Lab3: Exploratory Data Analysis

Overview

This is a two part lab where each part will focus on a different dataset: the first part will use a dataset containing a series of diagnostic measurements taken on members of the Akimel O'odham people (an indigenous group living in the Southwestern United States who are also called the Pima) to understand diabetes risk (click here to download diabetes.csv), and the second dataset contains information on traffic accidents in New York City in the months of July and August of this year, and was compiled by NYC Open Data (click here to download crashes.csv).

For this problem set you will need to install the `skimr` and `GGally` packages, and in particular the functions `skim` and `ggpairs`.

We will also explore the concept of an *inlier*, which is an erroneous value that occurs in the interior of the distribution of a variable, rather than in the tails of the variable. The US Census published an article on the problem of inliers here

Part 1: Health Diagnostics and Diabetes Incidence

Problem 1: Data Description and Outliers.

Load `diabetes.csv` into R and take a look at the data using the `skimr` package (make sure to install it if you don't have it). Skimr provides a tidy summary function called `skim`. Use `skim` on the data frame that you loaded from diabetes.csv.

```
install.packages("skimr")
install.packages("GGally")
```

```
library(skimr)
library(GGally)
library(tidyverse)
library(ggplot2)
```

```
library(readr)
diabetes <- read_csv("diabetes.csv")
View(diabetes)
```

Skim will list several variables. Pregnancies is the past number of pregnancies (this dataset includes women 21 years or older), glucose describes the concentration of glucose in the blood after an oral glucose tolerance test (drinking a sugary drink and measuring two hours later), skin thickness is the result of a skinfold thickness test taken at the triceps (upper arm), Insulin is the insulin concentration in the blood taken at the same time as the glucose measurement (Insulin is a hormone that transports glucose into cells), BMI is "Body Mass Index", Diabetes Pedigree Function is a measure of diabetes risk based on the family history of diabetes for each patient (this is an engineered feature) and outcome is equal to 1 if the patient was diagnosed with diabetes with 5 years and 0 otherwise.

```
skim(diabetes)
```

a) Skim should show no missing data, but should indicate potential data issues. Do any of the percentile ranges (p0, p25, p50, p75, or p100) for the reported variables suggest a potential problem?

Some of the values for the variables in the percentile ranges have outliers that could bring errors into the data. Some the variables share the same the same minimum value as the p0 range which represent the minimum value in the data. This will bring in some error if there is too many zeros within the data.

b) Further investigate the dataset to find potentially problematic variables using a qq-plot (`geom_qq`) or `group_by` combined with `count` and `arrange`. For which variables do you find repeated values and what are those values? Do you believe these values represent real measurements or could they correspond to missing data? Do the repeated variables occur in the same rows or different rows?

Write an overview of which values are missing and replace all missing values with NA for the next stage of analysis.

```
diabetes |>
  group_by(Glucose) |>
  count(Glucose) |>
  arrange(desc(Glucose))
```

```
diabetes |>
  group_by(Pregnancies) |>
  count(Pregnancies) |>
  arrange(desc(Pregnancies))
```

```
diabetes |>
  group_by(BloodPressure) |>
  count(BloodPressure) |>
  arrange(desc(BloodPressure))
```

```
diabetes |>
  group_by(SkinThickness) |>
  count(SkinThickness) |>
  arrange(desc(SkinThickness))
```

```
diabetes |>
  group_by(Insulin) |>
  count(Insulin) |>
  arrange(desc(Insulin))
```

```
diabetes = diabetes |>
  mutate( Pregnancies = if_else( Pregnancies == 0, NA , Pregnancies),
          Glucose = if_else(Glucose == 0, NA, Glucose),
          BloodPressure = if_else(BloodPressure == 0, NA, BloodPressure),
          Insulin = if_else(Insulin == 0, NA, Insulin),
          BMI = if_else(BMI == 0, NA, BMI),
          DiabetesPedigreeFunction = if_else( DiabetesPedigreeFunction  == 0, NA, DiabetesPe
          Age = if_else( Age == 0, NA, Age),
          Outcome = if_else( Outcome == 0, NA, Outcome)
          )
```

c) Perform Tukey Box plots on each variable to identify potential outliers. Which variables have the most outliers? Are there any outliers that you think come from measurement error? If so remove them.

```
{r}# diabetes_box <- diabetes %>%    pivot_longer( cols = )
```

```
diabetes_long <- diabetes %>%
  pivot_longer(cols = Pregnancies:Outcome, names_to = "variable", values_to = "value")
```

```
diabetes_long |>
  ggplot(diabetes_long, mapping =  aes(x = variable, y = value)) + geom_boxplot(outlier.colo
```

Problem 2: Pair Plots

Use the `GGally` package and its function `ggpair` on both the original dataset and the cleaned dataset. Which correlations change the most? What are the strongest correlations between variables overall and with the `Outcome`?

```
{r warning=FALSE} diabetes |> ggpairs()
```

```
{r message=FALSE, warning=FALSE} diabetes_long |> ggpairs()
```

- Remark: This dataset has been used an model dataset for the construction of binary classifiers using machine learning and there are a large number of published studies showing these analyses. However, many of these analyses did not exclude the missing values erroneously coded as zero, as is discussed in this interesting paper by Breault, leading to highly degraded accuracy.

Part 2: Car Crashes in NYC

Problem 3: Finding Inliers and Missing Data

Load the NYC car crash dataset using `read_csv`. You can download the data from the course website by clicking here.

```
library(readr)
crashes <- read_csv("crashes.csv")
View(crashes)
```

a) Which variables have missing data (use `skim` or another tool of your choosing)? Some missing values have a different interpretation than others- what does it mean when `VEHICLE TYPE CODE 2` is missing compared to `LATITUDE`?

When vehicle type code 2 is missing it means that only one vehicle type code was involved in the crash and when latitude is missing that means they have unavailable data about the precise location of the crash. Without the location data we cannot verify the location impact of the crash.

```
skim(crashes)
```

```
crashes |>
  skim() |>
    dplyr::filter(n_missing > 0)
```

```
summary(crashes)
```

b) Latitude and Longitude have the same number of missing values. Verify that they always occur in the same row. Check the counts of latitude and longitude values- do you find any hidden missing values? If so recode them as NA.

```
missing_location <- crashes |>
  summarise(
    missing_lat = sum(is.na(LATITUDE)), missing_long = sum(is.na(LONGITUDE))
  )
```

```
missing_location <- crashes |>
  filter(is.na(LATITUDE) | is.na(LONGITUDE) )
```

```
missing_location_1 <- crashes |>
  filter(is.na(LATITUDE) & is.na(LONGITUDE))
```

```
count(missing_location)
```

```
crashes = crashes |>
  mutate(
    LATITUDE = if_else(LATITUDE == 0,NA, LATITUDE),
    LONGITUDE = if_else(LONGITUDE == 0, NA, LONGITUDE)
  )
```

c) Many of the geographic values are missing, but geographic information is redundant in multiple variables in the dataset. For example, with effort you could determine the borough of an accident from the zip code, the latitude and longitude, or the streets (not part of the assignment for this week). Consider the borough variable- what percentage of the missing values of borough have values present of *at least* one of zip code or latitude. What about if we include all the street name variables? What fraction of rows don't have any detailed location information (latitude, zip code, or street names)?

```
missing_borough <- sum(is.na(crashes$BOROUGH))
borough_total <- nrow(crashes)
```

```
borough_percentage  <- (missing_borough/borough_total) * 100
```

```
zipcode_or_latitude <- crashes |>
  filter(is.na(BOROUGH)) |>
  filter(is.na(`ZIP CODE`) | is.na(LATITUDE))
```

```
zipcode_or_latitude_percenatge <- (nrow(zipcode_or_latitude) / missing_borough) * 100
```

```
zipcode_or_latitude_street <- crashes |>
  filter(is.na(BOROUGH)) |>
  filter(is.na(`ZIP CODE`) | is.na(LATITUDE) | is.na(`ON STREET NAME`) | is.na(`CROSS STREET
```

```
street_perecentage <- (nrow(zipcode_or_latitude_street) / missing_borough) * 100
```

```
zero_street_info <- crashes |>
  filter(is.na(LATITUDE) & is.na(x = `ZIP CODE`) & is.na(`ON STREET NAME`) & is.na(`CROSS ST
```

d) The `CRASH TIME` variable has no missing values. Compute the count of how many times each individual time occurs in the crash data set. This will suggest that there are some inliers in the data. Compute summary statistics on the count data, and determine how many inliers there are (define an inlier as a data value where the count is an outlier, i.e. the count of that value is greater than 1.5*IQR + P75, i.e. 1.5 times the interquartile range past the 75th percentile for the distribution of counts for values of that variable.) For which inliers do you believe the time is most likely to be accurate? For which is it least likely to be accurate and why do you think so?

```
crashes_count <- crashes |>
  group_by(`CRASH TIME`) |>
  summarise(count = n()) |>
  arrange(desc(count))
```

```
ggplot( data = crashes_count, mapping = aes(x = `CRASH TIME`, y = count)) + geom_point()
```

```
ggplot( data = crashes_count, mapping = aes(x = `CRASH TIME`, y = count)) + geom_boxplot()
```

```
ggplot( data = crashes_count, mapping = aes(x = count, y = `CRASH TIME`)) + geom_boxplot()
```

```
crash_summary_stats <- crashes_count |>
  summarise(
    crash_min = min(count),
    Q1 = quantile(count,0.25),
    crash_median_count = median(count),
    Q3 = quantile(count, 0.75),
    crash_max_count = max(count),
    crash_IQR = IQR(count)
  )
```

```
print(crash_summary_stats)
```

```
crash_outlier<- crash_summary_stats$Q3 +1.5 * crash_summary_stats$crash_IQR
```

```
crash_inlin <- crashes_count |>
  filter( count > crash_outlier)
```

```
install.packages("lubridate")
library(lubridate)
```

```
crashes <- crashes |>
  mutate(`CRASH TIME` = hms::as_hms(`CRASH TIME`))
```

```
crashes_count <- crashes |>
  group_by(`CRASH TIME`) |>
  summarise(count = n()) |>
 mutate(
    likely_to_be_accurate = case_when(
      `CRASH TIME` >= hms::as_hms("08:00:00") &
        `CRASH TIME` <= hms::as_hms("09:00:00") ~ "likely to be accurate",

    ),
    least_likely_to_b_accurate = case_when(
      `CRASH TIME` == hms::as_hms("00:00:00") ~ "least_likely to be accurate",
      `CRASH TIME` == hms::as_hms("12:00:00") ~ "least likely to be accurate",

    )
  )
```

```
head(crashes_count)
```

Problem 4: Finding Patterns in the Data

Formulate a question about crash data in NYC and make visualizations to explore your question. It could be related to the geographical distribution of accidents, the timing of accidents, which types of vehicles lead to more or less dangerous accidents, or anything else you want. Write comments/notes describing your observations in each visualizations you create and mention how these observations impact your initial hypotheses.

Useful questions to consider when you observe a pattern:

- Could this pattern be due to coincidence (i.e. random chance)?

- How can you describe the relationship implied by the pattern?
- How strong is the relationship implied by the pattern?
- What other variables might affect the relationship?
- Does the relationship change if you look at individual subgroups of the data?

Question - Does the borough help determine how serve the causality of the car accident is?

According the the visualization the average person in NYC is more likely to be killed in the car crash in any other borough in the city. A New Yorker is least likely to die in car in Staten island. Some variables that will affect the severity of the accident would be the the concentration of population, traffic volume, weather conditions, and road conditions however that data is not available here. The average severity is higher in Brooklyn than Bronx, Queens, Manhattan and Staten Island. This high average severity is the reason why Brooklyn has the highest rate of death in car accidents.

```
crashes_borough <- crashes |>
  mutate( severity = `NUMBER OF PERSONS INJURED` + `NUMBER OF PERSONS KILLED`) |>
  group_by(BOROUGH) |>
  filter(!is.na(BOROUGH), !is.na(severity))|>
  summarise( avg_severity = mean(severity, na.rm = TRUE),
  total_severity = sum(severity, na.rm = TRUE),
  accident_count = n()) |>
  arrange(desc(avg_severity))
```

```
print(crashes_borough)
```

```
ggplot(crashes_borough, aes(x = reorder(BOROUGH, avg_severity), y = avg_severity, fill = BOR
  geom_bar(stat = "identity")
```

```
ggplot(crashes, aes(x = BOROUGH, y = `NUMBER OF PERSONS INJURED` + `NUMBER OF PERSONS KILLED
  geom_boxplot(na.rm = TRUE) + scale_color_brewer(palette = "Set1")
```

```
ggplot( data = crashes_borough, mapping = aes(x = fct_reorder(BOROUGH, avg_severity) ))
```