Course: Machine Learning - Techniques
Graded Assignment Solutions
Week 3

---

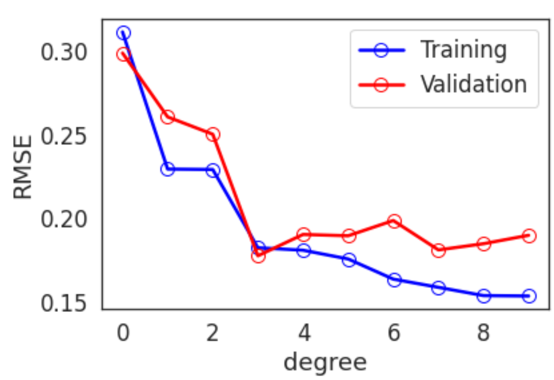
1. (1 point) If a data sample has 5 features, what will be the number of features after applying 4th degree polynomial transformation?

   Answer: 126

   Solution:
   Formula to calculate the number of features including the dummy feature is as follows: If there are n features, while fitting a N^{th} degree polynomial, the number of features after transformation is

   $$= \frac{(N+1)(N+2)...(N+n)}{n!}$$
   $$= \frac{(4+1)(4+2)(4+3)(4+4)(4+5)}{5!}$$
   $$= \frac{5*6*7*8*9}{5*4*3*2} = 126$$

2. (1 point) Anjali has some non linear data, she split the data into training and validation sets and then trained 10 polynomial regression models with degrees 0 to 9 on the training set. She used the trained models to make predictions on the respective test set. She used RMSE as the metric to find out the best model and the plot is given below.



   Which of the following models gives a reasonable fit?

   A. Model using degree 2 polynomial.

   B. Model using degree 3 polynomial.

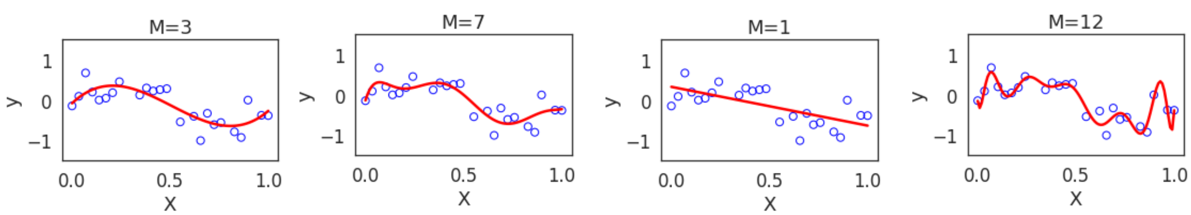
   C. Model using degree 4 polynomial.

   D. Model using degree 5 polynomial.

   Answer: B

Solution:
By inspecting the figure it can be seen that the Model using degree 3 polynomial has low value for both training and validation errors. For other models with degree 0, 1 and 2, both errors are high which means they underfit the data and models with degree 4 to 9, validation error rises while training errors are decreasing which means they slowly begin to overfit the data.

3. (1 point) (Multiple select) Ram trained four polynomial regression models of degree 1, 3, 7 and 12.The plots of data points (blue color) and the predicted models (red color) are provided below.



Which of the following is/are the true statement(s)?

A. Model with M=1 underfits the data.

B. Model with M=1 smoothly fits the data.

C. Model with M=3 underfits the data.

D. Model with M=3 smoothly fits the data.

E. Model with M=7 smoothly fits the data.

F. Model with M=7 overfits the data.

G. Model with M=12 smoothly fits the data.

H. Model with M=12 overfits the data.

Answer: A,D,F,H

Solution:
By inspecting the figure it can be seen that:

- Model with M=1 underfits the data.
- Model with M=3 smoothly fits the data.
- Model with M=7 overfits the data.
- Model with M=12 overfits the data.

(Common data Q4-Q7) Given below is the code snippet used for polynomial transformation of features.

```python
import itertools
import functools
import numpy as np
def polynomial_transform(x, degree):
  if x.ndim == 1:
```

```
    x = x[:, None]
  x_t = x.transpose()
  features = [np.ones(len(x))]
  for degree in range(1, degree + 1):
    for items in itertools.combinations_with_replacement(x_t, degree):
      features.append(functools.reduce(lambda x, y: x * y, items))
  return np.asarray(features).transpose()
```

4. (1 point) Which of the following is the output of the code snippet given below?

```
polynomial_transform(np.array([[5]]), degree=0)
```

A. array([[5.]])

B. array([[0.]])

C. array([[1.]])

D. array([[5,1]])

Answer: C

Solution:
Code output is array([[1.]]).

5. (1 point) Which of the following is the output of the code snippet given below?

```
polynomial_transform(np.array([[5,4]]), degree=2)
```

A. array([[1., 5., 4., 25., 20., 16.]])

B. array([[1., 5., 4., 25., 20., 16., 125., 100., 80., 64., 625.,500., 400., 320., 256.]])

C. array([[1., 5., 4., 25., 20., 16., 125., 100., 80., 64.]])

D. array([[1., 5., 4.]])

Answer: A

Solution:
Code output is array([[1., 5., 4., 25., 20., 16.]])

6. (1 point) Which of the following is the output of the code snippet given below?

```
polynomial_transform(np.array([[1,2],[7,9]]), degree=2)
```

A. array([[1., 1., 2.], [1., 7., 9.]])

B. array([[1., 2., 2., 1., 2., 2.], [1., 7., 7., 1., 9., 9.]])

C. array([[1., 2., 2.], [5., 7., 9.]])

D. array([[1., 1., 2., 1., 2., 4.], [1., 7., 9., 49., 63., 81.]])

Answer: D

Solution:
Code output is array([[1., 1., 2., 1., 2., 4.], [1., 7., 9., 49., 63., 81.]])

7. (1 point) Which of the following is the output of the code snippet given below?

```
polynomial_transform(np.array([[1],[5],[2]]), degree=2)
```

A. array([[1., 1., 1.], [1., 5., 25.], [1., 2., 4.]])

B. array([[1., 1., 1.], [5., 5., 5.], [2., 2., 2.]])

C. array([[1., 1., 1., 1.], [1., 5., 25., 125.], [1., 2., 4., 8.]])

D. array([[1., 1.], [1., 5.], [1., 2.]])

Answer: A

Solution:
Code output is array([[1., 1., 1.], [1., 5., 25.], [1., 2., 4.]])

8. (1 point) Consider the code snippet given below for generating non-linear training data:

```
def create_nonlin_training_set(func, sample_size, std):
    x = np.linspace(0, 1, sample_size)
    y = func(x) + np.random.normal(scale=std, size=x.shape)
    return x, y
def nonlin(x):
    return np.sin(2 * np.pi * x)
```

Using this function, some data points are generated with different noise values with a particular code.

```
    X, y = create_nonlin_training_set(nonlin, 50, std)
```

In the table given below, the plots are given and below the plots, a list of std values are given.

Fig.1

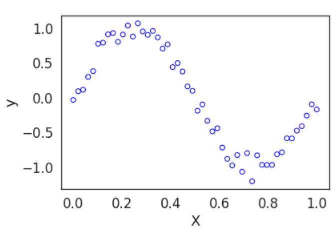


Fig.2

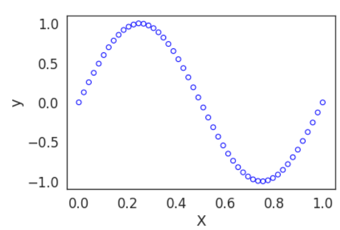


Fig.3

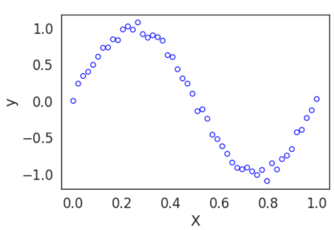
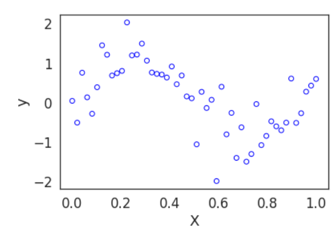


Fig.4



(i) std = 0.001

(ii) std = 0.05

(iii) std = 0.5

(iv) std = 0.1

Match the data plots with its correct std value.

A. (Fig.1 - (ii)), (Fig.2 - (iv)), (Fig.3 - (iii)), (Fig.4 - (i))

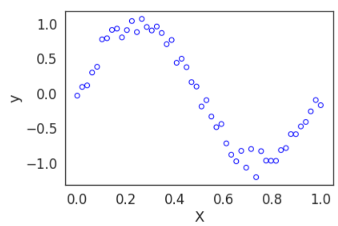
B. (Fig.1 - (ii)), (Fig.2 - (i)), (Fig.3 - (iv)), (Fig.4 - (iii))

C. (Fig.1 - (iv)), (Fig.2 - (i)), (Fig.3 - (ii)), (Fig.4 - (iii))

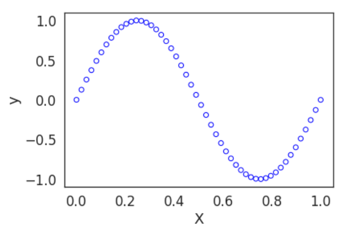
D. (Fig.1 - (iii)), (Fig.2 - (iv)), (Fig.3 - (i)), (Fig.4 - (ii))

Answer: C

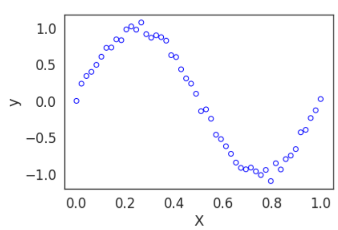
Solution:
The std parameter controls the amount of noise added to the sine function. Lesser the std value, lesser is the noise and the data points resembles sine curve.
By inspecting the figures, we can conclude that:



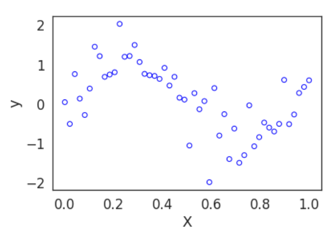
- Fig.1 - std = 0.1



- Fig.2 - std = 0.001



- Fig.3 - std = 0.05



- Fig.4 - std = 0.5

9. (1 point) (Multiple select) Consider that you have some non-linear data generated from a $sin(2 * \pi * x)$ function $0 \leq x \leq 1$. Choose the False statement(s) from the following.

A. The polynomial regression model of higher degree (say 8) adds a smooth fitting to non-linear data when there are large samples (say 1000).

B. The polynomial regression model of higher degree (say 8) overfits non-linear data when there are large samples (say 1000).

C. The polynomial regression model of lower degree (say 1) adds a smooth fitting to non-linear data when there are large samples (say 1000).

D. The polynomial regression model of lower degree (say 1) underfits non-linear data when there are large samples (say 1000).

Answer: B,C

Solution:

- B is false. Because polynomial regression model of higher degree (say 8) can fit the non-linear data when there are large samples (say 1000).
- C is false. Because polynomial regression model of lower degree (say 1) can underfit non-linear data eventhough there are large samples (say 1000).

10. (1 point) (Multiple select) Given below are two weight vectors that are obtained from training two polynomial regression models with degree 7 on two non-linear datasets generated with different sample size.

(i) array([3.932, -66.865, 534.784, -2042.464, 4161.895, -4746.534, 2874.862, -719.776])

(ii) array([0.026, 5.711, 6.288, -73.562, 73.387, 29.433, -55.949, 14.617])

Choose the correct statement that relates the weight vector with the sample data from the following.

A. Weight vector in (i) corresponds to the model trained on data with sample size = 20.

B. Weight vector in (i) corresponds to the model trained on data with sample size = 1000.

C. Weight vector in (ii) corresponds to the model trained on data with sample size = 20.

D. Weight vector in (ii) corresponds to the model trained on data with sample size = 1000.

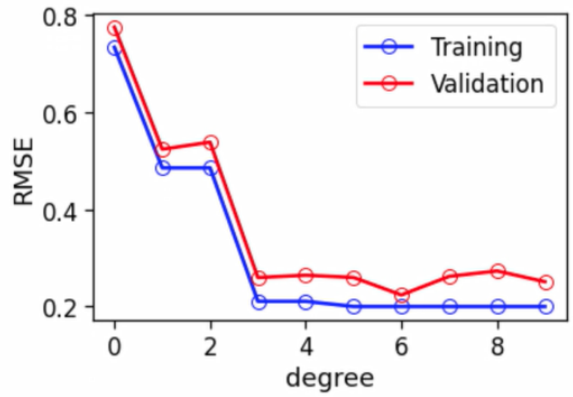
Answer: A,D

Solution:

- A is true. Because when there are less number of samples (20) the coefficients value for weight vector will be large.
- D is true. Because when there are more number of samples (1000) the coefficients value for weight vector will be less.

11. (1 point) Vikram generated some non-linear data and split the data into training and validation sets. He trained different polynomial regression models with varying degrees

and plotted the model selection curve using train and validation errors. From the model selection curve given in the below figure, select the degree of the model that gives the best fit.



A. 3

B. 6

C. 7

D. 9

Answer: B

Solution: From the degree vs RMSE plot, it can be observed that the value of RMSE is less for training and validation when degree is 6.

(Common data Q12, Q13) Consider the feature matrix $\mathbf{X}_{2\times 3}$ and corresponding label vector $\mathbf{y}_{2\times 1}$ in the below table. Let regularization rate, $\lambda = 0.1$.

X	**y**
[1 , 2 , 3]	2
[3 , 2, 4]	4

12. (1 point) Compute ridge regression loss using weight $\mathbf{w}_{ridge} = \begin{bmatrix} 0.1 \\ -0.2 \\ 0.3 \end{bmatrix}$. Enter the value of loss.

Answer: 1.9420, range: 1.8,2.2

Solution:
Ridge loss computation:

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

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

$$(\mathbf{Xw}\text{-}\mathbf{y}) = \begin{bmatrix} 1 & 1 & 2 & 3 \\ 1 & 3 & 2 & 4 \end{bmatrix} \times \begin{bmatrix} 1 \\ 0.1 \\ -0.2 \\ 0.3 \end{bmatrix} - \begin{bmatrix} 2 \\ 4 \end{bmatrix} = \begin{bmatrix} -0.4 \\ -1.9 \end{bmatrix} \tag{1}$$

$$\mathbf{w}^T\mathbf{w} = \begin{bmatrix} 1 & 0.1 & -0.2 & 0.3 \end{bmatrix} \times \begin{bmatrix} 1 \\ 0.1 \\ -0.2 \\ 0.3 \end{bmatrix} = 1.14 \tag{2}$$

Substituting these values in loss equation we get;

$$J(w) = 0.5 \times \begin{bmatrix} -0.4 & -1.9 \end{bmatrix} \times \begin{bmatrix} -0.4 \\ -1.9 \end{bmatrix} + 0.5 \times 0.1 \times 1.14 \tag{3}$$

$$J(w) = 1.885 + 0.057 = 1.942 \tag{4}$$

13. (1 point) Compute lasso regression loss using weight $\mathbf{w}_{ridge} = \begin{bmatrix} 0.01 \\ -0.5 \\ 0 \end{bmatrix}$. Enter the value of loss.

Answer: 9.936, range: 9.5,10.5

Solution:
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.

$$(\mathbf{Xw\text{-}y}) = \begin{bmatrix}1 & 1 & 2 & 3\\ 1 & 3 & 2 & 4\end{bmatrix} \times \begin{bmatrix}1\\ 0.01\\ -0.5\\ 0\end{bmatrix} - \begin{bmatrix}2\\ 4\end{bmatrix} = \begin{bmatrix}-1.99\\ -3.97\end{bmatrix} \tag{5}$$

$$\sum_{j=1}^{m}|\mathbf{w}_j| = 1 + 0.01 + 0.5 + 0 = 1.51 \tag{6}$$

Substituting these values in loss equation we get;

$$J(w) = 0.5 \times \begin{bmatrix}-1.99 & -3.97\end{bmatrix} \times \begin{bmatrix}-1.99\\ -3.97\end{bmatrix} + 0.5 \times 0.1 \times 1.51 \tag{7}$$

$$J(w) = 9.8605 + 0.0755 = 9.936 \tag{8}$$

14. (1 point) (Multiple select) Rajesh has a non-linear data generated from $sin(2 * \pi * x)$ function where $0 \leq x \leq 1$ with 50 samples. He wants to train ridge regression model by applying 9th degree polynomial transformation on his dataset. Provided some criteria on regularization rate, choose the True statement(s):

A. Model with regularization rate (say 0.001) underfits the data.

B. Model with regularization rate (say 0.01) smoothly fits the data.

C. Model with regularization rate (say 1) smoothly fits the data.

D. Model with regularization rate (say 10) underfits the data.

Answer: B,D

Solution:

- B is true. Because, for nonlinear data with 50 samples, if you apply 9^{th} degree polynomial, model with regularization rate = 0.01 smoothly fits the data as λ is less though the degree is high.
- D is true. Because, for nonlinear data with 50 samples, if you apply 9^{th} degree polynomial, model with regularization rate = 10 will underfit the data as λ is high and also degree is high.