Classification & Comparing models

2025-02-27

```
view(data)
```

```
summary(data)
```

```
##   tot_balance     avg_bal_cards     credit_age    credit_age_good_account
##  Min.   :     0   Min.   :    0   Min.   :  0.0   Min.   :  0.0
##  1st Qu.: 92213   1st Qu.:10151   1st Qu.:231.0   1st Qu.:120.0
##  Median :107711   Median :12239   Median :280.0   Median :146.0
##  Mean   :107439   Mean   :12231   Mean   :280.7   Mean   :146.1
##  3rd Qu.:122751   3rd Qu.:14286   3rd Qu.:330.0   3rd Qu.:172.0
##  Max.   :200000   Max.   :25000   Max.   :560.0   Max.   :300.0
##
##  credit_card_age num_acc_30d_past_due_12_months num_acc_30d_past_due_6_months
##  Min.   :  0.0   Min.   :0.0000                 Min.   :0.0000
##  1st Qu.:242.0   1st Qu.:0.0000                 1st Qu.:0.0000
##  Median :285.0   Median :0.0000                 Median :0.0000
##  Mean   :285.1   Mean   :0.1565                 Mean   :0.0297
##  3rd Qu.:330.0   3rd Qu.:0.0000                 3rd Qu.:0.0000
##  Max.   :550.0   Max.   :5.0000                 Max.   :2.0000
##
##  num_mortgage_currently_past_due tot_amount_currently_past_due num_inq_12_month
##  Min.   :0.00                    Min.   :    0.0               Min.   : 0.000
##  1st Qu.:0.00                    1st Qu.:    0.0               1st Qu.: 0.000
##  Median :0.00                    Median :    0.0               Median : 0.000
##  Mean   :0.03                    Mean   :  352.5               Mean   : 0.616
##  3rd Qu.:0.00                    3rd Qu.:    0.0               3rd Qu.: 1.000
##  Max.   :1.00                    Max.   :35000.0               Max.   :10.000
##
##  num_card_inq_24_month num_card_12_month num_auto_ 36_month uti_open_card
##  Min.   : 0.000        Min.   :0.000     Min.   :0.0000     Min.   :0.0000
##  1st Qu.: 0.000        1st Qu.:0.000     1st Qu.:0.0000     1st Qu.:0.4039
##  Median : 0.000        Median :0.000     Median :0.0000     Median :0.4904
##  Mean   : 1.053        Mean   :0.273     Mean   :0.1641     Mean   :0.4909
##  3rd Qu.: 1.000        3rd Qu.:1.000     3rd Qu.:0.0000     3rd Qu.:0.5783
##  Max.   :18.000        Max.   :3.000     Max.   :2.0000     Max.   :1.0000
##
##  pct_over_50_uti  uti_max_credit_line pct_card_over_50_uti    ind_XYZ
##  Min.   :0.0000   Min.   :0.0000      Min.   :0.0000       Min.   :0.00
##  1st Qu.:0.4011   1st Qu.:0.3778      1st Qu.:0.4642       1st Qu.:0.00
##  Median :0.4855   Median :0.4648      Median :0.5518       Median :0.00
##  Mean   :0.4842   Mean   :0.4650      Mean   :0.5510       Mean   :0.25
##  3rd Qu.:0.5680   3rd Qu.:0.5536      3rd Qu.:0.6383       3rd Qu.:0.25
##  Max.   :1.0000   Max.   :1.0000      Max.   :1.0000       Max.   :1.00
##                                       NA's   :1958
```

```
##    rep_income     rep_education         Def_ind
##  Min.   : 20000   Length:20000       Min.   :0.0
##  1st Qu.:143504   Class :character   1st Qu.:0.0
##  Median :166463   Mode  :character   Median :0.0
##  Mean   :166374                      Mean   :0.1
##  3rd Qu.:188904                      3rd Qu.:0.0
##  Max.   :300000                      Max.   :1.0
##  NA's   :1559
```

```
nrow(data)
```

```
## [1] 20000
```

```
ncol(data)
```

```
## [1] 21
```

#comments

Our summary statistics show us 20000 rows and 20 columns, we have 20 variables in which 20 are predictors and 1 is the response variable, also known as Def_ind in which it is an indicator of default 1, an account was opened and approved with the bank XYZ and 0 for not default.

```
colSums(is.na(data))
```

```
##                      tot_balance                      avg_bal_cards
##                                0                                 0
##                       credit_age          credit_age_good_account
##                                0                                 0
##                 credit_card_age   num_acc_30d_past_due_12_months
##                                0                                 0
##    num_acc_30d_past_due_6_months num_mortgage_currently_past_due
##                                0                                 0
##   tot_amount_currently_past_due                 num_inq_12_month
##                                0                                 0
##           num_card_inq_24_month                num_card_12_month
##                                0                                 0
##              num_auto_ 36_month                    uti_open_card
##                                0                                 0
##                 pct_over_50_uti              uti_max_credit_line
##                                0                                 0
##            pct_card_over_50_uti                          ind_XYZ
##                            1958                                 0
##                      rep_income                    rep_education
##                            1559                                 1
##                         Def_ind
##                                0
```

#comments

I am also able to notice there are variables with NAs, one of them is our pct_card_over_50_uti, percentage of open credit cards with 50% utilization and our rep_income also known as annual income and our rep education. We also notice that one of our rows is mispelled with an extra space "num_auto_ 36_month".

```r
names(data)[names(data) == "num_auto_ 36_month"] <- "num_auto_36_month"
```

#comments

We go on about changing the name of the mispelled variable

```r
data_clean <- data %>% drop_na()
```

#comments we also use the drop_na to remove our missing values

```r
colSums(is.na(data_clean))
```

```
##                      tot_balance                      avg_bal_cards
##                                0                                  0
##                       credit_age          credit_age_good_account
##                                0                                  0
##                  credit_card_age   num_acc_30d_past_due_12_months
##                                0                                  0
##    num_acc_30d_past_due_6_months  num_mortgage_currently_past_due
##                                0                                  0
##    tot_amount_currently_past_due                 num_inq_12_month
##                                0                                  0
##             num_card_inq_24_month                num_card_12_month
##                                0                                  0
##                num_auto_36_month                    uti_open_card
##                                0                                  0
##                  pct_over_50_uti              uti_max_credit_line
##                                0                                  0
##              pct_card_over_50_uti                          ind_XYZ
##                                0                                  0
##                       rep_income                    rep_education
##                                0                                  0
##                          Def_ind
##                                0
```

#comments we make sure our missing values have been dropped

```r
sum(duplicated(data_clean))
```

```
## [1] 0
```

#comments we check for duplicares in which we find none

```r
sapply(data_clean, class)
```

```
##                      tot_balance                      avg_bal_cards
##                        "numeric"                          "numeric"
##                       credit_age          credit_age_good_account
##                        "numeric"                          "numeric"
##                  credit_card_age   num_acc_30d_past_due_12_months
##                        "numeric"                          "numeric"
```

```
##   num_acc_30d_past_due_6_months num_mortgage_currently_past_due
##                       "numeric"                       "numeric"
##   tot_amount_currently_past_due                num_inq_12_month
##                       "numeric"                       "numeric"
##           num_card_inq_24_month               num_card_12_month
##                       "numeric"                       "numeric"
##              num_auto_36_month                   uti_open_card
##                       "numeric"                       "numeric"
##                pct_over_50_uti             uti_max_credit_line
##                       "numeric"                       "numeric"
##            pct_card_over_50_uti                         ind_XYZ
##                       "numeric"                       "numeric"
##                      rep_income                   rep_education
##                       "numeric"                     "character"
##                         Def_ind
##                       "numeric"
```

#comments we check the category of each variables we have and I notice that rep_education is described as character and it needs to be changed as a categorical value to define the classes of rep education so we have to recognize it as a factor.

```
data_clean$rep_education <- as.factor(data_clean$rep_education)
```

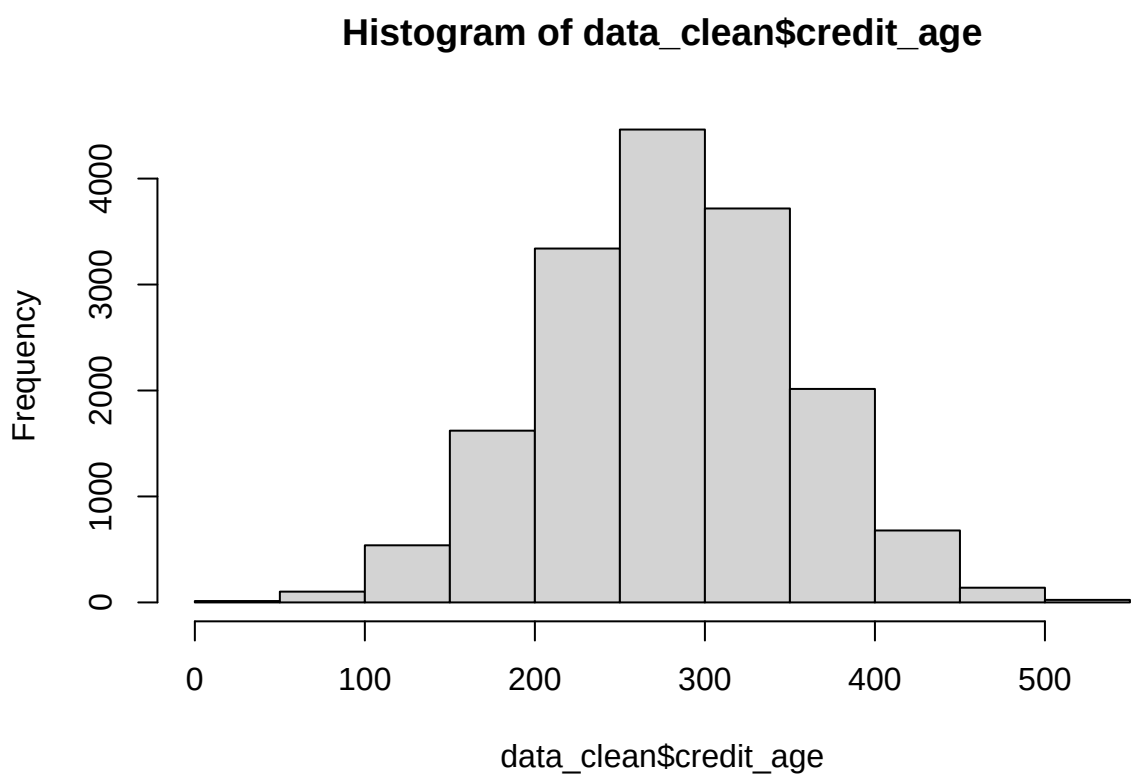
```
summary(data_clean)
```

```
##   tot_balance     avg_bal_cards     credit_age    credit_age_good_account
##  Min.   :     0   Min.   :    0   Min.   :  0.0   Min.   :  0.0
##  1st Qu.: 92142   1st Qu.:10135   1st Qu.:231.0   1st Qu.:120.0
##  Median :107740   Median :12237   Median :281.0   Median :146.0
##  Mean   :107503   Mean   :12226   Mean   :280.9   Mean   :146.2
##  3rd Qu.:122932   3rd Qu.:14297   3rd Qu.:330.0   3rd Qu.:172.0
##  Max.   :200000   Max.   :25000   Max.   :550.0   Max.   :300.0
##  credit_card_age num_acc_30d_past_due_12_months num_acc_30d_past_due_6_months
##  Min.   :  0.0   Min.   :0.0000                 Min.   :0.00000
##  1st Qu.:242.0   1st Qu.:0.0000                 1st Qu.:0.00000
##  Median :285.0   Median :0.0000                 Median :0.00000
##  Mean   :285.4   Mean   :0.1579                 Mean   :0.02936
##  3rd Qu.:330.0   3rd Qu.:0.0000                 3rd Qu.:0.00000
##  Max.   :550.0   Max.   :5.0000                 Max.   :2.00000
##  num_mortgage_currently_past_due tot_amount_currently_past_due
##  Min.   :0.0000                  Min.   :    0.0
##  1st Qu.:0.0000                  1st Qu.:    0.0
##  Median :0.0000                  Median :    0.0
##  Mean   :0.0299                  Mean   :  354.2
##  3rd Qu.:0.0000                  3rd Qu.:    0.0
##  Max.   :1.0000                  Max.   :35000.0
##  num_inq_12_month  num_card_inq_24_month num_card_12_month num_auto_36_month
##  Min.   : 0.0000   Min.   : 0.000        Min.   :0.0000    Min.   :0.000
##  1st Qu.: 0.0000   1st Qu.: 0.000        1st Qu.:0.0000    1st Qu.:0.000
##  Median : 0.0000   Median : 0.000        Median :0.0000    Median :0.000
##  Mean   : 0.6133   Mean   : 1.044        Mean   :0.2723    Mean   :0.165
##  3rd Qu.: 1.0000   3rd Qu.: 1.000        3rd Qu.:1.0000    3rd Qu.:0.000
##  Max.   :10.0000   Max.   :18.000        Max.   :3.0000    Max.   :2.000
```

```
##  uti_open_card    pct_over_50_uti  uti_max_credit_line pct_card_over_50_uti
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000      Min.   :0.0000
##  1st Qu.:0.4048   1st Qu.:0.4011   1st Qu.:0.3778      1st Qu.:0.4643
##  Median :0.4909   Median :0.4855   Median :0.4649      Median :0.5518
##  Mean   :0.4914   Mean   :0.4842   Mean   :0.4653      Mean   :0.5511
##  3rd Qu.:0.5783   3rd Qu.:0.5679   3rd Qu.:0.5541      3rd Qu.:0.6384
##  Max.   :1.0000   Max.   :0.9294   Max.   :1.0000      Max.   :1.0000
##     ind_XYZ         rep_income        rep_education      Def_ind
##  Min.   :0.0000   Min.   : 20000   college    :10104   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:143751   graduate   : 2026   1st Qu.:0.0000
##  Median :0.0000   Median :166630   high_school: 4403   Median :0.0000
##  Mean   :0.2487   Mean   :166504   other      :  120   Mean   :0.1019
##  3rd Qu.:0.0000   3rd Qu.:189020                       3rd Qu.:0.0000
##  Max.   :1.0000   Max.   :300000                       Max.   :1.0000
```

#comments Our data has now recognized it as a factor also known as college, graduates, high school(below), and others.

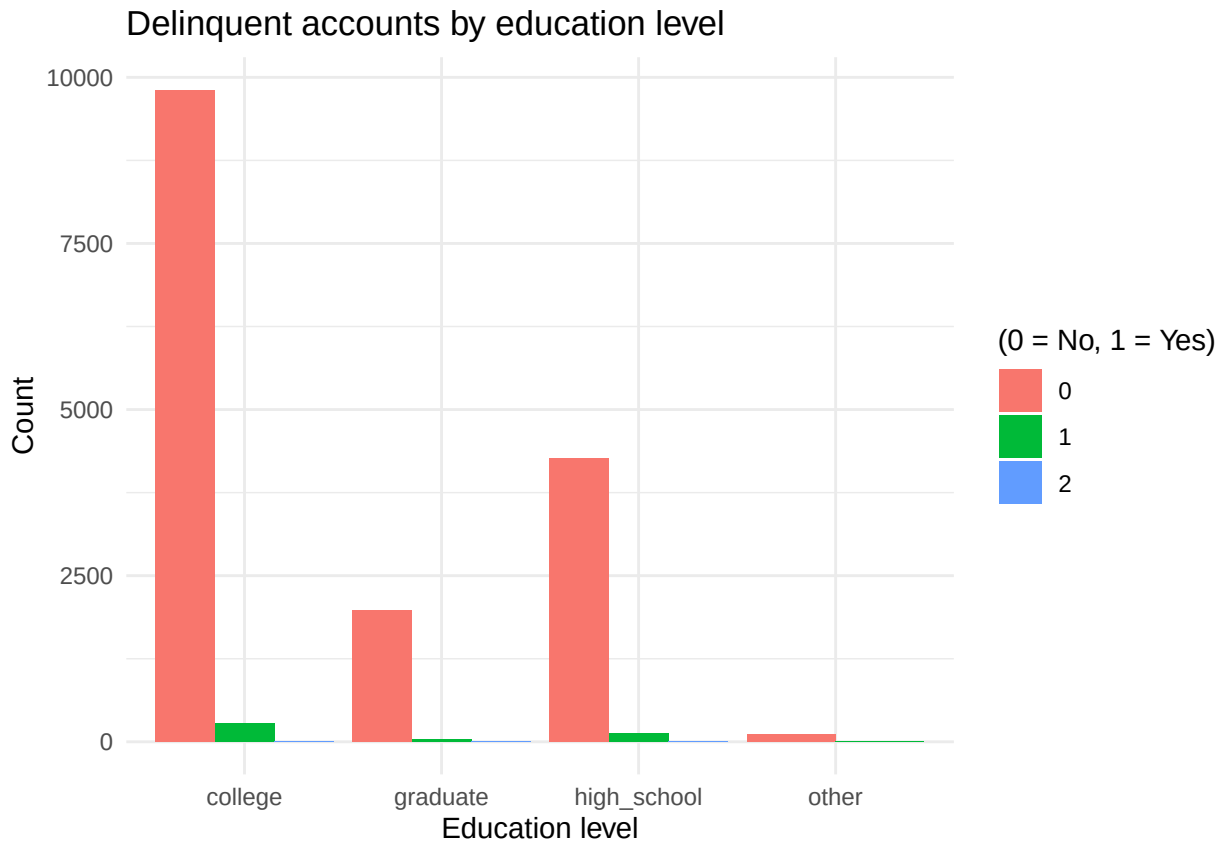
```
hist(data_clean$credit_age)
```



#comments I create a histogram to check the balance of credit age among people it seems like most people have been identified to have their credit card average for an around of 200-300 months. which means an average of 20 years.

```
library(ggplot2)
ggplot(data_clean, aes(x = rep_education, fill = as.factor(num_acc_30d_past_due_6_months))) +
```

```
  geom_bar(position = "dodge") +
  labs(title = "Delinquent accounts by education level",
       x = "Education level",
       y = "Count",
       fill = "(0 = No, 1 = Yes)") +
  theme_minimal()
```



#comments I also created another plot to see which education level had the most delinquent accounts in the last 30 days in the past 62 months. After conducting this graph I notice most people who marked college education and high school were on the top 2 columns with them as yes (green) and graduate holds the least amount of yes, though these are very small columns for all 4 rows of education we can still see a difference.

```
table(data_clean$rep_education)
```

```
##
##     college    graduate high_school       other
##       10104        2026        4403         120
```

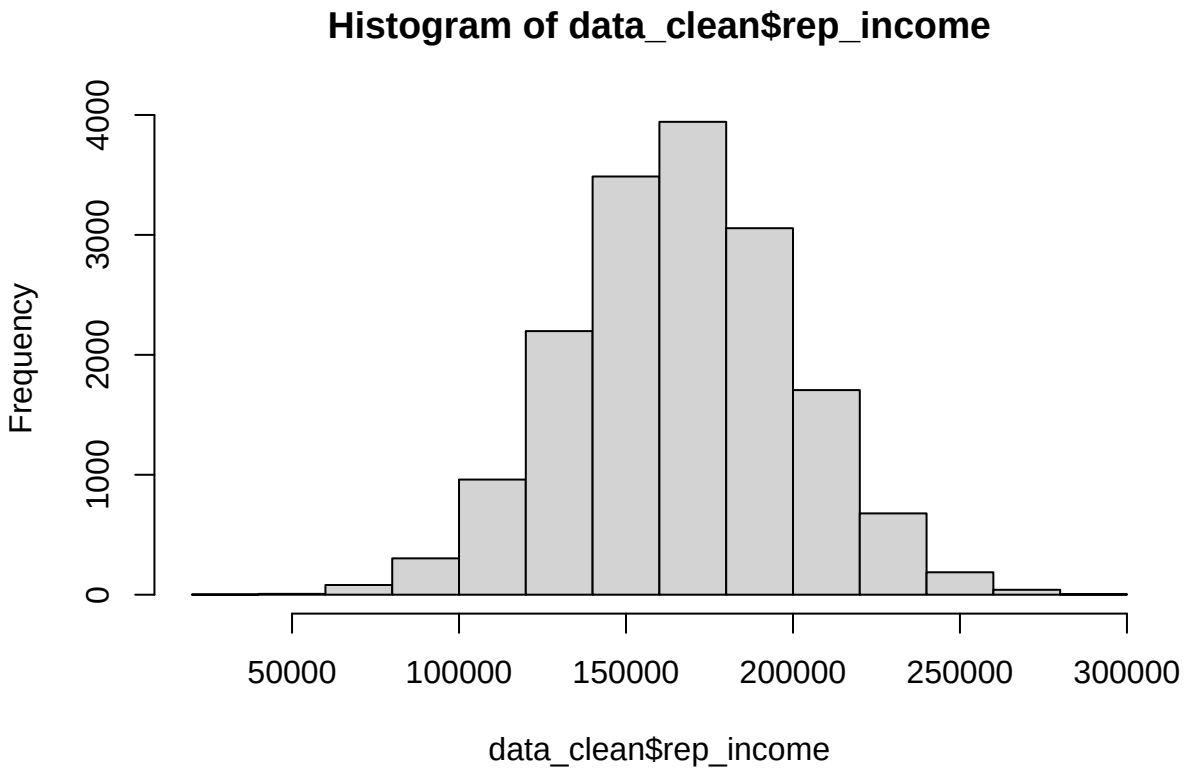
#comments We do have underrepresented levels of education with our 'other' being the most underrepresented and graduate and high school having a big difference amount than college. we have more college amount than everything else

```
table(data_clean$Def_ind)
```

```
##
##     0     1
## 14956  1697
```

#comments this data is imbalanced, some ways on how balance it would be oversampling by increasing the observations or duplicating data. we could also undersample to reduce the majority class, or try SMOTE, but i think we could adjust the class weights in our decision tree and knn models which might work best for this dataset or oversmaple the minority class but this would not determine accuracy since we would just predict it, our data could be imbalance due to more people with less higher education levels that we have counted for or that in general people tend to have high school, and college educations more.

```
hist(data_clean$rep_income)
```



#comments The histogram looks approximately normal with a small slight right skew

```
data_clean %>%
  group_by(rep_education) %>%
  summarise(Default_Rate = mean(Def_ind)) %>%
  arrange(desc(Default_Rate))
```

```
## # A tibble: 4 x 2
##   rep_education Default_Rate
##   <fct>                <dbl>
## 1 high_school         0.118
## 2 college             0.0990
```

```
## 3 graduate            0.0829
## 4 other               0.075
```

#comments High school graduates have the highest default rates, meaning the classes with least education are more likely to borrow loans compared to higher education levels. Our data has more information from high school and below education class than others.

```
set.seed(42)
trainIndex <- createDataPartition(data_clean$Def_ind, p=0.8, list=FALSE)
train <- data_clean[trainIndex, ]
test <- data_clean[-trainIndex, ]
```

```
train$Def_ind <- as.factor(train$Def_ind)
test$Def_ind <- as.factor(test$Def_ind)

knn_model <- train(Def_ind ~ ., data=train, method='knn', tuneLength=5)
```

```
pred_knn <- predict(knn_model, test)
print(confusionMatrix(pred_knn, test$Def_ind))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 2994  317
##          1    8   11
##
##                Accuracy : 0.9024
##                  95% CI : (0.8918, 0.9123)
##     No Information Rate : 0.9015
##     P-Value [Acc > NIR] : 0.4452
##
##                   Kappa : 0.0532
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.99734
##             Specificity : 0.03354
##          Pos Pred Value : 0.90426
##          Neg Pred Value : 0.57895
##              Prevalence : 0.90150
##          Detection Rate : 0.89910
##    Detection Prevalence : 0.99429
##       Balanced Accuracy : 0.51544
##
##        'Positive' Class : 0
##
```

#comments the model correctly predicted 2988 true negatives, and 16 true positives. They guessed incorrectly 13 false negatives and 313 false positives. The accuracy for the knn model is 90.21%, Kappa of 0.00746 which means this is our percentage of better accuracy than random guessing, our recall for default 0 is 99% accurate while 1 default is 4.86%. As well as our precision being higher for non default rather than default.

Looking also at the McNemars test our p value is low which means our model predicts the majority class as the non default class and they do not do very good at predicting actual default people.

```
dt_model <- train(Def_ind ~ ., data=train, method='rpart')

pred_dt <- predict(dt_model, test)
print(confusionMatrix(pred_dt, test$Def_ind))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 2993  302
##          1    9   26
##
##                Accuracy : 0.9066
##                  95% CI : (0.8962, 0.9163)
##     No Information Rate : 0.9015
##     P-Value [Acc > NIR] : 0.1688
##
##                   Kappa : 0.1267
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.99700
##             Specificity : 0.07927
##          Pos Pred Value : 0.90835
##          Neg Pred Value : 0.74286
##              Prevalence : 0.90150
##          Detection Rate : 0.89880
##    Detection Prevalence : 0.98949
##       Balanced Accuracy : 0.53814
##
##        'Positive' Class : 0
##
```

#comments

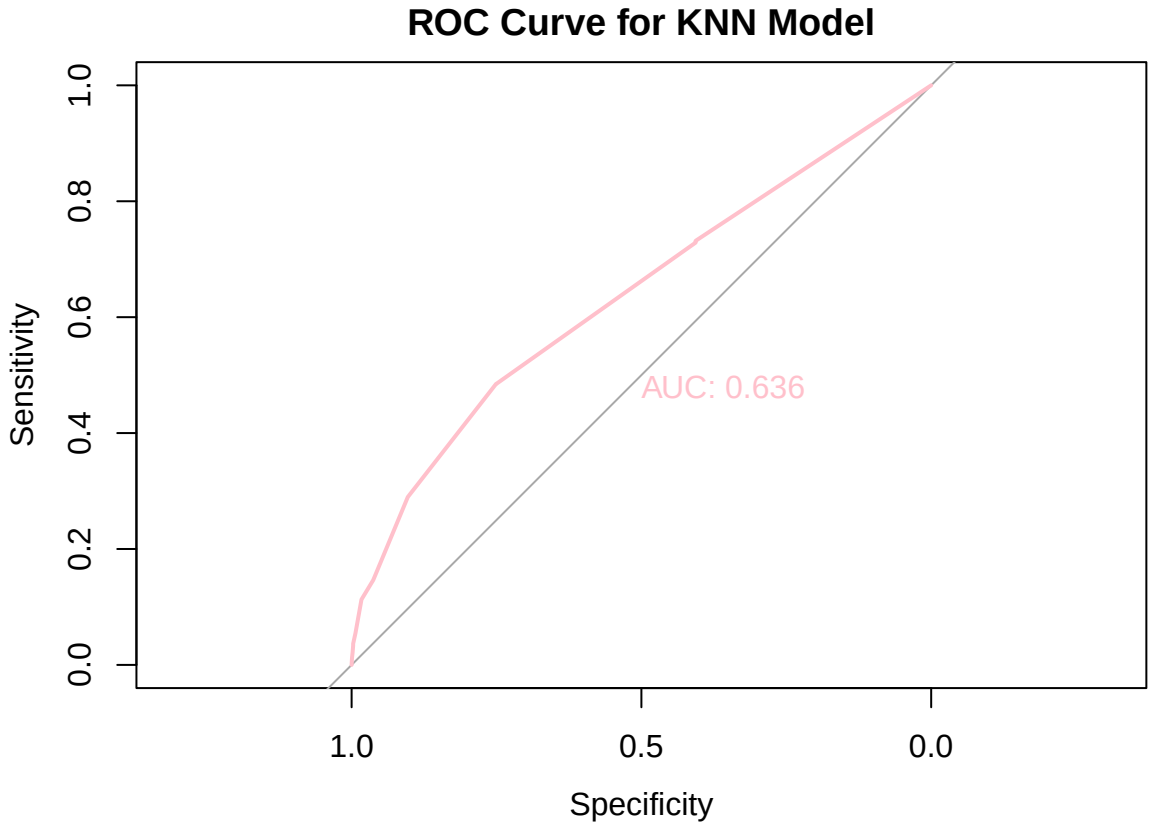
This Decision tree model shows a slight better accuracy than our knn models but we are still seeing a lot of values being predicted higher for our true negatives than our false positives probably due to our imablanced data that we had.

```
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.

##
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

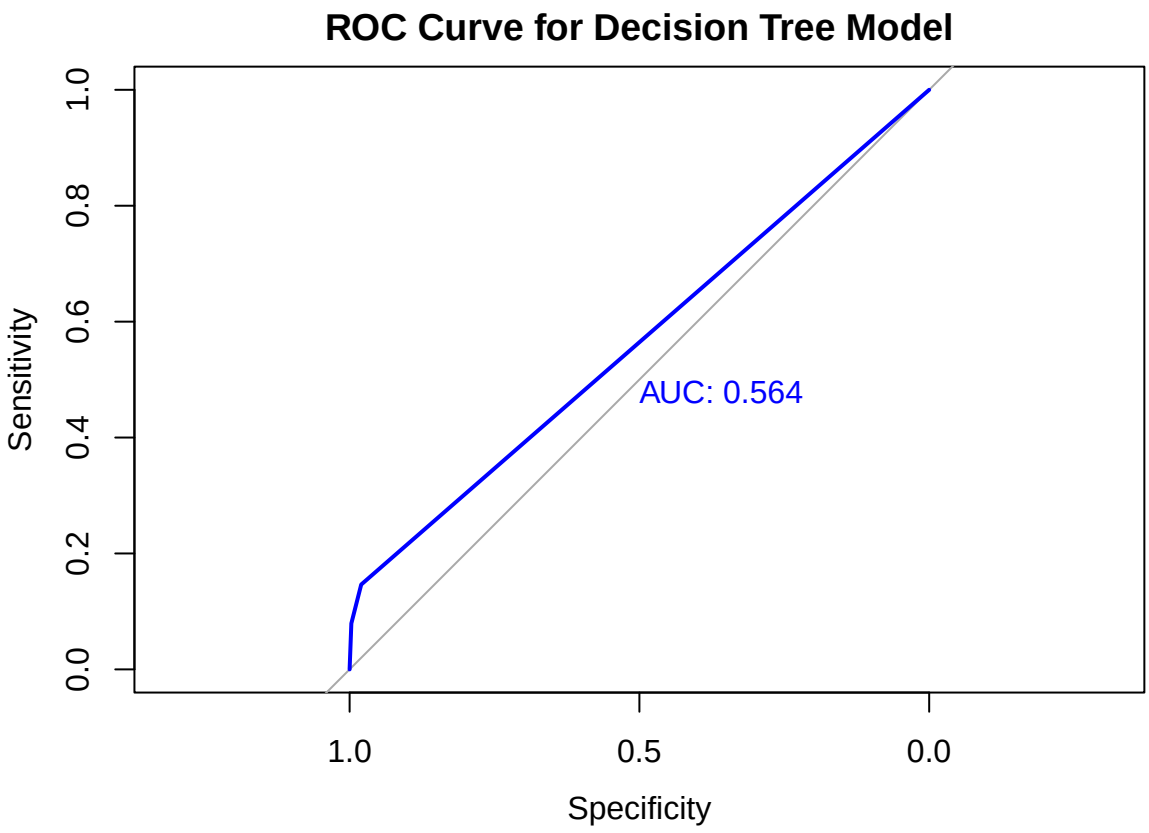
```
knn_pred_probs <- predict(knn_model, test, type = "prob")[, 2]

roc_knn <- roc(test$Def_ind, knn_pred_probs, plot = TRUE, col = "pink", main = "ROC Curve for KNN Model
```

```
## Setting levels: control = 0, case = 1

## Setting direction: controls < cases
```




```
dt_pred_probs <- predict(dt_model, test, type = "prob")[, 2]

roc_dt <- roc(test$Def_ind, dt_pred_probs, plot = TRUE, col = "blue", main = "ROC Curve for Decision Tr
```

```
## Setting levels: control = 0, case = 1

## Setting direction: controls < cases
```



#comments after seeing both scores of auc and how both models knn and decision tree perform, it is easy to say decision tree perform better, i believe our knn model does not help with this type of data set due to they perform the best with smaller and more predictive data sets while decision tree model can help us be more accurate due to our numeric and categorical variables and complex data. As we saw in our auc score it was a lot more accurate than our knn auc score that leaned more towards the left.

Seeing at the top our results for min median max, etc. Our rep income, credit usage history and utilization are the features that are best when predicting default status.