# **Part II**

## Introduction:

The ineffective way that Orange handles complaints is the problem we are attempting to fix. There are currently a number of issues that have an effect on both users and the organization. These difficulties include a lack of efficient procedures, l delays in resolving complaints, and difficulties in identifying the complaint product type (internet or mobile).

Our project's originality and contribution resulted from the use of data analytics techniques to improve Orange's complaint handling procedure. The contribution of our project is the development of specific strategies and recommendations for Orange to enhance their complaint management procedures, resulting in increased client satisfaction and organizational effectiveness.

## Deployment

### *Materials:*

* Source/description of the dataset:

The dataset was collected from Orange, a telecommunication company in Jordan. The dataset contains information about customer complaints in Orange, including details such as the type of customer, status of the complaint, escalation information, timestamps for complaint opening and closure, resolution details, and descriptions of the complaints. It also includes information about the customer group, age bracket of the complaint, preferred communication channel for callbacks, and classification of the complaint. (the meta data is attached with the submission)

* The dataset collection method:

When taking the complaints whether from phone calls or any other ways such as email, the system record every detail of the complaint such as the open date and close date, the employee who took this complaint, if it was escalated or not, and many other specifications.

* The dataset attributes:

CASE\_ID, OFFER\_NAME, CUSTOMER\_TYPE, CUSTOMER\_GROUP, CURRENT\_STATUS, ESCALATION\_FLAG, ESCALATED\_GROUP, OPEN\_DATE, OPEN\_USER, CLOSE\_DATE, CLOSE\_GROUP, CLOSE\_USER, AGE\_BRACKET, ACTUAL\_COMPLAINT, CALLBACK\_MECHANISM, RESOLUTION, RESOLUTION\_DESCRIPTION, CASE\_DESC, OPEN\_GR, COMPLAINT\_TYPE, PRODUCT, CASE.

* The size of the dataset:

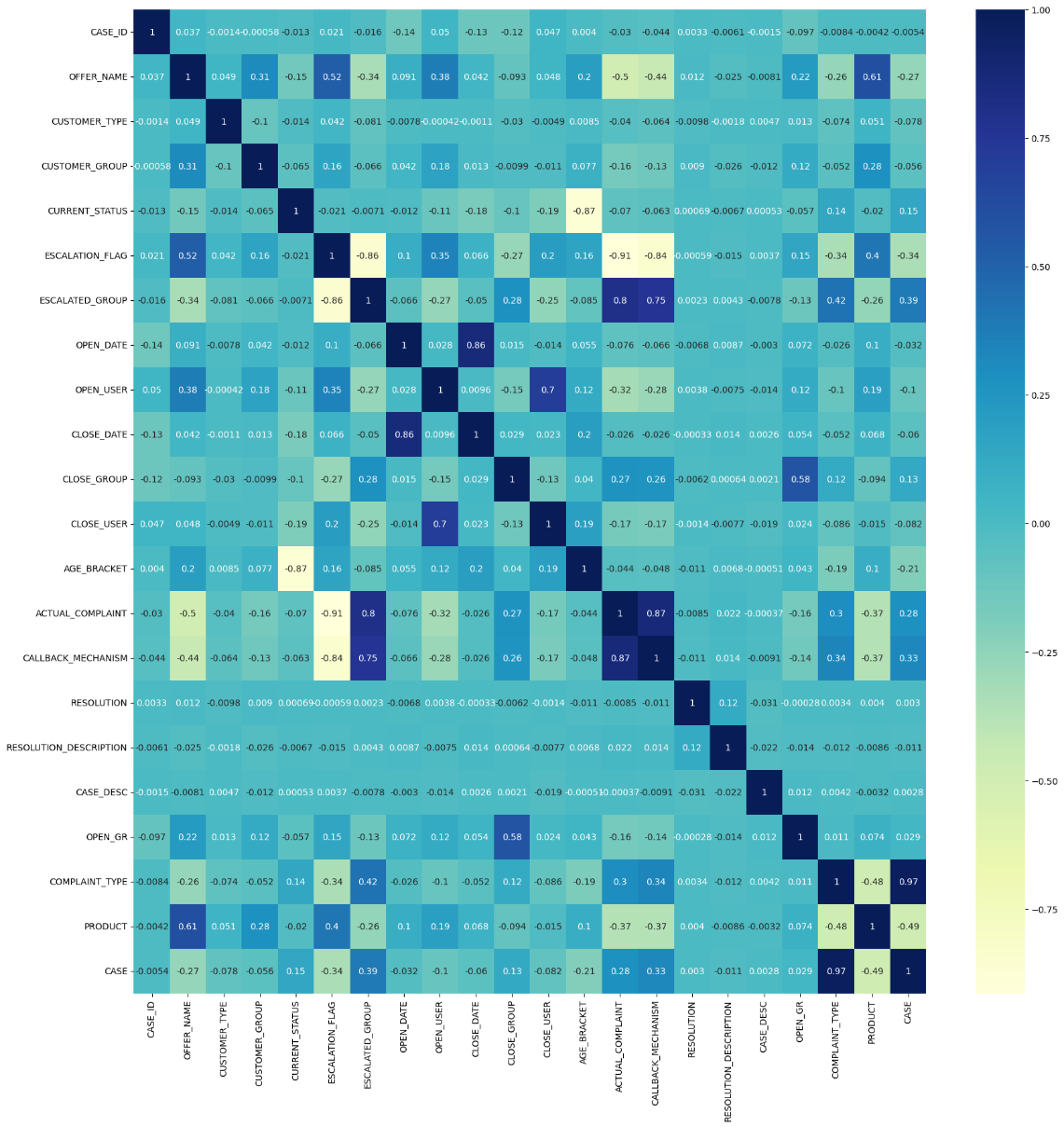
10416\*22

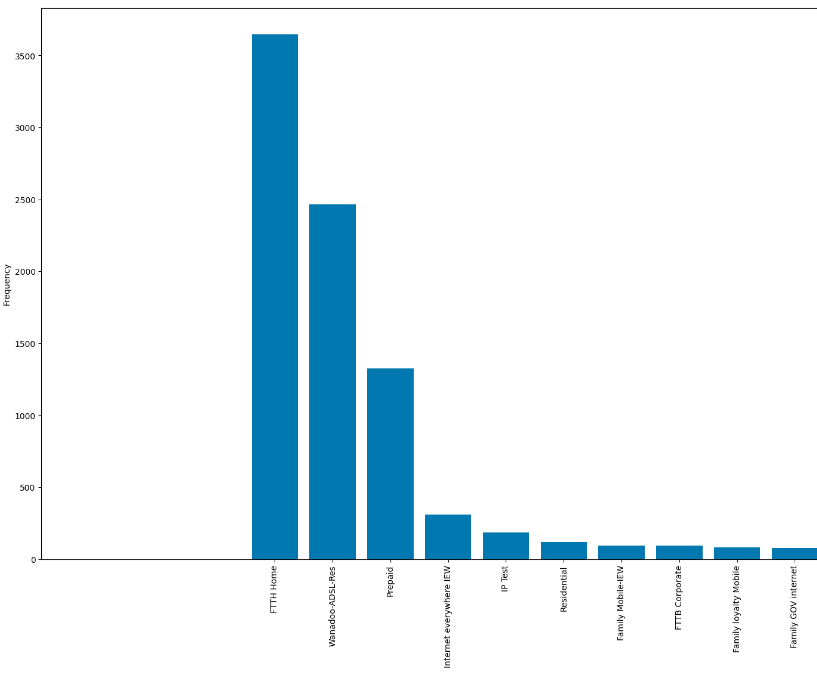
10416 rows, 22 column or feature.

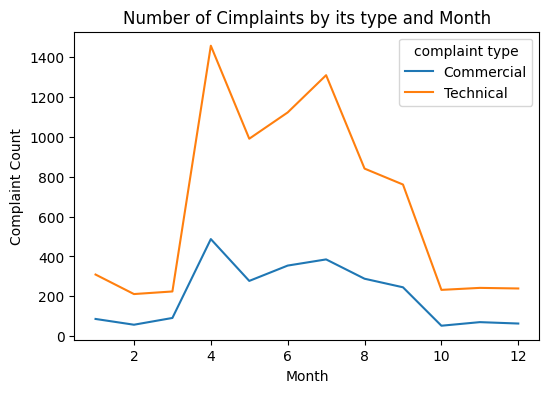
* Dataset pre-processing:

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Column/step Name** | **Description** | **Justification** |
|  | Check for the nulls and duplicated rows | I checked the number of nulls in each column and for duplicated rows | * The number of nulls in each column will lead me how to deal with it whether deleting the column or fill the nulls, etc. * There isn’t any duplicated rows, but if they exist, they should be dropped and removed from the data frame. As keeping them will mislead the model and decrease the accuracy of prediction |
|  | Drop columns | I dropped RESOLUTION, RESOLUTION\_DESCRIPTION, CASE\_DESC, and CASE\_ID columns using drop function | * I dropped these columns (RESOLUTION, RESOLUTION\_DESCRIPTION, CASE\_DESC) because they contains so many nulls that is above 9000 null in each column, so filling the nulls in them won’t be accurate and it can make the model inaccurate. * I dropped the case id column because it’s not necessary or needed in modelling and dropping it will make the data frame easier to deal with and faster in any computation or analysis performed. |
|  | Filling nulls in OFFER\_NAME, CUSTOMER\_GROUP, CLOSE\_GROUP, OPEN\_GR | I filled the nulls in these column by using group by function and take the mode after the grouping | I used the group by function to group the data frame base on specific columns and take the mode of each group and used transform function to fill it in all the nulls of each group. The columns which data frame were grouped on has been chosen based on the heatmap, so I was looking for the most three columns that has a high correlation value whether in positive or negative and use them in the group by function |
|  | drop the nulls in OPEN\_USER and CLOSE\_USER | 1. I dropped the nulls in the column. | 1. The names is hard to be predicted and they were just 516 row, so I dropped them. |
|  | Fill the nulls in ESCALATED\_GROUP | 1. Filled the nulls by “no escalation group” when the escalation flag is no 2. Then fill the rest of nulls by group by function as in step 2. | 1. When the escalation flag value is no means that the complaint hasn’t been escalated which means there is no escalation group, so I filled “no escalation group” in nulls. 2. The rest of nulls has been filled using group by as in step 2. |
|  | Fill the nulls in CLOSE\_DATE and AGE\_BRACKET | 1. I converted the open date and close date values from string to datetime data type. 2. Fill the null in age bracket column with (-999) 3. Fill the null in close date by added the age bracket from the open date. 4. Drop the rested null rows in close date which was 19 row | 1. I converted the date in these column from their string to datetime data type in order to do subtraction. 2. I filled the age bracket nulls with illogical value which is (-999) that represent that the case isn’t not closed yet. 3. I added the age bracket to open date and filled it in the close date, so the close date will be before the open date and it’s illogical value but represents that the case isn’t closed yet. 4. I dropped the rested rows which was 19 row because there is now way to know the open date and close date without any additional information, and deleting these rows won’t affect the model performance so much. |
|  | Fill the nulls in the CALLBACK\_MECHANISM | 1. I filled “no callback” when the current status of the complaint is active. 2. I filled the nulls with “phone” when the difference between the open date and close date is less then 1800 seconds = 30 minutes 3. Fill the rest with “no preferred call-back mechanism’’. | 1. When the current status is active it means that the complaint isn’t closed yet, so there is no call-back needed. 2. When the difference between the open and close date is less or equal 30 minutes, it means that is probably the complaint was from phone call because if it was email or SMS, it would take longer to respond so the difference will be more than 30 minutes. 3. The rest of nulls I filled with no preferred call-back because it would be inaccurate to predict customer’s preference. |

* The exploratory analysis made to understand the data?

Firstly, I did the heatmap so I can understand the correlations between all the columns and depend on it in filling the nulls by groupby function.

Moreover, I draw the bar plot of the frequency of each value in customer group column, which showed us that there is only three vales in this feature that repeats a lot, and the other values don’t exist a lot. So, the models can biased with these values. Which will require me to use a robust machine learning model so it cannot be very biased with these values.

Although, I draw this chart the shows the number of complaints by its type and the month the complaint was established. We can figure out that the number of trades for commercial type is higher of technical type at all times of the year. Also, the number of complaints increase from month3 to month 10 and reaches its peak in month 4.

### *Methods:*

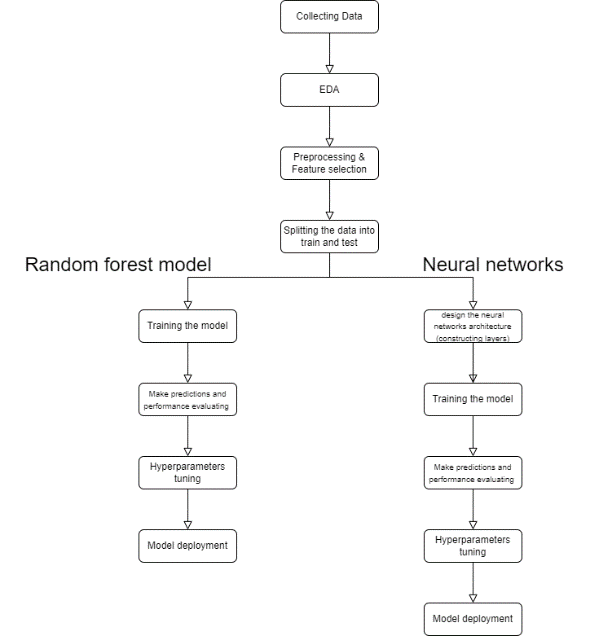
* The models selected are:

Random forest, neural networks

* The selection of each model:

Random Forest is a flexible and effective ensemble learning technique that combines different decision trees. It works well in handling complicated datasets with a lot of attributes. It has robustness and effectively handles missing values and outliers.

Neural Networks are a class of models founded on the neural network of the human brain. They have the ability to recognize complicated relationships and patterns in data. For datasets with high-dimensional input and non-linear relationships, neural networks are particularly suitable. They are able to extract useful representations from the data and capture complex dependencies. Neural networks can also modify and gain knowledge from the dataset, enhancing their performance with additional training data.

* ******Pipeline (architecture) for each model:

**The technical implementation of each model**:

***Model#1 Technical implementation.***

* Import the required libraries.
* Label encoding the categorical data.
* Splitting the data between dependant and independent features
* Splitting the data into training and testing
* Train the model by fit function
* Make predictions based using the trained model.
* Calculate the evaluation metrics.
* After running the model 50 times, calculate the average of these measures.

***Model#2 Technical implementation.***

* Import the required libraries.
* Label encoding the categorical data.
* Splitting the data between dependant and independent features
* Splitting the data into training and testing
* Add the dense layers to the model.
* Add the output layer to the model.
* Compile the model with an optimizer, loss function, and metrics.
* Train the model with the training data
* Make predictions on the test set.
* Convert the predicted probabilities to binary predictions using a threshold of 0.5.
* Calculate and print the accuracy, precision, recall, and classification report.
* How my models can work together with other models/tools/approaches in the organization:

My model can work with other models in the company by doing data analysis and providing valuable insights that can contribute to the decision of the company. and it also enhances resource allocation and optimizes the business process. and by collaborating with stakeholders, we ensure that the project aligns with the organization's objective and encourage a culture of creativity and progress. moreover, providing training to empower employees which can help them in adapting to the new technologies can foster a sense of creativity and improvement in the organization.

* Measures of performance I used to evaluate my model and the rationale for using such metrics:

1- accuracy: represents how well the model is making good predictions

2- recall: represents how many positive cases the model is identifying out of all the actual positive cases

3- precision: represents how many positive cases the model is identifying out of all cases predicted as positive

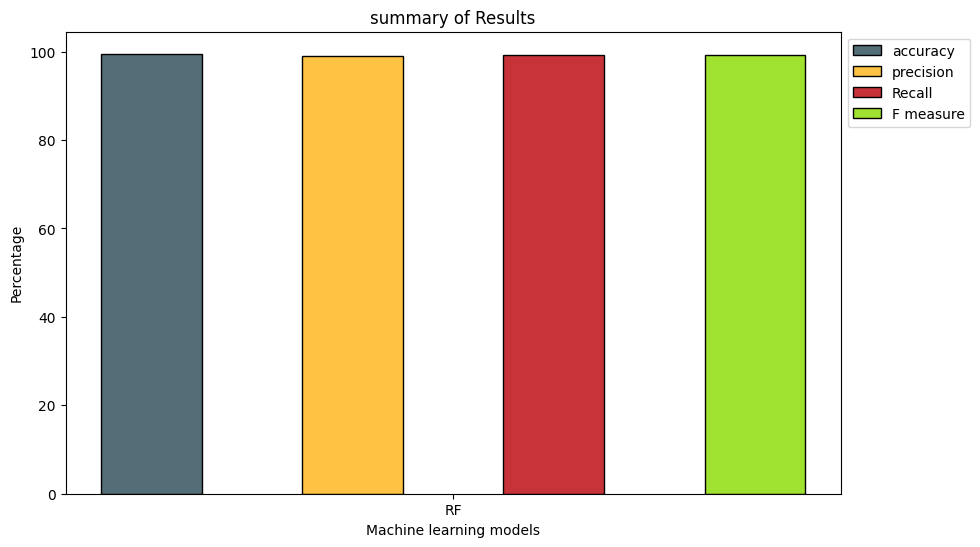
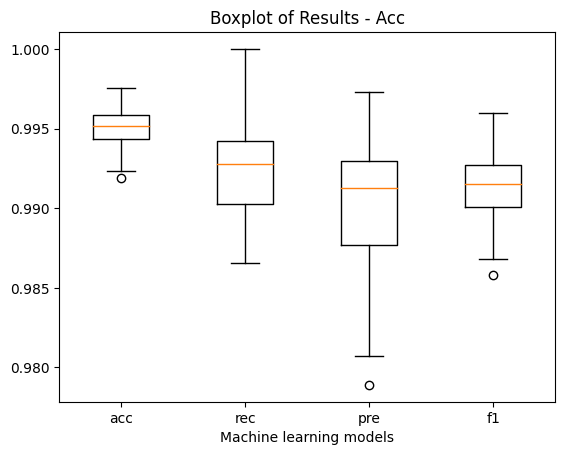
4- f1-score: it represents the balance between both recall and precision

* Evaluation of how based on the performance measures I was able to enhance the model:

by doing the EDA to understand the features and discover patterns or trends within it which helped me to decide whether a column affects the performance of the model. then by pre-processing the data by getting rid of the null values and the unimportant columns or features which means less noise. and by changing all the columns from categorical to numerical and by splitting it by dropping the useless features. and finally using the performance metrics I was able to decide whether the model is performing well or not and do some changes if it doesn't. and trying different parameters of different models I was able to decide if the model is performing well or if it needs to be modified to perform better.

### *Results and Discussion:*

* Analysis of the results.

**Random forest**

Random forest achieved high numbers in the evaluation metrics:

Average Accuracy: 99.502

Average Recall: 99.234

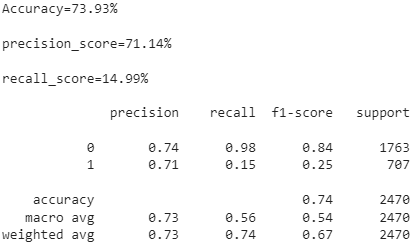
Average Precision: 99.039

Average F1-Score: 99.135

It is clear from the random forest classifier's average accuracy, recall, precision, and F1-score that the model excels at determining whether a complaint belongs to the internet or mobile category in the complaint dataset. The model accurately predicts the right category (internet or mobile) for the complaints, as evidenced by the high average accuracy of 99.502%. The model successfully identifies the pertinent internet or mobile complaints, minimizing false negatives, according to the average recall of 99.234%. The model has a low rate of false positives, as evidenced by the average precision of 99.039%, ensuring that the predicted categories (internet or mobile) are in fact accurate. The model was generally successful in capturing the appropriate categories for the complaints, as evidenced by the average F1-score of 99.135%, which reflects a balanced measure of precision and recall. These findings show that the random forest classifier is reliable and robust in determining whether a complaint falls under the internet or mobile category.

Neural Networks

A picture containing text, screenshot, diagram, colorfulness

Description automatically generated

The accuracy of the neural network model used to analyse the complaint dataset was 73.93%. While the precision score for mobile devices was lower at 14.99%, it was higher for the internet category at 71.14%. These findings suggest that while the model does an acceptable job of correctly classifying complaints about the internet, it struggles to do so for complaints about mobile devices. Recall scores for the internet and mobile categories were low, indicating that the model may not be able to capture all pertinent mobile complaints. Overall, the model's performance in differentiating between complaints made via the internet and those made via mobile devices was only moderate, pointing to the need for additional improvement and refinement.

* Possibility of implementing this project in Jordan:

The project could have advantages if it were implemented in Jordan, according to the outcomes of both the Random Forest and Neural Networks models. The average accuracy of the Random Forest model was 99.502%, while the accuracy of the Neural Networks model was 73.93%. With such high accuracy, it can be concluded that both models are capable of sorting complaints into those pertaining to mobile and the internet. The project's goal is to implement these models in Jordan in order to improve customer service efficiency and streamline the handling of complaints. To ensure a successful implementation, it is crucial to consider additional factors like the unique needs of the Jordanian market, data accessibility, and infrastructure readiness.

* Further enhancements can be done in the future:

In hopes to improve my results of the model, I would change the preprocessing process by choosing different columns for grouping in the group by operation. I would also look into different approaches to filling in the nulls, as they might help a particular model be more accurate. I would increase the number of iterations to improve the models' overall performance because this would allow for deeper learning and perhaps better outcomes. To further increase model effectiveness, I would incorporate feature selection techniques to determine the most crucial attributes for prediction. Last but not least, I would put effort into the Neural Networks model's parameter tuning, maximizing its configuration to achieve the best performance and increase accuracy.

* My role in building/improving this project:

I was the data scientist of the project, and I did everything within the project from EDA to preprocessing, choosing models and performance measures to building the model and evaluating it.

for future plans, I'm planning to improve my model by using different feature selection techniques and more advanced models, in addition to increasing the amount of data and the number of features.