BC3409 Individual Assignment

Q1) How to improve the results through programming by changing the features selected, data wrangling or parameter setting (optional: you may explain how and why you are doing so or you could explain why the result CANNOT be improved).

Improvements Made To Successfully Achieve Higher Accuracy:

A. Features Selected:

Feature	Definition
LIMIT_BAL	Amount of the given credit (NT dollar). Includes both the individual consumer credit and his/her family (supplementary) credit.
SEX	Gender (1 = male; 2 = female).
EDUCATION	Education (1 = graduate school; 2 = university; 3 = high school; 4 = others).
MARRIAGE	Marital status (1 = married; 2 = single; 3 = others).
AGE	Age (year).
PAY_1	Repayment status in September 2005 (where -2 = Balance paid in full and no transactions made in this period -1 = Balance paid in full and account has a positive balance at the end of period due to recent transactions which payment has not yet come due 0 = Paid minimum due amount, but not the entire balance 1 = Payment delay for 1 month, 2 = Payment delay for 2 months, etc.)
PAY_2	Repayment status in August 2005
PAY_3	Repayment status in July 2005
PAY_4	Repayment status in June 2005
PAY_5	Repayment status in May 2005
PAY_6	Repayment status in April 2005
BILL_AMT1	Amount of bill statement in September 2005
BILL_AMT2	Amount of bill statement in August 2005
BILL_AMT3	Amount of bill statement in July 2005
BILL_AMT4	Amount of bill statement in June 2005
BILL_AMT5	Amount of bill statement in May 2005
BILL_AMT6	Amount of bill statement in April 2005
PAY_AMT1	Amount paid in September 2005
PAY_AMT2	Amount paid in August 2005
PAY_AMT3	Amount paid in July 2005
PAY_AMT4	Amount paid in June 2005
PAY_AMT5	Amount paid in May 2005
PAY_AMT6	Amount paid in April 2005

I have decided to include all of the features in the models. Selecting a subset of features could probably improve the results as it could reduce overfitting and improve the models' ability to generalise to new data. However, I did not drop them as from the data visualization, it seems like these columns have some influence on default probability.

For example, people who default have a higher 'PAY_X' (where X = 1, 2, 3, 4, 5, 6) than people who did not default as shown in Figure 1 below.

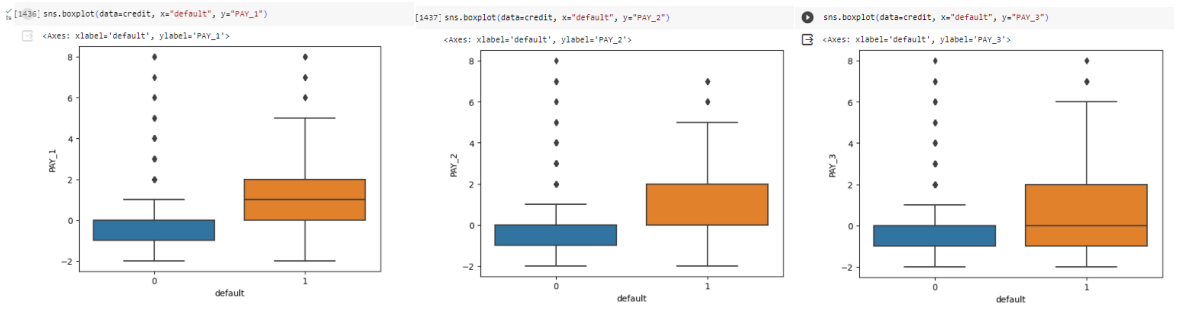

Figure 1

B. Data Wrangling:

Data Wrangling involves transforming raw data into a more usable form.

- **SMOTE Oversampling**: To generate synthetic samples from the minority class instead of duplicating them. It does this by selecting two or more similar data points from the minority class and randomly perturbing their features to create a new synthetic data point. SMOTE is supposed to avoid overfitting problems, but it does risk adding noise to the model.
- **Z-score Normalization**: Normalization is done by transforming the values of each feature to a range of 0 to 1, or to a standard normal distribution with a mean of 0 and a standard deviation of 1. This is done to facilitate data analysis and modeling, and to reduce the impact of different scales on the accuracy of machine learning models.

C. Hyperparameter Tuning

We want to finetune the parameters of the various models to get the best set of parameters. We can do this by using **GridSearchCV**, which gives a parameter space and tests our models on every point of this space. The parameters of the machine learning model are tuned using a technique called cross-validated grid search, which involves trying different values for the parameters and evaluating the performance of the model on a held-out dataset.

Figure 2 shows how GridSearchCV is implemented to find the best set of parameters for each model. The parameter space we want to test for each model is defined in ‘param_grid’. ‘cv’ is the number of folds to use for cross validation. The ‘scoring’ mode will be based on accuracy score. ‘refit’ is set to true so that the model will be using the best found parameters on the whole dataset.

```
[1410] # Logistic Regression with hyperparameter tuning based on accuracy
       param_grid = {'penalty': ['none', 'l2'],
                     'C': [0.1, 1, 10, 100]}
       acc_scorer = make_scorer(accuracy_score)
       grid_logreg = GridSearchCV(linear_model.LogisticRegression(max_iter=2000), param_grid, cv = 5, scoring= acc_scorer, refit=True)
       %time grid_logreg = grid_logreg.fit(X_train, Y_train)
       print(grid_logreg.best_estimator_)
       print(grid_logreg.best_score_)
```

Figure 2

Table 1 shows the best set of parameters found for each model.

Model	**Best set of parameters**
Logistic Regression	LogisticRegression(C=1, max_iter=2000)
Decision Tree	DecisionTreeClassifier(criterion='entropy', max_depth=9, max_leaf_nodes=100, min_samples_split=20)
Random Forest	RandomForestClassifier(criterion='entropy', n_estimators=600, n_jobs=-1)
Gradient Boosting	GradientBoostingClassifier(learning_rate=0.5, n_estimators=200)

Table 1

As for Neural Network model using Keras, GridSearchCV was not used as the search space would be too huge and would take too long to run. Instead, I have increased the epoch size from 10 to 150 and decreased the batch size from 32 to 16 in order to improve the accuracy. Increasing epoch size increases the accuracy as the model continues to refine its understanding of the training data. The parameter setting I used for Keras is: optimizer='adam', loss='binary_crossentropy', metrics=['accuracy', f1_score]

Table 2 shows the comparison of accuracy scores for every model, with and without hyperparameter tuning. As we can see, there is an increase in accuracy score for every model after searching the parameter space and adjusting the parameter settings. We shall refer to the optimized model as the model with hyperparameter tuning.

	Logistic Regression	Decision Tree	Random Forest	Gradient Boosting	Neural Network (Keras)
Without tuning	78.3%	68.4%	81.4%	82.4%	88.5%
With tuning	86.3%	83.6%	87.1%	85.5%	94.1%

Table 2

Table 3 shows the comparison of F1 scores for every model, with and without hyperparameter tuning. Overall, there is an improvement in F1 scores across all models, except Gradient Boosting. This is because the GridSearchCV was set on improving the models based on accuracy score, not F1 score. If we were to set based on F1 score, the F1 score will likely improve for all models.

	Logistic Regression	Decision Tree	Random Forest	Gradient Boosting	Neural Network (Keras)
Without tuning	0%	38%	49%	50%	87.3%
With tuning	47%	50%	50%	47%	93.4%

Table 3

Accuracy is not a good measure of model performance on imbalanced datasets, because it can be misleading. Instead, it is better to use a metric like F1-score, which is more robust to imbalanced datasets. Even if our training set is balanced, the test set may still be imbalanced. Therefore, using F1-score to evaluate the model is a better option.

Comparison Amongst Models:

Precision: Precision tells us the accuracy of positive predictions. It is the fraction of predicted True that are actually True. Having a high precision rate is important for banks because identifying the wrong customers for credit default will lead to a lower credit score of the bank. A bank with a high credit score will be able to borrow money more cheaply and will be more attractive to investors. By accurately identifying the customers who are most likely to default, banks can reduce the risk of loss, legal liability, and damage to their reputation.

Recall: Recall tells us the proportion of correctly identified positive predictions. It is the fraction of actual True that are predicted True. Having a high recall rate is important for banks because it allows them to identify the majority of customers who are likely to default. This is crucial because banks want to be able to prevent defaults and protect their bottom line. A high recall rate is important for banks because it allows them to take proactive measures to prevent defaults. For example, a bank with a high recall rate is more likely to be able to contact customers who are at risk of default and offer them financial assistance. This can help to prevent customers from defaulting on their loans and save the bank money.

F1 Score: F1 score measures precision and recall simultaneously by finding the harmonic mean of the two values. Comparing the F1 scores of the models, we can observe that by giving equal weightage to precision and recall, the optimized neural network has a high F1 score as there is equal trade off between precision and recall. The optimized logistic regression, decision tree, random forest and gradient boosting models all have low F1 scores due to the trade off of overfitting on these models. Therefore, the bank should employ neural network to accurately identify customers who are more likely to default.

Best Model To Predict Default:

Best Model: Keras Neural Network

Accuracy: 94.1%

F1 Score: 93.4%

Model Parameters: optimizer = 'adam', loss = 'binary_crossentropy', epochs = 150, batch_size = 16

Why Neural Network?

Neural networks can learn non-linear relationships between the independent and dependent variables. The probability of a person defaulting is a complex phenomenon that is influenced by a variety of factors, many of which are non-linear. Neural networks can learn these non-linear relationships and make accurate predictions even when the data is complex and noisy.

Neural networks can learn representation of the input data that is useful for making predictions. This representation learning can be used to extract features from the data that are not explicitly present in the input data. For example, a neural network could learn to represent the customer's financial history in a way that captures their risk of default.

Neural networks can be used to develop models that are robust to overfitting. Neural networks can be used to develop models that are less prone to overfitting by using techniques such as regularization and early stopping.

Neural networks can be used to develop models that are scalable to large datasets. Default prediction models often need to be trained on large datasets of historical data. Neural networks can be used to develop models that can be trained on these large datasets efficiently.

Changes Made That Did Not Improve Accuracy:

A. Data Wrangling

1. **GAP_X**: This refers to the difference between the limit balance and the bill amount for that month.
    a. Reason: My hypothesis was that the smaller the difference, the closer the bill amount is to the limit balance, the greater the probability of default. This hypothesis was derived from the data visualization that people with a smaller difference are more likely to default for all months.
2. **UNPAID_X**: This refers to the difference between the bill amount and payment made for that month.
    a. Reason: Similarly, my hypothesis was that the larger the difference, the larger the balance for that month after payment, the greater the probability of default.

Table 4 shows the comparison of accuracy scores of the models with and without GAP_X and UNPAID_X columns added. The accuracy scores did not increase, hence it is better to not include these columns. A possible reason that the scores did not improve could be that these columns are derived from the other columns that are already used as inputs in the models, and those columns do not have a high correlation with the target variable ‘default’ as well.

	Logistic Regression	Decision Tree	Random Forest	Gradient Boosting	Neural Network (Keras)
Without columns	78.3%	68.4%	81.4%	82.4%	88.5%
With columns	55.6%	69.1%	80.8%	82.0%	86.1%

Table 4

Therefore, the features used are still the original columns in the dataset, excluding ‘ID’ column.

Q2) Qualitatively, explain the pros and cons about all the models you use.

Logistic Regression:

Logistic Regression is a classification algorithm that is used to find the probability of whether an event happens or not. It is a supervised learning algorithm.

Advantages	Disadvantages
Performs well when the data is linearly separable	Does not work well on non-linearly separable data
Easy to implement and interpret,	difficult to get complex relationships using logistic regression.
Easily extended to classify multiple classes, using multinomial regression (Rout, 2023)	

Decision Tree:

CART, or a Classification and Regression Tree, is a supervised learning algorithm that is used for both classification and regression tasks.

Advantages	Disadvantages
Easily interpreted because decision trees are simply a series of if-else conditions	Sensitive to class imbalance. If some classes are much more common than others in the dataset, the tree will learn to predict the most common class, even if it is not the correct prediction for many of the data points.
Can perform multiclass classification	A small change in the dataset can make the tree structure unstable, which may lead to variance
Handles non-linear relationships between variables well. This is because they are able to learn complex relationships between features by splitting the data into smaller subsets based on the values of the features.	May not be the most accurate as it relies on variable importance generated by only one decision tree. This is not sufficient to represent the entire dataset, especially when we have a large dataset (Yadav, 2019)
Can handle both numerical and categorical data	

Random Forest:

Random forests are also used to solve classification and regression problems. They train on a large number of trees and combine them. The randomness comes from the fact that each tree is allowed to choose from a random subset of features to split on and each tree is trained on a random subset of observations.

Advantages	Disadvantages
More reliable than decision trees. It combines the output of numerous, randomly created decision trees to generate a final output.	Require a lot of computational power, resources and time to train, because they build many decision trees and combine their outputs
less likely to overfit the training data than decision trees, which can lead to improved accuracy	Difficult to interpret because they are an ensemble of decision trees
Can handle both categorical and continuous data	Difficult to determine the significance of each variable in the model
Can automatically handle missing values in the data (Team, 2023)	

Gradient Boosting:

Gradient boosting trees, unlike random forests, are trained sequentially, one tree at a time, each time correcting the errors of the previous trees. Random forests involve trees being constructed independently in parallel.

Advantages	Disadvantages
More accurate than random forests because we train them to correct the previous trees' errors	If the data is noisy, the gradient boosted trees may overfit and start modelling the noise
Capable of capturing complex patterns in the dataset (Simic, 2023)	

Gradient Tree Boosting tries to minimize a loss function, which is the difference between the real value and output of the learner, using the gradient descent method. XGBoost is an improved version of the Gradient Boosting algorithm that uses parallel processing, tree pruning, handling of missing values, and regularization to avoid overfitting and bias.

Neural Network:

Neural network is a machine learning model inspired by the human brain. It is composed of interconnected nodes, called neurons. Each neuron receives inputs from other neurons and produces an output that is then fed into other neurons.

Advantages	**Disadvantages**
Able to handle complex data	Requires large amounts of labelled training data
Non-linear modelling capabilities	Requires a lot of computational power
Robustness to noisy data	Lack of interpretability due to black box nature
Parallel processing for efficient computation	Probability of overfitting
Can handle high dimensional data	Complexity in model tuning

Q3) How to overcome the weakness of your all your models (future study).

Logistic Regression:

Handling non-linearity: Kernelization can be used to handle non-linearity. It transforms the data into a new space where the relationship between the features and the target variable is linear. Examples of kernelized logistic regression models include support vector machine (SVM) and radial basis function (RBF) kernel.

Regularize the model: Logistic Regression have the probability of overfitting the training data. To regularize the model, we can use L1 regularization or L2 regularization. L1 regularization penalizes the model for having large coefficients, while L2 regularization penalizes the model for having large coefficients in all directions.

Use ensemble methods: Ensemble methods combine the predictions of multiple models to produce a more accurate prediction. Ensemble methods can be used to improve the performance of logistic regression on a variety of tasks.

Decision Tree:

Handle imbalanced data: To handle imbalanced datasets, we can do oversampling or undersampling. Oversampling increases the number of samples in the minority class, while undersampling reduces the number of samples in the majority class.

Regularization: Regularization is a technique that penalizes the model for having complex features. This can help to reduce the variance of the model and make it more stable.

Bootstrapping: Bootstrapping is a technique that creates multiple training sets by sampling the original training set with replacement. This can help to reduce the variance of the model by training the model on multiple different datasets.

Tree pruning: Tree pruning is a technique that removes unnecessary branches from the decision tree. This can help to reduce the complexity of the tree and make it more stable.

Random Forest:

Reduce the computational cost: To reduce the computational cost, we can use a technique called parallel processing. Parallel processing involves training the random forest algorithm on multiple cores simultaneously.

Improve interpretability: To improve the interpretability of random forests, we can use a technique called feature importance. Feature importance involves identifying the features that are most important for the random forest algorithm to make predictions.

Gradient Boosting:

Feature selection: Feature selection involves selecting the most important features for the model. This can help to reduce the impact of noise on the model.

Noise injection: Noise injection involves adding noise to the training data in a controlled way. This can help the model to learn to ignore noise.

Regularization: Regularization is a technique that penalizes the model for having complex features. This can help to reduce overfitting and make the model more robust to noise.

Early stopping: Early stopping is a technique that stops training the model early when the validation performance starts to deteriorate. This can help to prevent the model from overfitting to the training data.

Neural Network:

Early stopping: Early stopping stops the training process early when the validation performance starts to deteriorate. Generally, accuracy increases with the number of epochs. However, after a certain point, increasing the epoch size may lead to overfitting. This causes the accuracy to plateau or even decrease. Early stopping helps to prevent overfitting and improves the generalization performance of the model.

Learning rate schedules: Allows the model to converge to the optimal solution and avoid getting stuck in local minima by slowly lowering the learning rate during training. It does so to help the model achieve the best results.

Q4) Please answer:

1. Why is credit card default prediction important for financial institutions?

Credit card default prediction is important for financial institutions for a number of reasons. First, it helps them to reduce their risk of losses. When a customer defaults on their credit card debt, the financial institution loses money. By predicting which customers are more likely to default, financial institutions can take steps to mitigate their risk, such as increasing interest rates for those customers or declining to issue them a credit card in the first place.

Second, credit card default prediction helps financial institutions to make better lending decisions. When a financial institution is considering whether to lend money to a customer, they will typically look at the customer's credit history to assess their risk of default. Credit card default prediction models can help financial institutions to get a more accurate picture of a customer's risk of default, which can lead to better lending decisions.

Third, credit card default prediction helps financial institutions to improve their customer service. By understanding which customers are more likely to default, financial institutions can reach out to those customers early on and offer them assistance, such as debt consolidation or financial counseling. This can help customers to avoid defaulting on their debt and improve their financial well-being.

2. How do consumers benefit from credit card default prediction?

Consumers benefit from credit card default prediction in a number of ways. First, it can help them to avoid defaulting on their debt. By understanding their risk of default, consumers can take steps to manage their credit more effectively and avoid getting into financial trouble.

Second, credit card default prediction can help consumers to get better credit card interest rates. When a consumer applies for a credit card, the financial institution will use their credit history to assess their risk of default. If the consumer has a low risk of default, they are more likely to be approved for a credit card with a lower interest rate.

Third, credit card default prediction can help consumers to get better credit card terms and conditions. When a consumer has a low risk of default, they are more likely to be approved for a credit card with better terms and conditions, such as a longer grace period or no annual fee.

3. How does credit card default prediction contribute to economic stability?

Credit card default prediction contributes to economic stability by helping financial institutions to reduce their risk of losses and make better lending decisions. When financial institutions are able to manage their risk more effectively, they are more likely to lend money to businesses and consumers. This can lead to increased investment and economic growth.

Additionally, credit card default prediction helps to protect consumers from financial hardship. By helping consumers to avoid defaulting on their debt, credit card default prediction can help to reduce the number of bankruptcies and foreclosures. This can have a positive impact on the overall economy.

4. What is the relationship between credit card default prediction and fraud detection?

Credit card default prediction and fraud detection are two closely related disciplines. Both disciplines use machine learning to identify patterns in data that can be used to predict future events.

Credit card default prediction models use data about customer behavior, such as their payment history and spending habits, to predict which customers are more likely to default on their debt. Fraud detection models use data about transactions, such as the time and location of the transaction, to identify fraudulent transactions.

Credit card default prediction and fraud detection models can be used together to improve the accuracy of both models. For example, a credit card default prediction model can be used to identify customers who are at a high risk of default. This information can then be used to flag transactions for those customers as suspicious, which can help fraud detection models to identify fraudulent transactions more accurately.

5. How has technology like AI transformed credit card default prediction?

Technology like AI has transformed credit card default prediction by making it possible to build more accurate and sophisticated models. AI models can learn complex patterns in data that would be difficult or impossible to identify using traditional methods.

Additionally, AI models can be trained on very large datasets. This allows them to learn patterns in the data that are not visible to the naked eye. This can lead to more accurate predictions of customer default risk.

AI models are also becoming increasingly accessible to financial institutions. This is due in part to the development of cloud computing platforms, which make it possible to train and deploy AI models without having to invest in expensive hardware and software.

As a result of these advances, AI is now widely used in credit card default prediction. AI models are helping financial institutions to reduce their risk of losses, make better lending decisions, and improve their customer service.

References

Rout, A. R. (2023, January 10). *Advantages and disadvantages of logistic regression*. GeeksforGeeks. https://www.geeksforgeeks.org/advantages-and-disadvantages-of-logistic-regression/

Yadav, A. (2019, January 11). *Decision trees*. Medium. https://towardsdatascience.com/decision-trees-d07e0f420175

Team, G. L. (2023, June 13). *Random Forest algorithm in Machine Learning: An overview*. Great Learning Blog: Free Resources what Matters to shape your Career! https://www.mygreatlearning.com/blog/random-forest-algorithm/#advantages-and-disadvantages-of-random-forest

Simic, W. by: M. (2023, May 15). *Gradient boosting trees vs. random forests*. Baeldung on Computer Science. https://www.baeldung.com/cs/gradient-boosting-trees-vs-random-forests#:~:text=4.3.-,Advantages%20and%20Disadvantages,and%20start%20modeling%20the%20noise.