Ozone_Random_Forest

Ryan Erickson

2023-06-05

Libraries

```
library(dplyr)
library(tidyr)
library(MASS)
library(rpart)
library(ranger)
library(pROC)
library(Metrics)
library(RColorBrewer)
```

Data

- Comes from data_processing.R, should run before hand for consistency
- pmax is monthly sum of daily max precipitation values

```
ozone_data=read.csv("../final_data/ozone_data.csv")%>%
  dplyr::select(site_name,date,lat,long,mda8,everything())

# ozone_data %>%
#   separate(date, c("Months","Years"),remove =F) %>%
#   mutate(Months=match(Months,month.abb)) %>%
#   group_by(Months) %>%
#   arrange(Years, Months)  %>%
#   ungroup() %>%
#   dplyr::select(mda8,everything()) %>%
#   dplyr::select(-c("Months","Years","X")) %>%
#   write.csv("../final_data/ozone_data_sorted.csv")

ozone_data$site_name = as.factor(ozone_data$site_name)
ozone_data$date = as.factor(ozone_data$date)

ozone_data_no.mda8_names = ozone_data %>%
  separate(date, c("Months","Years"),remove =F) %>%
  mutate(Months=match(Months,month.abb)) %>%
  group_by(Months) %>%
  arrange(Years, Months)  %>%
  ungroup() %>%
```

```
  dplyr::select(-c("mda8","site_name","date","Months","Years","X")) %>%
  as.data.frame()

ozone_data_mda8_first.no_name = ozone_data %>%
  separate(date, c("Months","Years"),remove =F) %>%
  mutate(Months=match(Months,month.abb)) %>%
  group_by(Months) %>%
  arrange(Years, Months)  %>%
  ungroup() %>%
  dplyr::select(mda8,everything()) %>%
  dplyr::select(-c("site_name","date","Months","Years","X")) %>%
  as.data.frame()

ozone_data %>%
  separate(date, c("Months","Years"),remove =F) %>%
  mutate(Months=match(Months,month.abb)) %>%
  group_by(Months) %>%
  arrange(Years, Months)  %>%
  ungroup() %>%
  dplyr::select(mda8,everything()) %>%
  dplyr::select(-c("Months","Years","X")) %>%
  write.csv("../final_data/ozone_data_sorted.csv")

head(ozone_data_no.mda8_names)
```

```
##       lat     long       ndvi    elev dist2road road_length     tmax    rhmax
## 1 4387727 536954.6 0.11474641 1793.14  11202.39       0.000 289.8151 64.13946
## 2 4435552 481219.8 0.13446639 1593.88    544.80       0.000 290.4293 66.00546
## 3 4400142 501060.2 0.07355672 1609.04   1174.89       0.000 291.2932 62.66088
## 4 4379800 503676.9 0.08371748 1746.35    280.62    3251.268 290.1372 62.51497
## 5 4403283 499556.4 0.17963080 1609.04    413.00    1231.818 291.2932 62.66088
## 6 4399329 484750.3 0.13003676 1766.56   1430.77       0.000 288.9421 63.72078
##       pmax apr_dummy may_dummy jun_dummy jul_dummy aug_dummy sep_dummy
## 1 21.08180         1         0         0         0         0         0
## 2 26.98729         1         0         0         0         0         0
## 3 27.10968         1         0         0         0         0         0
## 4 30.65294         1         0         0         0         0         0
## 5 27.10968         1         0         0         0         0         0
## 6 27.44648         1         0         0         0         0         0
##   oct_dummy yr_2018_dummy yr_2019_dummy yr_2020_dummy yr_2021_dummy
## 1         0             1             0             0             0
## 2         0             1             0             0             0
## 3         0             1             0             0             0
## 4         0             1             0             0             0
## 5         0             1             0             0             0
## 6         0             1             0             0             0
##   yr_2022_dummy
## 1             0
## 2             0
## 3             0
## 4             0
## 5             0
## 6             0
```

```r
head(ozone_data_mda8_first.no_name)
```

```
##       mda8     lat     long       ndvi    elev dist2road road_length     tmax
## 1 45.43727 4387727 536954.6 0.11474641 1793.14  11202.39       0.000 289.8151
## 2 43.20485 4435552 481219.8 0.13446639 1593.88    544.80       0.000 290.4293
## 3 37.63830 4400142 501060.2 0.07355672 1609.04   1174.89       0.000 291.2932
## 4 45.89848 4379800 503676.9 0.08371748 1746.35    280.62    3251.268 290.1372
## 5 37.24626 4403283 499556.4 0.17963080 1609.04    413.00    1231.818 291.2932
## 6 43.74330 4399329 484750.3 0.13003676 1766.56   1430.77       0.000 288.9421
##      rhmax     pmax apr_dummy may_dummy jun_dummy jul_dummy aug_dummy sep_dummy
## 1 64.13946 21.08180         1         0         0         0         0         0
## 2 66.00546 26.98729         1         0         0         0         0         0
## 3 62.66088 27.10968         1         0         0         0         0         0
## 4 62.51497 30.65294         1         0         0         0         0         0
## 5 62.66088 27.10968         1         0         0         0         0         0
## 6 63.72078 27.44648         1         0         0         0         0         0
##   oct_dummy yr_2018_dummy yr_2019_dummy yr_2020_dummy yr_2021_dummy
## 1         0             1             0             0             0
## 2         0             1             0             0             0
## 3         0             1             0             0             0
## 4         0             1             0             0             0
## 5         0             1             0             0             0
## 6         0             1             0             0             0
##   yr_2022_dummy
## 1             0
## 2             0
## 3             0
## 4             0
## 5             0
## 6             0
```

Summary of LM models with Dummy Variables Included

```r
line1 = glm(mda8~., data=ozone_data_mda8_first.no_name)
summary(line1)
```

```
##
## Call:
## glm(formula = mda8 ~ ., data = ozone_data_mda8_first.no_name)
##
## Coefficients: (2 not defined because of singularities)
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -5.242e+02  1.994e+02  -2.629 0.009080 **
## lat            9.600e-05  3.423e-05   2.805 0.005427 **
## long          -1.765e-04  5.990e-05  -2.947 0.003507 **
## ndvi          -9.681e+00  3.729e+00  -2.596 0.009989 **
## elev           5.089e-02  8.327e-03   6.112 3.72e-09 ***
## dist2road      6.616e-04  3.088e-04   2.143 0.033108 *
## road_length    1.132e-03  2.847e-04   3.977 9.13e-05 ***
## tmax           4.715e-01  1.442e-01   3.269 0.001229 **
## rhmax         -4.213e-02  4.263e-02  -0.988 0.323978
```

```
## pmax          -2.839e-02  1.051e-02  -2.702 0.007355 **
## apr_dummy      1.164e+01  6.201e-01  18.768  < 2e-16 ***
## may_dummy      1.104e+01  1.132e+00   9.756  < 2e-16 ***
## jun_dummy      1.083e+01  1.867e+00   5.801 1.97e-08 ***
## jul_dummy      1.332e+01  2.356e+00   5.654 4.24e-08 ***
## aug_dummy      1.361e+01  2.144e+00   6.345 1.02e-09 ***
## sep_dummy      5.481e+00  1.551e+00   3.533 0.000488 ***
## oct_dummy             NA         NA      NA       NA
## yr_2018_dummy -9.263e-01  5.206e-01  -1.779 0.076431 .
## yr_2019_dummy -1.310e+00  5.434e-01  -2.411 0.016612 *
## yr_2020_dummy -1.205e+00  5.192e-01  -2.322 0.021050 *
## yr_2021_dummy  2.741e+00  5.327e-01   5.146 5.36e-07 ***
## yr_2022_dummy         NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 6.538075)
##
##     Null deviance: 17900.9  on 271  degrees of freedom
## Residual deviance:  1647.6  on 252  degrees of freedom
## AIC: 1303.8
##
## Number of Fisher Scoring iterations: 2
```

Summary of LM models with no Dummy Variables

```
line_nd=ozone_data_mda8_first.no_name[,c(-2,-3,-11:-22)]
line_nd = glm(mda8~., data=line_nd)
summary(line_nd)
```

```
##
## Call:
## glm(formula = mda8 ~ ., data = line_nd)
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.276e+02  2.090e+01 -15.677  < 2e-16 ***
## ndvi         1.176e+01  6.075e+00   1.935   0.0540 .
## elev         6.951e-02  5.895e-03  11.791  < 2e-16 ***
## dist2road   -6.180e-04  1.342e-04  -4.606  6.4e-06 ***
## road_length -4.306e-04  3.399e-04  -1.267   0.2063
## tmax         8.317e-01  5.043e-02  16.494  < 2e-16 ***
## rhmax        6.650e-02  6.542e-02   1.017   0.3103
## pmax         2.916e-02  1.648e-02   1.769   0.0781 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 25.09887)
##
##     Null deviance: 17900.9  on 271  degrees of freedom
## Residual deviance:  6626.1  on 264  degrees of freedom
## AIC: 1658.4
```

```
##
## Number of Fisher Scoring iterations: 2
```

Checking Linearity for all Variables

```
n = 21
qual_col_pals = brewer.pal.info[brewer.pal.info$category == 'qual',]
col_vector = unlist(mapply(brewer.pal, qual_col_pals$maxcolors, rownames(qual_col_pals)))
col=sample(col_vector, n)

for (i in 1:ncol(ozone_data_no.mda8_names)) {
 plot(x=ozone_data_no.mda8_names[,i],
       y=ozone_data_mda8_first.no_name$mda8,
       main = paste0('Figure ',i,': MDA8 vs. ',colnames(ozone_data_no.mda8_names)[i]),
       xlab = paste0(colnames(ozone_data_no.mda8_names)[i]),
       ylab = "MDA8 Value",
       pch = 19)
  abline(lm(reformulate(paste0(names(ozone_data_no.mda8_names[i])),"mda8"),ozone_data_mda8_first.no_name
         col = col[i],
         lwd = 2)
}
```

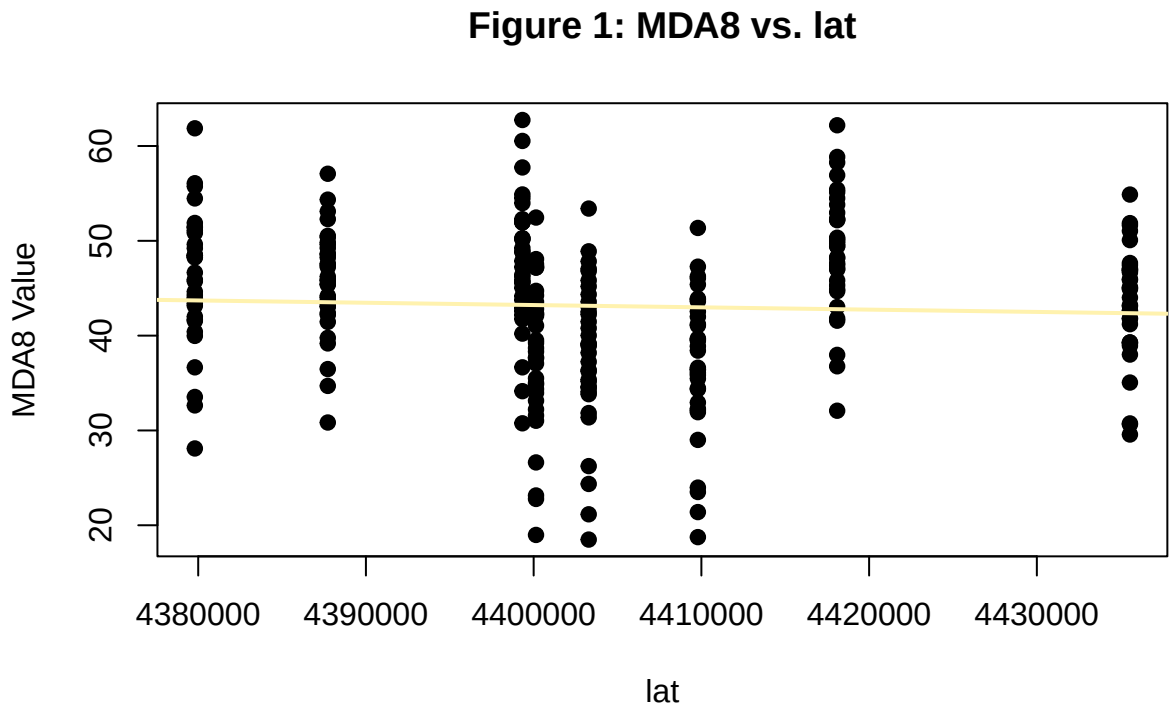


Figure 2: MDA8 vs. long

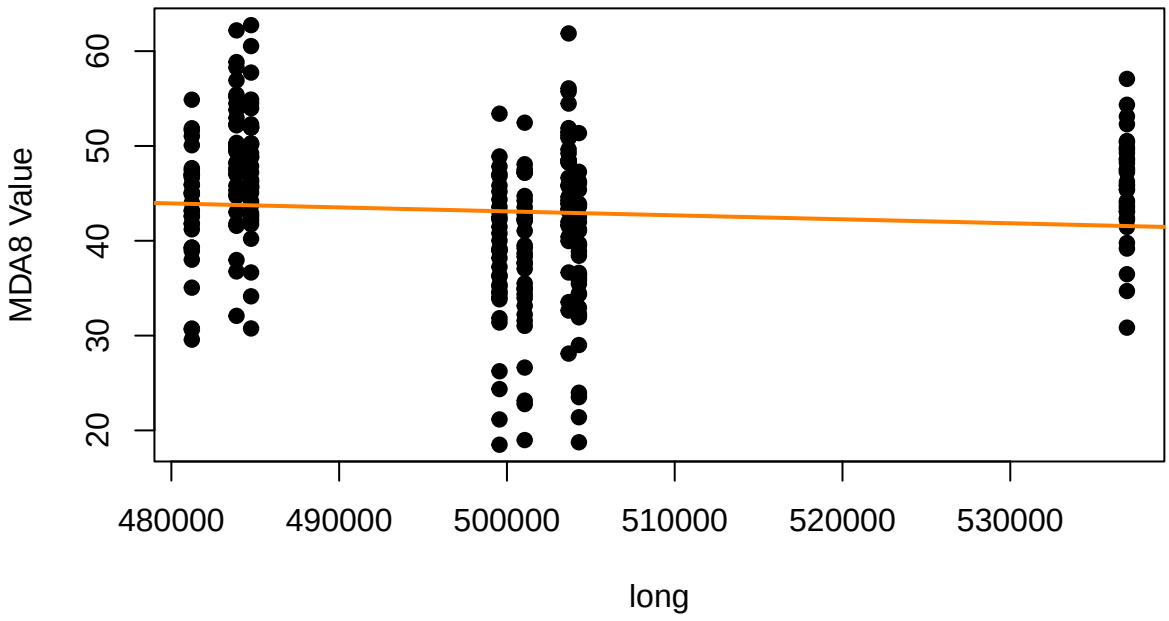


Figure 3: MDA8 vs. ndvi

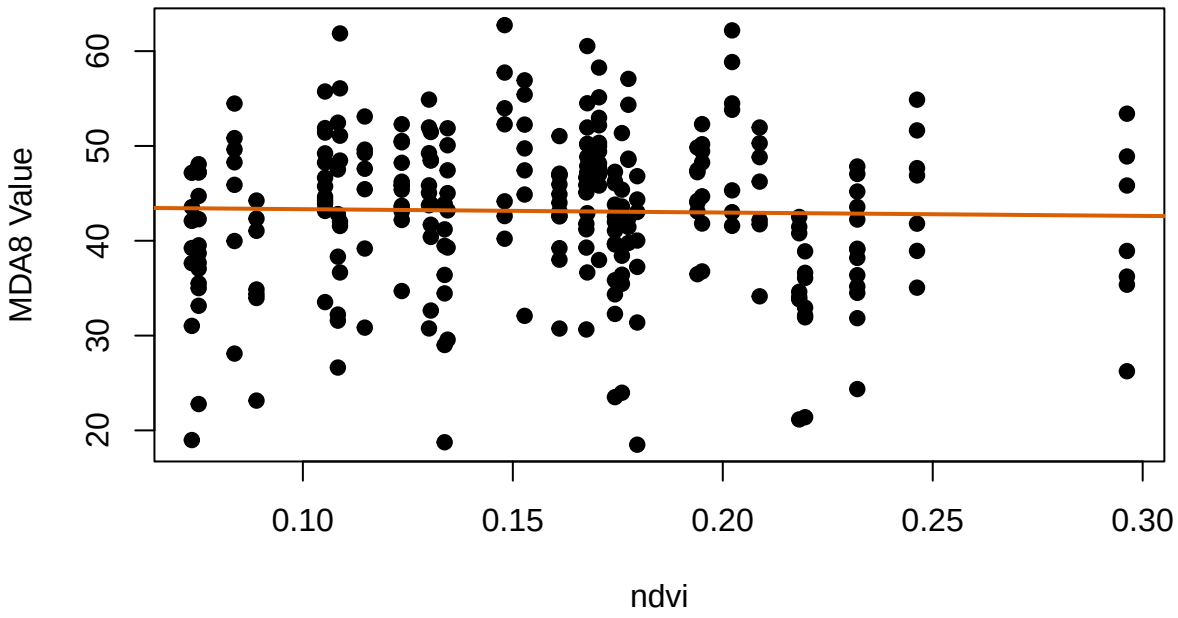


Figure 4: MDA8 vs. elev

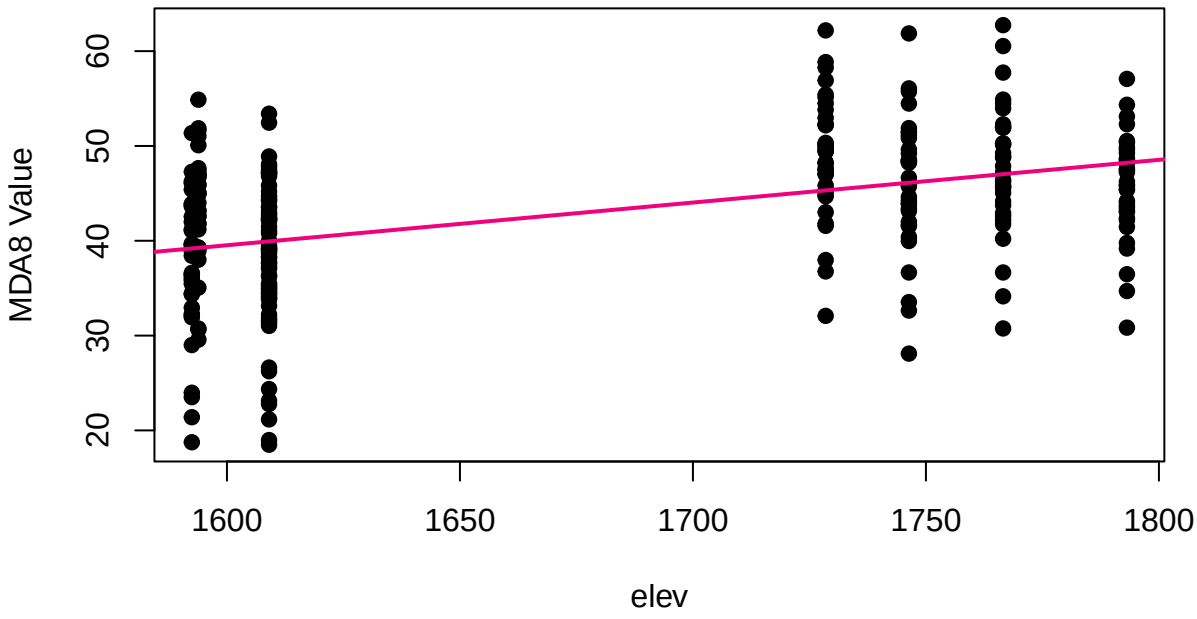


Figure 5: MDA8 vs. dist2road

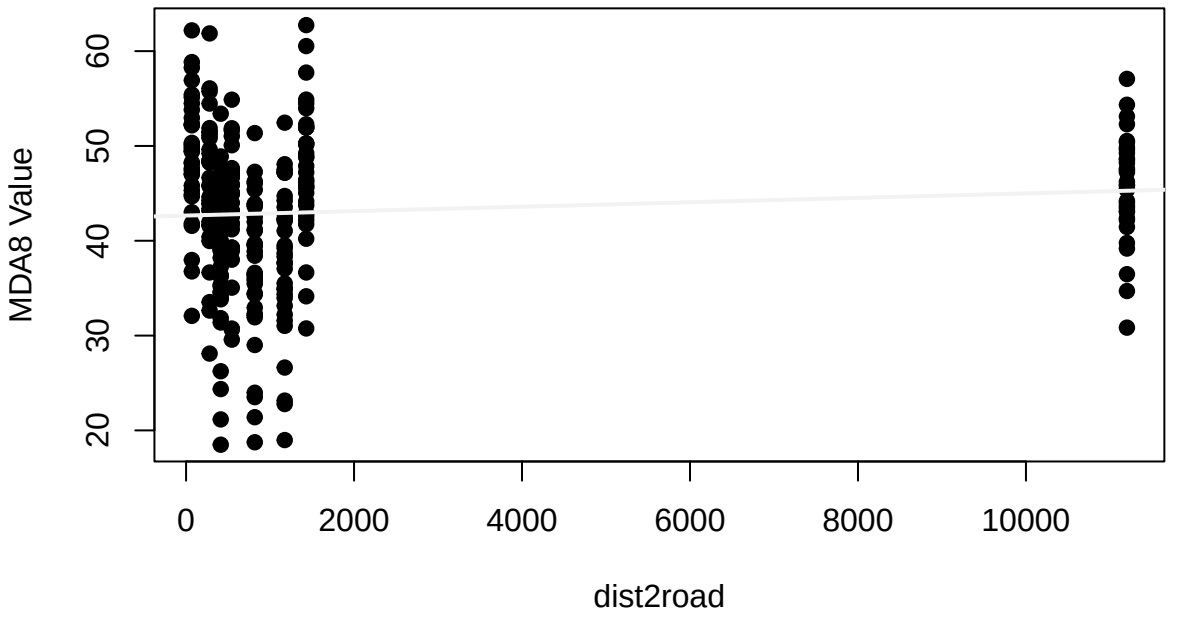


Figure 6: MDA8 vs. road_length

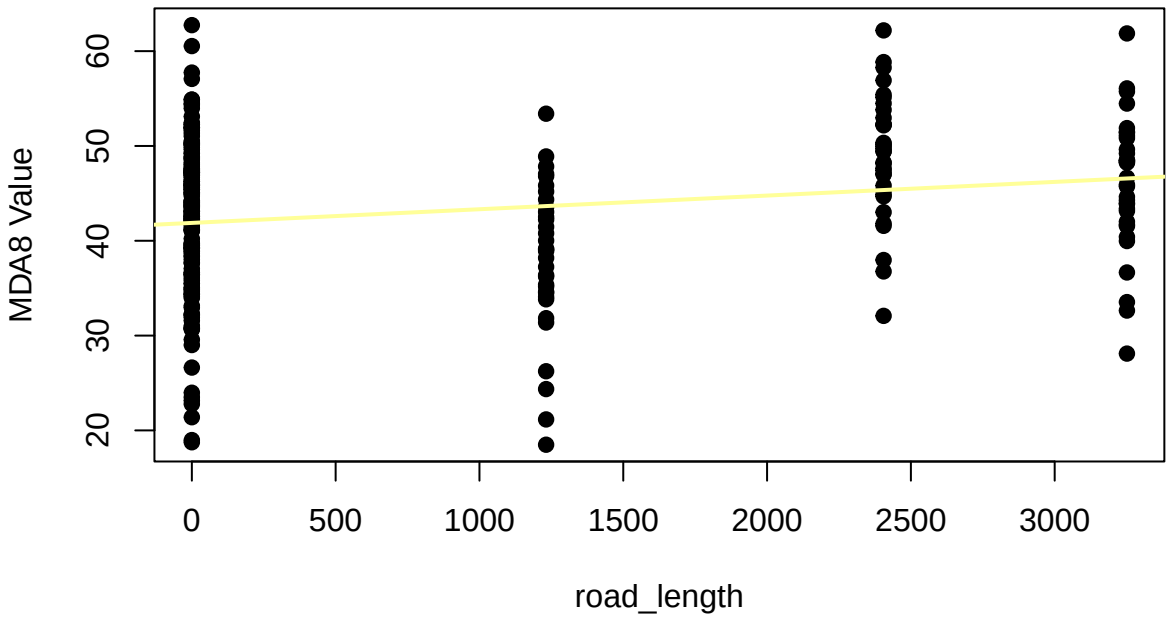


Figure 7: MDA8 vs. tmax

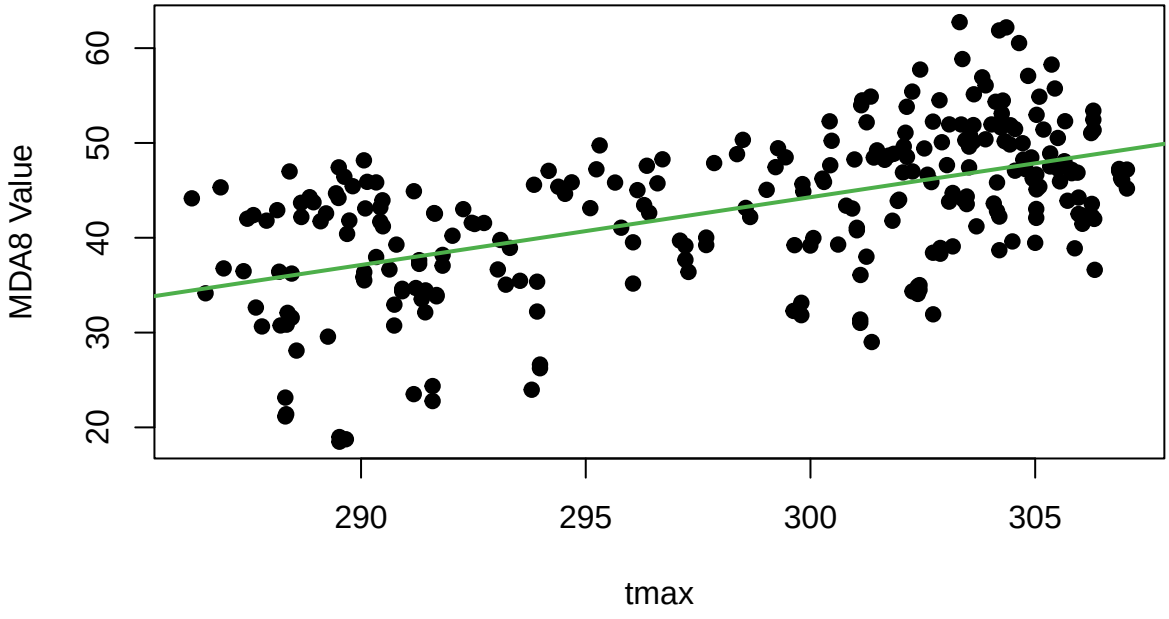


Figure 8: MDA8 vs. rhmax

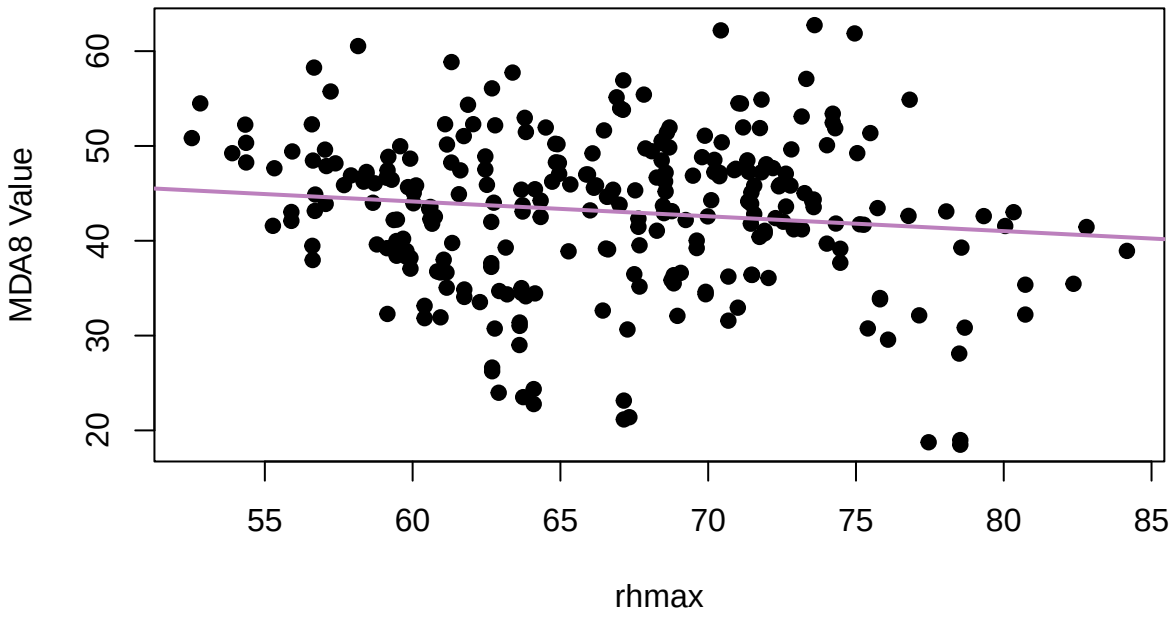


Figure 9: MDA8 vs. pmax

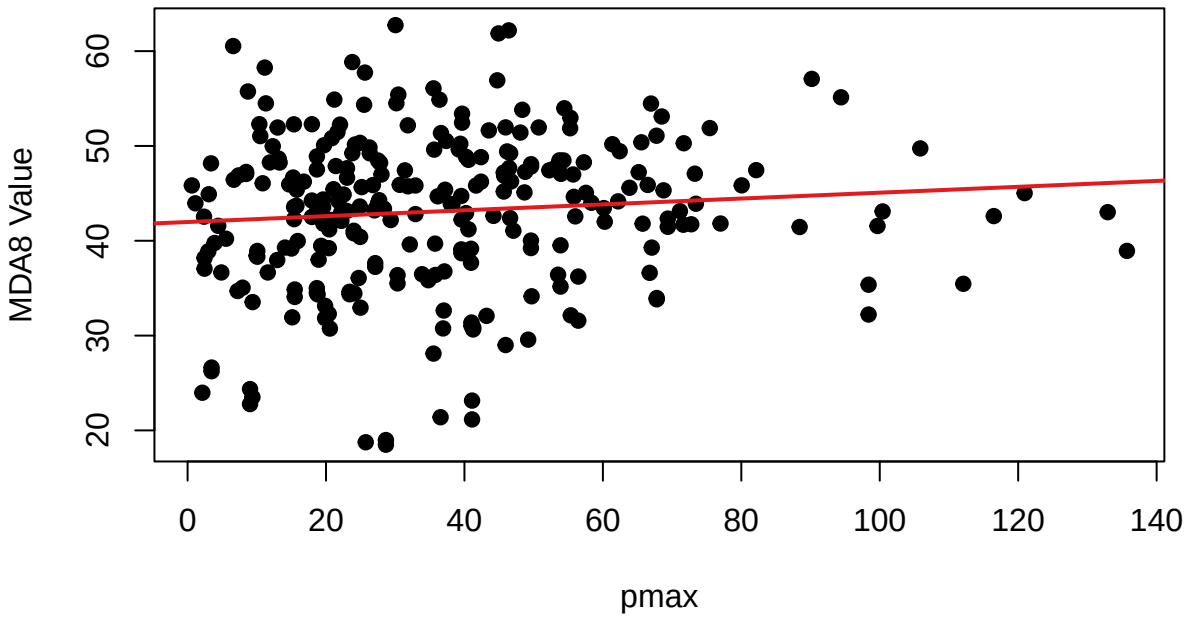


Figure 10: MDA8 vs. apr_dummy

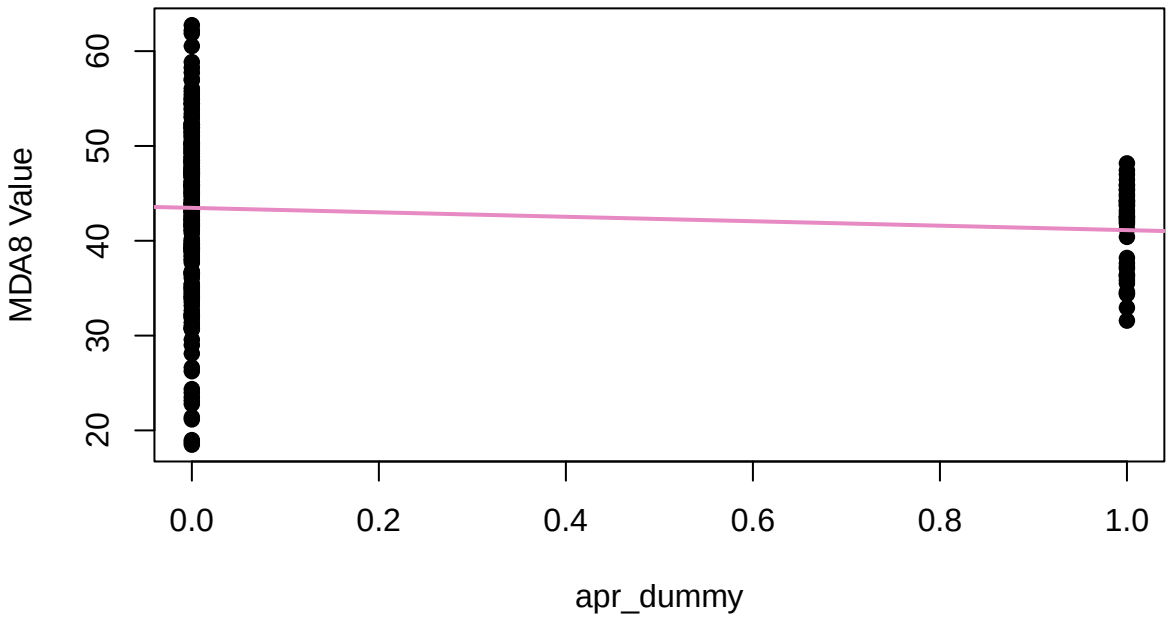


Figure 11: MDA8 vs. may_dummy

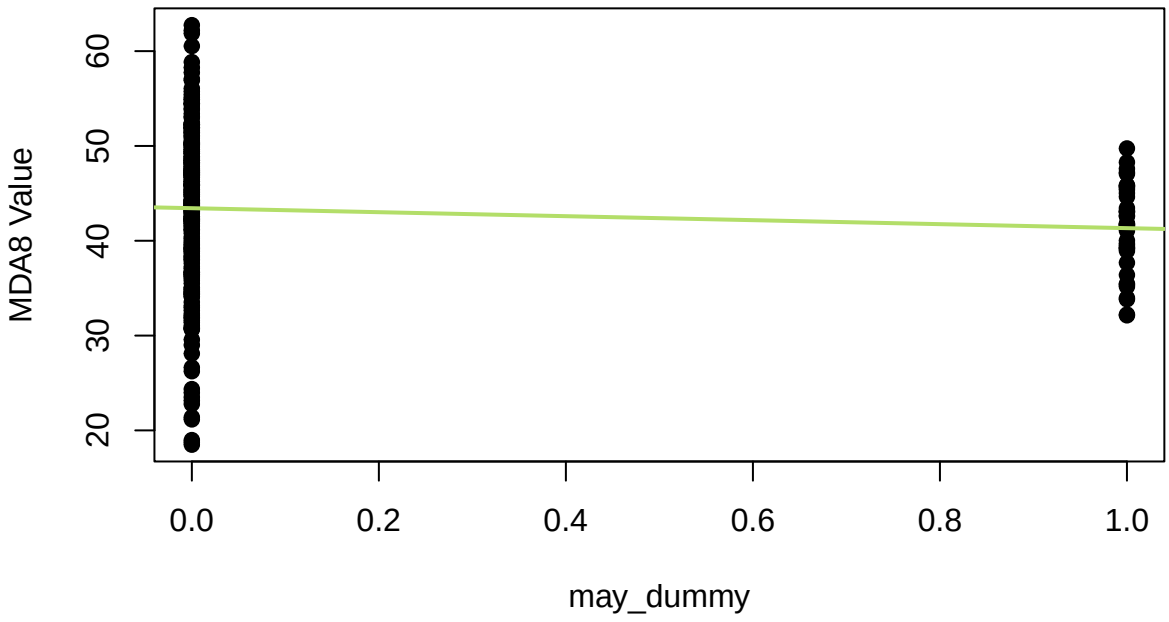


Figure 12: MDA8 vs. jun_dummy

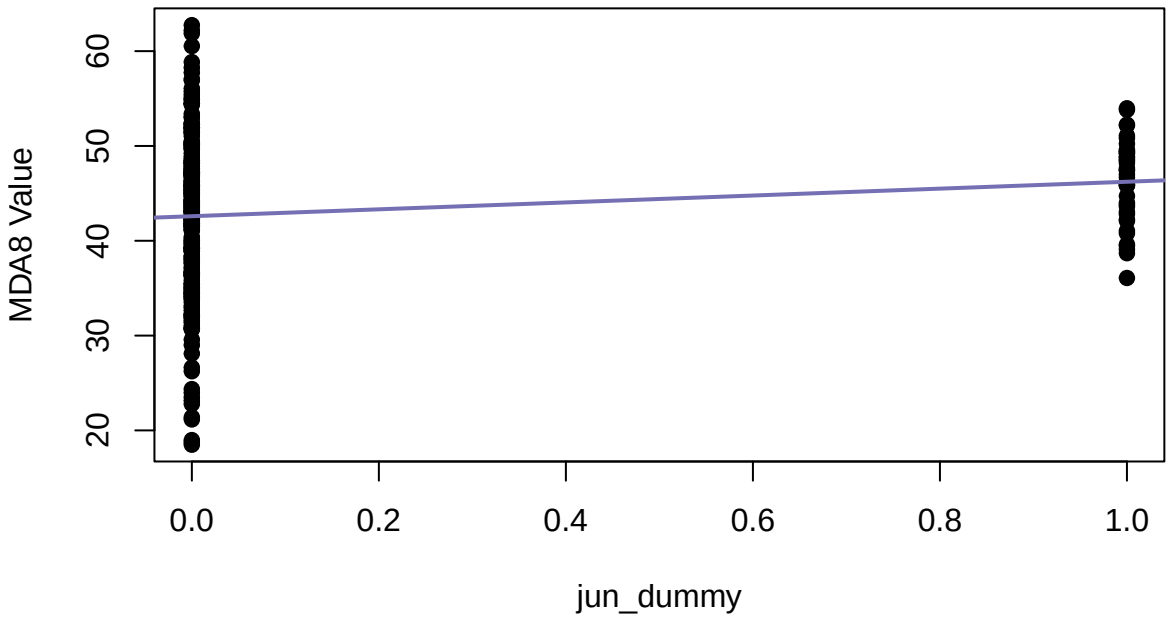


Figure 13: MDA8 vs. jul_dummy

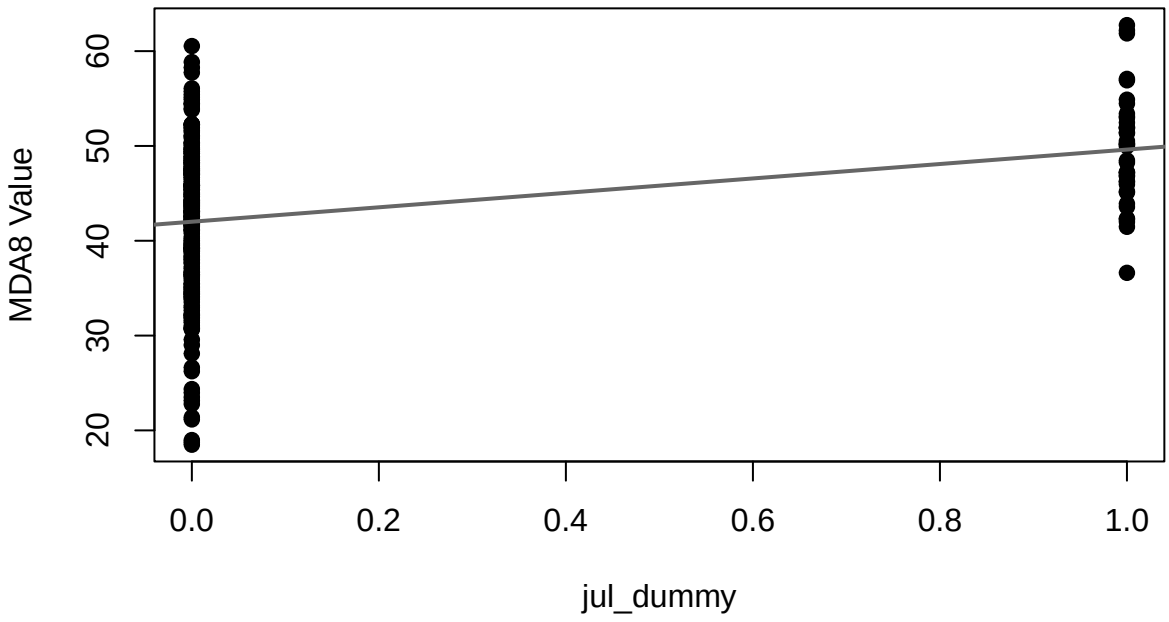


Figure 14: MDA8 vs. aug_dummy

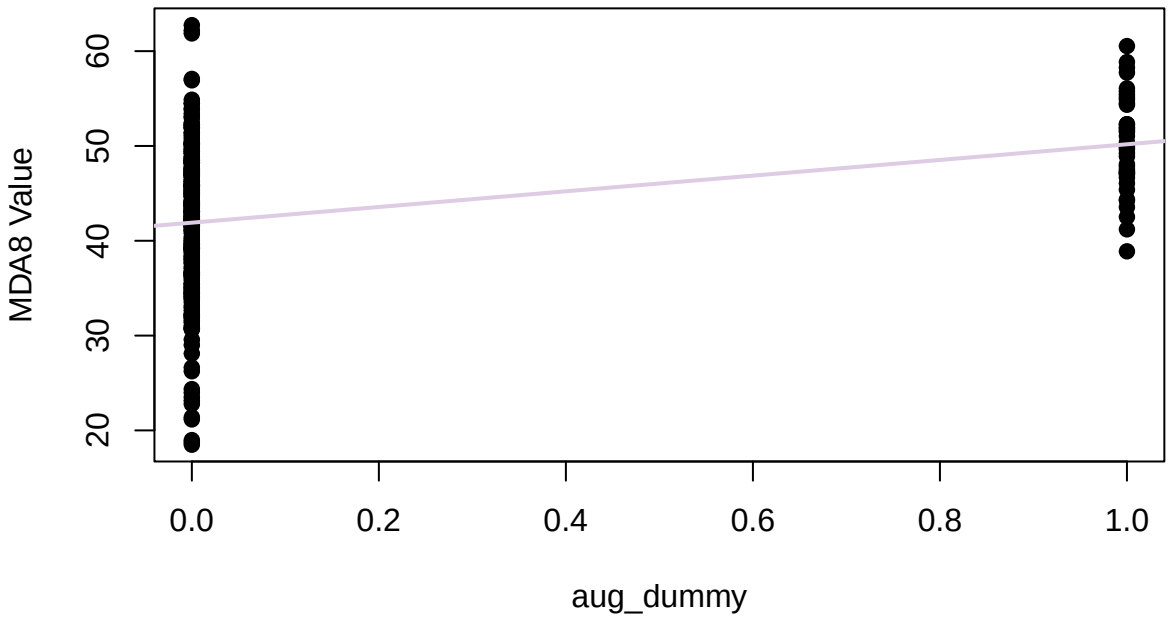


Figure 15: MDA8 vs. sep_dummy

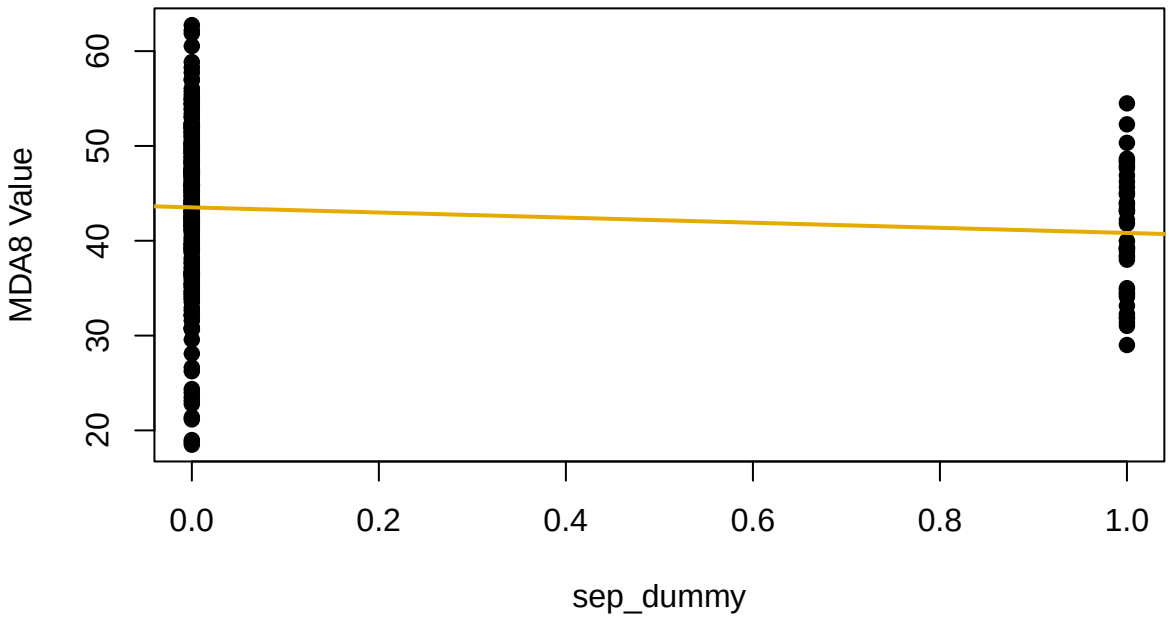


Figure 16: MDA8 vs. oct_dummy

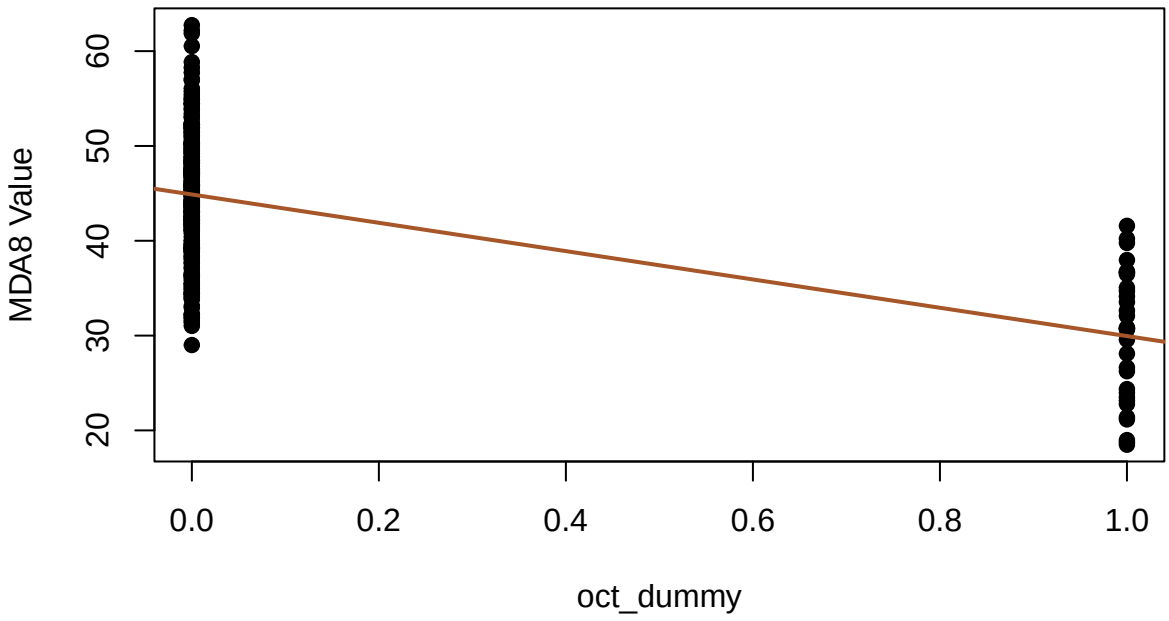


Figure 17: MDA8 vs. yr_2018_dummy

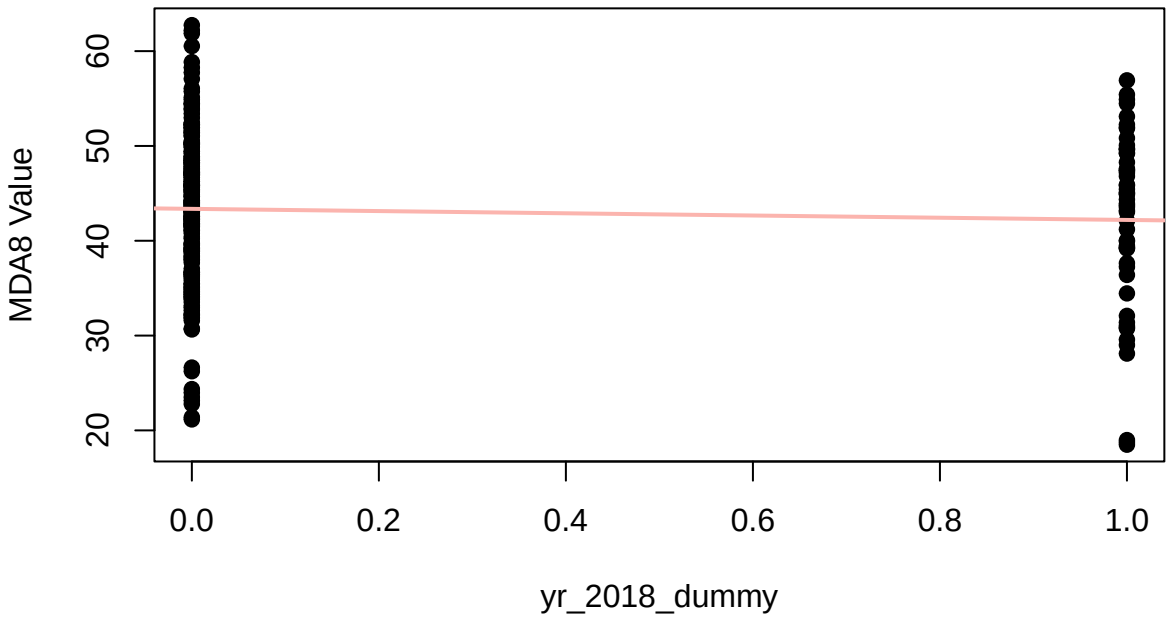


Figure 18: MDA8 vs. yr_2019_dummy

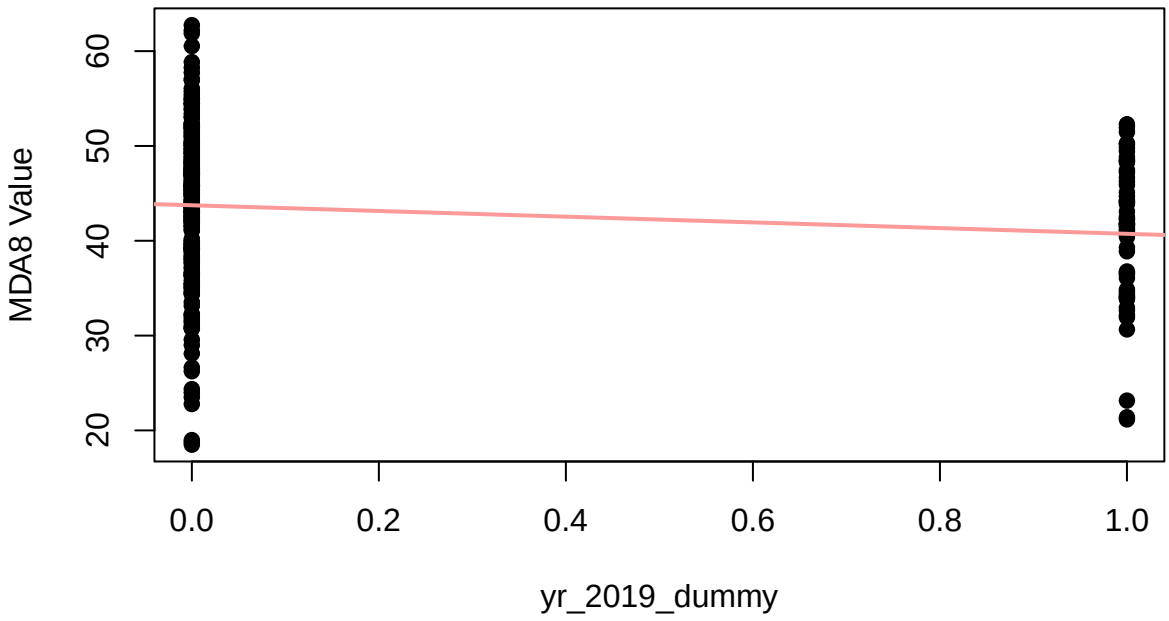


Figure 19: MDA8 vs. yr_2020_dummy

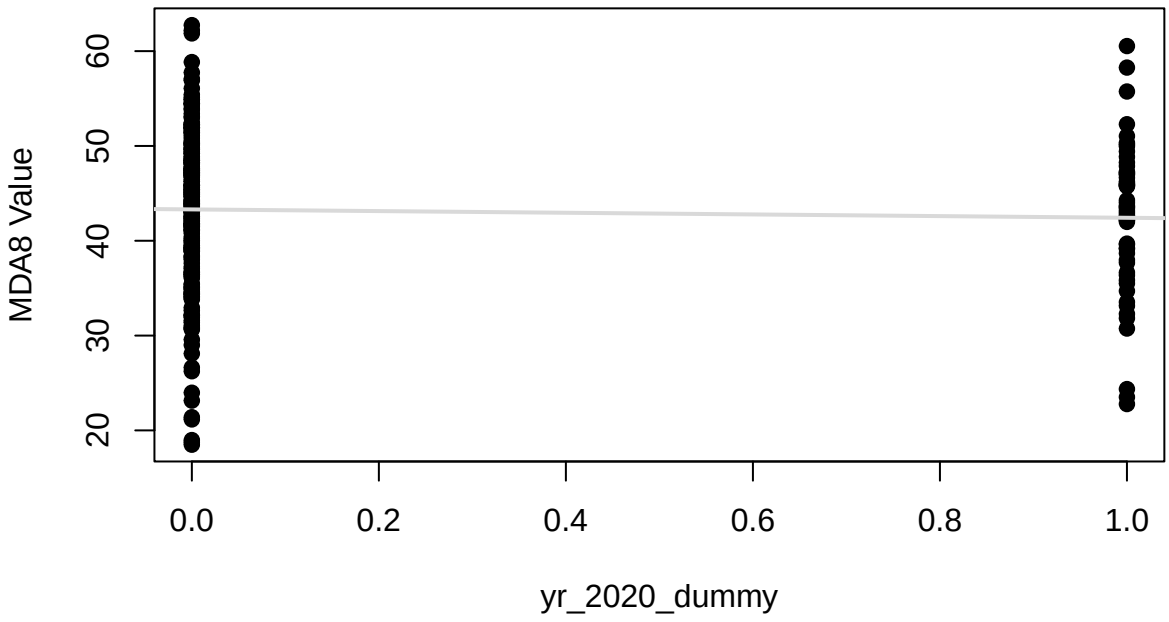


Figure 20: MDA8 vs. yr_2021_dummy

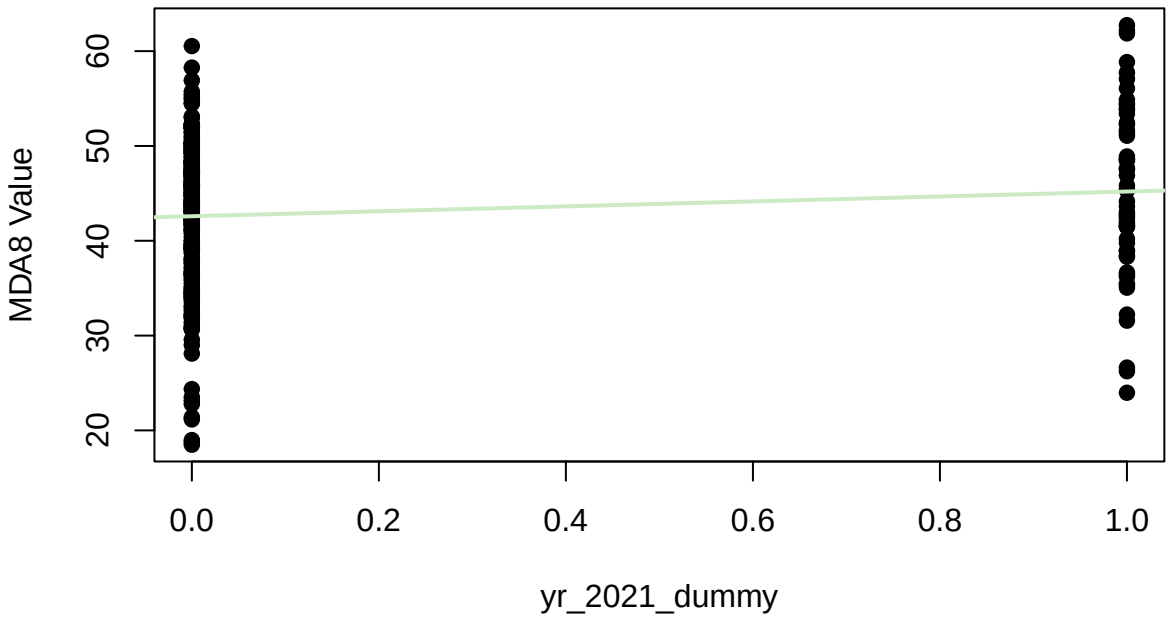
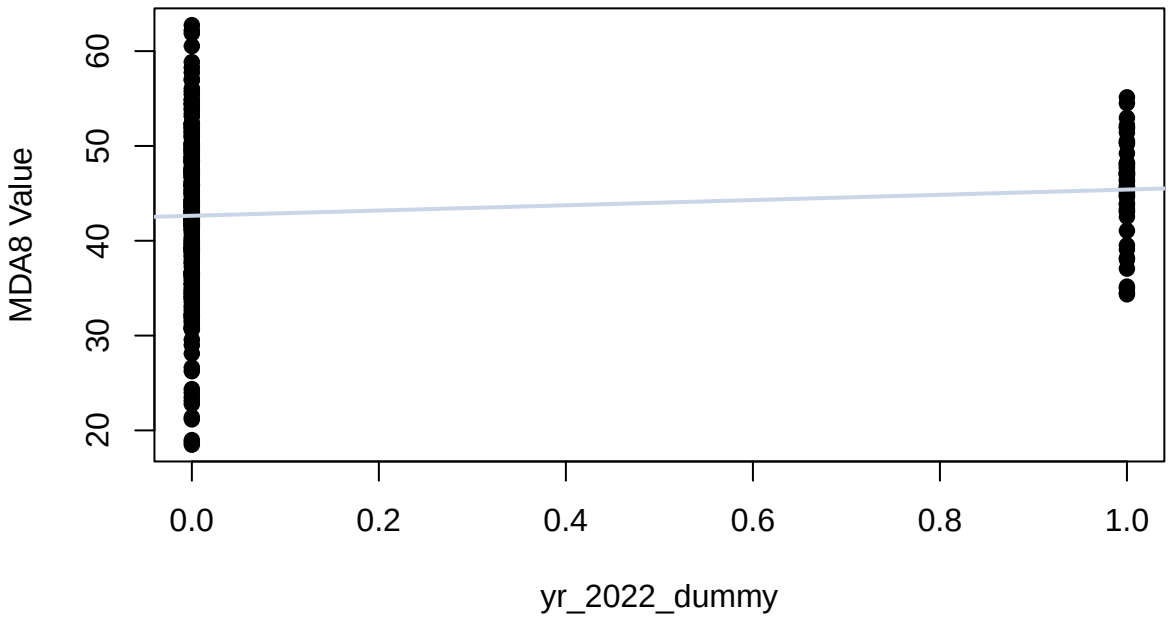


Figure 21: MDA8 vs. yr_2022_dummy



Checking Linearity for variables, exluding dummy variables

```
rf_model_nd.mda8=as.data.frame(ozone_data_mda8_first.no_name[,c(-2,-3,-11:-22)])
rf_model_nd=as.data.frame(ozone_data_mda8_first.no_name[,c(-1,-2,-3,-11:-22)])

n = 7
qual_col_pals = brewer.pal.info[brewer.pal.info$category == 'qual',]
col_vector = unlist(mapply(brewer.pal, qual_col_pals$maxcolors, rownames(qual_col_pals)))
col=sample(col_vector, n)

for (i in 1:ncol(rf_model_nd)) {
 plot(x=rf_model_nd[,i],
       y=rf_model_nd.mda8$mda8,
       main = paste0('Figure ',i,': MDA8 vs. ',colnames(rf_model_nd)[i]),
       xlab = paste0(colnames(rf_model_nd)[i]),
       ylab = "MDA8 Value",
       pch = 19)
  abline(lm(reformulate(paste0(names(rf_model_nd[i])),"mda8"),rf_model_nd.mda8),
         col = col[i],
         lwd = 2)
}
```

Figure 1: MDA8 vs. ndvi

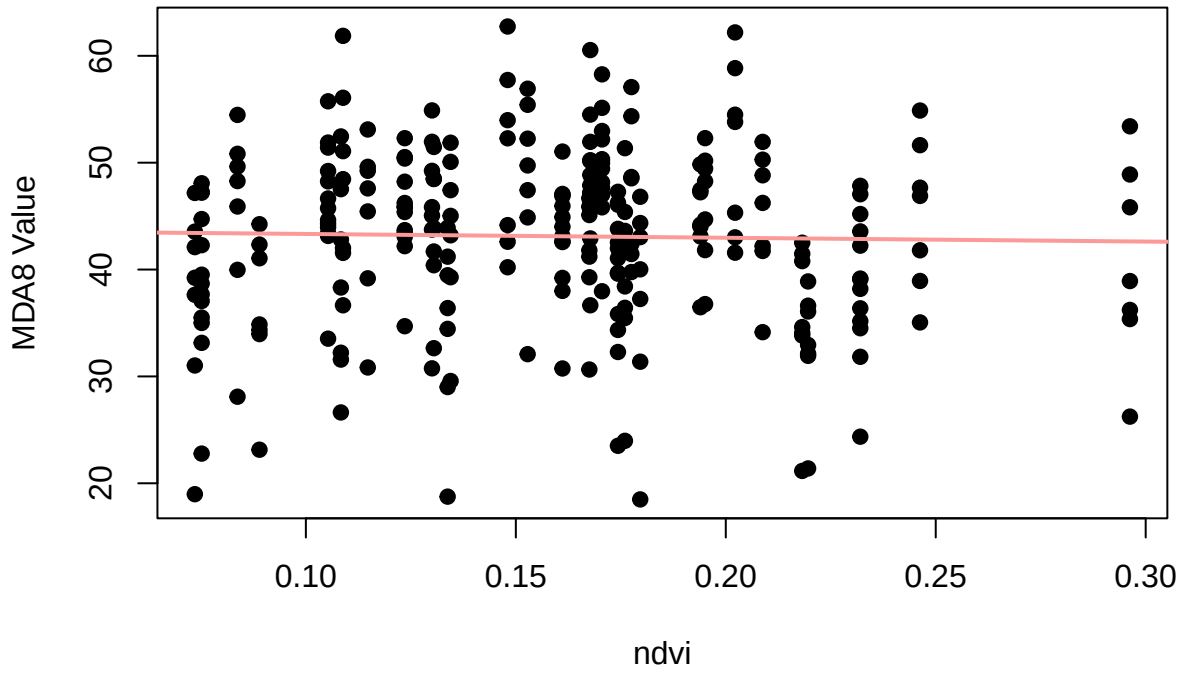


Figure 2: MDA8 vs. elev

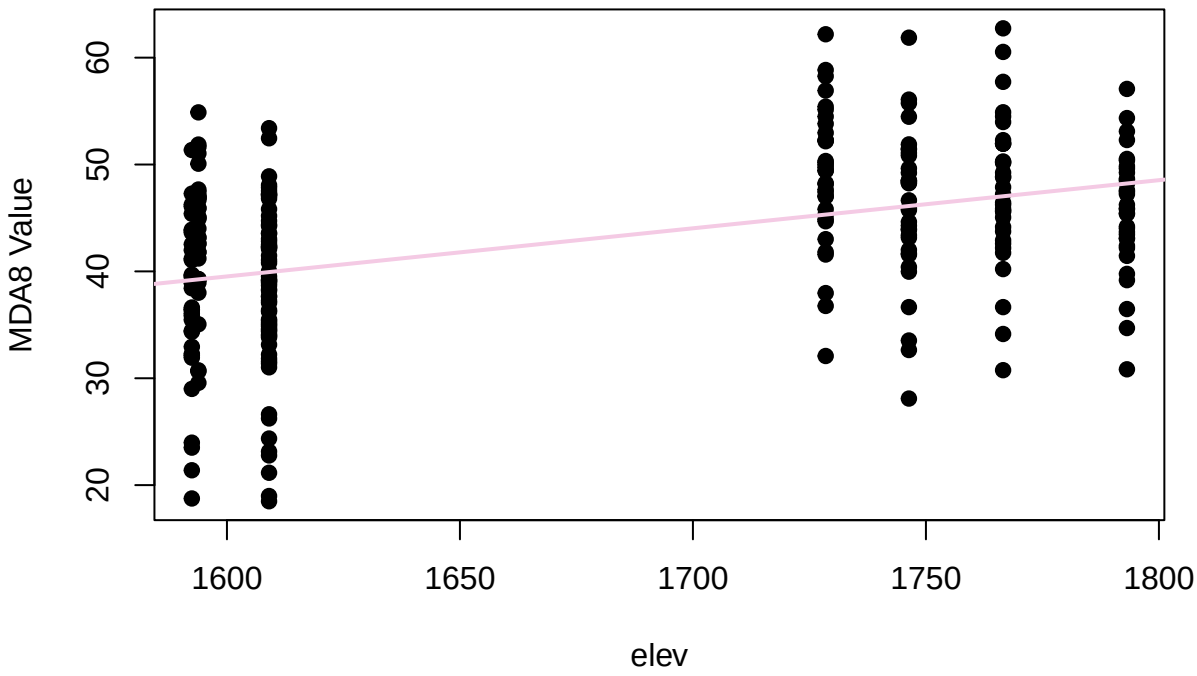


Figure 3: MDA8 vs. dist2road

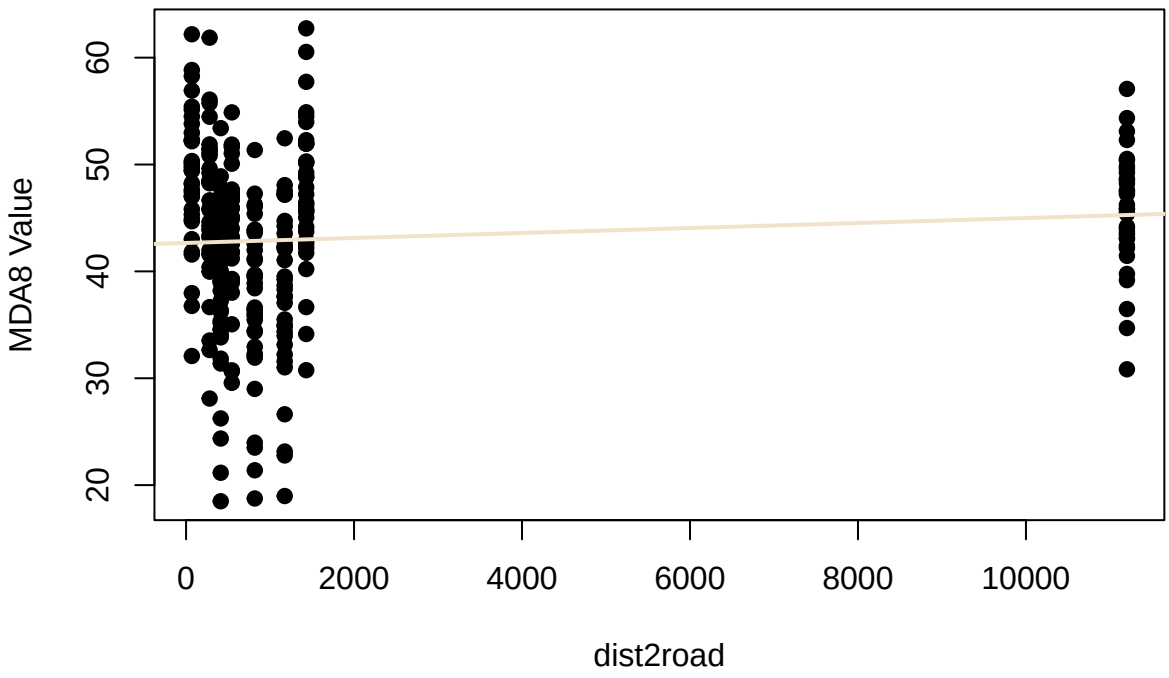


Figure 4: MDA8 vs. road_length

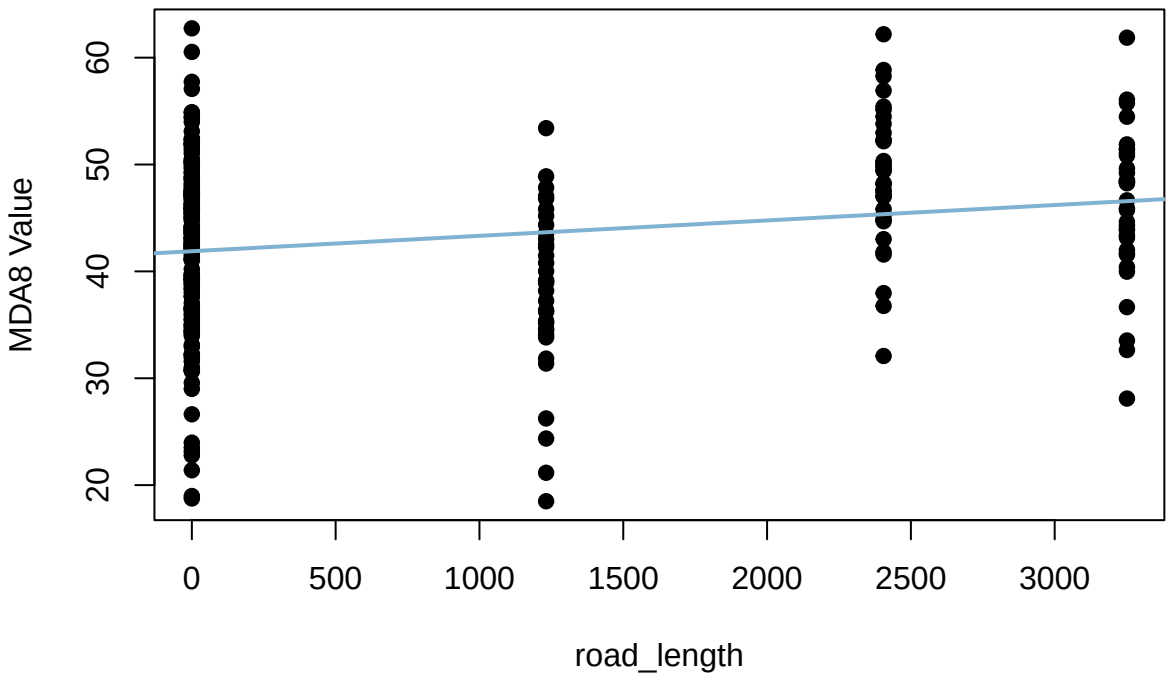


Figure 5: MDA8 vs. tmax

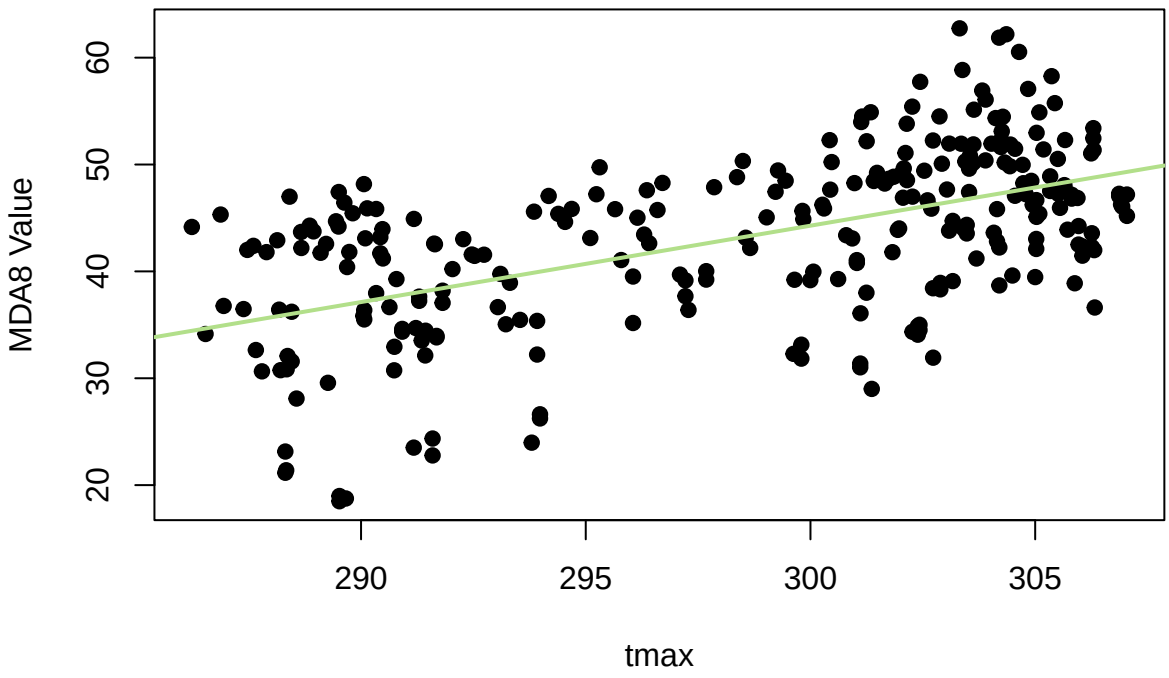


Figure 6: MDA8 vs. rhmax

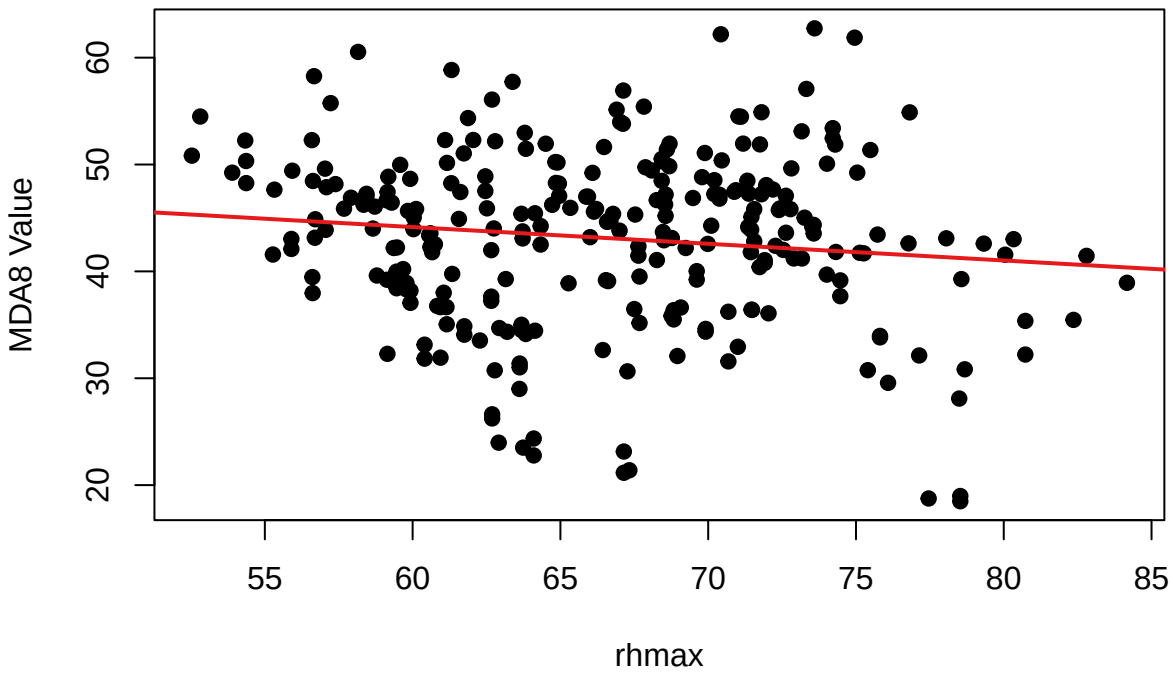
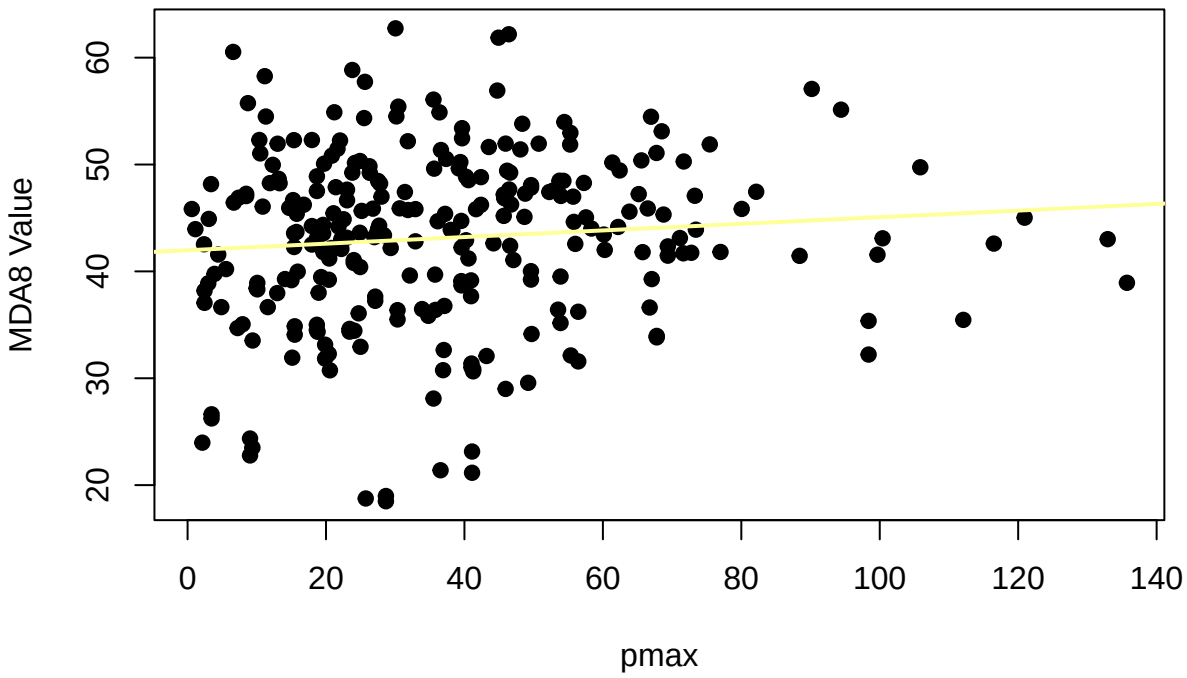


Figure 7: MDA8 vs. pmax



Splitting the Data - Testing Linear Models: 75% train/test split

```
sample_size = floor(0.75 * nrow(ozone_data_mda8_first.no_name))
set.seed(09111997)
splitting_data = sample(seq_len(nrow(ozone_data_mda8_first.no_name)), size = sample_size, replace=FALSE)

ozone_train_wd = ozone_data_mda8_first.no_name[splitting_data, ]
ozone_test_wd = ozone_data_mda8_first.no_name[-splitting_data, ]

lm.final_wd = glm(mda8~., data = ozone_train_wd)
###predicting on training date with test data
pred.vals = predict(object=lm.final_wd,new_data=ozone_test_wd,type = "response")
summary(lm.final_wd)
```

```
##
## Call:
## glm(formula = mda8 ~ ., data = ozone_train_wd)
##
## Coefficients: (2 not defined because of singularities)
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -4.990e+02  2.373e+02  -2.102 0.036888 *
## lat            1.018e-04  4.091e-05   2.489 0.013716 *
## long          -1.891e-04  7.218e-05  -2.620 0.009527 **
## ndvi          -1.318e+01  4.491e+00  -2.935 0.003764 **
```

```
## elev           4.821e-02  9.901e-03   4.869 2.41e-06 ***
## dist2road      7.575e-04  3.718e-04   2.038 0.043011 *
## road_length    1.287e-03  3.391e-04   3.796 0.000199 ***
## tmax           3.348e-01  1.643e-01   2.038 0.042998 *
## rhmax         -3.381e-02  4.979e-02  -0.679 0.497957
## pmax          -3.651e-02  1.229e-02  -2.970 0.003375 **
## apr_dummy      1.123e+01  7.334e-01  15.316  < 2e-16 ***
## may_dummy      1.190e+01  1.351e+00   8.804 9.58e-16 ***
## jun_dummy      1.220e+01  2.150e+00   5.671 5.41e-08 ***
## jul_dummy      1.556e+01  2.738e+00   5.682 5.13e-08 ***
## aug_dummy      1.527e+01  2.458e+00   6.213 3.39e-09 ***
## sep_dummy      6.761e+00  1.779e+00   3.800 0.000197 ***
## oct_dummy             NA         NA      NA       NA
## yr_2018_dummy -1.344e+00  6.313e-01  -2.128 0.034631 *
## yr_2019_dummy -1.242e+00  6.231e-01  -1.993 0.047790 *
## yr_2020_dummy -1.385e+00  6.160e-01  -2.249 0.025718 *
## yr_2021_dummy  2.772e+00  6.168e-01   4.494 1.23e-05 ***
## yr_2022_dummy         NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 6.667822)
##
##     Null deviance: 13413.6  on 203  degrees of freedom
## Residual deviance:  1226.9  on 184  degrees of freedom
## AIC: 986.93
##
## Number of Fisher Scoring iterations: 2
```

Linear Model RMSE = 2.45

Splitting the Data Without Dummy Variables - Testing Linear Models: 75% train/test split

```
sample_size2 = floor(0.75 * nrow(rf_model_nd.mda8))
set.seed(09111997)
split_data = sample(seq_len(nrow(rf_model_nd.mda8)), size = sample_size2, replace=FALSE)

ozone_train = rf_model_nd.mda8[split_data, ]
ozone_test = rf_model_nd.mda8[-split_data, ]

lm.final = glm(mda8~., data = ozone_train)
pred.vals2 = predict(lm.final, new_data=ozone_test)
summary(lm.final)
```

```
##
## Call:
## glm(formula = mda8 ~ ., data = ozone_train)
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.343e+02  2.375e+01 -14.075  < 2e-16 ***
## ndvi         7.716e+00  7.244e+00   1.065    0.288
```

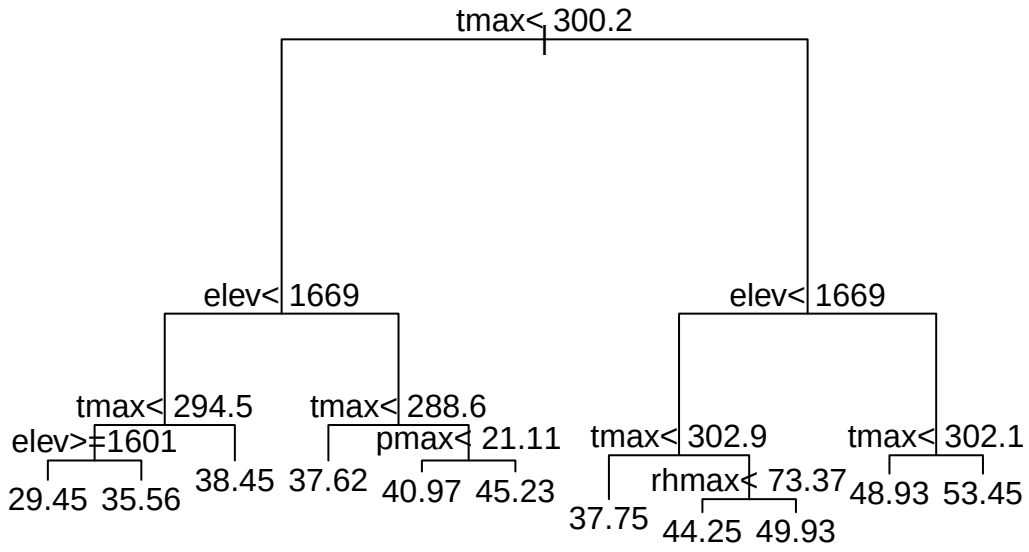
```
## elev         7.224e-02  6.668e-03  10.833  < 2e-16 ***
## dist2road   -7.173e-04  1.505e-04  -4.767 3.64e-06 ***
## road_length -5.756e-04  3.901e-04  -1.476    0.142
## tmax         8.332e-01  5.863e-02  14.211  < 2e-16 ***
## rhmax        1.119e-01  7.448e-02   1.502    0.135
## pmax         2.173e-02  1.926e-02   1.129    0.260
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 25.2699)
##
##     Null deviance: 13413.6  on 203  degrees of freedom
## Residual deviance:  4952.9  on 196  degrees of freedom
## AIC: 1247.6
##
## Number of Fisher Scoring iterations: 2
```

Linear Model RMSE = 4.93

Fitting a Regression Tree - No Dummy Variables since RMSE was a little better without them

- predict by month!!!

```
fit.tree = rpart(mda8~.,data=ozone_train)
#summary(fit.tree)
par(xpd = NA)
plot(fit.tree)
text(fit.tree)
```




```
pred.tree = predict(fit.tree,newdata=ozone_test)
```

Regression Tree RMSE = 5.17

Bagged tree for Comparison

```
pred.boot = ranger(mda8~.,data=ozone_train,mtry=dim(ozone_train)[2]-1,num.trees=500)
pred.bag = predict(pred.boot,data=ozone_test)$predictions

imp_feats = ranger(mda8~.,data=ozone_train,probability=TRUE,importance="impurity_corrected", mtry=dim(oz

cbind(sort(importance(imp_feats)))
```

```
##                    [,1]
## rhmax       -0.77806299
## elev        -0.34701364
## ndvi        -0.10119047
## pmax        -0.05385316
## dist2road   -0.04113476
## tmax        -0.03414605
## road_length  0.28286353
```

Bagging RMSE = 4.29

Random Forest

```
fit.rf = ranger(mda8~.,data=ozone_train, num.trees = 500)
pred.rf = predict(fit.rf,data=ozone_test)
pred.rf = pred.rf$predictions

imp_feats2 = ranger(mda8~.,data=ozone_train,probability=TRUE,importance="impurity_corrected", num.trees
cbind(sort(importance(imp_feats2)))
```

```
##                    [,1]
## rhmax       -0.49024403
## pmax        -0.21295019
## ndvi        -0.21212800
## dist2road   -0.19167245
## road_length -0.06108628
## elev         0.17390904
## tmax         0.18109816
```

RMSE = 4.82

LOOCV

```
library(caret)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'caret'
```

```
## The following objects are masked from 'package:Metrics':
##
##     precision, recall
```

```
#specify the cross-validation method
ctrl <- trainControl(method = "LOOCV")

#fit a regression model and use LOOCV to evaluate performance
model <- train(mda8 ~ ., data = rf_model_nd.mda8, method = "rf", trControl = ctrl)
model$results
```

```
##   mtry     RMSE  Rsquared      MAE
## 1    2 4.485391 0.7145471 3.422158
## 2    4 4.023657 0.7598116 3.161119
## 3    7 3.915686 0.7693547 3.073788
```