## Code and Output for the Random Forest-Based Prediction Model

```
> patent <- read.csv("patents.csv")
> install.packages("randomForest")
> library(randomForest)
> ind <- sample(2, nrow(patent), replace = TRUE, prob = c(0.5, 0.5))
> patent.train <- patent[ind == 1, ]
> patent.test <- patent[ind == 2, ]
> patent.train.rf <- randomForest(as.factor(status) ~ examinerInterviews + foreign + RCE, data =
patent.train, ntree = 5000, importance = TRUE, na.action = na.omit)
> patent.train.rf
```

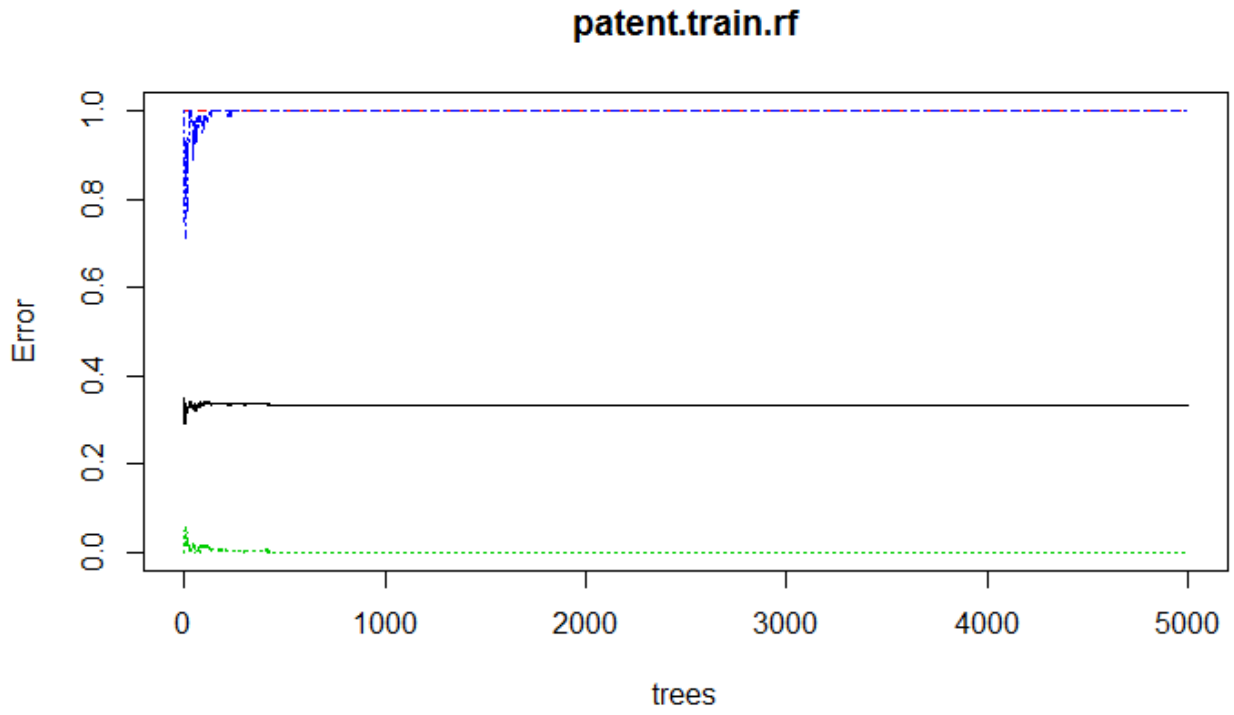
```
               Type of random forest: classification
                     Number of trees: 5000
No. of variables tried at each split: 1

        OOB estimate of  error rate: 33.25%
Confusion matrix:
                  Abandoned Allowed Unknown or Pending class.error
Abandoned                 0      47                   0           1
Allowed                   0     255                   0           0
Unknown or Pending        0      80                   0           1
> |
```

```
> table(predict(patent.train.rf), patent.train$status)
```

```
                    Abandoned Allowed Unknown or Pending
  Abandoned                 0       0                   0
  Allowed                  47     255                  80
  Unknown or Pending        0       0                   0
```

> plot(patent.train.rf)

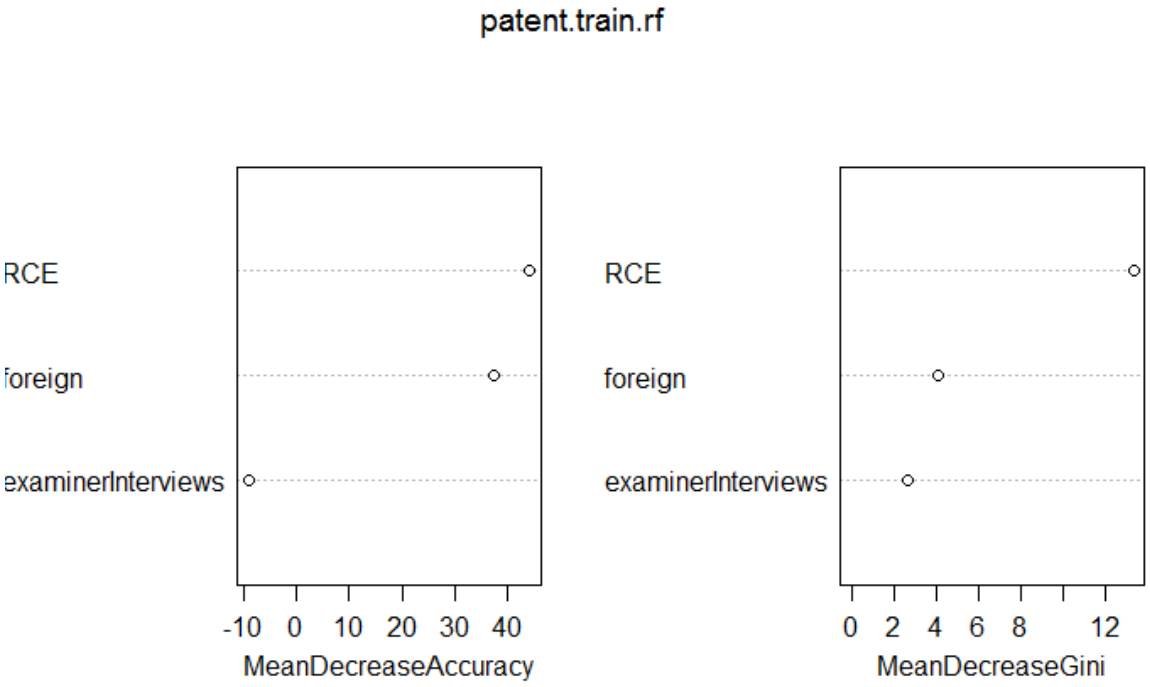



> importance(patent.train.rf)

| | Abandoned | Allowed | Unknown or Pending | MeanDecreaseAccuracy | MeanDecreaseGini |
|---|---|---|---|---|---|
| examinerInterviews | -2.1926227 | -9.139922 | 1.767663 | -9.054175 | 2.636602 |
| foreign | 0.5211465 | 35.896712 | 26.700647 | 37.310848 | 4.053628 |
| RCE | 18.3788555 | 24.266510 | 47.944250 | 44.072492 | 13.343200 |

> |

> varImpPlot(patent.train.rf)




> patent.prediction <- predict(patent.train.rf, newdata = patent.test)

> table(patent.prediction, patent.test$status)

| patent.prediction | Abandoned | Allowed | Unknown or Pending |
|---|---|---|---|
| Abandoned | 0 | 0 | 0 |
| Allowed | 59 | 258 | 91 |
| Unknown or Pending | 0 | 0 | 0 |

```
> plot(margin(patent.train.rf, patent.test$status))
```