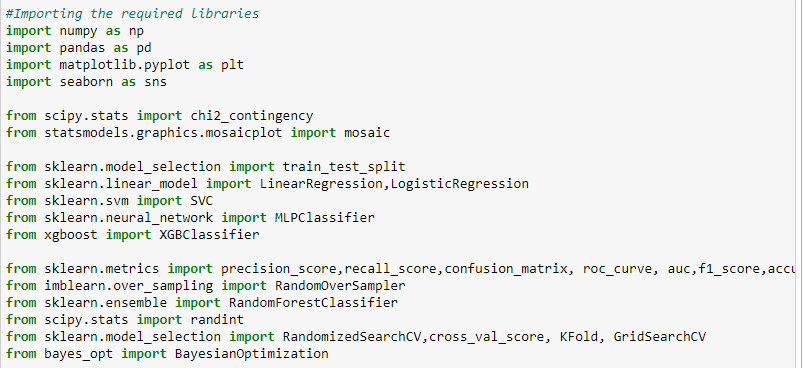
**German Credit Risk- With Target**

# Created by Kaushik Kar

# Employment id- 2216027

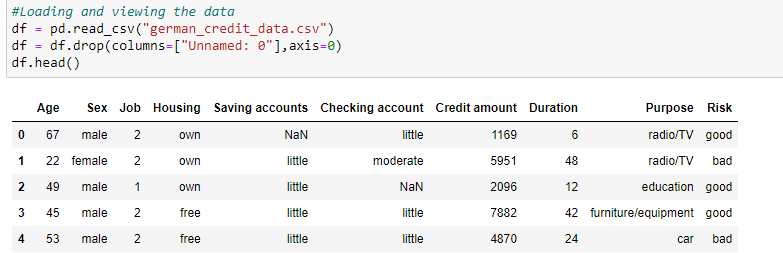
# Importing Libraries:

First, importing the important external Python packages using the pip package manager



1. NumPy is used for mathematical operations like addition, subtraction, multiplication, division, etc. on arrays and matrices.
2. Pandas provides data structures for efficiently storing and manipulating large datasets, and tools for reading and writing data to and from various file formats, including CSV, Excel, and SQL databases.
3. scipy.stats is a library for statistical functions and tests, including the chi2\_contingency function for performing chi-squared tests on contingency tables.
4. statsmodels.graphics.mosaicplot is a module for plotting mosaic plots.
5. sklearn (short for scikit-learn) is a popular machine learning library in Python. You have imported several modules from it, including train\_test\_split for splitting data into training and testing sets, LinearRegression and LogisticRegression for linear regression and logistic regression models, SVC for support vector machine models, MLPClassifier for multi-layer perceptron neural network models, and RandomForestClassifier for random forest models. You have also imported several metrics for evaluating models, including precision\_score, recall\_score, confusion\_matrix, roc\_curve, auc, f1\_score, and accuracy\_score. In addition, you have imported RandomOverSampler for oversampling imbalanced data, and RandomizedSearchCV, GridSearchCV, and KFold for hyperparameter tuning and cross-validation.
6. xgboost is a library for gradient boosting models, and you have imported XGBClassifier for using XGBoost models.
7. bayes\_opt is a library for Bayesian optimization, which can be used for hyperparameter tuning.
8. Seaborn is a data visualization library based on Matplotlib which is a plotting library used for creating static, interactive, and animated visualizations in Python.
9. **Uploading data and then analysis and preprocessing:**

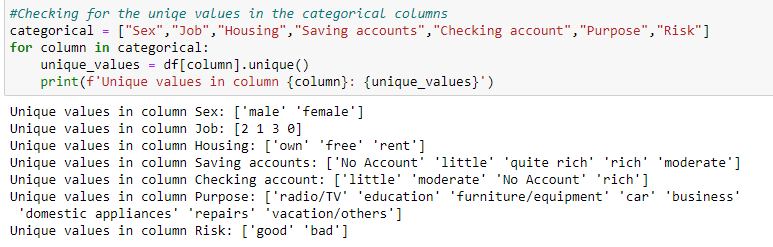
With the help of Pandas library, we can read and upload the data in csv form.



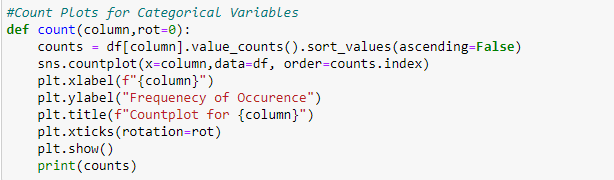
Now for the data preprocessing and analysis part firstly check whether there is any null value is present or not and as we can see in the notebook that there is null values in the data and we can fil those null values with the code .fillna and we can convert those null values in to “no account”.

Then for the description part in the data we can find the statistical information of the data such as mean,std min max values etc.

* Now for the Categorical columns we can first take out all the categorical columns from the data and then with the help of the for loop we can see all the unique values from every categorical column

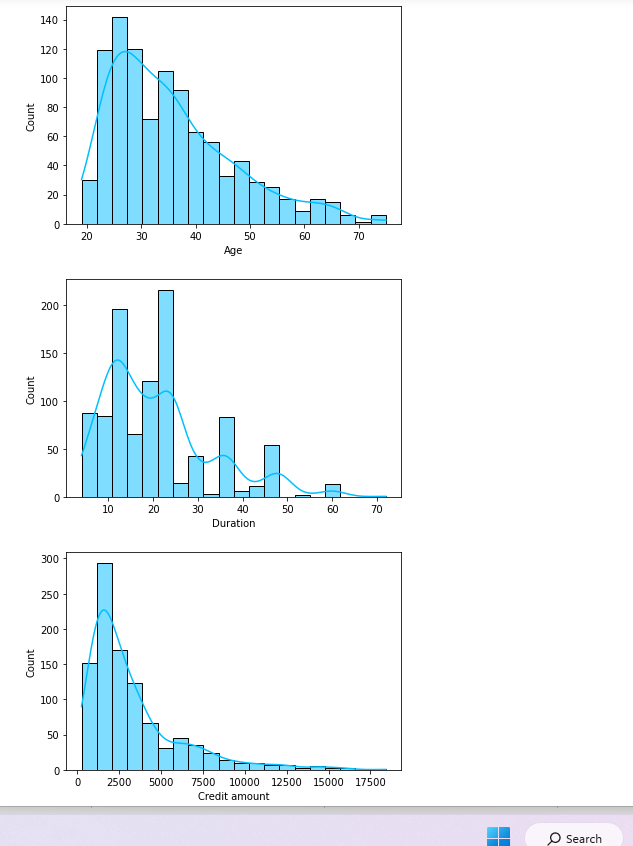


Now for the Univariate Analysis we want to visualize each categorical columns with its categories with the help of the for loop. As you can see in the figure example as “purpose”.



In the Risk Categorical Variable plots provide a distribution of the different classes and the target feature. Since the number of good credits are 70% of the dataset, we say that we have an imbalanced dataset.

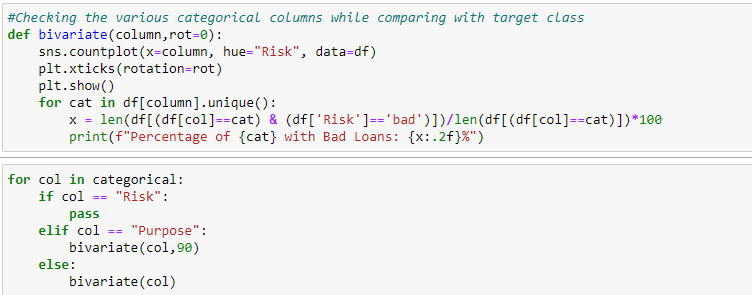
Also make the distribution plots for the numerical columns.



We see that all three distribtuions have a right skew.

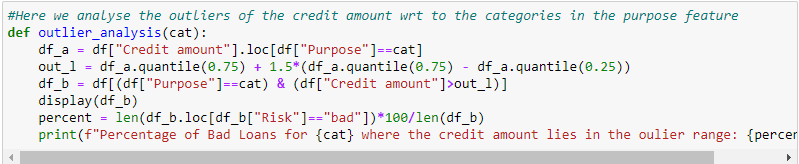
**Bivariate analysis** in Python involves analyzing the relationship between two variables in a dataset. It helps to understand how two variables are related and whether there is a cause-and-effect relationship between them.

Checking the various categorical columns while comparing with target class using the for loop.



**Outlier** is an observation that lies an abnormal distance from other values in a random sample from a population. Outliers can be caused by measurement or recording errors, or they can represent genuine but extreme values in the population being studied.

Here we analyse the outliers of the credit amount wrt to the categories in the purpose feature.

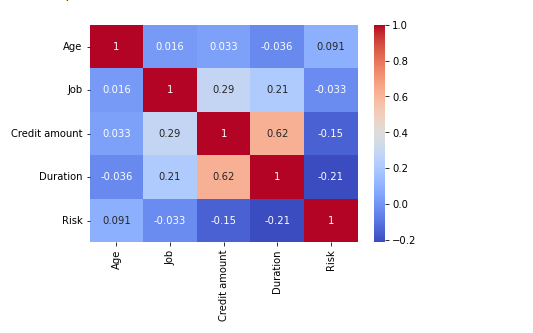


this function provides a useful approach to identifying and analyzing outliers in the credit dataset for a given categorical variable Such as radio/TV, car, Furniture/equipment.

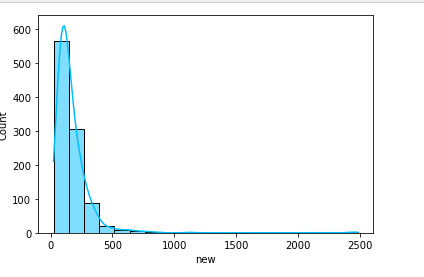
**Feature Engineering:**

Feature engineering is the process of creating new features or variables from existing data that can improve the performance of a machine learning model. It involves transforming raw data into a format that is more suitable for analysis and modelling.

The Categories in the Risk variable good and bad replaced by 0 and 1 and the we can visualize the correlation matrices through heatmap.

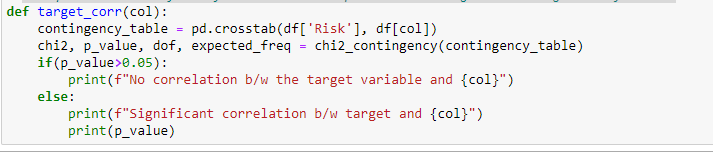


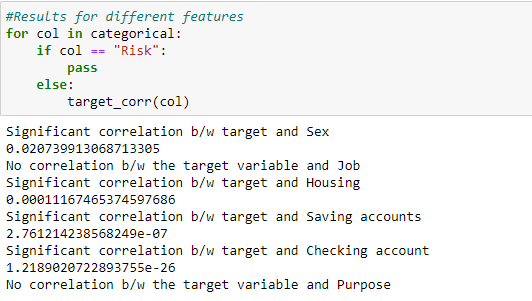
Trying to build a new variable which combines credit amount and duration as they are highly correlated and plot this new variable



**Chi-square test:** Chi-square test is a statistical hypothesis test that is used to determine whether there is a significant association between two categorical variables. The test is based on the difference between the observed frequencies of the two variables and the expected frequencies, assuming that there is no association between them.

Chi-Squared test to find if correlation is present b/w target and categorical features.

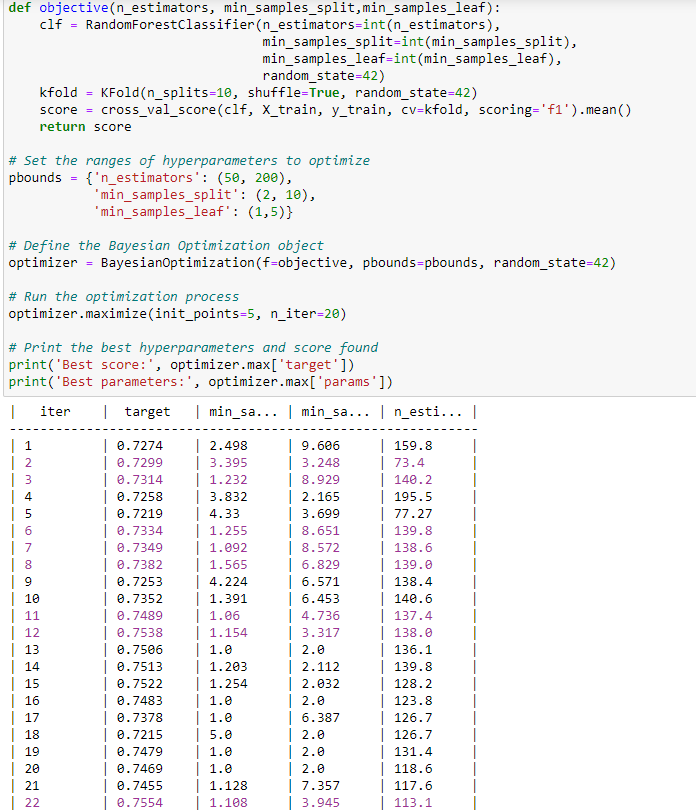




Same thing with the independent categorical variables

**Modelling:**

Random forest is a machine learning algorithm used for classification, regression, and other tasks that involve predicting a target variable from a set of input features. It is an ensemble learning method that combines the predictions of multiple decision trees to produce a more accurate prediction.

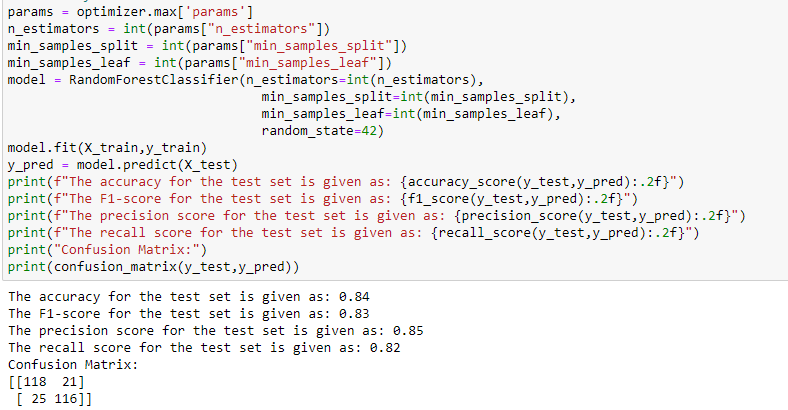


This code performs hyperparameter tuning using Bayesian optimization for a random forest classifier. The objective function takes the hyperparameters as input and returns the F1 score of the model trained using these hyperparameters, evaluated using cross-validation. The hyperparameters to optimize are the number of estimators, the minimum number of samples required to split an internal node, and the minimum number of samples required to be at a leaf node.

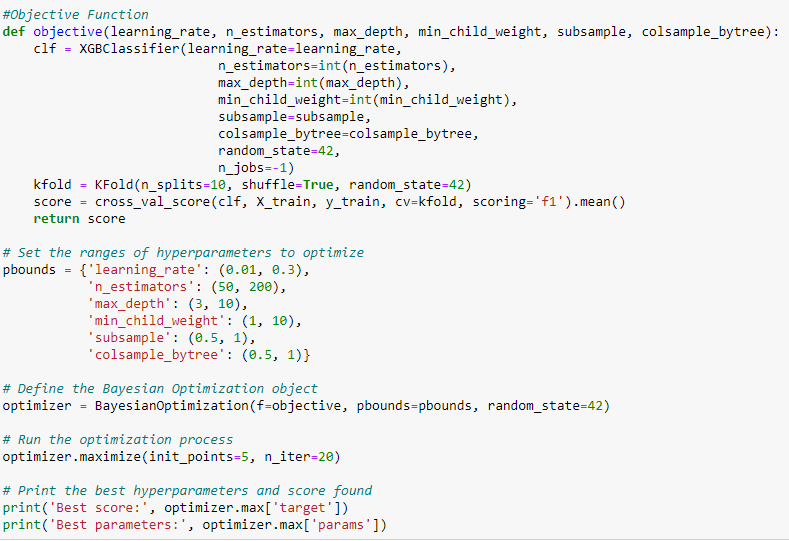
The ranges of hyperparameters to optimize are set using the pbounds dictionary, and the BayesianOptimization object is initialized with the objective function and pbounds. The optimization process is then run for a total of 25 iterations, with 5 initial random points and 20 subsequent iterations using the Gaussian process model.

Finally, the best hyperparameters and score found are printed. The best hyperparameters correspond to the maximum score found by the optimizer during the optimization process. These hyperparameters can be used to train the final model on the entire training dataset and evaluate its performance on a separate test dataset.

Now performance on the test set



**XGBoost:** XGBoost (Extreme Gradient Boosting) is a powerful and popular gradient boosting framework that is used for both classification and regression problems. It is an extension of the classic Gradient Boosting framework and uses a more regularized model formulation to prevent overfitting and improve generalization.

****

The code defines the BayesianOptimization object, which uses the objective function and pbounds to search for the optimal hyperparameters using Bayesian optimization. The maximize method runs the optimization process for a specified number of iterations, starting with a set of randomly chosen points (specified by init\_points).

Finally, the best hyperparameters and score found by the optimizer are printed. These hyperparameters can be used to train the final XGBoost model.

Model performance on the test set

