Data Glacier

Your Deep Learning Partner

Healthcare : Persistency of a Drug Final Project

Virtual Internship

Devin Chau, Ethan Dy, Rohan Khatri

Data Glacier

Your Deep Learning Partner

GitHub Repo

https://github.com/ethan05d/DataGlacier-Internship/tree/main/Week%2013

Data Glacier

Your Deep Learning Partner

Healthcare: Persistency of a Drug			
Group Members	Devin Chau	Ethan Dy	Rohan Khatri
Email	chau.devin031602@gmail.com	ethan05dy@gmail.com	rohankhatri0507@gmail.com
Country	United States	United States	United States
Specialization	Data Science	Data Science	Data Science
Internship Batch	LISUM39	LISUM39	LISUM39
Date	30 January 2025	30 January 2025	30 January 2025

Agenda

- Problem Statement
- Data Information
- Data Understanding
- Exploratory Data Analysis (EDA)
- Recommendations

Problem Statement

Context:

One of the challenge for all Pharmaceutical companies is to understand the persistency of drug as per the physician prescription. To solve this problem ABC pharma company approached an analytics company to automate this process of identification. With an objective to gather insights on the factors that are impacting the persistency, build a classification for the given dataset.

Problem Description:

We are building a predictive model that classifies patients into “persistent” or “non-persistent” categories based on factors like their demographics, medical history, physician characteristics, and treatment details. Factors like the patient level such as their age, risk factors, previous test results, or provider type allows for insights into why some patients continue therapy while others drop off. Thus understanding “persistence” levels. By analyzing these data points and finding patterns, the predictive model helps explain patient behavior and supports the creation of targeted interventions to improve adherence.

Data Information

Total number of observations	3424
Total number of files	1
Total number of features	69
Base format of the file	.csv
Size of the data	891 KB

Data Information

Bucket	Variable	Variable Description
Unique Row Id	Patient ID	Unique ID of each patient
Target Variable	Persistency_Flag	Flag indicating if a patient was persistent or not
Demographics	Age	Age of the patient during their therapy
	Race	Race of the patient from the patient table
	Region	Region of the patient from the patient table
	Ethnicity	Ethnicity of the patient from the patient table
	Gender	Gender of the patient from the patient table
	IDN Indicator	Flag indicating patients mapped to IDN

Data Information

Provider Attributes	NTM - Physician Specialty	Specialty of the HCP that prescribed the NTM Rx
	NTM - T-Score	T Score of the patient at the time of the NTM Rx (within 2 years prior from rxdate)
	Change in T Score	Change in Tscore before starting with any therapy and after receiving therapy (Worsened, Remained Same, Improved, Unknown)
	NTM - Risk Segment	Risk Segment of the patient at the time of the NTM Rx (within 2 years days prior from rxdate)
	Change in Risk Segment	Change in Risk Segment before starting with any therapy and after receiving therapy (Worsened, Remained Same, Improved, Unknown)
	NTM - Multiple Risk Factors	Flag indicating if patient falls under multiple risk category (having more than 1 risk) at the time of the NTM Rx (within 365 days prior from rxdate)

Data Information

Clinical Factors	NTM - Dexa Scan Frequency	Number of DEXA scans taken prior to the first NTM Rx date (within 365 days prior from rxdate)
	NTM - Dexa Scan Recency	Flag indicating the presence of Dexa Scan before the NTM Rx (within 2 years prior from rxdate or between their first Rx and Switched Rx; whichever is smaller and applicable)
	Dexa During Therapy	Flag indicating if the patient had a Dexa Scan during their first continuous therapy
	NTM - Fragility Fracture Recency	Flag indicating if the patient had a recent fragility fracture (within 365 days prior from rxdate)
	Fragility Fracture During Therapy	Flag indicating if the patient had fragility fracture during their first continuous therapy
	NTM - Glucocorticoid Recency	Flag indicating usage of Glucocorticoids (>=7.5mg strength) in the one year look-back from the first NTM Rx
	Glucocorticoid Usage During Therapy	Flag indicating if the patient had a Glucocorticoid usage during the first continuous therapy
	NTM - Injectable Experience	Flag indicating any injectable drug usage in the recent 12 months before the NTM OP Rx
	NTM - Risk Factors	Risk Factors that the patient is falling into. For chronic Risk Factors complete lookback to be applied and for non-chronic Risk Factors, one year lookback from the date of first OP Rx

Data Information

Disease/Treatment Factor	NTM - Comorbidity	Comorbidities are divided into two main categories - Acute and chronic, based on the ICD codes. For chronic disease we are taking complete look back from the first Rx date of NTM therapy and for acute diseases, time period before the NTM OP Rx with one year lookback has been applied
	NTM - Concomitancy	Concomitant drugs recorded prior to starting with a therapy(within 365 days prior from first rxdate)
	Adherence	Adherence for the therapies

Data Understanding

```
#   Column                                                        Non-Null Count  Dtype
---  ------                                                        --------------  -----
0   Ptid                                                          3424 non-null   object
1   Persistency_Flag                                              3424 non-null   object
2   Gender                                                        3424 non-null   object
3   Race                                                          3424 non-null   object
4   Ethnicity                                                     3424 non-null   object
5   Region                                                        3424 non-null   object
6   Age_Bucket                                                    3424 non-null   object
7   Ntm_Speciality                                                3424 non-null   object
8   Ntm_Specialist_Flag                                           3424 non-null   object
9   Ntm_Speciality_Bucket                                         3424 non-null   object
10  Gluco_Record_Prior_Ntm                                        3424 non-null   object
11  Gluco_Record_During_Rx                                        3424 non-null   object
12  Dexa_Freq_During_Rx                                           3424 non-null   int64
13  Dexa_During_Rx                                                3424 non-null   object
14  Frag_Frac_Prior_Ntm                                           3424 non-null   object
15  Frag_Frac_During_Rx                                           3424 non-null   object
16  Risk_Segment_Prior_Ntm                                        3424 non-null   object
17  Tscore_Bucket_Prior_Ntm                                       3424 non-null   object
18  Risk_Segment_During_Rx                                        3424 non-null   object
19  Tscore_Bucket_During_Rx                                       3424 non-null   object
20  Change_T_Score                                                3424 non-null   object
21  Change_Risk_Segment                                           3424 non-null   object
22  Adherent_Flag                                                 3424 non-null   object
23  Idn_Indicator                                                 3424 non-null   object
24  Injectable_Experience_During_Rx                               3424 non-null   object
25  Comorb_Encounter_For_Screening_For_Malignant_Neoplasms        3424 non-null   object
26  Comorb_Encounter_For_Immunization                             3424 non-null   object
27  Comorb_Encntr_For_General_Exam_W_O_Complaint,_Susp_Or_Reprtd_Dx  3424 non-null   object
28  Comorb_Vitamin_D_Deficiency                                   3424 non-null   object
29  Comorb_Other_Joint_Disorder_Not_Elsewhere_Classified          3424 non-null   object
30  Comorb_Encntr_For_Oth_Sp_Exam_W_O_Complaint_Suspected_Or_Reprtd_Dx  3424 non-null   object
31  Comorb_Long_Term_Current_Drug_Therapy                         3424 non-null   object
32  Comorb_Dorsalgia                                              3424 non-null   object
33  Comorb_Personal_History_Of_Other_Diseases_And_Conditions      3424 non-null   object
34  Comorb_Other_Disorders_Of_Bone_Density_And_Structure          3424 non-null   object
35  Comorb_Disorders_of_lipoprotein_metabolism_and_other_lipidemias  3424 non-null   object
36  Comorb_Osteoporosis_without_current_pathological_fracture     3424 non-null   object
37  Comorb_Personal_history_of_malignant_neoplasm                 3424 non-null   object
38  Comorb_Gastro_esophageal_reflux_disease                       3424 non-null   object
```

- The dataset consists of 3424 rows and 69 columns
- Types of Variables
  - **Numeric** (*2* columns):
    - Dexa_Freq_During_Rx
    - Count_Of_Risks
  - **Categorical** (*67* columns):
    - Examples: Persistency_Flag, Gender, Ntm_Speciality, etc
    - Many are binary flags (Y/N, etc.), while some have multiple categories (e.g., Ntm_Speciality has 36)
- Multiple columns (such as Risk_Segment_During_Rx, Change_T_Score, etc.) contain a large number of “Unknown” entries. This shows hidden missing data that could influence model training and interpretation

Data Understanding

```
Missing Values Summary:
                                                                                      Colu
Ptid                                                                                    Pt
Concom_Cephalosporins                                                  Concom_Cephalospori
Risk_Osteogenesis_Imperfecta                                        Risk_Osteogenesis_Imperfec
Risk_Type_1_Insulin_Dependent_Diabetes                        Risk_Type_1_Insulin_Dependent_Diabet
Concom_Viral_Vaccines                                                  Concom_Viral_Vaccin
...                                                                                      .
Comorb_Other_Joint_Disorder_Not_Elsewhere_Class...  Comorb_Other_Joint_Disorder_Not_Elsewhere_Clas.
Comorb_Encntr_For_Oth_Sp_Exam_W_O_Complaint_Sus...  Comorb_Encntr_For_Oth_Sp_Exam_W_O_Complaint_Su.
Comorb_Long_Term_Current_Drug_Therapy                      Comorb_Long_Term_Current_Drug_Thera
Comorb_Dorsalgia                                                          Comorb_Dorsalg
Count_Of_Risks                                                            Count_Of_Ris

                                                    Missing Values  \
Ptid                                                             0
Concom_Cephalosporins                                            0
Risk_Osteogenesis_Imperfecta                                     0
Risk_Type_1_Insulin_Dependent_Diabetes                           0
Concom_Viral_Vaccines                                            0
...                                                            ...
Comorb_Other_Joint_Disorder_Not_Elsewhere_Class...               0
Comorb_Encntr_For_Oth_Sp_Exam_W_O_Complaint_Sus...               0
Comorb_Long_Term_Current_Drug_Therapy                            0
Comorb_Dorsalgia                                                 0
Count_Of_Risks                                                   0

                                                    Percentage Missing
Ptid                                                               0.0
Concom_Cephalosporins                                              0.0
Risk_Osteogenesis_Imperfecta                                       0.0
Risk_Type_1_Insulin_Dependent_Diabetes                             0.0
Concom_Viral_Vaccines                                              0.0
...                                                                ...
Comorb_Other_Joint_Disorder_Not_Elsewhere_Class...                 0.0
Comorb_Encntr_For_Oth_Sp_Exam_W_O_Complaint_Sus...                 0.0
Comorb_Long_Term_Current_Drug_Therapy                              0.0
Comorb_Dorsalgia                                                   0.0
Count_Of_Risks                                                     0.0

[69 rows x 3 columns]
```

- Hidden “Unknown” Data
  - Some columns use the string "Unknown" instead of NaN. Like, Risk_Segment_During_Rx, Change_T_Score, and others have a lot of "Unknown" entries
- Outliers:
  - Two numeric variables, Dexa_Freq_During_Rx and Count_Of_Risks, have outliers:
    - Dexa_Freq_During_Rx shows ***460*** outliers (based on the Interquartile Range method)
    - Count_Of_Risks shows ***8*** outliers (also IQR-based)

Data Understanding

```
[ ]  scaler = MinMaxScaler()
     encoded_data[numerical_cols] = scaler.fit_transform(encoded_data[numerical_cols])
```

- MinMaxScaler was used to normalize numerical columns, bringing all variables to a common scale and ensuring equal contribution to the model.
- **Feature Scaling:**
    - Scaled numerical features using MinMaxScaler to normalize their range, ensuring model fairness.

```
duplicates = df.duplicated(subset='PatientID').sum()
```

- Duplicate rows were detected based on the 'PatientID' column to prevent skewed analyses, and they were logged for further review.
- Columns were standardized to appropriate data types, reducing potential errors during analysis.

Data Understanding

```
df = df[df['Dexa_Freq_During_Rx'] != 0]
```

- **Outlier Handling:**
  - Removed outliers from the Dexa_Freq_During_Rx column by excluding entries where its value was zero.

```
binary_columns = [col for col in df.columns if set(df[col].dropna().unique()) == {'N', 'Y'}]

for col in binary_columns:
    df[col] = df[col].replace({'N': 0, 'Y': 1}).astype(int)
```

- **Binary Encoding:**
  - Converted categorical binary values ('N' and 'Y') into numerical representations (0 and 1), ensuring compatibility with classifier models.

Data Understanding

```python
# Using IQR
Q1 = df['Dexa_Freq_During_Rx'].quantile(0.25)
Q3 = df['Dexa_Freq_During_Rx'].quantile(0.75)
IQR = Q3 - Q1
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR

df['Dexa_Freq_During_Rx_No_Outliers'] = np.where(
    df['Dexa_Freq_During_Rx'] > upper_bound, upper_bound,
    np.where(df['Dexa_Freq_During_Rx'] < lower_bound, lower_bound, df['Dexa_Freq_During_Rx'])
)
df['Dexa_Freq_During_Rx_No_Outliers'].describe()
```

- **Outlier Handling:**
  - Applied the IQR method to cap extreme values within acceptable ranges, addressing potential data skew.

Data Understanding

```python
risk_bins = [-1, 0, 2, 5, float('inf')]
risk_labels = ['None', 'Low', 'Moderate', 'High']
encoded_data['Risk_Level'] = pd.cut(encoded_data['Count_Of_Risks'].astype(int),
                                    bins=risk_bins,
                                    labels=risk_labels)

```

- **Risk Level Encoding:**
  - Categorized patient risk levels into bins such as 'None', 'Low', 'Moderate', and 'High' based on Count_Of_Risks values, enabling targeted risk analysis.

Exploratory Data Analysis (EDA)

During the exploratory data analysis (EDA), we performed the following steps:

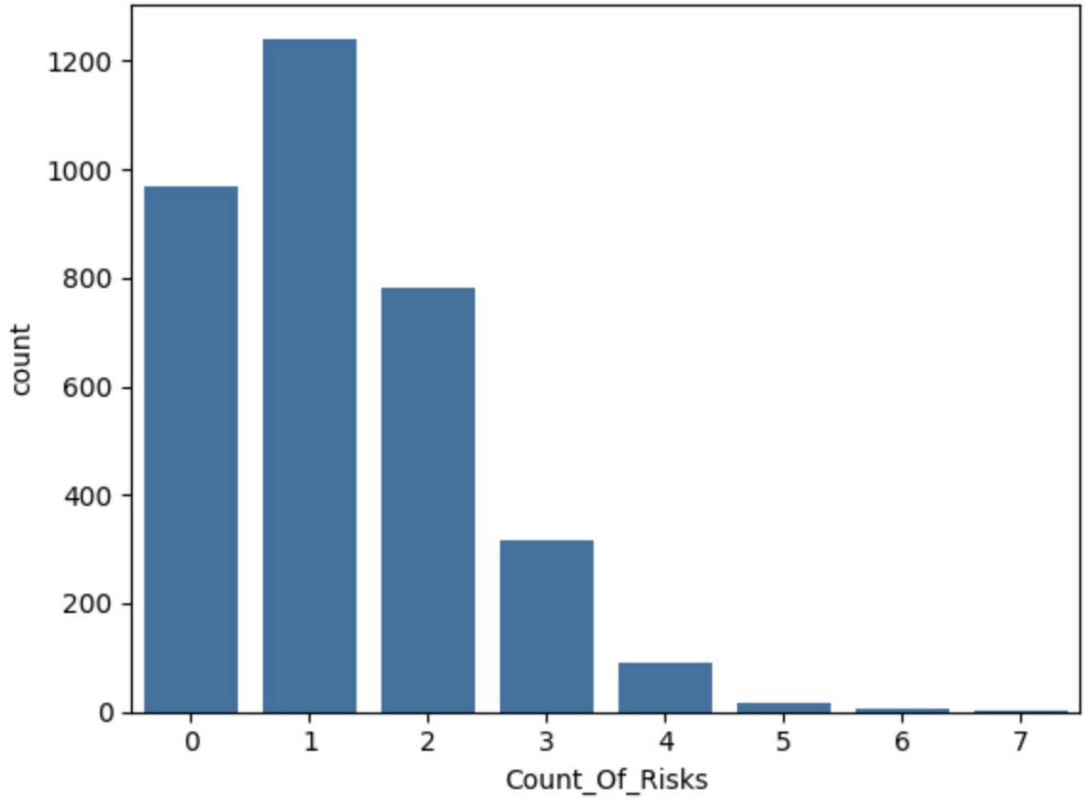
- **Data Visualization**: Visualized distributions of numerical features such as Dexa_Freq_During_Rx and Count_Of_Risks.
    - Histograms showed high skewness in Dexa_Freq_During_Rx, leading to the decision to apply log transformation for normalization

- **Correlation Analysis**: Calculated correlations between numerical variables. Count_Of_Risks showed a moderate correlation with the target variable Persistency_Flag

- **Categorical Feature Inspection**: Explored distributions for categorical variables like Ntm_Speciality and Gender. Identified the imbalanced classes in Ntm_Speciality which could maybe lead to overfitting (?)
- **Outlier Detection**: Used the Interquartile Range (IQR) method to detect outliers. Dexa_Freq_During_Rx had 460 extreme values,
    - while Count_Of_Risks had 8 outliers (addressed using capping methods)

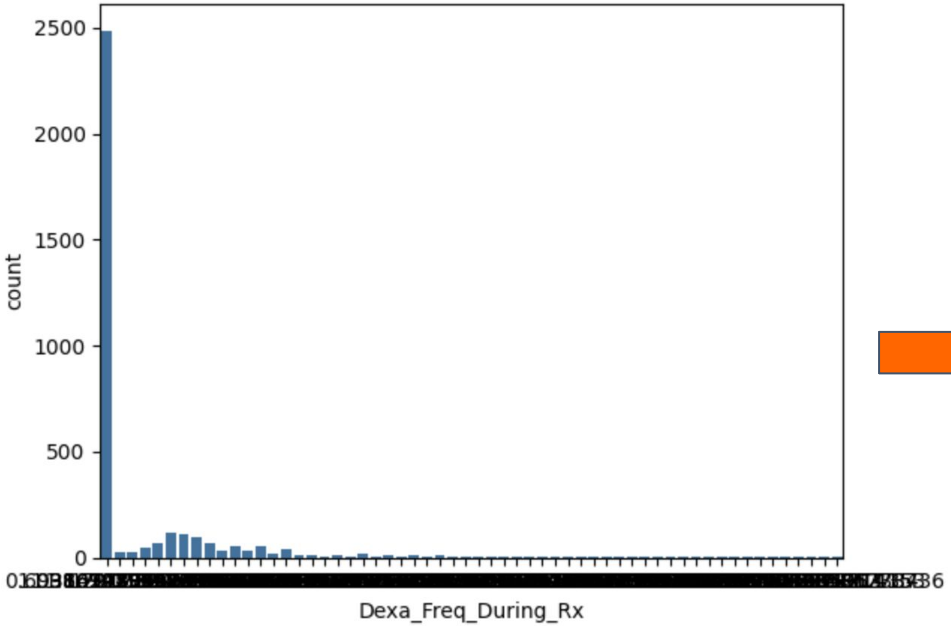
- **Handling Missing Values**: Discovered multiple columns with the value "Unknown" instead of NaNs, indicating hidden missing data. Handled as separate categories

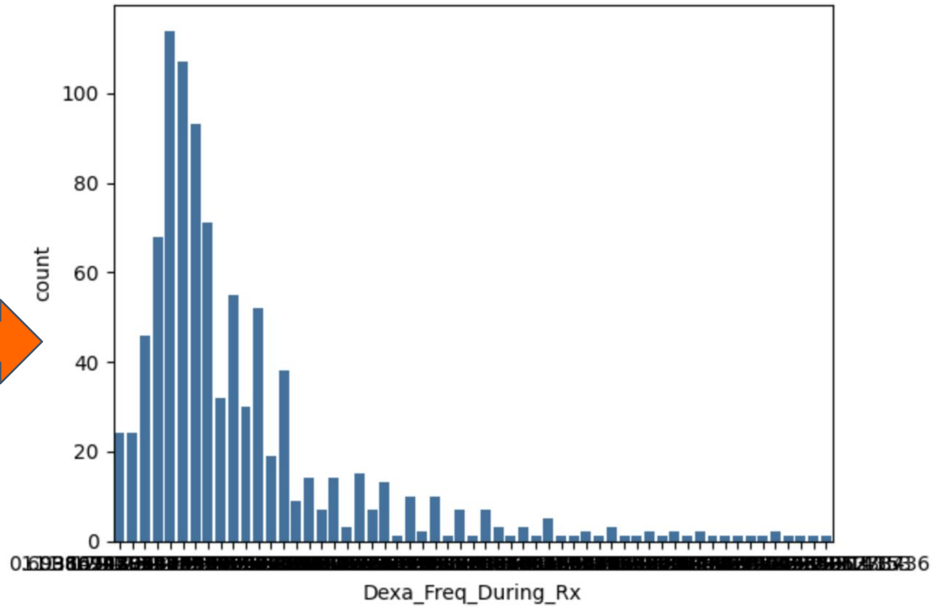
Exploratory Data Analysis (EDA)



- As we can see, most people tend to have 0-2 counts of risk in total, which is good to see. It is better to see less counts of risks in comparison to the higher numbers.

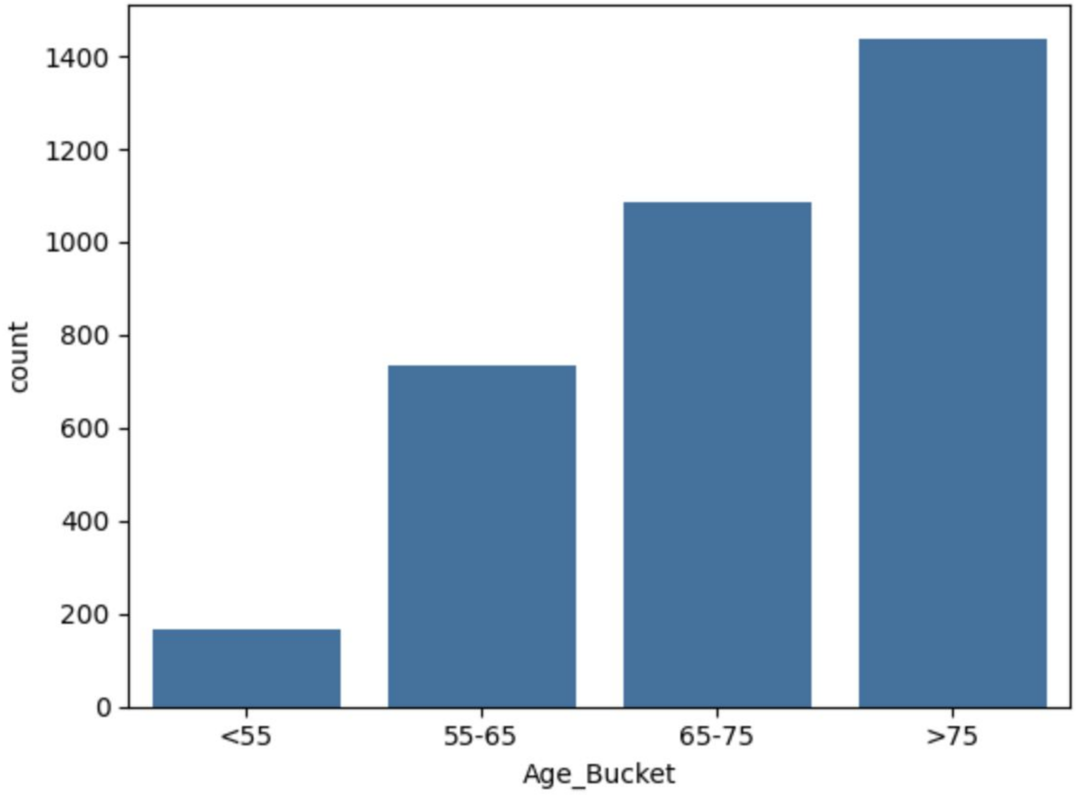
Exploratory Data Analysis (EDA)





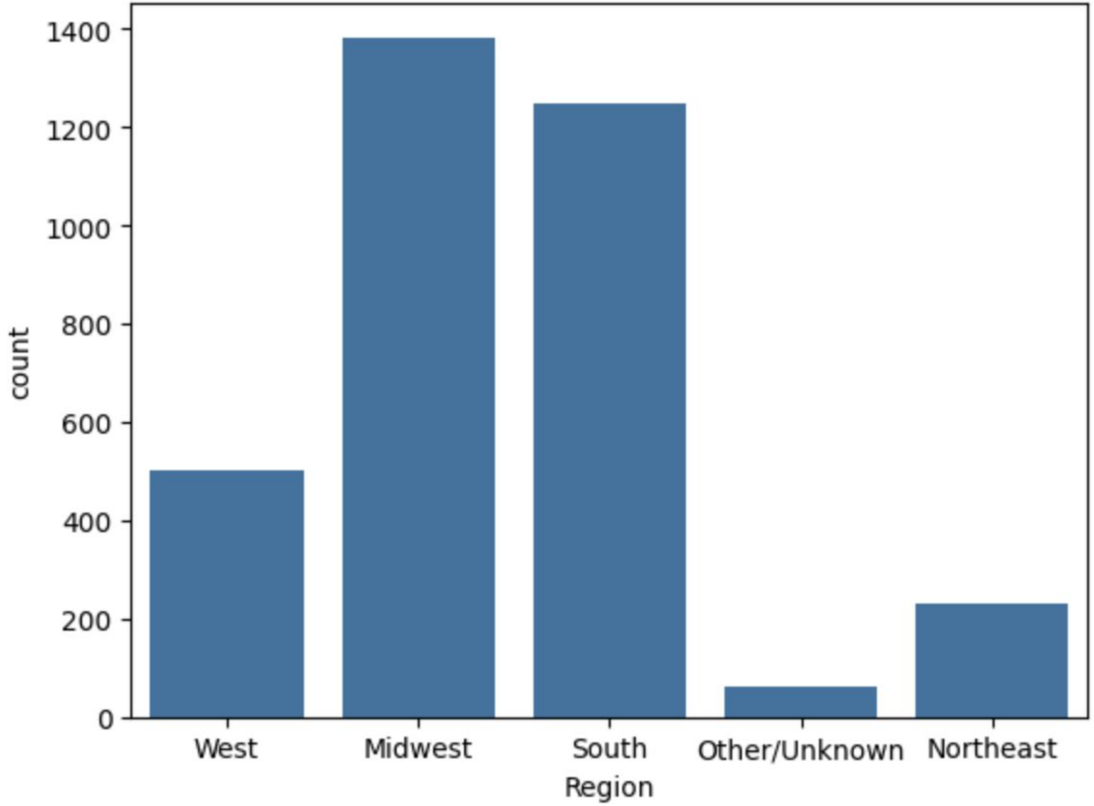
- This is what the distribution looks like without the outlier of 0

Exploratory Data Analysis (EDA)



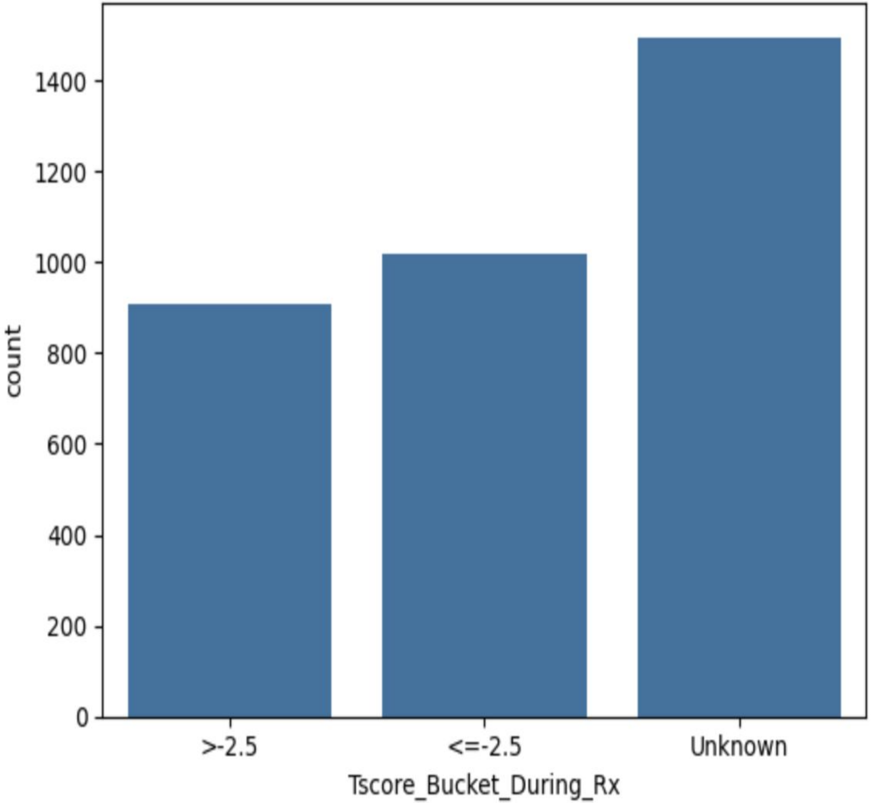
- The largest proportion of patients reported in this dataset belongs to the older age group, specifically those aged >75.

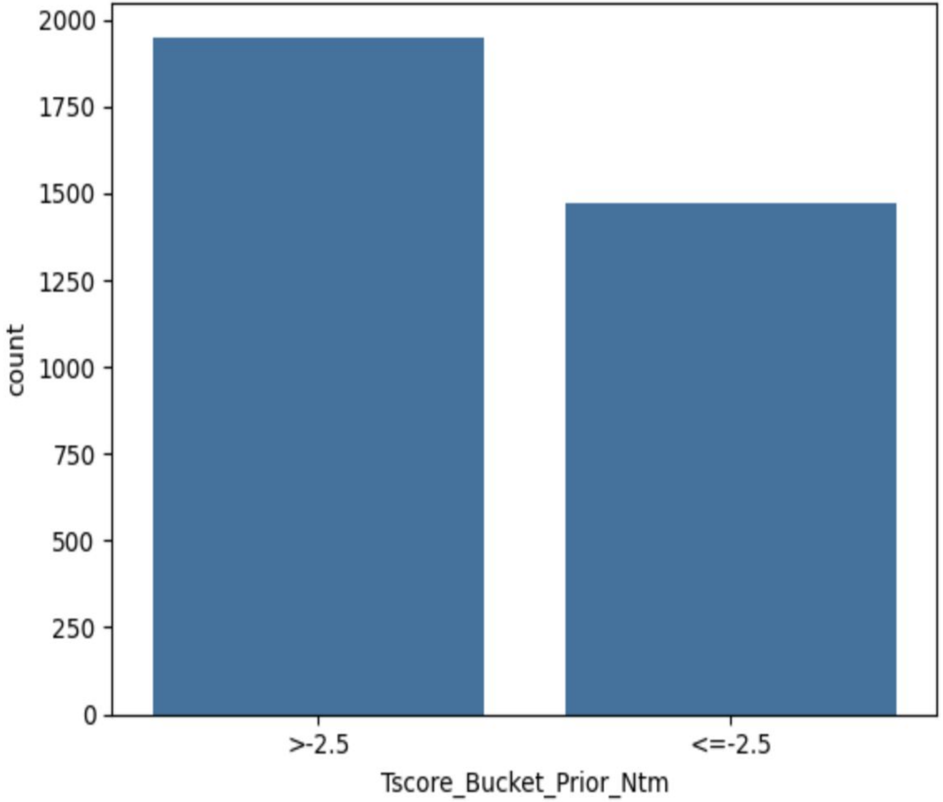
Exploratory Data Analysis (EDA)



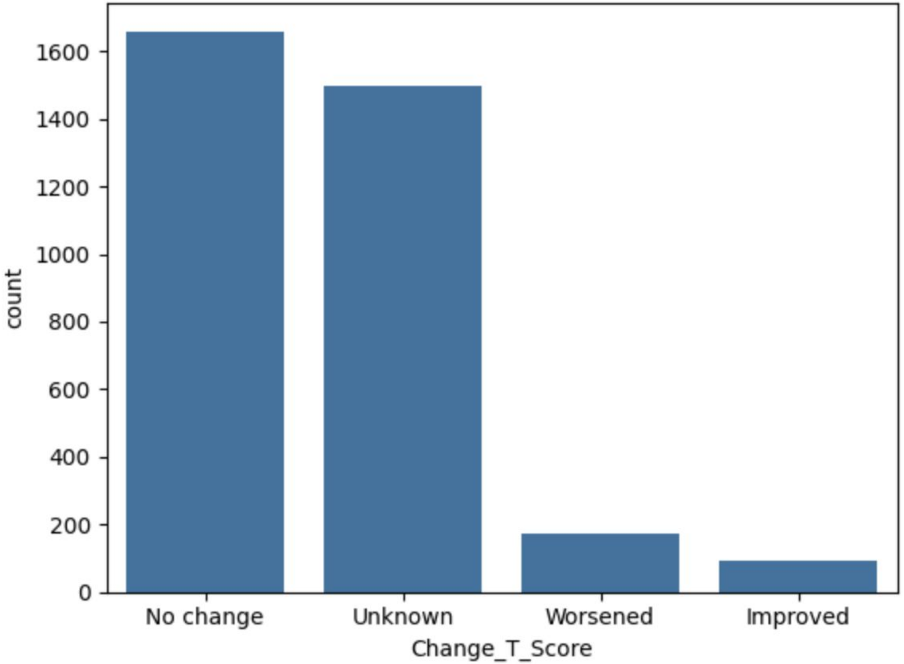
- Northeast and West seem to be severely underreported as compared to Midwest and South Region

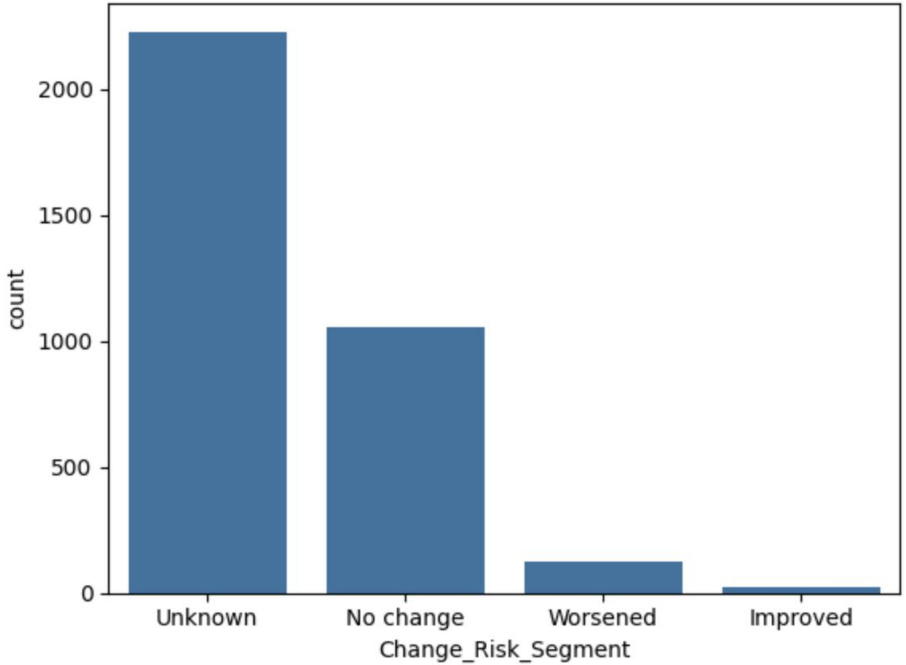
Exploratory Data Analysis (EDA)





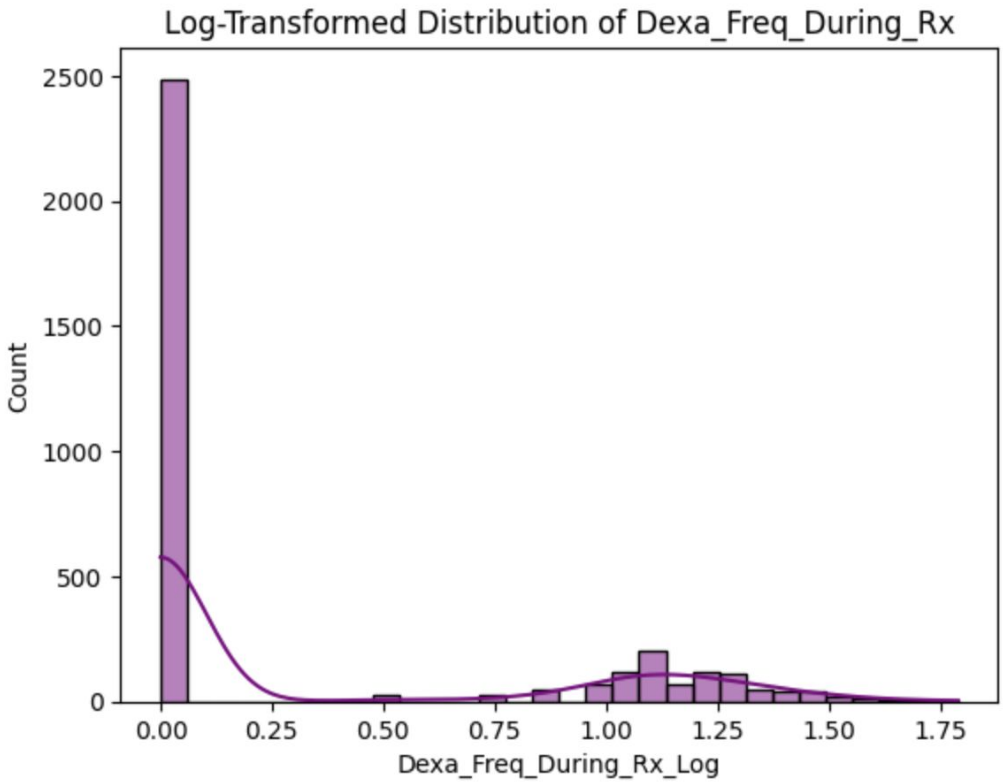
Exploratory Data Analysis (EDA)





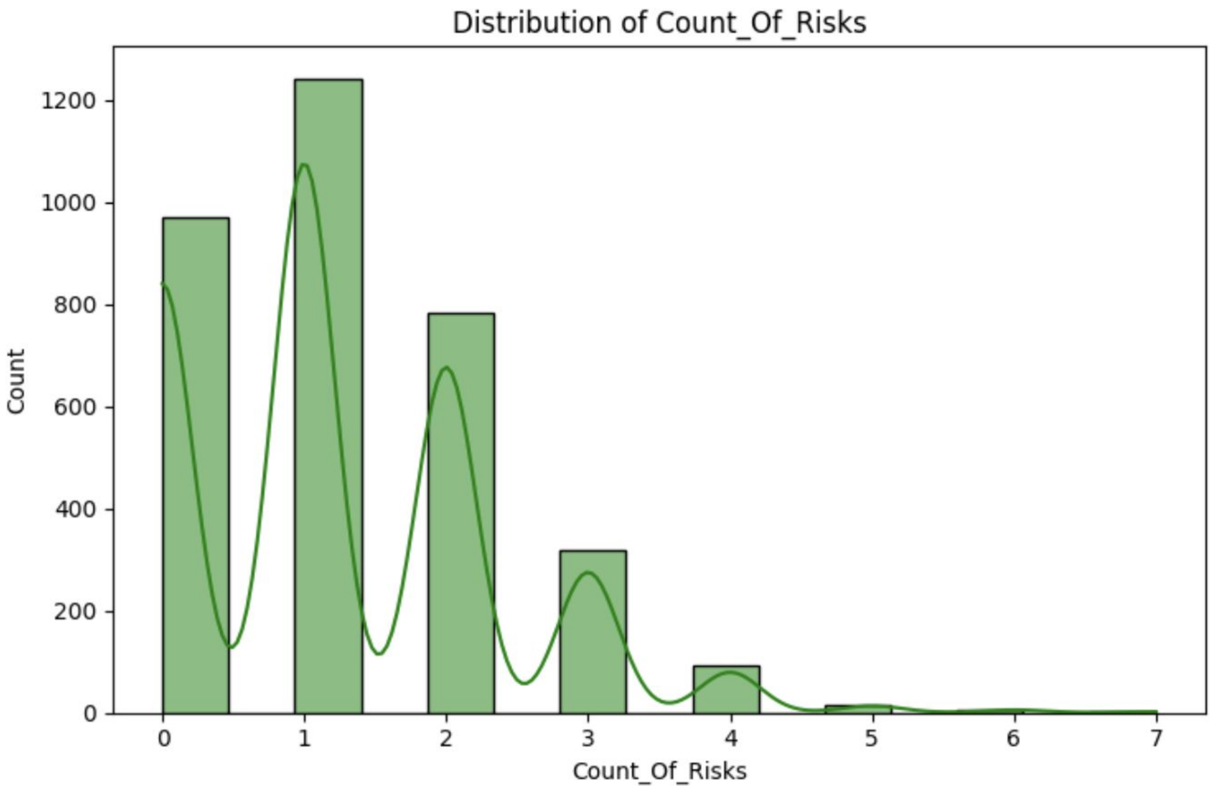
- We can see that typically there is no change. Worsened more than improved in terms of change risk segment.

Exploratory Data Analysis (EDA)



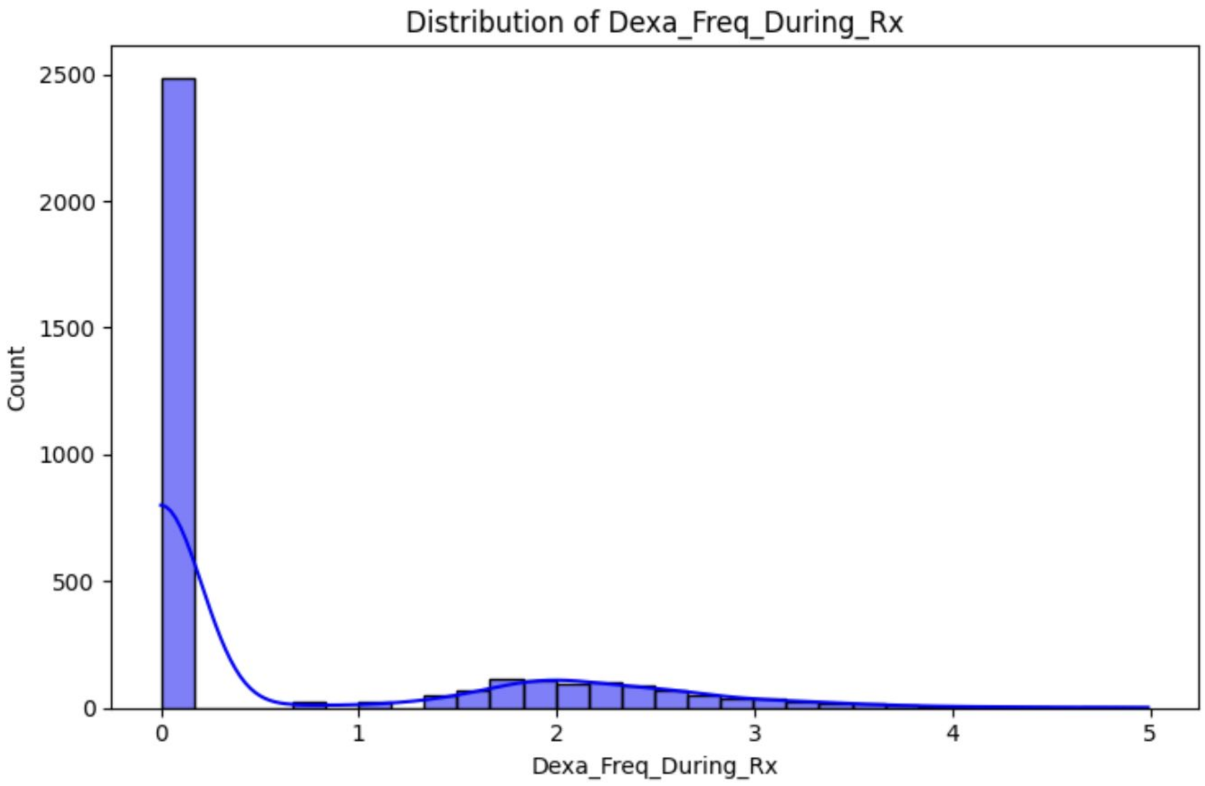
- Log transformation to Dexa_Freq_During_Rx for normalization

Exploratory Data Analysis (EDA)



- Histogram for Count_Of_Risks

Exploratory Data Analysis (EDA)

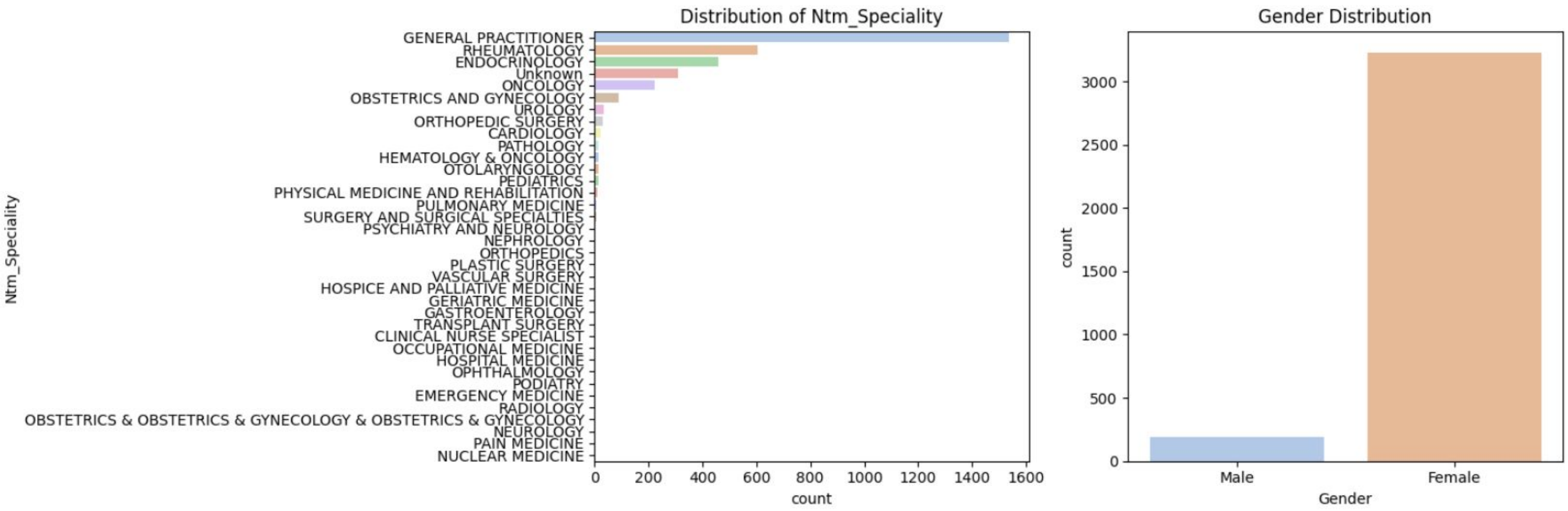


- Histogram for normalized Dexa_Freq_During_Rx

Exploratory Data Analysis (EDA)

- The variables are mostly uncorrelated except for a moderate relationship between Dexa_Freq_During_Rx and Persistency_Flag_Numeric.
- Higher Dexa scan frequency might be slightly associated with better persistence in treatment adherence.
- Count of risks does not show a meaningful correlation with either of the other variables.

Exploratory Data Analysis (EDA)



- The data suggests that most observations come from general practitioners and predominantly female participants.
- The gender imbalance might influence study outcomes if gender plays a role in the analysis being conducted.

Exploratory Data Analysis (EDA)

Final Recommendations

Based on the analysis and identified data issues, we recommend:

- **Handling Missing Values**: Maintain “Unknown” entries as a separate category to preserve data integrity and avoid loss of potentially valuable patterns
- **Normalization**: Continue using MinMaxScaler for numerical features like Dexa_Freq_During_Rx and Count_Of_Risks to ensure consistent scaling across variables
- **Outlier Handling**: Use the IQR method to cap extreme values for Dexa_Freq_During_Rx and Count_Of_Risks
- **Categorical Encoding**: Apply one-hot encoding for categorical variables with multiple categories. Using binary encoding for columns with simple Y/N values
- **Feature Engineering**: Consider grouping rare categories in columns like Ntm_Speciality to avoid overfitting due to high cardinality
- **Data Consistency**: Ensure proper standardization of data types and consistency across the entire dataset before model training

Exploratory Data Analysis (EDA)

Recommended models for this datasets:

- **Logistic Regression**:
  a. For binary classification.
- **Decision Trees**:
  a. Easy to interpret and handle categorical variables directly.
- **Random Forest**:
  - Robust and simple, handling mixed data types well.
    i. We will probably look into random forest more so than decision trees

Recommendations

Our recommendations for this is to have a model be built using XGBoost to classify patients into "persistent" or "non-persistent" categories

XGBoost (Extreme Gradient Boosting) is a great choice for this problem as it includes L1 and L2 regularization to avoid overfitting which could occur from this type of problem. XGBoost also uses max depth approach which helps with overfitting

Recommendations (cont).

XGBoost worked well during training and testing

- However, it could be improved upon as our accuracy was below 80%. We wish to optimize the model to give accurate predictions

Future Endeavors

- In the future, we look to add possibly more models, specifically models using ensemble learning techniques, and combine it with XGBoost to improve efficiency and accuracy.
    - Some possible techniques for the future would be:
        - Blending models, bagging, or have an autoencoder create new features

Demonstration of the Code

XGBoost Model

```
target_column = 'Persistency_Flag'
df[target_column] = LabelEncoder().fit_transform(df[target_column])

categorical_columns = df.select_dtypes(include=['object']).columns
df_encoded = df.copy()
for col in categorical_columns:
    df_encoded[col] = LabelEncoder().fit_transform(df_encoded[col])

# Split features and target
X = df_encoded.drop(columns=[target_column, 'Ptid'])
y = df_encoded[target_column]

# Train-test split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

# Re-run XGBoost
xgb_model = XGBClassifier(use_label_encoder=False, eval_metric='logloss', random_state=42)
xgb_model.fit(X_train, y_train)

# Make predictions and evaluate
y_pred_xgb = xgb_model.predict(X_test)
xgb_accuracy = accuracy_score(y_test, y_pred_xgb)

xgb_accuracy, classification_report(y_test, y_pred_xgb)
classification_results = classification_report(y_test, y_pred_xgb)

print(classification_results)
print("XGBoost accuracy:", xgb_accuracy)
```

```
              precision    recall  f1-score   support

           0       0.83      0.85      0.84       654
           1       0.73      0.70      0.72       374

    accuracy                           0.80      1028
   macro avg       0.78      0.78      0.78      1028
weighted avg       0.80      0.80      0.80      1028

XGBoost accuracy: 0.7976653696498055
```

Thank You

Data Glacier
Your Deep Learning Partner