DS200 Lab 9
Correlation Analysis for NFL Pass Prediction

Instructor: John Yen

LA: Luwei Lei

October 18, 2018

Learning Objectives (Part 1)

- Be able to use correlation analysis in Python (Pandas).
- Be able to compare the result of correlation analysis with features selected from decision tree learning

Load the CSV file used in Lab 4

DataFrame: a Pandas data structure for storing an array with labeled rows and columns (of the same data type)

jupyter Lab 9 Correlation Analysis for Predicting NFL Yards Gained (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code

In [5]:
```
import datascience
import numpy as np
import pandas as pd
import graphviz
from datascience import *
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.model_selection import train_test_split

from sklearn.pipeline import Pipeline
from sklearn import tree
from sklearn.tree import DecisionTreeClassifier
from sklearn import metrics

import os
os.environ["PATH"] += os.pathsep + 'C:/Program Files (x86)/Graphviz2.38/bin/'
```

In [15]:
```
Pitt_NE = pd.read_csv('Ben-NE-9-10-2015-pass-6.csv', sep=",")
```

In [17]:
```
type(Pitt_NE)
```

Out[17]:
```
pandas.core.frame.DataFrame
```

Display the Content of the DataFrame

```
In [19]: print(Pitt_NE)
```

	down	ydstogo	Yards.Gained.PrevPlay	AirYards	PassLocation	PassOutcome
0	1	10	18	-4	1	1
1	1	10	0	9	1	1
2	3	22	6	1	1	1
3	1	10	0	7	-1	1
4	1	10	13	6	-1	1
5	1	10	12	7	-1	1
6	1	10	0	5	1	0
7	2	10	0	25	1	0
8	3	5	-1	6	-1	1
9	1	15	4	-1	1	1
10	3	18	-6	17	-1	1
11	1	20	5	5	-1	1
12	2	11	9	4	-1	1
13	2	13	-3	-2	-1	1
14	3	6	7	6	0	1
15	2	7	0	11	1	1
16	1	10	13	16	1	1
17	1	10	19	6	1	1

Conduct Correlation Analysis

Corr is a Pandas method of DataFrame for calculating correlation among the columns of a DataFrame

```
In [20]: Pitt_NE.corr(method='pearson')
```

Out[20]:

	down	ydstogo	Yards.Gained.PrevPlay	AirYards	PassLocation	PassOutcome
down	1.000000	-0.293906	-0.306207	-0.032894	-0.060043	-0.076448
ydstogo	-0.293906	1.000000	-0.060572	0.091813	-0.058176	0.249902
Yards.Gained.PrevPlay	-0.306207	-0.060572	1.000000	0.022748	-0.081227	0.144037
AirYards	-0.032894	0.091813	0.022748	1.000000	0.056054	-0.286445
PassLocation	-0.060043	-0.058176	-0.081227	0.056054	1.000000	-0.223061
PassOutcome	-0.076448	0.249902	0.144037	-0.286445	-0.223061	1.000000

Inspecting the Correlation Analysis Results

- What correlation coefficient did you find interesting?

```
In [20]: Pitt_NE.corr(method='pearson')
```

Out[20]:

	down	ydstogo	Yards.Gained.PrevPlay	AirYards	PassLocation	PassOutcome
down	1.000000	-0.293906	-0.306207	-0.032894	-0.060043	-0.076448
ydstogo	-0.293906	1.000000	-0.060572	0.091813	-0.058176	0.249902
Yards.Gained.PrevPlay	-0.306207	-0.060572	1.000000	0.022748	-0.081227	0.144037
AirYards	-0.032894	0.091813	0.022748	1.000000	0.056054	-0.286445
PassLocation	-0.060043	-0.058176	-0.081227	0.056054	1.000000	-0.223061
PassOutcome	-0.076448	0.249902	0.144037	-0.286445	-0.223061	1.000000

Comparing Correlation Analysis Results with Features in Decision Trees

- What features did you find most often used by decision trees?
- Are they correlated with the prediction variable (PassOutcome)?

```
In [20]: Pitt_NE.corr(method='pearson')
```

Out[20]:

	down	ydstogo	Yards.Gained.PrevPlay	AirYards	PassLocation	PassOutcome
down	1.000000	-0.293906	-0.306207	-0.032894	-0.060043	-0.076448
ydstogo	-0.293906	1.000000	-0.060572	0.091813	-0.058176	0.249902
Yards.Gained.PrevPlay	-0.306207	-0.060572	1.000000	0.022748	-0.081227	0.144037
AirYards	-0.032894	0.091813	0.022748	1.000000	0.056054	-0.286445
PassLocation	-0.060043	-0.058176	-0.081227	0.056054	1.000000	-0.223061
PassOutcome	-0.076448	0.249902	0.144037	-0.286445	-0.223061	1.000000

Learning Objectives (Part 2)

- Be able to understand the importance of identifying and avoiding features that may not feasible to be used for prediction.
- Be able to use correlation analysis to identify such features

Load a new CSV file into Jupyter Notebook

- Download the file Ben-NE-9-10-2015-pass-4.csv and copy it to the directory from which you launch the Jupyter Notebook
- The file contains one more feature (Yards.Gained) than those used in previous labs

```
In [24]: Ben_pass_2 = pd.read_csv('Ben-NE-9-10-2015-pass-4.csv', sep=",")

In [25]: print(Ben_pass_2)
```

```
    down  ydstogo  Yards.Gained.PrevPlay  Yards.Gained  AirYards  \
0      1       10                    18             9        -4
1      1       10                     0            14         9
2      3       22                     6            10         1
3      1       10                     0            13         7
4      1       10                    13            12         6
5      1       10                    12            13         7
6      1       10                     0             0         5
7      2       10                     0             0        25
8      3        5                    -1            10         6
9      1       15                     4             3        -1
10     3       18                    -6            17        17
11     1       20                     5             9         5
12     2       11                     9             9         4
13     2       13                    -3             7        -2
14     3        6                     7             8         6
15     2        7                     0            13        11
16     1       10                    13            19        16
17     1       10                    19            16         6
```

Conduct Correlation Analysis

```
In [26]: Ben_pass_2.corr(method='pearson')
```

Out[26]:

	down	ydstogo	Yards.Gained.PrevPlay	Yards.Gained	AirYards	PassLocation	PassOutcome
down	1.000000	-0.293906	-0.306207	-0.054182	-0.032894	-0.060043	-0.076448
ydstogo	-0.293906	1.000000	-0.060572	0.222043	0.091813	-0.058176	0.249902
Yards.Gained.PrevPlay	-0.306207	-0.060572	1.000000	0.131102	0.022748	-0.081227	0.144037
Yards.Gained	-0.054182	0.222043	0.131102	1.000000	0.363336	0.019662	0.537075
AirYards	-0.032894	0.091813	0.022748	0.363336	1.000000	0.056054	-0.286445
PassLocation	-0.060043	-0.058176	-0.081227	0.019662	0.056054	1.000000	-0.223061
PassOutcome	-0.076448	0.249902	0.144037	0.537075	-0.286445	-0.223061	1.000000

Modify the columns of features to use and the column for prediction output

- X = Ben_pass_2.values[: , 0 : 6]
- Y = Ben_pass_2.values[: , 6]

```
X= Ben_pass_2.values[:,0:6]
print(X)
```

```
[[ 1 10 18  9 -4  1]
 [ 1 10  0 14  9  1]
 [ 3 22  6 10  1  1]
 [ 1 10  0 13  7 -1]
 [ 1 10 13 12  6 -1]
 [ 1 10 12 13  7 -1]
 [ 1 10  0  0  5  1]
 [ 2 10  0  0 25  1]
 [ 3  5 -1 10  6 -1]
 [ 1 15  4  3 -1  1]
 [ 3 18 -6 17 17 -1]
```

```
Y=Ben_pass_2.values[:,6]
print(Y)
```

```
[1 1 1 1 1 1 0 0 1 1 1 1 1 1 1 1 1 1 0 1 0 1 1 1 1 1 1 1 1 1 0 1 0 0 1 1 1
 1 1 1 1 1 1 1 1 0 1 0 1 0 1 0 0 0 0 1 0 1 0 0 1 1 0 1 1 1 0 1 1 1]
```

Modify the feature names to use for visualizing the tree

- In tree.export_graphviz statement, change the list of feature_names to

‘down’, ‘ydstogo’, ‘Yards.Gained.PrevPlay’, ‘Yards.Gained’, ‘AirYards’, ‘PassLocation’

Execute Each Cell of the Modified Jupyter Notebook from the First Cell to the Last one by one

- If you get an error message, check that you did not enter the file name incorrectly.
- Look at the tree constructed, how to interpret the tree?
- What is the performance of the model?

Lab 9 (due 10 pm, 10/19)

- Submit your Jupyter Notebook after you successfully modified and executed the notebook.
- Submit the correlation analysis results for both CSV file and the tree generated.
- Submit a document that discusses (1) the correlation analysis of the two CSV files, (2) performance of the tree using the second CSV file, (2) an interpretation of the tree, and (3) what you learned from the lab.

Additional Information about Decision Trees

- Decision tree can be used to predict the probability that an input data is in each output class.
- Clf.predict_proba(<an array of input data>)

It returns an array of predicted probability for each output class, each row contains the predicted probability for each input data.