0.1 Introduction

The question of how computers can change future careers have been a popular issue recently. With the fast development of automation technologies, more jobs have become purely automated. Many people argued that the increase in unemployment rate is partly because this. Would it be true that more people will become jobless? Or is it just a short term phenomenon that does not represent future trends? This report represent how this problem is tackled in the respect of machine learning techniques. In particular, the property of jobs are analysed through Gaussian Process and the possibility of automation of the job can be predicted.

0.2 Gaussian Process Regression

0.2.1 Covariance Function

The covariance function describes how observations are related to each other.

$$k(x, x\prime) = \sigma_f^2 exp[\frac{-(x - x\prime)}{2l^2}]$$

0.2.2 Hyperparameters

0.3 Gaussian Process Classification

From the last chapter, we know that regression involves continuous observation values. However, sometimes the output data is not continuous but a series of labels, where we wish to determine the class labels according to input x. This is the problem of classification. Similar ideas are adopted, except here regression is applied on a latent variable f and then the probability of the class label is determined by 'squashing' f into range [0,1].

0.3.1 'Squashing' Function

The 'squashing' function can be any sigmoid[explanation] funciton. Two typical sigmoid functions are logistic function $\lambda(f)$ and cumulative Gaussian function $\phi(f)$.

Inference is hence devided into two steps. First is to compute the distribution of the prediction latent variable f_* in terms of previous observations

$$p(\boldsymbol{f}_*|\boldsymbol{X},\boldsymbol{y},\boldsymbol{x}_*) = \int p(\boldsymbol{f}_*|\boldsymbol{X},\boldsymbol{x}_*,\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{X},\boldsymbol{y})\ d\boldsymbol{f}$$

where $p(\boldsymbol{f}|\boldsymbol{X},\boldsymbol{y}) = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{X})/p(\boldsymbol{y}|\boldsymbol{X})$ is the posterior of latent variables $\boldsymbol{f}$, which is estimated by MAP with respect to hyperpamaters(l and σ_f).

After obtaining the information about predictive latent variable, the probabilistic prediction is estimated by

$$\bar{\pi}_* = p(y_* = +1|X,y,x_*) = \int \sigma(f_*)p(f_*|X,y,x_*)df_*$$

This is called the averaged probility. One may argue that expectation of the prediction should be simply equal to the sigmoidal mean of f_*. These are actually two different things because of the non-linearity of sigmoid function, the first is $E[\pi_*|X,y,x_*]$ while the later one is $\sigma(E[f_*|y])$. However, the numerical values of these two are the same for binary classifications.

0.3.2 The Laplace Approximation

The integral for computing $p(f_*|X,y,x_*)$ is not analytically tractable. By doing a second order Taylor expansion of log$p(f|X,y)$ around its maximum point, we could obtain a Gaussian approximation of the posterior $q(f|X,y)$.

$$p(f_*|, X, y) = \mathcal{N}(f|\hat{f}, A^{-1}) \propto exp(-\frac{1}{2}(f - \hat{f})^T A(f - \hat{f}))$$

where $\hat{f} = argmax_f p(f|X, y)$ and $A = - \bigtriangledown \bigtriangledown logp(f|X, y)|_{f=\hat{f}}$

0.3.3 Estimating the Model

0.3.4 Probabilistic Prediction

0.3.5 Mathematical Implementation

Optimised latent variable f

The best latent f should be estimated for each set of hyperparameters by solving

$$\hat{f} = K(\bigtriangledown logp(y|\hat{f}))$$

Commonly used convergence criteria depend on the difference between successive $\Psi(f)$, the magnitude of gradient vetor $\bigtriangledown \Psi(f)$ or the defference between successive values of f. Note in practice, the convergence of objective functio is assured by checking that each iteration gives an increase in $\Psi(f)$. If not, a smaller step change in f should be used.

Maximised Marginal Likelihood

Gradients of likelihood wrt. hyperparameters:

Predictive Class Probability

Adding jitter -

0.3.6 Multi-class Classification