

Gas Turbine CO Emission Analysis

Aayushi Gupta, Kyle Kaminski, Rosa Lin, Ruben Martinez

Introduction

The combined cycle power plant, also known as combined cycle gas turbine plant, is an assembly of heat engines that combine to generate electricity (Tüfekci). A combined-cycle power plant (CCPP) is made up of gas turbines, steam turbines, and heat recovery steam generators. The electricity is generated and combined in one cycle by gas and steam turbines and then transferred from one turbine to another.

We are interested in identifying the process variables that impact carbon monoxide emissions. By determining the process variables that impact carbon monoxide emissions we will be able to find opportunities to reduce carbon monoxide emissions.

Gas Turbine CO and NOx Emission Data Set

The data comes from a gas turbine located in Turkey that studies the flue gas emissions of specifically carbon monoxide (CO) and nitrogen oxide (NOx) gases. The data set provides hourly statistics of 11 sensors. Data points were collected from a gas turbine from Jan 01 2011 to Dec 13 2015.

Description

The data file `gt_2015.csv` has 7384 observations and 11 variables from the UCI Gas Turbine CO and NOx Emission Data Set. We are going to explore and analyze the following variables:

- AT - Ambient Temperature
- AP - Ambient Pressure
- AH - Ambient Humidity
- AFDP - Air Filter Difference Pressure
- GTEP - Gas Turbine Exhaust Pressure
- TIT - Turbine Inlet Temperature
- TAT - Turbine After Temperature
- TEY - Turbine Energy Yield
- CDP - Compressor Discharge Pressure

Here's a quick peek at the data set:

AT	AP	AH	AFDP	GTEP	TIT	TAT	TEY	CDP	CO	NOX
1.95320	1020.1	84.985	2.5304	20.116	1048.7	544.92	116.27	10.799	7.4491	113.250
1.21910	1020.1	87.523	2.3937	18.584	1045.5	548.50	109.18	10.347	6.4684	112.020
0.94915	1022.2	78.335	2.7789	22.264	1068.8	549.95	125.88	11.256	3.6335	88.147
1.00750	1021.7	76.942	2.8170	23.358	1075.2	549.63	132.21	11.702	3.1972	87.078
1.28580	1021.6	76.732	2.8377	23.483	1076.2	549.68	133.58	11.737	2.3833	82.515
1.83190	1021.7	76.411	2.8410	23.495	1076.4	549.92	133.58	11.829	2.0812	81.193

AT	AP	AH	AFDP	GTEP	TIT	TAT	TEY	CDP	CO	NOX
1.007500	1021.7	76.942	2.8170	23.358	1075.2	549.63	132.21	11.702	3.1972	87.078
2.074000	1022.0	75.974	2.7981	22.945	1073.7	549.98	131.53	11.687	2.2529	83.171
0.052442	1024.0	64.823	2.7916	23.298	1070.9	550.23	130.43	11.546	3.6518	86.895
-1.084100	1022.3	70.733	2.8280	22.604	1071.9	550.31	130.41	11.526	1.7751	83.696
11.353000	1006.9	72.516	3.0802	26.554	1076.3	550.04	131.77	11.838	3.5554	69.506
7.573300	1009.0	71.839	2.9088	23.677	1076.0	550.17	132.51	11.760	3.5993	73.652

AT	AP	AH	AFDP	GTEP	TIT	TAT	TEY	CDP	CO	NOX
0.68481	1026.7	56.029	4.0703	34.213	1100.0	527.71	168.83	14.358	2.9790	59.354
1.99950	1026.3	54.000	3.9830	34.171	1100.1	530.64	166.13	14.182	2.5793	59.432
3.16630	1025.7	51.350	4.0683	35.162	1099.8	528.21	167.49	14.384	2.2228	58.432
3.64860	1025.3	51.649	4.0375	35.282	1100.0	530.04	165.89	14.257	2.2119	60.172
3.92070	1025.2	49.619	4.0455	34.648	1099.9	529.73	166.00	14.253	2.3487	60.255
3.91930	1025.1	51.181	4.0400	33.944	1100.0	529.56	166.46	14.283	2.3095	59.778

Here's some descriptive statistics of the data set:

```

##      AT          AP          AH          AFDP
## Min. :-6.235    Min. : 989.4   Min. :24.09   Min. :2.369
## 1st Qu.:11.073  1st Qu.:1009.7  1st Qu.:59.45  1st Qu.:3.117
## Median :17.456  Median :1014.0  Median :70.95  Median :3.538
## Mean   :17.225  Mean   :1014.5  Mean   :68.65  Mean   :3.599
## 3rd Qu.:23.685  3rd Qu.:1018.3  3rd Qu.:79.65  3rd Qu.:4.195
## Max.  :37.103   Max.  :1036.6   Max.  :96.67  Max.  :5.239
##      GTEP         TIT         TAT          TEY
## Min. :17.70     Min. :1016     Min. :516.0   Min. :100.0
## 1st Qu.:23.15   1st Qu.:1070   1st Qu.:544.7  1st Qu.:126.3
## Median :25.33   Median :1080     Median :549.7  Median :131.6
## Mean   :26.13   Mean   :1079     Mean   :546.6  Mean   :134.0
## 3rd Qu.:30.02   3rd Qu.:1100   3rd Qu.:550.0  3rd Qu.:147.2
## Max.  :40.72   Max.  :1100     Max.  :550.6  Max.  :179.5
##      CDP          CO          NOX
## Min. : 9.871   Min. : 0.2128  Min. : 25.91
## 1st Qu.:11.466  1st Qu.: 1.8082  1st Qu.: 52.40
## Median :11.933  Median : 2.5334  Median : 56.84
## Mean   :12.097  Mean   : 3.1300  Mean   : 59.89
## 3rd Qu.:13.148  3rd Qu.: 3.7026  3rd Qu.: 65.09
## Max.  :15.159   Max.  :41.0970  Max.  :119.68

##      AT          AP          AH          AFDP
## Min. :-5.785    Min. : 990.8   Min. :31.62  Min. :2.644
## 1st Qu.:10.505  1st Qu.:1010.6  1st Qu.:63.24  1st Qu.:3.207
## Median :15.460  Median :1015.2  Median :73.42  Median :3.344
## Mean   :15.661  Mean   :1015.1  Mean   :71.45  Mean   :3.441
## 3rd Qu.:20.797  3rd Qu.:1019.2  3rd Qu.:81.10  3rd Qu.:3.751
## Max.  :35.406   Max.  :1035.3   Max.  :96.67  Max.  :4.287
##      GTEP         TIT         TAT          TEY          CDP
## Min. :21.70     Min. :1052     Min. :529.2   Min. :127.0  Min. :11.29
## 1st Qu.:23.59   1st Qu.:1074   1st Qu.:549.8  1st Qu.:129.7  1st Qu.:11.65

```

```

## Median :24.20   Median :1077    Median :550.0   Median :130.4   Median :11.76
## Mean   :24.52   Mean   :1077    Mean   :549.9    Mean   :130.3   Mean   :11.77
## 3rd Qu.:24.99   3rd Qu.:1079    3rd Qu.:550.1   3rd Qu.:131.1   3rd Qu.:11.88
## Max.   :30.89   Max.   :1087    Max.   :550.5    Max.   :133.0   Max.   :12.28
##          CO           NOX
## Min.   : 0.4843   Min.   : 35.60
## 1st Qu.: 2.0296   1st Qu.: 52.57
## Median : 2.6196   Median : 57.35
## Mean   : 2.7564   Mean   : 59.60
## 3rd Qu.: 3.4285   3rd Qu.: 66.56
## Max.   :36.4540   Max.   :102.33

##          AT          AP          AH          AFDP
## Min.   :-6.235   Min.   :1006   Min.   :44.25   Min.   :3.608
## 1st Qu.: 3.041   1st Qu.:1021   1st Qu.:67.35   1st Qu.:4.084
## Median : 5.996   Median :1024   Median :75.36   Median :4.201
## Mean   : 5.277   Mean   :1023   Mean   :74.19   Mean   :4.293
## 3rd Qu.: 8.255   3rd Qu.:1025   3rd Qu.:82.20   3rd Qu.:4.516
## Max.   :15.830   Max.   :1037   Max.   :93.52   Max.   :5.239
##          GTEP        TIT        TAT          TEY          CDP
## Min.   :30.99   Min.   :1096   Min.   :516.0   Min.   :160.0   Min.   :13.63
## 1st Qu.:32.81   1st Qu.:1100   1st Qu.:529.7   1st Qu.:162.0   1st Qu.:14.00
## Median :33.43   Median :1100   Median :531.6   Median :163.8   Median :14.13
## Mean   :33.72   Mean   :1100   Mean   :530.9   Mean   :164.7   Mean   :14.17
## 3rd Qu.:34.21   3rd Qu.:1100   3rd Qu.:533.3   3rd Qu.:166.4   3rd Qu.:14.25
## Max.   :40.72   Max.   :1100   Max.   :538.5   Max.   :179.5   Max.   :15.16
##          CO           NOX
## Min.   : 0.2128   Min.   :45.92
## 1st Qu.: 2.1237   1st Qu.:50.27
## Median : 2.4638   Median :57.37
## Mean   : 2.3864   Mean   :55.55
## 3rd Qu.: 2.7363   3rd Qu.:59.70
## Max.   :4.0948   Max.   :69.88

```

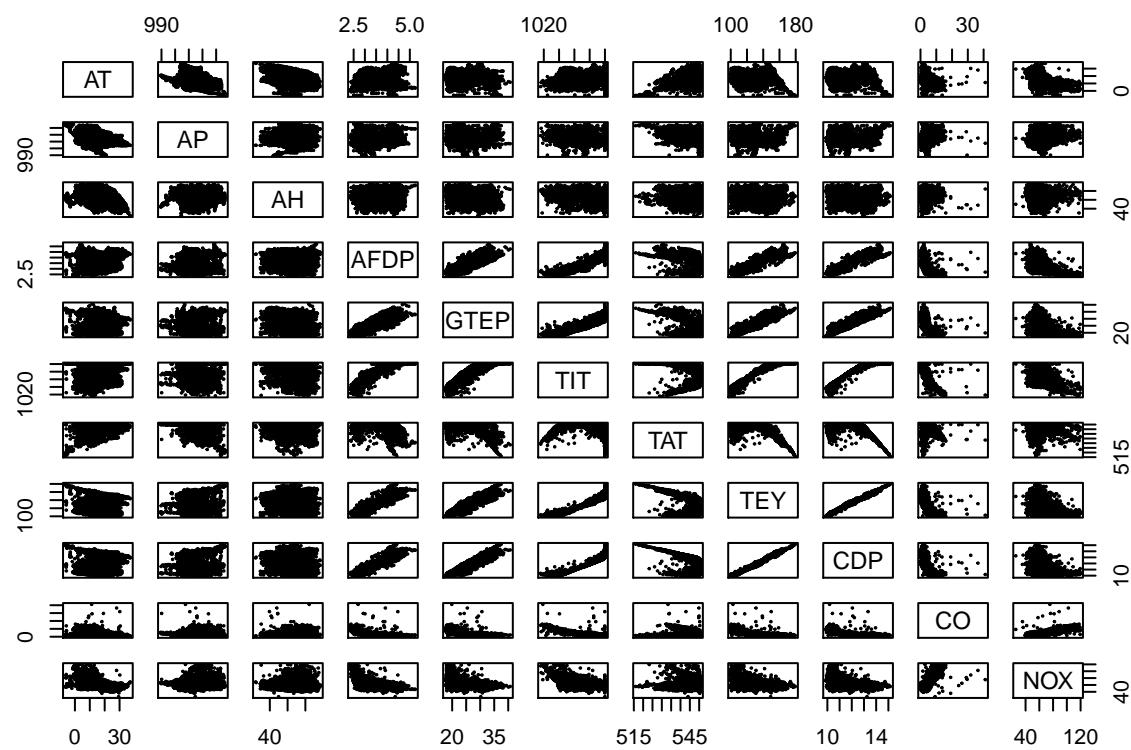
Goals

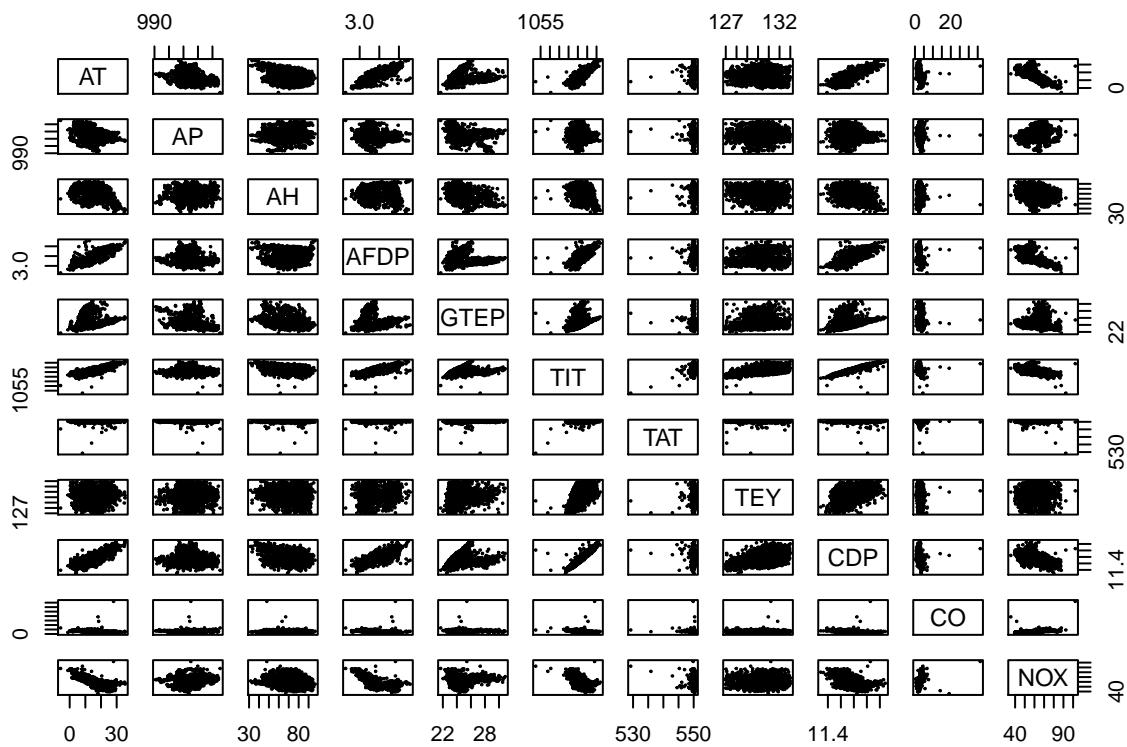
The goal for this project is to utilize this data set for the purpose of studying flue gas emissions, specifically carbon monoxide(CO) and nitrogen oxides (NOx). Our focus will be to find statistically significant relationships between the ambient and turbine variables and the emissions variables. We will limit the size of our model to more clearly demonstrate these relationships. Ultimately we will suggest which variables make the biggest impact on emission levels in order to decrease emissions overall.

Exploratory Data Analysis

Relationships between feature variables

Figure 1: Scatterplot Matrices to decide which feature variables have a linear relationship





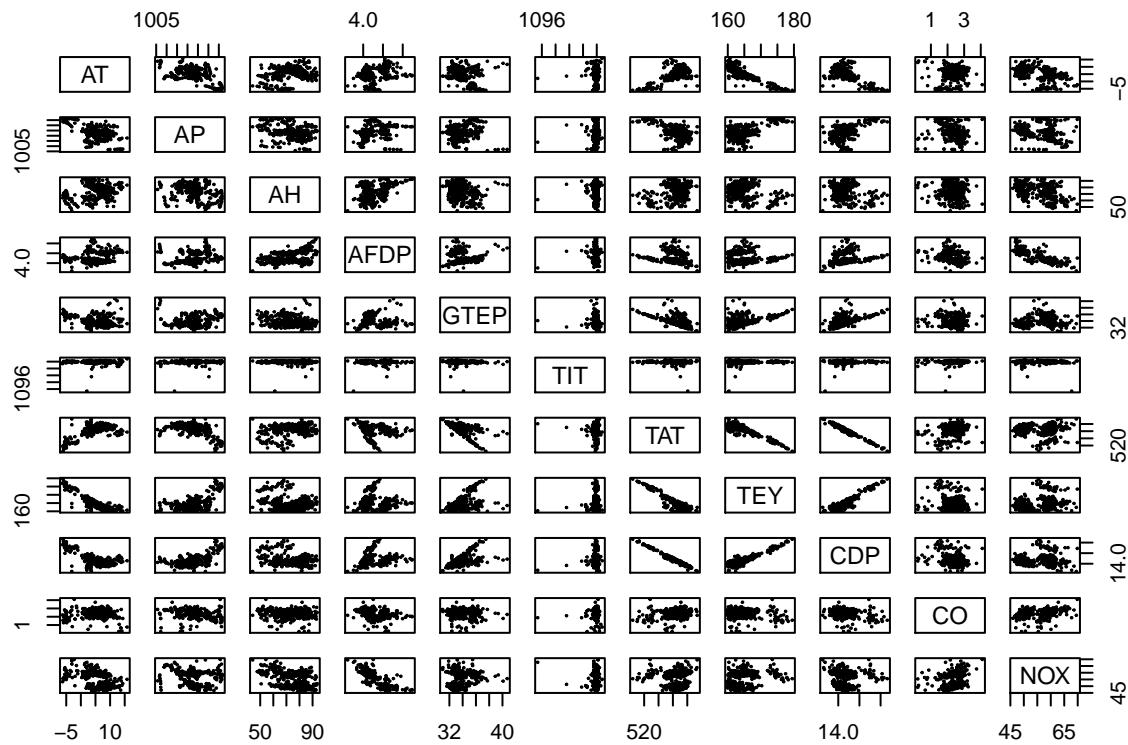


Figure 2:

Table 4: Pairwise Correlation Between Variables (All Data)

	AT	AP	AH	AFDP	GTEP	TIT	TAT	TEY	CDP	CO	NOX
AT	1.00	-0.49	-0.47	0.47	0.19	0.33	0.21	0.11	0.20	-0.39	-0.59
AP	-0.49	1.00	0.08	-0.09	-0.04	-0.08	-0.29	0.05	0.03	0.20	0.21
AH	-0.47	0.08	1.00	-0.25	-0.30	-0.26	0.03	-0.18	-0.22	0.16	0.07
AFDP	0.47	-0.09	-0.25	1.00	0.84	0.92	-0.52	0.88	0.92	-0.64	-0.58
GTEP	0.19	-0.04	-0.30	0.84	1.00	0.89	-0.62	0.93	0.94	-0.56	-0.37
TIT	0.33	-0.08	-0.26	0.92	0.89	1.00	-0.40	0.95	0.95	-0.74	-0.52
TAT	0.21	-0.29	0.03	-0.52	-0.62	-0.40	1.00	-0.63	-0.66	0.03	0.05
TEY	0.11	0.05	-0.18	0.88	0.93	0.95	-0.63	1.00	0.99	-0.62	-0.40
CDP	0.20	0.03	-0.22	0.92	0.94	0.95	-0.66	0.99	1.00	-0.61	-0.44
CO	-0.39	0.20	0.16	-0.64	-0.56	-0.74	0.03	-0.62	-0.61	1.00	0.68
NOX	-0.59	0.21	0.07	-0.58	-0.37	-0.52	0.05	-0.40	-0.44	0.68	1.00

Table 5: Pairwise Correlation Between Variables (Typical Energy Yield)

	AT	AP	AH	AFDP	GTEP	TIT	TAT	TEY	CDP	CO	NOX
AT	1.00	-0.33	-0.34	0.84	0.28	0.84	0.02	0.03	0.83	-0.45	-0.75
AP	-0.33	1.00	0.00	-0.18	-0.31	-0.13	-0.05	-0.03	-0.13	0.14	0.20
AH	-0.34	0.00	1.00	-0.13	-0.42	-0.33	0.01	-0.07	-0.32	-0.02	-0.13

	AT	AP	AH	AFDP	GTEP	TIT	TAT	TEY	CDP	CO	NOX
AFDP	0.84	-0.18	-0.13	1.00	0.12	0.82	0.00	0.18	0.81	-0.41	-0.67
GTEP	0.28	-0.31	-0.42	0.12	1.00	0.23	-0.04	0.15	0.23	0.01	0.04
TIT	0.84	-0.13	-0.33	0.82	0.23	1.00	0.21	0.48	0.91	-0.41	-0.61
TAT	0.02	-0.05	0.01	0.00	-0.04	0.21	1.00	0.06	-0.06	-0.06	-0.11
TEY	0.03	-0.03	-0.07	0.18	0.15	0.48	0.06	1.00	0.41	-0.05	0.06
CDP	0.83	-0.13	-0.32	0.81	0.23	0.91	-0.06	0.41	1.00	-0.40	-0.59
CO	-0.45	0.14	-0.02	-0.41	0.01	-0.41	-0.06	-0.05	-0.40	1.00	0.55
NOX	-0.75	0.20	-0.13	-0.67	0.04	-0.61	-0.11	0.06	-0.59	0.55	1.00

Table 6: Pairwise Correlation Between Variables (High Energy Yield)

	AT	AP	AH	AFDP	GTEP	TIT	TAT	TEY	CDP	CO	NOX
AT	1.00	-0.52	0.20	0.24	-0.29	0.03	0.67	-0.89	-0.70	0.01	-0.42
AP	-0.52	1.00	-0.22	0.23	0.11	0.07	-0.55	0.55	0.59	-0.18	-0.22
AH	0.20	-0.22	1.00	0.30	-0.21	-0.01	0.23	-0.24	-0.24	-0.03	-0.34
AFDP	0.24	0.23	0.30	1.00	-0.08	0.00	-0.17	-0.05	0.15	-0.40	-0.82
GTEP	-0.29	0.11	-0.21	-0.08	1.00	-0.02	-0.59	0.51	0.56	-0.03	0.26
TIT	0.03	0.07	-0.01	0.00	-0.02	1.00	-0.04	0.06	0.09	0.06	-0.04
TAT	0.67	-0.55	0.23	-0.17	-0.59	-0.04	1.00	-0.92	-0.99	0.29	-0.06
TEY	-0.89	0.55	-0.24	-0.05	0.51	0.06	-0.92	1.00	0.94	-0.15	0.27
CDP	-0.70	0.59	-0.24	0.15	0.56	0.09	-0.99	0.94	1.00	-0.26	0.06
CO	0.01	-0.18	-0.03	-0.40	-0.03	0.06	0.29	-0.15	-0.26	1.00	0.39
NOX	-0.42	-0.22	-0.34	-0.82	0.26	-0.04	-0.06	0.27	0.06	0.39	1.00

Remove variables that are highly correlated.

```
##      AT       AP       AH      AFDP      GTEP       TAT
## 3.866424 1.600597 1.718769 7.412520 5.909197 2.301015

##      AP       AH      AFDP      GTEP       TAT       TEY       CDP
## 1.175646 1.434192 3.568001 1.400706 1.032173 1.383516 4.562918

##      AT       AP       AH      AFDP      GTEP       TIT       TAT
## 3.084771 1.948954 1.321587 1.996187 1.897431 1.027967 4.191605
```

Exploratory analysis shows possible linear relationships between the response variable CO and the feature variables CDP, TEY, TIT, GTEP and AFDP. Collinearity between some of the feature variables (TIT, CDP, and TEY) could cause some problems in our analysis and will likely lead to the removal of the redundant variables.

Methods

Linear Regression

We will create a multiple linear regression model using all feature variables mentioned in the description of Section 1. The implementation and parameters of this model can be obtained by the following equation where we will find estimates for the parameters β using:

$$\hat{\beta} = (X^T X)^{-1} X^T y$$

Key assumptions are stated as:

- Linearity: can be written as a linear combination of the predictors.
- Independence: the errors are independent of each other (not highly correlated).
- Normality: the distribution of the errors follow a normal distribution.
- Equal Variance: the error variance is the same.¹

We will then use model selection using backward BIC to tune our model and remove any insignificant predictor variables. This selection prefers smaller models which aligns with our goal of limiting the size of our final model.

```
full_model = lm(CO ~ ., data = gt_2015)
linear_model = lm(CO ~ . - NOX - TIT - CDP - TEY, data = gt_2015)
summary(linear_model)

##
## Call:
## lm(formula = CO ~ . - NOX - TIT - CDP - TEY, data = gt_2015)
##
## Residuals:
##     Min      1Q Median      3Q     Max 
## -6.839 -0.673 -0.132  0.481 34.242 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 145.099775   4.408695 32.912 < 2e-16 ***
## AT           0.028276   0.004060  6.965 3.57e-12 ***
## AP           0.001918   0.003067  0.625   0.532    
## AH           -0.009753   0.001618 -6.026 1.76e-09 ***
## AFDP        -2.531044   0.074576 -33.939 < 2e-16 ***
## GTEP        -0.186308   0.009082 -20.513 < 2e-16 ***
## TAT          -0.237369   0.004619 -51.387 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.436 on 7377 degrees of freedom
## Multiple R-squared:  0.5874, Adjusted R-squared:  0.587 
## F-statistic: 1750 on 6 and 7377 DF,  p-value: < 2.2e-16

#picking a new variable to test
AT_model = lm(CO ~ AT, data = gt_2015)
AP_model = lm(CO ~ AP, data = gt_2015)
AH_model = lm(CO ~ AH, data = gt_2015)
AFDP_model = lm(CO ~ AFDP, data = gt_2015)
GTEP_model = lm(CO ~ GTEP, data = gt_2015)
TAT_model = lm(CO ~ TAT, data = gt_2015)
BIC(AT_model)
```

¹Dalpiaz David, Applied Statistics in R, <https://daviddalpiaz.github.io/appliedstats/model-diagnostics.html>

```

## [1] 31634.69

BIC(AP_model)

## [1] 32553.05

BIC(AH_model)

## [1] 32668.32

BIC(AFDP_model) #second best

## [1] 28953.71

BIC(GTEP_model)

## [1] 30112.68

BIC(TAT_model)

## [1] 32852.49

BIC(linear_model) #this is the best model

## [1] 26365.45

library(MASS)

n = length(resid(linear_model))
BIC_model = step(linear_model, direction = "backward", k = log(n))

## Start: AIC=5401.66
## CO ~ (AT + AP + AH + AFDP + GTEP + TIT + TAT + TEY + CDP + NOX) -
##      NOX - TIT - CDP - TEY
##
##          Df Sum of Sq    RSS    AIC
## - AP     1      0.8 15218 5393.1
## <none>            15217 5401.7
## - AH     1     74.9 15292 5429.0
## - AT     1    100.1 15317 5441.1
## - GTEP   1    868.0 16085 5802.4
## - AFDP   1   2376.0 17593 6464.1
## - TAT    1   5447.0 20664 7652.1
##
## Step: AIC=5393.14
## CO ~ AT + AH + AFDP + GTEP + TAT
##
##          Df Sum of Sq    RSS    AIC
## <none>            15218 5393.1

```

```

## - AH     1      86.8 15304 5426.2
## - AT     1     120.5 15338 5442.5
## - GTEP   1     991.0 16209 5850.1
## - AFDP   1    2564.3 17782 6534.1
## - TAT    1    5608.5 20826 7701.0

coef(BIC_model)

## (Intercept)          AT          AH          AFDP          GTEP          TAT
## 147.33755305  0.02702808 -0.01004686 -2.51758434 -0.18816259 -0.23782668

stepAIC(linear_model, direction = "backward")

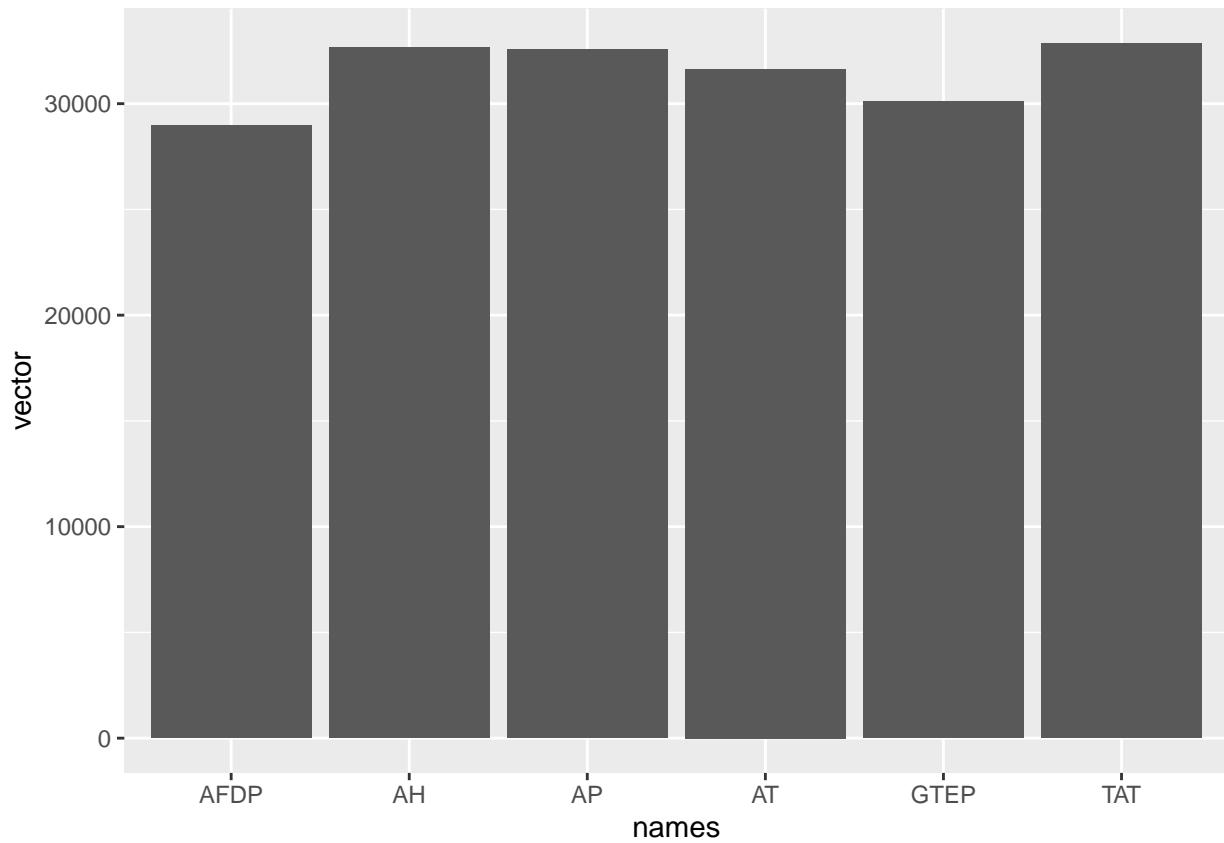
## Start: AIC=5353.31
## CO ~ (AT + AP + AH + AFDP + GTEP + TIT + TAT + TEY + CDP + NOX) -
##       NOX - TIT - CDP - TEY
##
##           Df Sum of Sq   RSS   AIC
## - AP     1      0.8 15218 5351.7
## <none>            15217 5353.3
## - AH     1     74.9 15292 5387.6
## - AT     1    100.1 15317 5399.7
## - GTEP   1     868.0 16085 5760.9
## - AFDP   1    2376.0 17593 6422.6
## - TAT    1    5447.0 20664 7610.7
##
## Step: AIC=5351.7
## CO ~ AT + AH + AFDP + GTEP + TAT
##
##           Df Sum of Sq   RSS   AIC
## <none>            15218 5351.7
## - AH     1      86.8 15304 5391.7
## - AT     1     120.5 15338 5408.0
## - GTEP   1     991.0 16209 5815.5
## - AFDP   1    2564.3 17782 6499.6
## - TAT    1    5608.5 20826 7666.5

##
## Call:
## lm(formula = CO ~ AT + AH + AFDP + GTEP + TAT, data = gt_2015)
##
## Coefficients:
## (Intercept)          AT          AH          AFDP          GTEP          TAT
## 147.33755      0.02703     -0.01005     -2.51758     -0.18816     -0.23783

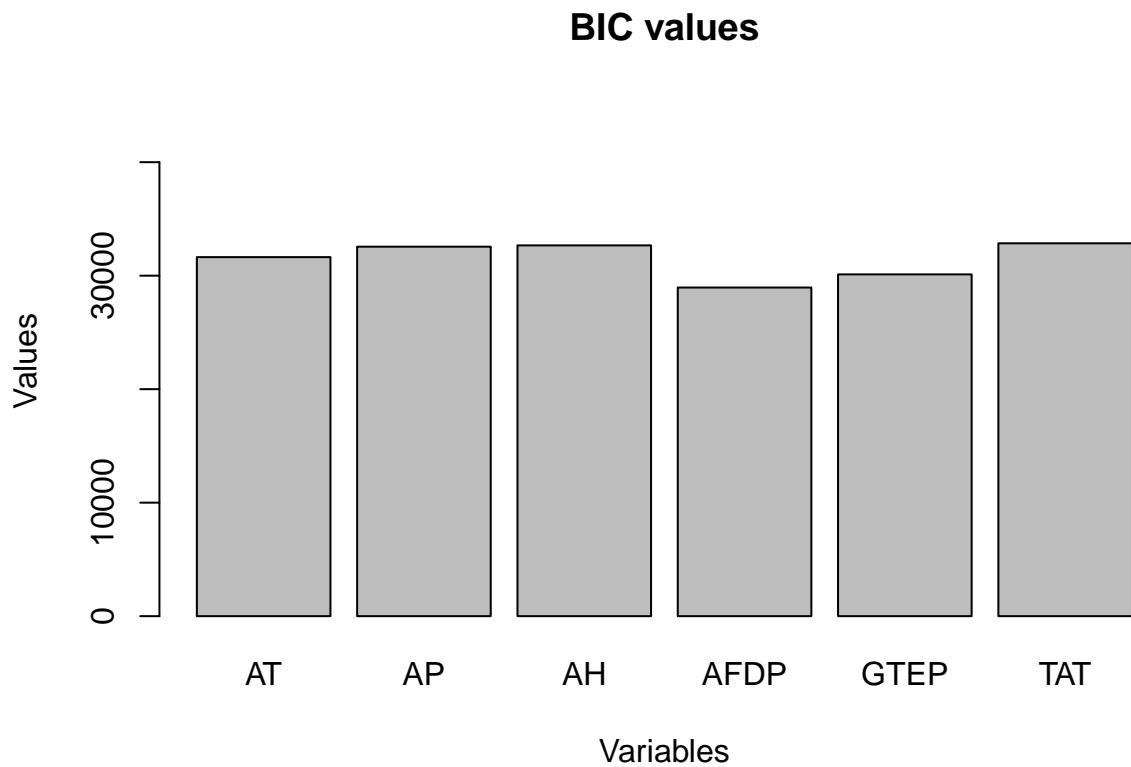
vector <- c(BIC(AT_model), BIC(AP_model), BIC(AH_model), BIC(AFDP_model), BIC(GTEP_model), BIC(TAT_model))

library(ggplot2)
df <- data.frame(vector = c(BIC(AT_model), BIC(AP_model), BIC(AH_model), BIC(AFDP_model), BIC(GTEP_model), BIC(TAT_model)))
ggplot(data = df, aes(x = names, y = vector), ylim = c(0, 50000)) + geom_bar(stat = "identity")

```



```
barplot(vector, main = "BIC values", xlab = "Variables", ylab = "Values", names.arg = c("AT", "AP", "AH"))
```



```
## Linear and Lasso stepwise AIC Models

library(caret)

## Loading required package: lattice

##
## Attaching package: 'lattice'

## The following object is masked from 'package:faraway':
## melanoma

#5-fold cross validation
cv_5 <- trainControl(method = "cv", number = 5)

#All Data

set.seed(10)
#AIC stepwise selected linear model
all_linear_mod <- train(
  form = CO ~ . - NOX - TIT - CDP - TEY ,
  data = gt_2015,
  method = "lmStepAIC",
  trControl = cv_5,
```

```

        trace = FALSE
    )

#Linear log model
all_log_linear_mod <- train(
    form = log(CO) ~ . - NOX - TIT - CDP - TEY ,
    data = gt_2015,
    method = "lmStepAIC",
    trControl = cv_5,
    nvmax = 10,
    trace = FALSE
)

#Lasso model
all_lasso_mod <- train(
    form = CO ~ . - NOX - TIT - CDP - TEY ,
    data = gt_2015,
    method = "lasso",
    trControl = cv_5
)

```

#Typical Energy Yield (130-136)

```

set.seed(10)
#AIC stepwise selected linear model
typical_linear_mod <- train(
    form = CO ~ . - NOX - TIT - AT,
    data = gt_2015_typical,
    method = "lmStepAIC",
    trControl = cv_5,
    trace = FALSE
)

#Linear log model
typical_log_linear_mod <- train(
    form = log(CO) ~ . - NOX - TIT - AT,
    data = gt_2015_typical,
    method = "lmStepAIC",
    trControl = cv_5,
    nvmax = 10,
    trace = FALSE
)

#Lasso model
typical_lasso_mod <- train(
    form = CO ~ . - NOX - TIT - AT,
    data = gt_2015_typical,
    method = "lasso",
    trControl = cv_5
)

```

#High Energy Yield (160+)

```

set.seed(10)

#AIC stepwise selected linear model
high_linear_mod <- train(
  form = CO ~ . - NOX - TEY - CDP,
  data = gt_2015_high,
  method = "lmStepAIC",
  trControl = cv_5,
  trace = FALSE
)

#Linear log model
high_log_linear_mod <- train(
  form = log(CO) ~ . - NOX - TEY - CDP,
  data = gt_2015_high,
  method = "lmStepAIC",
  trControl = cv_5,
  nvmax = 10,
  trace = FALSE
)

#Lasso model
high_lasso_mod <- train(
  form = CO ~ . - NOX - TEY - CDP,
  data = gt_2015_high,
  method = "lasso",
  trControl = cv_5
)

#Results
all_linear_mod$results

##   parameter      RMSE   Rsquared       MAE     RMSESD RsquaredSD       MAESD
## 1      none 1.433269  0.5894359  0.8272737  0.1219639  0.03158297  0.01273652

all_log_linear_mod$results

##   parameter      RMSE   Rsquared       MAE     RMSESD RsquaredSD       MAESD
## 1      none 0.3302711  0.6592231  0.2230641  0.01248279  0.01762377  0.003218697

all_lasso_mod$results

##   fraction      RMSE   Rsquared       MAE     RMSESD RsquaredSD       MAESD
## 1      0.1 1.993589  0.4162352  1.1803507  0.1700911  0.06042455  0.03400057
## 2      0.5 1.548607  0.5508411  0.8209167  0.2118693  0.08098964  0.03102373
## 3      0.9 1.427676  0.5921746  0.8187187  0.2237106  0.08165401  0.02995461

typical_linear_mod$results

##   parameter      RMSE   Rsquared       MAE     RMSESD RsquaredSD       MAESD
## 1      none 0.9697531  0.3377307  0.4738538  0.5080101  0.1978678  0.03731379

```

```

typical_log_linear_mod$results

##   parameter      RMSE  Rsquared       MAE      RMSESD RsquaredSD       MAESD
## 1      none 0.2430984 0.4535081 0.1681481 0.02249379 0.06104676 0.007462328

typical_lasso_mod$results

##   fraction      RMSE  Rsquared       MAE      RMSESD RsquaredSD       MAESD
## 1      0.1 1.0797560 0.3035173 0.6724230 0.4957577 0.1996600 0.05480752
## 2      0.5 0.9771476 0.3304005 0.5080683 0.5452057 0.2080158 0.04598878
## 3      0.9 0.9557657 0.3494965 0.4763786 0.5467614 0.2066107 0.04128323

high_linear_mod$results

##   parameter      RMSE  Rsquared       MAE      RMSESD RsquaredSD       MAESD
## 1      none 0.442826 0.2248008 0.3183321 0.04574879 0.05330011 0.02670168

high_log_linear_mod$results

##   parameter      RMSE  Rsquared       MAE      RMSESD RsquaredSD       MAESD
## 1      none 0.2796676 0.1914723 0.1751923 0.04989463 0.101253 0.02364425

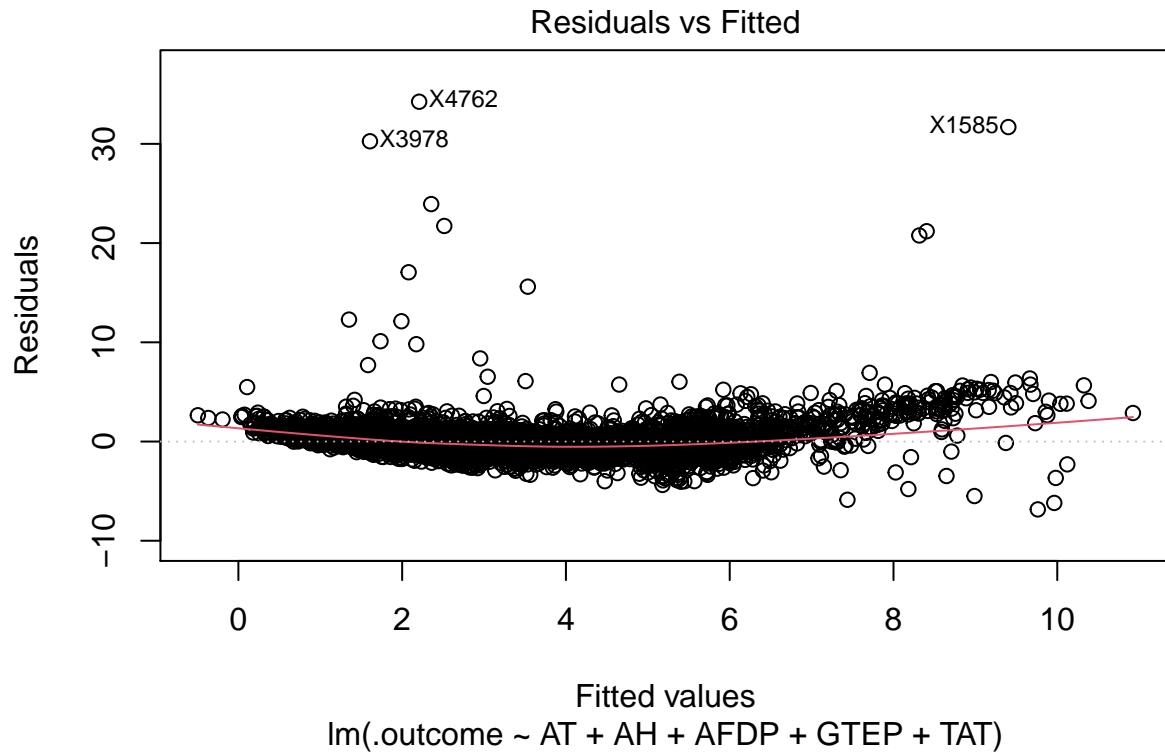
high_lasso_mod$results

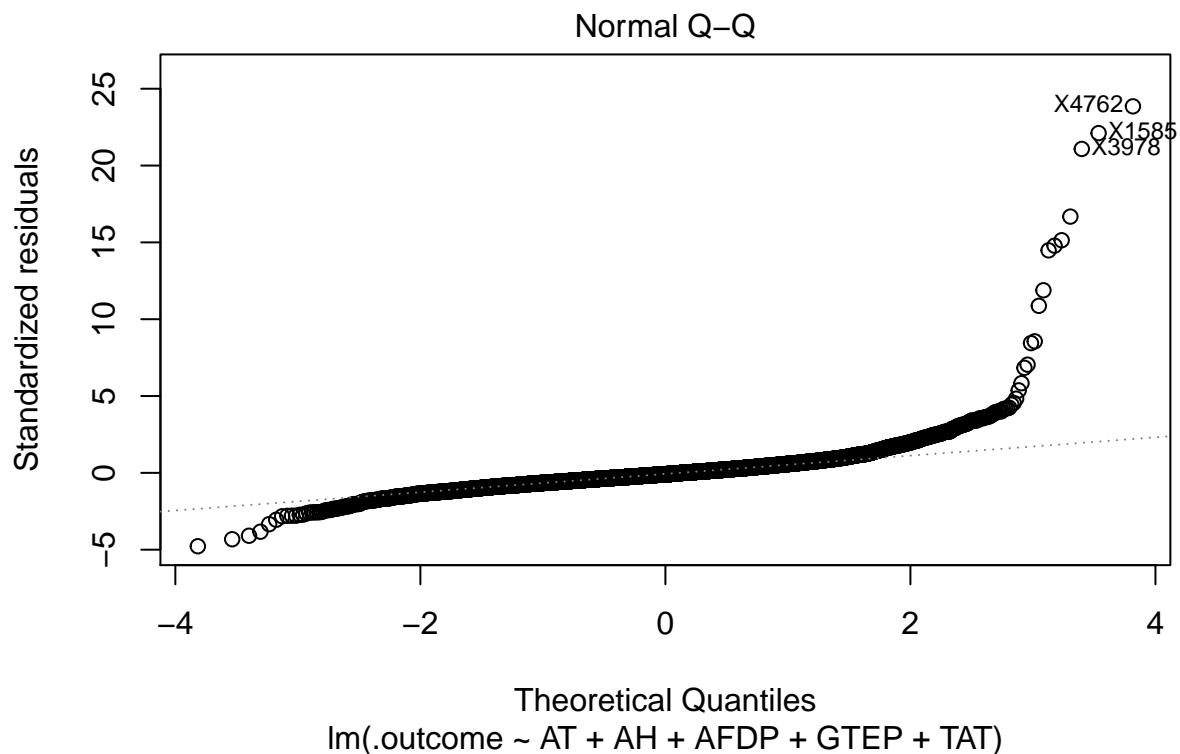
##   fraction      RMSE  Rsquared       MAE      RMSESD RsquaredSD       MAESD
## 1      0.1 0.4774531 0.1728564 0.3474956 0.03348756 0.06872672 0.01508288
## 2      0.5 0.4466801 0.2102170 0.3169699 0.04614911 0.08303450 0.02629183
## 3      0.9 0.4452885 0.2143768 0.3204783 0.04854922 0.07978382 0.02675016

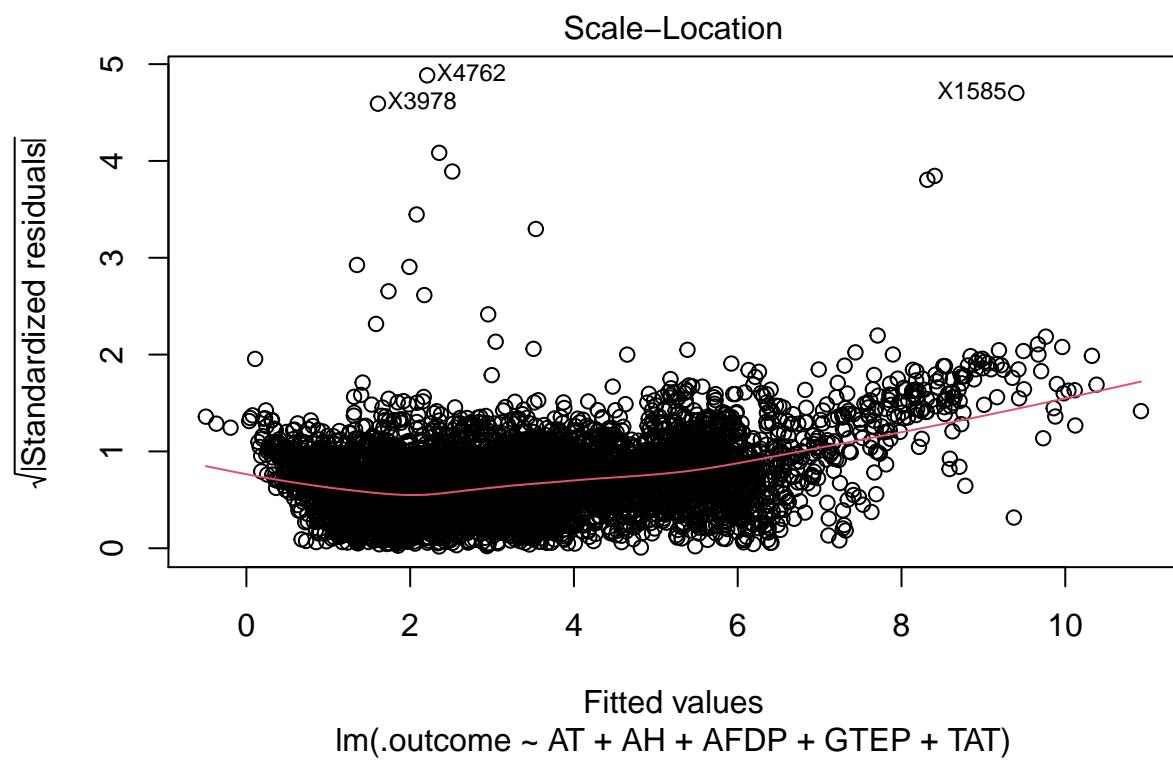
#plots

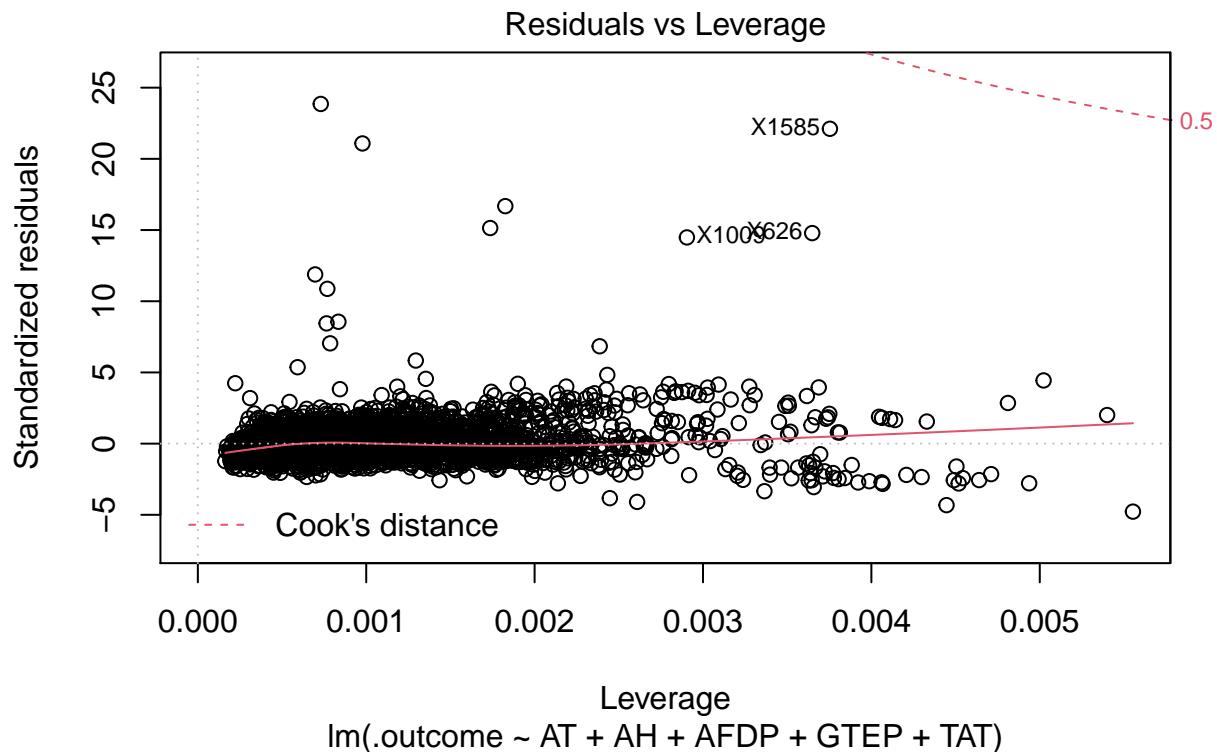
```

```
plot(all_linear_mod$finalModel)
```

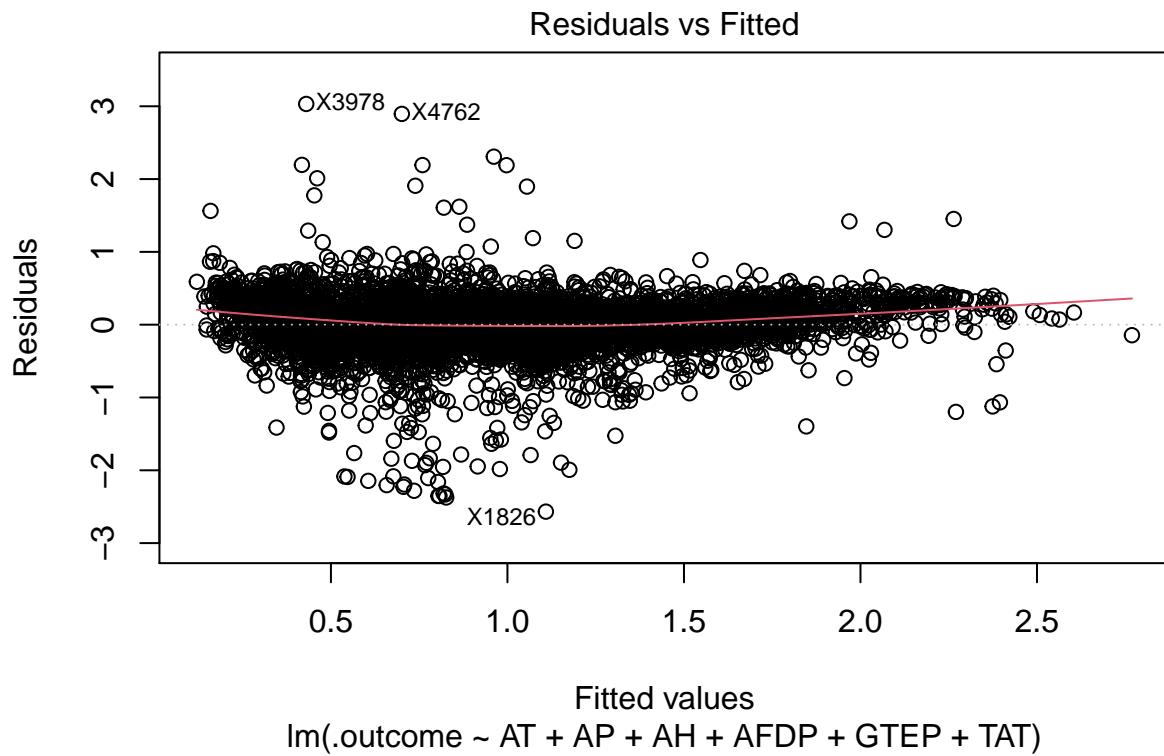


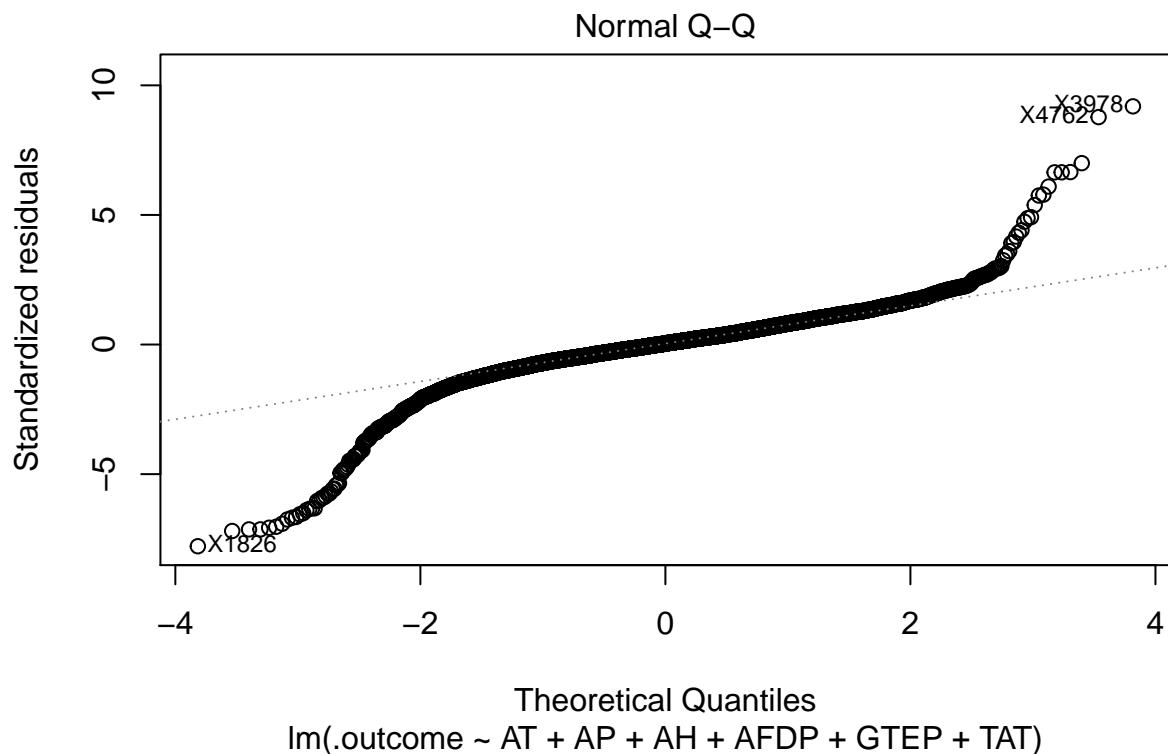


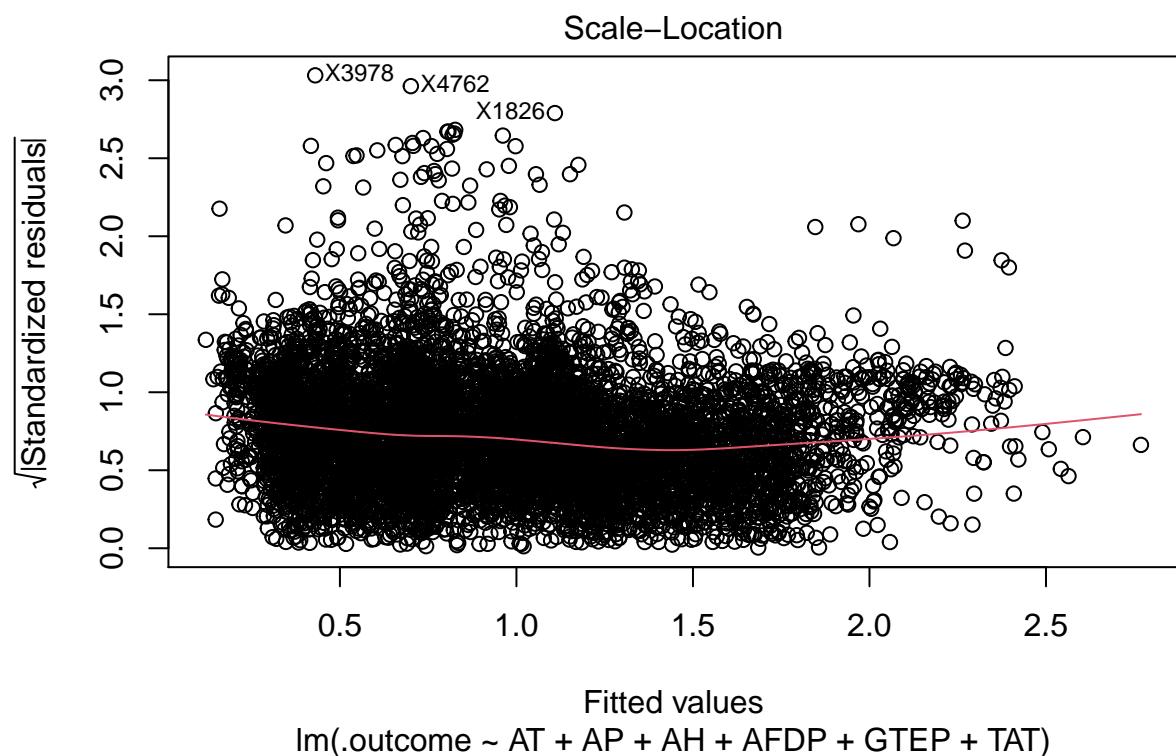


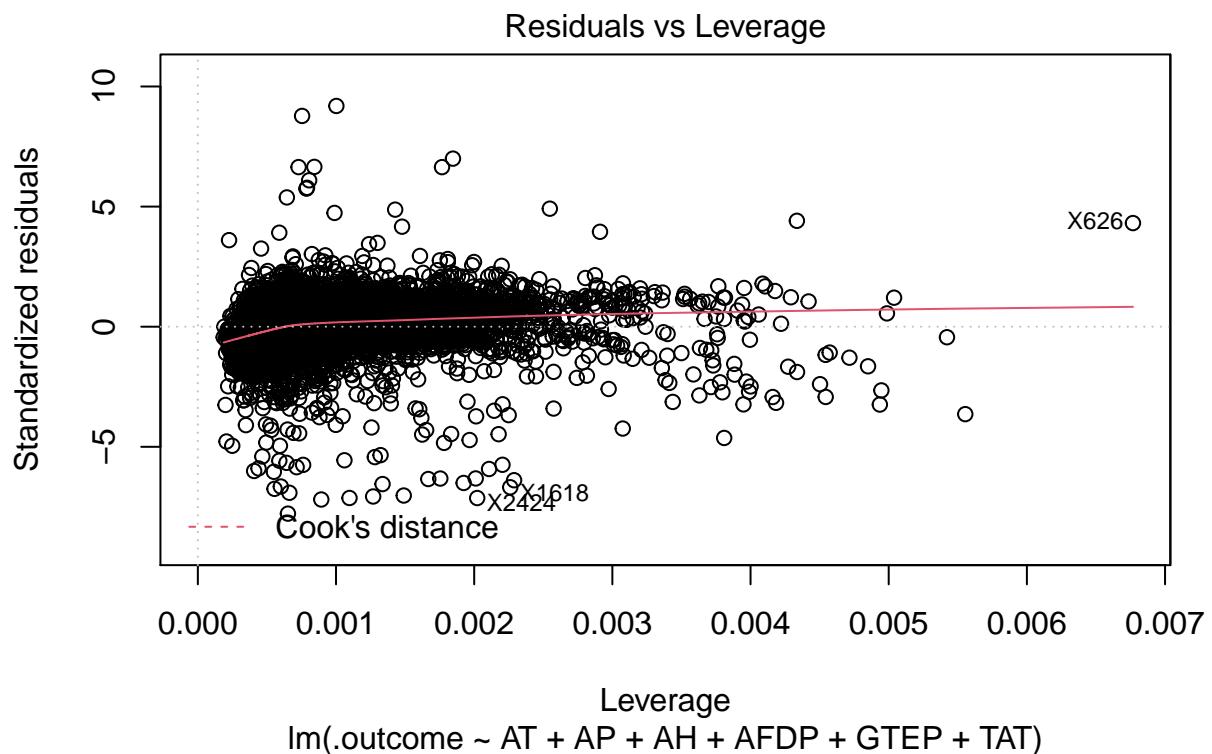


```
plot(all_log_linear_mod$finalModel)
```

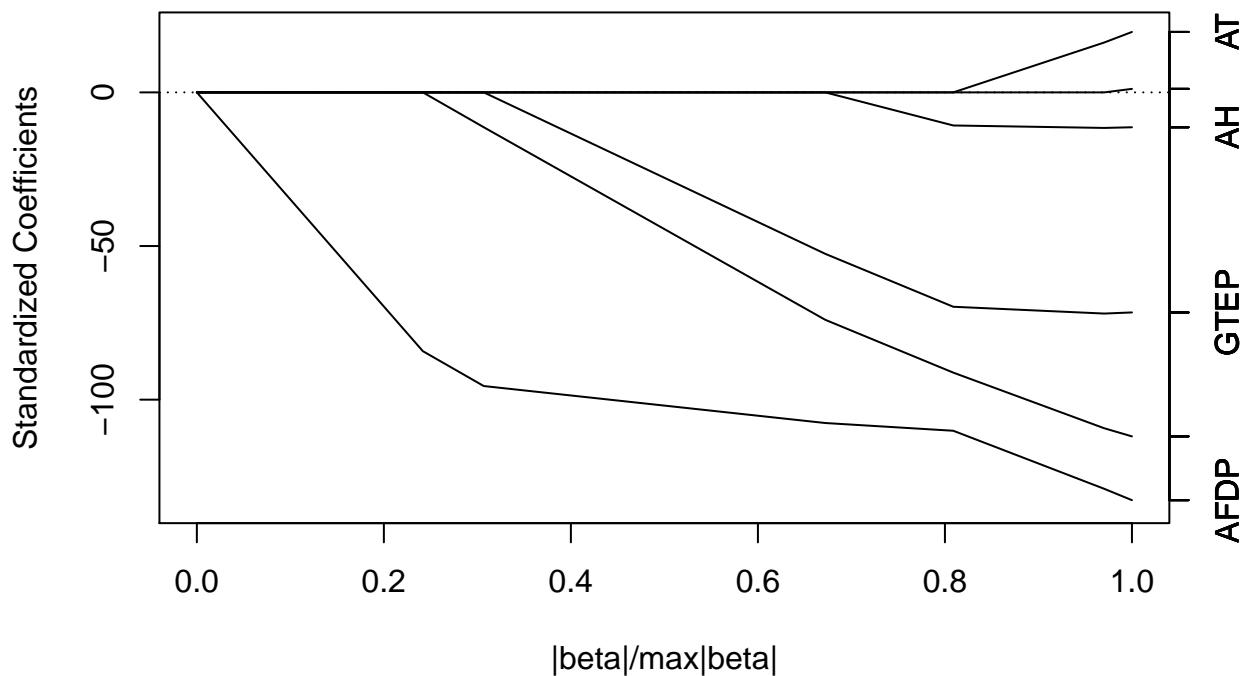




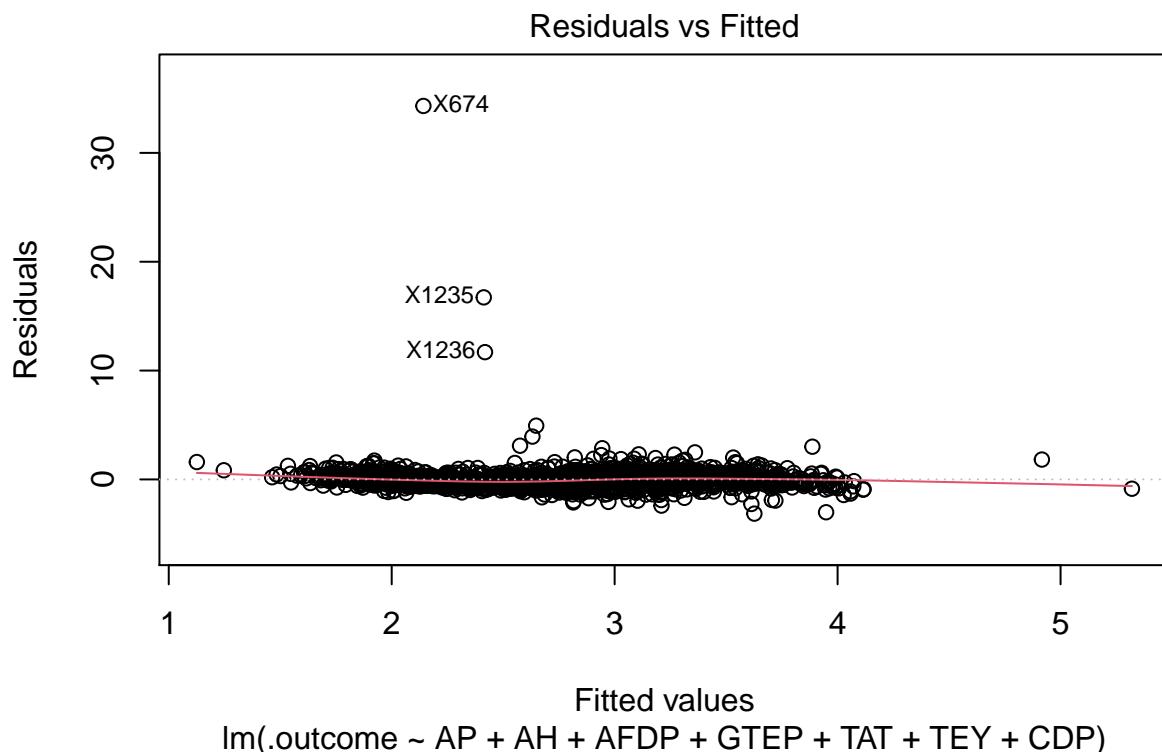


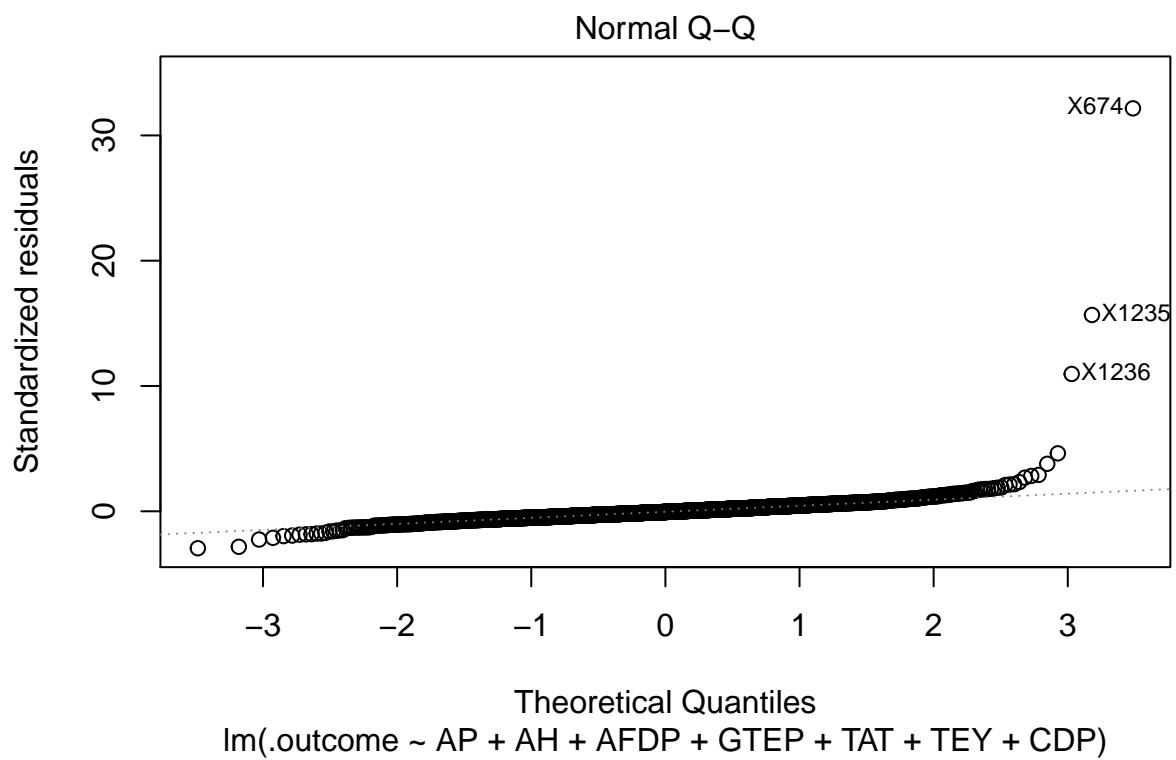


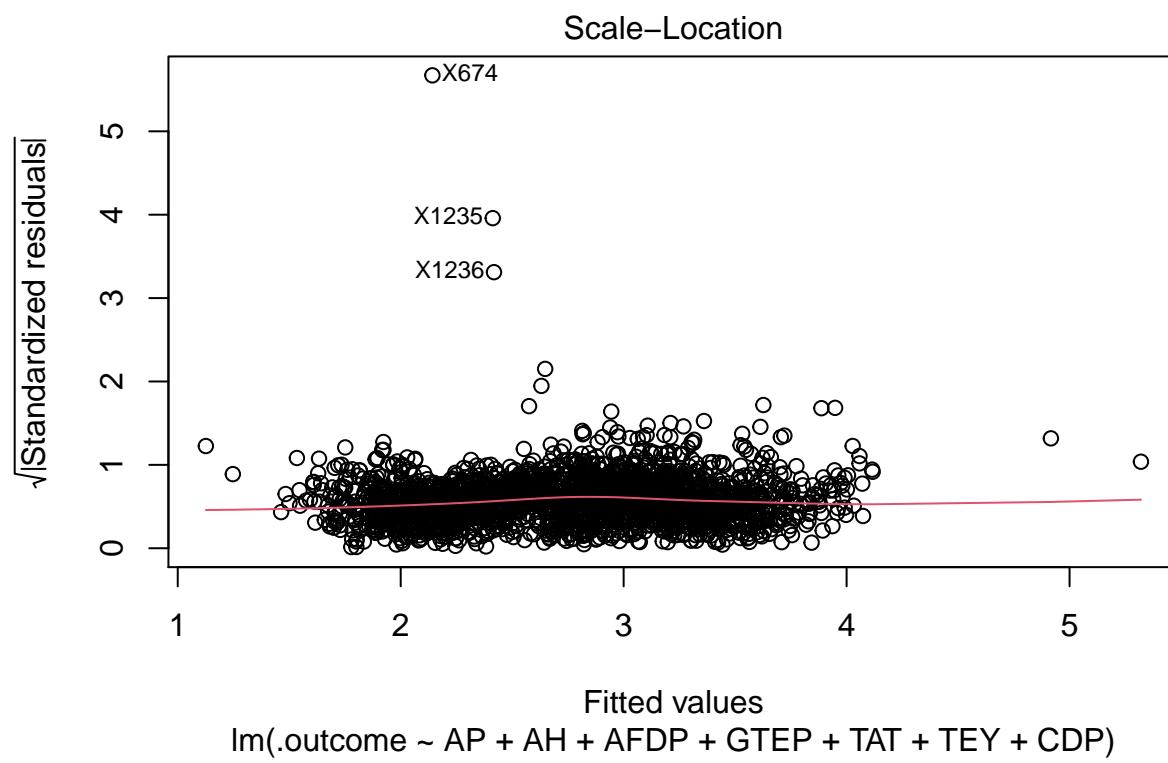
```
plot(all_lasso_mod$finalModel)
```

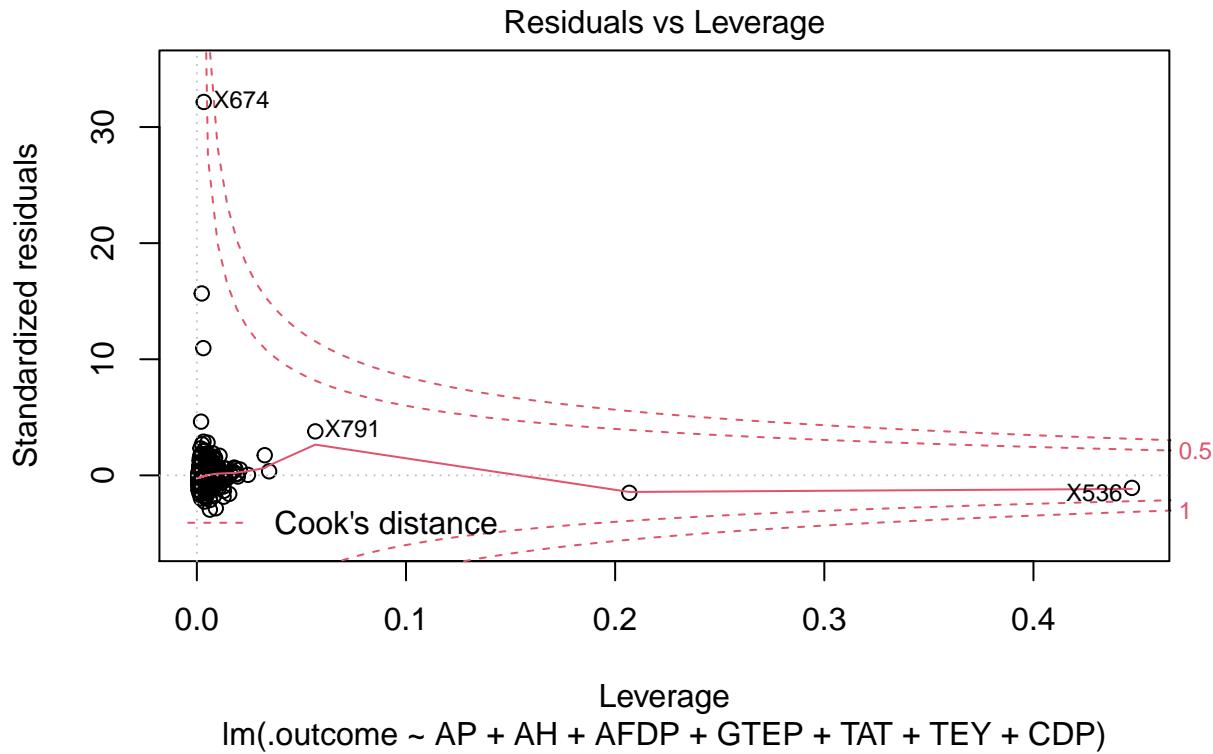


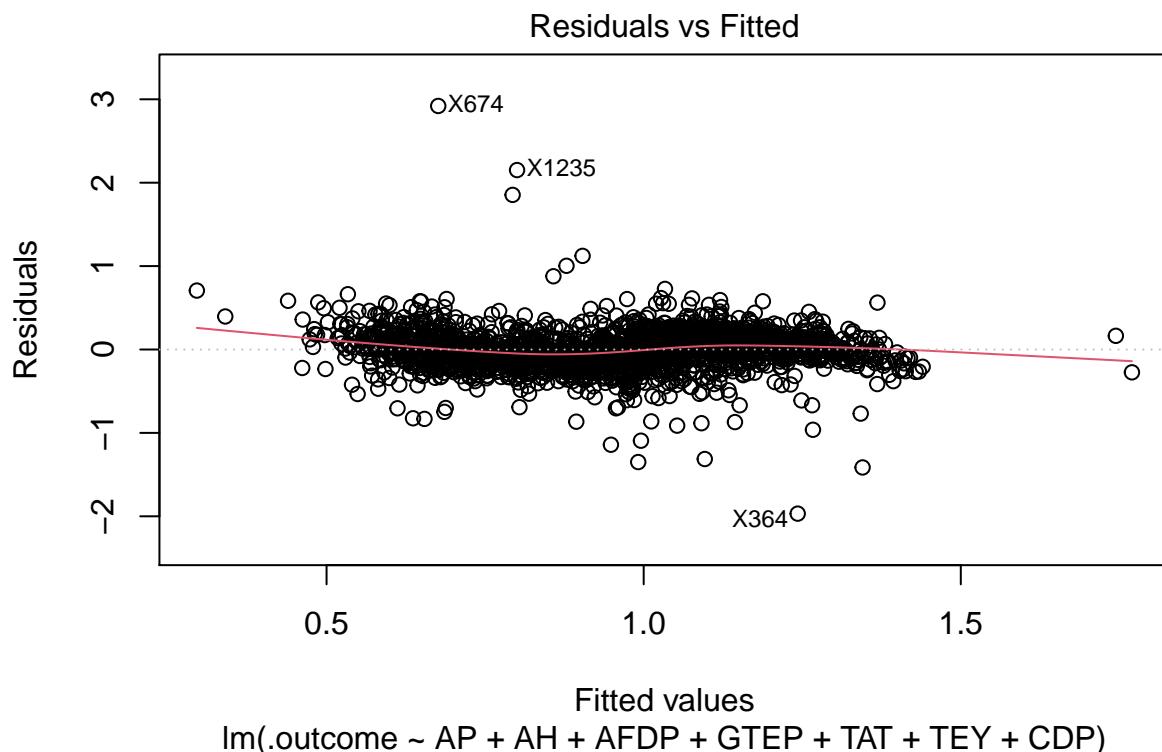
```
plot(typical_linear_mod$finalModel)
```

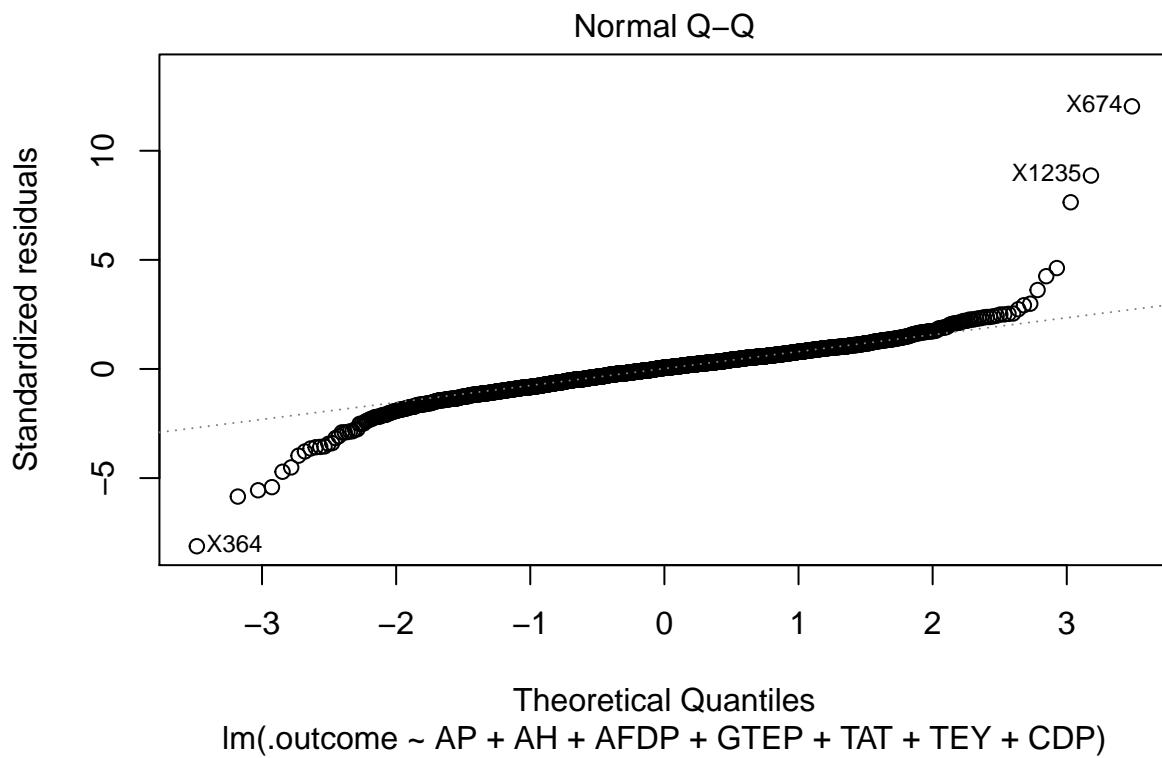


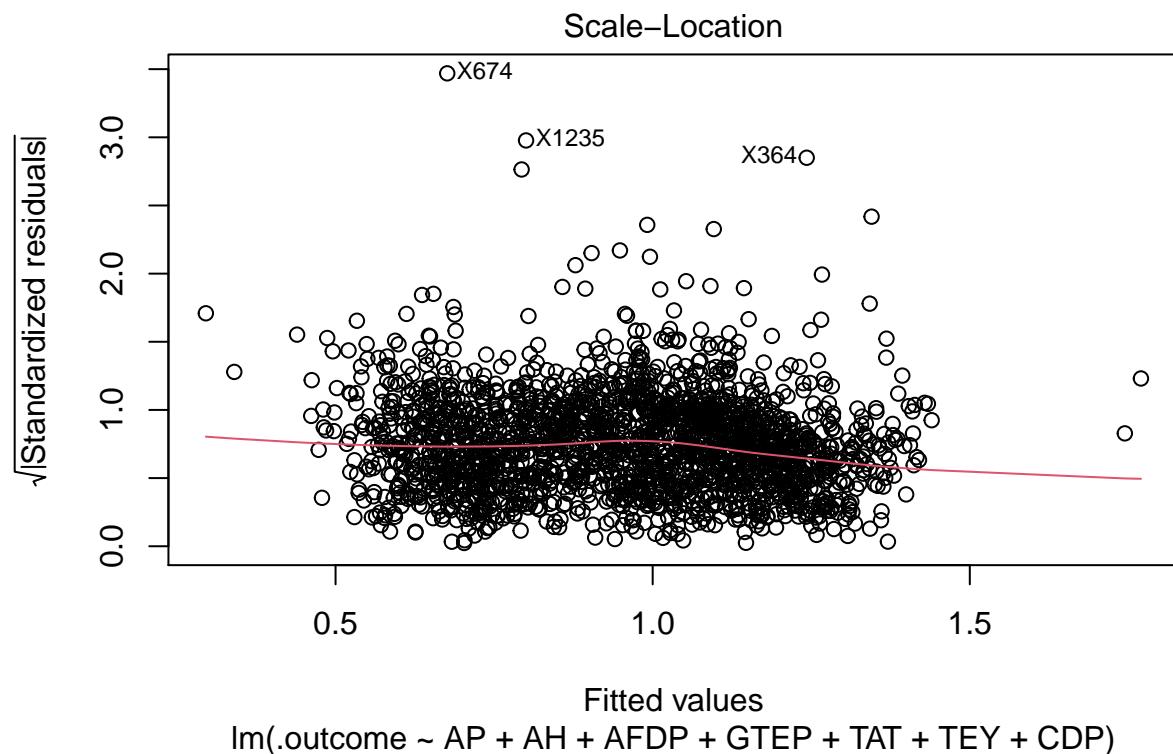


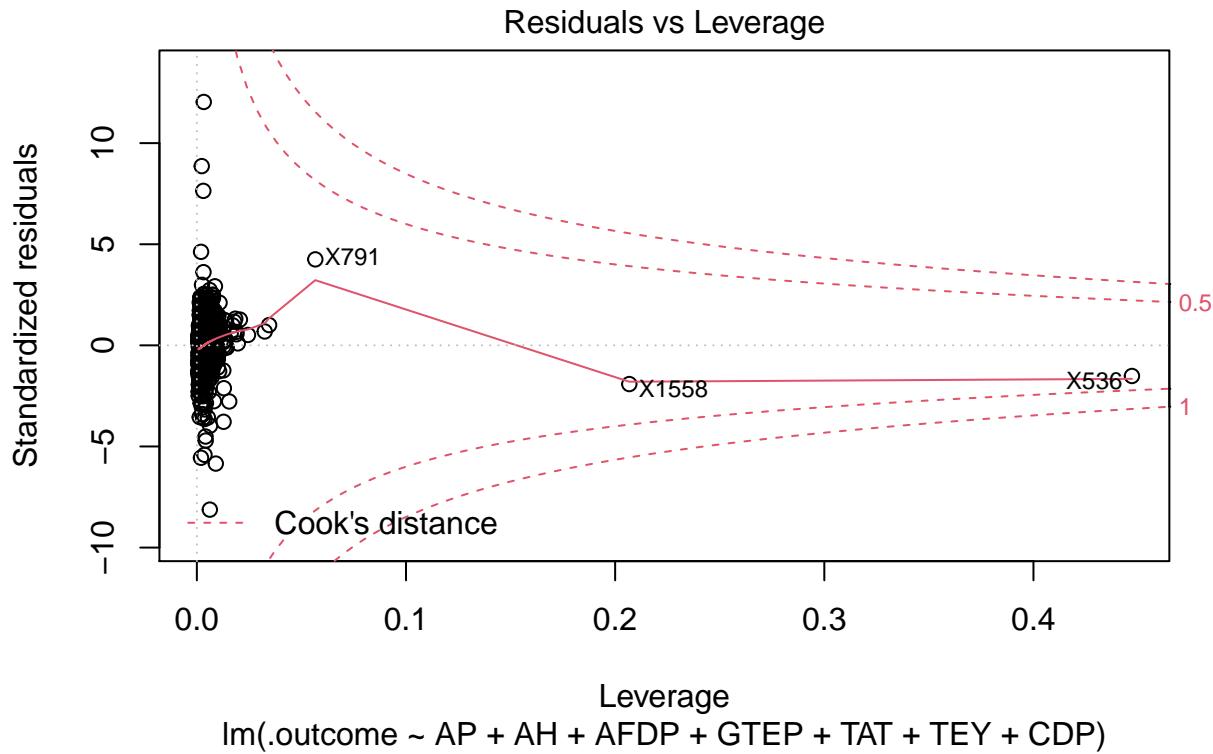


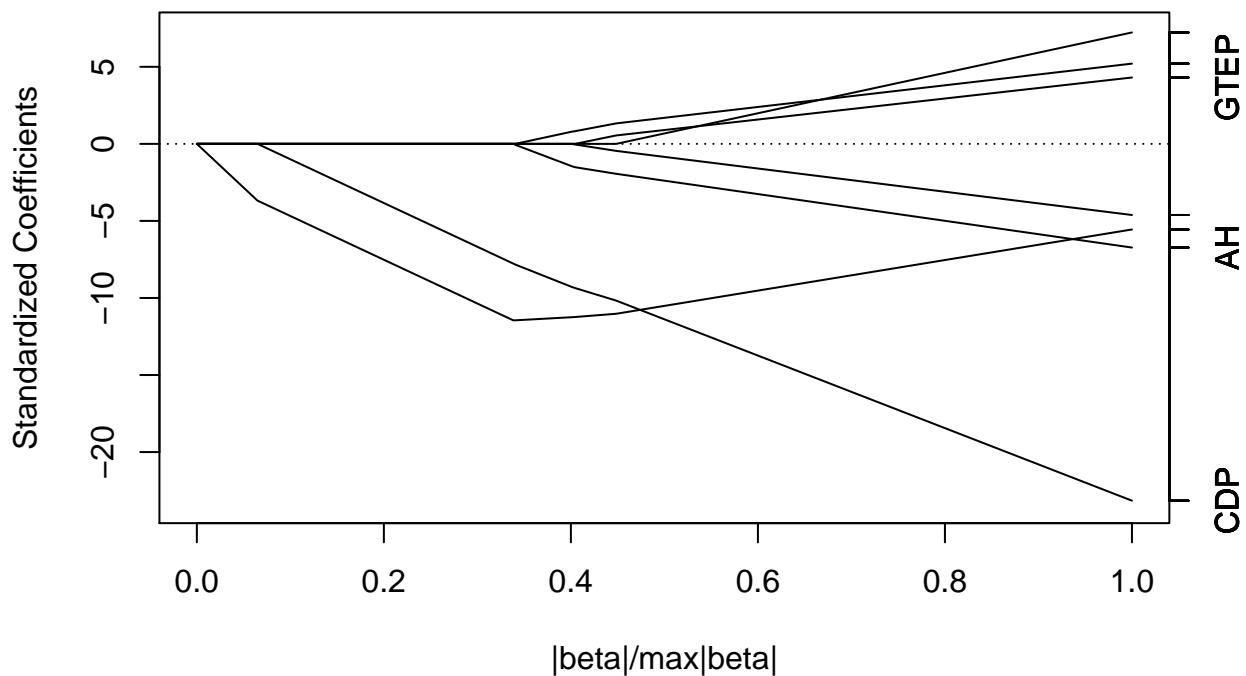




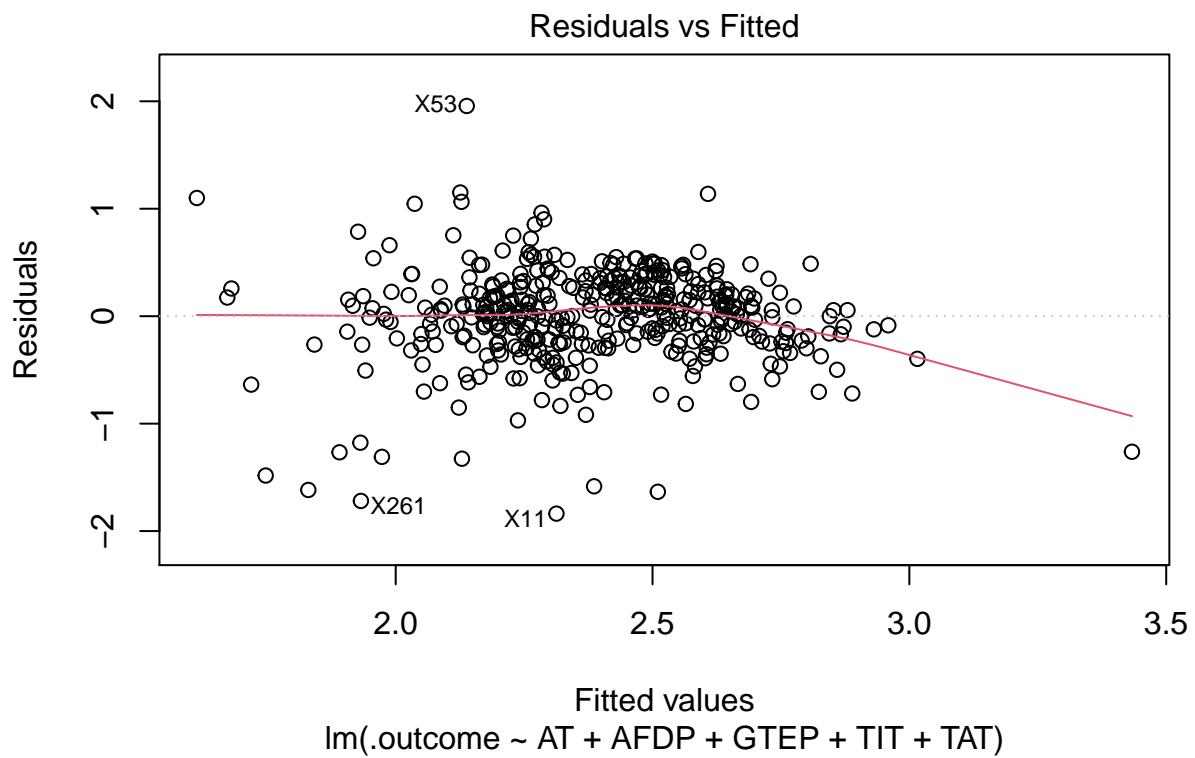


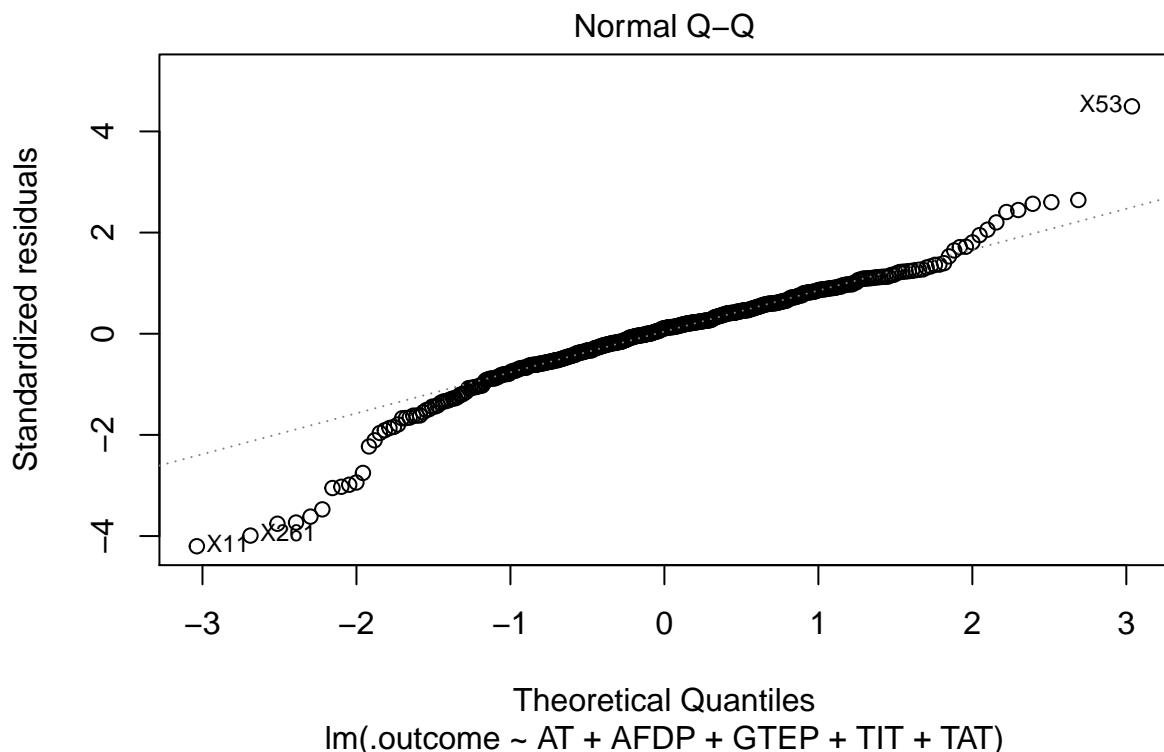


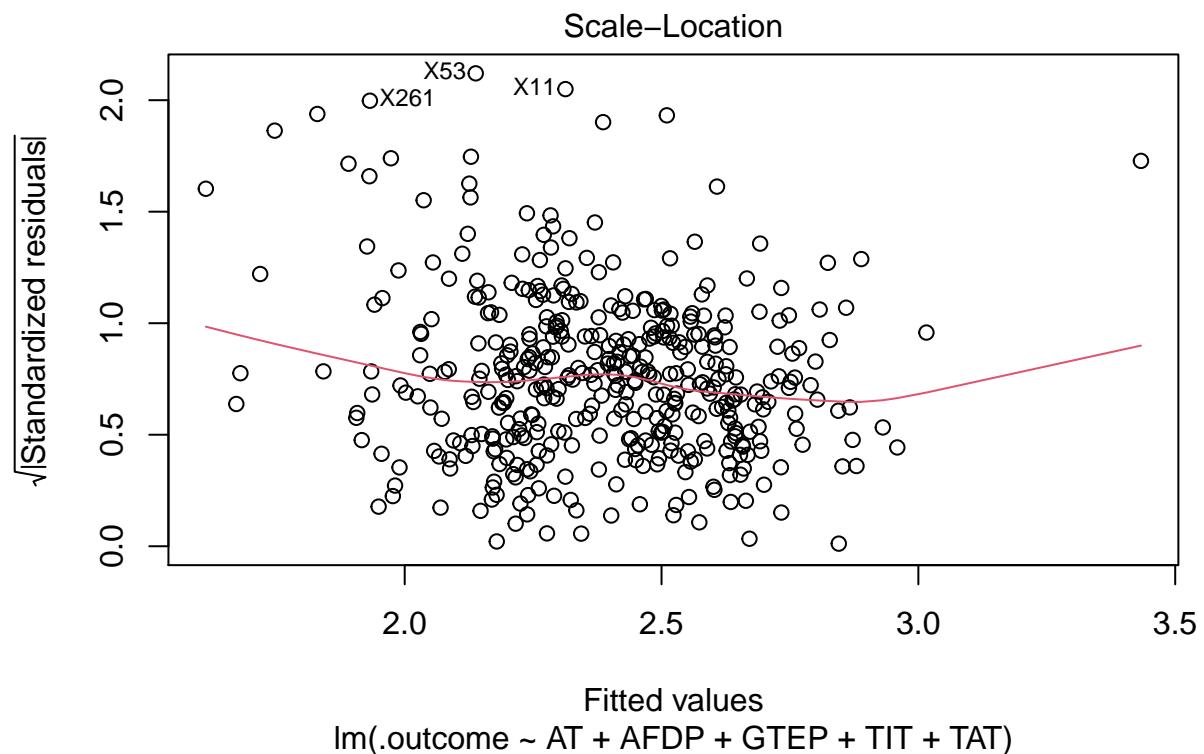


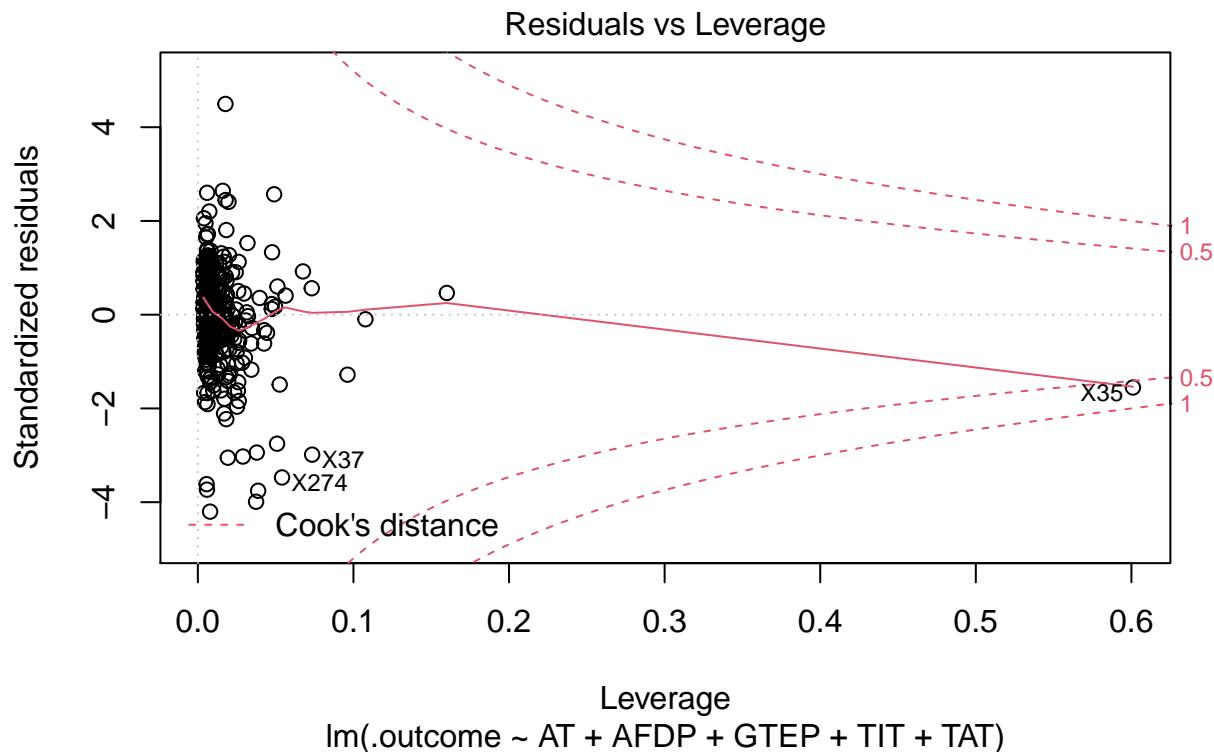


```
plot(high_linear_mod$finalModel)
```

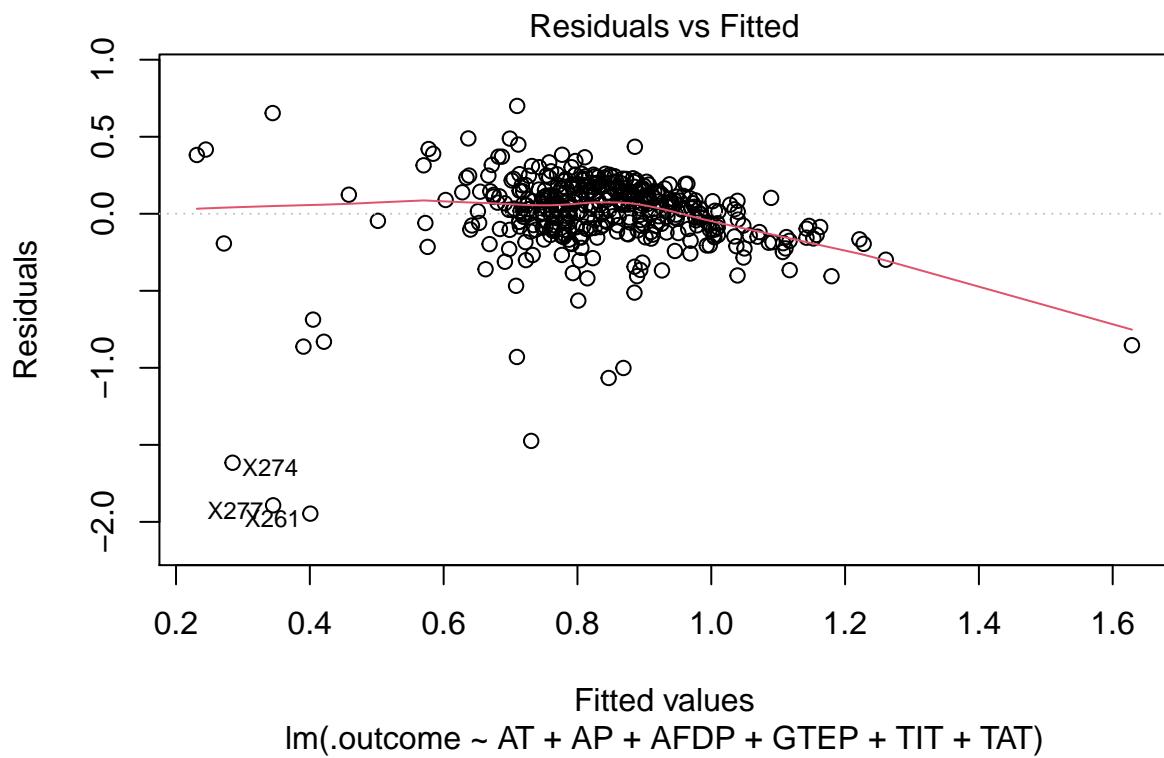


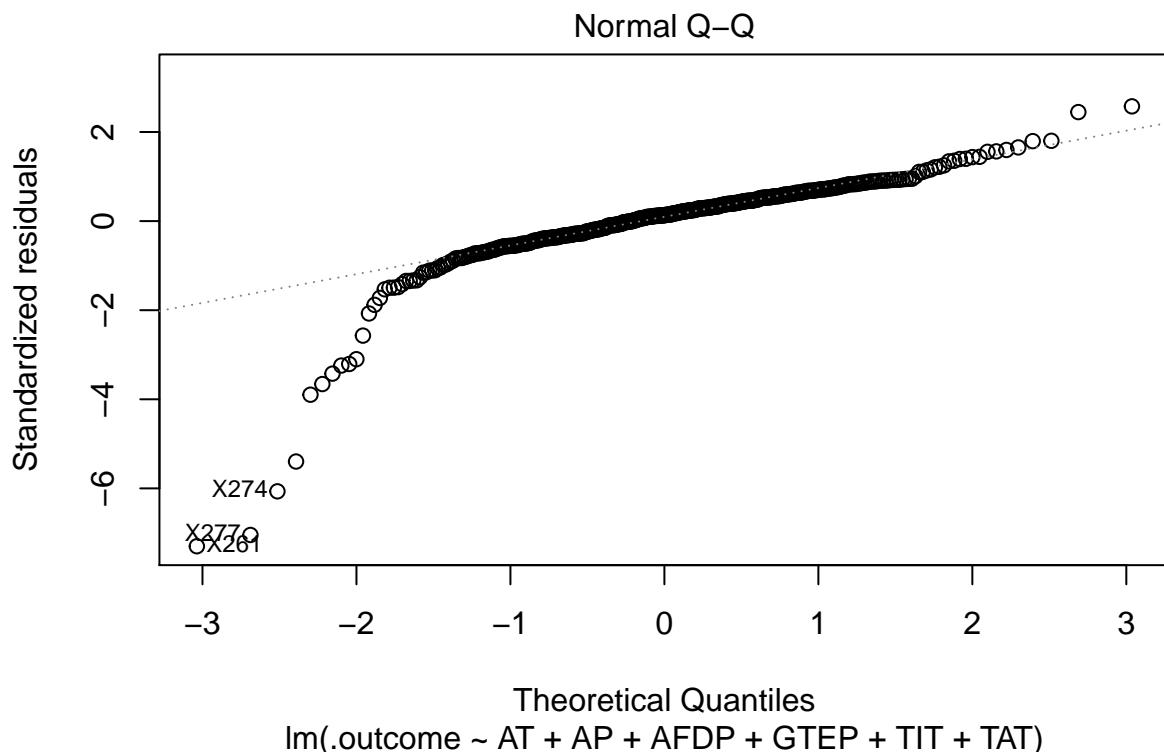


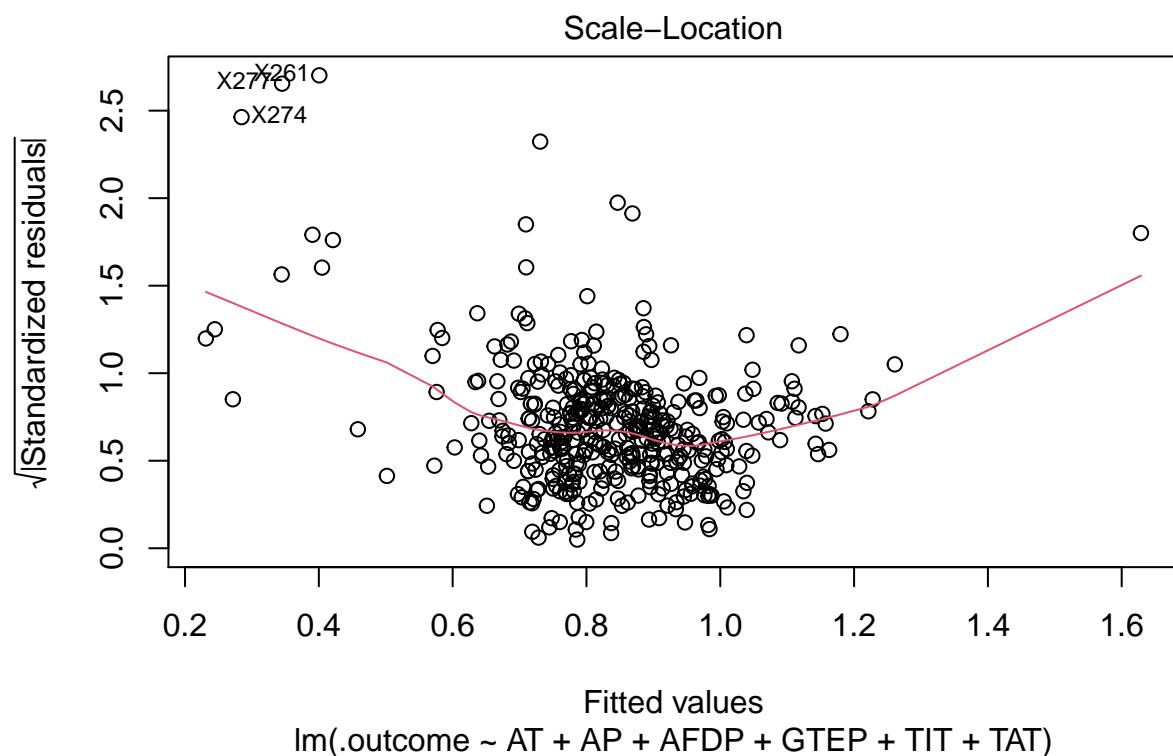


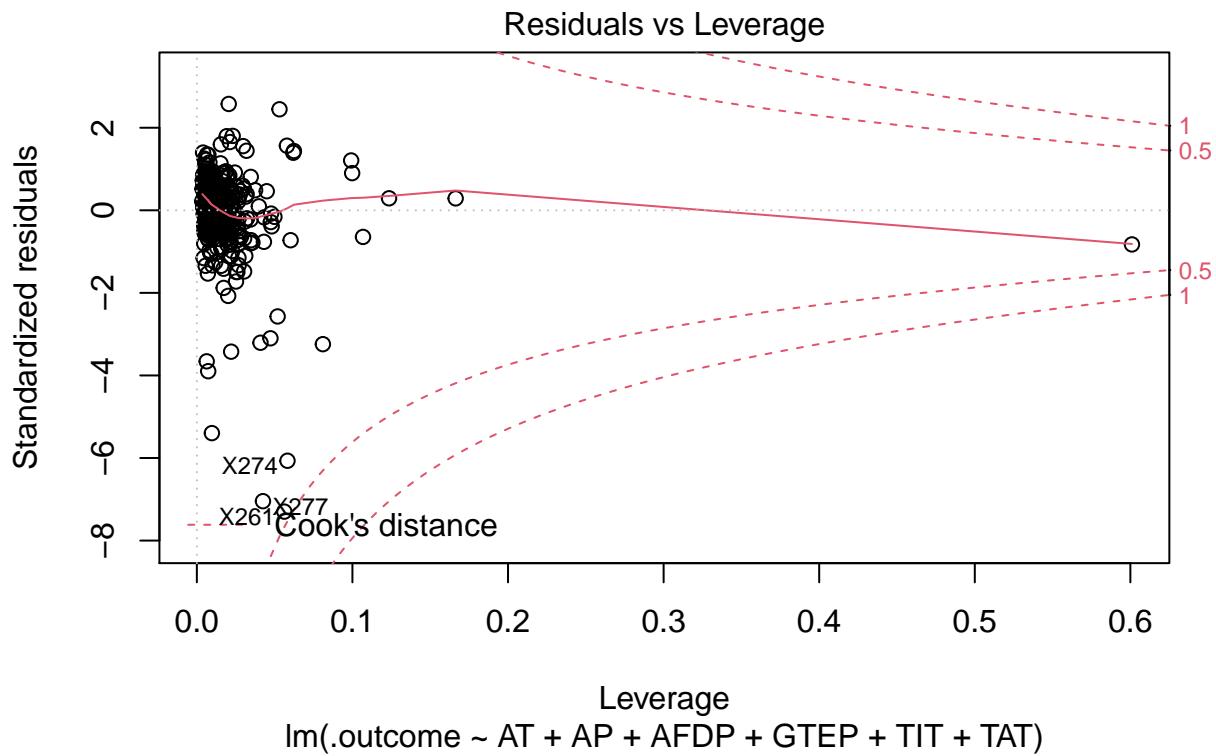


```
plot(high_log_linear_mod$finalModel)
```

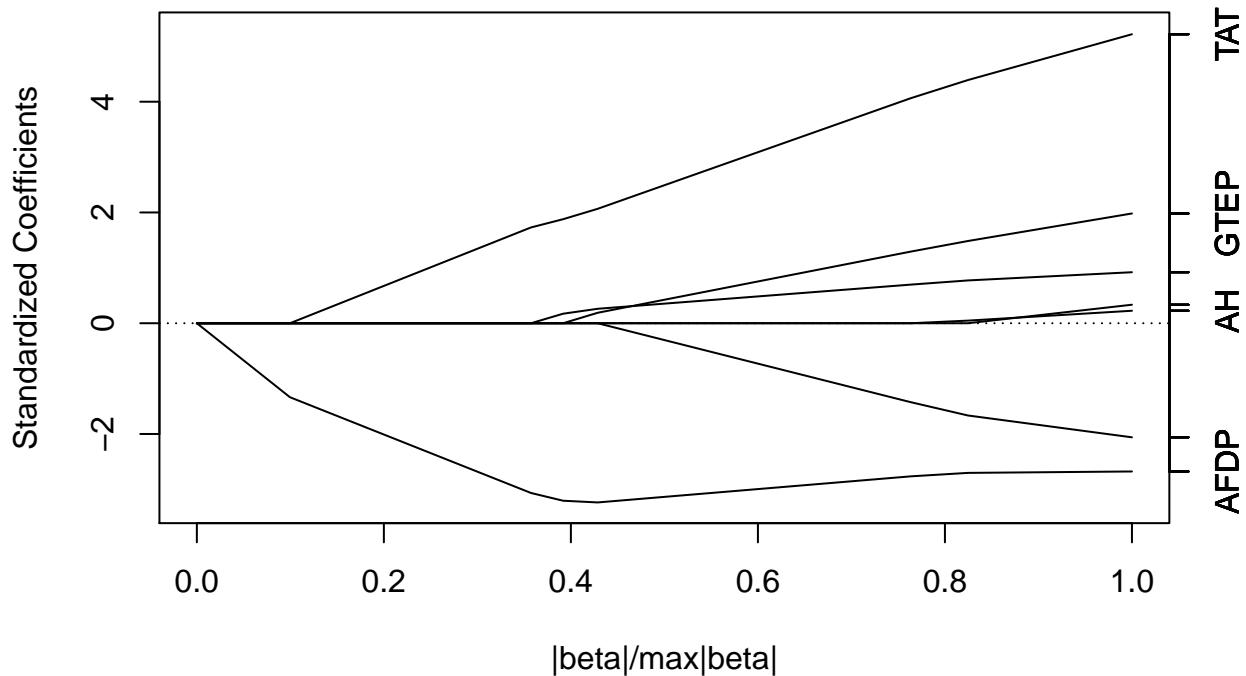








```
plot(high_lasso_mod$finalModel)
```



```
##Linear Model Diagnostic Plots
```

Decision Trees

```
#All Data
```

```
# install.packages('tree')
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.1 --

## v tibble 3.1.1     v dplyr  1.0.5
## v tidyverse 1.1.3    v stringr 1.4.0
## v purrr   0.3.4     v forcats 0.5.1

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x purrr::lift()   masks caret::lift()
## x dplyr::select() masks MASS::select()

library(tree)
```

```

## Registered S3 method overwritten by 'tree':
##   method      from
##   print.tree  cli

RMSE <- function(y, y_hat) {
  rmse <- sqrt(sum(((y_hat - y)^2)/length(y)))
  print(rmse)
}

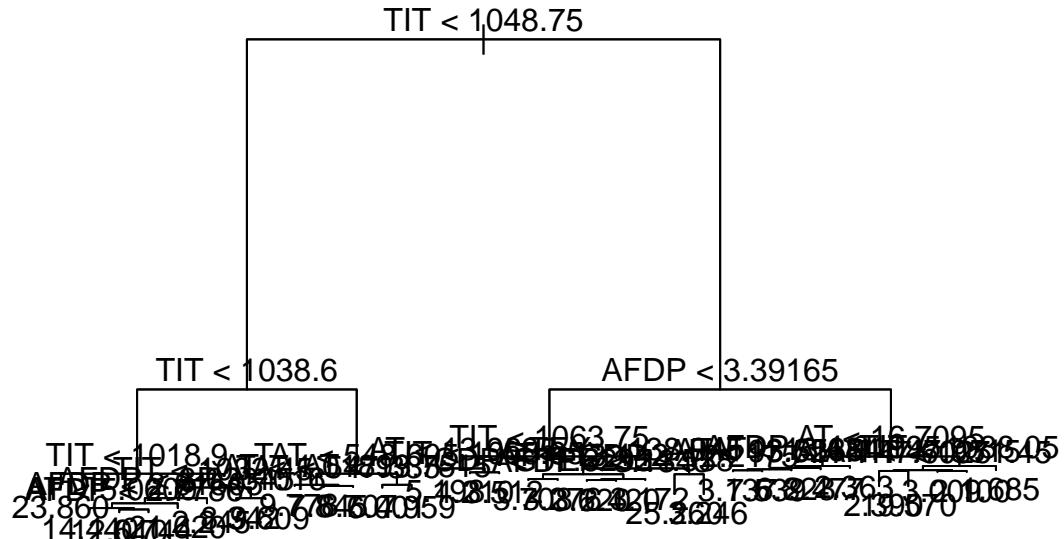
set.seed(10)
train <- gt_2015 %>% dplyr::select(-NOX) %>% sample_frac(0.8)
test <- gt_2015 %>% dplyr::select(-NOX) %>% setdiff(train)

tree_C0 <- tree(CO ~ . , train,
                  control = tree.control(nobs = length(train$CO),
                                         minsize = 4, mindev=0.001), method = "recursive.partition")
summary(tree_C0)

##
## Regression tree:
## tree(formula = CO ~ ., data = train, control = tree.control(nobs = length(train$CO),
##     minsize = 4, mindev = 0.001), method = "recursive.partition")
## Number of terminal nodes:  33
## Residual mean deviance:  1.012 = 5944 / 5874
## Distribution of residuals:
##      Min.    1st Qu.     Median      Mean    3rd Qu.      Max.
## -16.88000 -0.37240 -0.05792  0.00000  0.28590  30.18000

plot(tree_C0)
text(tree_C0, pretty = 0)

```

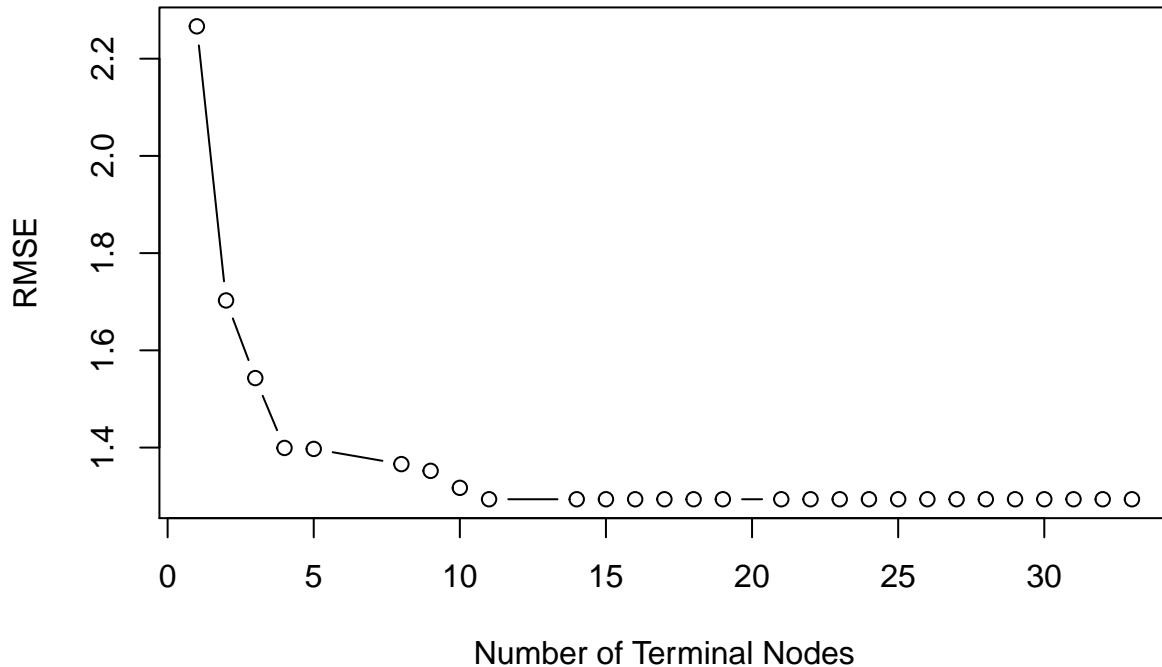


```
tree_pred <- predict(tree_C0, test)
RMSE(test$C0, tree_pred)
```

```
## [1] 1.316371
```

```
cv_info <- cv.tree(tree_C0, FUN = prune.tree)
plot(cv_info$size, sqrt(cv_info$dev / nrow(train)), type = "b", xlab = "Number of Terminal Nodes", ylab
```

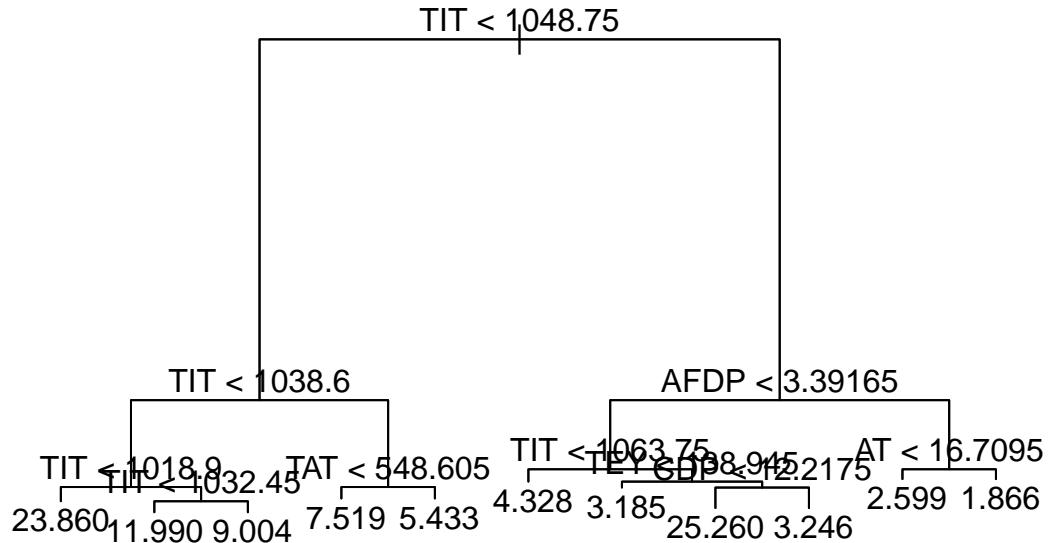
Decision Tree Cross Validation



```
pruned_tree <- prune.tree(tree_C0, best = 11)
summary(pruned_tree)
```

```
## 
## Regression tree:
## snip.tree(tree = tree_C0, nodes = c(19L, 10L, 14L, 11L, 12L,
## 26L, 18L, 15L))
## Variables actually used in tree construction:
## [1] "TIT"   "TAT"   "AFDP"  "TEY"   "CDP"   "AT"
## Number of terminal nodes: 11
## Residual mean deviance: 1.387 = 8180 / 5896
## Distribution of residuals:
##      Min.    1st Qu.     Median      Mean    3rd Qu.      Max. 
## -10.43000 -0.46410 -0.05351  0.00000  0.34680  34.59000
```

```
plot(pruned_tree)
text(pruned_tree, pretty = 0)
```



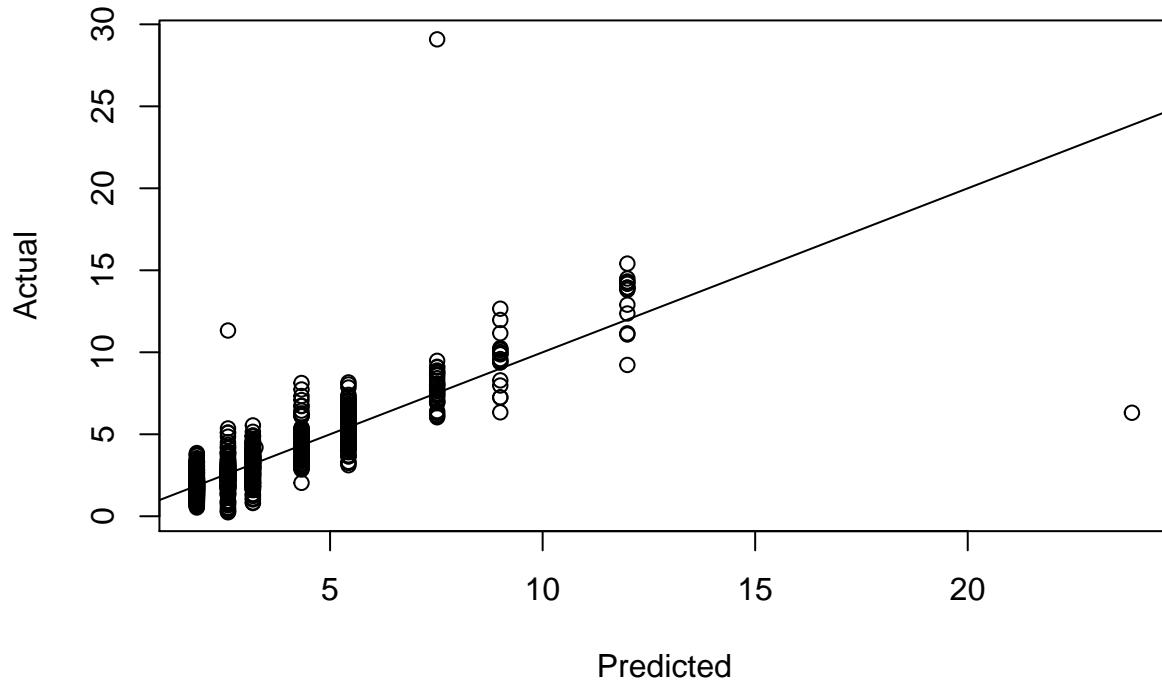
```

tree_pred <- predict(pruned_tree, test)
RMSE(test$C0, tree_pred)

## [1] 1.097033

plot(tree_pred, test$C0, xlab = "Predicted", ylab = "Actual")
abline(0, 1)

```



#Typical Energy Yield (130-136)

```

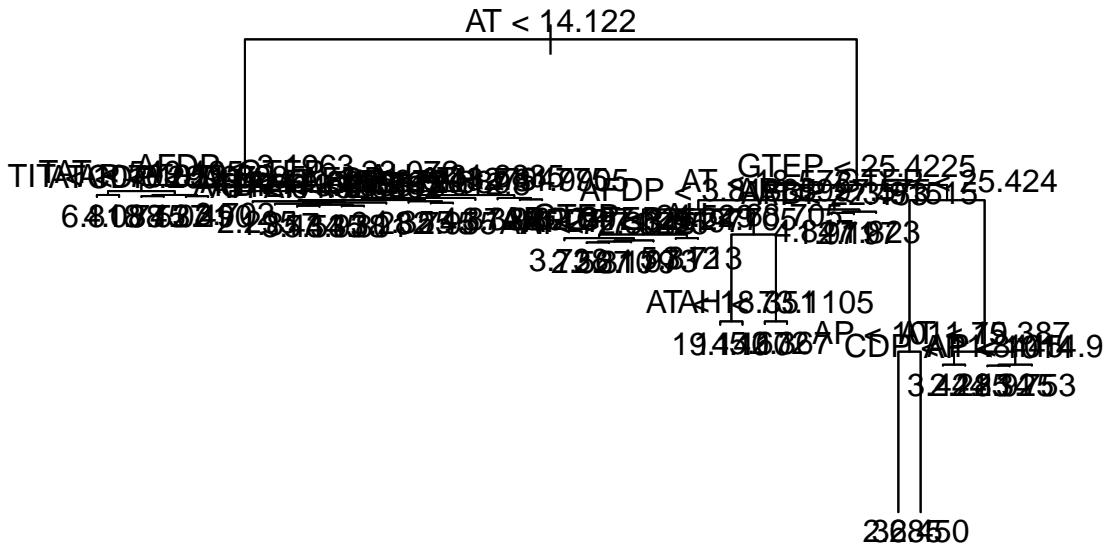
set.seed(10)
train_typical <- gt_2015_typical %>% dplyr::select(-NOX) %>% sample_frac(0.8)
test_typical <- gt_2015_typical %>% dplyr::select(-NOX) %>% setdiff(train_typical)

tree_C0_typical <- tree(C0 ~ . , train_typical,
                         control = tree.control(nobs = length(train_typical$C0),
                         minsize = 2, mindev=0.001), method = "recursive.partition")
summary(tree_C0_typical)

##
## Regression tree:
## tree(formula = C0 ~ ., data = train_typical, control = tree.control(nobs = length(train_typical$C0),
##     minsize = 2, mindev = 0.001), method = "recursive.partition")
## Variables actually used in tree construction:
## [1] "AT"    "AFDP"  "TAT"   "TIT"   "AP"    "CDP"   "GTEP"  "AH"
## Number of terminal nodes: 43
## Residual mean deviance: 0.1641 = 260.8 / 1589
## Distribution of residuals:
##      Min. 1st Qu. Median Mean 3rd Qu. Max.
## -2.29700 -0.23090 -0.01984 0.00000 0.19510 2.31900

plot(tree_C0_typical)
text(tree_C0_typical, pretty = 0)

```



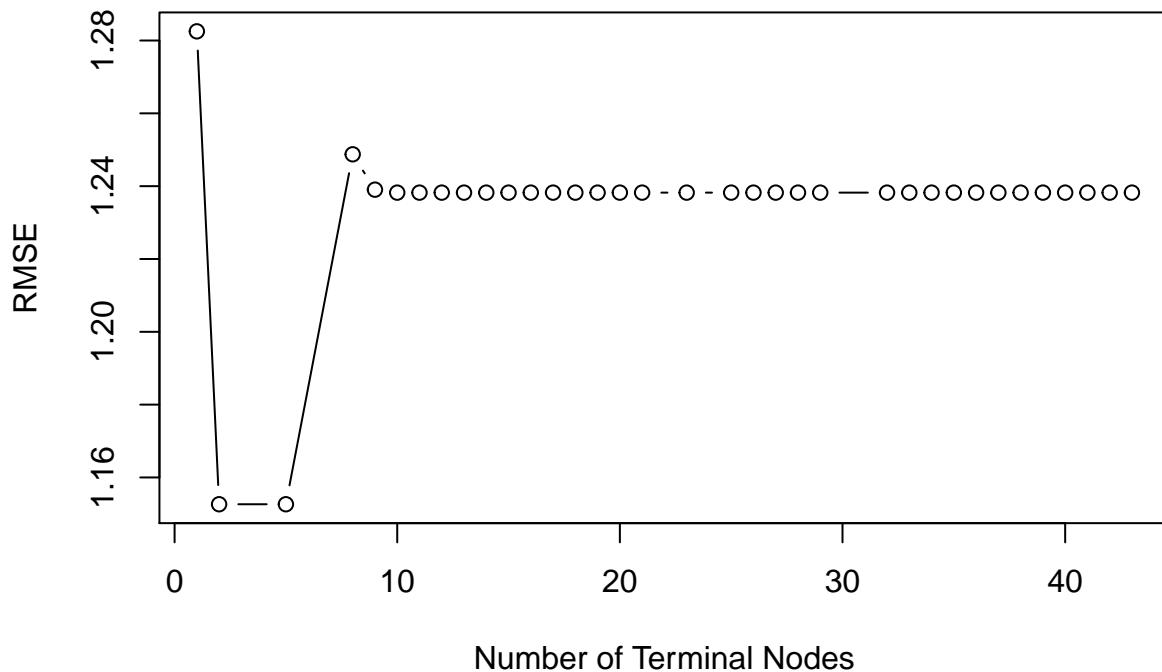
```
tree_pred_typical <- predict(tree_C0_typical, test_typical)
RMSE(test_typical$C0, tree_pred_typical)
```

```
## [1] 0.6818376
```

```
cv_info_typical <- cv.tree(tree_C0_typical, FUN = prune.tree)
```

```
plot(cv_info_typical$size, sqrt(cv_info_typical$dev / nrow(train_typical))), type = "b", xlab = "Number of observations")
```

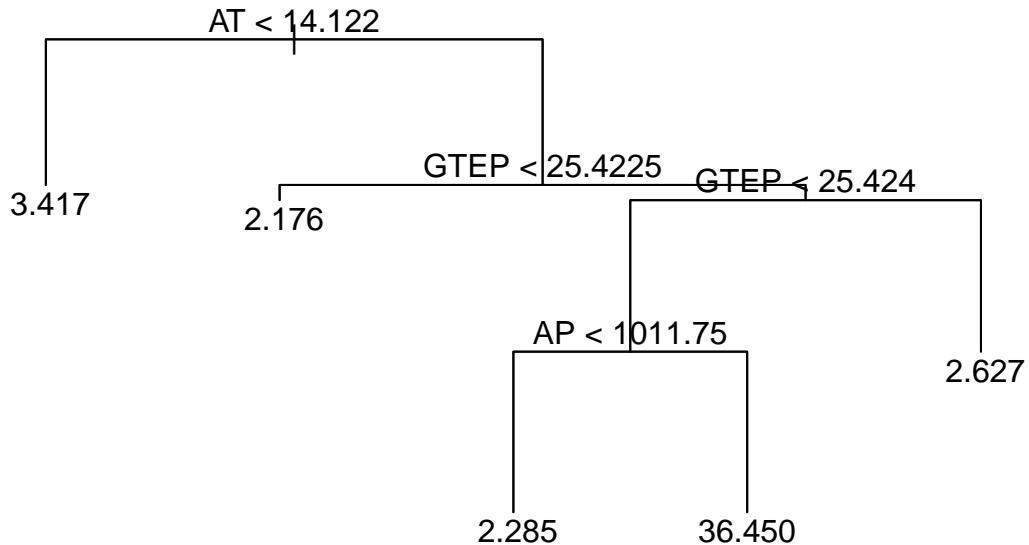
Decision Tree Cross Validation



```
pruned_tree_typical <- prune.tree(tree_C0_typical, best = 5)
summary(pruned_tree_typical)
```

```
## 
## Regression tree:
## snip.tree(tree = tree_C0_typical, nodes = c(2L, 15L, 6L))
## Variables actually used in tree construction:
## [1] "AT"    "GTEP"  "AP"
## Number of terminal nodes: 5
## Residual mean deviance:  0.5926 = 964.1 / 1627
## Distribution of residuals:
##      Min. 1st Qu. Median Mean 3rd Qu. Max.
## -2.61200 -0.35520 -0.05681 0.00000 0.27590 16.96000
```

```
plot(pruned_tree_typical)
text(pruned_tree_typical, pretty = 0)
```



```

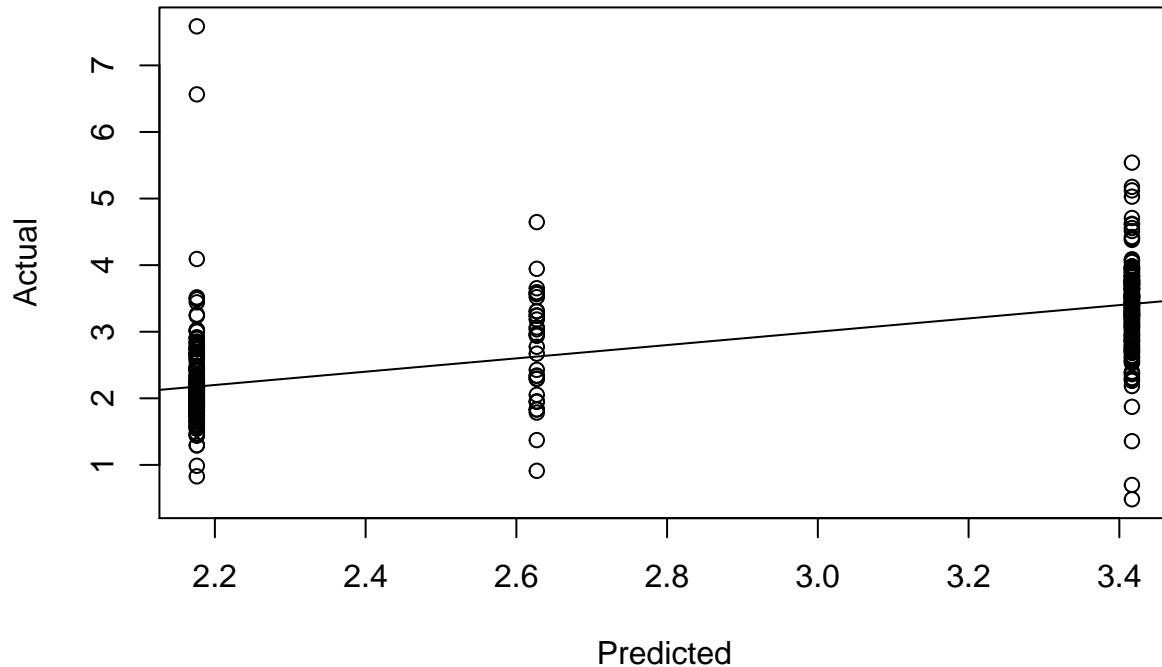
tree_pred_typical <- predict(pruned_tree_typical, test_typical)
RMSE(test_typical$C0, tree_pred_typical)
  
```

```

## [1] 0.6823446
  
```

```

plot(tree_pred_typical, test_typical$C0, xlab = "Predicted", ylab = "Actual")
abline(0, 1)
  
```



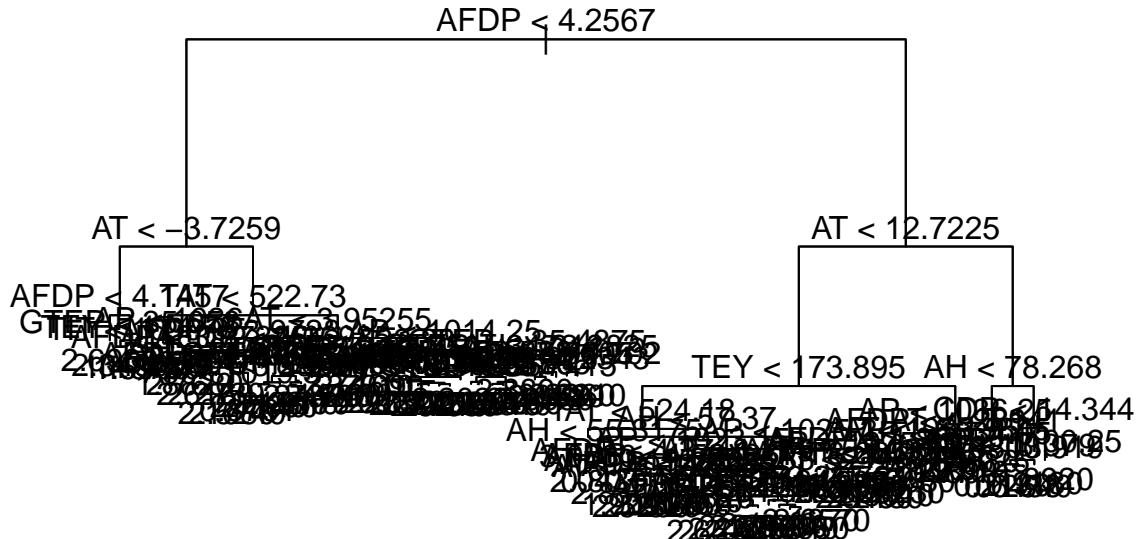
```
#High Energy Yield (160+)
```

```
set.seed(10)
train_high <- gt_2015_high %>% dplyr::select(-NOX) %>% sample_frac(0.8)
test_high <- gt_2015_high %>% dplyr::select(-NOX) %>% setdiff(train_high)

tree_CO_high <- tree(CO ~ . , train_high,
                      control = tree.control(nobs = length(train_high$CO),
                                             minsize = 2, mindev=0.001), method = "recursive.partition")
summary(tree_CO_high)

##
## Regression tree:
## tree(formula = CO ~ ., data = train_high, control = tree.control(nobs = length(train_high$CO),
##                     minsize = 2, mindev = 0.001), method = "recursive.partition")
## Number of terminal nodes:  89
## Residual mean deviance:  0.01502 = 3.679 / 245
## Distribution of residuals:
##      Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.28310 -0.06453  0.00000  0.00000  0.06841  0.26280

plot(tree_CO_high)
text(tree_CO_high, pretty = 0)
```

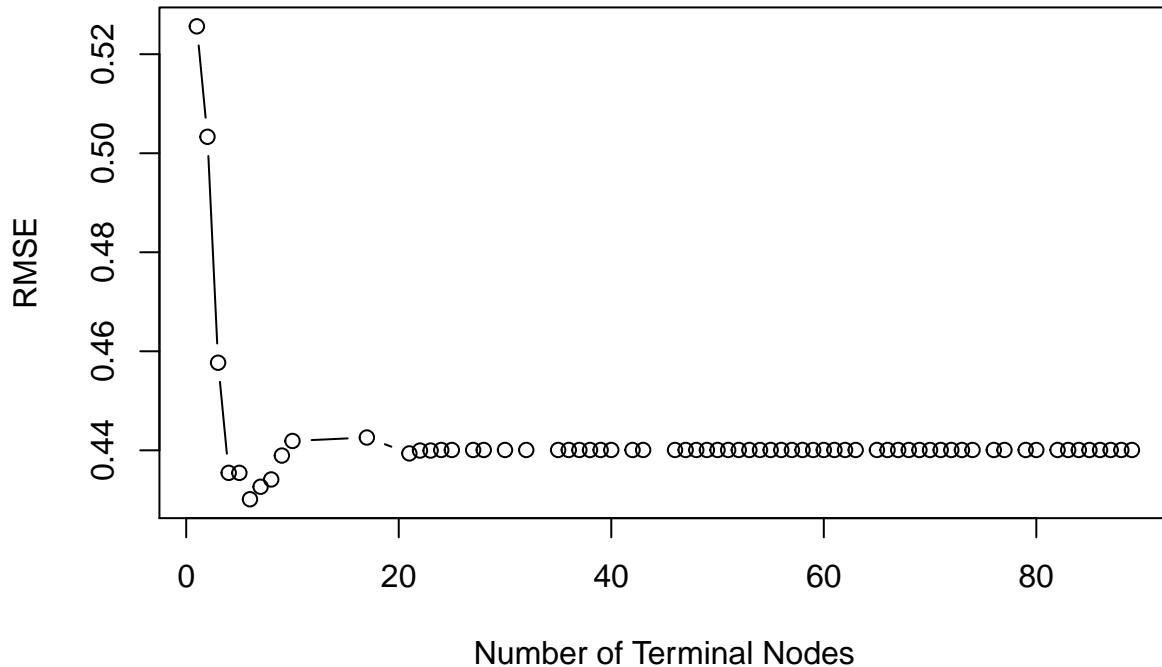


```
tree_pred_high <- predict(tree_C0_high, test_high)
RMSE(test_high$C0, tree_pred_high)
```

```
## [1] 0.507176
```

```
cv_info_high <- cv.tree(tree_C0_high, FUN = prune.tree)
plot(cv_info_high$size, sqrt(cv_info_high$dev / nrow(train_high)), type = "b", xlab = "Number of Terminal Nodes")
```

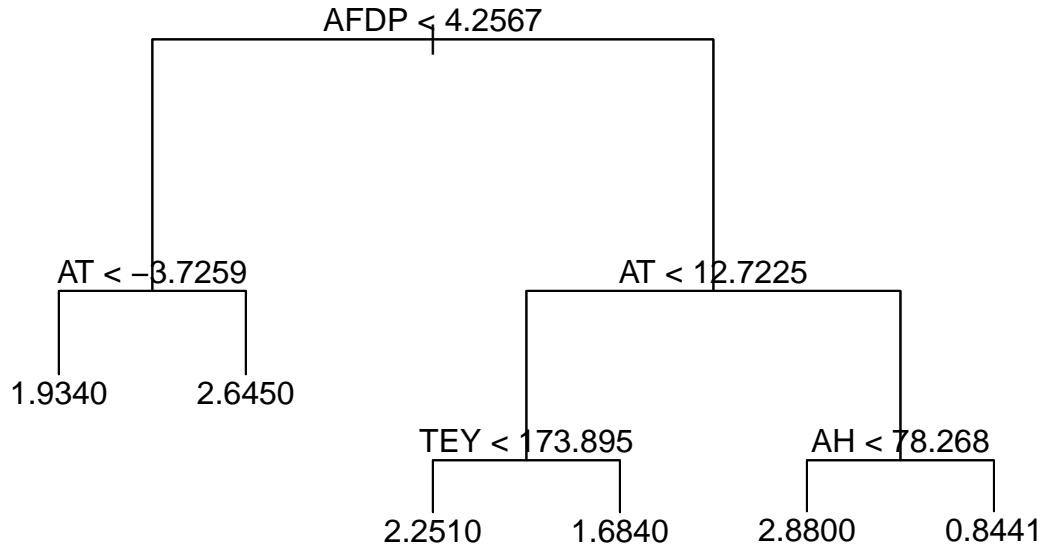
Decision Tree Cross Validation



```
pruned_tree_high <- prune.tree(tree_C0_high, best = 6)
summary(pruned_tree_high)
```

```
##
## Regression tree:
## snip.tree(tree = tree_C0_high, nodes = c(13L, 12L, 5L, 4L, 15L
## ))
## Variables actually used in tree construction:
## [1] "AFDP" "AT"    "TEY"   "AH"
## Number of terminal nodes: 6
## Residual mean deviance: 0.1334 = 43.76 / 328
## Distribution of residuals:
##      Min. 1st Qu. Median     Mean 3rd Qu.     Max.
## -1.76800 -0.19160 -0.01148  0.00000  0.21000  1.45000
```

```
plot(pruned_tree_high)
text(pruned_tree_high, pretty = 0)
```



```

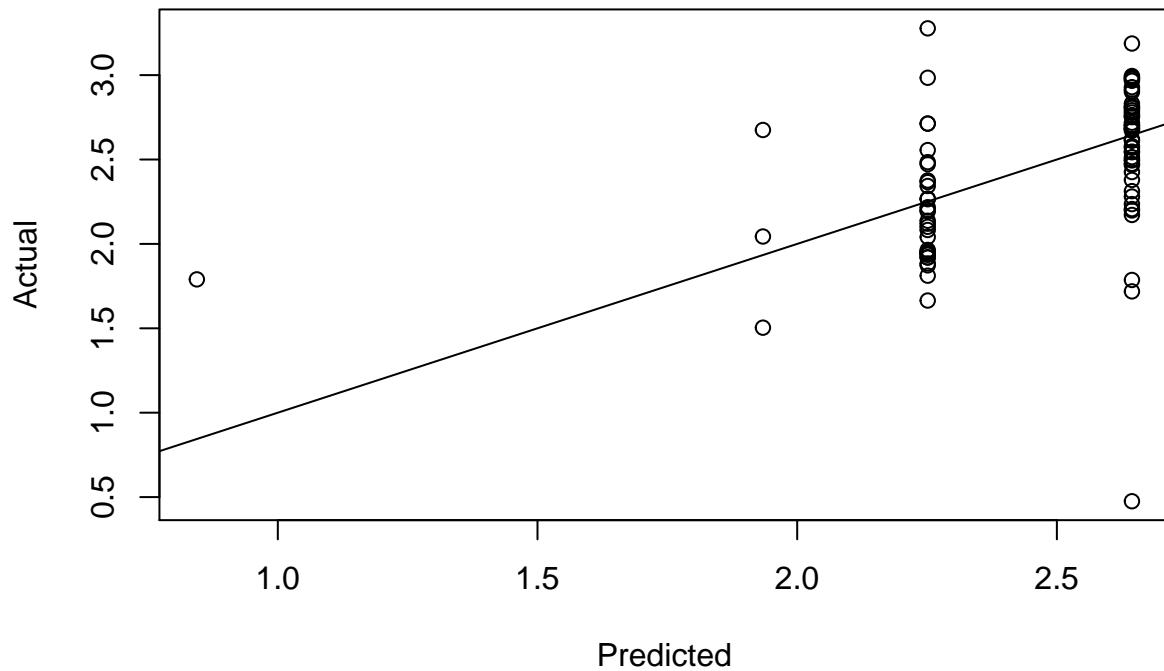
tree_pred_high <- predict(pruned_tree_high, test_high)
RMSE(test_high$C0, tree_pred_high)
  
```

```

## [1] 0.4154548
  
```

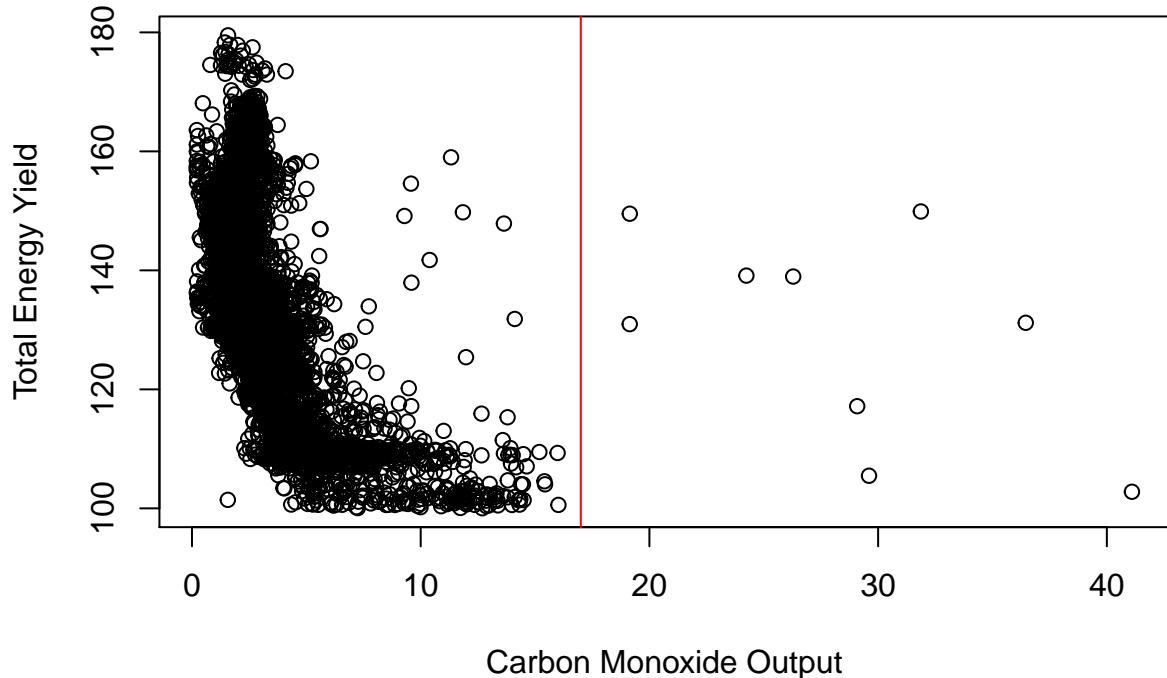
```

plot(tree_pred_high, test_high$C0, xlab = "Predicted", ylab = "Actual")
abline(0, 1)
  
```



Tree predicting high vs. low carbon monoxide output

```
plot(gt_2015$C0, gt_2015$TEY, ylab = "Total Energy Yield", xlab = "Carbon Monoxide Output")
abline(v = 17, col = "red")
```



```

data <- gt_2015 %>% mutate(Emissions = as.factor(ifelse(CO > 17, "High", "Low"))) %>% dplyr::select(-NO)
high_CO <- data %>% filter(CO > 17) %>% dplyr::select(-CO)
low_CO <- data %>% dplyr::select(-CO) %>% setdiff(high_CO)

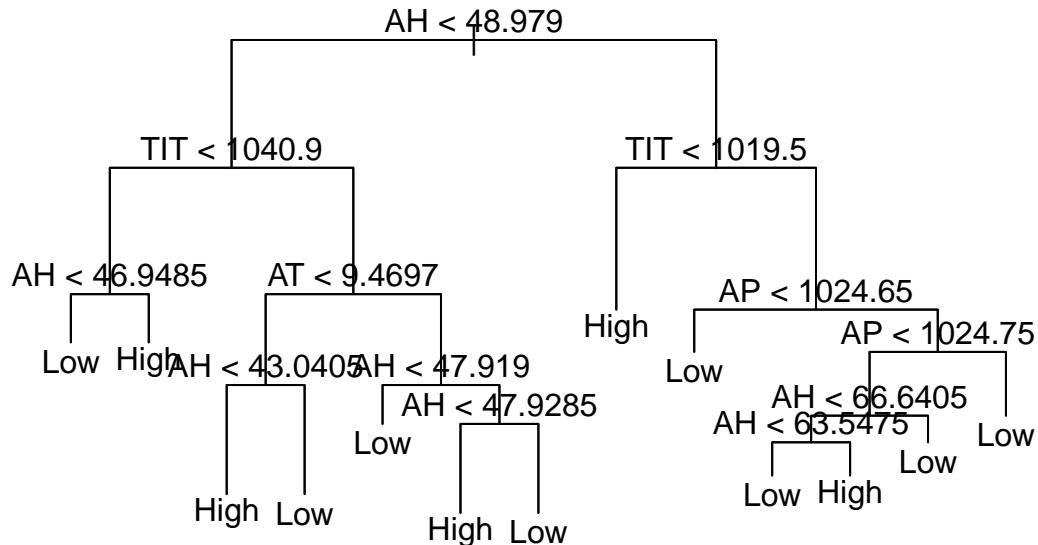
set.seed(10)
train <- bind_rows(low_CO %>% sample_frac(7/9), high_CO %>% sample_frac(7/9))
test <- data %>% dplyr::select(-CO) %>% setdiff(train)

tree <- tree(Emissions ~ . , train,
             control = tree.control(nobs = length(train$Emissions),
                                     minsize = 1))
summary(tree)

##
## Classification tree:
## tree(formula = Emissions ~ . , data = train, control = tree.control(nobs = length(train$Emissions),
##   minsize = 1))
## Variables actually used in tree construction:
## [1] "AH"   "TIT"  "AT"   "AP"
## Number of terminal nodes:  13
## Residual mean deviance:  0 = 0 / 5730
## Misclassification error rate: 0 = 0 / 5743

```

```
plot(tree)
text(tree, pretty = 0)
```



```
tree_pred <- predict(tree, train, type = "class")
table(predicted = tree_pred, actual = train$Emissions)
```

```
##           actual
## predicted High  Low
##       High     7    0
##       Low      0 5736
```

```
tree_pred <- predict(tree, test, type = "class")
table(predicted = tree_pred, actual = test$Emissions)
```

```
##           actual
## predicted High  Low
##       High     0    3
##       Low      2 1636
```

Tree predicting ratio of energy yield over carbon monoxide

```

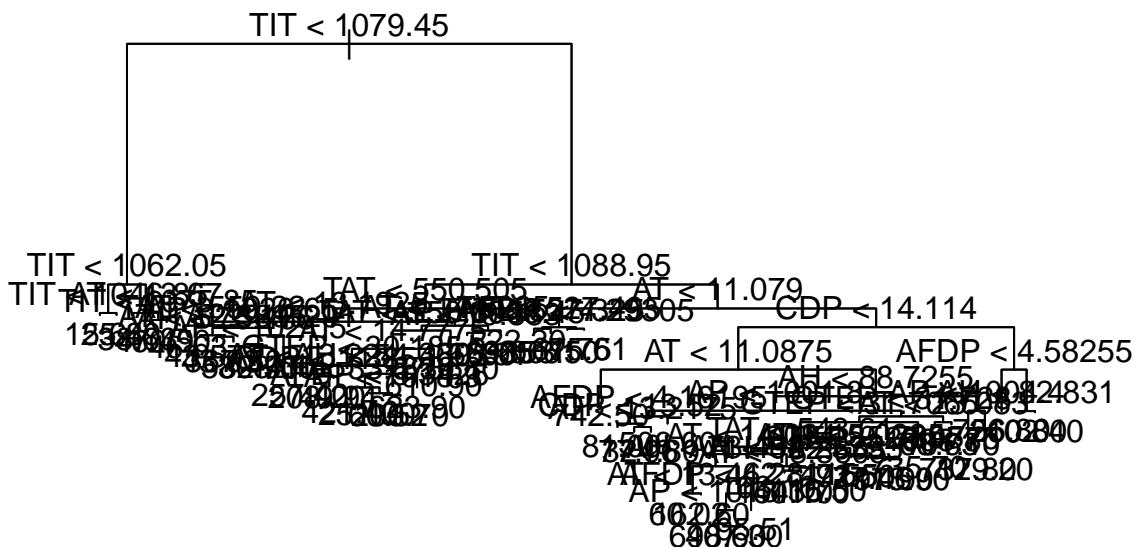
set.seed(10)
# train <- gt_2015 %>% mutate(Energy_CO_Ratio = TEY / CO) %>% sample_frac(0.8)
# test <- gt_2015 %>% mutate(Energy_CO_Ratio = TEY / CO) %>% setdiff(train)
train <- gt_2015 %>% mutate(Energy_CO_Ratio = TEY / CO) %>% dplyr::select(-c(NOX, TEY, CO)) %>% sample_
test <- gt_2015 %>% mutate(Energy_CO_Ratio = TEY / CO) %>% dplyr::select(-c(NOX, TEY, CO)) %>% setdiff(


tree_Energy_CO_Ratio <- tree(Energy_CO_Ratio ~ . , train,
                               control = tree.control(nobs = length(train$Energy_CO_Ratio),
                                                       minsize = 2, mindev=0.001), method = "recursive.partition")
summary(tree_Energy_CO_Ratio)

## 
## Regression tree:
## tree(formula = Energy_CO_Ratio ~ ., data = train, control = tree.control(nobs = length(train$Energy_-
##     minsize = 2, mindev = 0.001), method = "recursive.partition")
## Number of terminal nodes:  57
## Residual mean deviance:  476.9 = 2790000 / 5850
## Distribution of residuals:
##    Min. 1st Qu. Median 3rd Qu.   Max.
## -98.400 -8.495 -1.595  0.000  4.851 385.600

plot(tree_Energy_CO_Ratio)
text(tree_Energy_CO_Ratio, pretty = 0)

```



```

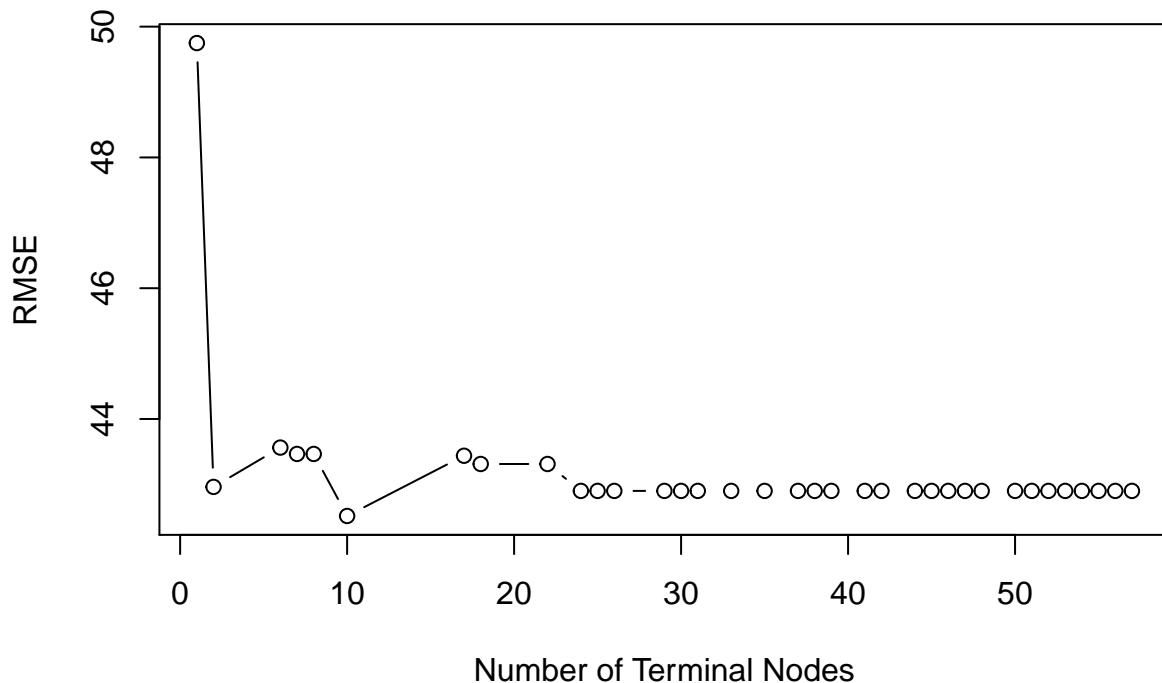
tree_pred <- predict(tree_Energy_CO_Ratio, test)
RMSE(test$Energy_CO_Ratio, tree_pred)

## [1] 39.95283

cv_info <- cv.tree(tree_Energy_CO_Ratio, FUN = prune.tree)
plot(cv_info$size, sqrt(cv_info$dev / nrow(train)), type = "b", xlab = "Number of Terminal Nodes", ylab =

```

Decision Tree Cross Validation



```

pruned_tree <- prune.tree(tree_Energy_CO_Ratio, best = 7)
summary(pruned_tree)

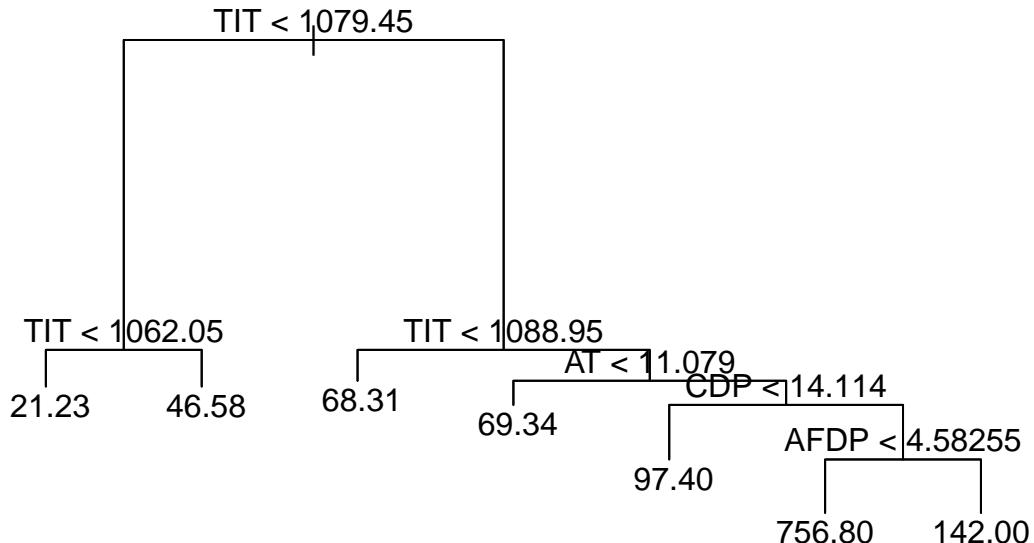
```

```

##
## Regression tree:
## snip.tree(tree = tree_Energy_CO_Ratio, nodes = c(63L, 4L, 14L,
## 5L, 6L, 30L))
## Variables actually used in tree construction:
## [1] "TIT"   "AT"    "CDP"   "AFDP"
## Number of terminal nodes: 7
## Residual mean deviance: 1430 = 8435000 / 5900
## Distribution of residuals:
##      Min. 1st Qu. Median  Mean 3rd Qu. Max.
## -92.690 -12.730 -4.005  0.000  6.385 645.100

```

```
plot(pruned_tree)
text(pruned_tree, pretty = 0)
```



```
tree_pred <- predict(pruned_tree, test)
RMSE(test$Energy_CO_Ratio, tree_pred)
```

```
## [1] 33.46526
```

```
plot(tree_pred, test$Energy_CO_Ratio, xlab = "Predicted", ylab = "Actual")
abline(0, 1)
```

