Data Analytics

Course: 18-899

Recitation 1

Adolphe & Felix

Spring 2019

Carnegie Mellon University

Assignment 1

Question 1:

Procedures:

- Download the CSV file posted on piazza of the historical daily weather data for France for 2017
- Save it as CSV file and load it into MATLAB/Jupiter Notebook, etc.,…
- Fill any missing gaps using linear interpolation

Question 1 (cont'd)

- **Missing values** Hint: use dataframe.info() to get the summary of your dataframe

```
Data columns (total 21 columns):
Date                                365 non-null object
high Temp. (°C)                     365 non-null int64
avg Temp. (°C)                      365 non-null int64
low Temp. (°C)                      365 non-null int64
high Dew Point (°C)                 365 non-null int64
avg Dew Point (°C)                  365 non-null int64
low Dew Point (°C)                  365 non-null int64
high Humidity (%)                   365 non-null int64
avg Humidity (%)                    365 non-null int64
low Humidity (%)                    365 non-null int64
high Sea Level Press. (hPa)         365 non-null int64
avg Sea Level Press. (hPa)          365 non-null int64
low Sea Level Press. (hPa)          365 non-null int64
high Visibility (km)                365 non-null object
avg Visibility (km)                 365 non-null object
low Visibility (km)                 365 non-null object
high Wind (km/h)                    365 non-null int64
avg Wind (km/h)                     365 non-null int64
high Gust Wind (km/h)               365 non-null object
sum Precip. (mm)                    365 non-null float64
Events                              226 non-null object
dtypes: float64(1), int64(14), object(6)
```

Question 1 (cont'd)

- Be careful of the (-, ?) .

```
Data columns (total 21 columns):
Date                                365 non-null object
high Temp. (°C)                     365 non-null int64
avg Temp. (°C)                      365 non-null int64
low Temp. (°C)                      365 non-null int64
high Dew Point (°C)                 365 non-null int64
avg Dew Point (°C)                  365 non-null int64
low Dew Point (°C)                  365 non-null int64
high Humidity (%)                   365 non-null int64
avg Humidity (%)                    365 non-null int64
low Humidity (%)                    365 non-null int64
high Sea Level Press. (hPa)         365 non-null int64
avg Sea Level Press. (hPa)          365 non-null int64
low Sea Level Press. (hPa)          365 non-null int64
high Visibility (km)                363 non-null object
avg Visibility (km)                 363 non-null object
low Visibility (km)                 363 non-null object
high Wind (km/h)                    365 non-null int64
avg Wind (km/h)                     365 non-null int64
high Gust Wind (km/h)               69 non-null object
sum Precip. (mm)                    365 non-null float64
Events                              226 non-null object
```

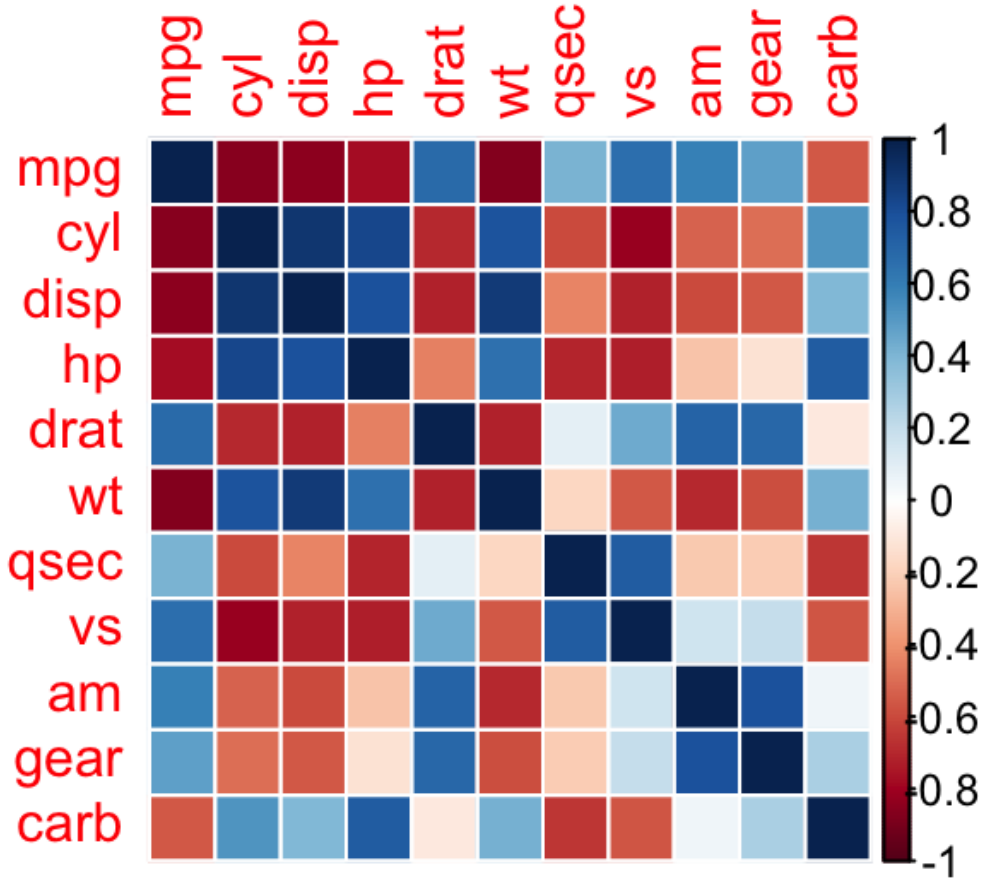
Question 2

Calculating and plotting the correlation matrix

A correlation matrix is a table showing correlation coefficients between variables. It is used to investigate the dependence between multiple variables at the same time.

Example of correlation matrix as a heat-map.

Source: http://www.sthda.com/english/wiki/visualize-correlation-matrix-using-correlogram



Question 3

Refer to Question 1

Question 4

Procedures:

- Extract the average/mean temperature data by indexing

- Synchronize Weather and Energy consumption timestamps

I. Extract weather and energy consumption dates Create time series for both weather and energy using the time series function (combines dates and data)

E.g.: you can use synchronize function for MATLAB and join, merge, etc.,… for python

Question 5

Fitting a quadratic model to the energy versus temperature.

A quadratic model is of the form **y = a2x^2 + a1x + a0**

Where a2, a1 and a0 are coefficients that minimize the squared error.

Sample code:

python

```
Import numpy as np
np.polyfit(x, y, 2)
```

MATLAB

```
model = polyfit(x,y,2);
```

Question 6

- The optimal minimum temperature corresponding to the lowest energy consumption.

Question 7

- Use multivariate linear regression
- instead of passing one variable you use multiple variables.
- MATLAB (Hint: you can use stewiselm,stepwisefit,…)
- Rsquared
- In python you use multiple stages

Question 8

- Feature variables: X, X^2

Same steps as Question 7

Question 9

- Feature variables: X, X^2, dummy variables of weekdays

Same steps as Question 7

Hint:

pandas: pd.get_dummies()

Matlab:

- dummyvar function or
- you can deal with it logically

Question 10

- Freestyle