Project 2 Data Transformation - Diabetes Dataset

shri Tripathi

2024-10-11

First, I will load the dataset into R from my Github repository and add an ID column. Next, I will clean the data by consolidating the diabetes predictor variables into a single column. Afterward, I will conduct basic descriptive analyses to identify the factors most strongly associated with diabetes outcomes.

Loading the Dataset & Introducing an ID Column:

```
diabetes <- read.csv("https://raw.githubusercontent.com/Shriyanshh/Project-2---Data-Transformation/refs/

# Load the dataset and add an ID column to track observations
diabetes <- diabetes %>% mutate(ID = row_number())

# Reorder the columns to place the ID column first
diabetes <- diabetes %>% select(ID, everything())
```

Reshaping the Dataframe to Long Format:

```
# Reshape the dataset to long format, pivoting the predictor columns
diabetes_long <- diabetes %>% pivot_longer(cols = c(Pregnancies:BMI, Age), names_to = "Predictors", valu

# Select relevant columns to keep in the long format dataframe
diabetes_long <- diabetes_long %>% select(ID, Predictors, Values, DiabetesPedigreeFunction, Outcome)
```

Calculating Mean Predictor Values by Outcome

```
# Split the dataset into two: one for positive (1) outcomes and one for negative (0) outcomes.
# This will help in calculating mean values for each predictor based on the outcome.

diabetes_pos <- diabetes_long %>% filter(Outcome == 1)
diabetes_neg <- diabetes_long %>% filter(Outcome == 0)
```

First, I will calculate the average Diabetes Pedigree Function for individuals with and without diabetes to evaluate whether it serves as a reliable marker for predicting diabetes.

```r
# Print the number of individuals without diabetes
print(paste0("Number of people without diabetes: ", nrow(diabetes_neg)))
```

```
## [1] "Number of people without diabetes: 3500"
```

```r
# Print the number of individuals with diabetes
print(paste0("Number of people with diabetes: ", nrow(diabetes_pos)))
```

```
## [1] "Number of people with diabetes: 1876"
```

As shown, the number of individuals without diabetes is greater than those with diabetes. This aligns with population trends, making it reasonable for generalization.

```r
# Calculate and print the mean Diabetes Pedigree Function for individuals without diabetes
mean_pedigree_neg <- diabetes_neg %>% summarize(mean_pedigree = mean(DiabetesPedigreeFunction))
print(mean_pedigree_neg)
```

```
## # A tibble: 1 x 1
##   mean_pedigree
##           <dbl>
## 1         0.430
```

```r
# Calculate and print the mean Diabetes Pedigree Function for individuals with diabetes
mean_pedigree_pos <- diabetes_pos %>% summarize(mean_pedigree = mean(DiabetesPedigreeFunction))
print(mean_pedigree_pos)
```

```
## # A tibble: 1 x 1
##   mean_pedigree
##           <dbl>
## 1         0.550
```

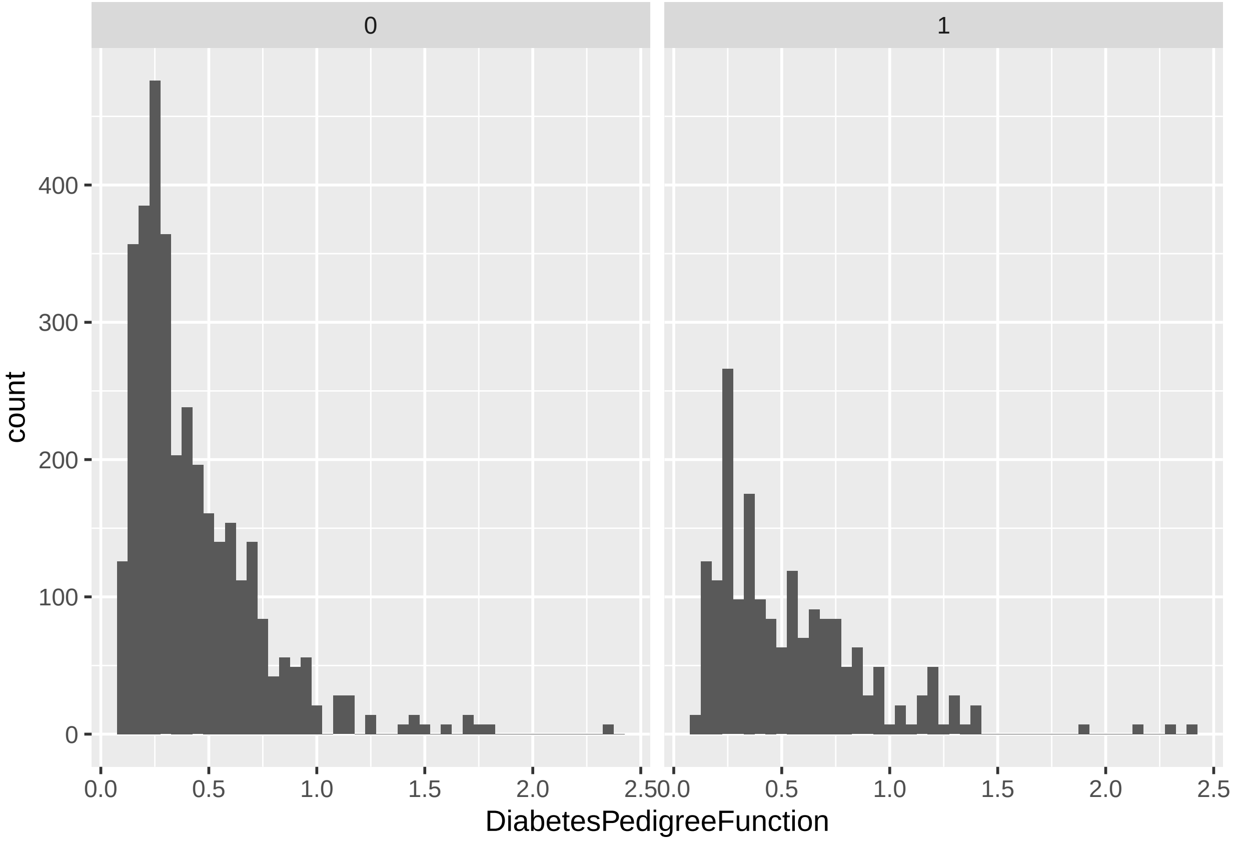
```r
# Calculate and print the median Diabetes Pedigree Function for individuals without diabetes
median_pedigree_neg <- diabetes_neg %>% summarize(median_pedigree = median(DiabetesPedigreeFunction))
print(median_pedigree_neg)
```

```
## # A tibble: 1 x 1
##   median_pedigree
##             <dbl>
## 1           0.336
```

```r
# Calculate and print the median Diabetes Pedigree Function for individuals with diabetes
median_pedigree_pos <- diabetes_pos %>% summarize(median_pedigree = median(DiabetesPedigreeFunction))
print(median_pedigree_pos)
```

```
## # A tibble: 1 x 1
##   median_pedigree
##             <dbl>
## 1           0.449
```

```
# Plot histogram of Diabetes Pedigree Function with a bin width of 0.05, faceted by Outcome
ggplot(diabetes_long, aes(x = DiabetesPedigreeFunction, y = after_stat(count))) +
  geom_histogram(binwidth = .05) +
  facet_wrap(~Outcome)
```



The analysis indicates that, on average, the Diabetes Pedigree Function is slightly higher for individuals diagnosed with diabetes compared to those without. The histograms reveal that the distributions for both groups are heavily positively skewed, which is expected for this type of risk indicator. For individuals with diabetes, values tend to cluster closer to 1.0, while for those without diabetes, the values are more dispersed. Given the skewness, I also calculated the median values, which show a noticeable difference between the two groups. However, despite this difference, the Diabetes Pedigree Function may not be the most reliable predictor for diabetes, as both the mean and median values for those diagnosed with diabetes hover around 0.5. It's important to keep in mind that this function reflects risk factors associated with diabetes rather than diabetes itself.

Now, I will find the averages for each predictor:

```
# Display unique predictor variables
unique(diabetes_long$Predictors)
```

```
## [1] "Pregnancies"   "Glucose"       "BloodPressure" "SkinThickness"
## [5] "Insulin"       "BMI"           "Age"
```

```
# Calculate and print mean values of predictors for individuals without diabetes
diabetes_means_neg <- diabetes_neg %>% group_by(Predictors) %>% summarize(Group_means = mean(Values))
print(diabetes_means_neg)
```

```
## # A tibble: 7 x 2
##   Predictors    Group_means
##   <chr>               <dbl>
## 1 Age                 31.2
## 2 BMI                 30.3
## 3 BloodPressure       68.2
## 4 Glucose            110.
## 5 Insulin             68.8
## 6 Pregnancies          3.30
## 7 SkinThickness       19.7
```

```
# Calculate and print mean values of predictors for individuals with diabetes
diabetes_means_pos <- diabetes_pos %>% group_by(Predictors) %>% summarize(Group_means = mean(Values))
print(diabetes_means_pos)
```

```
## # A tibble: 7 x 2
##   Predictors    Group_means
##   <chr>               <dbl>
## 1 Age                 37.1
## 2 BMI                 35.1
## 3 BloodPressure       70.8
## 4 Glucose            141.
## 5 Insulin            100.
## 6 Pregnancies          4.87
## 7 SkinThickness       22.2
```

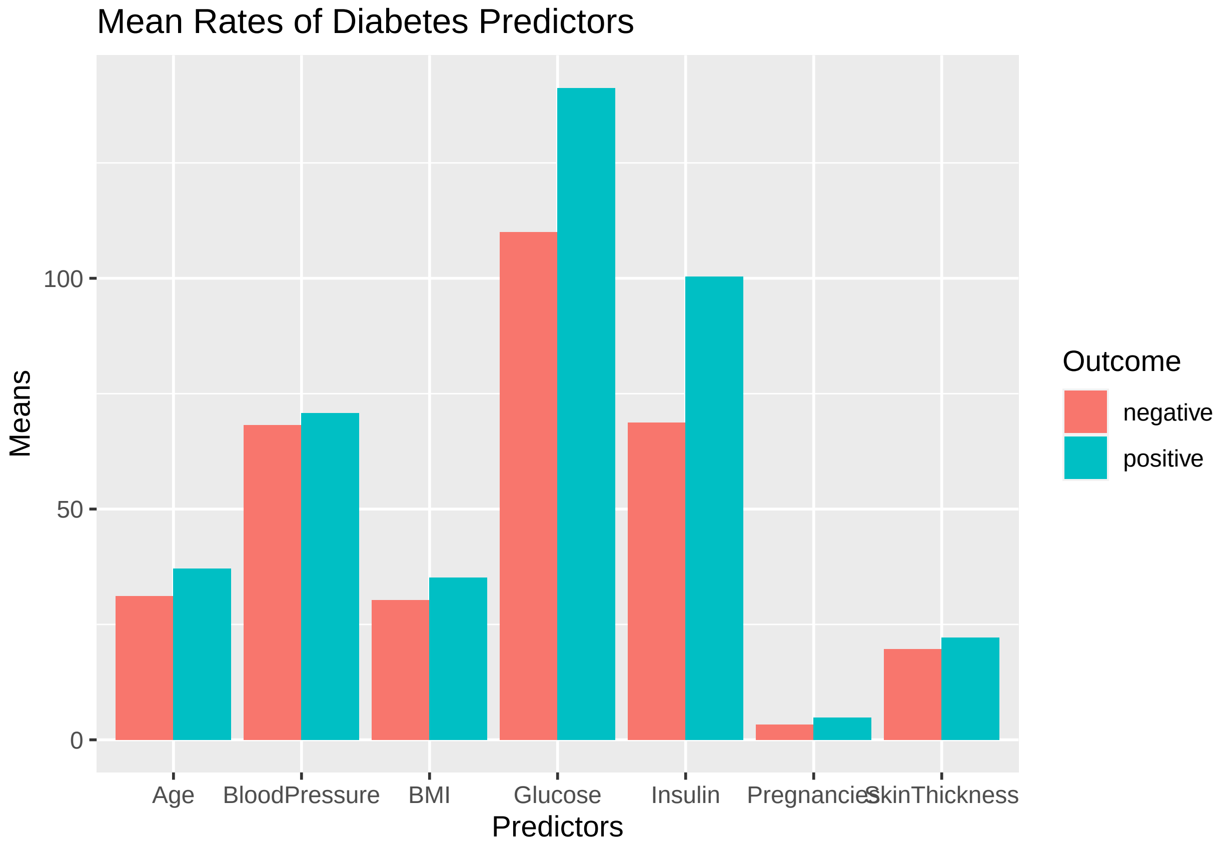
As we can see, all predictor categories show higher values in the positive group (those with diabetes) compared to the negative group, which aligns with expectations. Let's explore these differences further through graphical comparisons to visualize the trends.

```
# Combine the mean comparisons into a single dataframe
diabetes_means_neg <- data_frame(diabetes_means_neg)
```

```
## Warning: `data_frame()` was deprecated in tibble 1.1.0.
## i Please use `tibble()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
diabetes_means_pos <- data_frame(diabetes_means_pos)

# Add outcome labels for the negative and positive groups
diabetes_means_neg <- diabetes_means_neg %>% mutate(Outcome = 0)
diabetes_means_pos <- diabetes_means_pos %>% mutate(Outcome = 1)

# Combine the two dataframes into one
diabetes_mean_append <- rbind(diabetes_means_neg, diabetes_means_pos)

# Recode the outcome values to 'negative' and 'positive' for clarity
diabetes_mean_append <- diabetes_mean_append %>% mutate(Outcome = recode(Outcome, '0' = 'negative', '1'

# Create a bar plot to compare the mean rates of diabetes predictors by outcome
```

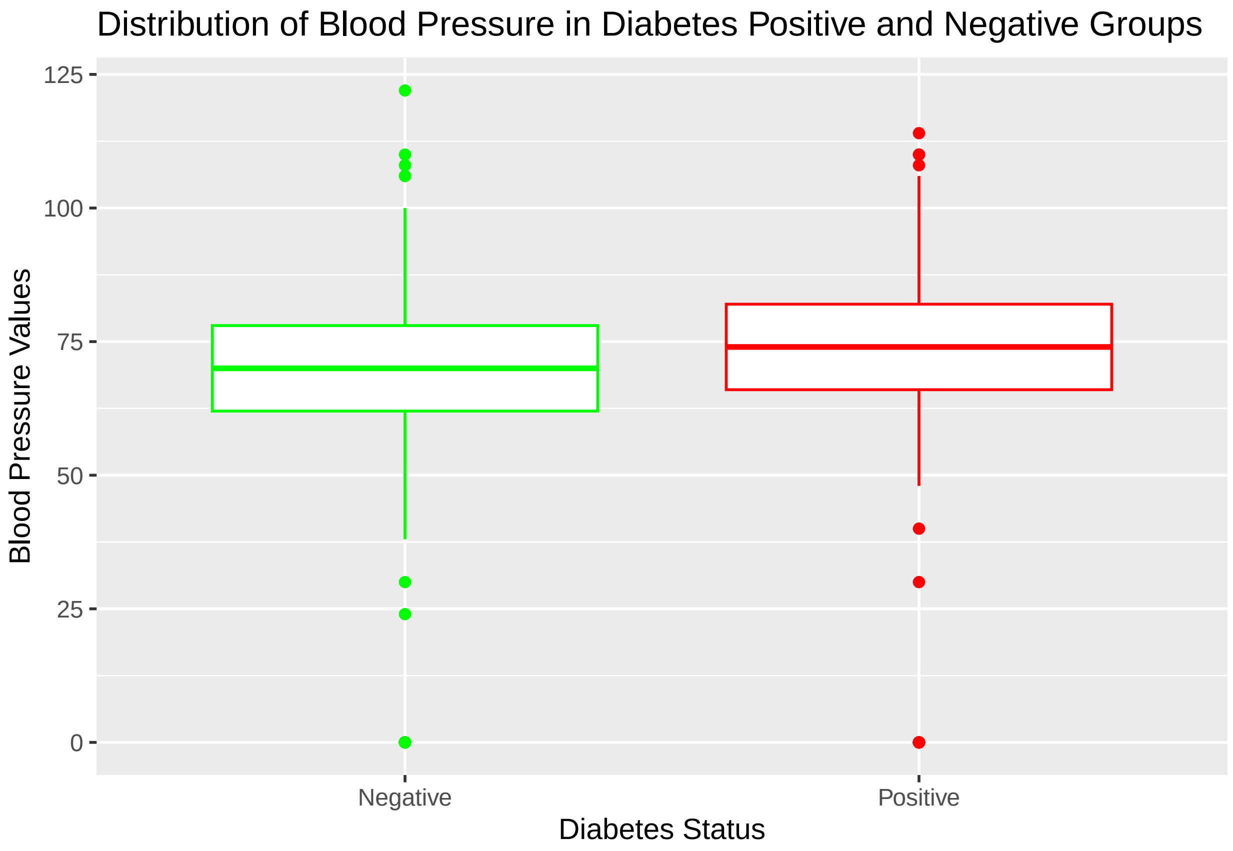
```r
ggplot(diabetes_mean_append, aes(x = Predictors, y = Group_means, fill = Outcome)) +
  geom_bar(stat= "identity", position = "dodge") +
  labs(title = "Mean Rates of Diabetes Predictors", y = "Means")
```



As expected, insulin and glucose levels are the strongest predictors of diabetes, with higher values closely linked to the presence of the condition. Next, I will examine how blood pressure distribution differs between the two groups.

```r
# Filter blood pressure values for positive and negative groups
bloodPressure_pos <- diabetes_pos %>%
  filter(Predictors == "BloodPressure") %>%
  mutate(Status = "Positive")

bloodPressure_neg <- diabetes_neg %>%
  filter(Predictors == "BloodPressure") %>%
  mutate(Status = "Negative")

# Create a boxplot comparing blood pressure distributions for both groups
bp = ggplot() +
  ggtitle("Distribution of Blood Pressure in Diabetes Positive and Negative Groups") +
  xlab("Diabetes Status") +
  ylab("Blood Pressure Values")

# Add boxplot for Positive group
bp = bp +
  geom_boxplot(data = bloodPressure_pos, aes(x = Status, y = Values), colour='red')
```

```
# Add boxplot for Negative group
bp = bp +
  geom_boxplot(data = bloodPressure_neg, aes(x = Status, y = Values), colour='green')

# Display the plot
print(bp)
```



Conclusion:

The blood pressure distributions between the diabetes-positive and diabetes-negative groups appear similar. However, the mean blood pressure for the diabetes-positive group is slightly higher than that of the negative group. This could suggest that individuals with diabetes tend to have higher blood pressure, or conversely, elevated blood pressure might be associated with an increased likelihood of developing diabetes.