

**0907531 Machine Learning (Spring 2018)**  
**Midterm Exam**

الاسم: ..... رقم التسجيل: ..... رقم التسلسل: .....

**Instructions:** Time **60** min. Open book and notes exam. No electronics. Please answer all problems in the space provided and limit your answer to the space provided. No questions are allowed. There are five problems. Each problem has 5 points except the last problem that has 10 points.

**P1.** Draw a line to connect each Problem/Algorithm of the following table with the Machine Learning type that best fits it.

[5 points]

No	Problem/Algorithm	Type	Answer
1	Spam mail filtering	Unsupervised clustering	3
2	Sorting fish using nearest neighbor classifier	Online learning	5
3	Analyzing land using satellite images	Supervised classification	1
4	Improving model accuracy using earlier models	Instance-based learning	2
5	Predicting gold prices	Reinforcement learning	4

**P2.** Complete the following Python code to loop through and print out all odd numbers from the numbers list in the same order they are received. Don't print any numbers that come after 390 in the sequence.

[5 points]

```
numbers = [  
    951, 402, 984, 651, 360, 69, 408, 319, 601, 485, 980, 507, 725,  
    615, 83, 165, 141, 501, 263, 617, 865, 575, 219, 390, 984, 592,  
    386, 527  
]  
  
# your code goes here  
for number in numbers:  
    if number == 390:  
        break  
  
    if number % 2 == 0:  
        continue  
  
    print(number)
```

**P3.** What is the output of the following Python code?

*[5 points]*

```
import numpy as np
x = np.arange(5, 17).reshape(4,3)
print(x)
```

```
[[ 5  6  7]
 [ 8  9 10]
 [11 12 13]
 [14 15 16]]
```

**P4.** Given the following Python statements, calculate the classification error, precision and recall.

*[5 points]*

```
>>> from sklearn.metrics import confusion_matrix
>>> confusion_matrix(y_train_5, y_train_pred)
array([[3, 2],
       [1, 4]])
```

**Error =  $(1+2) / (3+2+1+4) = 3/10 = 0.3$**

**Precision =  $4 / (4+2) = 4/6 = 0.67$**

**Recall =  $4 / (4+1) = 4/5 = 0.8$**

**P5.** Given the information in the box blow, complete the following Python code to split the data into train set (70%) and test set (30%), scale the features  $x_1$  and  $x_2$  using the standardization method, train a Linear Regressor using the two features to predict the response  $y$  using the scaled train set, and print the RMSE of predicting the response  $y$  of the test set.

[10 points]

```
>>> data.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20000 entries, 0 to 19999
Data columns (total 3 columns):
x1          20000 non-null float64
x2          20000 non-null float64
y           20000 non-null float64
dtypes: float64(3)
memory usage: 1.1+MB
```

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error

train_set, test_set = train_test_split(data, test_size=0.3)

X_train = train_set.drop("y", axis=1)
y_train = train_set["y"].copy()
X_test = test_set.drop("y", axis=1)
y_test = test_set["y"].copy()

scaler = StandardScaler()
scaler.fit(X_train)
X_train_scaled = scaler.transform(X_train)

lin_reg = LinearRegression()
lin_reg.fit(X_train_scaled, y_train)

X_test_scaled = scaler.transform(X_test)
y_predictions = lin_reg.predict(X_test_scaled)

print(np.sqrt(mean_squared_error(y_test, y_predictions)))
```

<Good Luck>