Adityanarayanan Radhakrishnan, Daniel Beaglehole, Parthe Pandit, and Mikhail Belkin. Feature learning in neural networks and kernel machines that recursively learn features. *arXiv preprint arXiv:2212.13881*, 2022.

Carl Edward Rasmussen and Christopher KI Williams. *Gaussian processes for machine learning*, volume 1. Springer, 2006.