Random Forest Results

Rsqu(new\_train\_data[,c(2:28)]$Count,predict(rf\_default,new\_train\_data[,c(2:28)]))

rmse(new\_train\_data[,c(2:28)]$Count,predict(rf\_default,new\_train\_data[,c(2:28)]))

MAE(new\_train\_data[,c(2:28)]$Count,predict(rf\_default,new\_train\_data[,c(2:28)]))

cv(new\_train\_data[,c(2:28)]$Count,predict(rf\_default,new\_train\_data[,c(2:28)]))\*100

Rsqu(new\_test\_data[,c(2:28)]$Count,predict(rf\_default,new\_test\_data[,c(2:28)]))

rmse(new\_test\_data[,c(2:28)]$Count,predict(rf\_default,new\_test\_data[,c(2:28)]))

MAE(new\_test\_data[,c(2:28)]$Count,predict(rf\_default,new\_test\_data[,c(2:28)]))

cv(new\_test\_data[,c(2:28)]$Count,predict(rf\_default,new\_test\_data[,c(2:28)]))\*100

plot(varImp(rf\_default))

library(gridBase)

library(gridExtra)

library(lattice)

x <-1:500

y <- sqrt(rf\_default$finalModel$mse)

n\_trees <- xyplot(y ~ x,

ylab="RMSE", xlab="Number of Trees")

dev.off()

rf\_g <-plot(1:500,sqrt(rf\_default$finalModel$mse),type='l',col='blue',axes=TRUE,xlab="Number of Trees",ylab="RMSE")

tress\_g <- plot(rf\_default)

class(tress\_g)

panel\_plot <- grid.arrange(n\_trees, tress\_g,ncol=2)

ggsave(file="rf\_model.png", panel\_plot)

cubist

Rsqu(new\_train\_data[,c(2:28)]$Count,predict(cubist,new\_train\_data[,c(2:28)]))

rmse(new\_train\_data[,c(2:28)]$Count,predict(cubist,new\_train\_data[,c(2:28)]))

MAE(new\_train\_data[,c(2:28)]$Count,predict(cubist,new\_train\_data[,c(2:28)]))

cv(new\_train\_data[,c(2:28)]$Count,predict(cubist,new\_train\_data[,c(2:28)]))\*100

Rsqu(new\_test\_data[,c(2:28)]$Count,predict(cubist,new\_test\_data[,c(2:28)]))

rmse(new\_test\_data[,c(2:28)]$Count,predict(cubist,new\_test\_data[,c(2:28)]))

MAE(new\_test\_data[,c(2:28)]$Count,predict(cubist,new\_test\_data[,c(2:28)]))

cv(new\_test\_data[,c(2:28)]$Count,predict(cubist,new\_test\_data[,c(2:28)]))\*100

plot(varImp(cubist))

plot(cubist)

cubist$bestTune

Classification and Regression Trees

Rsqu(new\_train\_data[,c(2:28)]$Count,predict(cart,new\_train\_data[,c(2:28)]))

rmse(new\_train\_data[,c(2:28)]$Count,predict(cart,new\_train\_data[,c(2:28)]))

MAE(new\_train\_data[,c(2:28)]$Count,predict(cart,new\_train\_data[,c(2:28)]))

cv(new\_train\_data[,c(2:28)]$Count,predict(cart,new\_train\_data[,c(2:28)]))\*100

Rsqu(new\_test\_data[,c(2:28)]$Count,predict(cart,new\_test\_data[,c(2:28)]))

rmse(new\_test\_data[,c(2:28)]$Count,predict(cart,new\_test\_data[,c(2:28)]))

MAE(new\_test\_data[,c(2:28)]$Count,predict(cart,new\_test\_data[,c(2:28)]))

cv(new\_test\_data[,c(2:28)]$Count,predict(cart,new\_test\_data[,c(2:28)]))\*100

plot(varImp(cart))

plot(cart)

cart$bestTune

K Nearest Neighbour

Rsqu(new\_train\_data[,c(2:28)]$Count,predict(knn,new\_train\_data[,c(2:28)]))

rmse(new\_train\_data[,c(2:28)]$Count,predict(knn,new\_train\_data[,c(2:28)]))

MAE(new\_train\_data[,c(2:28)]$Count,predict(knn,new\_train\_data[,c(2:28)]))

cv(new\_train\_data[,c(2:28)]$Count,predict(knn,new\_train\_data[,c(2:28)]))\*100

Rsqu(new\_test\_data[,c(2:28)]$Count,predict(knn,new\_test\_data[,c(2:28)]))

rmse(new\_test\_data[,c(2:28)]$Count,predict(knn,new\_test\_data[,c(2:28)]))

MAE(new\_test\_data[,c(2:28)]$Count,predict(knn,new\_test\_data[,c(2:28)]))

cv(new\_test\_data[,c(2:28)]$Count,predict(knn,new\_test\_data[,c(2:28)]))\*100

plot(varImp(knn))

plot(knn)

knn$bestTune

Conditional Inference Tree

Rsqu(new\_train\_data[,c(2:28)]$Count,predict(conditional,new\_train\_data[,c(2:28)]))

rmse(new\_train\_data[,c(2:28)]$Count,predict(conditional,new\_train\_data[,c(2:28)]))

MAE(new\_train\_data[,c(2:28)]$Count,predict(conditional,new\_train\_data[,c(2:28)]))

cv(new\_train\_data[,c(2:28)]$Count,predict(conditional,new\_train\_data[,c(2:28)]))\*100

Rsqu(new\_test\_data[,c(2:28)]$Count,predict(conditional,new\_test\_data[,c(2:28)]))

rmse(new\_test\_data[,c(2:28)]$Count,predict(conditional,new\_test\_data[,c(2:28)]))

MAE(new\_test\_data[,c(2:28)]$Count,predict(conditional,new\_test\_data[,c(2:28)]))

cv(new\_test\_data[,c(2:28)]$Count,predict(conditional,new\_test\_data[,c(2:28)]))\*100

plot(varImp(conditional))

plot(conditional)

conditional$bestTune

RF\_imp <- plot(varImp(rf\_default),main="RF Variable Importance")

CUBIST\_imp <- plot(varImp(cubist),main="CUBIST Variable Importance")

CART\_imp <- plot(varImp(cart),main="CART Variable Importance")

KNN\_imp <- plot(varImp(knn),main="KNN Variable Importance")

Conditional\_imp <- plot(varImp(conditional),main="CIT Variable Importance")

panelVIMP\_plot\_models <- grid.arrange(RF\_imp, CUBIST\_imp,KNN\_imp, KNN\_imp,Conditional\_imp,ncol=5)

ggsave(file="panel\_plot\_Cubist.png", panelVIMP\_plot\_models)