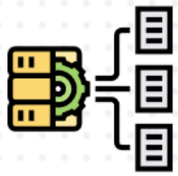
# PREDICTING HOSPITAL ADMISSION AT EMERGENCY DEPARTMENT TRIAGE

*By Carol Calderon*

# PROJECT WORKFLOW

BUSINESS QUESTION

DATA ACQUISITION



EXPLORATION

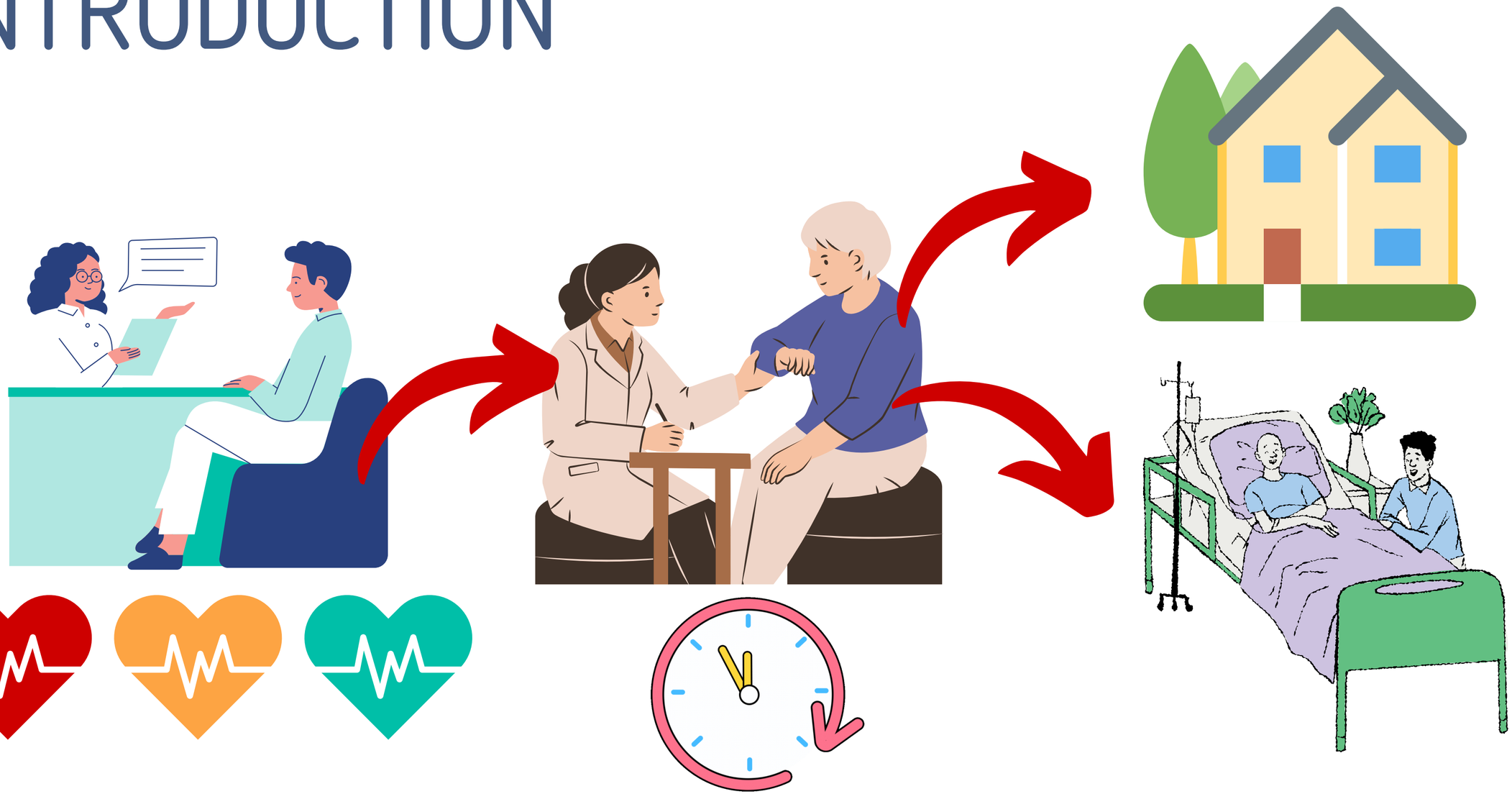
ANALISIS

MODELING

INSIGHTS & CONCLUSIONS

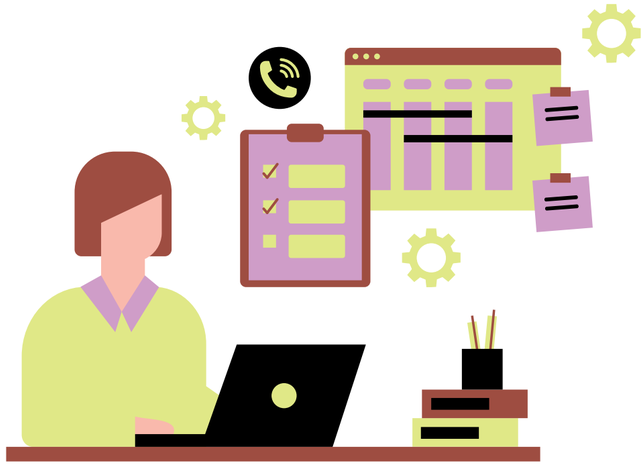
COMMUNICATE FINDINGS

# INTRODUCTION

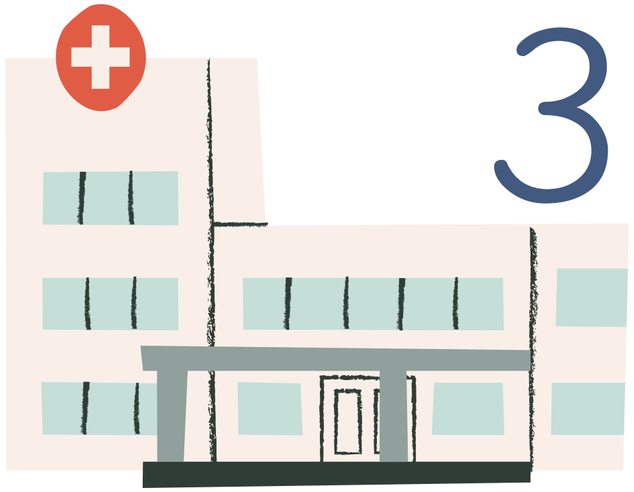


# INTRODUCTION

*Using machine learning, how might we predict the patient admission into the hospital in the emergency department to provide better information that helps the different stakeholders make better decisions*



# DATA ADQUISITION




85K
75K
35K



*The dataset used in this project was originally collected and published by Hong WS, Haimovich AD, Taylor RA (2018) in their paper, 'Predicting hospital admission at emergency department triage using machine learning.' PLoS ONE 13(7): e0201016. https://doi.org/10.1371/journal.pone.0201016*

# DATA DESCRIPTION

Data points
+560.000
Patients

~970
Features

Demographics:
Triage evaluation:
Chief complaint:
Hospital usage statistic:
Past medical history:
Outpatient medications:
Imaging / EKG counts:
Historical vitals and labs:

Admitted
or
Discharge

# CLEAN UP

*Data understanding*

*Regular expressions*

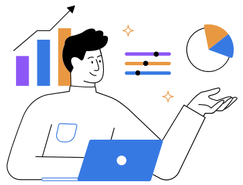
*Data types*

*Missing values: missingno.*

*Feature processing*

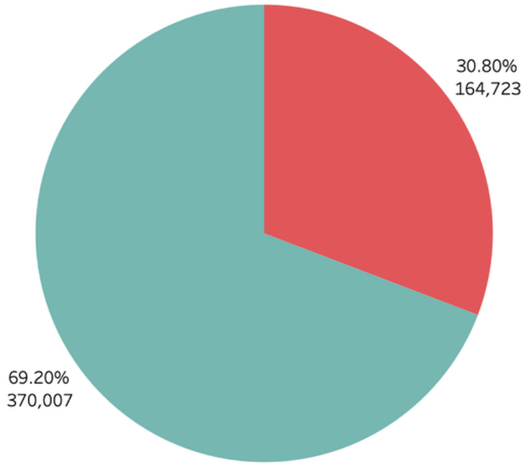
*One Hot Encoding*

534.730 *patients* 602 *features*

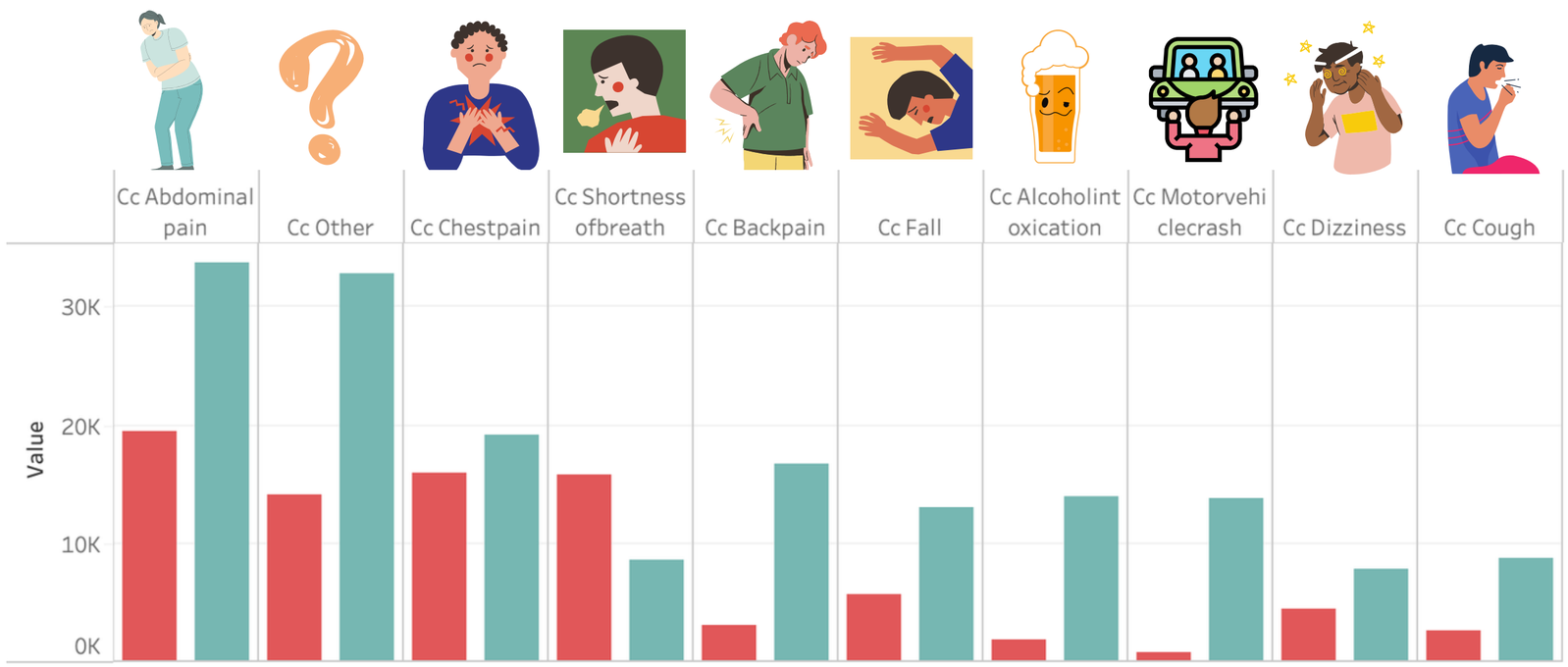


# EXPLORATION AND ANALYSIS

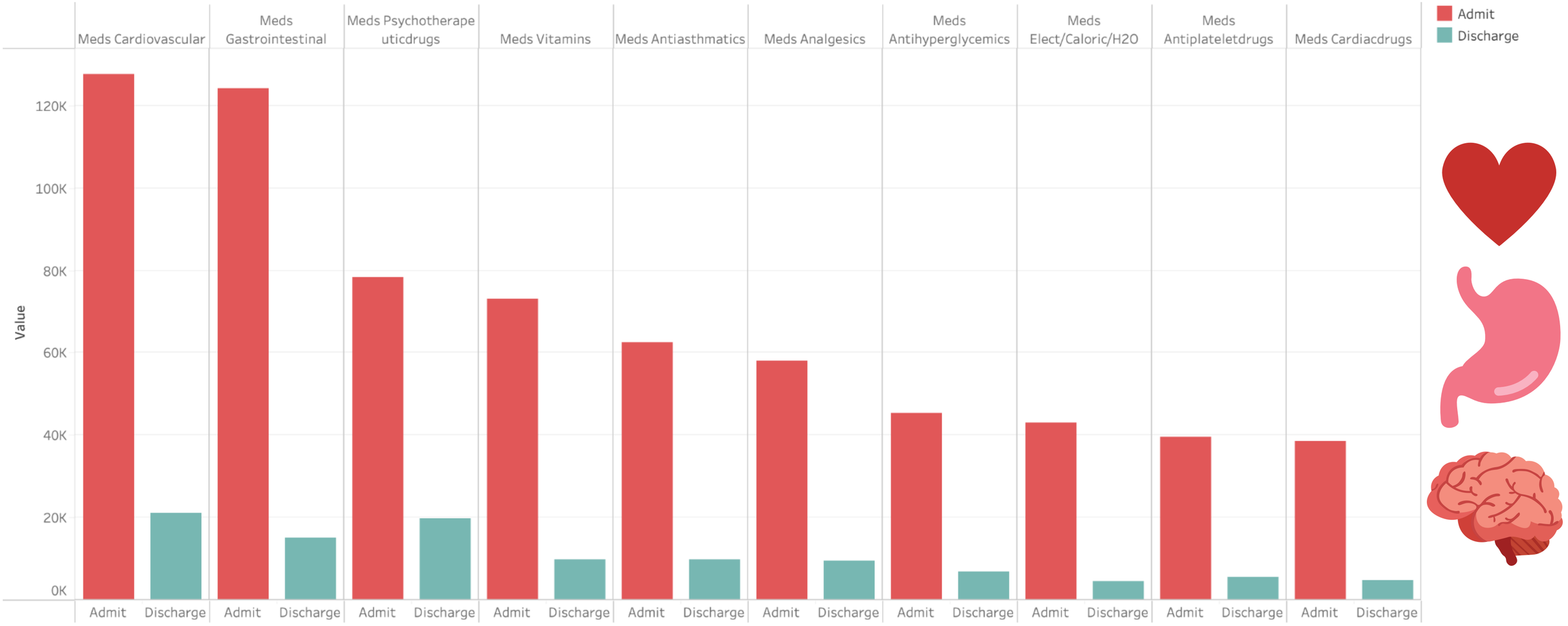
## *Distribution of admitted and discharged patients*




## *Most frequent chief complaints and its distribution between Admit and Discharge*



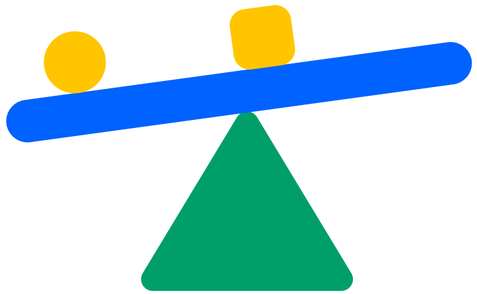
# Most frequent medication categories



# PRE-MODELING

*IMBALANCED DATA*

*DOWNSAMPLING*

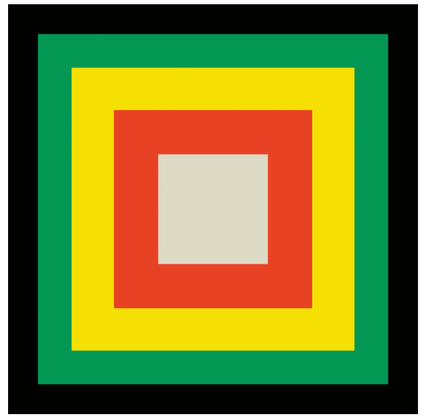


*VARIANCE THRESHOLD* ✓

*SELECT K-BEST*

*PCA*

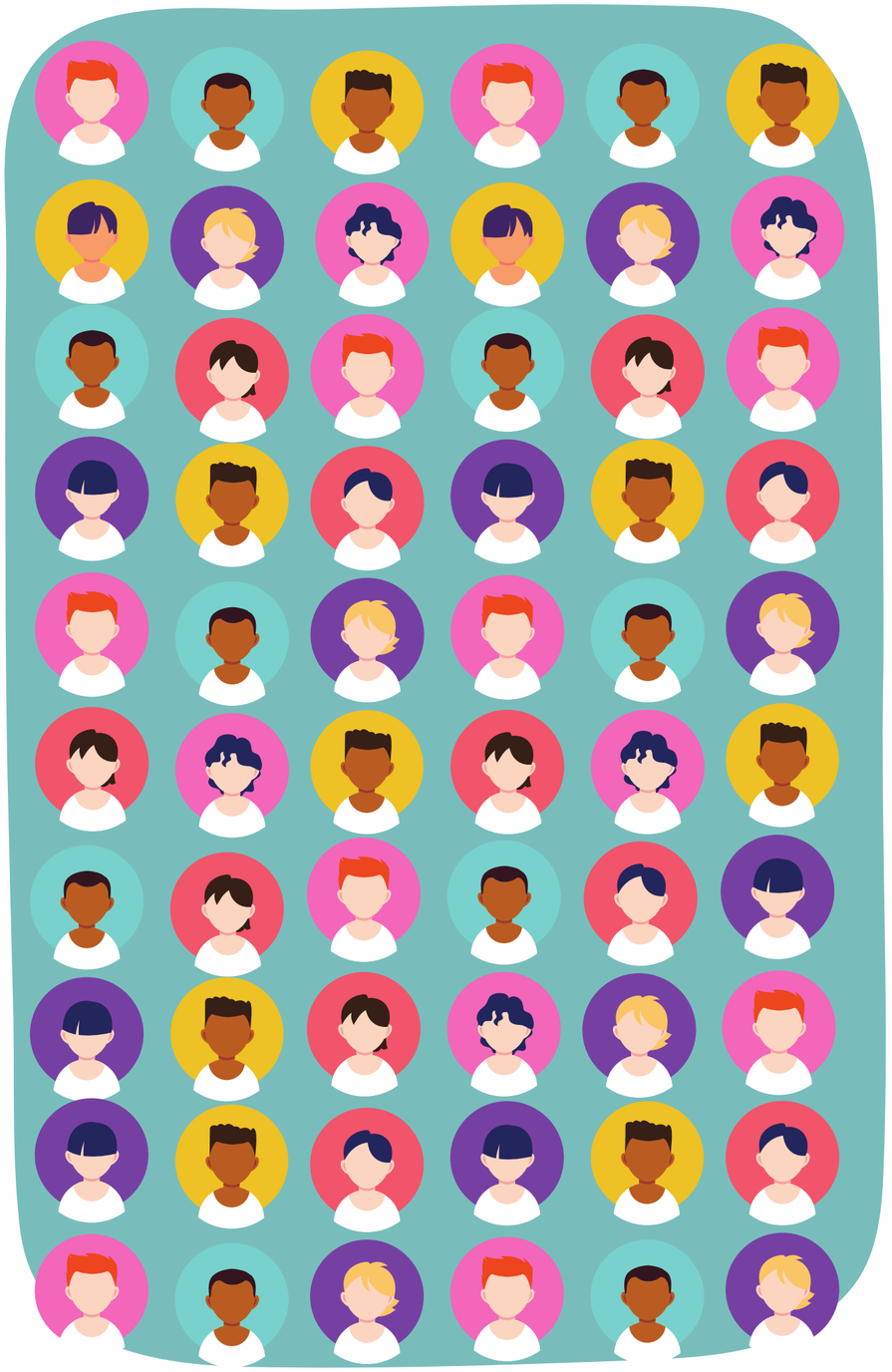
*HYPERPARAMETER OPTIMIZATION*



| MODEL | PARAMETERS | OVERAL ACCURACY | PRECISION Admitted | SENSITIVITY Admitted |
|---|---|---|---|---|
| BASELINE LOGISTIC REGRESSION | solver='lbfgs', max_iter=100000, random_state=1<br>NO balanced - NO reduced dataframe | 86% | 83% | 67% |
| OP. LOGISTIC REGRESSION ✓ | solver='lbfgs', max_iter=100000, random_state=1, C=0.1<br>Balanced and Reduced dataframe | 84% | 71% | 81% |
| OP. RANDOM FOREST | max_depth=11, n_estimators=100, random_state=1<br>Balanced and Reduced dataframe | 84% | 72% | 76% |
| OP. XGBOOST ✓ | max_depth=11, n_estimators=150, learning_rate=0.1, random_state=1<br>Balanced and Reduced dataframe | 85% | 72% | 81% |

# BEST MODEL RESULTS

- *TRUE ADMITTED*
- *TRUE DISCHARGE*
- *FALSE ADMITTED*
- *FALSE DISCHARGE*



# CONCLUSIONS

*Model Performance*

*Feature Importance*

*Model Interpretation*

*Generalizability*

*Real escenarios implementation*

*Future Directions*

# QUESTIONS

LET'S STAY IN CONTACT