HealthCare: Persistency-of-the-drug

- Group Name: sika (individual)
- Name: Mohamed Sayed Hassan
- Email: msiika70@gmail.com
- Country: Egypt
- College: Cairo university, Faculty of computers and artificial intelligence
- Specialization: Data Science

GitHub repo :

https://github.com/mohamedsiika/Persistency-of-the-drug/

Problem Description:

gather insights on the factors that are impacting the persistency, build a classification for the given dataset.

Business Understanding:

One of the challenges 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.

Drug persistence:

the extent to which a patient acts in accordance with the prescribed interval, and dose of a dosing regimen

Project Lifecycle:

1. Data understanding: discover the data and understands the attributes very well then find the outliers and the null values.
2. remove the unimportant features if there is any and do feature cleansing
3. Analyze the data using EDA to understand the relation between features and the conclusions that I can get from the data
4. Work on model, It is a classification problem (persistent or not persistent) so I am going to use 3 models 1 Linear model 1 ensemble and 1 for boosting.

Data Understanding:

The Dataset is Excel file with size 898 KB .The data consists of 67 features, id of each patient and the target variable. The target variable is the persitency_flag.It has also 3424 observations.

Feature 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

Provider Attributes	NTM - Physician Specialty	Specialty of the HCP that prescribed the NTM Rx
Clinical Factors	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)
	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
Disease/Treatme nt Factor	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
	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

Problems in the Data:

- Null values: 0 Null values in all data

- Outliers: the data is categorical so I searched about what is called “rare values” it’s the values that is represent less than 1% percent of all the observations I found 9 features that has rare values.

- Skewed variables: check if there are variables that has approximately on value.

Approaches to solve the problems.

- Rare values: I will try to solve this problem by substituting the rare values in each categorical with a default value to

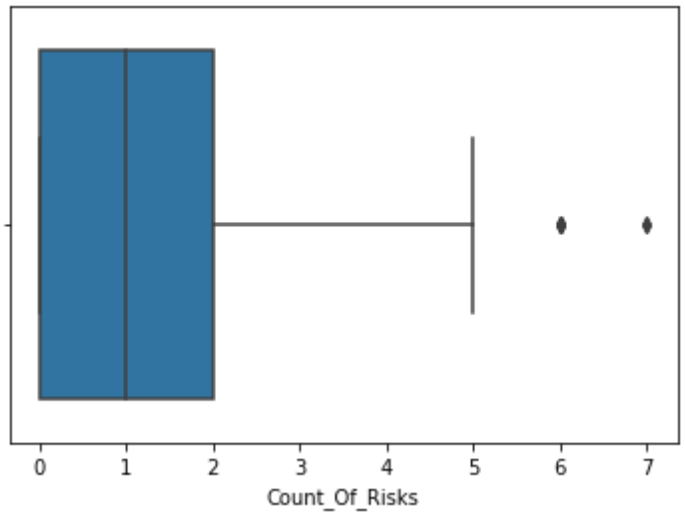
reduce the complexity and the overfitting problem during the training.

Data Cleansing and Transforming:

- **Outliers:** Our dataset contains 2 numerical features and the others are categorical feature so I dealed with them differently so let's start with the numerical features.

1. Numerical Features:

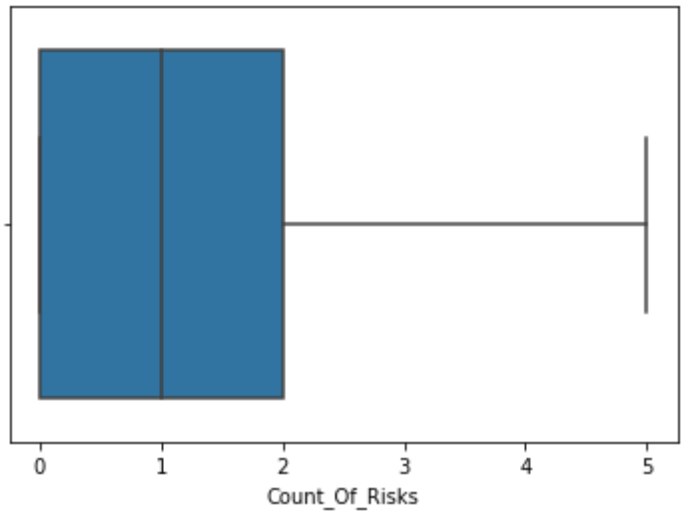
1.1. **Count of Risk** : I will use the box blot to detect the outliers



It appears from the blot that the observations that has values more than 5 are outliers.

Handling: swap the outliers with the median of the data the median in this case is **1.**

Transformed:

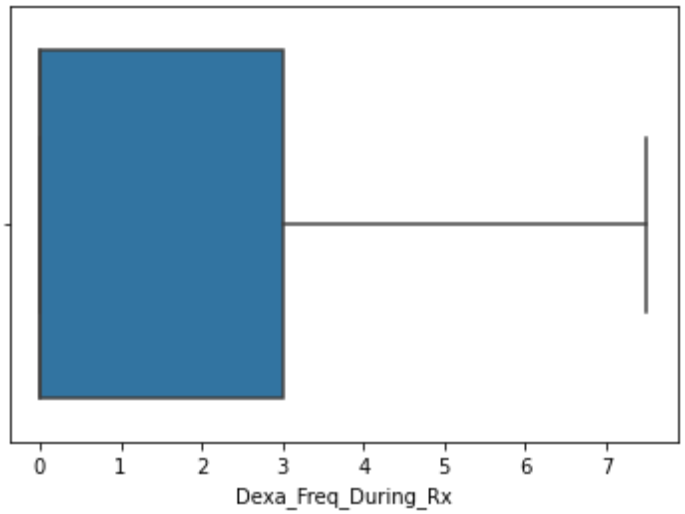


1.2 Dexa_freq_during_Rx: for "Dexa_Freq_During_Rx" i will use inter quartile range in detecting the outliers

```
sort=sorted(df['Dexa_Freq_During_Rx'])
q1=np.percentile(sort,25)
q3=np.percentile(sort,75)
IQR=q3-q1
lwr_bound = q1-(1.5*IQR)
upr_bound = q3+(1.5*IQR)
```

Handling:I will replace values that are higher than the upper bound with the upper bound and the values lower the lower bound with the lower bound.

Transformed:



2. Categorical Features:

I searched about what is called “rare values” it’s the values that is represent less than 1% percent of all the observations I found 9 features that has rare values.

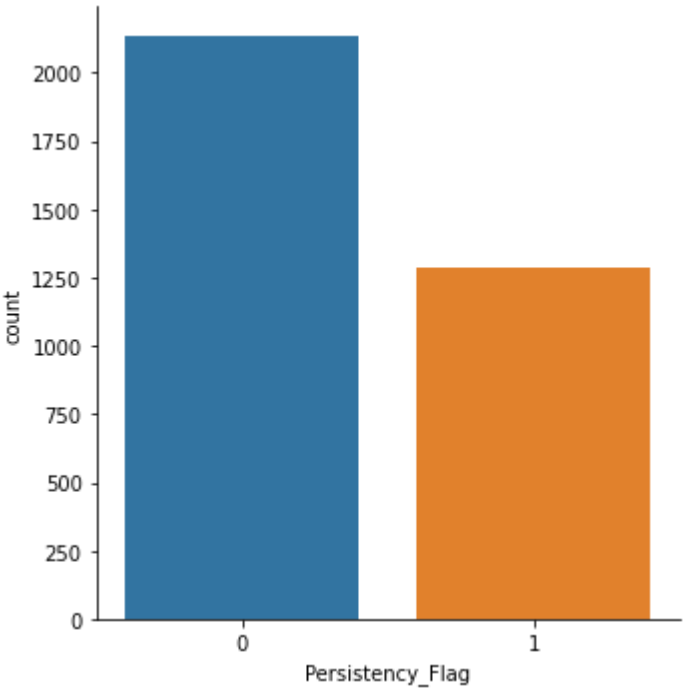
Handling:

I solved this problem by substituting the rare values in each categorical with a default value “Other” to reduce the complexity and the overfitting problem during the training.

```
for i in range(len(df.columns)):
    if df.columns[i]!= "Dexa_Freq_During_Rx" or df.columns[i]!=
"Count_Of_Risks":
        freq=df[df.columns[i]].value_counts(normalize=True)
        map=df[df.columns[i]].map(freq)
        df[df.columns[i]]=df[df.columns[i]].mask(map<0.01,'other')
```

EDA

- First exploring our data sample and check if there is any skewed data and check the percentage of the values inside the target variable to find out what is persistency of the majority of our patients. We find that most of our patients are not persistent by a percentage 62.3%



That will lead us to a point in our analysis that is normal to found Un persistent patients having a specific feature more than the persistent one so we will look to the features that exists in the persistent patients more than the not persistent

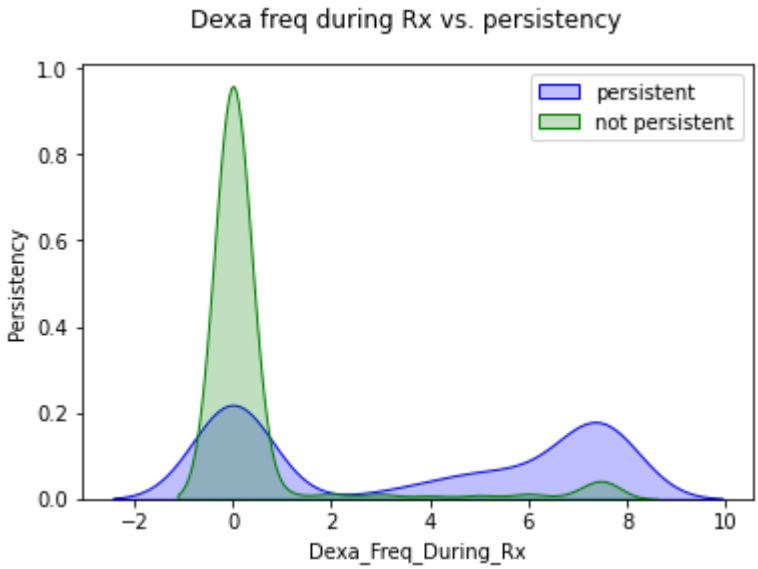
- Now checking the features if there are any skewed variables. I found that there are 6 features approximately have one value by percentage more than 99.7% those will be useless during the training of our model. So I will need to drop them out before the training.

1. Risk_Untreated_Early_Menopause :
    a. 3412 have No Risk
    b. 12 have Risk
2. Risk_Chronic_Liver_Disease:

    a. 3406 have No Risk
    b. 18 have Risk
3. Risk_Estrogen_Deficiency:
    a. 3413 have No Risk
    b. 11 have Risk
4. Risk_Immobilization:
    a. 3410 have No Risk
    b. 14 have Risk
5. Risk_Untreated_Chronic_Hyperthyroidism:
    a. 3422 Have No Risk
    b. 2 have Risk
6. Risk_Osteogenesis_Imperfecta:
    a. 3421 Have No Risk
    b. 3 Have Risk

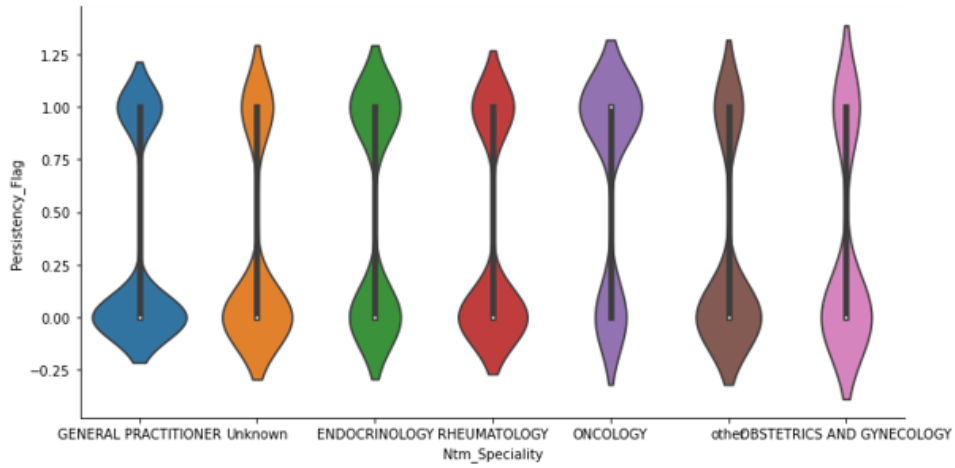
- Now after discovering our sample we need to find the correlation between the numerical variables There is a positive correlation between Dexa scans and the persistency. When the number of Dexa scans increases the patient tends to be more persistent to the drug.

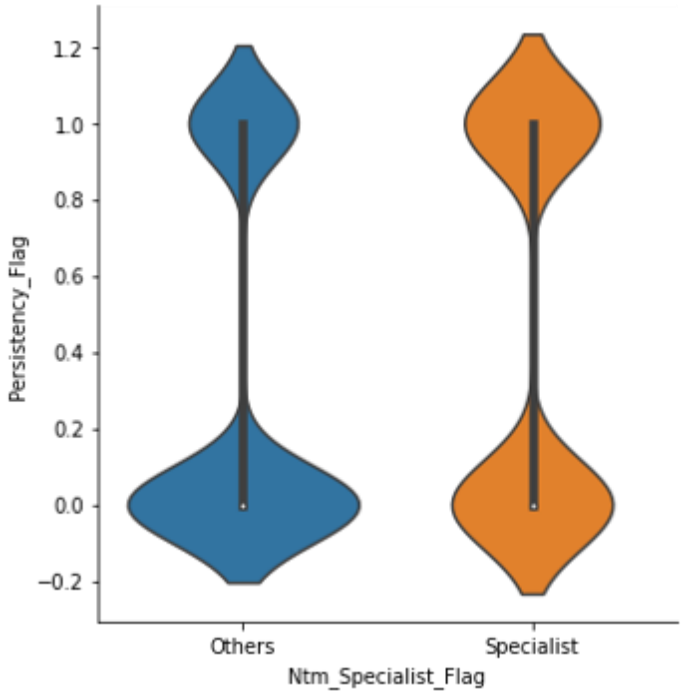
  Most of the not persistent patients have zero Dexa scans taken prior to the first NTM Rx.



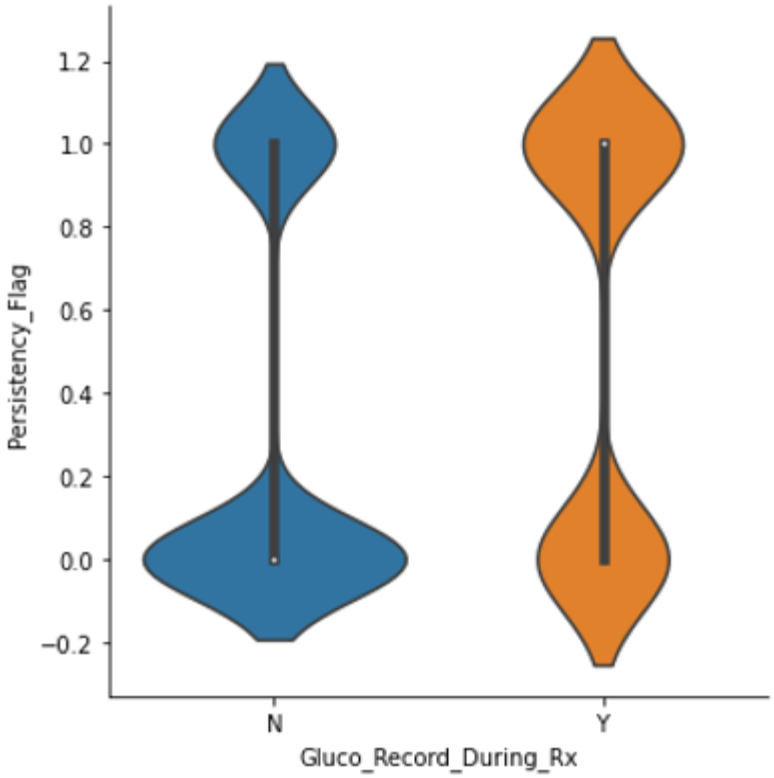
- After discovering the correlation between the numerical features now I have to find if there are relations between the categorical features and the persistency and which of the features is found in the most persistent patients I will use violin plot to visualize the relation between the number of the persistent patients and each feature and capture which of the has high effect. And I found that:

1. SPECIALITY OF THE HCP WHO WROTE THE NTM RX :
    - if the HCP who prescribed the NTM RX is a specialist the patient tends to be more persistent.
    - Non specialists have a large probability to have a non persistent patients . So Patients have to take The Ntm Rx from specialsts only .
    - The oncologists have more persistent patients than the other specialists
    - Oncologists do more Dexa scans to their patients and other scans as well also they do more follow ups so that make sense the have more persistent patients.
    - General practitioners have high percentage of unpersistent patients than the other specialists.

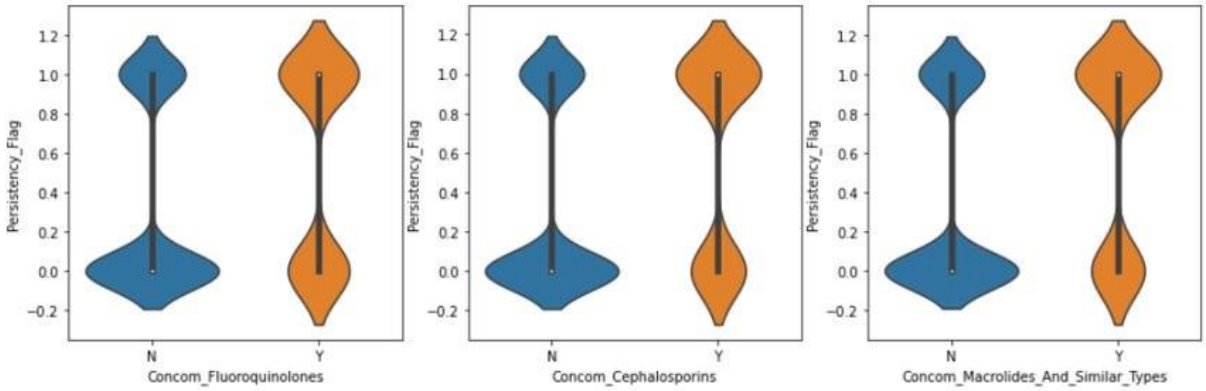




2. PATIENT HAD A GLUCOCORTICOID USAGE DURING THE FIRST CONTINUOUS THERAPY: it appears that a large percentage of the patients who hadn't Glucocorticoid during the first continuous therapy are un persistent.



3. CONCOMITANT DRUGS RECORDED PRIOR TO STARTING WITH A THERAPY: some concomitant drugs have a good effect on the persistency. Form the data we found three drugs highly affect the persistency in a positive way and they are Fluoroquinolones , Cephalosporins ,Macrolides And Similar Types. The patients who hadn't these drugs are less persistent.



The Technical conclusion:

- Dropping the skewed features from the data
- The data contains categorical variables so before training the model I have to encode these variables and I will use dummy encoding
- I will use 3 models:
    c. Support vector machine
    d. Boosting model I will use Ada Boost and Gradient Boosting classifiers
    e. Linear model I will use Linear regression

MODELS

To start building any of the upcoming model I had to drop the skewed features first the encode the data as most of them is categorical features I used Get dummies from pandas to encode my data. Also I built a function to calculate the accuracy.

```
df=df.drop(columns=['Unnamed: 0','Risk_Untreated_Early_Menopause','Risk_Chronic_Liver_Disease','Risk_Estrogen_Deficiency','Risk_Immobilization'])

def accuracy(pred):
  sum=0
  for i in range(len(pred)):
    if pred[i]==y_test[i]:
      sum+=1

  accuracy=sum/len(pred)*100

print(accuracy)
```

1. **Support Vector machine:** the accuracy resulted from this model is **81.89781021897811** I used sklearn SVC to train the model and sk train_test_split to divide the data to test and train and choosed my train size to be 80 percent from the data

2. **Boosting models:** I used 2 examples of ensemble boosting models the first one is Ada Boost Classifier. I adjusted the number of estimators to 250 and the learning rate to 1. the accuracy of this model is **82.35981308411215.** The other

boosting ensemble model is gradient boosting. I adjusted the number of estimators to 250 as the same like Ada boost but I used a small learning rate 0.1 .This model accuracy is 81.89252336448598

3. Linear Regression : Linear regression is a widely used model and the accuracy resulted from it is 80.6420233463035 . I used linear_regression from sklearn.linear_models and used train size 7o percent from the whole data

As you see from the resulting accuracies that the models are close from each other with average accuracy 81 %.