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) | Regression Tree | Classification Tree | |`psych_over17`| Small effect, but not significant (β = 0.184) | Regression Tree | Classification Tree | | `discharges1864`| Very small but significant effect (β = 0.0001) | Regression Tree | Classification Tree | | `psych_over17_beds_lic` | Negligible effect (β = 0.001) | Regression Tree | Classification Tree |

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.

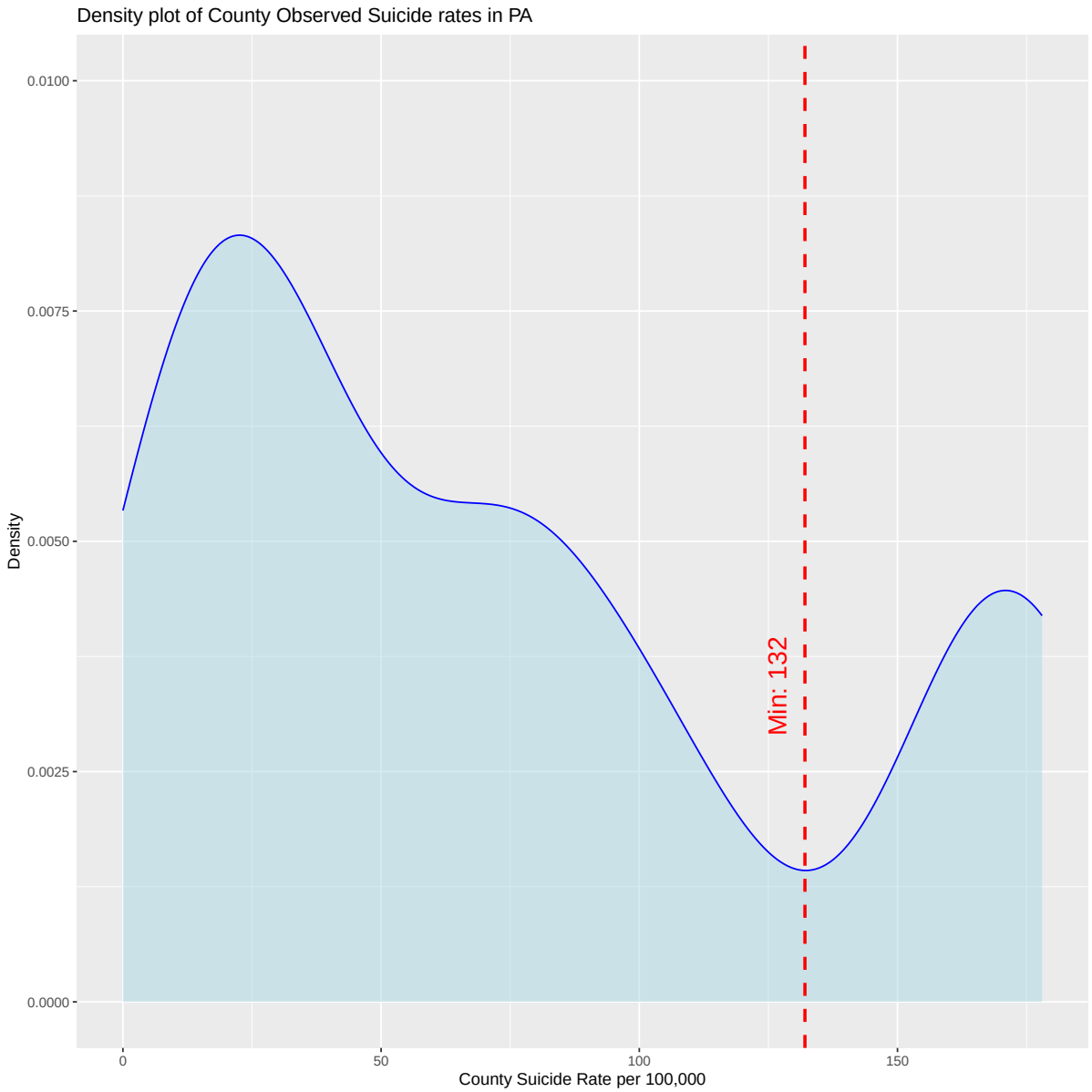
```
##
## Variable TYPES
##


# sapply(df, typeof)
sapply(df, class)
```

```
                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"
```

```r
##
## Visualize target values
##

# Histogram of target values
target_density <- density(df$target)

# Convert the density estimate to a function
dens_func <- approxfun(target_density$x, target_density$y)

# Use optimize() to find the minimum in a specified interval (choose based on your data)
result <- optimize(dens_func, interval = c(min(df$target), max(df$target)))
local_min_x <- result$minimum     # The x value where local minimum occurs
local_min_y <- result$objective   # The minimum density value

#  Create density plot with ggplot2 and add vertical line at minimum
df_density <- data.frame(x = df$target)
ggplot(df_density, aes(x = df$target)) +
  geom_density(fill = "lightblue", color = "blue", alpha = 0.5) +
  geom_vline(xintercept = local_min_x , color = "red", linetype = "dashed", size = 1) +
  annotate("text", x = local_min_x, y = local_min_y + 0.002,
    label = sprintf("Min: %.0f", local_min_x), color = "red", angle = 90, vjust = -1, size = 6) +
```

```
  labs(title = "Density plot of County Observed Suicide rates in PA", x = "County Suicide Rate per 100,000", y = "Density") +
  coord_cartesian(ylim = c(0, 0.01))
```




```
## Make TARGET binary for logistic regression
target_bin_cutoff <- local_min_x
```

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

```r
##
## Visualize ALL data across target values
## Normalized Scatter
##
minMax <- function(x) { (x - min(x)) / (max(x) - min(x))}
df_norm <- as.data.frame(lapply(df, minMax))
df_long <- pivot_longer(df_norm, cols = -target, names_to = "variable", values_to = "value")
ggplot(df_long, aes(x = target, y = value, color = variable)) +
  geom_point(size = 2) +
  theme_minimal() +
  theme(legend.position = "bottom") +
  guides(color = guide_legend(nrow = 7, title = "Feature")) +
  labs(x = "", y = "", color = "Feature")
```

No clear clustering patterns across all variables vs. County Suicide Rates (Regression Target).

Split Dataset into Training and Test

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

```
#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
```

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

dim(df)
```

```
[1] 218  27
```

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

```
[1] 174  27
```

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

```
[1] 44 27
```

Preliminary Analysis

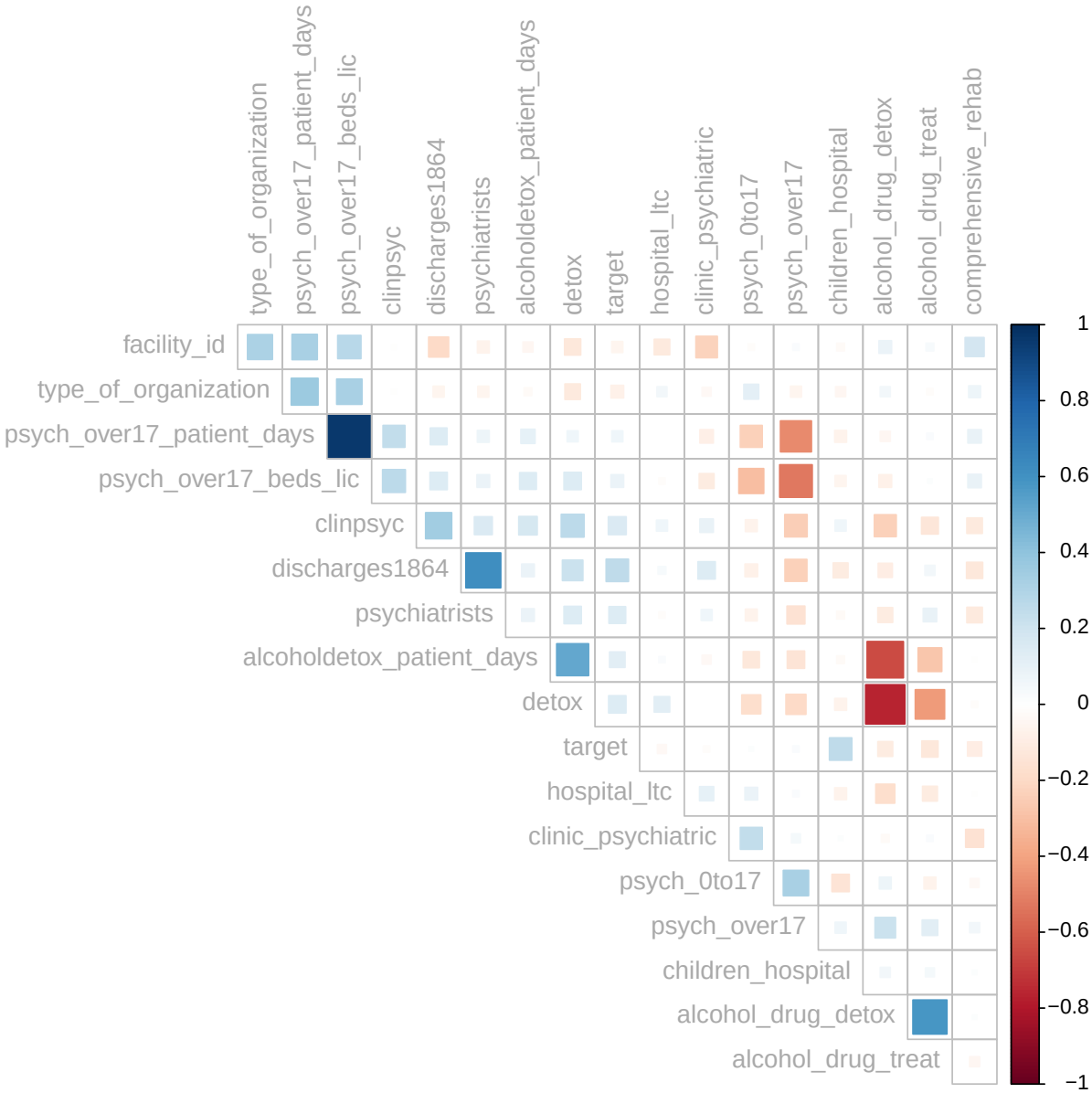
Evaluate Correlation Matrix

```
## 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)
```

```r
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
```

```r
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

```
# For more info: https://www.rdocumentation.org/packpsych_over17s/stats/versions/3.6.2/topics/step
```

```r
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

```
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 ONLY discharges1864
```

```r
logreg_ch <- glm(target ~ discharges1864, data=df, family="binomial")
summary(logreg_ch)
```

```
Call:
glm(formula = target ~ discharges1864, family = "binomial", data = df)

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)    -1.600e+00  2.084e-01  -7.677 1.63e-14 ***
discharges1864  9.391e-05  3.215e-05   2.921  0.00349 **
---
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: 221.17  on 216  degrees of freedom
AIC: 225.17

Number of Fisher Scoring iterations: 4
```

```r
# Logistic Regression with discharges1864 +
#  children_hospital + psych_over17  + psych_over17_beds_lic
logreg_chdisch <- glm(target ~ discharges1864  +  children_hospital + psych_over17
             + psych_over17_beds_lic, data=df, family="binomial")
summary(logreg_chdisch)
```

```
Call:
glm(formula = target ~ discharges1864 + children_hospital + psych_over17 +
    psych_over17_beds_lic, family = "binomial", data = df)

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)           -2.271e+00  6.144e-01  -3.697 0.000218 ***
discharges1864         1.180e-04  3.465e-05   3.404 0.000663 ***
children_hospital      2.849e+00  8.489e-01   3.356 0.000790 ***
psych_over17           1.836e-01  2.194e-01   0.837 0.402656
psych_over17_beds_lic  1.335e-03  3.948e-03   0.338 0.735195
---
```

```
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: 206.84  on 213  degrees of freedom
AIC: 216.84

Number of Fisher Scoring iterations: 4
```

```r
logreg <- logreg_chdisch
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)	-2.2712892545	0.1031791
discharges1864	0.0001179614	1.0001180
children_hospital	2.8490898212	17.2720540
psych_over17	0.1835961539	1.2015305
psych_over17_beds_lic	0.0013351614	1.0013361

```r
#
# 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.
 0.1110  0.1530  0.1606  0.2202  0.2018  0.8103
```

```r
# transform prediction into either 1 (Elevated Suicide Rate) or 0 (Suicide Rate) using a cutoff point
for (cutoff in c(0.25, 0.15, 0.1, 0.05))
{
  df$PredHighSuicide <- ifelse(df$PredProbs >= cutoff,1,0)
  summary(df$PredHighSuicide)
  cat(paste("For a cutoff point of",
    cutoff, "the proportion of Counties predicted to have high suicide rates is",
    round(mean(df$PredHighSuicide),2)), "\n\n")
}
```

```
For a cutoff point of 0.25 the proportion of Counties predicted to have high suicide rates is 0.18

For a cutoff point of 0.15 the proportion of Counties predicted to have high suicide rates is 0.83

For a cutoff point of 0.1 the proportion of Counties predicted to have high suicide rates is 1

For a cutoff point of 0.05 the proportion of Counties predicted to have high suicide rates is 1
```

Trees: Regression

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

rpart(target~ children_hospital, data=train.df)
```

```
n= 174

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

1) root 174 611057.6 73.31609 *
```

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

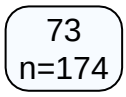
```
[1] 134.6
```

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

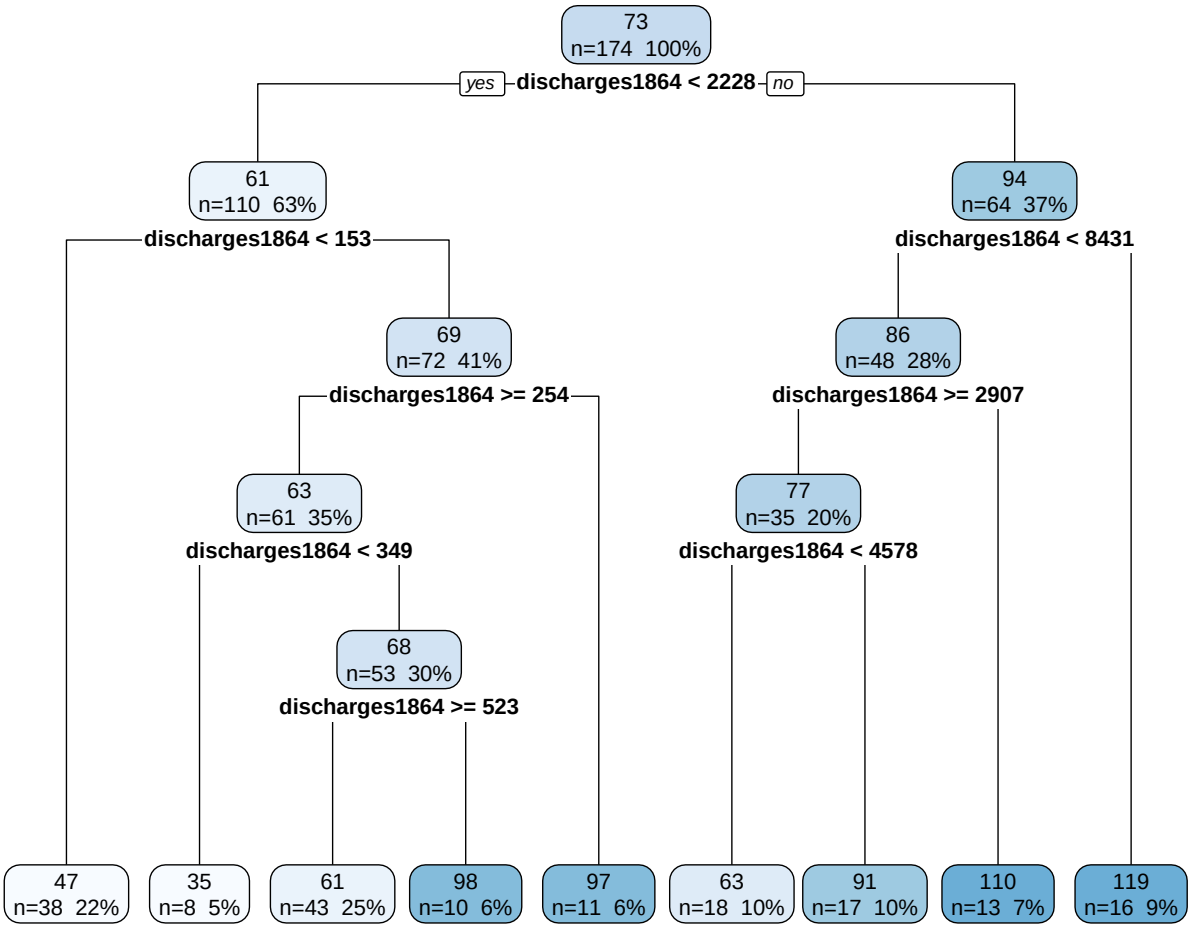
```
[1] 71.50296
```

Regression Tree with children_hospital and discharges1864

```
tree1<-rpart(target~ children_hospital, data=train.df)
rpart.plot(tree1,digits=-2,extra=1)
```



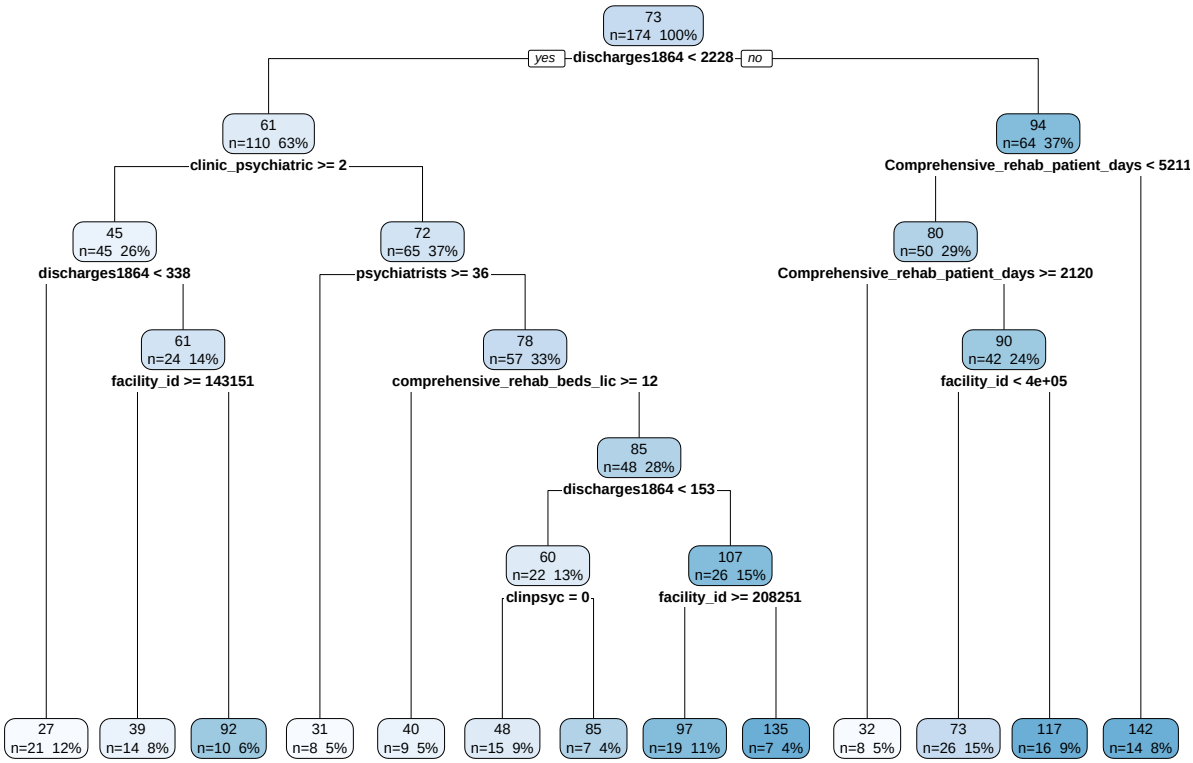
```
tree2<-rpart(target~ discharges1864, data=train.df)
rpart.plot(tree2,digits=-2,extra=101)
```




```
tree3<-rpart(target~ children_hospital+discharges1864, data=train.df)
rpart.plot(tree3,digits=-2,extra=101)
```

Regression Tree with all the variables

```
tree4<-rpart(target~ ., data=train.df)
rpart.plot(tree4,digits=-2,extra=101)
```



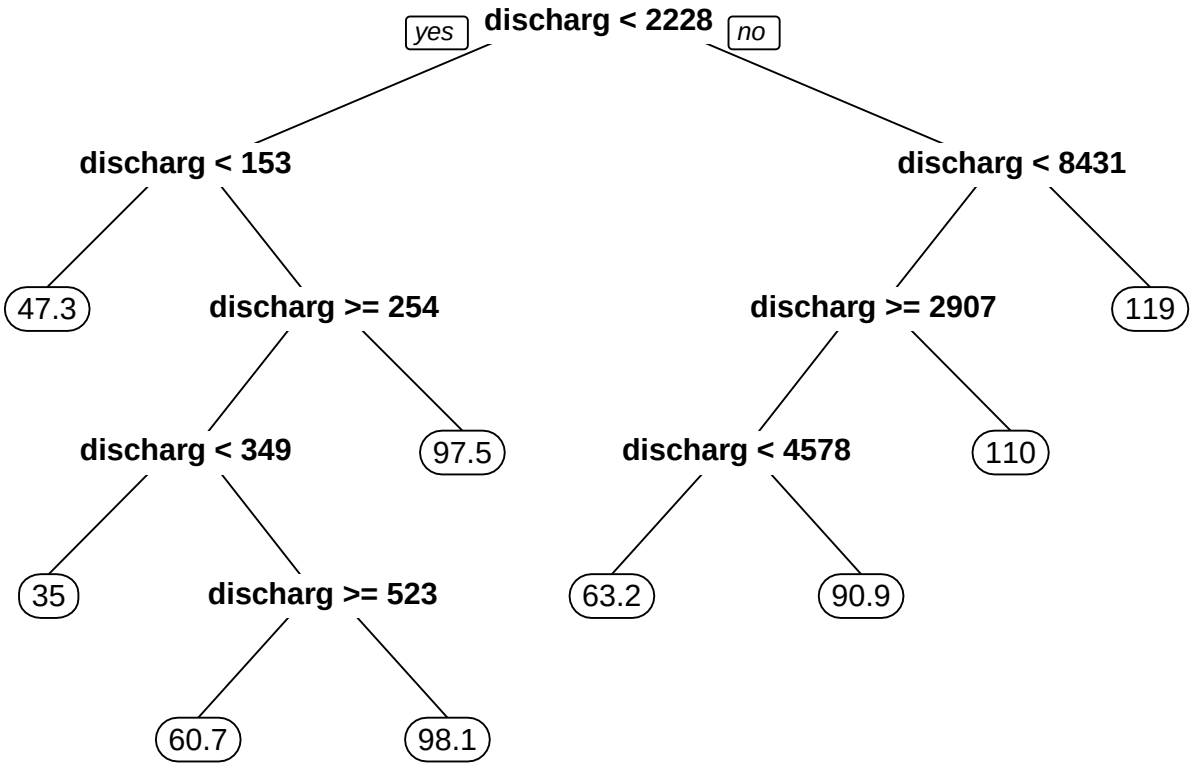

```
df$logtarget<-log(df$target)
summary(df$logtarget)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   -Inf   3.045   4.248    -Inf   4.682   5.182
```

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

prp(tree1,digits=-3)
```

73.3

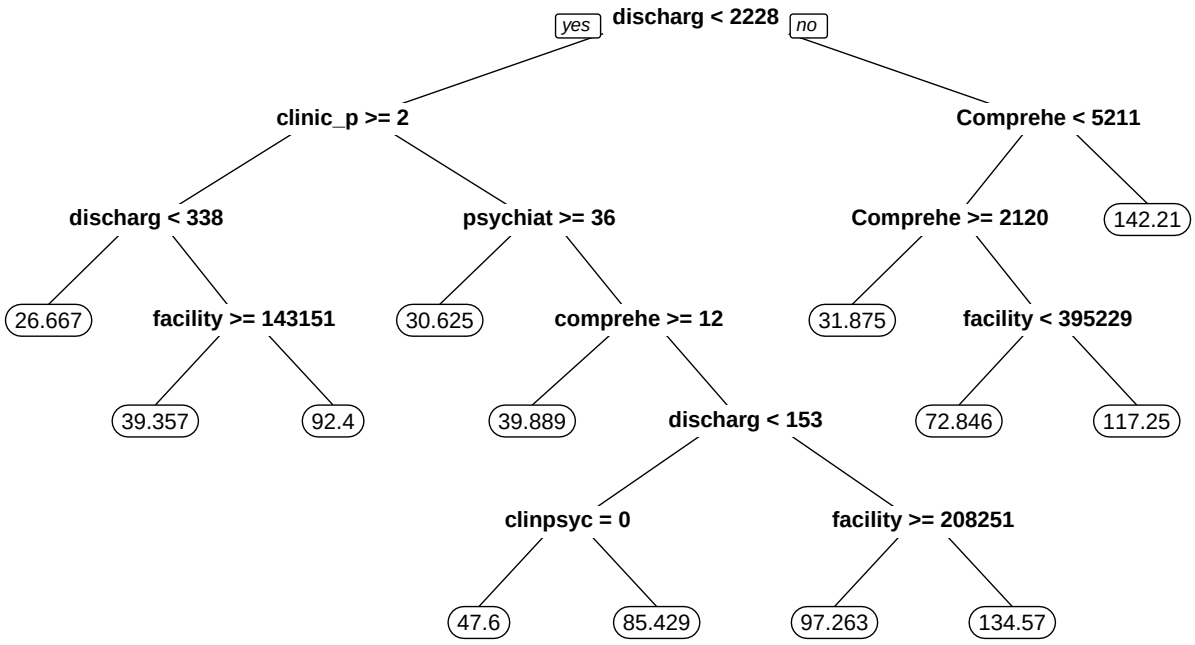
```
prp(tree3,digits=-3)
```



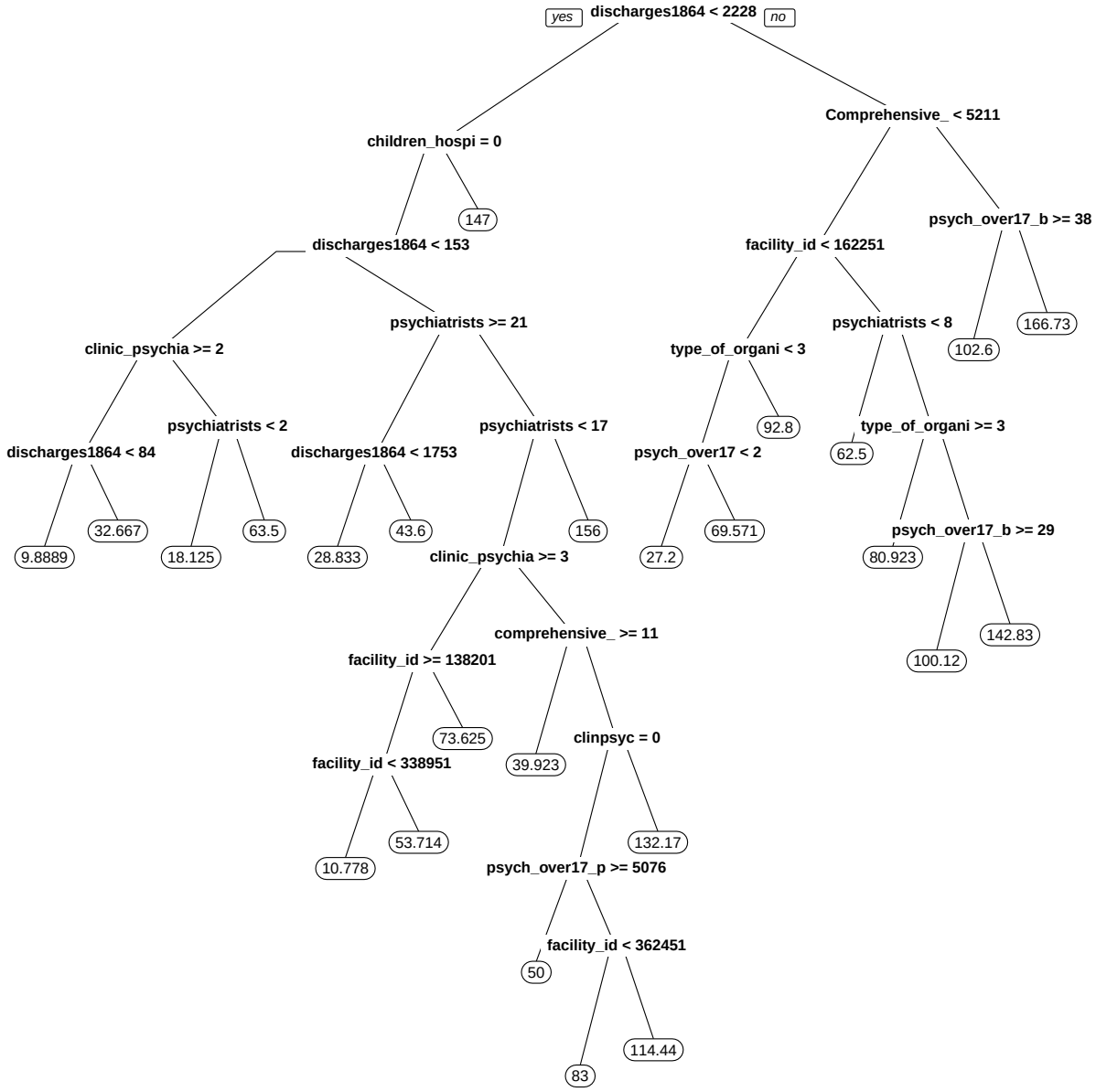

```
names(df)
```

```
 [1] "facility_id"                   "type_of_organization"
 [3] "children_hospital"             "hospital_ltc"
 [5] "on_site_ltc"                   "privateroomexist"
 [7] "semiprivateroomexist"          "discharges1864"
 [9] "alcohol_drug_detox"            "alcoholdetox_patient_days"
[11] "alcohol_drug_treat"            "alcoholtreat_beds_lic"
[13] "alcoholtreat_patient_days"     "comprehensive_rehab"
[15] "comprehensive_rehab_beds_lic"  "Comprehensive_rehab_patient_days"
[17] "psych_0to17"                   "psych_0to17_beds_lic"
[19] "psych_0to17_patient_days"      "psych_over17"
[21] "psych_over17_beds_lic"         "psych_over17_patient_days"
[23] "detox"                         "clinpsyc"
[25] "clinic_psychiatric"            "psychiatrists"
[27] "target"                        "logtarget"
```

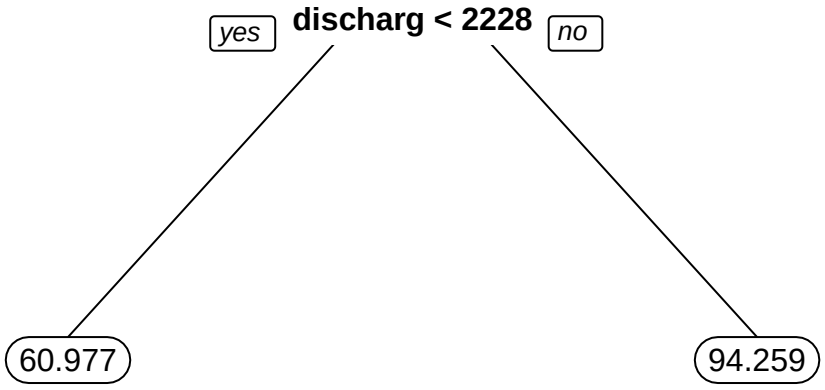
```
basetree<-rpart(target ~ .-logtarget,data=train.df)
prp(basetree,digits=-5)
```



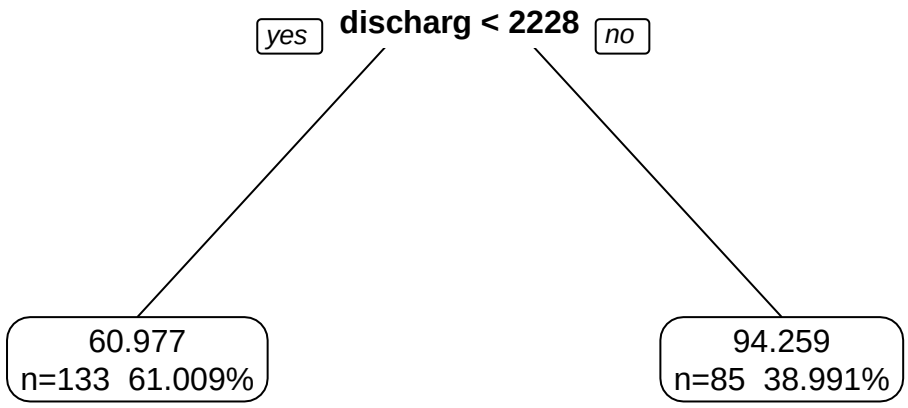

```
#try different cp values to get a bigger tree
prp(rpart(target ~ .-logtarget,data=df, method="anova",minbucket=5,cp=0.0001),digits=-5)
```




```
prp(rpart(target ~ .-logtarget,data=df, method="anova",minbucket=50,cp=0.001),digits=-5)
```



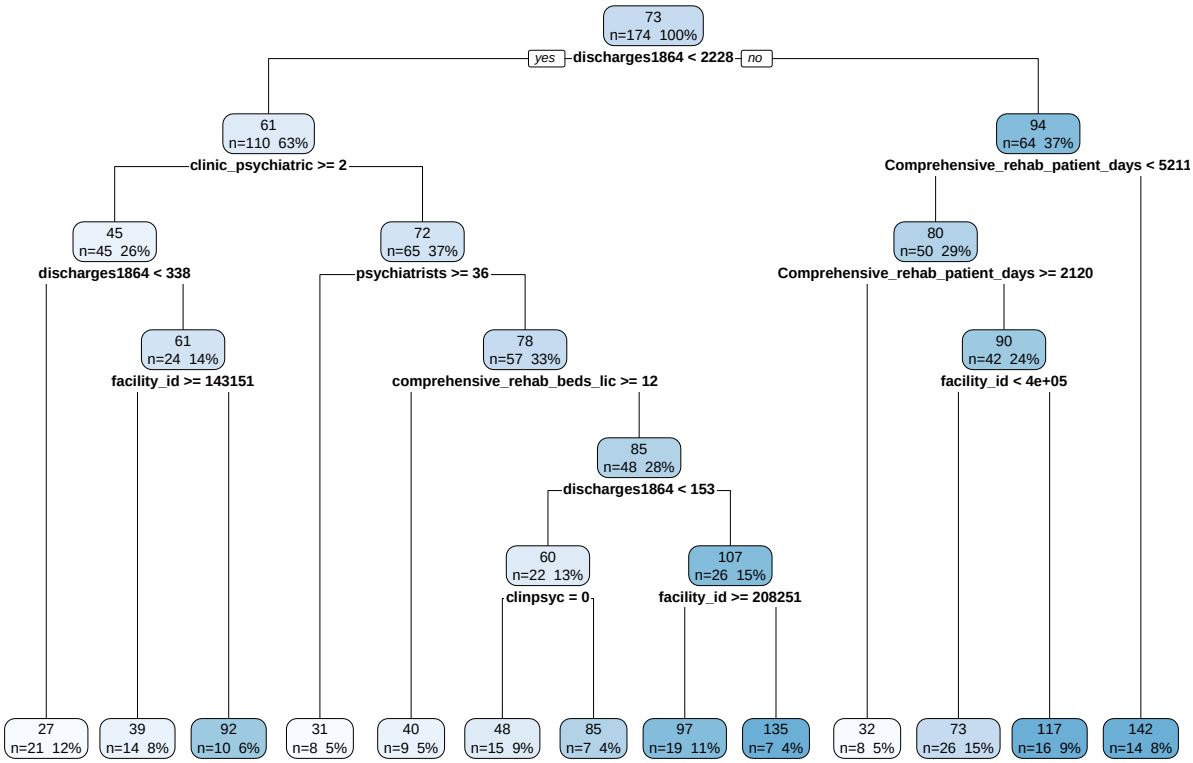

```
prp(rpart(target ~ .-logtarget,data=df, method="anova",minbucket=50,cp=0.01),digits=-5)
prp(rpart(target ~ .-logtarget,data=df, method="anova",minbucket=50,cp=0.01),digits=-5,extra=101)
```



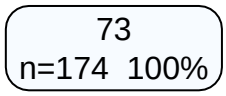

```
# extra = 101 displays observations in each leaf and percentpsych_over17
```

Cross Validation

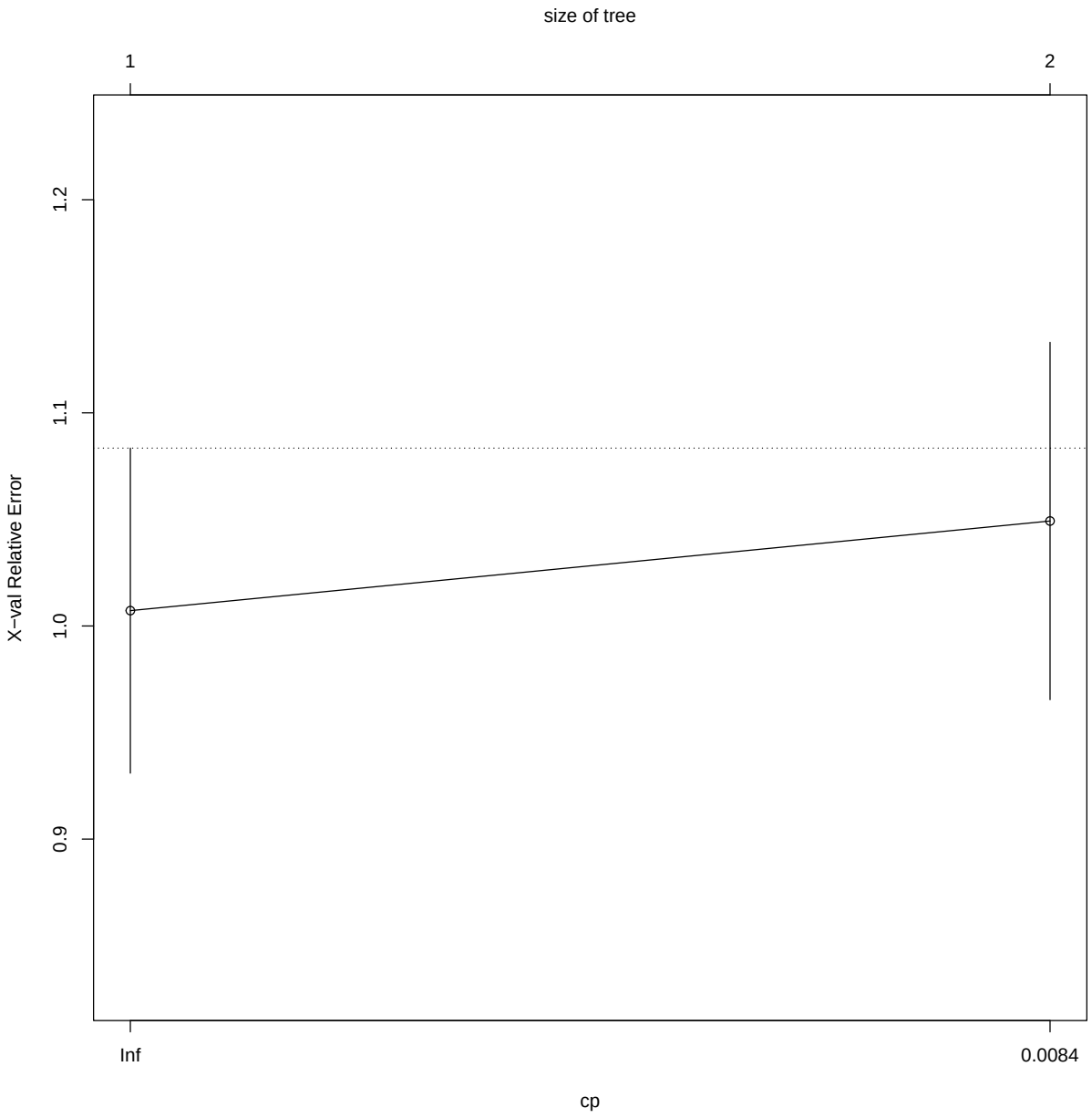
```
set.seed(1760, sample.kind = "Rejection")

#make a tree with a very small value of cp. Not 0 because it will take a long time creating too many splits
tree_cv = rpart(target ~ .-logtarget,data=train.df, method="anova")
rpart.plot(tree_cv,digits=-2,extra=101)
```



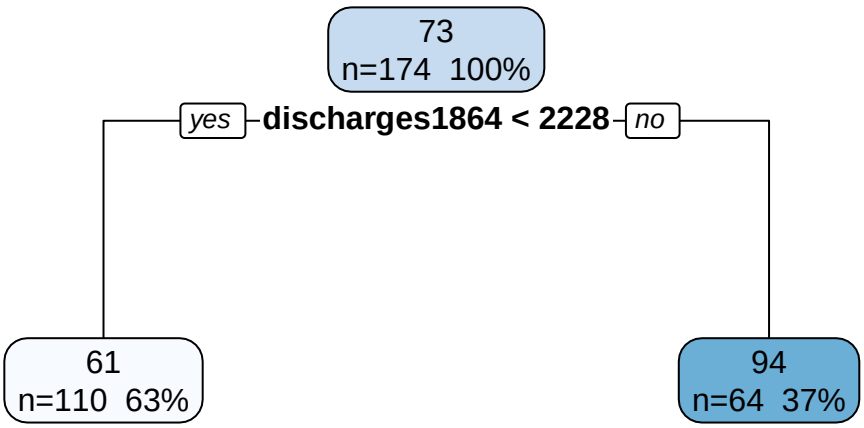

```
tree_cv = rpart(target ~ .-logtarget,data=train.df, method="anova",minbucket=5000,cp=0.1)
rpart.plot(tree_cv,digits=-2,extra=101)
```




```
tree_cv = rpart(target ~ .-logtarget,data=train.df, method="anova",minbucket=5000,cp=0.01)
rpart.plot(tree_cv,digits=-2,extra=101)

tree_cv = rpart(target ~ .-logtarget,data=train.df, method="anova",minbucket=5000,cp=0.001)
rpart.plot(tree_cv,digits=-2,extra=101)

tree_cv = rpart(target ~ .-logtarget,data=train.df, method="anova",minbucket=50,cp=0.001)
plotcp(tree_cv)
```




```
# plotcp will give us the relative error in the y axis for a 10-fold cross validation of our dataset, telling us the size of the tree

# The dotted line in the "plotcp" graph represents the minimum cross-validation error plus one standard deviation. One simple rule of

tree01 = rpart(target ~ .-logtarget,data=train.df, method="anova",minbucket=50,cp=0.01)
rpart.plot(tree01,digits=-2,extra=101)
```




```
treea = rpart(target ~ .-logtarget,data=train.df, method="anova",minbucket=50,cp=0.003)
rpart.plot(treea,digits=-2,extra=101)

treeb = rpart(target ~ .-logtarget,data=train.df, method="anova",minbucket=50,cp=0.00125)
rpart.plot(treeb,digits=-2,extra=101)
```

Which model is better? To get out of sample R square:

```
## Predictions for both models:
test.df$pred01 = predict(tree01, newdata= test.df)
test.df$preda = predict(treea, newdata= test.df)
test.df$predb = predict(treeb, newdata= test.df)
head(test.df)%>%relocate(preda,predb, pred01)
```

```
      preda    predb   pred01 facility_id type_of_organization
4  61.28182 61.28182 61.28182      291201                    2
5  94.00000 94.00000 94.00000      234101                    2
6  94.00000 94.00000 94.00000    50590101                    2
8  94.00000 94.00000 94.00000      311101                    4
9  94.00000 94.00000 94.00000      650401                    2
22 61.28182 61.28182 61.28182      429970                    2
   children_hospital hospital_ltc on_site_ltc privateroomexist
4                  1            0          NA                1
5                  0            0          NA                1
6                  0            0          NA                1
8                  0            0          NA                1
9                  0            0          NA                1
22                 1            0          NA                1
   semiprivateroomexist discharges1864 alcohol_drug_detox
```

```
4                     1            948                 3
5                     0           4028                 3
6                     0           2814                 3
8                     1           4984                 3
9                     1           2421                 1
22                    1             18                 3
   alcoholdetox_patient_days alcohol_drug_treat alcoholtreat_beds_lic
4                          0                  3                     0
5                          0                  3                     0
6                          0                  3                     0
8                          0                  3                     0
9                       4702                  1                     0
22                         0                  3                     0
   alcoholtreat_patient_days comprehensive_rehab comprehensive_rehab_beds_lic
4                          0                   1                          12
5                          0                   3                          67
6                          0                   3                           0
8                          0                   1                          20
9                       1771                   1                           0
22                         0                   3                           0
   Comprehensive_rehab_patient_days psych_0to17 psych_0to17_beds_lic
4                              3106           3                    0
5                             20813           3                    0
6                                 0           3                    0
8                              5935           3                    0
9                              2276           3                    0
22                                0           1                   74
   psych_0to17_patient_days psych_over17 psych_over17_beds_lic
4                         0            3                     0
5                         0            3                     0
6                         0            3                     0
8                         0            1                    37
9                         0            1                     0
22                    24576            1                     0
   psych_over17_patient_days detox clinpsyc clinic_psychiatric psychiatrists
4                          0     0        0                  2            47
5                          0     0        0                  1             2
6                          0     0        0                  3            35
8                       7542     0        1                  3            38
9                       9498     1        1                  3            11
22                       420     0        0                  1             4
   target logtarget
4     178  5.181784
```

```
5     178  5.181784
6     178  5.181784
8     178  5.181784
9     178  5.181784
22    178  5.181784
```

```
mean_train = mean(train.df$target)  #grab the mean for calc below

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

SSE01 = sum((test.df$target - test.df$pred01)^2)
SSEa = sum((test.df$target - test.df$preda)^2)
SSEb = sum((test.df$target - test.df$predb)^2)

SSE01
```

```
[1] 147207
```

```
SSEa
```

```
[1] 147207
```

```
SSEb
```

```
[1] 147207
```

```
print(paste("Tree CP=0.01 has a SSE of", SSE01))
```

```
[1] "Tree CP=0.01 has a SSE of 147206.990330579"
```

```
print(paste("Tree A has a SSE of", SSEa))
```

```
[1] "Tree A has a SSE of 147206.990330579"
```

```
print(paste("Tree B has a SSE of", SSEb))
```

```
[1] "Tree B has a SSE of 147206.990330579"
```

```
# And the total sum of squared errors (SST) using our simple benchmark model
# (the mean in the training set)

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

# With that, we finally get

OSR2.01 = 1 - SSE01/SST
OSR2a = 1 - SSEa/SST
OSR2b = 1 - SSEb/SST

OSR2.01
```

```
[1] 0.08797456
```

```
OSR2a
```

```
[1] 0.08797456
```

```
OSR2b
```

```
[1] 0.08797456
```

Let's see the MAE for comparisons:

```
MAE.01 = mean(abs(test.df$target - test.df$pred01))
MAEa = mean(abs(test.df$target - test.df$preda))
MAEb = mean(abs(test.df$target - test.df$predb))



MAE.01
```

```
[1] 48.31756
```

```
MAEa
```

```
[1] 48.31756
```

```
MAEb
```

```
[1] 48.31756
```

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,]



# df<-df%>%relocate(target)

summary(df)
```

```
  facility_id       type_of_organization children_hospital  hospital_ltc
 Min.   :    2984   Min.   :1.000        Min.   :0.0000    Min.   :0.0000
 1st Qu.:  141176   1st Qu.:2.000        1st Qu.:0.0000    1st Qu.:0.0000
 Median :  390628   Median :2.000        Median :0.0000    Median :0.0000
 Mean   : 8421653   Mean   :2.729        Mean   :0.0367    Mean   :0.1055
 3rd Qu.: 8402600   3rd Qu.:4.000        3rd Qu.:0.0000    3rd Qu.:0.0000
 Max.   :75140100   Max.   :7.000        Max.   :1.0000    Max.   :1.0000

  on_site_ltc    privateroomexist semiprivateroomexist discharges1864
 Min.   :0.000   Min.   :0.0000   Min.   :0.0000       Min.   :    0.0
 1st Qu.:1.000   1st Qu.:1.0000   1st Qu.:1.0000       1st Qu.:  244.8
 Median :1.000   Median :1.0000   Median :1.0000       Median : 1605.0
 Mean   :0.913   Mean   :0.8692   Mean   :0.7981       Mean   : 3115.4
 3rd Qu.:1.000   3rd Qu.:1.0000   3rd Qu.:1.0000       3rd Qu.: 3831.5
 Max.   :1.000   Max.   :1.0000   Max.   :1.0000       Max.   :27544.0
 NA's   :195     NA's   :4        NA's   :5
 alcohol_drug_detox alcoholdetox_patient_days alcohol_drug_treat
 Min.   :1.000      Min.   :   -1.0           Min.   :1.000
 1st Qu.:3.000      1st Qu.:    0.0           1st Qu.:3.000
 Median :3.000      Median :    0.0           Median :3.000
 Mean   :2.867      Mean   :  231.3           Mean   :2.936
```

```
3rd Qu.:3.000      3rd Qu.:    0.0           3rd Qu.:3.000
Max.   :3.000      Max.   :15468.0           Max.   :3.000

alcoholtreat_beds_lic alcoholtreat_patient_days comprehensive_rehab
Min.   : -1.000       Min.   :   -1.0           Min.   :1.000
1st Qu.:  0.000       1st Qu.:    0.0           1st Qu.:1.000
Median :  0.000       Median :    0.0           Median :3.000
Mean   :  2.558       Mean   :  476.6           Mean   :2.472
3rd Qu.:  0.000       3rd Qu.:    0.0           3rd Qu.:3.000
Max.   :220.000       Max.   :65169.0           Max.   :3.000
NA's   :1             NA's   :1
comprehensive_rehab_beds_lic Comprehensive_rehab_patient_days  psych_0to17
Min.   : -1.000              Min.   :   -1                     Min.   :1.000
1st Qu.:  0.000              1st Qu.:    0                     1st Qu.:3.000
Median :  0.000              Median :    0                     Median :3.000
Mean   :  9.866              Mean   : 2476                     Mean   :2.752
3rd Qu.:  9.000              3rd Qu.: 1734                     3rd Qu.:3.000
Max.   :187.000              Max.   :43378                     Max.   :3.000
NA's   :1                    NA's   :1
psych_0to17_beds_lic psych_0to17_patient_days  psych_over17
Min.   : -1.000      Min.   :   -1            Min.   :1.000
1st Qu.:  0.000      1st Qu.:    0            1st Qu.:1.000
Median :  0.000      Median :    0            Median :3.000
Mean   :  5.286      Mean   : 1400            Mean   :2.193
3rd Qu.:  0.000      3rd Qu.:    0            3rd Qu.:3.000
Max.   :186.000      Max.   :53196            Max.   :3.000
NA's   :1            NA's   :1
psych_over17_beds_lic psych_over17_patient_days     detox
Min.   : -1.00        Min.   :    -1            Min.   :0.0000
1st Qu.:  0.00        1st Qu.:     0            1st Qu.:0.0000
Median :  0.00        Median :     0            Median :0.0000
Mean   : 24.74        Mean   :  6484            Mean   :0.1101
3rd Qu.: 28.00        3rd Qu.:  5730            3rd Qu.:0.0000
Max.   :374.00        Max.   :126282            Max.   :1.0000

   clinpsyc      clinic_psychiatric psychiatrists        target
Min.   :0.0000   Min.   :1.00       Min.   :  0.00   Min.   :0.0000
1st Qu.:0.0000   1st Qu.:1.00       1st Qu.:  2.00   1st Qu.:0.0000
Median :0.0000   Median :1.00       Median :  7.00   Median :0.0000
Mean   :0.3578   Mean   :1.83       Mean   : 22.83   Mean   :0.2202
3rd Qu.:1.0000   3rd Qu.:3.00       3rd Qu.: 26.00   3rd Qu.:0.0000
Max.   :1.0000   Max.   :3.00       Max.   :221.00   Max.   :1.0000
```

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

```
  0   1
170  48
```

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

```
        0         1
0.7798165 0.2201835
```

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

```
 For the train dataset:
```

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

```
        0         1
0.7816092 0.2183908
```

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

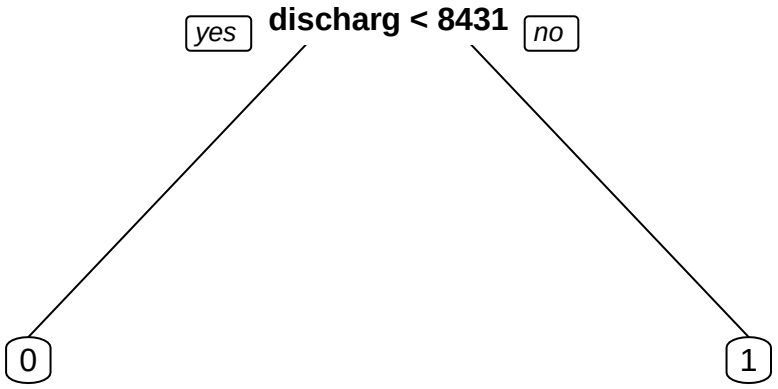
```
 For the test dataset:
```

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

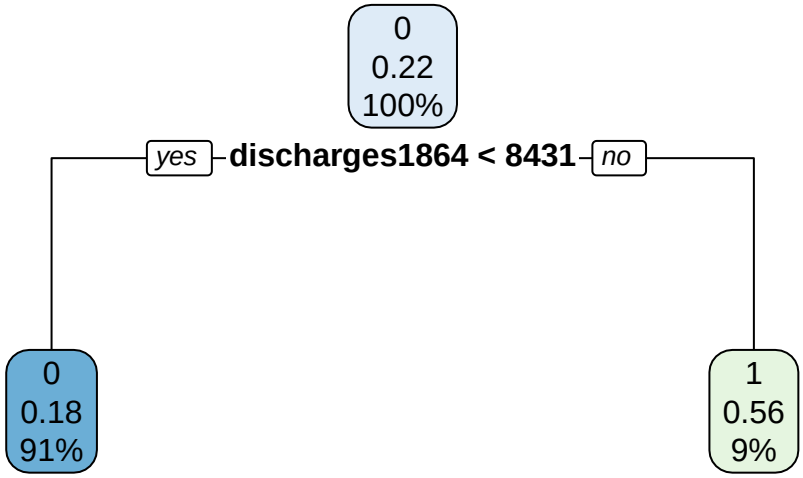
```
        0         1
0.7727273 0.2272727
```

First Classification Trees

```r
tree<-rpart(target ~ ., data=train.df, method="class",cp=0.05)
# simple graph:
# outcome is just the most common value (the mode)
prp(tree)
```



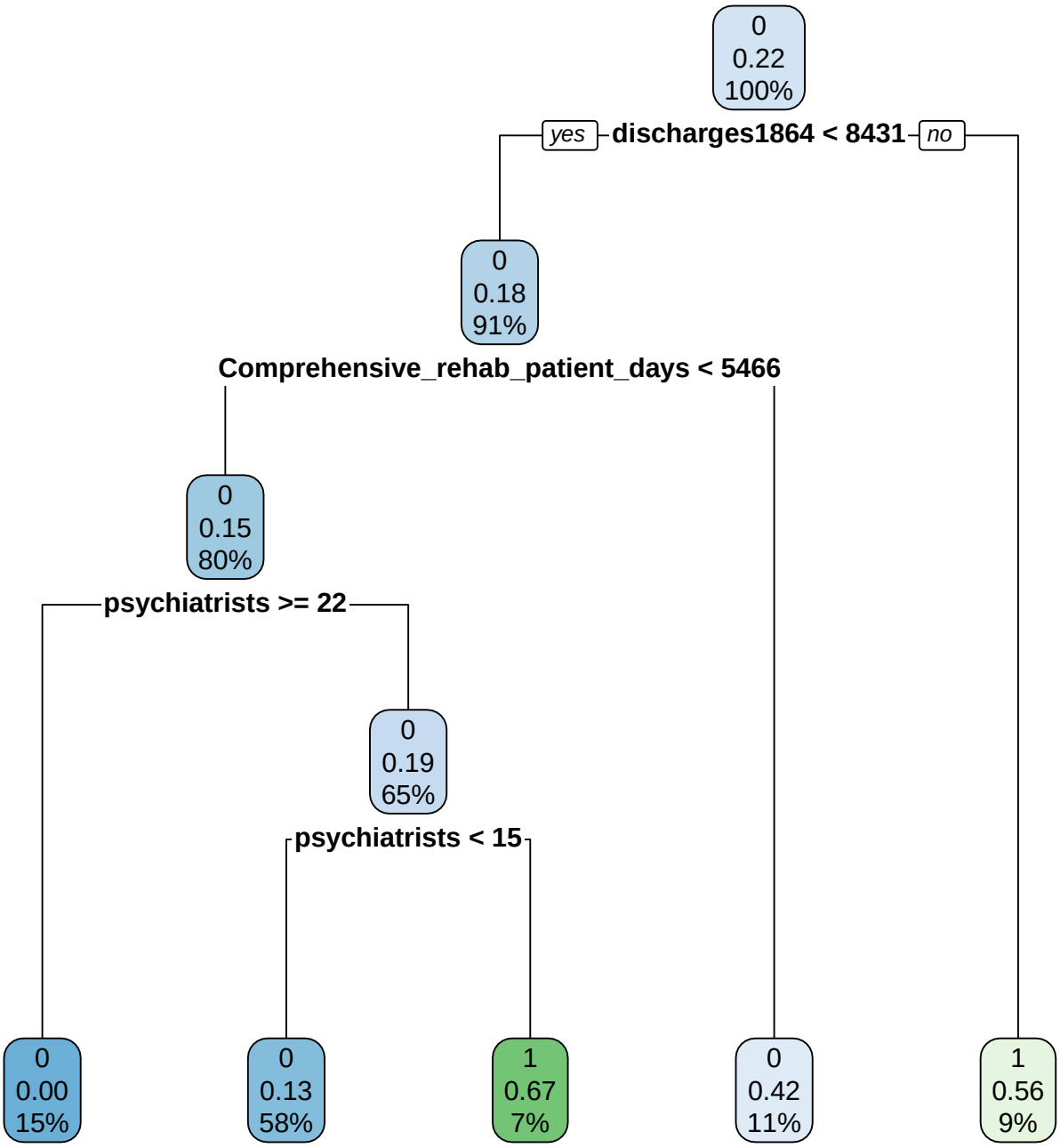

```r
# outcome by default is the most common value, the proportion of successes (variable==1) and the % of observations
rpart.plot(tree,digits=-2)
```



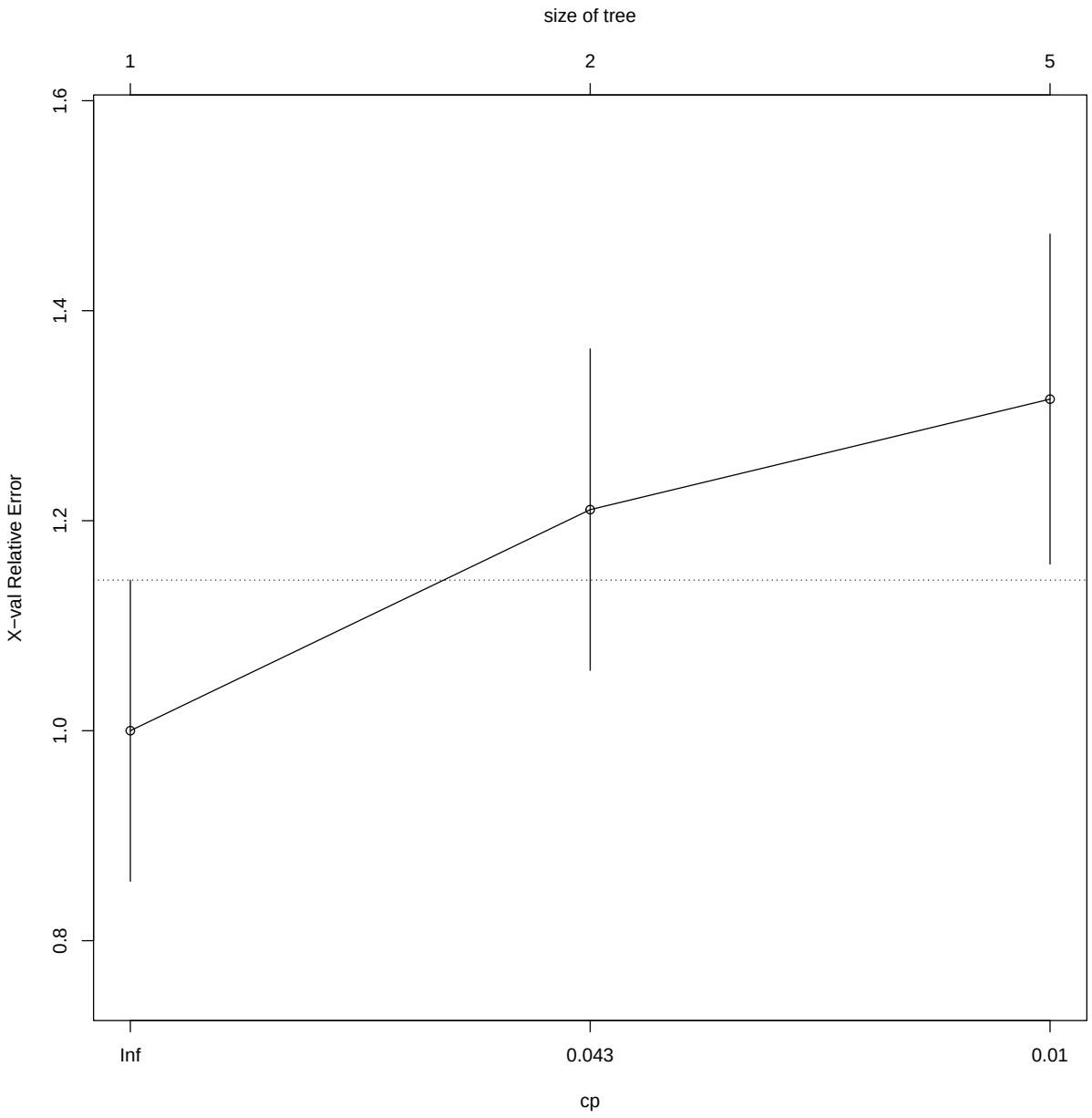

```r
tree<-rpart(target ~ .-target, data=train.df, method="class", cp=0.05)
rpart.plot(tree,digits=-2)
```

Cross-Validation

```
#pick a large enough minbucket you would trust the results and
#a cp value small enough we will get large trees for the cpplot

# how can we get some reference info to help us decide?

tree_cv = rpart(target ~ .-target,data=train.df, method="class",cp= 0.003 )
rpart.plot(tree_cv,digits=-2)
```

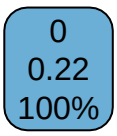



```
plotcp(tree_cv)
```



Let's see the larger trees. When do they stop making sense?

```
tree2 = rpart(target ~ .-target,data=train.df, method="class",minbucket=30,cp=.003)
rpart.plot(tree2,digits=-2)
```



```
tree3 = rpart(target ~ .-target,data=train.df, method="class",minbucket=30,cp=.0015)
rpart.plot(tree3,digits=-2)
```

Making predictions and comparing different trees

```
test.df$pred = predict(tree, newdata = test.df, type="class")
test.df$pred2 = predict(tree2, newdata = test.df, type="class")
test.df$pred3 = predict(tree3, newdata = test.df, type="class")
test.df<-test.df%>%relocate(target,pred,pred2,pred3)
head(test.df)
```

```
   target pred pred2 pred3 facility_id type_of_organization children_hospital
4       1    0     0     0      291201                    2                 1
5       1    0     0     0      234101                    2                 0
6       1    0     0     0    50590101                    2                 0
8       1    0     0     0      311101                    4                 0
9       1    0     0     0      650401                    2                 0
22      1    0     0     0      429970                    2                 1
   hospital_ltc on_site_ltc privateroomexist semiprivateroomexist
4             0          NA                1                    1
5             0          NA                1                    0
6             0          NA                1                    0
8             0          NA                1                    1
9             0          NA                1                    1
22            0          NA                1                    1
   discharges1864 alcohol_drug_detox alcoholdetox_patient_days
4             948                  3                         0
5            4028                  3                         0
6            2814                  3                         0
8            4984                  3                         0
9            2421                  1                      4702
22             18                  3                         0
   alcohol_drug_treat alcoholtreat_beds_lic alcoholtreat_patient_days
4                   3                     0                        0
5                   3                     0                        0
6                   3                     0                        0
8                   3                     0                        0
9                   1                     0                     1771
22                  3                     0                        0
   comprehensive_rehab comprehensive_rehab_beds_lic
```

```
4                   1                          12
5                   3                          67
6                   3                           0
8                   1                          20
9                   1                           0
22                  3                           0
   Comprehensive_rehab_patient_days psych_0to17 psych_0to17_beds_lic
4                              3106           3                    0
5                             20813           3                    0
6                                 0           3                    0
8                              5935           3                    0
9                              2276           3                    0
22                                0           1                   74
   psych_0to17_patient_days psych_over17 psych_over17_beds_lic
4                         0            3                     0
5                         0            3                     0
6                         0            3                     0
8                         0            1                    37
9                         0            1                     0
22                    24576            1                     0
   psych_over17_patient_days detox clinpsyc clinic_psychiatric psychiatrists
4                          0     0        0                  2            47
5                          0     0        0                  1             2
6                          0     0        0                  3            35
8                       7542     0        1                  3            38
9                       9498     1        1                  3            11
22                       420     0        0                  1             4
```

```
tail(test.df)
```

```
    target pred pred2 pred3 facility_id type_of_organization children_hospital
159      0    0     0     0    56100100                    7                 0
169      1    1     0     0      200701                    4                 0
180      1    0     0     0      195601                    4                 1
186      1    0     0     0    23830101                    2                 0
187      0    0     0     0       31801                    2                 0
216      0    0     0     0    22530101                    4                 0
    hospital_ltc on_site_ltc privateroomexist semiprivateroomexist
159            0          NA                0                    0
169            0          NA                1                    1
180            0          NA                1                    1
186            0          NA               NA                    1
```

```
187            1           1                0                    1
216            0          NA                1                    0
    discharges1864 alcohol_drug_detox alcoholdetox_patient_days
159             40                  3                         0
169          27544                  3                         0
180            320                  3                         0
186            202                  3                         0
187            334                  3                         0
216            303                  3                         0
    alcohol_drug_treat alcoholtreat_beds_lic alcoholtreat_patient_days
159                  3                     0                         0
169                  3                     0                         0
180                  3                     0                         0
186                  3                     0                         0
187                  3                     0                         0
216                  3                     0                         0
    comprehensive_rehab comprehensive_rehab_beds_lic
159                   3                           0
169                   3                           0
180                   3                           0
186                   3                           0
187                   3                           0
216                   3                           0
    Comprehensive_rehab_patient_days psych_0to17 psych_0to17_beds_lic
159                                0          3                    0
169                                0          3                    0
180                                0          3                    0
186                                0          3                    0
187                                0          3                    0
216                                0          3                    0
    psych_0to17_patient_days psych_over17 psych_over17_beds_lic
159                        0            1                   161
169                        0            1                   118
180                        0            3                     0
186                        0            3                    42
187                        0            3                     0
216                        0            3                     0
    psych_over17_patient_days detox clinpsyc clinic_psychiatric psychiatrists
159                     57844     0        0                  1             7
169                     41762     0        1                  3            49
180                         0     0        0                  3             7
186                     13803     0        1                  1             3
187                         0     0        0                  3             1
```

```
216                       0     0        0                 3             0
```

```
cat("1st tree (using target):")
```

```
1st tree (using target):
```

```
confusionMatrix(test.df$pred,as.factor(test.df$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
```

```
# Remember that this version of confusion matrix has "actual" in columns and
#    has the predicted in the rows.  Also we need to specify positive as 1
#    for the performance metrics (e.g., Sensitivity) to be what we expect

cat("2nd tree (using nine splits):")
```

```
2nd tree (using nine splits):
```

```
confusionMatrix(test.df$pred2,as.factor(test.df$target), positive="1")
```

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 34 10
         1  0  0

               Accuracy : 0.7727
                 95% CI : (0.6216, 0.8853)
    No Information Rate : 0.7727
    P-Value [Acc > NIR] : 0.583734

                  Kappa : 0

 Mcnemar's Test P-Value : 0.004427

            Sensitivity : 0.0000
            Specificity : 1.0000
         Pos Pred Value :    NaN
         Neg Pred Value : 0.7727
             Prevalence : 0.2273
         Detection Rate : 0.0000
   Detection Prevalence : 0.0000
      Balanced Accuracy : 0.5000

       'Positive' Class : 1
```

```
cat("3rd tree (using eleven splits):")
```

```
3rd tree (using eleven splits):
```

```
confusionMatrix(test.df$pred3,as.factor(test.df$target), positive="1")
```

```
Confusion Matrix and Statistics

          Reference
```

```
Prediction  0  1
         0 34 10
         1  0  0

               Accuracy : 0.7727
                 95% CI : (0.6216, 0.8853)
    No Information Rate : 0.7727
    P-Value [Acc > NIR] : 0.583734

                  Kappa : 0

 Mcnemar's Test P-Value : 0.004427

            Sensitivity : 0.0000
            Specificity : 1.0000
         Pos Pred Value :    NaN
         Neg Pred Value : 0.7727
             Prevalence : 0.2273
         Detection Rate : 0.0000
   Detection Prevalence : 0.0000
      Balanced Accuracy : 0.5000

       'Positive' Class : 1
```

```r
acc_tree <- sum(test.df$pred == test.df$target) / nrow(test.df)
acc_tree2 <- sum(test.df$pred2 == test.df$target) / nrow(test.df)
acc_tree3 <- sum(test.df$pred3 == test.df$target) / nrow(test.df)
print(paste("Accuracy for tree:", round(acc_tree * 100, 2), "%"))
```

```
[1] "Accuracy for tree: 72.73 %"
```

```r
print(paste("Accuracy for tree2:", round(acc_tree2 * 100, 2), "%"))
```

```
[1] "Accuracy for tree2: 77.27 %"
```

```r
print(paste("Accuracy for tree3:", round(acc_tree3 * 100, 2), "%"))
```

```
[1] "Accuracy for tree3: 77.27 %"
```

Compare Classification Tree with Logistic regressions

```
reg1<-glm(target~discharges1864+psych_over17_beds_lic+children_hospital+psych_over17,data=train.df,family='binomial')
summary(reg1)
```

```
Call:
glm(formula = target ~ discharges1864 + psych_over17_beds_lic +
    children_hospital + psych_over17, family = "binomial", data = train.df)

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)          -2.440e+00  6.917e-01  -3.528 0.000419 ***
discharges1864        1.232e-04  4.071e-05   3.027 0.002472 **
psych_over17_beds_lic  3.813e-03  4.391e-03   0.869 0.385094
children_hospital      2.050e+00  9.473e-01   2.164 0.030481 *
psych_over17           2.625e-01  2.458e-01   1.068 0.285544
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 182.65  on 173  degrees of freedom
Residual deviance: 168.75  on 169  degrees of freedom
AIC: 178.75

Number of Fisher Scoring iterations: 4
```

```
reg2<-glm(target~discharges1864+psych_over17_beds_lic+children_hospital+psych_over17+psych_over17_beds_lic+alcohol_drug_treat ,data=tr
summary(reg2)
```

```
Call:
glm(formula = target ~ discharges1864 + psych_over17_beds_lic +
    children_hospital + psych_over17 + psych_over17_beds_lic +
    alcohol_drug_treat, family = "binomial", data = train.df)

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)          -9.974e-01  1.381e+00  -0.722  0.47008
discharges1864        1.285e-04  4.128e-05   3.112  0.00186 **
```

```
psych_over17_beds_lic  4.135e-03  4.436e-03   0.932  0.35133
children_hospital      2.101e+00  9.501e-01   2.211  0.02704 *
psych_over17           3.023e-01  2.507e-01   1.206  0.22797
alcohol_drug_treat    -5.344e-01  4.552e-01  -1.174  0.24037
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 182.65  on 173  degrees of freedom
Residual deviance: 167.53  on 168  degrees of freedom
AIC: 179.53

Number of Fisher Scoring iterations: 4
```

Outcome is very different. We have to find a common metric. Common approaches are accuracy (in this case) or sensitivity or specificity, if they better fit your modeling needs.

```
test.df$predregprobs1 = predict(reg1, newdata = test.df, type="response")
test.df$predregprobs2 = predict(reg2, newdata = test.df, type="response")

test.df$predreg1<-ifelse(test.df$predregprobs1>0.5,1,0)
test.df$predreg2<-ifelse(test.df$predregprobs2>0.5,1,0)

acc_reg <- sum(test.df$predreg1 == test.df$target) / nrow(test.df)
acc_reg2 <- sum(test.df$predreg2 == test.df$target) / nrow(test.df)
print(paste("Accuracy for regression 1:", round(acc_reg * 100, 2), "%"))
```

```
[1] "Accuracy for regression 1: 84.09 %"
```

```
print(paste("Accuracy for regression 2:", round(acc_reg2 * 100, 2), "%"))
```

```
[1] "Accuracy for regression 2: 84.09 %"
```

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

```
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
```

```r
confusionMatrix(data=as.factor(test.df$predreg2),reference=as.factor(test.df$target), positive = "1")
```

```
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
```

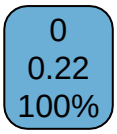
Loss Matrix

- To give a greater penalty to false negatives (predicting as negative when it is positive) we increase the second value.
- If you wanted to give a greater penalty to false positives you would have to increase the third value.

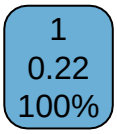
```
lossmatrix = matrix(c(0,5,1,0), byrow=FALSE, nrow=2)
lossmatrix
```

```
     [,1] [,2]
[1,]    0    1
[2,]    5    0
```

```
tree2new = rpart(target ~ .-target,data=train.df, method="class",minbucket=100,cp=0.0077)
rpart.plot(tree2new,digits=-2)
```




```
losstree2 = rpart(target ~ .-target,data=train.df, method="class",parms=list(loss=lossmatrix),minbucket=100,cp=0.0077)
rpart.plot(losstree2,digits=-2)
```



```
cat("Previous one:")
```

```
Previous one:
```

```
confusionMatrix(test.df$pred2,as.factor(test.df$target), positive="1")
```

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 34 10
         1  0  0

               Accuracy : 0.7727
                 95% CI : (0.6216, 0.8853)
    No Information Rate : 0.7727
    P-Value [Acc > NIR] : 0.583734

                  Kappa : 0

 Mcnemar's Test P-Value : 0.004427

            Sensitivity : 0.0000
            Specificity : 1.0000
         Pos Pred Value :    NaN
         Neg Pred Value : 0.7727
             Prevalence : 0.2273
         Detection Rate : 0.0000
   Detection Prevalence : 0.0000
      Balanced Accuracy : 0.5000

       'Positive' Class : 1
```

```
cat("New one:")
```

```
New one:
```

```
test.df$losspred2 = predict(losstree2, newdata = test.df, type="class")
confusionMatrix(test.df$losspred2,as.factor(test.df$target), positive="1")
```

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0  0  0
         1 34 10

               Accuracy : 0.2273
                 95% CI : (0.1147, 0.3784)
    No Information Rate : 0.7727
    P-Value [Acc > NIR] : 1

                  Kappa : 0

 Mcnemar's Test P-Value : 1.519e-08

            Sensitivity : 1.0000
            Specificity : 0.0000
         Pos Pred Value : 0.2273
         Neg Pred Value :    NaN
             Prevalence : 0.2273
         Detection Rate : 0.2273
   Detection Prevalence : 1.0000
      Balanced Accuracy : 0.5000

       '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