Beijing Data Air-Quality

Nicolas Makaroff & Mathieu Pavard

08/12/2019

1. Recall on the DataBase
   1. PM2.5 distribution against different features
2. Data Engineering
   1. Emulate time series
   2. Add Humidity features
   3. Projection of the wind direction, hour and month on the trigonometric circle
   4. Week day category
3. Model Selection
4. Method Comparison

1. Recall on the DataBase

- 12 Database of 12 sites in Beijing
- 35 064 rows in each

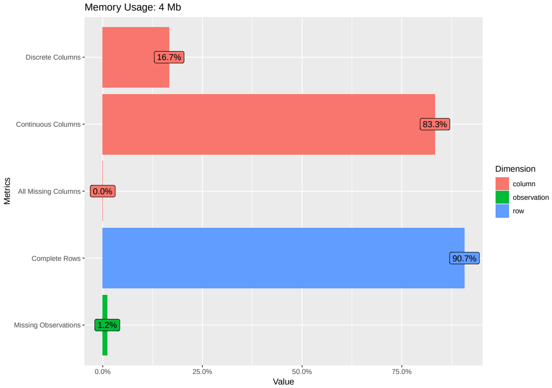

Figure 1: Description of the data of the Aotizhongxin data set

1.1. PM2.5 distribution against different features

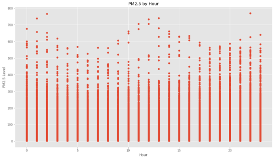

Figure 2: PM2.5 per hour in Aotizhonxin, Beijing

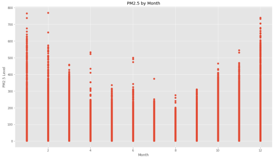

Figure 3: PM2.5 per month in Aotizhongxin, Beijing

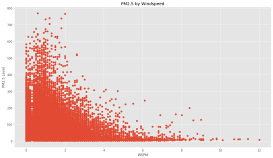

Figure 4: PM2.5 against wind speed in Aotizhongxin, Beijing

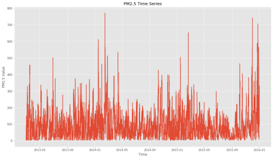

Figure 5: PM2.5 per day between 2013 - 2016 in Gucheng, Beijing

Season distribution of PM2.5

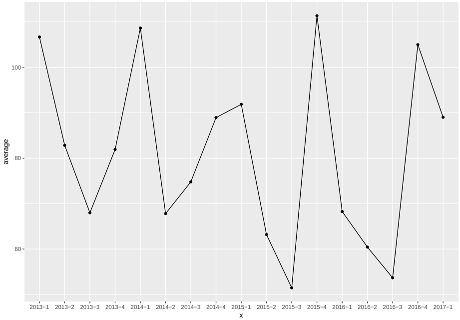

Figure 6: PM2.5 AVG per season in Beijing

2. Data Engineering

- Emulate time series
- Add Humidity variables
- Projection of the wind direction on the trigonometric circle
- Projection of hour and month on the trigonometric circle

Emulate time series and add humidity features

- Shift information on the timeline to add previous observation as features following **[1]**

```
def shift_timeline(tab):
    .
    .
    tab['PM2.5_5'] = tab['PM2.5'].shift(periods=5)
    tab['TEMP_5'] = tab.TEMP.shift(periods=5)
    tab['SPD_5'] = tab.WSPM.shift(periods=5)
    tab['DEWP_5'] = tab.DEWP.shift(periods=5)
    tab['WD_5'] = tab.wd.shift(periods=5)
```

[1] Mehdi Zamani Joharestani, Chunxiang Cao, Xiliang Ni, Barjeece Bashir, Somayeh Talebiesfandarani, PM2.5 Prediction Based on Random Forest, XGBoost,and Deep Learning Using Multisource Remote Sensing Data, 4 July 2019, Atmosphere 2019, 10, 373;

- Add humidity features

$$HUM = 100 * \frac{exp(\frac{17.625*DEWP}{243.04+DEWP})}{exp(\frac{17.625*TEMP}{243.04+TEMP})}$$

We used the formula out of **[2]**.

[2] Alduchov, O.A. and R.E. Eskridge, 1996: Improved Magnus Form Approximation of Saturation Vapor Pressure. J. Appl. Meteor., 35, 601–609, https://doi.org/10.1175/1520-0450(1996)035<0601:IMFAOS>2.0.CO;2

Projection on the trig. circle and week day category

```python
def change_wind_dir(df):
    df['WD_sin'] = np.sin(df.wd*(2.*np.pi/360))
    df['WD_cos'] = np.cos(df.wd*(2.*np.pi/360))
```

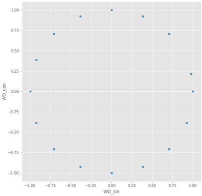

Figure 7: Wind direction

- We added week_day as a features

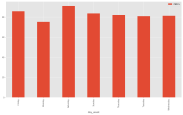

Figure 8: PM2.5 histogram on a week

Model Selection

3.1. Basic model

- Merge all station and take the mean for all features
- Split between train and test set

3.2. AIC Optimization

Initial AIC : 174873.98897595517

- After stepwise algorithm

Final AIC : 174868.47328581163

- No real improvement

```
Root Mean Squared Error: 17.88 (+/- 4.42) ug/m^3
R2: 0.98 (+/- 0.00)
```

Figure 9: Regression result for PM2.5 concentration in Beijing

Method used

- Simple Linear Regression
- Lasso Regression
- Ridge Regression
- ElasticNet Regression
- K-Nearest Neighbor Regression
- cross-validation for training
- k choosen using a gridsearch

4. Method comparison

- Adjusted R^2 because of the number of co-variates
- Classic Regression measure = **MSE** and **MAE**
- Different result if th size of the town isn't taken into account.
  - North
  - Center
  - South

Result for Beijing's center :

	Linear Regression	Ridge Regression	Lasso Regression	Knn
R2	0.98 (+/- 0.00)	0.98 (+/- 0.00)	0.98 (+/- 0.00)	0.97 (+/- 0.00)
RMSE_cv_train	10.29 (+/- 1.67)	10.29 (+/- 1.67)	10.59 (+/- 1.68)	13.93 (+/- 1.20)
RMSE_test	9.97	9.97	10.28	13.85
MAE_cv_test	5.91	5.91	5.92	7.83

left: expected values, **right:** predicted values