**FINAL PROJECT REPORT**

**PROJECT TITLE :** LOAN PREDICTION BASED ON CUSTOMER   
 BEHAVIOUR.

**BUSINESS PROBLEM :**

In the realm of financial services, specifically within the lending sector, there exists a critical need for an effective and accurate system to predict the likelihood of a customer defaulting on a loan based on their behavior and demographic information. The dataset in question encompasses vital attributes such as income, age, relationship status, car ownership, profession, state, city, house ownership, experience, current job years, and current house years.

The primary challenge at hand is to develop a robust predictive model that can analyze and interpret the intricate relationships between these customer-specific features and their propensity to default on loan repayments. The goal is to minimize financial risk for the lending institution by identifying high-risk customers while simultaneously ensuring that creditworthy applicants are not unjustly denied access to loans.

In addressing this business problem, the goal is to harness the power of advanced analytics and machine learning techniques to develop a predictive model that not only accurately assesses the creditworthiness of applicants by the predicting whether it is risky or not to lend a loan to the customer but also aligns with the broader strategic objectives of the lending institution. This solution aims to enhance decision-making, minimize financial losses, and foster a more resilient and adaptive lending environment.

**OBJECTIVE :**

The primary objective of this initiative is to develop a robust and accurate predictive model for loan approval, leveraging customer behaviour and demographic information. The model aims to achieve the following specific goals:

* **Risk Identification and Mitigation:** To develop a model that effectively identifies high-risk customers based on their behaviour and demographic attributes. To implement mechanisms to minimize the financial risk associated with loan approvals by accurately predicting the likelihood of customer default.
* **Precision in Decision-Making:** For designing and implementing a predictive model that ensures precision in decision-making, avoiding both false positives and false negatives. To strive for a balanced approach that optimally weighs risk aversion and inclusivity, ensuring that creditworthy applicants are not erroneously denied loans while identifying high-risk individuals.
* **Adaptability and Robustness:** Ensure the robustness of the model by incorporating features that allow it to adapt to emerging trends and maintain predictive accuracy in dynamic financial landscapes.
* **Enhanced Customer Experience:** To streamline the loan approval process for creditworthy applicants, minimizing unnecessary delays and enhancing overall customer experience and also aim for a customer-centric approach that balances risk mitigation with the goal of providing a positive and efficient lending experience.
* Incorporate mechanisms for ongoing evaluation.
* Align the development and deployment of the predictive model with the broader strategic objectives of the lending institution.
* Foster a balanced and resilient approach to loan prediction.
* **Empower Informed Decisions:** Provide financial institutions with a tool to make informed decisions regarding loan approvals, ultimately improving risk management.

* **Minimize Defaults:** Reduce the risk of defaults by identifying high-risk applicants early in the lending process.

This objective sets the stage for the development of a comprehensive solution that addresses the multifaceted challenges associated with loan prediction based on customer behaviour, promoting both financial prudence and customer-centricity in the lending process.

Hence, the overarching objective is to construct a sophisticated predictive machine learning model that predicts if it is a risk to grant the loan to a customer or not.

**SOLUTION APPROACH:**

The solution approach involves a systematic and iterative process, combining advanced analytics and machine learning methodologies to develop an accurate and adaptable predictive model for loan approval. The key steps are as follows:

1. Data Exploration and Understanding.
2. Data Pre-Processing.
3. Data Visualization.
4. Experimenting with Diverse Machine Learning Models.
5. Accuracy-Driven Model Selection.
6. Deploying the Chosen Machine Learning Model.

Throughout the development of this project, we have relied extensively on the capabilities of Python and its diverse libraries. Python's versatility has played a pivotal role in various stages of our work, from data preprocessing and visualization to the implementation of machine learning models. Additionally, we have utilized Tableau, a powerful data visualization tool, to gain deeper insights into the data and effectively communicate our findings.

In tackling the classification problem presented, we have leveraged the power of supervised and ensemble machine learning techniques. Our exploration of various algorithms has encompassed the following models :

1. Logistic Regression.
2. Decision Tree Classifier.
3. Random Forest Classifier.
4. Extra Trees Classifier.
5. Adaptive Boosting (AdaBoost Classifier).
6. AdaBoost Classifier with Decision Trees as base Estimator.
7. Gradient Boosting Classifier.
8. Extreme Gradient Boosting Classifier (XGBClassifier).
9. Light Gradient Boosting Classifier (LGBMClassifier).
10. Cat Boost Classifier.

In addition to the aforementioned supervised and ensemble machine learning algorithms, we have also explored the application of deep learning by implementing an artificial neural network (ANN) model.

**SCOPE:**

The project scope involves the end-to-end development and implementation of a predictive model for loan approval, leveraging customer behaviour and demographic data. This includes the collection and preprocessing of pertinent information such as income, age, relationship status, car ownership, profession, state, city, house ownership, experience, current job years, and current house years. The focus is on creating an advanced analytics and machine learning model that accurately assesses creditworthiness, with a specific emphasis on risk identification and mitigation. Precision in decision-making, adaptability to evolving market conditions, and compliance with regulatory standards are key pillars of the project. Additionally, the initiative aims to enhance the overall customer experience by streamlining the loan approval process for creditworthy applicants, while continuous improvement mechanisms and strategic alignment with the institution's goals ensure long-term effectiveness and relevance. The scope also encompasses comprehensive documentation, reporting, training, and considerations for scalability to facilitate a seamless and sustainable deployment of the predictive model.

The Deep learning techniques and models approach presents a promising avenue for future development or future scope in this project. Further exploration of deep learning architectures and algorithms holds immense potential for enhancing the performance and applicability of the proposed classification system. By leveraging the power of deep learning, we anticipate achieving greater accuracy, generalizability, and robustness in tackling the problem that is predicting the risk factor for lending a loan to the customer based on the customer behaviour.

**TEAM SIZE:**

Our team comprised six individuals who collaborated effectively to carry out the project. The team members are:

1. Pattan Shekshavali
2. Nellore Sai Nikhil
3. G. Chaitanya Sai
4. M. Pranai Kumar Reddy
5. Pujan Vittala
6. D. Surya Teja

**TIME LINE:**

**AGILE METHOD :**

**DATA SOURCES & DATA UNDERSTANDING :**

The dataset for this project was obtained from Kaggle, a popular platform for data sharing and machine learning competitions. The dataset contains information on a sample of loan applicants and their subsequent repayment history. The data was collected from a financial institution and includes a variety of demographic and financial attributes of the applicants, as well as their loan repayment status.

The dataset consists of 13 columns out of which , each representing a specific attribute of the loan applicant. The columns and their descriptions are as follows:

* id: A unique identifier for each loan applicant
* income: The annual income of the loan applicant
* age: The age of the loan applicant
* Married/Single: The marital status of the loan applicant
* car\_ownership: Whether the loan applicant owns a car (Yes/No)
* profession: The occupation of the loan applicant
* state: The state of residence of the loan applicant
* city: The city of residence of the loan applicant
* house\_ownership: Whether the loan applicant owns a house (Yes/No)
* experience: The professional experience of the loan applicant in years
* current\_job\_yrs: The number of years the loan applicant has been in their current job
* current\_house\_yrs: The number of years the loan applicant has lived in their current house
* risk\_flag: An indicator of whether the loan applicant has ever defaulted on a loan (1=Yes, 0=No)

The dataset used for this project is comprehensive and provides valuable information about loan applicants and their repayment behaviour. The data cleaning and preprocessing steps ensured the quality and consistency of the data, while the exploratory data analysis provided insights into the characteristics of the data and its potential patterns. This understanding of the data was crucial for developing effective machine learning models for loan risk prediction. The findings of the patterns and details from the exploratory data analysis are mentioned and described in the later part of the documentation.

Upon examination, the dataset was found to consist of 13 columns and 25,200 rows. These columns were categorized into two distinct data types: int64 and object. The int64 data type represented “seven numerical columns”, while the object data type represented “six categorical columns”. Notably, all values within the object-type columns were stored as strings.

**DATA PREPARATION :**

Data preparation, also known as data preprocessing, is a crucial step in the machine learning pipeline that involves transforming raw data into a format suitable for training and evaluating machine learning models. It encompasses a wide range of tasks, including data cleaning, wrangling, and feature engineering, aimed at ensuring data quality, consistency, and relevance for the intended machine learning task.

The key aspects of data preparation are :

* **Data Cleaning:** To ensure the integrity and reliability of the data, we meticulously performed data cleaning using Python libraries NumPy and Pandas, Matplotlib and Seaborn. This involves identifying and correcting errors, inconsistencies, and missing values in the data. Techniques like imputation, outlier removal, and error correction are employed to ensure data integrity.

Fortunately, we have found no such things like error, missing values, inconsistencies or outliers to deal with. The dataset doesn’t contain any duplicate rows too.

* **Data Wrangling:** This involves transforming the data into a format compatible with the chosen machine learning algorithm. This may include data type conversion, data normalization, and data encoding for categorical variables.

To handle the categorical data, we employed label encoding, a technique that transforms categorical values into numerical representations. **Label encoding**, also known as **ordinal encoding**, is a technique for converting categorical data into numerical values while preserving the inherent order between the categories. For example, the labels "low", "medium", and "high" could be encoded as 1, 2, and 3, respectively.

Subsequently, we applied the **Standard Scaler** technique to standardize the numerical data, ensuring consistency in the scale of the features. StandardScaler is a data preprocessing technique employed to normalize features by subtracting the mean and scaling to unit variance. This effectively centers each feature around the mean and assigns a standard deviation of one. By applying this transformation, StandardScaler ensures that all features contribute equally to the machine learning model, preventing any single feature from dominating the learning process and influencing the model's performance. Mathematically, StandardScaler operates by subtracting the mean of each feature from each data point and then dividing each data point by the standard deviation of the feature.

x\_std = (x - μ) / σ

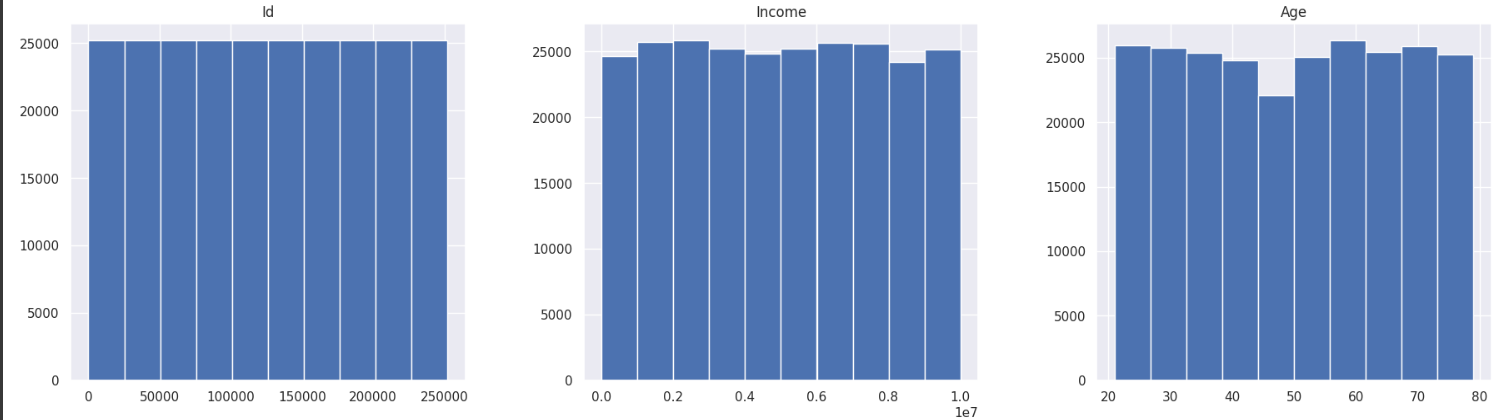
where,

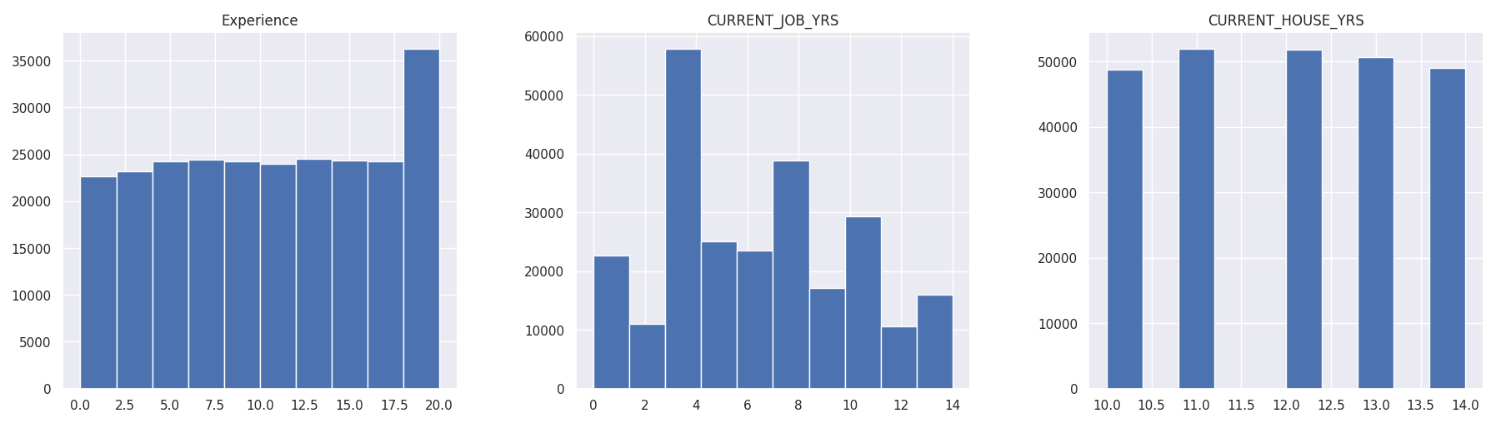
* x\_std is the standardized data point
* x is the original data point
* μ is the mean of the feature
* σ is the standard deviation of the feature

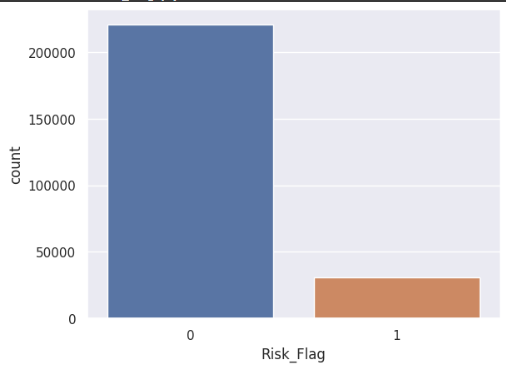
To address the imbalanced class distribution, we utilized the **SMOTE (Synthetic Minority Oversampling Technique)** algorithm. This technique effectively augmented the minority class by generating synthetic minority examples, resulting in a balanced dataset with an equal number of sample rows for both unique values of the 'risk\_flag' feature. The SMOTE algorithm commences by selecting a minority class data point. Subsequently, it identifies the k nearest neighbours of the chosen data point. Next, a random selection of one of the k nearest neighbours is performed. A new synthetic data point is then created by interpolating between the selected data point and its chosen neighbour. Finally, the newly generated synthetic data point is added to the dataset. Through this process, SMOTE effectively reduces bias, enhances the accuracy of models on minority class data, and mitigates the risk of model overfitting.

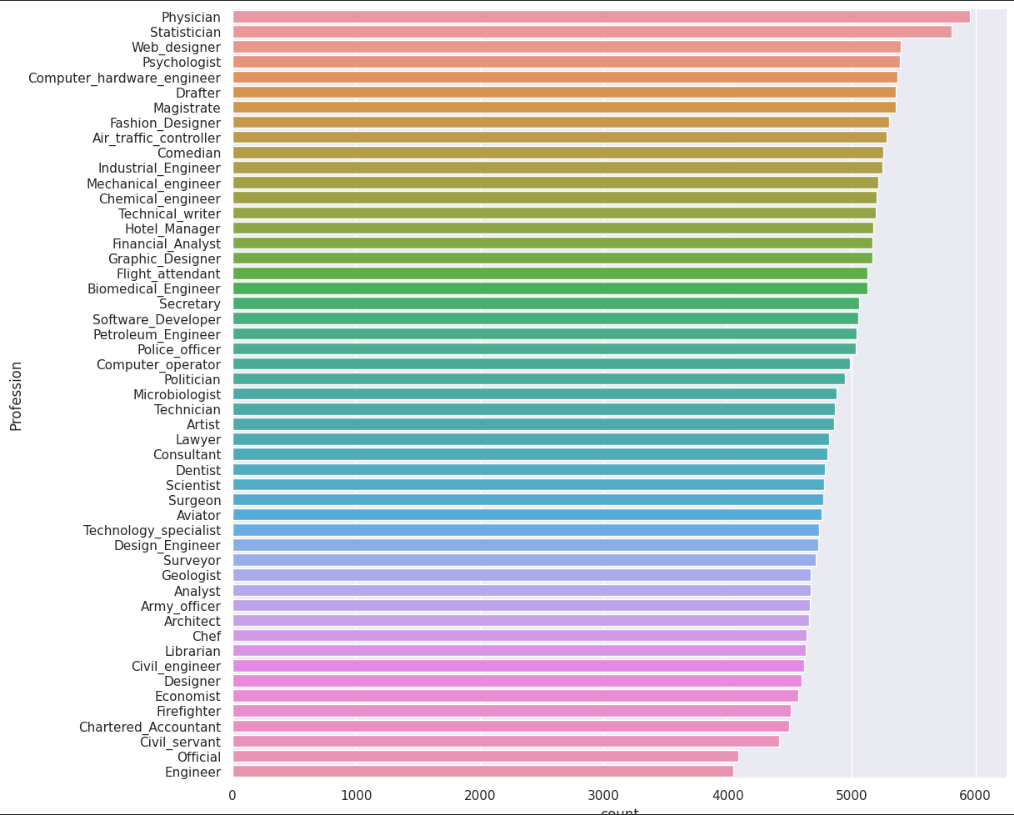
**DATA VISUALIZATION:**

Leveraging the powerful visualization capabilities of Matplotlib and Seaborn, we embarked on a journey to unveil the hidden patterns and relationships within the dataset. Through a series of insightful visualizations, we gained a deeper understanding of the data distribution, variable correlations, and potential outliers. These insights proved invaluable in guiding our subsequent analysis and model development. The below mentioned are the visualizations obtained:

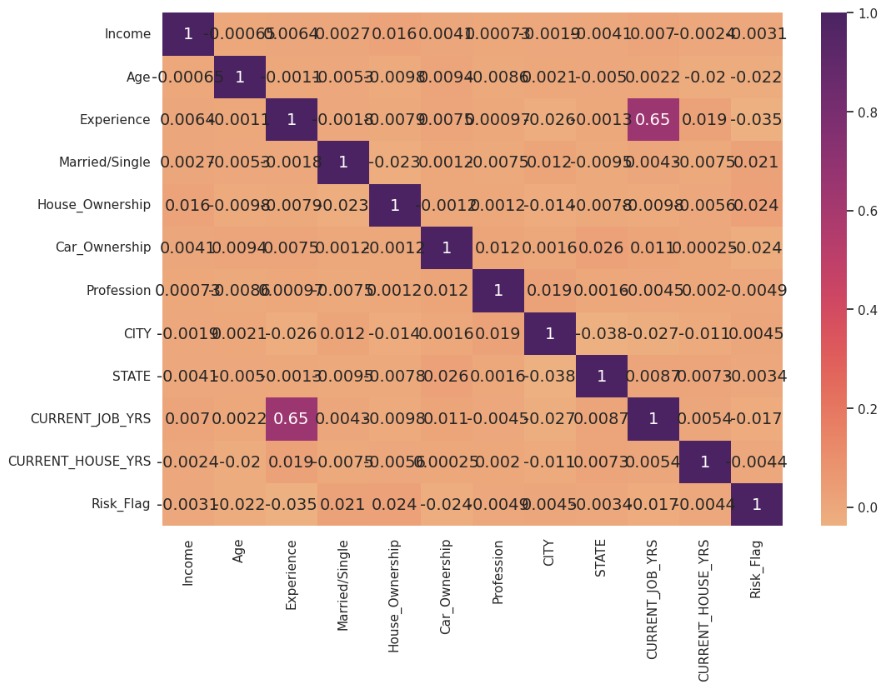








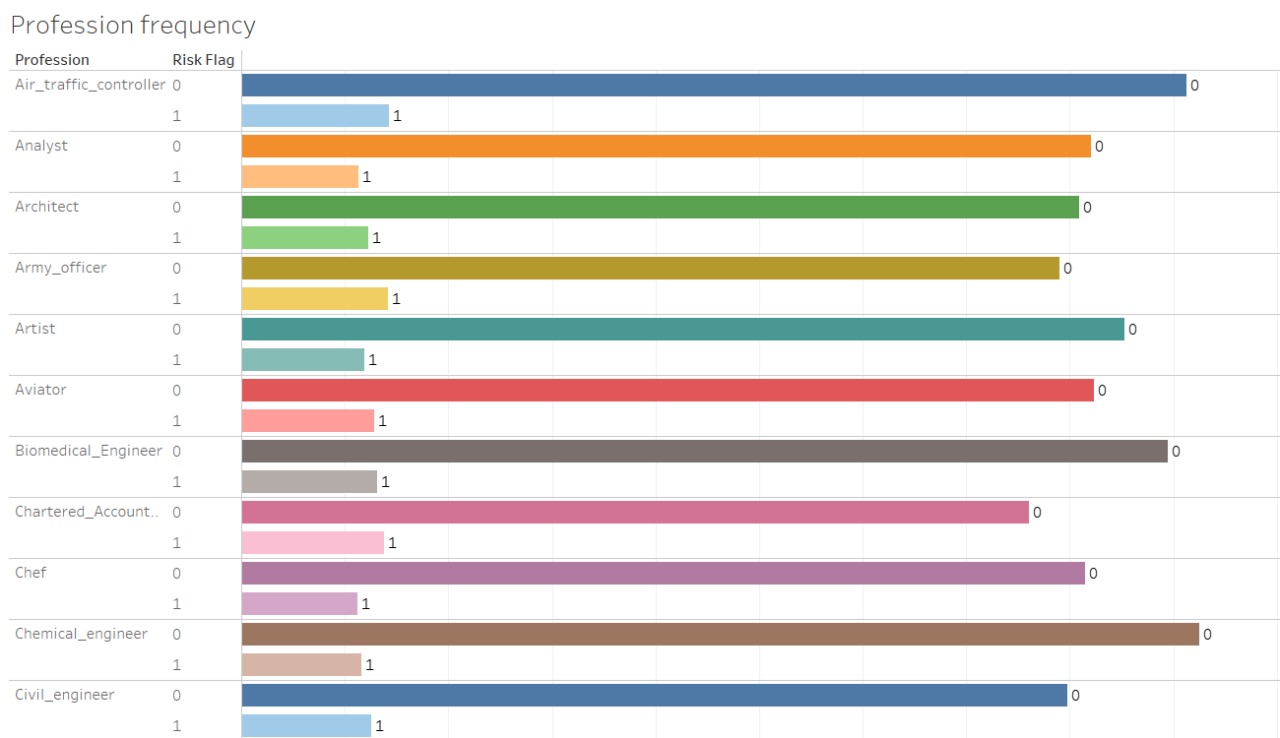
**CORRELATION HEAT MAP**

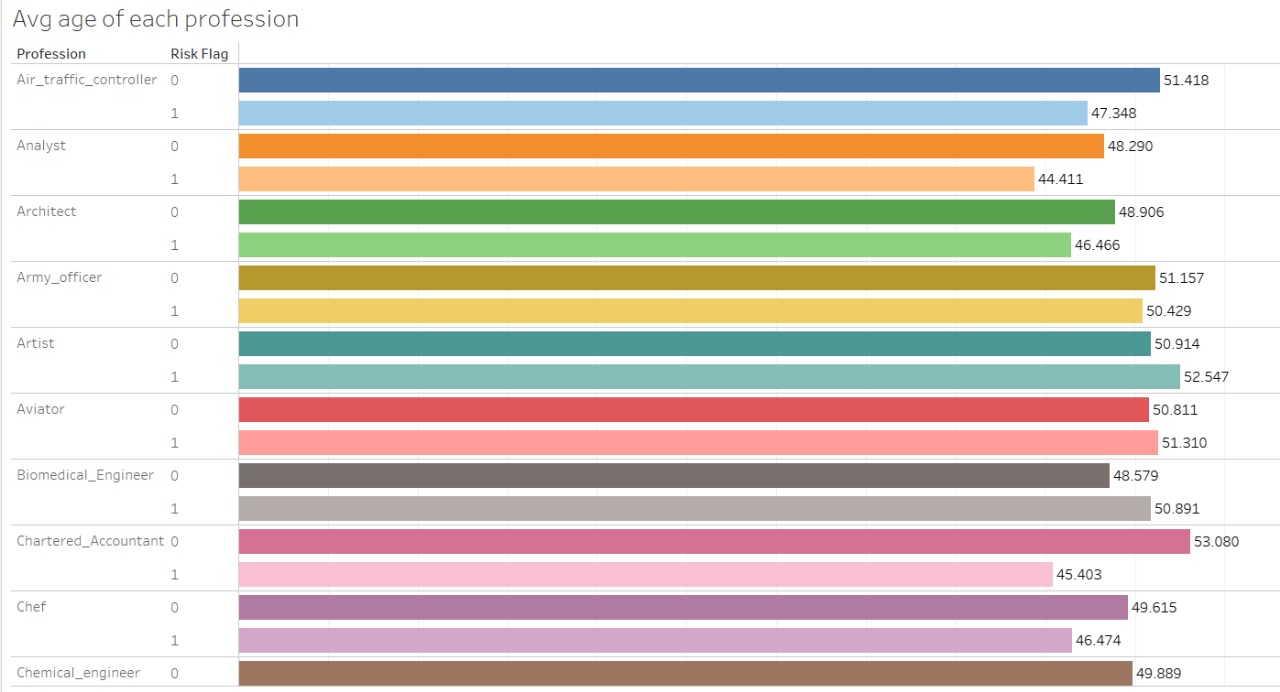


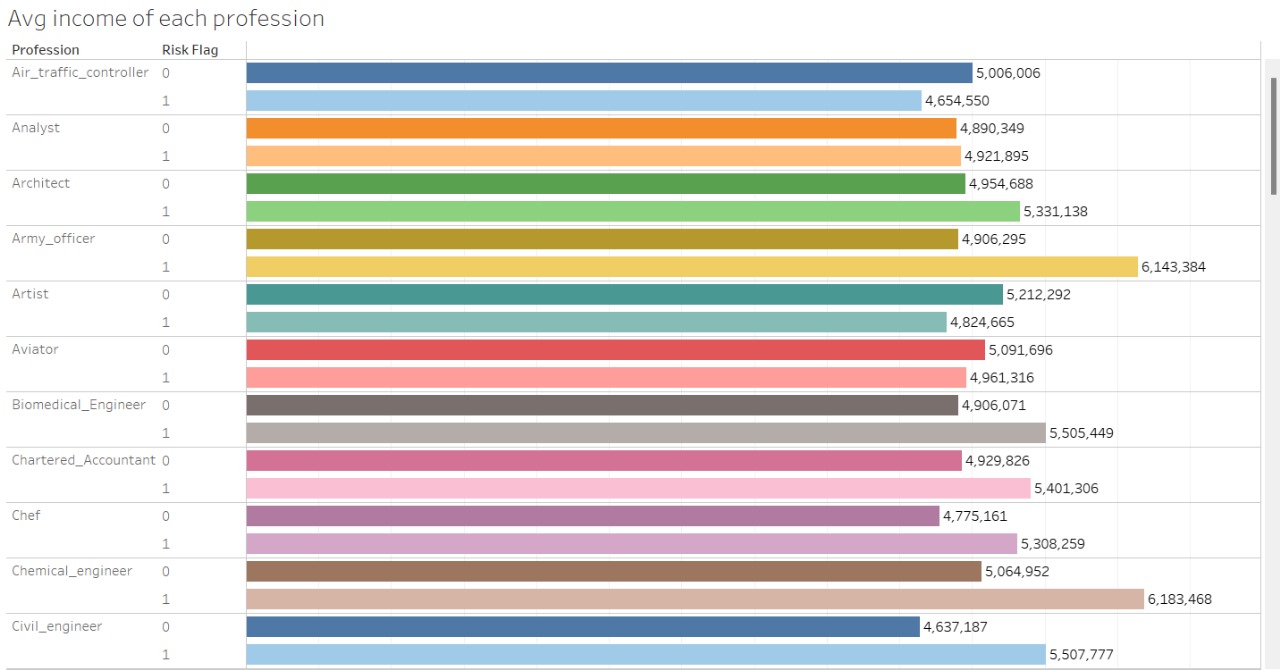
From the above correlation graph, we can clearly conclude that both Experience and current\_job\_yrs features are highly correlated to each other. But we neither removed anyone of the feature nor created any new feature that would replace them without effectively changing their removal. This is because both of them individually effect the prediction of the target variable risk\_flag.

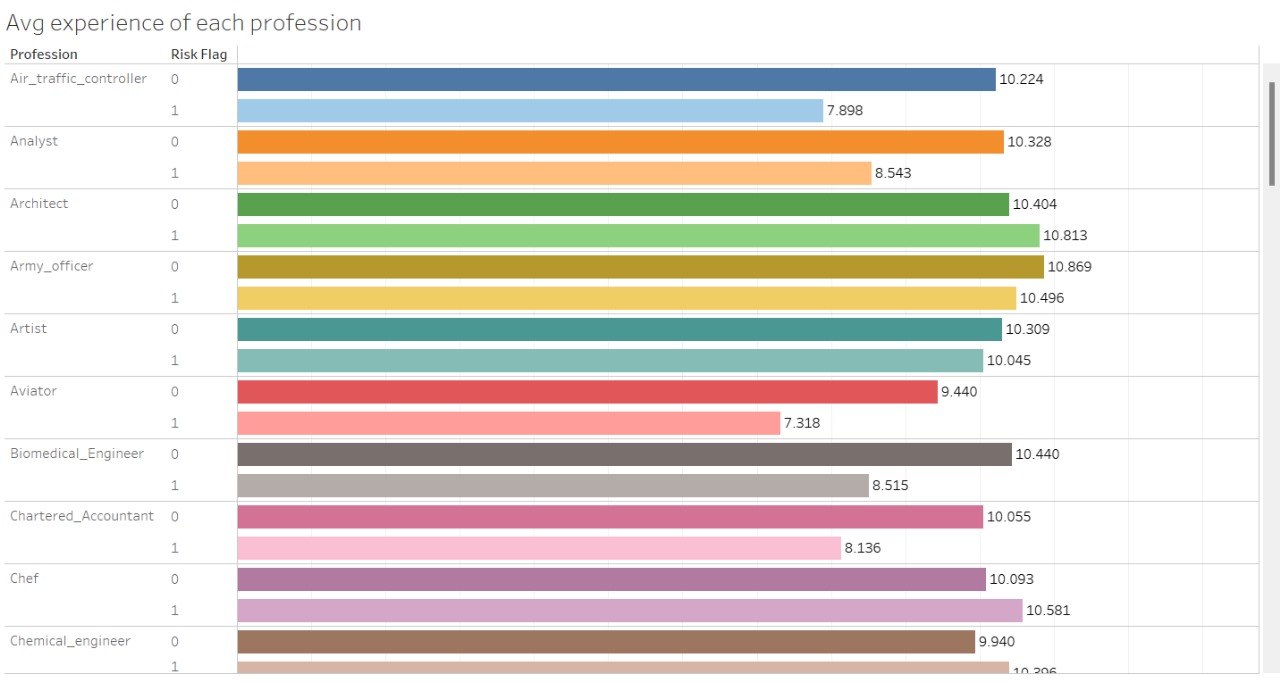
To further delve into the intricacies of the dataset, we turned to Tableau, a comprehensive data visualization and exploration tool. By harnessing the power of Tableau's interactive dashboards and charts, we were able to uncover intricate patterns, identify subtle trends, and gain a deeper understanding of the relationships between variables. This in-depth exploration provided us with valuable insights that informed our subsequent analysis and model development.

The below is the final dashboard and graphs obtained using Tableau:







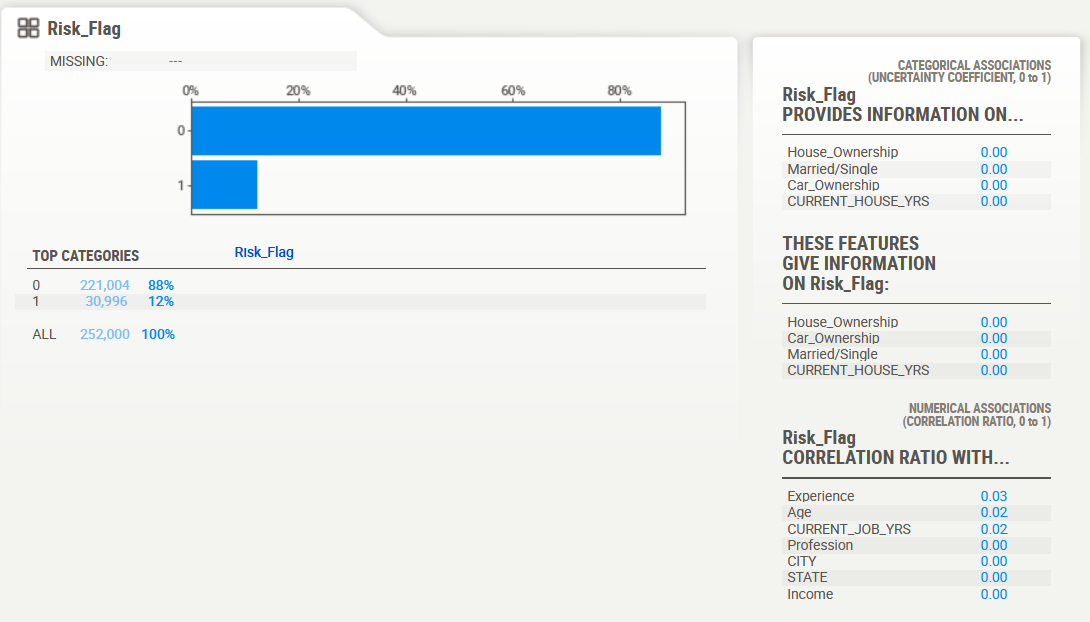


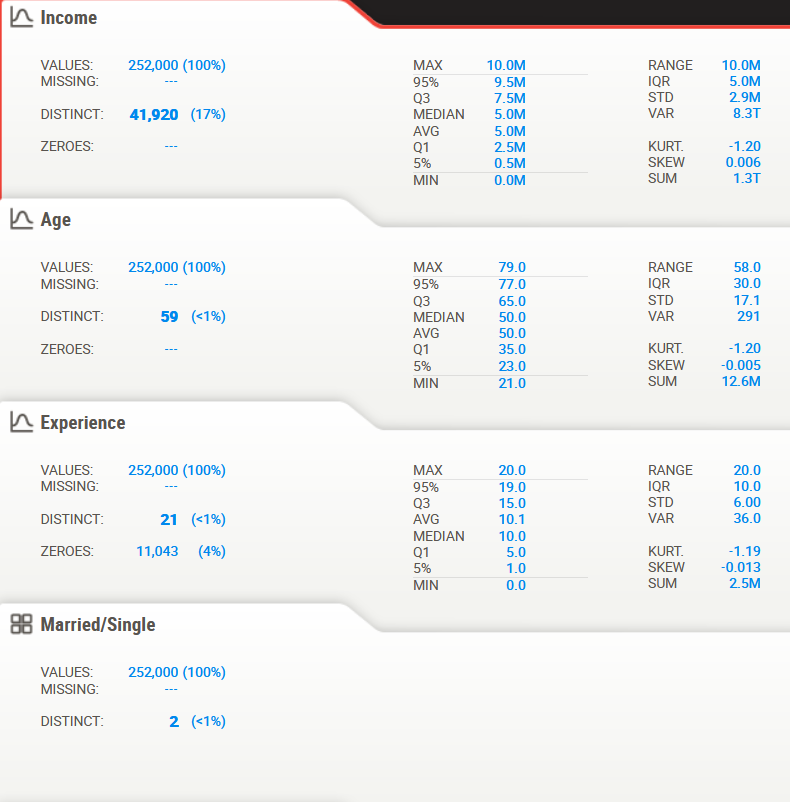
**AUTO EDA:**

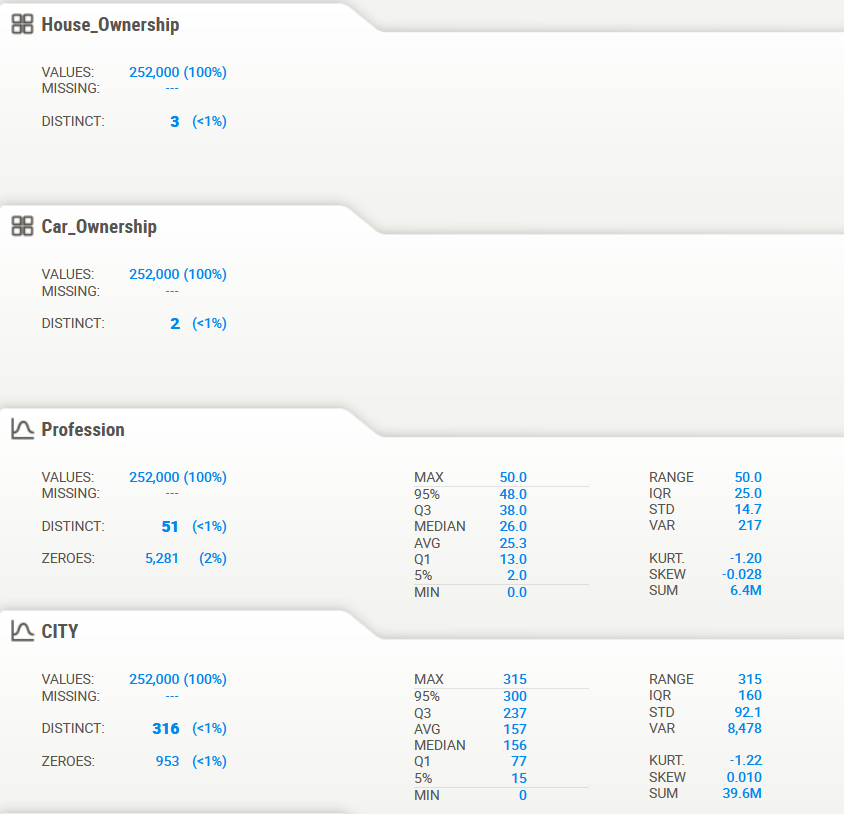
To gain comprehensive insights into the dataset, we employed the Sweetviz library, which enabled us to perform automated exploratory data analysis (EDA).

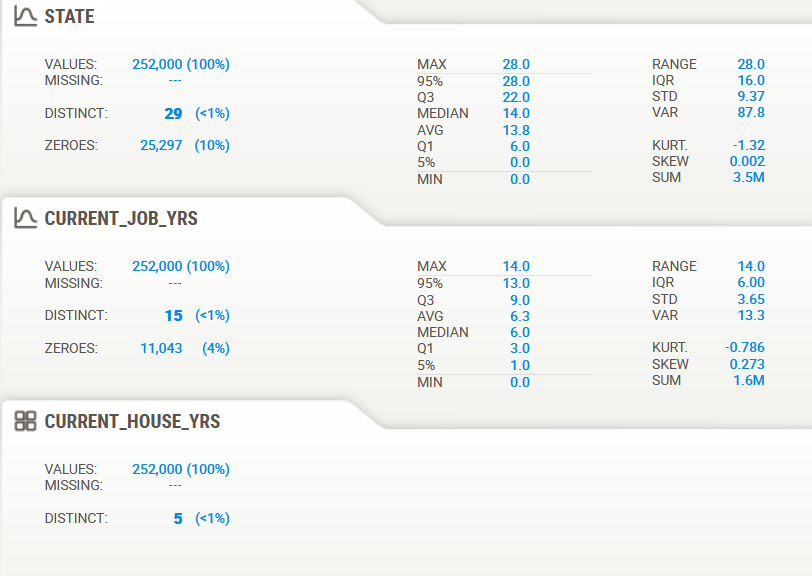
Sweetviz is an open-source Python library that generates beautiful, high-density visualizations to kickstart Exploratory Data Analysis (EDA) with just two lines of code. It produces a fully self-contained HTML application that allows you to interactively explore your data and gain insights quickly.

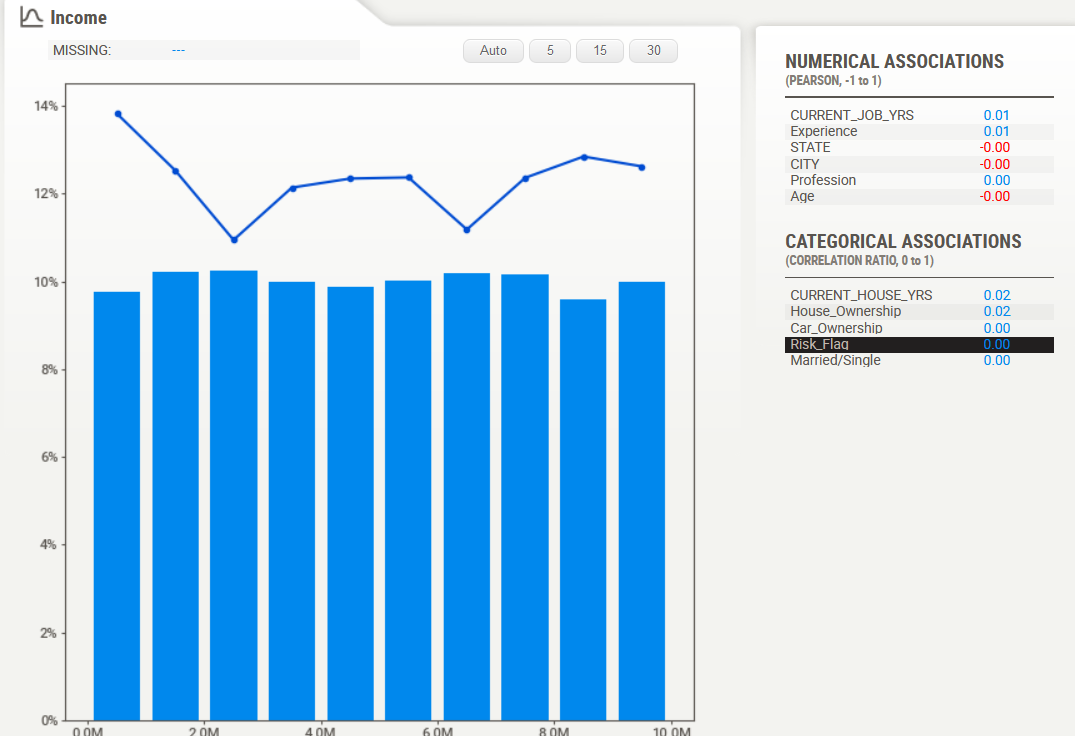
Below are the graphs obtained by performing EDA using Sweetviz library:

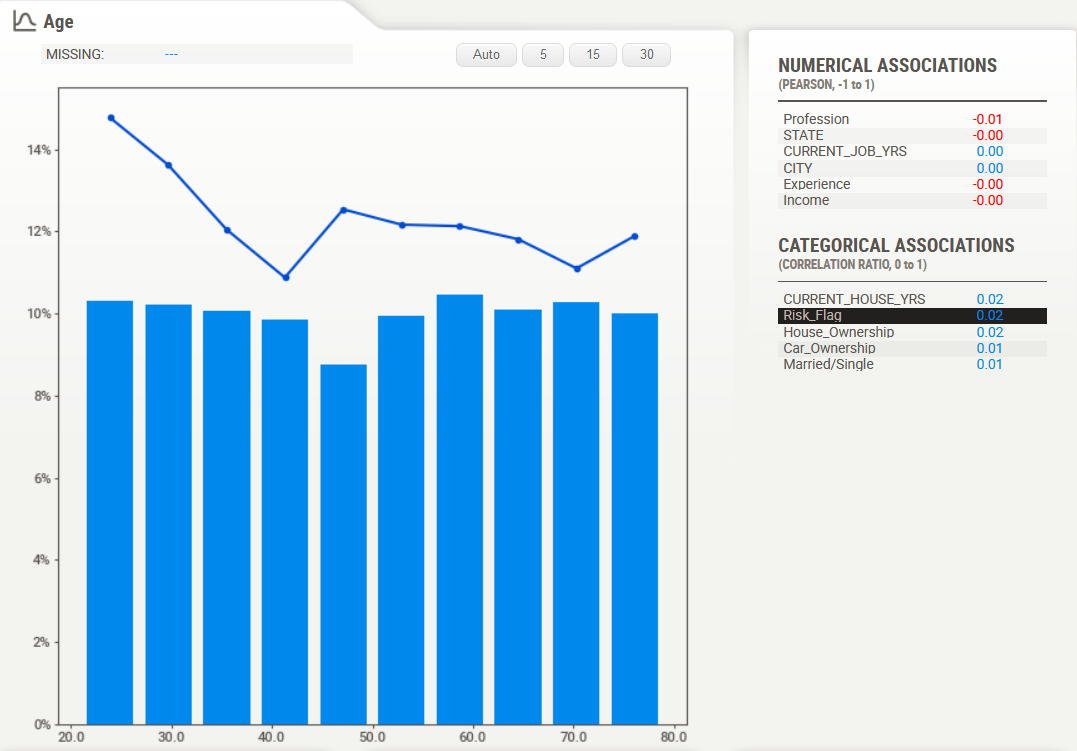


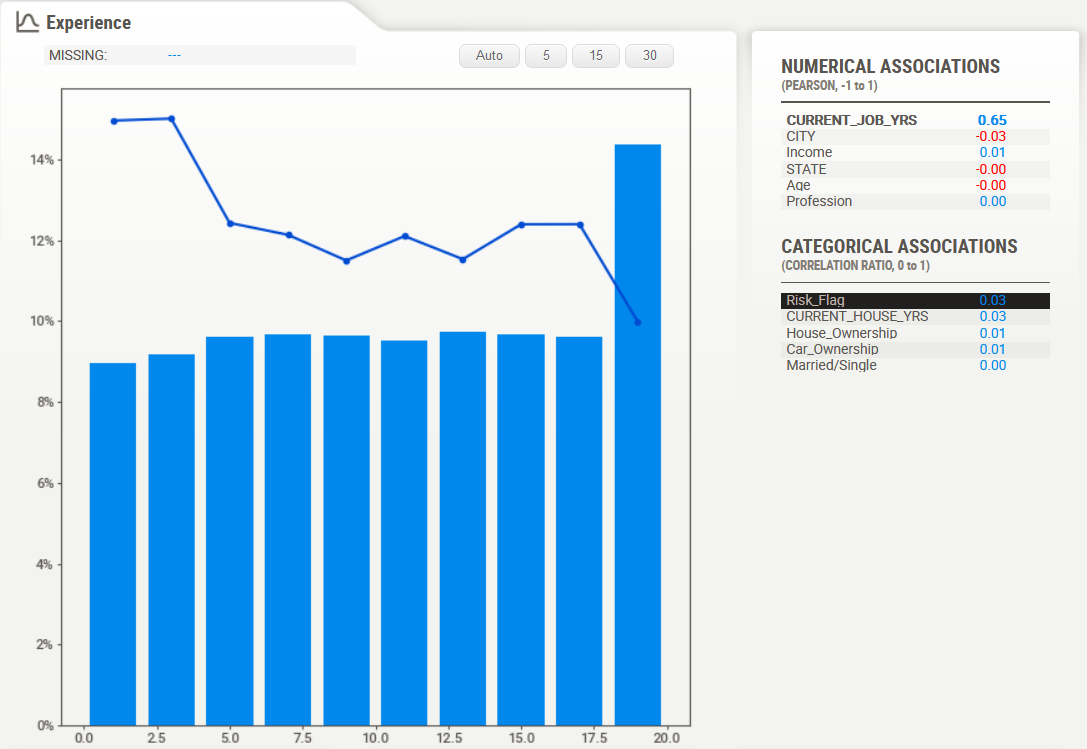


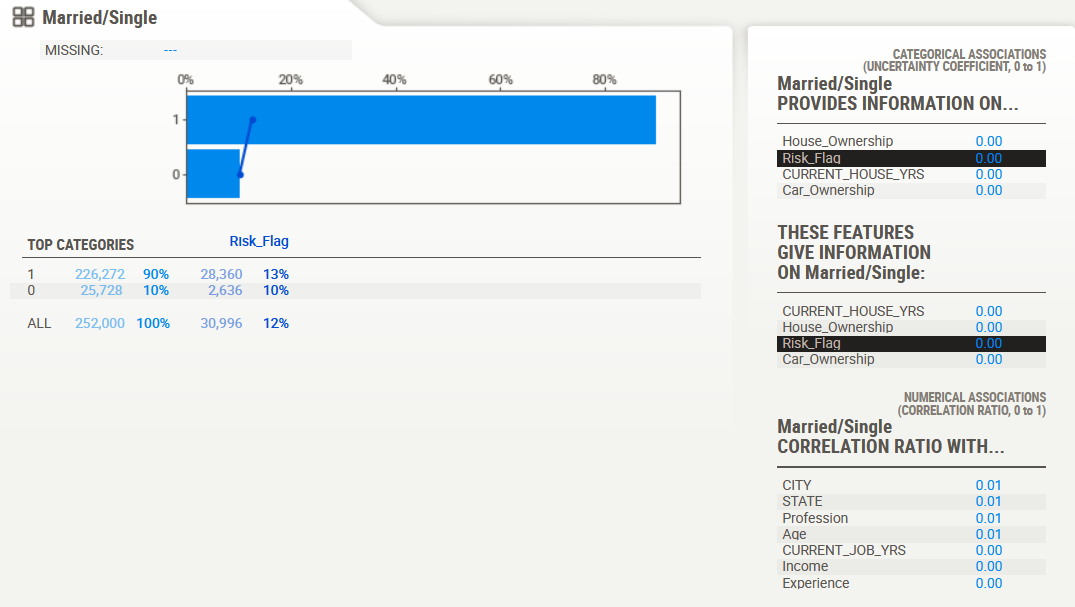


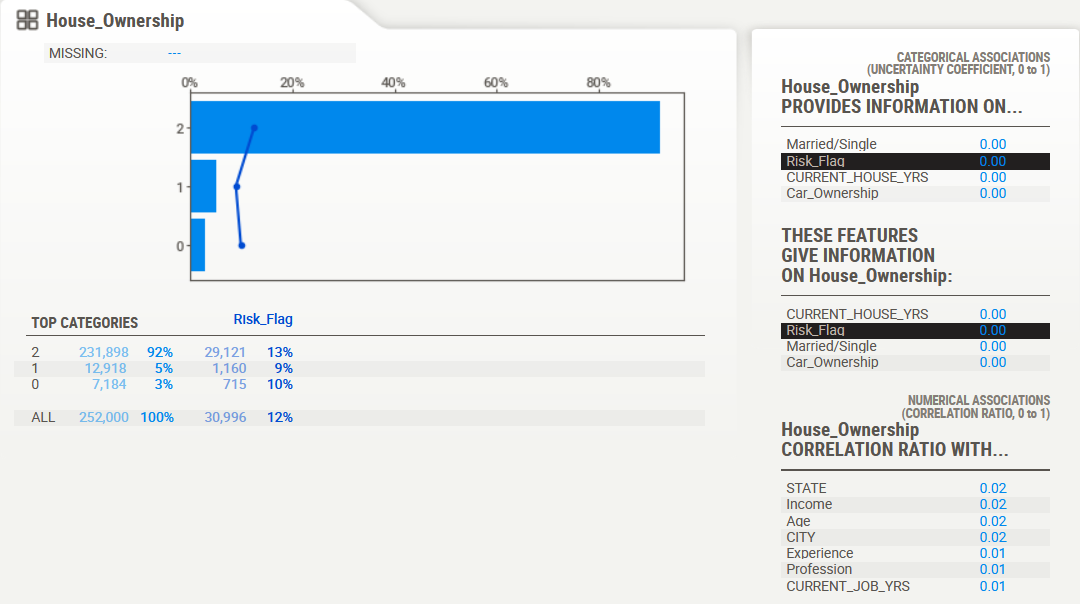


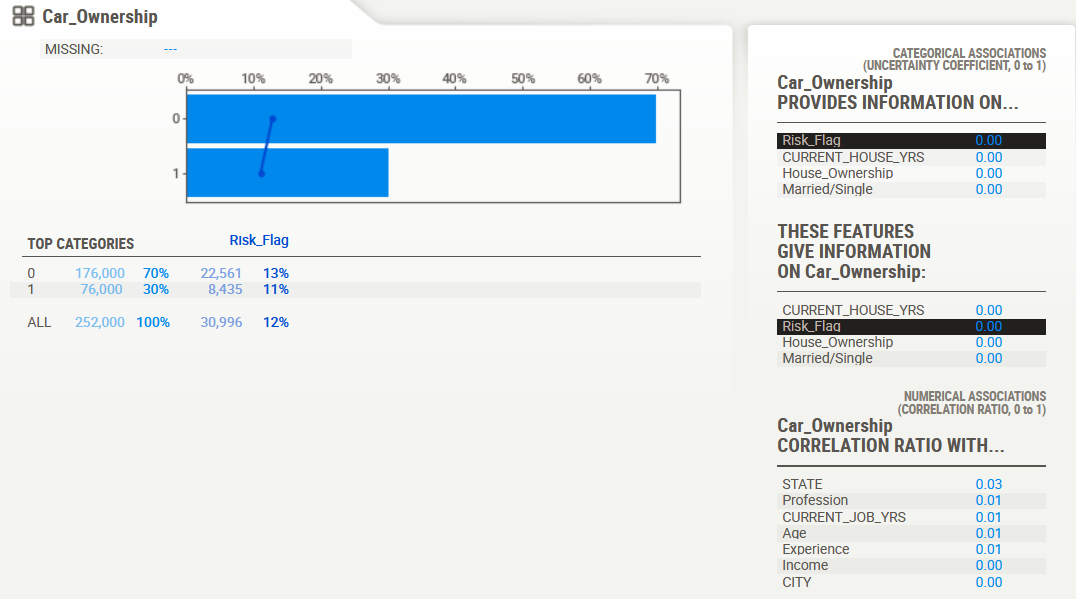


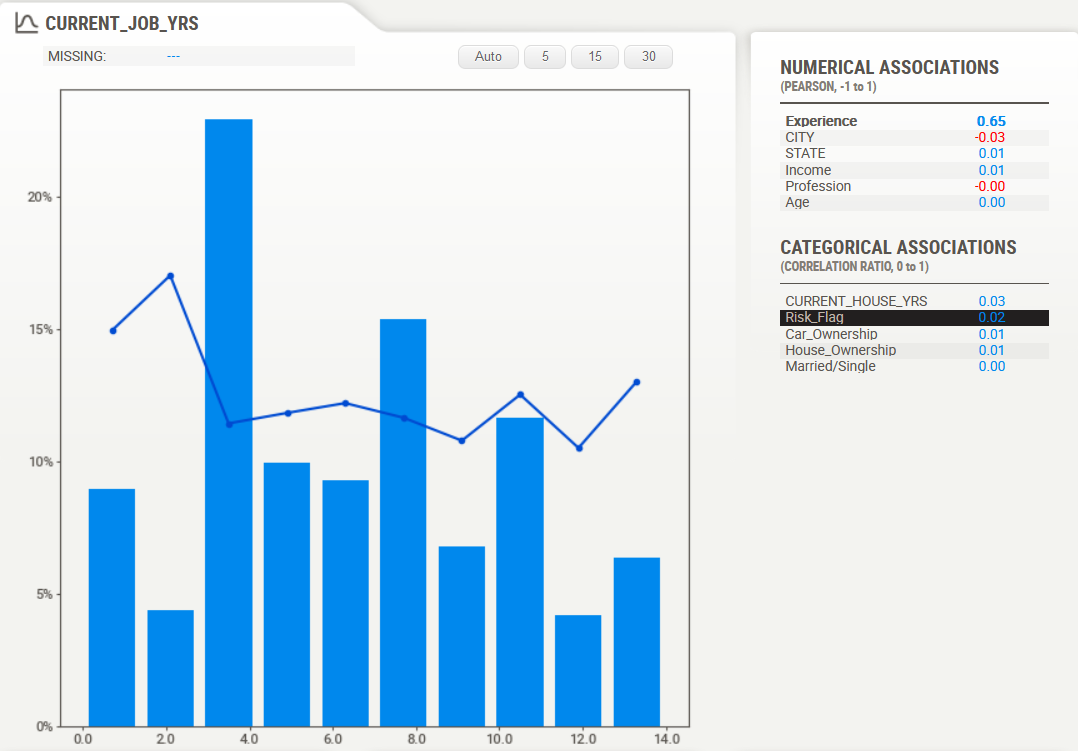


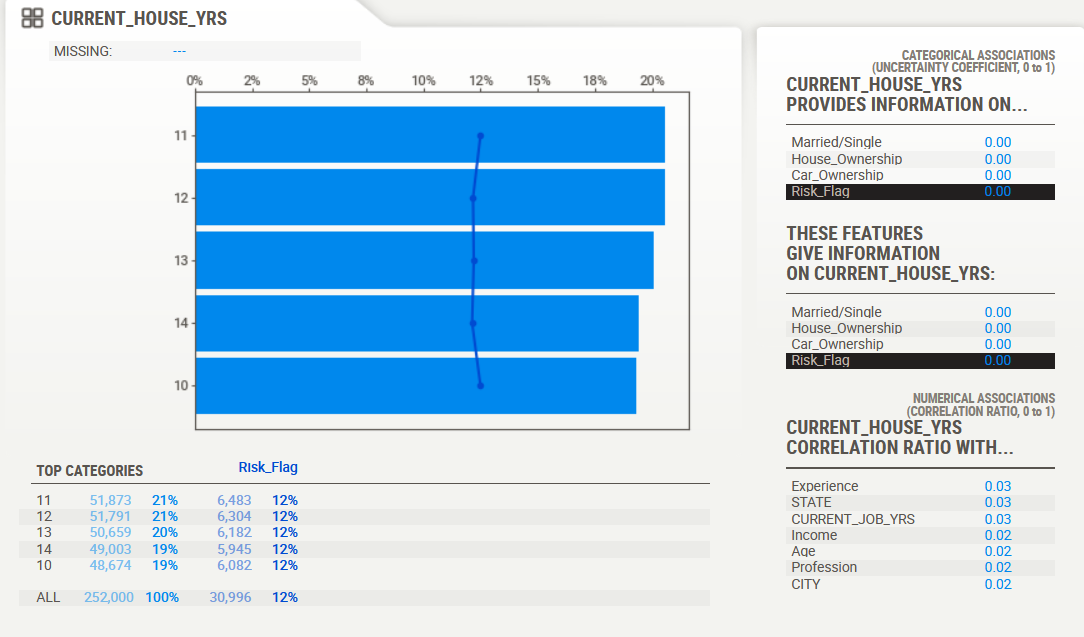












The conclusions drawn from the above graphs are :

1. The dataset is devoid of missing values and outliers, ensuring the integrity of the data for analysis and model development.
2. The zeroes observed in the dataset do not represent errors but rather encoded values for a particular string value. This encoding technique ensures compatibility with subsequent analysis and modelling steps.
3. Distinct, Mathematical and statistical values have also been mentioned in the above images for each feature or column in the given dataset.
4. The association between each feature and the target variable is visualized using appropriate graphical techniques, providing insights into the relationships between variables and facilitating informed decision-making.
5. Based on the analysis, the income group between 0.0M and 1.0M exhibits the highest risk of default, while the income group between 6.0M and 7.0M exhibits the lowest risk.
6. The age group between 21 and 26 years presents the highest risk of default, while the age group between 39 and 43 years presents the lowest risk.
7. The experience group between 0 and 4 years demonstrates the highest risk of default, while the experience group between 18 and 20 years demonstrates the lowest risk.
8. Married individuals exhibit a higher risk of default compared to single individuals.
9. Customers with car ownership demonstrate the lowest risk of default.
10. The risk of default increases for customers who do not own a house, live in a rented house, and own a house, respectively.

**MODEL TRAINING:**

Following a rigorous data cleaning process, we transformed categorical values using label encoding, standardized the data using StandardScaler, and applied SMOTE to address the imbalanced class distribution. These meticulous steps ensured that the dataset was meticulously prepared for the subsequent development of machine learning models.

To effectively train and evaluate the models, we split the data into two partitions: 80% for training and 20% for testing. This standard practice enabled us to assess the generalizability of the models and identify potential areas for improvement.

To effectively explore the predictive capabilities of various machine learning algorithms, we employed a diverse selection of models on the training dataset. This comprehensive approach allowed us to identify the models that best captured the underlying patterns and relationships within the data. The models utilized included:

1. **Random Forest Classifier:**   
   Random Forest, a powerful ensemble learning algorithm, employs a collection of decision trees to generate predictions. Each decision tree is trained on a random subset of the data, with the final prediction determined by aggregating the predictions of individual trees. This approach mitigates overfitting and enhances robustness to data variations. Random forest excels in both classification and regression tasks and effectively handles high-dimensional data.
2. **Decision Trees:**

Decision trees, powerful machine learning algorithms, utilize a tree-like structure to classify or predict continuous values. They recursively partition the data into smaller subsets based on decision rules, leading to predictions for each data point.

Constructing a decision tree involves data preparation, root node selection, recursive splitting, and leaf node creation. Data preparation ensures data quality, root node selection identifies an optimal feature for splitting, recursive splitting divides data into branches based on chosen features, and leaf node creation generates predictions based on the majority class or mean value.

1. **Logistic Regression:**   
   Logistic regression stands as a cornerstone of statistical modelling and is widely employed in machine learning for binary classification tasks. It leverages the logistic function to convert linear combinations of input features into probabilities between 0 and 1, representing the likelihood of belonging to a specific class. The model assumes a linear relationship between the input features and the logit, the logarithm of the odds ratio for the positive class. Parameter estimation techniques, such as maximum likelihood estimation, are utilized to determine the model parameters that best capture the underlying patterns in the data. For classification, the weighted sum of input features is calculated, and the logistic function is applied to determine the probability of belonging to the positive class. A threshold, typically set at 0.5, is employed to classify the data point based on the probability.
2. **Extra Trees Classifier:**  
   Extra Trees Classifier, a robust and versatile ensemble learning algorithm, excels in classification tasks. It constructs a collection of decision trees, each trained on a random subset of the data and utilizing random feature selection and split values, leading to a diversified and uncorrelated forest. Predictions are generated by majority vote among the individual trees. Extra Trees Classifier's robustness to overfitting, ability to handle high-dimensional data, and non-requirement for feature scaling make it a valuable tool for various classification problems.
3. **Gradient Boosting Classifier:**  
   Gradient Boosting Classifier, a robust and versatile ensemble learning algorithm, combines multiple weak learners, typically decision trees, to create a strong predictor. It operates iteratively, constructing new decision trees focused on reducing the prediction errors of the previous model. This process continues until a stopping criterion is met, ensuring optimal performance and minimizing overfitting. Gradient Boosting Classifier's ability to handle high-dimensional data and provide feature importance estimates makes it a valuable tool for various classification and regression tasks.
4. **LGBM Classifier:**  
   LightGBM is a gradient boosting framework that employs Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB) techniques to effectively handle large-scale data while maintaining accuracy, resulting in faster training and reduced memory consumption. Its key features include rapid training speed, lower memory usage, enhanced accuracy, support for parallel and GPU learning, and the ability to handle large datasets with millions of rows and thousands of features. These attributes make LightGBM a powerful and versatile machine learning algorithm suitable for a wide range of applications.
5. **XGB Classifier:**  
   XGBoost, an abbreviation for Extreme Gradient Boosting, is a powerful and widely used machine learning algorithm that efficiently and scalably builds an ensemble of decision trees. Unlike traditional gradient boosting algorithms, XGBoost employs several optimization techniques to improve both performance and efficiency. These techniques include regularization, approximate learning, and parallel processing, enabling XGBoost to handle large datasets with high accuracy and computational efficiency.
6. **CatBoost Classifier:**  
   CatBoost stands out as a robust gradient boosting library that leverages decision trees for classification and regression tasks. Its distinctive feature is the employment of symmetric trees, ensuring balance and preventing overfitting. This approach, coupled with ordered encoding of categorical features, gradient-based sample weighting, regularization techniques, and early stopping, contributes to CatBoost's efficiency and improved accuracy. These advantages make CatBoost a powerful and versatile machine learning algorithm suitable for a diverse range of applications. CatBoost assigns different weights to data points based on their importance, focusing more on those that contribute significantly to the overall loss. CatBoost implements an early stopping mechanism that halts the training process when further iterations no longer improve the model's performance.
7. **AdaBoost Classifier:**

AdaBoost stands out as an effective ensemble machine learning algorithm that harnesses multiple weak classifiers to construct a robust classifier. Its iterative approach involves sequentially training weak classifiers and adjusting their weights based on their performance, ensuring that the final classifier exhibits a lower error rate than its individual constituents. This adaptive nature, coupled with its robustness to noise and interpretable nature, makes AdaBoost a valuable tool for tackling a wide range of classification and regression tasks, including spam filtering, fraud detection, image classification, search engine ranking, recommender systems, and stock price prediction.

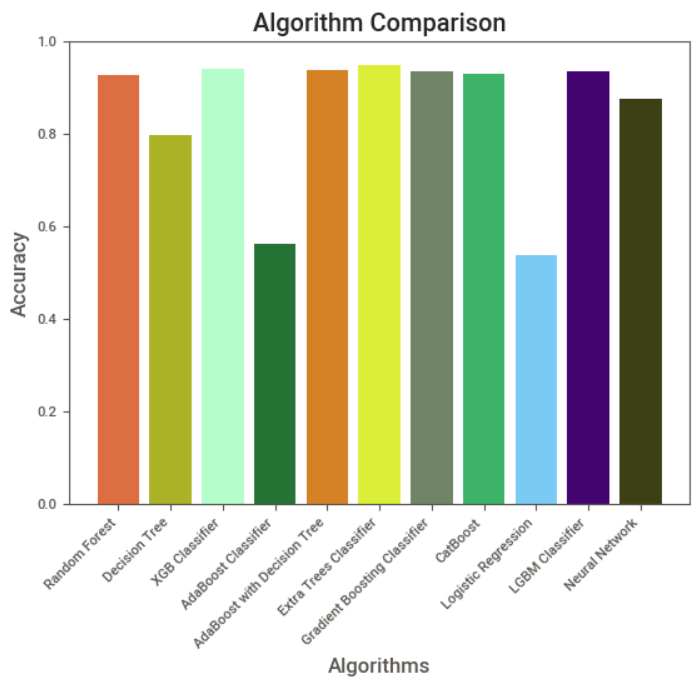
1. **AdaBoost with Decision Trees:**  
   AdaBoost frequently employs decision trees as its base classifiers due to their simplicity, interpretability, and computational efficiency. These characteristics align well with AdaBoost's objective of combining multiple weak classifiers to create a strong classifier. During the training process, AdaBoost iteratively trains decision trees, adjusting their weights based on their performance. Misclassified examples are assigned higher weights, guiding the subsequent decision trees to focus on those challenging instances. The final prediction is determined by a weighted vote of the individual decision trees. This approach has proven effective in a variety of applications, including classification, ranking, and regression.
2. **Artificial Neural Networks:**  
   Artificial neural networks (ANNs), inspired by the human brain's structure and function, are powerful machine learning algorithms that excel in pattern recognition and complex data analysis. ANNs consist of interconnected layers of processing units called neurons, resembling neural connections in the brain. Each neuron receives, processes, and transmits signals to its connected neurons. ANNs operate by preparing data, defining the network architecture, initializing neuron weights, propagating signals forward, calculating errors, adjusting weights through backpropagation, and iterating until a satisfactory error level or predefined iteration limit is reached. ANNs' advantages include non-linearity, feature learning, pattern recognition, and scalability.

**MODEL TESTING:**

Following the training of the aforementioned models using the training dataset, their performance was evaluated on the testing dataset. Accuracy, precision, recall, F-score, and AUC score were employed as the evaluation metrics. These metrics provide a comprehensive assessment of the models' ability to correctly classify the data. The below table shows it:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model Name** | **Accuracy** | **Precision** | **F1-Score** | **Recall** |
| Random Forest Classifier | 0.927 | 0.899 | 0.929 | 0.961 |
| Decision Trees | 0.842 | 0.788 | 0.855 | 0.935 |
| Logistic Regression | 0.534 | 0.533 | 0.547 | 0.561 |
| Extra Trees Classifier | 0.950 | 0.916 | 0.952 | 0.911 |
| Gradient Boosting | 0.934 | 0.911 | 0.936 | 0.963 |
| LGBM Classifier | 0.934 | 0.908 | 0.936 | 0.965 |
| XGB Classifier | 0.941 | 0.916 | 0.943 | 0.971 |
| CatBoost | 0.927 | 0.905 | 0.930 | 0.956 |
| AdaBoost Classifier | 0.565 | 0.562 | 0.574 | 0.585 |
| AdaBoost with Decision Trees | 0.937 | 0.906 | 0.939 | 0.974 |
| Artificial Neural Networks | 0.878 | 0.860 | 0.881 | 0.904 |

**GRAPH ON MODEL ACCURACY COMPARISION:**

****

From the above graph and A thorough evaluation of the classification algorithms revealed that Extra Trees Classifier emerged as the most promising and well-suited approach for predicting loan risk. Its remarkable accuracy of approximately 95% outperformed the other considered models, making it a compelling choice for this critical task.

**FINALIZED MODEL DETAILS:**

**DEPLOY:**

**CLOUD:**