Credit Card Approval Prediction

Submitted towards partial fulfilment of the criteria for the award of

Post Graduate Program in Data Science Engineering

Batch: PGP-DSE Jul 21' Online

Submitted By

Kunal Jangra (SXDTIQ9QT5)

Nevathithaa S (WZGL0OA7NY)

Nivedhitha Rajagopalan (5CTWL8RJNS)

Shivangi Bhaduri (1TLRPED3V4)

Sunil K S (QRPKWUB5SJ)

Mentored By

Animesh Tiwari

1. Industry Review

1.1 Background Research

The decision of approving a credit card or loan is majorly dependent on the personal and financial background of the applicant. Factors like, age, gender, income, employment status,credit history and other attributes all carry weight in the approval decision. Credit analysis involves the measure to investigate the probability of a third-party to pay back the loan to the bank on time and predict its default characteristic. Analysis focus on recognizing, assessing and reducing the financial or other risks that could lead to loss involved in the transaction. There are two basic risks: one is a business loss that results from not approving the good candidate, and the other is the financial loss that results from by approving the candidate who is at bad risk. It is very important to manage credit risk and handle challenges efficiently for credit decisions as it can have adverse effects on credit management. Therefore, evaluation of credit approval is significant before jumping to any granting decision.

1.2 Current Practice

Although a lot of methods have been used in banks across the nation followed by manual analysis to fill for the lack of efficient screening tool. Manually analyzing these applications is mundane, error-prone, and time-consuming. This also adds up to the economic cost of lending.

Current Algorithms that are used to decide the outcome of credit application vary from one bank to another and across sectors and geographies. However, there are high degrees of similarities in the attributes used to generate those algorithms. Most common tool of analysis is the logistical regression tool followed by manual screening. By using the historical data provided by the credit card applicants we can identify and predict the potential applicants by applying advance machine learning algorithms.

1.3 Literature Review

Wang, Ruoyu. (2021). AHP -Entropy Method Credit Risk Assessment Based on Python. 17-20.10.1109/ACCTCS52002.2021.00011. : This paper uses Bayesian network, Naive Bayes, Logistic regression Neural networks, and Radial basis function (RBF) to model the given data

Credit card customer churn prediction based on the RST and LS-SVM: This paper discusses Rough Set Theory (RST) and Least Squares Support Vector Machine (LS-SVM) techniques to model and predicts the data.

XGBoost Model and Its Application to Personal Credit Evaluation: This article investigates the application of the eXtreme Gradient Boosting (XGB) method to the credit evaluation problem based on big data. We first study the theoretical modeling of the credit classification problem using the XGB algorithm, and then we apply the XGB model to the personal loan scenario based on the open data set from Lending Club Platform in the USA. The empirical study shows that the XGB model has obvious advantages in both feature selection and classification performance compared to the logistic regression and the other three tree-based models.

Most studies used traditional statistical, machine learning, and deep learning techniques to detect credit card fraud and compared the results. However, as per the literature review, there is very little research work done to decide whether a customer is to be issued a credit card or not based on their information. Therefore, this study aims to support the decision-makers of whether a customer is to be issued a credit card or not using supervised machine learning techniques tools like Logistic Regression, k-Nearest Neighbors, Decision Trees, Support Vector Machine, Naive Bayes, and XGBoost Classifier.

2. Dataset and Domain

2.1 Project Justification

2.1.1 Problem Statement

The goal of a credit scoring model is to classify credit applicants into two classes: the "good credit" class that is liable to reimburse the financial obligation and the "bad credit" class that should be denied credit due to the high probability of defaulting on the financial obligation.

Credit score cards are a common risk control method in the financial industry. It uses personal information and data (historical data) submitted by credit card applicants to predict the probability of future defaults and credit card borrowings. The bank can decide whether to issue a credit card to the applicant or not. Credit scores are based on historical data and it can quantify the magnitude of risk.

2.1.2 Complexity Involved

We tried to move ahead by asking the below-mentioned questions.

1. Which model should we pick? Are the features that affect the credit card approval decision process correlated with each other?
2. Is a customer's debt background too validated along with credit score while approving their credit card application?
3. Is a customer's income as well as credit score is taken into account to determine whether a credit card must be given or not?
4. Is there any sort of discrimination within the approval process? If so, then is it possible to eliminate the manual intervention via ML techniques?

The dataset has a mixture of numerical and non-numerical features. Also, it is unbalanced data. This can be fixed with some data preprocessing. Apart from preprocessing like handling missing values and scaling the data, the choice of the model which is efficient as well as industrially acceptable has been a major challenge in the process.

What we need is something that will look like correlation, but will work with categorical values, or more formally, we're looking for a measure of association between two categorical features.

2.1.3 Project Outcome

Commercial: Adequate and timely screening can reduce friction in the economy by providing consumers convenient and secure access to their funds while reducing cash and check to handle for merchants and expanding the pool of customers who are guaranteed to pay. When credit grows, consumers can borrow and spend more, and enterprises can borrow and invest more. A rise in consumption and investments creates jobs and leads to a growth of both income and profit. Furthermore, the expansion of credit influences also the price of assets, thereby increasing their net value. We also gained some key business insights by which we can direct our sales and marketing teams to mold their approaches to attracting customers. Hence, we influenced a business decision-making process and acted as key in directing business processes. Reducing efforts and costs associated with the credit card Application approval process by deploying a machine learning model to select valid candidates

Academic: As we all know, no model is perfect so our attempt at building this model and analysis can be further built upon. The real-world application of predictive analytics and ML techniques will help in further research and model building. Even complex problems across domains can use the research gaps and advance academic research with the aim of efficient solution-finding. The Project nonetheless has tried to work upon the already existing methodological gaps and has tried to build a more efficient model than those under usage.

Social: While the use of credit can act as a catalyst for consumer spending and economic growth, a disproportionately large amount of credit use can introduce risk into the economy. The inability of consumers to pay back their debt can have serious repercussions on the personal and national levels. By developing a suitable filtering model, we can stop the vicious cycle of credit card indebtedness and identify suitable holders who can handle debts responsibly.

2.2 Data Dictionary

There are two datasets involved. One has the personal details of each applicant who has applied for the credit card (example; gender, age, income, years employed, etc.., – application_record.csv data). The other data set contains the details of each applicants' loan payment status for each month (credit_record.csv). The common feature for the two data sets is the Applicant ID. The two data set has to be merged for further processing.

Application_record dataset:

Feature	Explanation	Remarks
id	Client number	
gender	Gender	
own_car	Is there a car	
own_house	Is there a property	
count_children	Number of children	
income	Annual income	
income_type	Income category	
education	Education level	
family_status	Marital status	
housing_type	Way of living	
days_birth	Birthday	Birthday (Count backwards from current day (0), -1 means yesterday)
days_employed	Start date of employment	Count backwards from current day(0). If positive, it means the person currently unemployed.
flag_mobile	Is there a mobile phone	
flag_work_phone	Is there a work phone	
flag_phone	Is there a phone	
flag_email	Is there an email	
occupation_type	Occupation	
count_fam_members	Family size	

Credit_record dataset

Feature name	Explanation	Remarks
id	Client number	Client number
months_balance	Record month	The month of the extracted data is the starting point, backwards, 0 is the current month, -1 is the previous month, and so on
status	Status	0: 1-29 days past due 1: 30-59 days past due 2: 60-89 days overdue 3: 90-119 days overdue 4: 120-149 days overdue 5: Overdue or bad debts, write-offs for more than 150 days C: paid off that month X: No loan for the month

2.3 Variable categorization (count of numeric and categorical)

2.3.1 Application_record dataset

The total number of records in the Application_record dataset is 4,38,557

2.3.1.1 Numerical

Features	count	Unique	mean	std	min	25%	50%	75%	max
id	438557	438510	6022176	571637	5E+06	5609375	6047745	6456971	7999952
count_children	438557	12	0.42739	0.72488	0	0	0	1	19
income	438557	866	187524	110087	26100	121500	160781	225000	6750000
days_birth	438557	16379	-15998	4185.03	-25201	-19483	-15630	-12514	-7489
days_employed	438557	9406	60563.7	138768	-17531	-3103	-1467	-371	365243
flag_mobile	438557	1	1	0	1	1	1	1	1
flag_work_phone	438557	2	0.20613	0.40453	0	0	0	0	1
flag_phone	438557	2	0.28777	0.45272	0	0	0	1	1
flag_email	438557	2	0.10821	0.31064	0	0	0	0	1
count_fam_mem	438557	13	2.19447	0.89721	1	2	2	3	20

**There are no Null values in the numerical column

2.3.1.2 Categorical

Features	count	unique	Null Percentage
gender	438557	2	0
own_car	438557	2	0
own_house	438557	2	0
family_status	438557	5	0
housing_type	438557	6	0
occupation_type	438557	5	0
income_type	438557	5	0
occupation_type	134203	19	30.6

**There exist null values in one categorical column (occupation_type). The null value percentage is 30.6%

2.3.2 Credit_record dataset

The total number of records in the Credit_record dataset is 10,48,576

2.3.2.1 Column Details

Column_Name	Data_type	Unique_count	Null_count	Null_Perc
id	int64	45985	0	0
months_balance	int64	61	0	0
status	object	8	0	0

2.4 Pre Processing Data Analysis

2.4.1 Dataset definition

Application_record dataset:

- The features 'gender', 'own_car', 'own_house', 'income_type', 'education', 'family_status', 'housing_type', 'occupation_type' are in object datatypes as per the data definition.

- The variables 'id', 'count_children', 'income', 'days_birth', 'days_employed', 'flag_mobile', 'flag_work_phone', 'flag_phone', 'flag_email' are numerical as per data defined.
- The features 'id', 'own_car', 'count', 'income_type', 'education', 'family_status', 'housing_type', 'occupation_type' are in object datatypes as per the data definition.
- The variable 'count_fam_mem' according to the data definition, "count_fam_mem" is a categorical variable, which is wrongly interpreted as 'float64', so we will convert these variables data type to 'object'.
- There are no wrong entries like symbol -,?,#,* found in the dataset

**The datatypes in credit_record are defined as per the data definition

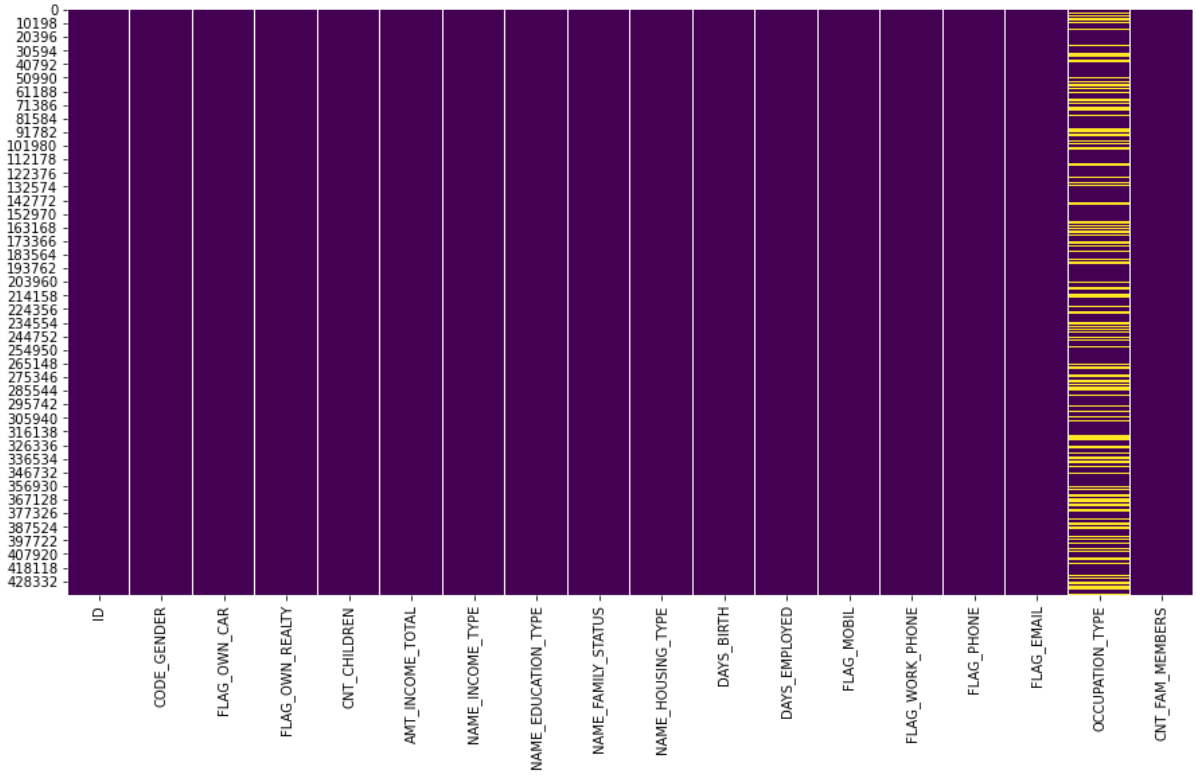
2.4.2 Checking duplicate values

On checking the whole data set there were no duplicate observations found. But on checking with the id column (applicants' id) there exist 47 duplicate ids. Those were removed from the dataset using drop.duplicates method in the application_record dataset. The id attribute is considered to be the unique column

There are no duplicates found in the credit_record dataset

2.4.3 Missing value treatment

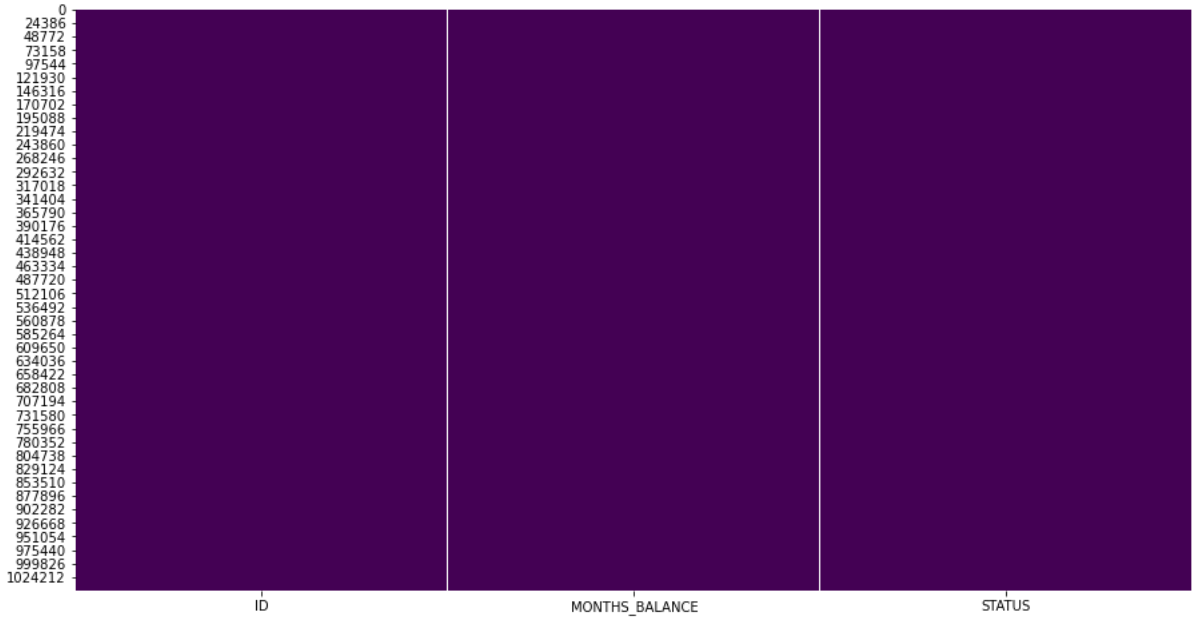
In the application_record dataset, only the occupation_type column contains missing values of about 134193 records (30% of the total records). As the null value percentage is huge, the null values are imputed based on the days_employed column.



Condition 1: Firstly the null values in the occupation_type are imputed as “Others” category

Condition 2: The applicant is considered to be “Un Employed”, wherever the days_employed holds positive value as per the data definition. So if the occupation_type is imputed again with “Un Employed” if above considered condition satisfies

There are no null values present in the credit_record dataset.



2.4.4 Creating target variable and Merging two Datasets

On analysing with the application_record and credit_record dataset the id feature is common in both datasets. So we are merging with the below insights;

The credit_record dataset has ‘status’ column which has the status of the applicant’s loan due for each month. The status column has below values along with description;

Values	Description
0	1-29 days past due
1	30-59 days past due
2	60-89 days overdue
3	90-119 days overdue
4	120-149 days overdue
5	Overdue or bad debts
C	paid off that month
X	No loan for the month

The columns are added based on the below condition;

Good_Debt – The applicants with loan due status ‘C’ and ‘X are considered as Good Debts

Bad_Debt – The applicants with loan due status other than ‘C’ and ‘X are considered as Bad Debts

The credit_record dataset is grouped based on the ‘id’ column, with number of Good_debt and Bad_debt for each applicants. The Target variable (status) is created based on the condition;

If the number of Good_debts are greater than the Bad_debts, then the status column is marked as 1 (considered as approved) else it is marked as 0 (declined)

Both the application_record and credit_record are merged with id column. The total number of observation of combined final dataset holds 36,457 records and 23 features

2.4.5 Target Variable Description

status (target)	Description
0	Applicants with Bad Credit Worthiness (Credit Card not approved)
1	Applicants with Good Credit Worthiness(Credit Card Approved)

2.4.6 Alternate sources of data

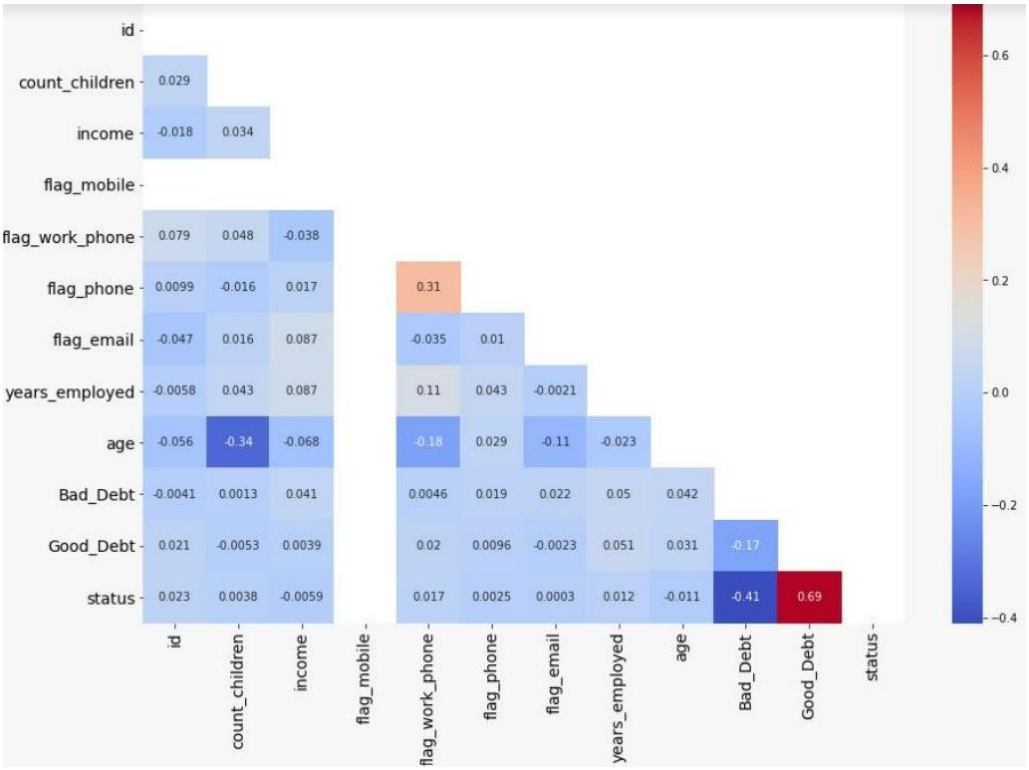
The days_birth and days_employed are column values was present in number days format with negative values. i.e., Count backwards from current day (0), -1 means yesterday. So both the fields are converted into years format. Also as per the description of days_employed, if it holds positive values then it is marked as 0 (The person is un employed)

Original_column	Derived_column
days_birth	age (in years)
days_employed	years_employed

2.4.7 Dropping insignificant features

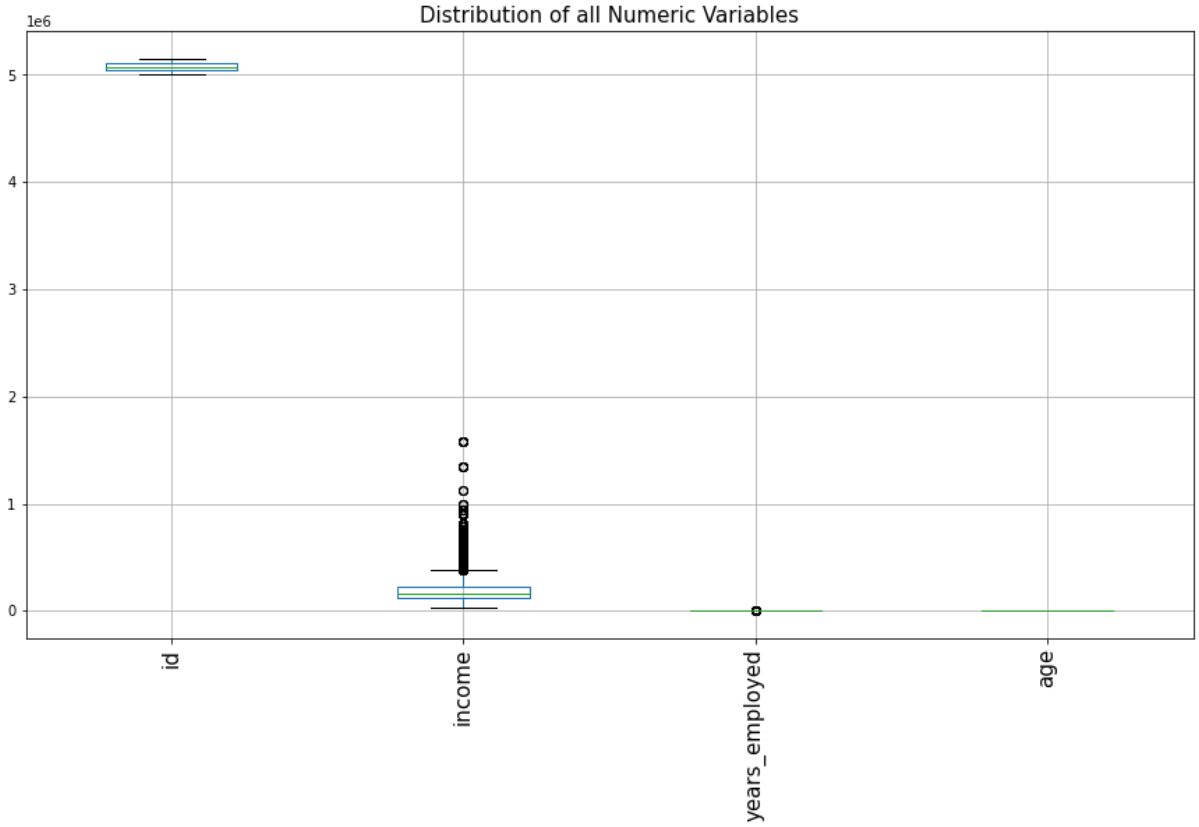
Based on the data analysis we found the below features are redundant in predicting the credit worthiness. So flag_mobile, flag_work_phone, flag_phone, flag_email, count_children, days_birth, days_employed. Also the Good_debt and Bad_debt columns are dropped as the target variable (status) is derived from that.

Numerical feature correlation plot before dropping the columns



2.4.8 Outlier checking and treatment

There are three numerical columns income, age, years_employed. Using IQR method outliers were detected. All three columns have 4% outliers of the total data. So the outliers are imputed using IQR treatment with upper and limit values



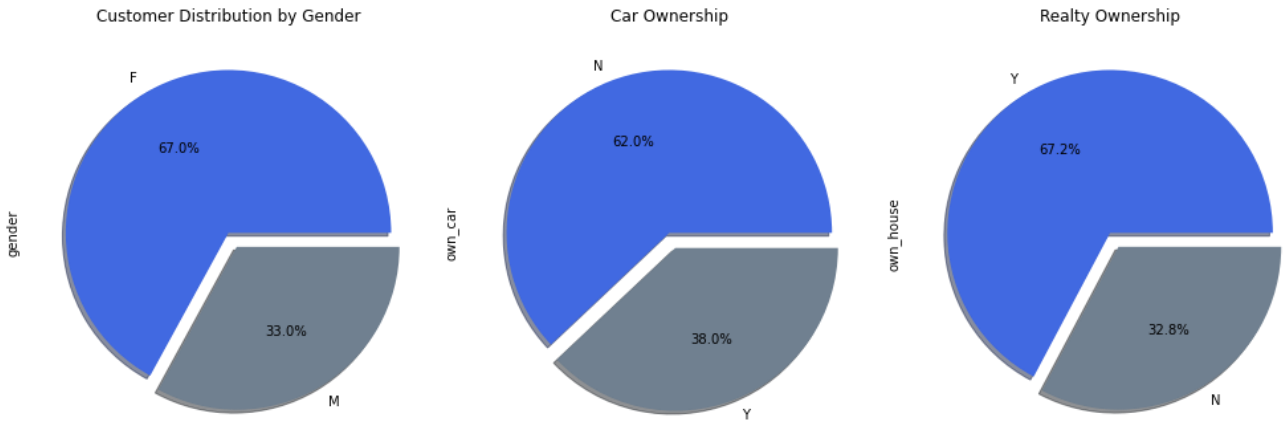
3. Exploratory Data Analysis

3.1 Univariate Analysis

In Univariate Analysis, we are exploring each feature separately using relevant plots to extract some insights about the data

3.1.1 Categorical Feature

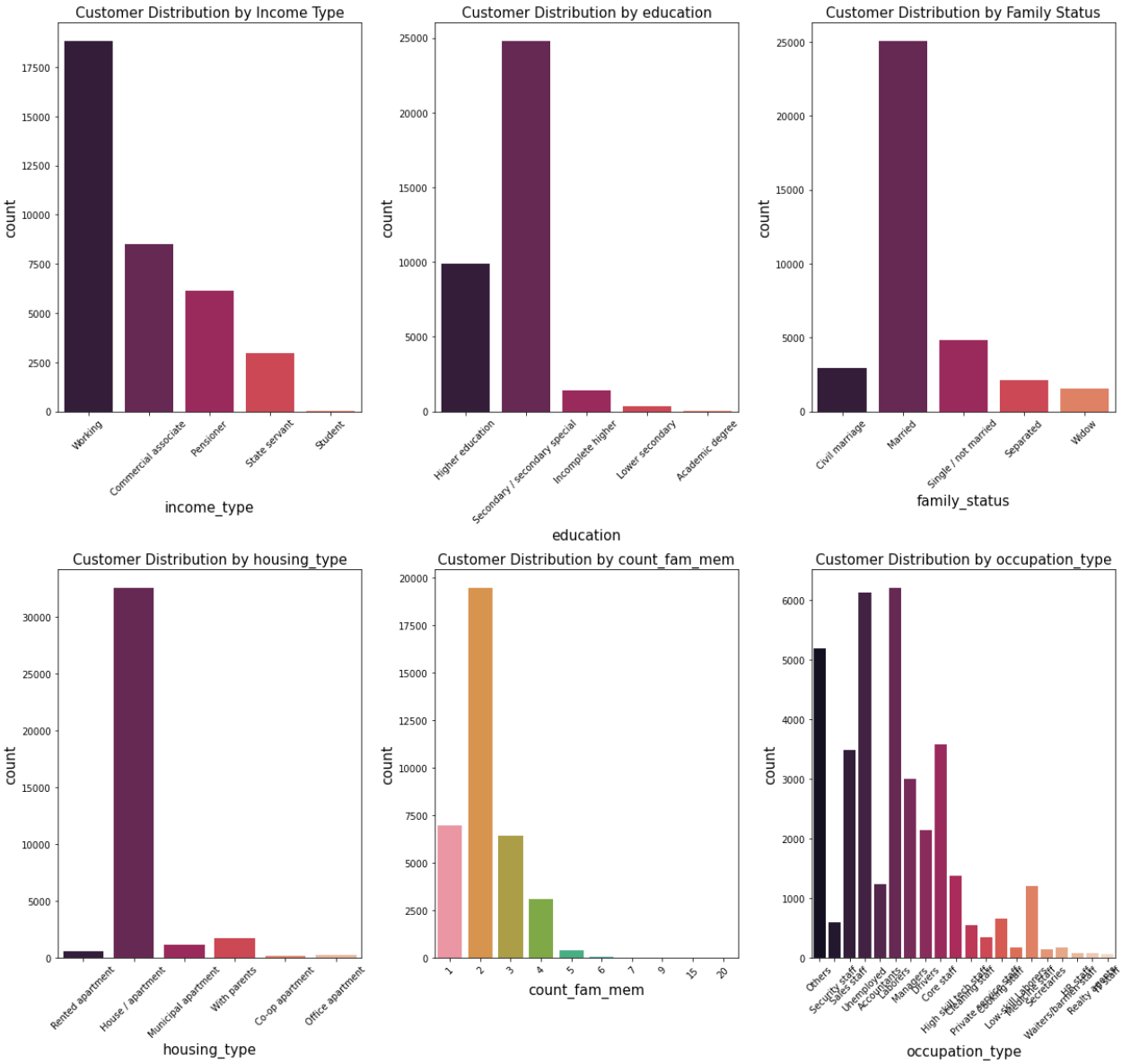
- **Gender Own_car and own_house**



Inference:

- The number of male applicants is higher than the number of female customers
- The number of female applicants is more than the number of male applicants.
- The percentage of female applicants is 67% and the male is 33%
- The count is nearly 12k for males and 24.5k for female applicants
- Most of the applicants does not own a car, only a few own the car
- Most of the applicants have own house

- **income_type, education, family_status, housing_type, count_fam_mem, occupation_type**



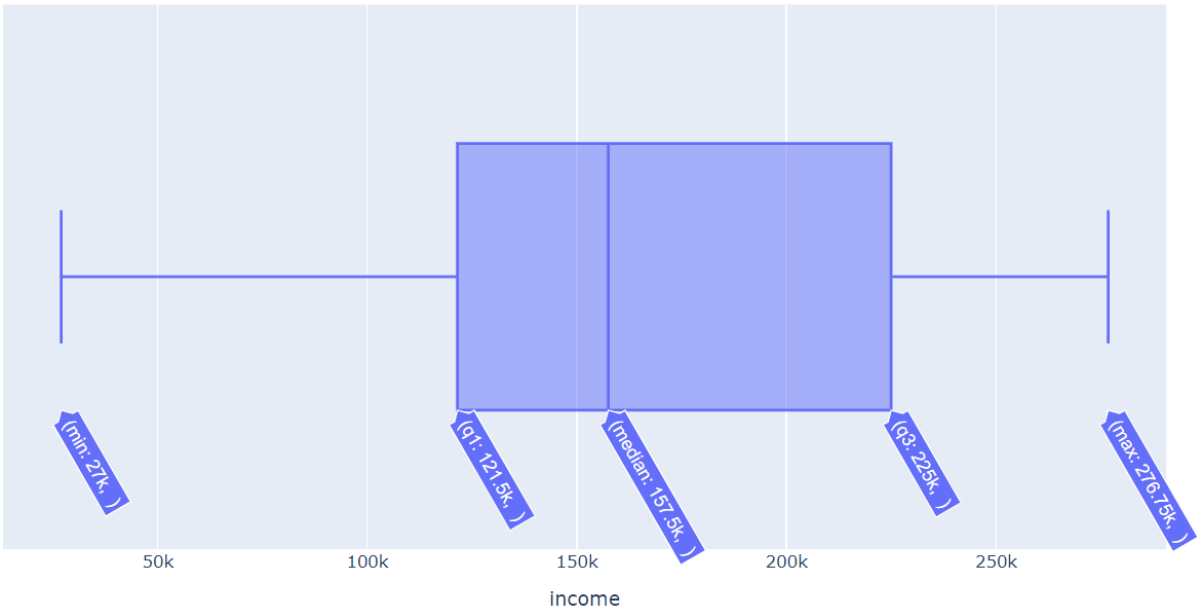
Inference:

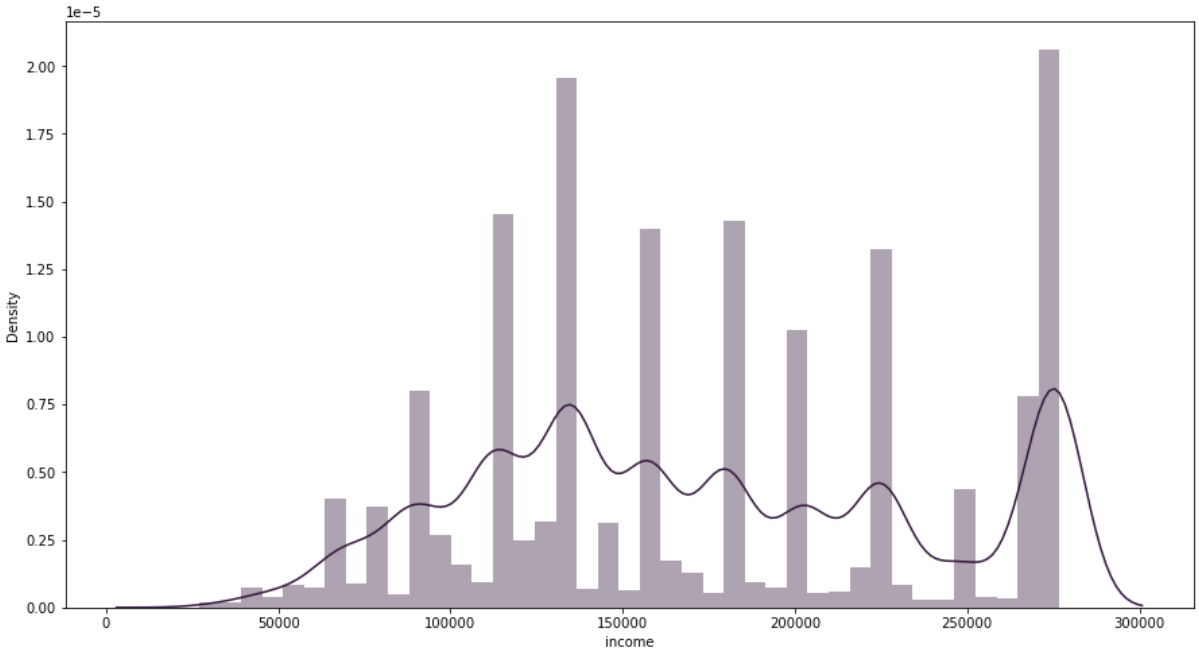
- The working professionals' **income_type** of the are high in comparison to other category types (Working, Commercial associate, State Servant). Whereas the State Servant professionals are low in number
- The maximum number of applicants has the **education** qualification up to Secondary level when compared to other education types
- The majority of the applicants' **family_status** is married
- The majority of the applicants own a House/Apartment **(Housing_type)**

- The family count of most of the applicants are 2 followed by 1,3,4 and 5 **(count_fam_mem).**
- For most of the applicants, the Occupation type is Laborers and unknown category (Others). Also, more number of applicants are Un-employed

3.1.2 Numerical Feature

- **income**

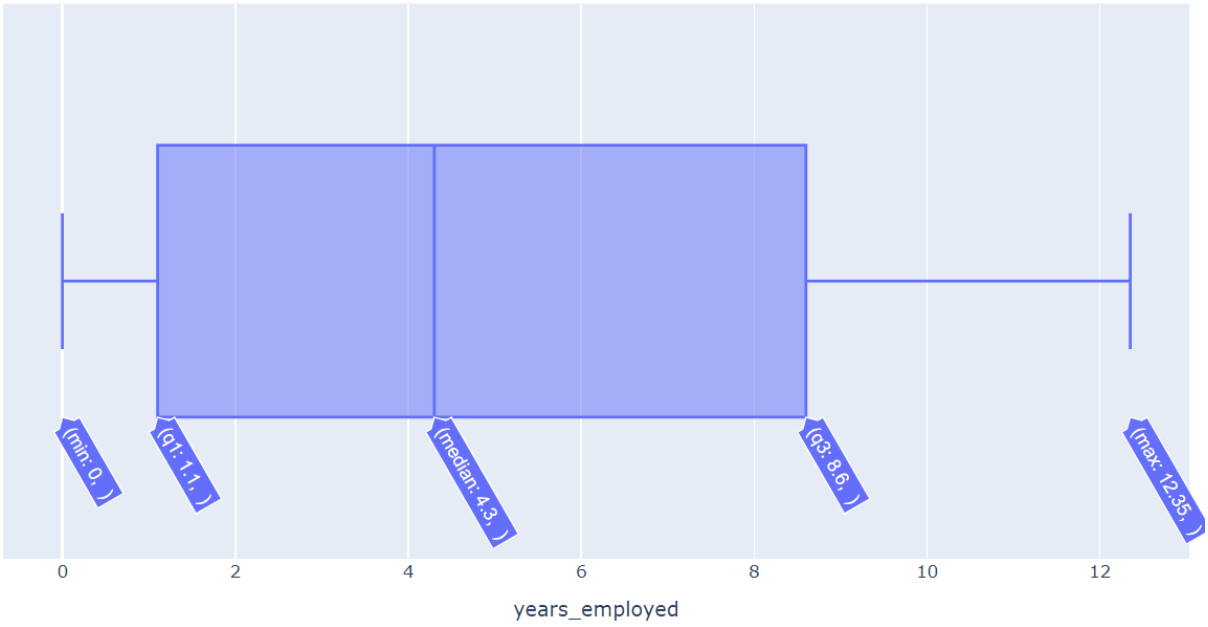


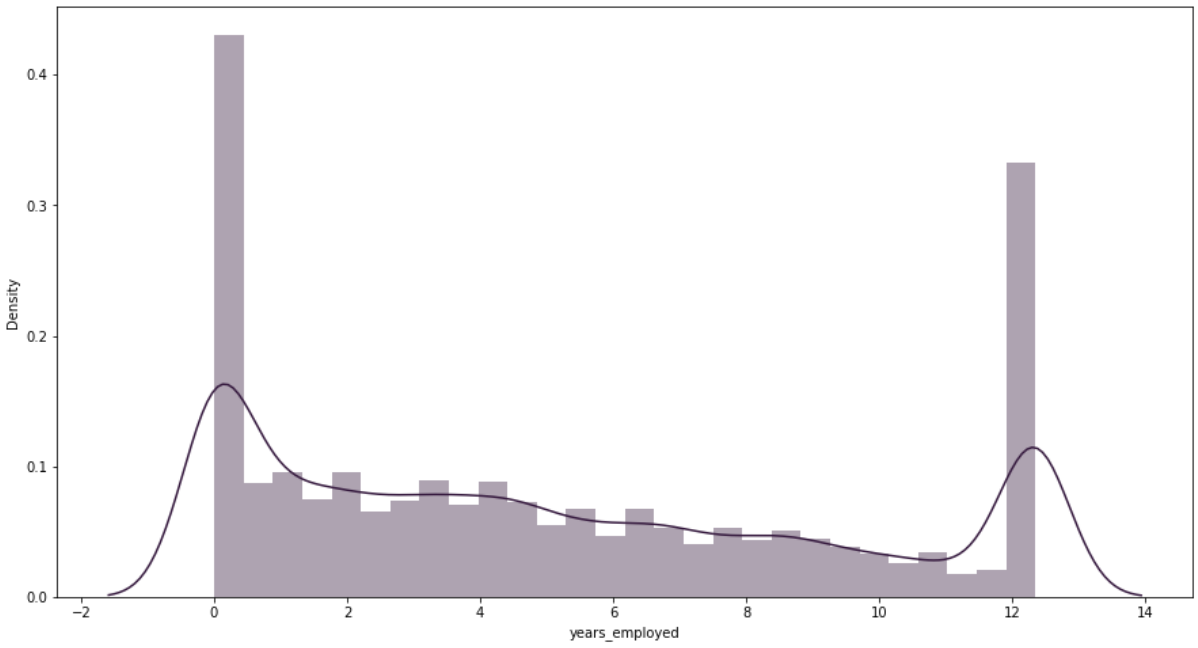


Inference:

- The distribution of income is slightly skewed towards the right i.e. “Positive Skew”
- The income of the applicants ranges from 27k to 276.75k.
- The median income of the applicants is 157.5k and the mean income of the applicants is 173.2k

- **years_employed**

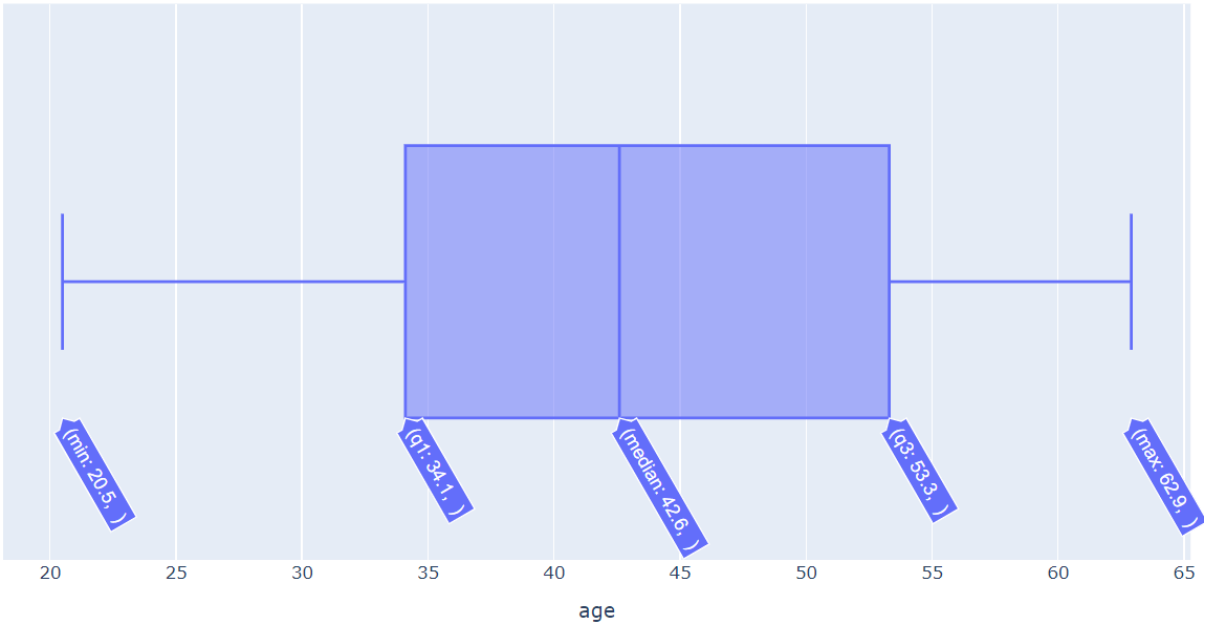


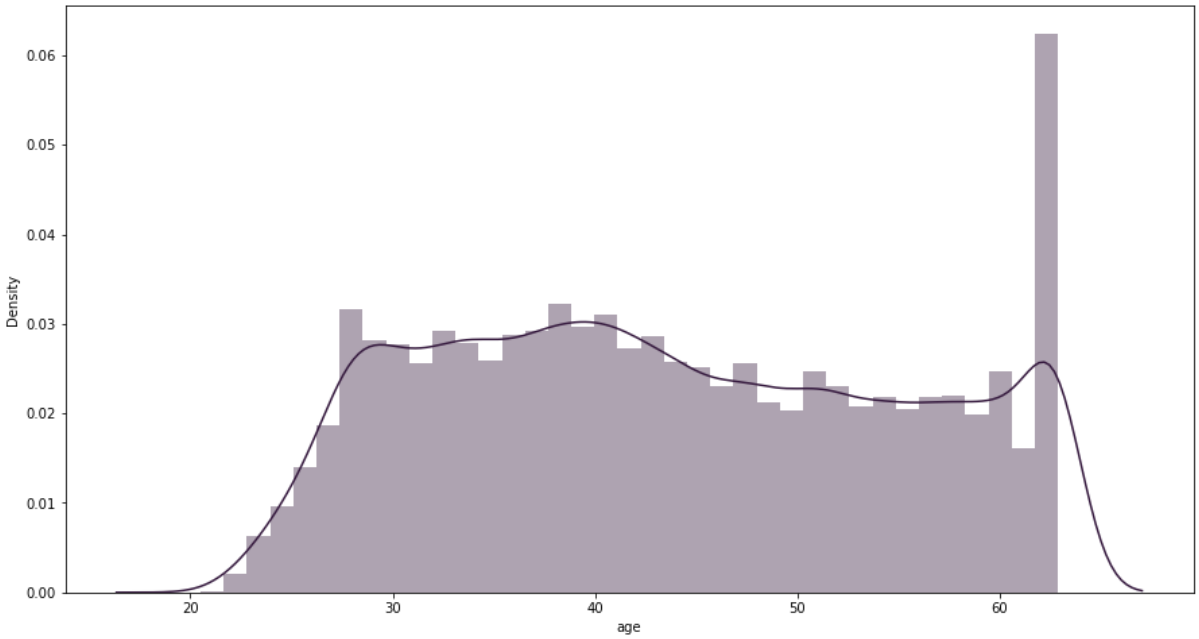


Inference:

- **The distribution of Employment years is highly skewed towards the right**
  - ➢ The Employment years of the applicants range from 0 to 12 years.

- **age**





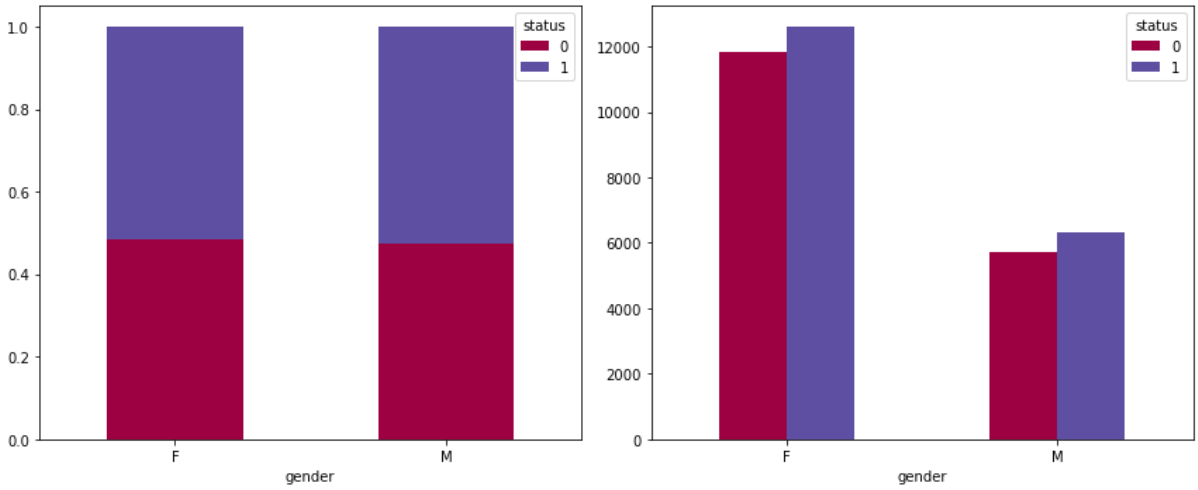
Inference:

- The Age distribution is slightly skewed towards the right i.e. “Positive Skew”
- The distribution of age ranges from 20 years to 62 years among several distinct groups.
- The mean and median of the applicants' age is nearly 42 years

3.2 Bivariate Analysis (Between the feature and target variable)

3.2.1 Analyzing the relationship between target and categorical variable

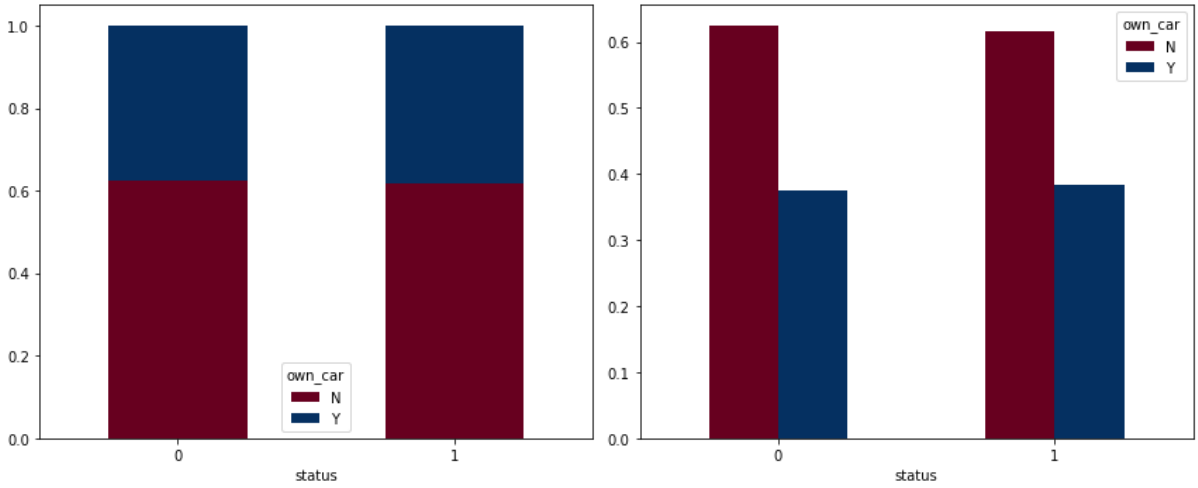
- **gender**



Inference:

- The count of male and female applicants has an equal proportion on creditworthiness, so there is no significant relation between gender and credit approval('status', target variable).

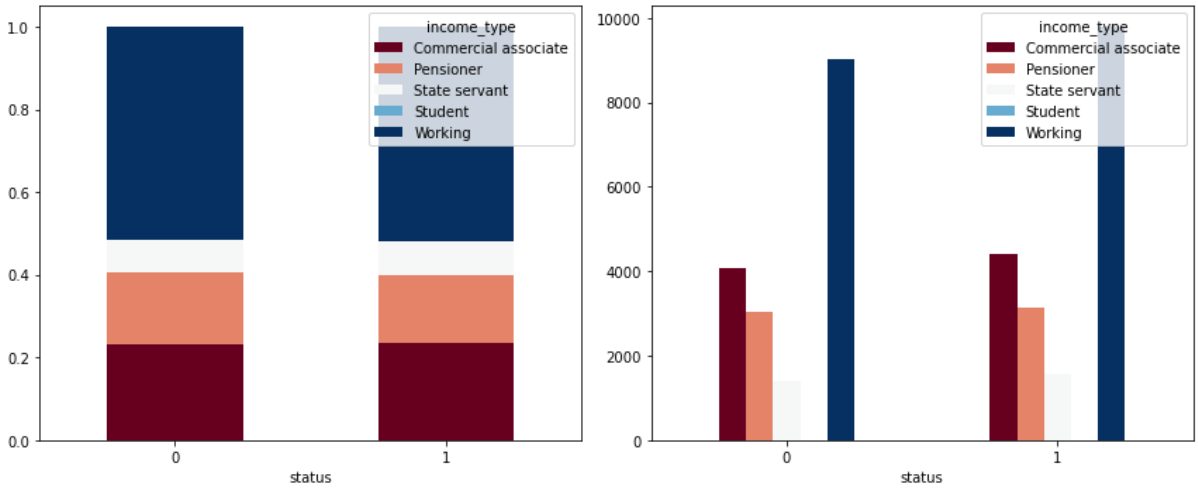
- **Own_car:**



Inference:

➢ The count of applicants who owns a car are having an equal proportion on their creditworthiness, so there is no significant relation between applicants with their own car and credit approval ('status', target variable).

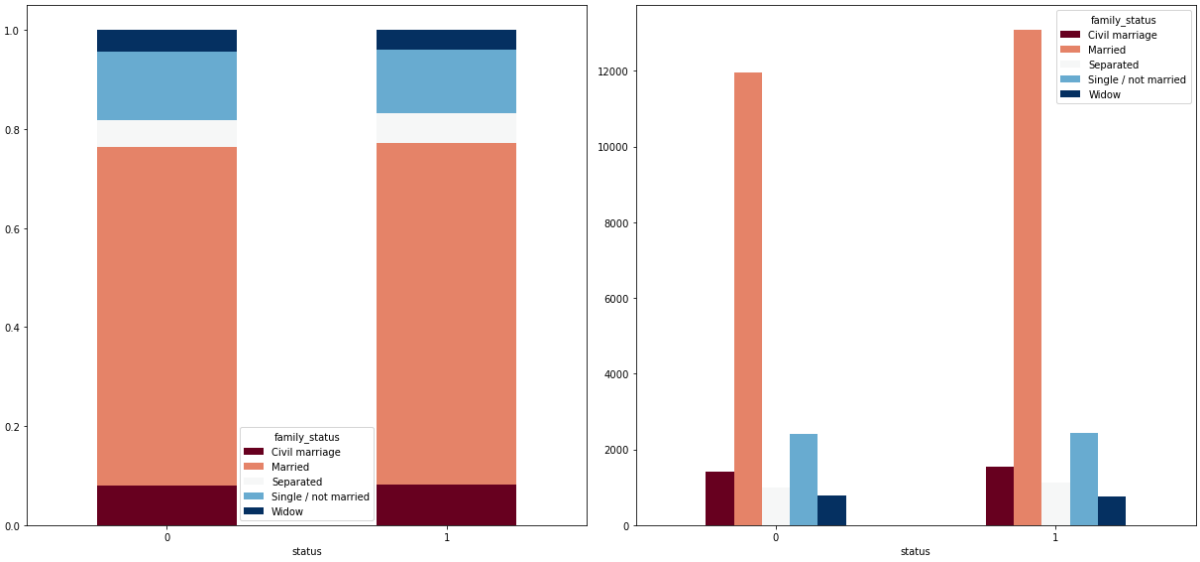
- **Income_type:**



- **Inference:**

➢ The working professional's income type has more creditworthiness,so there may be a significant relationship between applicants with their own house and credit approval ('status', target variable).

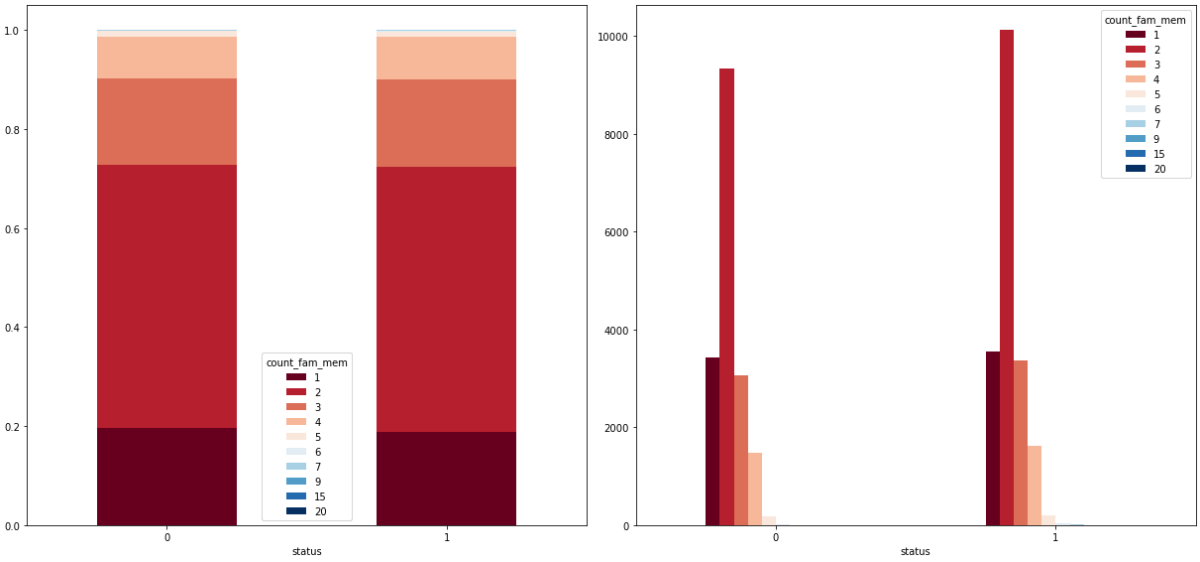
- **Family_status:**



Inference:

- The married applicants are high in count compared to other categories so there may be a significant relationship between family status and creditworthiness.

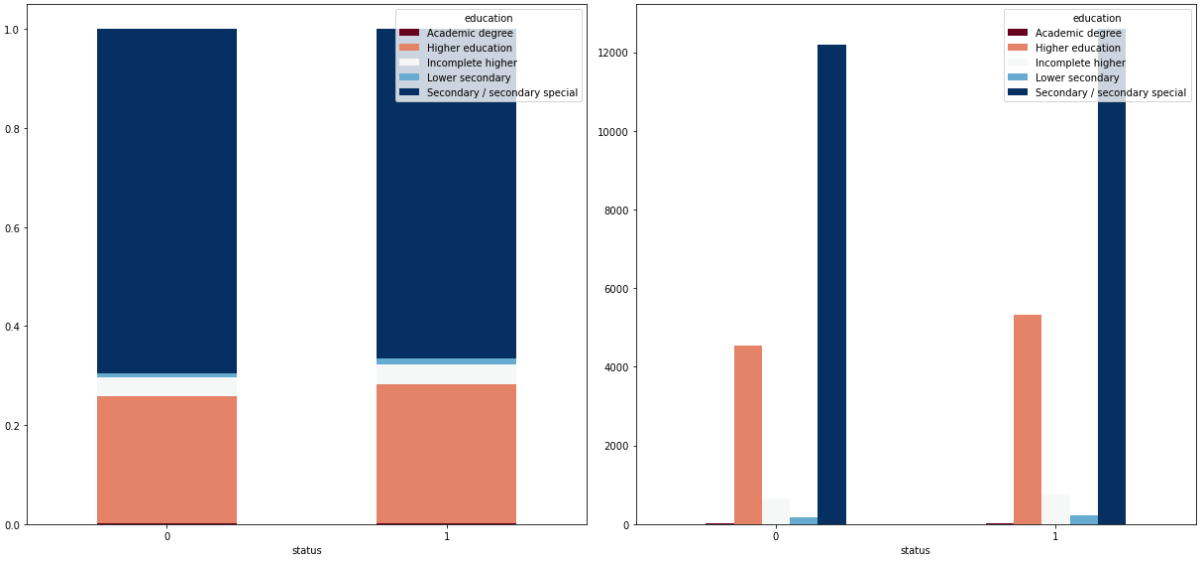
- **Count_fam_mem:**



Inference:

- The applicants with family sizes ranging more than five are less in the count so those are mapped with the family size 5.

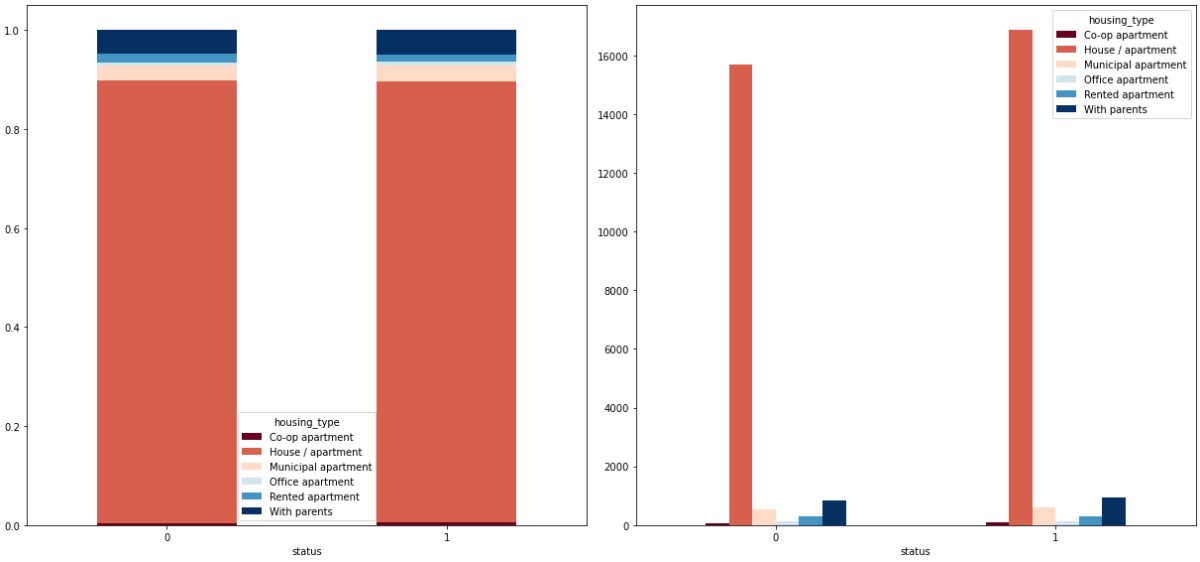
- **Education:**



Inference:

- The applicants with education qualifications as Higher and secondary education are having more creditworthiness compared with other qualifications.

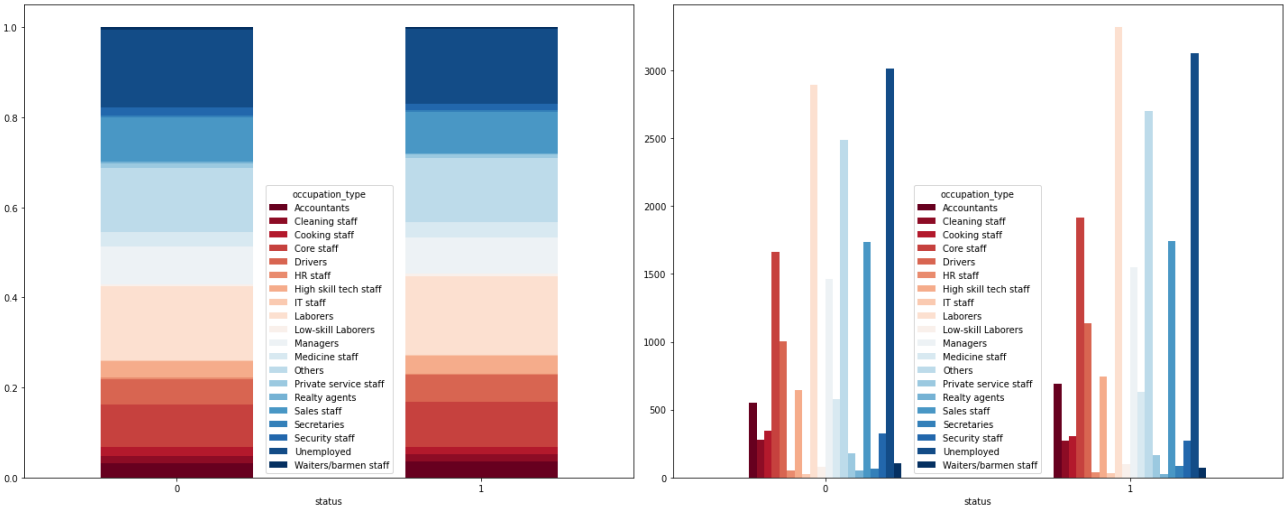
- **Housing_Type:**



Inference:

- From the above observation the mean income of "housing_type" with "With parents" and "House/apartment" are in a similar range.
- Also the median of both housing types "With parents" and "House/apartment" are equal while validating.so we have mapped it together as Own House Housing Type.

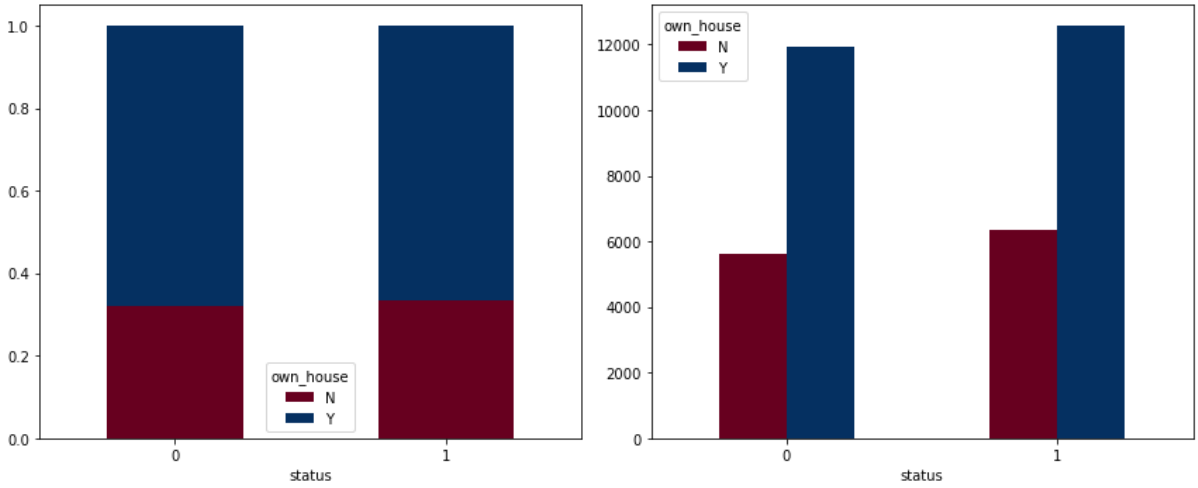
- **Occupation_type:**



Inference:

- The applicants with occupation type Laborers, managers, others, and unemployed are high in the distribution of creditworthiness among other occupation types.

- **Own_House:**



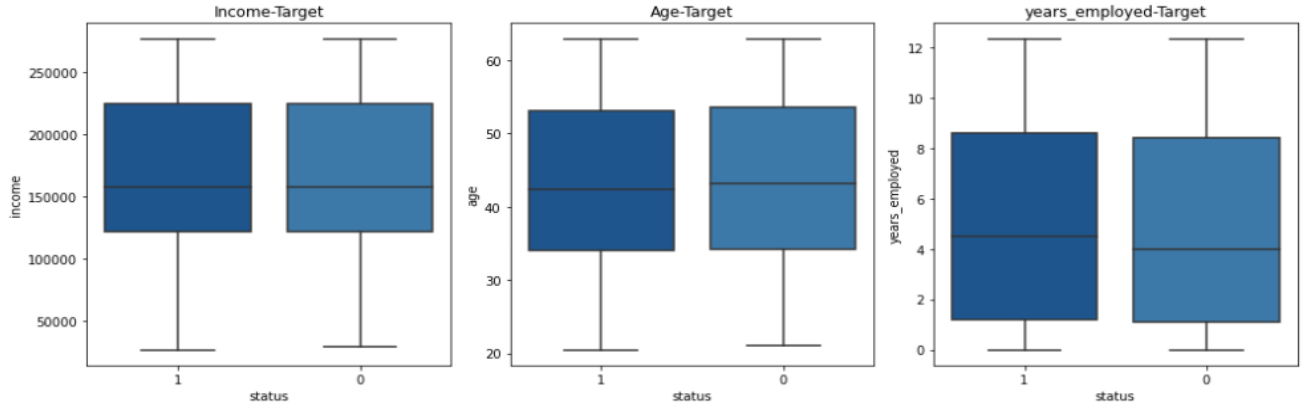
Inference:

- The count of applicants with their own house has higher creditworthiness, so there may be a significant relationship between applicants with their own house and credit approval ('status', target variable).

3.2.2 Analyzing relationship between target and continuous variable

- **Boxplot:**

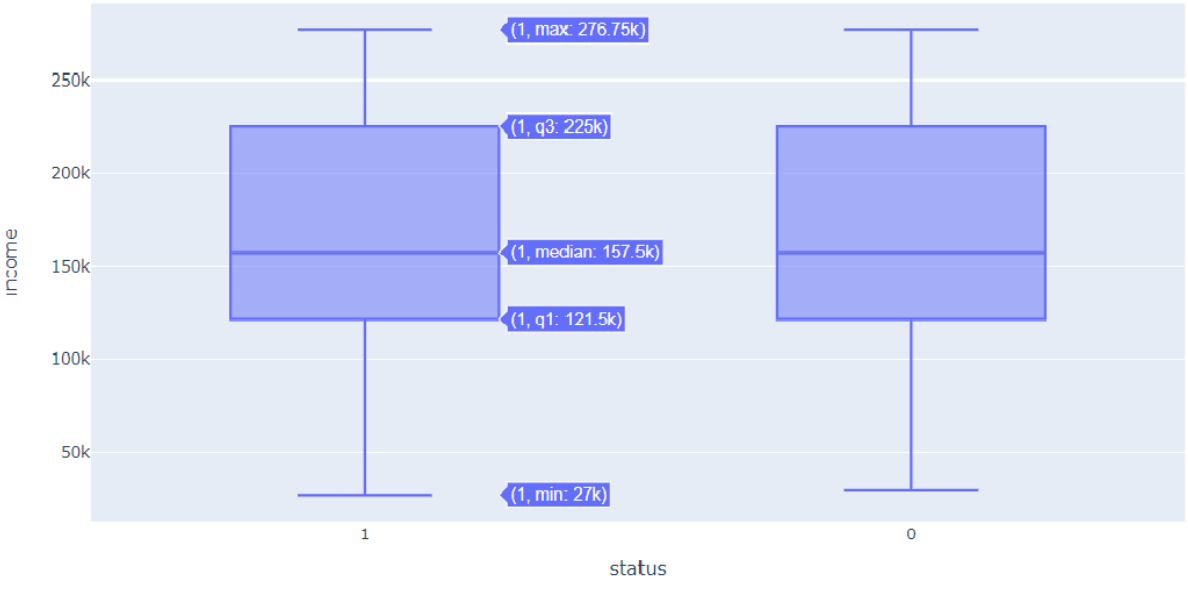
	income	age	years_employed
count	36446.000000	36446.000000	36446.000000
min	27000.000000	20.500000	0.000000
max	276750.000000	62.900000	12.350000
median	157500.000000	42.600000	4.300000
mean	173223.873800	43.664660	5.082342
skew	0.182443	0.130779	0.460523

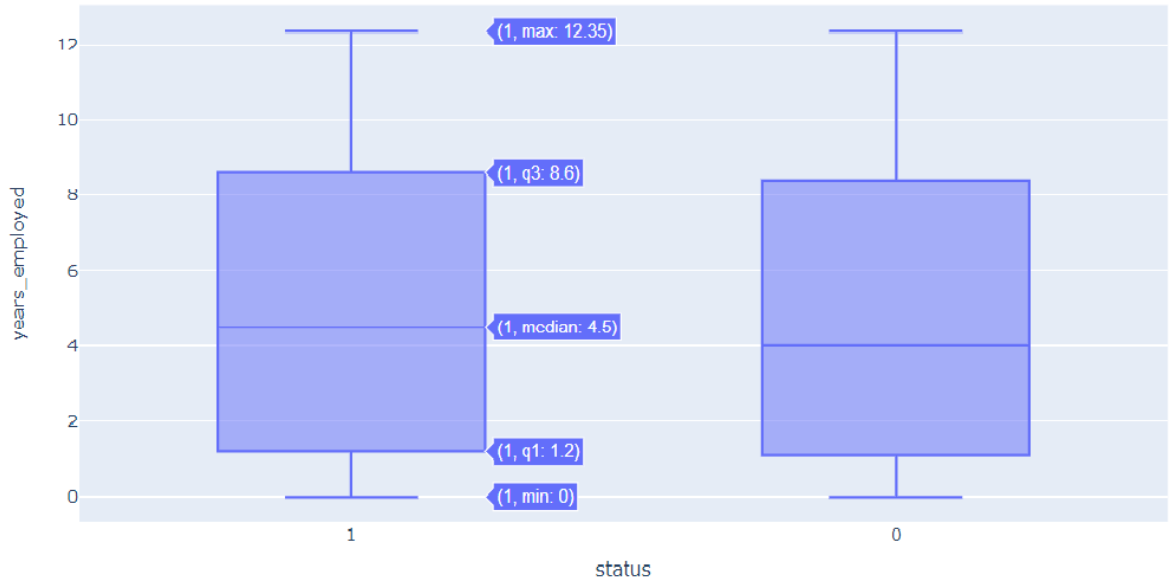


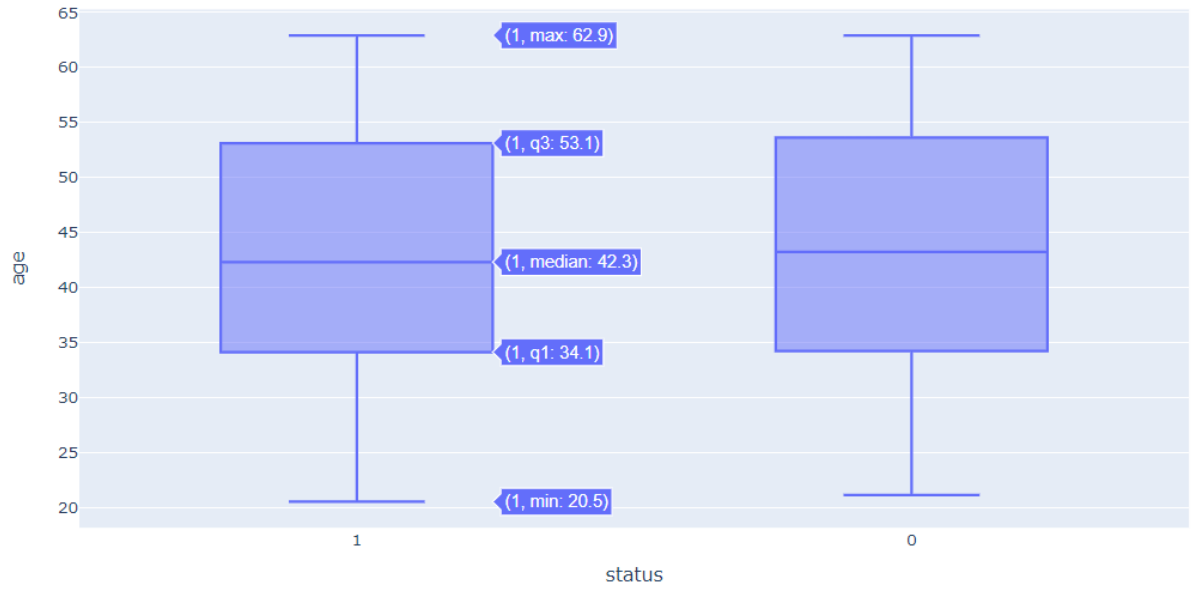
Inference:

- The applicants with higher work experience are having more creditworthiness for approval.
- The credit worthiness is equally distributed among age and income.

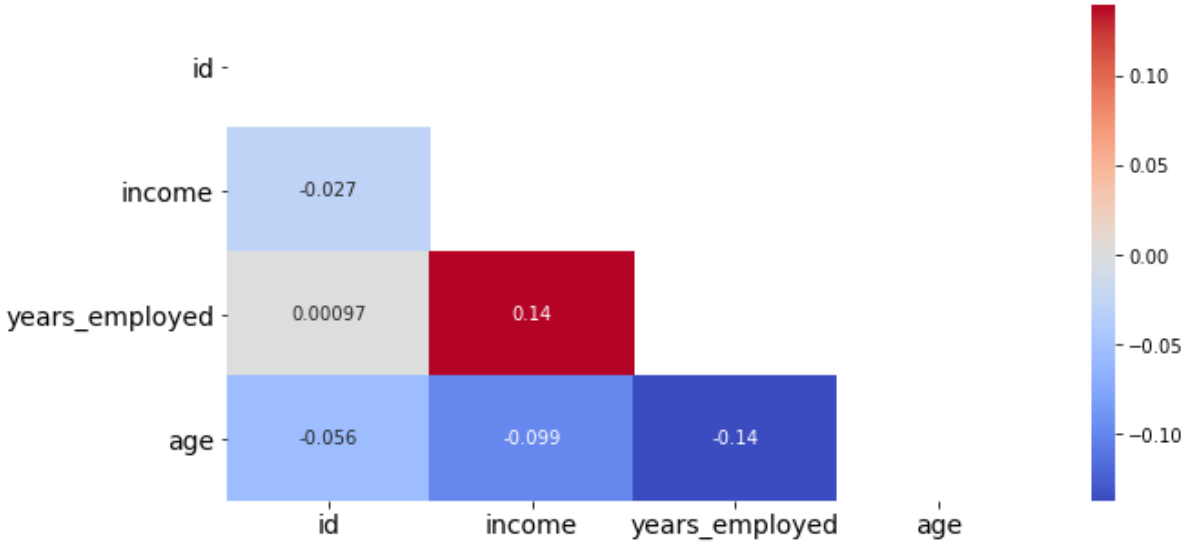
Interactive box plots for income, years_employed, and age




(1, max: 12.35)
(1, q3: 8.6)
(1, median: 4.5)
(1, q1: 1.2)
(1, min: 0)
years_employed
12
10
8
6
4
2
0
1
0
status


(1, max: 62.9)
(1, q3: 53.1)
(1, median: 42.3)
(1, q1: 34.1)
(1, min: 20.5)
age
65
60
55
50
45
40
35
30
25
20
1
0
status

3.3 Correlation for numerical feature



3.4 Modifications in category type based on the Insights

- **income_type**

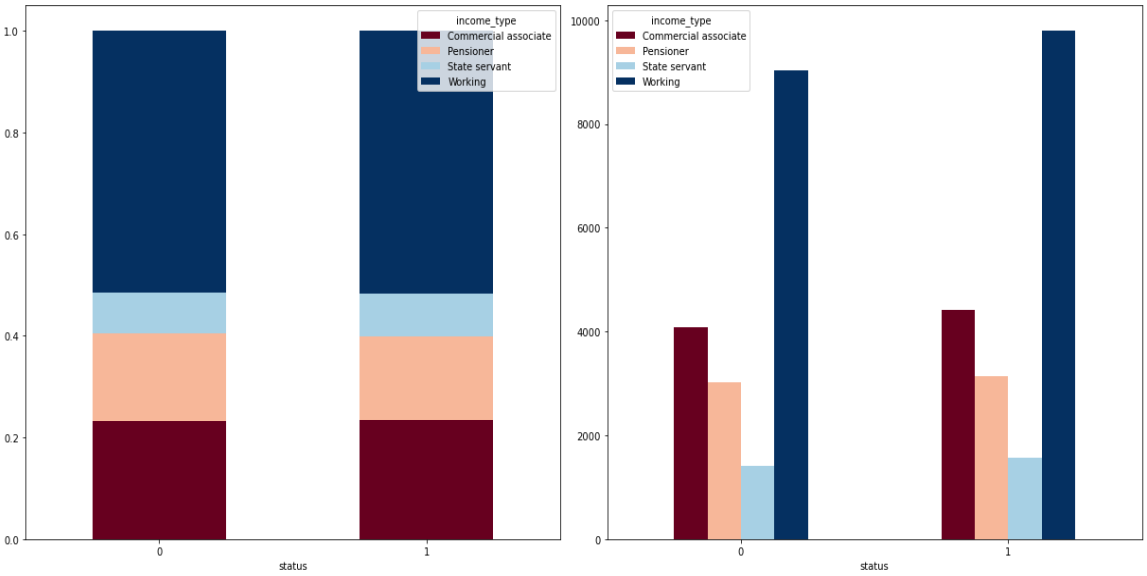
  value_counts for each category type of **income_type feature**

```
Working                 18819
Commercial associate     8490
Pensioner                6152
State servant            2985
Student                    11
```

EDA Performed:

The number of ‘Student’ category is very less so we are dropping records related to the students' income type

Plot for income_type and status after performing EDA



- **count_fam_mem:**

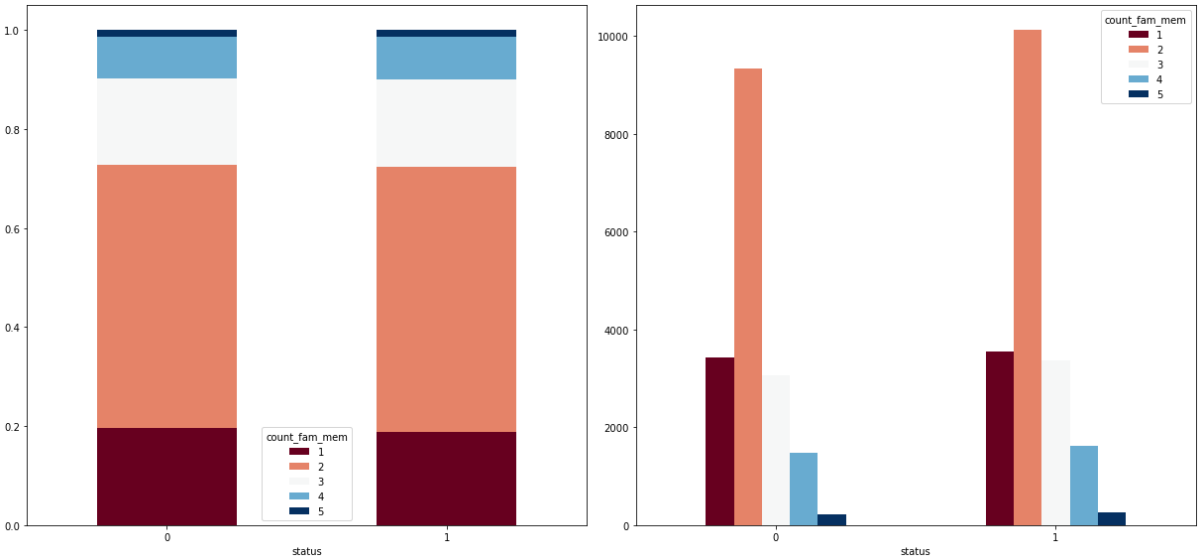
  value_counts for each category type of **count_fam_mem**

2	19455
1	6986
3	6419
4	3106
5	397
6	58
7	19
15	3
9	2
20	1

<u>EDA Performed:</u>

The applicants with family sizes ranging more than five are less in the count so those are mapped with the family size 5.

Plot for count_fam_mem and status after performing EDA



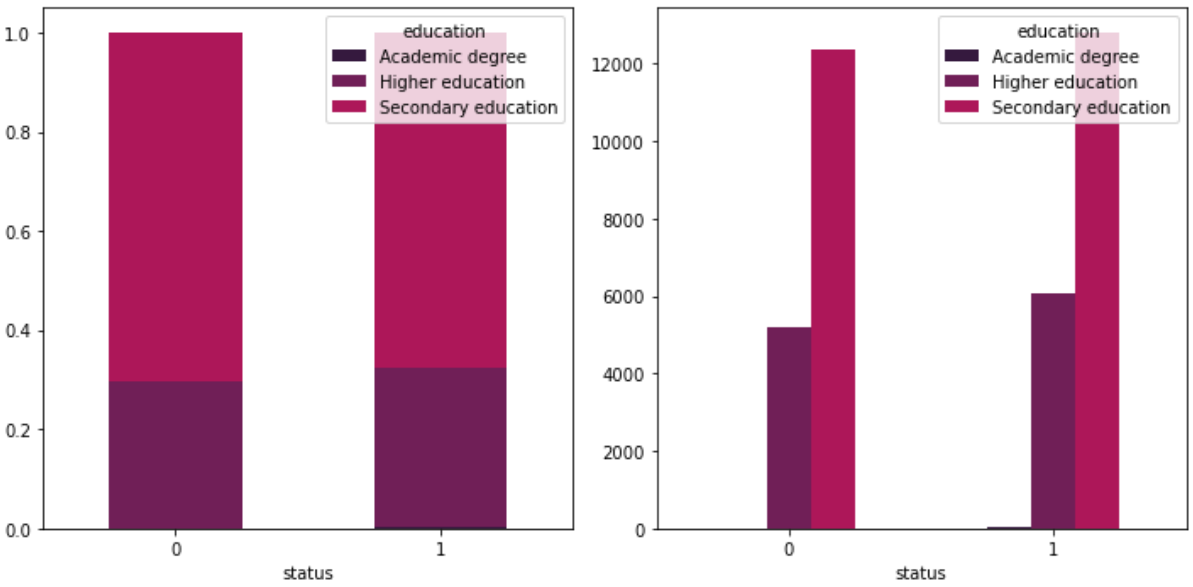
- **education**

```
Mean income for Higher education education type is: 196062.56
Mean income for Incomplete higher education type is: 186747.77
Mean income for Lower secondary education type is: 140546.79
Mean income for Secondary / secondary special education type is: 163793.39
```

<u>EDA Performed:</u>

On comparing mean values of Secondary/secondary special and lower secondary with the income they are nearly in a similar range. So the Secondary/secondary special and lower secondary are mapped together as secondary education category types, similarly while comparing the mean values of Incomplete Higher and Higher education falls under the similar range so we mapped it together.

Plot for education and status after performing EDA



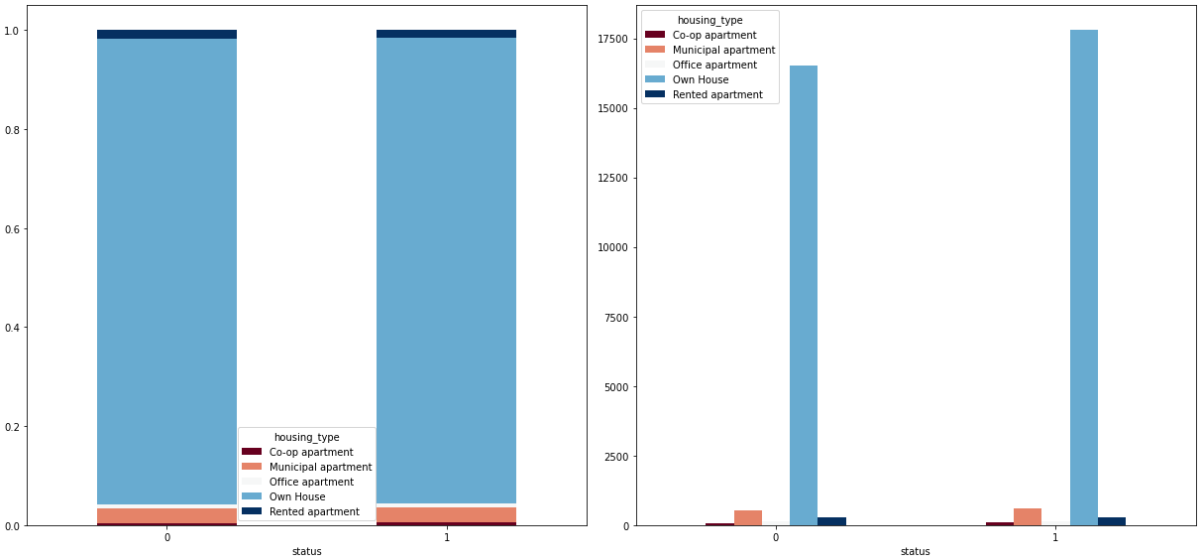
- **housing_type**

```
Mean income for  With parents housing_type is: 168118.48
Mean income for  With House / apartment housing_type is: 173175.5
```

EDA Performed:

From the above observation the mean income of "housing_type" with "With parents" and "House/apartment" are in a similar range. Also the median of both housing type "With parents" and "House/apartment" are equal while validating.so we have mapped it together as Own House Housing Type.

Plot for housing_type and status after performing EDA



- **occupation_type**

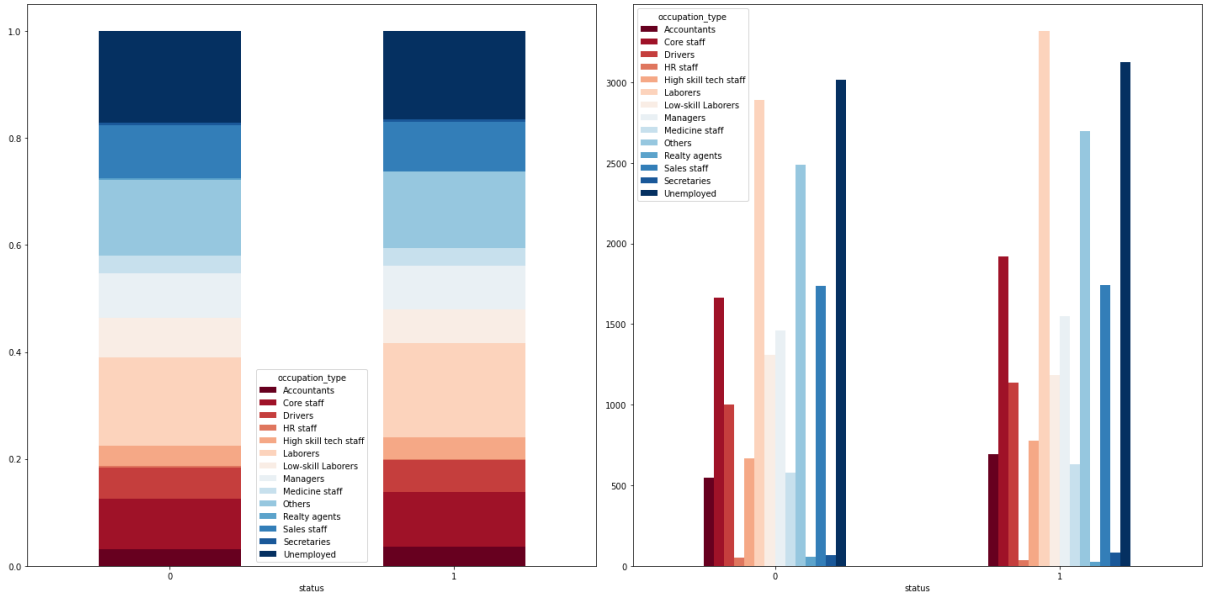
```
Mean income for Low-skill Laborers Occupation type 132865.71
Mean income for Cooking staff Occupation type 141951.98
Mean income for Security staff Occupation type 164822.38
Mean income for Cleaning staff Occupation type 146813.52
Mean income for Waiters/barmen staff Occupation type 153064.66
Mean income for Private service staff Occupation type 188058.14
```

```
Mean income for IT staff Occupation type 183172.5
Mean income for High skill tech staff Occupation type 184618.11
```

EDA Performed:

On comparing the mean income of the occupation types ['Low-skill Laborers', 'Cooking staff', 'Security staff', 'Cleaning staff', 'Waiters/barmen staff', 'Private service staff']they almost fall under similar range. So they are mapped with Low-skill Laborers.

Plot for occupation_type and status after performing EDA



3.5 Statistical Analysis

3.5.1 Test for Categorical features using chi-square

```
There is no significant between gender and status pvalue= 0.11098430030735769
There is no significant between own_car and status pvalue= 0.10209385764873544
There is a significant relationship between (Reject H0) own_house and status pvalue= 0.0016495654447334907
There is no significant between income_type and status pvalue= 0.3450550145102346
There is a significant relationship between (Reject H0) education and status pvalue= 5.6487488941586936e-08
There is a significant relationship between (Reject H0) family_status and status pvalue= 0.02482386595747338
There is no significant between housing_type and status pvalue= 0.22746154111738823
There is a significant relationship between (Reject H0) occupation_type and status pvalue= 9.275961988562151e-09
There is no significant between count_fam_mem and status pvalue= 0.3612385022435408
```

- Among the categorical features occupation_type, family_status, education, own_house variables are having a significant relationship with the target variable 'status'. (p-value less than the significant level 0.05)

3.5.2 Normality test for Numerical Feature

```
Income ShapiroResult(statistic=0.9428659677505493, pvalue=0.0)
Age ShapiroResult(statistic=0.956992506980896, pvalue=0.0)
years_employed ShapiroResult(statistic=0.8895602822303772, pvalue=0.0)
```

- From the above Shapiro test, the pvalue of all the continuous variables is less than the significant level(0.05) so the continuous variable does not follow the normal distribution

3.5.3 Test for Continuous features using Mann-Whitney

```
There is a significant relationship(Reject H0) between income and status pvalue= 0.0
There is a significant relationship(Reject H0) between years_employed and status pvalue= 0.0
There is a significant relationship(Reject H0) between age and status pvalue= 0.0
```

- Among the numerical features income,years_employed, age are having a significant relationship with the target variable 'status'. (p-value less than the significant level 0.05)

4. Encoding and Scaling

The categorical features are encoded using the One-Hot Encoding method. The numerical features are Scaled using the Standard Scaler method before modeling.

5. ML model

5.1 Simple model using Logistic Regression (without feature selection)

Confusion Matrix:

```
array([[ 845, 2651],
       [ 819, 2975]], dtype=int64)
```

Test_report:

	precision	recall	f1-score	support
0	0.51	0.24	0.33	3496
1	0.53	0.78	0.63	3794
accuracy			0.52	7290
macro avg	0.52	0.51	0.48	7290
weighted avg	0.52	0.52	0.49	7290

Train_report:

	precision	recall	f1-score	support
0	0.51	0.25	0.34	14039
1	0.53	0.77	0.63	15117
accuracy			0.52	29156
macro avg	0.52	0.51	0.48	29156
weighted avg	0.52	0.52	0.49	29156

6. Feature Selection

As we are having more number of categorical feature (with more sub-categories for each feature), so feature selection using RFE and other methods is not effective.

*****(We have tried to build a model with features selected based on RFE and statistical analysis, model score was not effective)**

7. Probability Cut-off for Logistic Regression

7.1 Manual method using score_card

S.No	Probability Cutoff	AUC Score	Precision Score	Recall Score	Accuracy Score	Kappa Score	f1-score
0	0.1	0.5	0.517377	1	0.517377	0	0.681936
1	0.2	0.5	0.517377	1	0.517377	0	0.681936
2	0.3	0.5	0.517377	1	0.517377	0	0.681936
3	0.4	0.50103	0.517893	0.997702	0.518292	0.002132	0.681848
4	0.5	0.508803	0.523101	0.77656	0.518109	0.017919	0.625116
5	0.6	0.500478	0.55	0.007778	0.483355	0.000923	0.015339
6	0.7	0.499905	0	0	0.482532	-0.000183	0
7	0.8	0.5	0	0	0.482623	0	0
8	0.9	0.5	0	0	0.482623	0	0

7.2 Youden Index

S.No	TPR	FPR	Threshold	Difference
0	0.528018	0.493462	0.518865	0.034556
1	0.472335	0.437938	0.522999	0.034397
2	0.527842	0.493462	0.518871	0.034379
3	0.473926	0.439644	0.52272	0.034282
4	0.474103	0.439833	0.522707	0.03427

In both the method the optimum probability cut-off is 0.5 and the same has been implemented for predicting the target values

8. ML model Result Comparision

Score	Model	Score
Train Accuracy Score	Logistic Regression	0.52
Test Accuracy Score		0.52
Train ROC AUC		0.5137

Test ROC AUC		0.5262
Train Confusion Matrix		([[845, 2651], [819, 2975]]
Test Confusion Matrix		([[3552, 10487], [3411, 11706]]
Train Accuracy score	Decision Tree (Random search)	0.541
Test Accuracy Score		0.53
Train ROC AUC		0.55
Test ROC AUC		0.53
Train Confusion Matrix		[[4483 9556] [3818 11299]]
Test Confusion Matrix		[[1128 2368] [1016 2778]]
Train Accuracy score	**Random Forest (Random Search)**	**0.8**
Test Accuracy Score		**0.65**
Train ROC AUC		**0.9**
Test ROC AUC		**0.71**
Train Confusion Matrix		[[10869 3170] [2689 12428]]
Test Confusion Matrix		[[2141 1355] [1154 2640]]
Train Accuracy score	XGB (tuned)	0/64
Test Accuracy Score		0.77
Train ROC AUC		0.86
Test ROC AUC		0.68
Train Confusion Matrix		[[10465 3574] [2961 12156]]
Test Confusion Matrix		[[2059 1437] [1198 2596]]
Train Accuracy score	**XGB**	**0.74**
Test Accuracy Score		**0.62**
Train ROC AUC		**0.82**
Test ROC AUC		**0.67**
Train Confusion Matrix		[[9750 4289] [3163 11954]]
Test Confusion Matrix		[[1938 1558] [1153 2641]]
Train Accuracy score	ADA Boost (tuned)	0.54

Test Accuracy Score		0.53
Train ROC AUC		0.86
Test ROC AUC		0.68
Train Confusion Matrix		[[4779 9260] [3981 11136]]
Test Confusion Matrix		[[1151 2345] [1040 2754]]
Train Accuracy score	Gradient Boosting	0.79
Test Accuracy Score		0.64
Test ROC AUC		0.69
Test Confusion Matrix		[[2056 1440] [1165 2629]]

9. AutoML (PyCaret)

S.No	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
1	Random Forest Classifier	0.6409	0.6971	0.665	0.6502	0.6574	0.2802	0.2804	1.718
2	Decision Tree Classifier	0.6396	0.6763	0.6115	0.666	0.6375	0.2805	0.2816	0.203
3	Extra Trees Classifier	0.6376	0.689	0.6159	0.6617	0.6379	0.2762	0.2769	1.276
4	K Neighbors Classifier	0.6121	0.6483	0.6269	0.6257	0.6262	0.2232	0.2232	0.325
5	Extreme Gradient Boosting	0.6084	0.6477	0.6381	0.6184	0.628	0.2148	0.215	1.632
6	Light Gradient Boosting Machine	0.5944	0.6287	0.6341	0.6035	0.6184	0.1862	0.1865	0.597
7	Gradient Boosting Classifier	0.5449	0.5599	0.59	0.5577	0.5733	0.0866	0.0868	1.196
8	Ada Boost Classifier	0.5288	0.5391	0.5421	0.5458	0.5438	0.0566	0.0567	0.528
9	Naive Bayes	0.5186	0.5175	0.4954	0.5386	0.516	0.0388	0.0389	0.167
10	Linear Discriminant Analysis	0.5146	0.5224	0.5096	0.5332	0.521	0.0295	0.0296	0.255
11	Ridge Classifier	0.5145	0	0.5095	0.5331	0.5209	0.0294	0.0294	0.195
12	Logistic Regression	0.5006	0.4972	0.4286	0.3664	0.381	0.0067	0.0068	0.887
13	SVM - Linear Kernel	0.4926	0	0.3	0.1554	0.2048	0	0	0.571
14	Quadratic Discriminant Analysis	0.4817	0.5	0	0	0	0	0	0.246
15	Dummy Classifier	0.4817	0.5	0	0	0	0	0	0.204

Best Model Output:

```
RandomForestClassifier(bootstrap=True, ccp_alpha=0.0, class_weight=None,
criterion='gini', max_depth=None, max_features='auto',
max_leaf_nodes=None, max_samples=None,
min_impurity_decrease=0.0, min_impurity_split=None,
min_samples_leaf=1, min_samples_split=2,
min_weight_fraction_leaf=0.0, n_estimators=100,
n_jobs=-1, oob_score=False, random_state=123, verbose=0,

warm_start=False)
```

10. Inference and Conclusion for the built Model

In this project credit worthiness of the potential applicants is identified. Out of the built model, XGB and Random Forest with hyperparameter tunning performed well for the train and test scores.

11. References

- https://www.kaggle.com/datasets/rikdifos/credit-card-approval-prediction/code?datasetId=426827&sortBy=voteCount
- https://www.kaggle.com/code/caesarmario/99-9-approval-prediction-w-pycaret
- https://www.analyticsvidhya.com/blog/2016/02/complete-guide-parameter-tuning-gradient-boosting-gbm-python/#h2_6
- https://www.kaggle.com/code/umerkk12/credit-card-predictive-analysis