BUSINESS CASE - LEAD QUALITY PREDICTION AND SALES EFFECTIVENESS

BUSINESS CASE

Generating Data Exploration insights for sales effectiveness and creating an ML model to predict lead quality for FicZon Inc, an IT solutions provider. The main task is to help the company increase **sales effectiveness** by automating the categorization of lead quality by using machine learning technology. This is a binary classification task that involves prediction of whether lead quality is high or low potential.

THE DATASET

The data has been retrieved from the Database server of the company using SQL. It consists of customer order details from the year 2018 with **7422 entries and 10 features**. All the features are categorical in nature with details of the order such as 'Sales Agent' 'Location', 'Source' etc. The status column containing 11 different labels has to categorized into high or low potential labels which would serve as our target variable.

The two major **project goals** of the business case are as follows-

1. Data exploration insights – Sales effectiveness.
2. ML model to predict the Lead Category (High Potential, Low Potential)

QUERYING THE DATA

The dataset was queried from MySQL Database server using PyMySQL library which is a python interface for connecting to this server. MySQL-connector (a MySQL driver written in python) was used to establish connection to the database server. After connecting to the database, the table was accessed using the username and password provided and the data was converted to a .csv file.

DATA PREPARATION

This is a preprocessing step involves the manipulation and consolidation of raw data into a standardized format so that it can be used in a model. This would include feature creation, cleaning data, data augmentation, transformations etc.

In this case, we need to convert the date and time related columns into a format which can be used for analysis and be understood by the model. This was done using the pandas .todatetime() function on the 'Created' column and 4 new features were generated which were = 'Day', 'Hour', 'Weekday' and 'Month'.

The new target variable also had to be created which was done by categorizing the status labels as low or high potential and assigning it to the respective values using the map() function.

Unique value columns i.e., email and mobile were dropped since these columns did not provide useful information that would help in prediction of the target variable. Moreover, most of the characters were hidden for privacy purposes.

DATA EXPLORATION INSIGHTS

The first project goal involved getting insights from data which was done during the exploratory data analysis phase.

A detailed **univariate analysis** was conducted and visualized using charts, count plots and histograms using the seaborn library. Some insights from the analysis were as follows-

- Direct calls were the most frequently used source with 34% observations followed by Live Chat-Direct with 25% and Website 22% of all observations.
- The number of customer orders in the domestic market were greater in number with almost 60% of the total as compared to sales in the international markets that were less than 6%. Bangalore was the domestic location with most sales while UAE was the international location with the most orders.
- Most orders were received on Monday while least on Sunday and the percentage of orders placed decreased as the week progressed.

- The maximum orders were in the month of June while the least were in the month of December. Most sales happened mid-year with the second and third quarter of the year making the maximum sales. The company's sales dip towards the end and the beginning of the year i.e., the months of December and January.

- More orders were placed during the beginning of the month as compared to the end or middle.
- The orders placed were less during the early morning hours i.e. 8:00 am, then gradually increase and peak at 11:00 am and finally decrease as the day progresses.

Bivariate analysis was also performed using count plots and histograms that revealed interesting insights in the relationship of each variable with the target variable. The insights were as follows-

- Product ID 19 and 20 had higher lead potential as compared to other products i.e., customers seem to purchase these products more.

- Clients received through Customer referrals and recommendations generated the highest lead potential. Existing clients/customers and personal contacts are sources that generate higher lead qualities as well.

- International locations seem to have more high potential customers as compared to domestic locations although a greater number of orders come from the domestic market. UK, Australia and UAE have almost half as more high potential customers as compared to low.

- The month of October had the highest potential customers while the month of May had the lowest potential customers.
- The first week i.e., 5th-8th days of the month have the lowest potential customers. Despite there being more orders placed during this time, most of them are low potential.

DATA PREPROCESSING

MISSING VALUES

The data set had 4 features with missing values. Rather than imputing the rows with these values were dropped since there were only 156 missing values which made up only about 0.02% of total observations. Moreover, since some rows contained multiple columns with NaN values, the actual number of rows dropped reduced to 94.

CATEGORICAL FEATURE ENCODING

One of the main challenges of this dataset is that it contains multiple high cardinality categorical features which need to be encoded. Using a combination of various types of encoding for different columns like target encoding, one hot encoding, manual encoding etc. would be the best approach since it would help reduce the number of new dimensions created and thus avoid the curse of dimensionality.

Since the target variable was binary in nature, it was manually encoded by assigning 0 to Low Potential and 1 to High Potential. One Hot Encoding was used for 'Sales_Agent', "Product_ID" and 'Delivery_Mode' features. ‘Source’ and ‘Location’ features had comparatively higher cardinality and thus one hot encoding would result in generation of too many new dimensions. To avoid the curse of dimensionality, target encoding was used for these features. Target encoding involves taking the mean of the response variable of each group and assigning it to all observations in that group.

For different models I experimented with various combinations of encoding strategies such as target encoding, cyclic feature encoding etc and the best performing ones were chosen and displayed in the notebook.

HANDLING OUTLIERS

Since all of our features were categorical in nature, there was no need to handle outliers.

MODEL SELECTION

After pre-processing, the data was fit to 4 models- Artificial Neural Network (ANN), Support Vector Classifier, Logistic Regression and XGBOOST. For all the models, the data was split into training and testing set using sklearn’s train test split method.

EVALUATION

The following evaluation metrics were calculated for each model-

1. Recall- it is a measure of from the total number of positive results how many positives were correctly predicted by the model. It is literally is how many of the true positives were recalled (found). Recall= $TP/(TP+FN)$
2. Precision- it is a measure of amongst all the positive predictions, how many of them were actually positive. Precision=$TP/(TP+FP)$
3. F1 score- is defined as the harmonic mean of Precision and Recall. It sums up the predictive performance of a model by combining two otherwise competing metrics — precision and recall.
4. Accuracy- it's defined as the total number of correct classifications divided by the total number of classifications. It tells us the fraction of predictions our model got right.

In this use case**, Recall** was selected as our **primary evaluation metric**. The number of false negatives i.e., a customer actually being high potential but model predicting low potential must be low. We could be lenient on the False Positives i.e., customer being low potential but model predicting customer to be high potential i.e., lower Precision score for higher Recall. This is because it would be more costly to the business to lose out on a high potential customer than following up with a low potential customer.

The model evaluation scores were as follows-

Model	**ANN**	**XGBOOST**	**Support Vector Classifier/ Hyper parameter tuned model**	**Logistic Regression**
Recall	0.78	0.56	0.52 /0.52	0.56
Precision	0.57	0.65	0.60 /0.62	0.68
Accuracy	0.66	0.69	0.65 /0.66	0.70

F1 score	0.66	0.60	0.55 /0.56	0.61
Avg cross validation score	0.66	0.32	0.30	0.22
Std of cross validation score	0.03	0.17	0.13	0.11

**The scores after the slash are the model's scores after hyperparameter tuning.*

From the above table it is evident that ANN performs best with respect to recall score which is our model selection criteria. The other models performed poorly on recall and even after hyperparameter tuning there was no significant change in the evaluation metrics. Moreover, cross validation scores revealed that the model's scores were poor and unreliable. Thus, **ANN** was selected as the **final model** with a recall score of 0.78.

The ANN architecture consisted of two hidden layers of 32 and 6 units each utilizing a ReLU activation function. A fully connected layer with a sigmoid activation function was the final layer. The model was compiled using the ADAM optimizer and 'binary cross entropy' as the loss function. I experimented with several combinations of hidden layers and number of neurons/ nodes and found the above combination of parameters to be the best performing.

CONCLUSION

In conclusion, the exploratory data analysis revealed some interesting data insights that were visualised using various bar charts, count plots etc. The dataset had almost all categorical features and thus were encoded using a combination of encoding strategies like target encoding and one hot encoding. Missing values were handled by dropping the rows since they made up less than 0.2% of observations. It was observed that most models performed poorly which could be because the dataset was small and didn't have much information. For instance, 2 features of mobile and email had to be dropped because they were unique values which left us with only 8 features. Addition of more informative features like the business type of the customer, business size, past purchases with this company etc. could perhaps help improve model performance. After much experimenting and trials, ANN was found to be the best performing model.