Course: Machine Learning - Techniques
Practice Assignment Solutions
Week 3

(Common data Q1, Q2, Q3) Rahul observed a set of training and validation root mean squared error (RMSE) values while fitting linear regression models on some non linear sample data by applying polynomial transformation of different degrees

degree	Train RMSE	Validation RMSE
0	0.555	0.538
1	0.277	0.475
2	0.222	0.354
3	0.129	0.323
4	0.128	0.241
5	0.108	0.434
6	0.104	0.591
7	0.103	1.194
8	0.102	1.727

Choose the correct answer from the given options for the following questions.

1. (1 point) The linear regression model with degree ______ polynomial underfits the data.

   (i) 0
   (ii) 1
   (iii) 2

   A. (i) only
   B. Both (i) and (ii)
   C. (i) and (iii)
   D. (ii) only
   E. (i), (ii) and (iii)

   Answer: E

   Solution:
   For models with degree 0, 1 and 2, the training and validation errors are high, hence they underfit the data.

2. (1 point) The linear regression model with degree ______ polynomial overfits the data.

   (i) 6
   (ii) 7
   (iii) 8

   A. (ii) only

B. (iii) only

C. (i), (ii) and (ii)

D. (ii) and (iii)

Answer: C

Solution:
For models with degree 6, 7 and 8, the training error is decreasing and validation errors are increasing, hence the model has learnt training data including noise and poorly generalizes on validation data. Therefore these models overfit the data.

3. (1 point) The linear regression model with degree _______ polynomial smoothly fits the data.

A. 3

B. 4

C. 5

D. Both A and B

Answer: D

Solution:
For models with degree 3 and 4, the training and validation errors are low, hence they smoothly fit the data.

4. (1 point) Consider a feature vector $\mathbf{X} = [1, 3]$. After applying 3^{rd} degree polynomial transformation, which of the following is the corresponding polynomial feature vector?

A. [1, 3, 1, 1, 3, 9]

B. [1, 1, 3, 1, 3, 9, 1, 3, 9, 27]

C. [1, 1, 3, 1, 3, 9]

D. [1, 1, 3, 1, 3, 9, 1, 3, 9, 27, 1, 3, 9, 27, 81]

Answer: B

Solution: As per polynomial transformation, for two features say (x_1, x_2),

$$\phi_3(x_1, x_2) = \left[1, x_1, x_2, x_1^2, x_1x_2, x_2^2, x_1^3, x_1^2x_2, x_1x_2^2, x_2^3\right]$$

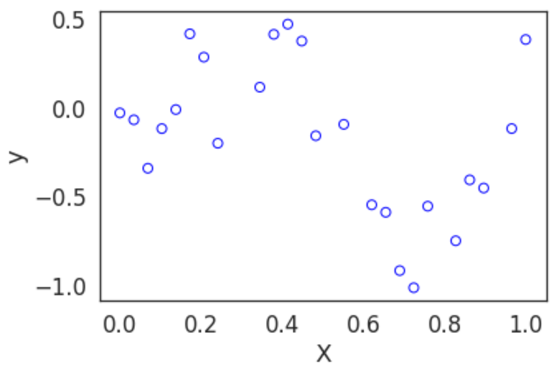
$$\phi_3(1, 3) = [1, 1, 3, 1, 3, 9, 1, 3, 9, 27]$$

5. (1 point) Observe the data points $\mathbf{X}$ and label $\mathbf{y}$ plotted in the scatterplot given below.

Which of the following model will best fit these points?

A. Classic linear regression

B. Polynomial regression of degree 3



C. Polynomial regression of degree 6 with regularization rate $\lambda = 0.5$

Answer: B

Solution: By visually observing the data you can deduce that it is a rsinusoidal distribution. A linear model cannot fit this data as the data is nonlinear. Also, a polynomial regression model of degree 6 might overfit the data as the degree value is high and the data samples are less.
Hence, a polynomial regression model of degree 3 will best fit the data.

(Common data Q6, Q7) Consider the feature matrix $\mathbf{X}_{3\times2}$ and corresponding label vector $\mathbf{y}_{3\times1}$ in the below table.

X	**y**
[1.1 , 1.2]	1
[2.3 , 1.5]	0.6
[2.9 , 2.3]	1.2

For this data, assume regularization rate as 0.01 and compute ridge regression loss and lasso regression loss using weights $\mathbf{w}_{ridge} = \begin{bmatrix} 0.5 \\ -0.5 \end{bmatrix}$ and $\mathbf{w}_{lasso} = \begin{bmatrix} 1 \\ -0.3 \end{bmatrix}$ respectively.

6. (1 point) Which model gives minimal loss value?

   A. Ridge regression model

   B. Lasso regression model

Answer: A

Solution: Ridge loss computation:

$$J(w) = \frac{1}{2}(\mathbf{Xw\text{-}y})^T(\mathbf{Xw\text{-}y}) + \frac{\lambda}{2}\mathbf{w}^T\mathbf{w}$$

Add a dummy column of ones for $\mathbf{X}$ and a 1 for $\mathbf{w}_{ridge}$ intercept.

$$(\mathbf{Xw\text{-}y}) = \begin{bmatrix} 1 & 1.1 & 1.2 \\ 1 & 2.3 & 1.5 \\ 1 & 2.9 & 2.3 \end{bmatrix} \times \begin{bmatrix} 1 \\ 0.5 \\ -0.5 \end{bmatrix} - \begin{bmatrix} 1 \\ 0.6 \\ 1.2 \end{bmatrix} = \begin{bmatrix} -0.05 \\ 0.8 \\ 0.1 \end{bmatrix} \tag{1}$$

$$\mathbf{w}^T\mathbf{w} = \begin{bmatrix} 1 & 0.5 & -0.5 \end{bmatrix} \times \begin{bmatrix} 1 \\ 0.5 \\ -0.5 \end{bmatrix} = 1.5 \tag{2}$$

Substituting these values in loss equation we get;

$$J(w) = 0.5 \times \begin{bmatrix} -0.05 & 0.8 & 0.1 \end{bmatrix} \times \begin{bmatrix} -0.05 \\ 0.8 \\ 0.1 \end{bmatrix} + \lambda \times 0.5 \times 1.5 \tag{3}$$

$$J(w) = 0.3262 + 0.0075 = 0.3337 \tag{4}$$

Lasso loss computation:

$$J(w) = \frac{1}{2}(\mathbf{Xw\text{-}y})^T(\mathbf{Xw\text{-}y}) + \frac{\lambda}{2}\sum_{j=1}^{m}|\mathbf{w}_j|$$

Add a dummy column of ones for $\mathbf{X}$ and a 1 for $\mathbf{w}_{lasso}$ intercept.

$$(\mathbf{Xw\text{-}y}) = \begin{bmatrix} 1 & 1.1 & 1.2 \\ 1 & 2.3 & 1.5 \\ 1 & 2.9 & 2.3 \end{bmatrix} \times \begin{bmatrix} 1 \\ 1 \\ -0.3 \end{bmatrix} - \begin{bmatrix} 1 \\ 0.6 \\ 1.2 \end{bmatrix} = \begin{bmatrix} 0.74 \\ 2.25 \\ 2.01 \end{bmatrix} \tag{5}$$

$$\sum_{j=1}^{m}|\mathbf{w}_j| = 1 + 1 + 0.3 = 2.3 \tag{6}$$

Substituting these values in loss equation we get;

$$J(w) = 0.5 \times \begin{bmatrix} 0.74 & 2.25 & 2.01 \end{bmatrix} \times \begin{bmatrix} 0.74 \\ 2.25 \\ 2.01 \end{bmatrix} + \lambda \times 0.5 \times 2.3 \tag{7}$$

$$J(w) = 4.8251 + 2.30 = 4.8366 \tag{8}$$

7. (1 point) Enter the value of minimum loss.

   Answer: 0.3337, range: 0.332,0.335

(Common data Q8,Q9,Q10) Given below is the code snippet used for generating combinations of polynomial features.

```
import itertools
import functools
def get_combinations(x,degree):
  return itertools.combinations_with_replacement(x,degree)
def compute_new_feature(items):
  return functools.reduce(lambda x, y: x * y, items)
```

8. (1 point) What is the output of the code shown below?

```
{items:compute_new_feature(items) for items in get_combinations
    ([1,2],2)}
```

   A. (1, 1): 2, (1, 2): 3, (2, 2): 4

   B. (1, 1): 1, (1, 2): 2, (2, 2): 4

   C. (1, 1): 1, (1, 2): 2

   D. (1, 2): 2, (2, 1): 2

   E. (1, 1): 1, (2, 2): 4

   Answer: B

   Solution: Code output is (1, 1): 1, (1, 2): 2, (2, 2): 4

9. (1 point) What is the output of the code shown below?

```
{items:compute_new_feature(items) for items in get_combinations
    ([4,5],3))}
```

   A. (4, 4, 4): 12, (4, 4, 5): 13, (4, 5, 5): 14, (5, 5, 5): 14

   B. (4, 4): 16, (4, 5): 20, (5, 5): 25

   C. (4, 4, 4): 64, (4, 4, 5): 80, (4, 5, 5): 100, (5, 5, 5): 125

   D. (4, 5, 4): 64, (4, 5, 5): 80, (4, 5, 4): 100, (4, 5, 5): 125

   E. (3, 3, 3): 27, (5, 3, 3): 45, (5, 5, 3): 75, (5, 5, 5): 125

   Answer: C

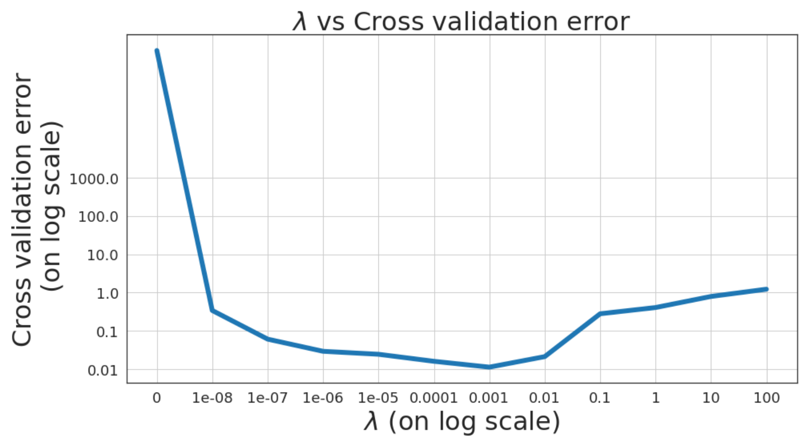
   Solution: Code output is (4, 4, 4): 64, (4, 4, 5): 80, (4, 5, 5): 100, (5, 5, 5): 125

10. (1 point) Ram trained a ridge regression model with various regularization rate λ values for some data by cross-validation procedure and selected a best λ by plotting the cross-validation errors (RMSE) as given in the plot below.

    From the above plot, which λ value did Ram select as the best regularization rate for the model to smoothly fit the data?

    Answer: 0.001

    Solution: The cross validation error is minimum for $\lambda = 0.001$.



11. (1 point) State True or False: The number of elements in weight vector for a polynomial regression model corresponds to the degree of the polynomial used for transformation.

    A. True

    B. False

Answer: B

Solution: False. Because it is equivalent to the number of features after polynomial transformation with specified degree.

12. (1 point) Given below are code snippets of different loss functions.

(i)
```
    def loss(features, labels, weights):
        e = predict(features, weights) - labels
        temp = np.transpose(e) @ e
        loss = (1/2) * temp
        return loss
```

(ii)
```
    def loss(features, labels, weights):
        e = predict(features, weights) - labels
        temp = np.transpose(e) @ e
        temp1 = np.transpose(weights) @ weights
        reg_rate = 0.01
        loss = (1/2) * temp + reg_rate*(1/2)* temp1
        return loss
```

Choose the correct statement from the following.

A. (i) - linear regression loss function and (ii) - lasso regression loss function

B. (i) - linear regression loss function and (ii) - ridge regression loss function

C. (i) - ridge regression loss function and (ii) - linear regression loss function

D. (i) - lasso regression loss function and (ii) - ridge regression loss function

Answer: B

Solution: Based on the implementation of loss, we can identify that (i) corresponds to linear regression loss function and (ii) corresponds to ridge regression loss function.

13. (1 point) (Multiple select) Choose the true statement(s):

A. Lasso regression assigns sparse weights. That is, it assigns non zero weights to only important features and zero weights to unimportant features.

B. Ridge regression can be useful if there are several features that are useless.

C. The penalty term added to lasso loss function is absolute value of magnitude of weight and ridge loss function is squared norm of weight.

Answer: A,C

Solution: A and C are true. B is false, because ridge regression does not assign sparse weights.