Project Report

STAT 380 Section 004

Gianna DeLorenzo & Melissa Kim

1 Title:

1.1 Introduction

Patient experiences with medications is crucial when it comes to the healthcare domain by providing insights on the side effects, drug effectiveness, and overall satisfaction.

Research Question

In this report, we will be exploring the following research question:

Can machine learning tasks predict the overall satisfaction of the patients with a particular class of drugs for a given medical condition?

This research question

1.2 Exploratory Data Analysis

1.2.1 Variable Description

The response variable of interest is the ratings of the drugs with the negative reviews from **0-6** and the positive reviews from **7-10**.

1.2.2 Data Visualization

Table 1: Variables used in Analysis

Variable	Type	Explanation
reviewID	Integer	Review ID
urlDrugName	Categorial	Name of drug
rating	Integer	Name of condition
effectiveness	Categorical	Patient on benefits
sideEffects	Categorical	Patient on side effects
condition	Categorical	Overall patient comment
benefitsReview	Categorical	10 star patient rating

Variable	Type	Explanation
sideEffectsReview	Categorical	5 step side effect rating
commentsReview	Categorical	5 step effectiveness rating

Table 1

1.2.3 Data Cleaning

This glimpse of the Drug Reviews train data displays a dataset in need of being cleaned and tidied.

```
Rows: 1,036
Columns: 9
$ ...1              <dbl> 1366, 3724, 3824, 969, 696, 1380, 45, 1939, 2576, 10~
$ urlDrugName       <chr> "biaxin", "lamictal", "depakene", "sarafem", "accuta~
$ rating            <dbl> 9, 9, 4, 10, 10, 2, 8, 10, 10, 1, 3, 9, 6, 10, 5, 5,~
$ effectiveness     <chr> "Considerably Effective", "Highly Effective", "Moder~
$ sideEffects       <chr> "Mild Side Effects", "Mild Side Effects", "Severe Si~
$ condition         <chr> "sinus infection", "bipolar disorder", "bipolar diso~
$ benefitsReview    <chr> "The antibiotic may have destroyed bacteria causing ~
$ sideEffectsReview <chr> "Some back pain, some nauseau.", "Drowsiness, a bit ~
$ commentsReview    <chr> "Took the antibiotics for 14 days. Sinus infection w~
```

```
# A tibble: 1,036 x 7
      id drug_name     condition   rating effectiveness side_effects full_review
   <dbl> <chr>         <chr>        <dbl> <fct>         <fct>        <chr>
 1  1366 Biaxin        Sinus Infe~      9 Considerably~ Mild Side E~ "The antib~
 2  3724 Lamictal      Bipolar Di~      9 Highly Effec~ Mild Side E~ "Lamictal ~
 3  3824 Depakene      Bipolar Di~      4 Moderately E~ Severe Side~ "Initial b~
 4   969 Sarafem       Bi-Polar /~     10 Highly Effec~ No Side Eff~ "It contro~
 5   696 Accutane      Nodular Ac~     10 Highly Effec~ Mild Side E~ "Within on~
 6  1380 Biaxin        Sinus Infe~      2 Marginally E~ No Side Eff~ "By the en~
 7    45 Carbamazepine Seizure          8 Considerably~ Moderate Si~ "reduction~
 8  1939 Ultram-Er     Cervical D~     10 Highly Effec~ Mild Side E~ "Ive been ~
 9  2576 Klonopin      Panic Diso~     10 Highly Effec~ No Side Eff~ "I immedia~
10  1093 Effexor       Depression       1 Marginally E~ Extremely S~ "the presu~
# i 1,026 more rows
```

1.3 Modeling

1.3.1 Logistic Regression: Predicting Positive vs. Negative Reviews

```
Confusion Matrix and Statistics
```

```
          Reference
Prediction    0    1
         0  548  121
         1  305 1716

               Accuracy : 0.8416
                 95% CI : (0.8273, 0.8552)
    No Information Rate : 0.6829
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.6119

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.6424
            Specificity : 0.9341
         Pos Pred Value : 0.8191
         Neg Pred Value : 0.8491
             Prevalence : 0.3171
         Detection Rate : 0.2037
   Detection Prevalence : 0.2487
      Balanced Accuracy : 0.7883

       'Positive' Class : 0
```

1.4 Discussion

1.5 References

2 Code Appendix

```
#Load necessary libraries
library(dplyr) #data manipulation
library(knitr) #table formats
library(readr)
library(kableExtra)
library(caret)
library(forcats)

#Import UC Irvine test csv file. (Initial Definition for All Code)
drug_train <- read_tsv(url("https://raw.githubusercontent.com/gkd5216/STAT380Project/refs/heads
drug_test <- read_tsv(url("https://raw.githubusercontent.com/gkd5216/STAT380Project/refs/heads/

# Summary Statistics grouping by Geographic Area.
```

```r
variable_analysis <- data.frame(
  Variable = c("reviewID", "urlDrugName", "rating", "effectiveness", "sideEffects", "condition",
               "sideEffectsReview", "commentsReview"),
  Type = c("Integer", "Categorial", "Integer", "Categorical", "Categorical", "Categorical",
           "Categorical", "Categorical", "Categorical"),
  Explanation = c(
    "Review ID",
    "Name of drug",
    "Name of condition",
    "Patient on benefits",
    "Patient on side effects",
    "Overall patient comment",
    "10 star patient rating",
    "5 step side effect rating",
    "5 step effectiveness rating"
  )
)

# Outputs Formatted Summary Table with Kable Styling tools
kable(
  variable_analysis,
  caption = "Variables used in Analysis"
  )

library(tidyverse)
library(janitor)
glimpse(drug_train)

tidy_drug_train <- drug_train %>%
  rename(id = `...1`) %>% #Renames ID column
  clean_names() %>%
  mutate(
    rating = as.numeric(rating),                # Ensure numeric
    effectiveness = as.factor(effectiveness),   # Treat as ordinal later if needed
    side_effects = as.factor(side_effects),
    condition = str_to_title(condition),        # Title case for condition
    drug_name = str_to_title(url_drug_name),    # Clean drug name
    full_review = paste(benefits_review, side_effects_review, comments_review, sep = " ") %>%
      str_squish()                              # Collapse and trim whitespace
  ) %>%
  select(id, drug_name, condition, rating, effectiveness, side_effects, full_review) %>%
  filter(!is.na(rating), !is.na(condition), condition != "")

tidy_drug_train
# Create binary sentiment variable: 1 (positive w/ rating 7-10), 0 (negative w/ rating 1-6)

drug_train <- drug_train %>%
```

```r
  mutate(sentiment = ifelse(rating >= 7, 1, 0))

drug_test <- drug_test %>%
  mutate(sentiment = ifelse(rating >= 7, 1, 0))

drug_train <- drug_train %>%
  mutate(across(c(urlDrugName, condition, effectiveness), as.factor))

drug_test <- drug_test %>%
  mutate(across(c(urlDrugName, condition, effectiveness), as.factor))

drug_train <- drug_train %>%
  mutate(urlDrugName = fct_lump(urlDrugName, n = 20),
         condition = fct_lump(condition, n = 20),
         effectiveness = fct_lump(effectiveness, n = 5))

# Match levels in test data to training data
drug_test <- drug_test %>%
  mutate(urlDrugName = fct_lump(urlDrugName, n = 20),
         condition = fct_lump(condition, n = 20),
         effectiveness = fct_lump(effectiveness, n = 5))

# Align factor levels
drug_test$urlDrugName <- factor(drug_test$urlDrugName, levels = levels(drug_train$urlDrugName))

drug_test$condition <- factor(drug_test$condition, levels = levels(drug_train$condition))
drug_test$effectiveness <- factor(drug_test$effectiveness, levels = levels(drug_train$effectiv

# Remove rows with NAs
test_data_clean <- drug_test %>%
  filter(!is.na(urlDrugName) & !is.na(condition) & !is.na(effectiveness))

# Logistic regression model
logit_model <- glm(sentiment ~ urlDrugName + condition + effectiveness,
                   data = drug_train, family = "binomial")

# Predict on clean test data
test_pred <- predict(logit_model, newdata = test_data_clean, type = "response")
test_pred_class <- ifelse(test_pred > 0.5, 1, 0)

# Confusion matrix
confusionMatrix(factor(test_pred_class), factor(test_data_clean$sentiment))
```