PYTHON PROJECT A4

Created by: Louise and Julie

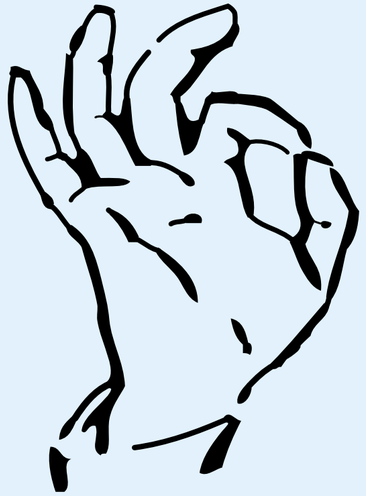
DIABETIC PATIENTS READMISSIONS

Understanding whether a patient is likely to be **readmitted** to a hospital is crucial. This information is essential for adjusting the **treatment plan** to prevent a recurrence of hospitalization.

IS OUR PROBLEM WORTH IT?

Personalization of Care

Resource Optimization



Cost Reduction

Improvement of Discharge Protocols

Continuous Monitoring

VARIABLES

race (Caucasian, Others)
gender (Male, Female)
age (0-80)
time_in_hospital (integer)
num_lab_procedures (integer)
num_procedures (integer)
num_medications (integer)
number_diagnoses (int)
max_glu_serum
(test, >200, <300, Norm)
diag 1/2/3 (type of diagnosis)

(test, >200, <300, Norm) A1Cresult
(variation : steady, no, up, down) metformin
(variation : steady, no, up, down) insulin
(yes or no) change
(yes or no if diabeth medicament taken)diabetesMed
(<30, >30, No) readmitted
(string) admission_type
(string) discharge_disposition
(string) admission_source

INS AND OUTS

we want to predict if a patient will return
as
being readmitted = 1 for ‘<30 days’ or ‘>30 days’
0 else

we are studuying correlations between “readmitted”
and the different variables previously described
like :
-Does one race readmitted more frequently than
another ?
-does test result influence readmission ?
-prescribed medicaments indicate a possible
readmission ?

HYPOTHESIS

1. One can assume that older patients are more likely to be readmitted.
2. Test results would also influence.
3. If it's an emergency, the patient will come back because they may be in a critical condition.
4. If they stay at the hospital for a long time, perhaps they will return more frequently.
5. If there have been many diagnoses, the patient will return.

TABLE OF CONTENT

1. Load data and dictionnaries related to variables
2. First data visualisation
3. Missing values, Duplicates, Reincoding, Value to int
4. Encoding
5. Visualisation
6. Prediction

VARIABLES ADDED AND DROPED

DATA PREPARATION

Droped values:

- Too many missing values: weight, payer_code, medical_specialty
- Not prescribed medicines: 'examide', 'citoglipton', 'metformin-rosiglitazone'
- Rows with remaining missing values: 3624 out of 98052

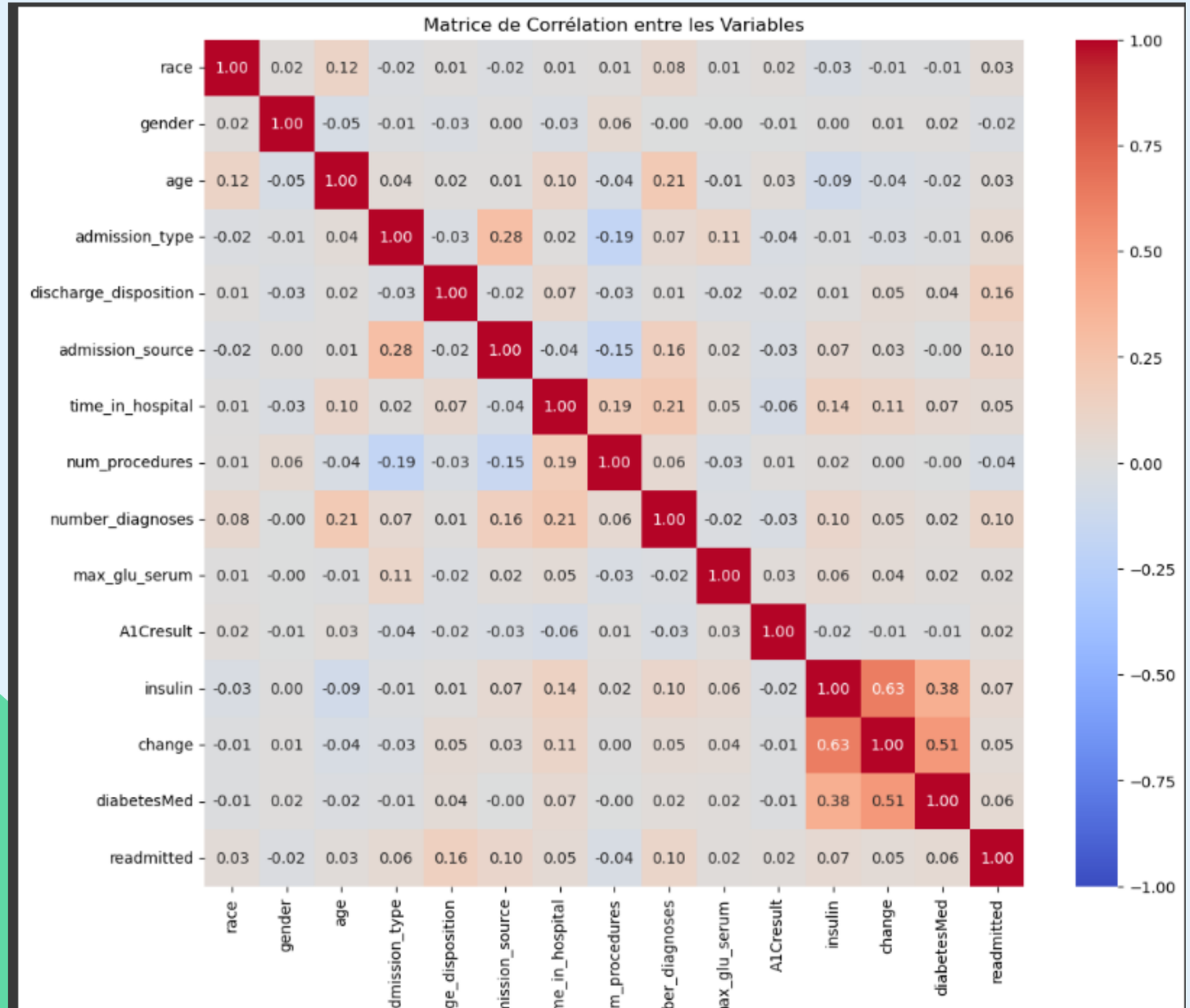
Changed values:

- Dico applications + ICD diag
- readmitted into binary column
- ‘age’ into integer column

New values:

- Rare_medication to remplace medication taken by less than 2% of the population
- readmitted_nbr: number of readmissions per patient
- df_medocs_diabeth : dataset for only patient whom diabeth medicines has been prescribed

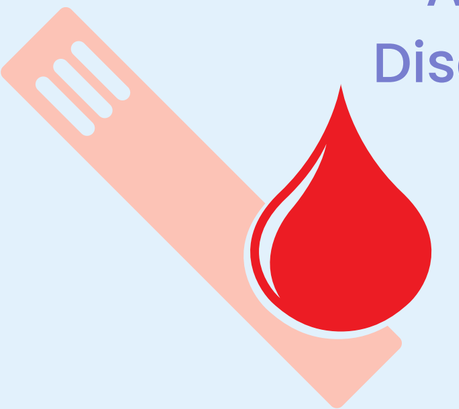
TENDANCES OBSERVED



TENDANCES OBSERVED

DISCRET VALUES

Race
Gender
Diabet medicines taken or not
Medicines taken
Admission Type
Admission source
Discharge disposition
Diagnostics



CONTINOUS VALUES

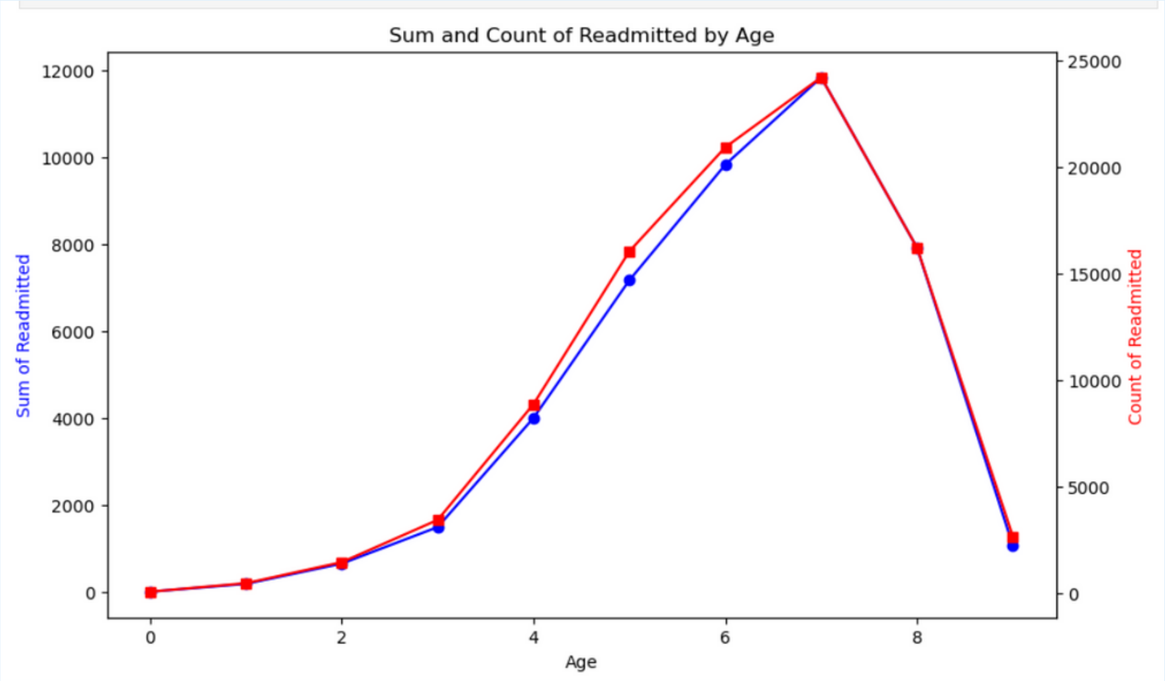
Number of emergency
Number of Lab procedures
Number of medications taken
Number outpatients
Number emergency
Number inpatients
Time spent in hospital
Age

TENDANCES OBSERVED

DISCRET VALUES

Seems to not have an impact on the target:

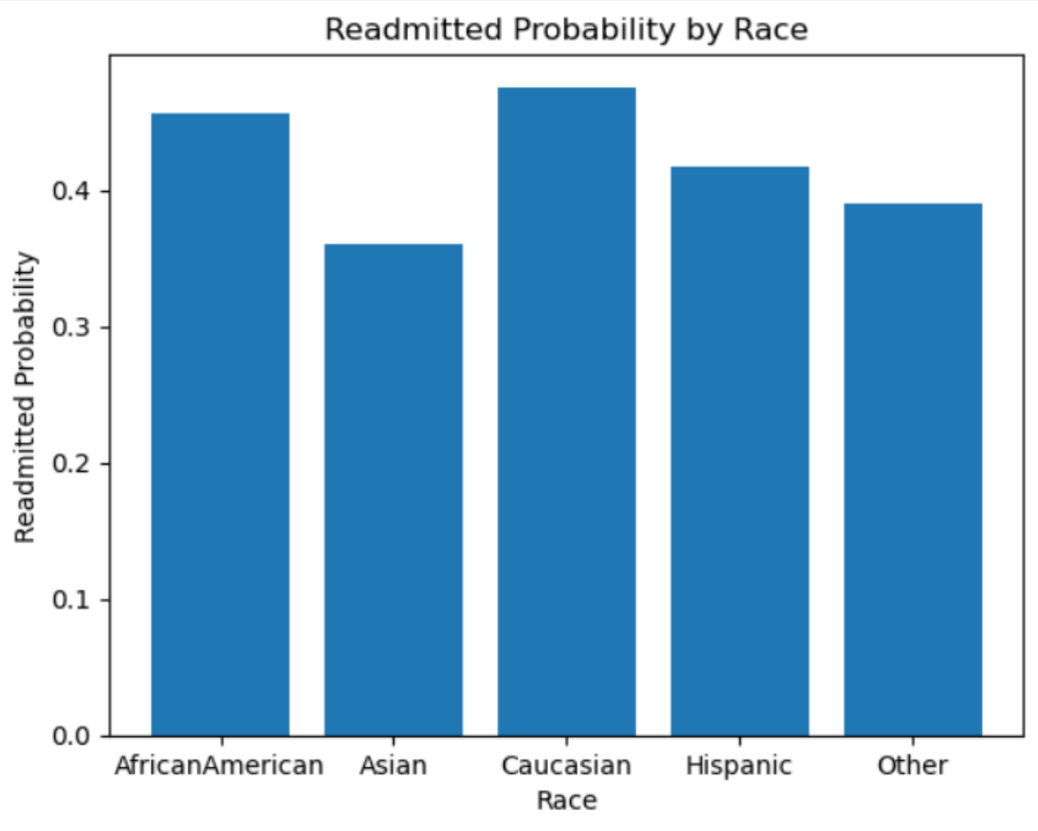
- gender
- consomation of diabets medicines
- age

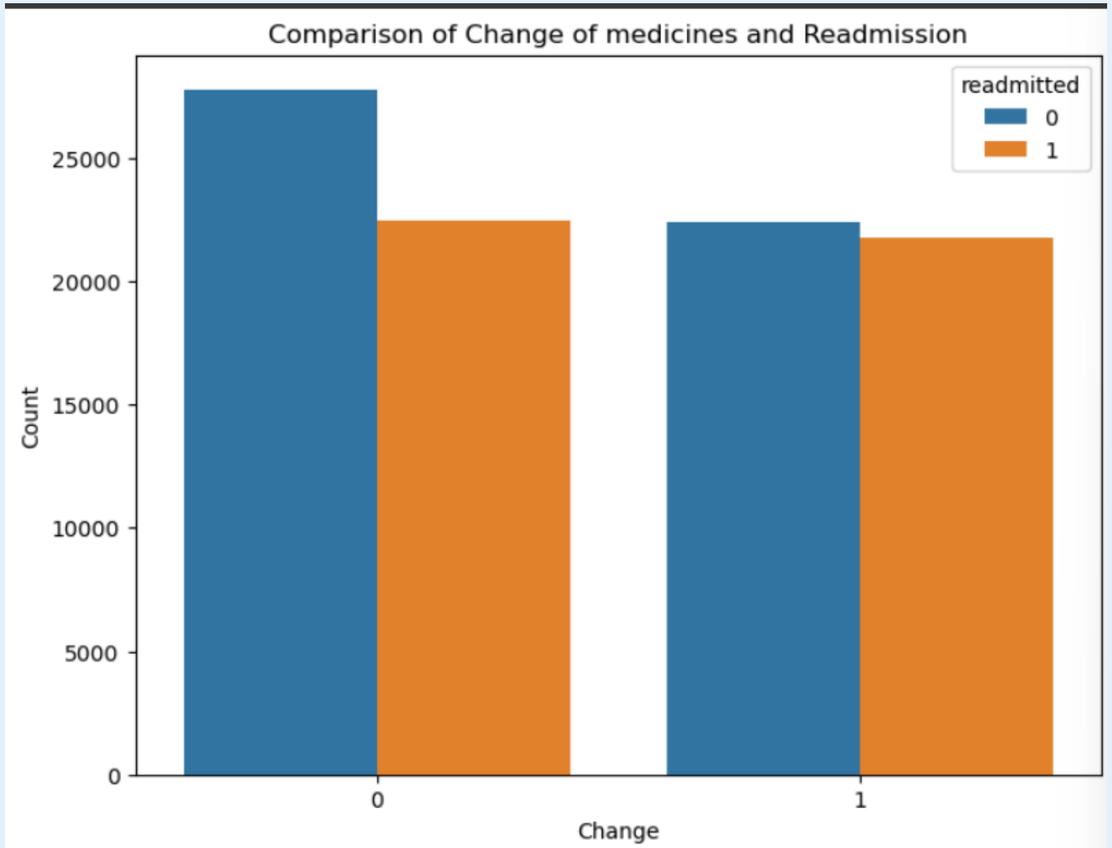


Others

TENDANCES OBSERVED

DISCRET VALUES

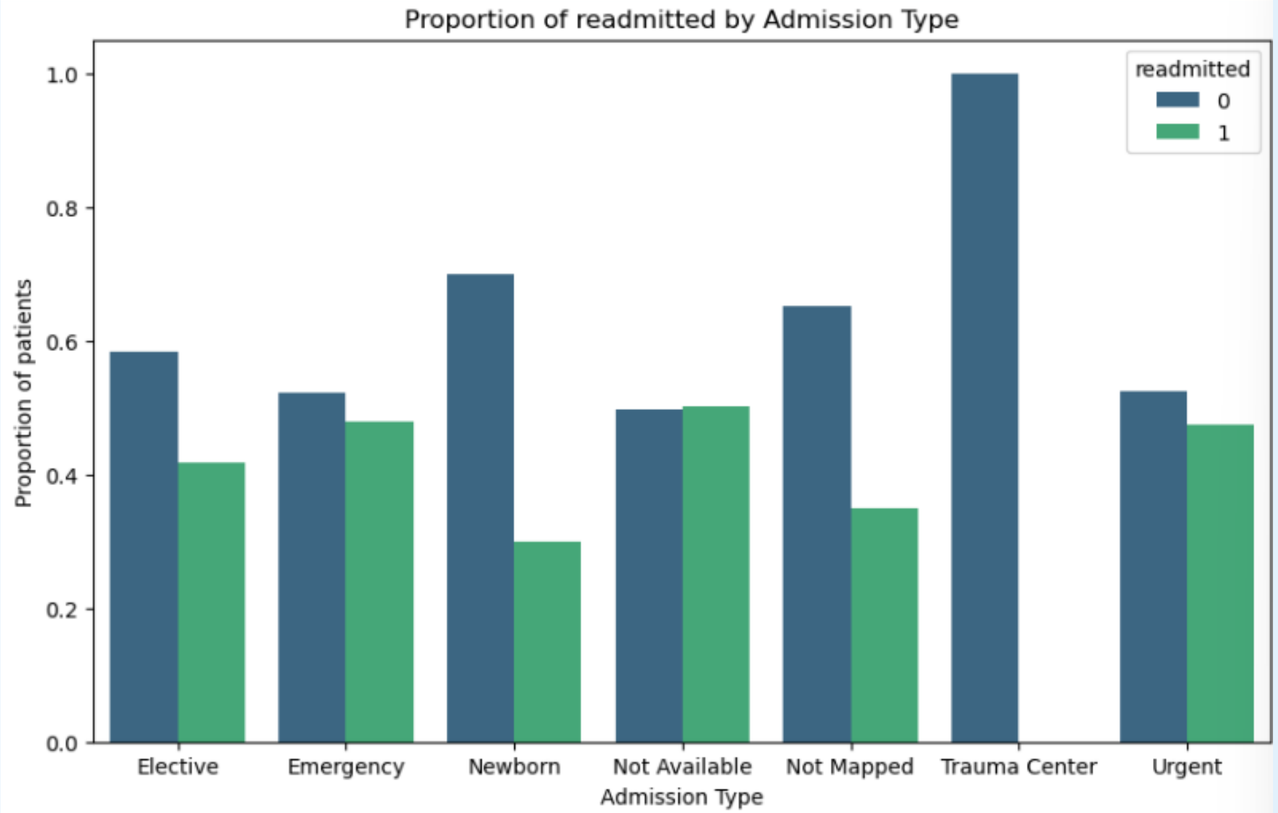


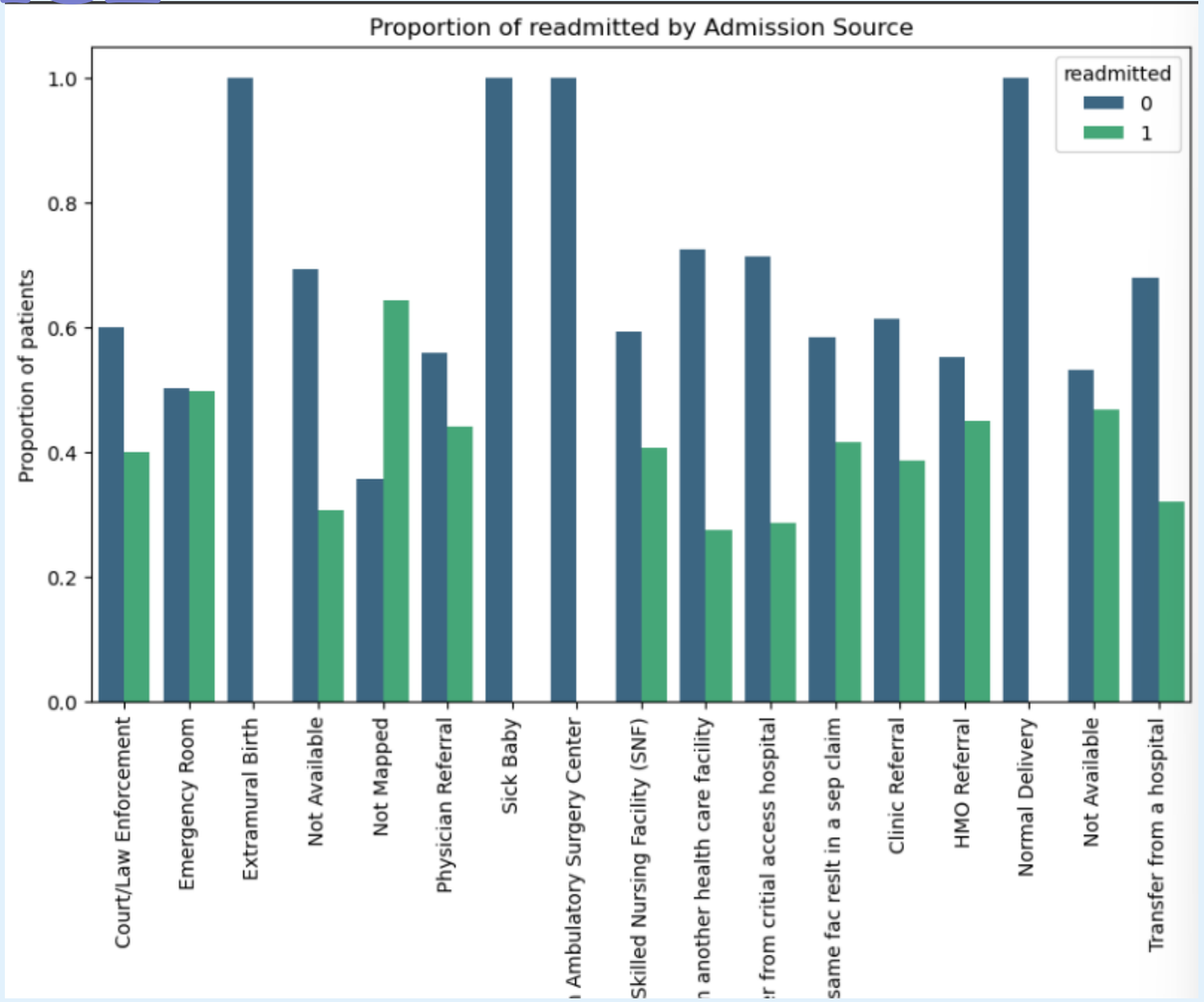


TENDANCES OBSERVED

DISCRET VALUES

ADMISSION TYPE VS SOURCE

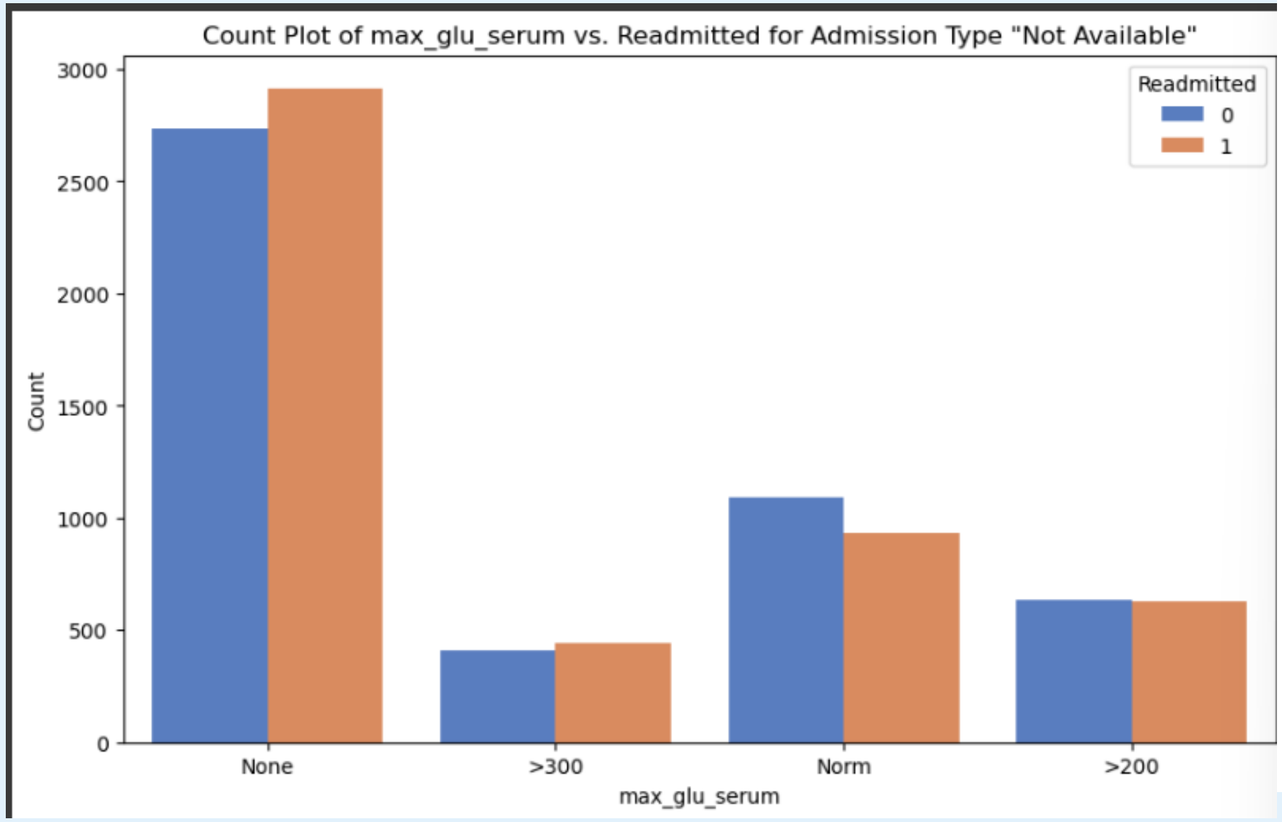


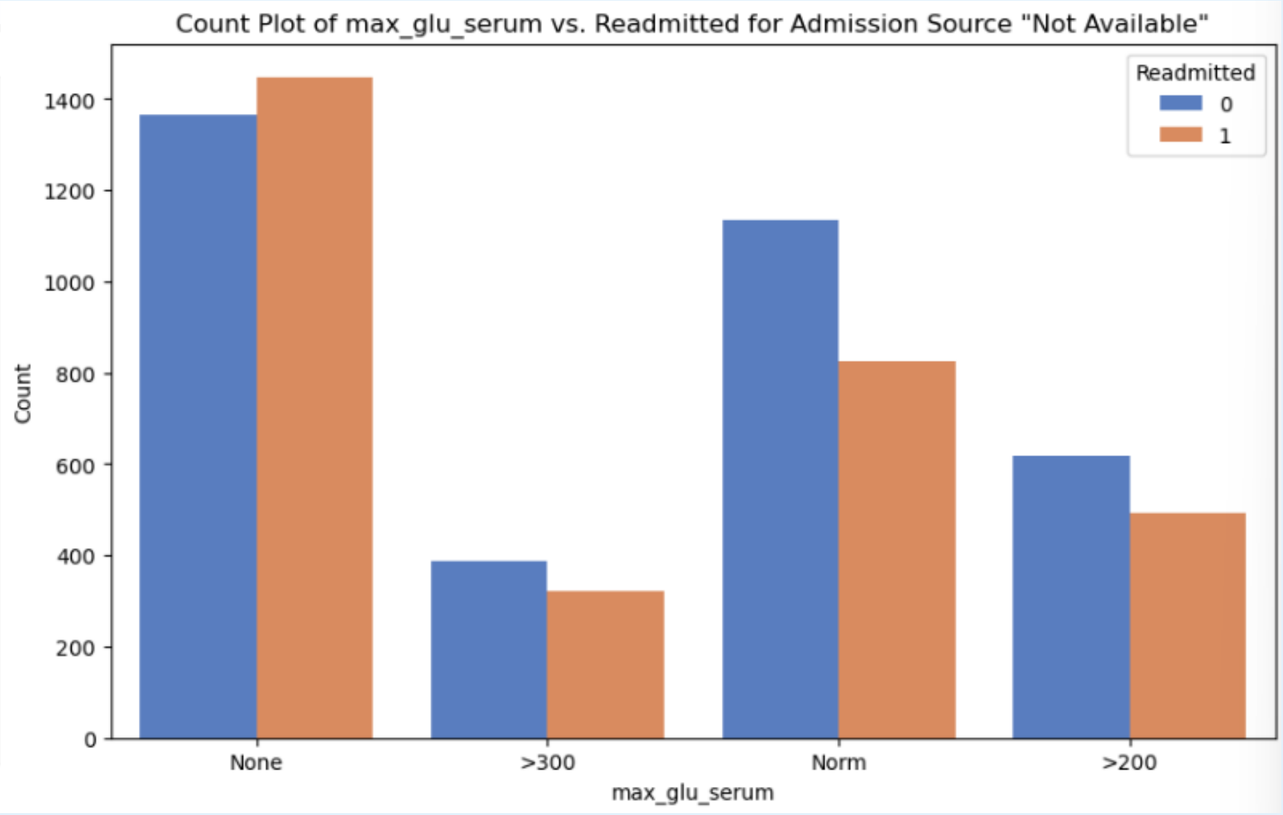


TENDANCES OBSERVED

DISCRET VALUES

ADMISSION TYPE VS SOURCE

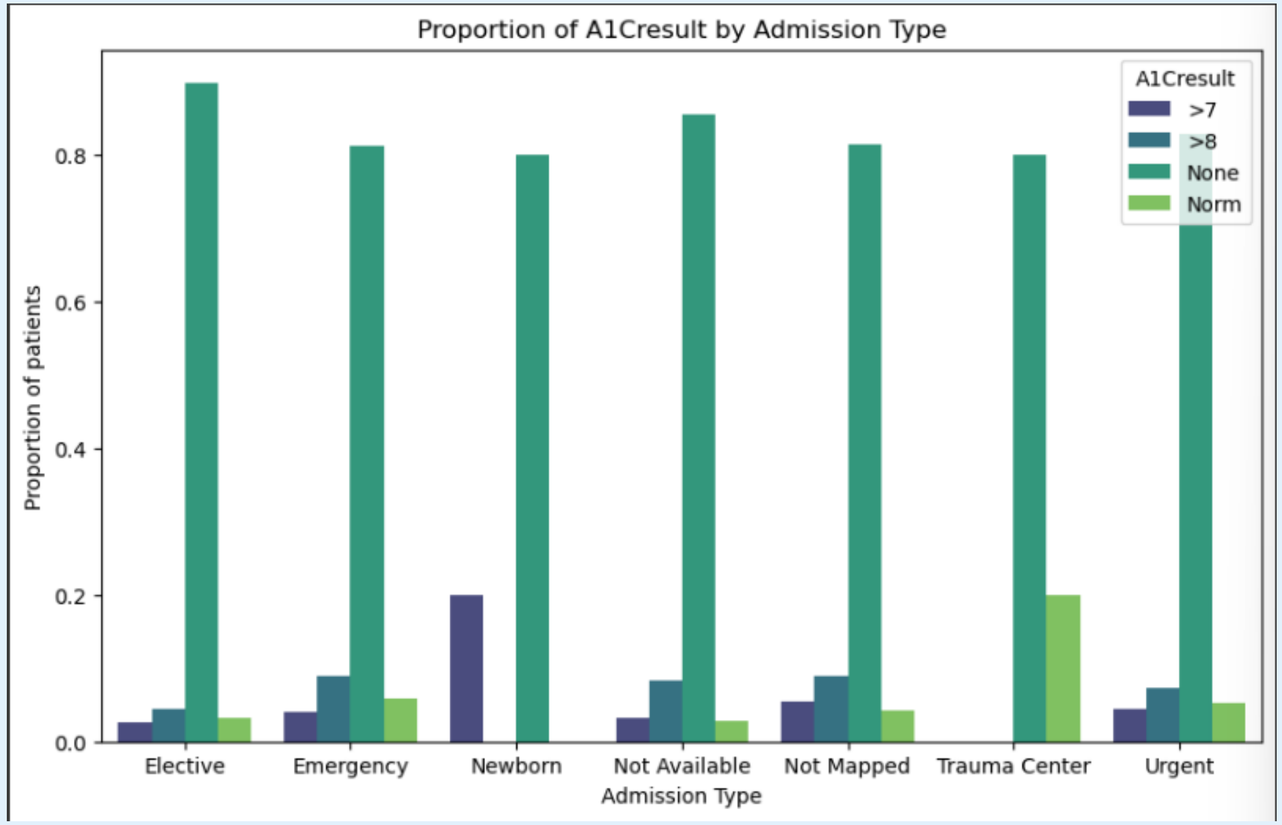


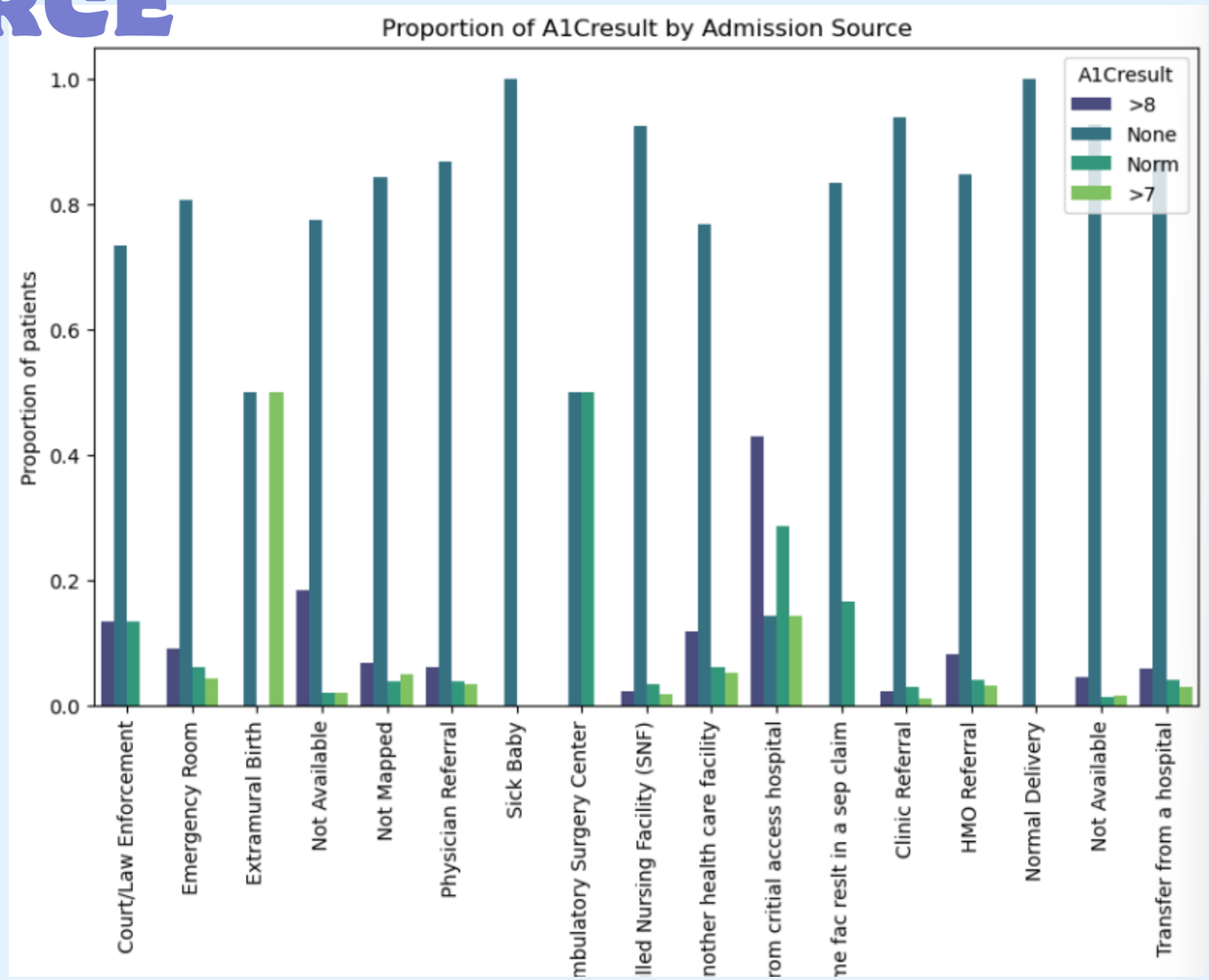


TENDANCES OBSERVED

DISCRET VALUES

ADMISSION TYPE VS SOURCE

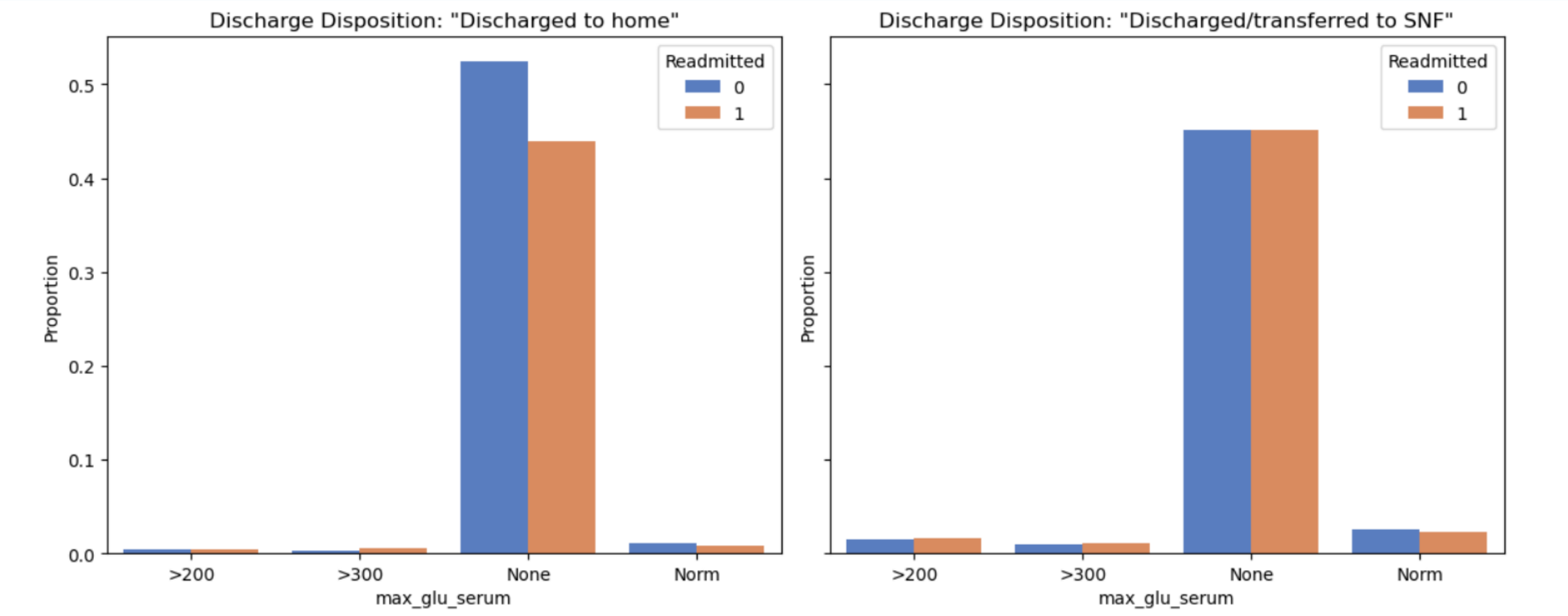




TENDANCES OBSERVED

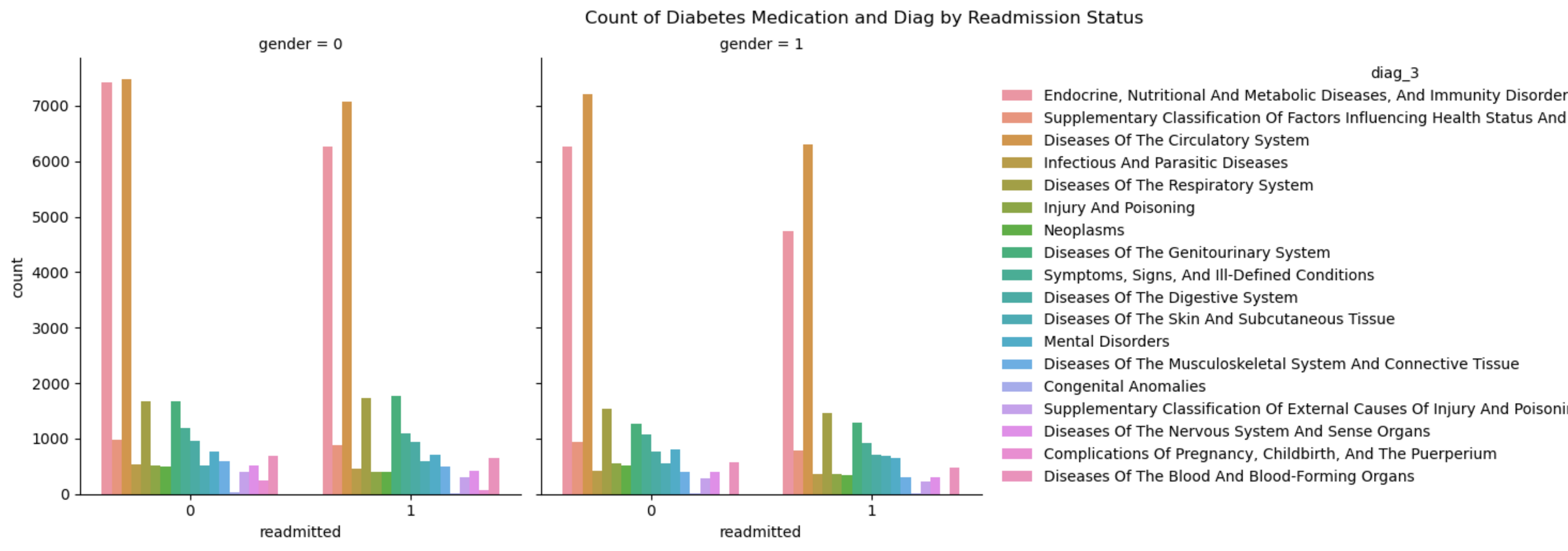
DISCRET VALUES

DISCHARGE FOR TWO MOST DESCRIBED



TENDANCES OBSERVED

DIAG 3 (SAME BEHAVIOUR FOR 1/2)



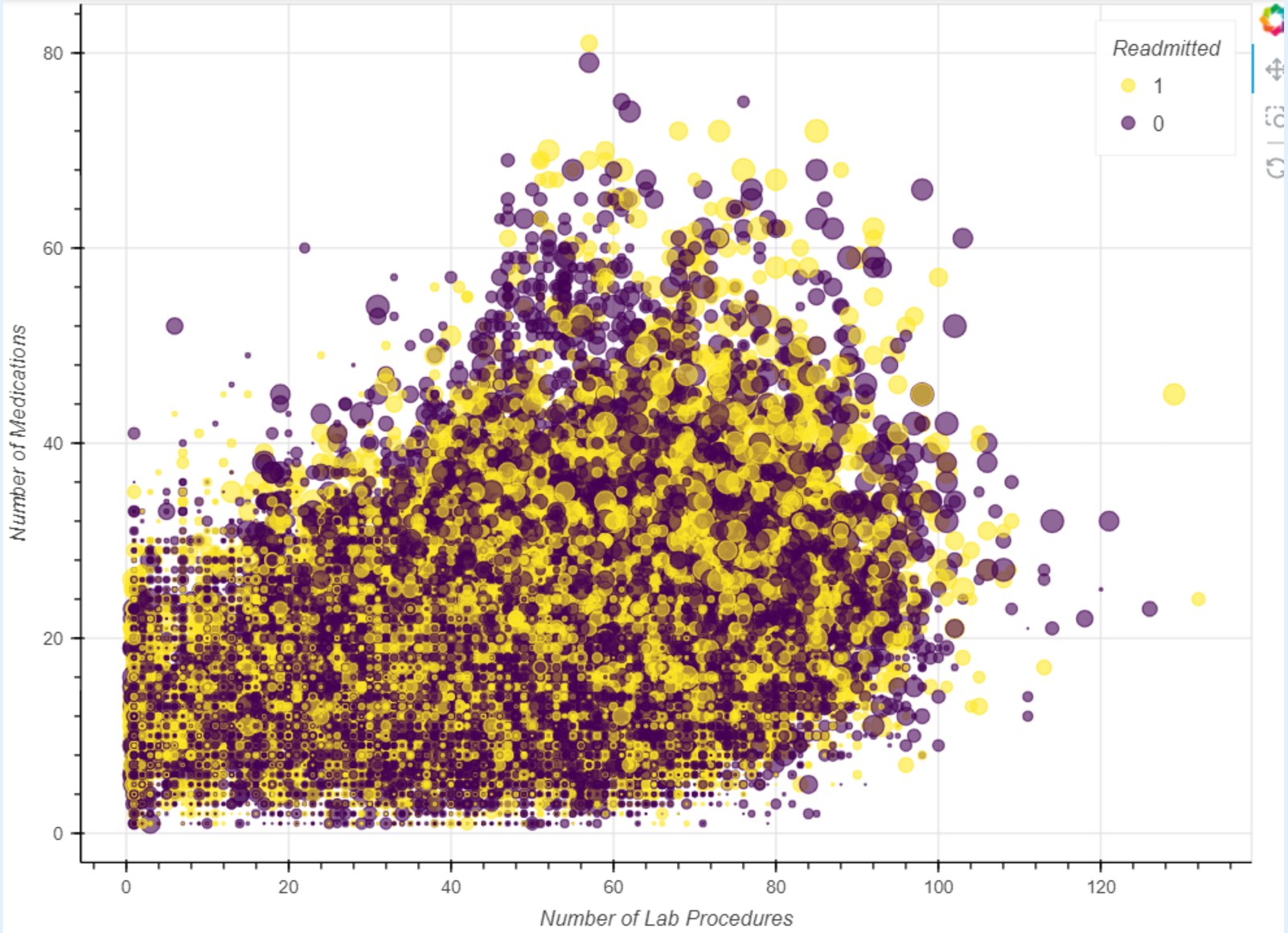
TENDANCES OBSERVED

CONCLUSION ON DISCRET VALUES

- We can see a clear difference in readmission probability depending on the race.
- On the other hand, the change of medicines seems to increase the readmission probability.
- Each admission type has its own readmitted proportion. But from the 4 main represented on the dataset, one is mostly different from the others: Not Available with a majority of readmitted. At the same time, Not Available is the only category where all max_glu_serum results are well represented. The same goes for admission source.
- We can see here how the gender and diag_3 categories, which both did not seem to have an impact on readmission before, are impacted in some categories.

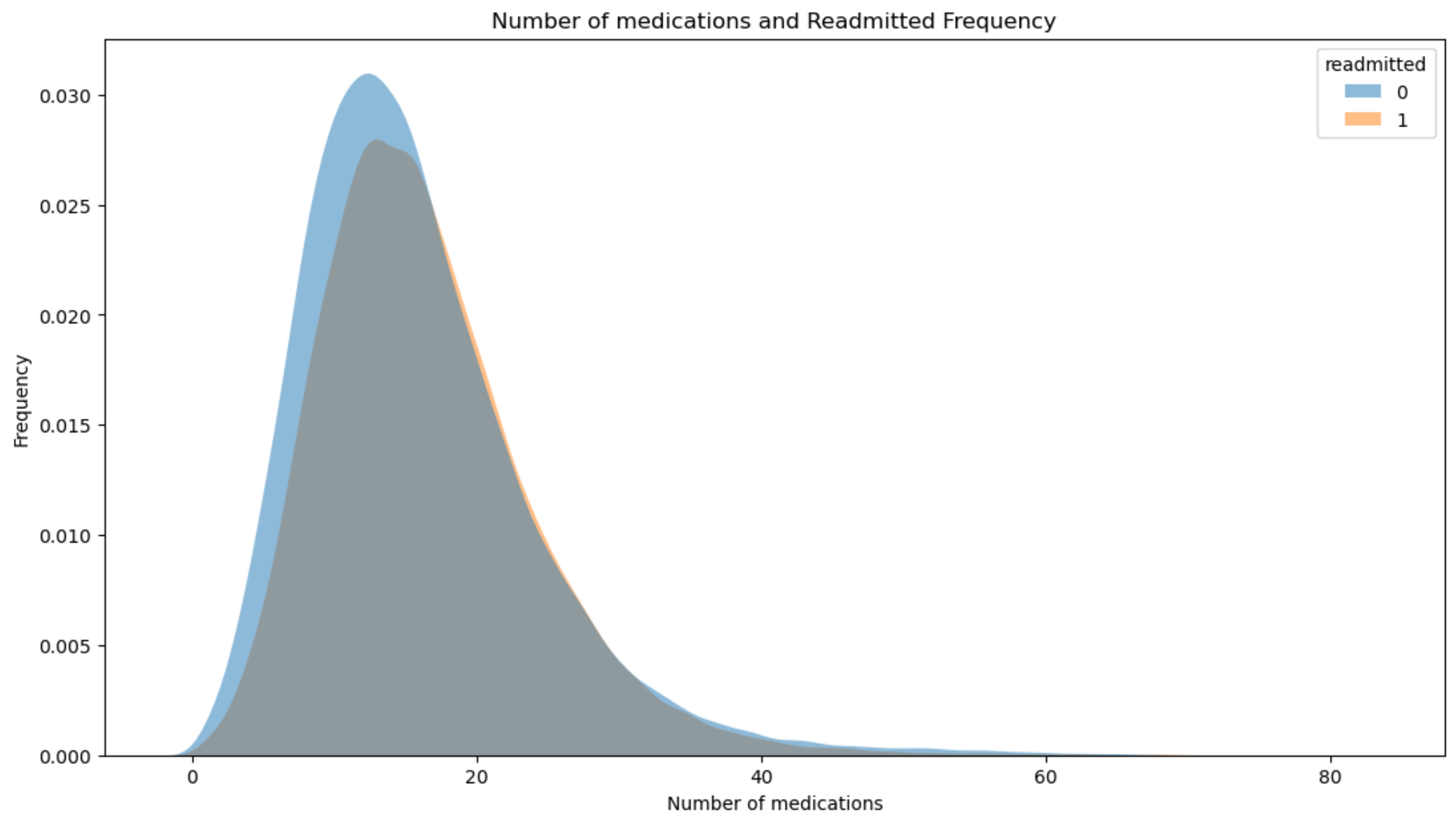
TENDANCES OBSERVED

CONTINUOUS VALUES



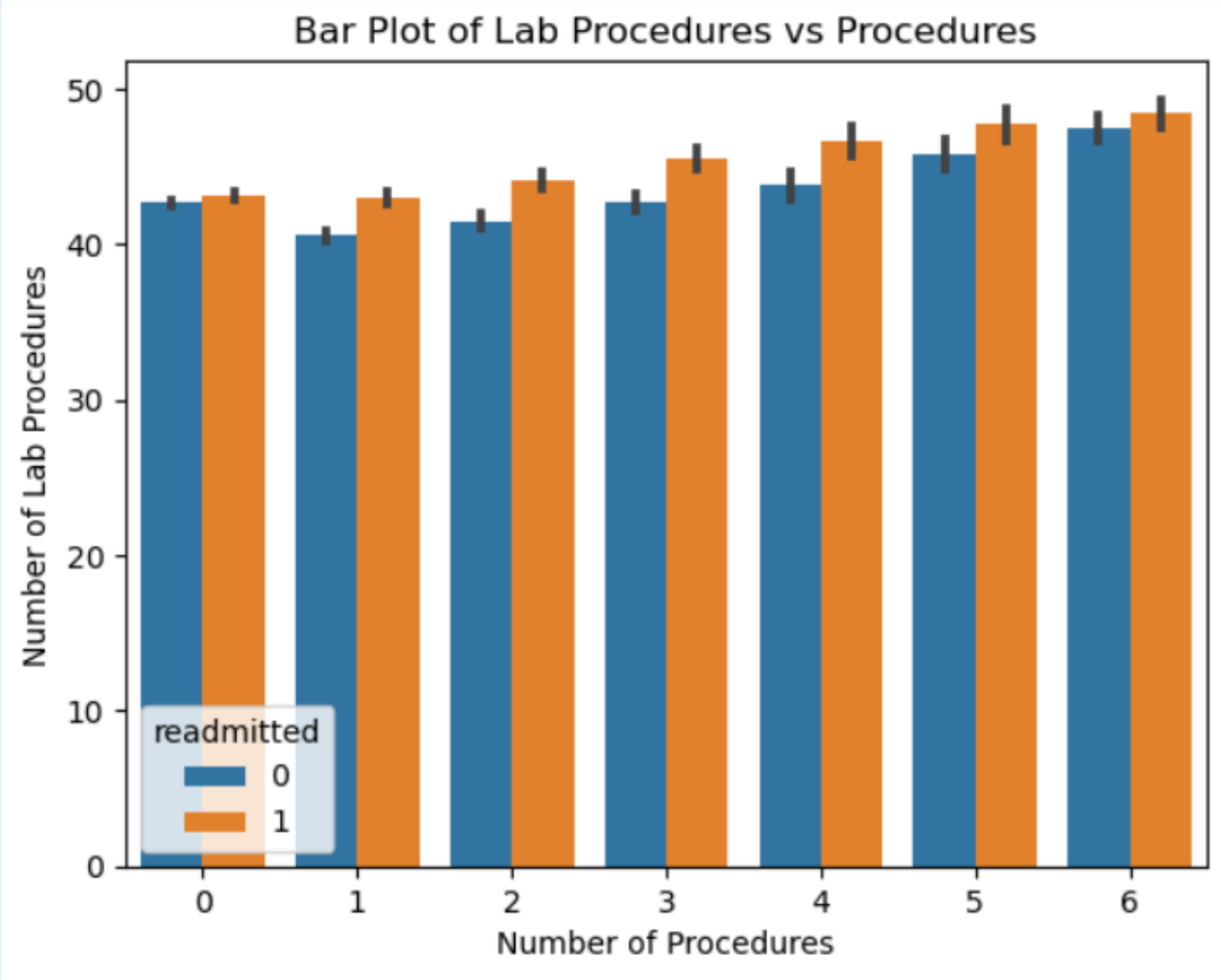
TENDANCES OBSERVED

CONTINUOUS VALUES



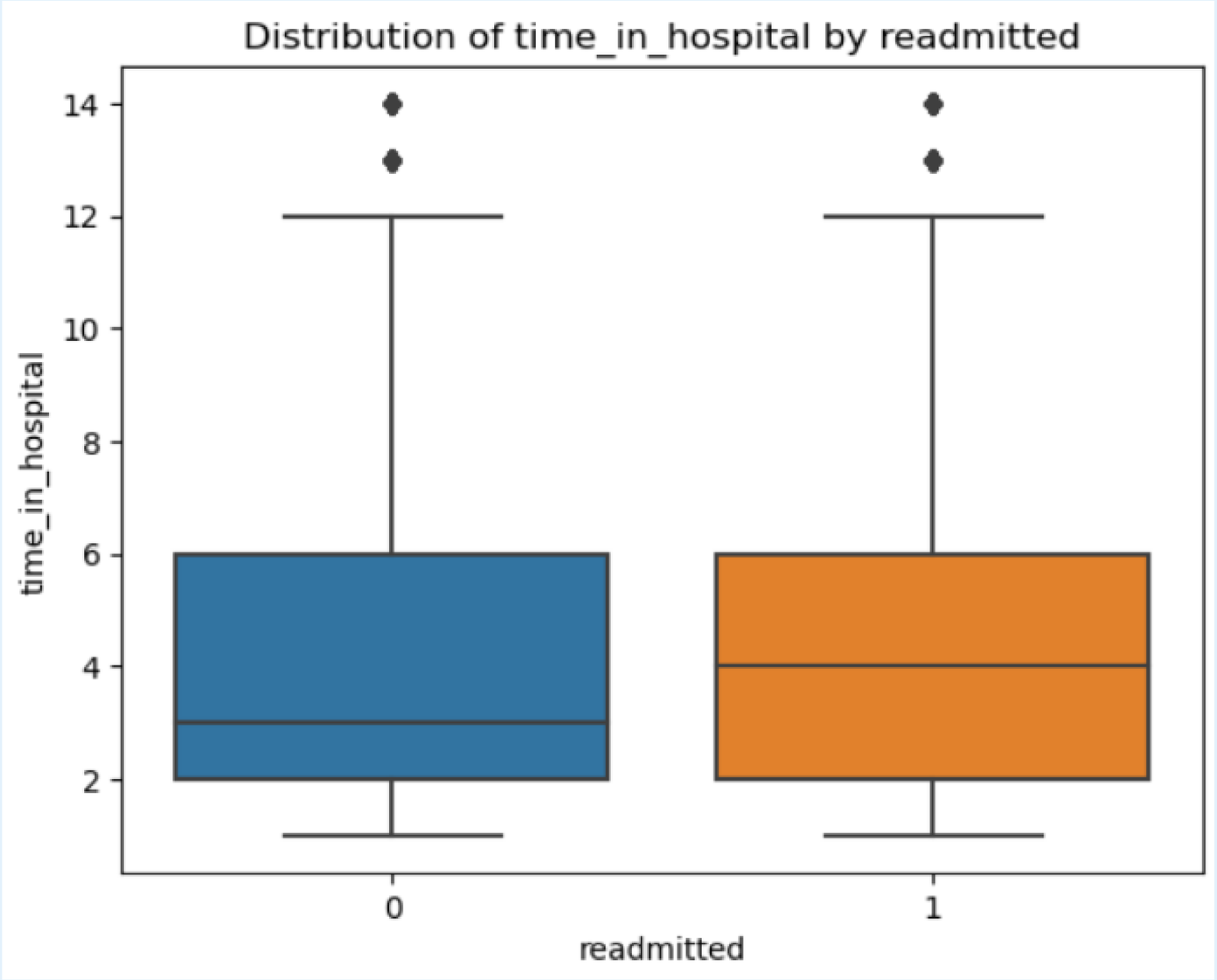
TENDANCES OBSERVED

CONTINUOUS VALUES



TENDANCES OBSERVED

CONTINUOUS VALUES



TENDANCES OBSERVED

CONCLUSION ON CONTINUOUS VALUES

- A certain quantity of procedures (over 80) is associated with at least 10 medicines. For example, if we zoom in between 40 and 70 medicines and between 40 and 80 lab procedures, we will find that in this category, the number of times spent in the hospital seems to have an impact on readmission.

- We can see that the difference between the two distributions happens for fewer than 20 medicines taken.

- The number of lab procedures increases with the number of procedures. The proportion of readmission does not change.

- The time spent in the hospital is less for not readmitted patients.

DATA PREDICTION

- DATA SPLITTING : 20% TEST, RANDOM_STATE 42

- NORMALISATION OF CONTINUOUS VALUES : 'AGE', 'TIME_IN_HOSPITAL', 'NUM_LAB_PROCEDURES','NUM_PROCEDURES', 'NUM_MEDICATIONS', 'NUMBER_OUTPATIENT','NUMBER_EMERGENCY', 'NUMBER_INPATIENT','NUMBER_DIAGNOSES'
- WITH ROBUST SCALAR (QUARTILS)

- MODELS' IMPORTATED : RANDOM FOREST,GRADIENTBOOSTINGCLASSIFIER, LOGISTICREGRESSION

- STATISTICS TESTS : MEAN_SQUARED_ERROR(Y_TEST, Y_PRED), CONFUSION_MATRIX(Y_TEST,Y_PRED), ACCURACY_SCORE(Y_TEST, Y_PRED)

WHY THOSE MODELS

WE ARE CHOOSING CATEGORICAL MODEL: ***LOGISTIC REGRESSION, GRADIENT BOOSTING CLASSIFIER AND RANDOM FOREST***. THE LAST TWO ARE SUPPOSED TO BE MORE EFFICIENT AS THEY CAN DETECT NON LINEAR CORRELATION BETWEEN VARIABLES.

IN OUR DATA SET THERE IS NOT A LOT OF VARIABLE LINEAR CORRELATED TO OUR TARGET VARIABLE, CONSEQUENTLY GRADIENT BOOSTING CLASSIFIER AND RANDOM FOREST SHOULD PROVIDE US BETTER RESULTS.

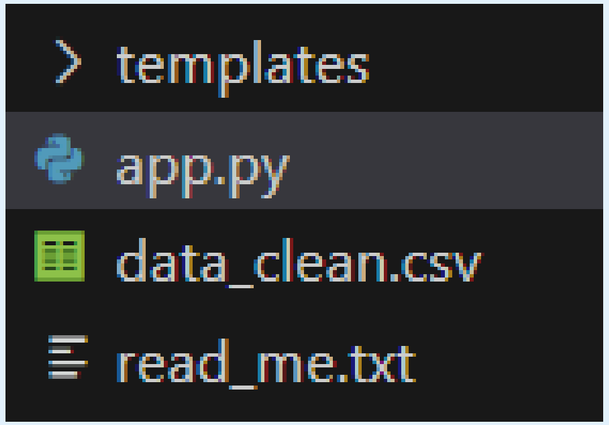
DATA PREDICTION RESULTS

```
--- GradientBoostingClassifier ---
Best Model Mean Squared Error: 0.226305199618765 22

Accuracy:  0.7736948003812347

Matrice de confusion:
 [[8163 1861]
 [2413 6449]]
              precision    recall  f1-score   support

           0       0.77      0.81      0.79     10024
           1       0.78      0.73      0.75      8862

    accuracy                           0.77     18886
   macro avg       0.77      0.77      0.77     18886
weighted avg       0.77      0.77      0.77     18886
```

```
--- Random Forest ---
Best Model Mean Squared Error: 0.22148681563062586

Accuracy:  0.7785131843693741

Matrice de confusion:
 [[8850 1174]
 [3009 5853]]
              precision    recall  f1-score   support

           0       0.75      0.88      0.81     10024
           1       0.83      0.66      0.74      8862

    accuracy                           0.78     18886
   macro avg       0.79      0.77      0.77     18886
weighted avg       0.79      0.78      0.78     18886
```

```
--- LogisticRegression ---
Best Model Mean Squared Error: 0.5115429418616965

Accuracy:  0.4884570581383035

Matrice de confusion:
 [[ 472 9552]
 [ 109 8753]]
              precision    recall  f1-score   support

           0       0.81      0.05      0.09     10024
           1       0.48      0.99      0.64      8862

    accuracy                           0.49     18886
   macro avg       0.65      0.52      0.37     18886
weighted avg       0.66      0.49      0.35     18886
```

ENVIRONNEMENT DE FLASK



TEMPLATES CONTAINS HTML CODE TO OUR DIFFRENT WINDOWS

APP.PY CONTAINS OUR BRUT CODE (GRAPHS AND FLASK CODE TU MAKE WINDOWS COMMUNICATE AMONG HTEM)

DATA_CLEAN.CSV IS OUR DATASET ALREADY CLEANED TO NOT CLEAN IT AGAIN

READ ME CONTAINS INSTRUCTIONS

ENVIRONNEMENT DE FLASK

ALL THE GRAPH FUNCTIONS RETURNED IMAGES :

```
# Affichage de la carte de corrélation
image_stream = BytesIO()
plt.savefig(image_stream, format='png',bbox_inches='tight')
plt.close()
image_stream.seek(0)
img_data = base64.b64encode(image_stream.read()).decode('utf-8')
img_html = f'<img src="{img_data}" alt="Graphique" >'
return img_html
```

FLASK CODE (PLOT_HTML IS THE IMAGE) :

```
return render_template('graph1.html', plot_html=plot_html)
```

THAT ARE PRINT IN “GRAPH1.HTML” TEMPLATE LIKE THAT :

```
{{ plot_html|safe }}
```

AMELIORATIONS

OUR PROJECT LACKS BETTER PREDICTION OPTIMIZATIONS.
OUR GRIDSEARCH OPERATIONS WERE VERY TIME-CONSUMING;

IT IS DIFFICULT TO KNOW WHICH HYPERPARAMETERS ARE THE MOST SUITABLE.

AN IN-DEPTH STUDY OF STATISTICS WOULD BE AN ADDITIONAL POINT.

THE RESULTS WOULD ALSO BE MORE CORRECTE IF WE KNEW THE TIME ORDER OF THE ENCOUNTERS

SOURCES

DATASET :
HTTPS://ARCHIVE.ICS.UCI.EDU/DATASET/296/DIABETES+130-US+HOSPITALS+FOR+YEARS+1999-2008

MODULS USED :
- MATPLOTLIB: HTTPS://MATPLOTLIB.ORG/STABLE/INDEX.HTML
- SEABORN: HTTPS://SEABORN.PYDATA.ORG/
- PANDAS: HTTPS://PANDAS.PYDATA.ORG/DOCS/REFERENCE/API/PANDAS.SERIES.HTML
- SCIKIT-LEARN: HTTPS://SCIKIT-LEARN.ORG/STABLE/INDEX.HTML
- FLASK: HTTPS://PIXEES.FR/INFORMATIQUELYCEE/N_SITE/NSI_PREM_FLASK.HTML
- HTML: HTTPS://DEVELOPER.MOZILLA.ORG/FR/DOCS/GLOSSARY/HTML
-STATISTICS : HTTPS://GT2.ARIIS.FR/LES-ALGORITHMES-DEXPLOITATION/LAPPRENTISSAGE-SUPERVISE/