

DISTRIBUTED NON-PARAMETRIC DEEP AND WIDE NETWORKS

Anonymous authors

Paper under double-blind review

ABSTRACT

In recent work, it was shown that combining multi-kernel based support vector machines (SVMs) can lead to near state-of-the-art performance on an action recognition dataset (HMDB-51 dataset). In the present work, we show that combining distributed Gaussian Processes with multi-stream deep convolutional neural networks (CNN) alleviate the need to augment a neural network with hand-crafted features. In contrast to prior work, we treat each deep neural convolutional network as an expert wherein the individual predictions (and their respective uncertainties) are combined into a Product of Experts (PoE) framework.

INTRODUCTION

Recognizing actions in a video stream requires the aggregation of temporal as well as spatial features (as in object classification). These video streams, unlike still images, have short and long temporal correlations, attributes that single frame convolutional neural networks fail to discover. Therefore, the first hurdle to reach human-level performance is designing feature extractors that can learn this latent temporal structure. Nonetheless, there has been much progress in devising novel neural network architecture since the work of [Karpathy et al. \(2014\)](#). Another problem is the large compute, storage and memory requirement for analysing moderately sized video snippets. One requires a relatively larger computing resource to train ultra deep neural networks that can learn the subtleties in temporal correlations, given varying lighting, camera angles, pose, etc. It is also difficult to utilise standard image augmentation (like random rotations, shears, flips, etc.) techniques on a video stream. Additionally, features for a video stream (unlike static images) evolve with a dynamics across several orders of time-scales.

Nonetheless, the action recognition problem has reached sufficient maturity using the two-stream deep convolutional neural networks (CNN) framework ([Simonyan & Zisserman, 2014](#)). Such a framework utilises a deep convolutional neural network (CNN) to extract static RGB (Red-Green-Blue) features as well as motion cues by deconstructing the optic-flow of a given video clip. Notably, there has been plenty of work in utilising a variety of network architectures for factorising the RGB and optical-flow based features. For example, an inception network ([Szegedy et al., 2016](#)) uses 1×1 convolutions in its inception block to estimate cross-channel corrections, which is then followed by the estimation of cross-spatial and cross-channel correlations. A residual network (ResNet), on the other hand, learns residuals on the inputs instead of learning unreferenced functions ([He et al., 2016](#)). While such frameworks have proven useful for many action recognition datasets (UCF101, UCF50, etc.), they are yet to show promise where videos have varying signal-to-noise ratio, viewing angles, etc.

We improve upon existing technology by combining Inception networks and ResNets using a Gaussian Process classifier that is further combined in a product-of-expert (PoE) framework to yield, to the best of our knowledge, a state-of-the-art performance on the HMDB51 data-set ([Kuehne et al., 2013](#)). Under a Bayesian setting, our framework provide not only mean predictions, but also the uncertainty associated with each prediction. Notably, our work forwards the following contributions:

- We introduce a framework that allow for independent multi-stream deep neural networks, enabling horizontal scalability
- Ability to classify video snippets that have heterogeneity regarding camera angle, video quality, pose, etc.

- Combine deep convolutional neural networks with non-parametric Bayesian models, wherein there is a possibility to train them using less amount of data
- Demonstrate the utility of model averaging that takes uncertainty around mean predictions into account

METHODS

In this section, we describe the dataset, the network architectures and the nonparametric Bayesian setup that we utilise in our four-stream CNN pillar network for activity recognition. We refer the readers to the original network architectures in [Wang et al. \(2016\)](#) and [Ma et al. \(2017\)](#) for further technical details. Utilising classification methodologies like AdaBoost, gradient boosting, random forests, etc. provide us with accuracies in the range of 5-55% for this dataset, for either the RGB or the optic-flow based features.

DATASET

The HMDB51 dataset ([Kuehne et al., 2013](#)) is an action classification dataset that comprises of 6,766 video clips which have been divided into 51 action classes. Although a much larger UCF-sports dataset exists with 101 action classes ([Soomro et al., 2012](#)), the HMDB51 has proven to be more challenging. This is because each video has been filmed using a variety of viewpoints, occlusions, camera motions, video quality, etc. anointing the challenges of video-based prediction problems. The second motivation behind using such a dataset lies in the fact that HMDB51 has storage and compute requirement that is fulfilled by a modern workstation with GPUs – alleviating deployment on expensive cloud-based compute resources.

All experiments were done on Intel Xeon E5-2687W 3 GHz 128 GB workstation with two 12GB nVIDIA TITAN Xp GPUs. As in the original evaluation scheme, we report accuracy as an average over the three training/testing splits.

INCEPTION LAYERS FOR RGB AND FLOW EXTRACTION

We use the inception layer architecture described in [Wang et al. \(2016\)](#). Each video is divided into N segments, and a short sub-segment is randomly selected from each segment so that a preliminary prediction can be produced from each snippet. This is later combined to form a video-level prediction. An Inception with Batch Normalisation network ([Ioffe & Szegedy, 2015](#)) is utilised for both the spatial and the optic-flow stream. The feature size of each inception network is fixed at 1024. For further details on network pre-training, construction, etc. please refer to [Wang et al. \(2016\)](#).

RESIDUAL LAYERS FOR RGB AND FLOW EXTRACTION

We utilise the network architecture proposed in [Ma et al. \(2017\)](#) where the authors leverage recurrent networks and convolutions over temporally constructed feature matrices as shown in Fig. 1. In our instantiation, we truncate the network to yield 2048 features, which is different from [Ma et al. \(2017\)](#) where these features feed into an LSTM (Long Short Term Memory) network. The spatial stream network takes in RGB images as input with a ResNet-101 ([He et al., 2016](#)) as a feature extractor; this ResNet-101 spatial-stream ConvNet has been pre-trained on the ImageNet dataset. The temporal stream stacks ten optical flow images using the pre-training protocol suggested in [Wang et al. \(2016\)](#). The feature size of each ResNet network is fixed at 2048. For further details on network pre-training, construction, etc. please refer to [Ma et al. \(2017\)](#).

NON-PARAMETRIC BAYESIAN CLASSIFICATION

Gaussian Processes (GP) emerged out of filtering theory ([Wiener, 1949](#)) in non-parametric Bayesian statistics via work done in geostatistics ([Matheron, 1973](#)). Put simply, GPs are collection of random variables that have a joint Gaussian distribution,

$$\begin{aligned}
\text{Observation:} \quad & y | f, \phi \sim \prod_{i=1}^n p(y_i | f_i, \phi) \\
\text{GP Prior:} \quad & f(x) | \theta \sim \mathcal{GP}(m(x), k(x, \tilde{x} | \theta)) \\
\text{Hyperprior:} \quad & \theta, \phi \sim p(\theta) p(\phi)
\end{aligned} \tag{1}$$

where, $k(x, \tilde{x} | \theta)$ is the kernel function parameterized by θ ; ϕ is the parameter of the observation model; $f(x)$ is the latent function evaluated at x i.e., the features. y denotes the class of the input features and $\{\phi, \theta\} \in \chi$ denote the set of hyper-parameters.

For multi-class problem with a non-Gaussian likelihood (softmax; $p(y_i | f(x_i)) = \frac{\exp(f_i^{y_i})}{\sum_{c=1}^{51} \exp(f_i^c)}$), the conditional posterior is approximated via the Laplace approximation (Williams & Barber, 1998) i.e., a second order Taylor expansion of $\log p(f | y, \theta, \phi)$ around the mode \hat{f} as,

$$\begin{aligned}
p(f | \mathcal{D}, \theta, \phi) &\approx q(f | \mathcal{D}, \theta, \phi) = \mathcal{N}(f | \hat{f}, \Sigma) \\
\hat{f} &= \arg \max_f p(f | \mathcal{D}, \theta, \phi) \\
\Sigma^{-1} &= -\nabla \nabla \log p(f | \mathcal{D}, \theta, \phi) |_{f=\hat{f}} = K_{f,f}^{-1} + W \\
W_{ii} &= \nabla_{f_i} \nabla_{f_i} \log p(y | f_i, \phi) |_{f_i=\hat{f}_i}
\end{aligned} \tag{2}$$

\mathcal{D} is the (input,output) tuple. After the Laplace approximations, the approximate posterior distribution becomes,

$$\begin{aligned}
\hat{f} | D, \theta, \phi &\sim GP(m_p(\tilde{x}), k_p(\tilde{x}, \tilde{x}')) \\
m_p(\tilde{x}) &= k(\tilde{x}, X) \nabla \log p(y | f) |_{f=\hat{f}} \\
k_p(\tilde{x}, \tilde{x}') &= k(\tilde{x}, \tilde{x}') - k(\tilde{x}, X) (K_{f,f} + W)^{-1} k(X, \tilde{x}')
\end{aligned} \tag{3}$$

Finally, we can evaluate the approximate conditional predictive density of \tilde{y}_i ,

$$p(\tilde{y}_i | D, \theta, \phi) \approx \int p(\tilde{y}_i | \tilde{f}_i, \phi) q(\tilde{f}_i | D, \theta, \phi) d\tilde{f}_i \tag{4}$$

PRODUCT OF EXPERTS

For each of the neural network, we subdivide the training set into $K = 7$ sub-sets so that K different GPs could be trained, giving us 28 GPs for the 4 deep networks (2 Inception networks and 2 ResNets) that we have trained in the first part of our training. We assume that each of the 7 GPs are independent, such that the marginal likelihood in our product of expert (PoE) becomes,

$$\begin{aligned}
p(y | X, \chi) &\approx \prod_{k=1}^K p_k(y^{(k)} | X^{(k)}, \chi) \\
p_k(y^{(k)} | X^{(k)}, \chi) &= \frac{1}{2} y^{(k)} \left(K_\theta^{(k)} + \sigma_\phi^2 \mathbf{I} \right)^{-1} y^{(k)} \\
&\quad - \frac{1}{2} \log |K_\theta^{(k)} + \sigma_\phi^2 \mathbf{I}|
\end{aligned} \tag{5}$$

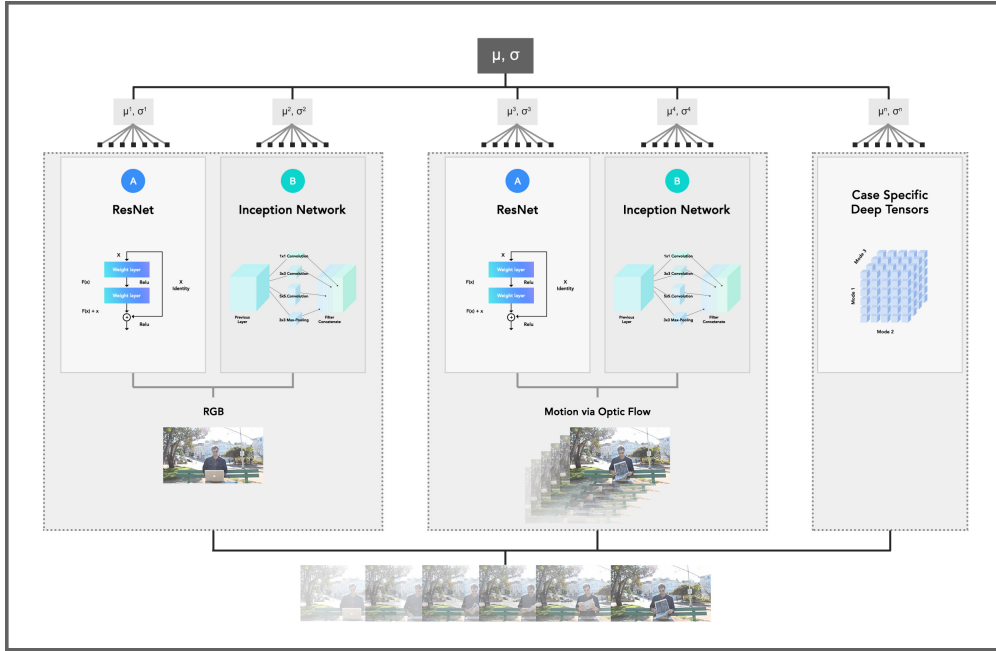


Figure 1: **The distributed non-parametric deep and wide network framework:** Each pillar represents either a single ultra-deep neural network or other feature tensors that can be learnt automatically from the input data. For action recognition, we factorize the static (RGB), and dynamic (optic flow) features using a ResNet and an Inception Network. Using the features of the last layer, we train seven Gaussian Processes for each of the network, which is combined under a Product-of-Experts (PoE) formalism. This hierarchy is then fused again to give us a prediction of action types.

What we have done is to reduce the computational expenditure from $\mathcal{O}(n^3)$ to $\mathcal{O}(n_k^3)$. Notice that unlike GPs with inducing inputs or variational parameters such a distributed GP does not require optimisation of additional parameters. Finally, a product-of-GP-experts is instantiated that predicts the function f_* at the test point x_* as,

$$\begin{aligned}
 p(f_* | x_*, D) &= \prod_{k=1}^K p_k(f_* | x_*, D^{(k)}) \\
 \mu_*^{poe} &= \frac{1}{(\sigma_*^{poe})^2} \sum_k \sigma_k^{-2}(x_*) \mu_k(x_*) \\
 \frac{1}{(\sigma_*^{poe})^2} &= \sum_k \sigma_k^{-2}(x_*) \tag{6}
 \end{aligned}$$

RESULTS

We used 3570 videos from HMDB51 as the training data-set; this was further split into seven sub-sets, each with 510 videos. We select ten videos randomly chosen from each category, and each sub-set is non-overlapping. Based on seven sub-sets, seven GPs are trained on different features (RGB and Flow) from different Networks (TSN-Inception Wang et al. (2016) and ResNet-LSTM Ma et al. (2017)). In total, twenty-eight GPs are generated. The features for both the RGB and the optical flow were extracted from the last connected layer with 1024 dimension for the Inception network and 2048 for the ResNet network. The fusion is then performed both vertically (seven sub-sets) and horizontally (four networks). The accuracies of individual GPs and different fusion combinations (PoE) on split-1 are shown in Table 1. Fusion-1 represents the results from the fusion of seven GPs for each feature; Fusion-2 show the fusion result of RGB and Flow using different

Table 1: GP-PoE and SVM results for the HMDB51 data-set on split-1

Accuracy [%]	Inception-RGB	Inception-Flow	ResNet-RGB	ResNet-Flow
GP-1	51.4	59.5	52.7	58.9
GP-2	52.0	59.7	51.9	59.1
GP-3	50.1	60.3	49.7	59.9
GP-4	48.7	58.5	49.5	59.1
GP-5	48.2	59.3	49.0	59.5
GP-6	52.0	59.5	52.2	57.9
GP-7	51.1	58.8	51.8	58.1
Average	50.5	59.4	51.0	58.9
Fusion-1	54.6	62.6	54.8	61.6
Fusion-2	69.7		68.2	
Fusion-all	75.7			
SVM-SingleKernel	54.0	61.0	53.1	58.5
SVM-MutliKernels-1	68.1		63.3	
SVM-MutliKernels-2	71.7			

Table 2: Accuracy scores for the HMDB51 data-set

Methods	Accuracy [%]	Reference
Two-stream	59.4	Simonyan & Zisserman (2014)
Rank Pooling (ALL)+ HRP (CNN)	65	Fernando & Gould (2017)
Convolutional Two-stream	65.4	Feichtenhofer et al. (2016)
Pillar Networks + soft-max + cross-entropy	67	Sengupta & Qian (2017)
Temporal-Inception	67.5	Ma et al. (2017)
TS-LSTM	69	Ma et al. (2017)
ST-multiplier network	68.9	Feichtenhofer et al. (2017)
ST-ResNet + iDT	70.3	Ma et al. (2017)
Temporal Segment Network (2/3/7 modalities)	68.5/69.4/71	Wang et al. (2016)
ST-multiplier network + iDT	72.2	Feichtenhofer et al. (2017)
Pillar Networks + SVM-MKL	72.8	Sengupta & Qian (2017)
Pillar Networks + iDT + SVM-MKL	73.0	Sengupta & Qian (2017)
Pillar Networks + MIFS + SVM-MKL	73.3	Sengupta & Qian (2017)
Deep Convolutional Networks + GP-PoE	73.6	this paper
Deep Convolutional Networks + iDT + GP-PoE	75.0	this paper

networks; Fusion-all shows the result by fusion of all the 28 GPs. The average result for three splits is displayed in Table 2. We also demonstrate the improvement when hand-crafted features like iDT are combined with dCNNs to yield an 1.4% improvement.

DISCUSSION

Here, we make two contributions – (a) we build on recently proposed pillar networks ([Sengupta & Qian, 2017](#)) and combine deep convolutional neural networks with non-parametric Bayesian models, wherein they have the possibility of being trained with less amount of data and (b) demonstrate the utility of model averaging that takes uncertainty around mean predictions into account. Combining different methodologies allow us to supersede the current state-of-the-art in video classification especially, action recognition.

We utilised the HMDB-51 dataset instead of UCF101 as the former has proven to be difficult for deep networks due to the heterogeneity of image quality, camera angles, etc. As is well-known, videos contain extensive long-range temporal structure; using different networks (2 ResNets and 2 Inception networks) to capture the subtleties of this temporal structure is an absolute requirement. Since each network implements a different non-linear transformation, one can utilise them to learn very deep yet different features. Utilising the distributed-GP architecture then enables us to parcel-

late the feature tensors into computable chunks (by being distributed) of input for a Gaussian Process classifier. Such an architectural choice, therefore, enables us to scale horizontally by plugging in a variety of networks *as per requirement*. While we have used this architecture for video based classification, there is a wide range of problems where we can apply this methodology – from speech processing (with different networks) to natural-language-processing (NLP).

Ultra deep convolutional networks have been influential for a variety of problems, from image classification to natural language processing (NLP). Recently, there has been work on combining the Inception network with that of a Residual network such that the resulting network builds on the advantages offered by either network in isolation (Szegedy et al., 2017). In future, it would be useful to see how different are the features when they are extracted from Inception module, ResNet module or a combination of both. Not only this, a wide variety of hand-crafted features can also be augmented as inputs to the distributed GPs; our initial experiments using the iDT features show that this is indeed the case (a 1.4% improvement). Input data can also be augmented using RGB difference or optic flow warps, as had been done in Wang et al. (2016).

Also, the second stage of training, i.e., the GP classifiers work with far fewer examples than what a deep learning network requires. It would be useful to see how our framework performs on immensely large datasets such as the Youtube-8m data-set (Abu-El-Haija et al., 2016). Additionally, recently published Kinetics human action video dataset from DeepMind (Kay et al., 2017) is equally attractive, as pre-training, our framework on this dataset before fine-grained training on HMDB-51 will invariably increase the accuracy of the current network architecture.

The Bayesian product-of-GPs would suffer from a problem were we to increase the number of experts. This is because the precision of the experts adds up which leads to overconfident predictions, especially in the absence of data. In unpublished work, we have utilised generalised Product of Experts (gPoE) (Cao & Fleet, 2014) and Bayesian Committee Machine (BCM) (Tresp, 2000) to increase the fidelity of our predictions. These would be reported in a subsequent publication along with results from a robust Bayesian Committee Machine (rBCM) which includes the product-of-GPs and the BCM as special cases (Deisenroth & Ng, 2015).

For inference, we have limited our experiments to the Laplace approximation inference under a distributed GP framework. An alternative inference methodology for multi-class classification include (stochastic) expectation propagation (Riihimäki et al.; Villacampa-Calvo & Hernández-Lobato, 2017) or variational approximations (Hensman et al., 2015). Free-energy minimization is attractive simply due to lower computational overhead. Indeed, it comes with its problems such as underestimation of the variability of the posterior density, inability to describe multi-modal densities and the inaccuracy due to the presence of multiple equilibrium points. All being said, some of these problems are also shared by state-of-the-art MCMC samplers for dynamical systems. Due to the flexibility of utilising GPUs, both methods (variational inference and EP) can prove to be computationally efficient, especially for streaming data. Thus, there is a scope of future work where one can apply these inference methodologies and compare it with vanilla Laplace approximations, as utilised here.

REFERENCES

- Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. YouTube-8M: a large-scale video classification benchmark. 2016.
- Yanshuai Cao and David J Fleet. Generalized product of experts for automatic and principled fusion of Gaussian process predictions. *arXiv preprint arXiv:1410.7827*, 2014.
- Marc Peter Deisenroth and Jun Wei Ng. Distributed Gaussian processes. *arXiv preprint arXiv:1502.02843*, 2015.
- Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1933–1941, 2016.

- Christoph Feichtenhofer, Axel Pinz, and Richard P Wildes. Spatiotemporal multiplier networks for video action recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.
- Basura Fernando and Stephen Gould. Discriminatively learned hierarchical rank pooling networks. *arXiv preprint arXiv:1705.10420*, 2017.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.
- James Hensman, Alexander G de G Matthews, and Zoubin Ghahramani. Scalable variational Gaussian process classification. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, 2015.
- Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pp. 448–456, 2015.
- Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 1725–1732, 2014.
- Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The Kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.
- Hilde Kuehne, Hueihan Jhuang, Rainer Stiefelhagen, and Thomas Serre. HMDB51: a large video database for human motion recognition. In *High Performance Computing in Science and Engineering 12*, pp. 571–582. Springer, 2013.
- Chih-Yao Ma, Min-Hung Chen, Zsolt Kira, and Ghassan AlRegib. TS-LSTM and Temporal-Inception: Exploiting spatiotemporal dynamics for activity recognition. *arXiv preprint arXiv:1703.10667*, 2017.
- Georges Matheron. The intrinsic random functions and their applications. *Advances in applied probability*, 5(3):439–468, 1973.
- Jaakko Riihimäki, Pasi Jylänki, and Aki Vehtari. Nested expectation propagation for Gaussian Process classification with a multinomial probit likelihood.
- B. Sengupta and Y. Qian. Multi-kernel deep learning of deep convolutional features for action recognition. *ICCV Joint Video and Language Understanding Workshop*, 2017.
- Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, pp. 568–576, 2014.
- Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: a dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.
- Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Re-thinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pp. 2818–2826, 2016.
- Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*, pp. 4278–4284, 2017.
- Volker Tresp. A Bayesian committee machine. *Neural computation*, 12(11):2719–2741, 2000.
- C. Villacampa-Calvo and D. Hernández-Lobato. Scalable Multi-Class Gaussian Process Classification using Expectation Propagation. *ArXiv e-prints*, 2017.

Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European Conference on Computer Vision*, pp. 20–36. Springer, 2016.

Norbert Wiener. *Extrapolation, interpolation, and smoothing of stationary time series*, volume 7. MIT press Cambridge, MA, 1949.

Christopher KI Williams and David Barber. Bayesian classification with Gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1342–1351, 1998.