UNIVERSITY OF CARDIFF

School of Mathematics

2020/2021-CMT307 Applied Machine Learning

Individual Coursework

Title: E-commerce Visitors' Purchasing Intention

By

Karakatsanis Charisios

Student ID number: 2080129

February 2021

QUESTION 1:

ID	PREDICTION	TARGET
1	TRUE	TRUE
2	TRUE	TRUE
3	FALSE	TRUE
4	TRUE	TRUE
5	TRUE	TRUE
6	FALSE	TRUE
7	TRUE	TRUE
8	TRUE	TRUE
9	TRUE	TRUE
10	FALSE	FALSE
11	FALSE	FALSE
12	FALSE	FALSE
13	TRUE	FALSE
14	FALSE	FALSE
15	TRUE	FALSE
16	FALSE	FALSE
17	FALSE	FALSE
18	TRUE	FALSE
19	TRUE	FALSE
20	FALSE	FALSE

CONFUSION MATRIX:

		PREDICTION		
		TRUE	FALSE	TOTAL POINTS
ACTUAL CLASS	TRUE	TP=7	FN=2	9
	FALSE	FP=4	TN=7	11

TP: True Positives: These are to the positive points that were correctly labelled by the classifier.

FP: False Positives: These are the negative points that were incorrectly labelled as positive.

FN: False negatives: These are the positive points that were mislabelled as negative.

TN: True Negatives: These are the negative points that were correctly labelled by the classifier.

Accuracy = $\frac{TP+TN}{TP+TN+FP+FN} = \frac{7+7}{7+7+4+2} = \frac{14}{20} = \frac{7}{10}$

Precision (p) = $\frac{\mathrm{TP}}{\mathrm{TP+FP}} = \frac{7}{7+4} = \frac{7}{11}$

Recall (r) = $\frac{TP}{TP+FN} = \frac{7}{7+2} = \frac{7}{9}$

F-Measure(F) = $\frac{2*precision*recall}{precision+recall} = \frac{2*p*r}{p+r} = \frac{2*\frac{7}{11}*\frac{7}{9}}{\frac{7}{11}+\frac{7}{9}} = \frac{7}{10}$

QUESTION 2:

Data Exploration

Firstly, we detected that our dataset did not contain any missing values.Secondly, through boxplots we noticed that every feature consisted of a large proportion of outliers,but we decided to include them in the dataset so as not to drop any crucial information.Regarding the numerical features we plotted histograms(Figure1) with their corresponding density function to observe how spread our data is and their

distribution.Concerning about categorical attributes we plotted their “countplot”(Figure1)to detect the number of observations in each category.Lastly,we created a correlation matrix in terms of noticing the linear or not dependence between the numerical features.

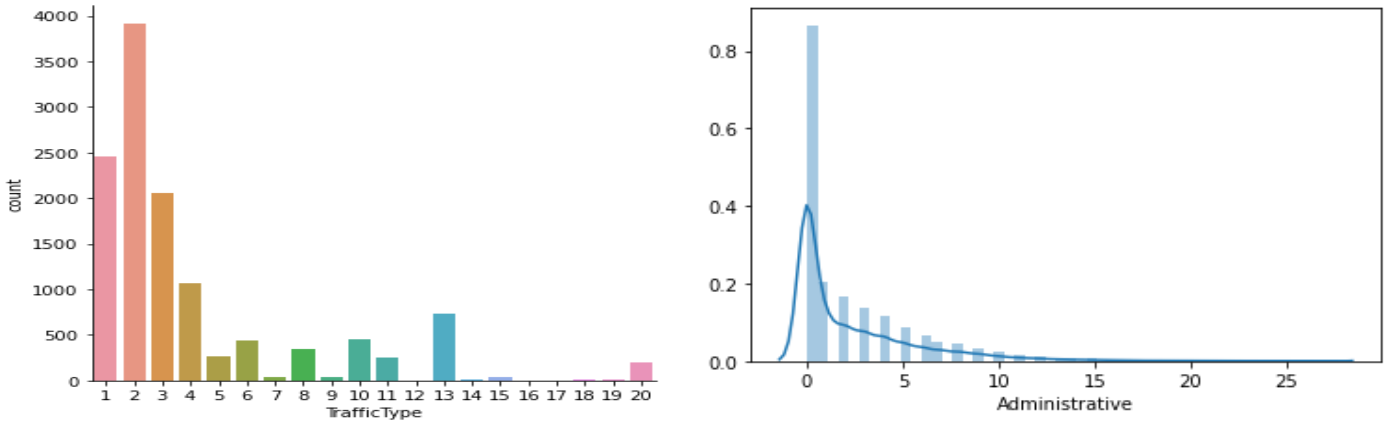

Figure1: On the left it is a ‘countplot’ and on the right it is a histogram to visualize the data.

Data Pre-processing

Regarding Pre-Processing we used 4 different techniques to achieve a higher and better overall performance.

- Multicollinearity refers to a situation in which two or more explanatory unbiased variables are linearly related in terms of a multiple regression model.We used Variance Inflation Factor(VIF) to reduce Multicollinearity[i].Hence,we dropped the values that had a value greater than 5(extreme multicollinearity) in the VIF function.Afterwards,every feature’s value was below 5 and thus we concluded that multicollinearity decreased to moderate levels.
- Data Binning minimizes the effect of small observation errors.For categorical columns,the labels with low frequencies have a negative effect on the robustness of statistical models[ii].Thus,assigning a general category to these less frequent values indicates that robustness of the model will remain at the same level.
- Encoder:OneHotEncoder creates additional attributes based on the number of values in each categorical feature.So,every unique value in every category will be added as a new column in the dataset. Moreover,this method of encoding works in a better way when the categorical data are not ordinal like ours.
- Scaler:PowerTransformer applies a power transformation to each feature to make the data more Gaussian-like to stabilize variance and minimize skewness.Furthermore, optimal scaling factor is determined by maximum likelihood estimation.So,in terms of outliers it can perform in a way that is likely to handle outliers because it is unaffected by them.

Model Implementation,Performance Evaluation

1. Support Vector Machine(SVM):

1.1. Merits:

- Has generalization in practice, so over-fit is less[iii].
- The parameter 'kernel' is the strength of SVM.That means with the proper kernel function any problem can be solved.
- Can find the best margin that separates the classes.This reduces the risk of error on the data.

1.1.1. Parameters:

- Kernel: Kernel function is a similarity measure between data points to map data from input(feature) space to a higher-dimensional feature space.
- Class_weight:In problems where it is desired to give more importance to certain classes, 'class_weight' can be implemented to balance a dataset.We established this parameter to reduce the ratio of 5:1(False:True respectively for output variable).
- C:Regularization parameter.The strength of the regularization is inversely proportional to C>0.

2. XGBoost:

2.1. Advantages:

- Has built-in optimization and regularization(prevents over-fitting).
- Utilizes the power of parallel processing and uses multiple CPU cores to execute the model.
- Makes splits up to the max_depth specified and then start pruning the tree backwards and remove splits beyond which there is no positive gain[iv].

2.1.1. Parameters:

- Scale_pos_weight:Useful for imbalanced classes.
- Max_depth:Increasing this value will make the model more complex.
- Learning_rate:Controls the magnitude of change that is permitted from one tree to another.
- N_estimators:Specifies how many sequential trees we want to make that attempt to correct for prior trees[v].
- Gamma:Minimum loss reduction required to make a further partition on a leaf.
- Min_child_weight:Minimum sum of instance weight needed in a child.
- Subsample:Determines how much of the initial dataset is fair for random sampling during each iteration of the boosting process.

3. Logistic Regression:

3.1. Perks:

- Due to its simple probabilistic interpretation,the training time of logistic regression algorithm comes out to be far less than most complex algorithms[vi].

- In a low dimensional dataset having enough training examples,logistic regression is less prone to over-fitting.
- Easy to implement and provides great training efficiency in some cases.Also,training a model with this algorithm does not require high computation power.

3.1.1. Parameters:

- C:Inverse of regularization strength.
- Class_weight:Can be implemented for the same reason as indicated in SVM.
- Penalty:Used to specify the norm used in the penalization.
- Solver:Represents which algorithm to use in the optimization problem.

Taking everything into consideration and for the advantages and the parameters outlined previously we decided to choose these 3 aforementioned classifiers.

4. Performance Matrices:

- For imbalanced data such as ours it is preferable to use Precision,Recall,F-measure to determine class wise performance of classifiers,while Accuracy usually fail because of imbalance.

5. Evaluate and Optimize:

- Evaluation:Stratified K-fold Cross Validation is done in a supervised way for classification and aims to ensure each class is(approximately) equally represented across each test fold.That's the main reason that this specific technique used for imbalance datasets.Repeated K-fold algorithm makes a robust model as it covers the maximum training and testing operations.Hence, we included them both in our Cross Validation(Evaluation) part.

- Optimization:The process of tuning hyperparameters indicates the best combination of hyperparameter values for a classifier.In our report we chose RandomizedSearchCV with Cross-Validation as mentioned above because it is very efficient in terms of optimization.Furthermore,the matrix in which we calculated the best possible solution,was F-Measure.

Results

1st model consists of the Encoding and the Scaling sections of the pre-processing.

Classifiers/Performance	Accuracy	Precision	Recall	F-Measure
Logistic Regression	0.893934	0.70	0.58	0.63
SVM	0.895556	0.71	0.59	0.64
XGBoost	0.896529	0.71	0.60	0.65

2nd model contains all 4 techniques as indicated in Pre-processing part.

Classifiers/Performance	Accuracy	Precision	Recall	F-Measure
Logistic Regression	0.894908	0.70	0.58	0.64
SVM	0.898476	0.72	0.58	0.65
XGBoost	0.894583	0.70	0.58	0.64

3rd model has every part of *2nd model* plus balance(using specific parameters).

Classifiers/Performance	Accuracy	Precision	Recall	F-Measure
Logistic Regression	0.891609	0.60	0.75	0.67
SVM	0.898690	0.59	0.80	0.68
XGBoost	0.891015	0.65	0.67	0.66

The *3rd model* with the hyperparameters as calculated after tuning.

Classifiers/Performance	F-Measure	AUC
Logistic Regression	0.671	0.671211
SVM	0.670	0.667274
XGBoost	0.674	0.703156

In coclusion,in all 3 models Accuracy is around 90% but it is not a reliable performance metric cause of imbalance.In *1st* and *2nd model* we can observe that Precision is higher than Recall while in the *3rd* it is the other way around.So,we consider F-Measure(Harmonic mean of Precision and Recall) as the most reliable metric of the 4.We chose to balance the *2nd model* based on the results which were slightly better than the ones of the *1st* in terms of F-Measure.Regarding the *3rd model* the results indicate an increase(around 3% for all classifiers) in F-Measure mostly because we used appropriate parameters to balance the data.F-Measure improves negligibly with the hyperparameters while AUC indicates that our model is not that good but there is overall rise regarding performance.Finally, looking the last table depending on AUC it is clear that XGBoost(also has more balanced metrics from the 3rd table)indicates a better overall model performance.

References:

[i] https://en.wikipedia.org/wiki/Variance_inflation_factor

[ii] https://towardsdatascience.com/feature-engineering-for-machine-learning-3a5e293a5114

[iii] https://statinfer.com/204-6-8-svm-advantages-disadvantages-applications/

[iv] http://theprofessionalspoint.blogspot.com/2019/03/advantages-of-xgboost-algorithm-in.html

[v] https://kevinvecmanis.io/machine%20learning/hyperparameter%20tuning/dataviz/python/2019/05/11/XGBoost-Tuning-Visual-Guide.html

[vi] https://iq.opengenus.org/advantages-and-disadvantages-of-logistic-regression/