**ECommerce Purchase Prediction**

**Using**

**Machine Learning**



Prof – Dr Ashok Kumar Patel

Sushanth Reddy Sandadi (02113925)

GVS Navneet (02114209)

Mukesh Kumar Karanam Rameshbabu (02121318)



**Data Description:**

The project uses the Online Shopper’s Purchasing Intentions Dataset compiled by C Okan Sakar and Yomi Kastro. C Okan Sakar is a faculty of Engineering and Natural Sciences at Bahcesehir University, Turkey.

Link(https://archive.ics.uci.edu/ml/datasets/Online+Shoppers+Purchasing+Int ention+Dataset)

The dataset consists of 10 numerical and categorical attributes. There are 12330 sessions that have been complied with. Each session’s data has been added after the closing of the session. Dataset has been created such that each session belongs to a different user in a 1-year period. Of the 12330 sessions in the dataset, 84.5% (10422) were negative class samples that did not end with shopping and the rest (1908) were positive class samples ending with shopping. The dataset consists of 10 numerical and 8 categorical attributes.

They are listed below with their data:

1. Administrative: Represents the number of administrative pages visited by the user in a session. Derived from URL information . (Integer value.)
2. Administrative Duration: Total time spent by the user on administrative pages during their session. Derived from user actions. (Integer value).
3. Informational: Indicates the number of informational pages visited by the user in a session.(Integer value)
4. Informational Duration: Total time spent by the user on informational pages during their session.(Integer value).
5. Product Related: Represents the number of pages related to products visited by the user in a session. (Integer value).
6. Product Related Duration: Total time spent by the user on product-related pages during their session. Derived from user actions and updated in real-time. (Continuous value).
7. Bounce Rates: Refers to the percentage of visitors who enter the site from a specific page and leave without any further interaction. (Continuous value).
8. Exit Rates: Indicates the percentage of visits to a specific page that were the last in the session. (Continuous value).
9. Page Values: Represents the average value for a webpage that a user visited before completing an e-commerce transaction. (Integer value).
10. Special Day: Indicates the proximity of the visit to a special day, affecting the likelihood of completing a transaction. Its value is determined based on e-commerce dynamics such as order and delivery dates. Integer value.
11. Month: Represents the month of the year in which the visit occurred. Categorical value.
12. Operating Systems: Indicates the operating system used by the visitor. Integer value.
13. Browser: Represents the web browser used by the visitor. Integer value.
14. Region: Indicates the geographical region of the visitor. Integer value.
15. Traffic Type: Represents the type of traffic source that brought the visitor to the website. Integer value. Visitor Type: Indicates whether the visitor is a returning or new visitor. Categorical value.
16. Weekend: Boolean value indicating whether the visit occurred on a weekend.
17. Revenue: Binary attribute indicating whether the visit resulted in revenue generation for the website. This is the target attribute.

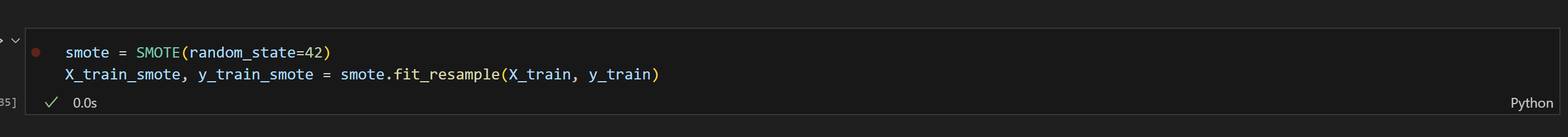
**Modeling:**

We have taken the training and testing split to be 75-25 respectively for the purpose of training the models stated below

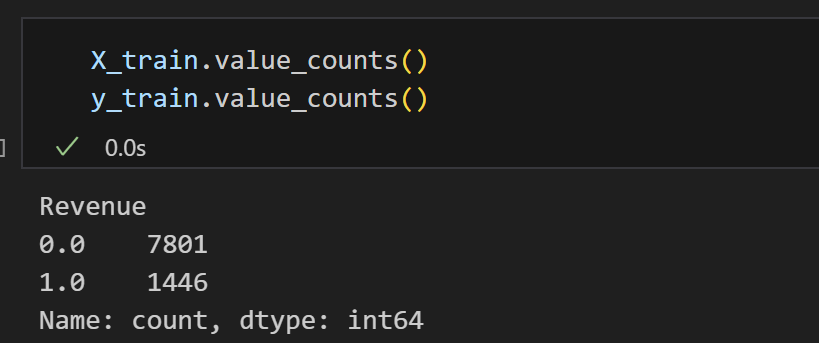
**Oversampling and Under sampling on dataset :**

*For Oversampled dataset,*

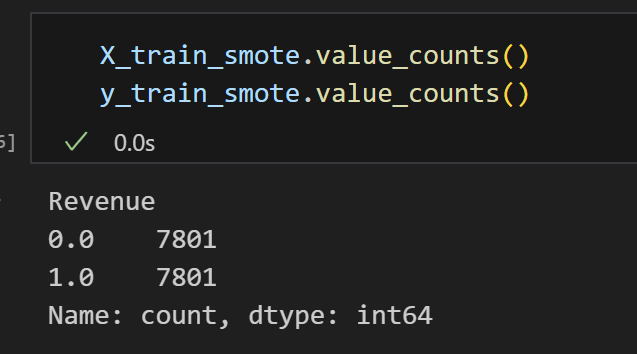
we use SMOTE algorithm. SMOTE stands for Synthetic Minority Oversampling Technique. This is a widely used technique to balance the training set before the learning phase to deal with the imbalance problem we simply duplicate examples from the minority class in the training dataset prior to fitting a model. This can balance the class distribution but does not provide any additional information to the model. An improvement in duplicating examples from the minority class is to synthesize new examples from the minority class. It can be grouped under oversampling algorithm and can be very effective way to balance the data.

****

Before smote,



After smote,

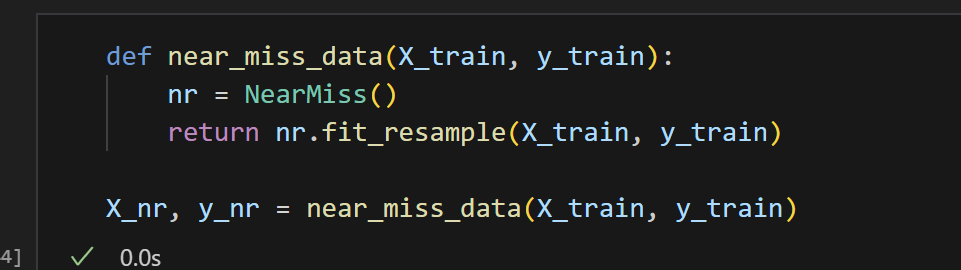


As you can see in the training set the samples pertaining to the yes case ie (1) was pushed up to match the samples of no cases (0)

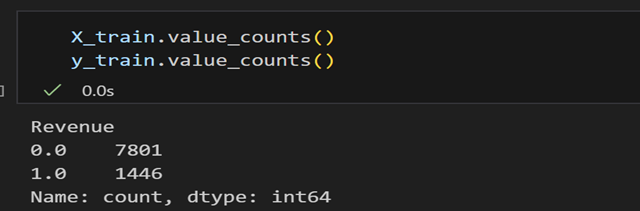
After performing this all the models were performed all over again to make the comparison.

*For Under sampled dataset,*

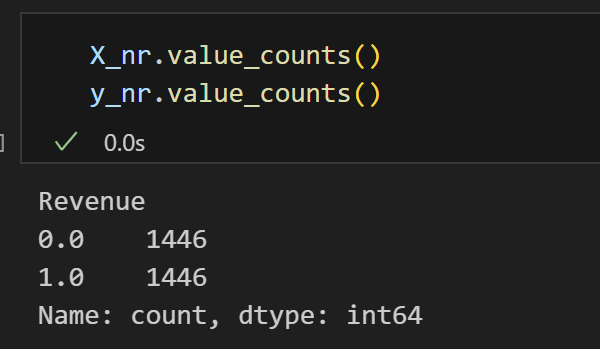
we use Near-Miss algorithm. It is an algorithm that can help in balancing an imbalanced dataset. It can be grouped under under-sampling algorithms and is an efficient way to balance the data. The algorithm does this by looking at the class distribution and randomly eliminating samples from the larger class. When two points belonging to different classes are very close to each other in the distribution, this algorithm eliminates the datapoint of the larger class thereby trying to balance the distribution.



Before near miss,



After near miss,



After performing this all the models were performed all over again to make the comparison.