SDM - ENSEMBLE

1. Introduction (E-Shop)

Ensemble Learning was performed out on the E-Shop dataset by training bagging and boosting models to predict if a transaction will take place during a web session or not.

2. Data Preparation

The dataset contains 12245 rows and 14 columns. There are **13 independent variables** and **1 dependent variable (Transaction)**.

Four categorical columns were identified within the E-shop dataset viz. Weekend, Transaction (Target Variable), VisitorType and Month.

The columns Weekend and Transaction contained Boolean datatype and were converted using the **converter function** with 1 & 0. Similarly, VisitorType & Month were converted to numerical values using **label encoding method**.

3. Model Building / Testing & Evaluation (Ensemble) / Observation

Three Decision Tree Ensembles viz. Random Forest (Bagging Technique) and 2 boosting techniques namely Adaptive boosting and Gradient boost were run on the E-Shop dataset to identify the best model that predicts if a transaction will occur during a given web session or not.

Random Forest

- A parallel ensemble learning, also know as Bagging where weak learners are paralleling produced during the training phase.
- In the random forest classifier, grid search cross validation method was carried out to evaluate combinations [200, 250, 300, 350, 400, 450] to identify optimal values in order to be used for hyperparameters. The scoring was set to '**recall**' with the idea to minimize false negatives and cv to 5.
- **It is important to minimize the error rate in the prediction model where an actual transaction has occurred in the real world, but the model predicts it the other way round. This would not only impact the customer base but also incur a major financial loss to E-Shop.**
- The n-estimators obtained from mean cross-validated score of best estimator is **200**. The n-estimators indicates the number of trees in the forest.
- The confusion matrix returned a **FN of 151** and a **TP of 418 for the entire**

dataset. In order to minimize the false negatives, one set of feature selections were done, **but the top 5 significant features couldn't produce a better model.**

Adaptive Boosting Model

Multiple weak learners are generated to combine their predictions to form one strong rule and is carried out sequentially.

Adaptive boosting is implemented by combining several weak learners into a single strong learner. It starts by **assigning equal weightage** to each of data points and a **decision stump is drawn** for a single input feature. The stumps are analyzed and checked **if misclassified in order to assign them higher weights**. New decision stumps will be drawn considering the observations with higher weights as more significant. Another stump will created based on observations if misclassified and will be assigned higher weights. This iteration loop will continue until all misclassified observations fall into the right class.

- In the Adaptive Boosting Model, grid search cross validation method was carried out to evaluate combinations [5, 10, 20, 30, 40, 50, 60, 70] to identify optimal values in order to be used for hyperparameters. The scoring was set to '**recall'** with the idea to minimize false negatives and **cv to 5**.
- The n-estimators obtained from mean cross-validated score of best estimator is **60**.
- **The FN obtained is 154 and the TP 415.**

Gradient Boost

Gradient boosting is also based on sequential ensemble learning that optimizes the loss function of the previous learner. The loss function or the errors needs to be reduced. Weak learners are required for computing predictions and forming strong learners. An additional model will be required to regularize the loss function.

- In the Gradient Boosting Model, grid search cross validation method was carried out to evaluate combinations [10, 20, 30, 40, 50, 60, 70] to identify optimal values in order to be used for hyperparameters. Max Depth defines the maximum length of the tree and it is used to **control overfitting**. The scoring was set to '**recall'** with the idea to minimize false negatives and cv to 5.
- The n-estimators obtained from mean cross-validated score of best estimator is **20, maximum depth obtained is 11** and **maximum leaf nodes is 32**.
- **The FN obtained is 138 and the TP 431.**

4. Final Classification Model

Gradient Boost

From the above 3 models (bagging and boosting), it was observed that, **Gradient Boost** produced the best **False Negative** value of **138** along with most significant variables ranked of the order:

PageValue	0.739696
Month	0.117919
Administrative	0.043157
ProductRelated	0.026075
BounceRate	0.019418
ExitRate	0.016555
Administrative_Duration	0.015116
ProductRelated_Duration	0.009651
VisitorType	0.004444
SpecialDay	0.002810
Informational_Duration	0.002649
Informational	0.002508
Weekend	0.000000

5. Recommendation

The analysis and model prediction suggests that the E-Shop must take into account the following measures:

- It can be confirmed from the data that, higher the **PageValue**, greater the probability that the user will proceed to the goal page and transaction may occur. Similarly, the **Month** of August, September, October and November proved to have the highest number of transactions.
- If the **Administrative** page is visited more than twice, there is higher probability of a transaction occurring during a given session. Similarly, if **ProductRelated** pages are visited more than 30 times, then a transaction will definitely take place during a session. Months like June, July, August, October, November are crucial for ProductRelated visits.
- Higher the **BounceRate** & **ExitRate** during a given session, lower the probability of a transaction occurring.