STAT 443 Appendix for Mid-Range Analysis

Jackie Kang, Jonas Reger | Glosemeyer

03 May, 2021

Appendix for Mid-Range Analysis

Packages & Functions

```
### PACKAGES ###
suppressMessages(library(tidyverse))
suppressMessages(library(MASS))
suppressMessages(library(faraway))
suppressMessages(library(leaps))
suppressMessages(library(lmtest))
suppressMessages(library(boot))

### FUNCTIONS ###
# Plot function to show outliers more clearly.
abplot = function(X, Y, Xa, Ya, Xo, Yo, c){
  sX = sd(X); sY = sd(Y); r = cor(X, Y); m = sY/sX*r
  sXa = sd(Xa); sYa = sd(Ya); ra = cor(Xa, Ya); ma = sYa/sXa*ra
  mX = mean(X); mY = mean(Y)
  mXa = mean(Xa); mYa = mean(Ya)
  yint = mY - mX*m
  yinta = mYa - mXa*ma
  plot(X, Y, pch=20, col="#13294b")
  points(Xo, Yo, pch=20, col=c)
  abline(yint, m, col="dodgerblue", lty=2, lwd=2)
  abline(yinta, ma, col="#e84a27", lwd=2)
}
#Fitted VS Residual Test
test_FvsR = function(model){
  plot(fitted(model), resid(model), xlab = "Fitted", ylab = "Residuals", main = "Residual Plot", col =
  abline(h = 0, col = "#e84a27")
}
#QQ plot & QQline
test_qq = function(model){
  qqnorm(resid(model), main = "Normal Q-Q plot", col = "#13294b", pch=20)
  qqline(resid(model), col = "#e84a27")
}
#BP test
test_bp = function(model, alpha) {
  p_val = bptest(model)$p.value
  decision = ifelse(p_val < alpha,"Reject", "Fail to Reject")
```

```
  list1 = list(bp.p_val = p_val, bp.decision = decision)
  return(list1)
}
# Shapiro-Wilk test (sw test)
test_sw = function(model, alpha) {
  p_val = shapiro.test(resid(model))$p.value
  decision = ifelse(p_val < alpha,"Reject", "Fail to Reject")
  list1 = list(sw.p_val = p_val, sw.decision = decision)
  return(list1)
}
#Calculate loocv_rmse
get_loocv_rmse = function(model) {
  sqrt(mean((resid(model) / (1 - hatvalues(model))) ^ 2))
}
#Calculate adr2
get_adj_r2 = function(model) {
  summary(model)$adj.r.squared
}
#Get number of parameters
get_num_params = function(model) {
  length(coef(model))
}
#Run All Test Diagnostic
run_all_diagnostic = function(model, visual=TRUE, alpha=0.05){
  if (visual) {
    par(mfrow = c(1,2))
    test_FvsR(model)
    test_qq(model)
  }
  loocv_rmse = get_loocv_rmse(model)
  adj.r.squared = get_adj_r2(model)
  params = get_num_params(model)
  bp = test_bp(model, alpha)
  sw = test_sw(model, alpha)
  data_frame = data.frame(loocv_rmse, adj.r.squared, params, bp, sw, row.names = '')
  return(data_frame)
}
# BIC exhaustive selection
gt_bic_sel = function(df, full){
  all_mod = summary(regsubsets(CO ~ ., data=df, nvmax=11, method='exhaustive'))
  p = length(coef(full))
  n = length(resid(full))
  mod_bic = n*log(all_mod$rss/n) + log(n)*(2:p)
  m = which.min(mod_bic)
  all_mod$which[m,]
}
# Final Model 1 - Shows changes in CO when changing one predictor while holding others constant.
pred.mb = function(x, mbc, chng, i, name){
  m = 2*sd(chng)
  ms = seq(mean(chng)-m, mean(chng)+m, length.out = 100)
  c=numeric(100)
  for(k in 1:100){
    xvals = x
```

```
    xvals[i] = ms[k]
    mbp = c(
      predict(object = poly(gt_mid$AT, 2), newdata = xvals[1]),
      predict(object = poly(gt_mid$AP, 2), newdata = xvals[2]),
      predict(object = poly(gt_mid$AH, 2), newdata = xvals[3]),
      predict(object = poly(gt_mid$AFDP, 2), newdata = xvals[4]),
      predict(object = poly(gt_mid$TIT, 2), newdata = xvals[5]),
      predict(object = poly(gt_mid$TAT, 2), newdata = xvals[6]),
      predict(object = poly(gt_mid$CDP, 2), newdata = xvals[7])
            )
    mb = mbc[1] + mbc[2]*xvals[1] + mbc[3]*xvals[2] + mbc[4]*xvals[3] +
      mbc[5]*mbp[7] + mbc[6]*mbp[8] +
      mbc[7]*mbp[9] + mbc[8]*mbp[10] +
      mbc[9]*mbp[11] + mbc[10]*mbp[12] +
      mbc[11]*mbp[13] + mbc[12]*mbp[14]
    c[k] = mb-1.602807
  }
  minval = ms[which.min(c)]
  minval1se = ms[which.min(c[26:75])+25]
  plot(ms, c, main=paste(name, "vs. CO predicted change"),
       xlab=paste0(name, " (CO Min at ", name, " = ", round(minval, 1), ")"),
       ylab = paste0("CO diff (", round(max(c)-min(c), 2), ")"),
       sub = paste0("CO diff Min = ", round(min(c), 2), ", Min 1se = ", round(min(c[26:75]), 2)),
       type='l', col="#13294b")
  abline(v=minval, col="#e84a27")
  abline(v=minval1se, col="#e84a27")
  abline(v=mean(chng), col="dodgerblue", lty=2)
  abline(h=0, col="dodgerblue", lty=2)
}
# Final Model 2 - Shows changes in CO when changing one predictor while holding others constant.
pred.mb2 = function(x, mbc, chng, i, name){
  m = 2*sd(chng)
  ms = seq(mean(chng)-m, mean(chng)+m, length.out = 100)
  c=numeric(100)
  for(k in 1:100){
    xvals = x
    xvals[i] = ms[k]
    jmp = c(
      predict(object = poly(gt_mid_all$AP, 2), newdata = xvals[1]),
      predict(object = poly(gt_mid_all$AT, 2), newdata = xvals[2]),
      predict(object = poly(gt_mid_all$AH, 2), newdata = xvals[3]),
      predict(object = poly(gt_mid_all$AFDP, 2), newdata = xvals[4]),
      predict(object = poly(gt_mid_all$TAT, 2), newdata = xvals[5]),
      predict(object = poly(gt_mid_all$TIT, 2), newdata = xvals[6]),
      predict(object = poly(gt_mid_all$NOX, 2), newdata = xvals[7])
            )
    jm = jmc[1] +
      jmc[2]*jmp[1] + jmc[3]*jmp[2] +
      jmc[4]*jmp[3] + jmc[5]*jmp[4] +
      jmc[6]*jmp[5] + jmc[7]*jmp[6] +
      jmc[8]*jmp[7] + jmc[9]*jmp[8] +
      jmc[10]*jmp[9] + jmc[11]*jmp[10] +
      jmc[12]*jmp[11] + jmc[13]*jmp[12] +
```

```
      jmc[14]*jmp[13] + jmc[15]*jmp[14]
    c[k] = jm-1.325417
  }
  minval = ms[which.min(c)]
  minval1se = ms[which.min(c[26:75])+25]
  plot(ms, c, main=paste(name, "vs. CO predicted change"),
       xlab=paste0(name, " (CO Min at ", name, " = ", round(minval, 1), ")"),
       ylab = paste0("CO diff (", round(max(c)-min(c), 2), ")"),
       sub = paste0("CO diff Min = ", round(min(c), 2), ", Min 1se = ", round(min(c[26:75]), 2)),
       type='l', col="#13294b")
  abline(v=minval, col="#e84a27")
  abline(v=minval1se, col="#e84a27")
  abline(v=mean(chng), col="dodgerblue", lty=2)
  abline(h=0, col="dodgerblue", lty=2)
}
```

Import & Modify Data

```
gt_2012 <- read_csv("gt_2012.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   AT = col_double(),
##   AP = col_double(),
##   AH = col_double(),
##   AFDP = col_double(),
##   GTEP = col_double(),
##   TIT = col_double(),
##   TAT = col_double(),
##   TEY = col_double(),
##   CDP = col_double(),
##   CO = col_double(),
##   NOX = col_double()
## )
```

```
gt_mid_all <- gt_2012 %>% filter(TEY >= 130 & TEY <=136)
gt_high_all <- gt_2012 %>% filter(TEY >= 160)
gt_mid <- subset(gt_mid_all, select = -NOX)
head(gt_mid)
```

```
## # A tibble: 6 x 10
##      AT    AP    AH  AFDP  GTEP   TIT   TAT   TEY   CDP    CO
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1  8.08 1015   88.6  4.06  23.4 1083   550.  132.  11.7  1.91
## 2  8.30 1016   86.3  4.09  23.7 1085.  550.  134.  11.7  1.71
## 3  8.47 1016.  86.5  4.05  23.7 1085.  550.  134.  11.8  1.47
## 4  8.89 1016.  83.0  4.05  23.9 1086.  550.  135.  11.9  1.71
## 5  9.37 1017.  80.0  4.04  23.9 1086.  550.  135.  11.9  1.66
## 6  9.80 1017.  78.1  4.06  24.0 1086.  550.  135.  11.9  1.41
```

Model Diagnostics

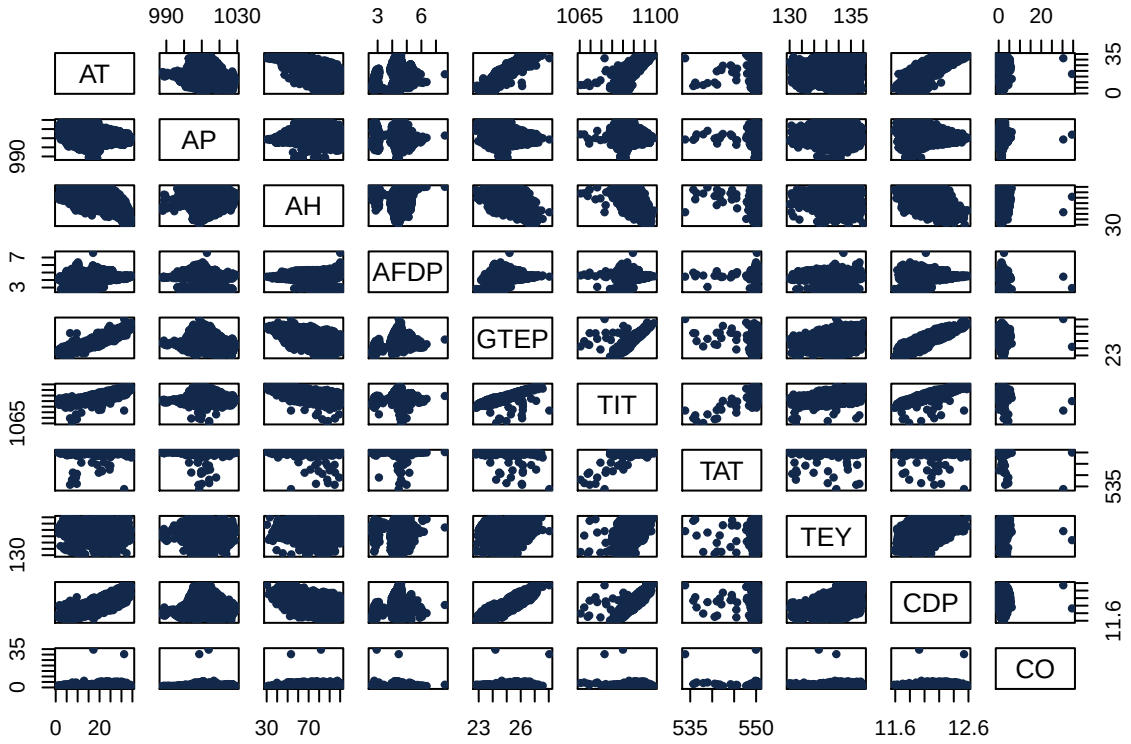
```
mid1 <- lm(CO ~., data = gt_mid)
summary(mid1)
```

```
##
## Call:
## lm(formula = CO ~ ., data = gt_mid)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -4.394 -0.369 -0.054  0.224 32.600
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 206.370163  13.635908  15.134  < 2e-16 ***
## AT            0.031552   0.012242   2.577  0.00999 **
## AP            0.009148   0.003796   2.410  0.01600 *
## AH            0.009878   0.001374   7.190 7.74e-13 ***
## AFDP         -0.088187   0.029368  -3.003  0.00269 **
## GTEP          0.018078   0.124452   0.145  0.88451
## TIT          -0.077685   0.026377  -2.945  0.00325 **
## TAT          -0.261402   0.042412  -6.163 7.84e-10 ***
## TEY           0.029471   0.035248   0.836  0.40315
## CDP           0.741095   0.246368   3.008  0.00265 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9409 on 3897 degrees of freedom
## Multiple R-squared:  0.0816, Adjusted R-squared:  0.07948
## F-statistic: 38.47 on 9 and 3897 DF,  p-value: < 2.2e-16
```

```
v = vif(mid1)
v[which(v > 5)]
```

```
##        AT      GTEP       TIT       CDP
## 36.443438 52.733917 40.940070  8.185557
```

```
pairs(gt_mid, pch=20, col="#13294b")
```




```
round(cor(gt_mid),2)
```

```
##          AT    AP    AH  AFDP  GTEP   TIT   TAT   TEY   CDP    CO
## AT    1.00 -0.30 -0.62  0.29  0.95  0.89  0.06 -0.11  0.86 -0.02
## AP   -0.30  1.00  0.04 -0.17 -0.30 -0.11  0.00  0.04 -0.13  0.01
## AH   -0.62  0.04  1.00 -0.14 -0.58 -0.58 -0.01  0.08 -0.56  0.10
## AFDP  0.29 -0.17 -0.14  1.00  0.37  0.29 -0.03  0.00  0.30 -0.05
## GTEP  0.95 -0.30 -0.58  0.37  1.00  0.92 -0.03  0.11  0.91 -0.02
## TIT   0.89 -0.11 -0.58  0.29  0.92  1.00  0.25  0.23  0.90 -0.09
## TAT   0.06  0.00 -0.01 -0.03 -0.03  0.25  1.00  0.05  0.00 -0.25
## TEY  -0.11  0.04  0.08  0.00  0.11  0.23  0.05  1.00  0.21 -0.03
## CDP   0.86 -0.13 -0.56  0.30  0.91  0.90  0.00  0.21  1.00 -0.01
## CO   -0.02  0.01  0.10 -0.05 -0.02 -0.09 -0.25 -0.03 -0.01  1.00
```

```
### Partial Correlation Coefficient between CO and AT
mod1a <-lm(CO ~.-AT, data = gt_mid)
mod1b <-lm(AT ~. - CO, data = gt_mid)
# Result Small, can remove the variable from the model
cor(resid(mod1a), resid(mod1b))
```

```
## [1] 0.04125209
```

```
### Partial Correlation Coefficient between CO and GTEP
mod1c <-lm(CO ~.-GTEP, data = gt_mid)
```

```r
mod1d <-lm(GTEP ~. - CO, data = gt_mid)
# Result Small, can remove the variable from the model
cor(resid(mod1c), resid(mod1d))
```

```
## [1] 0.002326901
```

```r
### Partial Correlation Coefficient between CO and CDP
mod1e<-lm(CO ~.-CDP, data = gt_mid)
mod1f <-lm(CDP ~. - CO, data = gt_mid)
# Result Small, can remove the variable from the model
cor(resid(mod1f), resid(mod1e))
```

```
## [1] 0.04813062
```

```r
### Partial Correlation Coefficient between CO and TIT
mod1h <-lm(CO ~.-TIT, data = gt_mid)
mod1g <-lm(TIT ~. - CO, data = gt_mid)
# Result Small, can remove the variable from the model
cor(resid(mod1h), resid(mod1g))
```

```
## [1] -0.0471264
```

Model 1 Building

```r
mid2 <- lm(CO ~. - GTEP, data = gt_mid)
summary(mid2)
```

```
##
## Call:
## lm(formula = CO ~ . - GTEP, data = gt_mid)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -4.394 -0.368 -0.055  0.223 32.598
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 206.802635  13.305245  15.543  < 2e-16 ***
## AT            0.032618   0.009797   3.329 0.000879 ***
## AP            0.008916   0.003441   2.591 0.009606 **
## AH            0.009906   0.001359   7.288 3.79e-13 ***
## AFDP         -0.086420   0.026725  -3.234 0.001233 **
## TIT          -0.075646   0.022328  -3.388 0.000711 ***
## TAT          -0.265397   0.032284  -8.221 2.73e-16 ***
## TEY           0.030750   0.034125   0.901 0.367599
## CDP           0.743763   0.245651   3.028 0.002480 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.9408 on 3898 degrees of freedom
## Multiple R-squared:  0.08159,    Adjusted R-squared:  0.07971
## F-statistic: 43.29 on 8 and 3898 DF,  p-value: < 2.2e-16
```

```
v = vif(mid2)
v[which(v > 5)]
```

```
##        AT       TIT       CDP
## 23.349252 29.344026  8.140064
```

```
anova(mid2, mid1)#mod2 is better
```

```
## Analysis of Variance Table
##
## Model 1: CO ~ (AT + AP + AH + AFDP + GTEP + TIT + TAT + TEY + CDP) - GTEP
## Model 2: CO ~ AT + AP + AH + AFDP + GTEP + TIT + TAT + TEY + CDP
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1   3898 3450.1
## 2   3897 3450.0  1   0.01868 0.0211 0.8845
```

```
mid3 <- lm(CO ~. - GTEP - TIT - AT, data = gt_mid)
summary(mid3)
```

```
##
## Call:
## lm(formula = CO ~ . - GTEP - TIT - AT, data = gt_mid)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -4.379 -0.365 -0.052  0.231 32.663
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 189.164523  12.341923  15.327  < 2e-16 ***
## AP            0.001642   0.002658   0.618  0.53686
## AH            0.010146   0.001298   7.819 6.82e-15 ***
## AFDP         -0.103873   0.026261  -3.955 7.78e-05 ***
## TAT          -0.341665   0.021575 -15.836  < 2e-16 ***
## TEY          -0.066205   0.019146  -3.458  0.00055 ***
## CDP           0.596718   0.114685   5.203 2.06e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.942 on 3900 degrees of freedom
## Multiple R-squared:  0.07867,    Adjusted R-squared:  0.07726
## F-statistic:  55.5 on 6 and 3900 DF,  p-value: < 2.2e-16
```

```
v = vif(mid3)
v[which(v > 5)]
```

```
## named numeric(0)
```

```r
mid4_aic_back <- step(mid1, trace = FALSE)
summary(mid4_aic_back) #REMOVE GTEP & TEY
```

```
##
## Call:
## lm(formula = CO ~ AT + AP + AH + AFDP + TIT + TAT + CDP, data = gt_mid)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -4.420 -0.368 -0.055  0.224 32.596
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 203.797813  12.880277  15.822  < 2e-16 ***
## AT            0.025345   0.005554   4.563 5.19e-06 ***
## AP            0.007391   0.002996   2.467 0.013677 *
## AH            0.010053   0.001349   7.450 1.14e-13 ***
## AFDP         -0.090705   0.026298  -3.449 0.000568 ***
## TIT          -0.061029   0.015343  -3.978 7.09e-05 ***
## TAT          -0.278850   0.028624  -9.742  < 2e-16 ***
## CDP           0.767980   0.244171   3.145 0.001672 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9408 on 3899 degrees of freedom
## Multiple R-squared:  0.0814, Adjusted R-squared:  0.07975
## F-statistic: 49.36 on 7 and 3899 DF,  p-value: < 2.2e-16
```

```r
anova(mid4_aic_back, mid2)#mid4_aic_back
```

```
## Analysis of Variance Table
##
## Model 1: CO ~ AT + AP + AH + AFDP + TIT + TAT + CDP
## Model 2: CO ~ (AT + AP + AH + AFDP + GTEP + TIT + TAT + TEY + CDP) - GTEP
##   Res.Df    RSS Df Sum of Sq     F Pr(>F)
## 1   3899 3450.8
## 2   3898 3450.1  1   0.71865 0.812 0.3676
```

```r
mid5 <- lm(CO ~ AT + AP + AH + poly(AFDP,2) + poly(TIT,2) + poly(TAT,2) + poly(CDP,2),data = gt_mid)

# boxcox transformation didn't help model
# boxcox(mid5, lambda = seq(0.25, 0.5, length.out=100))

summary(mid5) #ambient variable may not have strong influence over CO emission
```

```
##
## Call:
## lm(formula = CO ~ AT + AP + AH + poly(AFDP, 2) + poly(TIT, 2) +
##     poly(TAT, 2) + poly(CDP, 2), data = gt_mid)
##
## Residuals:
```

```
##    Min     1Q Median     3Q    Max
## -6.416 -0.351 -0.056  0.229 32.869
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -6.070530   3.042977  -1.995 0.046120 *
## AT               0.019646   0.005527   3.555 0.000383 ***
## AP               0.006471   0.002927   2.211 0.027091 *
## AH               0.009372   0.001354   6.924 5.11e-12 ***
## poly(AFDP, 2)1  -3.194292   0.990177  -3.226 0.001266 **
## poly(AFDP, 2)2  -2.977301   1.008120  -2.953 0.003163 **
## poly(TIT, 2)1   -3.934220   3.440565  -1.143 0.252909
## poly(TIT, 2)2  -18.349490   1.603611 -11.443  < 2e-16 ***
## poly(TAT, 2)1  -24.983758   1.570303 -15.910  < 2e-16 ***
## poly(TAT, 2)2   16.797544   0.950353  17.675  < 2e-16 ***
## poly(CDP, 2)1    2.510030   2.581421   0.972 0.330940
## poly(CDP, 2)2   12.503072   1.246289  10.032  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8984 on 3895 degrees of freedom
## Multiple R-squared:  0.1631, Adjusted R-squared:  0.1607
## F-statistic: 69.01 on 11 and 3895 DF,  p-value: < 2.2e-16
```

```
v = vif(mid5)
v[which(v > 5)]
```

```
##            AT poly(TIT, 2)1 poly(CDP, 2)1
##      8.147728     14.665877      8.255930
```

```
mid6_back_aic = step(mid5, trace = FALSE)
summary(mid6_back_aic)
```

```
##
## Call:
## lm(formula = CO ~ AT + AP + AH + poly(AFDP, 2) + poly(TIT, 2) +
##     poly(TAT, 2) + poly(CDP, 2), data = gt_mid)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.416 -0.351 -0.056  0.229 32.869
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -6.070530   3.042977  -1.995 0.046120 *
## AT               0.019646   0.005527   3.555 0.000383 ***
## AP               0.006471   0.002927   2.211 0.027091 *
## AH               0.009372   0.001354   6.924 5.11e-12 ***
## poly(AFDP, 2)1  -3.194292   0.990177  -3.226 0.001266 **
## poly(AFDP, 2)2  -2.977301   1.008120  -2.953 0.003163 **
## poly(TIT, 2)1   -3.934220   3.440565  -1.143 0.252909
## poly(TIT, 2)2  -18.349490   1.603611 -11.443  < 2e-16 ***
## poly(TAT, 2)1  -24.983758   1.570303 -15.910  < 2e-16 ***
```

```
## poly(TAT, 2)2   16.797544   0.950353  17.675  < 2e-16 ***
## poly(CDP, 2)1    2.510030   2.581421   0.972 0.330940
## poly(CDP, 2)2   12.503072   1.246289  10.032  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8984 on 3895 degrees of freedom
## Multiple R-squared:  0.1631, Adjusted R-squared:  0.1607
## F-statistic: 69.01 on 11 and 3895 DF,  p-value: < 2.2e-16
```

```
mod1 <- lm(CO~1, data = gt_mid)
mid6_aic_both = step(
  mod1,
  scope = CO ~ poly(AT,2) + poly(AP,2) + poly(AH,2) + poly(AFDP, 2) + poly(TIT, 2) +
    poly(TAT, 2) + poly(CDP, 2),
  direction = 'both', trace = FALSE
)
summary(mid6_aic_both)
```

```
##
## Call:
## lm(formula = CO ~ poly(TAT, 2) + poly(TIT, 2) + poly(CDP, 2) +
##     poly(AH, 2) + poly(AFDP, 2) + poly(AP, 2) + poly(AT, 2),
##     data = gt_mid)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.407 -0.356 -0.059  0.225 32.838
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       1.60300    0.01435 111.717  < 2e-16 ***
## poly(TAT, 2)1   -25.25392    1.58451 -15.938  < 2e-16 ***
## poly(TAT, 2)2    16.87723    0.95176  17.733  < 2e-16 ***
## poly(TIT, 2)1    -4.25673    3.48362  -1.222 0.221810
## poly(TIT, 2)2   -19.14042    1.70372 -11.234  < 2e-16 ***
## poly(CDP, 2)1     1.49161    2.59085   0.576 0.564837
## poly(CDP, 2)2    12.66316    1.29583   9.772  < 2e-16 ***
## poly(AH, 2)1      8.01760    1.28599   6.235 5.01e-10 ***
## poly(AH, 2)2     -0.56378    0.97985  -0.575 0.565068
## poly(AFDP, 2)1   -2.95986    0.99401  -2.978 0.002922 **
## poly(AFDP, 2)2   -2.77873    1.02840  -2.702 0.006923 **
## poly(AP, 2)1      1.86752    1.07205   1.742 0.081586 .
## poly(AP, 2)2     -3.87681    1.00344  -3.864 0.000114 ***
## poly(AT, 2)1      8.52539    2.67588   3.186 0.001454 **
## poly(AT, 2)2      1.30570    1.28758   1.014 0.310609
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8969 on 3892 degrees of freedom
## Multiple R-squared:  0.1666, Adjusted R-squared:  0.1636
## F-statistic: 55.57 on 14 and 3892 DF,  p-value: < 2.2e-16
```

```
v = vif(mid6_aic_both)
v[which(v > 5)]
```

```
## poly(TIT, 2)1 poly(CDP, 2)1  poly(AT, 2)1
##     15.086582      8.344776      8.901465
```

```
critval = qt(0.05/(2*nobs(mid5)), df=df.residual(mid5)-1, lower=FALSE)
out_ind = which(abs(rstudent(mid5)) > critval)
out_ind
```

```
##  217  308 1090 2088 2089 2090 3414 3544
##  217  308 1090 2088 2089 2090 3414 3544
```

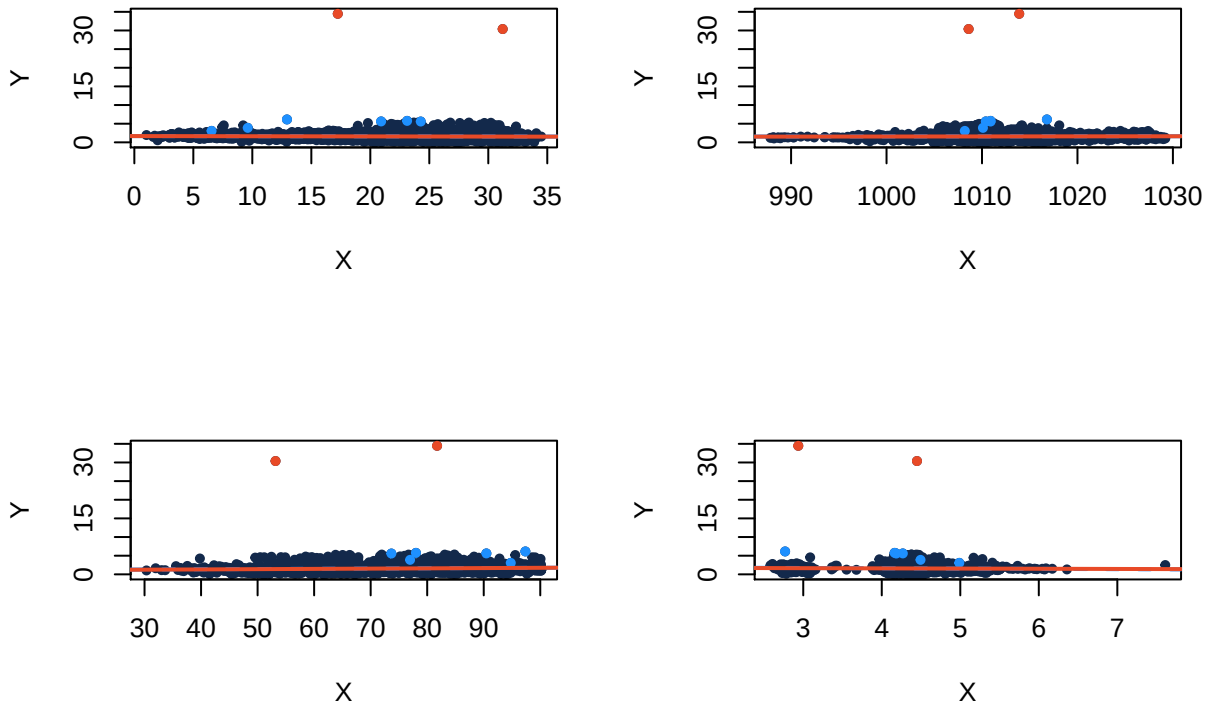
```
gt_adj = gt_mid[-c(out_ind)[c(3, 8)],]
mid5_adj = update(mid5, data=gt_adj)
# mod.diag(fit1_adj, "Adjusted Fit1 data model")

gt_out = gt_mid[c(out_ind),]
print("potential outlier observations")
```

```
## [1] "potential outlier observations"
```

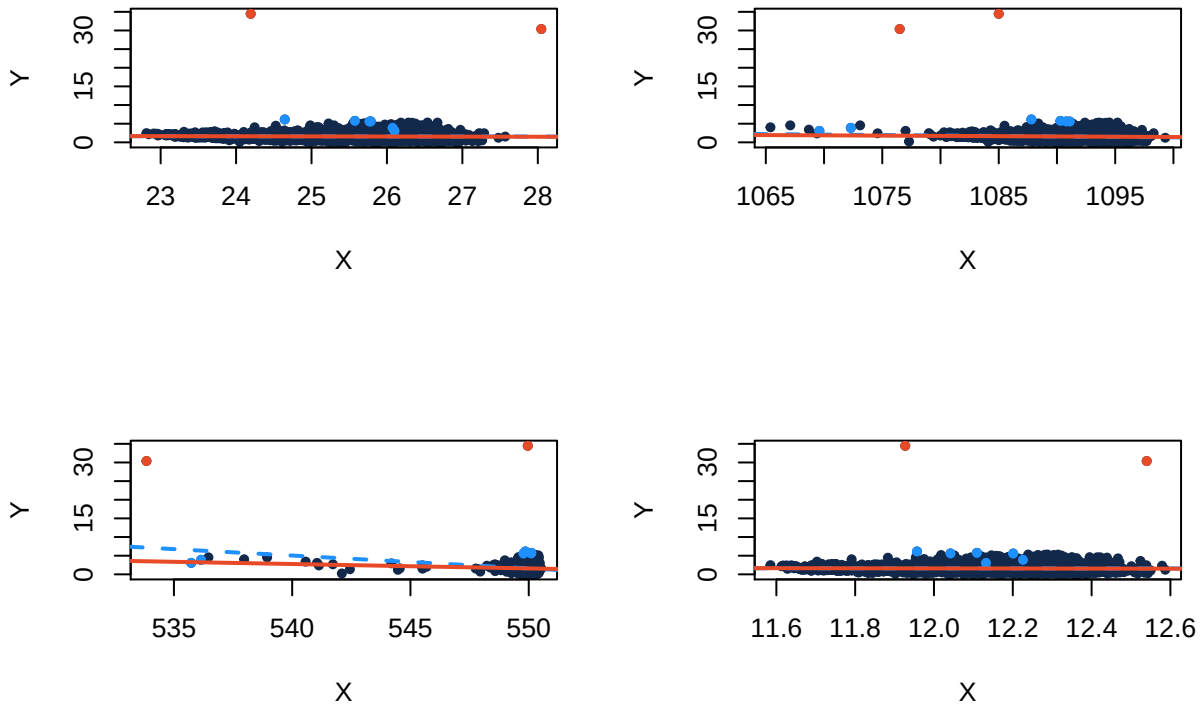
```
gt_out
```

```
## # A tibble: 8 x 10
##      AT    AP    AH  AFDP  GTEP   TIT   TAT   TEY   CDP    CO
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1  9.61 1010.  77.0  4.49  26.1 1072.  536.  136.  12.2  3.88
## 2  6.55 1008.  94.8  4.99  26.1 1070.  536.  135.  12.1  3.07
## 3 31.2  1009.  53.2  4.45  28.0 1076.  534.  134.  12.5 30.4
## 4 24.3  1011.  73.7  4.18  25.8 1091.  550.  134.  12.2  5.58
## 5 23.1  1011.  78.0  4.16  25.6 1090.  550.  134.  12.1  5.75
## 6 20.9  1010.  90.5  4.27  25.8 1091.  550.  135.  12.0  5.61
## 7 12.9  1017.  97.4  2.77  24.6 1088.  550.  134.  12.0  6.14
## 8 17.2  1014.  81.8  2.94  24.2 1085   550.  132.  11.9 34.5
```

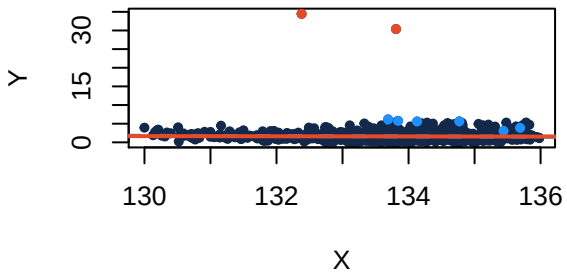
```
c = rep("dodgerblue", 8); c[c(3, 8)] = "#e84a27"

par(mfrow = c(2, 2))
abplot(gt_mid$AT, gt_mid$CO, gt_adj$AT, gt_adj$CO, gt_out$AT, gt_out$CO, c)
abplot(gt_mid$AP, gt_mid$CO, gt_adj$AP, gt_adj$CO, gt_out$AP, gt_out$CO, c)
abplot(gt_mid$AH, gt_mid$CO, gt_adj$AH, gt_adj$CO, gt_out$AH, gt_out$CO, c)
abplot(gt_mid$AFDP, gt_mid$CO, gt_adj$AFDP, gt_adj$CO, gt_out$AFDP, gt_out$CO, c)
```




```
abplot(gt_mid$GTEP, gt_mid$CO, gt_adj$GTEP, gt_adj$CO, gt_out$GTEP, gt_out$CO, c)
abplot(gt_mid$TIT, gt_mid$CO, gt_adj$TIT, gt_adj$CO, gt_out$TIT, gt_out$CO, c)
abplot(gt_mid$TAT, gt_mid$CO, gt_adj$TAT, gt_adj$CO, gt_out$TAT, gt_out$CO, c)
abplot(gt_mid$CDP, gt_mid$CO, gt_adj$CDP, gt_adj$CO, gt_out$CDP, gt_out$CO, c)
```




```
abplot(gt_mid$TEY, gt_mid$CO, gt_adj$TEY, gt_adj$CO, gt_out$TEY, gt_out$CO, c)
```




```
5/3097 < 0.01
```

```
## [1] TRUE
```

```
str(gt_mid)
```

```
## tibble [3,907 x 10] (S3: tbl_df/tbl/data.frame)
##  $ AT  : num [1:3907] 8.08 8.3 8.47 8.89 9.37 ...
##  $ AP  : num [1:3907] 1015 1016 1016 1016 1017 ...
##  $ AH  : num [1:3907] 88.6 86.3 86.5 83 80 ...
##  $ AFDP: num [1:3907] 4.06 4.09 4.05 4.05 4.04 ...
##  $ GTEP: num [1:3907] 23.4 23.7 23.7 23.9 23.9 ...
##  $ TIT : num [1:3907] 1083 1085 1085 1086 1086 ...
##  $ TAT : num [1:3907] 550 550 550 550 550 ...
##  $ TEY : num [1:3907] 132 134 134 135 135 ...
##  $ CDP : num [1:3907] 11.7 11.7 11.8 11.9 11.9 ...
##  $ CO  : num [1:3907] 1.91 1.71 1.47 1.71 1.66 ...
```

```
str(gt_adj)
```

```
## tibble [3,905 x 10] (S3: tbl_df/tbl/data.frame)
##  $ AT  : num [1:3905] 8.08 8.3 8.47 8.89 9.37 ...
##  $ AP  : num [1:3905] 1015 1016 1016 1016 1017 ...
##  $ AH  : num [1:3905] 88.6 86.3 86.5 83 80 ...
```

```
##  $ AFDP: num [1:3905] 4.06 4.09 4.05 4.05 4.04 ...
##  $ GTEP: num [1:3905] 23.4 23.7 23.7 23.9 23.9 ...
##  $ TIT : num [1:3905] 1083 1085 1085 1086 1086 ...
##  $ TAT : num [1:3905] 550 550 550 550 550 ...
##  $ TEY : num [1:3905] 132 134 134 135 135 ...
##  $ CDP : num [1:3905] 11.7 11.7 11.8 11.9 11.9 ...
##  $ CO  : num [1:3905] 1.91 1.71 1.47 1.71 1.66 ...
```

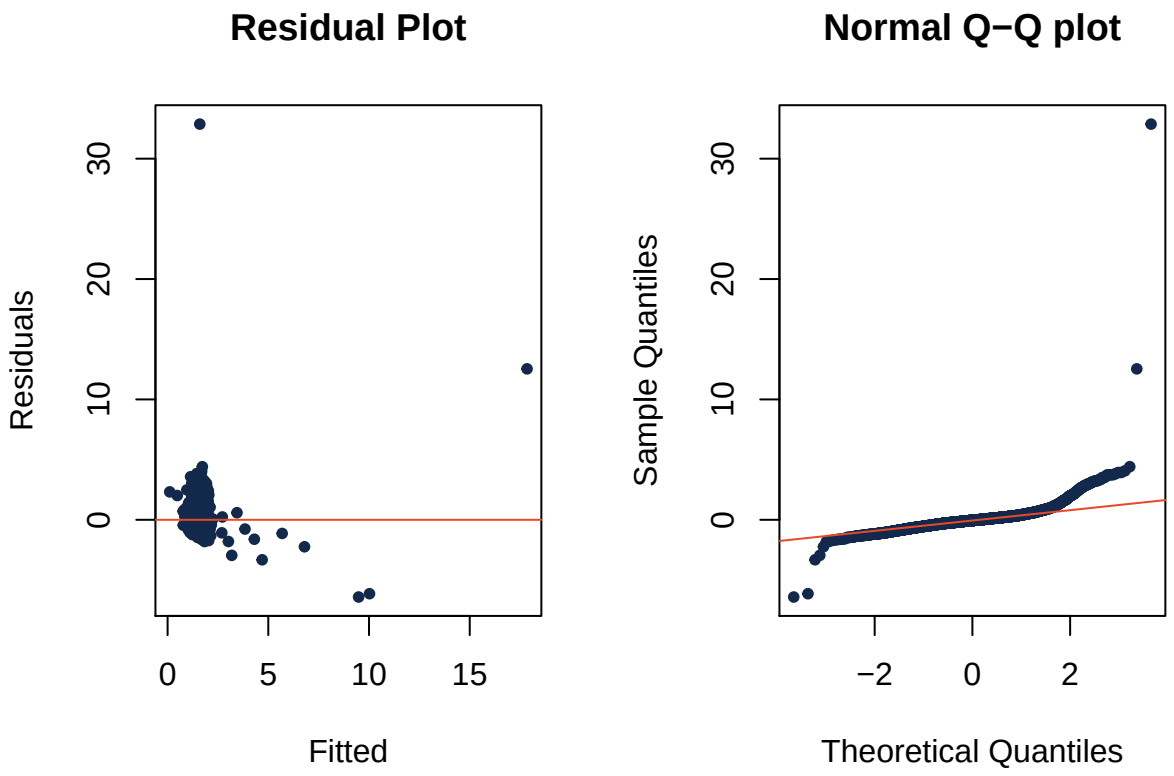
```
#BEST SO FAR
mid5_no <- lm(CO ~ AT + AP + AH + poly(AFDP,2) + poly(TIT,2) + poly(TAT,2) + poly(CDP,2),data = gt_adj)
summary(mid5_no) #ambient variable may not have strong influence over CO emission
```

```
##
## Call:
## lm(formula = CO ~ AT + AP + AH + poly(AFDP, 2) + poly(TIT, 2) +
##     poly(TAT, 2) + poly(CDP, 2), data = gt_adj)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.4002 -0.3293 -0.0428  0.2158  4.4971
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -6.02304    2.26703  -2.657 0.007921 **
## AT              0.01803    0.00412   4.375 1.25e-05 ***
## AP              0.00631    0.00218   2.894 0.003825 **
## AH              0.01109    0.00101  10.986  < 2e-16 ***
## poly(AFDP, 2)1 -1.04743    0.73826  -1.419 0.156048
## poly(AFDP, 2)2 -3.91324    0.75091  -5.211 1.97e-07 ***
## poly(TIT, 2)1  -8.69058    2.58066  -3.368 0.000766 ***
## poly(TIT, 2)2   0.66777    1.39571   0.478 0.632363
## poly(TAT, 2)1  -2.66268    1.27380  -2.090 0.036651 *
## poly(TAT, 2)2   2.05576    0.79368   2.590 0.009629 **
## poly(CDP, 2)1   3.92933    1.92512   2.041 0.041309 *
## poly(CDP, 2)2   1.62946    1.01950   1.598 0.110059
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6691 on 3893 degrees of freedom
## Multiple R-squared:  0.05644,    Adjusted R-squared:  0.05378
## F-statistic: 21.17 on 11 and 3893 DF,  p-value: < 2.2e-16
```

```
vif(mid5_no)
```

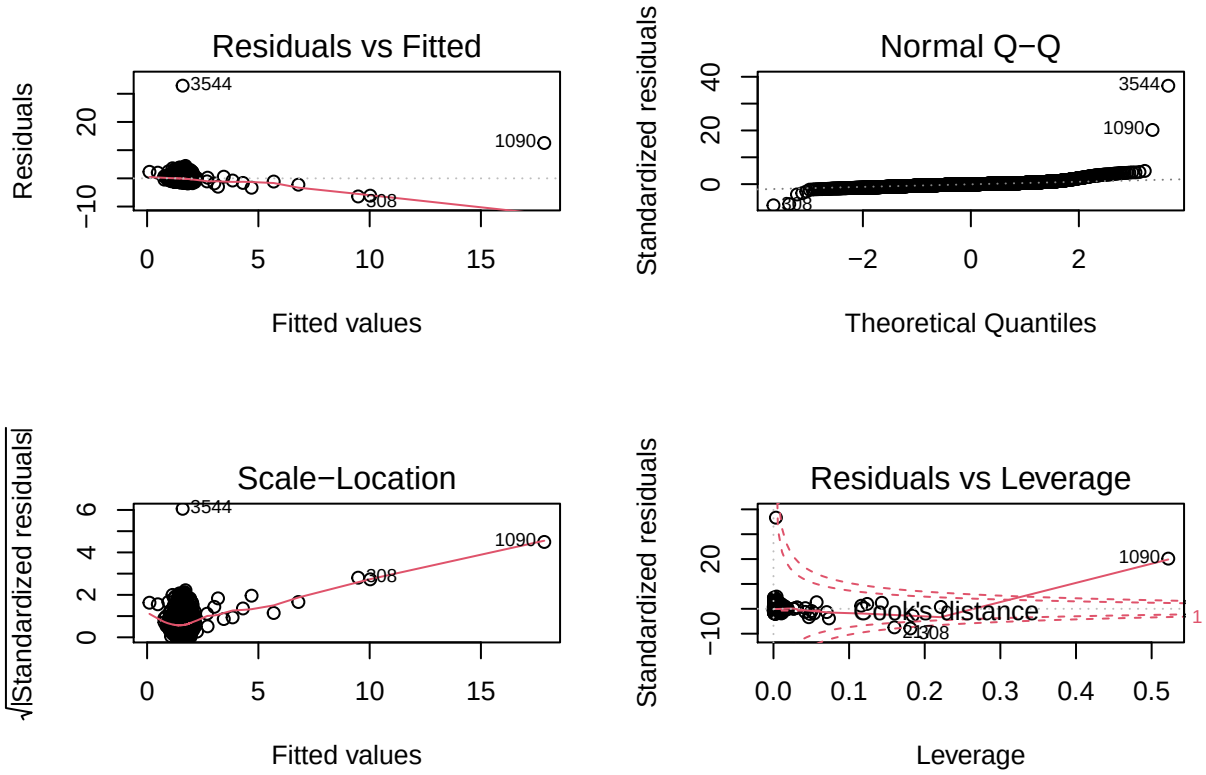
```
##             AT             AP             AH poly(AFDP, 2)1 poly(AFDP, 2)2
##       8.158075       1.391146       1.871009       1.217382       1.259447
##  poly(TIT, 2)1  poly(TIT, 2)2  poly(TAT, 2)1  poly(TAT, 2)2  poly(CDP, 2)1
##      14.875247       4.351048       3.624128       1.406993       8.277827
##  poly(CDP, 2)2
##       2.321540
```

```
run_all_diagnostic(mid5)
```

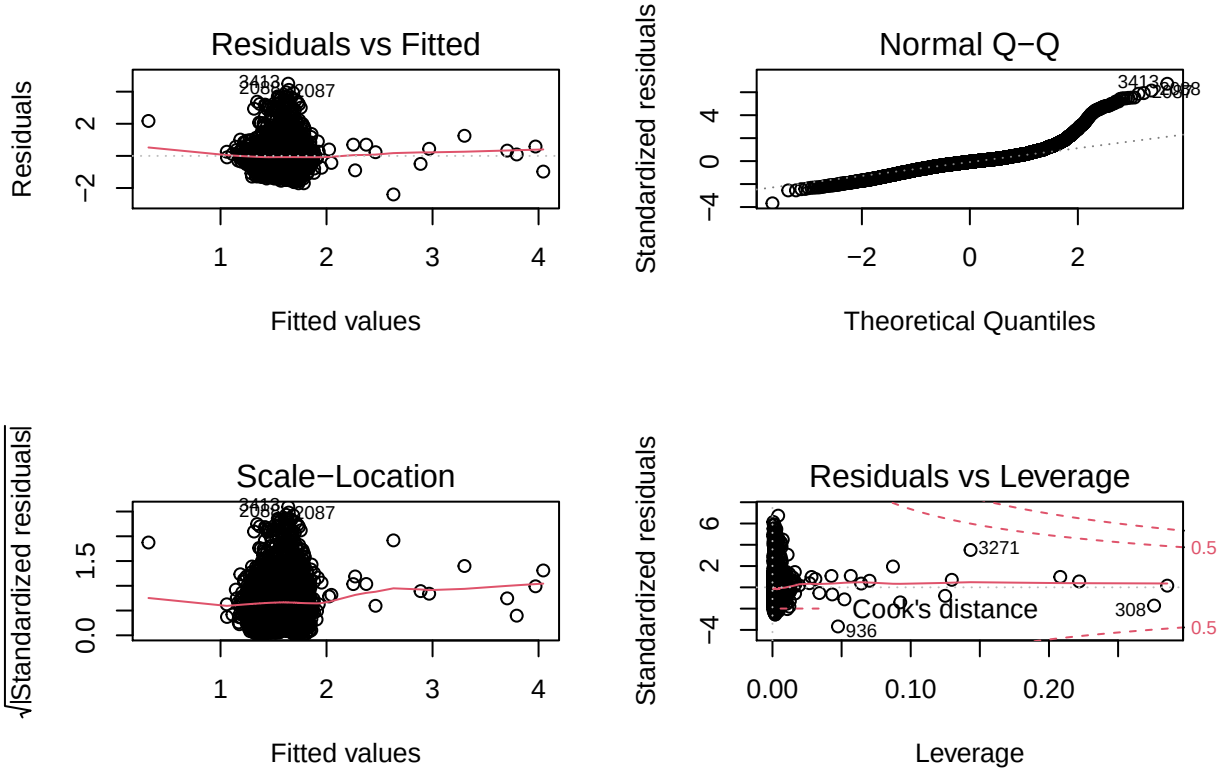



```
##  loocv_rmse adj.r.squared params     bp.p_val bp.decision     sw.p_val
##   0.9778249     0.1607462     12 3.741635e-10      Reject 2.482981e-70
##  sw.decision
##       Reject
```

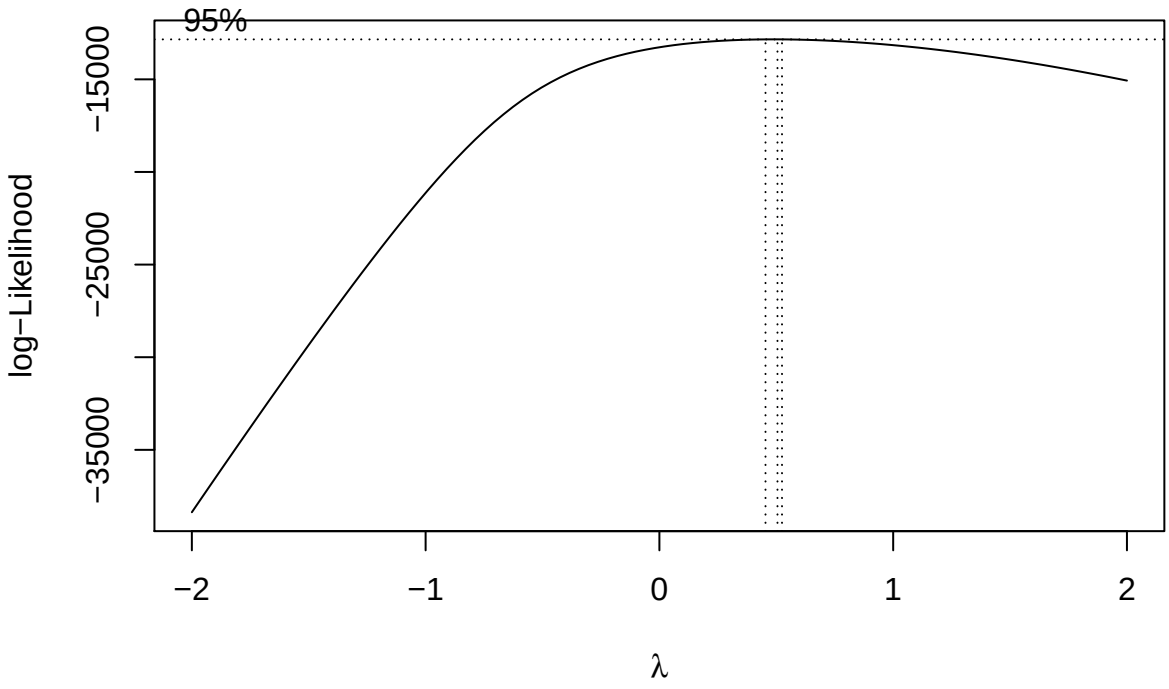
```
par(mfrow=c(2,2))
plot(mid5)
```




```
par(mfrow=c(2,2))
plot(mid5_no)
```




```
boxcox(mid5_no)
```




```
mid5_no <- lm(sqrt(CO)  ~ AT + AP + AH + poly(AFDP,2) + poly(TIT,2) + poly(TAT,2) + poly(CDP,2),data = g
summary(mid5_no) #ambient variable may not have strong influence over CO emission
```

```
##
## Call:
## lm(formula = sqrt(CO) ~ AT + AP + AH + poly(AFDP, 2) + poly(TIT,
##     2) + poly(TAT, 2) + poly(CDP, 2), data = gt_adj)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.13883 -0.11657  0.00603  0.10827  1.22018
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -2.0915389  0.8724704  -2.397 0.016565 *
## AT              0.0079385  0.0015856   5.007 5.79e-07 ***
## AP              0.0028022  0.0008391   3.339 0.000847 ***
## AH              0.0042125  0.0003887  10.838  < 2e-16 ***
## poly(AFDP, 2)1 -0.3334521  0.2841219  -1.174 0.240618
## poly(AFDP, 2)2 -1.5617050  0.2889890  -5.404 6.91e-08 ***
## poly(TIT, 2)1  -4.8724758  0.9931696  -4.906 9.68e-07 ***
## poly(TIT, 2)2  -0.2690571  0.5371410  -0.501 0.616466
## poly(TAT, 2)1  -0.6957421  0.4902224  -1.419 0.155910
## poly(TAT, 2)2   0.6359604  0.3054480   2.082 0.037402 *
## poly(CDP, 2)1   1.5562715  0.7408828   2.101 0.035743 *
```

```
## poly(CDP, 2)2   0.5656108  0.3923552   1.442 0.149502
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2575 on 3893 degrees of freedom
## Multiple R-squared:  0.06246,    Adjusted R-squared:  0.05981
## F-statistic: 23.58 on 11 and 3893 DF,  p-value: < 2.2e-16
```

```
vif(mid5_no)
```

```
##             AT             AP             AH poly(AFDP, 2)1 poly(AFDP, 2)2
##       8.158075       1.391146       1.871009       1.217382       1.259447
##  poly(TIT, 2)1  poly(TIT, 2)2  poly(TAT, 2)1  poly(TAT, 2)2  poly(CDP, 2)1
##      14.875247       4.351048       3.624128       1.406993       8.277827
##  poly(CDP, 2)2
##       2.321540
```

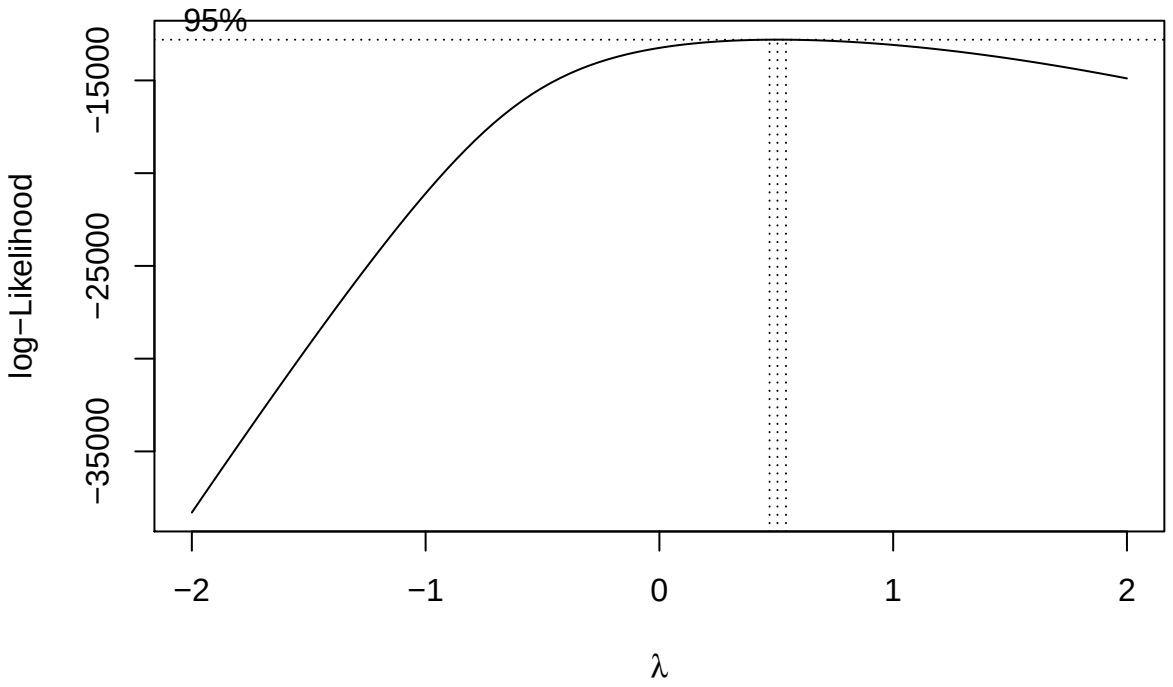
```
gt_adj_all = gt_mid[-c(out_ind),]
mod1 <- lm(CO~., data = gt_adj_all)
mid1_aic_back = step(mod1, trace = FALSE)
summary(mid1_aic_back)
```

```
##
## Call:
## lm(formula = CO ~ AT + AP + AH + AFDP + TIT + TAT + CDP, data = gt_adj_all)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.5025 -0.3467 -0.0419  0.2219  3.7272
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 69.7824393 10.8234698   6.447 1.28e-10 ***
## AT           0.0172299  0.0039018   4.416 1.03e-05 ***
## AP           0.0060668  0.0021027   2.885 0.003932 **
## AH           0.0098082  0.0009469  10.358  < 2e-16 ***
## AFDP        -0.0443998  0.0184711  -2.404 0.016274 *
## TIT         -0.0381825  0.0107751  -3.544 0.000399 ***
## TAT         -0.0720705  0.0227385  -3.170 0.001539 **
## CDP          0.4953731  0.1715593   2.887 0.003905 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6598 on 3891 degrees of freedom
## Multiple R-squared:  0.04269,    Adjusted R-squared:  0.04097
## F-statistic: 24.79 on 7 and 3891 DF,  p-value: < 2.2e-16
```

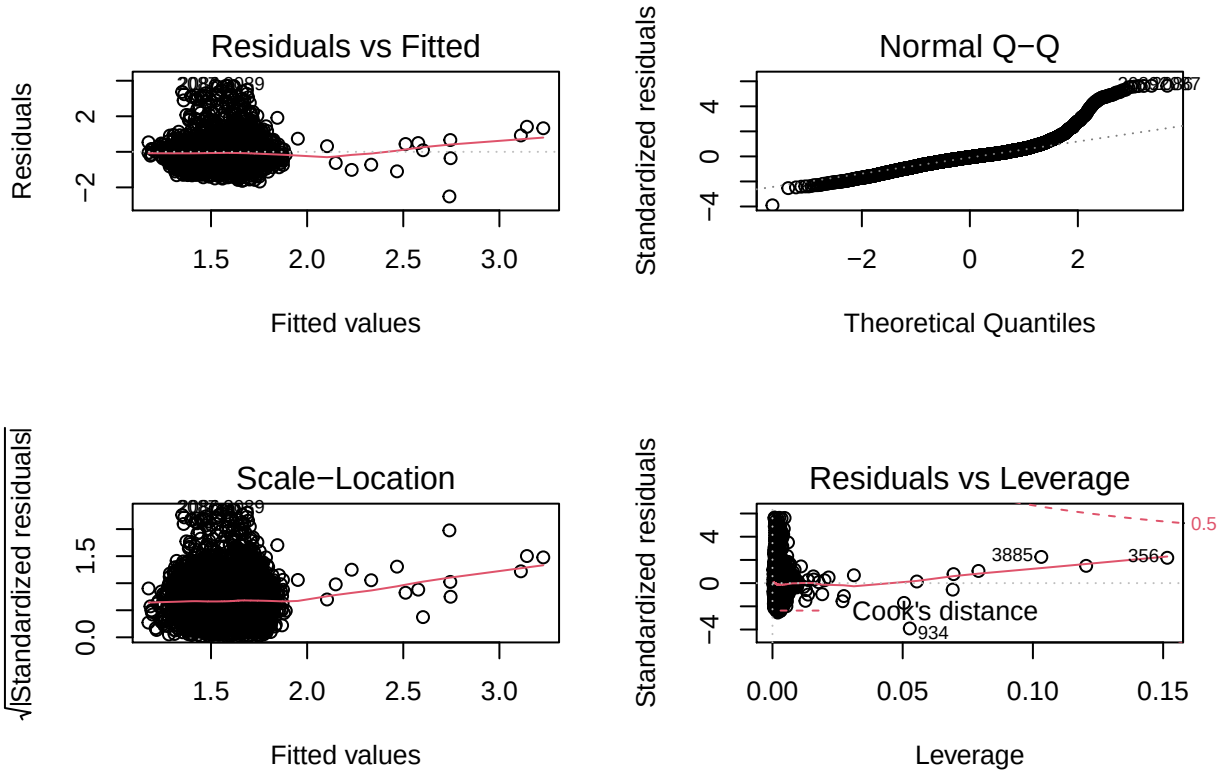
```
vif(mid1_aic_back)
```

```
##        AT        AP        AH      AFDP       TIT       TAT       CDP
##  7.510763  1.330007  1.689710  1.131227 13.657424  1.495483  8.050614
```

```
boxcox(mid1_aic_back)
```




```
par(mfrow=c(2,2))
plot(mod1)
```




```
midbest1 <- lm(CO ~ AT + AP + AH + poly(AFDP,2) + poly(TIT,2) + poly(TAT,2) + poly(CDP,2),data = gt_mid
summary(midbest1) #ambient variable may not have strong influence over CO emission
```

```
##
## Call:
## lm(formula = CO ~ AT + AP + AH + poly(AFDP, 2) + poly(TIT, 2) +
##     poly(TAT, 2) + poly(CDP, 2), data = gt_mid)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.416 -0.351 -0.056  0.229 32.869
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -6.070530   3.042977  -1.995 0.046120 *
## AT               0.019646   0.005527   3.555 0.000383 ***
## AP               0.006471   0.002927   2.211 0.027091 *
## AH               0.009372   0.001354   6.924 5.11e-12 ***
## poly(AFDP, 2)1  -3.194292   0.990177  -3.226 0.001266 **
## poly(AFDP, 2)2  -2.977301   1.008120  -2.953 0.003163 **
## poly(TIT, 2)1   -3.934220   3.440565  -1.143 0.252909
## poly(TIT, 2)2  -18.349490   1.603611 -11.443  < 2e-16 ***
## poly(TAT, 2)1  -24.983758   1.570303 -15.910  < 2e-16 ***
## poly(TAT, 2)2   16.797544   0.950353  17.675  < 2e-16 ***
## poly(CDP, 2)1    2.510030   2.581421   0.972 0.330940
```

```
## poly(CDP, 2)2   12.503072   1.246289  10.032  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8984 on 3895 degrees of freedom
## Multiple R-squared:  0.1631, Adjusted R-squared:  0.1607
## F-statistic: 69.01 on 11 and 3895 DF,  p-value: < 2.2e-16
```

```
vif(midbest1)
```

```
##             AT             AP             AH poly(AFDP, 2)1 poly(AFDP, 2)2
##       8.147728       1.390616       1.865517       1.214714       1.259137
##  poly(TIT, 2)1  poly(TIT, 2)2  poly(TAT, 2)1  poly(TAT, 2)2  poly(CDP, 2)1
##      14.665877       3.186006       3.055030       1.118969       8.255930
##  poly(CDP, 2)2
##       1.924358
```

```
midbest2 <-lm(CO ~. -AT-CDP-GTEP, data = gt_mid)
summary(midbest2)
```

```
##
## Call:
## lm(formula = CO ~ . - AT - CDP - GTEP, data = gt_mid)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -4.338 -0.363 -0.060  0.227 32.707
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 187.871988  12.429130  15.115  < 2e-16 ***
## AP            0.001204   0.002659   0.453 0.650837
## AH            0.009689   0.001361   7.117 1.30e-12 ***
## AFDP         -0.098367   0.026449  -3.719 0.000203 ***
## TIT           0.024247   0.005984   4.052 5.18e-05 ***
## TAT          -0.374532   0.022900 -16.355  < 2e-16 ***
## TEY          -0.061687   0.019395  -3.181 0.001481 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9433 on 3900 degrees of freedom
## Multiple R-squared:  0.07617,    Adjusted R-squared:  0.07475
## F-statistic: 53.59 on 6 and 3900 DF,  p-value: < 2.2e-16
```

```
vif(midbest2)
```

```
##       AP       AH     AFDP      TIT      TAT      TEY
## 1.041113 1.711583 1.138341 2.096469 1.127478 1.157575
```

```
# Removing influential observations
cd = cooks.distance(midbest2)
length(cd[which(cd > 4 / length(cd))])
```

```
## [1] 84
```

```
midbest1a = lm(CO ~ AT + AP + AH + poly(AFDP,2) + poly(TIT,2) + poly(TAT,2) + poly(CDP,2), data = gt_mi
summary(midbest1a)
```

```
##
## Call:
## lm(formula = CO ~ AT + AP + AH + poly(AFDP, 2) + poly(TIT, 2) +
##     poly(TAT, 2) + poly(CDP, 2), data = gt_mid, subset = cd <=
##     4/length(cd))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.7265 -0.2940 -0.0134  0.2385  4.1442
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -8.788e+00  1.900e+00  -4.625 3.86e-06 ***
## AT              2.313e-02  3.572e-03   6.474 1.07e-10 ***
## AP              8.995e-03  1.824e-03   4.931 8.53e-07 ***
## AH              1.030e-02  8.477e-04  12.152  < 2e-16 ***
## poly(AFDP, 2)1 -1.166e+00  6.227e-01  -1.872   0.0613 .
## poly(AFDP, 2)2 -4.929e+00  6.721e-01  -7.333 2.74e-13 ***
## poly(TIT, 2)1  -1.209e+01  2.301e+00  -5.253 1.58e-07 ***
## poly(TIT, 2)2   2.649e-01  1.640e+00   0.162   0.8717
## poly(TAT, 2)1  -4.557e+01  2.577e+01  -1.769   0.0771 .
## poly(TAT, 2)2   2.142e+01  1.232e+01   1.738   0.0822 .
## poly(CDP, 2)1   1.514e+00  1.626e+00   0.931   0.3519
## poly(CDP, 2)2   1.373e-01  1.017e+00   0.135   0.8926
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5522 on 3811 degrees of freedom
## Multiple R-squared:  0.07427,    Adjusted R-squared:  0.0716
## F-statistic: 27.79 on 11 and 3811 DF,  p-value: < 2.2e-16
```

```
vif(midbest1a)
```

```
##             AT             AP             AH poly(AFDP, 2)1 poly(AFDP, 2)2
##       8.846399       1.413827       1.894025       1.246335       1.296359
##  poly(TIT, 2)1  poly(TIT, 2)2  poly(TAT, 2)1  poly(TAT, 2)2  poly(CDP, 2)1
##      15.732382       3.775967     244.596359     244.335087       8.382152
##  poly(CDP, 2)2
##       3.224416
```

```
# Percentage of influential points in mid-range data
length(cd[which(cd > 4 / length(cd))])/nobs(midbest2)*100
```

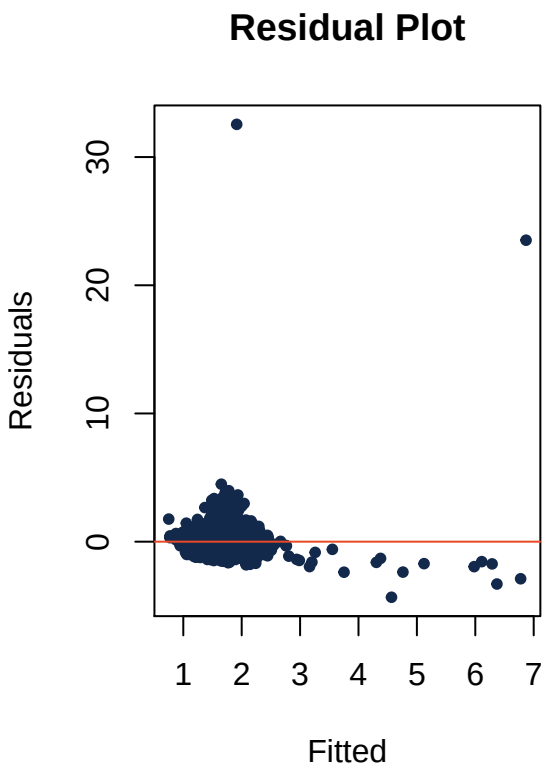
```
## [1] 2.149987
```

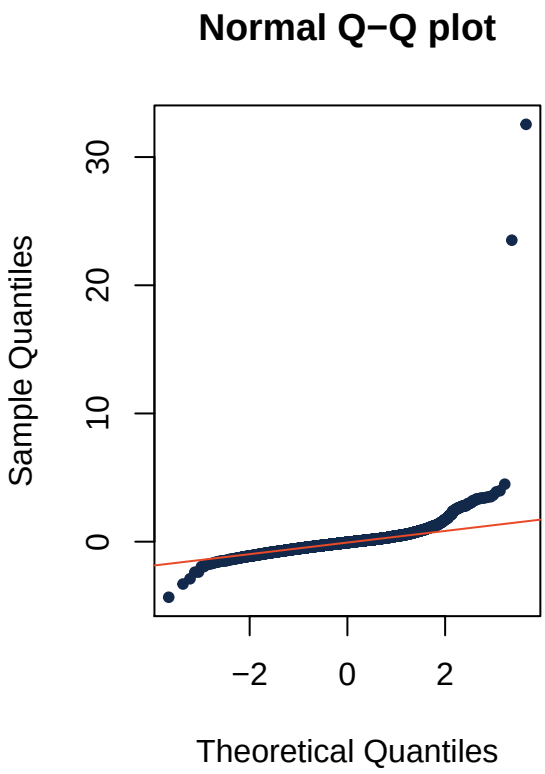
```
full = lm(CO ~ ., data = gt_mid)
fit1 = update(full, .~.-GTEP-AT-TIT)
j_model = lm(CO ~ poly(AP,2)+poly(AH,2) + poly(AFDP,2)+poly(TAT,2)+poly(TIT,2) + poly(TEY,2), data= gt_a
summary(j_model)
```

```
##
## Call:
## lm(formula = CO ~ poly(AP, 2) + poly(AH, 2) + poly(AFDP, 2) +
##     poly(TAT, 2) + poly(TIT, 2) + poly(TEY, 2), data = gt_adj)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.4909 -0.3392 -0.0431  0.2177  4.4380
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)     1.58721    0.01068 148.590  < 2e-16 ***
## poly(AP, 2)1    0.34201    0.68941   0.496 0.619860
## poly(AP, 2)2   -3.32455    0.74095  -4.487 7.44e-06 ***
## poly(AH, 2)1    9.00777    0.91989   9.792  < 2e-16 ***
## poly(AH, 2)2   -1.20397    0.71305  -1.688 0.091401 .
## poly(AFDP, 2)1 -1.58419    0.73751  -2.148 0.031773 *
## poly(AFDP, 2)2 -3.40517    0.75873  -4.488 7.40e-06 ***
## poly(TAT, 2)1  -3.58763    0.98229  -3.652 0.000263 ***
## poly(TAT, 2)2   2.13785    0.71937   2.972 0.002978 **
## poly(TIT, 2)1   1.58530    1.11984   1.416 0.156957
## poly(TIT, 2)2   1.75283    1.03578   1.692 0.090674 .
## poly(TEY, 2)1  -2.27070    0.72794  -3.119 0.001826 **
## poly(TEY, 2)2   1.90939    0.69552   2.745 0.006074 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6675 on 3892 degrees of freedom
## Multiple R-squared:  0.0612, Adjusted R-squared:  0.0583
## F-statistic: 21.14 on 12 and 3892 DF,  p-value: < 2.2e-16
```

Model 2 Building

```
gt_adj_all = gt_mid_all[-c(out_ind)[c(3, 8)],]

full1 = lm(CO ~ ., data = gt_mid_all)

fit1 = update(full1, .~.-GTEP-AT-TIT)
run_all_diagnostic(fit1)
```






```
##  loocv_rmse adj.r.squared params     bp.p_val bp.decision     sw.p_val
##   0.9500886     0.1163982      8 5.595682e-27      Reject 5.489629e-73
##  sw.decision
##       Reject
```

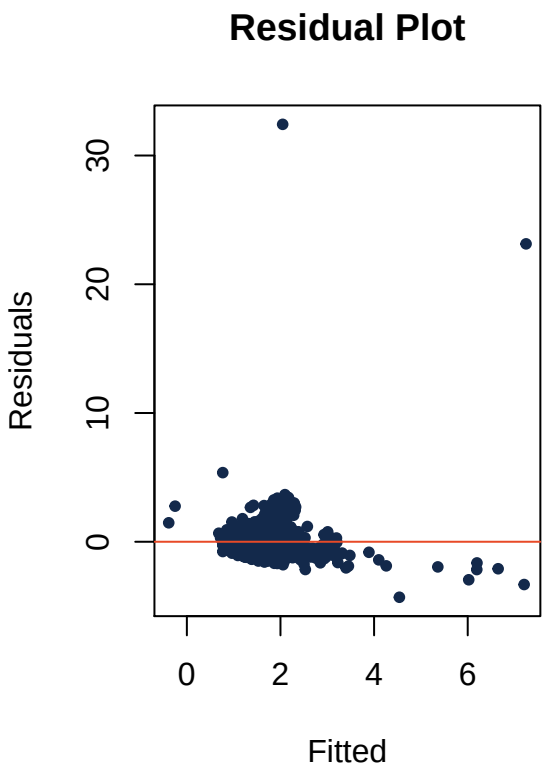
```
v = vif(fit1)
v[which(v > 5)]
```

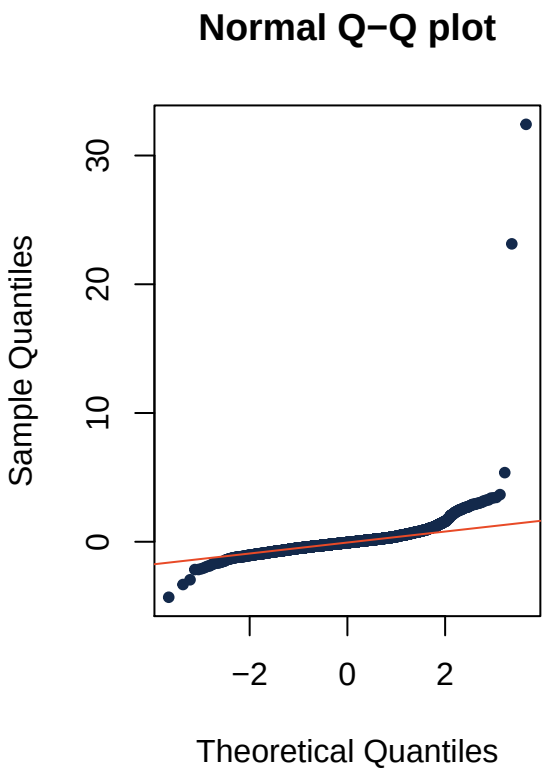
```
## named numeric(0)
```

```
gt_bic_sel(gt_mid_all, full1)
```

```
## (Intercept)          AT          AP          AH        AFDP        GTEP
##        TRUE        TRUE        TRUE        TRUE        TRUE        TRUE
##         TIT         TAT         TEY         CDP         NOX
##        TRUE        TRUE        TRUE       FALSE        TRUE
```

```
bic_mod1 = update(full1, .~.-CDP)
run_all_diagnostic(bic_mod1)
```



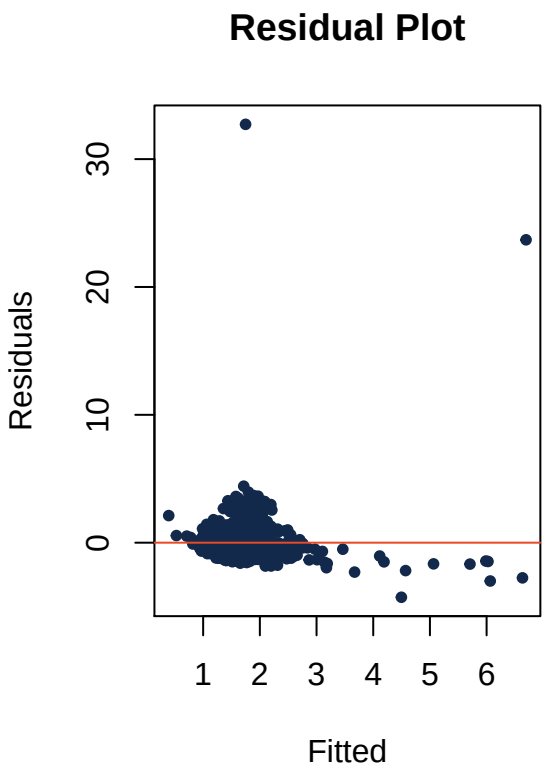


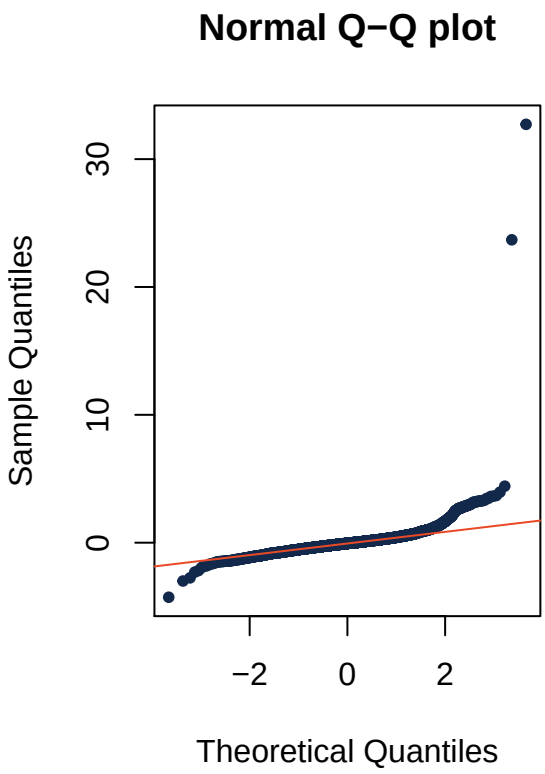

```
##  loocv_rmse adj.r.squared params     bp.p_val bp.decision     sw.p_val
##   0.9230876     0.1687327     10 6.637984e-26      Reject 9.231982e-74
##  sw.decision
##       Reject
```

```
v = vif(bic_mod1)
v[which(v > 5)]
```

```
##       AT     GTEP      TIT
## 52.58734 54.81048 37.36301
```

```
bic_mod2 = update(bic_mod1, .~.-GTEP-AT) # Leaving TIT in
run_all_diagnostic(bic_mod2)
```



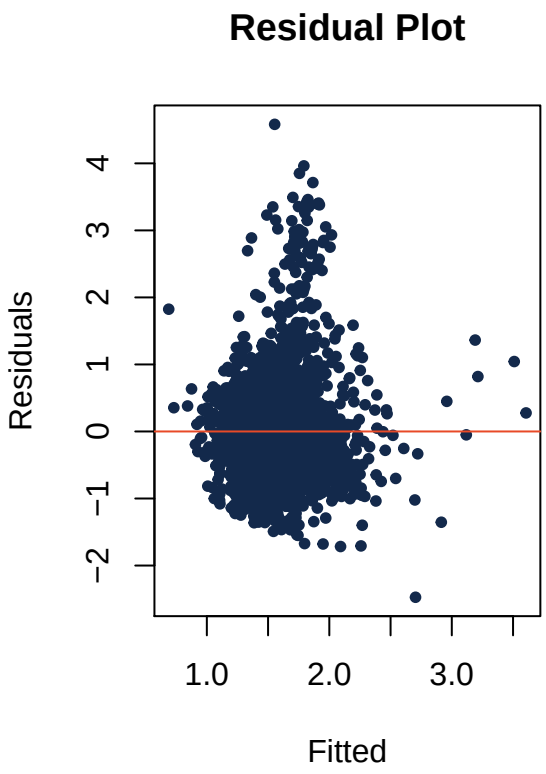


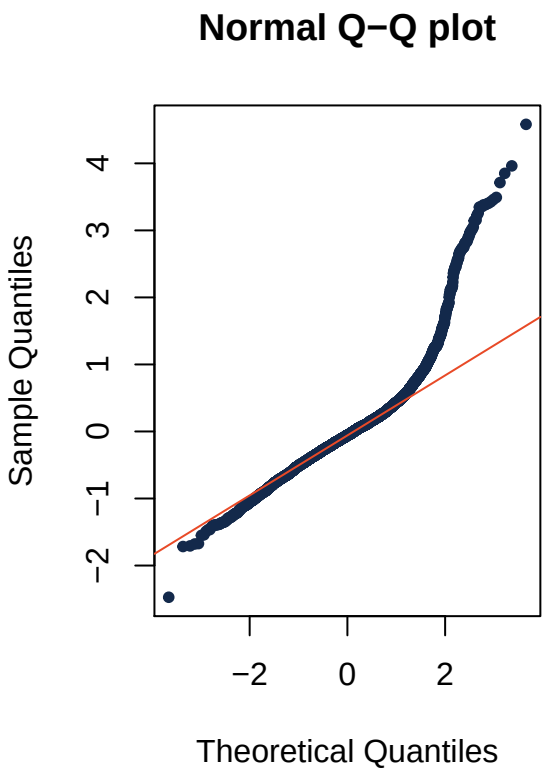

```
##  loocv_rmse adj.r.squared params     bp.p_val bp.decision     sw.p_val
##   0.9479408     0.1210341      8 4.157756e-26      Reject 3.285279e-73
##  sw.decision
##       Reject
```

```
v = vif(bic_mod2)
v[which(v > 5)]
```

```
##      TIT
## 5.490481
```

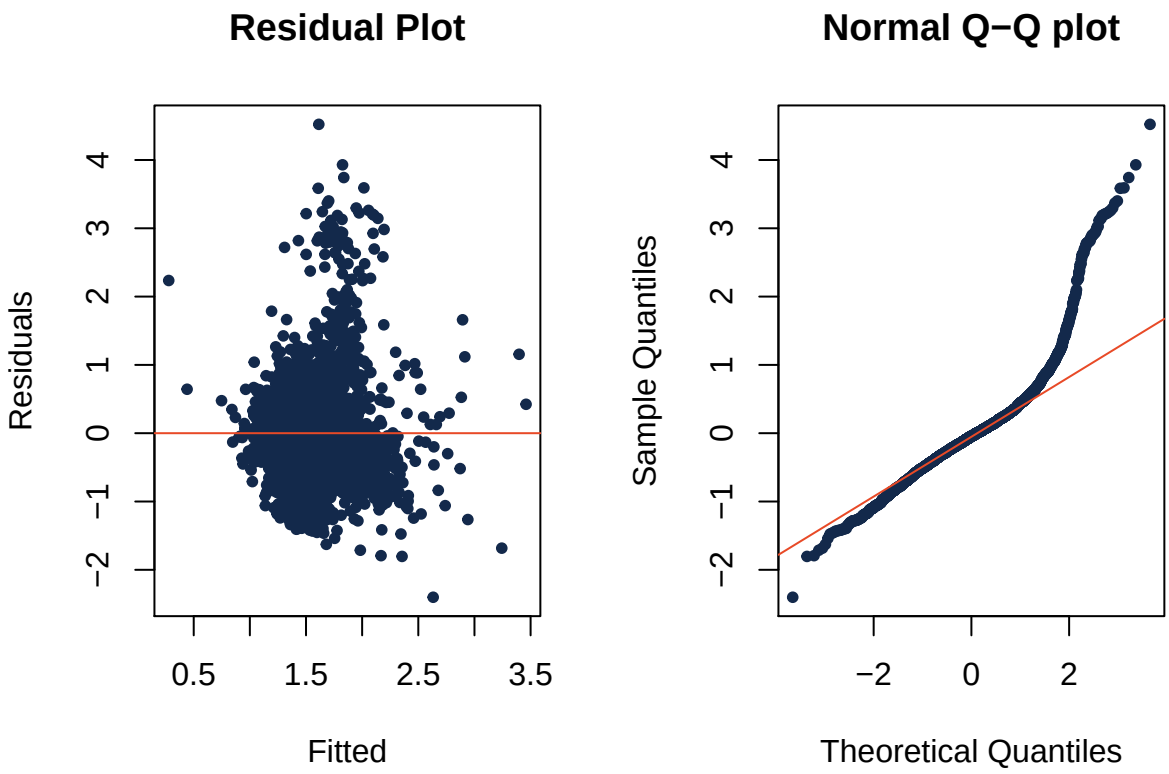
```
fit_adj = update(fit1, data=gt_adj_all)
run_all_diagnostic(fit_adj)
```





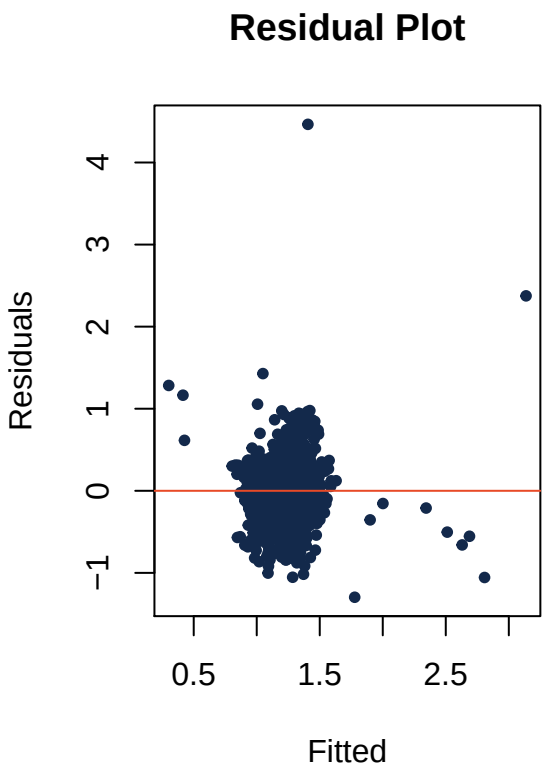

```
##  loocv_rmse adj.r.squared params     bp.p_val bp.decision     sw.p_val
##   0.6433733     0.1272433      8 8.176951e-40      Reject 2.119163e-47
##  sw.decision
##       Reject
```

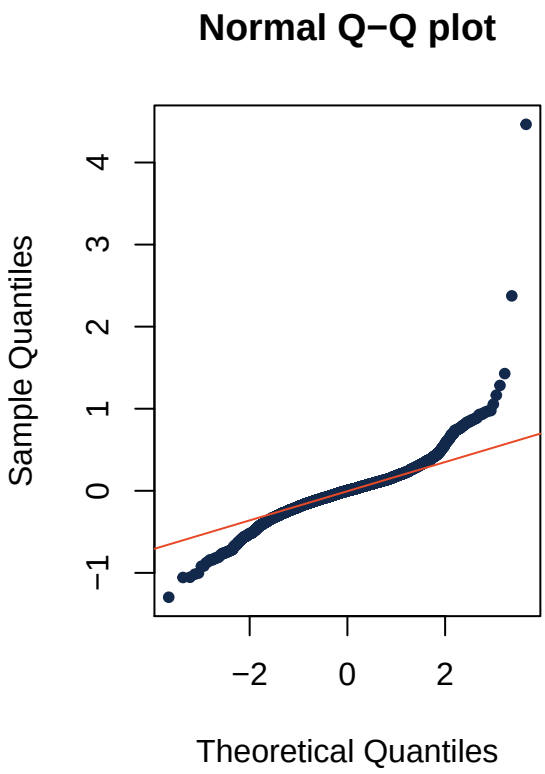
```
bic_adj = update(bic_mod2, data=gt_adj_all)
run_all_diagnostic(bic_adj)
```




```
##  loocv_rmse adj.r.squared params     bp.p_val bp.decision    sw.p_val
##   0.6359884     0.1476088      8 1.827801e-49      Reject 1.63508e-46
##  sw.decision
##       Reject
```

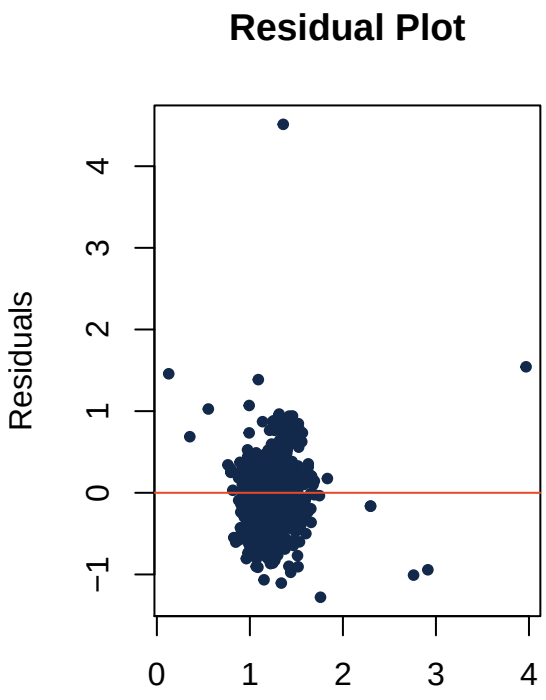
```
fit_refit = lm(sqrt(CO) ~ poly(AP,2)+poly(AH,2) + poly(AFDP,2)+poly(TAT,2)+poly(CDP,2) + poly(TEY,2) + p
run_all_diagnostic(fit_refit)
```

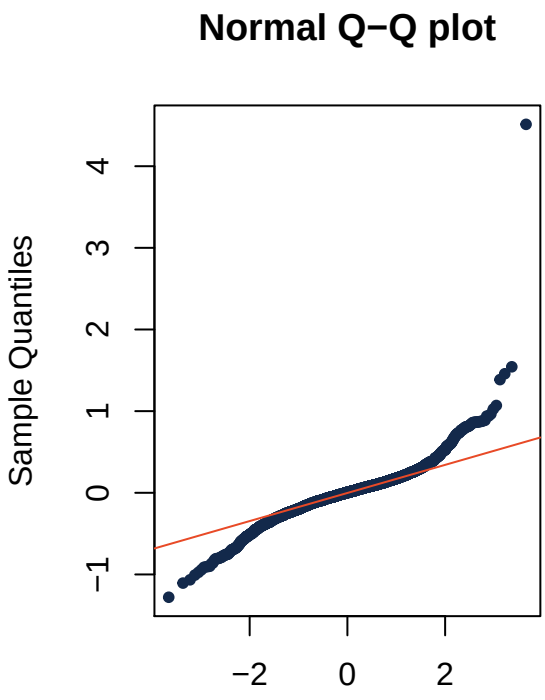





```
##  loocv_rmse adj.r.squared params     bp.p_val bp.decision     sw.p_val
##   0.2579594     0.2067914     15 1.167661e-27      Reject 9.746608e-46
##  sw.decision
##       Reject
```

```
bic_refit = lm(sqrt(CO) ~ poly(AP,2)+poly(AH,2) + poly(AFDP,2)+poly(TAT,2)+poly(TIT,2) + poly(TEY,2) + p
run_all_diagnostic(bic_refit)
```






```
##  loocv_rmse adj.r.squared params     bp.p_val bp.decision     sw.p_val
##   0.2510173     0.2424045     15 1.058968e-10      Reject 2.433198e-45
##  sw.decision
##       Reject
```

```
# Include these models for comparisons with -NOX model (Could be useful for client to see).
# Highest adj. R squared models for mid-range data.
j_mod = lm(sqrt(CO) ~ poly(AP,2)+poly(AT,2)+poly(AH,2) + poly(AFDP,2)+poly(TAT,2)+poly(TIT,2) + poly(NO
summary(j_mod)
```

```
##
## Call:
## lm(formula = sqrt(CO) ~ poly(AP, 2) + poly(AT, 2) + poly(AH,
##     2) + poly(AFDP, 2) + poly(TAT, 2) + poly(TIT, 2) + poly(NOX,
##     2), data = gt_mid_all)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.2781 -0.1187 -0.0004  0.1107  4.4755
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     1.233828   0.003852 320.276  < 2e-16 ***
## poly(AP, 2)1    2.572193   0.292404   8.797  < 2e-16 ***
## poly(AP, 2)2   -1.238311   0.267165  -4.635 3.69e-06 ***
```

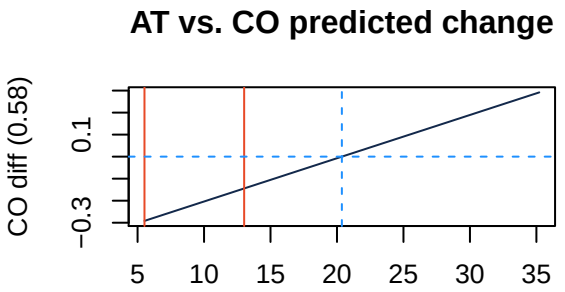
```
## poly(AT, 2)1   21.966603   0.945212  23.240  < 2e-16 ***
## poly(AT, 2)2    1.157332   0.378977   3.054  0.00227 **
## poly(AH, 2)1   13.891484   0.480146  28.932  < 2e-16 ***
## poly(AH, 2)2   -1.269242   0.267685  -4.742 2.20e-06 ***
## poly(AFDP, 2)1 -3.095761   0.282293 -10.966  < 2e-16 ***
## poly(AFDP, 2)2 -1.686331   0.278209  -6.061 1.48e-09 ***
## poly(TAT, 2)1  -4.003558   0.380485 -10.522  < 2e-16 ***
## poly(TAT, 2)2   2.487767   0.250173   9.944  < 2e-16 ***
## poly(TIT, 2)1  -5.307118   0.690971  -7.681 1.99e-14 ***
## poly(TIT, 2)2  -3.200082   0.408515  -7.833 6.08e-15 ***
## poly(NOX, 2)1  14.612148   0.497860  29.350  < 2e-16 ***
## poly(NOX, 2)2  -3.134226   0.288762 -10.854  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2408 on 3892 degrees of freedom
## Multiple R-squared:  0.284,  Adjusted R-squared:  0.2814
## F-statistic: 110.3 on 14 and 3892 DF,  p-value: < 2.2e-16
```

```
j_mod1 = lm(sqrt(CO) ~ AT + AP + AH + poly(AFDP,2) + poly(TIT,2) + poly(TAT,2) + poly(CDP,2) + poly(NOX
summary(j_mod1)
```

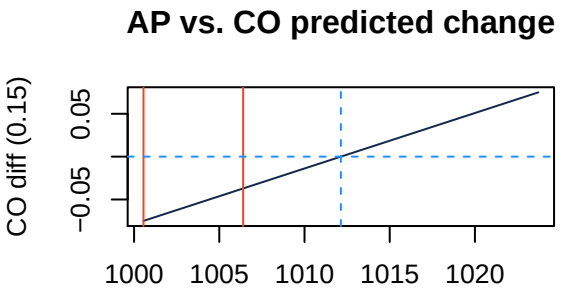
```
##
## Call:
## lm(formula = sqrt(CO) ~ AT + AP + AH + poly(AFDP, 2) + poly(TIT,
##     2) + poly(TAT, 2) + poly(CDP, 2) + poly(NOX, 2), data = gt_mid_all)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.2792 -0.1167  0.0004  0.1079  4.4980
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -9.2575556  0.8489140 -10.905  < 2e-16 ***
## AT              0.0479958  0.0020207  23.752  < 2e-16 ***
## AP              0.0082418  0.0008051  10.237  < 2e-16 ***
## AH              0.0151824  0.0005104  29.746  < 2e-16 ***
## poly(AFDP, 2)1 -3.0323078  0.2804180 -10.814  < 2e-16 ***
## poly(AFDP, 2)2 -2.0369282  0.2771670  -7.349 2.42e-13 ***
## poly(TIT, 2)1  -6.5071050  0.9278441  -7.013 2.74e-12 ***
## poly(TIT, 2)2  -4.0863497  0.4348726  -9.397  < 2e-16 ***
## poly(TAT, 2)1  -4.1766737  0.4223407  -9.889  < 2e-16 ***
## poly(TAT, 2)2   2.6208817  0.2554123  10.261  < 2e-16 ***
## poly(CDP, 2)1   1.4205410  0.6937495   2.048   0.0407 *
## poly(CDP, 2)2   1.5270130  0.3376730   4.522 6.30e-06 ***
## poly(NOX, 2)1  14.3949760  0.4919722  29.260  < 2e-16 ***
## poly(NOX, 2)2  -2.8121517  0.2584415 -10.881  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2414 on 3893 degrees of freedom
## Multiple R-squared:   0.28,  Adjusted R-squared:  0.2776
## F-statistic: 116.5 on 13 and 3893 DF,  p-value: < 2.2e-16
```

CO changes with one predictor changing with all things held constant (Model 1)

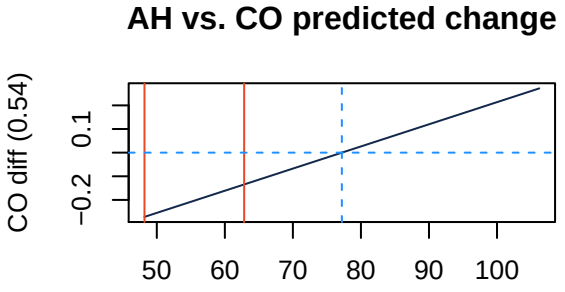
```
# midbest1
xvals = c(mean(gt_mid$AT), mean(gt_mid$AP), mean(gt_mid$AH), mean(gt_mid$AFDP), mean(gt_mid$TIT), mean(g
mbc = coef(midbest1)

par(mfrow=c(2,2))
pred.mb(xvals, mbc, gt_mid$AT, 1, "AT")
pred.mb(xvals, mbc, gt_mid$AP, 2, "AP")
pred.mb(xvals, mbc, gt_mid$AH, 3, "AH")

pred.mb(xvals, mbc, gt_mid$AFDP, 4, "AFDP")
```



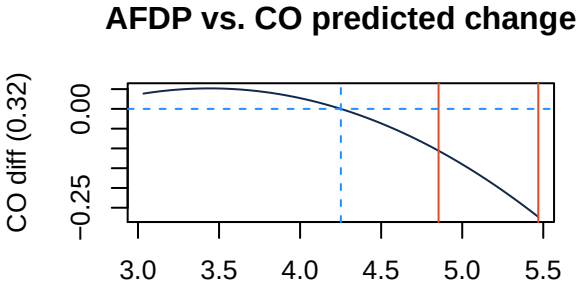
AT (CO Min at AT = 5.5)
CO diff Min = −0.29, Min 1se = −0.14



AP (CO Min at AP = 1000.5)
CO diff Min = −0.07, Min 1se = −0.04

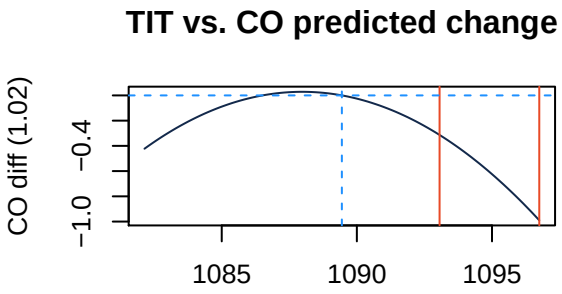


AH (CO Min at AH = 48.2)
CO diff Min = −0.27, Min 1se = −0.13

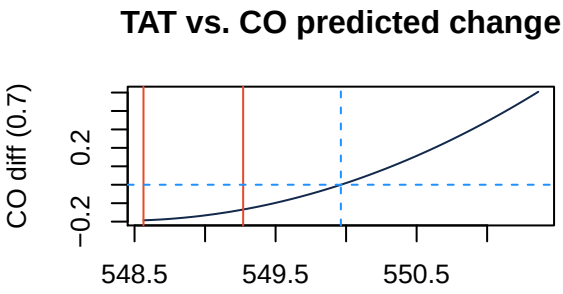


AFDP (CO Min at AFDP = 5.5)
CO diff Min = −0.27, Min 1se = −0.11

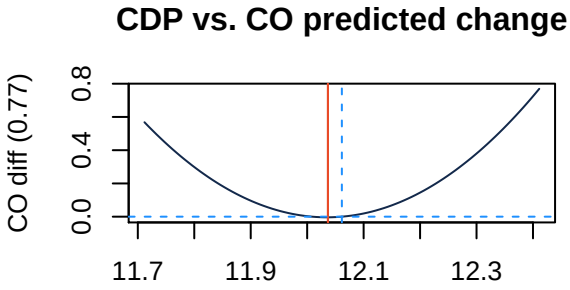
```
pred.mb(xvals, mbc, gt_mid$TIT, 5, "TIT")
pred.mb(xvals, mbc, gt_mid$TAT, 6, "TAT")
pred.mb(xvals, mbc, gt_mid$CDP, 7, "CDP")
```



TIT (CO Min at TIT = 1096.7)
CO diff Min = −0.99, Min 1se = −0.31



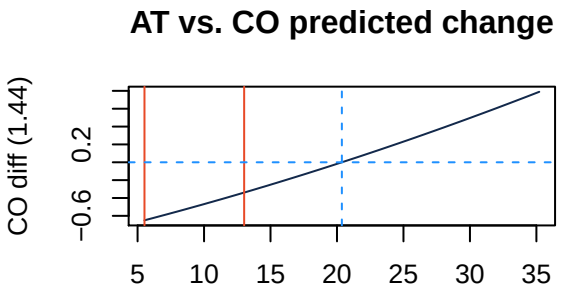
TAT (CO Min at TAT = 548.6)
CO diff Min = −0.19, Min 1se = −0.13

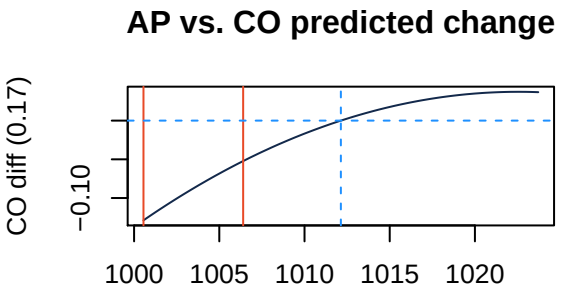


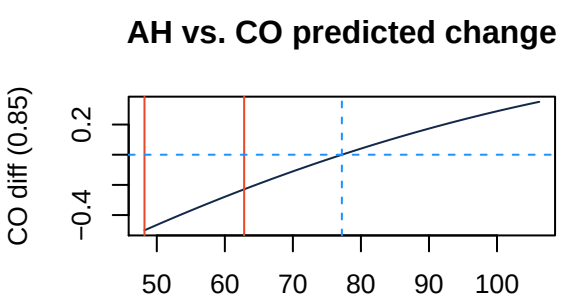
CDP (CO Min at CDP = 12)
CO diff Min = 0, Min 1se = 0

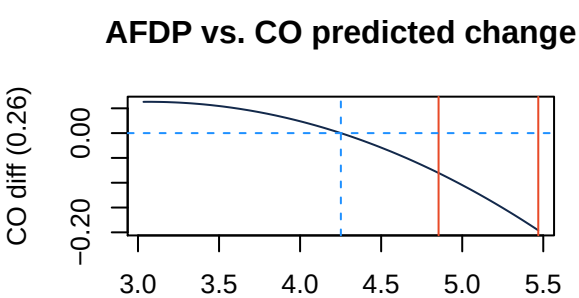
CO changes with one predictor changing with all things held constant (Model 2)

```
# j_mod
xvals = c(mean(gt_mid_all$AP), mean(gt_mid_all$AT), mean(gt_mid_all$AH), mean(gt_mid_all$AFDP), mean(gt
jmc = coef(j_mod)

par(mfrow=c(2,2))
pred.mb2(xvals, jmc, gt_mid_all$AT, 2, "AT")
pred.mb2(xvals, jmc, gt_mid_all$AP, 1, "AP")
pred.mb2(xvals, jmc, gt_mid_all$AH, 3, "AH")

pred.mb2(xvals, jmc, gt_mid_all$AFDP, 4, "AFDP")
```

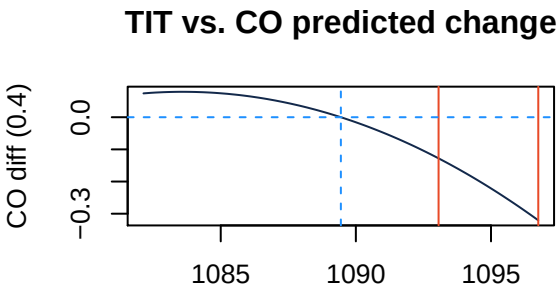




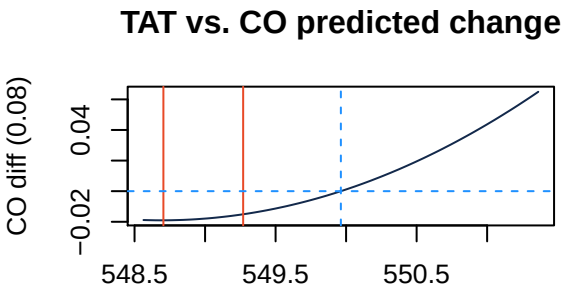




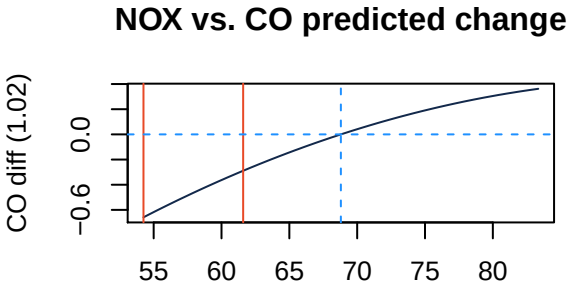

```
pred.mb2(xvals, jmc, gt_mid_all$TIT, 6, "TIT")
pred.mb2(xvals, jmc, gt_mid_all$TAT, 5, "TAT")
pred.mb2(xvals, jmc, gt_mid_all$NOX, 7, "NOX")
```



TIT (CO Min at TIT = 1096.7)
CO diff Min = −0.32, Min 1se = −0.13



TAT (CO Min at TAT = 548.7)
CO diff Min = −0.02, Min 1se = −0.02



NOX (CO Min at NOX = 54.2)
CO diff Min = −0.66, Min 1se = −0.29