COMP 652 - ECSE 608: Machine Learning - Assignment 2

Posted Wednesday, October 3, 2018
Due Wednesday, October 22 , 2018

---

1. [30 points] **Gaussian Processes**

   For this exercise, you can use either the Python Gaussian Process package from scikit-learn (`sklearn.gaussian_process`) or the GPML toolbox available in Matlab (we encourage you to use the scikit-learn Gaussian Process package). You can also use another programming language but then you need to check whether a GP toolbox is available.

   In order to get a clear idea of what the plots you have to do in this exercise should look like, have a look at the following pages:
   http://scikit-learn.org/stable/auto_examples/gaussian_process/plot_gpr_prior_posterior.html
   http://scikit-learn.org/0.17/auto_examples/gaussian_process/plot_gp_regression.html

   (a) [10 points] We consider the model $y = x \sin x$. Generate a training set of input-output examples (x_i, y_i) of size 50 drawn from this model, where the x_i are drawn uniformly between 0 and 10.

      i. Fit the data using Gaussian Process with a squared exponential covariance function with scale parameters $v_0 = 1.1$, using maximum likelihood estimation of the parameters. Use a starting point for the MLE estimation of the best set of hyperparameters as 10^{-1}, with a upper bound of 1 and a lower bound of 10^{-3} on the parameters

      For example, this can be done in scikit learn with

```
from sklearn.gaussian_process import GaussianProcessRegressor
from sklearn.gaussian_process.kernels import RBF

kernel = 1.1 * RBF(length_scale=0.1, length_scale_bounds=(1e-3, 1.0))
gp =  GaussianProcessRegressor(kernel=kernel)
gp.fit(X,y)
```

      where the matrix X and vector y are the inputs and outputs from the training set.

      Plot the mean and variance of the GP fit for x ranging from 0 to 15. You should also show the training instances on this plot. What does the variance tell you about the fit of the GP model? If you use a starting point for the length scale l of the RBF kernel to be 10.0, how does the fit of the GP model differ? Comment on the difference.

      ii. Instead of using a RBF kernel, now fit the GP model with a linear kernel (`DotProduct` in scikit learn). Plot the mean and variance of the GP fit. How does the fit of the GP model with this kernel compare to fitting it with an RBF kernel?

      iii. Again fit a GP model, this time with a rational quadratic kernel (`RationalQuadratic` in scikit learn) with initial length scale $l = 0.1$. Plot the mean and variance of the GP fit. Comment on the difference using this kernel, compared to using a RBF or linear kernel.

      iv. Instead of using a single kernel for the GP model fit, consider using a sum of kernels (in scikit learn, this can simply be done by using the addition operator, e.g. `kernel = DotProduct(...)  + RBF(...)`). Fit a GP model with a sum of RBF kernel and a RationalQuadratic kernel. How does the fit of the GP model with a sum of kernels differ, compared to using a single kernel? Plot the mean and variance of the GP model fit. Comment on the difference in your observations.

(b) [10 points] We again consider the model from the preceding question, but this time with noisy observations. Considering the model $y = x \sin x + \varepsilon$ where $\varepsilon \sim \mathcal{N}(0.0.5)$. Generate a training set of input-output examples (x_i, y_i) of size 50 drawn from this model, where the x_i are drawn uniformly between 0 and 10.

i. Fit a GP model for this noisy data using the kernels you considered above in questions $(c.i) - (c.iv)$, i.e. first fit the GP model with a RBF kernel, then with a DotProduct kernel, a RationalQuadratic kernel, and finally with a sum of RBF and RationalQuadratic kernel. How does the fit of the GP model with each of these kernels on a noisy data set differ, compared to what you observed above in (c)? Plot the mean and variance for each of the GP model fit.

ii. A White kernel is often used as part of a sum kernel to account for the noise component of the signal. Tuning the parameter of the White Kernel corresponds to estimating the noise level. For example, in scikit-learn, the White kernel is implemented in `sklearn.gaussian_process.kernels.WhiteKernel`.

For each kernel k you considered above in question $(d.i)$, use the sum of k and a White kernel. For example, for the RBF kernel, now consider a sum of RBF and White kernel. For each of these new kernels, fit a GP model on the data and plot the mean and variance for each of your model fit. What impact do you think the White kernel has on the model fitting? Comment on the difference in your observations.

iii. Report the log marginal likelihood on the training data for each of the kernels in questions (d.i) and (d.ii). Comment on the difference in the log marginal likelihood values. What does the log marginal likelihood value tell you about the fit of the model with these kernels? Explain the differences. Which kernel, or combination of kernels, do you think is good for the GP model fit on this noisy data?

(c) [10 points] Load the Mauna Loa Atmospheric CO2 data. The dataset can be downloaded using `sklearn.datasets.fetch_mldata` and is also available in the `mauna.csv` file. This is a time series of monthly average atmospheric Carbon Dioxide concentrations in parts per million (vol) measured at Mauna Loa in Hawaii. You need to center the data (both input and output). For this question, you will design a kernel that will give a good GP fit for this data.

First plot the data and comment on the trends in this dataset.

Then, you need to find a good kernel or combination of kernels to account for the trends that you observed in the previous plot. You should experiment with the kernels you used in the previous questions (and others if you want) and combinations of these kernels. While you experiment with different kernels, in order to check whether your kernel leads to a good fit of the data, you should

- fit a GP model with the given kernel on the data
- plot the mean and variance of the GP fit for values of x ranging from m to $M + 30$ where m (resp. M) is the minimum (resp maximum) value of the inputs in the training data
- look at the log marginal likelihood on the training data

(no need to include these plots in the report).

Once you have found a good kernel or combinations of kernels, that you think is a good fit to model this CO2 data, report the log marginal likelihood value for your GP model and plot the mean and variance of the model between m and M (defined above). Why do you think the combination of kernels you found is a good kernel for this dataset?

2. [25 points] **EM algorithm**

In this question we will explore a mixture model for modeling text. Suppose you have a vocabulary of M words. We consider each word in a document as a random variable W whose value is a vector of M components, such that $W(i) = 1$ if the value of W is the ith word in the vocabulary, and 0 otherwise. Hence, $\sum_{i=1}^{M} W(i) = 1$ (this is also known as a one-hot encoding). Suppose the words are generated from a discrete mixture of K latent topics:

$$P(W) = \sum_{k=1}^{K} \pi_k P(W|\mu_k)$$

where π_k is the probability for the kth latent topic and $P(W|\mu_k)$ is modeled as:

$$P(W|\mu_k) = \prod_{i=1}^{M} (\mu_k(i))^{W(i)}$$

Hence, we generate a word by drawing a topic k from π and then drawing the word from the topic's distribution, according to μ_k (i.e. the ith component of $\mu_k \in \mathbb{R}^M$ is the probability of drawing the ith word knowing that the topic is k).

(a) [5 points] Suppose we have documents consisting of N words, which have been drawn i.i.d. according to this process. Suppose that for each document we have a given topic, which is known. Compute the maximum likelihood estimators for the π and μ parameters.

(b) [15 points] Suppose now that the topics are not known, and in fact, one document may cover multiple topics. Derive an expectation maximization algorithm for learning the parameters π and μ. In this case, for the expectation step, you need to compute the probability of the topic associated with each word W_j, in order to complete the data, and in the maximization step, you need to re-compute the parameters that maximize the likelihood of the data.

(c) [5 points] The assumption that words are drawn iid from a topic is quite strong. It would make more sense to assume a word's probability is conditioned on the topic as well as the previous word in the document. Explain how many parameters would the model have in this case, and what is the bias-variance trade-off compared to the previous model.

---

3. [20 points] **Neural Networks and Backpropagation**

In this question, we will explore neural networks and convolutional neural networks, and understand how the backpropagation algorithm works.

A neural network, also known as Multi-Layer Perceptrons (MLP), with one hidden layer is to be used for a multi-class classification problem. We can explicitly write down the set of parameters for the model: we denote the matrix of weights from input to the hidden layer as W and from the hidden layer to the output layer as V. There are K classes, the input of the network is a d-dimensional feature vector and there are n examples to train the MLP, $x_1, ...x_n \in \mathbb{R}^d$. For each of the training samples there is a target vector $y_1, ...y_n \in \mathbb{R}^K$, which uses a 1-out-of-K coding for the class of each of the training examples.

(a) What needs to be considered when designing the neural network? Your answer should include a discussion of the number of MLP parameters for the designed neural network.

(b) The MLP parameters are to be trained using cross-entropy. This cost-function has the form

$$E = -\sum_{i=1}^{I} \sum_{j=1}^{J} a_{i,j} \log(b_{i,j}) \tag{1}$$

i. For this expression describe what the variables I, J, $a_{i,j}$ and $b_{i,j}$ represent.

ii. Write down the likelihood term for this model. Comments?

iii. To avoid overfitting, add an L2 regulariser to the model. Write down the overall objective function that needs to be minimized.

iv. An alternative to L2 regularisation in neural networks is to use Dropout during training. Explain how Dropout works as a regulariser in model fitting.

v. Explain the difference between using a L2 and a Dropout regulariser to avoid overfitting.

vi. With the expression for the overall objective function, find the derivatives with respect to the parameters needed to implement the the backpropagation algorithm.

(c) The Hessian matrix is to be used in the optimisation process for the model parameters in $b(ii)$.

i. How is the Hessian matrix derived, and how can it be used to improve the training of the MLP.

ii. Discuss the computational cost of using the Hessian in optimising the output-layer weights.

---

4. [35 points] **Multi-Armed Bandits**

(a) [5 points] Let $X_1, X_2, \dots, X_N$ denote i.i.d. samples of a distribution of mean μ, which enjoy the following concentration bound:

$$\mathbb{P}\left[\left|\mu - \frac{1}{N}\sum_{n=1}^{N} X_n\right| > \varepsilon\right] < \sqrt{1+N}\exp\left(-\frac{N^2\varepsilon^2}{2(1+N)}\right).$$

Compute the associated upper confidence bound that would hold with **probability at least 1$-\boldsymbol{\delta}$** after having observed N samples. **Pay attention to the signs of the inequalities.

(b) [10 points] Derive the Thompson sampling algorithm to learn in bandits settings where rewards are assumed to be sampled from Poisson distributions.

(c) [10 points] Implement the Thompson sampling algorithm from (b). Execute it 20 times over 10000 episodes on a 5-armed environment where rewards are sampled from Poisson distribution with the following rates: $\{1, 2, 3, 4, 5\}$.

On a single figure, plot the average cumulative regret per episode (averaged over the 20 repetitions) along with the best and worst cumulative regret (obtained the 20 repetitions). Justify the prior parameters that you use.

(d) [10 points] Consider the UCB algorithm which samples each action once, and then selects at episode t the action

$$k_t = \text{argmax}_{k\in\mathcal{K}}\, \hat{\mu}_k(t-1) + \sqrt{\frac{2\log(1/\delta)}{N_k(t-1)}} \qquad \text{for } t > K.$$

Design a two-armed, Bernoulli bandit experiment to determine the practical effect of the choice of δ. Run the experiment and discuss the impact of δ given the suboptimal gap. Present figures to support your claims.