Day1

January 8, 2019

0.1 1

```
In [1]: import numpy as np
        import pandas as pd
```

0.2 2

```
In [2]: dataset = pd.read_csv('../datasets/Data.csv')
        X = dataset.iloc[:, :-1].values
        Y = dataset.iloc[:, 3].values
        print(X)
```

```
[['France' 44.0 72000.0]
 ['Spain' 27.0 48000.0]
 ['Germany' 30.0 54000.0]
 ['Spain' 38.0 61000.0]
 ['Germany' 40.0 nan]
 ['France' 35.0 58000.0]
 ['Spain' nan 52000.0]
 ['France' 48.0 79000.0]
 ['Germany' 50.0 83000.0]
 ['France' 37.0 67000.0]]
```

0.3 3

```
In [3]: from sklearn.preprocessing import Imputer
        imputer = Imputer(missing_values = 'NaN', strategy = "mean", axis = 0)
        imputer = imputer.fit(X[:, 1:3])
        X[:, 1:3] = imputer.transform(X[: , 1:3])
        print(X)
```

```
[['France' 44.0 72000.0]
 ['Spain' 27.0 48000.0]
 ['Germany' 30.0 54000.0]
 ['Spain' 38.0 61000.0]
 ['Germany' 40.0 63777.77777777778]
 ['France' 35.0 58000.0]
```

```
 ['Spain' 38.77777777777778 52000.0]
 ['France' 48.0 79000.0]
 ['Germany' 50.0 83000.0]
 ['France' 37.0 67000.0]]
```

0.4 4

```
In [4]: from sklearn.preprocessing import LabelEncoder, OneHotEncoder
        encode_x = LabelEncoder()
        X[:, 0] = encode_x.fit_transform(X[:, 0])
        print(X)
```

```
[[0 44.0 72000.0]
 [2 27.0 48000.0]
 [1 30.0 54000.0]
 [2 38.0 61000.0]
 [1 40.0 63777.77777777778]
 [0 35.0 58000.0]
 [2 38.77777777777778 52000.0]
 [0 48.0 79000.0]
 [1 50.0 83000.0]
 [0 37.0 67000.0]]
```

```
In [5]: onehotencoder = OneHotEncoder(categorical_features=[0])
        X = onehotencoder.fit_transform(X).toarray()
        label_Y = LabelEncoder()
        Y = label_Y.fit_transform(Y)
        print(X, Y)
```

```
[[  1.00000000e+00   0.00000000e+00   0.00000000e+00   4.40000000e+01
    7.20000000e+04]
 [  0.00000000e+00   0.00000000e+00   1.00000000e+00   2.70000000e+01
    4.80000000e+04]
 [  0.00000000e+00   1.00000000e+00   0.00000000e+00   3.00000000e+01
    5.40000000e+04]
 [  0.00000000e+00   0.00000000e+00   1.00000000e+00   3.80000000e+01
    6.10000000e+04]
 [  0.00000000e+00   1.00000000e+00   0.00000000e+00   4.00000000e+01
    6.37777778e+04]
 [  1.00000000e+00   0.00000000e+00   0.00000000e+00   3.50000000e+01
    5.80000000e+04]
 [  0.00000000e+00   0.00000000e+00   1.00000000e+00   3.87777778e+01
    5.20000000e+04]
 [  1.00000000e+00   0.00000000e+00   0.00000000e+00   4.80000000e+01
    7.90000000e+04]
```

```
 [  0.00000000e+00   1.00000000e+00   0.00000000e+00   5.00000000e+01
    8.30000000e+04]
 [  1.00000000e+00   0.00000000e+00   0.00000000e+00   3.70000000e+01
    6.70000000e+04]] [0 1 0 0 1 1 0 1 0 1]
```

0.5 5

```
In [6]: from sklearn.cross_validation import train_test_split
        X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.2, random_state
```

```
/home/huiwen/anaconda3/lib/python3.6/site-packages/sklearn/cross_validation.py:41: DeprecationWa
  "This module will be removed in 0.20.", DeprecationWarning)
```

0.6 6

```
In [7]: from sklearn.preprocessing import StandardScaler
        sc_X = StandardScaler();
        X_train = sc_X.fit_transform(X_train)
        X_test = sc_X.fit_transform(X_test)
        print(X_train)
```

```
[[-1.          2.64575131 -0.77459667  0.26306757  0.12381479]
 [ 1.         -0.37796447 -0.77459667 -0.25350148  0.46175632]
 [-1.         -0.37796447  1.29099445 -1.97539832 -1.53093341]
 [-1.         -0.37796447  1.29099445  0.05261351 -1.11141978]
 [ 1.         -0.37796447 -0.77459667  1.64058505  1.7202972 ]
 [-1.         -0.37796447  1.29099445 -0.0813118  -0.16751412]
 [ 1.         -0.37796447 -0.77459667  0.95182631  0.98614835]
 [ 1.         -0.37796447 -0.77459667 -0.59788085 -0.48214934]]
```