

```

In [1]: #Importing Libraries
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline

import pandas as pd

from sklearn.preprocessing import LabelEncoder
from sklearn.utils import shuffle

#Constants

TRAIN_SIZE = 6772
TEST_SIZE = 1693

ACTIVATION_F = 'tanh'

#Import Training Set
df = pd.read_csv('SeoulBikeData.csv',engine='python')

dummies = pd.get_dummies(df.Seasons)
df= pd.concat([df,dummies],axis='columns')
df= df.drop(['Seasons','Winter'], axis='columns')

le= LabelEncoder()
dfle = df
df.Holiday=le.fit_transform(dfle.Holiday)
df['Functioning Day']=le.fit_transform(dfle['Functioning Day'])

```

```
In [2]: df.head()
```

```
Out[2]:
```

	Date	Rented Bike Count	Hour	Temperature(°C)	Humidity(%)	Wind speed (m/s)	Visibility (10m)	Dew point temperature(°C)	Solar Radiation (MJ/m <sup>2</sup> )
0	01/12/2017	254	0	-5.2	37	2.2	2000	-17.6	0.0
1	01/12/2017	204	1	-5.5	38	0.8	2000	-17.6	0.0
2	01/12/2017	173	2	-6.0	39	1.0	2000	-17.7	0.0
3	01/12/2017	107	3	-6.2	40	0.9	2000	-17.6	0.0
4	01/12/2017	78	4	-6.0	36	2.3	2000	-18.6	0.0

```

In [3]: X=df
X=X.drop("Date",axis=1)
X=X.drop("Rented Bike Count",axis=1)
y = df["Rented Bike Count"]

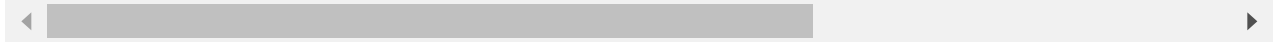
```

```
In [4]: X.head()
```

```
Out[4]:
```

	Hour	Temperature(°C)	Humidity(%)	Wind speed (m/s)	Visibility (10m)	Dew point temperature(°C)	Solar Radiation (MJ/m <sup>2</sup> )	Rainfall(mm)	Sn
0	0	-5.2	37	2.2	2000	-17.6	0.0	0.0	0.0

	Hour	Temperature(°C)	Humidity(%)	Wind speed (m/s)	Visibility (10m)	Dew point temperature(°C)	Solar Radiation (MJ/m2)	Rainfall(mm)	Sn
1	1	-5.5	38	0.8	2000	-17.6	0.0	0.0	
2	2	-6.0	39	1.0	2000	-17.7	0.0	0.0	
3	3	-6.2	40	0.9	2000	-17.6	0.0	0.0	
4	4	-6.0	36	2.3	2000	-18.6	0.0	0.0	



```
In [5]: from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(
    X,
    y,
    test_size=0.2,
    random_state=0)

X_train.shape, X_test.shape
```

Out[5]: ((7008, 14), (1752, 14))

```
In [6]: X_train.corr()
```

Out[6]:

	Hour	Temperature(°C)	Humidity(%)	Wind speed (m/s)	Visibility (10m)	Dew point temperature(°C)	Solar Radiation (MJ/m2)	Rainfall(mm)	Sn
Hour	1.000000	0.135492	-0.233281	0.282317	0.096367	0.016132	0.148713	0.011442	0.000357
Temperature(°C)	0.135492	1.000000	0.154051	-0.032333	0.036242	0.912724	0.363076	0.047771	0.052282
Humidity(%)	-0.233281	0.154051	1.000000	-0.344004	-0.547715	0.532832	-0.457705	0.229661	0.051985
Wind speed (m/s)	0.282317	-0.032333	-0.344004	1.000000	0.176518	-0.175369	0.332530	-0.020668	-0.023402
Visibility (10m)	0.096367	0.036242	-0.547715	0.176518	1.000000	-0.176538	0.145275	-0.159169	-0.032970
Dew point temperature(°C)	0.016132	0.912724	0.532832	-0.175369	-0.176538	1.000000	0.105797	0.121061	0.065061
Solar Radiation (MJ/m2)	0.148713	0.363076	-0.457705	0.332530	0.145275	0.105797	1.000000	0.064274	0.002597
Rainfall(mm)	0.011442	0.047771	0.229661	-0.020668	-0.159169	0.121061	0.064274	1.000000	0.652420
Snowfall (cm)	-0.019582	-0.218641	0.110899	-0.003987	-0.117943	-0.150324	-0.074000	0.652420	1.000000
Holiday	0.000357	0.052282	0.051985	-0.023402	-0.032970	0.065061	0.002597	0.652420	1.000000
Functioning Day	0.012371	-0.049998	-0.015002	0.017091	-0.032013	-0.050186	-0.002000	0.652420	1.000000
Autumn	-0.000024	0.056742	0.036346	-0.131273	0.109990	0.064274	-0.042000	0.652420	1.000000
Spring	-0.005970	0.010933	0.014811	0.089861	-0.193632	0.002597	0.093000	0.652420	1.000000
Summer	0.012511	0.665986	0.187273	-0.062988	0.065471	0.652420	0.134000	0.652420	1.000000



```
In [7]: import seaborn as sns
#Using Pearson Correlation
plt.figure(figsize=(12,10))
cor = X_train.corr()
sns.heatmap(cor, annot=True, cmap=plt.cm.CMRmap_r)
plt.show()
```

