MAST6100 FINAL PROJECT – GODWINS BRAIMOH GB514

Predicting 30-day Hospital Readmission in Diabetic Patients using Classification Models

1. Project Objective
   The objective of this project is to predict whether a hospital encounter for a diabetic patient results in readmission within 30 days, using routinely collected administrative and clinical data. This is formulated as a binary classification problem, with readmission within 30 days defined as the positive class. The analysis is limited to a single encounter-level dataset and focuses on predictive performance rather than causal inference. Because readmissions are rare, the evaluation focuses on the performance of minority classes rather than overall accuracy.

Data cleaning, outcome definition and exploratory analysis

The dataset was imported from a CSV file and inspected before any preprocessing to confirm its structure and contents. The initial inspection confirmed that the dataset comprised 101,766 encounter-level observations and 50 variables, which included a mixture of numeric and categorical variables stored as character strings. This step ensured that variable types were correctly understood before further manipulation, demonstrating careful data handling that the audience can trust.

```
# Load dataset ----------------------------------------------------------------
df <- read.csv("diabetic_data.csv", stringsAsFactors = FALSE)

# Inspect structure -----------------------------------------------------------
str(df)                                  # view variable types
summary(df)                               # summary statistics
names(df)
```

Missing values in the dataset were initially encoded using the character "?", which required explicit recoding to standard missing value indicators. After converting these entries to NA, a total of 192,849 missing values were identified across the dataset, indicating that missingness was a substantive feature of the data. Duplicate records were also checked, and none were found, confirming that all observations represented unique hospital encounters.

```
# Replace '?' with actual NA
df[df == "?"] <- NA

# Check missing values
colSums(is.na(df))                       # count missing per column
sum(is.na(df))                           # total missing values

################################################################################
# Remove duplicates -----------------------------------------------------------
################################################################################

sum(duplicated(df))                      # number of duplicates
df <- df[!duplicated(df), ]              # remove duplicates
sum(duplicated(df))                      # confirm removal
```

Because many predictors were stored as character strings, all character variables were systematically converted to factors to ensure correct handling of categorical information in downstream classification models. This step is crucial for models such as logistic regression and linear discriminant analysis, which rely on factor encoding to construct appropriate design matrices.

```
char_cols <- sapply(df, is.character)

# Convert all character columns to factors
df[char_cols] <- lapply(df[char_cols], factor)

str(df)
```

The response variable was defined as readmission within 30 days of discharge. The original outcome variable contained three categories, so a binary target was constructed by coding encounters with readmission within 30 days as the positive class and all other outcomes as the negative class. This formulation presents the task as a binary classification problem, aligning with the practical objective of identifying short-term readmissions. Examination of the resulting class distribution revealed substantial imbalance, with approximately 11.2% of encounters classified as readmissions.

```
> table(df$readmit_binary)

   NO   YES
90409 11357
> prop.table(table(df$readmit_binary))

       NO       YES
0.8884008 0.1115992
```

```
# Common choice: Readmission within 30 days (binary classification)
df$readmit_binary <- ifelse(df$readmitted == "<30", "YES", "NO")
df$readmit_binary <- factor(df$readmit_binary)

# Check class balance
table(df$readmit_binary)
prop.table(table(df$readmit_binary))
```

Exploratory analysis focused on the numeric variables, which primarily capture healthcare utilisation such as counts of procedures, medications, and hospital visits. These variables were extracted and summarised to assess their scale and basic distributional properties.

```
# Extract numeric variables
numeric_vars <- df %>% select(where(is.numeric))

summary(numeric_vars)
```

Distributional exploration revealed that many numeric predictors exhibited strong right skewness, characterised by a large concentration of observations at low values and a small number of extreme observations. This pattern was particularly pronounced for emergency, inpatient, and outpatient visit counts and is typical of administrative healthcare data. Such skewness can influence the behaviour of distance-based and optimisation-based models if not accounted for.

Formal skewness calculations supported these observations, revealing extreme positive skew for several utilisation-related variables.

Exploratory analysis of the numeric variables revealed distributional patterns typical of healthcare utilisation data. As shown in Appendix A.1, most numeric predictors are discrete and concentrated at low values. Visit count variables, including emergency, inpatient, and outpatient visits, exhibit a large mass at zero, with a small number of encounters showing very high counts. Density plots in Appendix A.2 confirm this pattern and highlight a strong right skew in several utilisation-related variables.

Boxplots of numeric variables in Appendix A.3 further illustrate the presence of extreme values and substantial differences in scale across predictors. Visit count variables display pronounced skewness and long upper tails, while time spent in hospital shows a more moderate but still asymmetric distribution. These characteristics are relevant for modelling, as they can affect the performance of distance-based methods and neural networks and motivate the scaling of numeric predictors.

Skewness statistics support the visual findings, with particularly high positive skew observed for emergency and outpatient visit counts. Correlation analysis of the numeric predictors, presented in Appendix A.7, reveals limited multicollinearity, with most pairwise correlations remaining weak to moderate in strength. This suggests that the numeric predictors capture distinct aspects of patient utilisation.

Comparisons between selected numeric predictors and readmission status show limited separation between outcome classes. As shown in Appendix A.5, the distribution of time spent in hospital overlaps substantially between readmitted and non-readmitted encounters. A similar pattern is observed for the number of laboratory procedures in Appendix A.6, with only minor differences in central tendency. These results suggest that individual predictors exhibit limited discriminatory power, underscoring the need for multivariate classification models to capture more complex relationships associated with readmission risk.

Preparation for modelling and train–test split

To prepare the data for modelling, a reduced modelling dataset was constructed from the full dataset to improve stability and interpretability. Clinically relevant demographic variables, admission characteristics, and healthcare utilisation measures were retained, while variables with extensive missingness or limited relevance were excluded. Observations with missing values in any selected predictor were removed to ensure consistent preprocessing across all models. After filtering, the final modelling dataset contained 99,493 observations and 15 variables, including the binary readmission outcome.

```r
# Start from original df (already has "?" -> NA)
df2 <- df

# Create binary outcome: readmission within 30 days
df2$readmitted30 <- ifelse(df2$readmitted == "<30", "YES", "NO")
df2$readmitted30 <- factor(df2$readmitted30, levels = c("NO", "YES"))

# Build a smaller, clean modelling dataset
df_model_small <- df2 %>%
  mutate(
    race                    = factor(race),
    gender                  = factor(gender),
    age                     = factor(age),
    admission_type_id       = factor(admission_type_id),
    discharge_disposition_id= factor(discharge_disposition_id),
    admission_source_id     = factor(admission_source_id)
  ) %>%
  dplyr::select(
    readmitted30,
    race, gender, age,
    admission_type_id, discharge_disposition_id, admission_source_id,
    time_in_hospital, num_lab_procedures, num_medications,
    num_procedures, number_diagnoses,
    number_emergency, number_inpatient, number_outpatient
  ) %>%
  drop_na()          # drop rows that have NA in any of these

str(df_model_small)
table(df_model_small$readmitted30)
```

The modelling dataset preserved the original class imbalance, with 11,169 readmissions and 88,324 non-readmissions, corresponding to approximately 11.2% of encounters resulting in readmission within 30 days. Retaining this imbalance ensures that model performance reflects realistic prediction conditions rather than artificially balanced data.

Model performance was evaluated using a stratified train–test split. Seventy per cent of the data were used for training, and thirty per cent for testing, while maintaining class

proportions in both subsets. A fixed random seed was used to ensure reproducibility of the split.

```
set.seed(123)
train_index <- createDataPartition(df_model_small$readmitted30, p = 0.7, list = FALSE)

train <- df_model_small[train_index, ]
test  <- df_model_small[-train_index, ]

X_train <- train %>% dplyr::select(-readmitted30)
y_train <- train$readmitted30

X_test  <- test %>% dplyr::select(-readmitted30)
y_test  <- test$readmitted30
```

Logistic regression and linear discriminant analysis

A logistic regression model was fitted as a baseline classifier using all selected predictors. Predicted probabilities were converted to class labels using a threshold of 0.5.

```
glm_model <- glm(readmitted30 ~ ., data = train, family = binomial)
summary(glm_model)

glm_prob <- predict(glm_model, test, type = "response")
glm_pred <- factor(ifelse(glm_prob > 0.5, "YES", "NO"), levels = c("NO", "YES"))

confusionMatrix(glm_pred, y_test)

glm_auc <- roc(y_test, glm_prob)
auc(glm_auc)
```

The model achieved an overall accuracy of 0.887, which is close to the no-information rate of 0.888. However, balanced accuracy was only 0.509, with very high sensitivity (0.997) and extremely low specificity (0.021), indicating that the model almost always predicted non-readmission. The AUC for logistic regression was 0.663, indicating moderate discrimination despite poor minority-class classification.

Linear discriminant analysis was fitted as an alternative linear classifier. Although LDA relies on stronger distributional assumptions than logistic regression, it provides insight into whether a generative linear model offers any improvement in this setting.

```
lda_model <- lda(readmitted30 ~ ., data = train)
lda_pred  <- predict(lda_model, X_test)$class
confusionMatrix(lda_pred, y_test)

lda_prob <- predict(lda_model, X_test)$posterior[,2]
lda_auc  <- roc(y_test, lda_prob)
auc(lda_auc)
```

LDA achieved an overall accuracy of 0.884 and a balanced accuracy of 0.521, slightly higher than that of logistic regression. Sensitivity remained high at 0.989, while specificity increased modestly to 0.053. The AUC for LDA was 0.663, essentially identical to that of logistic regression, suggesting limited improvement from the generative modelling approach.

K-Nearest neighbours and random forest models

A k-nearest neighbours classifier was implemented using only numeric predictors. Because KNN is sensitive to scale, all numeric variables were standardised using centring and scaling based on the training data.

```
num_cols <- c("time_in_hospital", "num_lab_procedures", "num_medications",
              "num_procedures", "number_diagnoses",
              "number_emergency", "number_inpatient", "number_outpatient")

X_train_num <- train[, num_cols]
X_test_num  <- test[, num_cols]

preproc <- preProcess(X_train_num, method = c("center", "scale"))
X_train_scaled <- predict(preproc, X_train_num)
X_test_scaled  <- predict(preproc, X_test_num)

knn_pred <- knn(train = X_train_scaled,
                test  = X_test_scaled,
                cl    = y_train,
                k = 5)

confusionMatrix(knn_pred, y_test)
```

With k set to five, the model achieved an overall accuracy of 0.878 but a balanced accuracy of only 0.511. Sensitivity remained high at 0.985, while specificity was low at 0.038, indicating a strong bias toward the majority class. These results are consistent with the application of distance-based classifiers to imbalanced datasets.

A random forest model was fitted to capture nonlinear relationships and interactions automatically. The model was trained using 300 trees, with five predictors considered at each split.

```
rf_model <- randomForest(readmitted30 ~ ., data = train, ntree = 300, mtry = 5)
rf_pred  <- predict(rf_model, test)
confusionMatrix(rf_pred, y_test)

rf_prob <- predict(rf_model, test, type = "prob")[,2]
rf_auc  <- roc(y_test, rf_prob)
auc(rf_auc)
```

The random forest achieved an overall accuracy of 0.887 and a balanced accuracy of 0.510, which is similar to the results obtained with the linear models. Sensitivity was high at 0.996, while specificity remained low at 0.023. The AUC was 0.644, slightly lower

than that of the linear classifiers, indicating limited gains from nonlinear modelling in this setting.

Neural network model

A feed-forward neural network was fitted using the exact scaled numeric predictors as the KNN model. The network consisted of a single hidden layer with ten units and included L2 regularisation.

```
y_train_num <- ifelse(y_train == "YES", 1, 0)
y_test_num  <- ifelse(y_test  == "YES", 1, 0)

X_train_nn <- as.matrix(X_train_scaled)
X_test_nn  <- as.matrix(X_test_scaled)

nn_model <- nnet(
  x = X_train_nn,
  y = y_train_num,
  size = 10,
  maxit = 200,
  decay = 0.001,
  linout = FALSE,   # classification, not regression
  trace = FALSE
)

# Predicted probabilities
nn_prob <- predict(nn_model, X_test_nn, type = "raw")

# Convert to class labels
nn_pred <- factor(ifelse(nn_prob > 0.5, "YES", "NO"),
                  levels = c("NO", "YES"))




confusionMatrix(nn_pred, y_test)

# AUC
nn_auc <- roc(y_test, as.numeric(nn_prob))
auc(nn_auc)
```

Despite achieving an overall accuracy of 0.887, the neural network classified almost all observations as non-readmissions. This resulted in a balanced accuracy of approximately 0.50, with sensitivity close to 1 and specificity near 0. The neural network produced the lowest AUC among all models, at approximately 0.62, highlighting its sensitivity to class imbalance when no reweighting or resampling is applied.

Model comparison and evaluation

Overall model performance was compared using accuracy, AUC, sensitivity, specificity, and balanced accuracy. Accuracy values were similar across all models and closely matched the no-information rate, reflecting the dominance of the non-readmission class. Balanced accuracy values ranged narrowly between approximately 0.50 and

0.52, indicating limited improvement in identifying readmissions across all classifiers.

```
results <- data.frame(
  Model = c("GLM", "LDA", "KNN (k=5)", "Random Forest", "Neural Network"),
  Accuracy = c(
    confusionMatrix(glm_pred, y_test)$overall["Accuracy"],
    confusionMatrix(lda_pred, y_test)$overall["Accuracy"],
    confusionMatrix(knn_pred, y_test)$overall["Accuracy"],
    confusionMatrix(rf_pred, y_test)$overall["Accuracy"],
    confusionMatrix(nn_pred, y_test)$overall["Accuracy"]
  ),
  AUC = c(
    auc(glm_auc),
    auc(lda_auc),
    NA,                    # KNN has no probability output
    auc(rf_auc),
    auc(nn_auc)
  )
)
```

Among the models considered, linear discriminant analysis achieved the highest balanced accuracy, while logistic regression and LDA produced the highest AUC values. The neural network did not outperform simpler models, demonstrating that increased model complexity does not necessarily lead to improved predictive performance in highly imbalanced healthcare datasets.

Conclusion

This project implemented and evaluated a complete classification pipeline for predicting 30-day hospital readmission among diabetic patients using routinely collected hospital encounter data. Five classification models were fitted and compared, including classical statistical methods and a neural network, using a consistent and reproducible workflow in R.

Across all models, overall accuracy remained close to the no-information rate, reflecting the strong class imbalance in the data. While logistic regression, linear discriminant analysis, k-nearest neighbours, random forest, and the neural network all achieved accuracies of approximately 0.88, balanced accuracy values remained low, ranging from 0.50 to 0.52. This indicates that none of the models were able to substantially improve the identification of readmissions without sacrificing performance on the majority class. Linear discriminant analysis achieved the highest balanced accuracy, while logistic regression and LDA produced the highest AUC values at approximately 0.66. The neural network achieved the lowest AUC and exhibited the most substantial bias toward predicting non-readmission.

These findings highlight that increased model complexity does not necessarily lead to improved predictive performance in highly imbalanced healthcare datasets. In

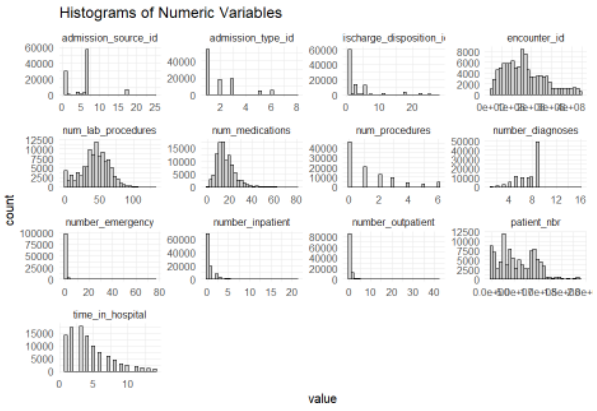
particular, the neural network failed to outperform simpler models when trained without reweighting or resampling strategies, underscoring the importance of explicitly addressing class imbalance rather than relying solely on model flexibility.

Overall, this analysis highlights the challenges of predicting hospital readmissions using administrative data and underscores the limitations of standard classification approaches in this context. Future work could explore alternative strategies such as cost-sensitive learning, resampling techniques, or threshold optimisation to improve minority-class detection. Despite these limitations, the project offers a transparent and reproducible comparison of classification models, meeting the methodological requirements of the assignment.

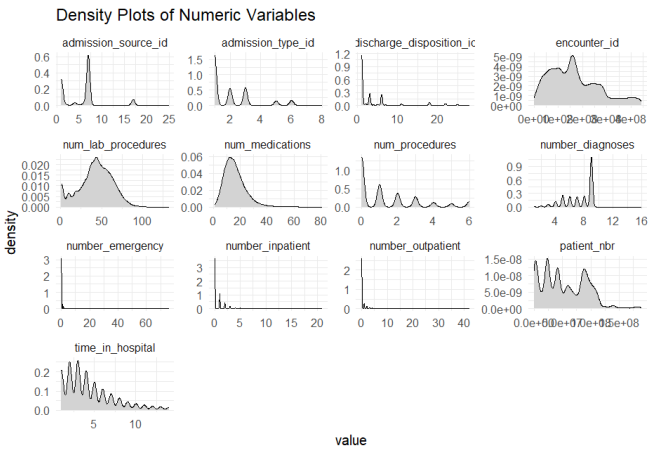
Appendix A: Exploratory Data Analysis Figures

This appendix contains the exploratory figures referenced in the main report.

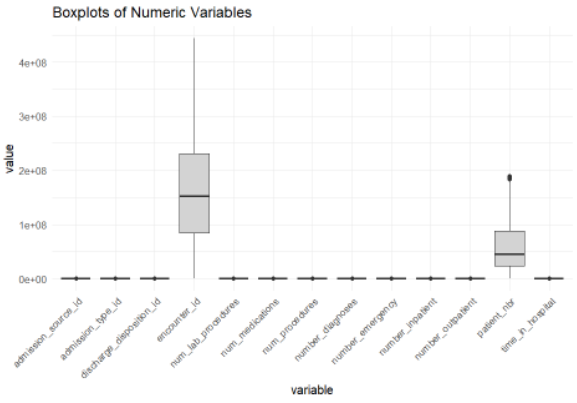
Appendix A.1



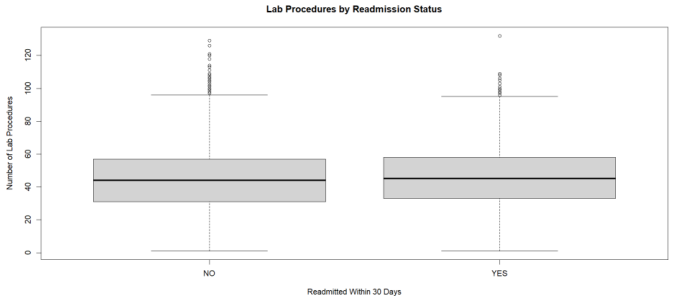
Appendix A.1 Histograms of numeric variables, showing discrete structure and right-skewed distributions in healthcare utilisation measures.



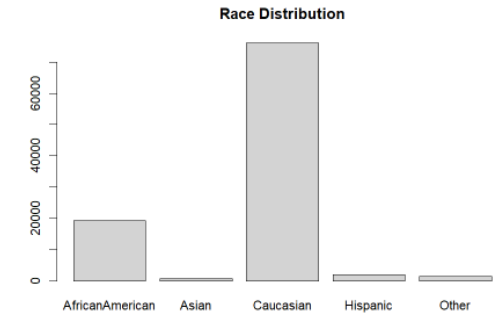
AppendixA.2 Density plots of numeric variables, highlighting heavy right tails and zero inflation in visit count variables.



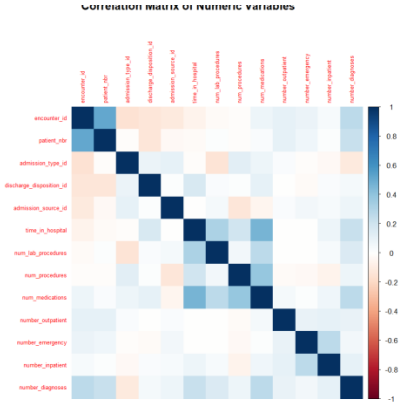
Appendix A.3 Boxplots of numeric variables on a common scale, illustrating extreme values and differences in scale across predictors.



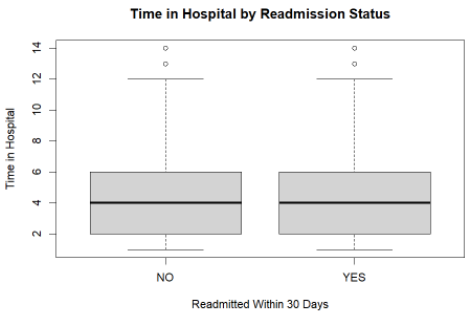
Appendix A.6 1Boxplot of number of laboratory procedures by readmission status, indicating limited separation between readmitted and non-readmitted encounters



Appendix A.4 Distribution of patient race categories in the dataset.



Appendix A.7 Correlation matrix of numeric predictors, showing generally weak to moderate pairwise correlations.



Appendix A.5 Boxplot of time in hospital by readmission status, showing substantial overlap between outcome classes.

Github Repository - https://github.com/Gbraimoh/MAST6100-IndividualProject