Machine Learning for Finance (FIN 570)
Midterm Exam

Instructor: Jaehyuk Choi

2023-24 Module 3 (2024. 4. 7.)

The acronyms are defined same as in the class. For example, machine learning (**ML**), logistic regression (**LR**), principal component analysis (**PCA**), support vector machine (**SVM**), neural network (**NN**), etc. If you are not sure, please ask.

1. (8 points) **[Linear Regression]** We collect a set of data ($n = 100$ observations) containing a single predictor and a quantitative response. We then fit a linear regression model to the data, as well as a separate cubic regression, i.e.,

$$Y = \beta_0 + \beta_1 X + \varepsilon \quad \text{and} \quad Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \varepsilon.$$

   (a) Suppose that the true relationship between X and Y is linear, i.e., $Y = \beta_0 + \beta_1 X + \varepsilon$. Consider the training residual sum of squares (RSS) for the linear regression, and also the training RSS for the cubic regression. Would we expect one to be lower than the other, would we expect them to be the same, or is there not enough information to tell? Justify your answer.

   (b) Answer (a) using test rather than training RSS.

   (c) Suppose that the true relationship between X and Y is not linear, but we don't know how far it is from linear. Consider the training RSS for the linear regression, and also the training RSS for the cubic regression. Would we expect one to be lower than the other, would we expect them to be the same, or is there not enough information to tell? Justify your answer.

   (d) Answer (c) using test rather than training RSS.

Solution: This question is from ISRL Exercise 3.7.4.

(a) Even if true relationship is linear, the cubic (polynomial) regression has a lower training RSS than the linear regression because the extra polynomial terms allow a tighter fit against data.

(b) Since the true relationship is linear, simple linear regression would generalize better to unseen data and has lower test RSS. The cubic regression has a higher test RSS as it overfits the traning data.

(c) Cubic regression has lower train RSS than the linear fit because of higher flexibility (more terms). No matter what the underlying true relationship is, more flexible model will tightly fit training data and reduces training RSS.

(d) There is not enough information to tell which RSS would be lower (or higher). It depends on "how far it is from linear." If it is closer to linear than cubic, the linear regression will have lower test RSS than the cubic regression. If it is closer to cubic than linear, the cubic regression will have lower test RSS than the linear regression.

2. (4 points) **[Linear Discriminant Analysis]** Suppose that we wish to predict whether a given stock will issue a dividend this year "Yes" or "No") based on X, last year's percent profit.We examine a large number of companies and discover that the mean value of X for companies that issued a dividend was $\bar{X} = 10$, while the mean for those that didn't was $\bar{X} = 0$. In addition, the variance of X for these two sets of companies was $\hat{\sigma}^2 = 36$. Finally, 80% of companies issued dividends. Assuming that X follows a normal distribution, predict the probability that a company will issue a dividend this year given that its percentage profit was $X = 4$ last year.

Solution: This question is from ISRL Exercise 4.7.7. Let the probability density belonging to each group:

$$\text{Dividend:} \quad A = n\left(\frac{(4-10)}{6}\right) = \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}\frac{(4-10)^2}{36}\right) = \frac{1}{\sqrt{2\pi}}e^{-1/2}$$

$$\text{No dividend:} \quad B = n\left(\frac{(4)}{6}\right) = \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}\frac{4^2}{36}\right) = \frac{1}{\sqrt{2\pi}}e^{-2/9}.$$

Then, the probability of issuing dividend is

$$P(\text{Issue dividend}) = \frac{0.8A}{0.8A + 0.2B} = \frac{0.8e^{-1/2}}{0.8e^{-1/2} + 0.2e^{-2/9}} \approx 0.75.$$

3. (3 points) **[LASSO]** Which of the followings are correct? Justify your answer. The LASSO (i.e., linear regression with L_1 regularization), relative to least squares, is:

   i. More flexible and hence will give improved prediction accuracy when its increase in bias is less than its decrease in variance.

   ii. More flexible and hence will give improved prediction accuracy when its increase in variance is less than its decrease in bias.

   iii. Less flexible and hence will give improved prediction accuracy when its increase in bias is less than its decrease in variance.

   iv. Less flexible and hence will give improved prediction accuracy when its increase in variance is less than its decrease in bias.

Solution: iii. Less flexible and better predictions because of less variance, more bias.

4. (4 points) In the kernel method, the kernel function $K(\cdot,\cdot)$ and the feature map $\phi(\cdot)$ is related by

$$K(\boldsymbol{x}, \boldsymbol{y}) = \phi(\boldsymbol{x})\phi(\boldsymbol{y})^T.$$

For example, if $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^2$, the quadratic kernel,

$$K(\boldsymbol{x}, \boldsymbol{y}) = (1 + \boldsymbol{x}\boldsymbol{y}^T)^2$$

corresponds to the six-dimensional feature map,

$$\phi(\boldsymbol{x}) = (1, \sqrt{2}x_1, \sqrt{2}x_2, x_1^2, \sqrt{2}x_1x_2, x_2^2) \quad \text{for} \quad \boldsymbol{x} = (x_1, x_2).$$

Find the feature map for the *Gaussian* (or radial basis) kernel for $x, y \in \mathbb{R}$:

$$K(x, y) = \exp\left(-\gamma(x-y)^2\right)$$

and show that the feature vector has infinite dimension. Hint:

$$e^x = 1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \cdots + \frac{x^n}{n!} + \cdots .$$

Solution: Since $(x-y) = x^2 - 2xy + y^2$,

$$\exp\left(-\gamma(x-y)^2\right) = \exp\left(-\gamma x^2\right)\exp\left(2\gamma xy\right)\exp\left(-\gamma y^2\right).$$

Using Taylor's expansion,

$$\begin{aligned}\exp\left(2\gamma xy\right) &= 1 + 2\gamma xy + \frac{1}{2}(2\gamma xy)^2 + \cdots \\ &= \left[1, \sqrt{2\gamma}x, \cdots, \frac{(2\gamma)^{n/2}}{\sqrt{n!}}x^n, \cdots\right] \cdot \left[1, \sqrt{2\gamma}y, \cdots, \frac{(2\gamma)^{n/2}}{\sqrt{n!}}y^n, \cdots\right].\end{aligned}$$

Therefore, the feature map is infinite dimensional:

$$\phi(x) = \exp\left(-\gamma x^2\right)\left[1, \sqrt{2\gamma}x, \cdots, \frac{(2\gamma)^{n/2}}{\sqrt{n!}}x^n, \cdots\right].$$

5. (4 points) For a loss function, $J(\boldsymbol{w})$, the gradient descent (GD) method is formulated by

$$\boldsymbol{w}^{(n+1)} = \boldsymbol{w}^{(n)} - \eta\boldsymbol{\nabla}J(\boldsymbol{w}^{(n)}),$$

where η is the learning rate. However, *momentum-based* or *accelerated* GD, formulated by

$$\begin{aligned}\boldsymbol{z}^{(n)} &= \beta\boldsymbol{z}^{(n-1)} + (1-\beta)\boldsymbol{\nabla}J(\boldsymbol{w}^{(n)}), \quad (\boldsymbol{z}^{(1)} = \boldsymbol{\nabla}J(\boldsymbol{w}^{(1)})) \\ \boldsymbol{w}^{(n+1)} &= \boldsymbol{w}^{(n)} - \eta\boldsymbol{z}^{(n)},\end{aligned}$$

is known to be better than the regular GD often. Discuss why. What is the role of the parameter β?

Solution: This question is based on this website. Contrast to the regular GD, momentum-based GD uses the interpolation between the previous direction and the current gradient for the new direction. Therefore, the direction does not change quickly, hence it has momentum.

When the gradient of loss function changes rapidly, oscillation can happen in regular GD. Momentum-based GD can avoid this, thereby converging to the minimum quickly.

The parameter β controls the degree of momentum. If $\beta = 1$, the direction never changes. If $\beta = 0$, the method is reduced to regular GD.

6. (4+3 points) **[Cross-entropy loss function]** We are building a NN model for a classification problem with K classes. From class, we learned that we use the soft max function for the output of the NN. When $\boldsymbol{a} = (a_1, \cdots, a_K)$ is the output from the last hidden layer of the NN network, the output values are given by the softmax function:

$$\hat{y}_k = \phi_k(\boldsymbol{a}) = \frac{e^{a_k}}{\sum_{i=1}^K e^{a_i}},$$

where $\hat{y}_k$ is the probability that the sample belongs to the k-th class.

Now we need to provide the loss function. The cross-entropy loss function is given by

$$J(\boldsymbol{a}) = -\boldsymbol{y} \cdot \log \hat{\boldsymbol{y}} = -\sum_{k=1}^{K} y_k \log \hat{y}_k,$$

where $\boldsymbol{y}$ is the true class represented by one-hot encoding, i.e., $\boldsymbol{y} = (0, \cdots, 1, \cdots, 0)$.

(a) Derive $\dfrac{\partial \hat{y}_k}{\partial a_i}$ by separating cases: (i) $i = k$ and (ii) $i \neq k$.

(b) Derive $\dfrac{\partial}{\partial a_i} J(\boldsymbol{a})$.

Solution: This questions is based on this website.

(a) If (i) $i = k$,

$$\frac{\partial \hat{y}_i}{\partial a_i} = \frac{e^{a_i}}{\sum_{i=1}^{K} e^{a_i}} - \frac{e^{2a_i}}{\left(\sum_{i=1}^{K} e^{a_i}\right)^2} = \frac{e^{a_i}}{\sum_{i=1}^{K} e^{a_i}} \left(1 - \frac{e^{a_i}}{\sum_{i=1}^{K} e^{a_i}}\right) = \hat{y}_i(1 - \hat{y}_i).$$

If (ii) $i \neq k$,

$$\frac{\partial \hat{y}_k}{\partial a_i} = -\frac{e^{a_k}\, e^{a_i}}{\left(\sum_{i=1}^{K} e^{a_i}\right)^2} = -\frac{e^{a_k}}{\sum_{i=1}^{K} e^{a_i}} \cdot \frac{e^{a_i}}{\sum_{i=1}^{K} e^{a_i}} = -\hat{y}_k\, \hat{y}_i.$$

(b) The derivative is given by

$$\begin{aligned} \frac{\partial}{\partial a_i} J(\boldsymbol{a}) &= -\sum_{k=1}^{K} y_k \frac{\partial}{\partial a_i} \log \hat{y}_k = -y_i \frac{1}{\hat{y}_i} \frac{\partial \hat{y}_i}{\partial a_i} - \sum_{k \neq i} y_k \frac{1}{\hat{y}_k} \frac{\partial \hat{y}_k}{\partial a_i} \\ &= -y_i(1 - \hat{y}_i) + \sum_{k \neq i} y_k \hat{y}_i = -y_i + \hat{y}_i \sum_{k=1}^{K} y_k = \hat{y}_i - y_i. \end{aligned}$$

In backpropagation, we defined

$$\delta_i := \frac{\partial}{\partial a_i} J(\boldsymbol{a})$$

for any unit and called it "error" term. We proved that, for the output unit, it is indeed error: $\delta_i = \hat{y}_i - y_i$.