Introduction_to_Python_for_Machine_Learning_-_Lesson_6_Quiz

July 13, 2020

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

[2]:
```
df = pd.read_csv("https://raw.githubusercontent.com/WalePhenomenon/climate_change/master/fuel_ferc1.csv")
df.head()
```

[2]:

	record_id	utility_id_ferc1	report_year	plant_name_ferc1	fuel_type_code_pudl	fuel_unit	fuel_qty_burned	fuel_mmbtu_per_unit	fuel_cost_per_unit_burned	fuel_cost_per_unit_delivered	fuel_cost_per_mmbtu
0	f1_fuel_1994_12_1_0_7	1	1994	rockport	coal	ton	5377489.0	16.590	18.59	18.53	1.121
1	f1_fuel_1994_12_1_0_10	1	1994	rockport total plant	coal	ton	10486945.0	16.592	18.58	18.53	1.120
2	f1_fuel_1994_12_2_0_1	2	1994	gorgas	coal	ton	2978683.0	24.130	39.72	38.12	1.650
3	f1_fuel_1994_12_2_0_7	2	1994	barry	coal	ton	3739484.0	23.950	47.21	45.99	1.970
4	f1_fuel_1994_12_2_0_10	2	1994	chickasaw	gas	mcf	40533.0	1.000	2.77	2.77	2.570

[3]:
```
len(df.report_year.unique())
```

[3]:
```
25
```

[4]:
```
df.columns
```

[4]:
```
Index(['record_id', 'utility_id_ferc1', 'report_year', 'plant_name_ferc1',
       'fuel_type_code_pudl', 'fuel_unit', 'fuel_qty_burned',
       'fuel_mmbtu_per_unit', 'fuel_cost_per_unit_burned',
       'fuel_cost_per_unit_delivered', 'fuel_cost_per_mmbtu'],
      dtype='object')
```

1 Question 1: Identity matrix

[5]:
```
np.identity(3)
```

[5]:
```
array([[1., 0., 0.],
       [0., 1., 0.],
       [0., 0., 1.]])
```

2 Question 2: Imputation techniques (Categorical and mode imputation)

[6]:
```
df.fuel_unit.mode(dropna=True)
```

[6]:
```
0    mcf
dtype: object
```

3 Question 3: 2nd and 3rd Lowest correlation with Fuel Cost Per Unit Burned

[7]:
```
df.corr()
#2nd lowest: fuel_mmbtu_per_unit
#3rd lowest: fuel_cost_per_unit_delivered
# Note: least correlation if nearest to zeros
```

[7]:

	utility_id_ferc1	report_year	fuel_qty_burned	\
utility_id_ferc1	1.000000	0.093323	-0.057447	
report_year	0.093323	1.000000	0.012952	
fuel_qty_burned	-0.057447	0.012952	1.000000	
fuel_mmbtu_per_unit	-0.066946	-0.110853	-0.080946	
fuel_cost_per_unit_burned	-0.037863	0.013599	-0.018535	

```
fuel_cost_per_unit_delivered         -0.016414    -0.014043        -0.003551
fuel_cost_per_mmbtu                   0.006122     0.010261        -0.001896

                              fuel_mmbtu_per_unit  fuel_cost_per_unit_burned  \
utility_id_ferc1                        -0.066946                  -0.037863
report_year                             -0.110853                   0.013599
fuel_qty_burned                         -0.080946                  -0.018535
fuel_mmbtu_per_unit                      1.000000                  -0.010034
fuel_cost_per_unit_burned               -0.010034                   1.000000
fuel_cost_per_unit_delivered            -0.009039                   0.011007
fuel_cost_per_mmbtu                     -0.005884                  -0.000437

                              fuel_cost_per_unit_delivered  \
utility_id_ferc1                                 -0.016414
report_year                                      -0.014043
fuel_qty_burned                                  -0.003551
fuel_mmbtu_per_unit                              -0.009039
fuel_cost_per_unit_burned                         0.011007
fuel_cost_per_unit_delivered                      1.000000
fuel_cost_per_mmbtu                              -0.000109

                              fuel_cost_per_mmbtu
utility_id_ferc1                         0.006122
report_year                              0.010261
fuel_qty_burned                         -0.001896
fuel_mmbtu_per_unit                     -0.005884
fuel_cost_per_unit_burned               -0.000437
fuel_cost_per_unit_delivered            -0.000109
fuel_cost_per_mmbtu                      1.000000
```

4 Question 4: Percentage change in the fuel cost per unit burned in 1998 compared to 1994

[8]:
```
df_fuel_type_coal = df[df.fuel_type_code_pudl.str.contains('coal')].copy()
df_fuel_type_coal = df_fuel_type_coal[['report_year',␣
 ↪'fuel_cost_per_unit_burned']]

# slicing data to retain report less and equal to 1998
df_coal_1994_1998 = df_fuel_type_coal[df_fuel_type_coal.report_year.le(1998)]

# finding the percent change for 1st index and last index
df_coal_1994_1998.iloc[[0,-1],[1,1]].pct_change()*100
```

[8]:
```
      fuel_cost_per_unit_burned  fuel_cost_per_unit_burned
0                           NaN                        NaN
5716                  33.808499                  33.808499
```

5 Question 5: Year with highest average fuel cost per unit delivered

[9]:
```
#df[df.fuel_cost_per_unit_delivered == df.fuel_cost_per_unit_delivered.
↪max()]['report_year']
df[['report_year','fuel_cost_per_unit_delivered']].
↪nlargest(1,columns='fuel_cost_per_unit_delivered')
```

[9]:
```
      report_year  fuel_cost_per_unit_delivered
3564         1997                     7964521.0
```

6 Question 6: standard deviation and 75th percentile of the measure of energy per unit

75th percentile of the measured energy per unit:

[10]:
```
round(df.fuel_mmbtu_per_unit.quantile(0.75), 2)
```

[10]:
```
17.01
```

Standard deviation of the measured energy per unit:

[11]:
```
round(df.fuel_mmbtu_per_unit.std(),2)
```

[11]:
```
10.6
```

7 Question 7: skewness and kurtosis for fuel qty burned

[12]:
```
df.fuel_qty_burned.skew()
```

[12]:
```
15.851495469109503
```

[13]:
```
df.fuel_qty_burned.kurtosis()
```

[13]:
```
651.3694501337732
```

8 Question 8: Missing values

[14]:
```
df.isnull().sum()  # the sum of 'NaN' is 180 at feature 'fuel_unit'
```

[14]:
```
record_id                         0
utility_id_ferc1                  0
report_year                       0
plant_name_ferc1                  0
fuel_type_code_pudl               0
fuel_unit                       180
fuel_qty_burned                   0
fuel_mmbtu_per_unit               0
fuel_cost_per_unit_burned         0
fuel_cost_per_unit_delivered      0
fuel_cost_per_mmbtu               0
dtype: int64
```

[15]:
```
# count the no. of rows for missing values divide by total no. of rows in the␣
↪dataset.
round(len(df[df.fuel_unit.isnull()]['fuel_unit'])/len(df)*100,3)
```

[15]:
```
0.61
```

9 Question 9: Fuel type code has the lowest average fule cost per unit burned

[16]:
```
df[['fuel_cost_per_unit_burned','fuel_type_code_pudl']].
↪groupby('fuel_type_code_pudl').mean()
```

[16]:

fuel_type_code_pudl	fuel_cost_per_unit_burned
coal	67.421830
gas	13.659397
nuclear	4955.157002
oil	168.877086
other	18.253856
waste	19.518122

10 Question 10: np.extend()

[17]:
```
A=[1,2,3,4,5,6]
B=[13,21,34]
```

[18]:
```
A.append(B) # append the whole list of B in to A. This is not the answer
A
```

[18]:
```
[1, 2, 3, 4, 5, 6, [13, 21, 34]]
```

[19]:
```
A=[1,2,3,4,5,6]
B=[13,21,34]
```

[20]:
```
A.extend(B) # append elements of B to A list.
A
```

[20]:
```
[1, 2, 3, 4, 5, 6, 13, 21, 34]
```