EDA.Rmd

Read in the data.

```
df = read.csv("data.csv")
```

Check variables for type issues etc.

```
str(df)
```

```
## 'data.frame':    26020 obs. of  17 variables:
##  $ Fiscal.Year.Released      : int  2010 2010 2010 2010 2010 2010 2010 2010 2010 2010 ...
##  $ Recidivism.Reporting.Year : int  2013 2013 2013 2013 2013 2013 2013 2013 2013 2013 ...
##  $ Main.Supervising.District : Factor w/ 11 levels "","1JD","2JD",..: 8 1 6 7 1 5 5 7 8 2 ...
##  $ Release.Type              : Factor w/ 13 levels "","Discharged - Expiration of Sentence",..: 5 3 5
##  $ Race...Ethnicity          : Factor w/ 12 levels "","American Indian or Alaska Native - Hispanic",
##  $ Age.At.Release            : Factor w/ 6 levels "","25-34","35-44",..: 2 2 3 2 3 2 2 3 2 2 ...
##  $ Sex                       : Factor w/ 3 levels "","Female","Male": 3 3 3 3 3 3 3 3 3 3 ...
##  $ Offense.Classification    : Factor w/ 15 levels "A Felony","Aggravated Misdemeanor",..: 4 5 3 3 5
##  $ Offense.Type              : Factor w/ 5 levels "Drug","Other",..: 5 3 1 2 5 1 1 4 3 5 ...
##  $ Offense.Subtype           : Factor w/ 26 levels "Alcohol","Animals",..: 17 22 24 11 4 24 24 15 8 1
##  $ Return.to.Prison          : Factor w/ 2 levels "No","Yes": 2 2 2 1 2 1 1 2 1 2 ...
##  $ Days.to.Return            : int  433 453 832 NA 116 NA NA 84 NA 274 ...
##  $ Recidivism.Type           : Factor w/ 3 levels "New","No Recidivism",..: 1 3 3 2 3 2 2 3 2 3 ...
##  $ New.Offense.Classification: Factor w/ 16 levels "","A Felony",..: 5 1 1 1 1 1 1 1 1 1 ...
##  $ New.Offense.Type          : Factor w/ 11 levels "","Assault","Drug",..: 3 1 1 1 1 1 1 1 1 1 ...
##  $ New.Offense.Sub.Type      : Factor w/ 26 levels "","Alcohol","Animals",..: 24 1 1 1 1 1 1 1 1 1 .
##  $ Target.Population         : Factor w/ 2 levels "No","Yes": 2 1 2 2 1 2 2 2 2 2 ...
```

Let's Return.to.Prison and Target.Population boolean

```
df$Return.to.Prison = as.character(df$Return.to.Prison)
df$Return.to.Prison[df$Return.to.Prison == "Yes"] = "TRUE"
df$Return.to.Prison[df$Return.to.Prison == "No"] = "FALSE"
df$Return.to.Prison = as.logical(df$Return.to.Prison)

df$Target.Population = as.character(df$Target.Population)
df$Target.Population[df$Target.Population == "Yes"] = "TRUE"
df$Target.Population[df$Target.Population == "No"] = "FALSE"
df$Target.Population = as.logical(df$Target.Population)

str(df)
```

```
## 'data.frame':    26020 obs. of  17 variables:
##  $ Fiscal.Year.Released      : int  2010 2010 2010 2010 2010 2010 2010 2010 2010 2010 ...
##  $ Recidivism.Reporting.Year : int  2013 2013 2013 2013 2013 2013 2013 2013 2013 2013 ...
##  $ Main.Supervising.District : Factor w/ 11 levels "","1JD","2JD",..: 8 1 6 7 1 5 5 7 8 2 ...
##  $ Release.Type              : Factor w/ 13 levels "","Discharged - Expiration of Sentence",..: 5 3 5
##  $ Race...Ethnicity          : Factor w/ 12 levels "","American Indian or Alaska Native - Hispanic",
##  $ Age.At.Release            : Factor w/ 6 levels "","25-34","35-44",..: 2 2 3 2 3 2 2 3 2 2 ...
##  $ Sex                       : Factor w/ 3 levels "","Female","Male": 3 3 3 3 3 3 3 3 3 3 ...
```

```
##  $ Offense.Classification    : Factor w/ 15 levels "A Felony","Aggravated Misdemeanor",..: 4 5 3 3 5
##  $ Offense.Type              : Factor w/ 5 levels "Drug","Other",..: 5 3 1 2 5 1 1 4 3 5 ...
##  $ Offense.Subtype           : Factor w/ 26 levels "Alcohol","Animals",..: 17 22 24 11 4 24 24 15 8 1
##  $ Return.to.Prison          : logi  TRUE TRUE TRUE FALSE TRUE FALSE ...
##  $ Days.to.Return            : int  433 453 832 NA 116 NA NA 84 NA 274 ...
##  $ Recidivism.Type           : Factor w/ 3 levels "New","No Recidivism",..: 1 3 3 2 3 2 2 3 2 3 ...
##  $ New.Offense.Classification: Factor w/ 16 levels "","A Felony",..: 5 1 1 1 1 1 1 1 1 1 ...
##  $ New.Offense.Type          : Factor w/ 11 levels "","Assault","Drug",..: 3 1 1 1 1 1 1 1 1 1 ...
##  $ New.Offense.Sub.Type      : Factor w/ 26 levels "","Alcohol","Animals",..: 24 1 1 1 1 1 1 1 1 1 .
##  $ Target.Population         : logi  TRUE FALSE TRUE TRUE FALSE TRUE ...
```

Let's get rid of NAs.

```
# find all columns with NA using an apply of anyNA on each column
colnames(df)[apply(df, 2, anyNA)]
```

```
## [1] "Days.to.Return"
```

Just Days.to.Return it seems. These are presumably those who did not reoffend, so these are valid values, so we will leave it as is. Let's see how severe it is.

```
cat("Total NA rows:", sum(is.na(df$Days.to.Return)),"\n")
```

```
## Total NA rows: 17339
```

```
cat("Rows total:", nrow(df), "\n")
```

```
## Rows total: 26020
```

```
cat("Proportion of rows NA:", sum(is.na(df$Days.to.Return)) / nrow(df))
```

```
## Proportion of rows NA: 0.666372
```

So around 2/3 of released prisoners did not return to prison, assuming NAs are not just missing values. Given the cleanliness of the dataset, this likely is a good assumption. We can check by seeing if these NAs are the same rows as FALSE values for Return.to.Prison:

```
# we can see if there any NAs that aren't matched with those that did not return to prison
length(df$Days.to.Return[(is.na(df$Days.to.Return)) & (df$Return.to.Prison == TRUE)])
```

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

Since there are no instances of a return to prison with an NA days to return, we can presume all of the NAs are valid.

Now we can look at the values of some of our most relevant factor variables.

```
levels(df$Offense.Classification)
```

```
##  [1] "A Felony"
##  [2] "Aggravated Misdemeanor"
##  [3] "B Felony"
##  [4] "C Felony"
##  [5] "D Felony"
##  [6] "Felony - Enhanced"
##  [7] "Felony - Enhancement to Original Penalty"
##  [8] "Felony - Mandatory Minimum"
##  [9] "Other Felony"
## [10] "Other Felony (Old Code)"
## [11] "Other Misdemeanor"
## [12] "Serious Misdemeanor"
## [13] "Sexual Predator Community Supervision"
## [14] "Simple Misdemeanor"
## [15] "Special Sentence 2005"
```

```
levels(df$Offense.Type)
```

```
## [1] "Drug"         "Other"        "Property"     "Public Order"
## [5] "Violent"
```

```
levels(df$Offense.Subtype)
```

```
##  [1] "Alcohol"                          "Animals"
##  [3] "Arson"                            "Assault"
##  [5] "Burglary"                         "Drug Possession"
##  [7] "Flight/Escape"                    "Forgery/Fraud"
##  [9] "Kidnap"                           "Murder/Manslaughter"
## [11] "Other Criminal"                   "Other Drug"
## [13] "Other Public Order"               "Other Violent"
## [15] "OWI"                              "Prostitution/Pimping"
## [17] "Robbery"                          "Sex"
## [19] "Sex Offender Registry/Residency" "Special Sentence Revocation"
## [21] "Stolen Property"                  "Theft"
## [23] "Traffic"                          "Trafficking"
## [25] "Vandalism"                        "Weapons"
```

We will likley use dummy variables if we use these for regression, and we likely will considering their potential usefulness in predicting recidivism, new offense, days until return, etc.

Let's see if offense types and subtypes differ between new and original offenses.

```
levels(df$New.Offense.Type)
```

```
##  [1] ""                "Assault"         "Drug"
##  [4] "Drug Possession" "Flight/Escape"   "Other"
##  [7] "Property"        "Public Order"    "Sex"
## [10] "Traffic"         "Violent"
```

So New.Offense.Type adds Assault, Drug Possession, Flight/Escape, Traffic, and Sex. These are all categories previously contained in offense subtype.

```
levels(df$New.Offense.Sub.Type)
```

```
##  [1] ""                      "Alcohol"              "Animals"
##  [4] "Arson"                 "Assault"              "Burglary"
##  [7] "Drug Possession"       "Flight/Escape"        "Forgery/Fraud"
## [10] "Kidnap"                "Murder/Manslaughter"  "Other Criminal"
## [13] "Other Drug"            "Other Property"       "Other Public Order"
## [16] "Other Violent"         "OWI"                  "Prostitution/Pimping"
## [19] "Robbery"               "Sex"                  "Stolen Property"
## [22] "Theft"                 "Traffic"              "Trafficking"
## [25] "Vandalism"             "Weapons"
```

The new offense type categories are still within subtype, though there are some differences between the subtype factors. Now, "Special sentence Revocation" is gone and we have a new blank factor, for example. Also note that new offense type has a blank factor. Maybe these are markers that the new offense matched the old offense type and/or subtype. Let's see if old and new types match otherwise:

```
sum(as.character(df$Offense.Type) == as.character(df$New.Offense.Type))
```

```
## [1] 3994
```

So, we have around 4000 indices where offense type and new offensive type match. Let's see what the values are. It is then unclear what the blank factors mean.

```
length(df$New.Offense.Type[(df$New.Offense.Type == "") | (df$New.Offense.Sub.Type == "")])
```

```
## [1] 19321
```

So, these occur very often in the data set. It is still possible that these indicate a match of original and new types, but it may warrent further investigation. They are often likely released prisoners who never reoffended, similar to NAs in Days.to.Return.

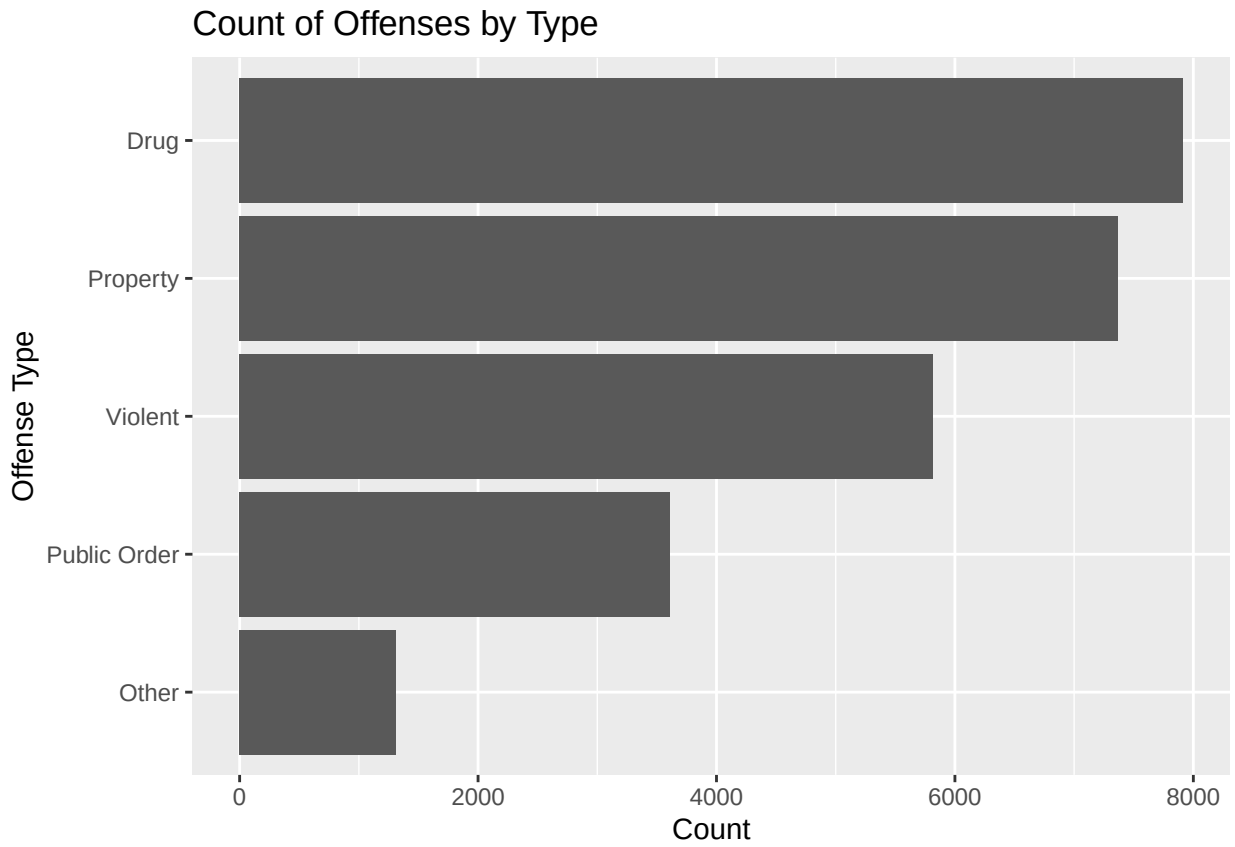
Let's reorder variables by frequency and plot their distributions.

```
library(ggplot2)

df$Offense.Type = reorder(df$Offense.Type, df$Offense.Type, length)
df$Offense.Subtype = reorder(df$Offense.Subtype, df$Offense.Subtype, length)
df$New.Offense.Type = reorder(df$New.Offense.Type, df$New.Offense.Type, length)
df$New.Offense.Sub.Type = reorder(df$New.Offense.Sub.Type, df$New.Offense.Sub.Type, length)
df$Age.At.Release = reorder(df$Age.At.Release, df$Age.At.Release, length)
```

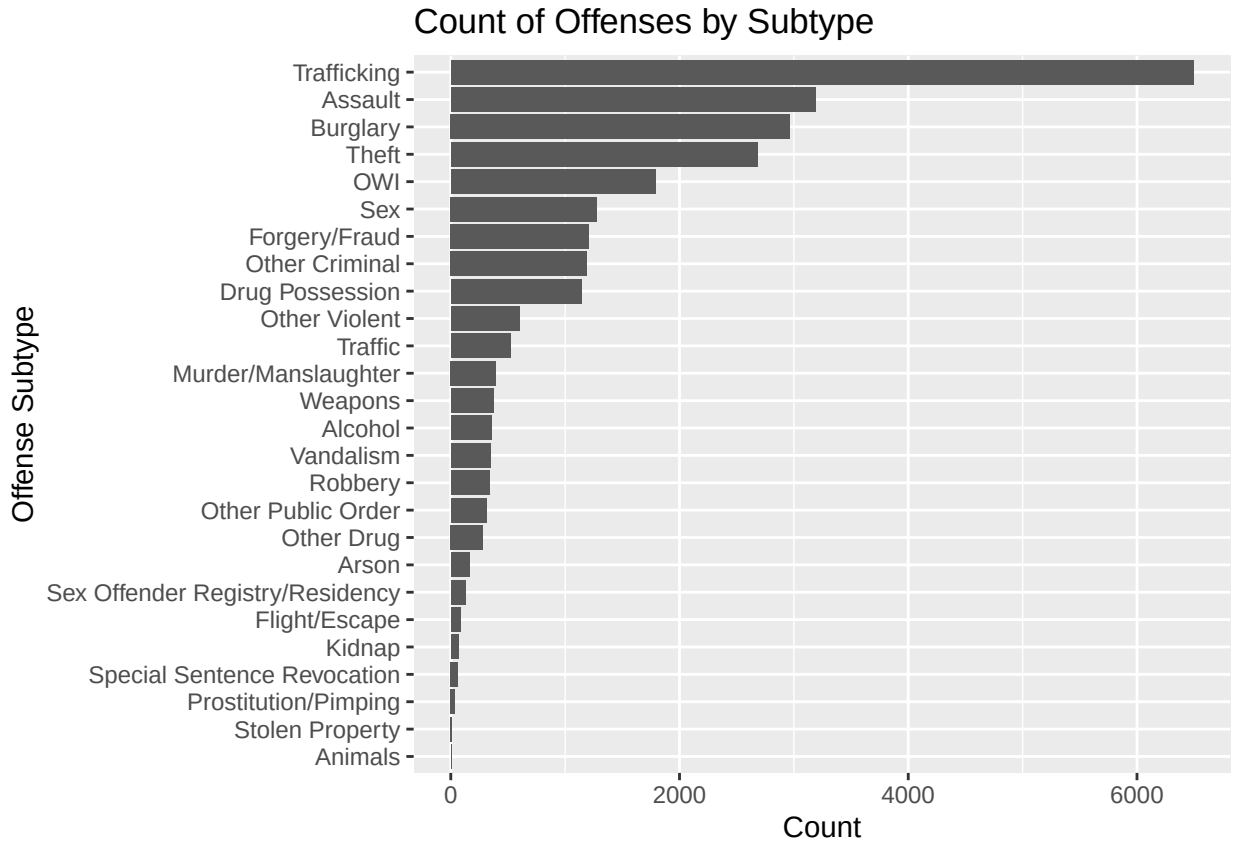
Plots:

```
ggplot(df, aes(x=Offense.Type)) + geom_bar() + labs(x="Offense Type", y="Count", title="Count of Offens
```



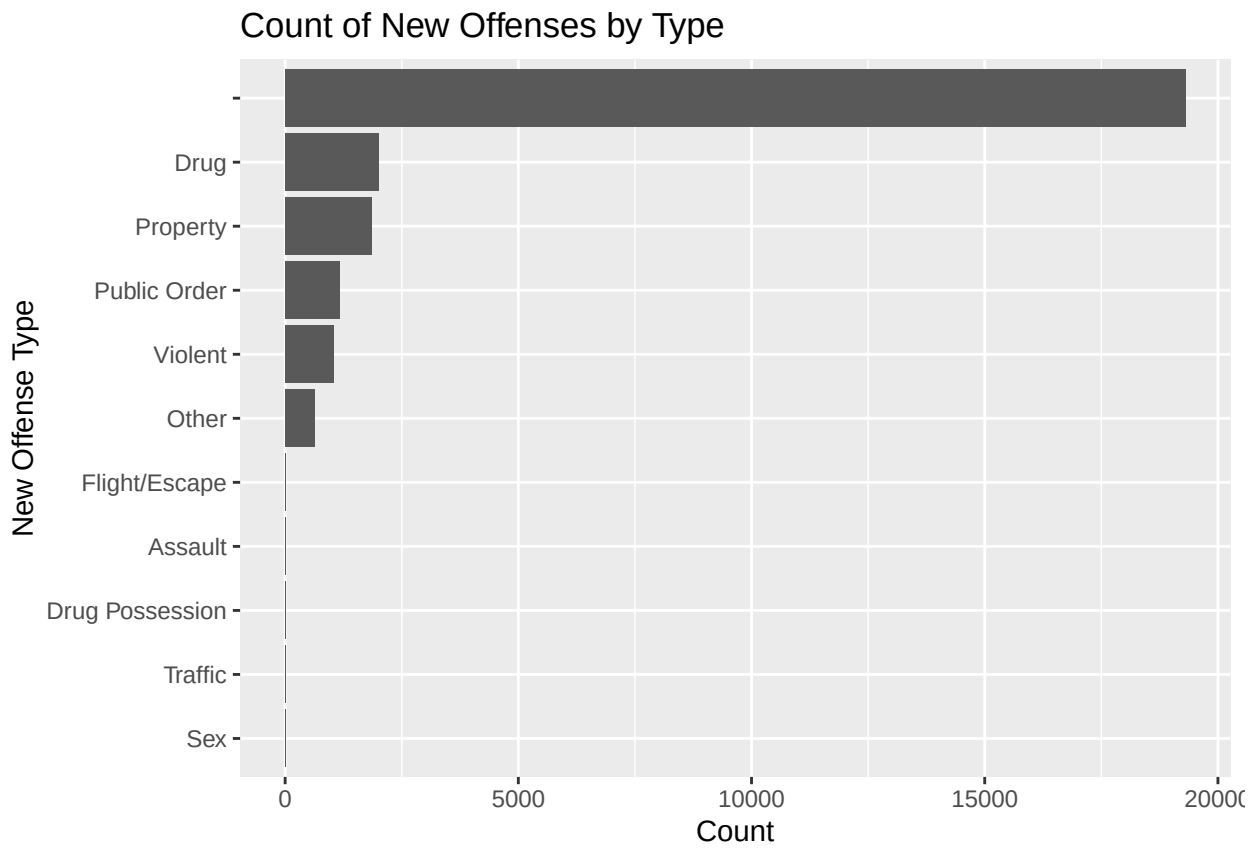
So, drug and proprtyb charges are our most common first offenses.

```
ggplot(df, aes(x=Offense.Subtype)) + geom_bar() + labs(x="Offense Subtype", y="Count", title="Count of
```



Trafficking is very high relative to its nearest competitors, which are assault and burglary.

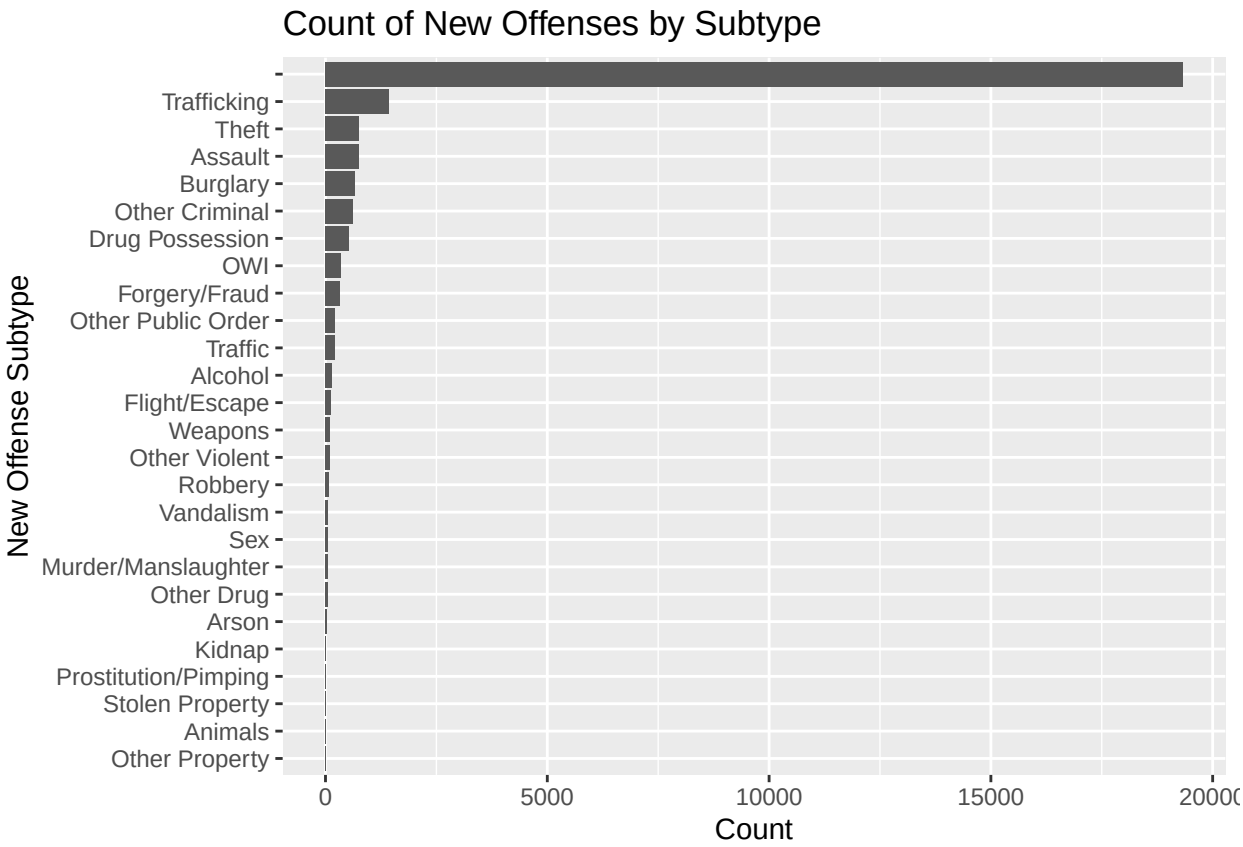
```
ggplot(df, aes(x=New.Offense.Type)) + geom_bar() + labs(x="New Offense Type", y="Count", title="Count o
```



We can see the vast majority of new offense types are empty values. Since these are so common, it is likely a mixture of non-reoffenders and those who reoffended with the same offense type. Besides empty values, drug and property offenses are most common.

What about subtypes?

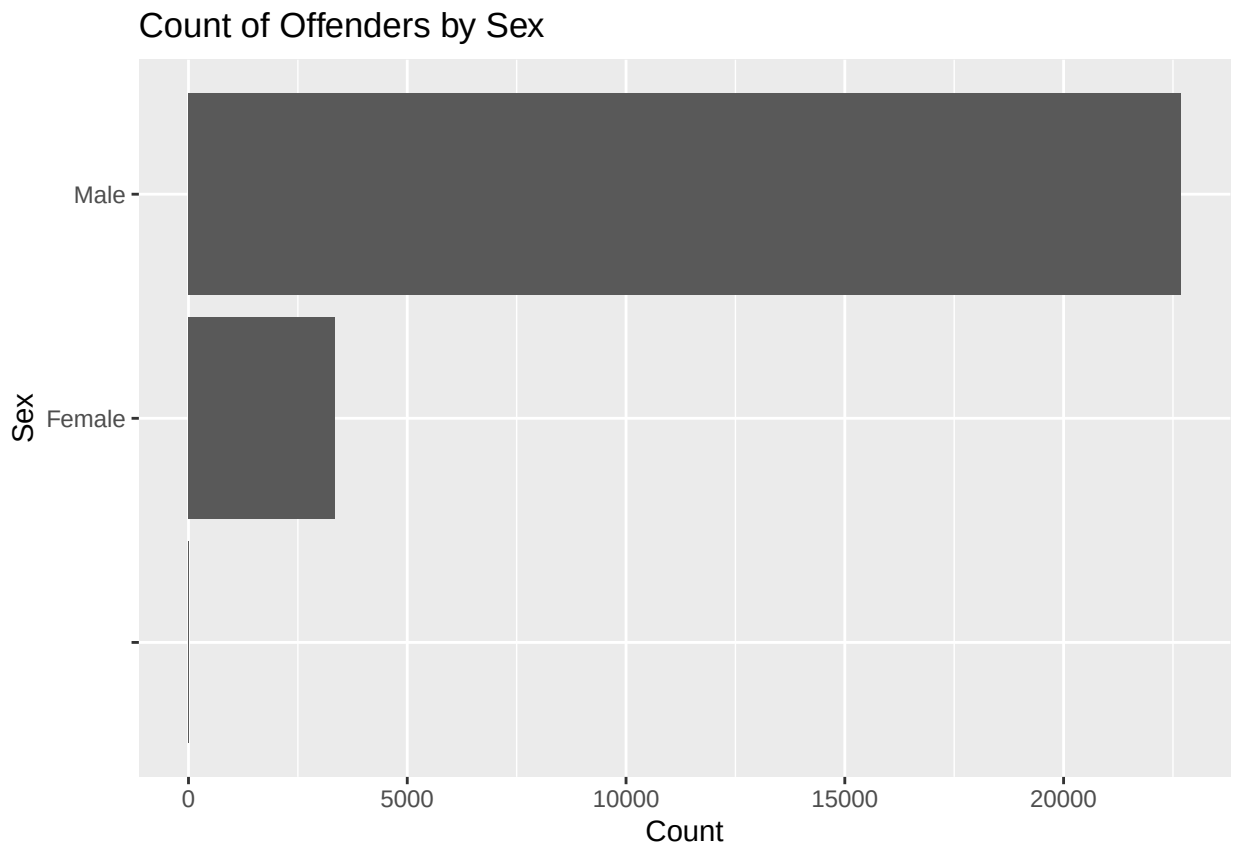
```
ggplot(df, aes(x=New.Offense.Sub.Type)) + geom_bar() + labs(x="New Offense Subtype", y="Count", title="
```



Again, the empty factor is by far the most common, followed by trafficking, theft, then assault.

Now, let's look at distribuitions for Sex, Day.to.Return, and Age.at.Release.

```
ggplot(df, aes(x=Sex)) + geom_bar() + labs(x="Sex", y="Count", title="Count of Offenders by Sex") + coo
```

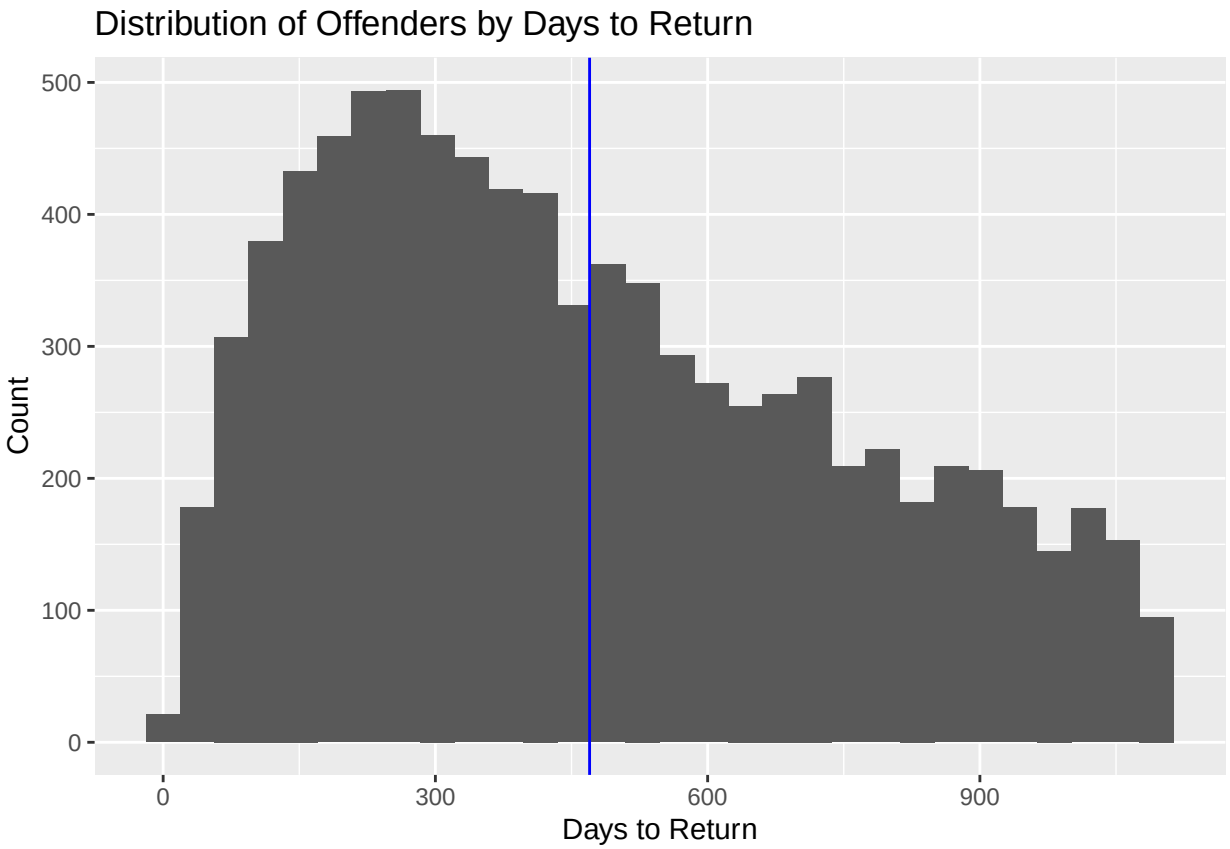


Mostly male.

```
days.mean = mean(df$Days.to.Return, na.rm=TRUE)
ggplot(df, aes(x=Days.to.Return)) + geom_histogram() + labs(x="Days to Return", y="Count", title="Distri
```

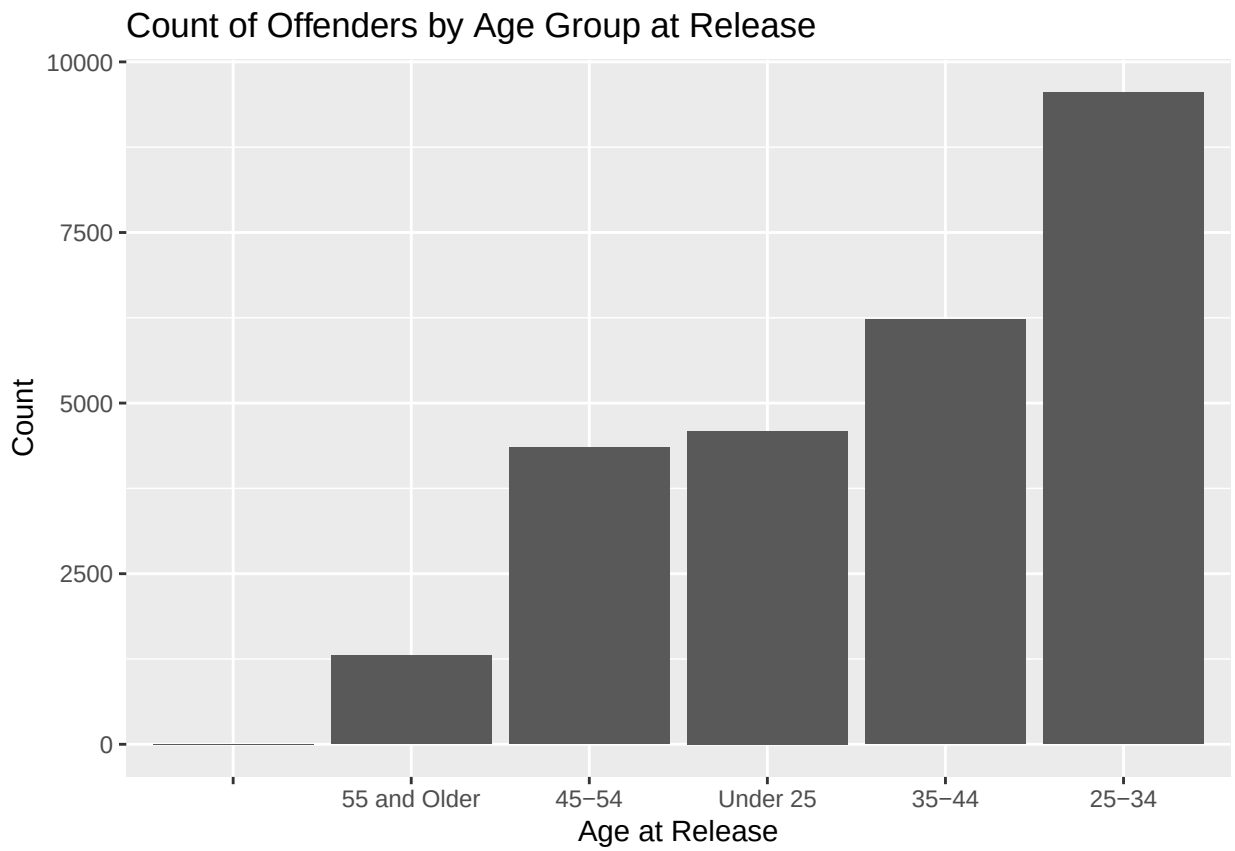
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 17339 rows containing non-finite values (stat_bin).
```



It looks like we have a right skewed distribution of days until return to prison with a mean of ~470.

```
ggplot(df, aes(x=Age.At.Release)) + geom_bar() + labs(x="Age at Release", y="Count", title="Count of Of
```



So, most offenders are 25-34 with the next highest group being 35-44.

Recidivism classifier

One question that the topic of the data is begging to be asked is if we can predict whether someone released will return to prison within the three year time frame of the data. First, let's see what proportion returned:

```
sum(df$Return.to.Prison) / length(df$Return.to.Prison)
```

```
## [1] 0.333628
```

So around a third of prisoners returned within three years.

Let's consider what values are useful for predicting Return.to.Prison. There are some variables we will need to consider altering, combining, or perhaps dropping, like Main.Supervising.District (MSD), which appears to be correlated with Release.Type (i.e., there is often no MSD when a prisoner is put on parole). Similar correlation is present in Target.Population. We will save these for later. Fiscal.Year.Released and Recidivism.Reporting.Year are 2010 and 2013 respectively for all inmates, since they are only labels of the dataset itself, so we will not use them for the model. Additionally, we are predicting whether recidivism will occur, so we will not include variables that confirm it, meaning any new-offense-related variables. These include: Days.to.Return, Recidivism.Type, New.Offense.Classification, New.Offense.Type, and New.Offense.Sub.Type. A non-null value for any of these would, of course, confirm recidivism. We will hold off on excluding or combining the potentially correlated variables, Main.Supervising.District, Release.Type, and Target.Population, for now. We can later perform Chi-squared tests of independence on them. We will

tentatively include them in the model. Below we also drop the single row with the value Interstate Compact Parole in Release.Type, as it is the only row with that value.

Before we fit the model, we will remove some rows with very rare values. While it is possible that these values may be good predictors, they are very rare (1-2 instances), so will not be especially useful in predicting the vast majority of cases. They also tend to cause issues when splitting training and testing sets.

```
# drop rows with very rare values (<3)
length(df$Release.Type)
```

```
## [1] 26020
```

```
indices_to_remove = vector()
indices_to_remove = append(indices_to_remove, which(df$Release.Type=="Interstate Compact Parole"))
indices_to_remove = append(indices_to_remove, which(df$Age.At.Release==""))
indices_to_remove = append(indices_to_remove, which(df$Race...Ethnicity=="Black -"))
indices_to_remove = append(indices_to_remove, which(df$Race...Ethnicity=="N/A -"))
indices_to_remove = append(indices_to_remove, which(df$Offense.Classification=="Other Misdemeanor"))
indices_to_remove = append(indices_to_remove, which(df$Offense.Classification=="Sexual Predator Communi
indices_to_remove = append(indices_to_remove, which(df$Offense.Classification=="Other Felony (Old Code)
df = df[-c(indices_to_remove),]
length(df$Release.Type)
```

```
## [1] 26007
```

Now, let's look at a full model using all of our potentially useful variables:

```
# first, train test split
set.seed(20)
train = sample(1:nrow(df),2*nrow(df)/3, replace=FALSE)
test = (-train)

# fit the model
model.fit = glm(Return.to.Prison~Main.Supervising.District+Release.Type+Race...Ethnicity+Age.At.Release
#summary(model.fit)
```

Uncomment the summary line if necessary. Here it is commented out because it is very long. Note that there are many many coefficients. This is because every variable is either boolean or categorical, and most of the categorical variables have several categories, resulting in a huge number of coefficients. Now, let's look at our actual predictions.

```
model.prob = predict(model.fit, df[test,],type='response')
head(model.prob)
```

```
##         3         5        11        13        16        20
## 0.4390969 0.2113999 0.4499045 0.3890737 0.5253218 0.3408298
```

```
model.pred = rep(FALSE, nrow(df)-length(test))
model.pred[model.prob >0.5] = TRUE
table(model.pred, df[test,]$Return.to.Prison)
```

```
##
## model.pred FALSE TRUE
##      FALSE  5358 2362
##      TRUE    431  518
```

```
1-mean(model.pred == df[test,]$Return.to.Prison)
```

```
## [1] 0.3221825
```

Not a fantastic accuracy, around 1/3. Next, we'll need to tune the model to see if we can improve it.