CSSM 502

HOMEWORK-3

REPORT

- First I replaced all the values in the data set corresponding to ‘missing’, ‘refuse to answer, and ‘don’t know’ with NaN. Then I replaced each with the “most frequent value” in the relevant column.
- I took ‘gender’, ‘education level’, ‘socio-economic status, ‘household income’, ‘religiosity’, ‘age’, and ‘rural or urban residence’ as variables to predict whether a citizen voted or not.
- For the education category, the number ‘96’ refers to the ones who didn’t get any education. Therefore, I replaced it with ‘0’ to put education level in an order with ‘0’ referring to non-educated and ‘9’ referring to doctoral-level education.
- Since ‘gender’, ‘rural or urban residence’, and ‘socio-economic status’ are categorical variables, I converted their types to string and used dictvectorizer for dummy variables.
- Then I split the data into training and test data.
- I used 4 machine learning algorithms which are ‘Gaussian Naïve Bayes’, ‘Random Forest Classifier’, ‘Support Vector Machine’, and ‘K-Nearest Neighbors’ to predict the binary outcome that is whether one voted or not.
- Gaussian NB has the lowest accuracy rate whereas the K-Neighbors algorithm has the best.
- Then, I optimized each model except GaussianNB in terms of hyperparameters using GridSearchCV. After optimization, each has a similar accuracy rate with Random Forest having the best accuracy rate.

Algorithm	Accuracy Rate	Accuracy Rate(Optimized)	Optimal Hyper Parameters
Random Forest	0,818	0,852	max_depth': 25, 'min_samples_leaf': 10, 'min_samples_split': 5, 'n_estimators': 800
SVC	0,819	0,85	C': 1, 'gamma': 0.01, 'kernel': 'rbf'
K-neighbors	0,841	0,85	metric': 'euclidean', 'n_neighbors': 30, 'weights': 'uniform'
GaussianNB	0,76		

Finally, I tried to add other variables which I think could improve the performance of the models. I added ‘main occupation’, ‘current employment status, and ‘marital status’ one by one, checked the accuracy rate for each one and also for different combinations of these three variables. As a result, I did not see any significant improvement in the accuracy rate for any model. Then removed some of the variables, again there was no significant reduction or improvement in terms of performance unless the ‘age’ column is removed. Additionally, when I was reducing the number of variables, I noticed that, after some point, even with the small number of variables the accuracy score stays stable. Then I checked the confusion matrix for each model and I think the reason was that the data was imbalanced meaning there are lots of ‘true’ values rather than ‘false’ values in the target column so even if the model predicted all the target values as ‘true’, the accuracy score was around 82%. Therefore, the accuracy score may not be an appropriate metric for the evaluation of the models. I decided to use. confusion matrix and also the other metrics like ‘precision’ ‘recall’ or “f-1 score” while making comparisons and there were no significant differences between models based on these metrics and confusion matrices instead of GaussianNB which has the worst performance.

Classification report output for RandomForestClassifier:

	Precision	recall	f1-score	support
False	0.78	0.25	0.38	561
True	0.86	0.98	0.92	2552
accuracy			0.85	3113
macro avg	0.82	0.62	0.65	3113
weighted avg	0.84	0.85	0.82	3113