EDA Exercise 3 and 4

Lance Price

Exercise 3

Problem 1

```
# download zip file from url and extract it

url <- "https://archive.ics.uci.edu/ml/machine-learning-databases/00296/dataset_diabetes.zip"

download.file(url, "~/Downloads/Diabetes.zip")

# extract the file into directory
unzip("~/Downloads/Diabetes.zip", exdir = "~/Downloads")

# read files into R Studio

csv_file_1 <- "~/Downloads/dataset_diabetes/diabetic_data.csv"

csv_file_2 <- "~/Downloads/dataset_diabetes/IDs_mapping.csv"

diabetic_data <- read.csv(csv_file_1, header=TRUE)

ID_mapping_data <- read.csv(csv_file_2, header=TRUE)
```

Problem 2

Document missing values for encounter_id and patient_nbr The article that documents the attributes says that these attributes are unique identifiers of encounters and patients, respectively. This is backed up by the fact that the number of unique elements in the lists of the values for these attributes are both 101766. This is the exact number of records.

```
# Document missing values for race:
# Missing values for race are shown as question marks.
# I will change these question marks to NA values so I can
# use the is.na function to count how many missing values there are.
# Alternatively, I could have just counted how many values were equal to "?".
# Sub NA for "?"
diabetic_data$race[diabetic_data$race == "?"] <- NA
# The following computation gives the percentage of missing values for the race attribute
percent_missing1 <- sum(is.na(diabetic_data$race))/length(diabetic_data$race) * 100
```

2.2335554 % of the race values are missing in this data. There are 2273 missing values in the race attribute.

```
# Document missing values for gender:
# Missing values for gender are shown as "Unknown/Invalid"
# Some might say they are not missing values, and I think there are valid
# arguments for either side of this judgement call.
# Sub NA for "Unknown/Invalid"
diabetic_data$gender[diabetic_data$gender == "Unknown/Invalid"] <- NA
# The following computation gives the percentage of missing values in the total
# values in the gender attribute
percent_missing2 <- sum(is.na(diabetic_data$gender))/length(diabetic_data$gender) * 100
```

0.0029479 % of the gender values are missing in this data. There are 3 missing values in the gender attribute.

The unique values of the age attribute are: [0-10), [10-20), [20-30), [30-40), [40-50), [50-60), [60-70), [70-80), [80-90), [90-100) Given this list, there are no missing values in the age attribute.

```
# Document missing values for weight
# Missing values for weight are shown as question marks
# Sub NA for "?"
diabetic_data$weight[diabetic_data$weight == "?"] <- NA
# The following computation gives the percentage of missing values for the weight attribute
percent_missing3 <- sum(is.na(diabetic_data$weight))/length(diabetic_data$weight) * 100
```

96.8584793 % of the weight values are missing in this data. There are 98569 missing values in the weight attribute.

```
# Document missing values for admission_type_id
# Missing values for admission_type_id are shown as 5, 6, and 8,
# which correspond to "Not Available","NULL", and "Not Mapped"
# according to the ID_mapping_data table.
# Sub NA for these numbers
diabetic_data$admission_type_id[diabetic_data$admission_type_id == 5] <- NA
diabetic_data$admission_type_id[diabetic_data$admission_type_id == 6] <- NA
diabetic_data$admission_type_id[diabetic_data$admission_type_id == 8] <- NA
# The following computation gives the percentage of missing values for the admission_type_id attribute
percent_missing4 <- sum(is.na(diabetic_data$admission_type_id))/length(diabetic_data$admission_type_id)
```

10.2155926 % of the admission_type_id values are missing in this data. There are 10396 missing values in the admission_type_id attribute.

```
# Document missing values for discharge_disposition_id
# Missing values for discharge_disposition_id are shown as 18, 25, and 26,
# which correspond to "NULL", "Not Mapped", and "Unknown/Invalid"
# according to the ID_mapping_data table.
# Sub NA for these numbers
diabetic_data$discharge_disposition_id[diabetic_data$discharge_disposition_id == 18] <- NA
diabetic_data$discharge_disposition_id[diabetic_data$discharge_disposition_id == 25] <- NA
diabetic_data$discharge_disposition_id[diabetic_data$discharge_disposition_id == 26] <- NA
# The following computation gives the percentage of missing values for the discharge_disposition_id att
percent_missing5 <- sum(is.na(diabetic_data$discharge_disposition_id))/length(diabetic_data$discharge_di
```

4.5987854 % of the discharge_disposition_id values are missing in this data. There are 4680 missing values in the discharge_disposition_id attribute.

```
# Document missing values for admission_source_id
# Missing values for admission_source_id are shown as 9, 15, 17, 20, and 21,
# which correspond to "Not Available","NULL","Not Mapped", and "Unknown/Invalid"
# according to the ID_mapping_data table.
# Sub NA for these numbers
diabetic_data$admission_source_id[diabetic_data$admission_source_id == 9] <- NA
diabetic_data$admission_source_id[diabetic_data$admission_source_id == 15] <- NA
diabetic_data$admission_source_id[diabetic_data$admission_source_id == 17] <- NA
diabetic_data$admission_source_id[diabetic_data$admission_source_id == 20] <- NA
diabetic_data$admission_source_id[diabetic_data$admission_source_id == 21] <- NA
# The following computation gives the percentage of missing values for the admission_source_id attribut
percent_missing6 <- sum(is.na(diabetic_data$admission_source_id))/length(diabetic_data$admission_source_
```

6.9443626 % of the admission_source_id values are missing in this data. There are 7067 missing values in the admission_source_id attribute.

The unique values of the time_in_hospital attribute are: 1, 3, 2, 4, 5, 13, 12, 9, 7, 10, 6, 11, 8, 14 Given this list, there are no missing values in the time_in_hospital attribute.

```
# Document missing values for payer_code
# Missing values for payer_code are shown as question marks
# Sub NA for "?"
diabetic_data$payer_code[diabetic_data$payer_code == "?"] <- NA
```

```
# The following computation gives the percentage of missing values for the payer_code attribute
percent_missing7 <- sum(is.na(diabetic_data$payer_code))/length(diabetic_data$payer_code) * 100
```

39.557416 % of the payer_code values are missing in this data. There are 40256 missing values in the payer_code attribute.

```
# Document missing values for medical_specialty
# Missing values for medical_specialty are shown as question marks
# Sub NA for "?"
diabetic_data$medical_specialty[diabetic_data$medical_specialty == "?"] <- NA
# The following computation gives the percentage of missing values for the medical_specialty attribute
percent_missing8 <- sum(is.na(diabetic_data$medical_specialty))/length(diabetic_data$medical_specialty)
```

49.0822082 % of the medical_specialty values are missing in this data. There are 49949 missing values in the medical_specialty attribute.

For the attribute num_lab_procedures, the values range from 1 to 132, and there are no missing values.

For the attribute num_procedures, the values range from 0 to 6, and there are no missing values.

For the attribute num_medications, the values range from 1 to 81, and there are no missing values.

For the attribute number_outpatient, the values range from 0 to 42, and there are no missing values.

For the attribute number_emergency, the values range from 0 to 76, and there are no missing values.

For the attribute number_inpatient, the values range from 0 to 21, and there are no missing values.

```
# Document missing values for diag_1
# Missing values for diag_1 are shown as question marks
# Sub NA for "?"
diabetic_data$diag_1[diabetic_data$diag_1 == "?"] <- NA
# The following computation gives the percentage of missing values for the diag_1 attribute
percent_missing9 <- sum(is.na(diabetic_data$diag_1))/length(diabetic_data$diag_1) * 100
```

0.0206356 % of the diag_1 values are missing in this data. There are 21 missing values in the diag_1 attribute.

```
# Document missing values for diag_2
# Missing values for diag_2 are shown as question marks
# Sub NA for "?"
diabetic_data$diag_2[diabetic_data$diag_2 == "?"] <- NA
# The following computation gives the percentage of missing values for the diag_2 attribute
percent_missing10 <- sum(is.na(diabetic_data$diag_2))/length(diabetic_data$diag_2) * 100
```

0.3517874 % of the diag_2 values are missing in this data. There are 358 missing values in the diag_2 attribute.

```
# Document missing values for diag_3
# Missing values for diag_3 are shown as question marks
# Sub NA for "?"
diabetic_data$diag_3[diabetic_data$diag_3 == "?"] <- NA
# The following computation gives the percentage of missing values for the diag_3 attribute
percent_missing11 <- sum(is.na(diabetic_data$diag_3))/length(diabetic_data$diag_3) * 100
```

1.3983059 % of the diag_3 values are missing in this data. There are 1423 missing values in the diag_3 attribute.

For the attribute num_diagnoses, the values range from 1 to 16, and there are no missing values.

Document missing values for the max_glu_serum attribute: The unique values of the max_glu_serum attribute are: None, >300, Norm, >200 Given this list, the missing values in the max_glu_serum attribute are shown as "None".

```
# Substitute NA for "None"
diabetic_data$max_glu_serum[diabetic_data$max_glu_serum == "None"] <- NA
# The following computation gives the percentage of missing values for the max_glu_serum attribute
percent_missing12 <- sum(is.na(diabetic_data$max_glu_serum))/length(diabetic_data$max_glu_serum) * 100
```

94.746772 % of the max_glu_serum values are missing in this data. There are 96420 missing values in the max_glu_serum attribute.

Document missing values for the A1Cresult attribute: The unique values of the A1Cresult attribute are: None, >7, >8, Norm Given this list, the missing values in the A1Cresult attribute are shown as "None".

```
# Substitute NA for "None"
diabetic_data$A1Cresult[diabetic_data$A1Cresult == "None"] <- NA
# The following computation gives the percentage of missing values for the A1Cresult attribute
percent_missing13 <- sum(is.na(diabetic_data$A1Cresult))/length(diabetic_data$A1Cresult) * 100
```

83.2773225 % of the A1Cresult values are missing in this data. There are 84748 missing values in the A1Cresult attribute.

Document missing values for the metformin attribute: The unique values of the metformin attribute are: No, Steady, Up, Down Given this list, the missing values in the metformin attribute are shown as "No".

```
# Substitute NA for "No"
diabetic_data$metformin[diabetic_data$metformin == "No"] <- NA
# The following computation gives the percentage of missing values for the metformin attribute
percent_missing14 <- sum(is.na(diabetic_data$metformin))/length(diabetic_data$metformin) * 100
```

80.3588625 % of the metformin values are missing in this data. There are 81778 missing values in the metformin attribute.

Document missing values for the repaglinide attribute: The unique values of the repaglinide attribute are: No, Up, Steady, Down Given this list, the missing values in the repaglinide attribute are shown as "No".

```
# Substitute NA for "No"
diabetic_data$repaglinide[diabetic_data$repaglinide == "No"] <- NA
# The following computation gives the percentage of missing values for the repaglinide attribute
percent_missing15 <- sum(is.na(diabetic_data$repaglinide))/length(diabetic_data$repaglinide) * 100
```

98.4877071 % of the repaglinide values are missing in this data. There are 100227 missing values in the repaglinide attribute.

Document missing values for the nateglinide attribute: The unique values of the nateglinide attribute are: No, Steady, Down, Up Given this list, the missing values in the nateglinide attribute are shown as "No".

```
# Substitute NA for "No"
diabetic_data$nateglinide[diabetic_data$nateglinide == "No"] <- NA
# The following computation gives the percentage of missing values for the nateglinide attribute
percent_missing16 <- sum(is.na(diabetic_data$nateglinide))/length(diabetic_data$nateglinide) * 100
```

99.3091995 % of the nateglinide values are missing in this data. There are 101063 missing values in the nateglinide attribute.

Document missing values for the chlorpropamide attribute: The unique values of the chlorpropamide attribute are: No, Steady, Down, Up Given this list, the missing values in the chlorpropamide attribute are shown as "No".

```
# Substitute NA for "No"
diabetic_data$chlorpropamide[diabetic_data$chlorpropamide == "No"] <- NA
# The following computation gives the percentage of missing values for the chlorpropamide attribute
percent_missing17 <- sum(is.na(diabetic_data$chlorpropamide))/length(diabetic_data$chlorpropamide) * 10
```

99.9154924 % of the chlorpropamide values are missing in this data. There are 101680 missing values in the chlorpropamide attribute.

Document missing values for the glimepiride attribute: The unique values of the glimepiride attribute are: No, Steady, Down, Up Given this list, the missing values in the glimepiride attribute are shown as "No".

```
# Substitute NA for "No"
diabetic_data$glimepiride[diabetic_data$glimepiride == "No"] <- NA
# The following computation gives the percentage of missing values for the glimepiride attribute
percent_missing18 <- sum(is.na(diabetic_data$glimepiride))/length(diabetic_data$glimepiride) * 100
```

94.8990822 % of the glimepiride values are missing in this data. There are 96575 missing values in the glimepiride attribute.

Document missing values for the acetohexamide attribute: The unique values of the acetohexamide attribute are: No, Steady Given this list, the missing values in the acetohexamide attribute are shown as "No".

```
# Substitute NA for "No"
diabetic_data$acetohexamide[diabetic_data$acetohexamide == "No"] <- NA
# The following computation gives the percentage of missing values for the acetohexamide attribute
percent_missing19 <- sum(is.na(diabetic_data$acetohexamide))/length(diabetic_data$acetohexamide) * 100
```

99.9990174 % of the acetohexamide values are missing in this data. There are 101765 missing values in the acetohexamide attribute.

Document missing values for the glipizide attribute: The unique values of the glipizide attribute are: No, Steady, Up, Down Given this list, the missing values in the glipizide attribute are shown as "No".

```
# Substitute NA for "No"
diabetic_data$glipizide[diabetic_data$glipizide == "No"] <- NA
# The following computation gives the percentage of missing values for the glipizide attribute
percent_missing20 <- sum(is.na(diabetic_data$glipizide))/length(diabetic_data$glipizide) * 100
```

87.534147 % of the glipizide values are missing in this data. There are 89080 missing values in the glipizide attribute.

Document missing values for the glyburide attribute: The unique values of the glyburide attribute are: No, Steady, Up, Down Given this list, the missing values in the glyburide attribute are shown as "No".

```
# Substitute NA for "No"
diabetic_data$glyburide[diabetic_data$glyburide == "No"] <- NA
# The following computation gives the percentage of missing values for the glyburide attribute
percent_missing21 <- sum(is.na(diabetic_data$glyburide))/length(diabetic_data$glyburide) * 100
```

89.5348152 % of the glyburide values are missing in this data. There are 91116 missing values in the glyburide attribute.

Document missing values for the tolbutamide attribute: The unique values of the tolbutamide attribute are: No, Steady Given this list, the missing values in the tolbutamide attribute are shown as "No".

```
# Substitute NA for "No"
diabetic_data$tolbutamide[diabetic_data$tolbutamide == "No"] <- NA
# The following computation gives the percentage of missing values for the tolbutamide attribute
percent_missing22 <- sum(is.na(diabetic_data$tolbutamide))/length(diabetic_data$tolbutamide) * 100
```

99.9773991 % of the tolbutamide values are missing in this data. There are 101743 missing values in the tolbutamide attribute.

Document missing values for the pioglitazone attribute: The unique values of the pioglitazone attribute are: No, Steady, Up, Down Given this list, the missing values in the pioglitazone attribute are shown as "No".

```
# Substitute NA for "No"
diabetic_data$pioglitazone[diabetic_data$pioglitazone == "No"] <- NA
# The following computation gives the percentage of missing values for the pioglitazone attribute
percent_missing23 <- sum(is.na(diabetic_data$pioglitazone))/length(diabetic_data$pioglitazone) * 100
```

92.7991667 % of the pioglitazone values are missing in this data. There are 94438 missing values in the pioglitazone attribute.

Document missing values for the rosiglitazone attribute: The unique values of the rosiglitazone attribute are: No, Steady, Up, Down Given this list, the missing values in the rosiglitazone attribute are shown as "No".

```
# Substitute NA for "No"
diabetic_data$rosiglitazone[diabetic_data$rosiglitazone == "No"] <- NA
# The following computation gives the percentage of missing values for the rosiglitazone attribute
percent_missing24 <- sum(is.na(diabetic_data$rosiglitazone))/length(diabetic_data$rosiglitazone) * 100
```

93.7454553 % of the rosiglitazone values are missing in this data. There are 95401 missing values in the rosiglitazone attribute.

Document missing values for the acarbose attribute: The unique values of the acarbose attribute are: No, Steady, Up, Down Given this list, the missing values in the acarbose attribute are shown as "No".

```
# Substitute NA for "No"
diabetic_data$acarbose[diabetic_data$acarbose == "No"] <- NA
# The following computation gives the percentage of missing values for the acarbose attribute
percent_missing25 <- sum(is.na(diabetic_data$acarbose))/length(diabetic_data$acarbose) * 100
```

99.6973449 % of the acarbose values are missing in this data. There are 101458 missing values in the acarbose attribute.

Document missing values for the miglitol attribute: The unique values of the miglitol attribute are: No, Steady, Down, Up Given this list, the missing values in the miglitol attribute are shown as "No".

```
# Substitute NA for "No"
diabetic_data$miglitol[diabetic_data$miglitol == "No"] <- NA
# The following computation gives the percentage of missing values for the miglitol attribute
percent_missing26 <- sum(is.na(diabetic_data$miglitol))/length(diabetic_data$miglitol) * 100
```

99.9626594 % of the miglitol values are missing in this data. There are 101728 missing values in the miglitol attribute.

It is now clear that we don't need to keep on showing the unique values for the rest of these drug attributes. They all have "Up", "Down", "Steady", or "No" as values.

```
# Substitute NA for "No" in the rest of the drug attributes
diabetic_data$troglitazone[diabetic_data$troglitazone == "No"] <- NA
diabetic_data$tolazamide[diabetic_data$tolazamide == "No"] <- NA
diabetic_data$examide[diabetic_data$examide == "No"] <- NA
diabetic_data$citoglipton[diabetic_data$citoglipton == "No"] <- NA
diabetic_data$insulin[diabetic_data$insulin == "No"] <- NA
diabetic_data$glyburide.metformin[diabetic_data$glyburide.metformin == "No"] <- NA
diabetic_data$glipizide.metformin[diabetic_data$glipizide.metformin == "No"] <- NA
diabetic_data$glimepiride.pioglitazone[diabetic_data$glimepiride.pioglitazone == "No"] <- NA
diabetic_data$metformin.rosiglitazone[diabetic_data$metformin.rosiglitazone == "No"] <- NA
diabetic_data$metformin.pioglitazone[diabetic_data$metformin.pioglitazone == "No"] <- NA
# The folowing computations give the percentage of missing values for the troglitazone,
# tolazamide, tolazamide, examide, citoglipton, insulin, glyburide.metformin,
# glipizide.metformin, glimepiride.pioglitazone, metformin.rosiglitazone,and
# metformin.pioglitazone attributes.
percent_missing <- vector()
percent_missing[1] <- sum(is.na(diabetic_data$troglitazone))/length(diabetic_data$troglitazone) * 100
percent_missing[2] <- sum(is.na(diabetic_data$tolazamide))/length(diabetic_data$tolazamide) * 100
percent_missing[3] <- sum(is.na(diabetic_data$examide))/length(diabetic_data$examide) * 100
percent_missing[4] <- sum(is.na(diabetic_data$citoglipton))/length(diabetic_data$citoglipton) * 100
percent_missing[5] <- sum(is.na(diabetic_data$insulin))/length(diabetic_data$insulin) * 100
percent_missing[6] <- sum(is.na(diabetic_data$glyburide.metformin))/length(diabetic_data$glyburide.metf
percent_missing[7] <- sum(is.na(diabetic_data$glipizide.metformin))/length(diabetic_data$glipizide.metf
percent_missing[8] <- sum(is.na(diabetic_data$glimepiride.pioglitazone))/length(diabetic_data$glimepiri
percent_missing[9] <- sum(is.na(diabetic_data$metformin.rosiglitazone))/length(diabetic_data$metformin.
percent_missing[10] <- sum(is.na(diabetic_data$metformin.pioglitazone))/length(diabetic_data$metformin.
# The following computations give th number of missing values for these remaining drugs.
num_missing <- vector()
num_missing[1] <- sum(is.na(diabetic_data$troglitazone))
num_missing[2] <- sum(is.na(diabetic_data$tolazamide))
num_missing[3] <- sum(is.na(diabetic_data$examide))
num_missing[4] <- sum(is.na(diabetic_data$citoglipton))
```

```
num_missing[5] <- sum(is.na(diabetic_data$insulin))
num_missing[6] <- sum(is.na(diabetic_data$glyburide.metformin))
num_missing[7] <- sum(is.na(diabetic_data$glipizide.metformin))
num_missing[8] <- sum(is.na(diabetic_data$glimepiride.pioglitazone))
num_missing[9] <- sum(is.na(diabetic_data$metformin.rosiglitazone))
num_missing[10] <- sum(is.na(diabetic_data$metformin.pioglitazone))
# Create simple table that summarizes the percent missing and number of missing values for these remain
matrix_drugs <- matrix(c(percent_missing[1], num_missing[1],
percent_missing[2], num_missing[2],
percent_missing[3], num_missing[3],
percent_missing[4], num_missing[4],
percent_missing[5], num_missing[5],
percent_missing[6], num_missing[6],
percent_missing[7], num_missing[7],
percent_missing[8], num_missing[8],
percent_missing[9], num_missing[9],
percent_missing[10], num_missing[10]), ncol = 2, byrow = TRUE)
colnames(matrix_drugs) <- c("Percent Missing", "Values Missing")
rownames(matrix_drugs) <- c("troglitazone", "tolazamide", "examide", "citoglipton", "insulin", "glyburi
"glipizide.metformin", "glimepiride.pioglitazone", "metformin.rosiglitazone", "metformin.pioglitazone")
table_drugs <- as.table(matrix_drugs)
```

The following table summarizes the remaining drugs with respect to percent missing and missing values.

```
##                           Percent Missing Values Missing
## troglitazone                     99.99705      101763.00000
## tolazamide                       99.96168      101727.00000
## examide                         100.00000      101766.00000
## citoglipton                     100.00000      101766.00000
## insulin                          46.56074       47383.00000
## glyburide.metformin              99.30625      101060.00000
## glipizide.metformin              99.98723      101753.00000
## glimepiride.pioglitazone         99.99902      101765.00000
## metformin.rosiglitazone          99.99803      101764.00000
## metformin.pioglitazone           99.99902      101765.00000
```

These values could also be interpreted as a legit "No", which would mean the people don't take these drugs. I was not sure if the "No" was also representing data that was missing and therefore could not say whether the dosage was being raised, lowered, or maintained as it is. Because of this abiguity, I chose to consider them missing values.

The change attribute says whether the dosage of any of the drugs were changed. In this case the "No" literally means that none of the attributes have an "Up" or "Down" as a value for that particular patient. The list of unique values is: No, Ch Given that "No" definitely means something, There are no missing values for this attribute.

This also makes me feel like the other "No" values should be taken as values with meaning.
In this case there are no missing values for all of those drug attributes.

The list of unique values for attribute diabetesMed is: No, Yes There are no missing values for this attribute because the "No" literally means that the patient was not taking any of the diabetes-related drugs in this data.

The list of unique values for the attribute readmitted is: NO, >30, <30 There are no missing values for this data.

Problem 3

```
# Compute the percentage of patients admitted from the emergency room
# According to the ID_mapping_data, the number 7 corresponds to emeregency room
# for the addmission_source_id attribute.
percent_ER <- sum(diabetic_data$admission_source_id == 7)/length(diabetic_data$patient_nbr) * 100
```

NA % of total patients are admitted from the ER.

```
# Compute the probability of patients with discharge status of expired, given
# that they were admitted from the emergency room.
# According to the ID_mapping_data, the number 11 corresponds to expired
# for the discharge_disposition_id attribute.
prob_expired_given_ER <- sum(diabetic_data$admission_source_id == 7 & diabetic_data$discharge_dispositi
```

Given a patient is admitted from the emergency room, the probablility that their discharge status will be "expired" is NA %.

Problem 4

```
# compute the frequencies of each admission status
freq <- vector()
freq[1] <- sum(diabetic_data$admission_type_id == 1)
freq[2] <- sum(diabetic_data$admission_type_id == 2)
freq[3] <- sum(diabetic_data$admission_type_id == 3)
freq[4] <- sum(diabetic_data$admission_type_id == 4)
freq[5] <- sum(diabetic_data$admission_type_id == 5)
freq[6] <- sum(diabetic_data$admission_type_id == 6)
freq[7] <- sum(diabetic_data$admission_type_id == 7)
freq[8] <- sum(diabetic_data$admission_type_id == 8)
```

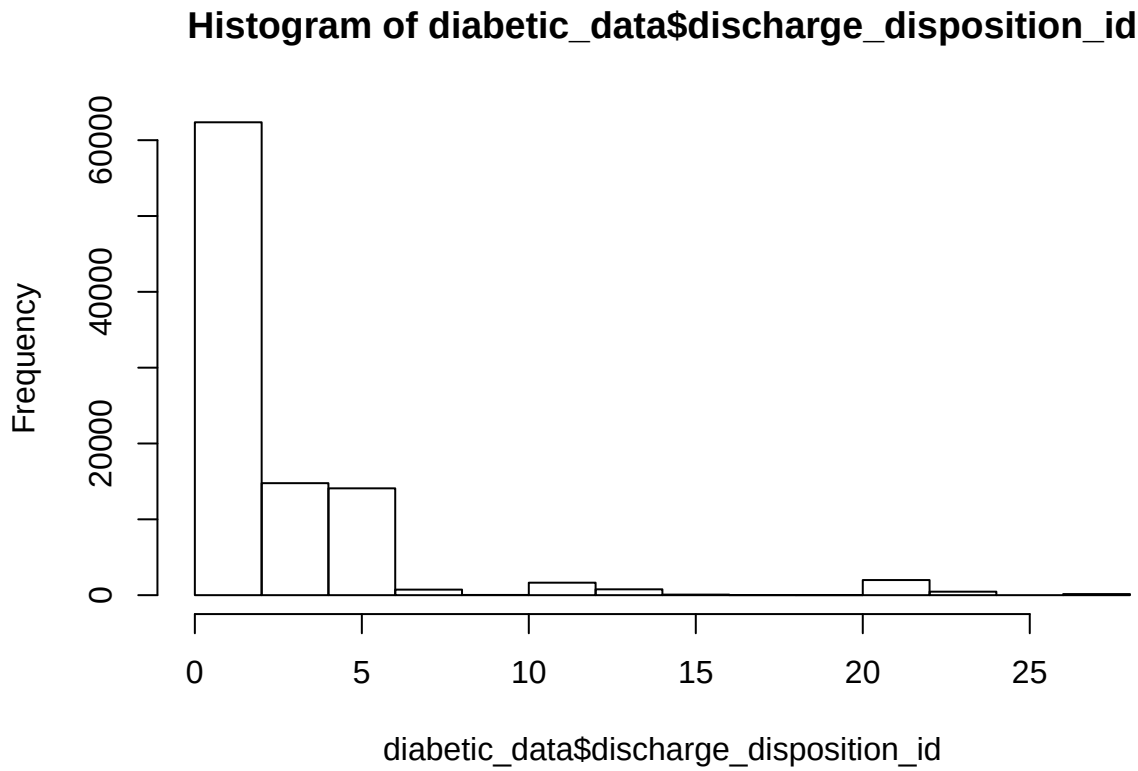
```r
freq_matrix <- matrix(freq, ncol = 1, byrow = TRUE)
colnames(freq_matrix) <- c("Frequency of Admission Status")
rownames(freq_matrix) <- c("Emergency", "Urgent", "Elective", "Newborn", "Not Available", "NULL",
"Trauma Center", "Not Mapped")
table_freq <- as.table(freq_matrix)
```

The following table shows the frequency of each admission status:

```
##               Frequency of Admission Status
## Emergency
## Urgent
## Elective
## Newborn
## Not Available
## NULL
## Trauma Center
## Not Mapped
```

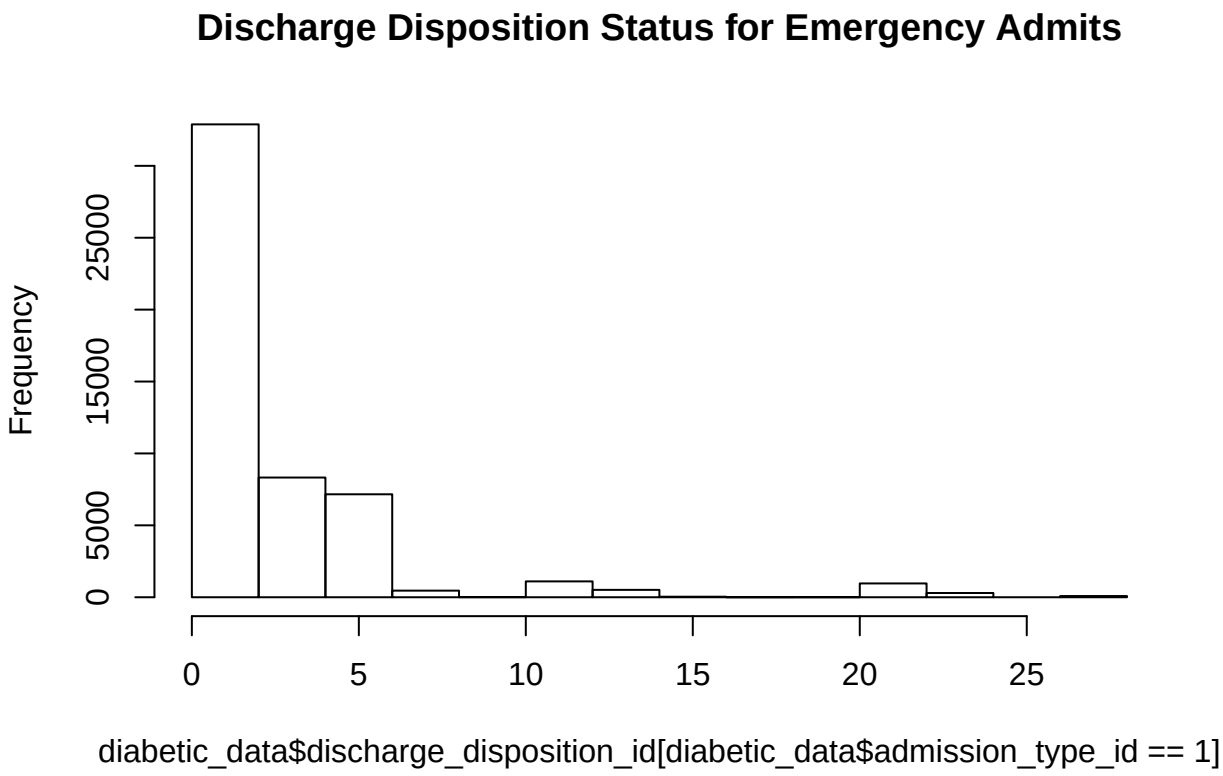
The most frequent admission status is “Emergency”.

I will now find the most frequent discharge status through another, much easier method, plotting a histogram.



The discharge status with the highest frequency is Discharged to home.

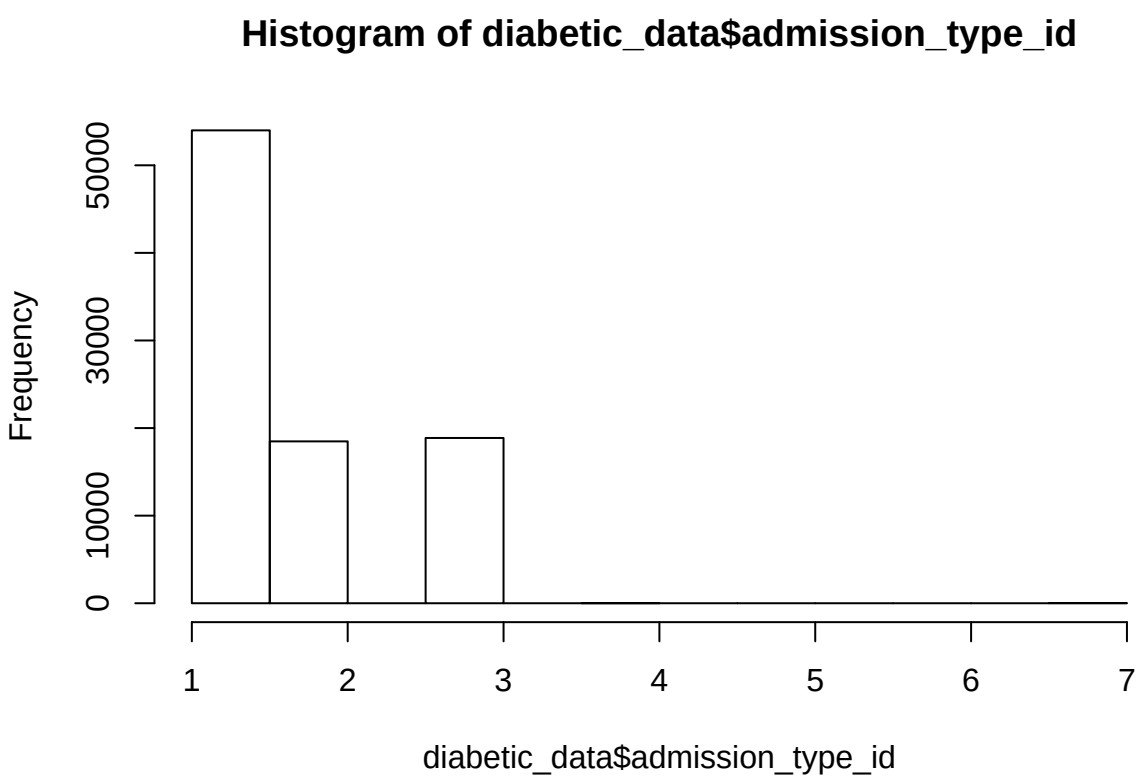
The folowing histogram is the frequency of discharge statuses for the most frequent admission status(i.e. Emergency):



The discharge status with the highest frequency for all patients that had an admission status of Emergency is still Discharged to home.

Problem 5

To characterize the distribution of the admission_type_id, I will plot a histogram:



We can't characterize a categorical distribution like we would somehthing that is continuous or at least ordered in any reasonable way.

All I can really say is that greater than half of the total admissions are by Emergency, and Urgent and Elective admissions each account for roughly a sixth of all admissions. Other admission types are relatively rare.

Exercise 4

Consider the following prompt: To what extent is it possible to predict discharge status?

I the use R tools/code to give a preliminary assessment of the quality of the data for a colleague asked to work on the same prompt.

I already calculated the percent of missing values for discharge_status_id. 4.5987854 % of the data is missing for this attribute. The converse of this analyss is to find the percentage of complete data for this attribute. I could simply subtract the percent missing from 100% and arrive at this answer, or I can use another function to explicitly calculate this number.

```
percent_complete_dis <- sum(complete.cases(diabetic_data$discharge_disposition_id))/length(diabetic_data
```

95.4012146 % of the total records have complete data for this attribute.

If I only wanted to predict discharge status based on this one attribute, I could be fairly sure that I am getting accurate prediction estimates.

I can compute one such estimate with the following code:

```
percent_home_discharge <- length(which(diabetic_data$discharge_disposition_id==1))/sum(complete.cases(di
```

62.041901 % percent of the admits (some patients more than once) were discharged to their homes,only considering non-missing data for discharge status. This can be an estimate of how likely they are to be discharged home after they are admitted. We can be very sure about this prediction because there is very little missing data.

Now if someone wanted to predict discharge status based on another attribute they would need to find the complete cases where both attributes were not-missing. For example, If I wanted to predict discharge status based on their admission status, I would need to access the completeness of the data for both of these attributes together.

```
percent_complete_dis_adm <- sum(complete.cases(diabetic_data$discharge_disposition_id)&complete.cases(di
```

86.1476328 % of the records have complete data for both admission status and discharge status.

If we were to make predictions of discharge status based on admission status, we could be fairly confident in our estimates based on the completeness of the data for these attributes.

The article mentioned that some of the records had multiple admissions for the same patient, and since these observations would be considered statistically dependent only one observation per patient shoud be considered in any regression model used for such a prediction. This would further limit our data to even lower than the 86% mark when actually using the data.