WuBenjaminAssignment3

March 9, 2022

1 CS156 (Introduction to AI), Spring 2022

2 Homework 3 submission

2.0.1 Roster Name: Benjamin Wu

2.0.2 Student ID: 013607880

2.0.3 Email address: benjamin.wu01@sjsu.edu

2.1 References and sources

https://www.statology.org/pandas-select-column-by-index/

2.2 Solution

Load libraries and set random number generator seed

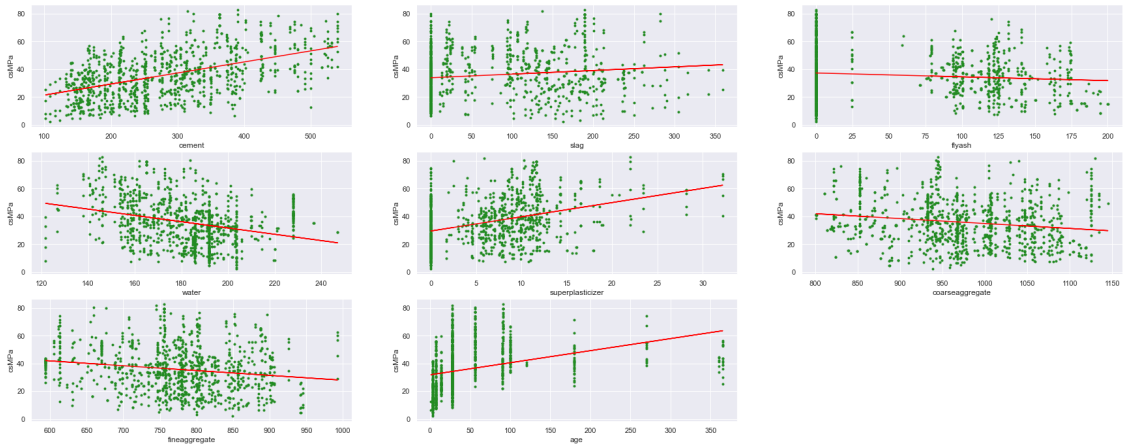
[ ]:
```
import numpy as np
import pandas as pd
from sklearn import datasets
from sklearn.datasets import load_boston
from sklearn import linear_model
from sklearn import preprocessing
from sklearn.preprocessing import PolynomialFeatures
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
import matplotlib.pyplot as plt
import matplotlib.ticker as ticker
import seaborn as sns
```

[ ]:
```
np.random.seed(42)
```

Code the solution

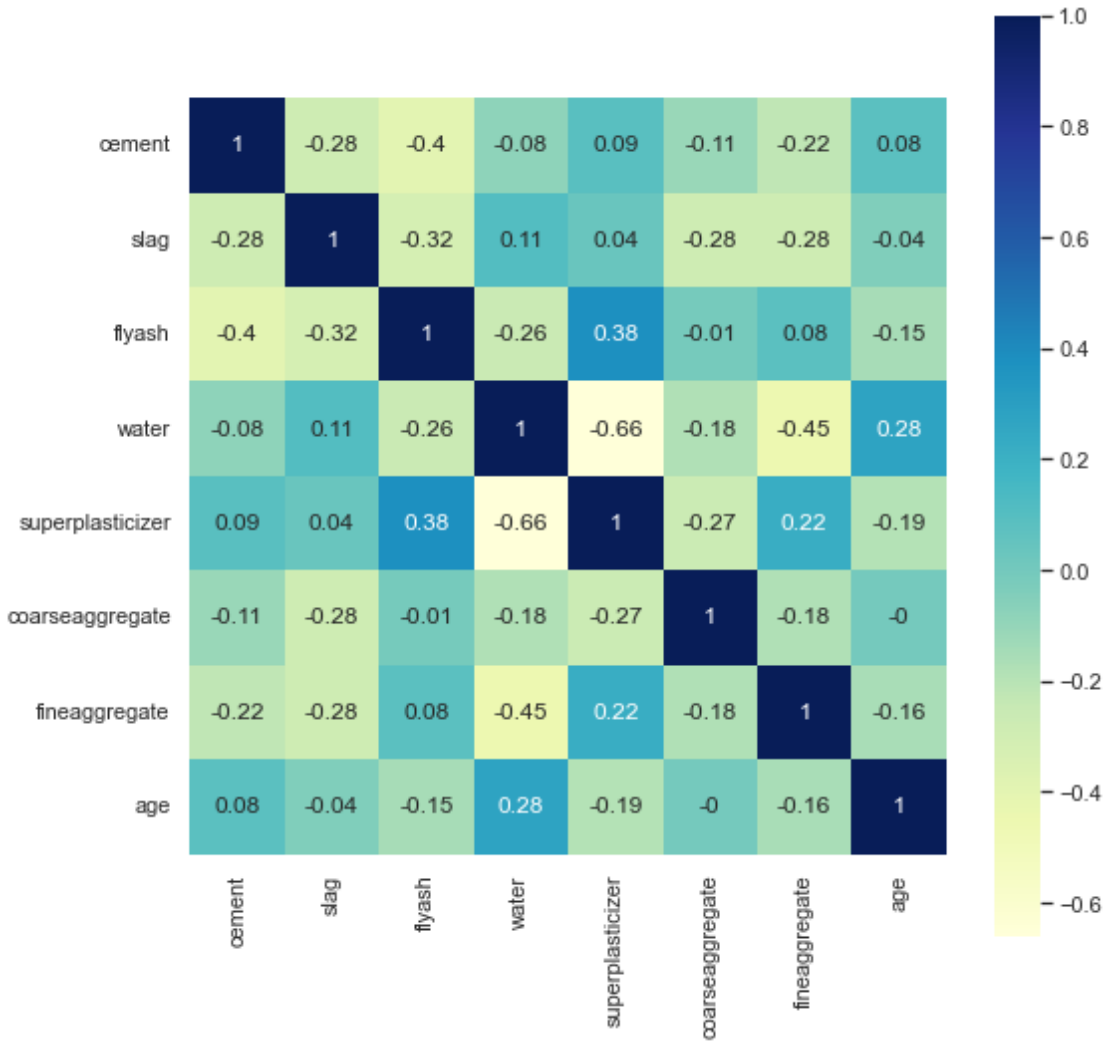
[ ]:
```
df = pd.read_csv("homework3_input_data.csv")



plt.figure(figsize=(30,20))
for i, col in enumerate(df.columns[0:8]):
    plt.subplot(5, 3, i+1)
```

```
    x = df[col]
    y = df['csMPa']
    plt.plot(x, y, '.', color="forestgreen")
    m, b = np.polyfit(x, y, 1)
    plt.plot(x, m*x + b, color="red")
    plt.xlabel(col)
    plt.ylabel('csMPa')
```



[ ]:
```
features =␣
  ↪df[['cement','slag','flyash','water','superplasticizer','coarseaggregate','fineaggregate','
sns.set(rc={'figure.figsize': (8.5,8.5)})
sns.heatmap(features.corr().round(2), square=True, cmap='YlGnBu', annot=True)
```

[ ]: <AxesSubplot:>



2.2.1 Splitting data into training and test datasets

[ ]:
```
X_train, X_test, Y_train, Y_test = train_test_split(df.iloc[:, 0:8], df.iloc[:,␣
↪8], test_size=0.2, random_state=0)
X_train.shape, Y_train.shape, X_test.shape, Y_test.shape
```

[ ]:
```
((824, 8), (824,), (206, 8), (206,))
```

2.2.2 Training model

[ ]:
```
model = linear_model.LinearRegression().fit(X_train, Y_train)
Y_test_pred = model.predict(X_test)
```

2.2.3 Calculating mean squared error and coefficient of determination

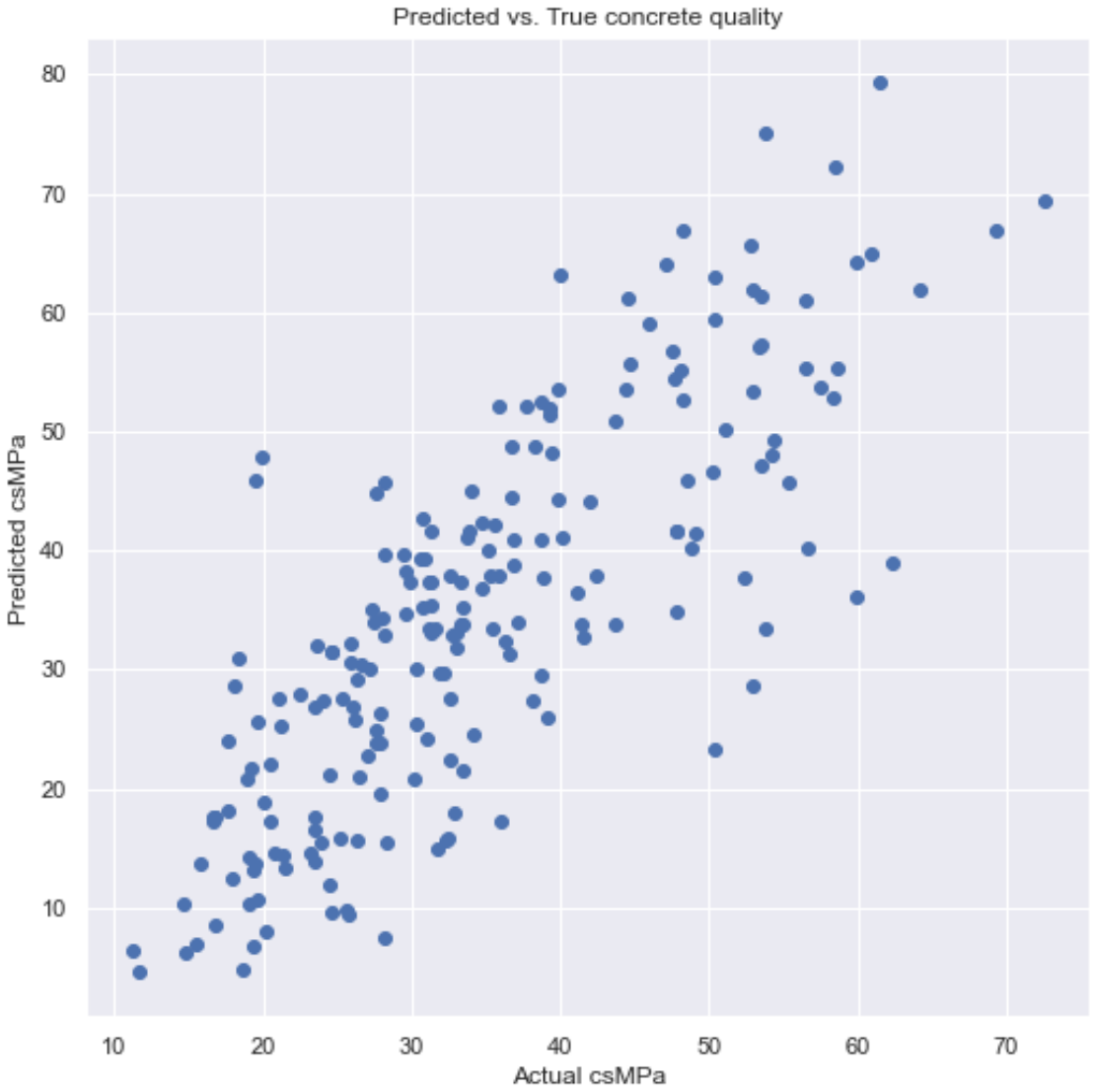
[ ]:
```
print('Mean squared error: %.2f' % mean_squared_error(Y_test, Y_test_pred))
print('Coefficient of determination: %.2f' % r2_score(Y_test, Y_test_pred))
```

```
Mean squared error: 95.62
Coefficient of determination: 0.64
```

2.2.4 Graphing predicted vs actual csMPa

[ ]:
```
plt.scatter(Y_test_pred,Y_test)
plt.title('Predicted vs. True concrete quality')
plt.xlabel('Actual csMPa')
plt.ylabel('Predicted csMPa')
plt.show
```

[ ]:
```
<function matplotlib.pyplot.show(close=None, block=None)>
```


Predicted vs. True concrete quality
Predicted csMPa
Actual csMPa
10
20
30
40
50
60
70
80