ML_Group_Project

2024-11-20

```
library(boot)
```

Data Exploration

ADD SAMARA'S part

```
#cor(df_train[,-c(1,10)])
```

It is worth to note that "decane_toluene" is not in the test data set

Missing Data

Missing data was found in the 'parentspecies' attribute. According to the definition of the 'parentspecies' attribute, missing values imply a meaning.They are not missing randomly but due to difficulty in retrieving the 'parentspecies'. Therefore, a new level was created as 'Unknown'.

```
test_data$parentspecies[test_data$parentspecies == ""] <- "Unknown"
train_data$parentspecies[train_data$parentspecies == ""] <- "Unknown"
```

Dummy model

A dummy model is a supervised learning model that gives the same constant output regardless of the values of the covariates. We first built a dummy model and calculated the training error and the cross validation error.

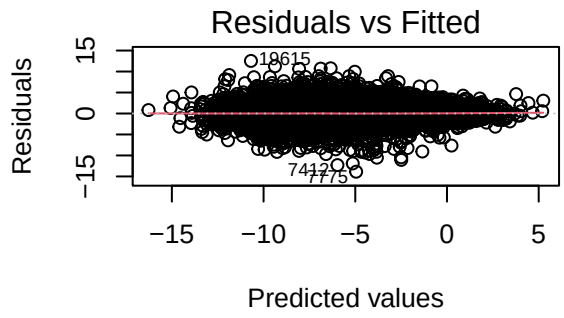
model	train	cv	kaggle_score
Dummy	9.735229	9.736679	-1e-04

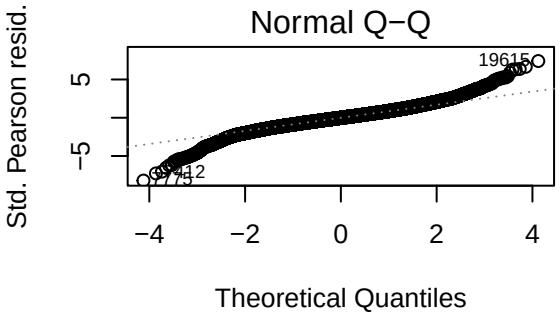
OLS as a baseline model

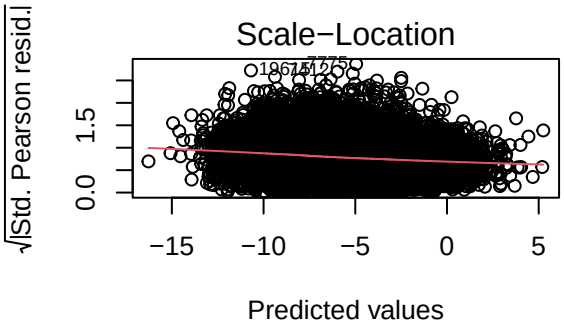
ten fold cross validation was done to get the cross validation error

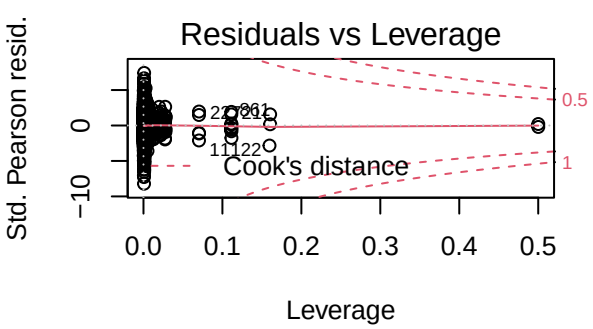
```
set.seed(123)
ols_fit = glm(log_pSat_Pa ~ ., data = train_data[,-1])
cv_error_5 = cv.glm(train_data[,-1] , ols_fit , K = 10)$delta[1]
error_train = mean((train_data$log_pSat_Pa - predict(ols_fit, train_data))^2)
```

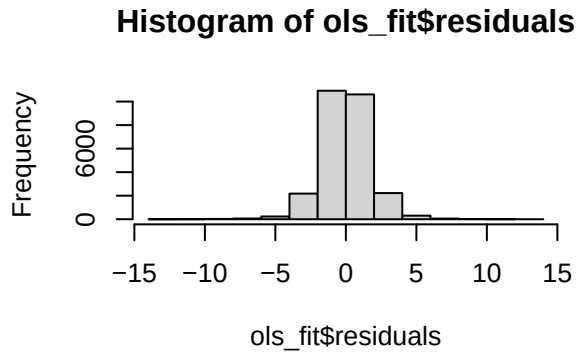
Plotting


Residuals vs Fitted
Residuals
Predicted values
19615


Normal Q−Q
Std. Pearson resid.
Theoretical Quantiles
19615


Scale−Location
√|Std. Pearson resid.|
Predicted values


Residuals vs Leverage
Std. Pearson resid.
Cook's distance
0.5
1
Leverage


Histogram of ols_fit$residuals
Frequency
ols_fit$residuals

model	train	cv	kaggle_score
Dummy	9.735229	9.736679	-0.0001
ols	2.855265	2.862134	0.7163

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading
```

Lasso

Before moving to non-linear models we tried regularization using Lasso

"decane_toluene" is not in the test data set. so it was handled in a way so that both train and test dataset has the same number of levels

Level	Freq_train	Freq_test
apin	6165	1195
apin_decane	46	5
apin_decane_toluene	9	2
apin_toluene	37	5
decane	2218	381
decane_toluene	2	0
toluene	17950	3379
Unknown	210	33

```
## Loading required package: Matrix
```

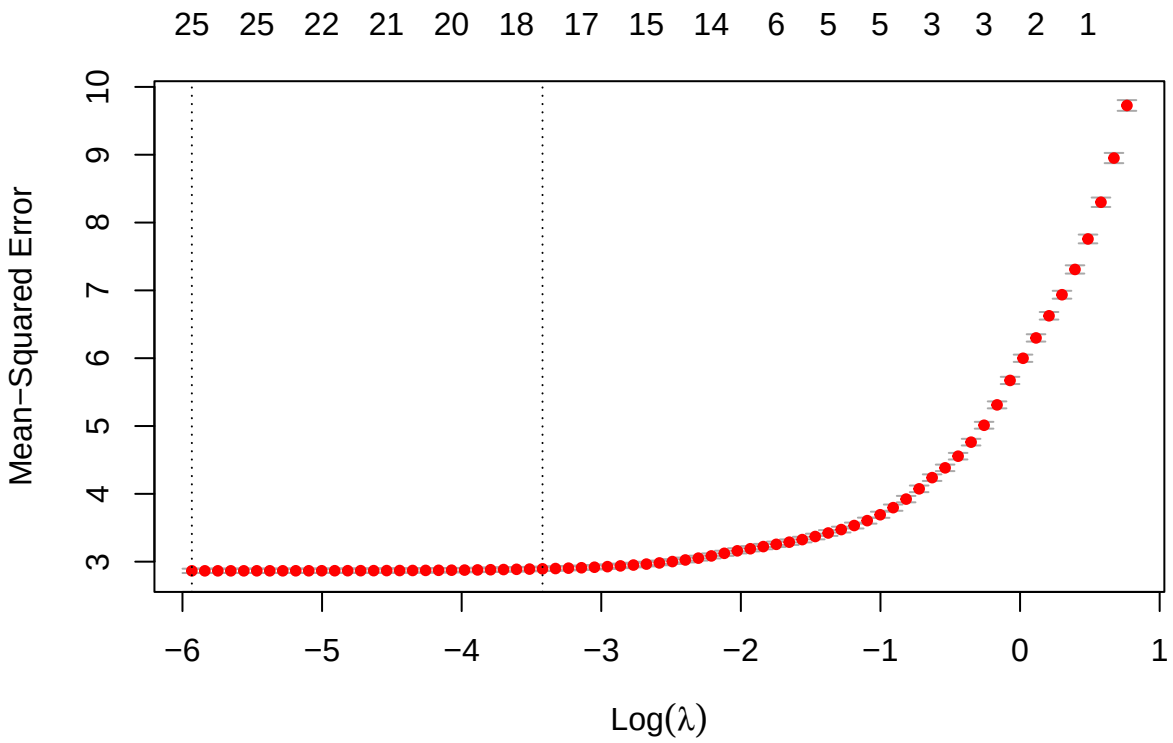
```
## Loaded glmnet 4.1-7
```

Cross validation to obtain the best lambda(tuning parameter)

best lambda obtained was 0.002651092

Plot

```
plot(cvfit)
```



WE can use lasso as a variable selection method

```
## <sparse>[ <logic> ]: .M.sub.i.logical() maybe inefficient
```

	x
(Intercept)	6.2437263
MW	-0.0026108
NumOfAtoms	0.0000000
NumOfC	-0.7950384
NumOfO	-0.0008376
NumOfN	0.0000000
NumHBondDonors	-1.7561728
NumOfConf	-0.0022532
NumOfConfUsed	-0.0000226
parentspeciesapin	0.0000000
parentspeciesapin_decane	-0.4952347
parentspeciesapin_decane_toluene	0.0766192
parentspeciesapin_toluene	-0.2154127
parentspeciesdecane	-0.8308856
parentspeciesdecane_toluene	-0.4904175
parentspeciestoluene	0.0000000

	x
parentspeciesUnknown	-0.8785030
C.C..non.aromatic.	-0.8968921
C.C.C.O.in.non.aromatic.ring	0.0000000
hydroxyl..alkyl.	0.0000000
aldehyde	-0.2922533
ketone	0.0637678
carboxylic.acid	-1.0328902
ester	-0.2404483
ether..alicyclic.	-0.6441586
nitrate	0.0000000

model	train	cv	kaggle_score
Dummy	9.735229	9.736679	-0.0001
ols	2.855265	2.862134	0.7163
Lasso	3.780056	2.857745	0.7160