Regression

Gaussian process is a distribution over functions where inference takes place in the space of functions (*function-space view*).

2.1 Weight-space View

Linear functions have been well studied. They are easy to interpret and implement. But, it's often times, too simple. One enhancement is to project inputs into high dimensional space where linear separation might be feasible. Applying calculations on features in higher dimensional space is called the *kernel trick*. A n-sized dataset is available, the input having size D, and y being a scalar. The entire input is aggregated into a $D \times N$ design matrix $\mathbf{X}$, and labels are collected into a vector $\mathbf{y}$.

2.1.1 The Standard Linear Model

We define a linear regression model with Gaussian noise as $f(x) = \mathbf{x}^T\mathbf{w}$ and $y = f(x) + \varepsilon$ A bias is included, but it will be integrated into the input vector as 1 and first weight in $\mathbf{w}$. We model the noise using a Gaussian: $\varepsilon \sim \mathcal{N}(0, \sigma_n^2)$ with zero mean and some variance. Combining the noise and the model, one ends up with likelihood, probability density of the output given the input and parameters.

$$p(y|X,w) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma_n)}} exp(-\frac{(y_i - x_i^T w)^2}{2\sigma_n^2}) p(y|X,w) \sim \mathcal{N}(X^T w, \sigma^2)$$

We need a prior over the weights, and we will use a Gaussian $w \sim \mathcal{N}(0, \Sigma_p)$

- Q: what's the dimension of weights?

In the Bayesian setup, inference is derived as posterior = prior times likelihood normalized with the marginal likelihood. Now, taking in the prior and combining it with the aforecalculated likelihood, we end up with:

$$p(w|X,y) = exp(\frac{1}{2}(w - \hat{w})^T(\frac{1}{\sigma^2}XX^T + \Sigma_p^{-1})(w - \hat(w))$$

$$p(w|X,y) \sim \mathcal{N}(\frac{1}{\sigma^2}A^{-1}Xy, A^{-1})$$

This estimation is called the maximum a posteriori (MAP) estimation. To make a prediction, we average over all possible parameter values, weighted by their posterior probability

$$p(f_*|x_*, X, y) = \int p(f_*|x_*, w)p(w|X, y)$$

$$p(f_*|x_*, X, y) = \mathcal{N}(\frac{1}{\sigma^2} x_*^T A^{-1} X y, x_*^T A^{-1} x_*)$$

- TODO: how to end up with this multiplication under a normal distribution.

Look at figure 2.1 The intercept should be a little smaller than 0 and slope around 1. THe weights go from uniform to the ones in picture d). Likelihood shapes them a lot (much more than the prior).

- TODO: what is c) exactly showing, since that can't be a distribution of y?

2.1.2 Projections of Inputs into Feature Space

We can project an input space into another using some transformation function. We perform polynomial regression by transforming using $\phi(x) = (1, x, x^2, ..)$. The projections **must be fixed functions independent of w**. We define a function $\phi(x)$ which maps from D-dimensional space to N dimensional feature space.

Now, we wish to make predictions over the transformation of x. And then the predictive distribution becomes:

$$f_*|x_*, X, y = \mathcal{N}(\frac{1}{\sigma^2} \phi(x_*)^T A^{-1} \Phi(X) y, \phi(x)_*^T A^{-1} \phi(x_*))$$

To make predictions, it is required to invert matrix A, which might be inconvinient in practice due to a large feature space, so the equation can be rewritten as using K and Σ. In this cse one needs to convert matrices of sample size n instead of feature space dimension N. Notice the occurence of the form $\phi(x)^T \Sigma_p \phi(x')$ appears often in eq (2.12). Thus, we define $k(x, x')$ of this form and name it covariance function or kernel. The kernel is an inner product with respect to a covariance Σ. To further simplify, Σ_p can be decomposed as it is positively definite (every scalar obtained with inner product is strictly positive).

If an algorithm is defined in terms of inner products in input space, it can be lifted in feature space by replacing inner products with $k(x, x')$, which is called the *kernel trick*.

2.2 Function-space View

Now, an alternative way of reaching identical results as the previous section, but by considering inference in the function space. We define a *Gaussian process* as a collection of random variables, which have a join Gaussian distribution. The process is specified by its mean and covariance. The random variables represent the value of the function f(x) at point x. Often GPs are defined over time, but we do not make this a requirement, but the input space is generalized to $\mathbb{R}^D$.

An example of a Gaussian process can come from linear models. Linear models are defined as $f(x) = \phi(x)^T w$ with prior $w \sim \mathcal{N}(0, \Sigma_p)$.