**Machine Learning Project Report**

**Problem Statement:** We have two datasets (Financial Risk Assessment) and (Credit Card Fraud Detection) and we want to apply machine learning algorithms on these datasets, Choosing the perfect algorithms that will fit perfectly on them.

Financial Risk Assessment: This dataset concerns predicting financial risk values as a function of a few financial features. Regression models are required to predict values that are in a continuous scale and our goal is to determine the most suitable algorithms in estimating these values with the least amount of error.

Credit Card Fraud Detection: This dataset deals with binary classification of credit card transactions as fraudulent or legit. It requires classification models and has the problem associated with class imbalance, where fraudulent transactions are the minority.

**Financial Risk Assessment Dataset**

1. **Data Exploration**

**Dataset Overview:**

* Number of samples: N (2000,37 ) rows & columns.
* Missing Data : No missing data.
* Duplicate Entries: No Found duplicates.

**Visualizations:**

* In RiskScore we found 94 outliers , by calculating the Interquartile Range(IQR).

1. **Data Preprocessing:**

* **Feature Scaling:** Quantitative features were scaled using MinMaxScaler function.
* **Encoding Categorical Features:** Converting qualitative features like (EmploymentStatus, EducationLevel) into numerical values using LabelEncoder function .
* **Feature Selection:** Removed irrelevant features (ApplicationDate) for cleaner modeling and better evaluation.

1. **Model Selection and Implementation:**

* **Linear regression :** Predict the target variable and evaluate the model's performance using (MSE), (MAE), and (RMSE).
* **Ridge regression :** Regularized linear model tested multiple alpha values to reduce overfitting.
* **KNN regressor :** tested various k values ranging from 1 to 15 to evaluate performance.

**Implementation Details:**

* Data split into training and testing sets (80% training , 20%testing).
* Standardized evaluation metrics (MSE, RMSE, R², MAE) were used for comparison

**Rationale:** Linear Regression and Ridge Regression are the better models. They both have very similar performance metrics with low MSE, RMSE, MAE, and high R2 scores.KNN Regression performs poorly in all metrics, with higher errors and a negative R2 score.

linear regression is more simple while ridge regression handles complexities better.

1. **Model Evaluation:**

**1-Linear Regression:**

**MSE\_Train: 3.7263158256775832**

**MSE\_Test: 4.058476108101882**

**RMSE\_Train: 1.9303667593692093**

**RMSE\_Test: 2.014565985045385**

**R2\_Train: 0.937968712725555**

**R2\_test: 0.9346329174290001**

**MAE\_Train: 1.4991649158556601**

**MAE\_Test: 1.5267375873663556**

**MSE\_TEST VALUES: [110.48151, 85.036155, 74.50665444444444, 70.158513125, 66.6930768, 65.2457547222222, 63.71250571428571, 62.476671874999994, 61.50113172839505, 60.78320539999999, 60.24913933884297, 59.54551687499999, 59.06166591715976, 58.7426731632653]**

**2-Ridge Regression:**

**MSE\_TRAIN\_RIDGE & MSE\_TEST\_RIDGE : 3.7267465238807334 4.056809177904589**

**R2\_SCORE\_TRAIN\_RIDGE & R2\_SCORE\_TEST\_RIDGE : 0.9379615429725824 0.9346597655268931**

**MAE\_TRAIN\_RIDGE & MAE\_TEST\_RIDGE 1.4998193507228634 1.5275398278699939**

**3-KNN Regression:**

**MSE\_TEST VALUES: [110.48151, 85.036155, 74.50665444444444, 70.158513125, 66.6930768, 65.2457547222222, 63.71250571428571, 62.476671874999994, 61.50113172839505, 60.78320539999999, 60.24913933884297, 59.54551687499999, 59.06166591715976, 58.7426731632653]**

**MSE\_Train: 46.4598726171875**

**MSE\_Test: 62.476671874999994**

**MAE\_Train: 5.365245312499999**

**MAE\_Test: 6.239862499999999**

**R2\_Train: 0.22659113186498958**

**R2\_Test: -0.006268771931839279**

**Fraud Detection Dataset**

**Data Exploration:**The dataset contains numerical features derived from credit card transactions.Initial analysis revealed a heavy imbalance in the Class target variable, with fraudulent cases being significantly underrepresented.

* **Visualization:** We used Histograms, Box plots, Scatter plots, and Correlation

**Data Preprocessing:**

* **Class Imbalance Handling:**

Random Under-Sampling: The data was unbalanced due to then bias in (Class) column that had more zeros than one’s , so we had to use ( Under Sampling Method ) that uses Random Under Sampler function that makes the minority (1) equal to the majority (0) by increasing the ones classes randomly to be equal to the zeros.

* **Data Cleaning:**

Removed duplicate and missing values to ensure data quality.

* **Feature Scaling:**

Applied StandardScaler to standardize features for improved model performance.

**Model Selection and Implementation:** We used **LogisticRegression** and **KNN regressor**  models because they will work perfectly with this datasets because it depends on Classification

* **Logistic Regression :**

-Chosen for its simplicity and interpretability in classification tasks.

-First we imported the sklearn library to import logistic regression function and then we identified the ( X train and test data , Y train and test data ) that the test size will take 20% of the main data, and then we fitted the X and Y train to model so it can learn properly and then we identified the Y predicted using the X test model, and then we got the accuracy of the model using the Y test and Y predicted to see how much is the model accurate.

* **KNN :**

-Selected to compare performance against a non-parametric method.

-First we imported the sklearn library to import KNeighborsClassifier function and, and then we identified the model and set the K to be 5, then we identified the ( X train and test data , Y train and test data ) that the test size will take 20% of the main data, and then we fitted the X and Y train to model so it can learn properly and then we identified the Y predicted using the X test model, and then we got the accuracy of the model using the Y test and Y predicted to see how much is the model accurate

* **Rationale**:

-Logistic Regression is effective for binary classification with linear relationships.

-KNN can capture complex patterns but requires careful scaling and tuning for optimal performance.

**Model Evaluation:**

**1-Logistic Model Evaluation:**

**Accuracy: 0.9441624365482234**

**Precision: 1.0**

**Recall: 0.8962264150943396**

**F1-Score: 0.945273631840796**

**Confusion Matrix:**

**[[91 0]**

**[11 95]]**

**Classification Report:**

**precision recall f1-score support**

**0 0.89 1.00 0.94 91**

**1 1.00 0.90 0.95 106**

**accuracy 0.94 197**

**macro avg 0.95 0.95 0.94 197**

**weighted avg 0.95 0.94 0.94 197**

**2-KNN Model Evaluation:**

**Accuracy: 0.8984771573604061**

**Precision: 0.9886363636363636**

**Recall: 0.8207547169811321**

**F1-Score: 0.8969072164948454**

**Confusion Matrix:**

**[[90 1]**

**[19 87]]**

**Classification Report:**

**precision recall f1-score support**

**0 0.83 0.99 0.90 91**

**1 0.99 0.82 0.90 106**

**accuracy 0.90 197**

**macro avg 0.91 0.90 0.90 197**

**weighted avg 0.91 0.90 0.90 197**

**Conclusion:**

**Financial Risk Assessment Dataset:**Ridge regression performed slightly better than linear regression which was expected because it utilizes ridge regularization. KNN performed poorly on the KNN regression. This could be attributed to the dataset used being a complex one.

**Fraud Detection Dataset:** Logistic Regression stands out as a good low bias high variance estimator.KNN performed well but needed careful dependency management through tuning hyperparameters.

**Limitations:**

**Financial Risk Assessment:**

The models are based on a linear relationship between features and the target variable.

If the data is not linear, then the performance of the models may not be as it should have been.

Using techniques which are more advanced and with no linear assumptions can improve the results.

**Fraud Detection Dataset:**

Random under-sampling used in balancing classes can result in the loss of important information belonging to the majority class, which consists of non-fraud cases.

This can reduce the model's capacity for learning from all the available data effectively.