SOUTH UNIVERSITY OF SCIENCE AND TECHNOLOGY OF CHINA

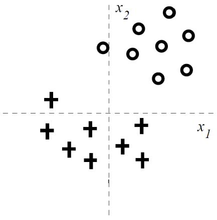
Intro to Big Data Science: Assignment 3

Due Date: April 19, 2024

Exercise 1

Log into Cookdata, and enroll the course “数据科学导引”. Finish the online exercise there.

Exercise 2 We consider here a discriminative approach for solving the classification problem illustrated in the following figure, where “+” corresponds to class $y = 1$ and “O” corresponds to class $y = 0$:



1. We attempt to solve the binary classification task depicted in Figure 1 with the simple linear logistic regression model

$$P(y = 1|\mathbf{x}, \mathbf{w}) = \sigma(w_0 + w_1x_1 + w_2x_2) := \frac{1}{1 + \exp(-w_0 - w_1x_1 - w_2x_2)}.$$

   where $\mathbf{x} = (x_1, x_2)^T$, $\mathbf{w} = (w_0, w_1, w_2)^T$, and σ is the sigmoid function defined by the last equality. Notice that the training data can be separated with zero training error with a linear separator.

Consider training regularized linear logistic regression models for the training set $\{(\mathbf{x}^{(i)}, y^{(i)})\}$, where we try to maximize

$$\sum_{i=1}^{n} \log P(y^{(i)}|\mathbf{x}^{(i)}, \mathbf{w}) - C w_j^2,$$

for very large C. The regularization penalties used in penalized conditional log-likelihood estimation are $-C w_j^2$ where $j = \{0, 1, 2\}$. In other words, only one of the parameters is regularized in each case. Given the training data in the above figure, how does the training error changes with regularization of each parameter w_j? State whether the training error increases or stays the same (zero) for each w_j ($j = 0, 1, 2$) for very large C. Provide a brief justification for each of your answers.

2. If we change the form of regularization to L_1-norm (absolute value) and regularize w_1 and w_2 only (but not w_0), we get the following penalized log-likelihood

$$\sum_{i=1}^{n} \log P(y^{(i)}|\mathbf{x}^{(i)}, \mathbf{w}) - C(|w_1| + |w_2|).$$

Consider again the problem in the above figure and the same linear logistic regression model $P(y = 1|\mathbf{x}, \mathbf{w}) = \sigma(w_0 + w_1 x_1 + w_2 x_2)$.

a) As we increase the regularization parameter C, which of the following scenarios do you expect to observe? (Choose only one) Briefly explain your choice.

(A) First w_1 will become 0, then w_2.

(B) First w_2 will become 0, then w_1.

(C) w_1 and w_2 will become zero simultaneously.

(D) None of the weights will become exactly zero, only smaller as C increases.

b) For very large C, with the same L_1-norm regularization for w_1 and w_2 as above, which value(s) do you expect w_0 to take? Explain your answer by doing some calculation. (Note that the number of points from each class is the same.) (You can give a range of values for w_0 if you deem necessary).

c) Assume that we obtain more data points from the "+" class that corresponds to $y = 1$ so that the class labels become unbalanced. Again for very large C, with the same L_1-norm regularization for w_1 and w_2 as above, which value(s) do you expect w_0 to take? Explain briefly. (You can give a range of values for w_0 if you deem necessary).

✏ **Exercise 3 (Linear regression)**

Consider a multivariate liner model $\mathbf{y} = \mathbf{X}\mathbf{w} + \epsilon$ with $\mathbf{y} \in \mathbb{R}^{n\times 1}$, $\mathbf{X} \in \mathbb{R}^{n\times(d+1)}$, $\mathbf{w} \in \mathbb{R}^{(d+1)\times 1}$, and $\epsilon \in \mathbb{R}^{n\times 1}$, where $\epsilon \sim N(\mathbf{0}, \sigma^2\mathbf{I})$, follows the normal distribution.

1. Show that the linear regression predictor is given by $\hat{\mathbf{y}} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$.

2. Let $\mathbf{P} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$, show that $\mathbf{P}$ has only 0 and 1 eigenvalues.

3. Show that $\hat{\mathbf{w}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$ is an unbiased estimator of $\mathbf{w}$, i.e., $\mathbf{E}(\hat{\mathbf{w}}) = \mathbf{w}$. Also show that $\mathrm{Var}(\hat{\mathbf{w}}) = (\mathbf{X}^T\mathbf{X})^{-1}\sigma^2$. (Note that by definition, $\mathrm{Var}(\hat{\mathbf{w}}) = \mathbf{E}[(\hat{\mathbf{w}} - \mathbf{E}(\hat{\mathbf{w}}))(\hat{\mathbf{w}} - \mathbf{E}(\hat{\mathbf{w}}))^T])$.
4. Recall the definition of R^2 score: $R^2 := 1 - \frac{SS_{res}}{SS_{tot}}$, where $SS_{tot} = \sum_{i=1}^n (y_i - \bar{y})^2$, $SS_{reg} = \sum_{i=1}^n (\hat{y}_i - \bar{y})^2$, and $SS_{res} = \sum_{i=1}^n (y_i - \hat{y}_i)^2$. Prove that for linear regression, $SS_{tot} = SS_{reg} + SS_{res}$. (So that R^2 score can also be defined as $R^2 = \frac{SS_{reg}}{SS_{tot}}$)
5. Repeat the questions in 1 and 3 if we are using ridge regression with a regularization parameter λ.

✏ **Exercise 4 (Generalized Cross-Validation)** Consider ridge regression:

$$\min_{\mathbf{w}} \left[(\mathbf{y} - \mathbf{X}\mathbf{w})^T(\mathbf{y} - \mathbf{X}\mathbf{w}) + \lambda \|\mathbf{w}\|_2^2 \right]$$

It has the solution $\hat{\mathbf{w}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$ and prediction $\hat{\mathbf{y}} = \mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y} = \mathbf{P}\mathbf{y}$ with $\mathbf{P} = \mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T$ be the projection matrix.

1. Define the leave-one-out cross validation estimator as

   $$\hat{\mathbf{w}}^{[k]} = \arg\min_{\mathbf{w}} \Big[\sum_{i=1, i\neq k}^{n} (y_i - \mathbf{x}_i^T\mathbf{w})^2 + \lambda \|\mathbf{w}\|_2^2 \Big].$$

   Show that $\hat{\mathbf{w}}^{[k]} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I} - \mathbf{x}_k\mathbf{x}_k^T)^{-1}(\mathbf{X}^T\mathbf{y} - \mathbf{x}_k y_k)$
2. (Optional) Define the ordinary cross-validation (OCV) mean squared error as $V_0(\lambda) = \frac{1}{n}\sum\limits_{k=1}^{n} (\mathbf{x}_k^T\hat{\mathbf{w}}^{[k]} - y_k)^2$. Show that $V_0(\lambda)$ can be rewritten as $V_0(\lambda) = \frac{1}{n}\sum\limits_{k=1}^{n} \left(\frac{\hat{y}_k - y_k}{1 - p_{kk}}\right)^2$, where $\hat{y}_k = \sum_{j=1}^n p_{kj} y_j$ and p_{kj} is the (k, j)-entry of $\mathbf{P}$.

   (Hint: You may need to use the Sherman-Morrison Formula for nonsingualar matrix $\mathbf{A}$ and vectors $\mathbf{x}$ and $\mathbf{y}$ with $\mathbf{y}^T\mathbf{A}^{-1}\mathbf{x} \neq -1$: $(\mathbf{A} + \mathbf{x}\mathbf{y}^T)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{x}\mathbf{y}^T\mathbf{A}^{-1}}{1 + \mathbf{y}^T\mathbf{A}^{-1}\mathbf{x}}\Big)$
3. (Optional) Define weights as $w_k = \left(\frac{1 - p_{kk}}{\frac{1}{n}tr(\mathbf{I} - \mathbf{P})}\right)^2$ and weighted OCV as $V(\lambda) = \frac{1}{n}\sum\limits_{k=1}^{n} w_k(\mathbf{x}_k^T\hat{\mathbf{w}}^{[k]} - y_k)^2$. Show that $V(\lambda)$ can be written as

   $$V(\lambda) = \frac{\frac{1}{n}\|(\mathbf{I} - \mathbf{A})\mathbf{y}\|^2}{\Big[1 - tr(\mathbf{P})/n\Big]^2}$$

✏ **Exercise 5** (Solving LASSO by ADMM) The alternating direction method of multipliers (ADMM) is a very useful algorithm for solving the constrained optimization problem:

$$\min_{\theta, z} f(\boldsymbol{\theta}) + g(\mathbf{z}), \qquad \text{subject to} \quad \mathbf{A}\boldsymbol{\theta} + \mathbf{B}\mathbf{z} = \mathbf{c}.$$

The algorithm is given by using Lagrange multiplier $\mathbf{u}$ for the constraint. The detail is as follows:

1. $\boldsymbol{\theta}^{(k+1)} = \arg\min\limits_{\boldsymbol{\theta}} L(\boldsymbol{\theta}, \mathbf{z}^{(k)}, \mathbf{u}^{(k)})$;

2. $\mathbf{z}^{(k+1)} = \arg\min_{\mathbf{z}} L(\boldsymbol{\theta}^{(k+1)}, \mathbf{z}, \mathbf{u}^{(k)})$;
3. $\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \mathbf{A}\boldsymbol{\theta}^{(k+1)} + \mathbf{B}\mathbf{z}^{(k+1)} - \mathbf{c}$;

where L is the augmented Lagrange function defined as

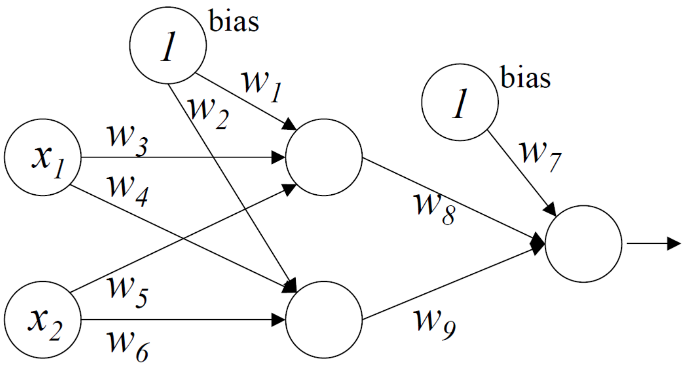
$$L(\boldsymbol{\theta}, \mathbf{z}, \mathbf{u}) = f(\boldsymbol{\theta}) + g(\mathbf{z}) + \mathbf{u}^T(\mathbf{A}\boldsymbol{\theta} + \mathbf{B}\mathbf{z} - \mathbf{c}) + \frac{1}{2}\|\mathbf{A}\boldsymbol{\theta} + \mathbf{B}\mathbf{z} - \mathbf{c}\|_2^2.$$

An advantage of ADMM is that no tuning parameter such as the step size in the gradient algorithm is involved. Please write down the ADMM steps for solving LASSO problem:

$$\min_{\mathbf{w}} \left[\|\mathbf{y} - \mathbf{X}\mathbf{w}\|_2^2 + \lambda \|\mathbf{w}\|_1 \right].$$

(Hint: In order to use ADMM, you have to introduce an auxiliary variable and a suitable constraint. Please give the explicit formulae by solving "arg min" in each step of ADMM.)

✏ **Problem 6** (NeuralNet) Consider a neural network for a binary classification which has one hidden layer as shown in the figure. We use a linear activation function $h(z) = cz$ at hidden units and a sigmoid activation function $g(z) = \frac{1}{1+\exp(-z)}$ at the output unit to learn the function for $\mathrm{P}(y = 1|\mathbf{x}, \mathbf{w})$ where $\mathbf{x} = (x_1, x_2)$ and $\mathbf{w} = (w_1, \ldots, w_9)$.



1. What is the output $\mathrm{P}(y = 1|\mathbf{x}, \mathbf{w})$ from the above neural net? Express it in terms of x_i, c and weights w_i. What is the final classification boundary?
2. Draw a neural net with no hidden layer which is equivalent to the given neural net, and write weights $\tilde{\mathbf{w}}$ of this new neural net in terms of c and w_i.
3. Is it true that any multi-layered neural net with linear activation functions at hidden layers can be represented as a neural net without any hidden layer? Briefly explain your answer.