Data Science HealthCare Project

Drug Persistance

Data Glacier

Your Deep Learning Partner

By : Sourour Cherif

Mail : Sourour.cherif@esprit .tn

Data Science Engineer Student

Esprit Tunisia

Problem description

ABC Pharma is looking for an automated way better than the traditional debilitating methods currently used to assess persistence of drugs as per the physician prescription, in order to have a deeper understanding on the factors impacting the persistence of their drug. The aim is to know if a patient, based on his/her information, will follow the prescription of the physician and continue taking the drug for all the treatment time. We have been provided with a dataset which contains patients' details.

Business understanding

We will create a classification model as a solution that divides patients into categories depending on their information, to determine if a patient was persistent or not.

Our goal is to create a web application that might be used as an automated solution to this process of identification.

Project Lifecycle Along with Deadline

The entire project, including all requirements, must be submitted by **the 30th of August 2022** . The project has been split into several subtasks .

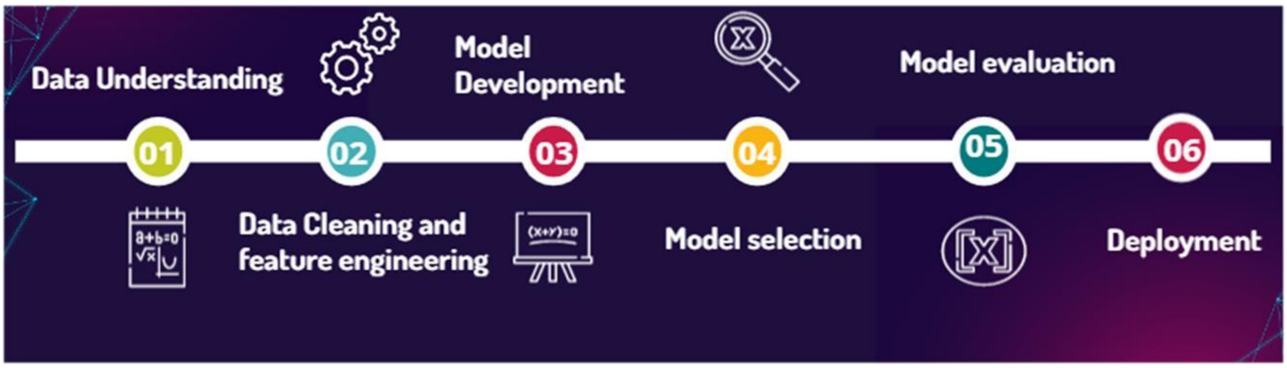


Figure 1 : Project Lifecycle

Data Intake Report

Name: Persistency of a Drug

Report Date: 09/08/2022

Internship Batch: LISUM10: 30

Data Intake: Sourour Cherif

Data Storage Location: https://github.com/Sururrrr/Drug_Persistance.git

Tabular Data Details: Healthcare_dataset.xlsx	
Total number of Observations	3424
Total number of File(s)	1
Total number of Features (Independent Variables or Predictors)	68
Base format of the File	.xlsx
Size of the dataset	899 KB

Data understanding

To fit any predictive model on a dataset, we need to understand the complexity of the dataset before deciding which predictive model to use to get optimal performance.

```
data.head()
```

	Persistency_Flag	Gender	Race	Ethnicity	Region	Age_Bucket	Ntm_Speciality	Ntm_Specialist_Flag	Ntm_Speciality_Bucket	Gluco_Record_Prior_Ntm	...
0	Persistent	Male	Caucasian	Not Hispanic	West	>75	GENERAL PRACTITIONER	Others	OB/GYN/Others/PCP/Unknown	N	...
1	Non-Persistent	Male	Asian	Not Hispanic	West	55-65	GENERAL PRACTITIONER	Others	OB/GYN/Others/PCP/Unknown	N	...
2	Non-Persistent	Female	Other/Unknown	Hispanic	Midwest	65-75	GENERAL PRACTITIONER	Others	OB/GYN/Others/PCP/Unknown	N	...
3	Non-Persistent	Female	Caucasian	Not Hispanic	Midwest	>75	GENERAL PRACTITIONER	Others	OB/GYN/Others/PCP/Unknown	N	...
4	Non-Persistent	Female	Caucasian	Not Hispanic	Midwest	>75	GENERAL PRACTITIONER	Others	OB/GYN/Others/PCP/Unknown	Y	...

5 rows × 68 columns

Type of data

```
[ ]  data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3424 entries, 0 to 3423
Data columns (total 68 columns):
 #   Column                                                   Non-Null Count  Dtype
---  ------                                                   --------------  -----
 0   Persistency_Flag                                         3424 non-null   object
 1   Gender                                                   3424 non-null   object
 2   Race                                                     3424 non-null   object
 3   Ethnicity                                                3424 non-null   object
 4   Region                                                   3424 non-null   object
 5   Age_Bucket                                               3424 non-null   object
 6   Ntm_Speciality                                           3424 non-null   object
 7   Ntm_Specialist_Flag                                      3424 non-null   object
 8   Ntm_Speciality_Bucket                                    3424 non-null   object
 9   Gluco_Record_Prior_Ntm                                   3424 non-null   object
 10  Gluco_Record_During_Rx                                   3424 non-null   object
 11  Dexa_Freq_During_Rx                                      3424 non-null   int64
 12  Dexa_During_Rx                                           3424 non-null   object
 13  Frag_Frac_Prior_Ntm                                      3424 non-null   object
 14  Frag_Frac_During_Rx                                      3424 non-null   object
 15  Risk_Segment_Prior_Ntm                                   3424 non-null   object
 16  Tscore_Bucket_Prior_Ntm                                  3424 non-null   object
 17  Risk_Segment_During_Rx                                   3424 non-null   object
 18  Tscore_Bucket_During_Rx                                  3424 non-null   object
 19  Change_T_Score                                           3424 non-null   object
 20  Change_Risk_Segment                                      3424 non-null   object
 21  Adherent_Flag                                            3424 non-null   object
 22  Idn_Indicator                                            3424 non-null   object
 23  Injectable_Experience_During_Rx                          3424 non-null   object
 24  Comorb_Encounter_For_Screening_For_Malignant_Neoplasms   3424 non-null   object
 25  Comorb_Encounter_For_Immunization                        3424 non-null   object
 26  Comorb_Encntr_For_General_Exam_W_O_Complaint,_Susp_Or_Reprtd_Dx  3424 non-null   object
 27  Comorb_Vitamin_D_Deficiency                              3424 non-null   object
```

```
 35  Comorb_Osteoporosis_without_current_pathological_fracture  3424 non-null   object
 36  Comorb_Personal_history_of_malignant_neoplasm            3424 non-null   object
 37  Comorb_Gastro_esophageal_reflux_disease                  3424 non-null   object
 38  Concom_Cholesterol_And_Triglyceride_Regulating_Preparations  3424 non-null   object
 39  Concom_Narcotics                                         3424 non-null   object
 40  Concom_Systemic_Corticosteroids_Plain                    3424 non-null   object
 41  Concom_Anti_Depressants_And_Mood_Stabilisers             3424 non-null   object
 42  Concom_Fluoroquinolones                                  3424 non-null   object
 43  Concom_Cephalosporins                                    3424 non-null   object
 44  Concom_Macrolides_And_Similar_Types                      3424 non-null   object
 45  Concom_Broad_Spectrum_Penicillins                        3424 non-null   object
 46  Concom_Anaesthetics_General                              3424 non-null   object
 47  Concom_Viral_Vaccines                                    3424 non-null   object
 48  Risk_Type_1_Insulin_Dependent_Diabetes                   3424 non-null   object
 49  Risk_Osteogenesis_Imperfecta                             3424 non-null   object
 50  Risk_Rheumatoid_Arthritis                                3424 non-null   object
 51  Risk_Untreated_Chronic_Hyperthyroidism                   3424 non-null   object
 52  Risk_Untreated_Chronic_Hypogonadism                      3424 non-null   object
 53  Risk_Untreated_Early_Menopause                           3424 non-null   object
 54  Risk_Patient_Parent_Fractured_Their_Hip                  3424 non-null   object
 55  Risk_Smoking_Tobacco                                     3424 non-null   object
 56  Risk_Chronic_Malnutrition_Or_Malabsorption               3424 non-null   object
 57  Risk_Chronic_Liver_Disease                               3424 non-null   object
 58  Risk_Family_History_Of_Osteoporosis                      3424 non-null   object
 59  Risk_Low_Calcium_Intake                                  3424 non-null   object
 60  Risk_Vitamin_D_Insufficiency                             3424 non-null   object
 61  Risk_Poor_Health_Frailty                                 3424 non-null   object
 62  Risk_Excessive_Thinness                                  3424 non-null   object
 63  Risk_Hysterectomy_Oophorectomy                           3424 non-null   object
 64  Risk_Estrogen_Deficiency                                 3424 non-null   object
 65  Risk_Immobilization                                      3424 non-null   object
 66  Risk_Recurring_Falls                                     3424 non-null   object
 67  Count_Of_Risks                                           3424 non-null   int64
dtypes: int64(2), object(66)
memory usage: 1.8+ MB
```

```
[ ] data.describe()
```

	Dexa_Freq_During_Rx	Count_Of_Risks
count	3424.000000	3424.000000
mean	3.016063	1.239486
std	8.136545	1.094914
min	0.000000	0.000000
25%	0.000000	0.000000
50%	0.000000	1.000000
75%	3.000000	2.000000
max	146.000000	7.000000

Unique elements in each Column

```
print(data.columns.unique)

<bound method Index.unique of Index(['Persistency_Flag', 'Gender', 'Race', 'Ethnicity', 'Region',
       'Age_Bucket', 'Ntm_Speciality', 'Ntm_Specialist_Flag',
       'Ntm_Speciality_Bucket', 'Gluco_Record_Prior_Ntm',
       'Gluco_Record_During_Rx', 'Dexa_Freq_During_Rx', 'Dexa_During_Rx',
       'Frag_Frac_Prior_Ntm', 'Frag_Frac_During_Rx', 'Risk_Segment_Prior_Ntm',
       'Tscore_Bucket_Prior_Ntm', 'Risk_Segment_During_Rx',
       'Tscore_Bucket_During_Rx', 'Change_T_Score', 'Change_Risk_Segment',
       'Adherent_Flag', 'Idn_Indicator', 'Injectable_Experience_During_Rx',
       'Comorb_Encounter_For_Screening_For_Malignant_Neoplasms',
       'Comorb_Encounter_For_Immunization',
       'Comorb_Encntr_For_General_Exam_W_O_Complaint,_Susp_Or_Reprtd_Dx',
       'Comorb_Vitamin_D_Deficiency',
       'Comorb_Other_Joint_Disorder_Not_Elsewhere_Classified',
       'Comorb_Encntr_For_Oth_Sp_Exam_W_O_Complaint_Suspected_Or_Reprtd_Dx',
       'Comorb_Long_Term_Current_Drug_Therapy', 'Comorb_Dorsalgia',
       'Comorb_Personal_History_Of_Other_Diseases_And_Conditions',
       'Comorb_Other_Disorders_Of_Bone_Density_And_Structure',
       'Comorb_Disorders_of_lipoprotein_metabolism_and_other_lipidemias',
       'Comorb_Osteoporosis_without_current_pathological_fracture',
       'Comorb_Personal_history_of_malignant_neoplasm',
       'Comorb_Gastro_esophageal_reflux_disease',
       'Concom_Cholesterol_And_Triglyceride_Regulating_Preparations',
       'Concom_Narcotics', 'Concom_Systemic_Corticosteroids_Plain',
       'Concom_Anti_Depressants_And_Mood_Stabilisers',
       'Concom_Fluoroquinolones', 'Concom_Cephalosporins',
       'Concom_Macrolides_And_Similar_Types',
       'Concom_Broad_Spectrum_Penicillins', 'Concom_Anaesthetics_General',
       'Concom_Viral_Vaccines', 'Risk_Type_1_Insulin_Dependent_Diabetes',
       'Risk_Osteogenesis_Imperfecta', 'Risk_Rheumatoid_Arthritis',
       'Risk_Untreated_Chronic_Hyperthyroidism',
       'Risk_Untreated_Chronic_Hypogonadism', 'Risk_Untreated_Early_Menopause',
       'Risk_Patient_Parent_Fractured_Their_Hip', 'Risk_Smoking_Tobacco',
       'Risk_Chronic_Malnutrition_Or_Malabsorption',
       'Risk_Chronic_Liver_Disease', 'Risk_Family_History_Of_Osteoporosis',
       'Risk_Low_Calcium_Intake', 'Risk_Vitamin_D_Insufficiency',
       'Risk_Poor_Health_Frailty', 'Risk_Excessive_Thinness',
       'Risk_Hysterectomy_Oophorectomy', 'Risk_Estrogen_Deficiency',
       'Risk_Immobilization', 'Risk_Recurring_Falls', 'Count_Of_Risks'],
      dtype='object')>
```

Data Problems

Data problems such as irrelevant columns, Null values, duplicates, skewed data, outliers and many others may cause bad predictions …

So we need to check if we have one of them to know then how to overcome it .

- **Skewed Data :**

```
[497] def measure_skew_kurtosis(cols):
          for col in cols:
              print(col)
              result = data[[col]].agg(['skew', 'kurtosis']).transpose()
              print(result)
      measure_skew_kurtosis(numeric_col)
```

```
Dexa_Freq_During_Rx
                        skew   kurtosis
Dexa_Freq_During_Rx  6.80873  74.758378
Count_Of_Risks
                    skew  kurtosis
Count_Of_Risks  0.879791  0.900486
```

```
[498] #skew and kurtosis values
      data.agg(['skew', 'kurtosis']).transpose()
```

	skew	kurtosis
Dexa_Freq_During_Rx	6.808730	74.758378
Count_Of_Risks	0.879791	0.900486

- **Outliers**

```
# creating a box plot of numerical columns against persitency flag to identify outliers
def boxplot(data, cols):
    for col in cols:
        sns.set_style('whitegrid')
        sns.boxplot(x='Persistency_Flag', y=col, data=data)
        plt.title('Boxplot of ' + col)
        plt.ylabel(col) #setting text for y axis
        plt.show()
boxplot(data, numeric_col)
```

- **Duplicates**

⇨ There is no duplicates, Having duplicates leads often to overfitting

- **Missing Values**

```
[ ] # Total number of missing values
    data.isnull().sum().sum()
```

```
0
```

⇨ No missing Values

Solutions

- **Removing duplicates if they exists**
- **Dropping unsignificant columns**
- **Eliminating Skewed data**

```
[501] #Checking skew after transformation
      data.agg(['skew', 'kurtosis']).transpose()
```

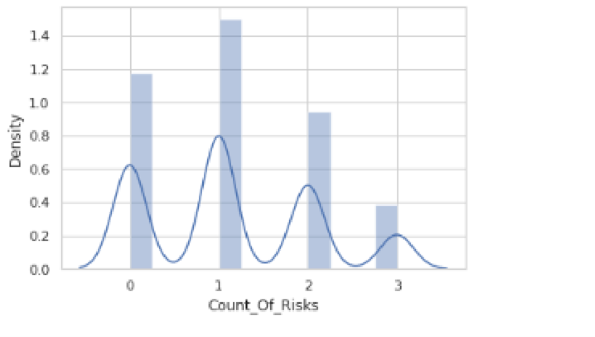
	skew	kurtosis
Dexa_Freq_During_Rx	6.808730	74.758378
Count_Of_Risks	0.879791	0.900486
log_Dexa	1.405860	0.624570
log_Count_Risks	-0.091583	-1.006414

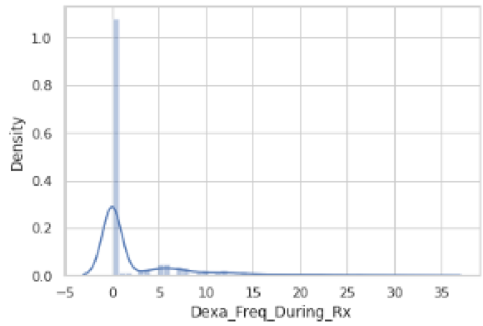
Example of removing 99% Percentile

```
[505] # To remove the 99th percentile
      q = data['Dexa_Freq_During_Rx'].quantile(0.99)
      data_1 = data[data['Dexa_Freq_During_Rx']<q]
      data_1.describe()
```

	Dexa_Freq_During_Rx	Count_Of_Risks	log_Dexa	log_Count_Risks
count	3389.000000	3389.000000	3389.000000	3389.000000
mean	2.440248	1.240484	0.572915	0.685941
std	5.183446	1.095904	0.997375	0.499826
min	0.000000	0.000000	0.000000	0.000000
25%	0.000000	0.000000	0.000000	0.000000
50%	0.000000	1.000000	0.000000	0.693147
75%	3.000000	2.000000	1.386294	1.098612
max	34.000000	7.000000	3.555348	2.079442

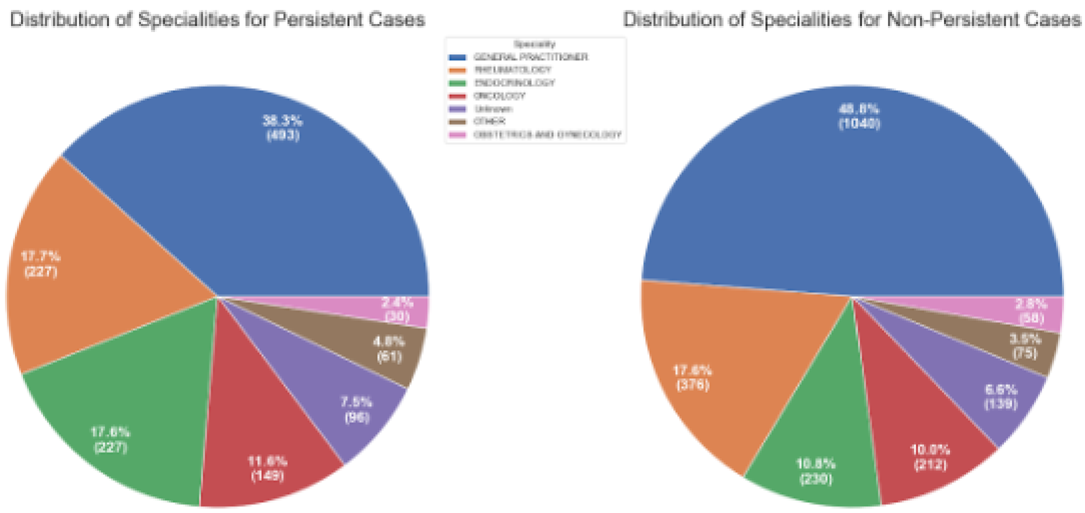
- **Removing outliers**





EDA

Does the speciality of the person who prescribed the drug have any effect on the persistent rate?



We see that both pie charts are pretty similar in distribution of frequency for each speciality. Thus, we can rule out the possibly that one of the factors that the drug is persistent or not is the speciality that prescribed the drug in the first place.

Does 'Ntm_Specialist_Flag' and 'Ntm_Speciality_Bucket' Variables have useful information for the classification task?

Persistency_Flag	0	1
Ntm_Specialist_Flag		
Others	0.686214	0.313786
Specialist	0.552553	0.447447

Persistency_Flag	0	1
Ntm_Speciality_Bucket		
Endo/Onc/Uro	0.473134	0.526866
OB/GYN/Others/PCP/Unknown	0.684884	0.315116
Rheum	0.629565	0.370435

It seems Rheum flag in Ntm_Speciality_Bucket have some useful information.

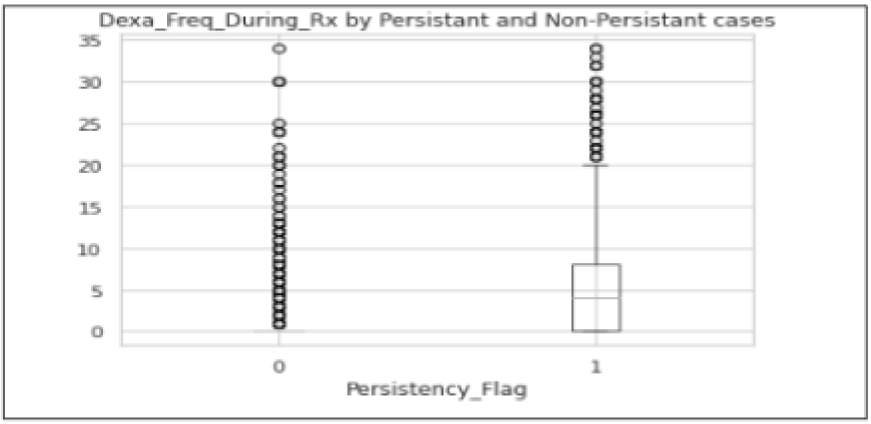
What about 'Gluco_Record_Prior_Ntm', 'Gluco_Record_During_Rx'?

Persistency_Flag	0	1
Gluco_Record_Prior_Ntm		
N	0.627879	0.372121
Y	0.645119	0.354881

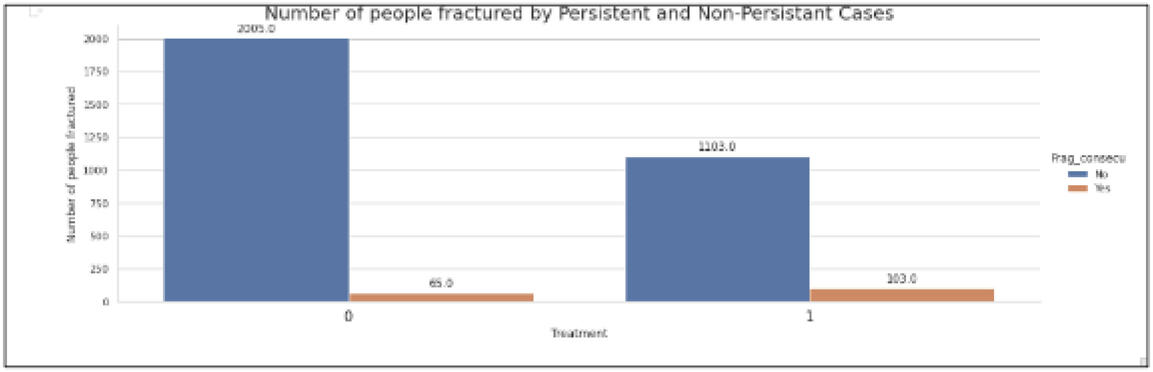
Persistency_Flag	0	1
Gluco_Record_During_Rx		
N	0.691044	0.308956
Y	0.460808	0.539192

Gluco_Record_During_Rx seems to be more useful than Gluco_Record_Prior_Ntm to predict the target

The distribution of Dexa_Freq_During_Rx numbers seems to be higher in the Persistent patients

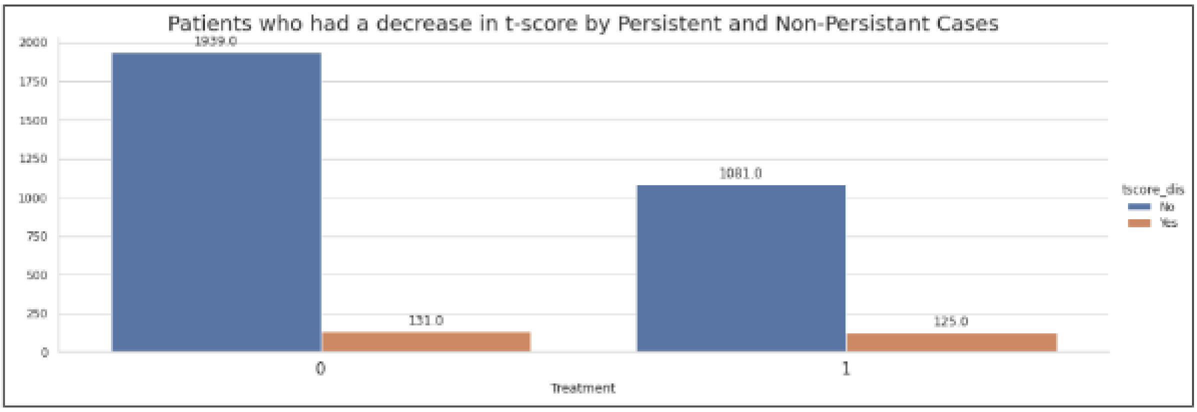


Variables that are recorded during the treatment have more useful information for the classification than others. It can be checked with the percentages shown by Dexa_During_Rx variable.



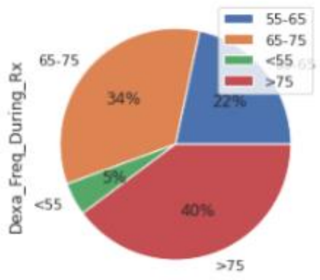
Of the total number of patients, 8% of people were affected by the treatment, weakening their bones

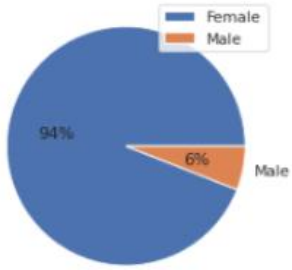
- The count of people affected by the treatment is small, and we can speculate that the treatment not affected considerably to the bones of the patients.

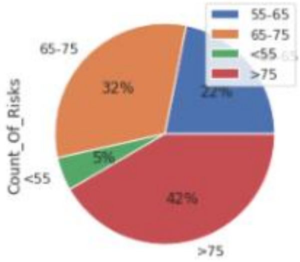


There is 10% of people with treatment who had a decrease in the t-score

- Then there is 90% approximately of people who maintained or improved their t-score.
- In conclusion, the treatment is improving the t-score of the patients.







Dexa_Freq_During_Rx
by Age_Bucket

Count_Of_Risks
by Gender

Counts_Of_Risks
by Age-Bucket

Most of the patients already hold comorbidity factors, while holding risk factors is less common.

Some highlights:

- The main comorbidity factor is related to lipoproteins and metabolism (cholesterol).
- The main risk factor is deficiency in vitamin D.
- More than one third has been found to have taken narcotics.
- 99 % of our sample hold at least one risk, comorbidity and/or concomitant factor.

There are some significant differences between genders:

- Women seem to be more affected by **vitamin D deficiencies**.
- More than twice as many women as men have passed as screening for **malignant neoplasms**.
- Four times as many men as women suffer from **Hypogonadism** (untreated).

- As expected, patients **older than 65** are affected by the mentioned factors in a higher proportion.

- There are some risks and other factors that seem to be significantly higher in **South and West regions**. It might be interesting to find out about socioeconomic factors aside.

- There seem to be some remarkable differences between **Asian and other** races. They are probably due to cultural factors and other behaviours, like medical reviews on a more regular basis (this is just a hypothesis to be found out).

⇨ EDA Summary

The file contained information of 3, 424 patients. For each patient it has demographic information, clinical records, others diseases as risk factor information and also about their physician's speciality.

There are some significant differences between genders (vitamin D deficiencies, screening for malignant neoplasms, Hypogonadism).

Most of the patients already hold comorbidity factors, while holding risk factors is less common.

Patients older than 65 are affected by the mentioned factors in a higher proportion.

There seem to be some remarkable differences between Asian and other races.

Variables that are recorded during the treatment like Dexa_Freq_During_Rx, Dexa_During_Rx and Gluco_Record_During_Rx have more useful information for the classification than others.

Modeling Techniques

Considering the nature of target variable the classification modeling techniques are most suitable for present study. This is a problem of binary classification and models logistic regression , decision tree can be used easily.

We conduct our experiment by implementing the following classification models

```
Classification_models = {"LogisticRegression":LogisticRegression(solver='lbfgs', verbose=0),
                         "DecisionTreeClassifier":DecisionTreeClassifier(),
                         "XGBClassifier":XGBClassifier(),
                         "BaggingClassifier":BaggingClassifier(),
                         "BernoulliNB":BernoulliNB(),
                         "SGDClassifier":SGDClassifier(),
                         "KNeighborsClassifier":KNeighborsClassifier(),
                         "RandomForestClassifier":RandomForestClassifier(),
                         "AdaBoostClassifier":AdaBoostClassifier(),
                         "Gradient Boosting Classifier":GradientBoostingClassifier(),
                         "LGBM Classifier":LGBMClassifier(),
                        }
```

Model Development and Evaluation

```
RandomForestClassifier

Confusion Matrix:
 [[346  64]
 [ 41 377]]
F1 Score:  87.77648428405122
Report :
               precision    recall  f1-score   support

           0       0.89      0.84      0.87       410
           1       0.85      0.90      0.88       418

    accuracy                           0.87       828
   macro avg       0.87      0.87      0.87       828
weighted avg       0.87      0.87      0.87       828
```

The RandomForest Classifier Model performed well on the dataset.