Master File

Joanna Duran, Nikhil Gupta, Max Moro

2019-02-16 10:54:08

# User Inputs

output.var = params$output.var   
  
transform.abs = FALSE  
log.pred = params$log.pred  
norm.pred = FALSE  
algo.forward.caret = params$algo.forward.caret  
algo.backward.caret = params$algo.backward.caret  
algo.stepwise.caret = params$algo.stepwise.caret  
algo.LASSO.caret = params$algo.LASSO.caret  
algo.LARS.caret = params$algo.LARS.caret  
message("Parameters used for training/prediction: ")

## Parameters used for training/prediction:

str(params)

## List of 7  
## $ output.var : chr "y3"  
## $ log.pred : logi TRUE  
## $ algo.forward.caret : logi TRUE  
## $ algo.backward.caret: logi TRUE  
## $ algo.stepwise.caret: logi TRUE  
## $ algo.LASSO.caret : logi TRUE  
## $ algo.LARS.caret : logi TRUE

# Setup Labels  
output.var.tr = if (log.pred == TRUE) paste0(output.var,'.log') else output.var.tr = output.var

# Loading Data

feat = read.csv('../../Data/features\_highprec.csv')  
labels = read.csv('../../Data/labels.csv')  
predictors = names(dplyr::select(feat,-JobName))  
data.ori = inner\_join(feat,labels,by='JobName')  
#data.ori = inner\_join(feat,select\_at(labels,c('JobName',output.var)),by='JobName')

# Data validation

cc = complete.cases(data.ori)  
data.notComplete = data.ori[! cc,]  
data = data.ori[cc,] %>% select\_at(c(predictors,output.var,'JobName'))  
message('Original cases: ',nrow(data.ori))

## Original cases: 10000

message('Non-Complete cases: ',nrow(data.notComplete))

## Non-Complete cases: 3020

message('Complete cases: ',nrow(data))

## Complete cases: 6980

summary(dplyr::select\_at(data,c('JobName',output.var)))

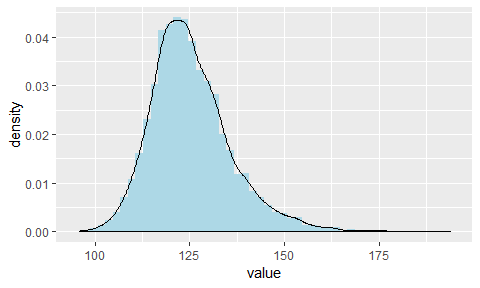
## JobName y3   
## Job\_00001: 1 Min. : 95.91   
## Job\_00002: 1 1st Qu.:118.29   
## Job\_00003: 1 Median :124.03   
## Job\_00004: 1 Mean :125.40   
## Job\_00007: 1 3rd Qu.:131.06   
## Job\_00008: 1 Max. :193.73   
## (Other) :6974

# Output Variable

The Output Variable **y3** shows right skewness, so will proceed with a log transformation

## Histogram

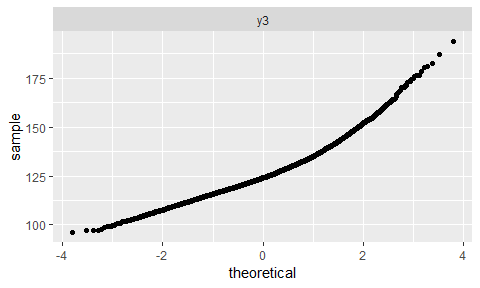
df=gather(select\_at(data,output.var))  
ggplot(df, aes(x=value)) +   
 geom\_histogram(aes(y=..density..),bins = 50,fill='light blue') +   
 geom\_density()



#stat\_function(fun = dnorm, n = 100, args = list(mean = mean(df$value), sd = sd(df$value)))

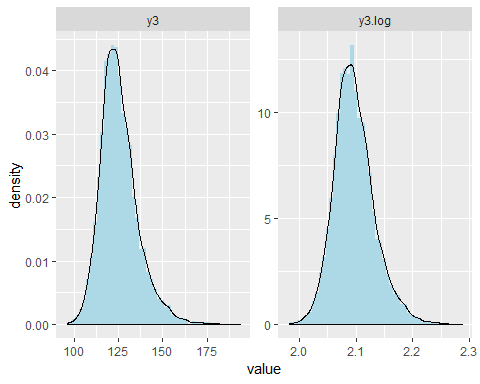
## QQPlot

ggplot(gather(select\_at(data,output.var)), aes(sample=value)) +   
 stat\_qq() +   
 facet\_wrap(~key, scales = 'free',ncol=4)

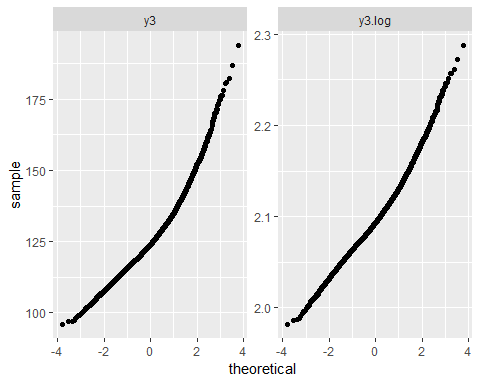


## Trasformation of Output Variable from y3 to y3.log

if(log.pred==TRUE) data[[output.var.tr]] = log(data[[output.var]],10) else  
 data[[output.var.tr]] = data[[output.var]]  
df=gather(select\_at(data,c(output.var,output.var.tr)))  
ggplot(df, aes(value)) +   
 geom\_histogram(aes(y=..density..),bins = 50,fill='light blue') +   
 geom\_density() +   
 # stat\_function(fun = dnorm, n = 100, args = list(mean = mean(df$value), sd = sd(df$value)))   
 facet\_wrap(~key, scales = 'free',ncol=2)



ggplot(gather(select\_at(data,c(output.var,output.var.tr))), aes(sample=value)) +   
 stat\_qq() +   
 facet\_wrap(~key, scales = 'free',ncol=4)



## Best Normalizator y3

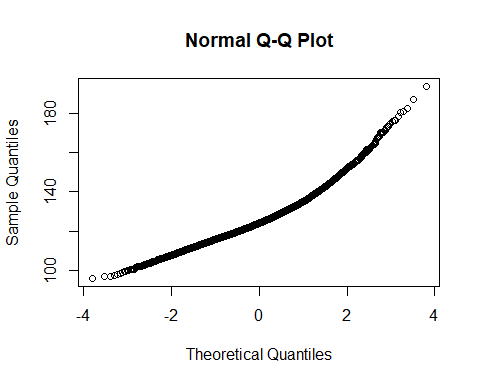
Normalization of **y3** using bestNormalize package. (suggested orderNorm) This is cool, but I think is too far for the objective of the project

t=bestNormalize::bestNormalize(data[[output.var]])

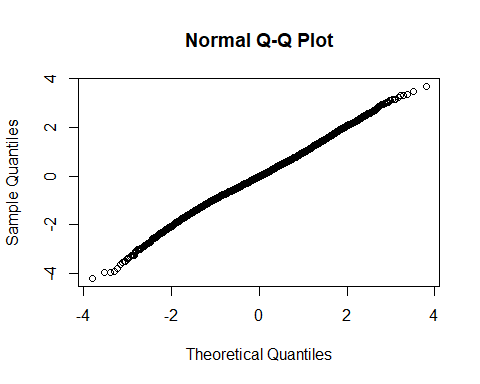
t

## Best Normalizing transformation with 6980 Observations  
## Estimated Normality Statistics (Pearson P / df, lower => more normal):  
## - No transform: 2.8969   
## - Box-Cox: 1.4044   
## - Log\_b(x+a): 1.9131   
## - sqrt(x+a): 2.3616   
## - exp(x): 749.0099   
## - arcsinh(x): 1.9131   
## - Yeo-Johnson: 1.1116   
## - orderNorm: 1.1278   
## Estimation method: Out-of-sample via CV with 10 folds and 5 repeats  
##   
## Based off these, bestNormalize chose:  
## Standardized Yeo-Johnson Transformation with 6980 nonmissing obs.:  
## Estimated statistics:  
## - lambda = -1.998639   
## - mean (before standardization) = 0.5003083   
## - sd (before standardization) = 5.108542e-06

qqnorm(data[[output.var]])



qqnorm(predict(t))



**orderNorm()** is a rank-based procedure by which the values of a vector are mapped to their percentile, which is then mapped to the same percentile of the normal distribution. Without the presence of ties, this essentially guarantees that the transformation leads to a uniform distribution

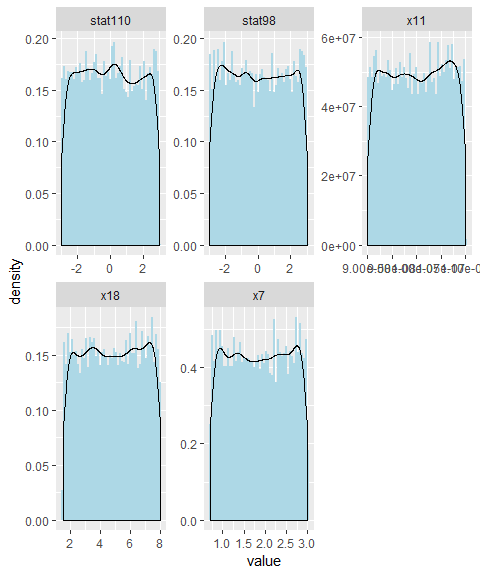
# Predictors

All predictors show a **Fat-Tail** situation, where the two tails are very tall, and a low distribution around the mean. The orderNorm transformation can help (see [Best Normalizator] section)

## Interesting Predictors

Histograms

cols = c('x11','x18','stat98','x7','stat110')  
df=gather(select\_at(data,cols))  
ggplot(df, aes(value)) +   
 geom\_histogram(aes(y=..density..),bins = 50,fill='light blue') +   
 geom\_density() +   
 # stat\_function(fun = dnorm, n = 100, args = list(mean = mean(df$value), sd = sd(df$value)))   
 facet\_wrap(~key, scales = 'free',ncol=3)



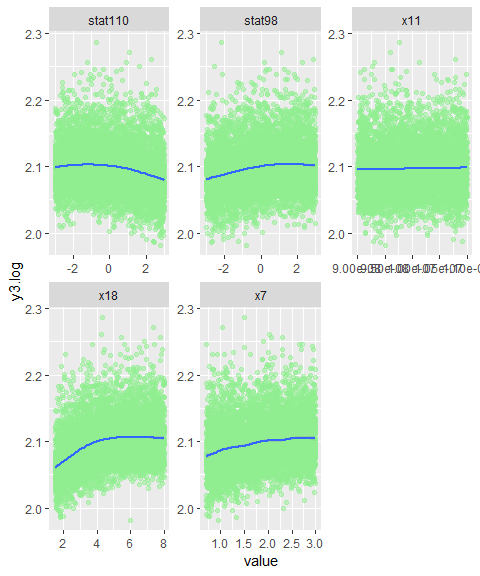
# ggplot(gather(select\_at(data,cols)), aes(sample=value)) +   
# stat\_qq()+  
# facet\_wrap(~key, scales = 'free',ncol=2)  
  
lapply(select\_at(data,cols),summary)

## $x11  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 9.000e-08 9.494e-08 1.001e-07 1.001e-07 1.052e-07 1.100e-07   
##   
## $x18  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 1.500 3.147 4.769 4.772 6.418 7.999   
##   
## $stat98  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -2.998619 -1.551882 -0.015993 -0.005946 1.528405 2.999499   
##   
## $x7  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.700 1.266 1.854 1.852 2.446 3.000   
##   
## $stat110  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -2.999543 -1.496865 -0.002193 -0.004129 1.504273 2.999563

Scatter plot vs. output variable \*\*y3.log

d = gather(dplyr::select\_at(data,c(cols,output.var.tr)),key=target,value=value,-!!output.var.tr)  
ggplot(data=d, aes\_string(x='value',y=output.var.tr)) +   
 geom\_point(color='light green',alpha=0.5) +   
 geom\_smooth() +   
 facet\_wrap(~target, scales = 'free',ncol=3)

## `geom\_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'

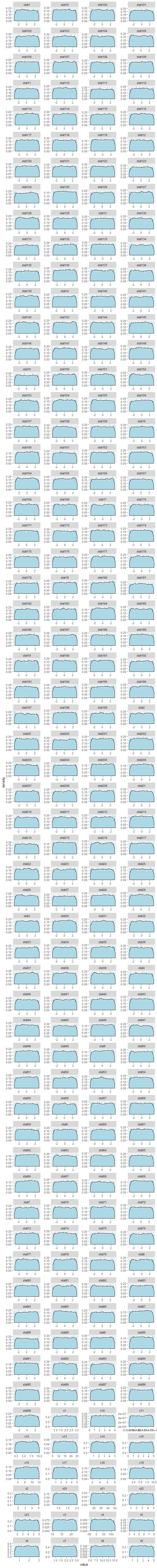


## All Predictors

### Histograms

All indicators have a strong indication of **Fat-Tails**

df=gather(select\_at(data,predictors))  
ggplot(df, aes(value)) +   
 geom\_histogram(aes(y=..density..),bins = 50,fill='light blue') +   
 geom\_density() +   
 # stat\_function(fun = dnorm, n = 100, args = list(mean = mean(df$value), sd = sd(df$value)))   
 facet\_wrap(~key, scales = 'free',ncol=4)



# Correlations

## With Output Variable

#chart.Correlation(select(data,-JobName), pch=21)  
t=as.data.frame(round(cor(dplyr::select(data,-one\_of(output.var.tr,'JobName'))  
 ,select\_at(data,output.var.tr)),4)) %>%  
 rownames\_to\_column(var='variable') %>% filter(variable != !!output.var) %>% arrange(-y3.log)  
#DT::datatable(t)  
message("Top Positive")

## Top Positive

kable(head(arrange(t,desc(y3.log)),20))

|  |  |
| --- | --- |
| variable | y3.log |
| x18 | 0.3120 |
| x7 | 0.2091 |
| stat98 | 0.1784 |
| x9 | 0.1127 |
| x17 | 0.0611 |
| x16 | 0.0489 |
| x10 | 0.0472 |
| x21 | 0.0412 |
| x11 | 0.0322 |
| x8 | 0.0318 |
| stat156 | 0.0287 |
| stat23 | 0.0234 |
| stat100 | 0.0206 |
| stat144 | 0.0203 |
| stat59 | 0.0202 |
| stat60 | 0.0199 |
| stat195 | 0.0199 |
| stat141 | 0.0194 |
| stat73 | 0.0192 |
| stat197 | 0.0185 |

message("Top Negative")

## Top Negative

kable(head(arrange(t,y3.log),20))

|  |  |
| --- | --- |
| variable | y3.log |
| stat110 | -0.1594 |
| x4 | -0.0603 |
| stat13 | -0.0345 |
| stat41 | -0.0345 |
| stat14 | -0.0317 |
| stat149 | -0.0309 |
| stat113 | -0.0279 |
| stat4 | -0.0248 |
| stat106 | -0.0236 |
| stat146 | -0.0236 |
| stat186 | -0.0217 |
| stat91 | -0.0210 |
| stat214 | -0.0209 |
| stat5 | -0.0207 |
| stat22 | -0.0202 |
| stat39 | -0.0202 |
| stat175 | -0.0194 |
| stat187 | -0.0193 |
| stat128 | -0.0192 |
| stat37 | -0.0191 |

## Between All Variables

#chart.Correlation(select(data,-JobName), pch=21)  
t=as.data.frame(round(cor(dplyr::select(data,-one\_of('JobName'))),4))  
#DT::datatable(t,options=list(scrollX=T))  
message("Showing only 10 variables")

## Showing only 10 variables

kable(t[1:10,1:10])

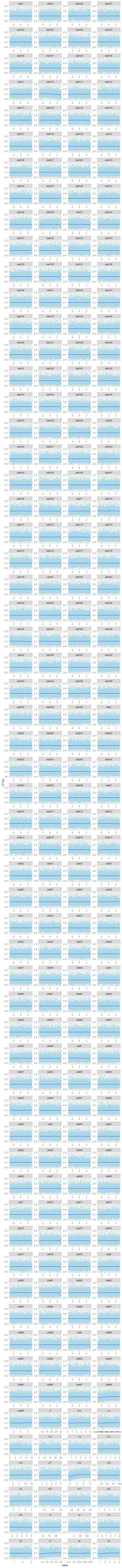
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | x1 | x2 | x3 | x4 | x5 | x6 | x7 | x8 | x9 | x10 |
| x1 | 1.0000 | 0.0034 | -0.0028 | 0.0085 | 0.0068 | 0.0159 | 0.0264 | -0.0012 | 0.0142 | 0.0013 |
| x2 | 0.0034 | 1.0000 | -0.0057 | 0.0004 | -0.0094 | -0.0101 | 0.0089 | 0.0078 | 0.0049 | -0.0214 |
| x3 | -0.0028 | -0.0057 | 1.0000 | 0.0029 | 0.0046 | 0.0006 | -0.0105 | -0.0002 | 0.0167 | -0.0137 |
| x4 | 0.0085 | 0.0004 | 0.0029 | 1.0000 | -0.0059 | 0.0104 | 0.0098 | 0.0053 | 0.0061 | -0.0023 |
| x5 | 0.0068 | -0.0094 | 0.0046 | -0.0059 | 1.0000 | 0.0016 | -0.0027 | 0.0081 | 0.0259 | -0.0081 |
| x6 | 0.0159 | -0.0101 | 0.0006 | 0.0104 | 0.0016 | 1.0000 | 0.0200 | -0.0157 | 0.0117 | -0.0072 |
| x7 | 0.0264 | 0.0089 | -0.0105 | 0.0098 | -0.0027 | 0.0200 | 1.0000 | -0.0018 | -0.0069 | -0.0221 |
| x8 | -0.0012 | 0.0078 | -0.0002 | 0.0053 | 0.0081 | -0.0157 | -0.0018 | 1.0000 | 0.0142 | -0.0004 |
| x9 | 0.0142 | 0.0049 | 0.0167 | 0.0061 | 0.0259 | 0.0117 | -0.0069 | 0.0142 | 1.0000 | 0.0149 |
| x10 | 0.0013 | -0.0214 | -0.0137 | -0.0023 | -0.0081 | -0.0072 | -0.0221 | -0.0004 | 0.0149 | 1.0000 |

## Scatter Plots with Output Variable

Scatter plots with all predictors and the output variable (y3.log)

d = gather(dplyr::select\_at(data,c(predictors,output.var.tr)),key=target,value=value,-!!output.var.tr)  
ggplot(data=d, aes\_string(x='value',y=output.var.tr)) +   
 geom\_point(color='light blue',alpha=0.5) +   
 geom\_smooth() +   
 facet\_wrap(~target, scales = 'free',ncol=4)

## `geom\_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'



## Multicollinearity - VIF

No Multicollinearity among predictors

Showing Top predictor by VIF Value

vifDF = usdm::vif(select\_at(data,predictors)) %>% arrange(desc(VIF))  
head(vifDF,15)

## Variables VIF  
## 1 stat153 1.060498  
## 2 stat186 1.060485  
## 3 stat11 1.059958  
## 4 stat129 1.059556  
## 5 stat142 1.059382  
## 6 x14 1.059104  
## 7 stat20 1.058742  
## 8 stat141 1.058514  
## 9 stat45 1.058514  
## 10 stat80 1.058367  
## 11 stat15 1.058318  
## 12 stat31 1.058295  
## 13 stat171 1.058217  
## 14 stat14 1.058106  
## 15 x13 1.058054

# Feature Eng

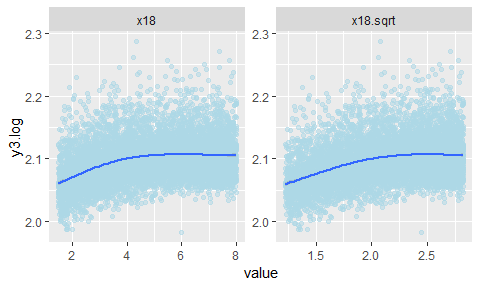
* Square Root transformation for **x18**

data.tr=data %>%  
 mutate(x18.sqrt = sqrt(x18))   
cols=c('x18','x18.sqrt')

## Comparing Pre and Post Transformation Density Plots

# ggplot(gather(select\_at(data.tr,cols)), aes(value)) +   
# geom\_histogram(aes(y=..density..),bins = 50,fill='light blue') +   
# geom\_density() +   
# facet\_wrap(~key, scales = 'free',ncol=4)  
  
d = gather(dplyr::select\_at(data.tr,c(cols,output.var.tr)),key=target,value=value,-!!output.var.tr)  
ggplot(data=d, aes\_string(x='value',y=output.var.tr)) +   
 geom\_point(color='light blue',alpha=0.5) +   
 geom\_smooth() +   
 facet\_wrap(~target, scales = 'free',ncol=4)

## `geom\_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'



#removing unwanted variables  
data.tr=data.tr %>%  
 dplyr::select\_at(names(data.tr)[! names(data.tr) %in% c('x18','y3','JobName')])  
  
data=data.tr  
label.names=output.var.tr

# Modeling

## Train Test Split

data = data[sample(nrow(data)),] # randomly shuffle data  
split = sample.split(data[,label.names], SplitRatio = 0.8)  
  
data.train = subset(data, split == TRUE)  
data.test = subset(data, split == FALSE)

## Common Functions

plot.diagnostics <- function(model, train) {  
 plot(model)  
   
 residuals = resid(model) # Plotted above in plot(lm.out)  
 r.standard = rstandard(model)  
 r.student = rstudent(model)  
   
 df = data.frame(x=predict(model,train),y=r.student)  
 p=ggplot(data=df,aes(x=x,y=y)) +  
 geom\_point(color='blue',alpha=0.5,shape=20,size=2) +  
 geom\_hline(yintercept = c(-2,0,2),size=1)+  
 ylab("Student Residuals") +  
 xlab("Predicted Values")+  
 ggtitle("Standardized Residual Plot")  
 plot(p)  
   
 # df = data.frame(x=predict(model,train),y=r.standard)  
 # p=ggplot(data=df,aes(x=x,y=y)) +  
 # geom\_point(color='blue',alpha=0.5,shape=20,size=2) +  
 # geom\_hline(yintercept = c(-2,0,2),size=1)+  
 # ylab("Standardized Residuals") +  
 # xlab("Predicted Values")+  
 # ggtitle("Student Residual Plot")  
 # plot(p)  
 # Histogram  
 df=data.frame(r.student)  
 p=ggplot(data=df,aes(r.student)) +  
 geom\_histogram(aes(y=..density..),bins = 50,fill='blue',alpha=0.6) +   
 stat\_function(fun = dnorm, n = 100, args = list(mean = 0, sd = 1)) +  
 ylab("Density")+  
 xlab("Studentized Residuals")+  
 ggtitle("Distribution of Studentized Residuals")  
 plot(p)  
 # http://www.stat.columbia.edu/~martin/W2024/R7.pdf  
 # Influential plots  
 inf.meas = influence.measures(model)  
 # print (summary(inf.meas)) # too much data  
   
 # Leverage plot  
 lev = hat(model.matrix(model))  
 df=tibble::rownames\_to\_column(as.data.frame(lev),'id')  
 p=ggplot(data=df,aes(x=as.numeric(id),y=lev)) +  
 geom\_point(color='blue',alpha=0.5,shape=20,size=2) +  
 ylab('Leverage - check') +   
 xlab('Index')  
 plot(p)  
 # Cook's Distance  
 cd = cooks.distance(model)  
 df=tibble::rownames\_to\_column(as.data.frame(cd),'id')  
 p=ggplot(data=df,aes(x=as.numeric(id),y=cd)) +  
 geom\_point(color='blue',alpha=0.5,shape=20,size=2) +  
 geom\_text(data=filter(df,cd>15/nrow(train)),aes(label=id),check\_overlap=T,size=3,vjust=-.5)+  
 ylab('Cooks distances') +   
 geom\_hline(yintercept = c(4/nrow(train),0),size=1)+  
 xlab('Index')  
 plot(p)  
 print (paste("Number of data points that have Cook's D > 4/n: ", length(cd[cd > 4/nrow(train)]), sep = ""))   
 print (paste("Number of data points that have Cook's D > 1: ", length(cd[cd > 1]), sep = ""))   
 return(cd)  
}  
  
# function to set up random seeds  
# Based on http://jaehyeon-kim.github.io/2015/05/Setup-Random-Seeds-on-Caret-Package.html   
setCaretSeeds <- function(method = "cv", numbers = 1, repeats = 1, tunes = NULL, seed = 1701) {  
 #B is the number of resamples and integer vector of M (numbers + tune length if any)  
 B <- if (method == "cv") numbers  
 else if(method == "repeatedcv") numbers \* repeats  
 else NULL  
 if(is.null(length)) {  
 seeds <- NULL  
 } else {  
 set.seed(seed = seed)  
 seeds <- vector(mode = "list", length = B)  
 seeds <- lapply(seeds, function(x) sample.int(n = 1000000  
 , size = numbers + ifelse(is.null(tunes), 0, tunes)))  
 seeds[[length(seeds) + 1]] <- sample.int(n = 1000000, size = 1)  
 }  
 # return seeds  
 seeds  
}  
  
train.caret.glmselect = function(formula, data, method  
 ,subopt = NULL, feature.names  
 , train.control = NULL, tune.grid = NULL, pre.proc = NULL){  
   
 if(is.null(train.control)){  
 train.control <- trainControl(method = "cv"  
 ,number = 10  
 ,seeds = setCaretSeeds(method = "cv"  
 , numbers = 10  
 , seed = 1701)  
 ,search = "grid"  
 ,verboseIter = TRUE  
 ,allowParallel = TRUE  
 )  
 }  
   
 if(is.null(tune.grid)){  
 if (method == 'leapForward' | method == 'leapBackward' | method == 'leapSeq'){  
 tune.grid = data.frame(nvmax = 1:length(feature.names))  
 }  
 if (method == 'glmnet' && subopt == 'LASSO'){  
 # Will only show 1 Lambda value during training, but that is OK  
 # https://stackoverflow.com/questions/47526544/why-need-to-tune-lambda-with-carettrain-method-glmnet-and-cv-glmnet  
 # Another option for LASSO is this: https://github.com/topepo/caret/blob/master/RegressionTests/Code/lasso.R  
 lambda = 10^seq(-2,0, length =100)  
 alpha = c(1)  
 tune.grid = expand.grid(alpha = alpha,lambda = lambda)  
 }  
 if (method == 'lars'){  
 # https://github.com/topepo/caret/blob/master/RegressionTests/Code/lars.R  
 fraction = seq(0, 1, length = 100)  
 tune.grid = expand.grid(fraction = fraction)  
 pre.proc = c("center", "scale")   
 }  
 }  
   
 # http://sshaikh.org/2015/05/06/parallelize-machine-learning-in-r-with-multi-core-cpus/  
 cl <- makeCluster(ceiling(detectCores()\*0.85)) # use 75% of cores only, leave rest for other tasks  
 registerDoParallel(cl)  
  
 set.seed(1)   
 # note that the seed has to actually be set just before this function is called  
 # settign is above just not ensure reproducibility for some reason  
 model.caret <- caret::train(formula  
 , data = data  
 , method = method  
 , tuneGrid = tune.grid  
 , trControl = train.control  
 , preProc = pre.proc  
 )  
   
 stopCluster(cl)  
 registerDoSEQ() # register sequential engine in case you are not using this function anymore  
   
 if (method == 'leapForward' | method == 'leapBackward' | method == 'leapSeq'){  
 print("All models results")  
 print(model.caret$results) # all model results  
 print("Best Model")  
 print(model.caret$bestTune) # best model  
 model = model.caret$finalModel  
  
 # Metrics Plot   
 dataPlot = model.caret$results %>%  
 gather(key='metric',value='value',-nvmax) %>%  
 dplyr::filter(metric %in% c('MAE','RMSE','Rsquared'))  
 metricsPlot = ggplot(data=dataPlot,aes(x=nvmax,y=value) ) +  
 geom\_line(color='lightblue4') +  
 geom\_point(color='blue',alpha=0.7,size=.9) +  
 facet\_wrap(~metric,ncol=2,scales='free')+  
 theme\_light()  
 plot(metricsPlot)  
   
 # Residuals Plot  
 # leap function does not support studentized residuals  
 dataPlot=data.frame(pred=predict(model.caret,data),res=resid(model.caret))  
 residPlot = ggplot(dataPlot,aes(x=pred,y=res)) +  
 geom\_point(color='light blue',alpha=0.7) +  
 geom\_smooth(method="lm")+  
 theme\_light()  
 plot(residPlot)  
   
 residHistogram = ggplot(dataPlot,aes(x=res)) +  
 geom\_histogram(aes(y=..density..),fill='light blue',alpha=1) +  
 #geom\_density(color='lightblue4') +   
 stat\_function(fun = dnorm, n = 100, args = list(mean = mean(dataPlot$res)  
 , sd = sd(dataPlot$res)),color='lightblue4')   
 theme\_light()  
 plot(residHistogram)  
 id = rownames(model.caret$bestTune)   
 # Provides the coefficients of the best model  
 # regsubsets doens return a full model (see documentation of regsubset), so we need to recalcualte themodel  
 # https://stackoverflow.com/questions/13063762/how-to-obtain-a-lm-object-from-regsubsets  
 print("Coefficients of final model:")  
 coefs <- coef(model, id=id)  
 #calculate the model to the the coef intervals  
 nams <- names(coefs)  
 nams <- nams[!nams %in% "(Intercept)"]  
 response <- as.character(formula[[2]])  
 form <- as.formula(paste(response, paste(nams, collapse = " + "), sep = " ~ "))  
 mod <- lm(form, data = data)  
 #coefs  
 #coef(mod)  
 print(car::Confint(mod))  
 return(list(model = model,id = id, residPlot = residPlot, residHistogram=residHistogram  
 ,modelLM=mod))  
 }  
 if (method == 'glmnet' && subopt == 'LASSO'){  
 print(model.caret)  
 print(plot(model.caret))  
 print(model.caret$bestTune)  
   
 print(model.caret$results)  
 model=model.caret$finalModel  
 # Metrics Plot   
 dataPlot = model.caret$results %>%  
 gather(key='metric',value='value',-lambda) %>%  
 dplyr::filter(metric %in% c('MAE','RMSE','Rsquared'))  
 metricsPlot = ggplot(data=dataPlot,aes(x=lambda,y=value) ) +  
 geom\_line(color='lightblue4') +  
 geom\_point(color='blue',alpha=0.7,size=.9) +  
 facet\_wrap(~metric,ncol=2,scales='free')+  
 theme\_light()  
 plot(metricsPlot)  
   
 # Residuals Plot   
 dataPlot=data.frame(pred=predict(model.caret,data),res=resid(model.caret))  
 residPlot = ggplot(dataPlot,aes(x=pred,y=res)) +  
 geom\_point(color='light blue',alpha=0.7) +  
 geom\_smooth(method="lm")+  
 theme\_light()  
 plot(residPlot)  
  
 residHistogram = ggplot(dataPlot,aes(x=res)) +  
 geom\_histogram(aes(y=..density..),fill='light blue',alpha=1) +  
 #geom\_density(color='lightblue4') +  
 stat\_function(fun = dnorm, n = 100, args = list(mean = mean(dataPlot$res)  
 , sd = sd(dataPlot$res)),color='lightblue4')   
 theme\_light()  
 plot(residHistogram)  
   
 print("Coefficients")   
 #no interval for glmnet: https://stackoverflow.com/questions/39750965/confidence-intervals-for-ridge-regression  
 t=coef(model,s=model.caret$bestTune$lambda)  
 model.coef = t[which(t[,1]!=0),]  
 print(as.data.frame(model.coef))  
 id = NULL # not really needed but added for consistency  
 return(list(model = model.caret,id = id, residPlot = residPlot, metricsPlot=metricsPlot ))  
 }  
 if (method == 'lars'){  
 print(model.caret)  
 print(plot(model.caret))  
 print(model.caret$bestTune)  
   
 # Metrics Plot  
 dataPlot = model.caret$results %>%  
 gather(key='metric',value='value',-fraction) %>%  
 dplyr::filter(metric %in% c('MAE','RMSE','Rsquared'))  
 metricsPlot = ggplot(data=dataPlot,aes(x=fraction,y=value) ) +  
 geom\_line(color='lightblue4') +  
 geom\_point(color='blue',alpha=0.7,size=.9) +  
 facet\_wrap(~metric,ncol=2,scales='free')+  
 theme\_light()  
 plot(metricsPlot)  
   
 # Residuals Plot  
 dataPlot=data.frame(pred=predict(model.caret,data),res=resid(model.caret))  
 residPlot = ggplot(dataPlot,aes(x=pred,y=res)) +  
 geom\_point(color='light blue',alpha=0.7) +  
 geom\_smooth(method="lm")+  
 theme\_light()  
 plot(residPlot)  
  
 residHistogram = ggplot(dataPlot,aes(x=res)) +  
 geom\_histogram(aes(y=..density..),fill='light blue',alpha=1) +  
 #geom\_density(color='lightblue4') +   
 stat\_function(fun = dnorm, n = 100, args = list(mean = mean(dataPlot$res)  
 , sd = sd(dataPlot$res)),color='lightblue4')   
 theme\_light()  
 plot(residHistogram)  
   
 print("Coefficients")   
 t=coef(model.caret$finalModel,s=model.caret$bestTune$fraction,mode='fraction')  
 model.coef = t[which(t!=0)]  
 print(model.coef)  
 id = NULL # not really needed but added for consistency  
 return(list(model = model.caret,id = id, residPlot = residPlot, residHistogram=residHistogram))  
 }  
}  
  
# https://stackoverflow.com/questions/48265743/linear-model-subset-selection-goodness-of-fit-with-k-fold-cross-validation  
# changed slightly since call[[2]] was just returning "formula" without actually returnign the value in formula  
predict.regsubsets <- function(object, newdata, id, formula, ...) {  
 #form <- as.formula(object$call[[2]])  
 mat <- model.matrix(formula, newdata) # adds intercept and expands any interaction terms  
 coefi <- coef(object, id = id)  
 xvars <- names(coefi)  
 return(mat[,xvars]%\*%coefi)  
}  
   
test.model = function(model, test, level=0.95  
 ,draw.limits = FALSE, good = 0.1, ok = 0.15  
 ,method = NULL, subopt = NULL  
 ,id = NULL, formula, feature.names, label.names  
 ,transformation = NULL){  
 ## if using caret for glm select equivalent functionality,   
 ## need to pass formula (full is ok as it will select subset of variables from there)  
 if (is.null(method)){  
 pred = predict(model, newdata=test, interval="confidence", level = level)   
 }  
   
 if (method == 'leapForward' | method == 'leapBackward' | method == 'leapSeq'){  
 pred = predict.regsubsets(model, newdata = test, id = id, formula = formula)  
 }  
   
 if (method == 'glmnet' && subopt == 'LASSO'){  
 xtest = as.matrix(test[,feature.names])   
 pred=as.data.frame(predict(model, xtest))  
 }  
   
 if (method == 'lars'){  
 pred=as.data.frame(predict(model, newdata = test))  
 }  
   
 # Summary of predicted values  
 print ("Summary of predicted values: ")  
 print(summary(pred[,1]))  
  
 test.mse = mean((test[,label.names]-pred[,1])^2)  
 print (paste(method, subopt, "Test MSE:", test.mse, sep=" "))  
   
 if(log.pred == TRUE || norm.pred == TRUE){  
 # plot transformewd comparison first  
 df=data.frame(x=test[,label.names],y=pred[,1])  
 ggplot(df,aes(x=x,y=y)) +  
 geom\_point(color='blue',alpha=0.5,shape=20,size=2) +  
 geom\_abline(slope=1,intercept=0,color='black',size=1) +  
 #scale\_y\_continuous(limits=c(min(df),max(df)))+  
 xlab("Actual (Transformed)")+  
 ylab("Predicted (Transformed)")  
 }  
   
 if (log.pred == FALSE && norm.pred == FALSE){  
 x = test[,label.names]  
 y = pred[,1]  
 }  
 if (log.pred == TRUE){  
 x = 10^test[,label.names]  
 y = 10^pred[,1]   
 }  
 if (norm.pred == TRUE){  
 x = predict(transformation, test[,label.names], inverse = TRUE)  
 y = predict(transformation, pred[,1], inverse = TRUE)  
 }  
  
 df=data.frame(x,y)  
 ggplot(df,aes(x,y)) +  
 geom\_point(color='blue',alpha=0.5,shape=20,size=2) +  
 geom\_abline(slope=c(1+good,1-good,1+ok,1-ok)  
 ,intercept=rep(0,4),color=c('dark green','dark green','dark red','dark red'),size=1,alpha=0.8) +  
 #scale\_y\_continuous(limits=c(min(df),max(df)))+  
 xlab("Actual")+  
 ylab("Predicted")   
   
   
}

## Setup Formulae

n <- names(data.train)  
 formula <- as.formula(paste(paste(n[n %in% label.names], collapse = " + ")  
 ," ~", paste(n[!n %in% label.names], collapse = " + ")))   
  
grand.mean.formula = as.formula(paste(paste(n[n %in% label.names], collapse = " + ")," ~ 1"))  
  
print(formula)

## y3.log ~ x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8 + x9 + x10 + x11 +   
## x12 + x13 + x14 + x15 + x16 + x17 + x19 + x20 + x21 + x22 +   
## x23 + stat1 + stat2 + stat3 + stat4 + stat5 + stat6 + stat7 +   
## stat8 + stat9 + stat10 + stat11 + stat12 + stat13 + stat14 +   
## stat15 + stat16 + stat17 + stat18 + stat19 + stat20 + stat21 +   
## stat22 + stat23 + stat24 + stat25 + stat26 + stat27 + stat28 +   
## stat29 + stat30 + stat31 + stat32 + stat33 + stat34 + stat35 +   
## stat36 + stat37 + stat38 + stat39 + stat40 + stat41 + stat42 +   
## stat43 + stat44 + stat45 + stat46 + stat47 + stat48 + stat49 +   
## stat50 + stat51 + stat52 + stat53 + stat54 + stat55 + stat56 +   
## stat57 + stat58 + stat59 + stat60 + stat61 + stat62 + stat63 +   
## stat64 + stat65 + stat66 + stat67 + stat68 + stat69 + stat70 +   
## stat71 + stat72 + stat73 + stat74 + stat75 + stat76 + stat77 +   
## stat78 + stat79 + stat80 + stat81 + stat82 + stat83 + stat84 +   
## stat85 + stat86 + stat87 + stat88 + stat89 + stat90 + stat91 +   
## stat92 + stat93 + stat94 + stat95 + stat96 + stat97 + stat98 +   
## stat99 + stat100 + stat101 + stat102 + stat103 + stat104 +   
## stat105 + stat106 + stat107 + stat108 + stat109 + stat110 +   
## stat111 + stat112 + stat113 + stat114 + stat115 + stat116 +   
## stat117 + stat118 + stat119 + stat120 + stat121 + stat122 +   
## stat123 + stat124 + stat125 + stat126 + stat127 + stat128 +   
## stat129 + stat130 + stat131 + stat132 + stat133 + stat134 +   
## stat135 + stat136 + stat137 + stat138 + stat139 + stat140 +   
## stat141 + stat142 + stat143 + stat144 + stat145 + stat146 +   
## stat147 + stat148 + stat149 + stat150 + stat151 + stat152 +   
## stat153 + stat154 + stat155 + stat156 + stat157 + stat158 +   
## stat159 + stat160 + stat161 + stat162 + stat163 + stat164 +   
## stat165 + stat166 + stat167 + stat168 + stat169 + stat170 +   
## stat171 + stat172 + stat173 + stat174 + stat175 + stat176 +   
## stat177 + stat178 + stat179 + stat180 + stat181 + stat182 +   
## stat183 + stat184 + stat185 + stat186 + stat187 + stat188 +   
## stat189 + stat190 + stat191 + stat192 + stat193 + stat194 +   
## stat195 + stat196 + stat197 + stat198 + stat199 + stat200 +   
## stat201 + stat202 + stat203 + stat204 + stat205 + stat206 +   
## stat207 + stat208 + stat209 + stat210 + stat211 + stat212 +   
## stat213 + stat214 + stat215 + stat216 + stat217 + x18.sqrt

print(grand.mean.formula)

## y3.log ~ 1

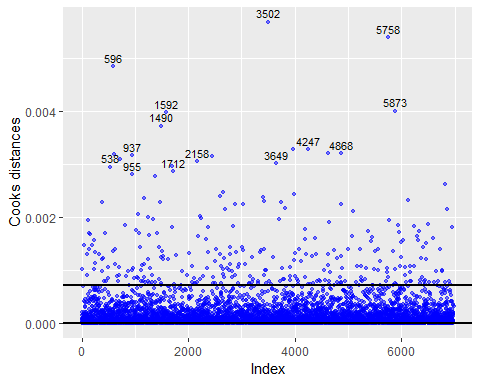
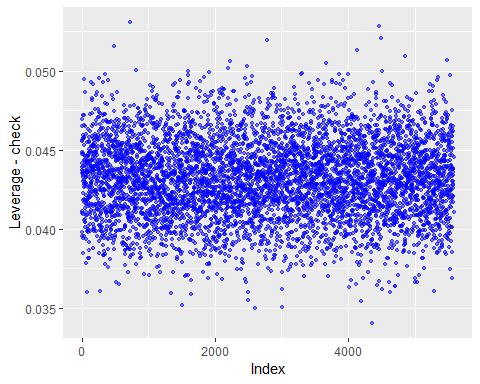
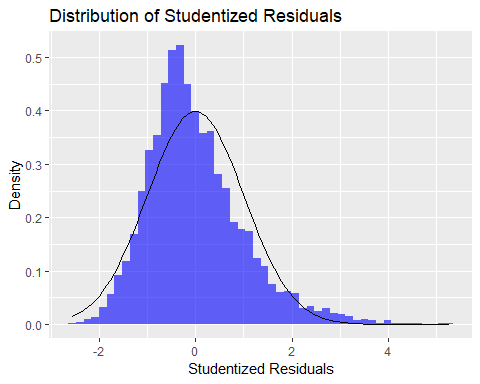
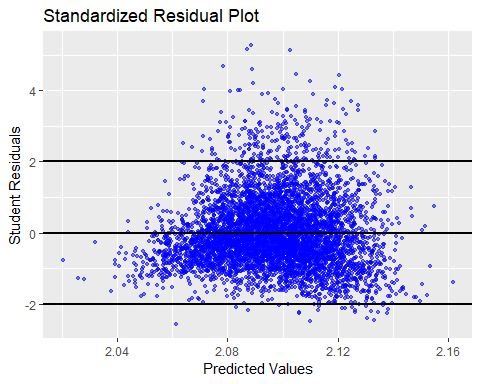
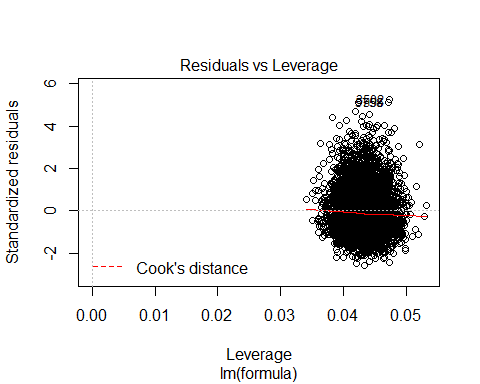
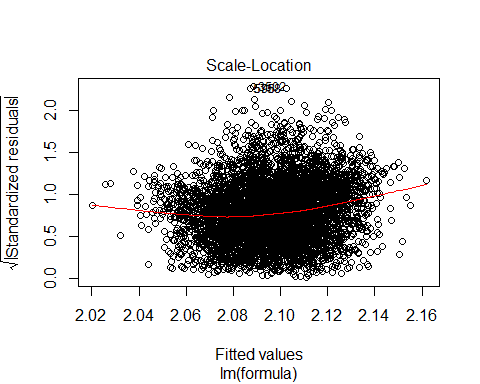
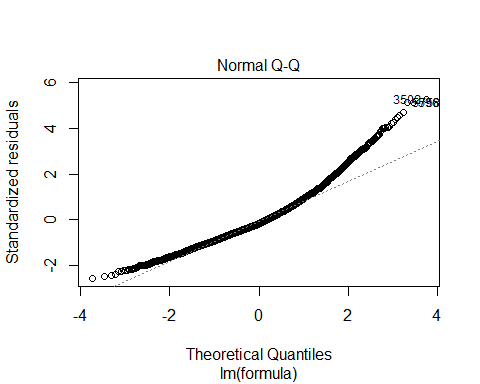
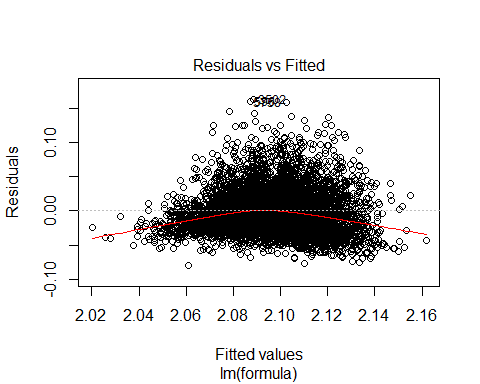
# Update feature.names because we may have transformed some features  
feature.names = n[!n %in% label.names]

## Full Model

model.full = lm(formula , data.train)  
summary(model.full)

##   
## Call:  
## lm(formula = formula, data = data.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.079266 -0.020590 -0.004983 0.016087 0.162401   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.972e+00 9.570e-03 206.035 < 2e-16 \*\*\*  
## x1 -1.548e-04 6.582e-04 -0.235 0.814020   
## x2 8.251e-05 4.193e-04 0.197 0.843997   
## x3 2.804e-05 1.152e-04 0.243 0.807723   
## x4 -5.063e-05 9.139e-06 -5.540 3.17e-08 \*\*\*  
## x5 2.432e-04 2.984e-04 0.815 0.414971   
## x6 7.898e-04 5.981e-04 1.320 0.186763   
## x7 1.172e-02 6.441e-04 18.192 < 2e-16 \*\*\*  
## x8 2.877e-04 1.501e-04 1.916 0.055393 .   
## x9 3.231e-03 3.354e-04 9.634 < 2e-16 \*\*\*  
## x10 1.209e-03 3.127e-04 3.866 0.000112 \*\*\*  
## x11 2.036e+05 7.423e+04 2.743 0.006107 \*\*   
## x12 -2.322e-04 1.894e-04 -1.226 0.220303   
## x13 -1.358e-06 7.615e-05 -0.018 0.985774   
## x14 -5.951e-04 3.286e-04 -1.811 0.070157 .   
## x15 1.310e-04 3.099e-04 0.423 0.672477   
## x16 8.847e-04 2.159e-04 4.098 4.23e-05 \*\*\*  
## x17 1.433e-03 3.283e-04 4.366 1.29e-05 \*\*\*  
## x19 2.750e-04 1.668e-04 1.649 0.099216 .   
## x20 -1.206e-03 1.156e-03 -1.043 0.296916   
## x21 1.333e-04 4.259e-05 3.130 0.001759 \*\*   
## x22 -4.010e-04 3.492e-04 -1.148 0.250896   
## x23 7.105e-05 3.299e-04 0.215 0.829470   
## stat1 -1.714e-04 2.507e-04 -0.684 0.494269   
## stat2 3.484e-04 2.498e-04 1.395 0.163156   
## stat3 3.921e-04 2.502e-04 1.567 0.117126   
## stat4 -4.829e-04 2.522e-04 -1.914 0.055610 .   
## stat5 -1.238e-04 2.505e-04 -0.494 0.621130   
## stat6 -1.263e-04 2.517e-04 -0.502 0.615837   
## stat7 -9.080e-05 2.498e-04 -0.363 0.716247   
## stat8 1.741e-04 2.507e-04 0.694 0.487477   
## stat9 1.476e-04 2.495e-04 0.591 0.554358   
## stat10 -2.858e-04 2.510e-04 -1.139 0.254770   
## stat11 4.029e-06 2.518e-04 0.016 0.987236   
## stat12 1.267e-04 2.488e-04 0.509 0.610664   
## stat13 -5.566e-04 2.476e-04 -2.248 0.024611 \*   
## stat14 -1.042e-03 2.488e-04 -4.189 2.85e-05 \*\*\*  
## stat15 -4.326e-04 2.476e-04 -1.747 0.080679 .   
## stat16 -9.828e-05 2.490e-04 -0.395 0.693090   
## stat17 3.575e-05 2.467e-04 0.145 0.884781   
## stat18 -2.841e-04 2.497e-04 -1.138 0.255284   
## stat19 -1.425e-04 2.479e-04 -0.575 0.565350   
## stat20 -4.647e-04 2.510e-04 -1.851 0.064214 .   
## stat21 -1.227e-04 2.502e-04 -0.490 0.623872   
## stat22 -4.432e-04 2.509e-04 -1.766 0.077403 .   
## stat23 6.309e-04 2.493e-04 2.530 0.011423 \*   
## stat24 -4.257e-04 2.509e-04 -1.696 0.089907 .   
## stat25 -3.738e-04 2.490e-04 -1.501 0.133286   
## stat26 -4.046e-04 2.499e-04 -1.619 0.105476   
## stat27 1.686e-04 2.497e-04 0.675 0.499639   
## stat28 1.598e-05 2.509e-04 0.064 0.949240   
## stat29 2.278e-04 2.522e-04 0.903 0.366524   
## stat30 2.142e-04 2.523e-04 0.849 0.395865   
## stat31 -5.606e-05 2.521e-04 -0.222 0.824006   
## stat32 1.383e-06 2.526e-04 0.005 0.995631   
## stat33 -1.963e-04 2.493e-04 -0.787 0.431201   
## stat34 1.322e-04 2.487e-04 0.532 0.594996   
## stat35 -3.998e-04 2.504e-04 -1.597 0.110432   
## stat36 -7.413e-05 2.478e-04 -0.299 0.764849   
## stat37 -3.161e-04 2.532e-04 -1.248 0.211925   
## stat38 4.465e-04 2.502e-04 1.785 0.074375 .   
## stat39 -2.341e-04 2.494e-04 -0.939 0.347973   
## stat40 -1.602e-04 2.516e-04 -0.637 0.524286   
## stat41 -4.041e-04 2.492e-04 -1.622 0.104908   
## stat42 -1.432e-04 2.497e-04 -0.574 0.566303   
## stat43 -2.419e-04 2.518e-04 -0.961 0.336689   
## stat44 3.639e-04 2.509e-04 1.451 0.146915   
## stat45 -2.480e-04 2.500e-04 -0.992 0.321343   
## stat46 2.675e-04 2.506e-04 1.068 0.285749   
## stat47 1.484e-04 2.510e-04 0.591 0.554298   
## stat48 4.330e-04 2.507e-04 1.727 0.084218 .   
## stat49 2.140e-04 2.494e-04 0.858 0.390945   
## stat50 2.552e-04 2.473e-04 1.032 0.302160   
## stat51 2.617e-04 2.502e-04 1.046 0.295612   
## stat52 8.110e-05 2.517e-04 0.322 0.747254   
## stat53 -1.292e-04 2.519e-04 -0.513 0.607929   
## stat54 -4.016e-04 2.515e-04 -1.597 0.110374   
## stat55 1.906e-04 2.486e-04 0.767 0.443171   
## stat56 -1.852e-04 2.500e-04 -0.741 0.458818   
## stat57 -2.769e-05 2.470e-04 -0.112 0.910759   
## stat58 -7.465e-05 2.493e-04 -0.299 0.764597   
## stat59 6.226e-05 2.509e-04 0.248 0.804025   
## stat60 4.224e-04 2.510e-04 1.683 0.092419 .   
## stat61 -2.405e-04 2.501e-04 -0.962 0.336216   
## stat62 -3.864e-04 2.492e-04 -1.550 0.121132   
## stat63 5.741e-05 2.510e-04 0.229 0.819103   
## stat64 -4.148e-05 2.475e-04 -0.168 0.866928   
## stat65 -3.716e-04 2.508e-04 -1.482 0.138469   
## stat66 8.675e-05 2.520e-04 0.344 0.730643   
## stat67 1.311e-05 2.508e-04 0.052 0.958320   
## stat68 -1.881e-04 2.517e-04 -0.747 0.455020   
## stat69 -1.565e-06 2.504e-04 -0.006 0.995013   
## stat70 2.905e-04 2.488e-04 1.168 0.242870   
## stat71 -3.240e-05 2.476e-04 -0.131 0.895895   
## stat72 -5.328e-06 2.512e-04 -0.021 0.983074   
## stat73 2.302e-04 2.509e-04 0.917 0.359064   
## stat74 -1.157e-04 2.505e-04 -0.462 0.644099   
## stat75 -3.840e-04 2.512e-04 -1.529 0.126351   
## stat76 5.739e-05 2.505e-04 0.229 0.818819   
## stat77 -1.464e-04 2.493e-04 -0.587 0.557045   
## stat78 -1.955e-04 2.500e-04 -0.782 0.434267   
## stat79 -2.309e-04 2.518e-04 -0.917 0.359151   
## stat80 3.865e-04 2.513e-04 1.538 0.124138   
## stat81 3.191e-05 2.513e-04 0.127 0.898971   
## stat82 2.441e-04 2.493e-04 0.979 0.327659   
## stat83 -1.339e-04 2.515e-04 -0.532 0.594600   
## stat84 -2.458e-04 2.503e-04 -0.982 0.326235   
## stat85 1.490e-04 2.500e-04 0.596 0.551334   
## stat86 -3.543e-05 2.493e-04 -0.142 0.886964   
## stat87 -4.339e-04 2.514e-04 -1.726 0.084481 .   
## stat88 -1.462e-04 2.479e-04 -0.590 0.555267   
## stat89 -4.325e-04 2.484e-04 -1.741 0.081731 .   
## stat90 -1.479e-04 2.498e-04 -0.592 0.553667   
## stat91 -4.684e-04 2.501e-04 -1.873 0.061137 .   
## stat92 -2.614e-04 2.492e-04 -1.049 0.294302   
## stat93 -1.597e-04 2.519e-04 -0.634 0.526187   
## stat94 -1.985e-04 2.516e-04 -0.789 0.430240   
## stat95 2.101e-05 2.490e-04 0.084 0.932773   
## stat96 -2.908e-04 2.510e-04 -1.159 0.246683   
## stat97 1.684e-04 2.486e-04 0.677 0.498221   
## stat98 3.647e-03 2.475e-04 14.737 < 2e-16 \*\*\*  
## stat99 3.682e-04 2.513e-04 1.465 0.142850   
## stat100 2.185e-04 2.492e-04 0.877 0.380618   
## stat101 -1.694e-04 2.522e-04 -0.672 0.501811   
## stat102 1.677e-04 2.521e-04 0.665 0.505984   
## stat103 -4.262e-04 2.524e-04 -1.688 0.091400 .   
## stat104 -2.092e-04 2.499e-04 -0.837 0.402719   
## stat105 3.724e-04 2.487e-04 1.497 0.134365   
## stat106 -3.882e-04 2.494e-04 -1.557 0.119600   
## stat107 3.695e-05 2.481e-04 0.149 0.881616   
## stat108 -2.444e-04 2.517e-04 -0.971 0.331691   
## stat109 8.658e-05 2.502e-04 0.346 0.729298   
## stat110 -3.605e-03 2.474e-04 -14.572 < 2e-16 \*\*\*  
## stat111 5.734e-06 2.492e-04 0.023 0.981648   
## stat112 -1.876e-04 2.513e-04 -0.747 0.455266   
## stat113 -1.901e-04 2.522e-04 -0.754 0.451011   
## stat114 6.675e-05 2.496e-04 0.267 0.789122   
## stat115 1.543e-04 2.498e-04 0.618 0.536808   
## stat116 9.930e-05 2.499e-04 0.397 0.691090   
## stat117 -1.421e-05 2.496e-04 -0.057 0.954596   
## stat118 -1.948e-04 2.478e-04 -0.786 0.431787   
## stat119 1.782e-04 2.487e-04 0.717 0.473574   
## stat120 2.116e-04 2.495e-04 0.848 0.396288   
## stat121 -4.135e-05 2.501e-04 -0.165 0.868683   
## stat122 1.301e-05 2.486e-04 0.052 0.958267   
## stat123 1.036e-04 2.523e-04 0.411 0.681455   
## stat124 -3.638e-04 2.484e-04 -1.465 0.143044   
## stat125 1.496e-04 2.509e-04 0.596 0.551114   
## stat126 -1.678e-05 2.483e-04 -0.068 0.946129   
## stat127 4.869e-05 2.496e-04 0.195 0.845335   
## stat128 -4.637e-05 2.490e-04 -0.186 0.852312   
## stat129 8.677e-05 2.492e-04 0.348 0.727713   
## stat130 3.031e-04 2.501e-04 1.212 0.225592   
## stat131 2.703e-04 2.498e-04 1.082 0.279310   
## stat132 -9.136e-05 2.498e-04 -0.366 0.714535   
## stat133 -1.304e-04 2.499e-04 -0.522 0.601793   
## stat134 -7.829e-05 2.482e-04 -0.315 0.752472   
## stat135 -1.715e-04 2.499e-04 -0.686 0.492563   
## stat136 4.967e-05 2.516e-04 0.197 0.843523   
## stat137 7.711e-05 2.473e-04 0.312 0.755219   
## stat138 -4.577e-05 2.498e-04 -0.183 0.854636   
## stat139 4.186e-05 2.525e-04 0.166 0.868349   
## stat140 8.976e-05 2.503e-04 0.359 0.719961   
## stat141 9.869e-05 2.488e-04 0.397 0.691685   
## stat142 -2.710e-04 2.521e-04 -1.075 0.282463   
## stat143 1.957e-04 2.486e-04 0.787 0.431139   
## stat144 3.389e-04 2.488e-04 1.362 0.173181   
## stat145 7.249e-05 2.535e-04 0.286 0.774947   
## stat146 -6.947e-04 2.512e-04 -2.766 0.005697 \*\*   
## stat147 -1.157e-04 2.513e-04 -0.460 0.645358   
## stat148 -3.642e-04 2.469e-04 -1.475 0.140193   
## stat149 -4.108e-04 2.504e-04 -1.641 0.100951   
## stat150 -1.839e-04 2.511e-04 -0.732 0.463961   
## stat151 -1.316e-04 2.532e-04 -0.520 0.603294   
## stat152 -2.035e-04 2.496e-04 -0.815 0.414865   
## stat153 -6.811e-05 2.550e-04 -0.267 0.789402   
## stat154 -1.601e-04 2.530e-04 -0.633 0.526970   
## stat155 -9.538e-05 2.488e-04 -0.383 0.701526   
## stat156 5.322e-04 2.535e-04 2.100 0.035798 \*   
## stat157 -1.705e-04 2.484e-04 -0.686 0.492431   
## stat158 -3.990e-05 2.523e-04 -0.158 0.874342   
## stat159 -5.850e-05 2.498e-04 -0.234 0.814875   
## stat160 1.240e-04 2.501e-04 0.496 0.619954   
## stat161 2.219e-04 2.518e-04 0.881 0.378350   
## stat162 -1.938e-05 2.489e-04 -0.078 0.937926   
## stat163 -5.958e-05 2.527e-04 -0.236 0.813618   
## stat164 1.966e-04 2.528e-04 0.778 0.436704   
## stat165 -2.054e-04 2.485e-04 -0.827 0.408476   
## stat166 -1.697e-04 2.496e-04 -0.680 0.496605   
## stat167 -2.623e-04 2.481e-04 -1.057 0.290533   
## stat168 9.870e-05 2.495e-04 0.396 0.692393   
## stat169 -2.784e-05 2.504e-04 -0.111 0.911491   
## stat170 -1.577e-04 2.506e-04 -0.629 0.529184   
## stat171 -9.483e-05 2.522e-04 -0.376 0.706948   
## stat172 3.042e-04 2.487e-04 1.224 0.221173   
## stat173 -4.315e-04 2.507e-04 -1.722 0.085213 .   
## stat174 -2.435e-04 2.506e-04 -0.972 0.331147   
## stat175 -4.961e-04 2.512e-04 -1.975 0.048347 \*   
## stat176 -1.525e-04 2.500e-04 -0.610 0.541896   
## stat177 -3.101e-04 2.515e-04 -1.233 0.217663   
## stat178 -3.044e-04 2.541e-04 -1.198 0.231113   
## stat179 9.087e-05 2.494e-04 0.364 0.715570   
## stat180 -2.264e-04 2.493e-04 -0.908 0.363980   
## stat181 -1.457e-05 2.524e-04 -0.058 0.953975   
## stat182 1.234e-04 2.519e-04 0.490 0.624188   
## stat183 2.148e-04 2.496e-04 0.861 0.389525   
## stat184 5.092e-05 2.518e-04 0.202 0.839730   
## stat185 -3.277e-04 2.482e-04 -1.320 0.186906   
## stat186 -5.558e-04 2.515e-04 -2.210 0.027167 \*   
## stat187 -2.228e-04 2.495e-04 -0.893 0.371785   
## stat188 2.435e-06 2.501e-04 0.010 0.992232   
## stat189 1.968e-04 2.510e-04 0.784 0.432989   
## stat190 9.530e-05 2.485e-04 0.384 0.701333   
## stat191 -4.841e-04 2.494e-04 -1.941 0.052320 .   
## stat192 -6.048e-05 2.531e-04 -0.239 0.811134   
## stat193 1.305e-04 2.523e-04 0.517 0.605185   
## stat194 1.116e-05 2.491e-04 0.045 0.964263   
## stat195 3.377e-04 2.499e-04 1.351 0.176621   
## stat196 -2.072e-04 2.557e-04 -0.810 0.417816   
## stat197 -3.009e-05 2.477e-04 -0.121 0.903311   
## stat198 -4.093e-04 2.510e-04 -1.631 0.103010   
## stat199 1.430e-04 2.471e-04 0.579 0.562739   
## stat200 -2.637e-04 2.474e-04 -1.066 0.286604   
## stat201 -5.282e-06 2.486e-04 -0.021 0.983051   
## stat202 -3.598e-04 2.530e-04 -1.422 0.155044   
## stat203 2.583e-04 2.477e-04 1.043 0.297185   
## stat204 -6.639e-04 2.482e-04 -2.674 0.007511 \*\*   
## stat205 -2.856e-06 2.493e-04 -0.011 0.990861   
## stat206 -7.487e-05 2.514e-04 -0.298 0.765879   
## stat207 3.299e-04 2.505e-04 1.317 0.188016   
## stat208 -5.920e-05 2.498e-04 -0.237 0.812698   
## stat209 -6.476e-06 2.498e-04 -0.026 0.979319   
## stat210 -2.152e-04 2.520e-04 -0.854 0.393200   
## stat211 -8.859e-05 2.512e-04 -0.353 0.724406   
## stat212 -1.449e-04 2.499e-04 -0.580 0.562082   
## stat213 -2.154e-04 2.508e-04 -0.859 0.390318   
## stat214 -3.578e-04 2.500e-04 -1.431 0.152403   
## stat215 -2.943e-04 2.504e-04 -1.175 0.239896   
## stat216 -6.929e-05 2.507e-04 -0.276 0.782248   
## stat217 1.521e-04 2.514e-04 0.605 0.545240   
## x18.sqrt 2.564e-02 9.551e-04 26.842 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.03167 on 5343 degrees of freedom  
## Multiple R-squared: 0.2707, Adjusted R-squared: 0.238   
## F-statistic: 8.265 on 240 and 5343 DF, p-value: < 2.2e-16

cd.full = plot.diagnostics(model=model.full, train=data.train)



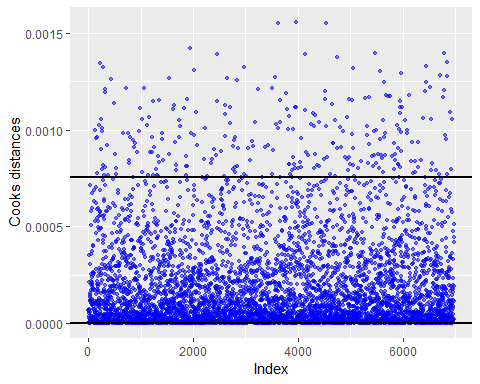
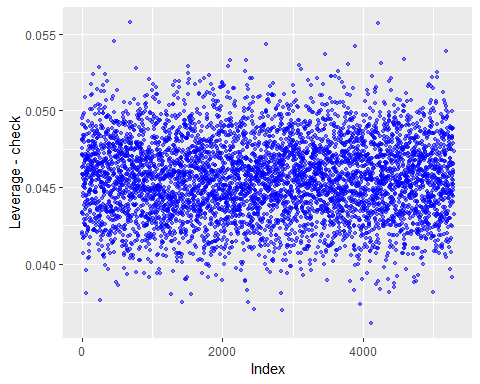
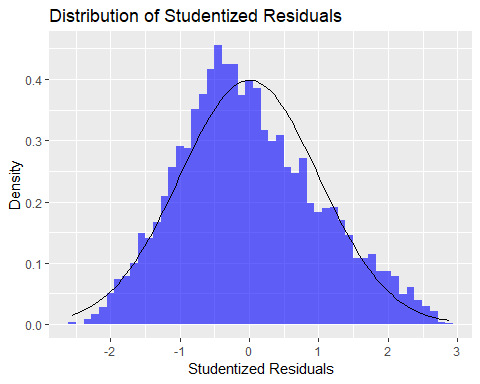
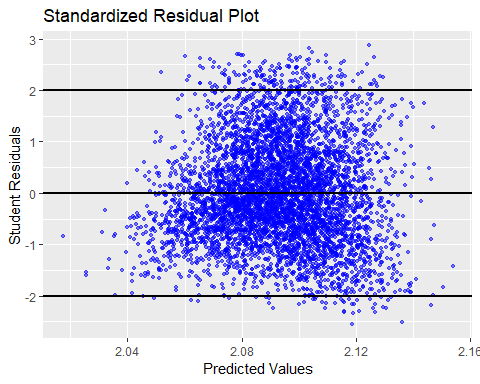
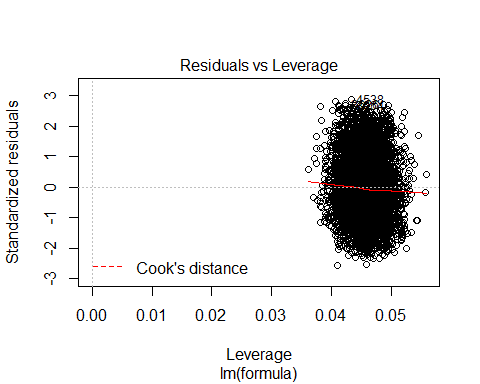
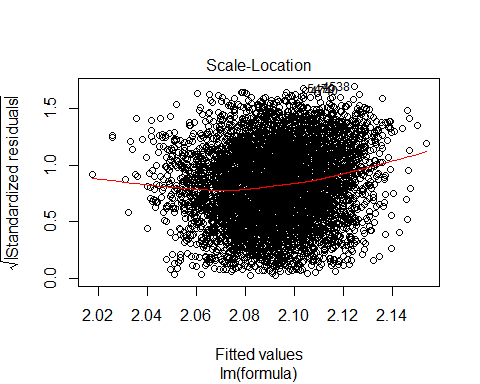
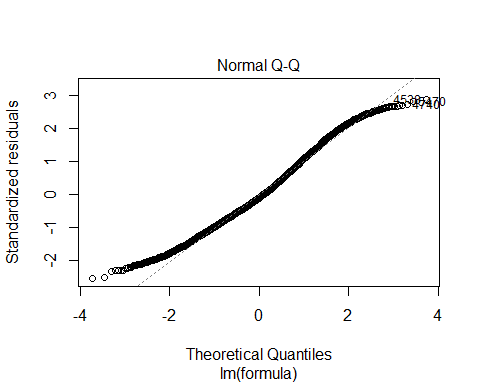
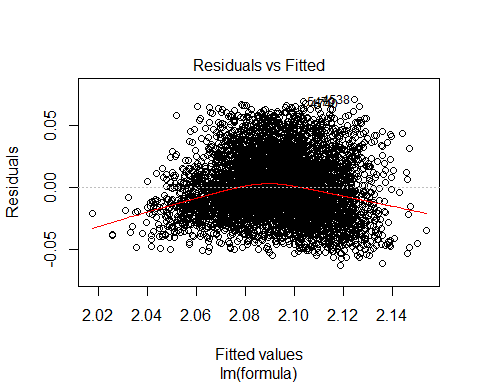
## [1] "Number of data points that have Cook's D > 4/n: 296"  
## [1] "Number of data points that have Cook's D > 1: 0"

## Checking with removal of high influence points

high.cd = names(cd.full[cd.full > 4/nrow(data.train)])  
data.train2 = data.train[!(rownames(data.train)) %in% high.cd,]  
model.full2 = lm(formula , data.train2)  
summary(model.full2)

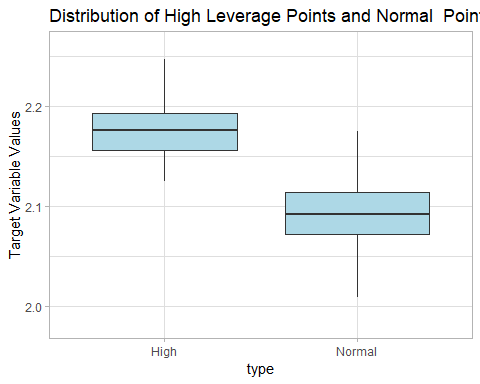
##   
## Call:  
## lm(formula = formula, data = data.train2)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.062564 -0.017352 -0.002707 0.016450 0.070396   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.961e+00 7.793e-03 251.685 < 2e-16 \*\*\*  
## x1 6.562e-05 5.363e-04 0.122 0.902614   
## x2 8.517e-05 3.410e-04 0.250 0.802772   
## x3 4.488e-05 9.338e-05 0.481 0.630830   
## x4 -5.858e-05 7.458e-06 -7.855 4.86e-15 \*\*\*  
## x5 5.043e-04 2.424e-04 2.080 0.037539 \*   
## x6 2.741e-04 4.864e-04 0.564 0.573066   
## x7 1.224e-02 5.232e-04 23.392 < 2e-16 \*\*\*  
## x8 3.792e-04 1.223e-04 3.100 0.001943 \*\*   
## x9 3.237e-03 2.721e-04 11.895 < 2e-16 \*\*\*  
## x10 1.496e-03 2.544e-04 5.882 4.32e-09 \*\*\*  
## x11 1.974e+05 6.049e+04 3.264 0.001107 \*\*   
## x12 -4.505e-05 1.537e-04 -0.293 0.769497   
## x13 9.758e-05 6.208e-05 1.572 0.116061   
## x14 -3.936e-04 2.671e-04 -1.474 0.140640   
## x15 2.083e-04 2.521e-04 0.826 0.408584   
## x16 8.805e-04 1.758e-04 5.009 5.67e-07 \*\*\*  
## x17 1.330e-03 2.673e-04 4.975 6.75e-07 \*\*\*  
## x19 2.604e-04 1.355e-04 1.922 0.054669 .   
## x20 -1.300e-03 9.426e-04 -1.379 0.168017   
## x21 1.402e-04 3.461e-05 4.051 5.17e-05 \*\*\*  
## x22 -6.003e-04 2.835e-04 -2.117 0.034312 \*   
## x23 1.301e-04 2.686e-04 0.484 0.628167   
## stat1 -3.497e-04 2.037e-04 -1.716 0.086144 .   
## stat2 3.013e-04 2.029e-04 1.485 0.137651   
## stat3 4.965e-04 2.038e-04 2.436 0.014878 \*   
## stat4 -6.517e-04 2.054e-04 -3.173 0.001520 \*\*   
## stat5 -1.960e-04 2.042e-04 -0.960 0.337117   
## stat6 -2.638e-04 2.044e-04 -1.291 0.196885   
## stat7 -2.414e-04 2.024e-04 -1.192 0.233174   
## stat8 -1.540e-05 2.037e-04 -0.076 0.939767   
## stat9 -1.371e-05 2.032e-04 -0.067 0.946202   
## stat10 -2.384e-04 2.035e-04 -1.172 0.241343   
## stat11 -2.289e-05 2.049e-04 -0.112 0.911060   
## stat12 5.242e-05 2.022e-04 0.259 0.795441   
## stat13 -5.131e-04 2.012e-04 -2.550 0.010805 \*   
## stat14 -1.089e-03 2.021e-04 -5.388 7.45e-08 \*\*\*  
## stat15 -6.229e-04 2.017e-04 -3.089 0.002021 \*\*   
## stat16 -3.603e-04 2.022e-04 -1.782 0.074814 .   
## stat17 2.911e-06 2.008e-04 0.014 0.988432   
## stat18 -3.858e-05 2.030e-04 -0.190 0.849280   
## stat19 -1.748e-04 2.026e-04 -0.863 0.388072   
## stat20 -7.497e-05 2.041e-04 -0.367 0.713347   
## stat21 -4.021e-05 2.034e-04 -0.198 0.843317   
## stat22 -1.921e-04 2.037e-04 -0.943 0.345787   
## stat23 4.904e-04 2.033e-04 2.412 0.015912 \*   
## stat24 -3.453e-04 2.040e-04 -1.692 0.090659 .   
## stat25 -2.080e-04 2.027e-04 -1.026 0.304812   
## stat26 -3.179e-04 2.037e-04 -1.561 0.118686   
## stat27 2.601e-04 2.037e-04 1.277 0.201536   
## stat28 -1.591e-04 2.043e-04 -0.779 0.436054   
## stat29 2.761e-04 2.054e-04 1.344 0.178954   
## stat30 2.124e-04 2.046e-04 1.038 0.299268   
## stat31 3.040e-05 2.050e-04 0.148 0.882122   
## stat32 9.615e-06 2.057e-04 0.047 0.962726   
## stat33 -2.923e-04 2.030e-04 -1.440 0.150010   
## stat34 3.389e-04 2.025e-04 1.673 0.094300 .   
## stat35 -5.630e-04 2.037e-04 -2.764 0.005737 \*\*   
## stat36 -9.835e-05 2.022e-04 -0.486 0.626775   
## stat37 -1.345e-05 2.062e-04 -0.065 0.948005   
## stat38 4.107e-04 2.028e-04 2.025 0.042904 \*   
## stat39 -2.585e-04 2.023e-04 -1.278 0.201225   
## stat40 -1.911e-04 2.048e-04 -0.933 0.350780   
## stat41 -3.953e-04 2.026e-04 -1.952 0.051049 .   
## stat42 -2.007e-04 2.031e-04 -0.988 0.322972   
## stat43 -2.143e-04 2.050e-04 -1.046 0.295755   
## stat44 3.231e-04 2.043e-04 1.582 0.113730   
## stat45 -1.329e-04 2.036e-04 -0.653 0.514000   
## stat46 1.743e-04 2.040e-04 0.855 0.392758   
## stat47 1.997e-04 2.042e-04 0.978 0.328091   
## stat48 3.978e-04 2.034e-04 1.956 0.050537 .   
## stat49 -2.563e-05 2.034e-04 -0.126 0.899715   
## stat50 4.184e-04 2.011e-04 2.081 0.037488 \*   
## stat51 1.492e-04 2.035e-04 0.733 0.463537   
## stat52 5.156e-05 2.045e-04 0.252 0.800927   
## stat53 -2.238e-04 2.049e-04 -1.093 0.274652   
## stat54 -4.360e-04 2.048e-04 -2.129 0.033281 \*   
## stat55 1.128e-04 2.027e-04 0.557 0.577862   
## stat56 9.491e-05 2.030e-04 0.468 0.640114   
## stat57 -8.983e-05 2.013e-04 -0.446 0.655479   
## stat58 -6.397e-05 2.026e-04 -0.316 0.752169   
## stat59 7.025e-05 2.042e-04 0.344 0.730802   
## stat60 5.331e-04 2.039e-04 2.615 0.008953 \*\*   
## stat61 -5.269e-05 2.033e-04 -0.259 0.795446   
## stat62 -4.613e-04 2.025e-04 -2.278 0.022774 \*   
## stat63 8.475e-05 2.041e-04 0.415 0.677960   
## stat64 2.407e-04 2.012e-04 1.196 0.231609   
## stat65 -2.644e-04 2.040e-04 -1.296 0.194981   
## stat66 1.034e-04 2.047e-04 0.505 0.613620   
## stat67 6.691e-05 2.040e-04 0.328 0.742962   
## stat68 -2.655e-04 2.046e-04 -1.298 0.194384   
## stat69 -2.368e-04 2.035e-04 -1.163 0.244797   
## stat70 2.368e-04 2.025e-04 1.169 0.242414   
## stat71 7.826e-05 2.017e-04 0.388 0.698018   
## stat72 5.752e-05 2.042e-04 0.282 0.778249   
## stat73 2.956e-04 2.043e-04 1.447 0.147998   
## stat74 1.785e-06 2.036e-04 0.009 0.993004   
## stat75 -1.163e-04 2.042e-04 -0.570 0.568896   
## stat76 4.550e-05 2.037e-04 0.223 0.823271   
## stat77 1.254e-04 2.026e-04 0.619 0.536141   
## stat78 -3.608e-04 2.027e-04 -1.780 0.075181 .   
## stat79 -9.692e-05 2.046e-04 -0.474 0.635770   
## stat80 4.340e-04 2.042e-04 2.125 0.033621 \*   
## stat81 -4.893e-05 2.043e-04 -0.239 0.810772   
## stat82 8.125e-05 2.032e-04 0.400 0.689263   
## stat83 -1.248e-04 2.043e-04 -0.611 0.541258   
## stat84 -2.152e-04 2.036e-04 -1.057 0.290437   
## stat85 -1.609e-04 2.035e-04 -0.791 0.429242   
## stat86 2.925e-05 2.027e-04 0.144 0.885285   
## stat87 -3.210e-04 2.043e-04 -1.571 0.116173   
## stat88 -8.302e-05 2.018e-04 -0.411 0.680871   
## stat89 -1.794e-04 2.025e-04 -0.886 0.375626   
## stat90 -1.576e-04 2.030e-04 -0.776 0.437627   
## stat91 -5.817e-04 2.025e-04 -2.872 0.004092 \*\*   
## stat92 -1.629e-04 2.026e-04 -0.804 0.421386   
## stat93 -4.378e-05 2.058e-04 -0.213 0.831566   
## stat94 1.073e-04 2.040e-04 0.526 0.599102   
## stat95 3.075e-04 2.027e-04 1.517 0.129343   
## stat96 -3.201e-04 2.042e-04 -1.567 0.117169   
## stat97 2.958e-04 2.022e-04 1.463 0.143529   
## stat98 3.489e-03 2.013e-04 17.328 < 2e-16 \*\*\*  
## stat99 4.617e-04 2.043e-04 2.260 0.023882 \*   
## stat100 3.101e-04 2.026e-04 1.531 0.125918   
## stat101 -1.130e-04 2.051e-04 -0.551 0.581571   
## stat102 2.619e-04 2.050e-04 1.278 0.201290   
## stat103 -3.176e-04 2.050e-04 -1.549 0.121382   
## stat104 -4.774e-05 2.037e-04 -0.234 0.814758   
## stat105 4.242e-04 2.025e-04 2.095 0.036206 \*   
## stat106 -4.393e-04 2.027e-04 -2.167 0.030254 \*   
## stat107 1.209e-04 2.017e-04 0.600 0.548816   
## stat108 -1.894e-05 2.050e-04 -0.092 0.926401   
## stat109 -5.759e-05 2.034e-04 -0.283 0.777125   
## stat110 -3.439e-03 2.009e-04 -17.122 < 2e-16 \*\*\*  
## stat111 6.260e-05 2.024e-04 0.309 0.757135   
## stat112 -1.122e-04 2.045e-04 -0.549 0.583226   
## stat113 -1.950e-04 2.048e-04 -0.952 0.340942   
## stat114 1.013e-04 2.029e-04 0.499 0.617519   
## stat115 2.926e-04 2.036e-04 1.437 0.150649   
## stat116 2.274e-04 2.033e-04 1.119 0.263388   
## stat117 -9.305e-05 2.025e-04 -0.460 0.645862   
## stat118 1.473e-04 2.012e-04 0.732 0.464318   
## stat119 3.717e-04 2.021e-04 1.839 0.065992 .   
## stat120 -9.743e-07 2.029e-04 -0.005 0.996168   
## stat121 -7.963e-05 2.032e-04 -0.392 0.695140   
## stat122 -1.432e-04 2.025e-04 -0.707 0.479626   
## stat123 2.903e-04 2.046e-04 1.418 0.156108   
## stat124 -3.027e-04 2.017e-04 -1.501 0.133383   
## stat125 8.422e-05 2.039e-04 0.413 0.679595   
## stat126 -3.643e-05 2.022e-04 -0.180 0.857015   
## stat127 -4.976e-05 2.027e-04 -0.245 0.806112   
## stat128 -3.684e-04 2.019e-04 -1.824 0.068176 .   
## stat129 1.313e-04 2.025e-04 0.649 0.516661   
## stat130 1.827e-04 2.032e-04 0.899 0.368548   
## stat131 2.173e-04 2.028e-04 1.072 0.283803   
## stat132 -1.761e-04 2.033e-04 -0.866 0.386521   
## stat133 -2.463e-05 2.036e-04 -0.121 0.903719   
## stat134 6.331e-05 2.019e-04 0.314 0.753909   
## stat135 -2.856e-04 2.030e-04 -1.407 0.159543   
## stat136 -8.295e-05 2.045e-04 -0.406 0.685029   
## stat137 9.828e-05 2.007e-04 0.490 0.624308   
## stat138 -1.438e-04 2.034e-04 -0.707 0.479765   
## stat139 -1.262e-04 2.055e-04 -0.614 0.539249   
## stat140 1.953e-04 2.030e-04 0.962 0.336112   
## stat141 3.408e-04 2.020e-04 1.687 0.091668 .   
## stat142 -2.154e-04 2.049e-04 -1.051 0.293166   
## stat143 5.773e-05 2.025e-04 0.285 0.775596   
## stat144 2.865e-04 2.022e-04 1.417 0.156619   
## stat145 -3.974e-06 2.063e-04 -0.019 0.984634   
## stat146 -8.362e-04 2.043e-04 -4.092 4.34e-05 \*\*\*  
## stat147 -4.039e-05 2.048e-04 -0.197 0.843665   
## stat148 -2.116e-04 2.010e-04 -1.053 0.292489   
## stat149 -4.194e-04 2.039e-04 -2.057 0.039745 \*   
## stat150 -3.311e-04 2.046e-04 -1.618 0.105720   
## stat151 2.569e-04 2.064e-04 1.245 0.213348   
## stat152 -9.362e-05 2.026e-04 -0.462 0.644123   
## stat153 1.660e-04 2.070e-04 0.802 0.422710   
## stat154 8.999e-05 2.063e-04 0.436 0.662746   
## stat155 2.542e-04 2.027e-04 1.254 0.209810   
## stat156 4.096e-04 2.059e-04 1.989 0.046703 \*   
## stat157 -1.763e-04 2.015e-04 -0.875 0.381579   
## stat158 2.951e-04 2.049e-04 1.440 0.149814   
## stat159 3.809e-05 2.028e-04 0.188 0.851042   
## stat160 1.181e-04 2.038e-04 0.579 0.562520   
## stat161 1.588e-05 2.050e-04 0.077 0.938274   
## stat162 -6.954e-05 2.022e-04 -0.344 0.730894   
## stat163 7.076e-05 2.061e-04 0.343 0.731361   
## stat164 4.717e-05 2.060e-04 0.229 0.818905   
## stat165 -5.839e-05 2.022e-04 -0.289 0.772776   
## stat166 -2.055e-04 2.023e-04 -1.016 0.309689   
## stat167 -3.827e-04 2.020e-04 -1.894 0.058245 .   
## stat168 7.745e-05 2.026e-04 0.382 0.702340   
## stat169 -1.699e-04 2.039e-04 -0.833 0.404684   
## stat170 1.306e-04 2.041e-04 0.640 0.522334   
## stat171 -1.806e-04 2.052e-04 -0.880 0.378852   
## stat172 4.812e-04 2.018e-04 2.385 0.017120 \*   
## stat173 -2.932e-04 2.037e-04 -1.439 0.150173   
## stat174 -8.056e-05 2.036e-04 -0.396 0.692337   
## stat175 -3.543e-04 2.041e-04 -1.736 0.082643 .   
## stat176 -4.209e-04 2.034e-04 -2.070 0.038541 \*   
## stat177 -7.333e-04 2.049e-04 -3.578 0.000349 \*\*\*  
## stat178 -3.167e-04 2.067e-04 -1.532 0.125517   
## stat179 4.429e-05 2.029e-04 0.218 0.827238   
## stat180 -1.458e-04 2.032e-04 -0.717 0.473167   
## stat181 3.677e-05 2.050e-04 0.179 0.857639   
## stat182 2.864e-04 2.051e-04 1.396 0.162767   
## stat183 2.362e-04 2.029e-04 1.164 0.244374   
## stat184 2.924e-04 2.044e-04 1.431 0.152516   
## stat185 -1.946e-04 2.019e-04 -0.964 0.335000   
## stat186 -1.682e-04 2.048e-04 -0.821 0.411506   
## stat187 -1.426e-04 2.026e-04 -0.704 0.481378   
## stat188 -4.873e-06 2.031e-04 -0.024 0.980865   
## stat189 -5.296e-05 2.049e-04 -0.258 0.796055   
## stat190 -1.263e-04 2.019e-04 -0.625 0.531672   
## stat191 -3.230e-04 2.031e-04 -1.590 0.111831   
## stat192 2.038e-05 2.061e-04 0.099 0.921261   
## stat193 1.851e-04 2.055e-04 0.900 0.367908   
## stat194 -1.811e-04 2.028e-04 -0.893 0.371761   
## stat195 -1.914e-05 2.035e-04 -0.094 0.925094   
## stat196 -2.800e-04 2.074e-04 -1.350 0.177121   
## stat197 -1.725e-04 2.016e-04 -0.856 0.392297   
## stat198 -3.045e-04 2.039e-04 -1.493 0.135402   
## stat199 1.477e-04 2.007e-04 0.736 0.461738   
## stat200 -1.254e-04 2.017e-04 -0.622 0.534162   
## stat201 1.290e-04 2.025e-04 0.637 0.524061   
## stat202 -1.876e-04 2.056e-04 -0.913 0.361499   
## stat203 2.870e-04 2.016e-04 1.424 0.154502   
## stat204 -2.755e-04 2.022e-04 -1.362 0.173112   
## stat205 2.213e-04 2.020e-04 1.095 0.273412   
## stat206 -1.107e-04 2.041e-04 -0.543 0.587397   
## stat207 5.047e-04 2.039e-04 2.475 0.013356 \*   
## stat208 9.686e-05 2.033e-04 0.476 0.633743   
## stat209 7.930e-05 2.028e-04 0.391 0.695826   
## stat210 -4.223e-04 2.048e-04 -2.062 0.039224 \*   
## stat211 7.699e-05 2.043e-04 0.377 0.706286   
## stat212 -1.311e-04 2.031e-04 -0.645 0.518743   
## stat213 -1.518e-04 2.036e-04 -0.746 0.455987   
## stat214 -1.587e-04 2.035e-04 -0.780 0.435499   
## stat215 -5.813e-05 2.040e-04 -0.285 0.775696   
## stat216 -6.252e-05 2.034e-04 -0.307 0.758529   
## stat217 8.147e-07 2.043e-04 0.004 0.996818   
## x18.sqrt 2.613e-02 7.753e-04 33.707 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.02503 on 5047 degrees of freedom  
## Multiple R-squared: 0.381, Adjusted R-squared: 0.3515   
## F-statistic: 12.94 on 240 and 5047 DF, p-value: < 2.2e-16

cd.full2 = plot.diagnostics(model.full2, data.train2)



## [1] "Number of data points that have Cook's D > 4/n: 270"  
## [1] "Number of data points that have Cook's D > 1: 0"

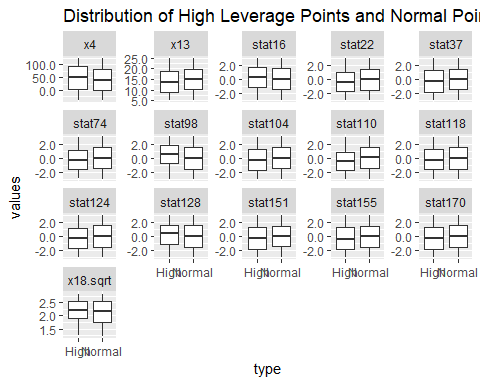
# much more normal residuals than before.   
# Checking to see if distributions are different and if so whcih variables  
# High Leverage Plot   
plotData = data.train %>%   
 rownames\_to\_column() %>%  
 mutate(type=ifelse(rowname %in% high.cd,'High','Normal')) %>%  
 dplyr::select(type,target=one\_of(label.names))  
  
ggplot(data=plotData, aes(x=type,y=target)) +  
 geom\_boxplot(fill='light blue',outlier.shape=NA) +  
 scale\_y\_continuous(name="Target Variable Values",label=scales::comma\_format(accuracy=.1)) +  
 theme\_light() +  
 ggtitle('Distribution of High Leverage Points and Normal Points')



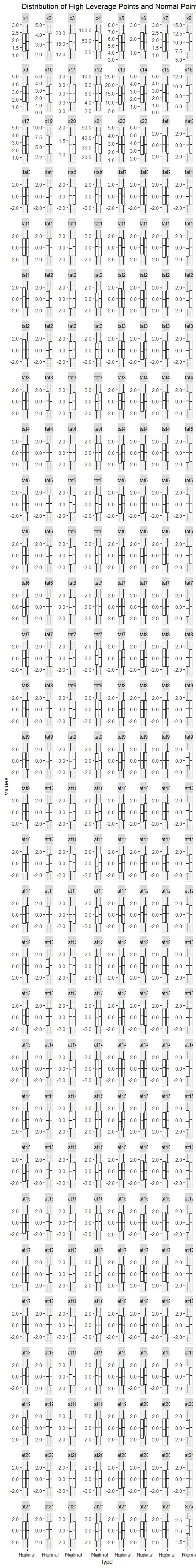
# 2 sample t-tests  
  
plotData = data.train %>%   
 rownames\_to\_column() %>%  
 mutate(type=ifelse(rowname %in% high.cd,'High','Normal')) %>%  
 dplyr::select(type,one\_of(feature.names))  
  
comp.test = lapply(dplyr::select(plotData, one\_of(feature.names))  
 , function(x) t.test(x ~ plotData$type, var.equal = TRUE))   
  
sig.comp = list.filter(comp.test, p.value < 0.05)  
sapply(sig.comp, function(x) x[['p.value']])

## x4 x13 stat16 stat22 stat37 stat74 stat98 stat104 stat110   
## 3.731826e-02 6.921376e-03 4.499620e-02 2.038699e-02 2.126469e-02 2.925538e-02 3.315695e-06 4.524951e-02 1.850834e-04   
## stat118 stat124 stat128 stat151 stat155 stat170 x18.sqrt   
## 3.654050e-02 4.840502e-02 3.154062e-03 2.110851e-02 2.771776e-02 3.909266e-02 2.354888e-02

mm = melt(plotData, id=c('type')) %>% filter(variable %in% names(sig.comp))  
  
ggplot(mm,aes(x=type, y=value)) +  
 geom\_boxplot()+  
 facet\_wrap(~variable, ncol=5, scales = 'free\_y') +  
 scale\_y\_continuous(name="values",label=scales::comma\_format(accuracy=.1)) +  
 ggtitle('Distribution of High Leverage Points and Normal Points')



# Distribution (box) Plots  
mm = melt(plotData, id=c('type'))  
  
ggplot(mm,aes(x=type, y=value)) +  
 geom\_boxplot()+  
 facet\_wrap(~variable, ncol=8, scales = 'free\_y') +  
 scale\_y\_continuous(name="values",label=scales::comma\_format(accuracy=.1)) +  
 ggtitle('Distribution of High Leverage Points and Normal Points')



## Grand Means Model

model.null = lm(grand.mean.formula, data.train)  
summary(model.null)

##   
## Call:  
## lm(formula = grand.mean.formula, data = data.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.114744 -0.023920 -0.003176 0.020902 0.164489   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.0966204 0.0004855 4319 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.03628 on 5583 degrees of freedom

## Variable Selection

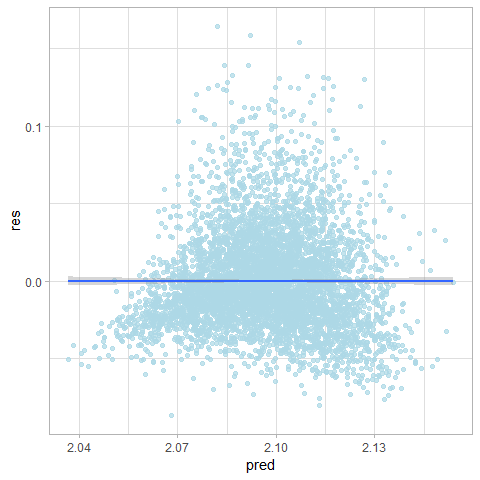
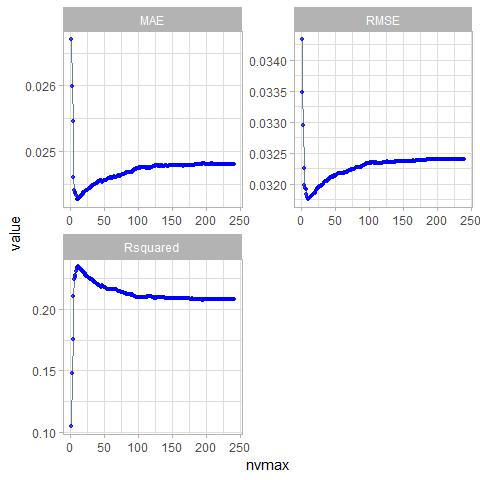
Basic: <http://www.stat.columbia.edu/~martin/W2024/R10.pdf> Cross Validation + Other Metrics: <http://www.sthda.com/english/articles/37-model-selection-essentials-in-r/154-stepwise-regression-essentials-in-r/>

### Forward Selection with CV

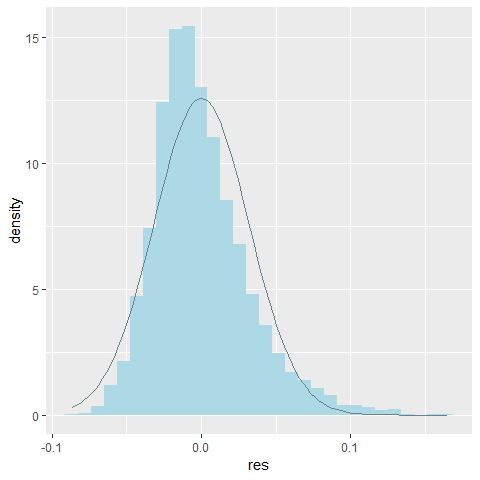
#### Train

if (algo.forward.caret == TRUE){  
 set.seed(1)  
 returned = train.caret.glmselect(formula = formula  
 , data = data.train  
 , method = "leapForward"  
 , feature.names = feature.names)  
 model.forward = returned$model  
 id = returned$id  
}

## Aggregating results  
## Selecting tuning parameters  
## Fitting nvmax = 11 on full training set  
## [1] "All models results"  
## nvmax RMSE Rsquared MAE RMSESD RsquaredSD MAESD  
## 1 1 0.03433074 0.1048652 0.02669982 0.0009261803 0.02117276 0.0006028036  
## 2 2 0.03348741 0.1480642 0.02598665 0.0009358138 0.02197751 0.0006196956  
## 3 3 0.03295185 0.1756793 0.02545556 0.0009655478 0.02940230 0.0006300332  
## 4 4 0.03225830 0.2100902 0.02460393 0.0010569927 0.03875907 0.0007157887  
## 5 5 0.03197411 0.2240441 0.02440412 0.0010723334 0.04180201 0.0007678828  
## 6 6 0.03193699 0.2259220 0.02437171 0.0010318862 0.03925322 0.0007656639  
## 7 7 0.03190821 0.2270003 0.02437009 0.0010314741 0.03680224 0.0007072754  
## 8 8 0.03183760 0.2303050 0.02432669 0.0010355665 0.03567746 0.0007199164  
## 9 9 0.03178572 0.2327543 0.02427982 0.0010201358 0.03519427 0.0007210205  
## 10 10 0.03176701 0.2336791 0.02427305 0.0010328434 0.03634847 0.0007382203  
## 11 11 0.03175425 0.2342165 0.02427299 0.0009983627 0.03444561 0.0007166804  
## 12 12 0.03177023 0.2334855 0.02429347 0.0010037969 0.03454673 0.0007159624  
## 13 13 0.03178538 0.2327640 0.02429594 0.0009901537 0.03335959 0.0007360250  
## 14 14 0.03180291 0.2319070 0.02430930 0.0009821608 0.03190600 0.0007207990  
## 15 15 0.03179509 0.2323210 0.02432426 0.0009716771 0.03218259 0.0007128379  
## 16 16 0.03181327 0.2314083 0.02433694 0.0009500490 0.03170264 0.0007017497  
## 17 17 0.03183806 0.2302426 0.02435454 0.0009385681 0.03097334 0.0006988639  
## 18 18 0.03183534 0.2303618 0.02435549 0.0009438311 0.03091633 0.0006877500  
## 19 19 0.03184665 0.2298190 0.02437313 0.0009513497 0.03098831 0.0006983417  
## 20 20 0.03185729 0.2293116 0.02438262 0.0009609730 0.03215801 0.0006924766  
## 21 21 0.03187280 0.2286825 0.02439268 0.0009535370 0.03302951 0.0006989450  
## 22 22 0.03190037 0.2274570 0.02440168 0.0009567643 0.03291584 0.0007150198  
## 23 23 0.03191691 0.2267484 0.02441089 0.0009418347 0.03352612 0.0007213483  
## 24 24 0.03192841 0.2262815 0.02440846 0.0009429044 0.03359696 0.0007180475  
## 25 25 0.03192377 0.2265322 0.02441062 0.0009393596 0.03443558 0.0007339263  
## 26 26 0.03194365 0.2256020 0.02442124 0.0009300575 0.03450378 0.0007457495  
## 27 27 0.03195792 0.2249732 0.02443978 0.0009230172 0.03424009 0.0007235536  
## 28 28 0.03196189 0.2248147 0.02444206 0.0009023624 0.03437973 0.0007022881  
## 29 29 0.03197138 0.2243609 0.02445168 0.0009173598 0.03407020 0.0007241480  
## 30 30 0.03197467 0.2242219 0.02446056 0.0009176890 0.03389315 0.0007181135  
## 31 31 0.03198152 0.2239145 0.02446690 0.0009203932 0.03407516 0.0007194675  
## 32 32 0.03199623 0.2232303 0.02448012 0.0009211640 0.03404566 0.0007203750  
## 33 33 0.03199210 0.2234870 0.02447805 0.0009335329 0.03355747 0.0007463249  
## 34 34 0.03202182 0.2221318 0.02449557 0.0009290047 0.03306992 0.0007402607  
## 35 35 0.03203967 0.2214082 0.02450423 0.0009262735 0.03366259 0.0007374438  
## 36 36 0.03203993 0.2214079 0.02450637 0.0009301327 0.03347200 0.0007394007  
## 37 37 0.03204736 0.2210620 0.02451606 0.0009166949 0.03243609 0.0007307364  
## 38 38 0.03204947 0.2209705 0.02451467 0.0009032075 0.03154889 0.0007356877  
## 39 39 0.03206359 0.2203745 0.02452345 0.0009008424 0.03148755 0.0007179593  
## 40 40 0.03207805 0.2197196 0.02453771 0.0008948702 0.03100220 0.0007254520  
## 41 41 0.03208468 0.2194826 0.02454701 0.0008952856 0.03161173 0.0007220193  
## 42 42 0.03209436 0.2190144 0.02455068 0.0008824038 0.03053588 0.0007164534  
## 43 43 0.03211059 0.2182860 0.02455723 0.0008678310 0.03008038 0.0006936157  
## 44 44 0.03211439 0.2181563 0.02456549 0.0008546985 0.02952843 0.0006808491  
## 45 45 0.03212105 0.2179084 0.02456605 0.0008722749 0.03017928 0.0006797222  
## 46 46 0.03210627 0.2186128 0.02454628 0.0008643037 0.03016265 0.0006780243  
## 47 47 0.03210208 0.2188179 0.02453504 0.0008596109 0.02990735 0.0006839507  
## 48 48 0.03211745 0.2181789 0.02454673 0.0008575772 0.02985177 0.0006758966  
## 49 49 0.03213786 0.2172560 0.02456902 0.0008540496 0.02926417 0.0006781853  
## 50 50 0.03212883 0.2177164 0.02456224 0.0008606391 0.02976748 0.0006937523  
## 51 51 0.03213431 0.2175186 0.02455770 0.0008507114 0.02982203 0.0006865940  
## 52 52 0.03214810 0.2170259 0.02456721 0.0008571236 0.03067770 0.0006949534  
## 53 53 0.03215498 0.2167372 0.02456875 0.0008372929 0.03102993 0.0006849708  
## 54 54 0.03217046 0.2161025 0.02457857 0.0008254974 0.03155766 0.0006733758  
## 55 55 0.03217741 0.2157977 0.02459306 0.0008313091 0.03182364 0.0006853653  
## 56 56 0.03217560 0.2159374 0.02459608 0.0008292518 0.03194393 0.0006836589  
## 57 57 0.03216706 0.2163289 0.02459119 0.0008290093 0.03148413 0.0006899334  
## 58 58 0.03217561 0.2159481 0.02460096 0.0008301783 0.03131185 0.0006898117  
## 59 59 0.03217953 0.2157570 0.02460085 0.0008206348 0.03053644 0.0006779749  
## 60 60 0.03217516 0.2159973 0.02459956 0.0008188799 0.03109559 0.0006778484  
## 61 61 0.03217721 0.2159593 0.02459562 0.0008220179 0.03163457 0.0006799356  
## 62 62 0.03217751 0.2159859 0.02460015 0.0008109456 0.03108377 0.0006829977  
## 63 63 0.03217819 0.2160169 0.02460458 0.0008054896 0.03129134 0.0006869452  
## 64 64 0.03218597 0.2156816 0.02461670 0.0007995414 0.03109923 0.0006839008  
## 65 65 0.03218884 0.2155571 0.02461218 0.0007888355 0.03104376 0.0006730536  
## 66 66 0.03218541 0.2157628 0.02460903 0.0007877134 0.03109743 0.0006816084  
## 67 67 0.03219056 0.2156076 0.02461204 0.0007841132 0.03168243 0.0006757358  
## 68 68 0.03220047 0.2152527 0.02462234 0.0007765333 0.03211651 0.0006739922  
## 69 69 0.03220564 0.2150061 0.02462788 0.0007754458 0.03132423 0.0006779663  
## 70 70 0.03221949 0.2143978 0.02463206 0.0007756493 0.03113979 0.0006885557  
## 71 71 0.03222407 0.2142201 0.02463348 0.0007757184 0.03095866 0.0006956091  
## 72 72 0.03222399 0.2142203 0.02462929 0.0007639430 0.02992694 0.0006915775  
## 73 73 0.03222269 0.2143187 0.02462807 0.0007585721 0.03011078 0.0006932040  
## 74 74 0.03223210 0.2138131 0.02463815 0.0007691710 0.02957044 0.0006970556  
## 75 75 0.03223446 0.2136412 0.02463936 0.0007582383 0.02953119 0.0006964906  
## 76 76 0.03224373 0.2132030 0.02464822 0.0007524433 0.02863780 0.0006953251  
## 77 77 0.03224357 0.2132308 0.02464866 0.0007446196 0.02853157 0.0006933613  
## 78 78 0.03224402 0.2131848 0.02465342 0.0007494173 0.02827525 0.0006968616  
## 79 79 0.03225163 0.2128788 0.02465914 0.0007458039 0.02873317 0.0006933530  
## 80 80 0.03225481 0.2127223 0.02466209 0.0007344625 0.02845259 0.0006866422  
## 81 81 0.03226469 0.2123739 0.02466980 0.0007276964 0.02872628 0.0006862923  
## 82 82 0.03227131 0.2120373 0.02467719 0.0007327510 0.02848031 0.0006924438  
## 83 83 0.03226824 0.2121648 0.02467885 0.0007634081 0.02862693 0.0007062214  
## 84 84 0.03227583 0.2118626 0.02468755 0.0007685710 0.02851679 0.0007119116  
## 85 85 0.03226971 0.2120973 0.02468210 0.0007770229 0.02829195 0.0007176127  
## 86 86 0.03226827 0.2121982 0.02468529 0.0007762245 0.02869777 0.0007266719  
## 87 87 0.03226900 0.2121205 0.02468376 0.0007786393 0.02879258 0.0007149121  
## 88 88 0.03227796 0.2117594 0.02468547 0.0007773979 0.02861836 0.0007185135  
## 89 89 0.03228502 0.2115008 0.02468435 0.0007867653 0.02869353 0.0007280330  
## 90 90 0.03228139 0.2116593 0.02468852 0.0007920362 0.02864311 0.0007252737  
## 91 91 0.03229748 0.2110372 0.02470333 0.0007928458 0.02854451 0.0007286113  
## 92 92 0.03230337 0.2108137 0.02471016 0.0007889861 0.02877433 0.0007249772  
## 93 93 0.03230973 0.2105809 0.02471031 0.0007965447 0.02888581 0.0007242775  
## 94 94 0.03231914 0.2102199 0.02472198 0.0008002198 0.02943370 0.0007289156  
## 95 95 0.03232584 0.2099802 0.02473566 0.0007979761 0.02961976 0.0007262895  
## 96 96 0.03233321 0.2096480 0.02474594 0.0007996722 0.02962789 0.0007306195  
## 97 97 0.03233065 0.2097656 0.02473382 0.0008013801 0.02943510 0.0007251149  
## 98 98 0.03233298 0.2097181 0.02474063 0.0008021512 0.02946354 0.0007315911  
## 99 99 0.03233470 0.2097054 0.02474190 0.0008183320 0.02977700 0.0007420428  
## 100 100 0.03233238 0.2098074 0.02474673 0.0008113700 0.03015699 0.0007322781  
## 101 101 0.03234274 0.2093527 0.02475453 0.0008005244 0.02942706 0.0007317030  
## 102 102 0.03234538 0.2092517 0.02475823 0.0008014185 0.02950118 0.0007228570  
## 103 103 0.03233657 0.2096614 0.02474512 0.0007998140 0.02957567 0.0007189592  
## 104 104 0.03234566 0.2093007 0.02475528 0.0008025429 0.02983491 0.0007197858  
## 105 105 0.03234584 0.2093316 0.02475286 0.0007999697 0.02978666 0.0007151665  
## 106 106 0.03234951 0.2092059 0.02475472 0.0008070015 0.02984070 0.0007229470  
## 107 107 0.03234796 0.2092679 0.02475555 0.0008123665 0.03007577 0.0007297732  
## 108 108 0.03234142 0.2095588 0.02475591 0.0008127759 0.02995777 0.0007266356  
## 109 109 0.03233953 0.2096829 0.02475265 0.0008095866 0.02977191 0.0007198375  
## 110 110 0.03234383 0.2095179 0.02475457 0.0008126598 0.02992582 0.0007183148  
## 111 111 0.03233894 0.2097084 0.02474931 0.0008125268 0.02997889 0.0007186687  
## 112 112 0.03233711 0.2098056 0.02475312 0.0008081366 0.02987177 0.0007179038  
## 113 113 0.03233228 0.2100649 0.02474611 0.0007986519 0.03009960 0.0007088480  
## 114 114 0.03233264 0.2100803 0.02474942 0.0007907128 0.03010827 0.0007020673  
## 115 115 0.03233329 0.2100542 0.02475421 0.0007840766 0.02993800 0.0006979955  
## 116 116 0.03233386 0.2100202 0.02475125 0.0007804054 0.02979280 0.0006954229  
## 117 117 0.03233308 0.2100517 0.02475079 0.0007831086 0.03012090 0.0006971071  
## 118 118 0.03233280 0.2100722 0.02475299 0.0007815459 0.03020846 0.0006955673  
## 119 119 0.03233439 0.2099792 0.02475689 0.0007809712 0.03016641 0.0006977745  
## 120 120 0.03234438 0.2095640 0.02476649 0.0007821131 0.03019661 0.0006953340  
## 121 121 0.03235330 0.2091792 0.02477194 0.0007824433 0.03013929 0.0006958711  
## 122 122 0.03235445 0.2091190 0.02477406 0.0007900863 0.02998183 0.0006949376  
## 123 123 0.03235924 0.2089169 0.02477699 0.0007833662 0.03026895 0.0006836072  
## 124 124 0.03235575 0.2090970 0.02477762 0.0007904932 0.03072200 0.0006878292  
## 125 125 0.03235836 0.2089693 0.02478468 0.0007927892 0.03076152 0.0006866133  
## 126 126 0.03236118 0.2088630 0.02478832 0.0008018046 0.03093563 0.0006903557  
## 127 127 0.03235535 0.2091136 0.02478734 0.0008081561 0.03085688 0.0006933645  
## 128 128 0.03235389 0.2091705 0.02478798 0.0008093371 0.03070458 0.0006956529  
## 129 129 0.03235067 0.2093361 0.02478111 0.0008184786 0.03118471 0.0007023543  
## 130 130 0.03235202 0.2093039 0.02478484 0.0008175649 0.03127166 0.0007013422  
## 131 131 0.03235248 0.2092877 0.02478066 0.0008144027 0.03129433 0.0006957642  
## 132 132 0.03235245 0.2093112 0.02478083 0.0008169077 0.03129939 0.0006950092  
## 133 133 0.03235535 0.2092083 0.02478337 0.0008235000 0.03174888 0.0007008460  
## 134 134 0.03235741 0.2091196 0.02478582 0.0008238096 0.03149220 0.0007013267  
## 135 135 0.03235537 0.2091994 0.02478083 0.0008232647 0.03177374 0.0006975629  
## 136 136 0.03235369 0.2092534 0.02477920 0.0008176033 0.03184013 0.0006933018  
## 137 137 0.03235606 0.2091556 0.02478078 0.0008192719 0.03179551 0.0006990661  
## 138 138 0.03235562 0.2091288 0.02477792 0.0008204583 0.03150413 0.0007026397  
## 139 139 0.03236216 0.2088382 0.02478344 0.0008153006 0.03167051 0.0006995035  
## 140 140 0.03236168 0.2088549 0.02478233 0.0008099874 0.03163203 0.0006979432  
## 141 141 0.03236754 0.2085728 0.02478349 0.0008069287 0.03150866 0.0006889715  
## 142 142 0.03236272 0.2087597 0.02478170 0.0008028085 0.03142637 0.0006826674  
## 143 143 0.03236163 0.2088320 0.02478142 0.0008051091 0.03138180 0.0006823354  
## 144 144 0.03236454 0.2087043 0.02478128 0.0008102315 0.03126542 0.0006894110  
## 145 145 0.03236824 0.2085530 0.02478249 0.0008081694 0.03138828 0.0006922552  
## 146 146 0.03237156 0.2084512 0.02478875 0.0008020129 0.03125911 0.0006898824  
## 147 147 0.03236595 0.2086867 0.02478312 0.0007936825 0.03115882 0.0006881755  
## 148 148 0.03236406 0.2087810 0.02478307 0.0008007856 0.03086597 0.0006942305  
## 149 149 0.03236654 0.2086748 0.02478516 0.0008004081 0.03081572 0.0006973134  
## 150 150 0.03236830 0.2086115 0.02478631 0.0007975283 0.03085215 0.0006969591  
## 151 151 0.03237071 0.2085204 0.02478402 0.0007965239 0.03082393 0.0006981321  
## 152 152 0.03236983 0.2085718 0.02478551 0.0007947086 0.03058653 0.0006954736  
## 153 153 0.03237171 0.2084870 0.02478811 0.0007926540 0.03053278 0.0006928436  
## 154 154 0.03236945 0.2085942 0.02478726 0.0007975225 0.03065385 0.0006953157  
## 155 155 0.03236834 0.2086742 0.02478678 0.0007978921 0.03068910 0.0006954742  
## 156 156 0.03236943 0.2086534 0.02478773 0.0007992089 0.03065845 0.0006945002  
## 157 157 0.03236671 0.2087803 0.02478398 0.0008051246 0.03053033 0.0006981939  
## 158 158 0.03236882 0.2087108 0.02478343 0.0008054232 0.03065258 0.0006987847  
## 159 159 0.03236689 0.2087878 0.02478273 0.0008030143 0.03057969 0.0007025523  
## 160 160 0.03236800 0.2087424 0.02478420 0.0008056812 0.03063395 0.0007041587  
## 161 161 0.03236721 0.2087571 0.02478461 0.0008040512 0.03040951 0.0006992335  
## 162 162 0.03236699 0.2087847 0.02478476 0.0008068600 0.03044708 0.0007022722  
## 163 163 0.03237014 0.2086673 0.02478573 0.0008063708 0.03057983 0.0007021442  
## 164 164 0.03237398 0.2084975 0.02478666 0.0008062930 0.03043318 0.0007028706  
## 165 165 0.03237588 0.2084319 0.02478652 0.0008034137 0.03035972 0.0006987328  
## 166 166 0.03237351 0.2085508 0.02478157 0.0008060066 0.03051666 0.0006994942  
## 167 167 0.03237306 0.2085680 0.02478546 0.0008032834 0.03017612 0.0006980311  
## 168 168 0.03237439 0.2085347 0.02478736 0.0008039534 0.03025936 0.0006957648  
## 169 169 0.03237811 0.2083763 0.02479192 0.0008019814 0.03033452 0.0006943542  
## 170 170 0.03237623 0.2084587 0.02479045 0.0008027386 0.03052162 0.0006925539  
## 171 171 0.03237652 0.2084385 0.02478908 0.0008015905 0.03045259 0.0006891059  
## 172 172 0.03237754 0.2084007 0.02478793 0.0008034427 0.03048104 0.0006888749  
## 173 173 0.03237974 0.2083003 0.02479114 0.0008016130 0.03052689 0.0006840234  
## 174 174 0.03237901 0.2083634 0.02479152 0.0008013607 0.03068637 0.0006818128  
## 175 175 0.03237870 0.2083753 0.02478919 0.0008026168 0.03062241 0.0006811644  
## 176 176 0.03238231 0.2082277 0.02479424 0.0007977887 0.03051883 0.0006791369  
## 177 177 0.03238355 0.2081666 0.02479439 0.0007953878 0.03021590 0.0006757618  
## 178 178 0.03238177 0.2082479 0.02479577 0.0007933780 0.03020045 0.0006755593  
## 179 179 0.03238145 0.2082781 0.02479406 0.0007935923 0.03031883 0.0006780768  
## 180 180 0.03238356 0.2081979 0.02479362 0.0007958783 0.03032880 0.0006782025  
## 181 181 0.03238350 0.2082049 0.02479445 0.0007968860 0.03036759 0.0006766965  
## 182 182 0.03238673 0.2080718 0.02479666 0.0007967943 0.03042475 0.0006753559  
## 183 183 0.03238900 0.2079689 0.02479935 0.0007961674 0.03029701 0.0006737435  
## 184 184 0.03239148 0.2078736 0.02480322 0.0007976518 0.03039666 0.0006776694  
## 185 185 0.03239248 0.2078399 0.02480340 0.0007978453 0.03038587 0.0006795790  
## 186 186 0.03239459 0.2077593 0.02480448 0.0008038633 0.03052225 0.0006835629  
## 187 187 0.03239489 0.2077392 0.02480546 0.0008062305 0.03042609 0.0006847939  
## 188 188 0.03239529 0.2077363 0.02480641 0.0008090850 0.03044222 0.0006859409  
## 189 189 0.03239625 0.2076958 0.02480765 0.0008099361 0.03027613 0.0006859452  
## 190 190 0.03239977 0.2075378 0.02481204 0.0008085578 0.03035598 0.0006827815  
## 191 191 0.03239773 0.2076259 0.02481120 0.0008090593 0.03037229 0.0006842250  
## 192 192 0.03239796 0.2076377 0.02481133 0.0008094023 0.03045552 0.0006876470  
## 193 193 0.03239812 0.2076314 0.02481208 0.0008080899 0.03035447 0.0006886239  
## 194 194 0.03240357 0.2074035 0.02481512 0.0008046067 0.03021263 0.0006879130  
## 195 195 0.03240376 0.2073883 0.02481628 0.0008061842 0.03012122 0.0006884299  
## 196 196 0.03240118 0.2074881 0.02481502 0.0008060417 0.03006750 0.0006863987  
## 197 197 0.03239968 0.2075522 0.02481432 0.0008061175 0.03001844 0.0006878562  
## 198 198 0.03239687 0.2076666 0.02481070 0.0008064856 0.03001489 0.0006900378  
## 199 199 0.03239582 0.2077213 0.02481047 0.0008078701 0.02997146 0.0006911710  
## 200 200 0.03239260 0.2078487 0.02480602 0.0008082229 0.02991484 0.0006918175  
## 201 201 0.03239211 0.2078691 0.02480536 0.0008104231 0.02990355 0.0006937748  
## 202 202 0.03239494 0.2077477 0.02481001 0.0008088736 0.02988667 0.0006922808  
## 203 203 0.03239481 0.2077451 0.02481025 0.0008071418 0.02987765 0.0006904695  
## 204 204 0.03239465 0.2077518 0.02481042 0.0008068616 0.02984773 0.0006898419  
## 205 205 0.03239582 0.2077072 0.02481160 0.0008088763 0.02989790 0.0006919871  
## 206 206 0.03239676 0.2076669 0.02481258 0.0008084379 0.02992827 0.0006927683  
## 207 207 0.03239976 0.2075370 0.02481419 0.0008086301 0.02995573 0.0006923394  
## 208 208 0.03240047 0.2075116 0.02481357 0.0008087189 0.03001085 0.0006926649  
## 209 209 0.03239942 0.2075493 0.02481263 0.0008085179 0.03000634 0.0006933612  
## 210 210 0.03239923 0.2075564 0.02481237 0.0008070833 0.02994547 0.0006933037  
## 211 211 0.03239826 0.2075913 0.02481136 0.0008071926 0.02992519 0.0006939208  
## 212 212 0.03239800 0.2076000 0.02481051 0.0008065144 0.02990305 0.0006931590  
## 213 213 0.03239736 0.2076214 0.02481032 0.0008071987 0.02989720 0.0006933715  
## 214 214 0.03239786 0.2076028 0.02481028 0.0008071055 0.02989012 0.0006946509  
## 215 215 0.03239767 0.2076130 0.02480988 0.0008054066 0.02989333 0.0006938246  
## 216 216 0.03239697 0.2076483 0.02480957 0.0008057682 0.02993806 0.0006935698  
## 217 217 0.03239756 0.2076262 0.02480998 0.0008065609 0.02992503 0.0006943024  
## 218 218 0.03239652 0.2076673 0.02480844 0.0008079287 0.02996277 0.0006945053  
## 219 219 0.03239664 0.2076691 0.02480927 0.0008076442 0.03001166 0.0006943636  
## 220 220 0.03239666 0.2076654 0.02480944 0.0008070100 0.03000765 0.0006941494  
## 221 221 0.03239748 0.2076345 0.02481028 0.0008077605 0.02996920 0.0006941388  
## 222 222 0.03239743 0.2076412 0.02481035 0.0008076651 0.02999370 0.0006940418  
## 223 223 0.03239714 0.2076569 0.02481013 0.0008074580 0.02997602 0.0006933393  
## 224 224 0.03239681 0.2076675 0.02480995 0.0008069051 0.02997305 0.0006924176  
## 225 225 0.03239751 0.2076396 0.02480993 0.0008073367 0.03000847 0.0006931620  
## 226 226 0.03239684 0.2076676 0.02480894 0.0008077286 0.02999928 0.0006933119  
## 227 227 0.03239684 0.2076647 0.02480880 0.0008086006 0.02998916 0.0006940725  
## 228 228 0.03239689 0.2076627 0.02480908 0.0008083397 0.02996282 0.0006936294  
## 229 229 0.03239718 0.2076518 0.02480947 0.0008083455 0.02997024 0.0006931002  
## 230 230 0.03239755 0.2076363 0.02480947 0.0008078492 0.02996797 0.0006929937  
## 231 231 0.03239724 0.2076487 0.02480871 0.0008079005 0.02994713 0.0006935089  
## 232 232 0.03239709 0.2076543 0.02480875 0.0008085886 0.02995841 0.0006936891  
## 233 233 0.03239735 0.2076432 0.02480902 0.0008083724 0.02996337 0.0006934086  
## 234 234 0.03239749 0.2076374 0.02480902 0.0008081391 0.02996127 0.0006933539  
## 235 235 0.03239728 0.2076478 0.02480859 0.0008080836 0.02996168 0.0006936570  
## 236 236 0.03239728 0.2076493 0.02480841 0.0008081013 0.02996454 0.0006938541  
## 237 237 0.03239759 0.2076357 0.02480851 0.0008080369 0.02996750 0.0006937463  
## 238 238 0.03239754 0.2076391 0.02480848 0.0008079081 0.02997088 0.0006935764  
## 239 239 0.03239751 0.2076405 0.02480845 0.0008078407 0.02997007 0.0006935219  
## 240 240 0.03239750 0.2076409 0.02480845 0.0008078701 0.02996875 0.0006935553  
## [1] "Best Model"  
## nvmax  
## 11 11



## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

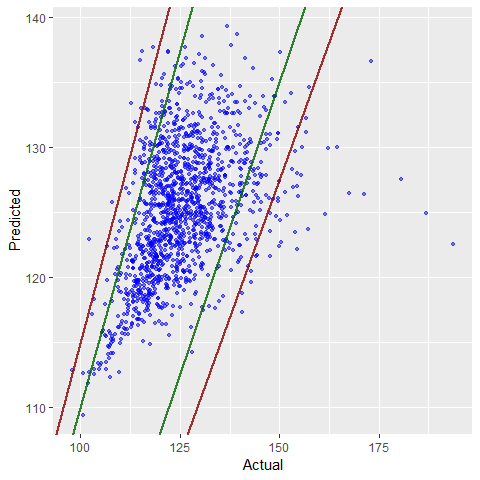


## [1] "Coefficients of final model:"  
## Estimate 2.5 % 97.5 %  
## (Intercept) 1.999228e+00 1.992687e+00 2.005768e+00  
## x4 -4.961729e-05 -6.718664e-05 -3.204795e-05  
## x7 1.138656e-02 1.015067e-02 1.262245e-02  
## x9 3.357169e-03 2.712443e-03 4.001895e-03  
## x10 1.139116e-03 5.398086e-04 1.738423e-03  
## x16 9.556884e-04 5.401986e-04 1.371178e-03  
## x17 1.357405e-03 7.256734e-04 1.989137e-03  
## stat14 -9.544654e-04 -1.430654e-03 -4.782771e-04  
## stat98 3.668932e-03 3.193272e-03 4.144592e-03  
## stat110 -3.579722e-03 -4.056280e-03 -3.103165e-03  
## stat146 -6.929336e-04 -1.174815e-03 -2.110520e-04  
## x18.sqrt 2.552906e-02 2.369269e-02 2.736542e-02

#### Test

if (algo.forward.caret == TRUE){  
 test.model(model=model.forward, test=data.test  
 ,method = 'leapForward',subopt = NULL  
 ,formula = formula, feature.names = feature.names, label.names = label.names  
 ,id = id  
 ,draw.limits = TRUE, transformation = t)  
}

## [1] "Summary of predicted values: "  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2.039 2.085 2.098 2.097 2.110 2.144   
## [1] "leapForward Test MSE: 0.000971233975337743"

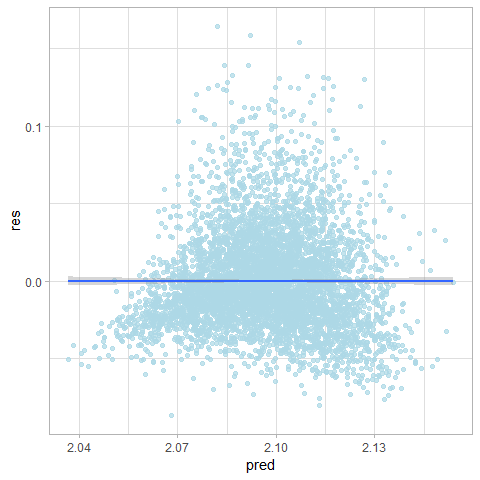
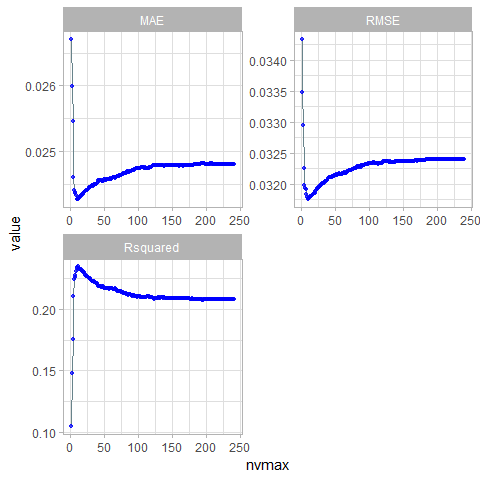


### Backward Elimination with CV

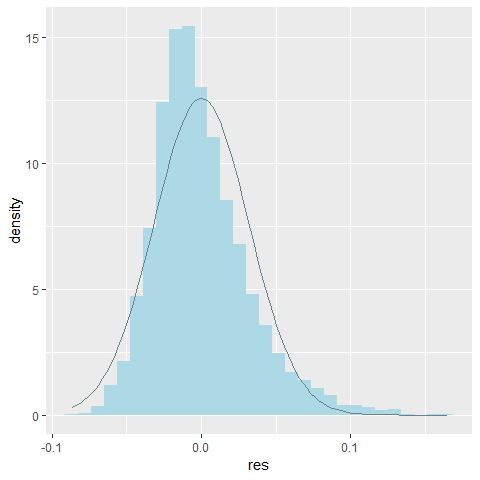
#### Train

if (algo.backward.caret == TRUE){  
 set.seed(1)  
 returned = train.caret.glmselect(formula = formula  
 ,data = data.train  
 ,method = "leapBackward"  
 ,feature.names = feature.names)  
 model.backward = returned$model  
 id = returned$id  
}

## Aggregating results  
## Selecting tuning parameters  
## Fitting nvmax = 11 on full training set  
## [1] "All models results"  
## nvmax RMSE Rsquared MAE RMSESD RsquaredSD MAESD  
## 1 1 0.03433074 0.1048652 0.02669982 0.0009261803 0.02117276 0.0006028036  
## 2 2 0.03348741 0.1480642 0.02598665 0.0009358138 0.02197751 0.0006196956  
## 3 3 0.03295185 0.1756793 0.02545556 0.0009655478 0.02940230 0.0006300332  
## 4 4 0.03225830 0.2100902 0.02460393 0.0010569927 0.03875907 0.0007157887  
## 5 5 0.03197411 0.2240441 0.02440412 0.0010723334 0.04180201 0.0007678828  
## 6 6 0.03193699 0.2259220 0.02437171 0.0010318862 0.03925322 0.0007656639  
## 7 7 0.03190821 0.2270003 0.02437009 0.0010314741 0.03680224 0.0007072754  
## 8 8 0.03183760 0.2303050 0.02432669 0.0010355665 0.03567746 0.0007199164  
## 9 9 0.03178572 0.2327543 0.02427982 0.0010201358 0.03519427 0.0007210205  
## 10 10 0.03176701 0.2336791 0.02427305 0.0010328434 0.03634847 0.0007382203  
## 11 11 0.03175425 0.2342165 0.02427299 0.0009983627 0.03444561 0.0007166804  
## 12 12 0.03177353 0.2333557 0.02429523 0.0010058286 0.03515755 0.0007217200  
## 13 13 0.03179507 0.2323029 0.02430312 0.0009982181 0.03356413 0.0007423370  
## 14 14 0.03179512 0.2323223 0.02431070 0.0009747765 0.03247881 0.0007212512  
## 15 15 0.03179509 0.2323210 0.02432426 0.0009716771 0.03218259 0.0007128379  
## 16 16 0.03181327 0.2314083 0.02433694 0.0009500490 0.03170264 0.0007017497  
## 17 17 0.03183617 0.2302780 0.02435031 0.0009376987 0.03092469 0.0006962906  
## 18 18 0.03183845 0.2301986 0.02435592 0.0009452669 0.03114140 0.0006880128  
## 19 19 0.03184914 0.2296999 0.02436855 0.0009551448 0.03138209 0.0007032536  
## 20 20 0.03185594 0.2294060 0.02437378 0.0009664038 0.03239511 0.0007033834  
## 21 21 0.03187919 0.2283837 0.02439052 0.0009519169 0.03329335 0.0007015664  
## 22 22 0.03190704 0.2271638 0.02440136 0.0009550511 0.03317403 0.0007154118  
## 23 23 0.03191691 0.2267484 0.02441089 0.0009418347 0.03352612 0.0007213483  
## 24 24 0.03192985 0.2261911 0.02441290 0.0009443770 0.03370292 0.0007249876  
## 25 25 0.03192495 0.2264604 0.02441486 0.0009406213 0.03452215 0.0007406169  
## 26 26 0.03194979 0.2253044 0.02443159 0.0009278987 0.03414422 0.0007429836  
## 27 27 0.03196240 0.2247674 0.02445196 0.0009217274 0.03412584 0.0007194554  
## 28 28 0.03196470 0.2246520 0.02444916 0.0009071121 0.03399810 0.0007022156  
## 29 29 0.03197928 0.2239526 0.02446095 0.0009138522 0.03370508 0.0007192025  
## 30 30 0.03198829 0.2235770 0.02447161 0.0009111082 0.03338807 0.0007081585  
## 31 31 0.03200579 0.2228381 0.02448173 0.0009107781 0.03433836 0.0006997107  
## 32 32 0.03200830 0.2227260 0.02447846 0.0009205360 0.03350584 0.0007145439  
## 33 33 0.03200862 0.2228070 0.02447629 0.0009390494 0.03380812 0.0007307310  
## 34 34 0.03201683 0.2224914 0.02448290 0.0009346251 0.03427023 0.0007365608  
## 35 35 0.03203930 0.2214826 0.02449474 0.0009337344 0.03387500 0.0007342145  
## 36 36 0.03204240 0.2213557 0.02450163 0.0009377271 0.03369755 0.0007408053  
## 37 37 0.03204682 0.2211141 0.02450271 0.0009277245 0.03282401 0.0007389277  
## 38 38 0.03205517 0.2207327 0.02450666 0.0009127044 0.03170941 0.0007392199  
## 39 39 0.03205935 0.2205567 0.02451140 0.0009124235 0.03106077 0.0007296402  
## 40 40 0.03209326 0.2190353 0.02453271 0.0008974609 0.03090737 0.0007208886  
## 41 41 0.03210729 0.2183862 0.02455600 0.0008849834 0.03039413 0.0007100284  
## 42 42 0.03210858 0.2183310 0.02455770 0.0008847987 0.03010993 0.0007061946  
## 43 43 0.03210819 0.2184102 0.02455758 0.0008697577 0.03011648 0.0006947700  
## 44 44 0.03211291 0.2182125 0.02455920 0.0008525041 0.02974112 0.0006853697  
## 45 45 0.03212215 0.2178932 0.02455902 0.0008568634 0.03070538 0.0006815434  
## 46 46 0.03211027 0.2184210 0.02454284 0.0008608280 0.03031634 0.0006823394  
## 47 47 0.03212225 0.2178671 0.02454680 0.0008387033 0.03024669 0.0006764293  
## 48 48 0.03211927 0.2179999 0.02454752 0.0008430339 0.02987679 0.0006774018  
## 49 49 0.03214848 0.2166951 0.02457040 0.0008485584 0.02942640 0.0006822460  
## 50 50 0.03214576 0.2168975 0.02457116 0.0008492438 0.02997110 0.0006842655  
## 51 51 0.03214337 0.2170548 0.02456827 0.0008439948 0.02994518 0.0006759457  
## 52 52 0.03215279 0.2167278 0.02457977 0.0008463984 0.03043079 0.0006913543  
## 53 53 0.03214571 0.2171004 0.02456801 0.0008315171 0.03075208 0.0006811334  
## 54 54 0.03215215 0.2168667 0.02456936 0.0008269529 0.03127875 0.0006861038  
## 55 55 0.03216026 0.2165451 0.02457623 0.0008268153 0.03143554 0.0006902876  
## 56 56 0.03216813 0.2162466 0.02458294 0.0008297667 0.03173275 0.0006933392  
## 57 57 0.03216090 0.2166347 0.02458341 0.0008321183 0.03225746 0.0007020872  
## 58 58 0.03216409 0.2165183 0.02458881 0.0008354616 0.03185470 0.0006957211  
## 59 59 0.03216039 0.2166965 0.02458032 0.0008325708 0.03115281 0.0006887325  
## 60 60 0.03216281 0.2165869 0.02458268 0.0008327096 0.03120370 0.0006942422  
## 61 61 0.03216846 0.2163999 0.02458975 0.0008309512 0.03150908 0.0006876374  
## 62 62 0.03217593 0.2161207 0.02460154 0.0008126828 0.03175356 0.0006933178  
## 63 63 0.03218030 0.2160404 0.02460672 0.0008049996 0.03193166 0.0007027999  
## 64 64 0.03218441 0.2158447 0.02460938 0.0007996331 0.03197257 0.0007021321  
## 65 65 0.03217844 0.2161378 0.02459610 0.0007904210 0.03206352 0.0006866304  
## 66 66 0.03217217 0.2163902 0.02460129 0.0007865448 0.03181321 0.0006839267  
## 67 67 0.03218230 0.2159734 0.02460636 0.0007758522 0.03180017 0.0006754205  
## 68 68 0.03221113 0.2147368 0.02462667 0.0007778205 0.03131358 0.0006850258  
## 69 69 0.03220480 0.2150389 0.02461810 0.0007797424 0.03099273 0.0006866774  
## 70 70 0.03221709 0.2145167 0.02462214 0.0007768608 0.03040598 0.0006895947  
## 71 71 0.03221856 0.2144309 0.02463007 0.0007801180 0.03029029 0.0006869799  
## 72 72 0.03222021 0.2143210 0.02463221 0.0007828286 0.02957634 0.0006860606  
## 73 73 0.03221979 0.2143664 0.02463130 0.0007828361 0.02975783 0.0006934537  
## 74 74 0.03222475 0.2141290 0.02463989 0.0007777256 0.02973337 0.0006934751  
## 75 75 0.03223212 0.2137687 0.02463896 0.0007760534 0.02906969 0.0006980193  
## 76 76 0.03223595 0.2135317 0.02464641 0.0007617645 0.02827648 0.0006915552  
## 77 77 0.03224364 0.2132402 0.02464910 0.0007562328 0.02828353 0.0006909161  
## 78 78 0.03225409 0.2127970 0.02466541 0.0007489516 0.02815396 0.0006902001  
## 79 79 0.03225963 0.2125938 0.02466939 0.0007477855 0.02866004 0.0006983787  
## 80 80 0.03225892 0.2125903 0.02467307 0.0007521993 0.02830268 0.0006962832  
## 81 81 0.03225795 0.2126560 0.02467368 0.0007696140 0.02909548 0.0007054159  
## 82 82 0.03226620 0.2122813 0.02468165 0.0007699780 0.02896481 0.0007116349  
## 83 83 0.03228109 0.2116094 0.02469522 0.0007715956 0.02867277 0.0007163354  
## 84 84 0.03228461 0.2114691 0.02469188 0.0007707317 0.02874356 0.0007094470  
## 85 85 0.03228070 0.2116897 0.02469064 0.0007798976 0.02889215 0.0007147048  
## 86 86 0.03227474 0.2119583 0.02469003 0.0007819043 0.02896966 0.0007150485  
## 87 87 0.03227955 0.2117659 0.02469595 0.0007765925 0.02906446 0.0007037563  
## 88 88 0.03227858 0.2117907 0.02469897 0.0007763896 0.02883195 0.0007049740  
## 89 89 0.03229834 0.2109699 0.02470569 0.0007733157 0.02880175 0.0007069561  
## 90 90 0.03229843 0.2109561 0.02471012 0.0007766102 0.02872211 0.0007041537  
## 91 91 0.03230677 0.2106601 0.02471535 0.0007800595 0.02884240 0.0007141477  
## 92 92 0.03230452 0.2108093 0.02471648 0.0007844417 0.02939173 0.0007145437  
## 93 93 0.03230700 0.2107395 0.02471395 0.0007961061 0.02944829 0.0007164143  
## 94 94 0.03231522 0.2104209 0.02472539 0.0008119932 0.02953098 0.0007237155  
## 95 95 0.03232190 0.2101929 0.02473615 0.0008116688 0.02962364 0.0007280932  
## 96 96 0.03232443 0.2100947 0.02474491 0.0008076858 0.02930287 0.0007231928  
## 97 97 0.03232216 0.2101936 0.02473878 0.0008119030 0.02923928 0.0007231684  
## 98 98 0.03233250 0.2097811 0.02474818 0.0008131878 0.02936749 0.0007215730  
## 99 99 0.03232242 0.2102497 0.02473727 0.0008151488 0.02973073 0.0007198020  
## 100 100 0.03232778 0.2100177 0.02474332 0.0008170108 0.02943337 0.0007230854  
## 101 101 0.03233024 0.2099132 0.02473978 0.0008108845 0.02952312 0.0007226829  
## 102 102 0.03233111 0.2099035 0.02474455 0.0008128451 0.02969186 0.0007207915  
## 103 103 0.03233617 0.2096828 0.02474541 0.0008109729 0.02961718 0.0007203957  
## 104 104 0.03234168 0.2094913 0.02475465 0.0008069150 0.02982484 0.0007189370  
## 105 105 0.03233996 0.2095829 0.02475291 0.0008093026 0.03010674 0.0007139439  
## 106 106 0.03234638 0.2093300 0.02476098 0.0008120161 0.03008425 0.0007210948  
## 107 107 0.03234498 0.2094167 0.02475827 0.0008074171 0.03012874 0.0007212773  
## 108 108 0.03234033 0.2096527 0.02475065 0.0008079688 0.02996480 0.0007217876  
## 109 109 0.03233904 0.2097053 0.02474898 0.0008087998 0.02993065 0.0007182904  
## 110 110 0.03234274 0.2095323 0.02475493 0.0007998324 0.02986622 0.0007088089  
## 111 111 0.03233650 0.2097899 0.02474569 0.0008045457 0.02982015 0.0007088137  
## 112 112 0.03233185 0.2100114 0.02474781 0.0008008501 0.02965911 0.0007036683  
## 113 113 0.03232486 0.2103445 0.02474297 0.0007950425 0.02983956 0.0007021030  
## 114 114 0.03232952 0.2101538 0.02474641 0.0007883157 0.02968481 0.0006989633  
## 115 115 0.03232881 0.2101826 0.02475128 0.0007768132 0.02963550 0.0006894929  
## 116 116 0.03234083 0.2096659 0.02476029 0.0007762038 0.02974805 0.0006917415  
## 117 117 0.03233910 0.2097526 0.02476394 0.0007802917 0.03015826 0.0007007919  
## 118 118 0.03233947 0.2097229 0.02476218 0.0007787281 0.03004422 0.0006982999  
## 119 119 0.03234068 0.2096834 0.02476365 0.0007800962 0.03018764 0.0006975156  
## 120 120 0.03234798 0.2093964 0.02477396 0.0007949737 0.03019001 0.0007050733  
## 121 121 0.03236143 0.2088083 0.02478390 0.0007983467 0.03039458 0.0007097842  
## 122 122 0.03236898 0.2085174 0.02479013 0.0008000431 0.03066372 0.0007098026  
## 123 123 0.03237521 0.2082474 0.02479578 0.0008069369 0.03088126 0.0007059787  
## 124 124 0.03236644 0.2086360 0.02479187 0.0008110146 0.03103777 0.0007066265  
## 125 125 0.03236558 0.2086825 0.02479619 0.0008082953 0.03087962 0.0006979651  
## 126 126 0.03236544 0.2087278 0.02479497 0.0008117784 0.03135771 0.0007001419  
## 127 127 0.03236044 0.2089375 0.02479385 0.0008130063 0.03140138 0.0007014783  
## 128 128 0.03235966 0.2089444 0.02479370 0.0008129288 0.03112525 0.0006987793  
## 129 129 0.03235358 0.2092170 0.02478876 0.0008180548 0.03124019 0.0007045327  
## 130 130 0.03235446 0.2091757 0.02478881 0.0008167358 0.03125354 0.0007012115  
## 131 131 0.03236105 0.2088925 0.02479147 0.0008178104 0.03130095 0.0007048369  
## 132 132 0.03235920 0.2090064 0.02478850 0.0008175437 0.03130579 0.0007039038  
## 133 133 0.03235710 0.2091314 0.02478515 0.0008193783 0.03173765 0.0007006480  
## 134 134 0.03235681 0.2091270 0.02478638 0.0008192002 0.03150681 0.0006981622  
## 135 135 0.03235421 0.2092354 0.02478250 0.0008202481 0.03179505 0.0006974045  
## 136 136 0.03235882 0.2089931 0.02478502 0.0008186477 0.03159481 0.0006965681  
## 137 137 0.03235560 0.2091449 0.02478077 0.0008186925 0.03180867 0.0006990511  
## 138 138 0.03235623 0.2090805 0.02477955 0.0008230613 0.03160101 0.0007049195  
## 139 139 0.03236139 0.2088560 0.02478370 0.0008179348 0.03173124 0.0006992685  
## 140 140 0.03236430 0.2087368 0.02478480 0.0008130241 0.03178566 0.0006950272  
## 141 141 0.03236481 0.2086860 0.02478597 0.0008069117 0.03142359 0.0006862592  
## 142 142 0.03236137 0.2088223 0.02478236 0.0008019717 0.03136408 0.0006794954  
## 143 143 0.03236304 0.2087675 0.02478329 0.0008079240 0.03141430 0.0006834207  
## 144 144 0.03236890 0.2085264 0.02478788 0.0008065953 0.03127697 0.0006880368  
## 145 145 0.03236835 0.2085657 0.02478363 0.0008041996 0.03131719 0.0006895237  
## 146 146 0.03236910 0.2085623 0.02478697 0.0008023886 0.03115992 0.0006914361  
## 147 147 0.03236301 0.2088351 0.02478278 0.0007945046 0.03105264 0.0006896161  
## 148 148 0.03236791 0.2086253 0.02478286 0.0007971775 0.03093588 0.0006946474  
## 149 149 0.03236998 0.2085474 0.02478454 0.0007989548 0.03093670 0.0006979382  
## 150 150 0.03237160 0.2084977 0.02478424 0.0008000695 0.03108622 0.0006992724  
## 151 151 0.03237317 0.2084397 0.02478377 0.0007967891 0.03093180 0.0006959766  
## 152 152 0.03237050 0.2085473 0.02478472 0.0007949846 0.03061533 0.0006950441  
## 153 153 0.03237397 0.2084041 0.02478936 0.0007935540 0.03062925 0.0006934967  
## 154 154 0.03237077 0.2085532 0.02478837 0.0007965955 0.03065644 0.0006942539  
## 155 155 0.03236778 0.2086882 0.02478556 0.0007947311 0.03042426 0.0006970656  
## 156 156 0.03236867 0.2086713 0.02478370 0.0007992208 0.03055056 0.0006996147  
## 157 157 0.03236555 0.2088240 0.02478284 0.0008061523 0.03060487 0.0006988074  
## 158 158 0.03236560 0.2088286 0.02478298 0.0008071845 0.03071742 0.0006976375  
## 159 159 0.03236323 0.2089307 0.02478208 0.0008040640 0.03042081 0.0006999206  
## 160 160 0.03236695 0.2087795 0.02478484 0.0008063442 0.03041714 0.0007021976  
## 161 161 0.03236582 0.2088225 0.02478364 0.0008049967 0.03052618 0.0006996836  
## 162 162 0.03236492 0.2088789 0.02478311 0.0008077468 0.03057161 0.0007005460  
## 163 163 0.03237095 0.2086387 0.02478601 0.0008080746 0.03069019 0.0006984338  
## 164 164 0.03237373 0.2085110 0.02478610 0.0008092460 0.03053995 0.0007031693  
## 165 165 0.03237559 0.2084559 0.02478556 0.0008059758 0.03044749 0.0006990558  
## 166 166 0.03237219 0.2086144 0.02478054 0.0008058543 0.03043227 0.0007008185  
## 167 167 0.03237463 0.2085188 0.02478690 0.0008058157 0.03022685 0.0006969376  
## 168 168 0.03237834 0.2083772 0.02478987 0.0008059088 0.03039104 0.0006965310  
## 169 169 0.03237647 0.2084553 0.02479061 0.0008067594 0.03043680 0.0006963493  
## 170 170 0.03237530 0.2085100 0.02478840 0.0008041647 0.03057278 0.0006907242  
## 171 171 0.03237588 0.2084698 0.02478602 0.0008040746 0.03051165 0.0006908857  
## 172 172 0.03237636 0.2084599 0.02478702 0.0008010623 0.03058616 0.0006834484  
## 173 173 0.03237767 0.2083955 0.02478871 0.0007991120 0.03059040 0.0006805956  
## 174 174 0.03237859 0.2083924 0.02479174 0.0007999069 0.03072965 0.0006773469  
## 175 175 0.03237824 0.2083918 0.02478779 0.0008015160 0.03060320 0.0006804155  
## 176 176 0.03238247 0.2082172 0.02479342 0.0007977250 0.03051013 0.0006794817  
## 177 177 0.03238355 0.2081666 0.02479439 0.0007953878 0.03021590 0.0006757618  
## 178 178 0.03238177 0.2082479 0.02479577 0.0007933780 0.03020045 0.0006755593  
## 179 179 0.03238145 0.2082781 0.02479406 0.0007935923 0.03031883 0.0006780768  
## 180 180 0.03238411 0.2081670 0.02479473 0.0007950618 0.03031473 0.0006774598  
## 181 181 0.03238483 0.2081383 0.02479580 0.0007947813 0.03034939 0.0006756518  
## 182 182 0.03238851 0.2079929 0.02479825 0.0007940142 0.03040126 0.0006741174  
## 183 183 0.03239026 0.2079141 0.02480004 0.0007938974 0.03029138 0.0006729245  
## 184 184 0.03239227 0.2078397 0.02480328 0.0007968570 0.03038787 0.0006764973  
## 185 185 0.03239206 0.2078597 0.02480272 0.0007977335 0.03038237 0.0006789407  
## 186 186 0.03239320 0.2078121 0.02480398 0.0008020247 0.03049001 0.0006833217  
## 187 187 0.03239444 0.2077544 0.02480523 0.0008061548 0.03040498 0.0006850708  
## 188 188 0.03239525 0.2077349 0.02480713 0.0008090262 0.03044288 0.0006869666  
## 189 189 0.03239532 0.2077293 0.02480712 0.0008083098 0.03026011 0.0006851768  
## 190 190 0.03239774 0.2076194 0.02480998 0.0008085238 0.03034548 0.0006833917  
## 191 191 0.03239884 0.2075807 0.02481185 0.0008069499 0.03036360 0.0006834206  
## 192 192 0.03240020 0.2075416 0.02481341 0.0008057019 0.03044354 0.0006851773  
## 193 193 0.03240115 0.2075052 0.02481452 0.0008057891 0.03035687 0.0006866103  
## 194 194 0.03240381 0.2073896 0.02481603 0.0008063061 0.03022746 0.0006877247  
## 195 195 0.03240293 0.2074198 0.02481628 0.0008059850 0.03011368 0.0006867904  
## 196 196 0.03240194 0.2074516 0.02481527 0.0008055856 0.03003505 0.0006848746  
## 197 197 0.03240078 0.2075009 0.02481408 0.0008061274 0.02998831 0.0006855326  
## 198 198 0.03239677 0.2076723 0.02480984 0.0008063134 0.03001209 0.0006888173  
## 199 199 0.03239582 0.2077213 0.02481047 0.0008078701 0.02997146 0.0006911710  
## 200 200 0.03239260 0.2078487 0.02480602 0.0008082229 0.02991484 0.0006918175  
## 201 201 0.03239257 0.2078548 0.02480626 0.0008097737 0.02989751 0.0006932060  
## 202 202 0.03239540 0.2077331 0.02481089 0.0008082153 0.02988048 0.0006917240  
## 203 203 0.03239528 0.2077303 0.02481116 0.0008064706 0.02987127 0.0006898869  
## 204 204 0.03239568 0.2077087 0.02481082 0.0008064452 0.02981071 0.0006896892  
## 205 205 0.03239636 0.2076860 0.02481170 0.0008089197 0.02986576 0.0006921452  
## 206 206 0.03239710 0.2076514 0.02481278 0.0008081237 0.02992468 0.0006925743  
## 207 207 0.03239954 0.2075475 0.02481418 0.0008085610 0.02997954 0.0006923438  
## 208 208 0.03240024 0.2075224 0.02481354 0.0008086491 0.03003529 0.0006926726  
## 209 209 0.03239942 0.2075493 0.02481263 0.0008085179 0.03000634 0.0006933612  
## 210 210 0.03239923 0.2075564 0.02481237 0.0008070833 0.02994547 0.0006933037  
## 211 211 0.03239826 0.2075913 0.02481136 0.0008071926 0.02992519 0.0006939208  
## 212 212 0.03239800 0.2076000 0.02481051 0.0008065144 0.02990305 0.0006931590  
## 213 213 0.03239736 0.2076214 0.02481032 0.0008071987 0.02989720 0.0006933715  
## 214 214 0.03239786 0.2076028 0.02481028 0.0008071055 0.02989012 0.0006946509  
## 215 215 0.03239767 0.2076130 0.02480988 0.0008054066 0.02989333 0.0006938246  
## 216 216 0.03239697 0.2076483 0.02480957 0.0008057682 0.02993806 0.0006935698  
## 217 217 0.03239756 0.2076262 0.02480998 0.0008065609 0.02992503 0.0006943024  
## 218 218 0.03239652 0.2076673 0.02480844 0.0008079287 0.02996277 0.0006945053  
## 219 219 0.03239664 0.2076691 0.02480927 0.0008076442 0.03001166 0.0006943636  
## 220 220 0.03239691 0.2076547 0.02480955 0.0008071000 0.03002004 0.0006942007  
## 221 221 0.03239774 0.2076224 0.02481055 0.0008078536 0.02998321 0.0006942601  
## 222 222 0.03239744 0.2076398 0.02481049 0.0008076685 0.02999528 0.0006941037  
## 223 223 0.03239714 0.2076569 0.02481013 0.0008074580 0.02997602 0.0006933393  
## 224 224 0.03239681 0.2076675 0.02480995 0.0008069051 0.02997305 0.0006924176  
## 225 225 0.03239751 0.2076396 0.02480993 0.0008073367 0.03000847 0.0006931620  
## 226 226 0.03239684 0.2076676 0.02480894 0.0008077286 0.02999928 0.0006933119  
## 227 227 0.03239684 0.2076647 0.02480880 0.0008086006 0.02998916 0.0006940725  
## 228 228 0.03239696 0.2076600 0.02480917 0.0008082791 0.02996223 0.0006935391  
## 229 229 0.03239725 0.2076490 0.02480956 0.0008082831 0.02996963 0.0006930079  
## 230 230 0.03239755 0.2076363 0.02480947 0.0008078492 0.02996797 0.0006929937  
## 231 231 0.03239724 0.2076487 0.02480871 0.0008079005 0.02994713 0.0006935089  
## 232 232 0.03239709 0.2076543 0.02480875 0.0008085886 0.02995841 0.0006936891  
## 233 233 0.03239735 0.2076432 0.02480902 0.0008083724 0.02996337 0.0006934086  
## 234 234 0.03239749 0.2076374 0.02480902 0.0008081391 0.02996127 0.0006933539  
## 235 235 0.03239728 0.2076478 0.02480859 0.0008080836 0.02996168 0.0006936570  
## 236 236 0.03239728 0.2076493 0.02480841 0.0008081013 0.02996454 0.0006938541  
## 237 237 0.03239759 0.2076357 0.02480851 0.0008080369 0.02996750 0.0006937463  
## 238 238 0.03239754 0.2076391 0.02480848 0.0008079081 0.02997088 0.0006935764  
## 239 239 0.03239751 0.2076405 0.02480845 0.0008078407 0.02997007 0.0006935219  
## 240 240 0.03239750 0.2076409 0.02480845 0.0008078701 0.02996875 0.0006935553  
## [1] "Best Model"  
## nvmax  
## 11 11



## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

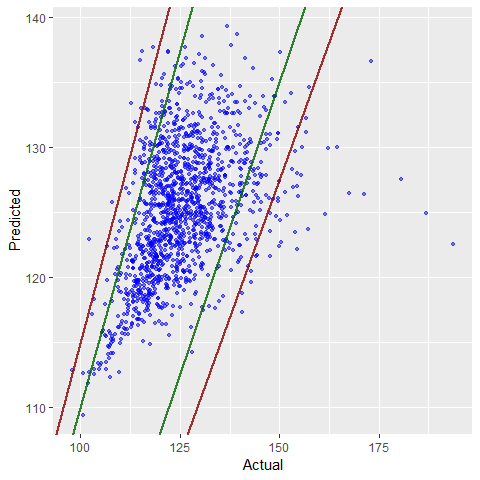


## [1] "Coefficients of final model:"  
## Estimate 2.5 % 97.5 %  
## (Intercept) 1.999228e+00 1.992687e+00 2.005768e+00  
## x4 -4.961729e-05 -6.718664e-05 -3.204795e-05  
## x7 1.138656e-02 1.015067e-02 1.262245e-02  
## x9 3.357169e-03 2.712443e-03 4.001895e-03  
## x10 1.139116e-03 5.398086e-04 1.738423e-03  
## x16 9.556884e-04 5.401986e-04 1.371178e-03  
## x17 1.357405e-03 7.256734e-04 1.989137e-03  
## stat14 -9.544654e-04 -1.430654e-03 -4.782771e-04  
## stat98 3.668932e-03 3.193272e-03 4.144592e-03  
## stat110 -3.579722e-03 -4.056280e-03 -3.103165e-03  
## stat146 -6.929336e-04 -1.174815e-03 -2.110520e-04  
## x18.sqrt 2.552906e-02 2.369269e-02 2.736542e-02

#### Test

if (algo.backward.caret == TRUE){  
 test.model(model.backward, data.test  
 ,method = 'leapBackward',subopt = NULL  
 ,formula = formula, feature.names = feature.names, label.names = label.names  
 ,id = id  
 ,draw.limits = TRUE, transformation = t)  
}

## [1] "Summary of predicted values: "  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2.039 2.085 2.098 2.097 2.110 2.144   
## [1] "leapBackward Test MSE: 0.000971233975337743"

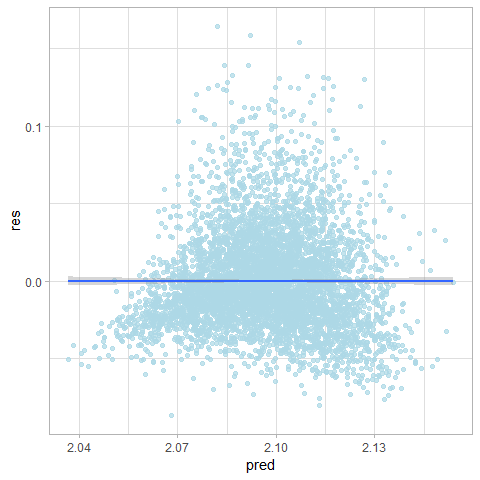
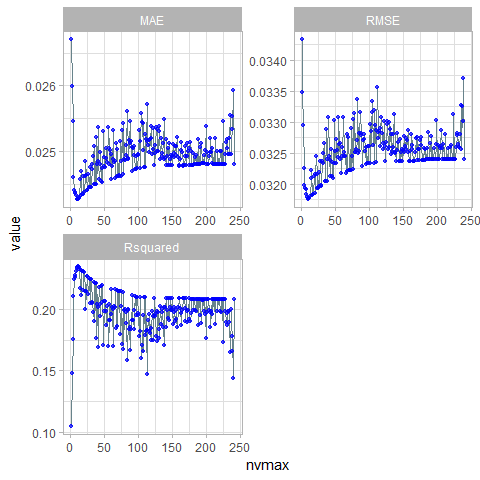


### Stepwise Selection with CV

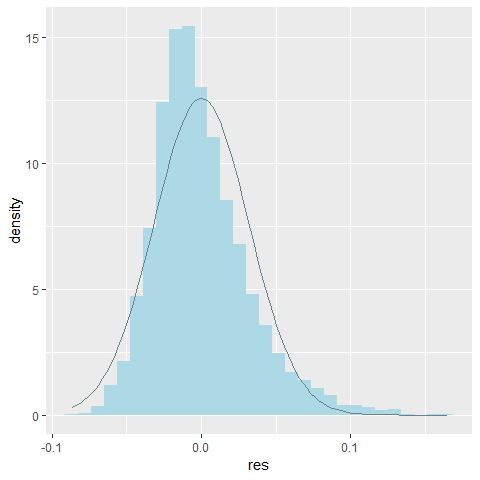
#### Train

if (algo.stepwise.caret == TRUE){  
 set.seed(1)  
 returned = train.caret.glmselect(formula = formula  
 ,data = data.train  
 ,method = "leapSeq"  
 ,feature.names = feature.names)  
 model.stepwise = returned$model  
 id = returned$id  
}

## Aggregating results  
## Selecting tuning parameters  
## Fitting nvmax = 11 on full training set  
## [1] "All models results"  
## nvmax RMSE Rsquared MAE RMSESD RsquaredSD MAESD  
## 1 1 0.03433074 0.1048652 0.02669982 0.0009261803 0.02117276 0.0006028036  
## 2 2 0.03348741 0.1480642 0.02598665 0.0009358138 0.02197751 0.0006196956  
## 3 3 0.03295185 0.1756793 0.02545556 0.0009655478 0.02940230 0.0006300332  
## 4 4 0.03225830 0.2100902 0.02460393 0.0010569927 0.03875907 0.0007157887  
## 5 5 0.03197411 0.2240441 0.02440412 0.0010723334 0.04180201 0.0007678828  
## 6 6 0.03193699 0.2259220 0.02437171 0.0010318862 0.03925322 0.0007656639  
## 7 7 0.03190821 0.2270003 0.02437009 0.0010314741 0.03680224 0.0007072754  
## 8 8 0.03183760 0.2303050 0.02432669 0.0010355665 0.03567746 0.0007199164  
## 9 9 0.03178572 0.2327543 0.02427982 0.0010201358 0.03519427 0.0007210205  
## 10 10 0.03176701 0.2336791 0.02427305 0.0010328434 0.03634847 0.0007382203  
## 11 11 0.03175425 0.2342165 0.02427299 0.0009983627 0.03444561 0.0007166804  
## 12 12 0.03177023 0.2334855 0.02429347 0.0010037969 0.03454673 0.0007159624  
## 13 13 0.03179507 0.2323029 0.02430312 0.0009982181 0.03356413 0.0007423370  
## 14 14 0.03209981 0.2164996 0.02457704 0.0015435783 0.06817586 0.0013250169  
## 15 15 0.03222618 0.2112625 0.02469418 0.0018966888 0.05983664 0.0014294061  
## 16 16 0.03181327 0.2314083 0.02433694 0.0009500490 0.03170264 0.0007017497  
## 17 17 0.03183806 0.2302426 0.02435454 0.0009385681 0.03097334 0.0006988639  
## 18 18 0.03183845 0.2301986 0.02435592 0.0009452669 0.03114140 0.0006880128  
## 19 19 0.03212733 0.2149298 0.02461678 0.0013825014 0.06621475 0.0011443242  
## 20 20 0.03185313 0.2295074 0.02438153 0.0009671527 0.03230596 0.0006934559  
## 21 21 0.03215945 0.2142361 0.02461495 0.0009421198 0.04921239 0.0008424286  
## 22 22 0.03244276 0.1995346 0.02485546 0.0013729145 0.06981918 0.0012796644  
## 23 23 0.03215772 0.2140636 0.02463717 0.0011565565 0.05868826 0.0007905172  
## 24 24 0.03192841 0.2262815 0.02440846 0.0009429044 0.03359696 0.0007180475  
## 25 25 0.03192495 0.2264604 0.02441486 0.0009406213 0.03452215 0.0007406169  
## 26 26 0.03219423 0.2123055 0.02464569 0.0014136222 0.06365716 0.0012535026  
## 27 27 0.03196350 0.2247245 0.02445037 0.0009213284 0.03406414 0.0007191229  
## 28 28 0.03237018 0.2047481 0.02476022 0.0014996373 0.06503908 0.0011404690  
## 29 29 0.03197138 0.2243609 0.02445168 0.0009173598 0.03407020 0.0007241480  
## 30 30 0.03197987 0.2239527 0.02446247 0.0009212191 0.03337186 0.0007190525  
## 31 31 0.03237373 0.2046383 0.02479960 0.0014514849 0.05478828 0.0013021871  
## 32 32 0.03232912 0.2056780 0.02472695 0.0010163843 0.06320278 0.0008479140  
## 33 33 0.03250021 0.1969521 0.02492861 0.0017452343 0.07874226 0.0014725749  
## 34 34 0.03266541 0.1899082 0.02507387 0.0018119829 0.06579513 0.0013159955  
## 35 35 0.03204110 0.2213609 0.02450226 0.0009308666 0.03398675 0.0007345564  
## 36 36 0.03204510 0.2212206 0.02450458 0.0009358962 0.03371891 0.0007380511  
## 37 37 0.03288341 0.1765012 0.02521502 0.0015451868 0.08953714 0.0014220998  
## 38 38 0.03205116 0.2209048 0.02450647 0.0009151077 0.03171351 0.0007393635  
## 39 39 0.03240503 0.2034907 0.02480485 0.0014495529 0.05397633 0.0013096256  
## 40 40 0.03308260 0.1686192 0.02536449 0.0014424330 0.08142893 0.0011582281  
## 41 41 0.03244600 0.2016641 0.02484818 0.0014325901 0.05339055 0.0012939708  
## 42 42 0.03211884 0.2179227 0.02456972 0.0008830995 0.03053845 0.0007065253  
## 43 43 0.03238394 0.2041347 0.02481194 0.0010850318 0.05344784 0.0009212166  
## 44 44 0.03256839 0.1936653 0.02498941 0.0014438450 0.07303248 0.0011954370  
## 45 45 0.03211871 0.2180394 0.02454926 0.0008580674 0.03010344 0.0006959837  
## 46 46 0.03235325 0.2056874 0.02475025 0.0008751956 0.04779634 0.0008488574  
## 47 47 0.03209707 0.2190559 0.02453306 0.0008586886 0.03018476 0.0006943208  
## 48 48 0.03303758 0.1704228 0.02530623 0.0015219959 0.08155716 0.0014360733  
## 49 49 0.03233819 0.2062732 0.02477393 0.0010529636 0.05382638 0.0007331117  
## 50 50 0.03253699 0.1979697 0.02491683 0.0016785597 0.04856821 0.0012994783  
## 51 51 0.03234021 0.2063019 0.02476511 0.0010538993 0.05420452 0.0007414347  
## 52 52 0.03272476 0.1865815 0.02505302 0.0010242634 0.06954121 0.0009030787  
## 53 53 0.03254238 0.1977846 0.02489140 0.0014352777 0.06178440 0.0011152588  
## 54 54 0.03242977 0.2024905 0.02483013 0.0010427273 0.05383218 0.0008893624  
## 55 55 0.03307545 0.1695034 0.02536182 0.0018662558 0.09045629 0.0015872932  
## 56 56 0.03264688 0.1907516 0.02505387 0.0011628501 0.06795129 0.0008747249  
## 57 57 0.03240591 0.2036337 0.02479936 0.0008556025 0.04935328 0.0008579470  
## 58 58 0.03265149 0.1909250 0.02501341 0.0017535108 0.07526235 0.0015138207  
## 59 59 0.03216992 0.2162251 0.02458325 0.0008185042 0.03135137 0.0006741400  
## 60 60 0.03259311 0.1958113 0.02495897 0.0016420401 0.04763773 0.0013085266  
## 61 61 0.03245097 0.2010747 0.02477610 0.0008826194 0.05868026 0.0007594969  
## 62 62 0.03307766 0.1695011 0.02532268 0.0013459422 0.07535954 0.0013082769  
## 63 63 0.03251417 0.1994462 0.02487630 0.0013699715 0.05416863 0.0012801205  
## 64 64 0.03244157 0.2019484 0.02483370 0.0012242813 0.06344782 0.0010749181  
## 65 65 0.03245450 0.2013560 0.02484527 0.0012180212 0.06367926 0.0010641707  
## 66 66 0.03217341 0.2163242 0.02459745 0.0007938411 0.03169275 0.0006947540  
## 67 67 0.03280331 0.1846187 0.02513105 0.0014307310 0.06527680 0.0013414614  
## 68 68 0.03220643 0.2149313 0.02462563 0.0007774359 0.03133741 0.0006747752  
## 69 69 0.03275660 0.1861439 0.02503357 0.0014472148 0.07302898 0.0012252410  
## 70 70 0.03222353 0.2142070 0.02462755 0.0007785586 0.03082117 0.0006903726  
## 71 71 0.03222905 0.2140180 0.02463263 0.0007771298 0.03094352 0.0006953961  
## 72 72 0.03222794 0.2140547 0.02462745 0.0007688617 0.02986250 0.0006915803  
## 73 73 0.03290643 0.1796512 0.02516159 0.0013759165 0.07463026 0.0011612732  
## 74 74 0.03222418 0.2141473 0.02462954 0.0007769196 0.02995974 0.0007023030  
## 75 75 0.03244368 0.2021625 0.02485415 0.0009972475 0.05471515 0.0007059730  
## 76 76 0.03304091 0.1714268 0.02532088 0.0014810930 0.07507993 0.0014598670  
## 77 77 0.03223288 0.2136956 0.02464651 0.0007515106 0.02819379 0.0006976875  
## 78 78 0.03313916 0.1669311 0.02542219 0.0015748554 0.07108102 0.0011844009  
## 79 79 0.03250634 0.1990807 0.02488372 0.0012003731 0.06188145 0.0010668282  
## 80 80 0.03250908 0.1998430 0.02487232 0.0014367063 0.05563275 0.0012075173  
## 81 81 0.03256882 0.1970478 0.02493597 0.0013238055 0.05229306 0.0012689778  
## 82 82 0.03270369 0.1884687 0.02508686 0.0009359021 0.06189079 0.0007709490  
## 83 83 0.03336817 0.1580157 0.02560948 0.0019998450 0.07645649 0.0016293772  
## 84 84 0.03249593 0.1999696 0.02487664 0.0007877135 0.04495494 0.0008423719  
## 85 85 0.03253151 0.1989376 0.02490258 0.0014485186 0.05576488 0.0012262617  
## 86 86 0.03281735 0.1838133 0.02517240 0.0014729905 0.06853682 0.0012452408  
## 87 87 0.03316131 0.1666662 0.02546636 0.0017599045 0.06731746 0.0015508971  
## 88 88 0.03278971 0.1852621 0.02513217 0.0017375216 0.07311527 0.0015259787  
## 89 89 0.03281912 0.1832760 0.02515818 0.0013121688 0.07282690 0.0011647591  
## 90 90 0.03228582 0.2114643 0.02469347 0.0007879155 0.02846139 0.0007230398  
## 91 91 0.03271189 0.1911667 0.02506259 0.0016408795 0.04765376 0.0013380051  
## 92 92 0.03271965 0.1909147 0.02505140 0.0014186889 0.06035296 0.0011827428  
## 93 93 0.03279784 0.1837965 0.02507866 0.0007443227 0.06232840 0.0008343605  
## 94 94 0.03253195 0.1985553 0.02494331 0.0010170119 0.05493301 0.0007319738  
## 95 95 0.03232330 0.2100968 0.02473412 0.0008018083 0.02960922 0.0007278768  
## 96 96 0.03259845 0.1950057 0.02492949 0.0008397884 0.05572425 0.0007735390  
## 97 97 0.03232894 0.2098358 0.02473397 0.0007982016 0.02939284 0.0007253713  
## 98 98 0.03233400 0.2096887 0.02474434 0.0008048868 0.02956716 0.0007358941  
## 99 99 0.03260421 0.1961719 0.02496371 0.0014765575 0.05619506 0.0012480340  
## 100 100 0.03260739 0.1947172 0.02494084 0.0008344518 0.05493398 0.0007621502  
## 101 101 0.03296629 0.1778920 0.02532160 0.0017071366 0.06223830 0.0012732527  
## 102 102 0.03233633 0.2096546 0.02474978 0.0008095783 0.02942577 0.0007187076  
## 103 103 0.03285521 0.1815186 0.02515622 0.0012727470 0.07311913 0.0011839186  
## 104 104 0.03334021 0.1597317 0.02558733 0.0018786648 0.08377485 0.0016775005  
## 105 105 0.03315989 0.1704345 0.02544631 0.0019108577 0.06429461 0.0015303040  
## 106 106 0.03262722 0.1947268 0.02501381 0.0010232743 0.05357831 0.0009196278  
## 107 107 0.03318707 0.1655236 0.02543404 0.0014559895 0.07769072 0.0012611774  
## 108 108 0.03233949 0.2096568 0.02475451 0.0008093832 0.02990732 0.0007244723  
## 109 109 0.03295712 0.1789308 0.02526840 0.0018387804 0.07072152 0.0016512660  
## 110 110 0.03234398 0.2095202 0.02475669 0.0008083945 0.02998933 0.0007147289  
## 111 111 0.03286912 0.1815271 0.02519153 0.0009685539 0.06239068 0.0009563255  
## 112 112 0.03355846 0.1473256 0.02572254 0.0017464590 0.07101603 0.0014956849  
## 113 113 0.03283004 0.1828432 0.02515481 0.0009936734 0.07101217 0.0007117888  
## 114 114 0.03273725 0.1908118 0.02509512 0.0014166241 0.06039962 0.0011947479  
## 115 115 0.03285924 0.1816167 0.02516977 0.0012708096 0.07390152 0.0011745167  
## 116 116 0.03260545 0.1958042 0.02498136 0.0012413312 0.06303332 0.0010935388  
## 117 117 0.03289269 0.1802490 0.02518859 0.0011992986 0.07623346 0.0010813638  
## 118 118 0.03307118 0.1743049 0.02537938 0.0017728435 0.07295978 0.0016015615  
## 119 119 0.03281852 0.1839992 0.02519815 0.0013539298 0.07608451 0.0010732573  
## 120 120 0.03256386 0.1982355 0.02496139 0.0008845305 0.04533374 0.0007624231  
## 121 121 0.03265331 0.1943753 0.02503106 0.0011664586 0.04904626 0.0010007175  
## 122 122 0.03301283 0.1747613 0.02534554 0.0011335695 0.06228576 0.0009167072  
## 123 123 0.03236536 0.2086641 0.02478595 0.0007862413 0.03058242 0.0006893431  
## 124 124 0.03260704 0.1957417 0.02497861 0.0007346636 0.04956893 0.0007123503  
## 125 125 0.03268141 0.1937077 0.02509520 0.0013510298 0.03367265 0.0011346026  
## 126 126 0.03295898 0.1778719 0.02536217 0.0016405801 0.07304886 0.0013567184  
## 127 127 0.03302139 0.1737887 0.02533381 0.0009398852 0.06846583 0.0010144038  
## 128 128 0.03255672 0.1984670 0.02499736 0.0012011256 0.05345885 0.0011231461  
## 129 129 0.03259490 0.1974967 0.02500007 0.0013858482 0.05180850 0.0011730632  
## 130 130 0.03300201 0.1763703 0.02537586 0.0016497454 0.06824804 0.0014101250  
## 131 131 0.03287822 0.1832838 0.02530662 0.0015857674 0.05223091 0.0013990348  
## 132 132 0.03248265 0.2023149 0.02488413 0.0009193564 0.04472900 0.0006465169  
## 133 133 0.03280742 0.1876915 0.02520309 0.0014965580 0.04037576 0.0012372220  
## 134 134 0.03253401 0.1995780 0.02492415 0.0007087236 0.04231006 0.0006651799  
## 135 135 0.03259070 0.1976669 0.02497255 0.0010288809 0.04138111 0.0008552733  
## 136 136 0.03280694 0.1851719 0.02512183 0.0009481391 0.05354710 0.0007901139  
## 137 137 0.03312704 0.1708324 0.02549569 0.0015496291 0.05063995 0.0014037783  
## 138 138 0.03250219 0.2015564 0.02488374 0.0010239569 0.03792669 0.0009175652  
## 139 139 0.03251162 0.2011136 0.02489122 0.0010260122 0.03828702 0.0009189762  
## 140 140 0.03236292 0.2087944 0.02478259 0.0008132541 0.03173845 0.0006979045  
## 141 141 0.03281378 0.1849986 0.02518104 0.0011111042 0.05957621 0.0010308355  
## 142 142 0.03253790 0.1993166 0.02491960 0.0006841420 0.04230752 0.0006429052  
## 143 143 0.03268113 0.1920860 0.02504143 0.0011336485 0.05987955 0.0008454518  
## 144 144 0.03236572 0.2086519 0.02478361 0.0008122149 0.03128736 0.0006927376  
## 145 145 0.03272942 0.1905565 0.02512051 0.0012312061 0.05047139 0.0010645690  
## 146 146 0.03237204 0.2084235 0.02478961 0.0008013001 0.03124693 0.0006893251  
## 147 147 0.03250780 0.2010807 0.02487397 0.0007106102 0.03694675 0.0006925988  
## 148 148 0.03255473 0.1989844 0.02493809 0.0010573291 0.05100973 0.0009143214  
## 149 149 0.03236755 0.2086423 0.02478381 0.0007987109 0.03081185 0.0006989031  
## 150 150 0.03254615 0.1990283 0.02492555 0.0006913349 0.04249469 0.0006751349  
## 151 151 0.03277642 0.1877656 0.02510473 0.0008838842 0.04784145 0.0008037456  
## 152 152 0.03251433 0.2010000 0.02491506 0.0009286672 0.04603276 0.0006405269  
## 153 153 0.03260305 0.1971640 0.02497434 0.0010194661 0.04117377 0.0008625039  
## 154 154 0.03273445 0.1899729 0.02507553 0.0012455698 0.05537255 0.0011248369  
## 155 155 0.03256292 0.1987538 0.02494220 0.0010630780 0.05121355 0.0009120674  
## 156 156 0.03270546 0.1913268 0.02508002 0.0010608590 0.05202637 0.0010830135  
## 157 157 0.03286344 0.1850482 0.02521668 0.0013143340 0.03426686 0.0010598808  
## 158 158 0.03251242 0.2011867 0.02490922 0.0009423395 0.04627546 0.0006459176  
## 159 159 0.03236498 0.2088814 0.02478143 0.0008042800 0.03064416 0.0007031043  
## 160 160 0.03236806 0.2087386 0.02478579 0.0008059732 0.03061696 0.0007027660  
## 161 161 0.03252998 0.2002330 0.02488305 0.0007182969 0.03743561 0.0007047780  
## 162 162 0.03280185 0.1871620 0.02513652 0.0012205285 0.05647033 0.0010303964  
## 163 163 0.03262956 0.1965260 0.02502339 0.0011984476 0.02657848 0.0009729424  
## 164 164 0.03256098 0.1991660 0.02499115 0.0011734984 0.05013291 0.0011321605  
## 165 165 0.03237559 0.2084559 0.02478556 0.0008059758 0.03044749 0.0006990558  
## 166 166 0.03237219 0.2086144 0.02478054 0.0008058543 0.03043227 0.0007008185  
## 167 167 0.03237222 0.2086019 0.02478447 0.0008031962 0.03013096 0.0006993161  
## 168 168 0.03251967 0.2009413 0.02491982 0.0009338253 0.04548406 0.0006489932  
## 169 169 0.03274262 0.1900824 0.02508498 0.0011743710 0.04898670 0.0010735773  
## 170 170 0.03237575 0.2084856 0.02478934 0.0008044781 0.03056596 0.0006922446  
## 171 171 0.03275932 0.1893465 0.02512883 0.0010953012 0.05677484 0.0009459805  
## 172 172 0.03256183 0.1984887 0.02492651 0.0006989060 0.04273120 0.0006655366  
## 173 173 0.03256468 0.1991212 0.02500029 0.0011625532 0.04973515 0.0011220074  
## 174 174 0.03237874 0.2083884 0.02479076 0.0008007299 0.03074901 0.0006780193  
## 175 175 0.03270072 0.1912465 0.02505335 0.0008079591 0.05290865 0.0005774235  
## 176 176 0.03274986 0.1899671 0.02514011 0.0012462617 0.05084291 0.0010730720  
## 177 177 0.03261875 0.1967240 0.02499965 0.0010172601 0.04049827 0.0008703861  
## 178 178 0.03238177 0.2082479 0.02479577 0.0007933780 0.03020045 0.0006755593  
## 179 179 0.03238289 0.2082128 0.02479404 0.0007960658 0.03034640 0.0006780530  
## 180 180 0.03238356 0.2081979 0.02479362 0.0007958783 0.03032880 0.0006782025  
## 181 181 0.03257433 0.1988925 0.02497361 0.0012344799 0.04510522 0.0010606961  
## 182 182 0.03238775 0.2080281 0.02479792 0.0007952956 0.03040547 0.0006745077  
## 183 183 0.03255974 0.1991620 0.02491195 0.0007260188 0.03816557 0.0006916738  
## 184 184 0.03239223 0.2078341 0.02480378 0.0007967953 0.03039034 0.0006772071  
## 185 185 0.03257778 0.1978876 0.02494158 0.0006985478 0.04295520 0.0006565143  
## 186 186 0.03239356 0.2078033 0.02480416 0.0008020828 0.03050205 0.0006831010  
## 187 187 0.03283070 0.1865873 0.02519133 0.0013693392 0.05097088 0.0011682619  
## 188 188 0.03239529 0.2077363 0.02480641 0.0008090850 0.03044222 0.0006859409  
## 189 189 0.03239625 0.2076958 0.02480765 0.0008099361 0.03027613 0.0006859452  
## 190 190 0.03266302 0.1951616 0.02506207 0.0012162988 0.02766843 0.0009959245  
## 191 191 0.03239884 0.2075807 0.02481185 0.0008069499 0.03036360 0.0006834206  
## 192 192 0.03276739 0.1886578 0.02512393 0.0011319134 0.05210986 0.0010143278  
## 193 193 0.03240035 0.2075356 0.02481419 0.0008043832 0.03034208 0.0006861373  
## 194 194 0.03240320 0.2074163 0.02481569 0.0008052200 0.03021430 0.0006872370  
## 195 195 0.03240339 0.2074011 0.02481691 0.0008067983 0.03012294 0.0006876968  
## 196 196 0.03257713 0.1985190 0.02493657 0.0007448412 0.03847108 0.0007124699  
## 197 197 0.03240056 0.2075118 0.02481540 0.0008057420 0.02998300 0.0006874204  
## 198 198 0.03264104 0.1958841 0.02501537 0.0010377467 0.04080355 0.0008844765  
## 199 199 0.03303509 0.1761807 0.02538292 0.0013764362 0.04562679 0.0011950708  
## 200 200 0.03239260 0.2078487 0.02480602 0.0008082229 0.02991484 0.0006918175  
## 201 201 0.03261964 0.1970820 0.02499361 0.0011442000 0.05301152 0.0009781120  
## 202 202 0.03239592 0.2077130 0.02481101 0.0008085811 0.02987503 0.0006919282  
## 203 203 0.03239528 0.2077303 0.02481116 0.0008064706 0.02987127 0.0006898869  
## 204 204 0.03257386 0.1987099 0.02492907 0.0007482520 0.03845007 0.0007162259  
## 205 205 0.03261415 0.1970495 0.02499407 0.0011704973 0.04289943 0.0011011381  
## 206 206 0.03286160 0.1851385 0.02520313 0.0013040679 0.04868340 0.0012070696  
## 207 207 0.03258133 0.1983886 0.02499074 0.0008432763 0.04008857 0.0007425610  
## 208 208 0.03240024 0.2075224 0.02481354 0.0008086491 0.03003529 0.0006926726  
## 209 209 0.03265003 0.1958061 0.02505410 0.0011857894 0.02603468 0.0009854286  
## 210 210 0.03239923 0.2075564 0.02481237 0.0008070833 0.02994547 0.0006933037  
## 211 211 0.03239826 0.2075913 0.02481136 0.0008071926 0.02992519 0.0006939208  
## 212 212 0.03281055 0.1878337 0.02519814 0.0012640406 0.04179478 0.0009201419  
## 213 213 0.03239736 0.2076214 0.02481032 0.0008071987 0.02989720 0.0006933715  
## 214 214 0.03239786 0.2076028 0.02481028 0.0008071055 0.02989012 0.0006946509  
## 215 215 0.03239767 0.2076130 0.02480988 0.0008054066 0.02989333 0.0006938246  
## 216 216 0.03258451 0.1982525 0.02499827 0.0008501374 0.04069003 0.0007651182  
## 217 217 0.03239756 0.2076262 0.02480998 0.0008065609 0.02992503 0.0006943024  
## 218 218 0.03239652 0.2076673 0.02480844 0.0008079287 0.02996277 0.0006945053  
## 219 219 0.03239664 0.2076691 0.02480927 0.0008076442 0.03001166 0.0006943636  
## 220 220 0.03258572 0.1981952 0.02500024 0.0008522858 0.04085267 0.0007687091  
## 221 221 0.03262272 0.1969695 0.02500661 0.0011355277 0.05263324 0.0009943666  
## 222 222 0.03239744 0.2076398 0.02481049 0.0008076685 0.02999528 0.0006941037  
## 223 223 0.03239714 0.2076569 0.02481013 0.0008074580 0.02997602 0.0006933393  
## 224 224 0.03312110 0.1732893 0.02545612 0.0014565752 0.04967609 0.0012253485  
## 225 225 0.03239751 0.2076396 0.02480993 0.0008073367 0.03000847 0.0006931620  
## 226 226 0.03239684 0.2076676 0.02480894 0.0008077286 0.02999928 0.0006933119  
## 227 227 0.03239684 0.2076647 0.02480880 0.0008086006 0.02998916 0.0006940725  
## 228 228 0.03259555 0.1978779 0.02495642 0.0007692385 0.04008207 0.0007519870  
## 229 229 0.03278750 0.1890003 0.02514504 0.0012275204 0.06210529 0.0009198127  
## 230 230 0.03282339 0.1867986 0.02519077 0.0012096936 0.05048970 0.0011453541  
## 231 231 0.03263504 0.1968896 0.02504748 0.0012994519 0.05347158 0.0012185505  
## 232 232 0.03259890 0.1977612 0.02495492 0.0007731096 0.04031771 0.0007502331  
## 233 233 0.03259910 0.1977533 0.02495513 0.0007730290 0.04032378 0.0007499943  
## 234 234 0.03280244 0.1888410 0.02519394 0.0013815307 0.06290778 0.0011639263  
## 235 235 0.03327159 0.1652459 0.02554565 0.0011171066 0.06139351 0.0011754282  
## 236 236 0.03256341 0.1996743 0.02495485 0.0009774268 0.04659836 0.0006589850  
## 237 237 0.03325127 0.1660522 0.02553978 0.0015099360 0.06722577 0.0014312696  
## 238 238 0.03300786 0.1776559 0.02533877 0.0013190248 0.06426026 0.0011974207  
## 239 239 0.03371248 0.1439361 0.02593720 0.0012714371 0.04843086 0.0011851233  
## 240 240 0.03239750 0.2076409 0.02480845 0.0008078701 0.02996875 0.0006935553  
## [1] "Best Model"  
## nvmax  
## 11 11



## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

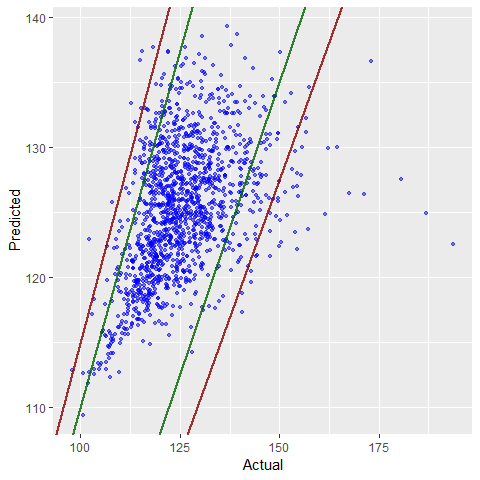


## [1] "Coefficients of final model:"  
## Estimate 2.5 % 97.5 %  
## (Intercept) 1.999228e+00 1.992687e+00 2.005768e+00  
## x4 -4.961729e-05 -6.718664e-05 -3.204795e-05  
## x7 1.138656e-02 1.015067e-02 1.262245e-02  
## x9 3.357169e-03 2.712443e-03 4.001895e-03  
## x10 1.139116e-03 5.398086e-04 1.738423e-03  
## x16 9.556884e-04 5.401986e-04 1.371178e-03  
## x17 1.357405e-03 7.256734e-04 1.989137e-03  
## stat14 -9.544654e-04 -1.430654e-03 -4.782771e-04  
## stat98 3.668932e-03 3.193272e-03 4.144592e-03  
## stat110 -3.579722e-03 -4.056280e-03 -3.103165e-03  
## stat146 -6.929336e-04 -1.174815e-03 -2.110520e-04  
## x18.sqrt 2.552906e-02 2.369269e-02 2.736542e-02

#### Test

if (algo.stepwise.caret == TRUE){  
 test.model(model.stepwise, data.test  
 ,method = 'leapSeq',subopt = NULL  
 ,formula = formula, feature.names = feature.names, label.names = label.names  
 ,id = id  
 ,draw.limits = TRUE, transformation = t)  
   
}

## [1] "Summary of predicted values: "  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2.039 2.085 2.098 2.097 2.110 2.144   
## [1] "leapSeq Test MSE: 0.000971233975337744"



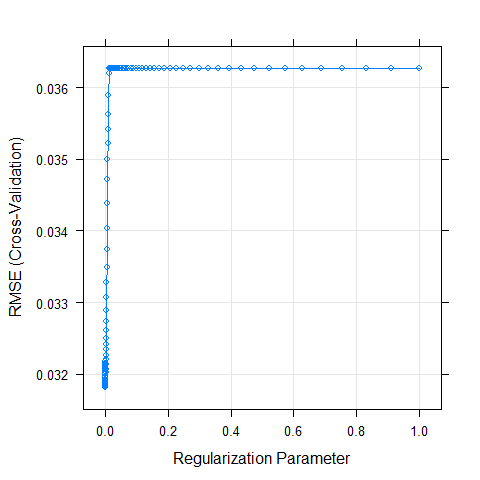
### LASSO with CV

#### Train

if (algo.LASSO.caret == TRUE){  
 set.seed(1)  
 tune.grid= expand.grid(alpha = 1,lambda = 10^seq(from=-4,to=0,length=100))  
 returned = train.caret.glmselect(formula = formula  
 ,data = data.train  
 ,method = "glmnet"  
 ,subopt = 'LASSO'  
 ,tune.grid = tune.grid  
 ,feature.names = feature.names)  
 model.LASSO.caret = returned$model  
}

## Warning in nominalTrainWorkflow(x = x, y = y, wts = weights, info = trainInfo, : There were missing values in resampled  
## performance measures.

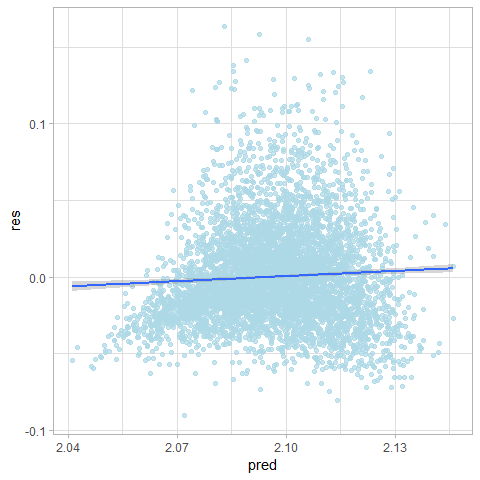
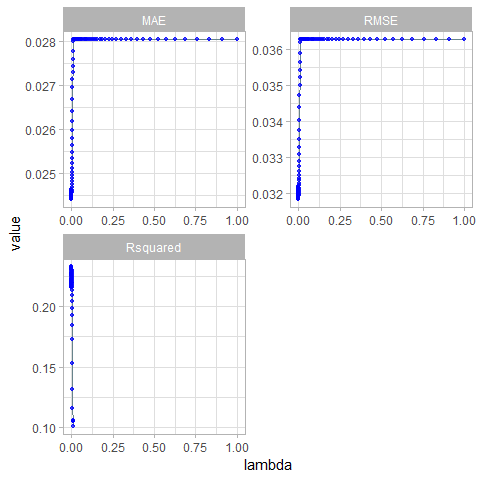
## Aggregating results  
## Selecting tuning parameters  
## Fitting alpha = 1, lambda = 0.000643 on full training set  
## glmnet   
##   
## 5584 samples  
## 240 predictor  
##   
## No pre-processing  
## Resampling: Cross-Validated (10 fold)   
## Summary of sample sizes: 5026, 5026, 5026, 5025, 5025, 5026, ...   
## Resampling results across tuning parameters:  
##   
## lambda RMSE Rsquared MAE   
## 0.0001000000 0.03217337 0.2155012 0.02463278  
## 0.0001097499 0.03215639 0.2161334 0.02461969  
## 0.0001204504 0.03213894 0.2167887 0.02460628  
## 0.0001321941 0.03212070 0.2174827 0.02459214  
## 0.0001450829 0.03210128 0.2182349 0.02457704  
## 0.0001592283 0.03208098 0.2190347 0.02456134  
## 0.0001747528 0.03206038 0.2198589 0.02454525  
## 0.0001917910 0.03203949 0.2207111 0.02452975  
## 0.0002104904 0.03201773 0.2216227 0.02451385  
## 0.0002310130 0.03199599 0.2225554 0.02449799  
## 0.0002535364 0.03197424 0.2235123 0.02448262  
## 0.0002782559 0.03195219 0.2245166 0.02446717  
## 0.0003053856 0.03193130 0.2254994 0.02445224  
## 0.0003351603 0.03191052 0.2265202 0.02443821  
## 0.0003678380 0.03189097 0.2275323 0.02442645  
## 0.0004037017 0.03187220 0.2285611 0.02441528  
## 0.0004430621 0.03185624 0.2295034 0.02440851  
## 0.0004862602 0.03184372 0.2303223 0.02440506  
## 0.0005336699 0.03183287 0.2311139 0.02440137  
## 0.0005857021 0.03182464 0.2318345 0.02439922  
## 0.0006428073 0.03182206 0.2323331 0.02440047  
## 0.0007054802 0.03182595 0.2325738 0.02440519  
## 0.0007742637 0.03183325 0.2327119 0.02441536  
## 0.0008497534 0.03184640 0.2326222 0.02443330  
## 0.0009326033 0.03186904 0.2321178 0.02445805  
## 0.0010235310 0.03189824 0.2313681 0.02448761  
## 0.0011233240 0.03193218 0.2304840 0.02452334  
## 0.0012328467 0.03197213 0.2294074 0.02456623  
## 0.0013530478 0.03201893 0.2281107 0.02461400  
## 0.0014849683 0.03207297 0.2265796 0.02466883  
## 0.0016297508 0.03213479 0.2247982 0.02473255  
## 0.0017886495 0.03220083 0.2230051 0.02480054  
## 0.0019630407 0.03226906 0.2213827 0.02487197  
## 0.0021544347 0.03233923 0.2200443 0.02494548  
## 0.0023644894 0.03241375 0.2189542 0.02502305  
## 0.0025950242 0.03250177 0.2175790 0.02511229  
## 0.0028480359 0.03260807 0.2156641 0.02521439  
## 0.0031257158 0.03273564 0.2130434 0.02533174  
## 0.0034304693 0.03288858 0.2094025 0.02546668  
## 0.0037649358 0.03307180 0.2042516 0.02562343  
## 0.0041320124 0.03327698 0.1981400 0.02579431  
## 0.0045348785 0.03348909 0.1925040 0.02597168  
## 0.0049770236 0.03373466 0.1848057 0.02617109  
## 0.0054622772 0.03402773 0.1726919 0.02640489  
## 0.0059948425 0.03437705 0.1531354 0.02667796  
## 0.0065793322 0.03472129 0.1312232 0.02693751  
## 0.0072208090 0.03499622 0.1161551 0.02713454  
## 0.0079248290 0.03522312 0.1061937 0.02729185  
## 0.0086974900 0.03541329 0.1048652 0.02742392  
## 0.0095454846 0.03563082 0.1048652 0.02757849  
## 0.0104761575 0.03589111 0.1048652 0.02776428  
## 0.0114975700 0.03620125 0.1010599 0.02798513  
## 0.0126185688 0.03626118 NaN 0.02802751  
## 0.0138488637 0.03626118 NaN 0.02802751  
## 0.0151991108 0.03626118 NaN 0.02802751  
## 0.0166810054 0.03626118 NaN 0.02802751  
## 0.0183073828 0.03626118 NaN 0.02802751  
## 0.0200923300 0.03626118 NaN 0.02802751  
## 0.0220513074 0.03626118 NaN 0.02802751  
## 0.0242012826 0.03626118 NaN 0.02802751  
## 0.0265608778 0.03626118 NaN 0.02802751  
## 0.0291505306 0.03626118 NaN 0.02802751  
## 0.0319926714 0.03626118 NaN 0.02802751  
## 0.0351119173 0.03626118 NaN 0.02802751  
## 0.0385352859 0.03626118 NaN 0.02802751  
## 0.0422924287 0.03626118 NaN 0.02802751  
## 0.0464158883 0.03626118 NaN 0.02802751  
## 0.0509413801 0.03626118 NaN 0.02802751  
## 0.0559081018 0.03626118 NaN 0.02802751  
## 0.0613590727 0.03626118 NaN 0.02802751  
## 0.0673415066 0.03626118 NaN 0.02802751  
## 0.0739072203 0.03626118 NaN 0.02802751  
## 0.0811130831 0.03626118 NaN 0.02802751  
## 0.0890215085 0.03626118 NaN 0.02802751  
## 0.0977009957 0.03626118 NaN 0.02802751  
## 0.1072267222 0.03626118 NaN 0.02802751  
## 0.1176811952 0.03626118 NaN 0.02802751  
## 0.1291549665 0.03626118 NaN 0.02802751  
## 0.1417474163 0.03626118 NaN 0.02802751  
## 0.1555676144 0.03626118 NaN 0.02802751  
## 0.1707352647 0.03626118 NaN 0.02802751  
## 0.1873817423 0.03626118 NaN 0.02802751  
## 0.2056512308 0.03626118 NaN 0.02802751  
## 0.2257019720 0.03626118 NaN 0.02802751  
## 0.2477076356 0.03626118 NaN 0.02802751  
## 0.2718588243 0.03626118 NaN 0.02802751  
## 0.2983647240 0.03626118 NaN 0.02802751  
## 0.3274549163 0.03626118 NaN 0.02802751  
## 0.3593813664 0.03626118 NaN 0.02802751  
## 0.3944206059 0.03626118 NaN 0.02802751  
## 0.4328761281 0.03626118 NaN 0.02802751  
## 0.4750810162 0.03626118 NaN 0.02802751  
## 0.5214008288 0.03626118 NaN 0.02802751  
## 0.5722367659 0.03626118 NaN 0.02802751  
## 0.6280291442 0.03626118 NaN 0.02802751  
## 0.6892612104 0.03626118 NaN 0.02802751  
## 0.7564633276 0.03626118 NaN 0.02802751  
## 0.8302175681 0.03626118 NaN 0.02802751  
## 0.9111627561 0.03626118 NaN 0.02802751  
## 1.0000000000 0.03626118 NaN 0.02802751  
##   
## Tuning parameter 'alpha' was held constant at a value of 1  
## RMSE was used to select the optimal model using the smallest value.  
## The final values used for the model were alpha = 1 and lambda = 0.0006428073.



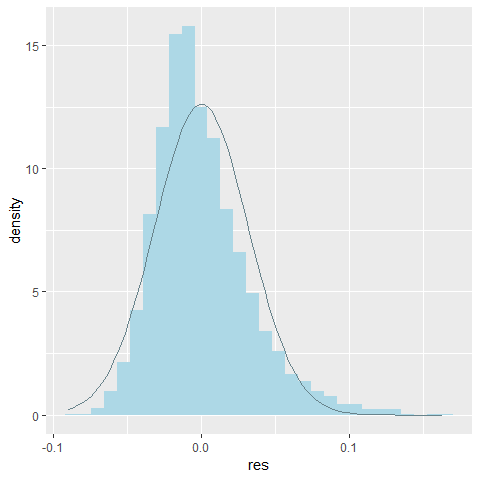
## alpha lambda  
## 21 1 0.0006428073  
## alpha lambda RMSE Rsquared MAE RMSESD RsquaredSD MAESD  
## 1 1 0.0001000000 0.03217337 0.2155012 0.02463278 0.0008173492 0.03084170 0.0006913930  
## 2 1 0.0001097499 0.03215639 0.2161334 0.02461969 0.0008187873 0.03090224 0.0006916334  
## 3 1 0.0001204504 0.03213894 0.2167887 0.02460628 0.0008206847 0.03097215 0.0006920436  
## 4 1 0.0001321941 0.03212070 0.2174827 0.02459214 0.0008231023 0.03103911 0.0006926507  
## 5 1 0.0001450829 0.03210128 0.2182349 0.02457704 0.0008262124 0.03110180 0.0006936529  
## 6 1 0.0001592283 0.03208098 0.2190347 0.02456134 0.0008299029 0.03115870 0.0006945017  
## 7 1 0.0001747528 0.03206038 0.2198589 0.02454525 0.0008340536 0.03123274 0.0006954447  
## 8 1 0.0001917910 0.03203949 0.2207111 0.02452975 0.0008381595 0.03130292 0.0006961486  
## 9 1 0.0002104904 0.03201773 0.2216227 0.02451385 0.0008429429 0.03136772 0.0006966327  
## 10 1 0.0002310130 0.03199599 0.2225554 0.02449799 0.0008489496 0.03143650 0.0006981015  
## 11 1 0.0002535364 0.03197424 0.2235123 0.02448262 0.0008556360 0.03148663 0.0006990338  
## 12 1 0.0002782559 0.03195219 0.2245166 0.02446717 0.0008632466 0.03160799 0.0006996160  
## 13 1 0.0003053856 0.03193130 0.2254994 0.02445224 0.0008719750 0.03173987 0.0007002435  
## 14 1 0.0003351603 0.03191052 0.2265202 0.02443821 0.0008821279 0.03194046 0.0007002181  
## 15 1 0.0003678380 0.03189097 0.2275323 0.02442645 0.0008930604 0.03221816 0.0006995757  
## 16 1 0.0004037017 0.03187220 0.2285611 0.02441528 0.0009047167 0.03253443 0.0006984332  
## 17 1 0.0004430621 0.03185624 0.2295034 0.02440851 0.0009174146 0.03292453 0.0006967388  
## 18 1 0.0004862602 0.03184372 0.2303223 0.02440506 0.0009309685 0.03340020 0.0006942755  
## 19 1 0.0005336699 0.03183287 0.2311139 0.02440137 0.0009448719 0.03398628 0.0006937491  
## 20 1 0.0005857021 0.03182464 0.2318345 0.02439922 0.0009564821 0.03459694 0.0006932106  
## 21 1 0.0006428073 0.03182206 0.2323331 0.02440047 0.0009680132 0.03510093 0.0006943088  
## 22 1 0.0007054802 0.03182595 0.2325738 0.02440519 0.0009766080 0.03548750 0.0006965676  
## 23 1 0.0007742637 0.03183325 0.2327119 0.02441536 0.0009822002 0.03591351 0.0006964120  
## 24 1 0.0008497534 0.03184640 0.2326222 0.02443330 0.0009858179 0.03626945 0.0006965488  
## 25 1 0.0009326033 0.03186904 0.2321178 0.02445805 0.0009890151 0.03672520 0.0006978250  
## 26 1 0.0010235310 0.03189824 0.2313681 0.02448761 0.0009921357 0.03727640 0.0006998365  
## 27 1 0.0011233240 0.03193218 0.2304840 0.02452334 0.0009935007 0.03772737 0.0007001487  
## 28 1 0.0012328467 0.03197213 0.2294074 0.02456623 0.0009941365 0.03808183 0.0006994761  
## 29 1 0.0013530478 0.03201893 0.2281107 0.02461400 0.0009942794 0.03844786 0.0006981565  
## 30 1 0.0014849683 0.03207297 0.2265796 0.02466883 0.0009933308 0.03866211 0.0006963136  
## 31 1 0.0016297508 0.03213479 0.2247982 0.02473255 0.0009911140 0.03881318 0.0006942539  
## 32 1 0.0017886495 0.03220083 0.2230051 0.02480054 0.0009864432 0.03891452 0.0006910211  
## 33 1 0.0019630407 0.03226906 0.2213827 0.02487197 0.0009812401 0.03883340 0.0006859912  
## 34 1 0.0021544347 0.03233923 0.2200443 0.02494548 0.0009744218 0.03872023 0.0006787890  
## 35 1 0.0023644894 0.03241375 0.2189542 0.02502305 0.0009693834 0.03851888 0.0006740270  
## 36 1 0.0025950242 0.03250177 0.2175790 0.02511229 0.0009640172 0.03804113 0.0006690945  
## 37 1 0.0028480359 0.03260807 0.2156641 0.02521439 0.0009583158 0.03733951 0.0006642847  
## 38 1 0.0031257158 0.03273564 0.2130434 0.02533174 0.0009532470 0.03644957 0.0006594023  
## 39 1 0.0034304693 0.03288858 0.2094025 0.02546668 0.0009492784 0.03530356 0.0006545778  
## 40 1 0.0037649358 0.03307180 0.2042516 0.02562343 0.0009469241 0.03380385 0.0006518792  
## 41 1 0.0041320124 0.03327698 0.1981400 0.02579431 0.0009471942 0.03231889 0.0006535084  
## 42 1 0.0045348785 0.03348909 0.1925040 0.02597168 0.0009497159 0.03107460 0.0006589196  
## 43 1 0.0049770236 0.03373466 0.1848057 0.02617109 0.0009559234 0.02893659 0.0006652691  
## 44 1 0.0054622772 0.03402773 0.1726919 0.02640489 0.0009660899 0.02591862 0.0006726031  
## 45 1 0.0059948425 0.03437705 0.1531354 0.02667796 0.0009808372 0.02186849 0.0006815502  
## 46 1 0.0065793322 0.03472129 0.1312232 0.02693751 0.0009853654 0.02082959 0.0006804479  
## 47 1 0.0072208090 0.03499622 0.1161551 0.02713454 0.0009895185 0.02278881 0.0006828750  
## 48 1 0.0079248290 0.03522312 0.1061937 0.02729185 0.0009959335 0.02064611 0.0006962213  
## 49 1 0.0086974900 0.03541329 0.1048652 0.02742392 0.0010034175 0.02117276 0.0007125344  
## 50 1 0.0095454846 0.03563082 0.1048652 0.02757849 0.0010144647 0.02117276 0.0007307376  
## 51 1 0.0104761575 0.03589111 0.1048652 0.02776428 0.0010271101 0.02117276 0.0007496433  
## 52 1 0.0114975700 0.03620125 0.1010599 0.02798513 0.0010407483 0.01847740 0.0007703484  
## 53 1 0.0126185688 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 54 1 0.0138488637 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 55 1 0.0151991108 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 56 1 0.0166810054 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 57 1 0.0183073828 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 58 1 0.0200923300 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 59 1 0.0220513074 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 60 1 0.0242012826 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 61 1 0.0265608778 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 62 1 0.0291505306 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 63 1 0.0319926714 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 64 1 0.0351119173 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 65 1 0.0385352859 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 66 1 0.0422924287 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 67 1 0.0464158883 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 68 1 0.0509413801 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 69 1 0.0559081018 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 70 1 0.0613590727 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 71 1 0.0673415066 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 72 1 0.0739072203 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 73 1 0.0811130831 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 74 1 0.0890215085 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 75 1 0.0977009957 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 76 1 0.1072267222 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 77 1 0.1176811952 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 78 1 0.1291549665 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 79 1 0.1417474163 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 80 1 0.1555676144 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 81 1 0.1707352647 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 82 1 0.1873817423 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 83 1 0.2056512308 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 84 1 0.2257019720 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 85 1 0.2477076356 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 86 1 0.2718588243 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 87 1 0.2983647240 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 88 1 0.3274549163 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 89 1 0.3593813664 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 90 1 0.3944206059 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 91 1 0.4328761281 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 92 1 0.4750810162 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 93 1 0.5214008288 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 94 1 0.5722367659 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 95 1 0.6280291442 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 96 1 0.6892612104 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 97 1 0.7564633276 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 98 1 0.8302175681 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 99 1 0.9111627561 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602  
## 100 1 1.0000000000 0.03626118 NaN 0.02802751 0.0010192174 NA 0.0007521602

## Warning: Removed 48 rows containing missing values (geom\_path).

## Warning: Removed 48 rows containing missing values (geom\_point).



## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

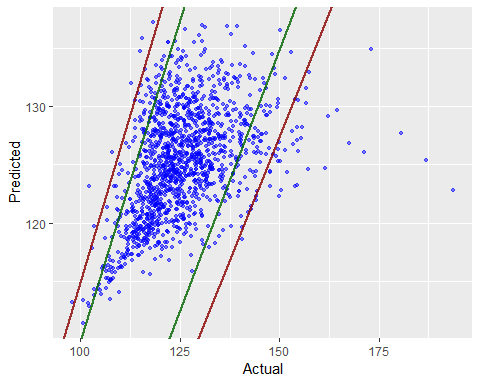


## [1] "Coefficients"  
## model.coef  
## (Intercept) 1.999065e+00  
## x4 -3.542505e-05  
## x7 1.039479e-02  
## x8 1.075399e-04  
## x9 2.804671e-03  
## x10 6.922743e-04  
## x11 8.409804e+04  
## x14 -1.592932e-04  
## x16 6.436982e-04  
## x17 9.064521e-04  
## x19 1.288956e-05  
## x21 5.399885e-05  
## stat3 4.552004e-06  
## stat4 -1.363803e-04  
## stat13 -1.779710e-04  
## stat14 -5.876592e-04  
## stat15 -3.416841e-05  
## stat20 -3.450890e-05  
## stat22 -6.795319e-05  
## stat23 2.026714e-04  
## stat24 -1.565351e-06  
## stat26 -8.693179e-07  
## stat35 -4.361200e-06  
## stat38 9.877132e-05  
## stat41 -9.293586e-05  
## stat48 4.804888e-05  
## stat62 -2.193323e-05  
## stat65 -1.450745e-05  
## stat80 5.025853e-05  
## stat87 -5.545546e-05  
## stat89 -9.694512e-05  
## stat91 -1.522195e-04  
## stat98 3.289814e-03  
## stat103 -3.423596e-05  
## stat106 -1.040502e-05  
## stat110 -3.235877e-03  
## stat124 -2.782429e-06  
## stat130 2.666212e-06  
## stat131 2.224694e-05  
## stat144 4.479178e-05  
## stat146 -3.298818e-04  
## stat149 -6.781153e-05  
## stat156 1.304413e-04  
## stat173 -5.368451e-06  
## stat175 -7.932207e-05  
## stat185 -4.117126e-06  
## stat186 -1.640671e-04  
## stat191 -3.928988e-07  
## stat198 -3.298455e-05  
## stat204 -2.438854e-04  
## x18.sqrt 2.414216e-02

#### Test

if (algo.LASSO.caret == TRUE){  
 test.model(model.LASSO.caret, data.test  
 ,method = 'glmnet',subopt = "LASSO"  
 ,formula = formula, feature.names = feature.names, label.names = label.names  
 ,draw.limits = TRUE, transformation = t)  
}

## [1] "Summary of predicted values: "  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2.047 2.086 2.098 2.097 2.109 2.138   
## [1] "glmnet LASSO Test MSE: 0.000967265746638055"



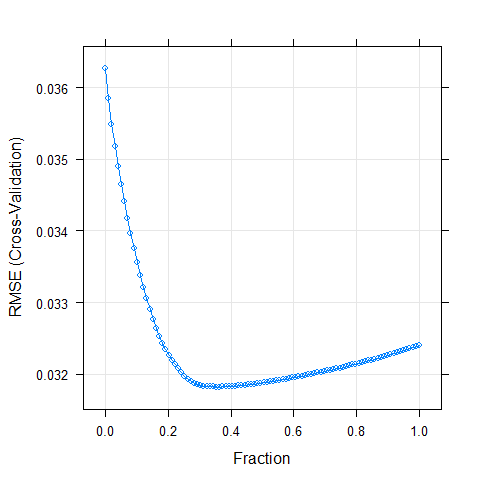
### LARS with CV

#### Train

if (algo.LARS.caret == TRUE){  
 set.seed(1)  
 returned = train.caret.glmselect(formula = formula  
 ,data = data.train  
 ,method = "lars"  
 ,subopt = 'NULL'  
 ,feature.names = feature.names)  
 model.LARS.caret = returned$model  
}

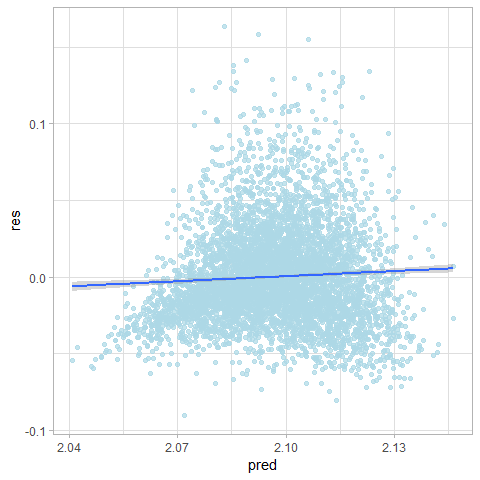
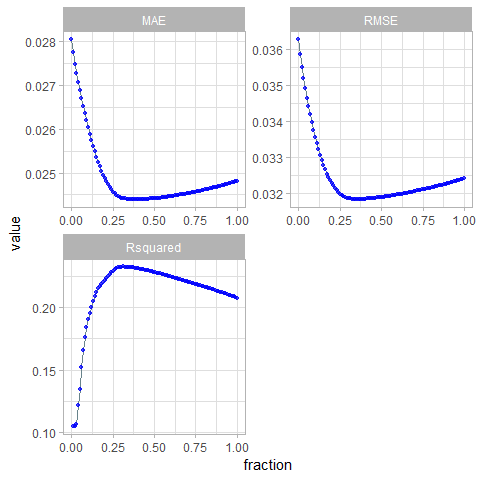
## Warning in nominalTrainWorkflow(x = x, y = y, wts = weights, info = trainInfo, : There were missing values in resampled  
## performance measures.

## Aggregating results  
## Selecting tuning parameters  
## Fitting fraction = 0.354 on full training set  
## Least Angle Regression   
##   
## 5584 samples  
## 240 predictor  
##   
## Pre-processing: centered (240), scaled (240)   
## Resampling: Cross-Validated (10 fold)   
## Summary of sample sizes: 5026, 5026, 5026, 5025, 5025, 5026, ...   
## Resampling results across tuning parameters:  
##   
## fraction RMSE Rsquared MAE   
## 0.00000000 0.03626118 NaN 0.02802751  
## 0.01010101 0.03585259 0.1048652 0.02773603  
## 0.02020202 0.03548965 0.1048652 0.02747644  
## 0.03030303 0.03517699 0.1062883 0.02726013  
## 0.04040404 0.03490052 0.1215464 0.02706700  
## 0.05050505 0.03464547 0.1346996 0.02688335  
## 0.06060606 0.03440508 0.1517319 0.02669960  
## 0.07070707 0.03417549 0.1655019 0.02652110  
## 0.08080808 0.03395777 0.1761724 0.02634976  
## 0.09090909 0.03375217 0.1843703 0.02618544  
## 0.10101010 0.03355889 0.1906457 0.02602971  
## 0.11111111 0.03337864 0.1954984 0.02588042  
## 0.12121212 0.03321212 0.1998822 0.02574078  
## 0.13131313 0.03305485 0.2048277 0.02560939  
## 0.14141414 0.03290754 0.2089402 0.02548342  
## 0.15151515 0.03277048 0.2122690 0.02536318  
## 0.16161616 0.03264381 0.2149542 0.02524927  
## 0.17171717 0.03252764 0.2171117 0.02513962  
## 0.18181818 0.03242739 0.2186830 0.02503923  
## 0.19191919 0.03233948 0.2200780 0.02494789  
## 0.20202020 0.03226410 0.2214303 0.02486840  
## 0.21212121 0.03219681 0.2229974 0.02479787  
## 0.22222222 0.03213327 0.2246889 0.02473256  
## 0.23232323 0.03207348 0.2264138 0.02467022  
## 0.24242424 0.03201995 0.2279162 0.02461450  
## 0.25252525 0.03197154 0.2292727 0.02456587  
## 0.26262626 0.03193146 0.2303343 0.02452346  
## 0.27272727 0.03189883 0.2311808 0.02448903  
## 0.28282828 0.03187339 0.2318668 0.02446219  
## 0.29292929 0.03185455 0.2323258 0.02444255  
## 0.30303030 0.03184237 0.2325097 0.02442847  
## 0.31313131 0.03183459 0.2325609 0.02441710  
## 0.32323232 0.03182923 0.2325343 0.02440962  
## 0.33333333 0.03182502 0.2324946 0.02440469  
## 0.34343434 0.03182225 0.2324179 0.02440151  
## 0.35353535 0.03182139 0.2322743 0.02439946  
## 0.36363636 0.03182199 0.2320822 0.02439830  
## 0.37373737 0.03182340 0.2318684 0.02439825  
## 0.38383838 0.03182526 0.2316502 0.02439781  
## 0.39393939 0.03182828 0.2313895 0.02439840  
## 0.40404040 0.03183208 0.2310963 0.02439954  
## 0.41414141 0.03183576 0.2308183 0.02440069  
## 0.42424242 0.03183957 0.2305437 0.02440204  
## 0.43434343 0.03184362 0.2302645 0.02440291  
## 0.44444444 0.03184781 0.2299855 0.02440376  
## 0.45454545 0.03185227 0.2297010 0.02440481  
## 0.46464646 0.03185718 0.2293992 0.02440631  
## 0.47474747 0.03186278 0.2290695 0.02440889  
## 0.48484848 0.03186895 0.2287178 0.02441221  
## 0.49494949 0.03187542 0.2283569 0.02441614  
## 0.50505051 0.03188203 0.2279965 0.02442009  
## 0.51515152 0.03188829 0.2276590 0.02442361  
## 0.52525253 0.03189470 0.2273207 0.02442732  
## 0.53535354 0.03190126 0.2269816 0.02443110  
## 0.54545455 0.03190837 0.2266220 0.02443550  
## 0.55555556 0.03191608 0.2262398 0.02444069  
## 0.56565657 0.03192383 0.2258614 0.02444604  
## 0.57575758 0.03193155 0.2254906 0.02445143  
## 0.58585859 0.03193959 0.2251099 0.02445719  
## 0.59595960 0.03194796 0.2247194 0.02446334  
## 0.60606061 0.03195633 0.2243327 0.02446946  
## 0.61616162 0.03196469 0.2239516 0.02447523  
## 0.62626263 0.03197347 0.2235580 0.02448139  
## 0.63636364 0.03198244 0.2231605 0.02448780  
## 0.64646465 0.03199139 0.2227682 0.02449415  
## 0.65656566 0.03200052 0.2223714 0.02450067  
## 0.66666667 0.03200976 0.2219742 0.02450732  
## 0.67676768 0.03201908 0.2215781 0.02451406  
## 0.68686869 0.03202848 0.2211832 0.02452096  
## 0.69696970 0.03203803 0.2207865 0.02452791  
## 0.70707071 0.03204756 0.2203944 0.02453516  
## 0.71717172 0.03205725 0.2200008 0.02454268  
## 0.72727273 0.03206699 0.2196088 0.02455029  
## 0.73737374 0.03207688 0.2192137 0.02455788  
## 0.74747475 0.03208711 0.2188075 0.02456579  
## 0.75757576 0.03209772 0.2183890 0.02457396  
## 0.76767677 0.03210838 0.2179729 0.02458222  
## 0.77777778 0.03211891 0.2175658 0.02459047  
## 0.78787879 0.03212943 0.2171641 0.02459867  
## 0.79797980 0.03214018 0.2167563 0.02460689  
## 0.80808081 0.03215121 0.2163402 0.02461530  
## 0.81818182 0.03216252 0.2159161 0.02462392  
## 0.82828283 0.03217417 0.2154818 0.02463288  
## 0.83838384 0.03218603 0.2150428 0.02464213  
## 0.84848485 0.03219811 0.2145978 0.02465163  
## 0.85858586 0.03221029 0.2141524 0.02466131  
## 0.86868687 0.03222254 0.2137073 0.02467113  
## 0.87878788 0.03223493 0.2132593 0.02468105  
## 0.88888889 0.03224754 0.2128068 0.02469116  
## 0.89898990 0.03226030 0.2123508 0.02470123  
## 0.90909091 0.03227324 0.2118924 0.02471161  
## 0.91919192 0.03228640 0.2114292 0.02472221  
## 0.92929293 0.03229978 0.2109612 0.02473292  
## 0.93939394 0.03231346 0.2104855 0.02474371  
## 0.94949495 0.03232713 0.2100153 0.02475440  
## 0.95959596 0.03234083 0.2095475 0.02476516  
## 0.96969697 0.03235470 0.2090764 0.02477603  
## 0.97979798 0.03236878 0.2086010 0.02478690  
## 0.98989899 0.03238309 0.2081210 0.02479770  
## 1.00000000 0.03239750 0.2076409 0.02480845  
##   
## RMSE was used to select the optimal model using the smallest value.  
## The final value used for the model was fraction = 0.3535354.

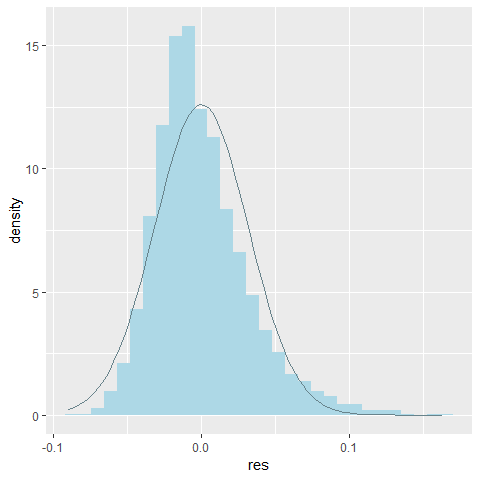


## fraction  
## 36 0.3535354

## Warning: Removed 1 rows containing missing values (geom\_point).



## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

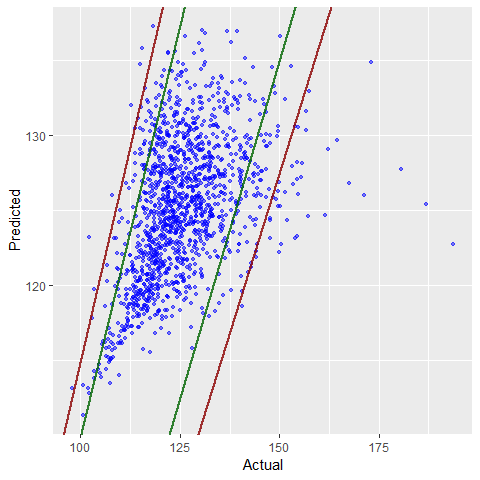


## [1] "Coefficients"  
## x4 x7 x8 x9 x10 x11 x14 x16   
## -1.685695e-03 7.006074e-03 3.178270e-04 3.628137e-03 9.688277e-04 4.966752e-04 -2.182458e-04 1.296057e-03   
## x17 x19 x21 stat3 stat4 stat13 stat14 stat15   
## 1.203067e-03 4.149125e-05 5.566560e-04 1.352520e-06 -2.418515e-04 -3.175865e-04 -1.034850e-03 -6.719675e-05   
## stat20 stat22 stat23 stat38 stat41 stat48 stat62 stat65   
## -6.727737e-05 -1.263498e-04 3.589162e-04 1.797035e-04 -1.688353e-04 9.030713e-05 -4.585482e-05 -3.252831e-05   
## stat80 stat87 stat89 stat91 stat98 stat103 stat106 stat110   
## 9.359698e-05 -1.035002e-04 -1.759916e-04 -2.712225e-04 5.765274e-03 -6.645710e-05 -2.519513e-05 -5.658522e-03   
## stat131 stat144 stat146 stat149 stat156 stat173 stat175 stat185   
## 4.557172e-05 8.405849e-05 -5.772197e-04 -1.242569e-04 2.308789e-04 -9.069132e-06 -1.453625e-04 -7.074058e-06   
## stat186 stat198 stat204 x18.sqrt   
## -2.892949e-04 -6.403769e-05 -4.332137e-04 1.095750e-02

#### Test

if (algo.LARS.caret == TRUE){  
 test.model(model.LARS.caret, data.test  
 ,method = 'lars',subopt = NULL  
 ,formula = formula, feature.names = feature.names, label.names = label.names  
 ,draw.limits = TRUE, transformation = t)  
}

## [1] "Summary of predicted values: "  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2.047 2.086 2.098 2.097 2.109 2.138   
## [1] "lars Test MSE: 0.000967245442650916"



# Session Info

sessionInfo()

## R version 3.5.2 (2018-12-20)  
## Platform: x86\_64-w64-mingw32/x64 (64-bit)  
## Running under: Windows 10 x64 (build 17763)  
##   
## Matrix products: default  
##   
## locale:  
## [1] LC\_COLLATE=English\_United States.1252 LC\_CTYPE=English\_United States.1252 LC\_MONETARY=English\_United States.1252  
## [4] LC\_NUMERIC=C LC\_TIME=English\_United States.1252   
##   
## attached base packages:  
## [1] parallel stats graphics grDevices utils datasets methods base   
##   
## other attached packages:  
## [1] knitr\_1.21 htmltools\_0.3.6 reshape2\_1.4.3 lars\_1.2   
## [5] doParallel\_1.0.14 iterators\_1.0.10 caret\_6.0-81 leaps\_3.0   
## [9] ggforce\_0.1.3 rlist\_0.4.6.1 car\_3.0-2 carData\_3.0-2   
## [13] bestNormalize\_1.3.0 scales\_1.0.0 onewaytests\_2.0 caTools\_1.17.1.1   
## [17] mosaic\_1.5.0 mosaicData\_0.17.0 ggformula\_0.9.1 ggstance\_0.3.1   
## [21] lattice\_0.20-38 DT\_0.5 ggiraphExtra\_0.2.9 ggiraph\_0.6.0   
## [25] investr\_1.4.0 glmnet\_2.0-16 foreach\_1.4.4 Matrix\_1.2-15   
## [29] MASS\_7.3-51.1 PerformanceAnalytics\_1.5.2 xts\_0.11-2 zoo\_1.8-4   
## [33] forcats\_0.3.0 stringr\_1.4.0 dplyr\_0.8.0.1 purrr\_0.3.0   
## [37] readr\_1.3.1 tidyr\_0.8.2 tibble\_2.0.1 ggplot2\_3.1.0   
## [41] tidyverse\_1.2.1 usdm\_1.1-18 raster\_2.8-19 sp\_1.3-1   
## [45] pacman\_0.5.0   
##   
## loaded via a namespace (and not attached):  
## [1] readxl\_1.3.0 backports\_1.1.3 plyr\_1.8.4 lazyeval\_0.2.1 splines\_3.5.2 mycor\_0.1.1   
## [7] crosstalk\_1.0.0 leaflet\_2.0.2 digest\_0.6.18 magrittr\_1.5 mosaicCore\_0.6.0 openxlsx\_4.1.0   
## [13] recipes\_0.1.4 modelr\_0.1.3 gower\_0.1.2 colorspace\_1.4-0 rvest\_0.3.2 ggrepel\_0.8.0   
## [19] haven\_2.0.0 xfun\_0.4 crayon\_1.3.4 jsonlite\_1.6 survival\_2.43-3 glue\_1.3.0   
## [25] registry\_0.5 gtable\_0.2.0 ppcor\_1.1 ipred\_0.9-8 sjmisc\_2.7.7 abind\_1.4-5   
## [31] rngtools\_1.3.1 bibtex\_0.4.2 Rcpp\_1.0.0 xtable\_1.8-3 units\_0.6-2 foreign\_0.8-71   
## [37] stats4\_3.5.2 lava\_1.6.5 prodlim\_2018.04.18 prediction\_0.3.6.2 htmlwidgets\_1.3 httr\_1.4.0   
## [43] RColorBrewer\_1.1-2 pkgconfig\_2.0.2 farver\_1.1.0 nnet\_7.3-12 labeling\_0.3 tidyselect\_0.2.5   
## [49] rlang\_0.3.1 later\_0.8.0 munsell\_0.5.0 cellranger\_1.1.0 tools\_3.5.2 cli\_1.0.1   
## [55] generics\_0.0.2 moments\_0.14 sjlabelled\_1.0.16 broom\_0.5.1 evaluate\_0.13 ggdendro\_0.1-20   
## [61] yaml\_2.2.0 ModelMetrics\_1.2.2 zip\_1.0.0 nlme\_3.1-137 doRNG\_1.7.1 mime\_0.6   
## [67] xml2\_1.2.0 compiler\_3.5.2 rstudioapi\_0.9.0 curl\_3.3 tweenr\_1.0.1 stringi\_1.3.1   
## [73] highr\_0.7 gdtools\_0.1.7 stringdist\_0.9.5.1 pillar\_1.3.1 data.table\_1.12.0 bitops\_1.0-6   
## [79] httpuv\_1.4.5.1 R6\_2.4.0 promises\_1.0.1 gridExtra\_2.3 rio\_0.5.16 codetools\_0.2-15   
## [85] assertthat\_0.2.0 pkgmaker\_0.27 withr\_2.1.2 nortest\_1.0-4 mgcv\_1.8-26 hms\_0.4.2   
## [91] quadprog\_1.5-5 grid\_3.5.2 rpart\_4.1-13 timeDate\_3043.102 class\_7.3-14 rmarkdown\_1.11   
## [97] snakecase\_0.9.2 shiny\_1.2.0 lubridate\_1.7.4