Data Challenge: Exploratory Data Analysis

Aldo Iturrios

3/27/2021

NA's in the Data

Question: What percentage of patients have at least one record for a particular variable (ie. 60% of patients have at least 1 blood pressure reading)

```
##       measures patients_atleast_one
## 1    Bilirubin                37.61
## 2  Cholesterol                 7.08
## 3   Creatinine                98.90
## 4      DiasABP                98.58
## 5         FiO2                65.29
## 6          GCS                98.58
## 7      Glucose                96.81
## 8         HCO3                98.36
## 9          HCT                98.99
## 10          HR                98.58
## 11           K                97.70
## 12     Lactate                49.82
## 13         MAP                98.58
## 14    MechVent                61.47
## 15          Mg                97.10
## 16   NIDiasABP                86.64
## 17       NIMAP                86.37
## 18    NISysABP                86.87
## 19          Na                98.22
## 20       PaCO2                72.29
## 21        PaO2                72.09
## 22   Platelets                98.50
## 23    RespRate                29.73
## 24        SaO2                42.77
## 25      SysABP                98.58
## 26        Temp                98.57
## 27   TroponinI                 3.90
## 28   TroponinT                19.02
## 29       Urine                97.61
## 30         WBC                98.41
## 31          pH                72.81
## 32         ALP                36.80
## 33         ALT                37.48
## 34         AST                37.73
## 35     Albumin                34.35
## 36         BUN                98.86
```

```
## 37        SAPS                 5.77
```

Question: Distribution of patient data at each time point (ie. 98% of patients have data at 00:30, 80% of patients have data at 06:30, etc. . .)

```
data.frame(timestamps, patients_atleast_one = round(times_vec * 100, 2))
```

```
##    timestamps patients_atleast_one
## 1       00.00                 2.07
## 2       00.30                50.18
## 3       01.00                55.99
## 4       01.30                63.32
## 5       02.00                66.25
## 6       02.30                68.65
## 7       03.00                68.04
## 8       03.30                69.82
## 9       04.00                67.99
## 10      04.30                69.22
## 11      05.00                67.73
## 12      05.30                68.84
## 13      06.00                67.19
## 14      06.30                67.75
## 15      07.00                65.79
## 16      07.30                66.68
## 17      08.00                64.94
## 18      08.30                65.73
## 19      09.00                63.85
## 20      09.30                63.80
## 21      10.00                62.51
## 22      10.30                63.27
## 23      11.00                61.80
## 24      11.30                62.56
## 25      12.00                61.65
## 26      12.30                62.40
## 27      13.00                60.72
## 28      13.30                59.99
## 29      14.00                59.82
## 30      14.30                60.38
## 31      15.00                60.15
## 32      15.30                60.42
## 33      16.00                59.38
## 34      16.30                60.40
## 35      17.00                59.24
## 36      17.30                60.50
## 37      18.00                59.19
## 38      18.30                59.85
## 39      19.00                59.30
## 40      19.30                58.80
## 41      20.00                59.17
## 42      20.30                59.06
## 43      21.00                58.15
## 44      21.30                59.12
```

```
## 45      22.00               57.29
## 46      22.30               58.86
## 47      23.00               58.41
## 48      23.30               58.89
## 49      24.00               97.97
## 50      24.30               59.11
## 51      25.00               56.99
## 52      25.30               58.10
## 53      26.00               56.55
## 54      26.30               57.65
## 55      27.00               56.99
## 56      27.30               57.69
## 57      28.00               56.59
## 58      28.30               57.45
## 59      29.00               55.80
## 60      29.30               57.27
## 61      30.00               55.88
## 62      30.30               57.13
## 63      31.00               55.77
## 64      31.30               56.81
## 65      32.00               55.02
## 66      32.30               56.18
## 67      33.00               55.10
## 68      33.30               55.53
## 69      34.00               55.03
## 70      34.30               55.99
## 71      35.00               54.78
## 72      35.30               55.81
## 73      36.00               54.98
## 74      36.30               55.92
## 75      37.00               54.97
## 76      37.30               55.40
## 77      38.00               54.27
## 78      38.30               55.00
## 79      39.00               54.70
## 80      39.30               54.90
## 81      40.00               54.73
## 82      40.30               55.66
## 83      41.00               55.59
## 84      41.30               56.19
## 85      42.00               54.94
## 86      42.30               55.30
## 87      43.00               54.31
## 88      43.30               55.08
## 89      44.00               53.18
## 90      44.30               54.06
## 91      45.00               53.79
## 92      45.30               54.31
## 93      46.00               53.68
## 94      46.30               53.70
## 95      47.00               53.43
## 96      47.30               53.77
## 97      48.00               52.71
```

EDA on Training Data (Looking at variables in relation to Outcome)

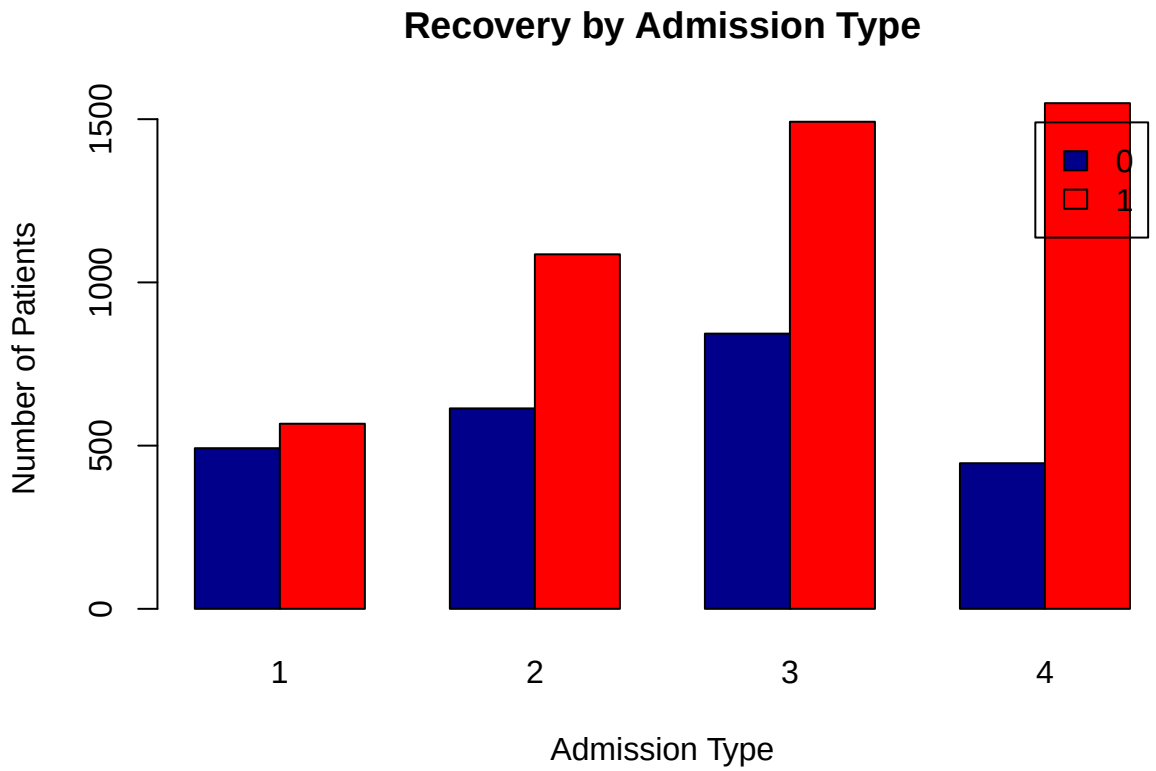
Outcome:

```
# Training Data
table(train_data$outcome)
```

```
##
##    0    1
## 2395 4694
```

Recovery by Admission Type

```
counts <- table(train_data$outcome, train_data$AdmissionType)
barplot(counts, main="Recovery by Admission Type", xlab="Admission Type", ylab = "Number of Patients",
```



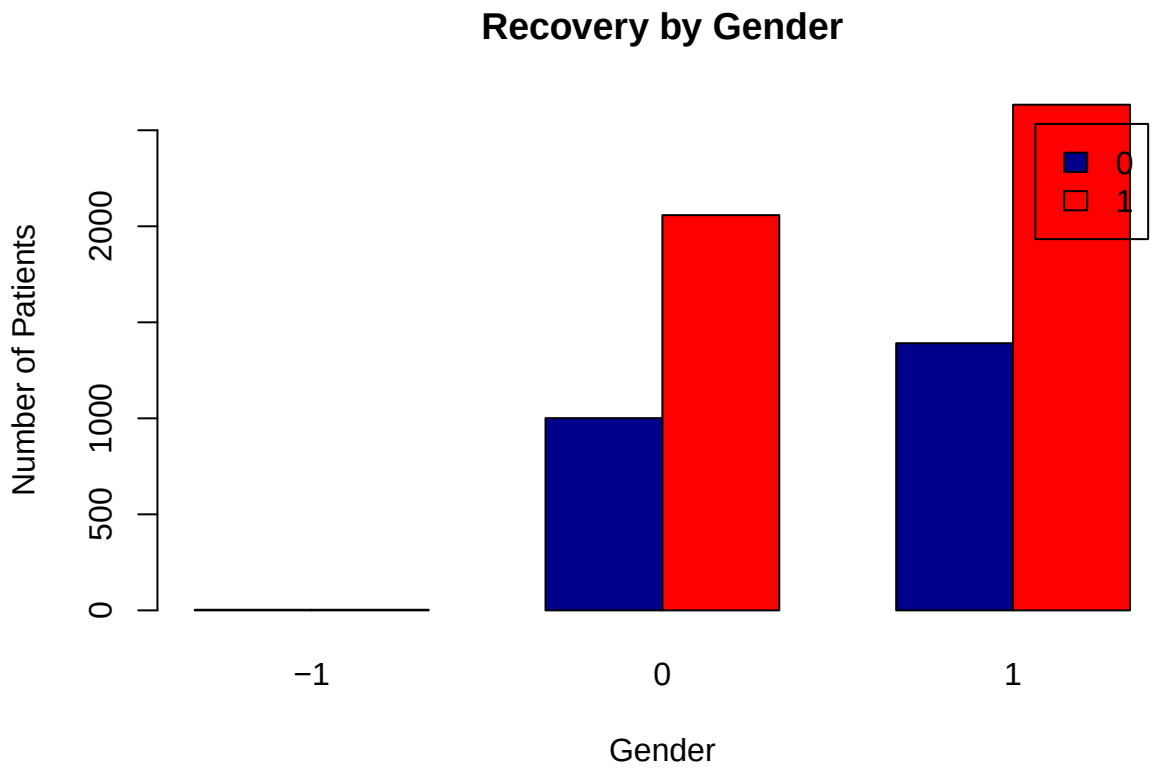

```
addmargins(counts)
```

```
##
##          1    2    3    4  Sum
```

```
##   0    492  614  843  446 2395
##   1    567 1086 1492 1549 4694
##   Sum 1059 1700 2335 1995 7089
```

Recovery by Gender

```
gender_counts <- table(train_data$outcome, train_data$Gender)
barplot(gender_counts, main="Recovery by Gender", xlab="Gender", ylab = "Number of Patients", col=c("da
```



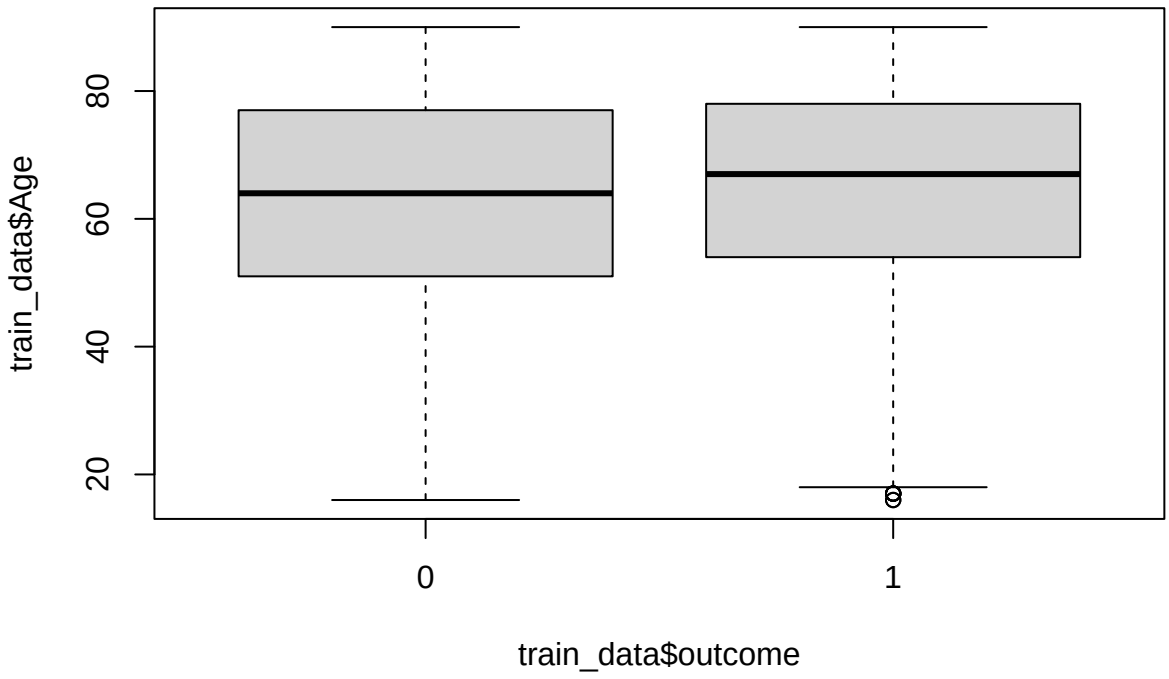

```
addmargins(gender_counts)
```

```
##
##         -1    0    1  Sum
##   0      3 1001 1391 2395
##   1      3 2058 2633 4694
##   Sum    6 3059 4024 7089
```

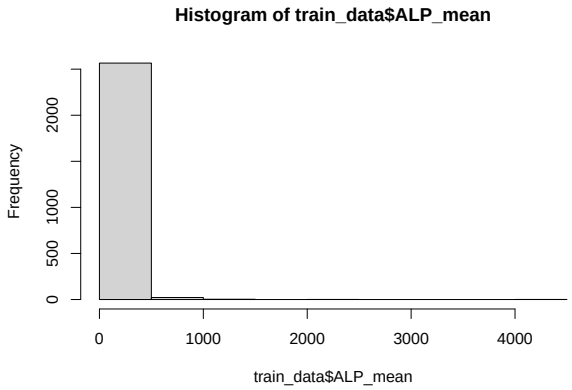
```
summary(train_data$Age)
```

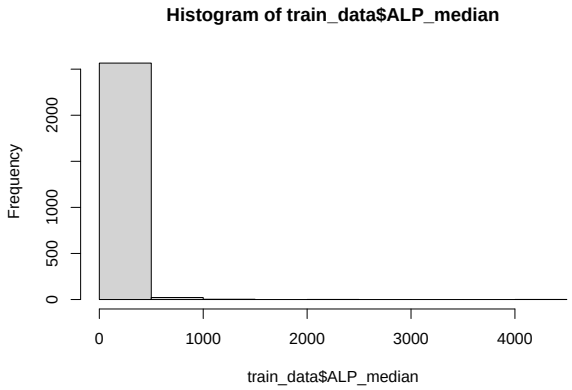
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   16.00   53.00   66.00   63.87   78.00   90.00
```

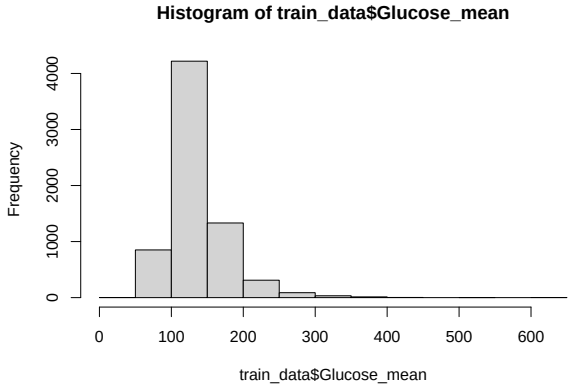
```
boxplot(train_data$Age ~ train_data$outcome)
```

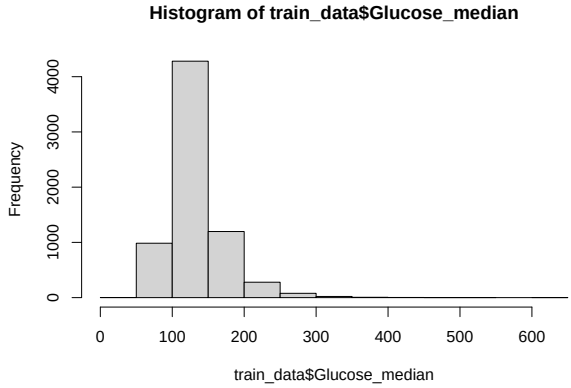


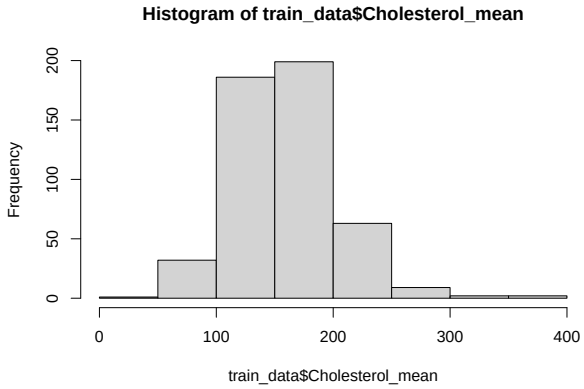
Look at a few measure summary variables

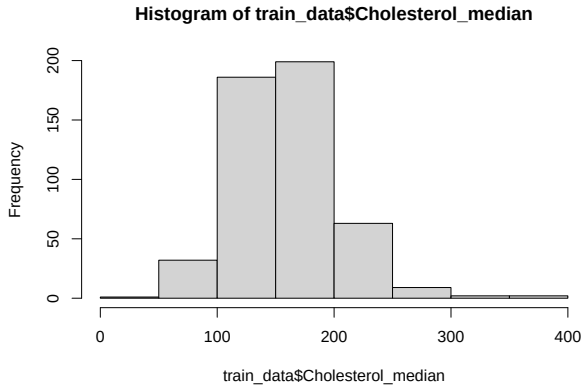


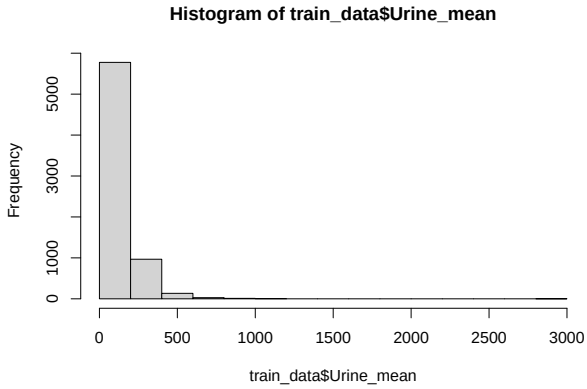


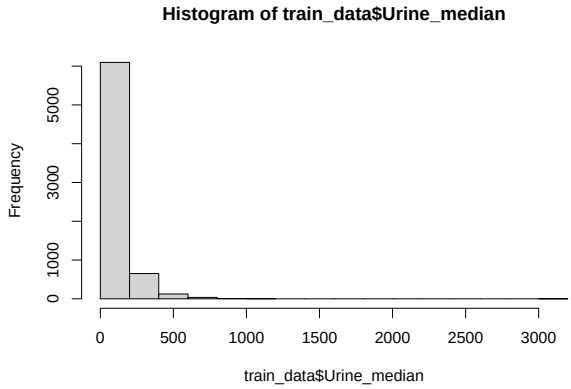

Histogram of train_data$Glucose_mean
Frequency
0
1000
2000
3000
4000
0
100
200
300
400
500
600
train_data$Glucose_mean


Histogram of train_data$Glucose_median
Frequency
0
1000
2000
3000
4000
0
100
200
300
400
500
600
train_data$Glucose_median


Histogram of train_data$Cholesterol_mean
Frequency
0
50
100
150
200
0
100
200
300
400
train_data$Cholesterol_mean


Histogram of train_data$Cholesterol_median
Frequency
0
50
100
150
200
0
100
200
300
400
train_data$Cholesterol_median


Histogram of train_data$Urine_mean
Frequency
0
1000
3000
5000
0
500
1000
1500
2000
2500
3000
train_data$Urine_mean


Histogram of train_data$Urine_median
Frequency
0
1000
3000
5000
0
500
1000
1500
2000
2500
3000
train_data$Urine_median