VEF ACADEMY

# MACHINE LEARNING 2018

## Homework 3 Solutions

November 28, 2018

- This homework is due at 2 PM, December 8, 2018.
- Please submit the HW via Google Form (Link will be sent out shortly). Code for programming problems should be submitted as .py files.
- You can discuss HW problems with the instructor, TAs, classmates, or others, but the work you submit must be your own work.

- You may write your answers in Vietnamese or English or a mix of both languages.
- You may consult textbooks and print and online materials.
- Please show all of your work. Answers without appropriate justification will receive very little credit. For programming questions, please submit all the code.

**Problem 1.** *(10 points)* In the "New York City Taxi Fare Prediction" Competition on Kaggle (https://www.kaggle.com/c/new-york-city-taxi-fare-prediction/data), competitors have to predict the cost (in USD) of a taxi ride in New York City given the following features:

- *pickup_datetime* – timestamp value indicating when the taxi ride started.
- *pickup_longitude* – float for longitude coordinate of where the taxi ride started.
- *pickup_latitude* – float for latitude coordinate of where the taxi ride started.
- *dropoff_longitude* – float for longitude coordinate of where the taxi ride ended.
- *dropoff_latitude* – float for latitude coordinate of where the taxi ride ended.
- *passenger_count* – integer indicating the number of passengers in the taxi ride.

Competitors are given a training dataset (*train.csv*) with input features and target *fare_amount* values. They will then have to predict the *fare_amount* for each row of input features in a test set (*test.csv*). Using Tom M. Mitchell's definition of Machine Learning discussed in class (for Parts (a) and (b)),
**(a)** Describe the experience $E$ and the class of tasks $T$ of the algorithms used to solve this problem.
**(b)** Propose a performance measure $P$ that we can use to rank the competitors' submissions.
**(c)** Is this problem a supervised learning one or an unsupervised learning one? Justify your answer.
**(d)** Is this problem a classification or a regression problem? Justify your answer.

**Solution:**

**(a)** *(3 points)* Experience $E$: Learning the training dataset (*train.csv*) with input features and target *fare_amount* values.
Task $T$: Predict the *fare_amount* for each row of input features in the test set (*test.csv*).
**(b)** *(3 points)* A performance measure $P$ that we can use is the root mean-squared error or RMSE. RMSE measures the difference between the predicted

fare amount $\hat{y}$ of a model, and the corresponding ground truth $y$ (the real fare amount).

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}$$

**(c)** *(2 points)* This is a supervised learning problem as we have a training dataset (*train.csv*) with input features and labels (target *fare_amount* values).
**(d)** *(2 points)* This problem can be considered as a regression problem because $y$ (the fare amount) is real-valued (To be precise, the fare amount can only have finite precision, say to \$0.01, but we can ignore this fact.)

**Problem 2.** *(20 points)* In a homework at the Machine Learning class, Toan uses Logistic Regression to classify customers of a Consumer Finance Company (CFC) into two categories: Low-risk (Negative) and High-risk (Positive). Comparing the output of his model with the loan performance data of 1000 customers,Toan ends up with the following confusion matrix (For a description, see, for example https://www.dataschool.io/simple-guide-to-confusion-matrix-terminology/):

| n=1000 | Predicted Low-risk | Predicted High-risk |
|---|---|---|
| Actual Low-risk | 850 | 50 |
| Actual High-risk | 20 | 80 |

**(a)** Calculate True Positive rate, False Positive rate, True Negative rate, and False Negative rate.
**(b)** Discuss the costs (to the CFC) of a False Positive and a False Negative.
**(c)** Calculate the Accuracy, Precision, Recall, and $F_1$ score of this classifier.

For references, see also

- https://en.wikipedia.org/wiki/Precision_and_recall,
- https://en.wikipedia.org/wiki/F1_score.

**Solution:**

**(a)** *(4 points – 1 points for each item)*

$$\begin{aligned}
\text{TP rate} &= \frac{TP}{P} == \frac{80}{20+80} = 0.8 \\
\text{FP rate} &= \frac{FP}{N} = \frac{50}{50+850} \approx 0.0556 \\
\text{TN rate} &= \frac{TN}{N} = \frac{850}{50+850} \approx 0.9444 \\
\text{FN rate} &= \frac{FN}{P} = \frac{20}{20+80} = 0.2
\end{aligned}$$

**(b)** *(8 points – 4 points for each item)*
**False Positive**: A low-risk customer that is predicted as high-risk. In this case a good customer will not be offered a loan. The CFC will lose the net profit they would have gotten had they offered the loan to the customer (The net profit could be approximated by: fee + real interest - costs - cost of capital).

**False Negative**: A high-risk customer that is predicted as low-risk. In this case there is a high probability that the customer will not be able to repay the principal and the interest. The CFC will lose a part of or the whole sum.

**(b)** *(8 points – 2 points for each item)*

$$\begin{aligned}
\text{Accuracy} &= \frac{TP+TN}{P+N} = \frac{80+850}{100+900} = 0.93 \\
\text{Precision} &= \frac{TP}{TP+FP} = \frac{80}{80+50} \approx 0.6154 \\
\text{Recall} &= \text{TP rate} = \frac{TP}{P} = \frac{TP}{TP+FN} = \frac{80}{80+20} = 0.8 \\
F_1 \text{ score} &= 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \approx 0.6957
\end{aligned}$$

**Problem 3** *(35 points)* To prevent overfitting in Linear Regression, we can use a technique called regularization in which we add a penalty term in the loss function to discourage higher values of the coefficients $w_j, j = 1, \ldots, D$. The loss function discussed in class will then become

$$L(\mathbf{w}) = \frac{1}{2}\sum_{i=1}^{N}\left(\mathbf{y}^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)}\right)^2 + \frac{\lambda}{2}\sum_{j=1}^{D} w_j^2 \tag{1}$$

where we use the constant $\lambda$ to adjust the effect of the regularization term. Calculate the gradient $\nabla_{\mathbf{w}} L(\mathbf{w})$.

**Solution:**

Let

$$\begin{aligned} L_1(\mathbf{w}) &= \frac{1}{2}\sum_{i=1}^{N}\left(y^{(i)} - \mathbf{w}^T x^{(i)}\right)^2 \\ R(\mathbf{w}) &= \frac{\lambda}{2}\sum_{j=1}^{D} w_j^2 \end{aligned}$$

From Lecture 5 – Linear Regression, we have that

$$\begin{aligned} L_1(\mathbf{w}) &= \frac{1}{2}(\mathbf{y} - \mathbf{X}\mathbf{w})^T(\mathbf{y} - \mathbf{X}\mathbf{w}) \\ &= \mathbf{y}^T\mathbf{y} - 2\mathbf{w}^T\mathbf{X}^T\mathbf{y} + \mathbf{w}^T\mathbf{X}^T\mathbf{X}\mathbf{w}. \end{aligned}$$

and

$$\nabla_{\mathbf{w}} L_1(\mathbf{w}) = -\mathbf{X}^T\mathbf{y} + \mathbf{X}^T\mathbf{X}\mathbf{w} \tag{2}$$

where

$$\mathbf{y} = \begin{bmatrix} y^{(1)} \\ \ldots \\ y^{(N)} \end{bmatrix}; \quad \mathbf{X} = \begin{bmatrix} \mathbf{x}^{(1)} \\ \ldots \\ \mathbf{x}^{(N)} \end{bmatrix} = \begin{bmatrix} x_0^{(1)} \ldots x_D^{(1)} \\ \ldots \\ x_0^{(N)} \ldots x_D^{(N)} \end{bmatrix}; \quad \mathbf{w} = \begin{bmatrix} w_0 \\ \ldots \\ w_D \end{bmatrix}; \tag{3}$$

As $w_0$ is not in $R(w)$, we have that

$$\frac{\partial R(\mathbf{w})}{\partial w_0} = 0$$

For $w_j, \; j = 1 \ldots D$,

$$\frac{\partial R(\mathbf{w})}{\partial w_j} = \lambda w_j$$

Thus

$$\nabla_{\mathbf{w}} R(\mathbf{w}) = \lambda \begin{bmatrix} 0 \\ w_1 \\ \dots \\ w_D \end{bmatrix} \tag{4}$$

We finally have

$$\nabla_{\mathbf{w}} L(\mathbf{w}) = -\mathbf{X}^T \mathbf{y} + \mathbf{X}^T \mathbf{X} \mathbf{w} + \lambda \begin{bmatrix} 0 \\ w_1 \\ \dots \\ w_D \end{bmatrix}$$

**Problem 4** *(35 points)* In Regularized Logistic Regression, the loss function can be written as

$$L(\mathbf{w}) = -\frac{1}{N}\sum_{i=1}^{N}\left[y^{(i)}log(\sigma(\mathbf{w}^T\mathbf{x}^{(i)})) + (1-y^{(i)})log(1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)}))\right] + \frac{\lambda}{2N}\sum_{j=1}^{D}w_j^2$$

where we use the constant $\lambda$ to adjust the effect of the regularization term. Calculate the gradient $\nabla_{\mathbf{w}}L(\mathbf{w})$.

**Solution:**

Let

$$\begin{aligned} L_1(\mathbf{w}) &= -\frac{1}{N}\sum_{i=1}^{N}\left[y^{(i)}log(\sigma(\mathbf{w}^T\mathbf{x}^{(i)})) + (1-y^{(i)})log(1-\sigma(\mathbf{w}^T\mathbf{x}^{(i)}))\right] && (5)\\ R(\mathbf{w}) &= \frac{\lambda}{2N}\sum_{j=1}^{D}w_j^2 && (6)\end{aligned}$$

We first prove the following result:

Let $\sigma(z) = \frac{1}{1+e^{-z}}$ be the sigmoid function. We then have

$$\frac{d\sigma(z)}{dz} = \sigma(z)(1-\sigma(z)) \tag{7}$$

Indeed,

$$\begin{aligned} \frac{d\sigma(z)}{dz} &= -\frac{-e^{-z}}{(1+e^{-z})^2}\\ &= \frac{1}{1+e^{-z}}\quad\frac{e^{-z}}{1+e^{-z}}\\ &= \sigma(z)(1-\sigma(z))\end{aligned}$$

Let $z^{(i)} = \mathbf{w}^T\mathbf{x}^{(i)}$, Equation (5) can be rewritten as

$$L_1(\mathbf{w}) = -\frac{1}{N}\sum_{i=1}^{N}\left[y^{(i)}log(\sigma(z^{(i)})) + (1-y^{(i)})log(1-\sigma(z^{(i)}))\right]$$

In the above equation for $L_1(\mathbf{w})$, the only functions of $w_j,\ j = 0,\dots,D$, are

$z^{(i)},\ i = 1, \ldots, N$, ($y^{(i)}$ are constants). We have that

$$
\begin{aligned}
\frac{\partial L_1(\mathbf{w})}{\partial w_j} &= -\frac{1}{N}\sum_{i=1}^{N}\left[y^{(i)}\frac{\partial log(\sigma(z^{(i)}))}{\partial w_j} + (1-y^{(i)})\frac{\partial log(1-\sigma(z^{(i)}))}{\partial w_j}\right] \\
&= -\frac{1}{N}\sum_{i=1}^{N}\left[\frac{y^{(i)}}{\sigma(z^{(i)})}\frac{\partial(\sigma(z^{(i)}))}{\partial w_j} + \frac{(1-y^{(i)})}{(1-\sigma(z^{(i)}))}\frac{\partial(1-\sigma(z^{(i)}))}{\partial w_j}\right] \\
&= -\frac{1}{N}\sum_{i=1}^{N}\left[\frac{y^{(i)}}{\sigma(z^{(i)})}\frac{\partial\sigma(z^{(i)})}{\partial w_j} - \frac{(1-y^{(i)})}{1-\sigma(z^{(i)})}\frac{\partial\sigma(z^{(i)})}{\partial w_j}\right] \\
&= -\frac{1}{N}\sum_{i=1}^{N}\left[\frac{y^{(i)}}{\sigma(z^{(i)})}\frac{d\sigma(z^{(i)})}{dz^{(i)}}\frac{\partial z^{(i)}}{\partial w_j} - \frac{(1-y^{(i)})}{1-\sigma(z^{(i)})}\frac{d\sigma(z^{(i)}}{z^{(i)}}\frac{\partial z^{(i)}}{\partial w_j}\right] \\
&= -\frac{1}{N}\sum_{i=1}^{N}\left[\frac{y^{(i)}}{\sigma(z^{(i)})}\frac{d\sigma(z^{(i)})}{dz^{(i)}} - \frac{(1-y^{(i)})}{1-\sigma(z^{(i)})}\frac{d\sigma(z^{(i)}}{z^{(i)}}\right]\left(\frac{\partial z^{(i)}}{\partial w_j}\right) \\
&= -\frac{1}{N}\sum_{i=1}^{N}\left[\frac{y^{(i)}}{\sigma(z^{(i)})}\sigma(z^{(i)})(1-\sigma(z^{(i)})) - \frac{(1-y^{(i)})}{1-\sigma(z^{(i)})}\sigma(z^{(i)})(1-\sigma(z^{(i)}))\right]\left(\frac{\partial z^{(i)}}{\partial w_j}\right) \\
&= -\frac{1}{N}\sum_{i=1}^{N}\left[y^{(i)}(1-\sigma(z^{(i)})) - (1-y^{(i)})\sigma(z^{(i)})\right]\left(\frac{\partial z^{(i)}}{\partial w_j}\right) \qquad (8)
\end{aligned}
$$

Simplifying the first factor in the summation and note that $\frac{\partial z^{(i)}}{\partial w_j} = x_j^{(i)}$, we have that

$$
\begin{aligned}
\frac{\partial L_1(\mathbf{w})}{\partial w_j} &= \frac{1}{N}\sum_{i=1}^{N}\left[\sigma(z^{(i)}) - y^{(i)}\right]\left(x_j^{(i)}\right) \\
&= \frac{1}{N}\sum_{i=1}^{N}\left[\sigma(\mathbf{w}^T\mathbf{x}^{(i)}) - y^{(i)}\right]\left(x_j^{(i)}\right)
\end{aligned}
$$

The above equation can be written in the matrix form as follows

$$\nabla_{\mathbf{w}} L_1(\mathbf{w}) = \frac{1}{N}\mathbf{X}^T(\mu - \mathbf{y})$$

where

$$\mathbf{y} = \begin{bmatrix} y^{(1)} \\ \ldots \\ y^{(N)} \end{bmatrix}; \quad \mu = \begin{bmatrix} \sigma(\mathbf{w}^T\mathbf{x}^{(1)}) \\ \ldots \\ \sigma(\mathbf{w}^T\mathbf{x}^{(N)}) \end{bmatrix}; \quad \mathbf{X} = \begin{bmatrix} \mathbf{x}^{(1)} \\ \ldots \\ \mathbf{x}^{(N)} \end{bmatrix} = \begin{bmatrix} x_0^{(1)} \ldots x_D^{(1)} \\ \ldots \\ x_0^{(N)} \ldots x_D^{(N)} \end{bmatrix}; \quad \mathbf{w} = \begin{bmatrix} w_0 \\ \ldots \\ w_D \end{bmatrix};$$

Similar to the solution to Problem 3, we have that

$$\nabla_{\mathbf{w}} R(\mathbf{w}) = \frac{\lambda}{N} \begin{bmatrix} 0 \\ w_1 \\ \dots \\ w_D \end{bmatrix} \tag{9}$$

Thus, we finally have

$$\nabla_{\mathbf{w}} L(\mathbf{w}) = \frac{1}{N} \mathbf{X}^T (\mu - \mathbf{y}) + \frac{\lambda}{N} \begin{bmatrix} 0 \\ w_1 \\ \dots \\ w_D \end{bmatrix}$$