ELEN 4720 Problem 1.

Zhuoyu Feng

zf2272

(a). Given $p(y_0=y|\pi) = \text{Bernoulli}(y|\pi) = \begin{cases} \pi^y(1-\pi)^{1-y} & y=0,1 \\ 0, & y\neq 0,1 \end{cases}$

$$\hat{\pi} = \arg\max_{\pi} \sum_{i=1}^{n} \ln p(y_i|\pi) = \arg\max_{\pi} \sum_{i=1}^{n} \ln[\pi^{y_i}(1-\pi)^{1-y_i}]$$

$$= \arg\max_{\pi} \sum_{i=1}^{n} [y_i \ln\pi + (1-y_i)\ln(1-\pi)]$$

Let $\frac{\partial\left(\sum_{i=1}^{n}[y_i\ln\pi + (1-y_i)\ln(1-\pi)]\right)}{\partial\pi} = 0$, $\Rightarrow \sum_{i=1}^{n}\left(\frac{y_i}{\pi} - \frac{1-y_i}{1-\pi}\right) = 0$.

Hence, $\hat{\pi} = \frac{1}{n}\sum_{i=1}^{n} y_i$

(b). $\hat{\lambda}_{y,d} = \arg\max_{\lambda_{y,d}}\left(\ln p(\lambda_{0,d}) + \ln p(\lambda_{1,d}) + \sum_{i=1}^{n}\ln p(x_{i,d}|\lambda_{y_i,d})\right)$

$$= \arg\max_{\lambda_{y,d}}\left(\ln p(\lambda_{0,d}) + \ln p(\lambda_{1,d}) + \sum_{i=1}^{n}\ln p(x_{i,d}|\lambda_{y,d})\mathbb{1}(y_i=y)\right)$$

$\ln P(\lambda_{0,d})$, $\ln P(\lambda_{1,d})$ can be viewed as constants, since y are fixed.

Given, $x_{i,d}|y_i \sim \text{Pois}(\lambda_{y_i,d})$, $p(x_{i,d}|\lambda_{y,d}) = \frac{\lambda_{y,d}^{x_{i,d}} \cdot e^{-\lambda_{y,d}}}{x_{i,d}!}$

$$\therefore \hat{\lambda}_{y,d} = \arg\max_{\lambda_{y,d}} \sum_{i=1}^{n}(x_{i,d}\ln\lambda_{y,d} - \lambda_{y,d} - \ln x_{i,d}!)\mathbb{1}(y_i=y)$$

$$\frac{\partial \sum_{i=1}^{n}(x_{i,d}\ln\lambda_{y,d} - \lambda_{y,d} - \ln x_{i,d}!)\mathbb{1}(y_i=y)}{\partial\lambda_{y,d}} = 0.$$

Hence, $\hat{\lambda}_{y,d} = \frac{\sum_{i=1}^{n} x_{i,d}\mathbb{1}(y_i=y)}{\sum_{i=1}^{n}\mathbb{1}(y_i=y)}$

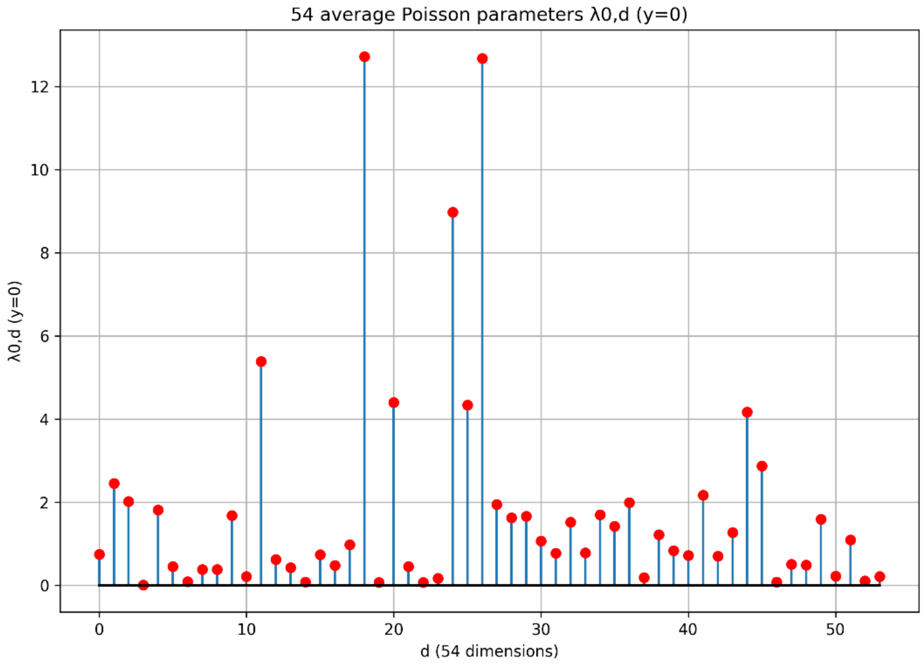
Problem2

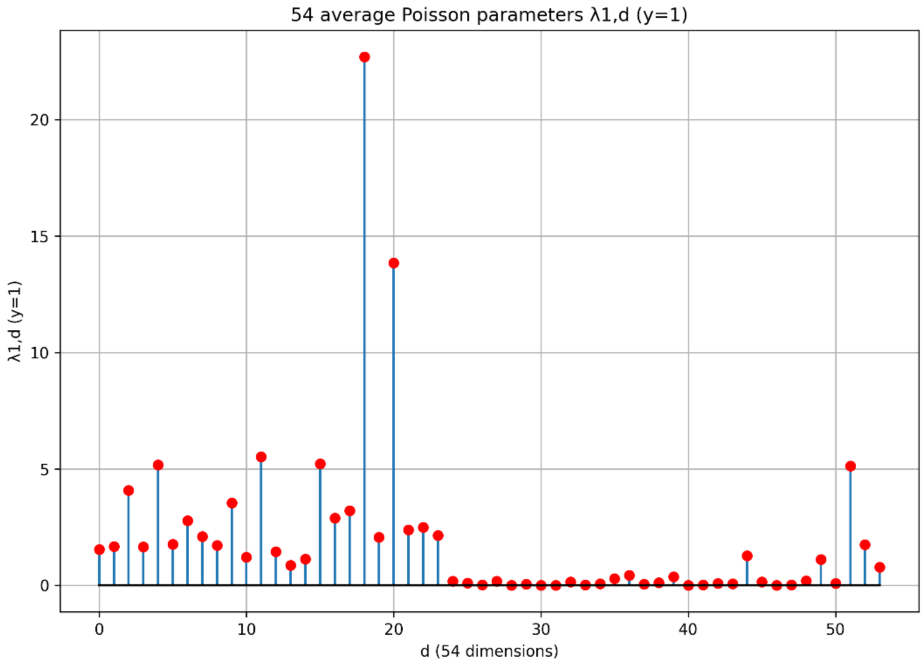
(a) Prediction Table:

	Model prediction y' = 0	Model prediction y' = 1
Ground truth y = 0	2297	490
Ground truth y = 1	110	1703

Prediction accuracy = (2297 + 1703)/4600 = 0.870

(b) A stem plot of the 54 Poisson parameters for each class averaged across the 10 runs:





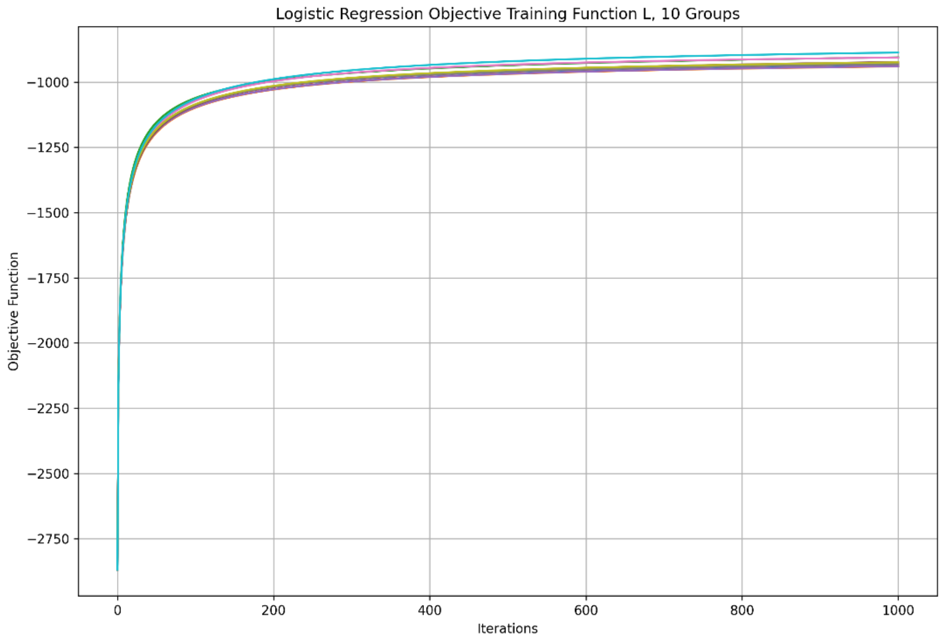
According to the README file, dimension 16 denotes the word “free”, dimension 52 denotes the word “!”.

Since for Poisson distribution, parameter λ is equal to the expected value of X, i.e., λ = E(X), which indicates that larger parameter λ means a larger expected value of X, closer to 1, so it is more likely that it is extracted from a spam email.

For d=16, when y=0, λy,d $\approx$ 8, when y=1, λy,d $\approx$ 52. It indicates that when an email contains the word “free”, the probability of this email being spam is about 6 times compared to when it doesn’t contain that word. Because it may be a sales promotion emails that use “free” price to attract customers.

For d=52, when y=0, λy,d $\approx$ 12, when y=1, λy,d $\approx$ 52. It indicates that when an email contains the word “!”, the probability of this email being spam is about 4 times compared to when it doesn’t contain that word. Because it may use “!” to emphasize and induce people.

(c) Logistic Regression objective training function L per iteration for each of the 10 training runs:



(d) Update for Wt+1:

(d). Derive the update for W_{t+1} for the logistic regression problem:

$$\mathcal{L}(w) \approx \mathcal{L}'(w) \equiv \mathcal{L}(W_t) + (w-W_t)^T \nabla \mathcal{L}(W_t) + \frac{1}{2}(W-W_t)^T \nabla^2 \mathcal{L}(W_t)(W-W_t)$$

Comput the derivation on W at both sides:

$$\nabla \mathcal{L}'(w) = \nabla \mathcal{L}(w_t) + \frac{1}{2}(w-W_t)^T \nabla^2 \mathcal{L}(W_t)^T + \frac{1}{2}\nabla^2 \mathcal{L}(W_t)(W-W_t)$$

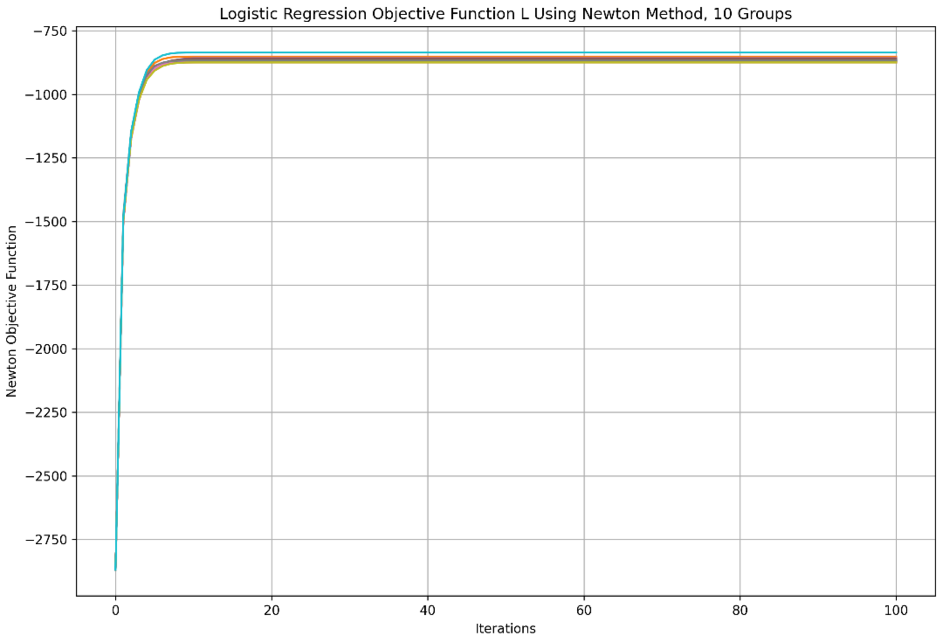
$$= \nabla \mathcal{L}(W_t) + \nabla^2 \mathcal{L}(W_t)(W-W_t)$$

Since $W_{t+1} = \arg\max_W \mathcal{L}'(w)$,

let $\nabla \mathcal{L}(W_t) + \nabla^2 \mathcal{L}(W_t)(W-W_t) = 0$, and set $W = W_{t+1}$, we have:

$$\begin{cases} W_{t+1} = W_t - (\nabla^2 \mathcal{L}(W_t))^{-1} \nabla \mathcal{L}(W_t). \\ \nabla \mathcal{L}(W_t) = \sum_{i=1}^{n} (1-\sigma_i(y_i \cdot w))\, y_i x_i \\ \nabla^2 \mathcal{L}(w_t) = -\sum_{i=1}^{n} \sigma_i(y_i w) \cdot [1-\sigma_i(y_i \cdot w)] \cdot x_i^T x_i \end{cases}$$

The objective function L on the training data as a function of t = 1, …, 100 for each of the 10 training runs:



(e) The testing results using Newton's method:

	Model prediction y' = 0	**Model prediction y' = 1**
Ground truth y = 0	2645	142
Ground truth y = 1	211	1602

Prediction accuracy = (2645 + 1602)/4600 = 0.923

Problem3

(a) RMSE table on the 42 test points, use the mean of the Gaussian process at the test point as prediction:

		σ^2									
		0.1	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1**
b	**5**	1.97	1.93	1.92	1.92	1.92	1.93	1.93	1.94	1.95	1.95
	7	1.92	1.90	1.91	1.92	1.92	1.93	1.94	1.95	1.96	1.97
	9	1.90	1.90	1.92	1.93	1.95	1.96	1.97	1.98	1.98	1.99
	11	1.89	1.91	1.94	1.96	1.97	1.99	2.00	2.01	2.01	2.02
	13	1.90	1.94	1.96	1.99	2.00	2.01	2.02	2.03	2.04	2.05
	15	1.91	1.96	1.99	2.01	2.03	2.04	2.05	2.06	2.07	2.07

(b) The best value (RMSE = 1.89) comes from b = 11 and σ^2 = 0.1. In Problem3 of Homework1, the lowest RMSE value is approximately 2.08 at λ=51, p=3.

A drawback of using the approach in this homework compared with homework 1 is using Gaussian process model for regression may cost more time and space. There are two parameters: b and σ^2, both have a wide range of values needs to be computed and compared. In this problem it provides us two groups of values, which saves us some time. But in practice everything is unknow and we need to try by ourselves. On the contrary, in Homenwork1, when we implement pth-order polynomial regression, once p is fixed, we only have to deal with one parameter – λ. What's more, in Gaussian process, it contains some inverse matrix and kernel manipulation, which are time-consuming. As for the space, it needs larger matrixes to store the intermediate results.

(c) A scatter plot of the data (x[4] versus y for each point) and a solid line of the predictive mean of the Gaussian process at each point in the training set: