Project

B203349

2022-12-11

Contents

Click here to link to my presentation

Loading the required packages required for this script

```
library(sparklyr)
library(dplyr)
library(ggplot2)
library(cowplot)
library(knitr)
library(kableExtra)
library(tidyverse)
library(lubridate)
library(ggplot2)
library(dbplot)
library(janitor)
library(broom)
library(formatR)
```

Connecting to Spark in local mode

```
sc = spark_connect(master = 'local')
```

Loading datasets

Data

The **Diabetes 130-US hospitals for years 1999-2008 Data Set** is an extract representing 10 years (1999–2008) of clinical care at 130 hospitals and integrated delivery networks throughout the United States. The dataset was compiled Strack et al and is in .csv format.

Read .csv downloaded from and create dataframe

```
diabetic_data = spark_read_csv(sc,
    "/home/jovyan/Matt/diabetes_readmissions/RawData/diabetic_data.csv")
```

Exploratory analysis

Summary of data

The dataset contains over 50 variables, broadly these describe: patient demographics, patient diagnoses, admission and discharge dispositions, length of stay, blood glucose levels, medication types and changes, and readmission data. Furthermore, procedures, medications, and outpatient / inpatient / ED visits are all also quantified. The full list of variables is outlined below:

```
glimpse(diabetic_data)
```

```
## Rows: ??
## Columns: 50
## Database: spark_connection
```

```
## $ encounter_id             <int> 2278392, 149190, 64410, 500364, 16680, 35754,~
## $ patient_nbr              <int> 8222157, 55629189, 86047875, 82442376, 425192~
## $ race                     <chr> "Caucasian", "Caucasian", "AfricanAmerican", ~
## $ gender                   <chr> "Female", "Female", "Female", "Male", "Male",~
## $ age                      <chr> "[0-10)", "[10-20)", "[20-30)", "[30-40)", "[~
## $ weight                   <chr> "?", "?", "?", "?", "?", "?", "?", "?", "?", ~
## $ admission_type_id        <int> 6, 1, 1, 1, 1, 2, 3, 1, 2, 3, 1, 2, 1, 1, 3, ~
## $ discharge_disposition_id <int> 25, 1, 1, 1, 1, 1, 1, 1, 1, 3, 1, 1, 3, 6, 1,~
## $ admission_source_id      <int> 1, 7, 7, 7, 7, 2, 2, 7, 4, 4, 7, 4, 7, 7, 2, ~
## $ time_in_hospital         <int> 1, 3, 2, 2, 1, 3, 4, 5, 13, 12, 9, 7, 7, 10, ~
## $ payer_code               <chr> "?", "?", "?", "?", "?", "?", "?", "?", "?", ~
## $ medical_specialty        <chr> "Pediatrics-Endocrinology", "?", "?", "?", "?~
## $ num_lab_procedures       <int> 41, 59, 11, 44, 51, 31, 70, 73, 68, 33, 47, 6~
## $ num_procedures           <int> 0, 0, 5, 1, 0, 6, 1, 0, 2, 3, 2, 0, 0, 1, 5, ~
## $ num_medications          <int> 1, 18, 13, 16, 8, 16, 21, 12, 28, 18, 17, 11,~
## $ number_outpatient        <int> 0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ number_emergency         <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, ~
## $ number_inpatient         <int> 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ diag_1                   <chr> "250.83", "276", "648", "8", "197", "414", "4~
## $ diag_2                   <chr> "?", "250.01", "250", "250.43", "157", "411",~
## $ diag_3                   <chr> "?", "255", "V27", "403", "250", "250", "V45"~
## $ number_diagnoses         <int> 1, 9, 6, 7, 5, 9, 7, 8, 8, 8, 9, 7, 8, 8, 8, ~
## $ max_glu_serum            <chr> "None", "None", "None", "None", "None", "None~
## $ A1Cresult                <chr> "None", "None", "None", "None", "None", "None~
## $ metformin                <chr> "No", "No", "No", "No", "No", "No", "Steady",~
## $ repaglinide              <chr> "No", "No", "No", "No", "No", "No", "No", "No~
## $ nateglinide              <chr> "No", "No", "No", "No", "No", "No", "No", "No~
## $ chlorpropamide           <chr> "No", "No", "No", "No", "No", "No", "No", "No~
## $ glimepiride              <chr> "No", "No", "No", "No", "No", "No", "Steady",~
## $ acetohexamide            <chr> "No", "No", "No", "No", "No", "No", "No", "No~
## $ glipizide                <chr> "No", "No", "Steady", "No", "Steady", "No", "~
## $ glyburide                <chr> "No", "No", "No", "No", "No", "No", "No", "St~
## $ tolbutamide              <chr> "No", "No", "No", "No", "No", "No", "No", "No~
## $ pioglitazone             <chr> "No", "No", "No", "No", "No", "No", "No", "No~
## $ rosiglitazone            <chr> "No", "No", "No", "No", "No", "No", "No", "No~
## $ acarbose                 <chr> "No", "No", "No", "No", "No", "No", "No", "No~
## $ miglitol                 <chr> "No", "No", "No", "No", "No", "No", "No", "No~
## $ troglitazone             <chr> "No", "No", "No", "No", "No", "No", "No", "No~
## $ tolazamide               <chr> "No", "No", "No", "No", "No", "No", "No", "No~
## $ examide                  <chr> "No", "No", "No", "No", "No", "No", "No", "No~
## $ citoglipton              <chr> "No", "No", "No", "No", "No", "No", "No", "No~
## $ insulin                  <chr> "No", "Up", "No", "Up", "Steady", "Steady", "~
## $ glyburidemetformin       <chr> "No", "No", "No", "No", "No", "No", "No", "No~
## $ glipizidemetformin       <chr> "No", "No", "No", "No", "No", "No", "No", "No~
## $ glimepiridepioglitazone  <chr> "No", "No", "No", "No", "No", "No", "No", "No~
## $ metforminrosiglitazone   <chr> "No", "No", "No", "No", "No", "No", "No", "No~
## $ metforminpioglitazone    <chr> "No", "No", "No", "No", "No", "No", "No", "No~
## $ change                   <chr> "No", "Ch", "No", "Ch", "Ch", "No", "Ch", "No~
## $ diabetesMed              <chr> "No", "Yes", "Yes", "Yes", "Yes", "Yes", "Yes~
## $ readmitted               <chr> "NO", ">30", "NO", "NO", "NO", ">30", "NO", "~
```

Data cleaning, exploration, and feature engineering

Conversion of '?' to NA

From review of the dataframe is a apparent that '?' is used instead of NA. To faciliate data cleaning and wrangling, '?' is switched to NA.

```
#replace '?' with NA
diabetic_data <- diabetic_data %>%
  mutate(across(where(is.character), ~na_if(., "?")))
```

Duplicate patients

```
# total number of patient encounters
total_number_of_patient_encounters <- pull(diabetic_data,
    patient_nbr) %>%
    length()

# number of patients with repeat encounters
number_of_patient_with_repeat_encounters <- diabetic_data %>%
    group_by(patient_nbr) %>%
    filter(n() > 1) %>%
    tally() %>%
    sdf_nrow()

# number of patient encounters that can be
# classified as 'repeat'
number_of_patient_classed_as_repeat <- diabetic_data %>%
    group_by(patient_nbr) %>%
    filter(n() > 1) %>%
    sdf_nrow()

patient_encounters_table <- data.frame(n = c(total_number_of_patient_encounters,
    number_of_patient_classed_as_repeat, number_of_patient_with_repeat_encounters))

row.names(patient_encounters_table) <- c("Total number of patient encounters:",
    "Number of patient encounters that can be classified as 'repeat':",
    "Number of patients with repeat encounters:")

kable(patient_encounters_table, caption = "Summary of patient encounters",
    digits = 3, format.args = list(big.mark = ",",
        scientific = FALSE)) %>%
    kable_styling(latex_options = "HOLD_position")
```

Table 1: Summary of patient encounters

	n
Total number of patient encounters:	101,766
Number of patient encounters that can be classified as 'repeat':	47,021
Number of patients with repeat encounters:	16,773

As outlined above. The dataset contains 16,773 patients with multiple admissions (accounting for 47,021 of the overall observed admissions).

Removing duplicate patients

It cannot be assumed that the reason for these repeat admissions are independent. Statistical independence between observations is an assumption in some predictive models e.g. logistic regression. Therefore the dataset was cleaned to include only the **initial encounter** of repeat patients using the following code:

```
#group by patient number then select only the earliest patient encounter
diabetic_data <- diabetic_data %>%
  group_by(patient_nbr) %>%
  slice_min(encounter_id) %>% #slice_min selects the rows with lowest values
  ungroup()
```

Once this is done `patient_nbr` and `ecnounter_id` are redundant so these columns are removed

```
diabetic_data <- select(diabetic_data, -c(patient_nbr, encounter_id))
```

Missing variables

```
#count the number of  NAs per column
NA_count <- diabetic_data %>%
  summarise_all(~sum(as.integer(is.na(.)))) %>%
  collect()
```

```
## Warning: Missing values are always removed in SQL aggregation functions.
## Use `na.rm = TRUE` to silence this warning
## This warning is displayed once every 8 hours.
```

```
#transpose dataframe (convert from wide to long)
NA_count <- t(NA_count)

#rename column 1 to missing_values
colnames(NA_count)[1] ="missing_values"

kable(
  NA_count,
  caption = "Total of missing observations per feature",
  digits = 3,
  format.args = list(
    big.mark = ",",
    scientific = FALSE)
  ) %>%
kable_styling(latex_options = "HOLD_position")
```

Table 2: Total of missing observations per feature

	missing_values
race	1,948
gender	0
age	0
weight	68,665
admission_type_id	0
discharge_disposition_id	0
admission_source_id	0
time_in_hospital	0
payer_code	31,043
medical_specialty	34,477
num_lab_procedures	0
num_procedures	0
num_medications	0
number_outpatient	0
number_emergency	0
number_inpatient	0
diag_1	11
diag_2	294
diag_3	1,225
number_diagnoses	0
max_glu_serum	0
A1Cresult	0
metformin	0
repaglinide	0
nateglinide	0
chlorpropamide	0
glimepiride	0
acetohexamide	0
glipizide	0
glyburide	0
tolbutamide	0
pioglitazone	0
rosiglitazone	0
acarbose	0
miglitol	0
troglitazone	0
tolazamide	0
examide	0
citoglipton	0
insulin	0
glyburidemetformin	0
glipizidemetformin	0
glimepiridepioglitazone	0
metforminrosiglitazone	0
metforminpioglitazone	0
change	0
diabetesMed	0
readmitted	0

From the table above it is clear that `weight`, `payer_code` and `medical_speciality` have a significant number of missing variables. The exact percentage of missing variables is further explored below:

```
missing_medical_specialty <- diabetic_data %>%
  count(medical_specialty) %>%
  mutate(percent_missing = ((n / sum(n))*100)) %>%
  mutate(percent_missing = round(percent_missing, 2)) %>%
  filter(is.na(medical_specialty))%>%
  pull(percent_missing)

missing_payer_code <- diabetic_data %>%
  count(payer_code) %>%
  mutate(percent_missing = ((n / sum(n))*100)) %>%
  mutate(percent_missing = round(percent_missing, 2)) %>%
  filter(is.na(payer_code))%>%
  pull(percent_missing)

missing_weight <- diabetic_data %>%
  count(weight) %>%
  mutate(percent_missing = ((n / sum(n))*100)) %>%
  mutate(percent_missing = round(percent_missing, 2)) %>%
  filter(is.na(weight)) %>%
  pull(percent_missing)

missing_variables_table <- data.frame( percent = c(missing_medical_specialty, missing_payer_code, missi
row.names(missing_variables_table) <- c("medical_specialty", "payer_code", "weight" )
kable(missing_variables_table, caption = "Variables with a high proportion of missing values", digits =
  scientific = FALSE))%>%
kable_styling(latex_options = "HOLD_position")
```

Table 3: Variables with a high proportion of missing values

	percent
medical_specialty	48.21
payer_code	43.41
weight	96.01

Given the amount of missing values, `weight` and `payer_code` are removed from inclusion in further analysis. Although `medical_speciality` also has a high rate of missing values, this variable was retained as it was postulated that it may be very relevant to the further analysis.

```
diabetic_data <- select(diabetic_data, -c(weight, payer_code))
```

Exploration of numerical variables

Summary stats numerical variables

```
summary_stats_num_var <- sdf_describe(diabetic_data,
    cols = c("time_in_hospital", "num_medications",
        "number_inpatient", "num_lab_procedures", "number_outpatient",
        "number_diagnoses", "num_procedures", "number_emergency")) %>%
    collect()
```

```
kable(t(summary_stats_num_var), caption = "Summary statistics for numerical variables",
    digits = 3, format.args = list(big.mark = ",",
        scientific = FALSE)) %>%
    kable_styling(latex_options = "scale_down") %>%
    kable_styling(latex_options = "HOLD_position")
```

Table 4: Summary statistics for numerical variables

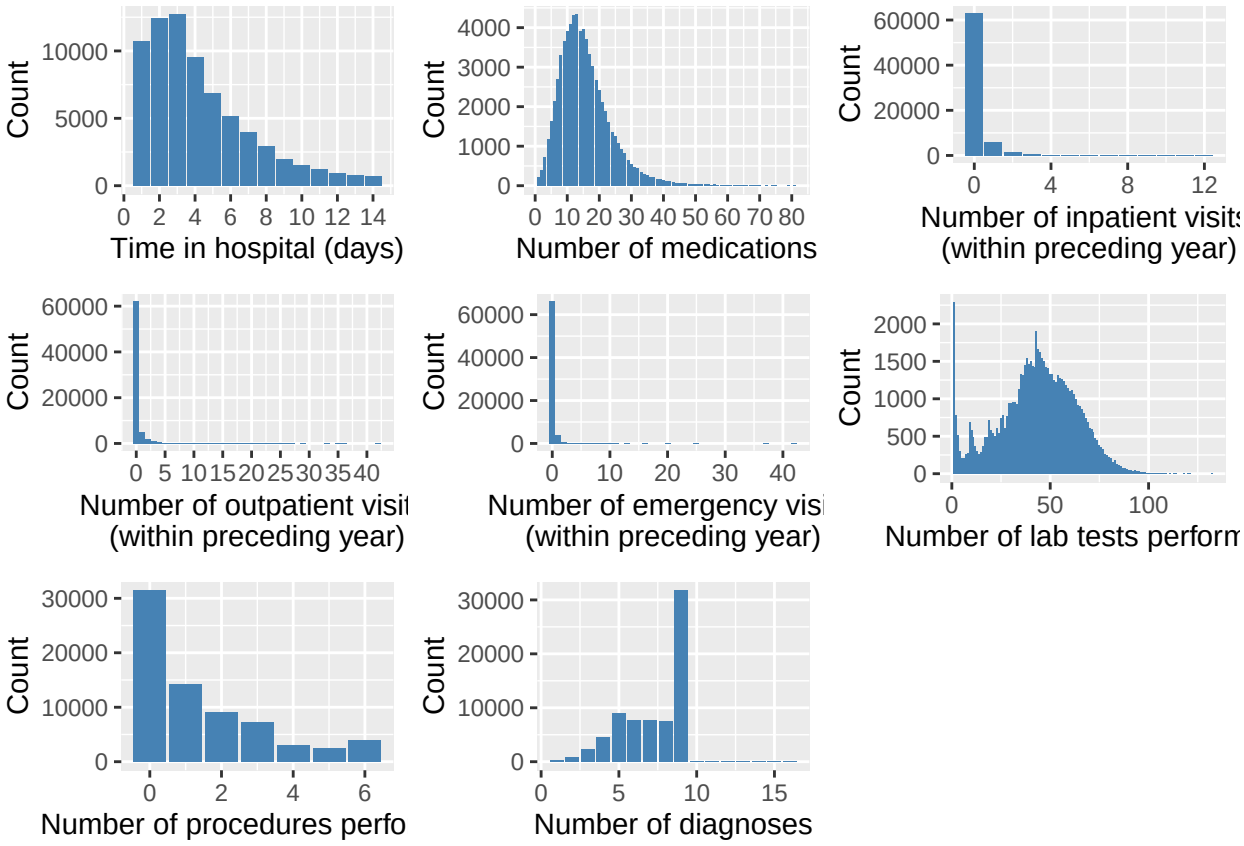
summary	count	mean	stddev	min	max
time_in_hospital	71518	4.289130009228446	2.949209936640064	1	14
num_medications	71518	15.70502530831399	8.311162711543068	1	81
number_inpatient	71518	0.1778293576442294	0.6037895326226642	0	12
num_lab_procedures	71518	43.07547750216729	19.952337943882533	1	132
number_outpatient	71518	0.2800693531698314	1.0689566542163038	0	42
number_diagnoses	71518	7.245700383120333	1.9946744288895086	1	16
num_procedures	71518	1.4305769176990408	1.759863962227284	0	6
number_emergency	71518	0.10354036745994015	0.5091865990151034	0	42

Visualisation of numerical variables

```
#create summary groups of numerical variable to collect and use in ggplot
time_in_hospital_group = diabetic_data %>%
  count(time_in_hospital) %>%
  arrange(time_in_hospital) %>%
  collect()

num_medications_group = diabetic_data %>%
  count(num_medications) %>%
  arrange(num_medications) %>%
  collect()

number_inpatient_group = diabetic_data %>%
  count(number_inpatient) %>%
  arrange(number_inpatient) %>%
  collect()

num_lab_procedures_group = diabetic_data %>%
  count(num_lab_procedures) %>%
  arrange(num_lab_procedures) %>%
  collect()

number_outpatient_group = diabetic_data %>%
  count(number_outpatient) %>%
  arrange(number_outpatient) %>%
  collect()

number_diagnoses_group = diabetic_data %>%
  count(number_diagnoses) %>%
  arrange(number_diagnoses) %>%
  collect()
```

```
num_procedures_group = diabetic_data %>%
  count(num_procedures) %>%
  arrange(num_procedures) %>%
  collect()

number_emergency_group = diabetic_data %>%
  count(number_emergency) %>%
  arrange(number_emergency) %>%
  collect()

#create plots
time_in_hospital_plot <-
  ggplot(aes(as.numeric(time_in_hospital), n), data = time_in_hospital_group) +
  geom_col(fill = 'SteelBlue') +
  scale_x_continuous(breaks=seq(0, 20, 2))  +
  xlab('Time in hospital (days)') +
  ylab('Count')

num_medications_plot <-
  ggplot(aes(as.numeric(num_medications), n), data = num_medications_group) +
  geom_col(fill = 'SteelBlue') +
  scale_x_continuous(breaks=seq(0, 80, 10))  +
  xlab('Number of medications') +
  ylab('Count')

number_inpatient_plot <-
  ggplot(aes(as.numeric(number_inpatient), n), data = number_inpatient_group) +
  geom_col(fill = 'SteelBlue') +
  #scale_x_continuous(breaks=seq(0, 15, 5))  +
  xlab('Number of inpatient visits \n(within preceding year)') +
  ylab('Count')

number_outpatient_plot <-
  ggplot(aes(as.numeric(number_outpatient), n), data = number_outpatient_group) +
  geom_col(fill = 'SteelBlue') +
  scale_x_continuous(breaks=seq(0, 40, 5))  +
  xlab('Number of outpatient visits \n(within preceding year)') +
  ylab('Count')

number_emergency_plot <-
  ggplot(aes(as.numeric(number_emergency), n), data = number_emergency_group) +
  geom_col(fill = 'SteelBlue') +
  xlab('Number of emergency visits \n(within preceding year)') +
  ylab('Count')

num_lab_procedures_plot <-
  ggplot(aes(as.numeric(num_lab_procedures), n), data = num_lab_procedures_group) +
  geom_col(fill = 'SteelBlue') +
  xlab('Number of lab tests performed') +
  ylab('Count')

num_procedures_plot <-
  ggplot(aes(as.numeric(num_procedures), n), data = num_procedures_group) +
```

```
  geom_col(fill = 'SteelBlue') +
  xlab('Number of procedures performed') +
  ylab('Count')

number_diagnoses_plot <-
  ggplot(aes(as.numeric(number_diagnoses), n), data = number_diagnoses_group) +
  geom_col(fill = 'SteelBlue') +
  xlab('Number of diagnoses') +
  ylab('Count')

#combine plots into one grid
plot_grid(time_in_hospital_plot,
          num_medications_plot,
          number_inpatient_plot,
          number_outpatient_plot,
          number_emergency_plot,
          num_lab_procedures_plot,
          num_procedures_plot,
          number_diagnoses_plot)
```

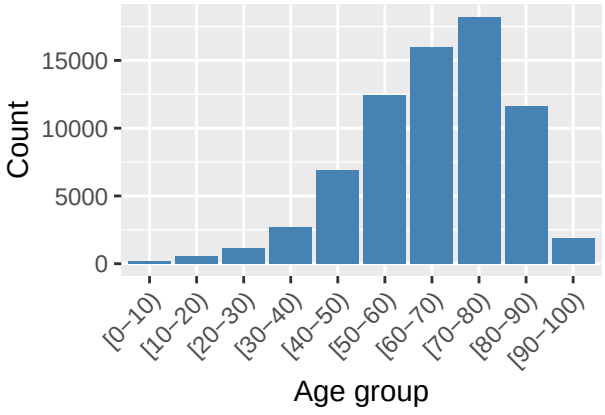


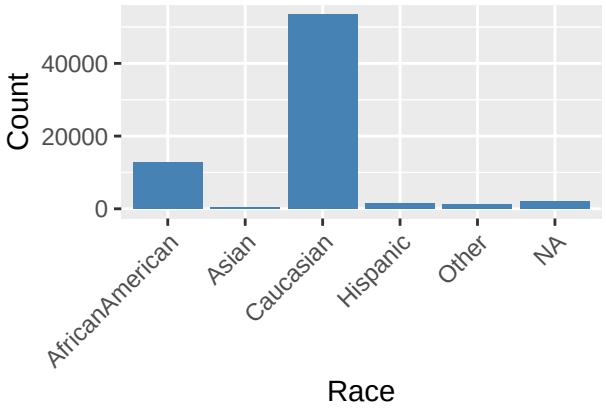
Exploration and feature engineering of categorical variables

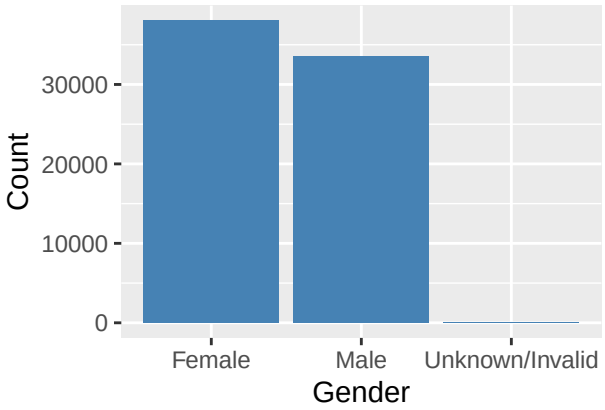
Age, Race, and gender

Visualisation of Age, Race, and gender

```
# Data manipulations are done first using spark and collected
age_group = diabetic_data %>%
  count(age) %>%
  arrange(age) %>%
  collect()

race_group = diabetic_data %>%
  count(race) %>%
  arrange(race) %>%
  collect()

gender_group = diabetic_data %>%
  count(gender) %>%
  arrange(gender) %>%
  collect()


#plots created with ggplot
age_plot <-
  ggplot(aes(as.factor(age), n), data = age_group) +
  geom_col(fill = 'SteelBlue') +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  xlab('Age group') +
  ylab('Count')

race_plot <-
  ggplot(aes(as.factor(race), n), data = race_group) +
  geom_col(fill = 'SteelBlue') +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  xlab('Race') +
  ylab('Count')

gender_plot <-
  ggplot(aes(as.factor(gender), n), data = gender_group) +
  geom_col(fill = 'SteelBlue') +
  xlab('Gender') +
  ylab('Count')


plot_grid(age_plot, race_plot, gender_plot)
```







Feature engineering of 'Age'

Age is converted to an ordinal scale, using the central age from each category. i.e. patients classed as age '[10-20)' are given the value.

```
diabetic_data <- diabetic_data %>%
  mutate(
    age_contin = case_when(
      age == '[0-10)'    ~ 5,
      age == '[10-20)'   ~ 15,
      age == '[20-30)'   ~ 25,
      age == '[30-40)'   ~ 35,
      age == '[40-50)'   ~ 45,
      age == '[50-60)'   ~ 55,
      age == '[60-70)'   ~ 65,
      age == '[70-80)'   ~ 75,
      age == '[80-90)'   ~ 85,
      age == '[90-100)'  ~ 95
    ))
```

Feature engineering of 'Race'

Race is one hot encoded

```
diabetic_data <- diabetic_data %>%
  mutate(
    unknown_race = ifelse(is.na(race), 1,0),
```

```
    asian = ifelse(race == 'Asian' & !is.na(race), 1,0),
    african_american = ifelse(race == 'AfricanAmerican' & !is.na(race), 1,0),
    caucasian = ifelse(race == 'Caucasian' & !is.na(race), 1,0),
    hispanic = ifelse(race == 'Hispanic' & !is.na(race), 1,0),
    other = ifelse(race == 'Other' & !is.na(race), 1,0),
  )
```

Feature engineering of 'Gender'

Gender is on hot encoded

```
diabetic_data <- diabetic_data %>%
  mutate(
    female = ifelse(gender == 'Female', 1,0),
    gender_unknown_invalid = ifelse(gender == 'Unknown/Invalid', 1,0),
    male = ifelse(gender == 'Male', 1,0)
  )
```

Readmissions

View the breakdown of readmissions

```
diabetic_data %>%
  count(readmitted) %>%
kable(
  caption = "Summary of readmitted feature",
  digits = 3,
  format.args = list(
    big.mark = ",",
    scientific = FALSE)
  ) %>%
  kable_styling(latex_options = "HOLD_position")
```

Table 5: Summary of readmitted feature

readmitted	n
<30	6,293
NO	42,985
>30	22,240

Feature engineering of readmission data

A variable (`early_readmission`) is created that classifies those readmitted within 30 days and those not, as this is the specific question posed by the challenge.

```
## create a new column with readmission <30
diabetic_data = mutate(diabetic_data, early_readmission = ifelse(readmitted ==
    "<30", 1, 0))

diabetic_data %>%
    group_by(early_readmission) %>%
    tally() %>%
    kable(caption = "Summary of early readmission feature",
        digits = 3, format.args = list(big.mark = ",",
```

```
            scientific = FALSE)) %>%
    kable_styling(latex_options = "HOLD_position")
```

Table 6: Summary of early readmission feature

early_readmission	n
1	6,293
0	65,225

Diagnoses

With regard to the diagnosis variables (`diag_1`, `diag_2`, and `diag_3`) initial exploration determines the exact number of different diagnosis categories.

```
# count of the number of unique primary Dx (698)
n_primary_dx <- diabetic_data %>%
    summarise(count = n_distinct(diag_1)) %>%
    pull()

# count of the number of unique secondary Dx
# (749)
n_secondary_dx <- diabetic_data %>%
    summarise(count = n_distinct(diag_2)) %>%
    pull()

# count of the number of unique tertiary Dx (759)
n_tertiary_dx <- diabetic_data %>%
    summarise(count = n_distinct(diag_3)) %>%
    pull()

diagnosis_n <- data.frame(n = c(n_primary_dx, n_secondary_dx,
    n_tertiary_dx))
row.names(diagnosis_n) <- c("Primary Dx", "Secondary Dx:",
    "Tertiary Dx")
kable(diagnosis_n, caption = "Total number of different diagnosis categories",
    digits = 3, format.args = list(big.mark = ",",
        scientific = FALSE)) %>%
    kable_styling(latex_options = "HOLD_position")
```

Table 7: Total number of different diagnosis categories

	n
Primary Dx	697
Secondary Dx:	726
Tertiary Dx	759

Feature engineering of Diagnosis variables

As illustrated above there are 698 unique primary diagnoses, 749 unique secondary diagnoses, and 759 tertiary diagnoses. Maintaining categorical variables with such high levels will be computationally expensive, diagnoses will be consolidated into more manageable levels. This is performed below using the ICD-9 code. Diagnoses have been consolidated according to the ICD-9 chapters, with each chapter essentially representing

a different bodily system. A seperate category for diabetes has also been created. It should be noted that this still results in 19 categories.

```
#consolidate according to ICD9 code
diabetic_data <- diabetic_data %>%
  mutate(
    diag_1_cat = case_when(
      rlike(diag_1, "250") ~ 'diabetes', #case_when works in order therefore 'diabetes' will be classed
      diag_1 >= 000 & diag_1 < 140 ~ 'infection',
      diag_1 >= 140 & diag_1 < 240 ~ 'neoplasms',
      diag_1 >= 240 & diag_1 < 280 ~ 'endo_metabolic_immunity',
      diag_1 >= 280 & diag_1 < 290 ~ 'haematology',
      diag_1 >= 290 & diag_1 < 320 ~ 'mental',
      diag_1 >= 320 & diag_1 < 390 ~ 'neurology',
      diag_1 >= 390 & diag_1 < 460 ~ 'circulatory',
      diag_1 >= 460 & diag_1 < 520 ~ 'respiratory',
      diag_1 >= 520 & diag_1 < 580 ~ 'digestive',
      diag_1 >= 580 & diag_1 < 630 ~ 'genitourinary',
      diag_1 >= 630 & diag_1 < 680 ~ 'preg_birth_puerperium',
      diag_1 >= 680 & diag_1 < 710 ~ 'dermatology',
      diag_1 >= 710 & diag_1 < 740 ~ 'musculoskeletal',
      diag_1 >= 740 & diag_1 < 760 ~ 'congenital',
      diag_1 >= 760 & diag_1 < 780 ~ 'perinatal',
      diag_1 >= 780 & diag_1 < 800 ~ 'ill_defined',
      diag_1 >= 800 & diag_1 < 1000 ~ 'injury_poisoning',
      rlike(diag_1, "V") | rlike(diag_1, "E") ~ 'supplementary',
      is.na(diag_1) ~ 'unknown_diag_1', #NA variables do not get special treatment
      TRUE ~ diag_1),
    diag_2_cat = case_when(
      rlike(diag_2, "250") ~ 'diabetes', #case_when work in order
      diag_2 >= 000 & diag_2 < 140 ~ 'infection',
      diag_2 >= 140 & diag_2 < 240 ~ 'neoplasms',
      diag_2 >= 240 & diag_2 < 280 ~ 'endo_metabolic_immunity',
      diag_2 >= 280 & diag_2 < 290 ~ 'haematology',
      diag_2 >= 290 & diag_2 < 320 ~ 'mental',
      diag_2 >= 320 & diag_2 < 390 ~ 'neurology',
      diag_2 >= 390 & diag_2 < 460 ~ 'circulatory',
      diag_2 >= 460 & diag_2 < 520 ~ 'respiratory',
      diag_2 >= 520 & diag_2 < 580 ~ 'digestive',
      diag_2 >= 580 & diag_2 < 630 ~ 'genitourinary',
      diag_2 >= 630 & diag_2 < 680 ~ 'preg_birth_puerperium',
      diag_2 >= 680 & diag_2 < 710 ~ 'dermatology',
      diag_2 >= 710 & diag_2 < 740 ~ 'musculoskeletal',
      diag_2 >= 740 & diag_2 < 760 ~ 'congenital',
      diag_2 >= 760 & diag_2 < 780 ~ 'perinatal',
      diag_2 >= 780 & diag_2 < 800 ~ 'ill_defined',
      diag_2 >= 800 & diag_2 < 1000 ~ 'injury_poisoning',
      rlike(diag_2, "V") | rlike(diag_2, "E") ~ 'supplementary',
      is.na(diag_2) ~ 'unknown_diag_2', #NA variable do not get special treatment
      TRUE ~ diag_2),
    diag_3_cat = case_when(
      rlike(diag_3, "250") ~ 'diabetes', #case_when work in order
      diag_3 >= 000 & diag_3 < 140 ~ 'infection',
```

```
      diag_3 >= 140 & diag_3 < 240 ~ 'neoplasms',
      diag_3 >= 240 & diag_3 < 280 ~ 'endo_metabolic_immunity',
      diag_3 >= 280 & diag_3 < 290 ~ 'haematology',
      diag_3 >= 290 & diag_3 < 320 ~ 'mental',
      diag_3 >= 320 & diag_3 < 390 ~ 'neurology',
      diag_3 >= 390 & diag_3 < 460 ~ 'circulatory',
      diag_3 >= 460 & diag_3 < 520 ~ 'respiratory',
      diag_3 >= 520 & diag_3 < 580 ~ 'digestive',
      diag_3 >= 580 & diag_3 < 630 ~ 'genitourinary',
      diag_3 >= 630 & diag_3 < 680 ~ 'preg_birth_puerperium',
      diag_3 >= 680 & diag_3 < 710 ~ 'dermatology',
      diag_3 >= 710 & diag_3 < 740 ~ 'musculoskeletal',
      diag_3 >= 740 & diag_3 < 760 ~ 'congenital',
      diag_3 >= 760 & diag_3 < 780 ~ 'perinatal',
      diag_3 >= 780 & diag_3 < 800 ~ 'ill_defined',
      diag_3 >= 800 & diag_3 < 1000 ~ 'injury_poisoning',
      rlike(diag_3, "V") | rlike(diag_3, "E") ~ 'supplementary',
      is.na(diag_3) ~ 'unknown_diag_3', #NA variable do not get special treatment
      TRUE ~ diag_3)
  )
```

Consolidation of `diag_1`, `diag_2`, and `diag_3` variables

One hot encoding of diagnnsis categories The diagnosis categories are then one hot encoded

```
# one_hot_encode diagnosis
diabetic_data <- diabetic_data %>%
    mutate(diag_1_infection = ifelse(diag_1_cat ==
        "infection" & !is.na(diag_1_cat), 1, 0), diag_1_neoplasms = ifelse(diag_1_cat ==
        "neoplasms" & !is.na(diag_1_cat), 1, 0), diag_1_endo_metabolic_immunity = ifelse(diag_1_cat ==
        "endo_metabolic_immunity" & !is.na(diag_1_cat),
        1, 0), diag_1_haematology = ifelse(diag_1_cat ==
        "haematology" & !is.na(diag_1_cat), 1, 0),
        diag_1_mental = ifelse(diag_1_cat == "mental" &
            !is.na(diag_1_cat), 1, 0), diag_1_neurology = ifelse(diag_1_cat ==
            "neurology" & !is.na(diag_1_cat), 1, 0),
        diag_1_circulatory = ifelse(diag_1_cat == "circulatory" &
            !is.na(diag_1_cat), 1, 0), diag_1_respiratory = ifelse(diag_1_cat ==
            "respiratory" & !is.na(diag_1_cat), 1,
            0), diag_1_digestive = ifelse(diag_1_cat ==
            "digestive" & !is.na(diag_1_cat), 1, 0),
        diag_1_genitourinary = ifelse(diag_1_cat ==
            "genitourinary" & !is.na(diag_1_cat), 1,
            0), diag_1_preg_birth_puerperium = ifelse(diag_1_cat ==
            "preg_birth_puerperium" & !is.na(diag_1_cat),
            1, 0), diag_1_dermatology = ifelse(diag_1_cat ==
            "dermatology" & !is.na(diag_1_cat), 1,
            0), diag_1_musculoskeletal = ifelse(diag_1_cat ==
            "musculoskeletal" & !is.na(diag_1_cat),
            1, 0), diag_1_congenital = ifelse(diag_1_cat ==
            "congenital" & !is.na(diag_1_cat), 1, 0),
        diag_1_perinatal = ifelse(diag_1_cat == "perinatal" &
            !is.na(diag_1_cat), 1, 0), diag_1_ill_defined = ifelse(diag_1_cat ==
            "ill_defined" & !is.na(diag_1_cat), 1,
```

```
        0), diag_1_injury_poisoning = ifelse(diag_1_cat ==
        "injury_poisoning" & !is.na(diag_1_cat),
        1, 0), diag_1_supplementary = ifelse(diag_1_cat ==
        "supplementary" & !is.na(diag_1_cat), 1,
        0), diag_1_diabetes = ifelse(diag_1_cat ==
        "diabetes" & !is.na(diag_1_cat), 1, 0),
    diag_1_unknown = ifelse(diag_1_cat == "unknown_diag_1" &
        !is.na(diag_1_cat), 1, 0), diag_2_infection = ifelse(diag_2_cat ==
        "infection" & !is.na(diag_2_cat), 1, 0),
    diag_2_neoplasms = ifelse(diag_2_cat == "neoplasms" &
        !is.na(diag_2_cat), 1, 0), diag_2_endo_metabolic_immunity = ifelse(diag_2_cat ==
        "endo_metabolic_immunity" & !is.na(diag_2_cat),
        1, 0), diag_2_haematology = ifelse(diag_2_cat ==
        "haematology" & !is.na(diag_2_cat), 1,
        0), diag_2_mental = ifelse(diag_2_cat ==
        "mental" & !is.na(diag_2_cat), 1, 0), diag_2_neurology = ifelse(diag_2_cat ==
        "neurology" & !is.na(diag_2_cat), 1, 0),
    diag_2_circulatory = ifelse(diag_2_cat == "circulatory" &
        !is.na(diag_2_cat), 1, 0), diag_2_respiratory = ifelse(diag_2_cat ==
        "respiratory" & !is.na(diag_2_cat), 1,
        0), diag_2_digestive = ifelse(diag_2_cat ==
        "digestive" & !is.na(diag_2_cat), 1, 0),
    diag_2_genitourinary = ifelse(diag_2_cat ==
        "genitourinary" & !is.na(diag_2_cat), 1,
        0), diag_2_preg_birth_puerperium = ifelse(diag_2_cat ==
        "preg_birth_puerperium" & !is.na(diag_2_cat),
        1, 0), diag_2_dermatology = ifelse(diag_2_cat ==
        "dermatology" & !is.na(diag_2_cat), 1,
        0), diag_2_musculoskeletal = ifelse(diag_2_cat ==
        "musculoskeletal" & !is.na(diag_2_cat),
        1, 0), diag_2_congenital = ifelse(diag_2_cat ==
        "congenital" & !is.na(diag_2_cat), 1, 0),
    diag_2_perinatal = ifelse(diag_2_cat == "perinatal" &
        !is.na(diag_2_cat), 1, 0), diag_2_ill_defined = ifelse(diag_2_cat ==
        "ill_defined" & !is.na(diag_2_cat), 1,
        0), diag_2_injury_poisoning = ifelse(diag_2_cat ==
        "injury_poisoning" & !is.na(diag_2_cat),
        1, 0), diag_2_supplementary = ifelse(diag_2_cat ==
        "supplementary" & !is.na(diag_2_cat), 1,
        0), diag_2_diabetes = ifelse(diag_2_cat ==
        "diabetes" & !is.na(diag_2_cat), 1, 0),
    diag_2_unknown = ifelse(diag_2_cat == "unknown_diag_2" &
        !is.na(diag_2_cat), 1, 0), diag_3_infection = ifelse(diag_3_cat ==
        "infection" & !is.na(diag_3_cat), 1, 0),
    diag_3_neoplasms = ifelse(diag_3_cat == "neoplasms" &
        !is.na(diag_3_cat), 1, 0), diag_3_endo_metabolic_immunity = ifelse(diag_3_cat ==
        "endo_metabolic_immunity" & !is.na(diag_3_cat),
        1, 0), diag_3_haematology = ifelse(diag_3_cat ==
        "haematology" & !is.na(diag_3_cat), 1,
        0), diag_3_mental = ifelse(diag_3_cat ==
        "mental" & !is.na(diag_3_cat), 1, 0), diag_3_neurology = ifelse(diag_3_cat ==
        "neurology" & !is.na(diag_3_cat), 1, 0),
    diag_3_circulatory = ifelse(diag_3_cat == "circulatory" &
```

```
            !is.na(diag_3_cat), 1, 0), diag_3_respiratory = ifelse(diag_3_cat ==
            "respiratory" & !is.na(diag_3_cat), 1,
            0), diag_3_digestive = ifelse(diag_3_cat ==
            "digestive" & !is.na(diag_3_cat), 1, 0),
        diag_3_genitourinary = ifelse(diag_3_cat ==
            "genitourinary" & !is.na(diag_3_cat), 1,
            0), diag_3_preg_birth_puerperium = ifelse(diag_3_cat ==
            "preg_birth_puerperium" & !is.na(diag_3_cat),
            1, 0), diag_3_dermatology = ifelse(diag_3_cat ==
            "dermatology" & !is.na(diag_3_cat), 1,
            0), diag_3_musculoskeletal = ifelse(diag_3_cat ==
            "musculoskeletal" & !is.na(diag_3_cat),
            1, 0), diag_3_congenital = ifelse(diag_3_cat ==
            "congenital" & !is.na(diag_3_cat), 1, 0),
        diag_3_perinatal = ifelse(diag_3_cat == "perinatal" &
            !is.na(diag_3_cat), 1, 0), diag_3_ill_defined = ifelse(diag_3_cat ==
            "ill_defined" & !is.na(diag_3_cat), 1,
            0), diag_3_injury_poisoning = ifelse(diag_3_cat ==
            "injury_poisoning" & !is.na(diag_3_cat),
            1, 0), diag_3_supplementary = ifelse(diag_3_cat ==
            "supplementary" & !is.na(diag_3_cat), 1,
            0), diag_3_diabetes = ifelse(diag_3_cat ==
            "diabetes" & !is.na(diag_3_cat), 1, 0),
        diag_3_unknown = ifelse(diag_1_cat == "unknown_diag_3" &
            !is.na(diag_3_cat), 1, 0), )
```

Blood sugars

Feature engineering of blood sugar variables

The blood sugar variable (i.e. `max_glu_serum` and `A1Cresult`) are one hot encoded

```
diabetic_data <- diabetic_data %>%
  mutate(
    max_glu_serum_none = ifelse(max_glu_serum == 'None', 1,0),
    max_glu_serum_norm = ifelse(max_glu_serum == 'Norm', 1,0),
    max_glu_serum_300 = ifelse(max_glu_serum == '>300', 1,0),
    max_glu_serum_200 = ifelse(max_glu_serum == '>200', 1,0),
    A1Cresult_none = ifelse(A1Cresult == 'None', 1,0),
    A1Cresult_norm = ifelse(A1Cresult == 'Norm', 1,0),
    A1Cresult_7 = ifelse(A1Cresult == '>7', 1,0),
    A1Cresult_8 = ifelse(A1Cresult == '>8', 1,0))
```

Exploration and feature engineering of `medical_speciality`

First the list of unique medical specialites is compiled along with the frequency of each observation

```
list_of_medical_specialty <- diabetic_data %>%
  group_by(medical_specialty) %>%
  tally() %>%
  mutate(percent = ((n / sum(n))*100)) %>%
  mutate(percent = round(percent, 2)) %>%
  arrange(desc(n)) %>%
  collect()
```

```
kable(list_of_medical_specialty,
      "latex", booktabs = TRUE, longtable = TRUE, caption = "List of medical specialites") %>%
  kable_styling(latex_options = c("hold_position", "repeat_header"))
```

Table 8: List of medical specialites

medical_specialty	n	percent
NA	34477	48.21
InternalMedicine	10919	15.27
Family/GeneralPractice	5118	7.16
Emergency/Trauma	4465	6.24
Cardiology	4266	5.96
Surgery-General	2221	3.11
Orthopedics	1134	1.59
Orthopedics-Reconstructive	1043	1.46
Radiologist	831	1.16
Nephrology	828	1.16
Pulmonology	653	0.91
Psychiatry	614	0.86
ObstetricsandGynecology	595	0.83
Urology	530	0.74
Surgery-Cardiovascular/Thoracic	497	0.69
Surgery-Neuro	409	0.57
Gastroenterology	398	0.56
Surgery-Vascular	362	0.51
Oncology	218	0.30
Pediatrics	196	0.27
PhysicalMedicineandRehabilitation	194	0.27
Neurology	168	0.23
Pediatrics-Endocrinology	147	0.21
Hematology/Oncology	122	0.17
Otolaryngology	110	0.15
Endocrinology	98	0.14
Surgery-Thoracic	92	0.13
Surgery-Cardiovascular	85	0.12
Pediatrics-CriticalCare	73	0.10
Podiatry	63	0.09
Gynecology	54	0.08
Psychology	53	0.07
Surgeon	40	0.06
Osteopath	38	0.05
Radiology	38	0.05
Hematology	37	0.05
Hospitalist	36	0.05
Ophthalmology	35	0.05
Surgery-Plastic	30	0.04
InfectiousDiseases	29	0.04
SurgicalSpecialty	26	0.04
Obsterics&Gynecology-GynecologicOnco	18	0.03
Obstetrics	17	0.02

Table 8: List of medical specialites *(continued)*

medical_specialty	n	percent
Anesthesiology-Pediatric	13	0.02
Surgery-Maxillofacial	10	0.01
Rheumatology	10	0.01
Surgery-Colon&Rectal	9	0.01
OutreachServices	9	0.01
PhysicianNotFound	8	0.01
Endocrinology-Metabolism	7	0.01
Cardiology-Pediatric	7	0.01
Pathology	7	0.01
Anesthesiology	7	0.01
Pediatrics-Neurology	7	0.01
AllergyandImmunology	6	0.01
Surgery-Pediatric	6	0.01
Pediatrics-Pulmonology	6	0.01
Psychiatry-Child/Adolescent	6	0.01
Dentistry	4	0.01
DCPTEAM	4	0.01
Pediatrics-EmergencyMedicine	3	0.00
Pediatrics-Hematology-Oncology	3	0.00
Psychiatry-Addictive	1	0.00
Speech	1	0.00
SportsMedicine	1	0.00
Dermatology	1	0.00
Resident	1	0.00
Surgery-PlasticwithinHeadandNeck	1	0.00
Neurophysiology	1	0.00
Proctology	1	0.00
Perinatology	1	0.00

Feature engineering of `medical_specialty`

Each category of `medical_speciality` is one hot encoded. I did not consolidate this group as getting a granular understanding of the influence that each group has on the readmission rate is important. As this can be directly fed back to the respective group to affect change.

```
# one_hot_encode medical_specialty
diabetic_data <- diabetic_data %>%
    mutate(Cardiology = ifelse(medical_specialty ==
        "Cardiology" & !is.na(medical_specialty), 1,
        0), ObstetricsandGynecology = ifelse(medical_specialty ==
        "ObstetricsandGynecology" & !is.na(medical_specialty),
        1, 0), Pediatrics = ifelse(medical_specialty ==
        "Pediatrics" & !is.na(medical_specialty), 1,
        0), SurgeryColonRectal = ifelse(medical_specialty ==
        "Surgery-Colon&Rectal" & !is.na(medical_specialty),
        1, 0), PediatricsCriticalCare = ifelse(medical_specialty ==
        "Pediatrics-CriticalCare" & !is.na(medical_specialty),
        1, 0), Anesthesiology_Pediatric = ifelse(medical_specialty ==
```

```
        "Anesthesiology-Pediatric" & !is.na(medical_specialty),
        1, 0), Ophthalmology = ifelse(medical_specialty ==
        "Ophthalmology" & !is.na(medical_specialty),
        1, 0), InfectiousDiseases = ifelse(medical_specialty ==
        "InfectiousDiseases" & !is.na(medical_specialty),
        1, 0), SurgeryMaxillofacial = ifelse(medical_specialty ==
        "Surgery-Maxillofacial" & !is.na(medical_specialty),
        1, 0), PsychiatryAddictive = ifelse(medical_specialty ==
        "Psychiatry-Addictive" & !is.na(medical_specialty),
        1, 0), SurgeryCardiovascular = ifelse(medical_specialty ==
        "Surgery-Cardiovascular" & !is.na(medical_specialty),
        1, 0), Speech = ifelse(medical_specialty ==
        "Speech" & !is.na(medical_specialty), 1, 0),
        Endocrinology_Metabolism = ifelse(medical_specialty ==
            "Endocrinology-Metabolism" & !is.na(medical_specialty),
            1, 0), FamilyGeneralPractice = ifelse(medical_specialty ==
            "Family/GeneralPractice" & !is.na(medical_specialty),
            1, 0), SurgeryGeneral = ifelse(medical_specialty ==
            "Surgery-General" & !is.na(medical_specialty),
            1, 0), Orthopedics = ifelse(medical_specialty ==
            "Orthopedics" & !is.na(medical_specialty),
            1, 0), EmergencyTrauma = ifelse(medical_specialty ==
            "Emergency/Trauma" & !is.na(medical_specialty),
            1, 0), HematologyOncology = ifelse(medical_specialty ==
            "Hematology/Oncology" & !is.na(medical_specialty),
            1, 0), Otolaryngology = ifelse(medical_specialty ==
            "Otolaryngology" & !is.na(medical_specialty),
            1, 0), Oncology = ifelse(medical_specialty ==
            "Oncology" & !is.na(medical_specialty),
            1, 0), SurgeryPediatric = ifelse(medical_specialty ==
            "Surgery-Pediatric" & !is.na(medical_specialty),
            1, 0), PediatricsEmergencyMedicine = ifelse(medical_specialty ==
            "Pediatrics-EmergencyMedicine" & !is.na(medical_specialty),
            1, 0), AllergyandImmunology = ifelse(medical_specialty ==
            "AllergyandImmunology" & !is.na(medical_specialty),
            1, 0), PediatricsInfectiousDiseases = ifelse(medical_specialty ==
            "Pediatrics-InfectiousDiseases" & !is.na(medical_specialty),
            1, 0), Osteopath = ifelse(medical_specialty ==
            "Osteopath" & !is.na(medical_specialty),
            1, 0), SurgicalSpecialty = ifelse(medical_specialty ==
            "SurgicalSpecialty" & !is.na(medical_specialty),
            1, 0), Dermatology = ifelse(medical_specialty ==
            "Dermatology" & !is.na(medical_specialty),
            1, 0), SportsMedicine = ifelse(medical_specialty ==
            "SportsMedicine" & !is.na(medical_specialty),
            1, 0), Resident = ifelse(medical_specialty ==
            "Resident" & !is.na(medical_specialty),
            1, 0), InternalMedicine = ifelse(medical_specialty ==
            "InternalMedicine" & !is.na(medical_specialty),
            1, 0), Gastroenterology = ifelse(medical_specialty ==
            "Gastroenterology" & !is.na(medical_specialty),
            1, 0), SurgeryCardiovascularThoracic = ifelse(medical_specialty ==
            "Surgery-Cardiovascular/Thoracic" & !is.na(medical_specialty),
```

```
        1, 0), Nephrology = ifelse(medical_specialty ==
        "Nephrology" & !is.na(medical_specialty),
        1, 0), OrthopedicsReconstructive = ifelse(medical_specialty ==
        "Orthopedics-Reconstructive" & !is.na(medical_specialty),
        1, 0), ObstericsGynecologyGynecologicOnco = ifelse(medical_specialty ==
        "Obsterics&Gynecology-GynecologicOnco" &
        !is.na(medical_specialty), 1, 0), Endocrinology = ifelse(medical_specialty ==
        "Endocrinology" & !is.na(medical_specialty),
        1, 0), Pediatrics_Pulmonology = ifelse(medical_specialty ==
        "Pediatrics-Pulmonology" & !is.na(medical_specialty),
        1, 0), Neurology = ifelse(medical_specialty ==
        "Neurology" & !is.na(medical_specialty),
        1, 0), Psychology = ifelse(medical_specialty ==
        "Psychology" & !is.na(medical_specialty),
        1, 0), Podiatry = ifelse(medical_specialty ==
        "Podiatry" & !is.na(medical_specialty),
        1, 0), Gynecology = ifelse(medical_specialty ==
        "Gynecology" & !is.na(medical_specialty),
        1, 0), SurgeryPlastic = ifelse(medical_specialty ==
        "Surgery-Plastic" & !is.na(medical_specialty),
        1, 0), SurgeryThoracic = ifelse(medical_specialty ==
        "Surgery-Thoracic" & !is.na(medical_specialty),
        1, 0), SurgeryPlasticwithinHeadandNeck = ifelse(medical_specialty ==
        "Surgery-PlasticwithinHeadandNeck" & !is.na(medical_specialty),
        1, 0), PhysicalMedicineandRehabilitation = ifelse(medical_specialty ==
        "PhysicalMedicineandRehabilitation" & !is.na(medical_specialty),
        1, 0), Rheumatology = ifelse(medical_specialty ==
        "Rheumatology" & !is.na(medical_specialty),
        1, 0), PediatricsAllergyandImmunology = ifelse(medical_specialty ==
        "Pediatrics-AllergyandImmunology" & !is.na(medical_specialty),
        1, 0), Surgeon = ifelse(medical_specialty ==
        "Surgeon" & !is.na(medical_specialty),
        1, 0), SurgeryVascular = ifelse(medical_specialty ==
        "Surgery-Vascular" & !is.na(medical_specialty),
        1, 0), Pathology = ifelse(medical_specialty ==
        "Pathology" & !is.na(medical_specialty),
        1, 0), Hospitalist = ifelse(medical_specialty ==
        "Hospitalist" & !is.na(medical_specialty),
        1, 0), OutreachServices = ifelse(medical_specialty ==
        "OutreachServices" & !is.na(medical_specialty),
        1, 0), CardiologyPediatric = ifelse(medical_specialty ==
        "Cardiology-Pediatric" & !is.na(medical_specialty),
        1, 0), Neurophysiology = ifelse(medical_specialty ==
        "Neurophysiology" & !is.na(medical_specialty),
        1, 0), PediatricsEndocrinology = ifelse(medical_specialty ==
        "Pediatrics-Endocrinology" & !is.na(medical_specialty),
        1, 0), Psychiatry = ifelse(medical_specialty ==
        "Psychiatry" & !is.na(medical_specialty),
        1, 0), Pulmonology = ifelse(medical_specialty ==
        "Pulmonology" & !is.na(medical_specialty),
        1, 0), SurgeryNeuro = ifelse(medical_specialty ==
        "Surgery-Neuro" & !is.na(medical_specialty),
        1, 0), Urology = ifelse(medical_specialty ==
```

```
            "Urology" & !is.na(medical_specialty),
            1, 0), PsychiatryChildAdolescent = ifelse(medical_specialty ==
            "Psychiatry-Child/Adolescent" & !is.na(medical_specialty),
            1, 0), Radiology = ifelse(medical_specialty ==
            "Radiology" & !is.na(medical_specialty),
            1, 0), PediatricsHematologyOncology = ifelse(medical_specialty ==
            "Pediatrics-Hematology-Oncology" & !is.na(medical_specialty),
            1, 0), PediatricsNeurology = ifelse(medical_specialty ==
            "Pediatrics-Neurology" & !is.na(medical_specialty),
            1, 0), Anesthesiology = ifelse(medical_specialty ==
            "Anesthesiology" & !is.na(medical_specialty),
            1, 0), Dentistry = ifelse(medical_specialty ==
            "Dentistry" & !is.na(medical_specialty),
            1, 0), PhysicianNotFound = ifelse(medical_specialty ==
            "PhysicianNotFound" & !is.na(medical_specialty),
            1, 0), Hematology = ifelse(medical_specialty ==
            "Hematology" & !is.na(medical_specialty),
            1, 0), Proctology = ifelse(medical_specialty ==
            "Proctology" & !is.na(medical_specialty),
            1, 0), Obstetrics = ifelse(medical_specialty ==
            "Obstetrics" & !is.na(medical_specialty),
            1, 0), Radiologist = ifelse(medical_specialty ==
            "Radiologist" & !is.na(medical_specialty),
            1, 0), Perinatology = ifelse(medical_specialty ==
            "Perinatology" & !is.na(medical_specialty),
            1, 0), DCPTEAM = ifelse(medical_specialty ==
            "DCPTEAM" & !is.na(medical_specialty),
            1, 0), medical_specialty_unkown = ifelse(is.na(medical_specialty),
            1, 0))
```

Admission type

Consolidation of `admission_type_id`

`admission_type_id` in consolidated into 4 categories. Using the IDs_mapping.csv file the numerical value for `admission_type_id` is exchanged for character / descriptive value

```
diabetic_data <- diabetic_data %>%
  mutate(
    admission_type_consolidated = case_when(
      admission_type_id %in% c(1) ~ 'emergency',
      admission_type_id %in% c(2) ~ 'urgent',
      admission_type_id %in%c(3) ~ 'elective',
      admission_type_id %in% c(4,5,6,7,8) ~ 'admisison_type_other',
      is.na(admission_type_id) ~ 'admisison_type_other', #NA variables do not get special treatment
      TRUE ~ admission_type_id))
```

One hot encoding of `admission_type_consolidated`

The `admission_type_consolidated` variable is then one hot encoded

```
diabetic_data <- diabetic_data %>%
    mutate(emergency = ifelse(admission_type_consolidated ==
        "emergency", 1, 0), urgent = ifelse(admission_type_consolidated ==
        "urgent", 1, 0), elective = ifelse(admission_type_consolidated ==
```

```
        "elective", 1, 0), admisison_type_other = ifelse(admission_type_consolidated ==
        "admisison_type_other", 1, 0))
```

Discharge disposition

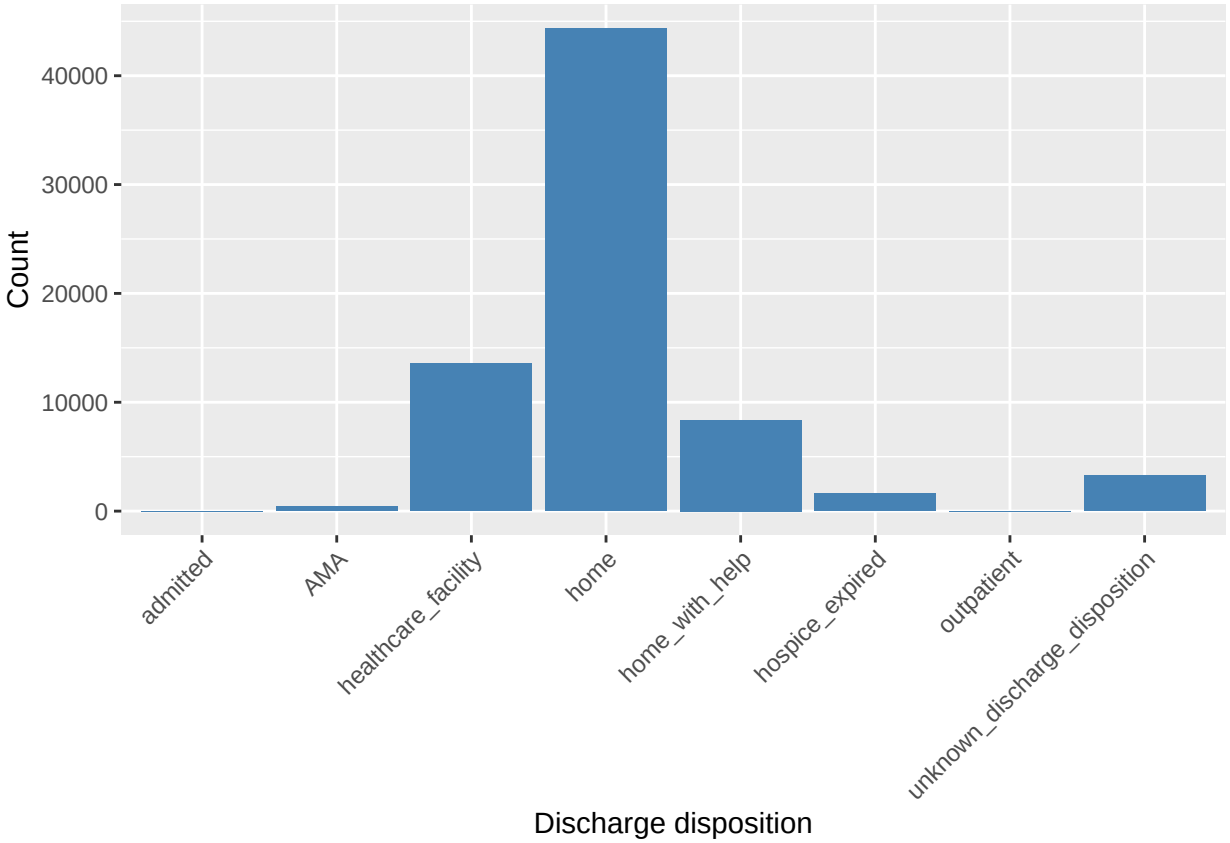
Consolidation of `discharge_disposition` variable

`discharge_disposition` in consolidated into 9 categories. Using the IDs_mapping.csv file the numerical value for `discharge_disposition` is exchanged for a character / descriptive value

```
diabetic_data <- diabetic_data %>%
    mutate(discharge_disposition_consolidated = case_when(discharge_disposition_id %in%
        c(1) ~ "home", discharge_disposition_id %in%
        c(2, 3, 4, 5, 10, 22, 23, 24, 30, 27, 28, 29) ~
        "healthcare_facility", discharge_disposition_id %in%
        c(6, 8) ~ "home_with_help", discharge_disposition_id %in%
        c(7) ~ "AMA", discharge_disposition_id %in%
        c(9) ~ "admitted", discharge_disposition_id %in%
        c(11, 13, 14, 15, 19, 20, 21) ~ "hospice_expired",
        discharge_disposition_id %in% c(12, 16, 17) ~
            "outpatient", discharge_disposition_id %in%
            c(18, 25, 26) ~ "unknown_discharge_disposition",
        is.na(discharge_disposition_id) ~ "unknown_discharge_disposition",
        TRUE ~ discharge_disposition_id))
```

Graphing of consolidated discharge dispositions

```
discharge_disposition_group = diabetic_data %>%
    count(discharge_disposition_consolidated) %>%
    arrange(discharge_disposition_consolidated) %>%
    collect()

discharge_disposition_plot <- ggplot(aes(as.factor(discharge_disposition_consolidated),
    n), data = discharge_disposition_group) + geom_col(fill = "SteelBlue") +
    theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
    xlab("Discharge disposition") + ylab("Count")

discharge_disposition_plot
```



Removal of patients with `hospice_expired` discharge disposition

It is likely that discharge destination could bias the model as certain destinations such as 'hospice', 'expired' will unlikely result in short-term readmission irrespective of the true predictive value of the other variables. An effective model will therefore have to account for this bias by removing patient whho had this discharge disposition.

```
diabetic_data <- diabetic_data %>%
  filter(discharge_disposition_consolidated!='hospice_expired')
```

One hot encoding of `discharge_disposition_consolidated`

The consolidated discharge disposition variable (`discharge_disposition_consolidated`) is then one hot encoded

```
# one_hot_encode
# discharge_disposition_consolidated
diabetic_data <- diabetic_data %>%
    mutate(home = ifelse(discharge_disposition_consolidated ==
        "home", 1, 0), healthcare_facility = ifelse(discharge_disposition_consolidated ==
        "healthcare_facility", 1, 0), home_with_help = ifelse(discharge_disposition_consolidated ==
        "home_with_help", 1, 0), AMA = ifelse(discharge_disposition_consolidated ==
        "AMA", 1, 0), admitted = ifelse(discharge_disposition_consolidated ==
        "admitted", 1, 0), hospice_expired = ifelse(discharge_disposition_consolidated ==
        "hospice_expired", 1, 0), outpatient = ifelse(discharge_disposition_consolidated ==
        "outpatient", 1, 0), unknown_discharge_disposition = ifelse(discharge_disposition_consolidated =
        "unknown_discharge_disposition",
```

```
        1, 0))
```

Change to diabetic medication

The `change` variable indicates if there was a change in diabetic medications (either dosage or generic name). Initial values are: "change" and "no change". These are respectively encoded to 1 and 0.

```
diabetic_data <- diabetic_data %>%
  mutate(
    change = ifelse(change == 'Ch', 1,0)
  )
```

Diabetes medication

The `diabetesMed` variable indicates if there was any diabetic medication prescribed. Values are: "yes" and "no". These are respectively encoded to 1 and 0.

```
diabetic_data <- diabetic_data %>%
  mutate(
    diabetesMed = ifelse(diabetesMed == 'Yes', 1,0)
  )
```

Partitioning of dataset

The `diabetic_data` dataset is partition into a **training** and **test** dataset. Importantly this is done before creating the z-scores for the respective numerical variables. Otherwise there would be cross over of data.

```
diabetic_data_partitions <- diabetic_data %>%
  sdf_random_split(diabetic_data_training = 0.3, diabetic_data_test = 0.7, seed = 1099)

diabetic_data_training <- diabetic_data_partitions$diabetic_data_training
diabetic_data_test <- diabetic_data_partitions$diabetic_data_test
```

Z-scoring numerical variables

The z-scores for the training and test dataset are calculated independently

```
# z-score partitioned training dataset
diabetic_data_training <- diabetic_data_training %>%
    mutate(z_age_contin = (age_contin - mean(age_contin,
        na.rm = TRUE))/sd(age_contin, na.rm = TRUE),
        z_time_in_hospital = (time_in_hospital - mean(time_in_hospital,
            na.rm = TRUE))/sd(time_in_hospital, na.rm = TRUE),
        z_num_lab_procedures = (num_lab_procedures -
            mean(num_lab_procedures, na.rm = TRUE))/sd(num_lab_procedures,
            na.rm = TRUE), z_num_procedures = (num_procedures -
            mean(num_procedures, na.rm = TRUE))/sd(num_procedures,
            na.rm = TRUE), z_number_outpatient = (number_outpatient -
            mean(number_outpatient, na.rm = TRUE))/sd(number_outpatient,
            na.rm = TRUE), z_number_emergency = (number_emergency -
            mean(number_emergency, na.rm = TRUE))/sd(number_emergency,
            na.rm = TRUE), z_number_inpatient = (number_inpatient -
            mean(number_inpatient, na.rm = TRUE))/sd(number_inpatient,
            na.rm = TRUE), z_number_diagnoses = (number_diagnoses -
```

```
            mean(number_diagnoses, na.rm = TRUE))/sd(number_diagnoses,
            na.rm = TRUE))

# z-score partitioned test dataset
diabetic_data_test <- diabetic_data_test %>%
    mutate(z_age_contin = (age_contin - mean(age_contin,
        na.rm = TRUE))/sd(age_contin, na.rm = TRUE),
        z_time_in_hospital = (time_in_hospital - mean(time_in_hospital,
            na.rm = TRUE))/sd(time_in_hospital, na.rm = TRUE),
        z_num_lab_procedures = (num_lab_procedures -
            mean(num_lab_procedures, na.rm = TRUE))/sd(num_lab_procedures,
            na.rm = TRUE), z_num_procedures = (num_procedures -
            mean(num_procedures, na.rm = TRUE))/sd(num_procedures,
            na.rm = TRUE), z_number_outpatient = (number_outpatient -
            mean(number_outpatient, na.rm = TRUE))/sd(number_outpatient,
            na.rm = TRUE), z_number_emergency = (number_emergency -
            mean(number_emergency, na.rm = TRUE))/sd(number_emergency,
            na.rm = TRUE), z_number_inpatient = (number_inpatient -
            mean(number_inpatient, na.rm = TRUE))/sd(number_inpatient,
            na.rm = TRUE), z_number_diagnoses = (number_diagnoses -
            mean(number_diagnoses, na.rm = TRUE))/sd(number_diagnoses,
            na.rm = TRUE))
```

Machine learning modelling

Model formula

```
ml_formula <- formula(early_readmission ~
                        diag_1_infection +
                        diag_1_neoplasms +
                        diag_1_endo_metabolic_immunity +
                        diag_1_haematology +
                        diag_1_mental +
                        diag_1_neurology +
                        diag_1_circulatory +
                        diag_1_respiratory +
                        diag_1_digestive +
                        diag_1_genitourinary +
                        diag_1_preg_birth_puerperium +
                        diag_1_dermatology +
                        diag_1_musculoskeletal +
                        diag_1_congenital +
                        diag_1_perinatal +
                        diag_1_ill_defined +
                        diag_1_injury_poisoning +
                        diag_1_supplementary +
                        diag_1_diabetes +
                        diag_2_unknown +
                        diag_2_infection +
                        diag_2_neoplasms +
                        diag_2_endo_metabolic_immunity +
                        diag_2_haematology +
```

```
                        diag_2_mental +
                        diag_2_neurology +
                        diag_2_circulatory +
                        diag_2_respiratory +
                        diag_2_digestive +
                        diag_2_genitourinary +
                        diag_2_preg_birth_puerperium +
                        diag_2_dermatology +
                        diag_2_musculoskeletal +
                        diag_2_congenital +
                        diag_2_perinatal +
                        diag_2_ill_defined +
                        diag_2_injury_poisoning +
                        diag_2_supplementary +
                        diag_2_diabetes +
                        diag_2_unknown +
                        diag_3_infection +
                        diag_3_neoplasms +
                        diag_3_endo_metabolic_immunity +
                        diag_3_haematology +
                        diag_3_mental +
                        diag_3_neurology +
                        diag_3_circulatory +
                        diag_3_respiratory +
                        diag_3_digestive +
                        diag_3_genitourinary +
                        diag_3_preg_birth_puerperium +
                        diag_3_dermatology +
                        diag_3_musculoskeletal +
                        diag_3_congenital +
                        diag_3_perinatal +
                        diag_3_ill_defined +
                        diag_3_injury_poisoning +
                        diag_3_supplementary +
                        diag_3_diabetes +
                        diag_3_unknown +
                        home +
                        healthcare_facility +
                        home_with_help +
                        AMA +
                        admitted +
                        outpatient +
                        unknown_discharge_disposition +
                        asian +
                        african_american +
                        caucasian +
                        hispanic +
                        unknown_race +
                        emergency +
                        urgent +
                        elective +
                        admisison_type_other +
                        max_glu_serum_none +
```

```
                    max_glu_serum_norm +
                    max_glu_serum_300 +
                    max_glu_serum_200 +
                    A1Cresult_none +
                    A1Cresult_norm +
                    A1Cresult_7 +
                    A1Cresult_8 +
                    z_age_contin +
                    z_time_in_hospital +
                    z_num_lab_procedures +
                    z_num_procedures +
                    z_number_outpatient +
                    z_number_emergency +
                    z_number_inpatient +
                    z_number_diagnoses +
                    Cardiology +
                    ObstetricsandGynecology +
                    Pediatrics +
                    SurgeryColonRectal +
                    PediatricsCriticalCare +
                    Anesthesiology_Pediatric +
                    Ophthalmology +
                    InfectiousDiseases +
                    SurgeryMaxillofacial +
                    PsychiatryAddictive +
                    SurgeryCardiovascular +
                    Speech +
                    Endocrinology_Metabolism +
                    FamilyGeneralPractice +
                    SurgeryGeneral +
                    Orthopedics +
                    EmergencyTrauma +
                    HematologyOncology +
                    Otolaryngology +
                    Oncology +
                    SurgeryPediatric +
                    PediatricsEmergencyMedicine +
                    AllergyandImmunology +
                    PediatricsInfectiousDiseases +
                    Osteopath +
                    SurgicalSpecialty +
                    Dermatology +
                    SportsMedicine +
                    Resident +
                    InternalMedicine +
                    Gastroenterology +
                    SurgeryCardiovascularThoracic +
                    Nephrology +
                    OrthopedicsReconstructive +
                    ObstericsGynecologyGynecologicOnco +
                    Endocrinology +
                    Pediatrics_Pulmonology +
                    Neurology +
```

```
                        Psychology +
                        Podiatry +
                        Gynecology +
                        SurgeryPlastic +
                        SurgeryThoracic +
                        SurgeryPlasticwithinHeadandNeck +
                        PhysicalMedicineandRehabilitation +
                        Rheumatology +
                        PediatricsAllergyandImmunology +
                        Surgeon +
                        SurgeryVascular +
                        Pathology +
                        Hospitalist +
                        OutreachServices +
                        CardiologyPediatric +
                        Neurophysiology +
                        PediatricsEndocrinology +
                        Psychiatry +
                        Pulmonology +
                        SurgeryNeuro +
                        Urology +
                        PsychiatryChildAdolescent +
                        Radiology +
                        PediatricsHematologyOncology +
                        PediatricsNeurology +
                        Anesthesiology +
                        Dentistry +
                        PhysicianNotFound +
                        Hematology +
                        Proctology +
                        Obstetrics +
                        Radiologist +
                        Perinatology +
                        DCPTEAM +
                        medical_specialty_unkown +
                        change +
                        diabetesMed)
```

Logistic regression

```
lg_model = ml_logistic_regression(diabetic_data_training,
    ml_formula, fit_intercept = FALSE, family = "binomial")
```

Gradient boosted trees

```
gbt_model = ml_gradient_boosted_trees(diabetic_data_training,
    ml_formula, type = "classification")
```

Multilayer perceptron

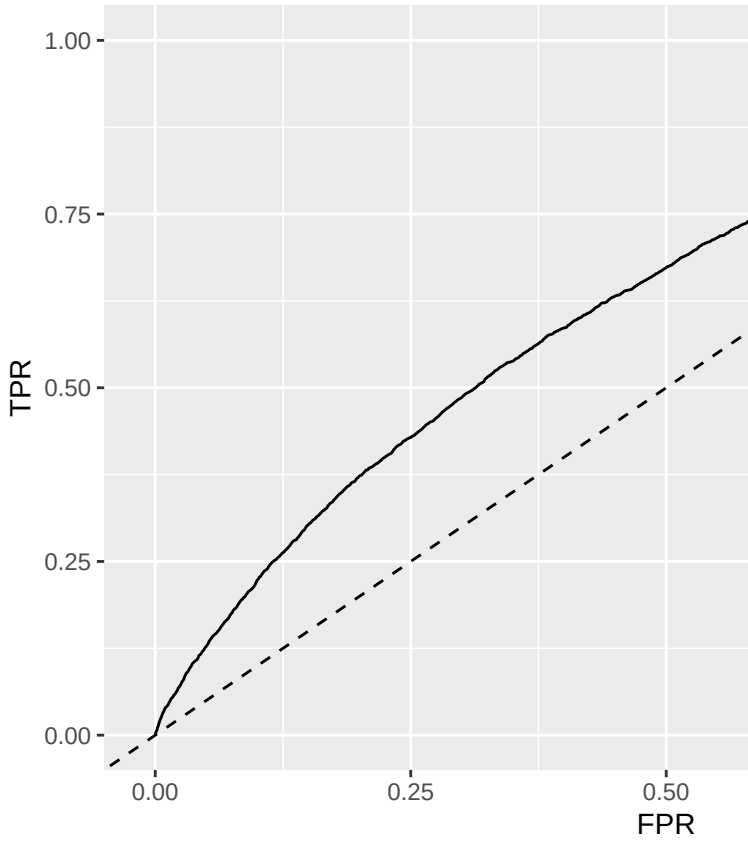
```
mlp_model = ml_multilayer_perceptron_classifier(diabetic_data_training,
    ml_formula, layers = c(166, 80, 80, 2))
```

Evaluating models

Logistic regression

```
lg_model_metrics = ml_evaluate(lg_model, diabetic_data_test)
lg_auc <- lg_model_metrics$area_under_roc()
```

```
lg_roc <- lg_model_metrics$roc() %>%
  collect()

ggplot(lg_roc, aes(x = FPR, y = TPR)) +
  geom_line() + geom_abline(lty = "dashed")
```



Graph ROC curve for the logistic regression model

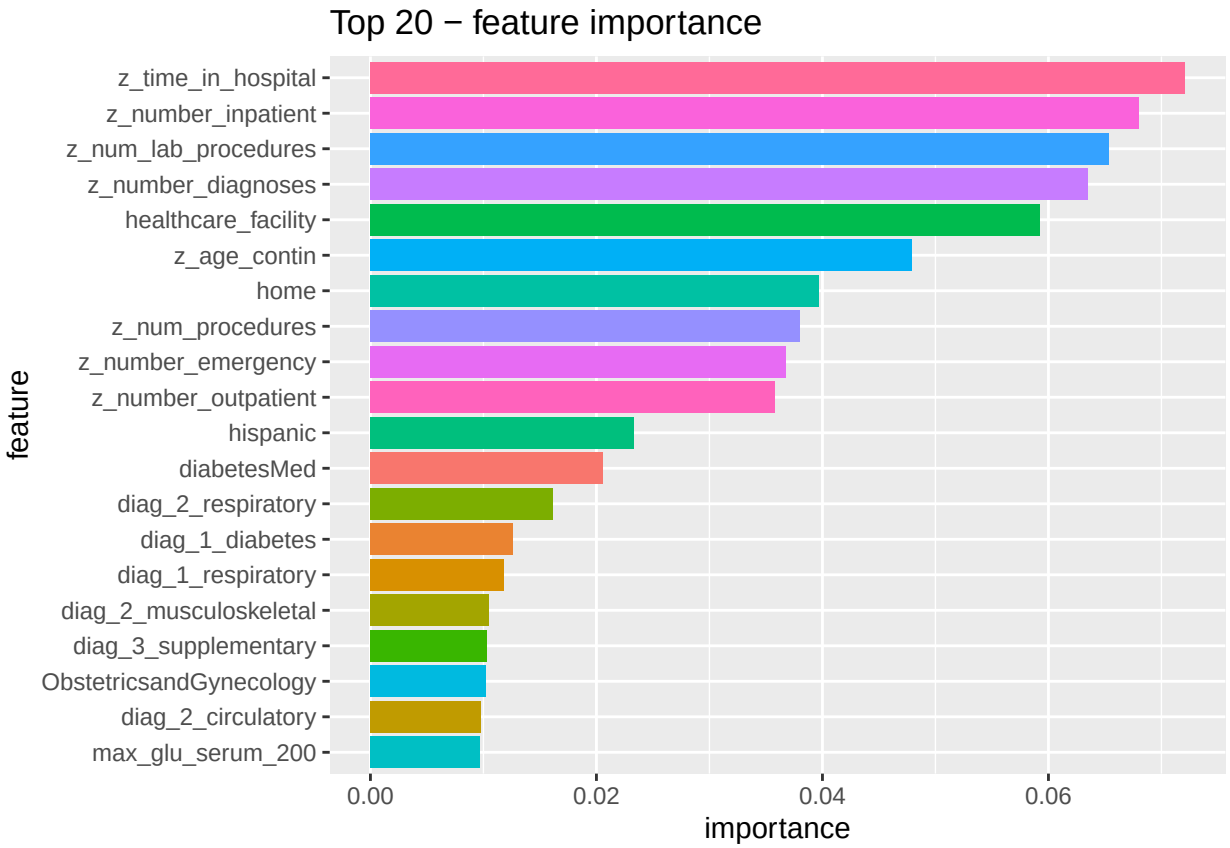
Gradient boosted trees

```
gbt_predictions = ml_predict(gbt_model, diabetic_data_test)
gbt_auc <- ml_binary_classification_evaluator(gbt_predictions,
    label_col = "early_readmission")
```

Graphing of feature importance

```
featureImport <- ml_tree_feature_importance(gbt_model)
featureImport[1:20, ] %>%
    ggplot(aes(reorder(feature, importance), importance,
        fill = feature)) + geom_bar(stat = "identity") +
    coord_flip() + ggtitle("Top 20 - feature importance") +
```

```
    theme(legend.position = "none") + ylab("importance") +
    xlab("feature")
```



Multilayer perceptron classifier

```
mlp_predictions = ml_predict(mlp_model, diabetic_data_test)
mlp_auc <- ml_binary_classification_evaluator(mlp_predictions,
    label_col = "early_readmission")
```

Compare models

```
auc <- data.frame( AUC = c(lg_auc, gbt_auc, mlp_auc))
row.names(auc) <- c("Logisitic Regression", "Gradient Boosted Trees", "Multilayer Perceptron" )
kable(auc, caption = "Model comparison - Area Under the ROC curve", digits = 3, format.args = list(big.m
  scientific = FALSE))%>%
  kable_styling(latex_options = "HOLD_position")
```

Table 9: Model comparison - Area Under the ROC curve

	AUC
Logisitic Regression	0.629
Gradient Boosted Trees	0.638
Multilayer Perceptron	0.627