Assignment2- Markdown

Ameya Karnad

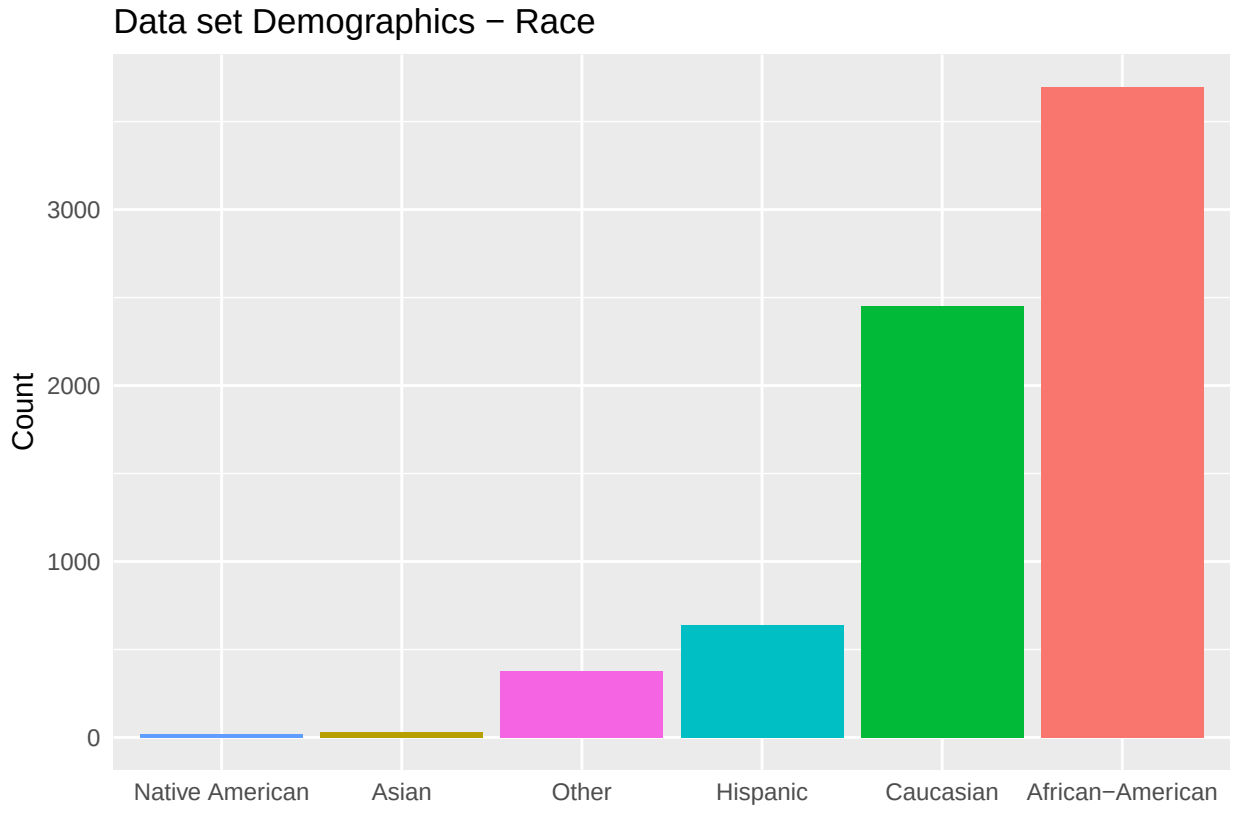
February 18, 2019

```
library(tidyverse)
library(data.table)
library(DT)

data <- fread(input = "D:/Columbia/Spring 2019/Data Science and Public Policy/Data Assignment 2/compas-a
```

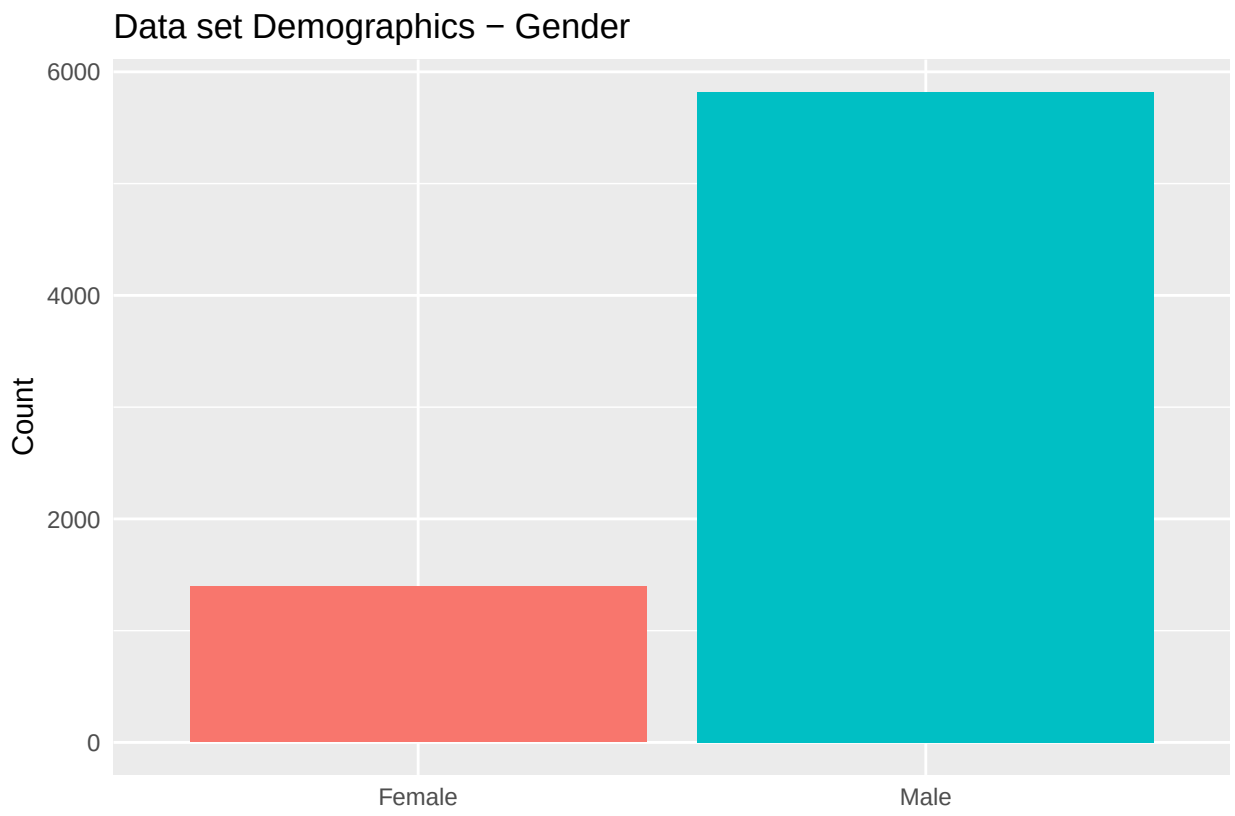
Including Plots

```
ggplot(data[, .(count = .N), race], aes(x = reorder(race, count), y=count, fill= race)) +
  geom_bar(stat = "identity") +
  theme(axis.ticks = element_blank(),plot.title = element_text(size = rel(1.2)),
        legend.position="none") +
  labs( x = "",y = "Count",
       title = "Data set Demographics - Race")
```

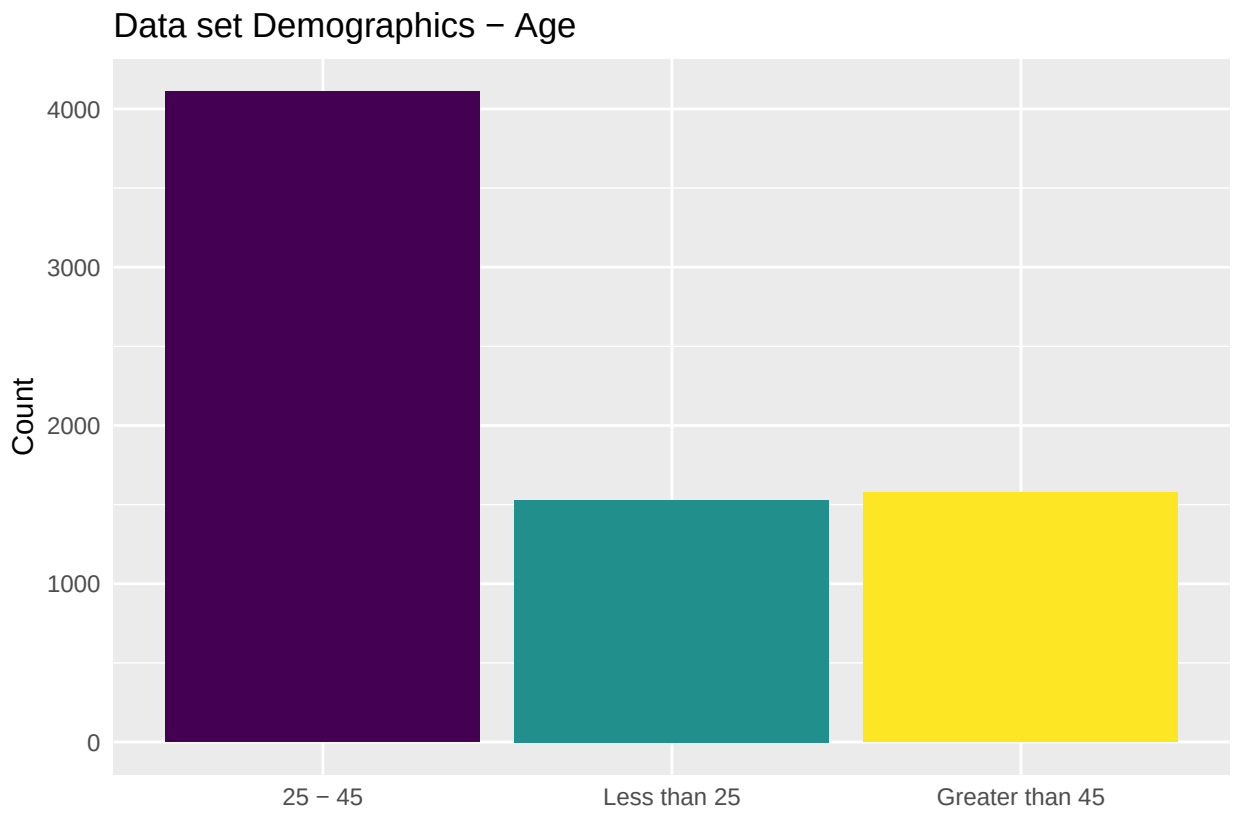



```
ggplot(data[, .(count = .N), sex], aes(x = reorder(sex, count), y=count, fill= sex)) +
  geom_bar(stat = "identity") +
  theme(axis.ticks = element_blank(),plot.title = element_text(size = rel(1.2)),
        legend.position="none") +
```

```
  labs( x = "",y = "Count",
        title = "Data set Demographics - Gender")
```




```
data$age_cat <- ordered(data$age_cat, levels = c( "25 - 45", "Less than 25", "Greater than 45"))
ggplot(data[, .(count = .N), age_cat], aes(x = age_cat, y=count, fill= age_cat)) +
  geom_bar(stat = "identity") +
  theme(axis.ticks = element_blank(),plot.title = element_text(size = rel(1.2)),
        legend.position="none") +
  labs( x = "",y = "Count",
        title = "Data set Demographics - Age")
```




```
data[score_text == "LOW"][.N]
```

```
## Empty data.table (0 rows) of 53 cols: id,name,first,last,compas_screening_date,sex...
```

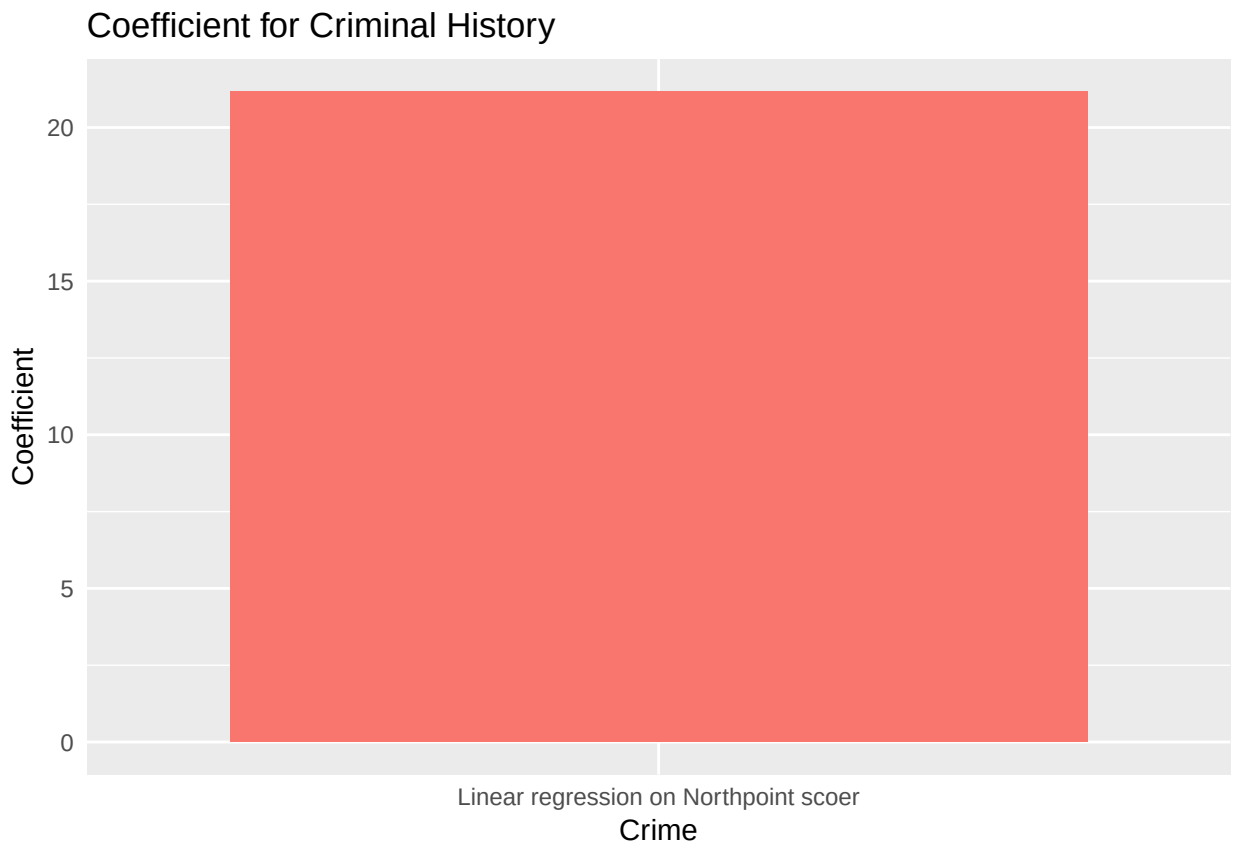
```
type = c('Correlation','Correlation', 'Coefficient', 'Coefficient')
gender = c('Male','Female', 'Male','Female')
value = c(-0.058, 0.058, 0.033, -0.033)

gender_score <- data.frame( "Analysis Type" = type, "Gender" = gender, "Value" = value)

ggplot(gender_score, aes(x = Analysis.Type, y = Value, fill = Gender)) +
  geom_bar(stat = "identity", position = "dodge") +
  theme(axis.ticks = element_blank(),plot.title = element_text(size = rel(1.2))) +
  labs(x = "Gender", y = "Score",
  title = "Correlation vs Regresion Coffiecient ",
  subtitle = "Corelation is not Causation")
```


```
Race = c('Linear regression on Northpoint scoer')
Score = c(5.86129229,  2.22553838,  3.4098577 ,  9.66977697)
Race_coeff <- data.frame( "Race" = Race, "Score" = Score)

ggplot(Race_coeff, aes(x = reorder(Race, desc(abs(Score))), y=Score, fill= Race)) +
  geom_bar(stat = "identity") +
  theme(axis.ticks = element_blank(),plot.title = element_text(size = rel(1.2)),
        legend.position="none") +
  labs( x = "Crime",y = "Coefficient",
        title = "Coefficient for Criminal History")
```



This is for Medium High - 1 and Low = 0

OVerall

```
print("Overall")
```

```
## [1] "Overall"
```

```
TP = data[is_recid == 1 & (score_text == "Medium" |score_text == "High"), .N]
FN = data[is_recid == 1 & score_text == "Low", .N ]
FP = data[is_recid == 0 & (score_text == "Medium" |score_text == "High"), .N]
TN = data[is_recid == 0 & score_text == "Low", .N]
names = c('Positive', 'Negative')
P_value = c(TP,FN)
N_value = c(FP,TN)
data.frame( "Type" = names, "Positive" = P_value, "Negative" = N_value)
```

```
##       Type Positive Negative
## 1 Positive     2140     1177
## 2 Negative     1331     2566
```

```
print(sprintf("Accuracy = %.2f",round((TP+TN)/(TP+TN+FP+FN),2) ))
```

```
## [1] "Accuracy = 0.65"
```

```
print(sprintf("False Discovery Rate = %.2f",round((FP)/(TP+FP),2) ))
```

```
## [1] "False Discovery Rate = 0.35"
```

```
print(sprintf("False Omission Rate = %.2f",round((FN)/(TN+FN),2) ))
```

```
## [1] "False Omission Rate = 0.34"
```

```
print(sprintf("False Positive Rate = %.2f",round((FP)/(TN+FP),2) ))
```

```
## [1] "False Positive Rate = 0.31"
```

African American

```
print("Race = African American")
```

```
## [1] "Race = African American"
```

```
aad =  data[race == "African-American"]

TP = aad[is_recid == 1 & (score_text == "Medium" |score_text == "High"), .N]
FN = aad[is_recid == 1 & score_text == "Low", .N ]
FP = aad[is_recid == 0 & (score_text == "Medium" |score_text == "High"), .N]
TN = aad[is_recid == 0 & score_text == "Low", .N]
names = c('Positive', 'Negative')
P_value = c(TP,FN)
N_value = c(FP,TN)
data.frame( "Type" = names, "Positive" = P_value, "Negative" = N_value)
```

```
##       Type Positive Negative
## 1 Positive     1445      729
## 2 Negative      591      931
```

```
print(sprintf("Accuracy = %.2f",round((TP+TN)/(TP+TN+FP+FN),2) ))
```

```
## [1] "Accuracy = 0.64"
```

```
print(sprintf("False Discovery Rate = %.2f",round((FP)/(TP+FP),2) ))
```

```
## [1] "False Discovery Rate = 0.34"
```

```
print(sprintf("False Omission Rate = %.2f",round((FN)/(TN+FN),2) ))
```

```
## [1] "False Omission Rate = 0.39"
```

```
print(sprintf("False Positive Rate = %.2f",round((FP)/(TN+FP),2) ))
```

```
## [1] "False Positive Rate = 0.44"
```

Caucasian

```
print("Race = Caucasian")
```

```
## [1] "Race = Caucasian"
```

```
cad =  data[race == "Caucasian"]
TP = cad[is_recid == 1 & (score_text == "Medium" |score_text == "High"), .N]
FN = cad[is_recid == 1 & score_text == "Low", .N ]
FP = cad[is_recid == 0 & (score_text == "Medium" |score_text == "High"), .N]
TN = cad[is_recid == 0 & score_text == "Low", .N]
names = c('Positive', 'Negative')
```

```
P_value = c(TP,FN)
N_value = c(FP,TN)
data.frame( "Type" = names, "Positive" = P_value, "Negative" = N_value)
```

```
##       Type Positive Negative
## 1 Positive      523      331
## 2 Negative      502     1098
```

```
print(sprintf("Accuracy = %.2f",round((TP+TN)/(TP+TN+FP+FN),2) ))
```

```
## [1] "Accuracy = 0.66"
```

```
print(sprintf("False Discovery Rate = %.2f",round((FP)/(TP+FP),2) ))
```

```
## [1] "False Discovery Rate = 0.39"
```

```
print(sprintf("False Omission Rate = %.2f",round((FN)/(TN+FN),2) ))
```

```
## [1] "False Omission Rate = 0.31"
```

```
print(sprintf("False Positive Rate = %.2f",round((FP)/(TN+FP),2) ))
```

```
## [1] "False Positive Rate = 0.23"
```

AGE CATEGORY

Less than 25

```
print("age_cat = Less than 25")
```

```
## [1] "age_cat = Less than 25"
```

```
low_age =  data[age_cat == "Less than 25"]
TP = low_age[is_recid == 1 & (score_text == "Medium" |score_text == "High"), .N]
FN = low_age[is_recid == 1 & score_text == "Low", .N ]
FP = low_age[is_recid == 0 & (score_text == "Medium" |score_text == "High"), .N]
TN = low_age[is_recid == 0 & score_text == "Low", .N]
names = c('Positive', 'Negative')
P_value = c(TP,FN)
N_value = c(FP,TN)
data.frame( "Type" = names, "Positive" = P_value, "Negative" = N_value)
```

```
##       Type Positive Negative
## 1 Positive      669      330
## 2 Negative      244      286
```

```
print(sprintf("Accuracy = %.2f",round((TP+TN)/(TP+TN+FP+FN),2) ))
```

```
## [1] "Accuracy = 0.62"
```

```
print(sprintf("False Discovery Rate = %.2f",round((FP)/(TP+FP),2) ))
```

```
## [1] "False Discovery Rate = 0.33"
```

```
print(sprintf("False Omission Rate = %.2f",round((FN)/(TN+FN),2) ))
```

```
## [1] "False Omission Rate = 0.46"
```

```
print(sprintf("False Positive Rate = %.2f",round((FP)/(TN+FP),2) ))
```

```
## [1] "False Positive Rate = 0.54"
```

25 - 45

```
print("age_cat = 25 - 45")
```

```
## [1] "age_cat = 25 - 45"
```

```
med_age =  data[age_cat == "25 - 45"]
TP = med_age[is_recid == 1 & (score_text == "Medium" |score_text == "High"), .N]
FN = med_age[is_recid == 1 & score_text == "Low", .N ]
FP = med_age[is_recid == 0 & (score_text == "Medium" |score_text == "High"), .N]
TN = med_age[is_recid == 0 & score_text == "Low", .N]
names = c('Positive', 'Negative')
P_value = c(TP,FN)
N_value = c(FP,TN)
data.frame( "Type" = names, "Positive" = P_value, "Negative" = N_value)
```

```
##       Type Positive Negative
## 1 Positive     1244      680
## 2 Negative      776     1409
```

```
print(sprintf("Accuracy = %.2f",round((TP+TN)/(TP+TN+FP+FN),2) ))
```

```
## [1] "Accuracy = 0.65"
```

```
print(sprintf("False Discovery Rate = %.2f",round((FP)/(TP+FP),2) ))
```

```
## [1] "False Discovery Rate = 0.35"
```

```
print(sprintf("False Omission Rate = %.2f",round((FN)/(TN+FN),2) ))
```

```
## [1] "False Omission Rate = 0.36"
```

```
print(sprintf("False Positive Rate = %.2f",round((FP)/(TN+FP),2) ))
```

```
## [1] "False Positive Rate = 0.33"
```

Greater than 45

```
print("age_cat = Greater than 45")
```

```
## [1] "age_cat = Greater than 45"
```

```
high_age =  data[age_cat == "Greater than 45"]
TP = high_age[is_recid == 1 & (score_text == "Medium" |score_text == "High"), .N]
FN = high_age[is_recid == 1 & score_text == "Low", .N ]
FP = high_age[is_recid == 0 & (score_text == "Medium" |score_text == "High"), .N]
TN = high_age[is_recid == 0 & score_text == "Low", .N]
names = c('Positive', 'Negative')
P_value = c(TP,FN)
N_value = c(FP,TN)
data.frame( "Type" = names, "Positive" = P_value, "Negative" = N_value)
```

```
##       Type Positive Negative
## 1 Positive      227      167
## 2 Negative      311      871
```

```
print(sprintf("Accuracy = %.2f",round((TP+TN)/(TP+TN+FP+FN),2) ))
```

```
## [1] "Accuracy = 0.70"
```

```
print(sprintf("False Discovery Rate = %.2f",round((FP)/(TP+FP),2) ))
```

```
## [1] "False Discovery Rate = 0.42"
```

```
print(sprintf("False Omission Rate = %.2f",round((FN)/(TN+FN),2) ))
```

```
## [1] "False Omission Rate = 0.26"
```

```
print(sprintf("False Positive Rate = %.2f",round((FP)/(TN+FP),2) ))
```

```
## [1] "False Positive Rate = 0.16"
```

Gender

MALE

```
print("sex = Male")
```

```
## [1] "sex = Male"
```

```
md =  data[sex == "Male"]
TP = md[is_recid == 1 & (score_text == "Medium" |score_text == "High"), .N]
FN = md[is_recid == 1 & score_text == "Low", .N ]
FP = md[is_recid == 0 & (score_text == "Medium" |score_text == "High"), .N]
TN = md[is_recid == 0 & score_text == "Low", .N]
names = c('Positive', 'Negative')
P_value = c(TP,FN)
N_value = c(FP,TN)
data.frame( "Type" = names, "Positive" = P_value, "Negative" = N_value)
```

```
##       Type Positive Negative
## 1 Positive     1821      905
## 2 Negative     1120     1973
```

```
print(sprintf("Accuracy = %.2f",round((TP+TN)/(TP+TN+FP+FN),2) ))
```

```
## [1] "Accuracy = 0.65"
```

```
print(sprintf("False Discovery Rate = %.2f",round((FP)/(TP+FP),2) ))
```

```
## [1] "False Discovery Rate = 0.33"
```

```
print(sprintf("False Omission Rate = %.2f",round((FN)/(TN+FN),2) ))
```

```
## [1] "False Omission Rate = 0.36"
```

```
print(sprintf("False Positive Rate = %.2f",round((FP)/(TN+FP),2) ))
```

```
## [1] "False Positive Rate = 0.31"
```

Female

```
print("sex = Female")
```

```
## [1] "sex = Female"
```

```
fd =  data[sex == "Female"]
TP = fd[is_recid == 1 & (score_text == "Medium" |score_text == "High"), .N]
FN = fd[is_recid == 1 & score_text == "Low", .N ]
FP = fd[is_recid == 0 & (score_text == "Medium" |score_text == "High"), .N]
```

```
TN = fd[is_recid == 0 & score_text == "Low", .N]
names = c('Positive', 'Negative')
P_value = c(TP,FN)
N_value = c(FP,TN)
data.frame( "Type" = names, "Positive" = P_value, "Negative" = N_value)
```

```
##       Type Positive Negative
## 1 Positive      319      272
## 2 Negative      211      593
```

```
print(sprintf("Accuracy = %.2f",round((TP+TN)/(TP+TN+FP+FN),2) ))
```

```
## [1] "Accuracy = 0.65"
```

```
print(sprintf("False Discovery Rate = %.2f",round((FP)/(TP+FP),2) ))
```

```
## [1] "False Discovery Rate = 0.46"
```

```
print(sprintf("False Omission Rate = %.2f",round((FN)/(TN+FN),2) ))
```

```
## [1] "False Omission Rate = 0.26"
```

```
print(sprintf("False Positive Rate = %.2f",round((FP)/(TN+FP),2) ))
```

```
## [1] "False Positive Rate = 0.31"
```

This is for High - 1 and Low, Medium = 0

OVerall

```
print("Overall")
```

```
## [1] "Overall"
```

```
TP = data[is_recid == 1 & score_text == "High", .N]
FN = data[is_recid == 1 & (score_text == "Medium" |score_text == "Low"), .N ]
FP = data[is_recid == 0 & score_text == "High", .N]
TN = data[is_recid == 0 & (score_text == "Medium" |score_text == "Low"), .N]
names = c('Positive', 'Negative')
P_value = c(TP,FN)
N_value = c(FP,TN)
data.frame( "Type" = names, "Positive" = P_value, "Negative" = N_value)
```

```
##       Type Positive Negative
## 1 Positive     1041      362
## 2 Negative     2430     3381
```

```
print(sprintf("Accuracy = %.2f",round((TP+TN)/(TP+TN+FP+FN),2) ))
```

```
## [1] "Accuracy = 0.61"
```

```
print(sprintf("False Discovery Rate = %.2f",round((FP)/(TP+FP),2) ))
```

```
## [1] "False Discovery Rate = 0.26"
```

```
print(sprintf("False Omission Rate = %.2f",round((FN)/(TN+FN),2) ))
```

```
## [1] "False Omission Rate = 0.42"
```

```
print(sprintf("False Positive Rate = %.2f",round((FP)/(TN+FP),2) ))
```

```
## [1] "False Positive Rate = 0.10"
```

African-American

```
print("Race = African American")
```

```
## [1] "Race = African American"
```

```
aad =  data[race == "African-American"]

TP = aad[is_recid == 1 & score_text == "High", .N]
FN = aad[is_recid == 1 & (score_text == "Medium" |score_text == "Low"), .N ]
FP = aad[is_recid == 0 & score_text == "High", .N]
TN = aad[is_recid == 0 & (score_text == "Medium" |score_text == "Low"), .N]
names = c('Positive', 'Negative')
P_value = c(TP,FN)
N_value = c(FP,TN)
data.frame( "Type" = names, "Positive" = P_value, "Negative" = N_value)
```

```
##       Type Positive Negative
## 1 Positive      771      254
## 2 Negative     1265     1406
```

```
print(sprintf("Accuracy = %.2f",round((TP+TN)/(TP+TN+FP+FN),2) ))
```

```
## [1] "Accuracy = 0.59"
```

```
print(sprintf("False Discovery Rate = %.2f",round((FP)/(TP+FP),2) ))
```

```
## [1] "False Discovery Rate = 0.25"
```

```
print(sprintf("False Omission Rate = %.2f",round((FN)/(TN+FN),2) ))
```

```
## [1] "False Omission Rate = 0.47"
```

```
print(sprintf("False Positive Rate = %.2f",round((FP)/(TN+FP),2) ))
```

```
## [1] "False Positive Rate = 0.15"
```

Caucasian

```
print("Race = Caucasian")
```

```
## [1] "Race = Caucasian"
```

```
cad =  data[race == "Caucasian"]
TP = cad[is_recid == 1 & score_text == "High", .N]
FN = cad[is_recid == 1 & (score_text == "Medium" |score_text == "Low"), .N ]
FP = cad[is_recid == 0 & score_text == "High", .N]
TN = cad[is_recid == 0 & (score_text == "Medium" |score_text == "Low"), .N]
names = c('Positive', 'Negative')
P_value = c(TP,FN)
N_value = c(FP,TN)
data.frame( "Type" = names, "Positive" = P_value, "Negative" = N_value)
```

```
##       Type Positive Negative
```

```
## 1 Positive      201       75
## 2 Negative      824     1354
```

```
print(sprintf("Accuracy = %.2f",round((TP+TN)/(TP+TN+FP+FN),2) ))
```

```
## [1] "Accuracy = 0.63"
```

```
print(sprintf("False Discovery Rate = %.2f",round((FP)/(TP+FP),2) ))
```

```
## [1] "False Discovery Rate = 0.27"
```

```
print(sprintf("False Omission Rate = %.2f",round((FN)/(TN+FN),2) ))
```

```
## [1] "False Omission Rate = 0.38"
```

```
print(sprintf("False Positive Rate = %.2f",round((FP)/(TN+FP),2) ))
```

```
## [1] "False Positive Rate = 0.05"
```

AGE

Less than 25

```
print("age_cat = Less than 25")
```

```
## [1] "age_cat = Less than 25"
```

```
low_age =  data[age_cat == "Less than 25"]
TP = low_age[is_recid == 1 & score_text == "High", .N]
FN = low_age[is_recid == 1 & (score_text == "Medium" |score_text == "Low"), .N ]
FP = low_age[is_recid == 0 & score_text == "High", .N]
TN = low_age[is_recid == 0 & (score_text == "Medium" |score_text == "Low"), .N]
names = c('Positive', 'Negative')
P_value = c(TP,FN)
N_value = c(FP,TN)
data.frame( "Type" = names, "Positive" = P_value, "Negative" = N_value)
```

```
##       Type Positive Negative
## 1 Positive      343      110
## 2 Negative      570      506
```

```
print(sprintf("Accuracy = %.2f",round((TP+TN)/(TP+TN+FP+FN),2) ))
```

```
## [1] "Accuracy = 0.56"
```

```
print(sprintf("False Discovery Rate = %.2f",round((FP)/(TP+FP),2) ))
```

```
## [1] "False Discovery Rate = 0.24"
```

```
print(sprintf("False Omission Rate = %.2f",round((FN)/(TN+FN),2) ))
```

```
## [1] "False Omission Rate = 0.53"
```

```
print(sprintf("False Positive Rate = %.2f",round((FP)/(TN+FP),2) ))
```

```
## [1] "False Positive Rate = 0.18"
```

25 - 45

```
print("age_cat = 25 - 45")
```

```
## [1] "age_cat = 25 - 45"
```

```
mid_age =  data[age_cat == "25 - 45"]
TP = mid_age[is_recid == 1 & score_text == "High", .N]
FN = mid_age[is_recid == 1 & (score_text == "Medium" |score_text == "Low"), .N ]
FP = mid_age[is_recid == 0 & score_text == "High", .N]
TN = mid_age[is_recid == 0 & (score_text == "Medium" |score_text == "Low"), .N]
names = c('Positive', 'Negative')
P_value = c(TP,FN)
N_value = c(FP,TN)
data.frame( "Type" = names, "Positive" = P_value, "Negative" = N_value)
```

```
##       Type Positive Negative
## 1 Positive      615      207
## 2 Negative     1405     1882
```

```
print(sprintf("Accuracy = %.2f",round((TP+TN)/(TP+TN+FP+FN),2) ))
```

```
## [1] "Accuracy = 0.61"
```

```
print(sprintf("False Discovery Rate = %.2f",round((FP)/(TP+FP),2) ))
```

```
## [1] "False Discovery Rate = 0.25"
```

```
print(sprintf("False Omission Rate = %.2f",round((FN)/(TN+FN),2) ))
```

```
## [1] "False Omission Rate = 0.43"
```

```
print(sprintf("False Positive Rate = %.2f",round((FP)/(TN+FP),2) ))
```

```
## [1] "False Positive Rate = 0.10"
```

Greater than 45

```
print("age_cat = Greater than 45")
```

```
## [1] "age_cat = Greater than 45"
```

```
high_age =  data[age_cat == "Greater than 45"]
TP = high_age[is_recid == 1 & score_text == "High", .N]
FN = high_age[is_recid == 1 & (score_text == "Medium" |score_text == "Low"), .N ]
FP = high_age[is_recid == 0 & score_text == "High", .N]
TN = high_age[is_recid == 0 & (score_text == "Medium" |score_text == "Low"), .N]
names = c('Positive', 'Negative')
P_value = c(TP,FN)
N_value = c(FP,TN)
data.frame( "Type" = names, "Positive" = P_value, "Negative" = N_value)
```

```
##       Type Positive Negative
## 1 Positive       83       45
## 2 Negative      455      993
```

```
print(sprintf("Accuracy = %.2f",round((TP+TN)/(TP+TN+FP+FN),2) ))
```

```
## [1] "Accuracy = 0.68"
```

```
print(sprintf("False Discovery Rate = %.2f",round((FP)/(TP+FP),2) ))
```

```
## [1] "False Discovery Rate = 0.35"
```

```
print(sprintf("False Omission Rate = %.2f",round((FN)/(TN+FN),2) ))
```

```
## [1] "False Omission Rate = 0.31"
```

```
print(sprintf("False Positive Rate = %.2f",round((FP)/(TN+FP),2) ))
```

```
## [1] "False Positive Rate = 0.04"
```

Gender

MALE

```
print("sex = Male")
```

```
## [1] "sex = Male"
```

```
md =  data[sex == "Male"]
TP = md[is_recid == 1 & score_text == "High", .N]
FN = md[is_recid == 1 & (score_text == "Medium" |score_text == "Low"), .N ]
FP = md[is_recid == 0 & score_text == "High", .N]
TN = md[is_recid == 0 & (score_text == "Medium" |score_text == "Low"), .N]
names = c('Positive', 'Negative')
P_value = c(TP,FN)
N_value = c(FP,TN)
data.frame( "Type" = names, "Positive" = P_value, "Negative" = N_value)
```

```
##       Type Positive Negative
## 1 Positive      921      292
## 2 Negative     2020     2586
```

```
print(sprintf("Accuracy = %.2f",round((TP+TN)/(TP+TN+FP+FN),2) ))
```

```
## [1] "Accuracy = 0.60"
```

```
print(sprintf("False Discovery Rate = %.2f",round((FP)/(TP+FP),2) ))
```

```
## [1] "False Discovery Rate = 0.24"
```

```
print(sprintf("False Omission Rate = %.2f",round((FN)/(TN+FN),2) ))
```

```
## [1] "False Omission Rate = 0.44"
```

```
print(sprintf("False Positive Rate = %.2f",round((FP)/(TN+FP),2) ))
```

```
## [1] "False Positive Rate = 0.10"
```

Female

```
print("sex = Female")
```

```
## [1] "sex = Female"
```

```
fd =  data[sex == "Female"]
TP = fd[is_recid == 1 & score_text == "High", .N]
FN = fd[is_recid == 1 & (score_text == "Medium" |score_text == "Low"), .N ]
FP = fd[is_recid == 0 & score_text == "High", .N]
TN = fd[is_recid == 0 & (score_text == "Medium" |score_text == "Low"), .N]
names = c('Positive', 'Negative')
P_value = c(TP,FN)
```

```r
N_value = c(FP,TN)
data.frame( "Type" = names, "Positive" = P_value, "Negative" = N_value)
```

```
##       Type Positive Negative
## 1 Positive      120       70
## 2 Negative      410      795
```

```r
print(sprintf("Accuracy = %.2f",round((TP+TN)/(TP+TN+FP+FN),2) ))
```

```
## [1] "Accuracy = 0.66"
```

```r
print(sprintf("False Discovery Rate = %.2f",round((FP)/(TP+FP),2) ))
```

```
## [1] "False Discovery Rate = 0.37"
```

```r
print(sprintf("False Omission Rate = %.2f",round((FN)/(TN+FN),2) ))
```

```
## [1] "False Omission Rate = 0.34"
```

```r
print(sprintf("False Positive Rate = %.2f",round((FP)/(TN+FP),2) ))
```

```
## [1] "False Positive Rate = 0.08"
```