EE2211 Tutorial 6

(Ridge Regression in Dual Form)
Question 1:

Derive the solution for linear ridge regression in dual form (see Lecture 6 notes page 16).

1 a) $(X^TX + \lambda I)w = X^Ty$

$X^TXw + \lambda I w = X^Ty$

$\lambda w = X^Ty - X^TXw$

$\lambda w = X^T(y - Xw)$

$w = X^T \frac{(y-Xw)}{\lambda}$

$\therefore w = X^Ta \qquad a = \frac{(y-Xw)}{\lambda}$

$\lambda a = y - Xw$

$\lambda a = y - XX^Ta$

$XX^Ta + \lambda a = y$

$(XX^T + \lambda)a = y$

$a = (XX^T+\lambda)^{-1}y$

$\therefore w = X^Ta = X^T(XX^T+\lambda)^{-1}y$

(Polynomial Regression, 1D data)
Question 2:

Given the following data pairs for training

$$\{x = -10\} \rightarrow \{y = 5\}$$
$$\{x = -8\} \rightarrow \{y = 5\}$$
$$\{x = -3\} \rightarrow \{y = 4\}$$
$$\{x = -1\} \rightarrow \{y = 3\}$$
$$\{x = 2\} \rightarrow \{y = 2\}$$
$$\{x = 8\} \rightarrow \{y = 2\}$$

$$\begin{bmatrix} 1 & x_1 & x_1^2 & x_1^3 \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ w_3 \end{bmatrix}$$

(a) Perform a 3rd-order polynomial regression and sketch the result of line fitting.

(b) Given a test point $\{x = 9\}$ predict y using the polynomial model.

(c) Compare this prediction with that of a linear regression.

constant input a → order

a) $\overset{\text{constant}}{w_0} + \overbrace{w_1x_1 + w_2x_2 + w_3x_3}^{C(3,1)} + \overbrace{w_{11}x_1^2 + w_{22}x_2^2 + w_{33}x_3^2 + w_{12}x_1x_2 + w_{13}x_1x_3 + w_{23}x_2x_3}^{C(3,2)} + w_{111}x_1^3 + w_{222}x_2^3 + w_{333}x_3^3 + w_{122}x_1x_2^2 + w_{133}x_1x_3^2 + w_{211}x_2x_1^2 + w_{233}x_2x_3^2 + w_{311}x_3x_1^2 + w_{322}x_3x_2^2 + w_{123}x_1x_2x_3$

(Polynomial Regression, 3D data, Python)
Question 3:

(a) Write down the expression for a 3rd order polynomial model having a 3-dimensional input.

(b) Write down the **P** matrix for this polynomial given $\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & -1 & 1 \end{bmatrix}$.

(c) Given $\mathbf{y} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, can a unique solution be obtained in dual form? If so, proceed to solve it.

(d) Given $\mathbf{y} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, can the primal ridge regression be applied to obtain a unique solution? If so, proceed to solve it.

(e)

(Binary Classification, Python)
Question 4:
Given the training data:

$$\{x = -1\} \rightarrow \{\, y = class1 \,\}$$
$$\{x = \ \ 0\,\} \rightarrow \{\, y = class1 \,\}$$
$$\{x = 0.5\} \rightarrow \{\, y = class2 \,\}$$
$$\{x = 0.3\} \rightarrow \{\, y = class1 \,\}$$
$$\{x = 0.8\} \rightarrow \{\, y = class2 \,\}$$

Predict the class label for $\{x = -0.1\}$ and $\{x = 0.4\}$ using linear regression with signum discrimination.

(Multi-Category Classification, Python)
Question 5:
Given the training data:

5 samples
$$\begin{cases} \{x = -1\} \rightarrow \{ y = class1 \} \\ \{x = \ \ 0\ \} \rightarrow \{ y = class1 \} \\ \{x = 0.5\} \rightarrow \{ y = class2 \} \\ \{x = 0.3\} \rightarrow \{ y = class3 \} \\ \{x = 0.8\} \rightarrow \{ y = class2 \} \end{cases}$$

(a) Predict the class label for $\{x = -0.1\}$ and $\{x = 0.4\}$ based on linear regression towards a one-hot encoded target.

one input feature, x

(b) Predict the class label for $\{x = -0.1\}$ and $\{x = 0.4\}$ using a polynomial model of 5th order and a one-hot encoded target.

$f(x) = w_0 + w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4 + w_5 x^5$

(Multi-Category Classification, Python)
Question 6 (continued from Q3 of Tutorial 2):
Get the data set "`from sklearn.datasets import load_iris`". Use Python to perform the following tasks.

(a) Split the database into two sets: 74% of samples for training, and 26% of samples for testing. Hint: you might want to utilize `from sklearn.model_selection import train_test_split` for the splitting.

(b) Construct the target output using one-hot encoding.

(c) Perform a linear regression for classification (without inclusion of ridge, utilizing one-hot encoding for the learning target) and compute the number of test samples that are classified correctly.

(d) Using the same training and test sets as in above, perform a 2nd order polynomial regression for classification (again, without inclusion of ridge, utilizing one-hot encoding for the learning target) and compute the number of test samples that are classified correctly. Hint: you might want to use `from sklearn.preprocessing import PolynomialFeatures` for generation of the polynomial matrix.

d) $X = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 3 & 3 \end{bmatrix}$ (row 3 is 3 times of row 1)

c) unique solution: matrix must be square, full rank

inf. solns, more unknowns than equations, more cols than rows (chap 4)

Question 7

MCQ: there could be more than one answer. Given $\underbrace{\text{three samples}}_{\text{3 rows}}$ of $\underbrace{\text{two-dimensional}}_{\text{2 inputs}}$ data points $\mathbf{X} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 3 & 3 \end{bmatrix}$ with corresponding target vector $\mathbf{y} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$. Suppose you want to use a full third-order polynomial model to fit these data. Which of the following is/are true?

b) $X = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 3 & 3 \end{bmatrix}$

$P = \begin{bmatrix} 1 & \cdots \\ 1 & \cdots \end{bmatrix}$

a) The polynomials model has 10 parameters to learn
b) The polynomial learning system is an under-determined one
c) The learning of the polynomial model has infinite number of solutions
d) The input matrix **X** has linearly dependent samples
e) None of the above

$m \times d$

2 rows x 10 cols

$(m < d) \rightarrow$ wide matrix, underdetermined

num of input features $\rightarrow d$, order

a) $C(2,3) = \text{constant} + C(2,1) + C(2,2) + C(2,3)$
$= 1 + 2 + 3 + 4$
$= 10$

$P = [1, x_1, x_2, x_1^2, x_2^2, x_1 x_2, x_1^3, x_2^3, x_1 x_2^2, x_1^2 x_2]$

Question 8

MCQ: there could be more than one answer. Which of the following is/are true?

a) The polynomial model can be used to solve problems with nonlinear decision boundary.

b) The ridge regression cannot be applied to multi-target regression.

c) The solution for learning feature $\mathbf{X}$ with target $\mathbf{y}$ based on linear ridge regression can be written as $\hat{\mathbf{w}} = (\mathbf{X}^T\mathbf{X}+\lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$ for $\lambda > 0$. As λ increases, $\hat{\mathbf{w}}^T\hat{\mathbf{w}}$ decreases.

d) If there are four data samples with two input features each, the full second-order polynomial model is an over-determined system.

four data samples $\downarrow$ 4 rows

two input features $\downarrow$ 2 cols (x_1, x_2)

a) True. Polynomial models can capture non-linear decision boundaries by introducing higher degree terms

b) False. Ridge regression can be applied to multi-target regression. In multi-target regression, y is a matrix (instead of a vector) and ridge regression works by applying L2 regularization to all target variables.

c) True.

Using a linear model:

$$\min_{\mathbf{w}} (\mathbf{X}\mathbf{w}-\mathbf{y})^T(\mathbf{X}\mathbf{w}-\mathbf{y}) + \lambda \mathbf{w}^T\mathbf{w}$$

Solution:

$$\frac{\partial}{\partial \mathbf{w}}\left((\mathbf{X}\mathbf{w}-\mathbf{y})^T(\mathbf{X}\mathbf{w}-\mathbf{y}) + \lambda \mathbf{w}^T\mathbf{w}\right) = \mathbf{0}$$

$$\Rightarrow 2\mathbf{X}^T\mathbf{X}\mathbf{w} - 2\mathbf{X}^T\mathbf{y} + 2\lambda\mathbf{w} = \mathbf{0}$$

$$\Rightarrow \mathbf{X}^T\mathbf{X}\mathbf{w} + \lambda\mathbf{w} = \mathbf{X}^T\mathbf{y}$$

$$\Rightarrow (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})\mathbf{w} = \mathbf{X}^T\mathbf{y}$$

where **I** is the *dxd* identity matrix

d)
$$X = \overbrace{\begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \\ x_{31} & x_{32} \\ x_{41} & x_{42} \end{bmatrix}}^{\text{2 input features}} \Big\} \text{ 4 samples}$$

2x4

(mxd)

<u>2nd order</u>

$$P = [\,1 \quad x_1 \quad x_2 \quad x_1^2 \quad x_2^2 \quad x_1x_2\,] \rightarrow \overbrace{\begin{bmatrix} 1 & \cdot & \cdot & \cdot & \cdot \\ 1 & \cdot & \cdot & \cdot & \cdot \\ 1 & \cdot & \cdot & \cdot & \cdot \\ 1 & \cdot & \cdot & \cdot & \cdot \end{bmatrix}}^{6} \Big\} 4$$

(mxd) : (4x6)

↳ wide matrix

(underdetermined)

(right-inverse)

$$Xw = y$$

- X → Input (feature matrix, shape: $m \times n$)
- w → Solution (weights/parameters, shape: $n \times 1$)
- y → Target (output vector, shape: $m \times 1$)

This is a standard **linear regression** equation, where we try to find w that minimizes the error between Xw and y.