Machine Learning 2020

Homework 2

Due date : 2020/12/6 23:55:00 (Hard Deadline)

1 Sequential Bayesian Learning (40%)

Conjugate prior assures that the posterior distribution has the same functional form as the prior. The posterior is computed and viewed as the prior for the next parameter updating. This property plays an important role in sequential Bayesian learning.

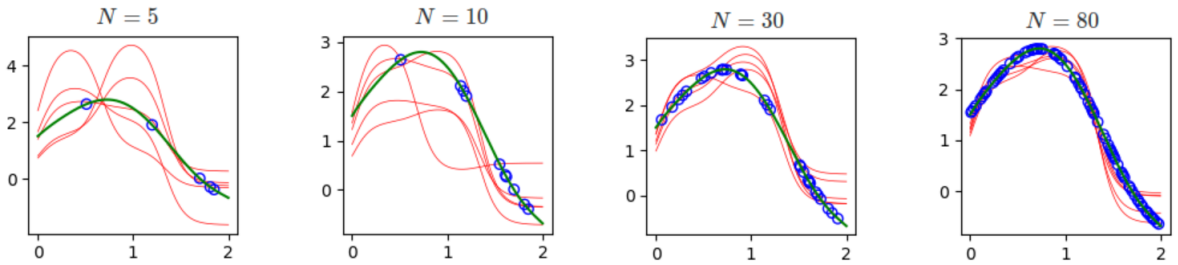
Dataset:
The file data.csv contains two sequences $\mathbf{x} = \{x_1, x_2, \ldots, x_{100} | 0 \leq x_i \leq 2\}$ and $\mathbf{t} = \{t_1, t_2, \ldots, t_{100}\}$ which represent the input sequence and the corresponding target sequence, respectively.

Basis Function:
Please apply the sigmoid basis functions $\boldsymbol{\phi} = [\phi_0, \ldots, \phi_{M-1}]^\top$ of the form $\phi_j(x) = \sigma(\frac{x-\mu_j}{s})$. In this exercise, please take the following parameter settings for your basis functions: $M = 3$, $s = 0.6$ and $\mu_j = \frac{2j}{M}$ with $j = 0, \ldots, M-1$. Please take the data size to be $N = 5, 10$, 30 and 80 for each of the following questions.

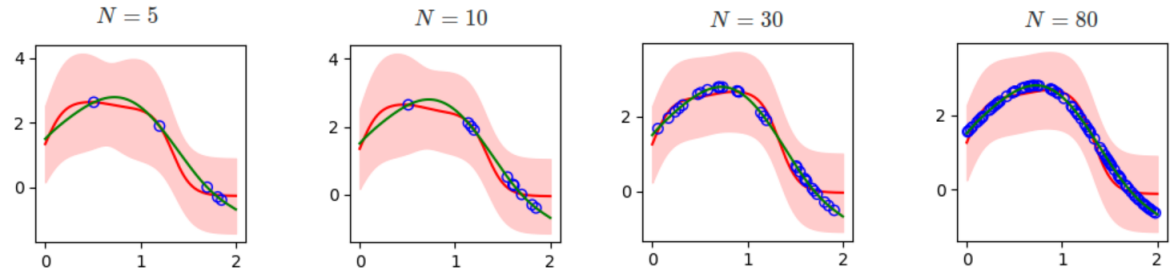
Bayesian Learning:
Please compute the mean vector $\mathbf{m}_N$ and the covariance matrix $\mathbf{S}_N$ for the posterior distribution $p(\mathbf{w}|\mathbf{t}) = \mathcal{N}(\mathbf{w}|\mathbf{m}_N, \mathbf{S}_N)$ with the given prior $p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{m}_0 = \mathbf{0}, \mathbf{S}_0^{-1} = 10^{-6}\mathbf{I})$. The precision of likelihood function $p(\mathbf{t}|\mathbf{w}, \beta)$ or $p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta)$ is chosen to be $\beta = 1$.

Note: You need to train your model by fitting data sequentially, this means that when you have calculated the result of case $N = 5$, you can use another 5 data points to calculate the result of case $N = 10$.

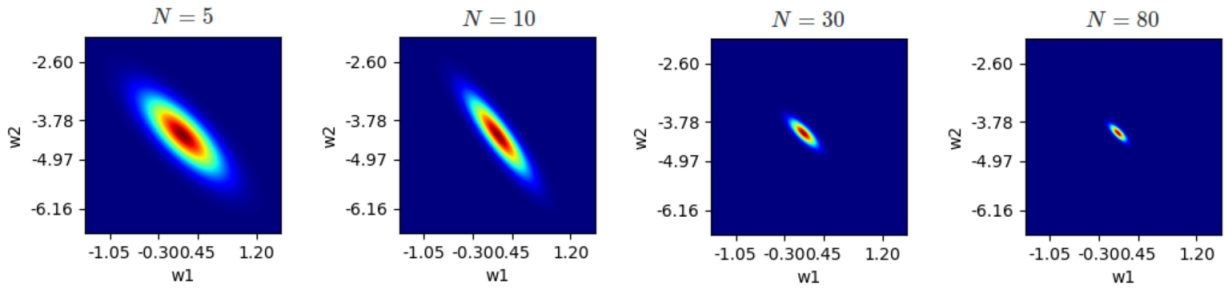
1. **Plot** five curves sampled from the parameter posterior distribution and N data points, e.g. (10%)



2. **Plot** the predictive distribution of target value t by showing the mean curve, the region of variance with one standard deviation on both sides of the mean curve and N data points, e.g. (10%)



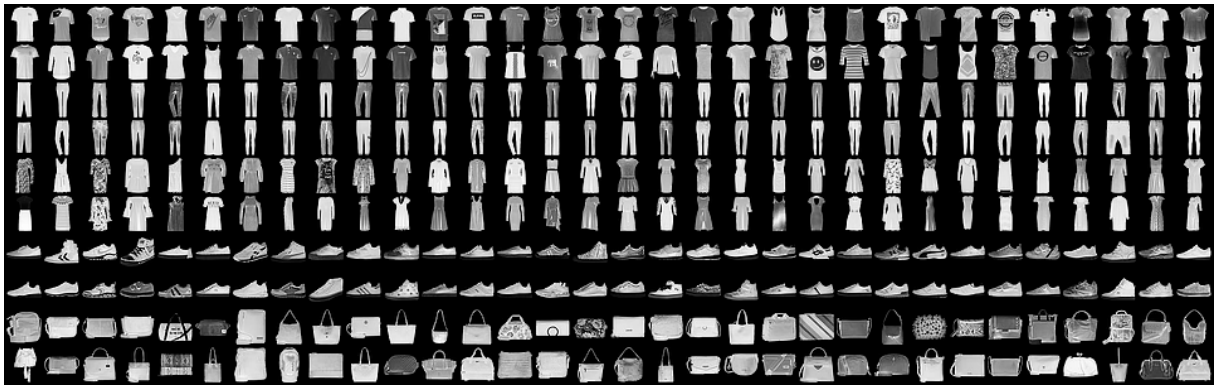
3. **Plot** the prior distributions by arbitrarily selecting two weights, e.g. (10%)



4. Make some discussion on the results of different N in 1, 2 and 3. (10%)

2 Logistic Regression (60%)

You are given the dataset [1] of fashion products (Fashion_MNIST.zip). This dataset contains 5 classes. There are 64 different images in each class. In this exercise, you need to implement batch GD (batch gradient descent), SGD (stochastic gradient descent), mini-batch SGD and Newton-Raphson algorithms to construct a multiclass logistic regression model with softmax transformation ($p(C_k|\boldsymbol{\phi}_n) = \exp(a_{nk})/\sum_j \exp(a_{nj})) = y_k(\boldsymbol{\phi}_n) \triangleq y_{nk}$. The error function is formed by $E(\mathbf{w}) = -\sum_{m=1}^{N}\sum_{k=1}^{K} t_{nk} \log y_{nk}$.



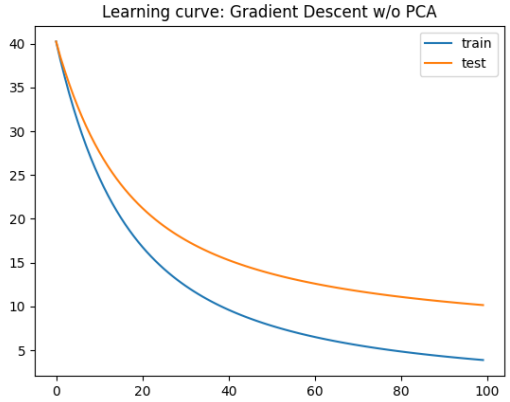
Algorithms	Batch size	Iterations in one epoch
batch GD	N	1
SGD	1	N
mini-batch SGD	B	N/B
Newton-Raphson	N	1

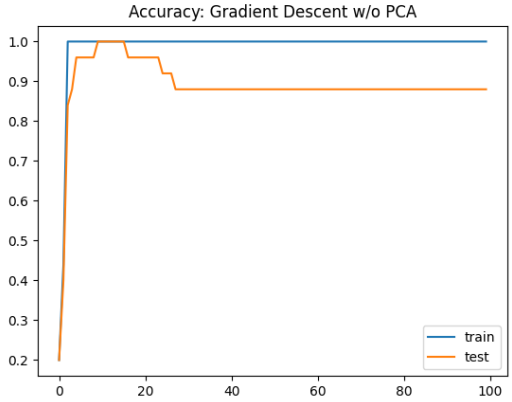
N = number of training data, B = batch size

Note: You need to normalize the data samples before training and randomly select 32 images as test data for each class.

1. Set the initial weight vector $\mathbf{w}_k = [w_{k1}, \ldots, w_{kF}]$ to be a zero vector where F is the number of features and k is the number of classes. Implement batch GD, SGD, mini-batch SGD (batch size = 32) and Newton-Raphson algorithms to construct a multiclass logistic regression. (15%)

   (a) **Plot** the learning curves of $E(\mathbf{w})$ and the accuracy of classification versus the number of epochs until convergence for training data as well as test data, e.g.





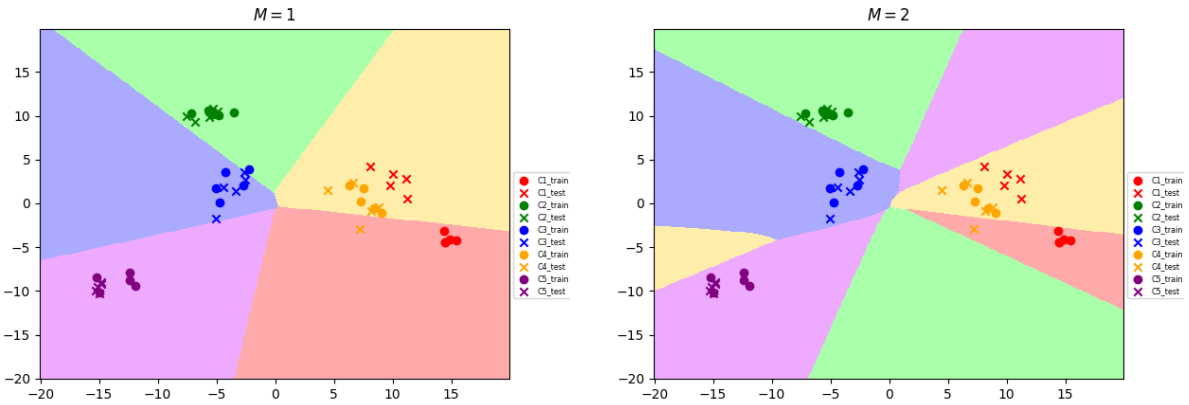
(b) **Show** the classification results of training and test data, e.g.

```
TYPE: Gradient Descent
w/o PCA
Training accuracy    : 1.00
Test accuracy        : 0.88
```

2. Use principal component analysis (PCA) to reduce the dimension of images to $d = 2, 5, 10$. (15%)
   (a) Repeat 1 by using PCA to reduce the dimension of images to d.
   (b) **Plot** d eigenvectors corresponding to top d eigenvalues, e.g.

Left: Olivetti faces dataset [2]; **Right:** Examples of top 2 eigenvectors

3. What do the decision regions and data points look like on the vector space? (15%)
   (a) **Plot** the decision regions and data points of the images on the span of top 2 eigenvectors by using PCA to reduce the dimension of images to 2.
   (b) Repeat 3(a) by changing the order from $M = 1$ to $M = 2$, e.g.



4. Make some discussion on the results of 1, 2 and 3. (15%)

3 Rule

- In your submission, you need to submit three files.
  Note : Only the following three files are accepted, so the code of two exercises should be written in one **.py** file.
  - **hw2_StudentID.ipynb** file which contains all the results and codes for this homework.
  - **hw2_StudentID.py** the content of this file must be consistent with .ipynb file.
  - **hw2_StudentID.pdf** file which is the report that contains your description for this homework.
    (e.g. hw2_0123456.ipynb; hw2_0123456.py; hw2_0123456.pdf)
- Implementation will be graded by
  - Completeness
  - Algorithm Correctness
  - Description of model design
  - Discussion and analysis
- Only Python implementation is acceptable.
- Numpy, Pandas and Matplotlib library are recommended for the implementation.
- Don't use high level toolbox/module functions (e.g. sklearn, polyfit).
- DO NOT PLAGIARISM. (We will check program similarity score.)

References

[1] Han Xiao, Kashif Rasul, and Roland Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017.

[2] Ferdinando S Samaria and Andy C Harter, "Parameterisation of a stochastic model for human face identification," in *Proc. of IEEE Workshop on Applications of Computer Vision*. IEEE, 1994, pp. 138–142.