Machine Learning

Homework 03

Due on 12/22/2023

Min-Kuan Chang

minkuanc@nchu.edu.tw

EE, College of EECS

Problem 01

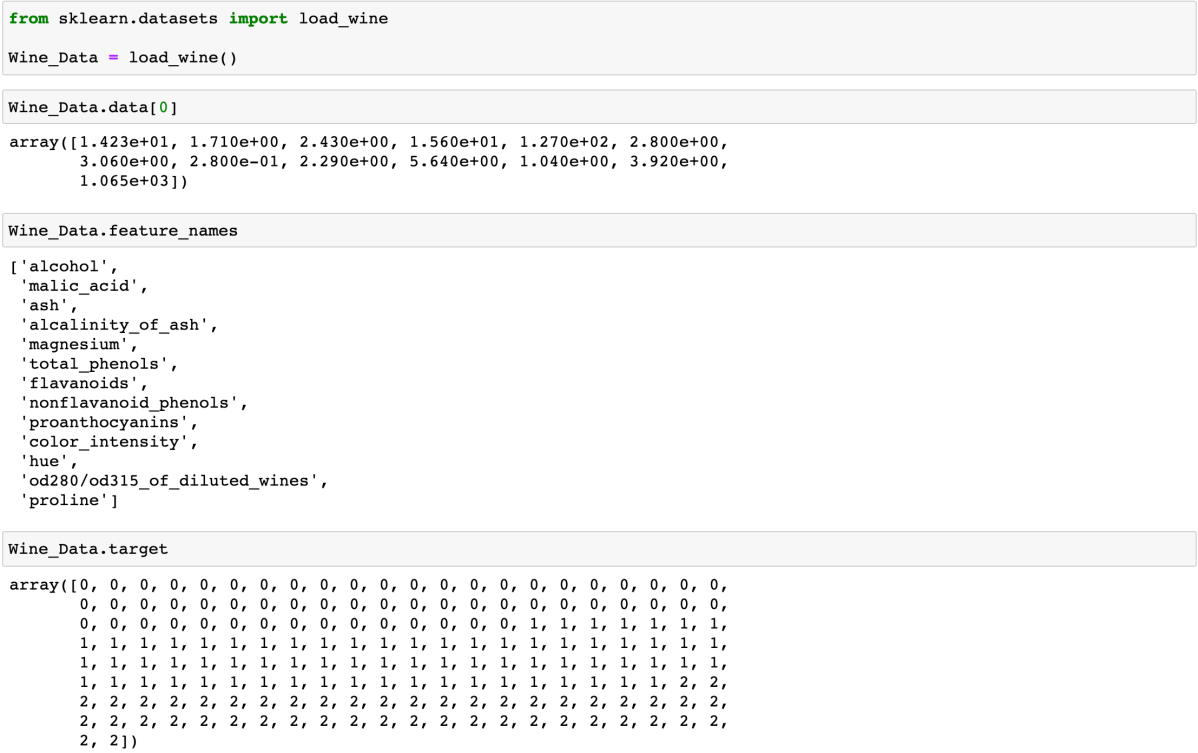
- You are given a data set column2Cweka.csv (file with two class labels)
- In this data set, each patient is represented in the data set by six biomechanical attributes derived from the shape and orientation of the pelvis and lumbar spine (each one is a column):
  - pelvic incidence
  - pelvic tilt
  - lumbar lordosis angle
  - sacral slope
  - pelvic radius
  - grade of spondylolisthesis
- Pleas use KNN to build a model to classify a patient into either normal or abnormal
  - See how number of neighbors affects the accuracy and determine the best number of neighbors

Problem 01

- Pleas use KNN to build a model to classify a patient into either normal or abnormal
  - See how number of neighbors affects the accuracy and determine the best number of neighbors
- Please use Random Forest to build a classification model
  - See how the number of estimator in the Random Forest affects the accuracy and determine the best choice of the number of estimator

Problem 02

- The wine data set
  - **from sklearn.datasets import load_wine**
- Develop a Decision Tree Model to classify the wine
  - See how the max depth affects the accuracy
  - Draw the feature importance under the best max depth



```
from sklearn.datasets import load_wine

Wine_Data = load_wine()
```

```
Wine_Data.data[0]
```

```
array([1.423e+01, 1.710e+00, 2.430e+00, 1.560e+01, 1.270e+02, 2.800e+00,
       3.060e+00, 2.800e-01, 2.290e+00, 5.640e+00, 1.040e+00, 3.920e+00,
       1.065e+03])
```

```
Wine_Data.feature_names
```

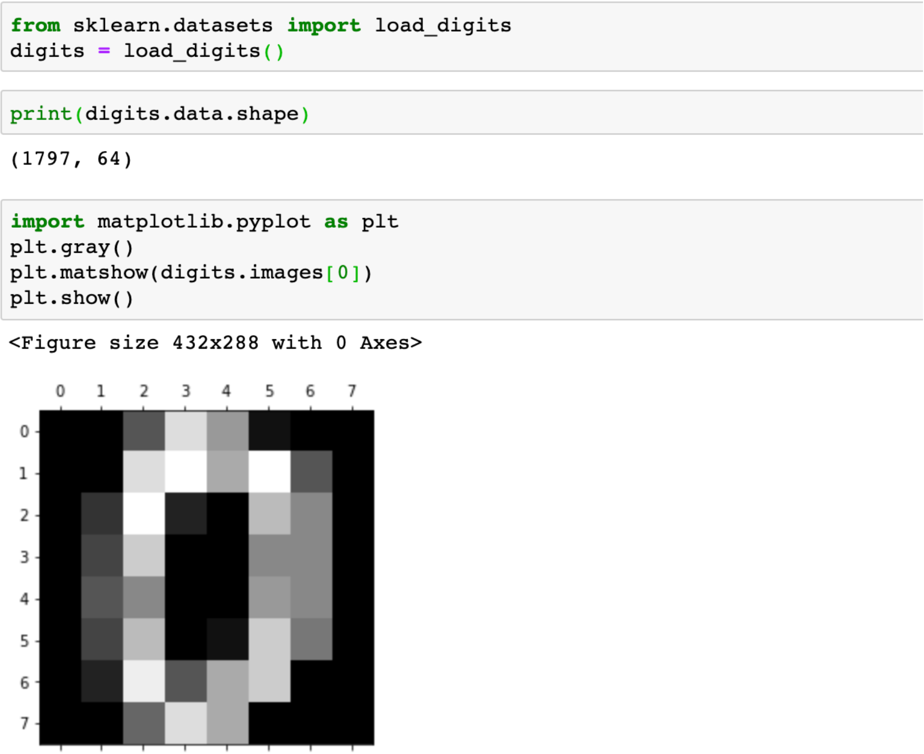
```
['alcohol',
 'malic_acid',
 'ash',
 'alcalinity_of_ash',
 'magnesium',
 'total_phenols',
 'flavanoids',
 'nonflavanoid_phenols',
 'proanthocyanins',
 'color_intensity',
 'hue',
 'od280/od315_of_diluted_wines',
 'proline']
```

```
Wine_Data.target
```

```
array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
       2, 2])
```

Problem 03

- The Digit Dataset
  - This dataset is made up of 1797 8x8 images
  - Each image is of a hand-written digit
  - In order to utilize an 8x8 figure like this, we'd have to first transform it into a feature vector with length 64
  - **from sklearn.datasets import load_digits**



```
from sklearn.datasets import load_digits
digits = load_digits()
```

```
print(digits.data.shape)
```

```
(1797, 64)
```

```
import matplotlib.pyplot as plt
plt.gray()
plt.matshow(digits.images[0])
plt.show()
```

```
<Figure size 432x288 with 0 Axes>
```

Problem 03

- Develop a Random Forest model to classify the hand-written digits
  - See how the number of estimators affects the accuracy
  - Draw the feature importance under the best number of estimators

Problem 04

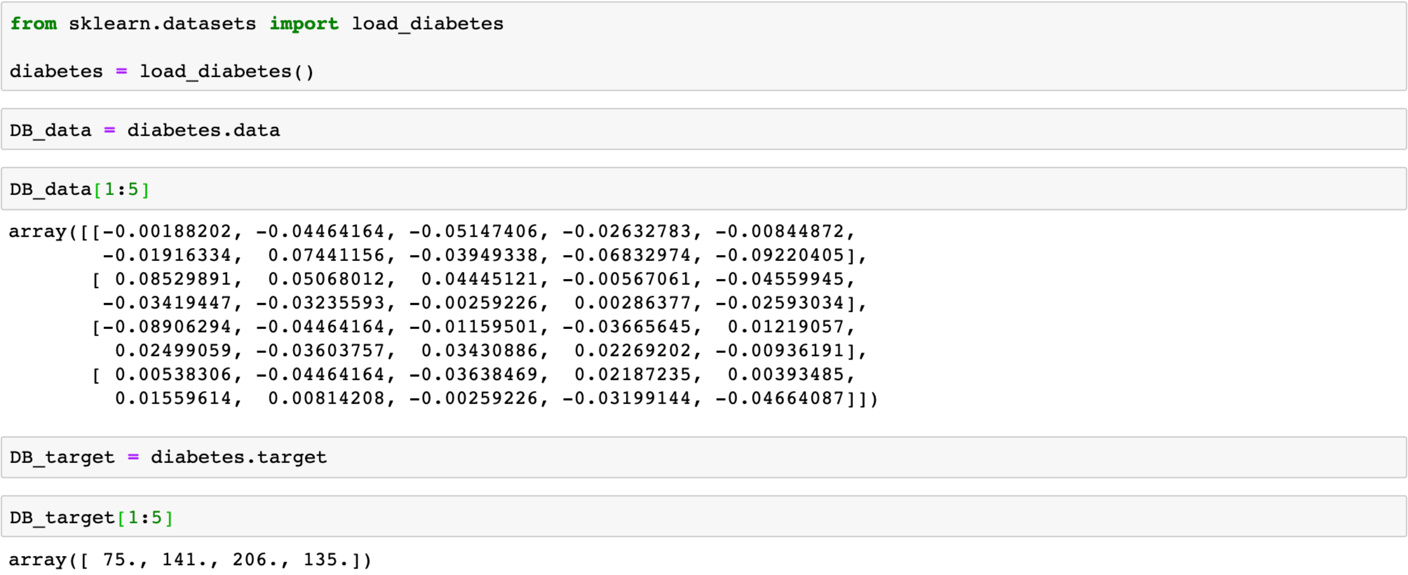
- The Digit Dataset
  - This dataset is made up of 1797 8x8 images
  - Each image is of a hand-written digit
  - In order to utilize an 8x8 figure like this, we’d have to first transform it into a feature vector with length 64
  - **from sklearn.datasets import load_digits**

Problem 04

- (a) Use logistic regression to build a model to predict the handwritten digits
  - Discuss how the parameter ‘C’ affects the accuracy
- (b) Use LinearSVC to build a model to predict the handwritten digits
  - Discuss how the parameter ‘C’ affects the accuracy
- (c) Use GaussianNB to build a model to predict the handwritten digits
- (d) Compare these results against KNN

Problem 05

- In this problem we will deal with the diabetes dataset



```
from sklearn.datasets import load_diabetes

diabetes = load_diabetes()
```

```
DB_data = diabetes.data
```

```
DB_data[1:5]
```

```
array([[-0.00188202, -0.04464164, -0.05147406, -0.02632783, -0.00844872,
        -0.01916334,  0.07441156, -0.03949338, -0.06832974, -0.09220405],
       [ 0.08529891,  0.05068012,  0.04445121, -0.00567061, -0.04559945,
        -0.03419447, -0.03235593, -0.00259226,  0.00286377, -0.02593034],
       [-0.08906294, -0.04464164, -0.01159501, -0.03665645,  0.01219057,
         0.02499059, -0.03603757,  0.03430886,  0.02269202, -0.00936191],
       [ 0.00538306, -0.04464164, -0.03638469,  0.02187235,  0.00393485,
         0.01559614,  0.00814208, -0.00259226, -0.03199144, -0.04664087]])
```

```
DB_target = diabetes.target
```

```
DB_target[1:5]
```

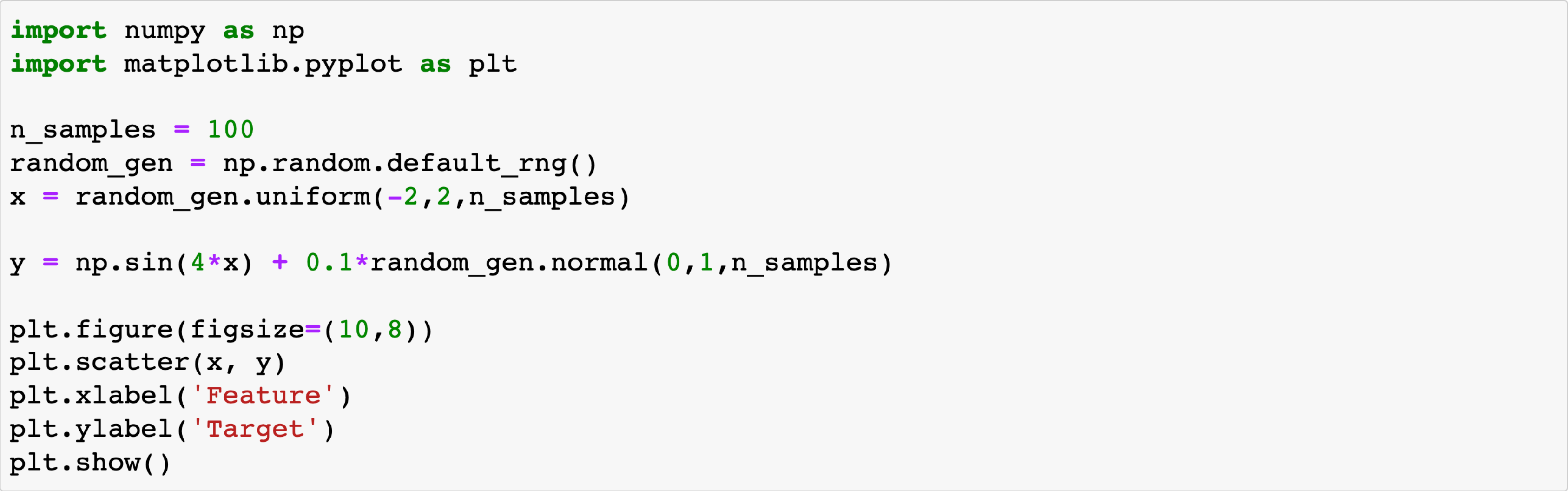
```
array([ 75., 141., 206., 135.])
```

Problem 05

- Use the linear regression to construct a prediction model
- Use the ridge regression to construct a prediction model
  - Discuss how the strength of regularization affects the prediction model
- Use the lasso regression to construct a prediction model
  - Discuss how the strength of regularization affects the prediction model

Problem 06

- Use the following code to generate the data



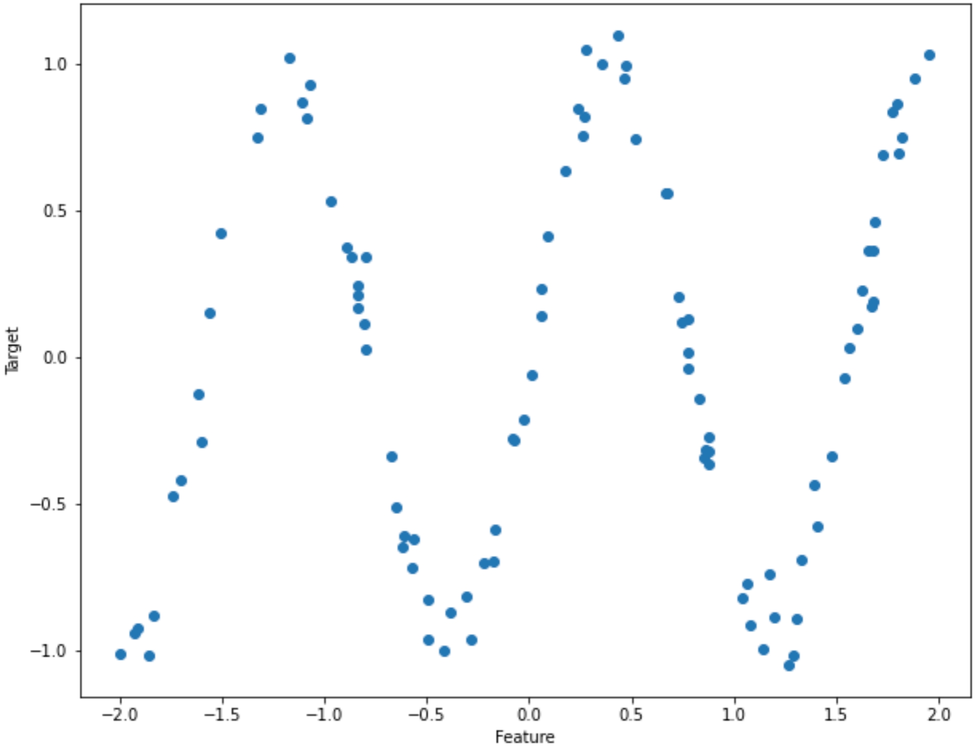
```
import numpy as np
import matplotlib.pyplot as plt

n_samples = 100
random_gen = np.random.default_rng()
x = random_gen.uniform(-2,2,n_samples)

y = np.sin(4*x) + 0.1*random_gen.normal(0,1,n_samples)

plt.figure(figsize=(10,8))
plt.scatter(x, y)
plt.xlabel('Feature')
plt.ylabel('Target')
plt.show()
```

Problem 06

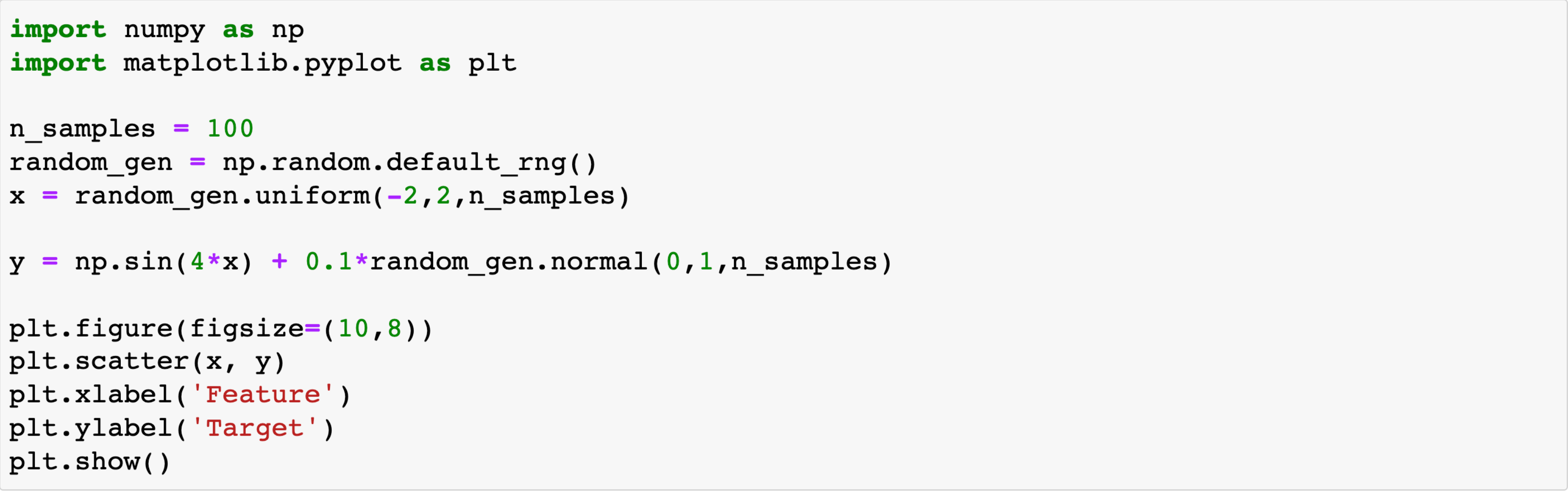


Problem 06

- Use the polynomial basis function to transform the input space to the feature space
- Discuss how the order of the polynomial basis function affect the prediction results when the linear regression model is utilized

Problem 07

- Use the following code to generate the data



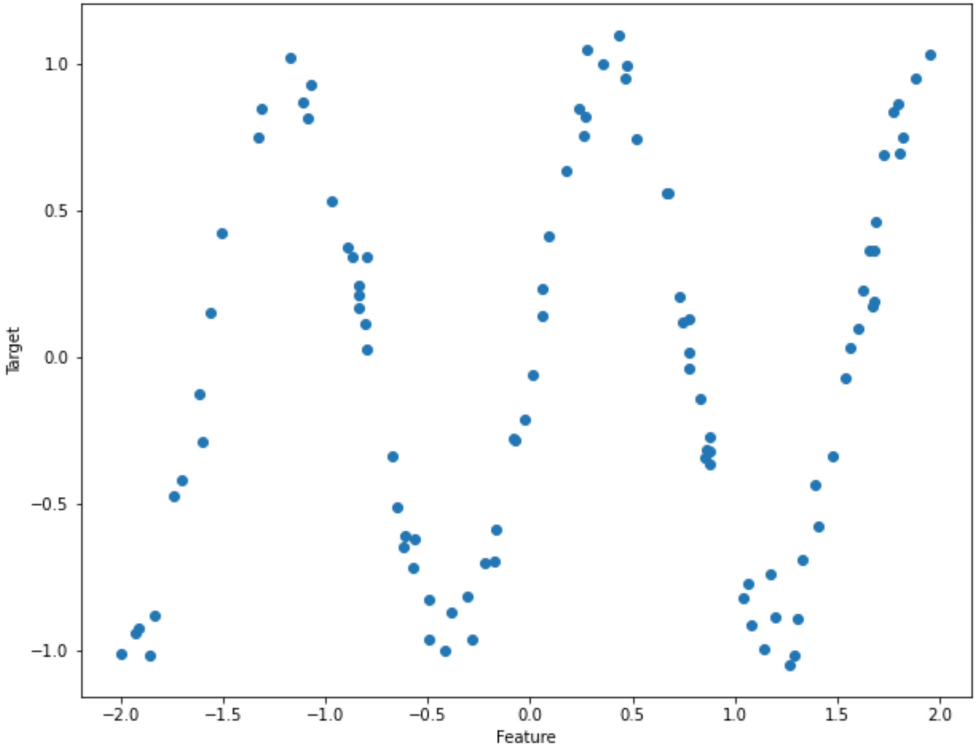
```python
import numpy as np
import matplotlib.pyplot as plt

n_samples = 100
random_gen = np.random.default_rng()
x = random_gen.uniform(-2,2,n_samples)

y = np.sin(4*x) + 0.1*random_gen.normal(0,1,n_samples)

plt.figure(figsize=(10,8))
plt.scatter(x, y)
plt.xlabel('Feature')
plt.ylabel('Target')
plt.show()
```

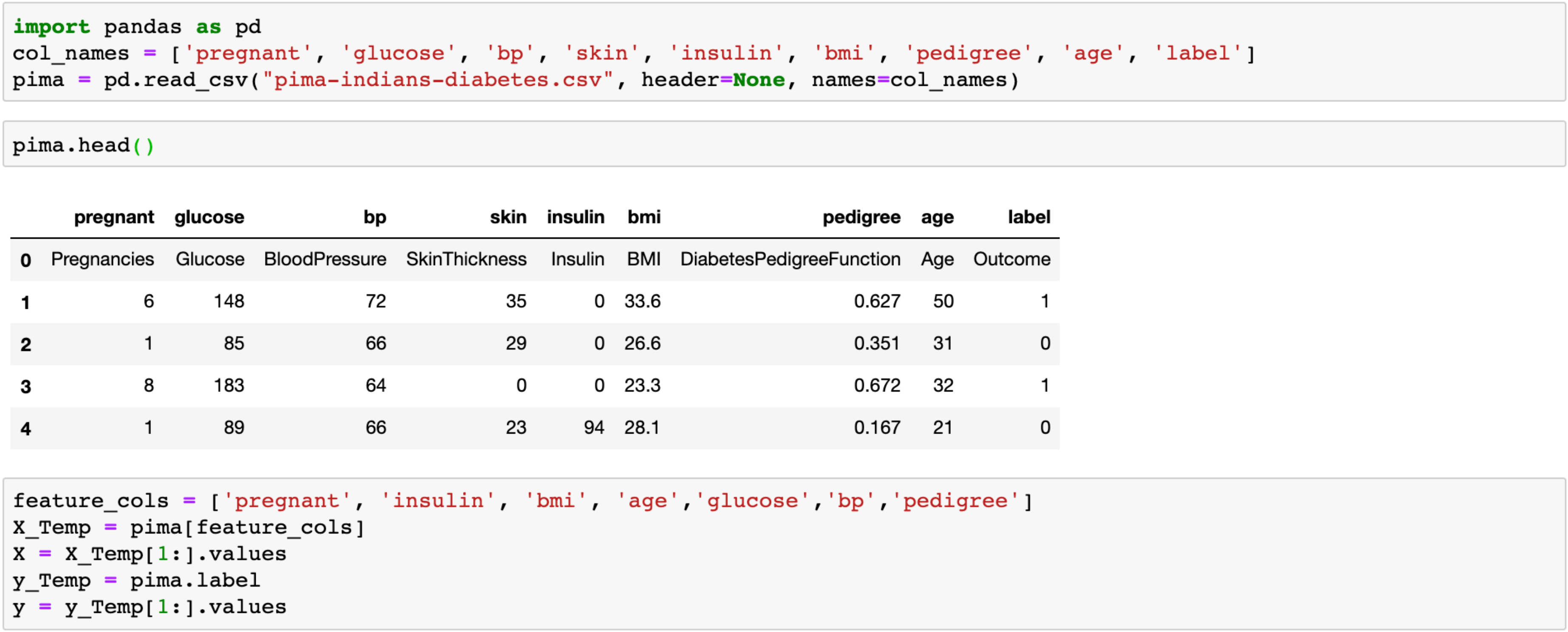
Problem 07



Problem 07

- Use the Gaussian basis function to transform the input space to the feature space
- Suppose the number of basis functions is $n + 1$ and the μ_j of the jth basis function is chosen to be $-2 + \frac{4}{n}(j - 1)$ for $j = 1,2, \cdots n + 1$
- Discuss how the σ affect the prediction results when the linear regression model is utilized

Problem 08



```
import pandas as pd
col_names = ['pregnant', 'glucose', 'bp', 'skin', 'insulin', 'bmi', 'pedigree', 'age', 'label']
pima = pd.read_csv("pima-indians-diabetes.csv", header=None, names=col_names)
```

```
pima.head()
```

	pregnant	glucose	bp	skin	insulin	bmi	pedigree	age	label
0	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
1	6	148	72	35	0	33.6	0.627	50	1
2	1	85	66	29	0	26.6	0.351	31	0
3	8	183	64	0	0	23.3	0.672	32	1
4	1	89	66	23	94	28.1	0.167	21	0

```
feature_cols = ['pregnant', 'insulin', 'bmi', 'age','glucose','bp','pedigree']
X_Temp = pima[feature_cols]
X = X_Temp[1:].values
y_Temp = pima.label
y = y_Temp[1:].values
```

Problem 08

- In this problem, we will use the Pima Indian Diabetes dataset to build a model to predict whether a person has the diabetes or not
  - (a) Use logistic regression to build a prediction model
    - Discuss how the parameter ‘C’ affects the accuracy
  - (b) Use LinearSVC to build a model to build a prediction model
    - Discuss how the parameter ‘C’ affects the accuracy
  - (c) Use Random Forest to build a model to build a prediction model
    - Discuss how the number of estimators affects the accuracy