Assignment 04 Trees

BQOM 2578 | Data Mining

Theresa Wohlever

Sunday, October 19, 2025

Table of contents

Executive Summary

How well can we predict county Suicide Rates from the hospital information on a per county basis within Pennsylvania? Combine both county Suicide rate data with all PA hospital data to address this question. Dependent Variable is the County Suicide rate. The Data preparation includes removing a large number of features expected to be unrelated to suicide rates, changing the representation of categorical variables to integers, and joining hospital data and county level suicide data.

Logistic regression provides insight into the impact of included features. Selected features were those shown to demonstrate significant impact on the linear regression. Using these features in a Logistic Regression we can better percieve their impact on County Suicide Rates.

Feature	Logistic Regression	Regression Tree	Classification Tree
`children_hospital`	Very large effect (β = 2.849)	Not Used	Not Used
`psych_over17`	Small effect, but not significant (β = 0.184)	Not Used	Not Used
`discharges1864`	Very small but significant effect (β = 0.0001)	Primary split variable First split at 2,228 Subsequent splits: 153, 254, 349, 523, 8431, etc. Most important predictor OSR^2 = 0.088, MAE = 48.32	Primary and only split variable Main split at 8,431 discharges Tree 1 accuracy: 72.73% Trees 2 & 3 accuracy: 77.27%
`psych_over17_beds_lic`	Negligible effect (β = 0.001)	Not Used	Not Used
`psychiatrists`	x	x	x
`clinic_psychiatric`	x	x	x
`comprehensive_rehab`	x	x	x
`type_of_organization`	x	x	x

These findings are discussed in detail in Logistic Regression Model 2 (Multi-Variable) : Beta Coefficients Discussion.

Data Preparation

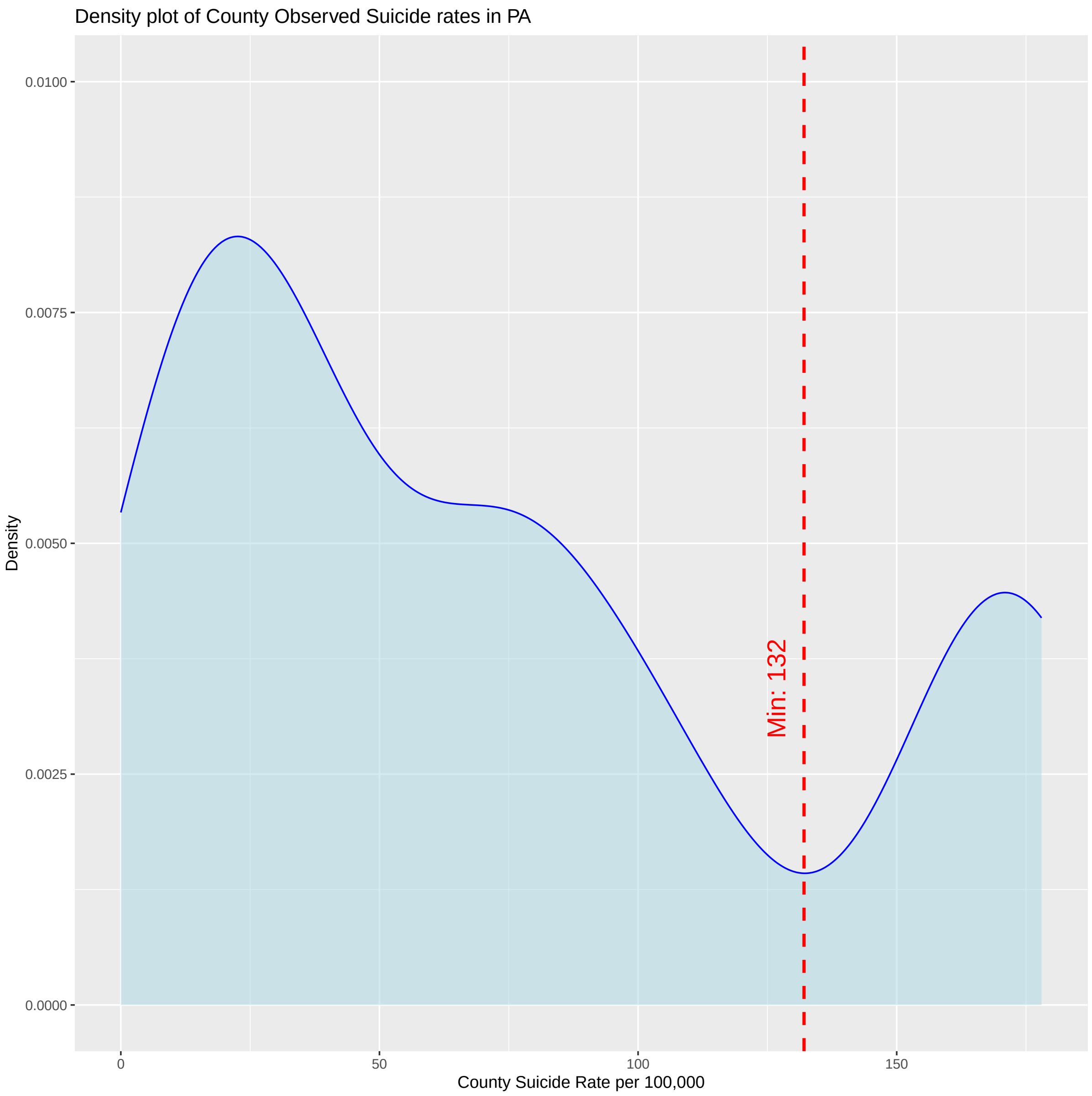
Importing Data, Cleaning, & Wrangling

Data Review

Numeric variables are continuous.

Integer variables are categorical.

```
                 facility_id             type_of_organization
                   "integer"                        "integer"
           children_hospital                     hospital_ltc
                   "numeric"                        "numeric"
                 on_site_ltc                 privateroomexist
                   "numeric"                        "numeric"
        semiprivateroomexist                   discharges1864
                   "numeric"                        "integer"
          alcohol_drug_detox        alcoholdetox_patient_days
                   "integer"                        "integer"
          alcohol_drug_treat            alcoholtreat_beds_lic
                   "integer"                        "integer"
   alcoholtreat_patient_days              comprehensive_rehab
                   "integer"                        "integer"
comprehensive_rehab_beds_lic Comprehensive_rehab_patient_days
                   "integer"                        "integer"
                 psych_0to17             psych_0to17_beds_lic
                   "integer"                        "integer"
    psych_0to17_patient_days                     psych_over17
                   "integer"                        "integer"
       psych_over17_beds_lic        psych_over17_patient_days
                   "integer"                        "integer"
                       detox                         clinpsyc
                   "numeric"                        "numeric"
          clinic_psychiatric                    psychiatrists
                   "integer"                        "integer"
                      target
                   "numeric"
```



Set the cut-off value for 1 or 0 (binary) for Logistic regression is the local minimum of county Suicide Rates.

Split Dataset into Training and Test

We will leave 80% of observations in the training set and 20% in the test set.

```r
#set.seed keeps results random but constant for all using the same seed (so we all will have the same results)
set.seed(1760, sample.kind = "Rejection")
spl = sample(nrow(df),0.8*nrow(df))
head(spl)
```

```
[1] 193  59 139 177 122  20
```

```r
# Split into train and test:
train.df = df[spl,]
test.df = df[-spl,]

dim(df)
```

```
[1] 218  27
```

```r
dim(train.df)
```

```
[1] 174  27
```

```r
dim(test.df)
```

```
[1] 44 27
```

Preliminary Analysis

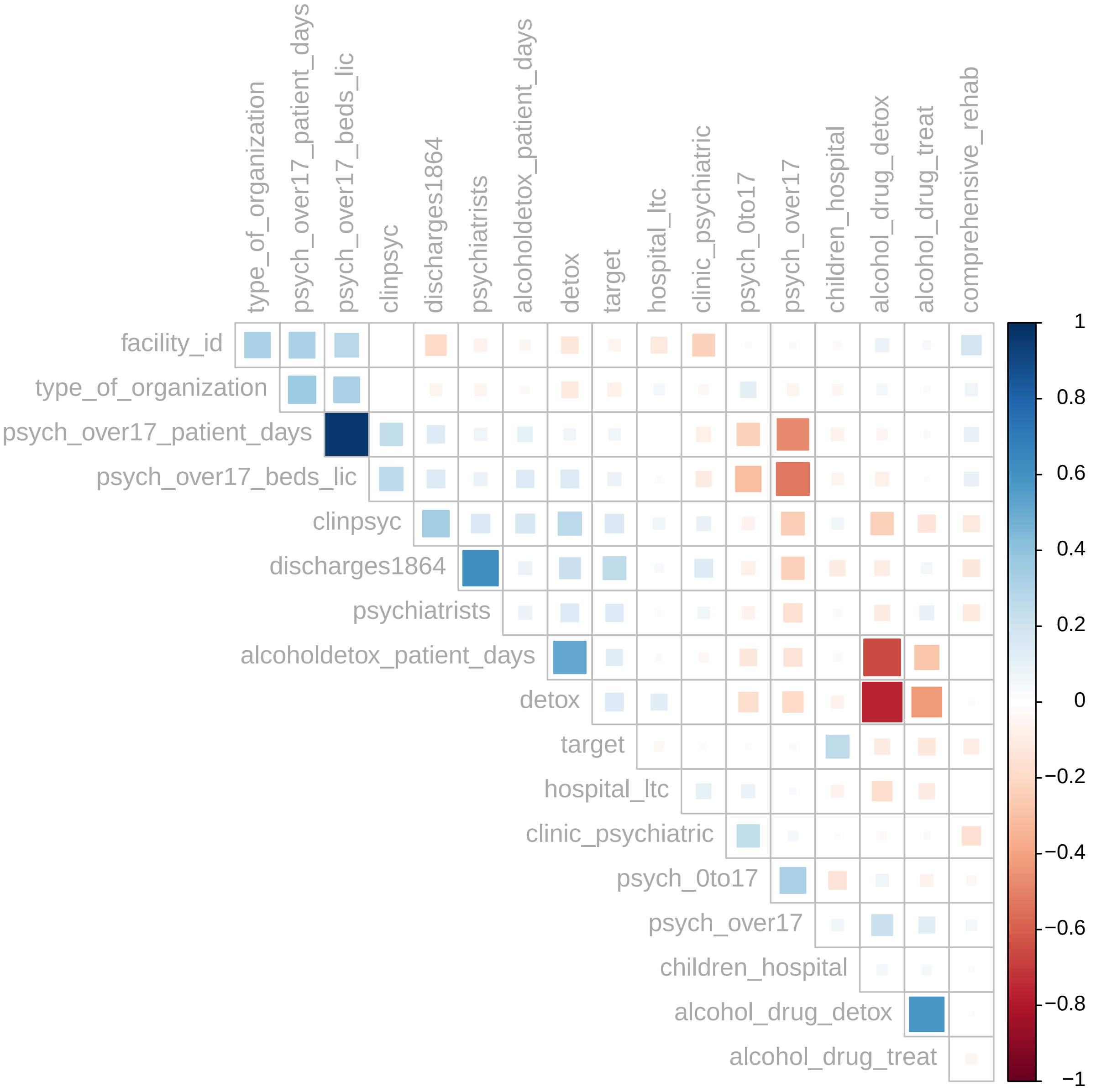
Evaluate Correlation Matrix

```r
## Prep for correlation
df_cor <- df_clean

cor_mat <- cor(df)
cor_threshold <- 0
cor_threshold_count <- 2

cols_above_threshold <- which( colSums(abs(cor_mat) > cor_threshold, na.rm = TRUE) >= cor_threshold_count)
df <- subset(df, select = colnames(cor_mat)[cols_above_threshold] )
cor_mat <- cor(df)
```

```
cor_mat_plot <- round(cor_mat, 2)
cor_mat_plot[is.na(cor_mat_plot)] <- 0 # Replace all NA values with zero
cat(paste(colnames(cor_mat_plot), collapse = "\n"))
```

```
facility_id
type_of_organization
children_hospital
hospital_ltc
discharges1864
alcohol_drug_detox
alcoholdetox_patient_days
alcohol_drug_treat
comprehensive_rehab
psych_0to17
psych_over17
psych_over17_beds_lic
psych_over17_patient_days
detox
clinpsyc
clinic_psychiatric
psychiatrists
target
```

```
corrplot(cor_mat_plot,
  method="square",
  type="upper",
  order="AOE",
  tl.col="darkgrey",
  cl.align.text = "r",
  diag=FALSE,
  number.cex=0.6)
```



Regression

Stepwise Linear Regression

```
model <- lm(target ~ ., data = df)
summary(model)
```

```
Call:
lm(formula = target ~ ., data = df)

Residuals:
   Min     1Q Median     3Q    Max 
-98.80 -40.27 -14.39  26.85 132.00 

Coefficients:
                            Estimate Std. Error t value Pr(>|t|)    
(Intercept)                9.837e+01  5.475e+01   1.797 0.073863 .  
facility_id                5.033e-09  2.712e-07   0.019 0.985212    
type_of_organization      -4.438e+00  3.035e+00  -1.462 0.145204    
children_hospital          9.625e+01  2.053e+01   4.688  5.1e-06 ***
hospital_ltc              -8.800e+00  1.261e+01  -0.698 0.485998    
discharges1864             4.268e-03  1.134e-03   3.763 0.000221 ***
alcohol_drug_detox         6.592e+00  1.572e+01   0.419 0.675419    
alcoholdetox_patient_days  3.616e-03  3.944e-03   0.917 0.360344    
alcohol_drug_treat        -2.762e+01  1.356e+01  -2.037 0.042938 *  
comprehensive_rehab       -7.007e+00  4.397e+00  -1.594 0.112576    
psych_0to17                1.157e+01  6.609e+00   1.750 0.081633 .  
psych_over17               1.091e+01  4.858e+00   2.245 0.025842 *  
psych_over17_beds_lic      3.940e-01  2.758e-01   1.429 0.154690    
psych_over17_patient_days -5.272e-04  8.691e-04  -0.607 0.544840    
detox                      8.600e+00  2.020e+01   0.426 0.670717    
clinpsyc                  -8.861e-01  8.850e+00  -0.100 0.920344    
clinic_psychiatric        -5.928e+00  4.407e+00  -1.345 0.180073    
psychiatrists             -6.321e-02  1.256e-01  -0.503 0.615476    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 54.57 on 200 degrees of freedom
Multiple R-squared:  0.229,	Adjusted R-squared:  0.1634 
F-statistic: 3.494 on 17 and 200 DF,  p-value: 1.028e-05
```

```r
# Perform stepwise regression
step_model_back <- step(model, direction = "backward",trace=0)
summary(step_model_back)
```

```
Call:
lm(formula = target ~ type_of_organization + children_hospital + 
    discharges1864 + alcohol_drug_treat + comprehensive_rehab + 
    psych_0to17 + psych_over17 + psych_over17_beds_lic + clinic_psychiatric, 
    data = df)

Residuals:
    Min      1Q  Median      3Q     Max 
-106.02  -40.33  -10.98   28.88  134.42 

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)    
(Intercept)           1.295e+02  3.853e+01   3.361 0.000924 ***
type_of_organization -5.079e+00  2.881e+00  -1.763 0.079397 .  
children_hospital     9.551e+01  1.995e+01   4.788 3.20e-06 ***
discharges1864        4.001e-03  8.419e-04   4.753 3.74e-06 ***
alcohol_drug_treat   -2.928e+01  1.057e+01  -2.772 0.006083 ** 
comprehensive_rehab  -6.924e+00  4.281e+00  -1.617 0.107307    
psych_0to17           9.837e+00  6.380e+00   1.542 0.124614    
psych_over17          1.085e+01  4.695e+00   2.312 0.021772 *  
psych_over17_beds_lic  2.496e-01  9.096e-02   2.744 0.006595 ** 
clinic_psychiatric   -6.337e+00  4.253e+00  -1.490 0.137781    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 53.87 on 208 degrees of freedom
Multiple R-squared:  0.2186,    Adjusted R-squared:  0.1848 
F-statistic: 6.466 on 9 and 208 DF,  p-value: 4.224e-08
```

```r
step_model_forward <- step(model, direction = "forward",trace=0)
summary(step_model_forward)
```

```
Call:
lm(formula = target ~ facility_id + type_of_organization + children_hospital + 
    hospital_ltc + discharges1864 + alcohol_drug_detox + alcoholdetox_patient_days + 
    alcohol_drug_treat + comprehensive_rehab + psych_0to17 + 
    psych_over17 + psych_over17_beds_lic + psych_over17_patient_days + 
    detox + clinpsyc + clinic_psychiatric + psychiatrists, data = df)

Residuals:
   Min     1Q Median     3Q    Max 
-98.80 -40.27 -14.39  26.85 132.00 
```

```
Coefficients:
                            Estimate Std. Error t value Pr(>|t|)    
(Intercept)                9.837e+01  5.475e+01   1.797 0.073863 .  
facility_id                5.033e-09  2.712e-07   0.019 0.985212    
type_of_organization      -4.438e+00  3.035e+00  -1.462 0.145204    
children_hospital          9.625e+01  2.053e+01   4.688  5.1e-06 ***
hospital_ltc              -8.800e+00  1.261e+01  -0.698 0.485998    
discharges1864             4.268e-03  1.134e-03   3.763 0.000221 ***
alcohol_drug_detox         6.592e+00  1.572e+01   0.419 0.675419    
alcoholdetox_patient_days  3.616e-03  3.944e-03   0.917 0.360344    
alcohol_drug_treat        -2.762e+01  1.356e+01  -2.037 0.042938 *  
comprehensive_rehab       -7.007e+00  4.397e+00  -1.594 0.112576    
psych_0to17                1.157e+01  6.609e+00   1.750 0.081633 .  
psych_over17               1.091e+01  4.858e+00   2.245 0.025842 *  
psych_over17_beds_lic      3.940e-01  2.758e-01   1.429 0.154690    
psych_over17_patient_days -5.272e-04  8.691e-04  -0.607 0.544840    
detox                      8.600e+00  2.020e+01   0.426 0.670717    
clinpsyc                  -8.861e-01  8.850e+00  -0.100 0.920344    
clinic_psychiatric        -5.928e+00  4.407e+00  -1.345 0.180073    
psychiatrists             -6.321e-02  1.256e-01  -0.503 0.615476    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 54.57 on 200 degrees of freedom
Multiple R-squared:  0.229,	Adjusted R-squared:  0.1634 
F-statistic: 3.494 on 17 and 200 DF,  p-value: 1.028e-05
```

Logistic Regression

```r
df <- df_clean
# Update target to be binomial
df$target <- ifelse(df$target < target_bin_cutoff, 0, 1)
train.df = df[spl,]
test.df = df[-spl,]



# Logistic Regression with FOI
logreg <- glm(model_feature_input, data=df, family="binomial")
summary(logreg)
```

```
Call:
glm(formula = model_feature_input, family = "binomial", data = df)

Coefficients:
                        Estimate Std. Error z value Pr(>|z|)
(Intercept)           -1.288e+00  8.774e-01  -1.468 0.142029
children_hospital      2.865e+00  8.650e-01   3.313 0.000925 ***
psych_over17           3.007e-01  2.305e-01   1.305 0.192016
discharges1864         1.023e-04  4.452e-05   2.298 0.021540 *
psych_over17_beds_lic  5.740e-03  4.584e-03   1.252 0.210548
psychiatrists          7.944e-04  5.246e-03   0.151 0.879633
clinic_psychiatric     1.199e-01  2.028e-01   0.591 0.554350
comprehensive_rehab   -3.175e-01  1.955e-01  -1.624 0.104344
type_of_organization  -3.022e-01  1.666e-01  -1.814 0.069601 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 229.83  on 217  degrees of freedom
Residual deviance: 199.78  on 209  degrees of freedom
AIC: 217.78

Number of Fisher Scoring iterations: 5
```

```
coeftable <- data.frame(col1=coef(logreg),col2=exp(coef(logreg)))
colnames(coeftable)<-c('Coefficient (log-odds)','e^coefficient (odds)')
coeftable
```

```
                      Coefficient (log-odds) e^coefficient (odds)
(Intercept)                    -1.2882266421            0.2757594
children_hospital               2.8654745262           17.5573827
psych_over17                    0.3007478757            1.3508687
discharges1864                  0.0001023188            1.0001023
psych_over17_beds_lic           0.0057395644            1.0057561
psychiatrists                   0.0007943842            1.0007947
clinic_psychiatric              0.1198965844            1.1273803
comprehensive_rehab            -0.3175226706            0.7279502
type_of_organization           -0.3022347150            0.7391646
```

```
#
# Confusion Matrix
#
df$PredLogOdds <- df$PredProbs <- predict(logreg, newdata=df)
df$PredProbs <- predict(logreg, newdata=df, type="response")
# type="response" gives the probability
summary(df$PredProbs)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
0.01934 0.12504 0.17154 0.22018 0.25243 0.87779 
```

Trees: Regression

```
df <- df_clean
train.df = df[spl,]
test.df = df[-spl,]



rpart(model_feature_input, data=train.df)
```

```
n= 174 

node), split, n, deviance, yval
      * denotes terminal node

 1) root 174 611057.600  73.31609  
   2) discharges1864< 2228 110 348748.300  61.28182  
     4) clinic_psychiatric>=1.5 45 127295.800  45.22222  
       8) discharges1864< 337.5 21  28082.670  26.66667 *
       9) discharges1864>=337.5 24  85655.960  61.45833 *
     5) clinic_psychiatric< 1.5 65 201811.600  72.40000  
      10) psychiatrists>=35.5 8    703.875  30.62500 *
      11) psychiatrists< 35.5 57 185187.100  78.26316  
        22) psychiatrists< 15.5 49 143652.000  71.79592  
          44) discharges1864< 153 20  46627.800  52.90000  
            88) discharges1864>=68 10   8729.600  32.20000 *
            89) discharges1864< 68 10  29328.400  73.60000 *
          45) discharges1864>=153 29  84958.140  84.82759  
            90) discharges1864>=254 21  54546.950  75.38095 *
            91) discharges1864< 254 8  23617.880 109.62500 *
        23) psychiatrists>=15.5 8  26932.880 117.87500 *
   3) discharges1864>=2228 64 218998.000  94.00000
     6) psych_over17_beds_lic< 62 57 194989.700  88.59649
      12) psychiatrists< 7.5 11  13195.640  45.81818 *
      13) psychiatrists>=7.5 46 156850.600  98.82609
        26) psych_over17_beds_lic>=27 16  43872.000  72.00000 *
        27) psych_over17_beds_lic< 27 30  95323.470 113.13330
          54) comprehensive_rehab>=2 18  49460.940  96.05556 *
          55) comprehensive_rehab< 2 12  32738.250 138.75000 *
     7) psych_over17_beds_lic>=62 7   8792.000 138.00000 *
```

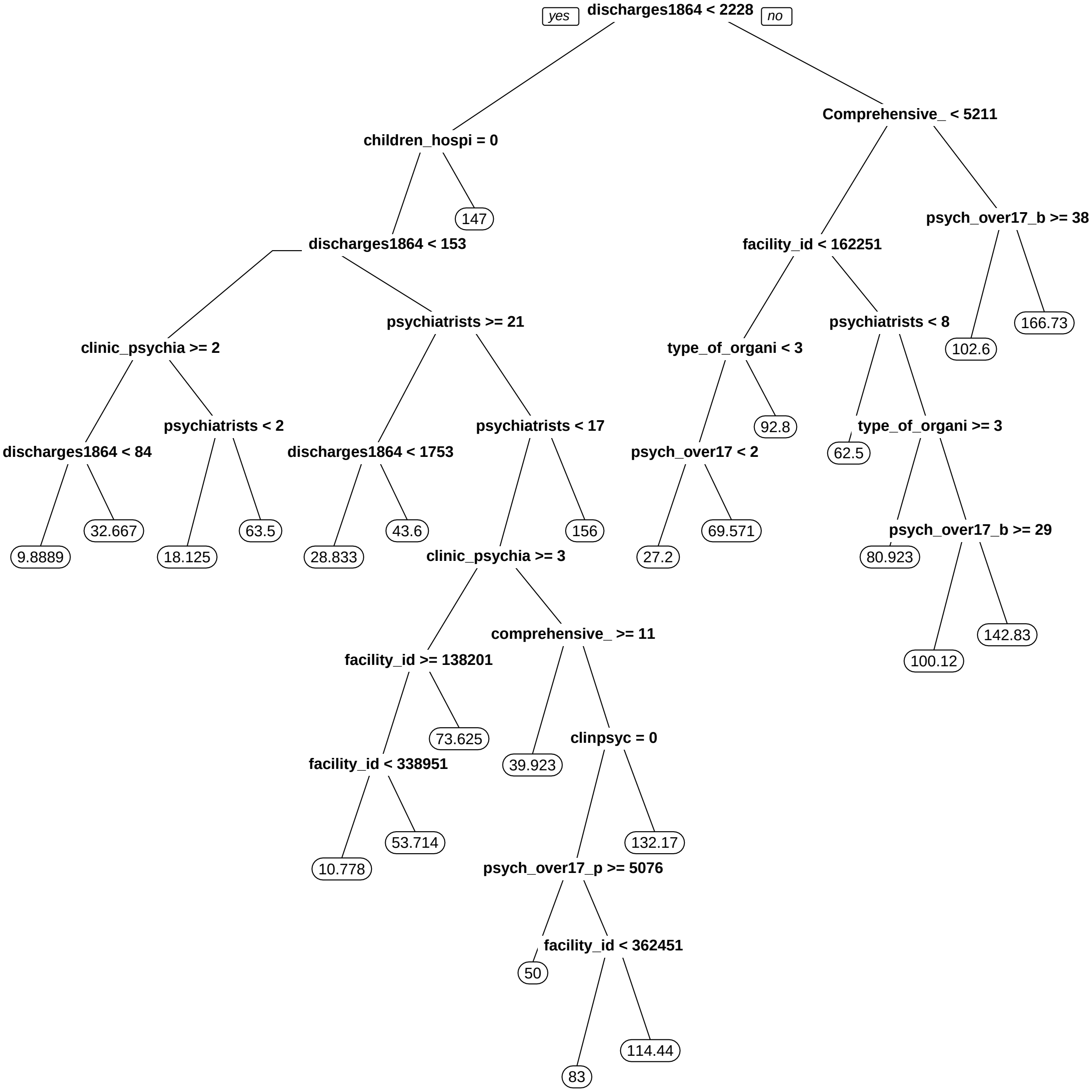
```
(train.df%>%filter(children_hospital==1))$target%>%mean()
```

```
[1] 134.6
```

```
(train.df%>%filter(children_hospital==0))$target%>%mean()
```

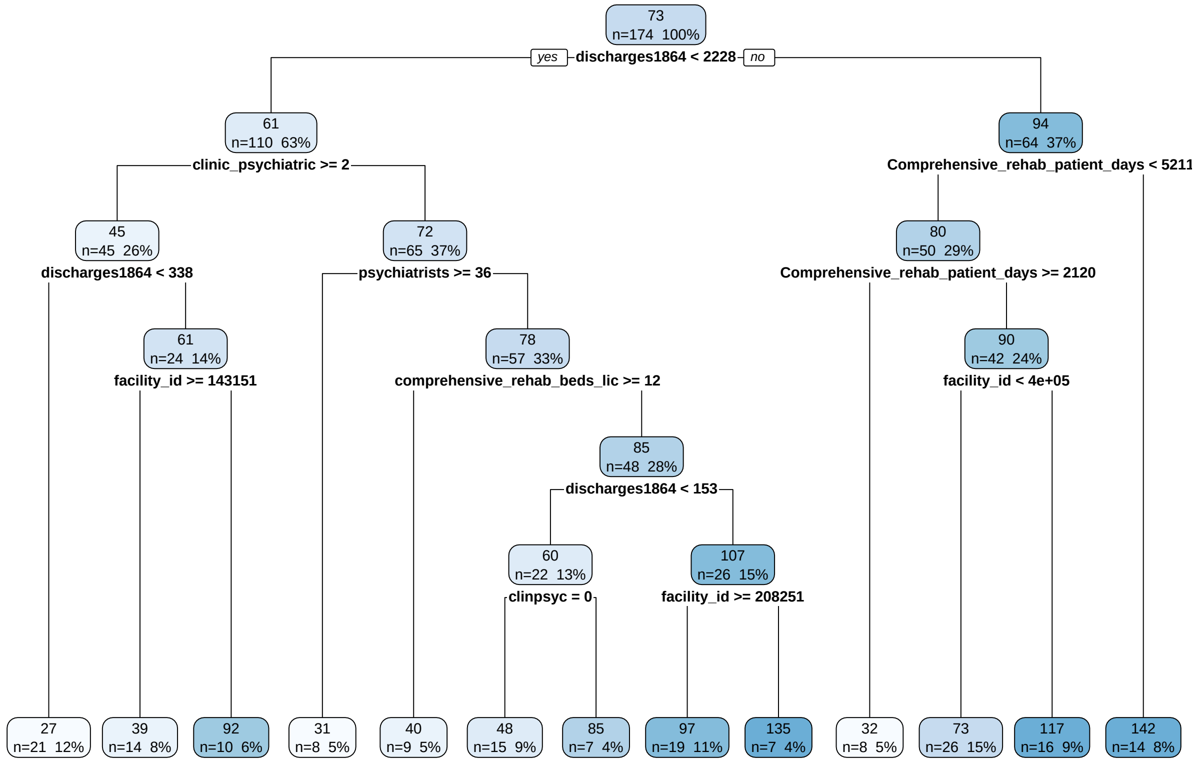
```
[1] 71.50296
```

```
prp(rpart(target ~ .,data=df, method="anova",minbucket=5,cp=0.0001),digits=-5)
```

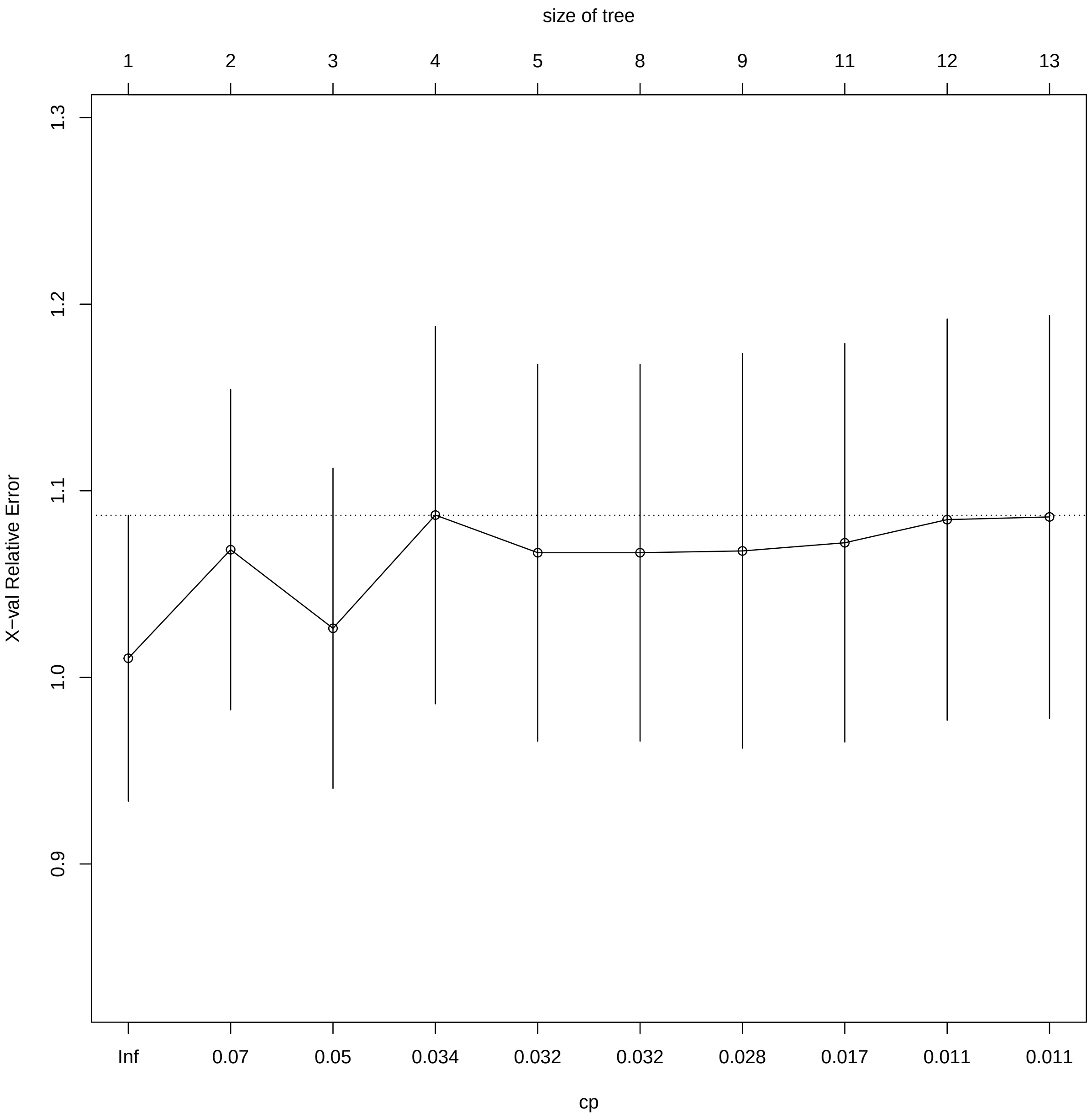


Cross Validation

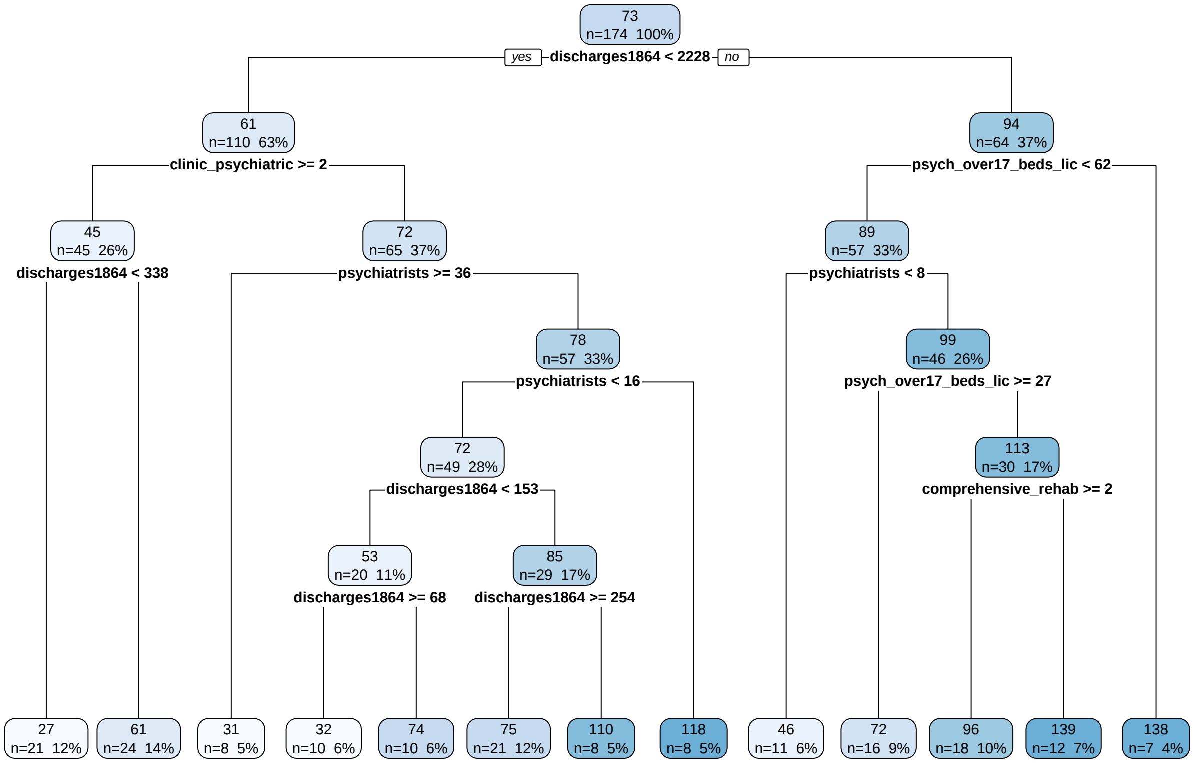
```
set.seed(1760, sample.kind = "Rejection")

tree_cv_all <- rpart(target ~ ., data=train.df, method="anova")
rpart.plot(tree_cv_all, digits=-2, extra=101)
```



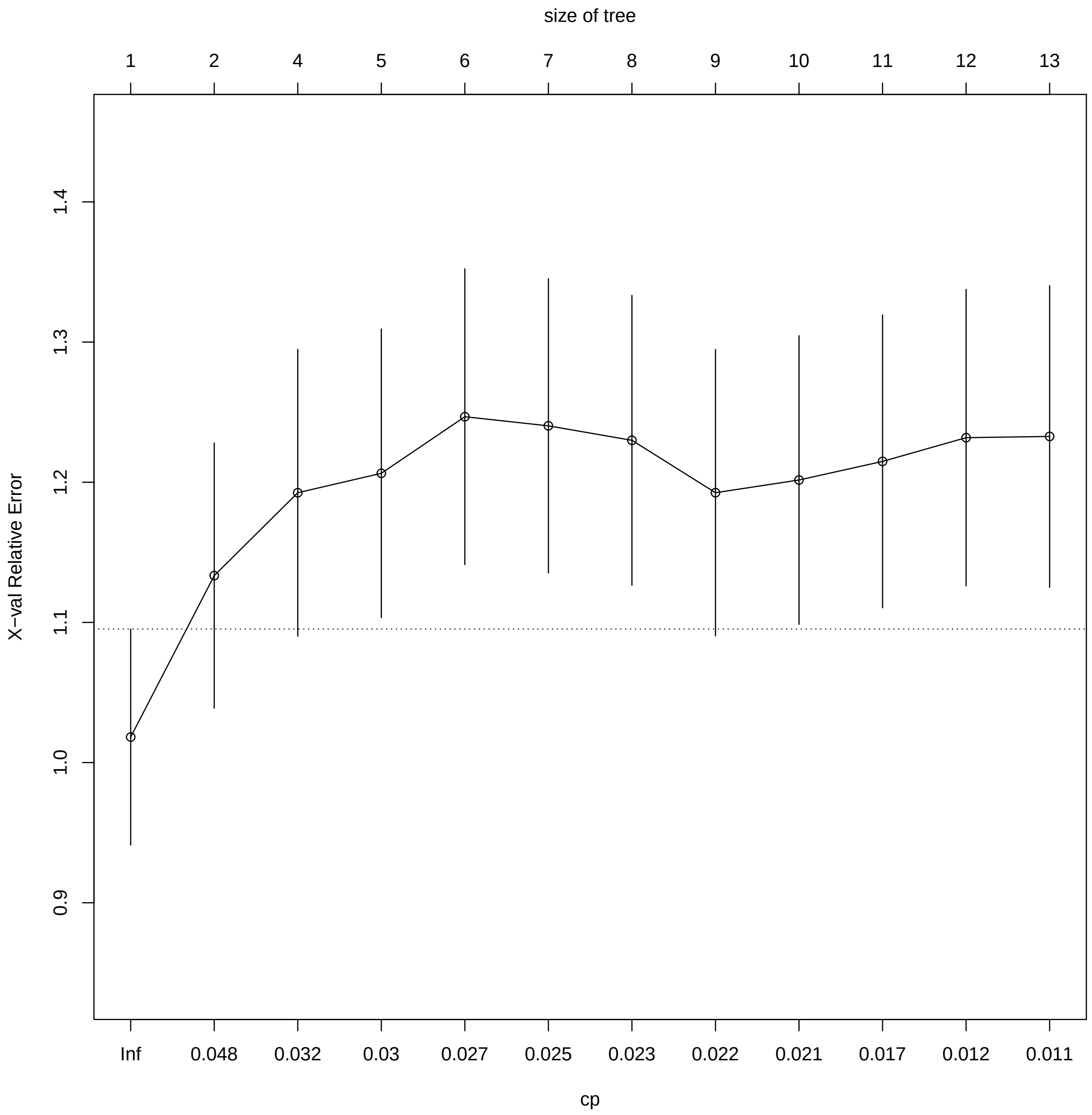
```
plotcp(tree_cv_all)
```



```
tree_cv_foi <- rpart(model_feature_input,data=train.df, method="anova")
rpart.plot(tree_cv_foi, digits=-2, extra=101)
```



```r
plotcp(tree_cv_foi)
```



```
mean_train = mean(train.df$target)
SST = sum((test.df$target - mean_train)^2)

## Predictions
test.df$pred_cv_all = predict(tree_cv_all, newdata=test.df)
test.df$pred_cv_foi = predict(tree_cv_foi, newdata=test.df)

# Compute the sum of squared errors (SSE) using our tree:
```

```
SSE_all = sum((test.df$target - test.df$pred_cv_all)^2)
print(paste("Tree All has a SSE of", SSE_all))
```

```
[1] "Tree All has a SSE of 186610.784586498"
```

```
OSR2_all = 1 - SSE_all/SST
OSR2_all
```

```
[1] -0.1561529
```

```
# Compute the sum of squared errors (SSE) using our tree:
SSE_foi = sum((test.df$target - test.df$pred_cv_foi)^2)
print(paste("Tree FOI has a SSE of", SSE_foi))
```

```
[1] "Tree FOI has a SSE of 155112.10444095"
```

```
OSR2_foi = 1 - SSE_foi/SST
OSR2_foi
```

```
[1] 0.03899819
```

Let’s see the MAE for comparisons:

```
MAE = mean(abs(test.df$target - test.df$pred01))


MAE
```

```
[1] NaN
```

Trees: Classification

```
df <- df_clean
# Update target to be binomial
df$target <- ifelse(df$target < target_bin_cutoff, 0, 1)
train.df = df[spl,]
test.df = df[-spl,]



#It is always a good practice to see the proportion of observations we have for each case
table(df$target)
```

```
  0   1 
170  48 
```

```r
prop.table(table(df$target))  # prop is "proportion"
```

```
        0         1 
0.7798165 0.2201835 
```

```r
cat("\n For the train dataset: ")      #  \n is a "new line" printing control
```

```
 For the train dataset: 
```

```r
prop.table(table(train.df$target))
```

```
        0         1 
0.7816092 0.2183908 
```

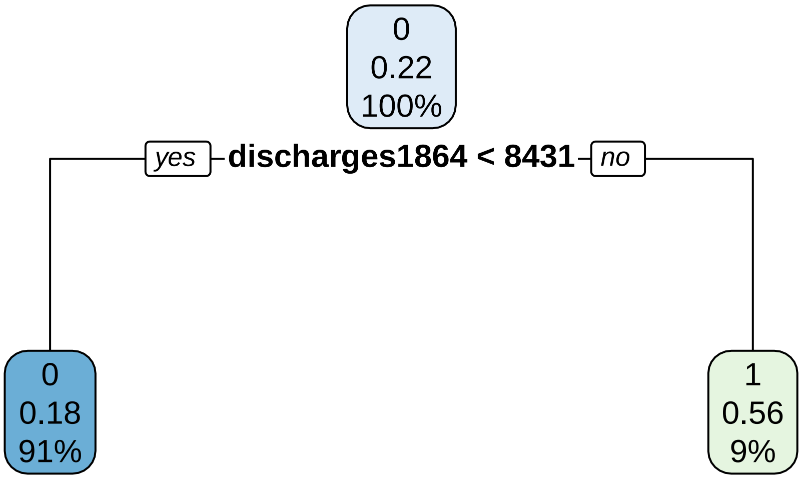
```r
cat("\n For the test dataset: ")
```

```
 For the test dataset: 
```

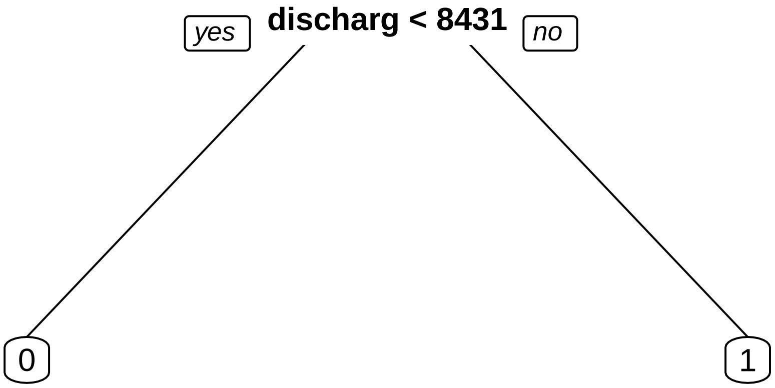
```r
prop.table(table(test.df$target))
```

```
        0         1 
0.7727273 0.2272727 
```

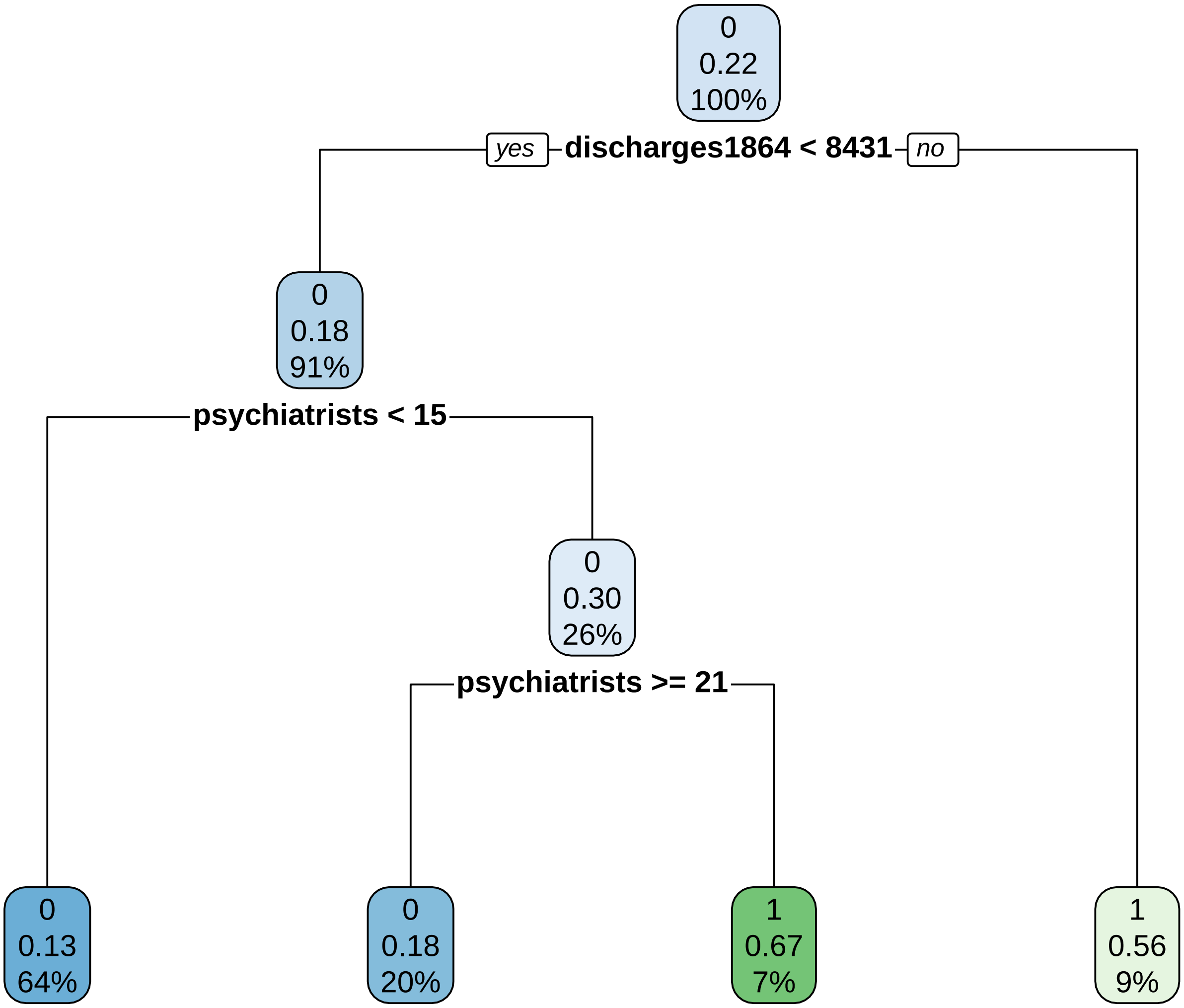
```r
tree_all <-rpart(target ~ ., data=train.df, method="class",cp=0.05)
rpart.plot(tree_all, digits=-2)
```



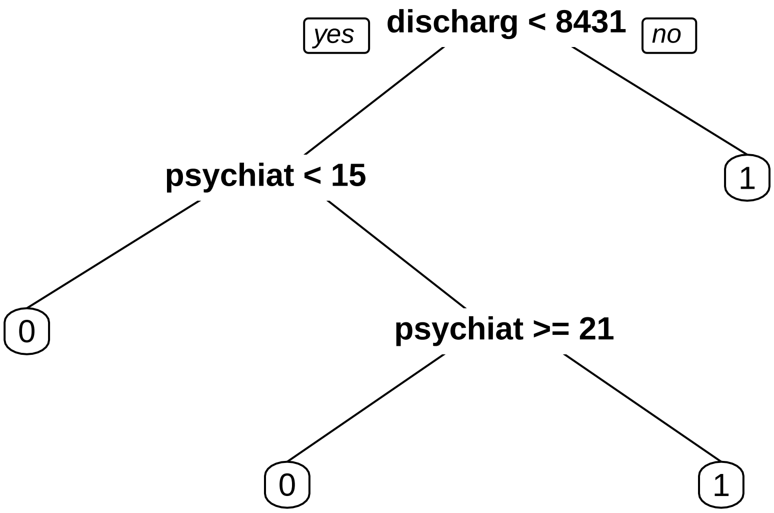
```
prp(tree_all)
```



```
tree_foi<-rpart(model_feature_input, data=train.df, method="class", cp=0.05)
rpart.plot(tree_foi,digits=-2)
```


0
0.22
100%
yes
discharges1864 < 8431
no
0
0.18
91%
psychiatrists < 15
0
0.30
26%
psychiatrists >= 21
0
0.13
64%
0
0.18
20%
1
0.67
7%
1
0.56
9%

```
prp(tree_foi)
```



Cross-Validation

```
cat("All Features Tree:")
```

```
All Features Tree:
```

```
test.df$pred_all = predict(tree_all, newdata = test.df, type="class")
confusionMatrix(test.df$pred_all,as.factor(test.df$target), positive="1")
```

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 30  8
         1  4  2

               Accuracy : 0.7273
                 95% CI : (0.5721, 0.8504)
    No Information Rate : 0.7727
    P-Value [Acc > NIR] : 0.8176

                  Kappa : 0.0959

 Mcnemar's Test P-Value : 0.3865
```

```
            Sensitivity : 0.20000
            Specificity : 0.88235
         Pos Pred Value : 0.33333
         Neg Pred Value : 0.78947
             Prevalence : 0.22727
         Detection Rate : 0.04545
   Detection Prevalence : 0.13636
      Balanced Accuracy : 0.54118

       'Positive' Class : 1
```

```
acc_tree_all <- sum(test.df$pred_all == test.df$target) / nrow(test.df)
print(paste("Accuracy for All Features Tree", round(acc_tree_all * 100, 2), "%"))
```

```
[1] "Accuracy for All Features Tree 72.73 %"
```

```
cat("FOI Tree:")
```

```
FOI Tree:
```

```
test.df$pred_foi = predict(tree_foi, newdata = test.df, type="class")
confusionMatrix(test.df$pred_foi,as.factor(test.df$target), positive="1")
```

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 27  8
         1  7  2

               Accuracy : 0.6591
                 95% CI : (0.5008, 0.7951)
    No Information Rate : 0.7727
    P-Value [Acc > NIR] : 0.9717

                  Kappa : -0.0061

 Mcnemar's Test P-Value : 1.0000

            Sensitivity : 0.20000
            Specificity : 0.79412
         Pos Pred Value : 0.22222
         Neg Pred Value : 0.77143
             Prevalence : 0.22727
         Detection Rate : 0.04545
   Detection Prevalence : 0.20455
      Balanced Accuracy : 0.49706

       'Positive' Class : 1
```

```
acc_tree_foi <- sum(test.df$pred_foi == test.df$target) / nrow(test.df)
print(paste("Accuracy for FOI Tree:", round(acc_tree_foi * 100, 2), "%"))
```

```
[1] "Accuracy for FOI Tree: 65.91 %"
```

Compare Classification Tree with Logistic regressions

```
## Logistic Regression
test.df$predregprobs_foi = predict(logreg, newdata = test.df, type="response")
test.df$predreg1<-ifelse(test.df$predregprobs_foi>0.5,1,0)
acc_reg <- sum(test.df$predregprobs_foi == test.df$target) / nrow(test.df)
print(paste("Accuracy for Logistic Regression 1:", round(acc_reg * 100, 2), "%"))
```

```
[1] "Accuracy for Logistic Regression 1: 0 %"
```

```
ConfMatReg <- confusionMatrix(data=as.factor(test.df$predreg1),reference=as.factor(test.df$target), positive = "1")
ConfMatReg
```

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 33  6
         1  1  4

               Accuracy : 0.8409
                 95% CI : (0.6993, 0.9336)
    No Information Rate : 0.7727
    P-Value [Acc > NIR] : 0.1859

                  Kappa : 0.45

 Mcnemar's Test P-Value : 0.1306

            Sensitivity : 0.40000
            Specificity : 0.97059
         Pos Pred Value : 0.80000
         Neg Pred Value : 0.84615
             Prevalence : 0.22727
         Detection Rate : 0.09091
   Detection Prevalence : 0.11364
      Balanced Accuracy : 0.68529

       'Positive' Class : 1
```

```
# Classification Tree
test.df$predregtreeprobs_foi = predict(tree_foi, newdata = test.df, type="class")
test.df$predreg2 <- ifelse(test.df$predregtreeprobs_foi >0.5 ,1 ,0)
acc_reg_tree <- sum(test.df$predregtreeprobs_foi == test.df$target) / nrow(test.df)
print(paste("Accuracy for Classification Tree:", round(acc_reg_tree * 100, 2), "%"))
```

```
[1] "Accuracy for Classification Tree: 65.91 %"
```

```
ConfMatTreeReg <-confusionMatrix(data=as.factor(test.df$predregtreeprobs_foi),reference=as.factor(test.df$target), positive = "1")
ConfMatTreeReg
```

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 27  8
         1  7  2

               Accuracy : 0.6591
                 95% CI : (0.5008, 0.7951)
    No Information Rate : 0.7727
    P-Value [Acc > NIR] : 0.9717
```

```
                 Kappa : -0.0061

Mcnemar's Test P-Value : 1.0000

           Sensitivity : 0.20000
           Specificity : 0.79412
        Pos Pred Value : 0.22222
        Neg Pred Value : 0.77143
            Prevalence : 0.22727
        Detection Rate : 0.04545
  Detection Prevalence : 0.20455
     Balanced Accuracy : 0.49706

      'Positive' Class : 1
```

References

Hospital Data: https://www.pa.gov/psych_over17ncies/health/health-statistics/health-facilities/hospital-reports

Suicide by County Data: https://www.phaim.health.pa.gov/EDD/WebForms/DeathCntySt.aspx