ANALYZING PORTUGESE BANK TELEMARKETING DATASET

In this project, we took the task of analysing a Marketing campaign dataset of a Portuguese banking institution. The classification goal is to predict if the client will subscribe to a term deposit. Analyzing marketing campaign datasets of banks is crucial for banks to stay competitive, improve their marketing strategies, and ultimately, better serve their customers. Analyzing marketing campaign datasets of banks is significant for several reasons:

- Understanding Customer Behavior: Analyzing the marketing campaign datasets of banks can help in understanding customer behavior, such as their preferences, interests, and demographics. By understanding customer behavior, banks can tailor their marketing campaigns to better target their intended audience and increase the chances of success.
- Improving Campaign Effectiveness: Analyzing the data from previous marketing campaigns can help banks identify what worked and what didn't. This information can be used to

improve the effectiveness of future campaigns by optimizing the channels, messaging, and timing of the campaigns.

- Measuring ROI(Rate of Interests): By analyzing the marketing campaign datasets, banks can track the ROI of their campaigns. This information is important to determine whether the investment in the campaign was worthwhile or not.
- Identifying Opportunities: Analyzing marketing campaign datasets can help banks identify opportunities for new products or services. By identifying patterns in customer behavior, banks can develop new products or services that meet customer needs and preferences.
- Benchmarking Performance: Analyzing marketing campaign datasets can help banks benchmark their performance against competitors. This information is important to determine how well the bank is performing in the market and what improvements can be made to remain competitive.

The dataset had many columns pertaining to different aspects of the campaigning journey. Following were the classification of the features :

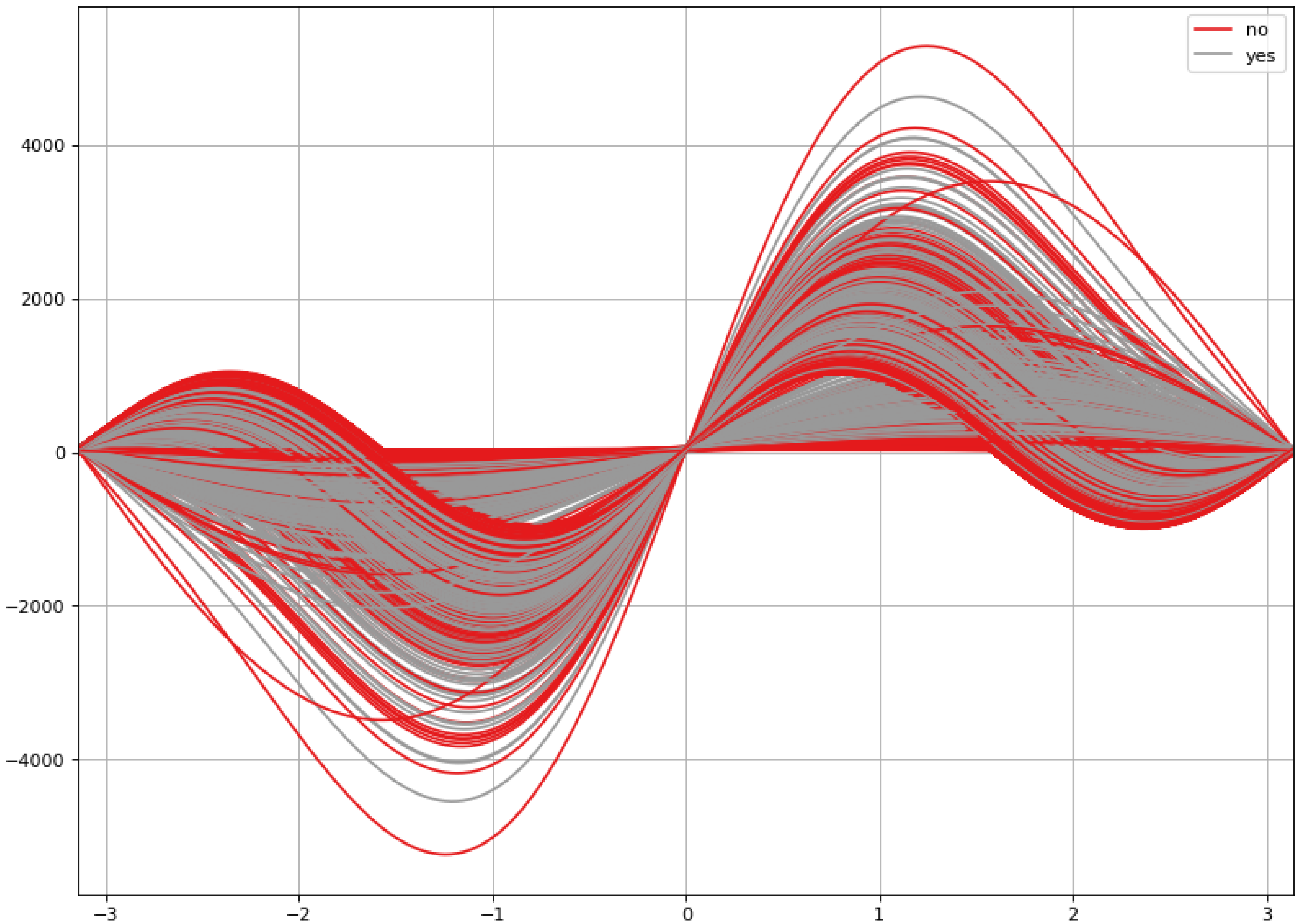
	age	job	marital	education	default	housing	loan	contact	month	day_of_week	...	campaign	pdays	previous	poutcome	emp.var.rate	cons.price.idx	cons.conf.idx	euribor3m	nr.employed	y
0	56	housemaid	married	basic.4y	no	no	no	telephone	may	mon	...	1	999	0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no
1	57	services	married	high.school	unknown	no	no	telephone	may	mon	...	1	999	0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no
2	37	services	married	high.school	no	yes	no	telephone	may	mon	...	1	999	0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no
3	40	admin.	married	basic.6y	no	no	no	telephone	may	mon	...	1	999	0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no
4	56	services	married	high.school	no	no	yes	telephone	may	mon	...	1	999	0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no

- Client Data : age, job, marital status, education, previous housing loan and previous personal loan.
- Related to last contact of the current campaign : Last communication type, Last contact month, Day of the week contacted and duration.
- Miscellaneous attributes : No of contacts in this campaign, Days since last contact, No of contacts in previous campaign and Outcome of previous campaign
- Social and Economics Attributes : Employee Variation Rate, Consumer Price Index, Consumer Confidence Index, Euribor 3 month rate and Number of Employees

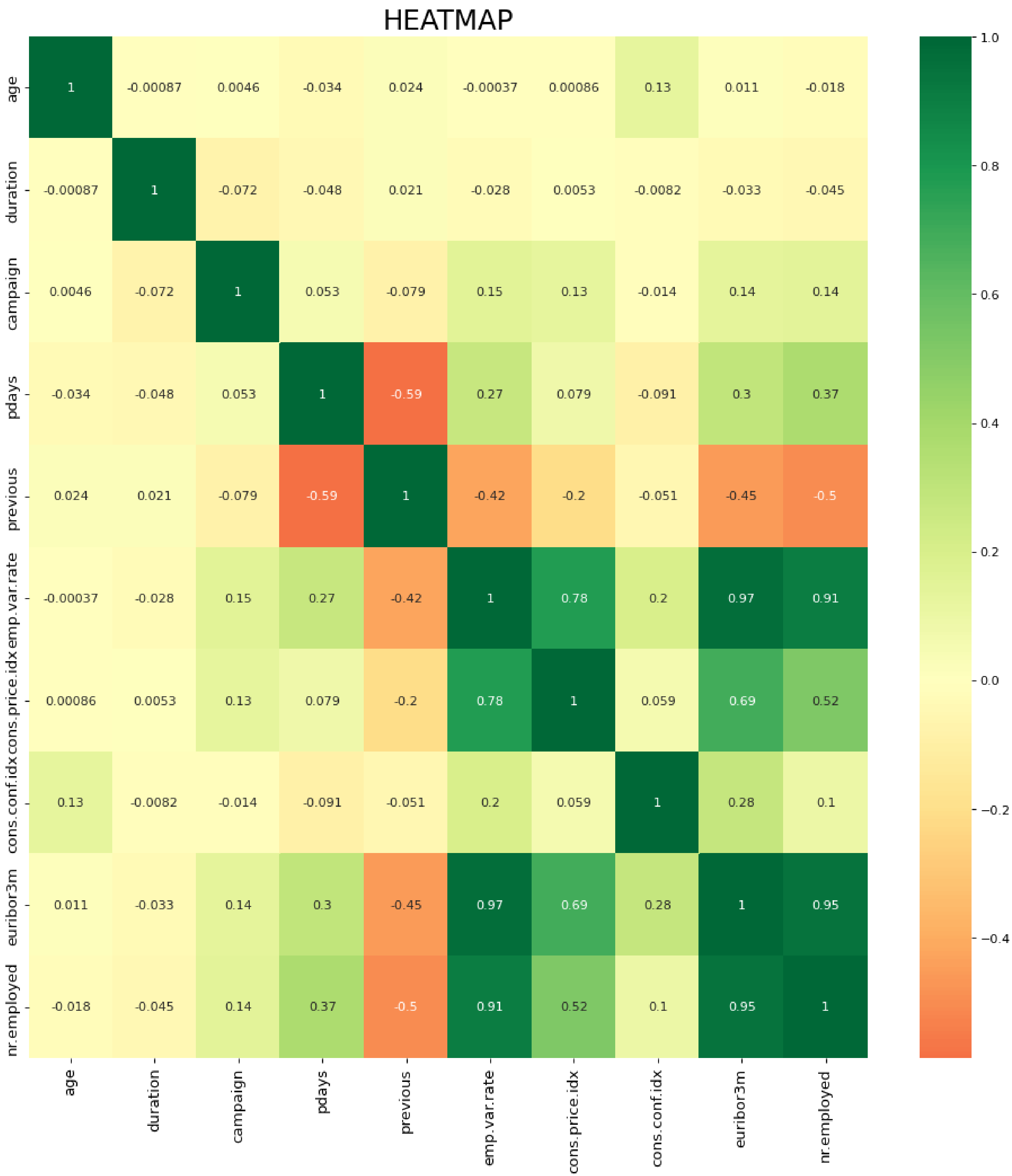
Exploratory Data Analysis for the dataset, provided with the following insights

- The subscription depends on continuous and discrete features such as age, job, education, housing loan, etc.
- Andrews curve was plotted for the given data to get a broader view regarding the correlations in the data. Following conclusion were drawn with it:
  - Highly overlapping suggest that there is a lot of noise in the current state of the dataset.

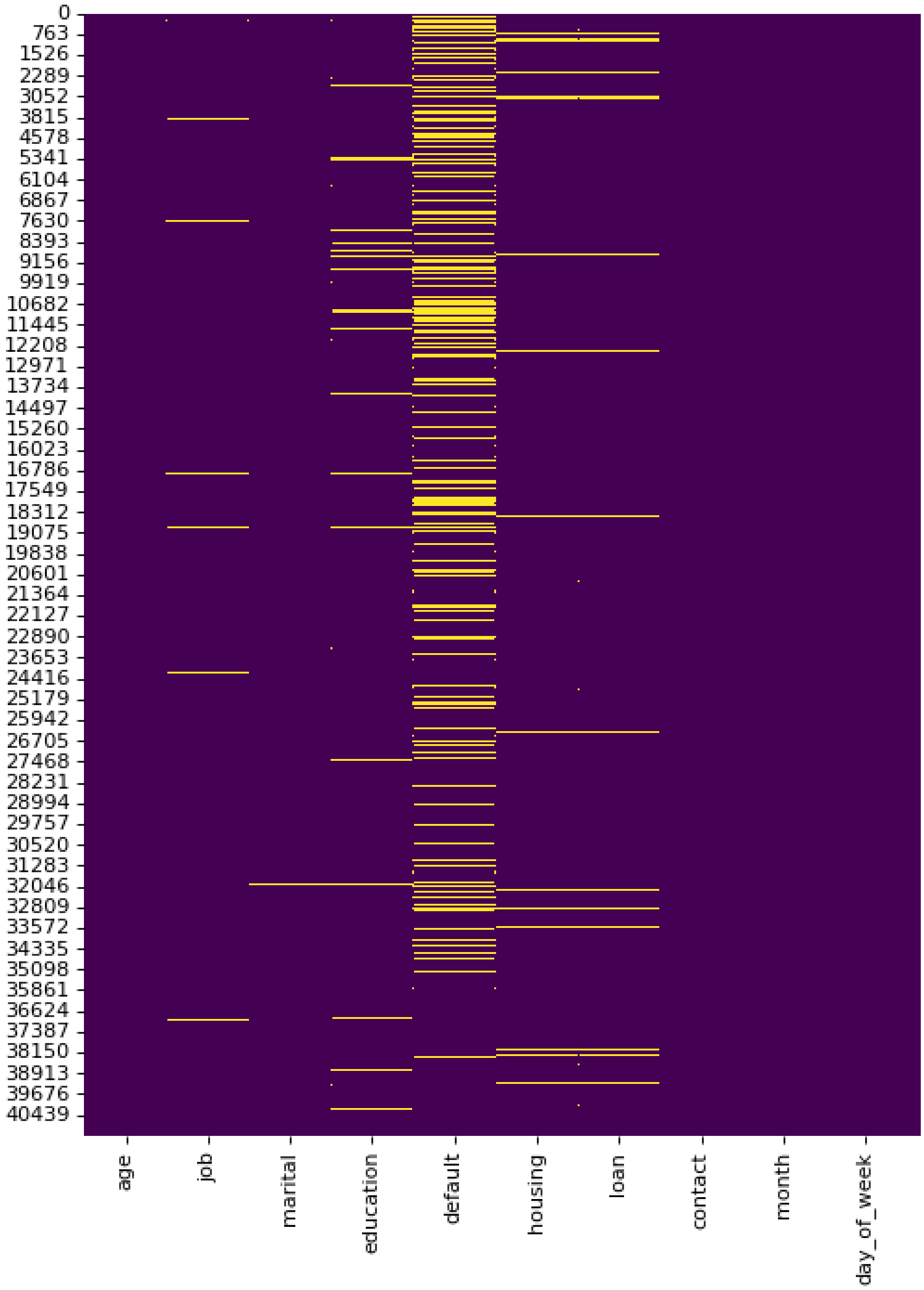
- Similar shape and size conclude that neither yes nor no is exclusively discriminated by the data and it cannot separate the two groups well.
- However, visible and distinguishable curves suggest that there is some degree of separation, and thus proper data cleaning, feature selection, and handling outliers would solve the underlying problems.



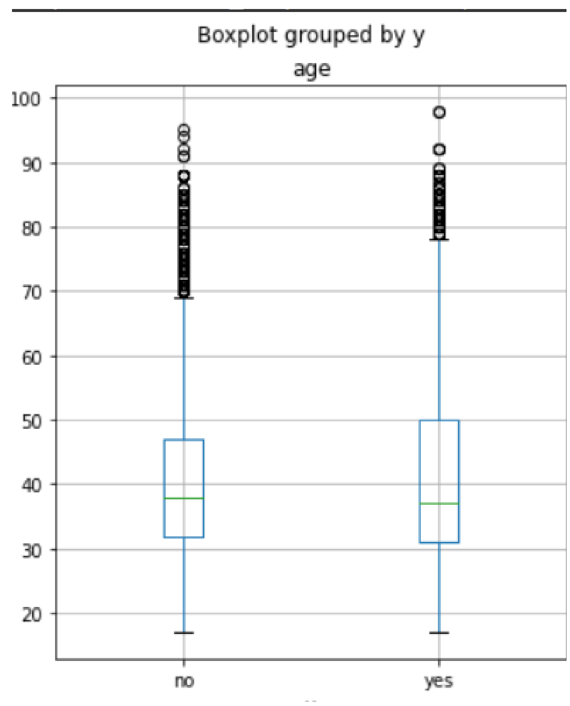
- Heatmap between all the continuous numeric variables shows there existed high correlations between (nr.employed , emp.var.rate) (nr.employed , euribor3m) (emp.var.rate , euribor3m) (emp.var.rate , cons.price.idx). These variables were concluded to be insignificant and were removed.


HEATMAP
age
duration
campaign
pdays
previous
emp.var.rate
cons.price.idx
cons.conf.idx
euribor3m
nr.employed
1 -0.00087 0.0046 -0.034 0.024 -0.00037 0.00086 0.13 0.011 -0.018
-0.00087 1 -0.072 -0.048 0.021 -0.028 0.0053 -0.0082 -0.033 -0.045
0.0046 -0.072 1 0.053 -0.079 0.15 0.13 -0.014 0.14 0.14
-0.034 -0.048 0.053 1 -0.59 0.27 0.079 -0.091 0.3 0.37
0.024 0.021 -0.079 -0.59 1 -0.42 -0.2 -0.051 -0.45 -0.5
-0.00037 -0.028 0.15 0.27 -0.42 1 0.78 0.2 0.97 0.91
0.00086 0.0053 0.13 0.079 -0.2 0.78 1 0.059 0.69 0.52
0.13 -0.0082 -0.014 -0.091 -0.051 0.2 0.059 1 0.28 0.1
0.011 -0.033 0.14 0.3 -0.45 0.97 0.69 0.28 1 0.95
-0.018 -0.045 0.14 0.37 -0.5 0.91 0.52 0.1 0.95 1
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4

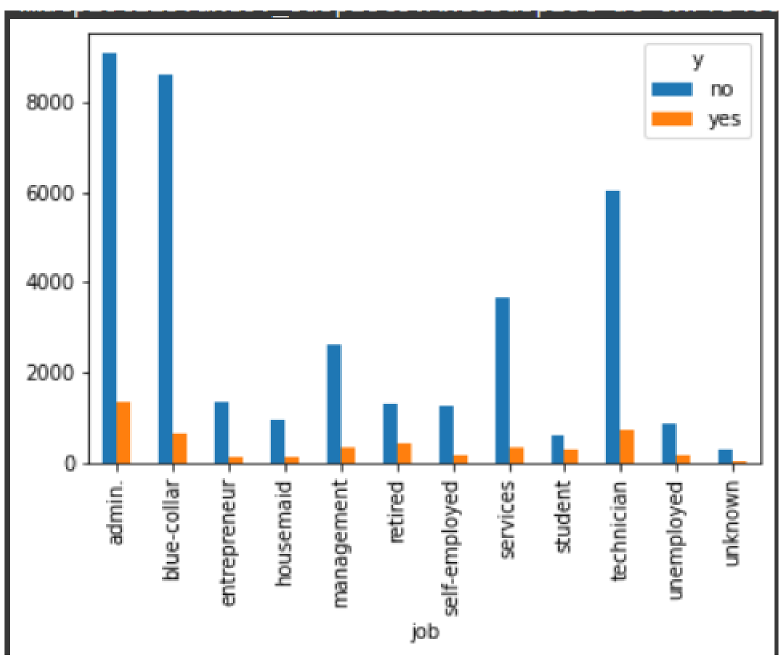
- Plotting heatmap of various variables shows that feature ‘Default’ could be removed to avoid any fake data in the analysis.



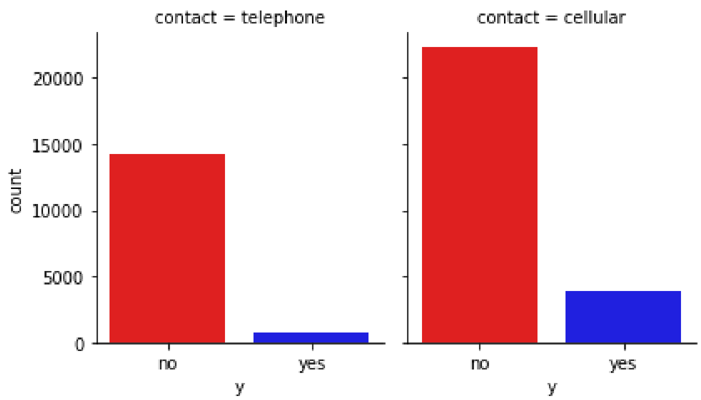
- While the client's age also dramatically affects the classification, around 50 percentile of the people in the age category of 35 to 45 are likely to subscribe to the term deposit.

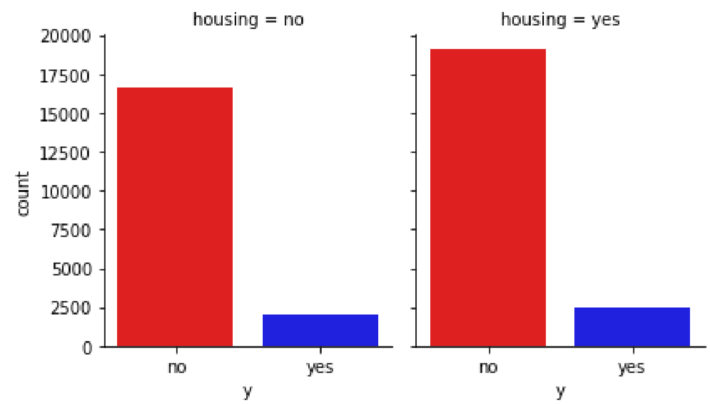


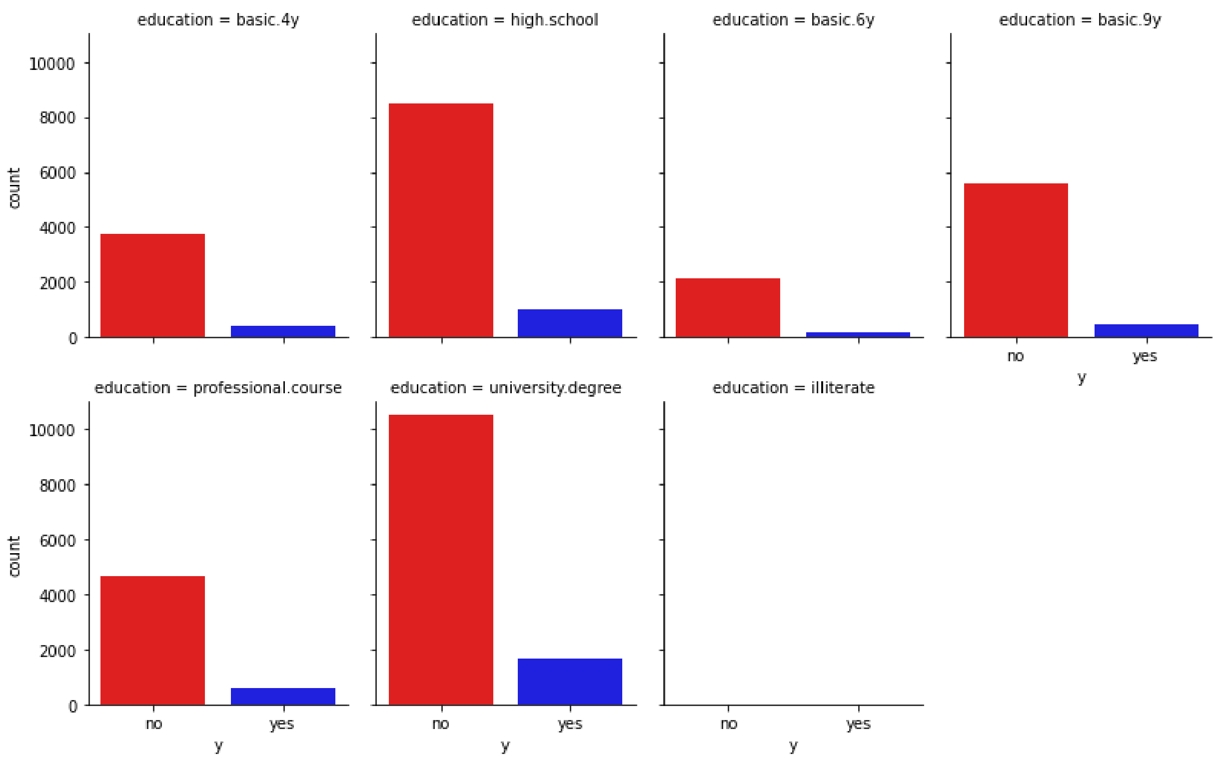
- From the plot, we have concluded that people in admin-related jobs have 12%, i.e., the highest chance of subscribing to a term deposit. Similarly, we found that singles with university degrees have a high chance of subscribing to a term deposit.



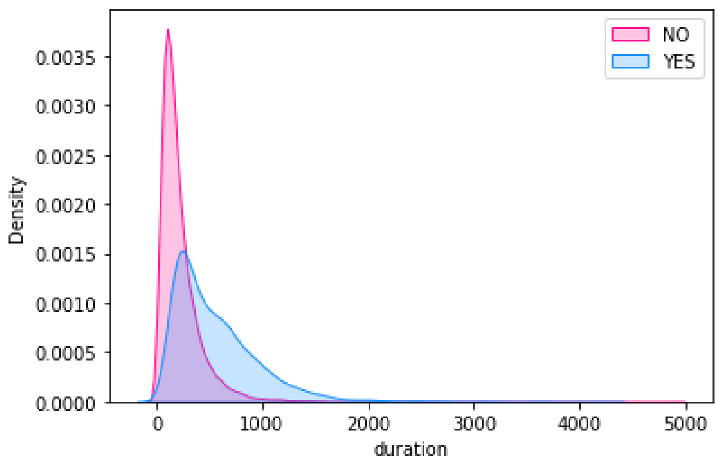
- Plotting count plots category-wise for most of the categorical features like marital, education, housing, loan, contact etc gave no significant anomaly and followed regular trends without any discrimination or partiality. Thus these features were directly fitted after encoding to the model.

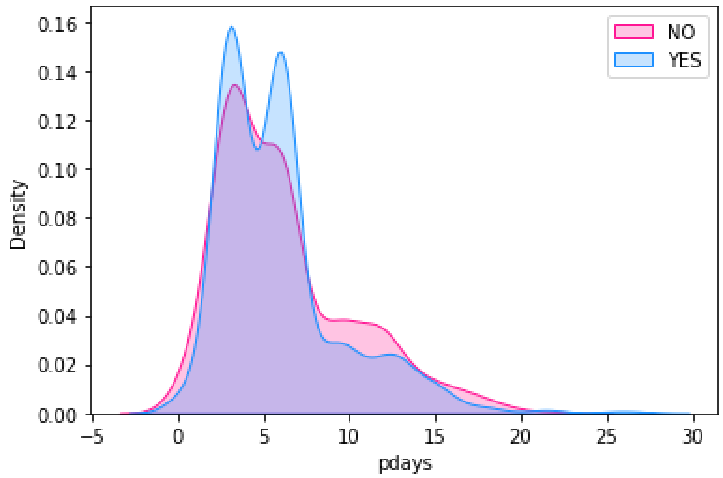


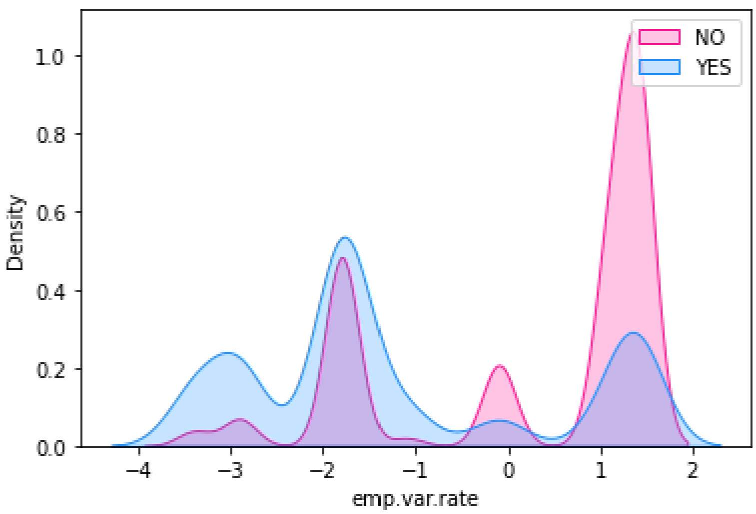


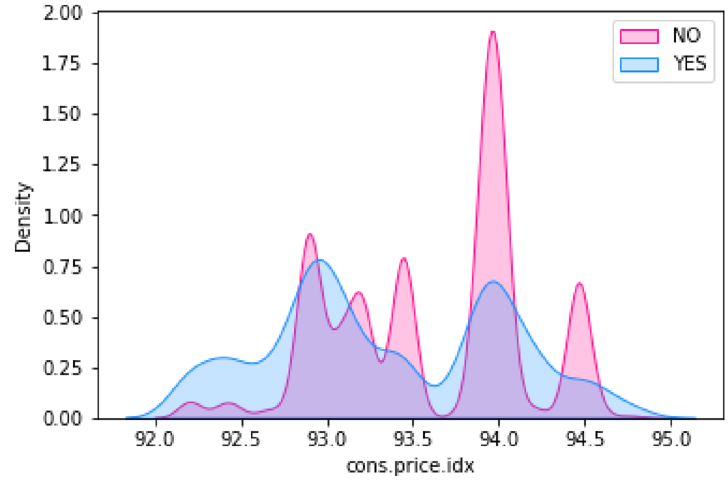


- On analysing continuous variables in the same way using density plots, symmetric and nearly normal trends were obtained for most of these features.

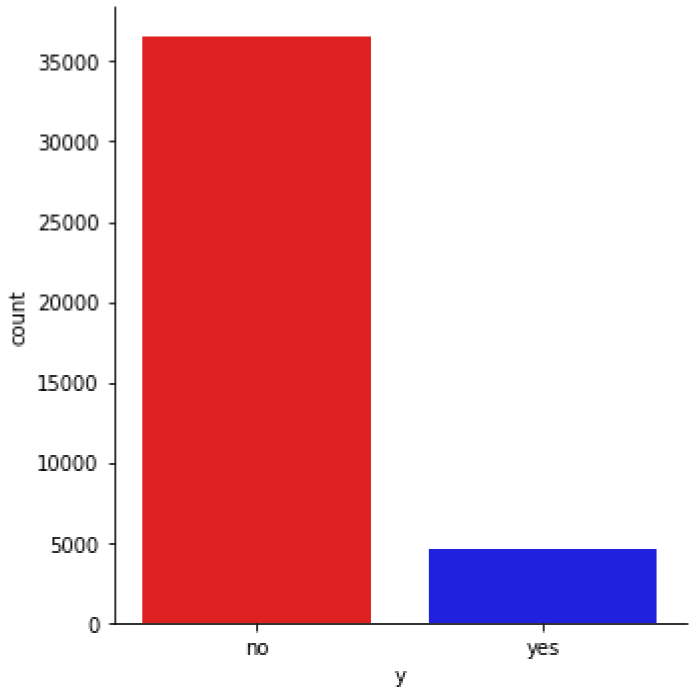






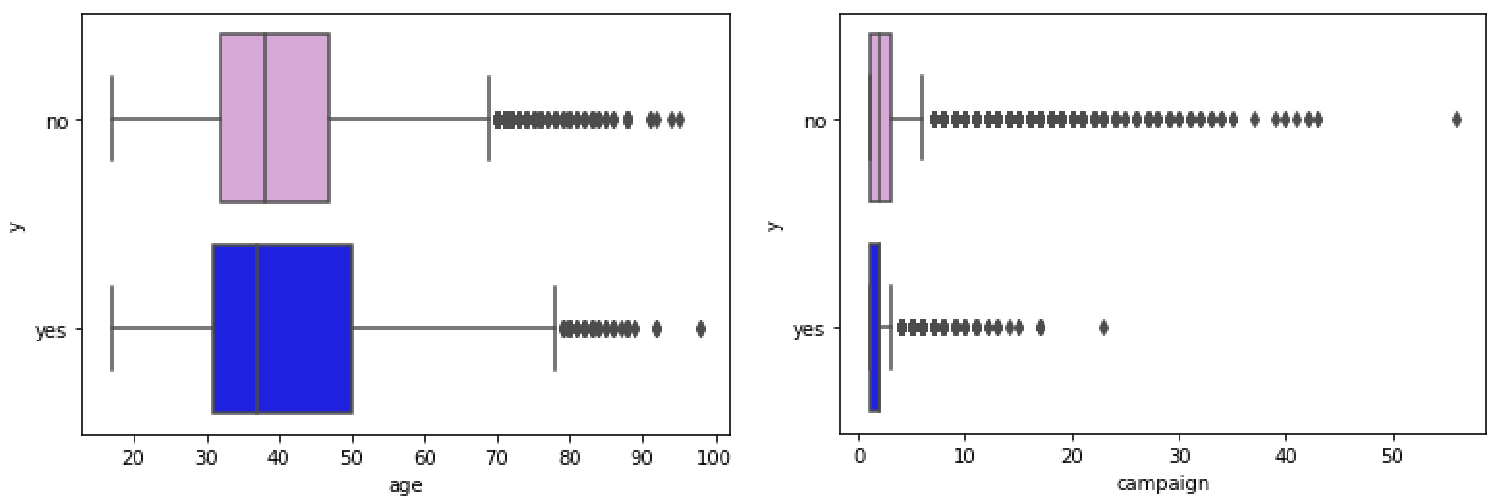


- The feature pdays had significant amount of 999(i.e. Unknown values). But since it was a significant factor, it was encoded with yes or no.
- ‘Poutcome’ was dropped due to high count of ‘nonexistent’ category.
- While checking the data about the day of the week, it doesn’t significantly affect the subscription. Still, clients subscribing in December, March, October, and September have very high chances of subscribing to the term deposit.
- On doing EDA, it was obtained that the data had high class imbalance, with NO being the dominating class. Since ML models doesnt work well with this distribution of data, The data was balanced using SMOTE algorithm, which also increased the size of data for better learning for the model.



Outlier analysis :

Features ‘age’ and ‘campaign’ were found to have significant amount of outliers. To counter this, age was scaled using StandardScaler and campaign by Log Transform, looking at their range of values.



From the exploratory data analysis, we have come to the above intuitions. After applying all the pre processing techniques thatw ere discussed above and scaling the required features, following dataset was obtained for further analysis.

	age	job	education	housing	loan	month	campaign	previous	emp.var.rate	cons.conf.idx	y	marital_enc
0	56	3	1	0	0	3	1	0	1.1	-36.4	0	0.607167
1	57	7	4	0	0	3	1	0	1.1	-36.4	0	0.607167
2	37	7	4	1	0	3	1	0	1.1	-36.4	0	0.607167
3	40	0	2	0	0	3	1	0	1.1	-36.4	0	0.607167
4	56	7	4	0	1	3	1	0	1.1	-36.4	0	0.607167
5	45	7	3	0	0	3	1	0	1.1	-36.4	0	0.607167
6	59	0	6	0	0	3	1	0	1.1	-36.4	0	0.607167
7	41	1	5	0	0	3	1	0	1.1	-36.4	0	0.607167
8	24	9	6	1	0	3	1	0	1.1	-36.4	0	0.280859
9	25	7	4	1	0	3	1	0	1.1	-36.4	0	0.280859

Further, we modeled the data using DNN, gradient boosting, KNN, and naive byes. We preprocessed the available data and filled in the unknown values using the mode, but further, we have tried implementing machine learning techniques to replace them more appropriately. Many features were found to be insignificant in the classification, like the day of the week. Similarly, the features which were very much correlated were removed not to overfit the data.

While using the KNN model had a good accuracy of 83.6% in predicting the subscription, but models like naive byes didn't fit the model very greatly with an accuracy of 71.3%

The data had a significant amount of outliers and quite a few missing values. Additionally, since the data was a mixture of numerical and categorical variables, it was expected that Tree based models would work well on the dataset.

On applying Tree-Based models, Random Forest, Gradient Boost and Decision Trees, using a pipeline along with GridSearchCV and 3-folds Cross validation, for tuning the hyper-parameters, it was found that Gradient Boost technique was working best with the data. Following were the results obtained :

```
# Initialzing the estimators
clf1 = RandomForestClassifier(random_state=42)
clf2 = DecisionTreeClassifier(random_state=42)
clf3 = GradientBoostingClassifier(random_state=42)

# Initiazing the hyperparameters for each dictionary

# Random Forest
param1 = {}
param1['classifier__n_estimators'] = [10, 50, 100, 250]
param1['classifier__max_depth'] = [5, 10, 20]
param1['classifier__class_weight'] = [None, {0:1,1:5}, {0:1,1:10}, {0:1,1:25}]
param1['classifier'] = [clf1]

# Decision Tree
param2 = {}
param2['classifier__max_depth'] = [5,10,25,None]
param2['classifier__min_samples_split'] = [2,5,10]
param2['classifier__class_weight'] = [None, {0:1,1:5}, {0:1,1:10}, {0:1,1:25}]
param2['classifier'] = [clf2]

# Gradient Boost
param3 = {}
param3['classifier__n_estimators'] = [10, 50, 100, 250]
param3['classifier__max_depth'] = [5, 10, 20]
param3['classifier'] = [clf3]

# Integrating pipeline
pipeline = Pipeline([('classifier', clf1)])
params = [param1, param2, param3]
```

```
Best Parameters : {'classifier': GradientBoostingClassifier(max_depth=10, n_estimators=250, random_state=42),
```

The auc-roc score obtained for these parameters in the test data was :

```
Best Score after Cross Validation : 0.9701343848506845
```

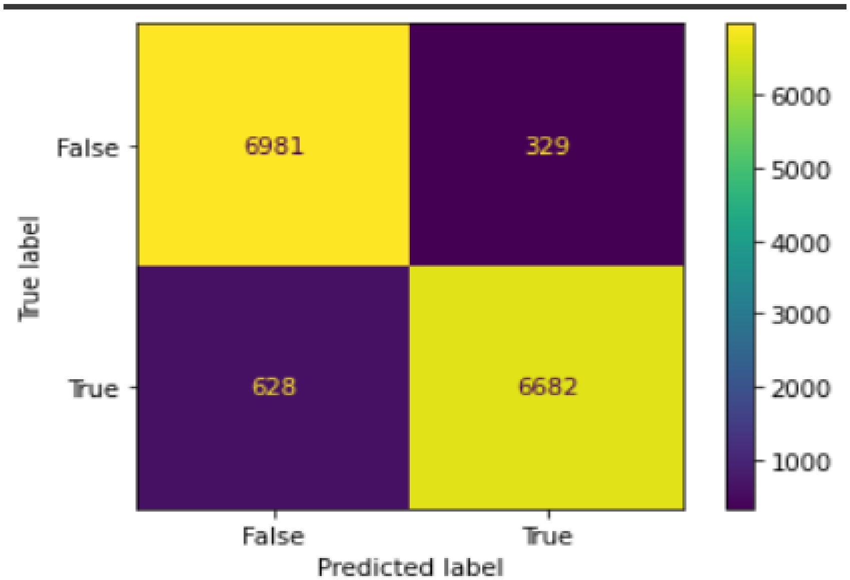
The test data was fit into model, along with the obtained hyper-parameters, gave highly accurate and satisfactory scores.

```
              precision    recall  f1-score   support

           0       0.92      0.95      0.94      7310
           1       0.95      0.91      0.93      7310

    accuracy                           0.93     14620
   macro avg       0.94      0.93      0.93     14620
weighted avg       0.94      0.93      0.93     14620
```

```
Test Accuracy: 0.9345417236662107
Test F1 Score: 0.9331750576077089
Test Precision: 0.9140902872777018
Test Recall: 0.9530737412637285
Test ROC AUC Score: 0.9352699498801228
```

Looking at training and testing scores, it was inferred that since the scores were close to each other and the test accuracy score is well below 97%(which is usually the threshold limit for overfitting model), the model had the best parameters possible that fits the dataset.

Following was the Confusion Matrix obtained :



The Classification Report shows that the model was appropriately trained, without any overfitting issues