COMS4721 Machine Learning for Data Science Homework 2

Po-Chieh Liu

UNI: pl2441

Problem 1.

(a) Derive $\hat{\pi}$ using the given objective function:

Take derivative on f in regards to π, and set the equation equals to 0, then we can derive the maximum likelihood estimator.

$$\frac{\partial f}{\partial \pi} = 0$$

$$\frac{\partial}{\partial \pi}\sum_{i=1}^{n} ln\big(p(y_i|\pi)\big) = \frac{\partial}{\partial \pi}\sum_{i=1}^{n} ln(\pi^{y_i}(1-\pi)^{1-y_i})$$

$$= \frac{\partial}{\partial \pi}\sum_{i=1}^{n}(y_i\, ln\, \pi + (1-y_i)\, ln(1-\pi)) = \sum_{i=1}^{n}\left(\frac{y_i}{\pi} - \frac{1-y_i}{1-\pi}\right)$$

$$= \frac{\sum_{i=1}^{n} y_i}{n\pi} - \frac{n - \sum_{i=1}^{n} y_i}{n - n\pi} = 0$$

$$(1-\pi)\sum_{i=1}^{n} y_i - n\pi + \pi\sum_{i=1}^{n} y_i = 0$$

$$\sum_{i=1}^{n} y_i - n\pi = 0$$

Rearrange the equation, we can derive the maximum likelihood estimator:

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

(b) Derive $\widehat{\lambda_{y,d}}$ using the given objective function:

Take derivative on f in regards to $\lambda_{y,d}$ for both $y = 0$ and $y = 1$, and set the equation equals to 0. Then we can derive the maximum likelihood estimators.

$$\frac{\partial f}{\partial \lambda_{y,d}} = 0$$

$$\frac{\partial}{\partial \lambda_{y,d}}\left(ln\, p\big(\lambda_{y,d}\big)+\sum_{i=1}^{n} ln\, p(x_{i,d}|\lambda_{y,d})\right)=\frac{\partial}{\partial \lambda_{y,d}}\left(ln\frac{\lambda_{y,d}e^{-\lambda_{y,d}}}{\Gamma(2)}+\sum_{i=1}^{n} ln\frac{\lambda_{y,d}^{x_{i,d}}e^{-\lambda_{y,d}}}{x_{i,d}!}\right)$$

$$=\frac{\partial}{\partial \lambda_{y,d}}\Bigg(ln\,\lambda_{y,d}-\lambda_{y,d}-ln\,\Gamma(2)$$

$$+\sum_{i=1}^{n}\big(x_{i,d}\,ln\,\lambda_{y,d}-\lambda_{y,d}-ln\big(x_{i,d}!\big)\big)\Bigg)=\frac{1}{\lambda_{y,d}}-1+\frac{\sum_{i=1}^{n}x_{i,d}}{\lambda_{y,d}}-n$$

$$=0$$

Rearrange:

$$\widehat{\lambda_{y,d}}=\frac{\sum_{i=1}^{n}x_{i,d}+1}{n_{y_i}+1}$$

For $\lambda_{y,d}$, y can be either 1 or 0, thus, we can use indicator function to separate $\widehat{\lambda_{0,d}}$ and $\widehat{\lambda_{1,d}}$:

$$\widehat{\lambda_{0,d}}=\frac{\sum_{i=1}^{n}x_{i,d}+1}{n_{y_i}+1}\mathbb{I}(y_i=0)$$

$$\widehat{\lambda_{1,d}}=\frac{\sum_{i=1}^{n}x_{i,d}+1}{n_{y_i}+1}\mathbb{I}(y_i=1)$$

Problem 2.

(a) The model outputs are listed below:

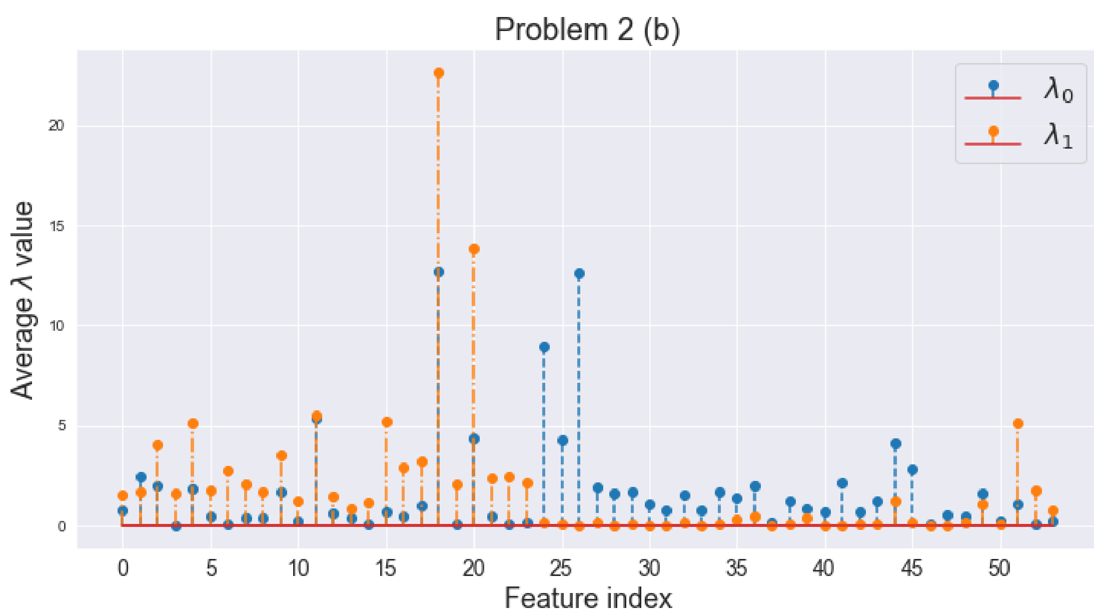
	Label 1	Label 0
Predict 1	1714	490
Predict 0	99	2297

Model accuracy is around 0.872

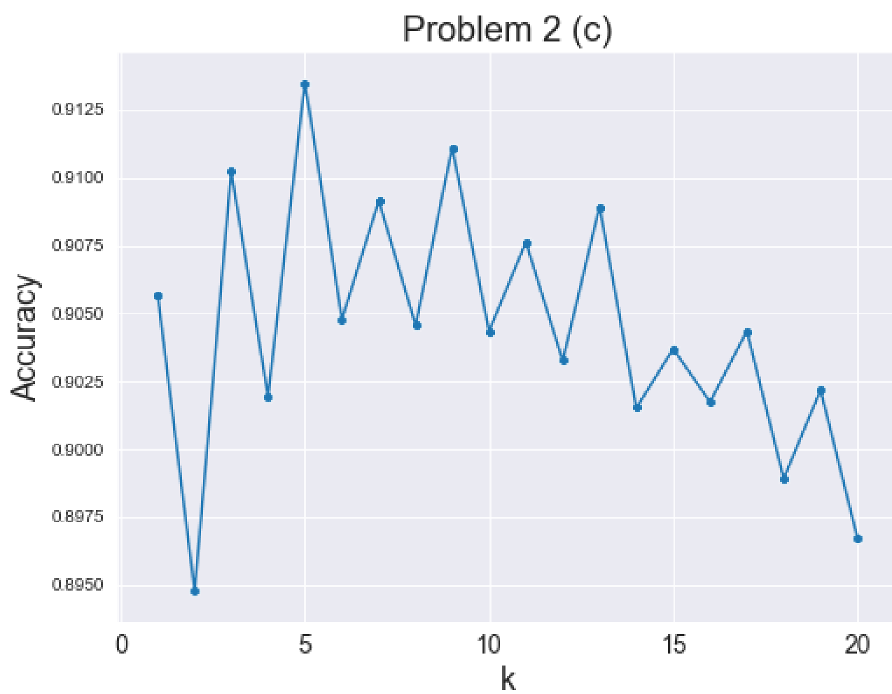
Note, the prediction formula is modified by taking log and ignore the comment term factorial of x in my code.

$$\mathrm{y}_0 = \arg\max_y \ln\left(p(y_0 = y|\hat{\pi}) \prod_{d=1}^{D} p\left(x_{0,d}|\widehat{\lambda_{y,d}}\right)\right)$$

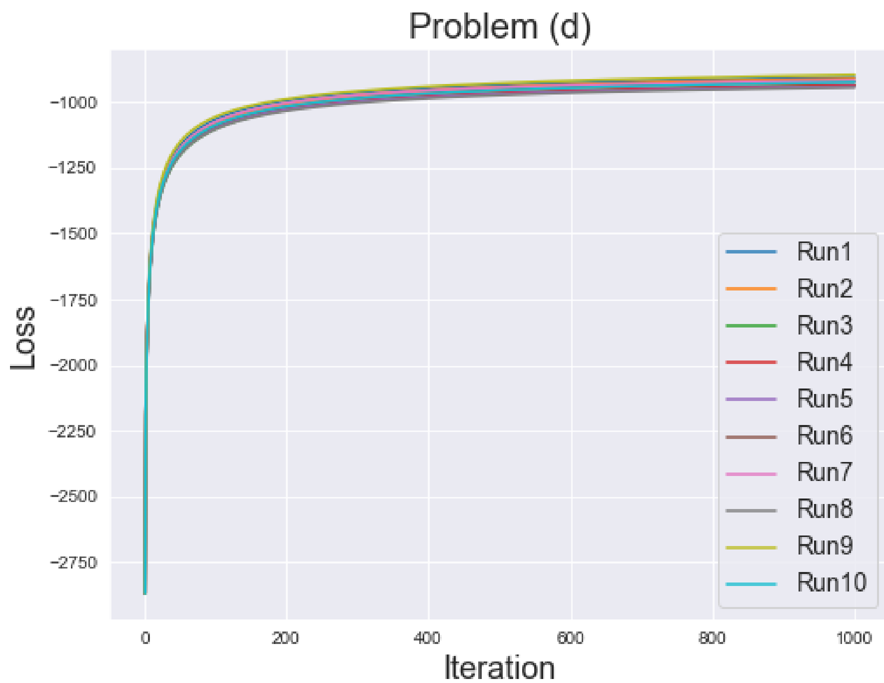
(b) The stem plot of two categories' λ is attached. From the REAEME file, dimensions 16 and 52 (corresponding indexes in the plot are 15 and 51) are "free" and "!". We can find for both features, the $\widehat{\lambda_1}$ coefficients are higher than $\widehat{\lambda_0}$. Checking the actual values of $\widehat{\lambda_{1,16}}$ and $\widehat{\lambda_{0,16}}$, the values imply a mail contains "free" is 5 times higher chance to be spam mail than mail without "free". Similarly, the values show that probability of a mail contains "!" is spam is around 3 times higher than mail without "!" for dimension 52.



(c) The prediction accuracy plot is attached below. The average accuracy is around 0.9025. The most optimized k is 5 with the maximum accuracy across k equals 1 to 20. We can roughly conclude that the accuracy increased when k increased from 1 to 5, then decreased from k equals 5 to 20. Note, I used round function to estimate the final decision of class. For all tie case, the prediction will be 1.



(d) The $\mathcal{L}$ plot is attached. The loss function values increased dramatically at very beginning, and started being plateau around iteration 200. The highest loss value is around -1000.



(e) The equation concept is basically applying Tyler expansion and truncate the high order term:

$$f(x) = f(x_0) + (x - x_0)f'(x_0) + \frac{1}{2!}(x - x_0)^2 f''(x_0) + \cdots$$

Then we can take derivative in regarding to x and we want find the 0 point:

$$0 = f'(x) = f'(x_0) + \frac{1}{2!}2(\mathrm{x} - \mathrm{x}_0)\mathrm{f}''(\mathrm{x}_0)$$

Rearrange:

$$x = x_0 - \frac{f'(x_0)}{f''(x_0)}$$

From above, we need to find the second order derivative to update weights. In matrix notation, we can express the 2nd order derivative using Hessian Matrix.

$$x = x_0 - H^{-1}(x_0)\nabla f(x_0)$$

Apply to Logistic regression and using class notation:

$$w^{t+1} = w^t - (\nabla_w^2 \mathcal{L})^{-1} \nabla_w \mathcal{L}$$

$\nabla_w \mathcal{L}$ and $\nabla_{\mathrm{w}}^2 \mathcal{L}$ can be expressed as following:

$$\nabla_w \mathcal{L} = \nabla_w \left(\sum \ln \sigma_i (y_i \cdot w) \right) = \nabla_w \left(\sum \ln \frac{e^{y_i x_i^T w}}{1 + e^{y_i x_i^T w}} \right)$$

$$= \nabla_w \left(\sum y_i x_i^T w - \ln \left(1 + e^{y_i x_i^T w} \right) \right)$$

$$= \sum \left(y_i x_i^T - \frac{y_i x_i^T e^{y_i x_i^T w}}{1 + e^{y_i x_i^T w}} \right) = \sum \left(1 - \frac{e^{y_i x_i^T w}}{1 + e^{y_i x_i^T w}} \right) y_i x_i^T$$

$$= \sum \big(1 - \sigma(y_i w) \big) y_i x_i^T$$

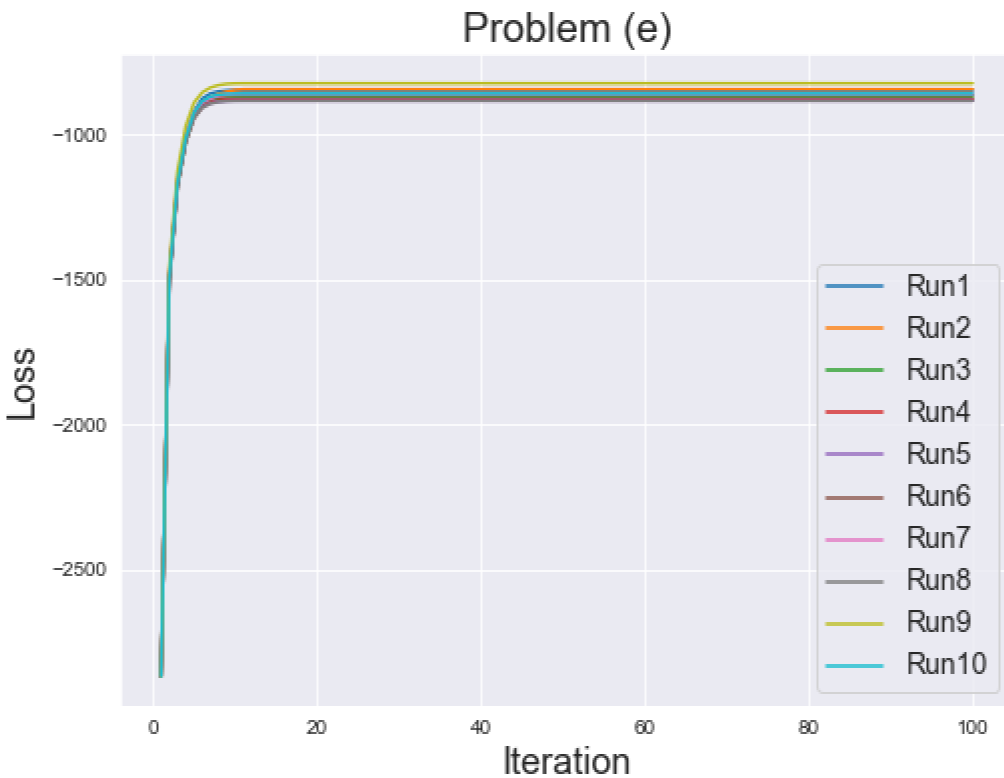
$$\nabla_w^2 \mathcal{L} = \nabla_w (\nabla_w \mathcal{L}) = \nabla_w \left(\sum \big(1 - \sigma(y_i w) \big) y_i x_i^T \right) = \nabla_w \left(- \sum \sigma(y_i w) y_i x_i^T \right)$$

$$= -\nabla_w \sum y_i x_i \left(\frac{e^{y_i x_i^T w}}{1 + e^{y_i x_i^T w}} \right)$$

$$= -\sum y_i x_i \frac{\left(1 - e^{y_i x_i^T w} \right) \left(y_i x_i^T e^{y_i x_i^T w} \right) - \left(e^{y_i x_i^T w} \right) \left(-y_i x_i^T e^{y_i x_i^T w} \right)}{\left(1 - e^{y_i x_i^T w} \right)^2}$$

$$= -\sum y_i x_i \frac{y_i x_i^T e^{y_i x_i^T w}}{\left(1 - e^{y_i x_i^T w} \right)^2} = -\sum_{i=1}^{n} \sigma(y_i w)(1 - \sigma(y_i w)) x_i x_i^T$$

The plot is attached. We can observe when applying Newton method, the loss function converges faster than problem (d).



(f) The model outputs are listed below:

	Label 1	Label 0
Predict 1	1595	147
Predict 0	218	2640

Model accuracy is around 0.921.