MRR project

Beijing's PM 2.5 pollution

LIU Hao

18/01/2018

Dataset descrption

Number of instance : 43824

Number of variables : 13

No: row number
Year: year of data in this row
Month: month of data in this row
Day: day of data in this row
Hour: hour of data in this row
DEWP: Dew Point (â„ƒ)
PM2.5: PM2.5 concentration (ug/m^3) ----- (Y)

TEMP: Temperature (â„ƒ)
PRES: Pressure (hPa)
cbwd: Combined wind direction
Iws: Cumulated wind speed (m/s)
Is: Cumulated hours of snow
Ir: Cumulated hours of rain

Object and problems

By building a linear regression model, to predict the PM 2.5 value at a given situation in the future.

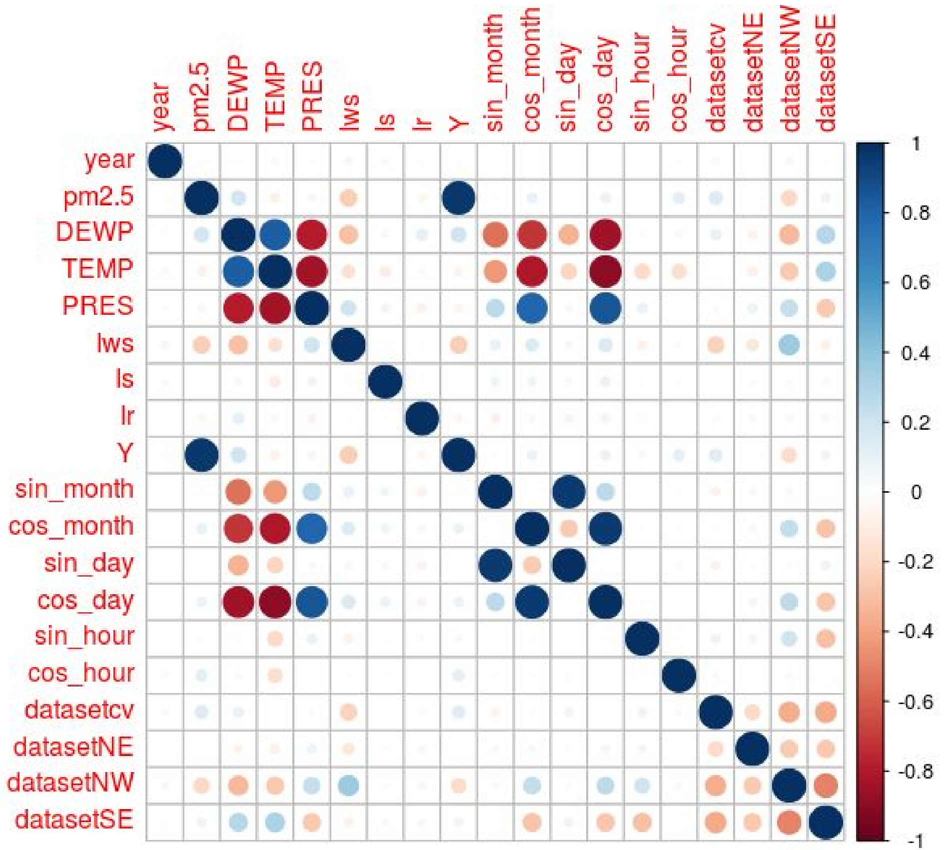
Problems:

1: Certain PM2.5 values (Y) are missing NA
2: Quite a lot instances, low performance with a normal computer
3: Four variables they are in some way related with each other. (Year, Month, Day, Hour)
4: Categorical variable instead of numerical variable (Combined wind direction)

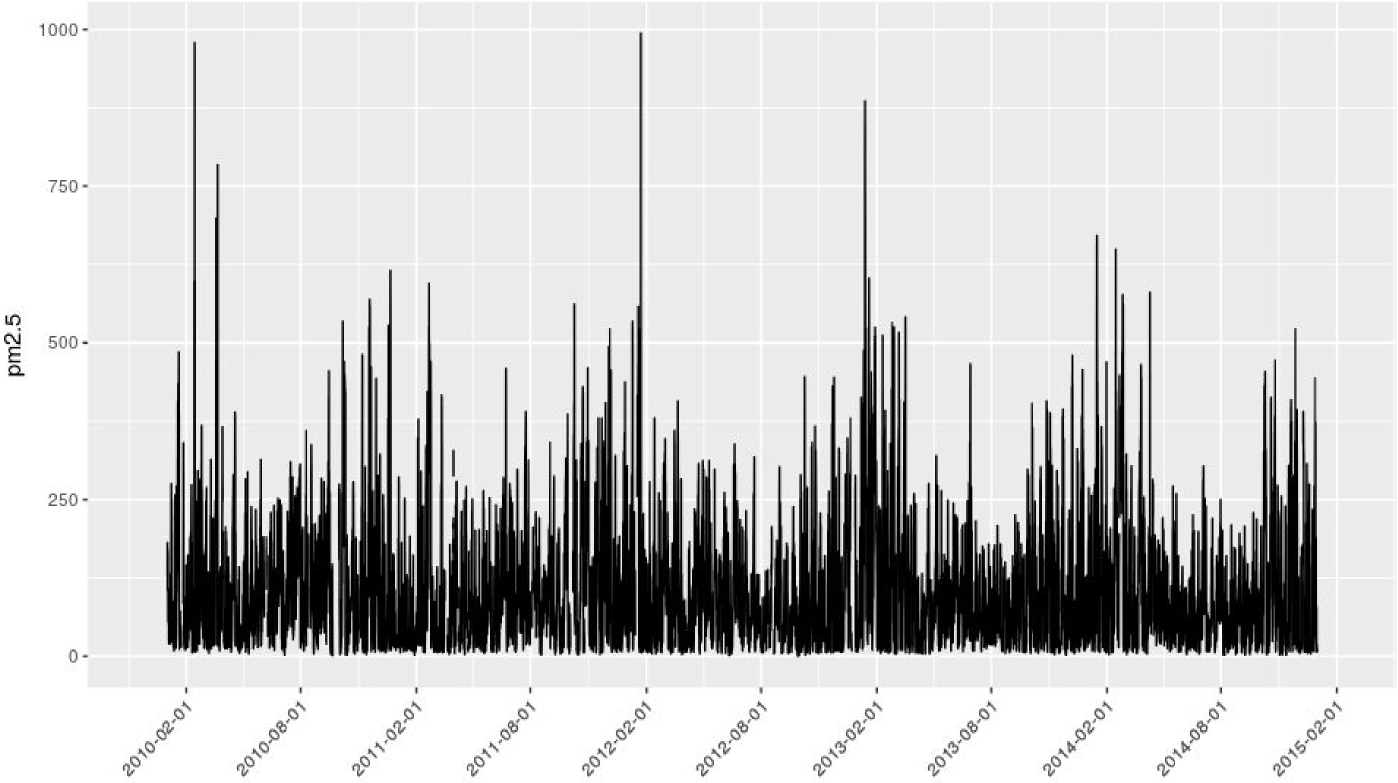
Study of dataset

Correlation matrix:

```
correlation = cor(dataset)
corrplot(correlation, method="circle")
```



Dependence of time



Data processing

Change of variables : Month, year, days, hour

$$X\_cos = \cos(i*2\pi/X)$$

$$X\_sin = \sin(i*2\pi/X)$$

Missing values:
```
dataset = na.omit(dataset)
```

Dummy Coding :
```
cbwd = dummy(dataset[,'cbwd'])
```

Time serie :
```
Y = dataset[,'pm2.5']
Y = Y[1:length(Y)-1]
dataset = cbind(dataset,Y)
```

Result : 19 variables

Data separation

Separation dataset to training and test sets :

```
sample = sample(c(1:nrow(dataset)),n)

subset = dataset[sample,]

training_size = 0.8*nrow(subset)
test_size = 0.2*nrow(subset)
train_Y = Y[1:training_size]
train_X = X[1:training_size,]
test_Y = Y[(training_size+1):nrow(subset)]
test_X = X[(training_size+1):nrow(subset),]
```

Normalisation

Idea :

```
(trainData - mean(trainData)) / sd(trainData)
```

```
(testData - mean(trainData)) / sd(trainData)
```

Implementation :

```
mean <- colMeans(train_X)
```

```
sd <- colSds(as.matrix(train_X))
```

```
train_X = scale(train_X)
```

First trial

Linear models :

```
reg =
 lm('train_Y~.', data = as.data.frame(cbind(train_X,train_Y)))
```

Mean square error :

```
sum((train_Y-reg$fitted.values)^2)/training_size
```

MSE(train) = 547.4764

MSE(test) = 573.7022

```
##                 Pr(>|t|)
## (Intercept)     0.634800
## year            0.703091
## DEWP            < 2e-16 ***
## TEMP            < 2e-16 ***
## PRES            3.41e-08 ***
## Iws             0.122921
## Is              0.000431 ***
## Ir              3.88e-16 ***
## Y               < 2e-16 ***
## sin_month       0.902409
## cos_month       0.717136
## sin_day         0.256935
## cos_day         0.021958 *
## sin_hour        < 2e-16 ***
## cos_hour        0.850291
## cv              0.002876 **
## NE              7.82e-13 ***
## NW              < 2e-16 ***
## SE              NA
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Model selection

Forward selection :

`regbackward = step(reg,direction = 'backward');`

train_Y ~ Y + NW + cos_month + sin_hour + NE +   DEWP + TEMP + Ir + cv + sin_day + PRES

AIC : 12   202769.9   MSE(train) = 547.4968

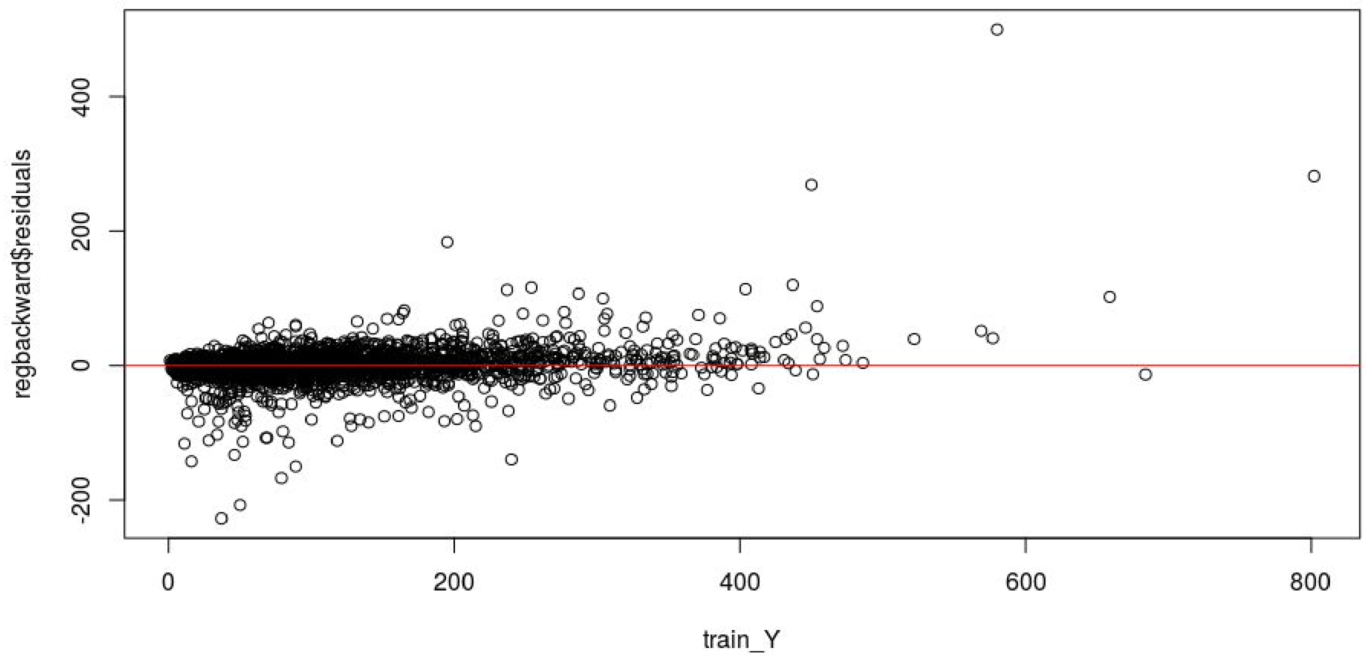
Backward selection :

`regforward = step(reg,list(upper=reg),direction = 'forward');`

train_Y ~ DEWP + TEMP + PRES + Ir + Y + cos_month + sin_day + sin_hour + cv + NE + NW

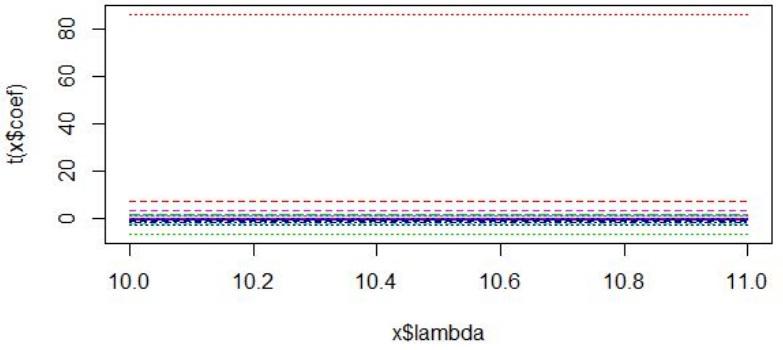
AIC : 12   202769.9   MSE(train) = 547.4764

Residuals plot



L2 regularization

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}}(||Y - X\beta||^2 + \lambda||\beta||^2)$$



Ridge :

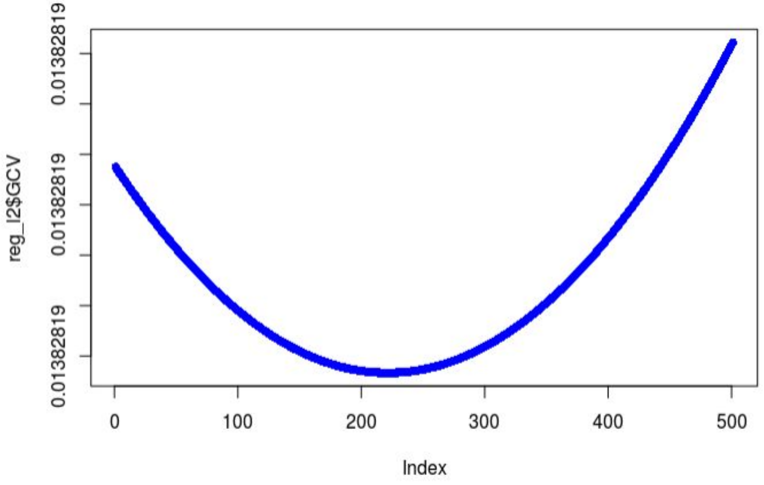
ridge <- lm.ridge('pm2.5~.', data = as.data.frame(subset), lambda = seq(10,15,0.01))

smallest value of GCV at 12.2

GCV.MIN<-ridge$GCV[which.min(reg_l2$GCV)]

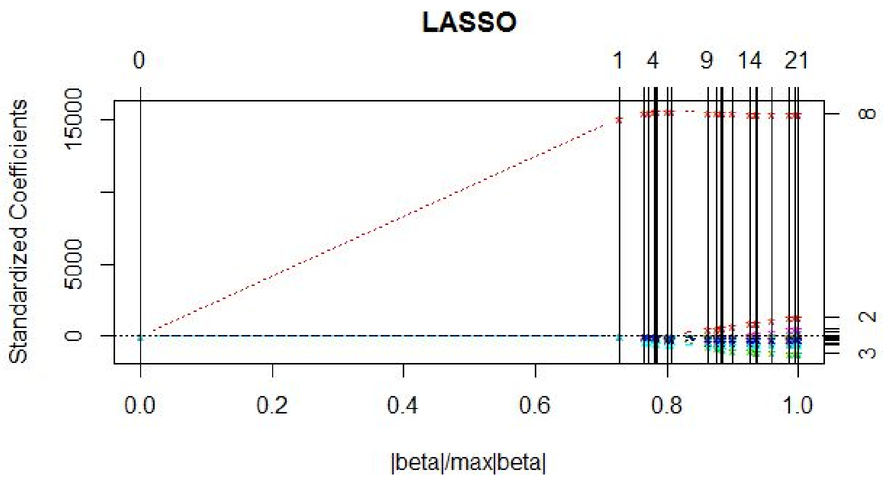
MSE(train) = 17784.68

MSE(test) = 17949.88



L1 regularization

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}}(||Y - X\beta||^2 + \lambda||\beta||_1)$$



Lasso :

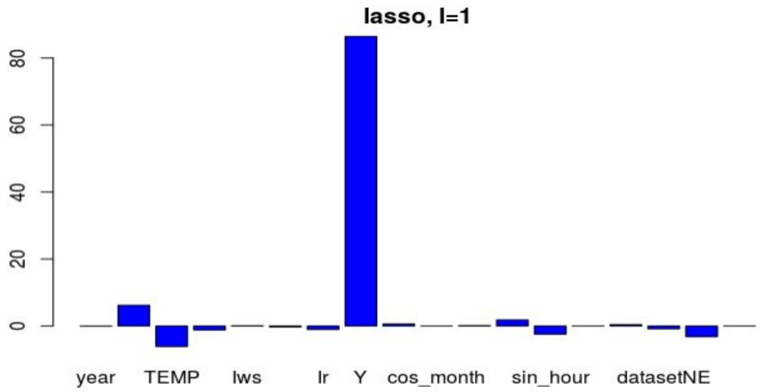
lasso <- lars(as.matrix(train_X),as.matrix(train_Y),type="lasso")

predict.lars(lasso,train_X,type='coefficients',mode='lambda',s=13.0)

4 variables removed : cos_month, sin_day, cos_hour, datasetSE

MSE(train) = 547.8338

MSE(test) = 574.3286



Polynomial regression

Expression :

$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_{11} x_1^2 + \beta_{22} x_2^2 + \beta_{33} x_3^2 + \ldots + \varepsilon$

Implementation :

```
poly_reg <- lm(train_Y ~ poly(train_X[,1],train_X[,2]train_X[,3],....,degree = 2))
```

MSE(train) = 516.3504

MSE(test) = 65424.36   (Overfitting)

Elastic Net regularization

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}}(||Y - X\beta||^2 + \lambda((1-\alpha)||\beta||^2 + \alpha||\beta||_1))$$

MSE(train) = 569.9181

MSE(test) = 592.3498

cvm <dbl>	lambda.1se <dbl>	alpha <dbl>
578.2529	5.541698	0.05
578.3614	5.314240	0.10
575.9144	4.683408	0.15
579.9925	4.643392	0.20
595.8421	5.914875	0.25
576.5218	3.728655	0.30
569.3507	2.653367	0.35
582.3067	4.057230	0.40
589.1240	4.767484	0.45
582.6376	3.909558	0.50
586.2465	4.280979	0.55
576.5973	2.968537	0.60
585.4732	3.975543	0.65
580.5482	3.363626	0.70
585.1212	3.781400	0.75
580.8580	3.230129	0.80
571.7995	1.909285	0.85
584.6648	3.458401	0.90
578.3657	2.720108	0.95