DD2434 Machine Learning, Advanced Course Assignment 1

Thomas Gaddy

November 30, 2017

Abstract

In this assignment several different aspects of building models of data are examined. The first task consists of a supervised scenario involving a model of a specific relationship between two different domains. The second task concerns unsupervised learning and how to learn a new representation of the data. This is related to finding hidden structures or patterns in the data which might contain important information. Finally, model selection is considered. This is important as it gives us the tool to design different models and then choose the one that best represents our data.

1 The Prior $p(\mathbf{X}), p(\mathbf{W}), p(f)$

1.1 Theory

We assume the following form of the likelihood:

$$p(\mathbf{t}_i|f, \mathbf{x}_i) \sim \mathcal{N}(f(\mathbf{x}_i), \sigma^2\mathbf{I}) \tag{1}$$

Question 1: *Why is a Gaussian likelihood a sensible choice? What does it mean that we have chosen a spherical covariance matrix for the likelihood?*

A Gaussian likelihood is sensible because if we assume that the noise around the target variables is due to various independently and identically distributed errors, then by the central limit theorem there will be a Gaussian distribution of noise, i.e. $\epsilon \sim \mathcal{N}(0, \sigma^2)$. This means that the distribution around each target variable will be Gaussian with a mean equal to $f(\mathbf{x}_i)$ and variance equal to the variance of the noise. Spherical covariance implies that there is no covariance between outputs (the elements of the $\mathbf{t}_i$ vector) and that the variances are the same for each output.

Question 2: *If we do not assume that the data points are independent how would the likelihood look then? Remember that* $\mathbf{T} = [\mathbf{t}_1, \ldots, \mathbf{t}_N]$

If it is assumed that each output point is conditionally independent given the input and the mapping, then we can write the likelihood of the data as follows,

$$p(\mathbf{T}|f, \mathbf{X}) = \prod_i^n p(\mathbf{t}_i|f, \mathbf{x}_i) \tag{2}$$

However, if we assume that each output point is not independent, we can determine the likelihood via the chain rule of probability:

$$p(\mathbf{T}|f, \mathbf{X}) = p(\mathbf{t}_1|f, \mathbf{x}_1)p(\mathbf{t}_2|f, \mathbf{t}_1, \mathbf{x}_2)\ldots p(\mathbf{t}_N|f, \mathbf{t}_{1:N-1}, \mathbf{x}_N) \tag{3}$$

We wish to find the mapping f from $\mathbf{x}_i$ to $\mathbf{t}_i$ from the observed data. More specifically, taking uncertainty into account, what we wish to reach is the posterior distribution over the mapping given the observations,

$$p(f|\mathbf{X}, \mathbf{T})$$

1.1.1 Linear Regression

Question 3: *What is the specific form of the likelihood below? Complete the right hand side of the expression in (4)*

If we make an assumption about the structure of noise in the observations such that

$$\mathbf{t}_i = \mathbf{W}\mathbf{x}_i + \boldsymbol{\epsilon},$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma^2)$ and that each out put point is conditionally independent, then we can formulate the likelihood of the data,

$$p(\mathbf{T}|\mathbf{X}, \mathbf{W}) = \prod_i^N \mathcal{N}(\mathbf{W}\mathbf{x}_i, \sigma^2\mathbf{I}) \tag{4}$$

We would like to infer $\mathbf{W}$ from the data. In order to do so, we must define a prior $p(\mathbf{W})$ over the model parameters. A sensible choice would be to pick the conjugate prior, i.e. a Gaussian prior over the parameters,

$$p(\mathbf{W}) = \mathcal{N}(\mathbf{W}_0, \tau^2\mathbf{I}). \tag{5}$$

This prior distribution provides a valuable tool to tell us how likely a parameter is or "how far" it is from our belief.

Question 4: *The prior in (5) is a spherical Gaussian. This means that the "preference" is encoded in terms of a L_2 distance in the space of the parameters. With this view, how would the preference change if the preference was rather encoded using a L_1 norm? Compare and discuss the different type of solutions these two priors would encode*

The preference would be encoded via a Laplacian prior. This would result in the LASSO regularization for a point estimate versus the ridge regression obtained with a spherical Gaussian prior. This could also lead to a sparser model, as weight values can be driven to zero with an L_1 norm. This can be especially useful when inference is important, as irrelevant or less important input variables are disregarded. An L_2 regularization term would lead to smaller weights, but they would not be driven to zero.

Question 5: *Derive the posterior over the parameters. Please, do these calculations by hand as it is very good practice. However, in order to pass the assignment you only need to outline the calculation and highlight the important steps. You can make derivations for individual samples $(\mathbf{x}_i, \mathbf{t}_i)$ and then generalize to the dataset or operate on matrices keeping the concept of vectorization in mind.*

- Briefly comment/discuss the form (mean and covariance).
- What is the effect of the constant Z, are we interested in this?

The posterior is the object that integrates our prior belief with the data. Since both the prior and likelihood are Gaussian, the posterior will also be Gaussian. For each input-output pair we can either consider D (dimension of the outputs) separate regression problems or a single regression problem. However, in order to facilitate the derivation, we can consider D conditionally independent regression problems for all input-output pairs where weight vectors for each regression problem are stacked horizontally and subsequently generalize to a multidimensional output. In this case, we still consider the entire NXQ input matrix $\mathbf{X}$ but only consider a column (one dimension of outputs) of the NXD matrix $\mathbf{T}$ and the corresponding column of the QXD weight matrix $\mathbf{W}$. Let us call these vectors $\mathbf{t}$ and $\mathbf{w}$, where $p(\mathbf{t}|\mathbf{X}, \mathbf{w}) = \mathcal{N}(\mathbf{X}\mathbf{w}, \sigma^2\mathbf{I})$ and $p(\mathbf{w}) = \mathcal{N}(\mathbf{w}_0, \tau^2\mathbf{I})$. If we call the mean $\mathbf{m}_N$ and the variance $\mathbf{S}_N$ for the posterior over the weight vector $\mathbf{w}$ we have,

$$p(\mathbf{w}|\mathbf{X}, \mathbf{t}) = \mathcal{N}(\mathbf{w}|\mathbf{w}_N, \mathbf{S}_N). \tag{6}$$

For now we are concerned with the exponential part of the Gaussian, which can be written as,

$$-\frac{1}{2}(\mathbf{w}-\mathbf{w}_N)^T\mathbf{S}_N^{-1}(\mathbf{w}-\mathbf{w}_N). \tag{7}$$

Expanding this out we are left with,

$$\overbrace{-\frac{1}{2}\mathbf{w}^T\mathbf{S}_N^{-1}\mathbf{w}}^{\text{Quad}}+\overbrace{\mathbf{w}^T\mathbf{S}_N^{-1}\mathbf{w}_N}^{\text{Lin}}-\overbrace{\frac{1}{2}\mathbf{w}_N^T\mathbf{S}_N^{-1}\mathbf{w}_N^T}^{\text{Const}} \tag{8}$$

Where the quadratic, linear, and constant terms in $\mathbf{w}$ have been highlighted. By Bayes' Rule, we know that the posterior is proportional to the product of the prior (5) and the likelihood (4),

$$p(\mathbf{w}|\mathbf{X},\mathbf{t})=\frac{1}{Z}p(\mathbf{t}|\mathbf{X},\mathbf{w})p(\mathbf{w}), \tag{9}$$

where Z represents the *evidence*. This factor is the same for all possible weights considered. As a result, we are not directly concerned with this value for now as we learn the weights from the data. This will become useful, however, when we perform Bayesian model comparison. Considering only the exponential part of this Gaussian we have,

$$-\frac{1}{2}(\mathbf{w}-\mathbf{w}_0)^T\frac{1}{\tau^2}\mathbf{I}(\mathbf{w}-\mathbf{w}_0)-\frac{1}{2}(\mathbf{t}-\mathbf{X}\mathbf{w})^T\frac{1}{\sigma^2}\mathbf{I}(\mathbf{t}-\mathbf{X}\mathbf{w}) \tag{10}$$

We can now multiply this out an collect the appropriate terms in order to "complete the square".

$$\overbrace{\frac{-1}{2\tau^2}\mathbf{w}^T\mathbf{w}-\frac{1}{2\sigma^2}(\mathbf{X}\mathbf{w})^T(\mathbf{X}\mathbf{w})}^{\text{Quad}}+\overbrace{\frac{1}{\tau^2}\mathbf{w}^T\mathbf{w}_0+\frac{1}{\sigma^2}\mathbf{w}^T\mathbf{X}^T\mathbf{t}}^{\text{Lin}}-\overbrace{\frac{1}{2\tau^2}\mathbf{w}_0^T\mathbf{w}_0-\frac{1}{2\sigma^2}\mathbf{t}^T\mathbf{t}}^{\text{Const}} \tag{11}$$

Now we can equalize terms in order to find $\mathbf{S}_N$ and $\mathbf{w}_N$. First we can equalize the quadratic terms.

$$\begin{aligned}
-\frac{1}{2}\mathbf{w}^T\mathbf{S}_N^{-1}\mathbf{w} &= \frac{-1}{2\tau^2}\mathbf{w}^T\mathbf{w}-\frac{1}{2\sigma^2}\mathbf{w}^T\mathbf{X}^T\mathbf{X}\mathbf{w}) \\
&= \frac{-1}{2}\mathbf{w}^T\frac{1}{\tau^2}\mathbf{w}-\frac{1}{2}\mathbf{w}^T(\frac{1}{\sigma^2}\mathbf{X}^T\mathbf{X})\mathbf{w} \\
&= \frac{-1}{2}\mathbf{w}^T(\frac{1}{\tau^2}+\frac{1}{\sigma^2}\mathbf{X}^T\mathbf{X})\mathbf{w} \\
\mathbf{S}_N^{-1} &= (\frac{1}{\tau^2}+\frac{1}{\sigma^2}\mathbf{X}^T\mathbf{X})
\end{aligned}$$

Similarly, we can equalize the linear terms.

$$\begin{aligned}
\mathbf{w}^T\mathbf{S}_N^{-1}\mathbf{w}_N &= \frac{1}{\tau^2}\mathbf{w}^T\mathbf{w}_0+\frac{1}{\sigma^2}\mathbf{w}^T\mathbf{X}^T\mathbf{t} \\
\mathbf{w}_N &= \mathbf{S}_N(\frac{1}{\sigma^2}\mathbf{X}^T\mathbf{t}+\frac{1}{\tau^2}\mathbf{w}_0)
\end{aligned}$$

We have therefore derived the posterior distribution for one of the D independent regression problems,

$$p(\mathbf{w}|\mathbf{t},\mathbf{X})=\mathcal{N}([\frac{1}{\tau^2}+\frac{1}{\sigma^2}\mathbf{X}^T\mathbf{X}]^{-1}[\frac{1}{\sigma^2}\mathbf{X}^T\mathbf{t}+\frac{1}{\tau^2}\mathbf{w}_0],[\frac{1}{\tau^2}+\frac{1}{\sigma^2}\mathbf{X}^T\mathbf{X}]^{-1}) \tag{12}$$

We note that our D posterior estimates for each column of $\mathbf{W}$ each have the same variance. We can therefore define the complete posterior distribution across all dimensions as the following matrix normal distribution:

$$p(\mathbf{W}|\mathbf{T},\mathbf{X})\sim\mathcal{MN}_{QXD}(\mathbf{W}_N,\mathbf{S}_N,\mathbf{I}), \tag{13}$$

where each column i of $\mathbf{W}_N$ is equal to the corresponding mean vector $\mathbf{w}_{N,i}$, $\mathbf{S}_N$ is the QXQ covariance matrix defined above, and $\mathbf{I}$ is the DXD identity matrix, indicating that the columns of are independent.

1.1.2 Non-parametric Regression

There is not just a great deal of uncertainty surrounding the parameters of the mapping, but also in the mappings themselves. As a result, we should formulate our uncertainty in the mapping via a prior over mappings and then use Bayes' rule to reach the posterior. Gaussian Processes can be used to represent prior distributions over the space of functions. Here, the linear assumption in the mapping used in the regression problem will be replaced by by a non-parametric prior over the space of functions. The same assumptions as in the linear case are made,

$$\mathbf{t}_i = f(\mathbf{x}_i) + \boldsymbol{\epsilon}.$$

The output of the function f can be defined as its own variable,

$$\mathbf{t}_i = f_i + \boldsymbol{\epsilon},$$

where $\mathbf{f}_i$ is the output of the function at the input $\mathbf{x}_i$. If we formulate the prior over the output of a function using a Gaussian Process we have,

$$p(f|\mathbf{X}, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{0}, k(\mathbf{X}, \mathbf{X}))$$

where $k(\cdot, \cdot)$ is the covariance function and $\boldsymbol{\theta}$ are its parameters (the *hyper-parameters*). Here we assume the data have zero-mean and as such we do not require a mean function in the prior.

Question 6: *Explain what this prior does? Why is it a sensible choice? Use images to show your reasoning. Clue: use the marginal distribution to explain the prior*

We want our data to influence our choice of function. Since the mean is zero, the relationships in the data are entirely encapsulated by the covariance. This allows us to describe the probability distribution over functions explicitly. This is represented by a kernel function which, for "similar" input values would be greater, thereby indicating more correlation in the function evaluated at those points. This is sensible in that we will have higher confidence for closer, more "correlated" data points and a lower confidence for far away points. This prior also allows us to work in a finite space despite the fact that we are considering a continuous range of functions by only considering the value of the functions at the set of input values. This prior is powerful in that it can model a wide range of functions (see fig. 2), thereby modeling our uncertainty in the form of the function.

Given this formulation we can now formulate the full model.

Question 7: *Formulate the joint likelihood of the full model that you have defined above,*

$$p(\mathbf{T}, \mathbf{X}, f, \boldsymbol{\theta})$$

(Try to draw a very simple graphical model to clearly show the assumptions that you have made.)

The joint likelihood can be given by the following equation and the graphical model below:

$$p(\mathbf{T}, \mathbf{X}, f, \boldsymbol{\theta}) = p(\mathbf{T}|f)p(f|\mathbf{X}, \boldsymbol{\theta})p(\mathbf{X})p(\boldsymbol{\theta}) \tag{14}$$

Here we have modeled the relationship between $\mathbf{T}$ and f and also between f and $\mathbf{X}$ when in reality we are concerned in the relationship between $\mathbf{T}$ and $\mathbf{X}$. We now have the possibility to model uncertainty in each of these stages: in our beliefs of the functions, and in how we believe the output of the function have generated the observed data. Since we are not interested in f, however, we should marginalize out this variable. This implies calculating a specific integral,

$$p(\mathbf{T}|\mathbf{X}, \boldsymbol{\theta}) = \int p(\mathbf{T}|f)p(f|\mathbf{X}, \boldsymbol{\theta})df \tag{15}$$

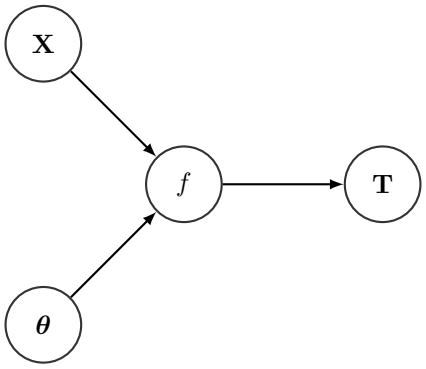


Figure 1: Graphical Model for Question 7.

Question 8: *Complete the marginalization formula in Eq.15 (general form) and discuss,*

- *Explain how this connects the prior and the data?*
- *How does the uncertainty "filter" through this?*
- *What does it imply that $\boldsymbol{\theta}$ is left on the left-hand side of the expression after marginalization?*

This marginalization uses the prior over functions as well as the likelihood of the data. In this manner we incorporate both our previous beliefs as well as new observations by taking the likelihood of the data over all possible functions. Our uncertainty in functions is represented via our prior over the mappings. In addition, we have uncertainty in the form of the likelihood; our output at a particular input value is due not only to the mapping but also noise. The fact that $\boldsymbol{\theta}$ remains on the left-hand side of the expression means that we have not marginalized it out and can still examine the impact of different hyperparameter values. This means we can optimize the posterior with respect to $\boldsymbol{\theta}$, leading to a type II maximum likelihood estimate.

1.2 Practical

1.2.1 Linear Regression

Here the linear regression discussed above is implemented. Both the prior and posterior over the the parameters $\mathbf{W}$ are examined. First some data were generated according to the relations given below.

$$
\begin{aligned}
t_i &= w_0 x_i + w_1 + \epsilon \\
\mathbf{x} &= [-2, -1.98, \ldots, 1.98, 2] \\
\epsilon &\sim \mathcal{N}(0, 0.2) \\
\mathbf{W} &= [1.5, -0.8]
\end{aligned}
$$

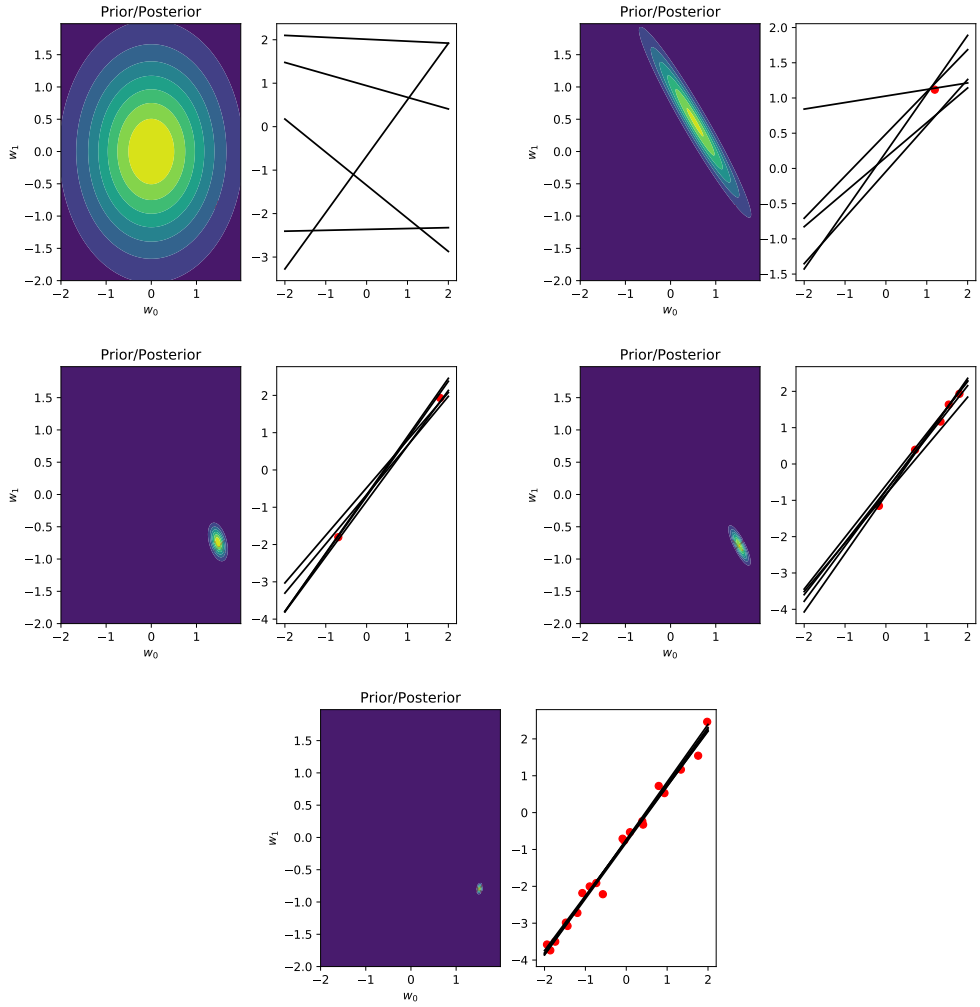


Figure 2: Sequential Bayesian Learning.

Question 9:

1. *Set the prior distribution over* $\mathbf{W}$ *and visualise it.*
2. *Pick a single data-point from the data and visualise the posterior distribution over* $\mathbf{W}$*.*
3. *Sample from the posterior and show a couple of functions.*
4. *Repeat 2–3 by adding additional data points.*

Describe the plots and the behavior when adding more data? Is this a desirable behavior? Provide an intuitive explanation.

Before we see any data, all of our previous beliefs are encoded via the prior. In this particular case the prior is very uninformative, and as a result when we sample from the prior we get a wide range of weights. Even after we pick only one point from the data, however, we already see the posterior distribution start to narrow. As a result, there is also a smaller range of possible functions when we sample from the posterior. We see how the posterior continues to narrow until it is essentially a delta function centered at the true parameter values after seeing only 20 points. Sampling from this posterior thus leads to a fairly uniform group of functions that fit the data points well. This behavior is exactly what is desired with probabilistic linear regression. Prior to seeing any data we have no idea what the possible weights could be and therefore we have a fairly

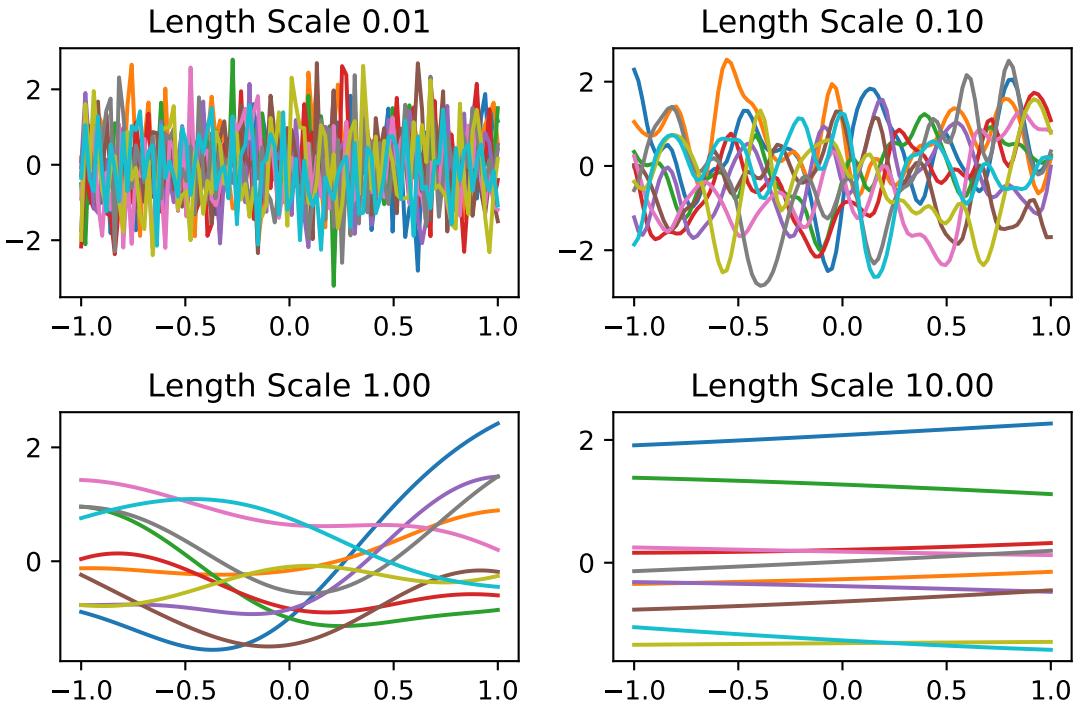


Figure 3: Sampling from a $\mathcal{GP}$-prior with different length-scales.

uninformative prior over the weights. As we begin to see more points, however, the data begins to overwhelm the prior and the posterior becomes constrained by the likelihood of the data. This process is therefore an example of a *consistent* estimator, meaning that it eventually recovers the true parameters that generated the data as the sample size goes to infinity.

1.2.2 Non-parametric Regression

Here the effect of a $\mathcal{GP}$ (Gaussian Process) prior was implemented and evaluated. A squared exponential covariance function was used:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \sigma_f^2 e^{-\frac{(\mathbf{x}_i - \mathbf{x}_j)^T(\mathbf{x}_i - \mathbf{x}_j)}{l^2}}$$

Question 10:

1. *Create a $\mathcal{GP}$-prior with a squared exponential co-variance function.*
2. *Sample from this prior and visualise the samples.*
3. *Show samples using different length-scales for the squared exponential.*

Explain the behavior of altering the length-scale of the covariance function.

As we can see, a smaller length scale leads to rather erratic samples. This makes intuitive sense; as the length scale is decreased the term in the exponential becomes much smaller. As a result, the value of the kernel function becomes much smaller, meaning that there is essentially zero correlation between nearby points. As the value of the covariance tends towards zero, we will eventually get white noise in which the variables are completely uncorrelated. We already start to see the beginnings of such behavior at a length scale of 0.01. As the length scale is increased, there is a stronger relationship between nearby points. The result is that at high length scales (such as 10 in the figure) even relatively "far away" points will still have a strong covariance.

In the next exercise highly non-linear data were generated,

$$y_i = cos(x_i) + \epsilon_i$$

$$\mathbf{x} = [0, \ldots, 2]^T$$

$$\epsilon_i \sim \mathcal{N}(0, 0.5)$$

We have seven observations of a noisy sine wave. Together with the prior we can use this data to examine the posterior distribution over functions.

Question 11:

1. *How do we interpret the posterior before we observe any data?*
2. *Compute the predictive posterior distribution of the model.*
3. *Sample from this posterior with points both close to the data and far away from the observed data.*
4. *Plot the data, the predictive mean and the predictive variance of the posterior from the data.*

Explain the behavior of the samples and compare the samples of the posterior with the ones from the prior. Is this behavior desirable? What would happen if you would add a diagonal covariance matrix to the squared exponential?.

Before any data is observed, the prior and the posterior are equivalent. In other words, all our knowledge about the problem is encapsulated in our previous beliefs via the prior. In figure 4 we see the result of sampling from the predictive posterior with points both close to and far away from the data. We are highly confident in the predictions close to and in between the data points. The posterior predictive mean essentially "connects the dots" of the data points. Outside of these data points, however, we are highly uncertain. Overall, this is not desirable behavior. We are not capturing the underlying structure of the data and instead over-fitting the noise in the observed data. In addition, we are too certain about our estimates within the observed data points. We would like our variance to reflect the fact that we are not completely certain about our estimates. In addition, we want to do a better job capturing the form of the data. Adding a diagonal covariance to the squared exponential as in figure 5 can help address some of these issues. Here we see that we are not simply following the noise in the data. The addition of the diagonal covariance also increases our uncertainty about the form of the function at points close to the observed data. This behavior is much more desirable. As we see more data points, we can expect our confidence in our predictive estimates to increase in addition to doing a better job of capturing the underlying structure of the data.

2 The Posterior $p(\mathbf{X}|\mathbf{Y})$

This task involves representation learning. We only observe the outputs $\mathbf{Y}$ and wish to deduce the inputs $\mathbf{X}$ that could represent $\mathbf{Y}$.

2.1 Theory

Here linear models are still considered; however, the input locations $\mathbf{X}$ are not known *a priori* and instead we wish to infer them from the data. These input locations are the *latent representations* of $\mathbf{Y}$. We begin with our linear model,

$$p(\mathbf{Y}, \mathbf{X}, \mathbf{W}) = p(\mathbf{Y}|\mathbf{X}, \mathbf{W})p(\mathbf{X})p(\mathbf{W})$$

We also specify a prior over the latent variables as a spherical Gaussian,

$$p(\mathbf{X}) = \mathcal{N}(0, \mathbf{I})$$

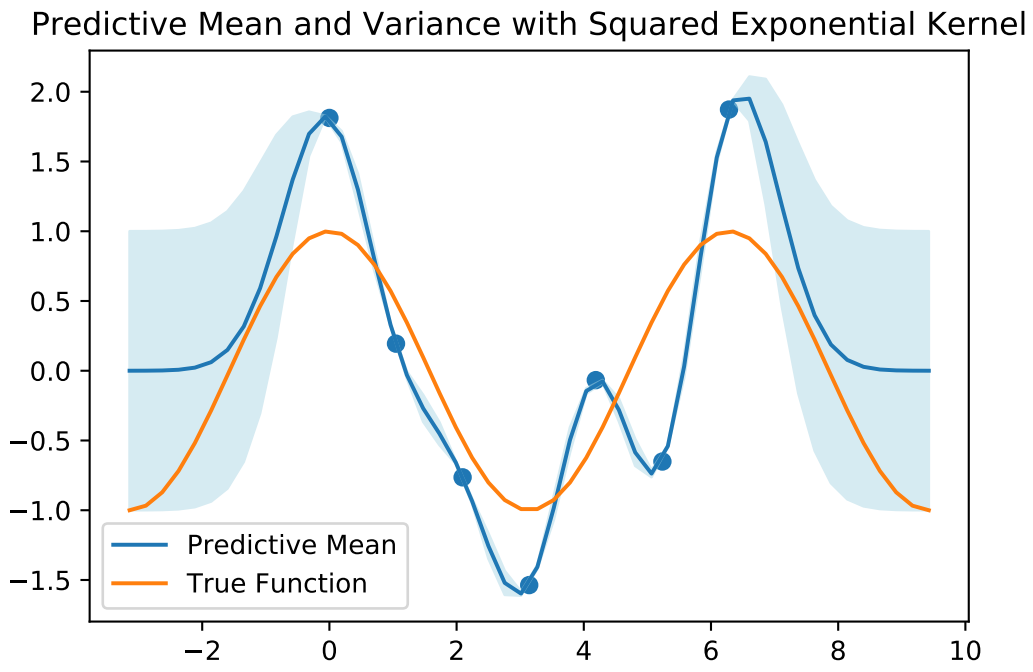


Figure 4: The predictive mean (solid blue line) and variance (light blue shading) with $l = 1$.

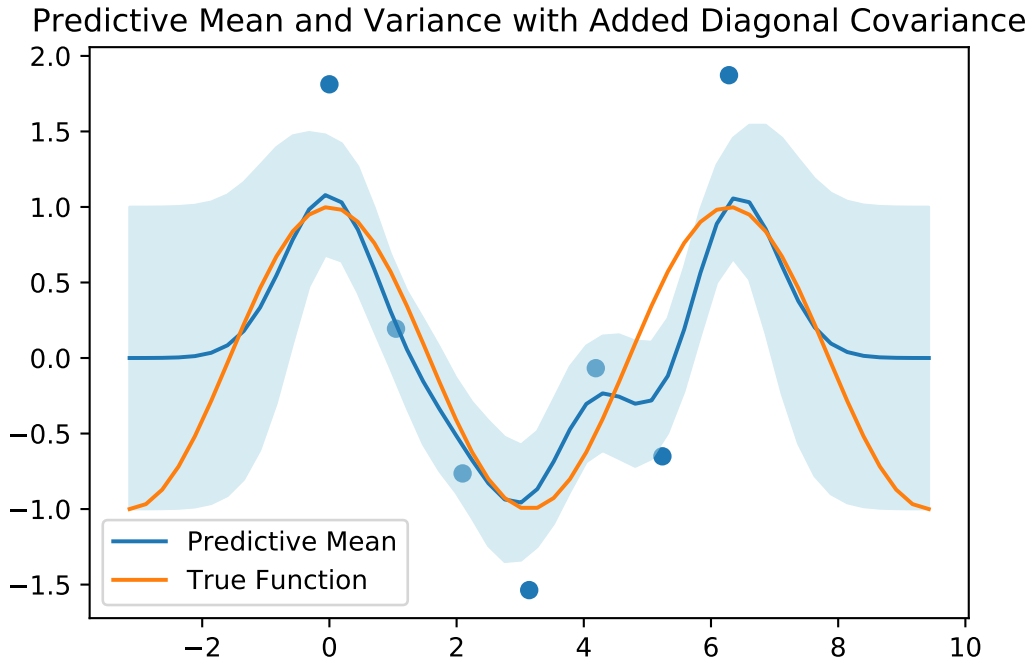


Figure 5: The predictive mean (solid blue line) and variance (light blue shading) with $l = 1$.

Question 12: *What type of "preference" does this prior encode?*

This prior essentially encodes zero "preference" for the latent variable. This actually makes sense because we have no reason to believe the latent variables have some bias. We are also assuming heterogeneity with independent dimensions. We want the generative variables to be as independent as possible. This is essentially the whole point of projecting our data into a lower-dimensional manifold. This prior can be combined with the likelihood and integrated over $\mathbf{X}$ in order to reach the marginal distribution,

$$p(\mathbf{Y}|\mathbf{W}) = \int p(\mathbf{Y}|\mathbf{W}, \mathbf{X})p(\mathbf{X})dx \tag{16}$$

Question 13: *Perform the marginalization in Eq. 16 and write down the expression. As previously, it is recommended that you do this by hand even though you only need to outline the calculations and show the approach that you would take to pass the assignment.*
Hint: *The marginal can be computed by integrating out* $\mathbf{X}$ *with the use of Gaussian algebra we exploited in the exercise derivations and, in particular, by completing the square. However it is much easier to derive the mean and covariance, knowing that the marginal is Gaussian, from the linear equation of* $\mathbf{Y}(\mathbf{X})$.

If we take advantage of the hint we can first write down the form of the linear equation for a single multidimensional data point $\mathbf{y}_i$,

$$\mathbf{y}_i = \mathbf{W}\mathbf{x}_i + \boldsymbol{\epsilon}$$

Where $\epsilon \sim \mathcal{N}(0, \sigma^2\mathbf{I})$. Recognizing that the predictive distribution will be Gaussian, the mean and covariance can be evaluated directly,

$$\begin{aligned}
\mathbb{E}[\mathbf{y}] &= \mathbb{E}[\mathbf{W}\mathbf{x} + \boldsymbol{\epsilon}] \\
\mathbb{E}[\mathbf{y}] &= \mathbf{W}\mathbb{E}[\mathbf{x}] + \mathbb{E}[\boldsymbol{\epsilon}] \\
\mathbb{E}[\mathbf{y}] &= 0
\end{aligned}$$

$$\begin{aligned}
cov[\mathbf{y}] &= \mathbb{E}[(\mathbf{W}\mathbf{x} + \boldsymbol{\epsilon})(\mathbf{W}\mathbf{x} + \boldsymbol{\epsilon})^T] \\
cov[\mathbf{y}] &= \mathbb{E}[(\mathbf{W}\mathbf{x}\mathbf{x}^T\mathbf{W}^T)] + \mathbb{E}[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T] \\
cov[\mathbf{y}] &= \mathbf{W}\mathbf{W}^T + \sigma^2\mathbf{I}
\end{aligned}$$

Here we took advantage of the fact that $\mathbf{x}$ and $\boldsymbol{\epsilon}$ are independent and therefore uncorrelated. Additionally, we note that we already know the covariance of each of these variables. We can then generalize from one data point to the entire dataset. We have N independent D-dimensional random variables $\mathbf{y}$ which are identically distributed. We can combine them into the complete matrix normal distribution,

$$p(\mathbf{Y}|\mathbf{W}) = \mathcal{MN}_{NXD}(\mathbf{0}, \mathbf{I}_{NXN}, \mathbf{W}\mathbf{W}^T + \sigma^2\mathbf{I}) \tag{17}$$

2.1.1 Learning

We will now compare three methods for learning the model parameters: maximum likelihood (ML),

$$\hat{\mathbf{W}} = argmax_{\mathbf{W}} p(\mathbf{Y}|\mathbf{X}, \mathbf{W}) \tag{18}$$

maximum a posteriori (MAP),

$$\hat{\mathbf{W}} = argmax_{\mathbf{W}} \frac{p(\mathbf{Y}|\mathbf{X}, \mathbf{W})p(\mathbf{W})}{\int p(\mathbf{Y}|\mathbf{X}, \mathbf{W})p(\mathbf{W})d\mathbf{W}} = argmax_{\mathbf{W}} p(\mathbf{Y}|\mathbf{X}, \mathbf{W})p(\mathbf{W}) \tag{19}$$

and Type-II Maximum-Likelihood,

$$\hat{\mathbf{W}} = argmax_{\mathbf{W}} \int p(\mathbf{Y}|\mathbf{X}, \mathbf{W})p(\mathbf{X})d\mathbf{X}. \tag{20}$$

Question 14: *Compare these estimation procedures in log-space.*

- *How are they different?*
- *How are MAP and ML different when we observe more data?*
- *Why are the last two expressions of eq. 14 equal?*

.

In log-space the MAP and ML estimates are essentially the same with the exception of the additional regularization term $\mathbf{W}^T\mathbf{W}$ in the MAP estimate that comes about via the inclusion of a prior. This regularization term can help to reduce over-fitting since we consider the prior over parameters and do not simply maximize the likelihood. As we observe more data, $\mathbf{W}_{MAP}$ will become more similar to $\mathbf{W}_{ML}$ as the data begins to overwhelm the prior. In the limit of infinite data points, the two estimations will be the same. The last two expressions of equation 19 are equal because the denominator is the normalization factor that represents the evidence, which would be equivalent for all possible sets of parameter values. Therefore, if we wish to find the parameter values that maximize the expression, we need only consider the numerator. Type II maximum likelihood takes a slightly different approach. Here we marginalize over X and then optimize the likelihood with respect to $\mathbf{W}$ directly as in the normal ML estimate. In this manner, the model complexity can be optimized without the use of an independent training set as in ML and MAP estimates.

The representation task involves two variables $\mathbf{W}$ and $\mathbf{X}$ that interact and is therefore appropriate for Type-II ML estimation.

Question 15:

1. *Write down the objective function $-log(p(\mathbf{Y}|\mathbf{W})) = \mathcal{L}(\mathbf{W})$ for the marginal distribution in Eq. 13.*
2. *Write down the gradients of the objective with respect to the parameters $\frac{d\mathcal{L}}{d\mathbf{W}}$.*

The objective function (negative log likelihood) can be written as,

$$\mathcal{L}(\mathbf{W}) = \frac{N}{2}\{Dln(2\pi) + ln|\mathbf{C}| + Tr(\mathbf{C}^{-1}\mathbf{S})\} \tag{21}$$

where $\mathbf{C} = \mathbf{W}\mathbf{W}^T + \sigma^2\mathbf{I}$ and $\mathbf{S}$ is the data covariance matrix. In order to evaluate the gradients of the objective we can use some matrix identities from the *Matrix Cookbook*,

$$\frac{dTr(\mathbf{X}^{-1}\mathbf{A})}{d\mathbf{X}} = -\mathbf{X}^{-1}\mathbf{A}\mathbf{X}^{-1}$$
$$\frac{dln|\mathbf{X}|}{d\mathbf{X}} = \mathbf{X}^{-T}$$

Using these two identities we have,

$$\begin{aligned}\mathcal{L}(\mathbf{W}) &= \frac{N}{2}\{\mathbf{C}^{-1}2\mathbf{W} - \mathbf{C}^{-1}\mathbf{S}\mathbf{C}^{-1}2\mathbf{W}\} \\ &= N\{\mathbf{C}^{-1}\mathbf{W} - \mathbf{C}^{-1}\mathbf{S}\mathbf{C}^{-1}\mathbf{W}\}\end{aligned}$$

2.2 Practical

Here we have generated a dataset $\mathbf{Y} \in \mathbb{R}^{100X10}$ and we wish to recover the generating parameter. The dataset was generated according to the following relations:

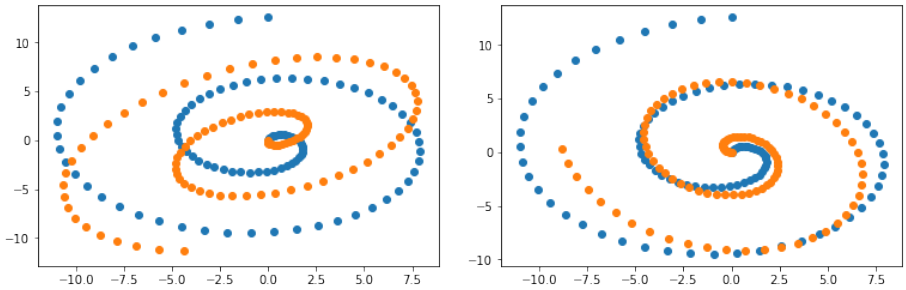


Figure 6: Two learned representations of $\mathbf{x}'$ using different initial guesses for $\mathbf{A}$. **Blue**: True $\mathbf{x}'$, **Orange**: Learned Representation

$$
\begin{aligned}
\mathbf{Y} &= f_{lin}(f_{non-lin}(\mathbf{x})) \\
\mathbf{x} &= [0, ..., 4\pi]^T \\
|\mathbf{x}| &= 100 \\
f_{non-lin}(x_i) &= [x_i sin(x_i), x_i cos(x_i)] \\
f_{lin}(x') &= \mathbf{x}'\mathbf{A}^T \\
\mathbf{A} &= \mathbb{R}^{10x2} \\
\mathbf{A}_{ij} &\sim \mathcal{N}(0, 1)
\end{aligned}
$$

2.2.1 Linear Representation Learning

Question 16: *Plot the representation you have learned (hint: plot* $\mathbf{X}$ *as a two dimensional representation). Explain the outcome and discuss the key features, elaborate on any invariance you observe. Did you expect this result?*

As we can see above, the representation we have learned using a probabilistic PCA does capture the general shape of the hidden variable, though the exact orientation is highly dependent on our initial guesses for the linear transformation matrix $\mathbf{A}$. These results are to be expected. It is perhaps not particularly impressive that we have captured the general shape of the latent variable given that we used a $10X2$ matrix as the initial guess for $\mathbf{A}$, so we already knew the dimensions of the latent variable space. It is also to be expected that different initial guesses for the transformation matrix led to rotations in the latent space coordinates. This can be easily seen through the following example, which is taken from *Bishop (p.573)*. If we consider a matrix $\tilde{\mathbf{A}} = \mathbf{AR}$ where $\mathbf{R}$ is an orthogonal matrix representing a rotation matrix in the latent space, then $\mathbf{RR}^T = \mathbf{I}$ and the quantity $\tilde{\mathbf{A}}\tilde{\mathbf{A}}^T = \mathbf{AA}^T$. Therefore, the covariance matrix $\mathbf{C} = \mathbf{AA}^T + \sigma^2\mathbf{I}$ of the observed variable is independent of $\mathbf{R}$ and there are many different matrices $\tilde{\mathbf{A}}$ that would give the same predictive density.

3 The Evidence $p(\mathcal{D})$

3.1 Theory

3.1.1 Data

Here we consider a simple data domain $\mathcal{D} = {t^i}_{i=1}^9$ where $t^i \in -1, 1$. The data is structured according to a grid whose locations can be parameterized by $\mathcal{X} = {\mathbf{x}^i}_{i=1}^9$ where $\mathbf{x}^i = (-1, 0, 1, -1, 0, 1)$. The data domain $\mathcal{D}$ therefore contains 512 different datasets.

3.1.2 Models

The simplest model spreads its probability mass uniformly across the entire data space,

$$p(\mathcal{D}|M_0, \boldsymbol{\theta}_0) = \frac{1}{512}$$

Question 17: *Why is this the simplest model, and what does it actually imply? Discuss its implications, why is this a bad model and why is it a good model?*

This is the "simplest" model because there are no free parameters. However, I do not necessarily think "simple" is the best way to describe the model—some other word such as "naive" may be more appropriate. This is because in a certain sense this is a complex model in that it assigns probability to many different types of datasets. In other words, it spreads its probability mass over all possible datasets. This is both the strength and weakness of this model; it can model a wide range of datasets though it assigns very little probability to simple data sets.

We can also define a model with parameters which can factorize into 9 separate models if we assume all t^n are independent.

$$p(\mathcal{D}|M_1, \boldsymbol{\theta}_1) = \prod_{n=1}^{9} \frac{1}{1 + e^{-t^n \theta_1^1 x_1^n}}.$$

Question 18: *Explain how each separate model works. In what way is this model more or less flexible compared to M_0? How does this model spread its probability mass over $\mathcal{D}$?*

These models relate the value t^n to its location, and in particular, its location only along one dimension; it gives the probability of the output value at a particular value of x_1^n. This model is more flexible in that it has a free parameter, but it is incapable of modeling more complex datasets or even simple data sets with a decision boundary along the other dimension. This model therefore concentrates its probability mass over a particular subset of simple models in $\mathcal{D}$.

We can also add parameters and create further models,

$$p(\mathcal{D}|M_2, \boldsymbol{\theta}_2) = \prod_{n=1}^{9} \frac{1}{1 + e^{-t^n (\theta_2^1 x_1^n + \theta_2^2 x_2^n)}}$$

$$p(\mathcal{D}|M_3, \boldsymbol{\theta}_3) = \prod_{n=1}^{9} \frac{1}{1 + e^{-t^n (\theta_2^1 x_1^n + \theta_2^2 x_2^n) + \theta_3^3}}$$

Question 19: *How have the choices we made above restricted the distribution of the model? What datasets are each model suited to model? What does this actually imply in terms of uncertainty? In what way are the different models more flexible and in what way are they more restrictive? Discuss and compare the models to each other.*

M_1 is the simplest model in that it can only model decision boundaries along one dimension. M_2 is more complex; it can model decision boundaries in either dimension. Finally, M_3 incorporates a bias term, which allows decision boundaries to be offset from the origin. M_3 can be seen as encompassing M_1 and M_2 and it therefore spreads its probability mass over a wider range of datasets. M_0 spreads its probability mass the most and is therefore best suited for datasets without a sharp linear boundary since none of the other models are able to capture this type of behavior. M_3 and M_0 are penalized for spreading their probability mass, however, and will not be chosen for simpler datasets. In effect, we are more certain when we choose M_1 and to a lesser extent M_2 because their probability mass is so concentrated that the evidence is likely much higher for datasets that can be described by either of these models. On the other hand, we are always relatively uncertain when choosing M_0 because the evidence for any dataset is always very small.

3.1.3 Evidence

The evidence of a given model M_i is represented via the distribution $p(\mathcal{D}|M_i)$. We can marginalize out the parameters of the model in order to reach the evidence of a given model.

$$p(\mathcal{D}|M_i) = \int_{\forall \boldsymbol{\theta}} p(\mathcal{D}|M_i, \boldsymbol{\theta}) p(\boldsymbol{\theta}) d\boldsymbol{\theta} \tag{22}$$

Question 20: *Explain the process of marginalization and briefly discuss its implications*

The process of marginalization allows us to compare models across all possible parameter values. To some extent it also allows us to avoid the over-fitting associated with classical maximum likelihood estimates, allowing us to directly compare models on training sets. Marginalization and Bayesian model comparison also allow us to use probabilities to represent uncertainty in the choice of model. Essentially marginalization lets us view the evidence as the probability of generating the data set $\mathcal{D}$ from a model whose parameters are sampled at random from the prior, thus also incorporating our prior beliefs about the parameters values, which are encoded via $p(\boldsymbol{\theta})$.

This marginalization requires a *prior* over the parameters. In this task we will use a simple spherical Gaussian with zero mean and variance values of $\sigma^2 = 10^3$.

Question 21: *What does this choice of prior imply? How does the choice of the parameters of the prior $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ affect the model?*

This choice of prior indicates that we do not know of any bias in the parameter values, since they have zero mean. Additionally, the parameter values have equal and uncorrelated variances. Finally, the very high variance will make larger parameter values more common and will therefore encourage sharper decision boundaries. Different $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ values would make certain decision boundaries more or less likely.

Instead of solving the integral from equation 21 directly, a Monte Carlo approach will be used to approximate the marginalization.

$$p(\mathcal{D}|M_i) \approx \frac{1}{S} \sum_{s=1}^{S} p(\mathcal{D}|M_i, \boldsymbol{\theta}^s),$$
$$\boldsymbol{\theta}^s \sim p(\boldsymbol{\theta}|M_i)$$

where s indexes the samples from the prior of the parameters.

3.2 Practical

Question 22: *Plot the evidence over the whole dataset for each model (and sum the evidence for the whole of $\mathcal{D}$, explain the numbers you get). The **x-axis** index the different instances in $\mathcal{D}$ and each model's evidence is on the **y-axis**. How do you interpret this? Relate this to the parametrization of each model.*

In figure 6 the datasets have been ordered from most to least total evidence across the models. As we see from the first figure, all of the models concentrate much of their probability mass over a relatively small subset of datasets. This is to be expected as the majority of datasets do not have very sharp linear boundaries, which are favored by all of the models with the exception of M_0. If we zoom in to the 50 datasets with the most combined evidence, we see that M_3 spreads its probability mass the most. This is to be expected since it is the most complex of the models other than M_0. This also means that there is relatively little evidence for very simple datasets, though it does assign some probability mass to datasets which have high evidence for models 1 and 2. Interestingly, however, it also has a very sharp peak likely corresponding to datasets with highly unequal distributions of target values, which can be accounted for with the bias term present in

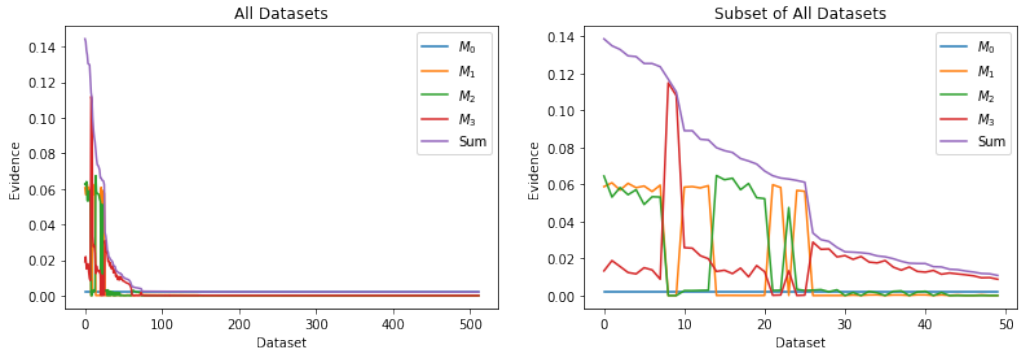


Figure 7: Evidence Over All Datasets for Each Model.

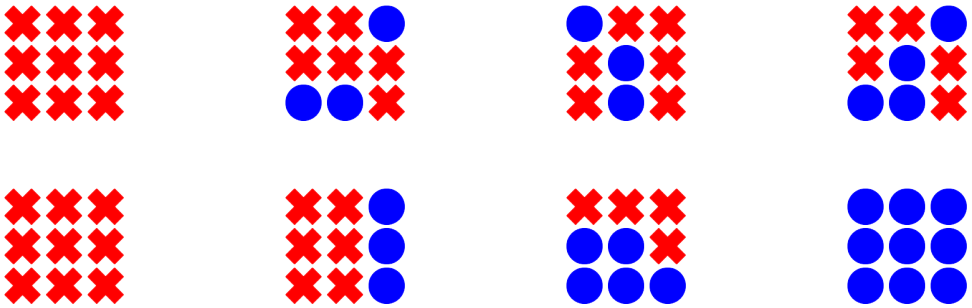


Figure 8: Highest and Lowest Evidence for Each Model. L–R M_0, M_1, M_2, M_3. Top Row: Lowest Evidence. Bottom Row: Most Evidence.

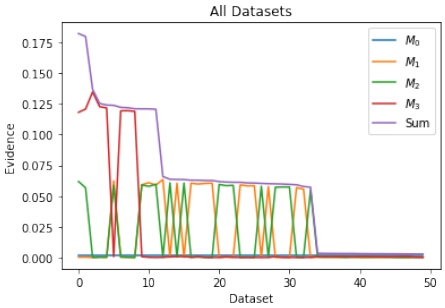
M_3. Both models M_2 and M_1 concentrate their probability mass more over about 20 datasets, with M_2 being slightly more spread out, which is to be expected since it can model decision boundaries in two dimensions as opposed to one.

Question 23: *Find using np.argmax and np.argmin which part of the $\mathcal{D}$ that is given most and least probability mass by each model. Plot the data-sets which are given the highest and lowest evidence for each model. Discuss these results, does it make sense?*

Figure 7 shows datasets with the highest and lowest evidence for each model. The leftmost plots are for M_0 and are therefore trivial since every dataset has equal evidence. The top row is the lowest evidence for each dataset. These all make sense since they have clearly nonlinear decision boundaries that cannot be well-modeled by any of the other three models. The highest evidence plots for each model on the bottom row also make sense. The dataset with the highest evidence for M_1 has a decision boundary along only the x_1 dimension. Since M_1 concentrates much of its probability mass for such models it makes sense that it would give this dataset high probability. The highest evidence dataset for M_2 has a decision boundary along the x_2 dimension. Such datasets are not well modeled by M_1, whereas M_2 is well suited to model such simple decision boundaries. Finally, M_3 is good at modeling highly unequal distributions of target values because of the bias parameter, so it makes sense that this dataset would be assigned a high probability.

Question 24: *What is the effect of the prior $p(\boldsymbol{\theta})$.*

- *What happens if we change its parameters?*
- *What happens if we use a non-diagonal covariance matrix for the prior?*
- *Alter the prior to have a non-zero mean, such that $\mu = [5, 5]^T$?*
- *Redo evidence plot for these and explain the changes compared to using zero-mean.*



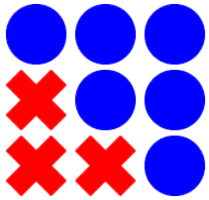


Figure 9: (L) Evidence with non-diagonal covariance. (R) Highest evidence dataset for model 2

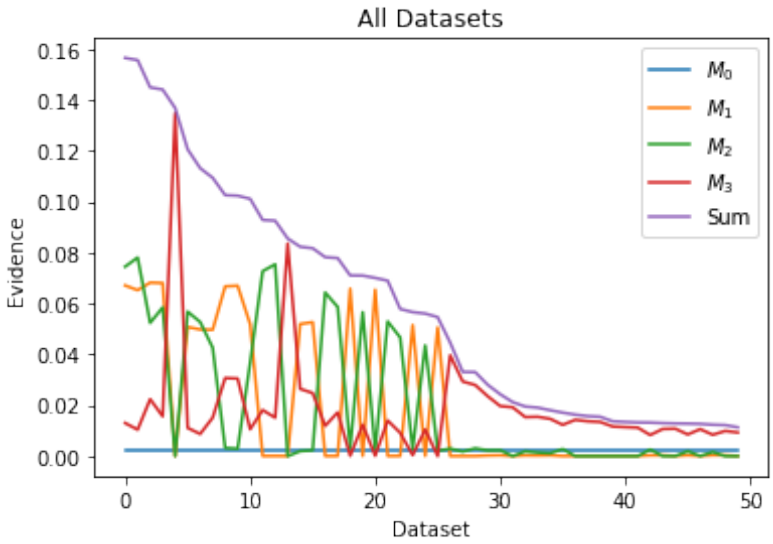


Figure 10: Evidence Plot for prior with nonzero mean

Changing the hyperparameters in the prior affect the probability of getting certain parameter values for each model. Smaller covariance values would not favor sharp linear boundaries as much in the examples above. If we use a non-diagonal covariance matrix such that all of the parameters have high covariance we see that certain datasets would become more probable. In model 2, for example, parameter values that were highly correlated would favor datasets with a diagonal decision boundary. Indeed, the highest evidence dataset for model 2 had a diagonal decision boundary and is shown in figure 8. In plotting the evidence over a subset of datasets as in figure 6, we also see that each model favors very particular datasets with almost equal probability. Interestingly, model 3 is also less capable of modeling a wide variety of datasets and no longer spreads its probability mass. If, for example, all the parameter values are very positive (including the bias term), only datasets with a large number of positive target values would be favored in order to minimize the term in the exponential and therefore maximize the overall evidence.

If we change the mean for the prior but maintain a diagonal covariance, we see in figure 9 that certain datasets become very favored, resulting in an evidence plot that has many sharp "peaks". However since the shift in the mean is relatively small compared to the still very large variance, we see that M_3 still spreads its probability mass over a wide range of datasets and, in general, the evidence plot looks a little more like the original evidence plot than when non-diagonal covariance was introduced.

References

[1] C. M. Bishop. *Pattern Recognition and Machine Learning*, volume 53. 2013.

[2] N. Lawrence. Probabilistic non-linear Principal Component Analysis with Gaussian Process Latent Variable Models. *Journal ofMachine Learning Research*, 6:1783–1816, 2005.

[3] K. P. Murphy. *Machine Learning: A Probabilistic Perspective*. 2012.

[4] K. B. Petersen and M. S. Pedersen. The Matrix Cookbook. *Citeseer*, 16(4):1–66, 2007.

[5] M. E. Tipping and C. M. Bishop. Probabilistic Principal Component Analysis. 61(3):611–622, 1999.