**Comp4211 PA1 Report**

**Q1 Dataset Overview**

The dataset has 3539 instances and 33 features, including 31 data features and two targets. Besides, the dataset has 18 numerical features and 15 categorical features.

**Q2 Missing Values**

After using the info() functions, it provides all the information on the features. The feature with missing values and their corresponding non-null counts are as follows:

C0: 3510 non-null values (29 missing)

C4: 3490 non-null values (49 missing)

C5: 3431 non-null values (108missing)

C8: 3457 non-null values (82missing)

C9: 3397 non-null values (142missing)

C11: 3381 non-null values (158missing)

C12: 3369 non-null values (170missing)

C13: 3419 non-null values (120missing)

C15: 3401 non-null values (138missing)

C17: 3391 non-null values (148missing)

C20: 3368 non-null values (171missing)

C22: 3523 non-null values (16missing)

C23: 3511 non-null values (28missing)

C25: 3395 non-null values (144missing)

C29: 3379 non-null values (160missing)

These features have missing values in the data frame.

Missing values can have a potential impact on data analysis and model performance in several ways:

First is biased analysis, which means the missing value may be related to the target variable. So, it is an essential feature in target prediction. Missing it would lead to biased results and incorrect predictions.

The second is reducing the training samples. The missing value may decrease the adequate training size, limiting statistical test power and leading to unreliable estimates. With fewer data points, the analysis may have reduced precision.

For the model performance, many algorithms in machine learning, like MLP classification, can not handle the missing data, which leads to errors or biased results. So, training the model with missing values will decrease its performance and may cause it to struggle to generalize well to new data.

**Q3: Feature Distribution**

Discrete features: C16,C21,C22,C27

Continuous features:C6,C14,C15,C17:C20,C23:C26,C28:30

The first three numerical features are C6, C14, and C16. The district-

button (e.g., mean, median, range, variance) of them are the following:

文本

描述已自动生成

The box plots about the three numerical features are the following:

图表, 箱线图

描述已自动生成

So, from the above box plot, **we can see that C6 and C14 are continuous, but C16 is discrete**. For continuous features, the box plot shows the quantile of the data, including the minimum, first quartile (lower quartile), median, third quartile (upper quartile), and maximum. However, for the discrete feature, the box plot may show some discrete values or a horizontal line. Box lengths are usually small or invisible because most data points have the same value.

Categorical Feature

Binary:C4,C8,C9,C10,C11,C12,C13,C15

Ordinal:C1,C2,C5

Nominal:C0,C3,C7

The first three categorical features are C0, C1, C2

图表, 直方图

描述已自动生成Here is the figure of C0

single 3115

married 296

divorced 75

facto union 18

legally separated 3

widower 3

图表, 直方图

描述已自动生成Here is the figure of C1

1st phase - general contingent 1351

2nd phase - general contingent 708

Over 23 years old, 630

Change of course 253

Technological specialization diploma holders 160

Holders of other higher courses 109

3rd phase - general contingent 105

Transfer 67

Change of institution/course 45

Short cycle diploma holders 30

1st phase - special contingent(Madeira Island) 29

International student (bachelor) 26

1st phase - special contingent(Azores Island) 14

Ordinance No. 854-B/99 7

Ordinance No. 612/93 2

Ordinance No. 533-A/99, item b3 (Other Institution)1

Ordinance No. 533-A/99, item b2) (Different Plan) 1

Change of institution/course (International) 1

图表, 直方图

描述已自动生成Here is the figure of C2

second choice 2402

third choice 457

fourth choice 247

fifth choice 194

sixth choice 125

seventh choice 112

last choice 1

first choice 1

表格

描述已自动生成Q4 Outliers

To identify the potential outliers in the first three numerical features.

**The first method is using the Z-score.** Import the states from the Scipy library and use the Zscore function to calculate the first three numerical features of Zscore. To check which samples the Zscores are more significant than three or smaller than -3, those samples will be the outliers.

表格

描述已自动生成

**The second method is using the isolation forest.** Import the IsolationForest from sklearn.ensemble. Set the contamination rate to 0.05962136 because I want to show the same number of sample Z-score outputs. After the isolation forest prediction, it gives out a true-false array; the false mean is the outlier.

How do outliers affect my preprocessing and modeling strategy?

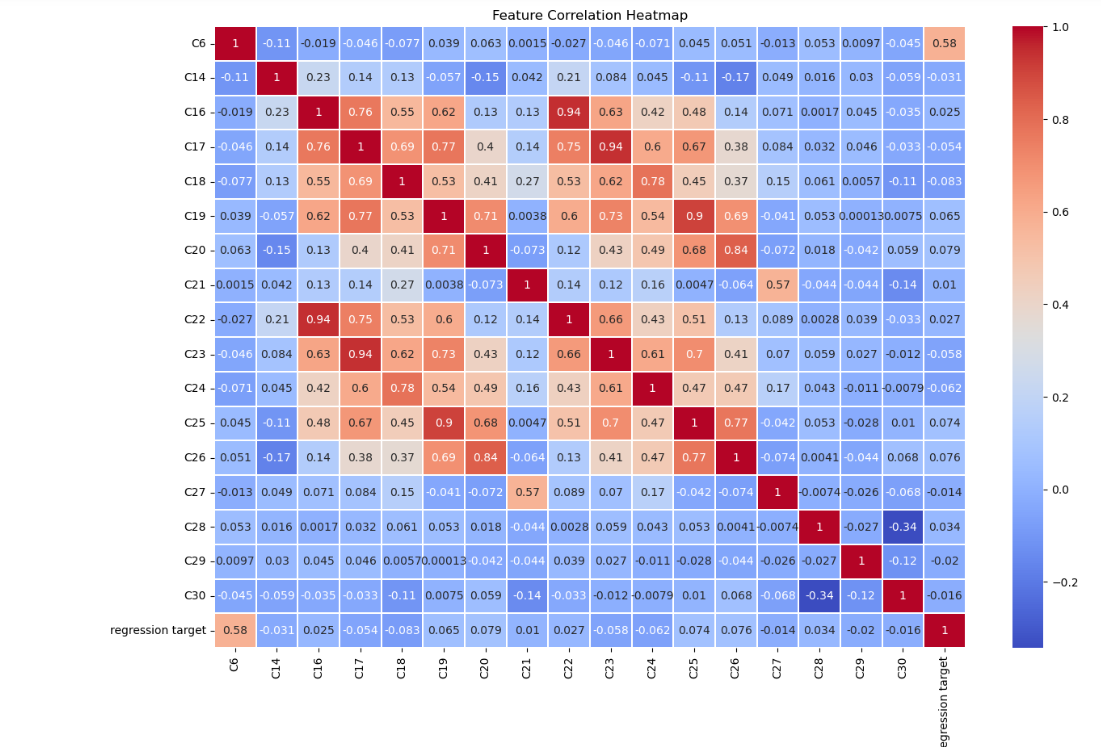
First, when considering the outliers, one must carefully evaluate their nature before deciding how to handle them. Because to duel with different outliers, we need to choose unique methods (e.g., remove it, replace it with mean or median). Other skills will have different effects on future training.

Second, the outliers can affect feature engineering techniques such as scaling, binning, or encoding. For example, outliers can disproportionately influence the scaling process and distort the data distribution if we use a scaling method like standardization or min-max scaling.

For the model strategy, outliers may affect the model selection. Since some of the models, such as linear regression, are sensitive to the outliers. So, the existing outliers will cause the model to fit the data poorly and may lead to a biased prediction. So, in this situation, we might consider using robust models less affected by the outliers. Therefore, we need to select a suitable model.

Q5 Correlation Analysis

The heatmap is the following:



By analyzing the heatmap, you can identify strong correlations between features. The higher the related, the deeper the colors. Strong positive correlations (values close to 1) indicate that the features tend to increase or decrease together, while strong negative correlations (values relative to -1) show an inverse relationship.

The heatmap shows that C6 has the highest correlation with the regression target; the second one is the C20, and the third is the C26. In feature engineering, if we use Kbestselection, we will select k features with a higher correlation with the target label, which we can look at heatmap: the higher the correlation, the more critical in feature engineering.

Q6

Necessary preprocessing steps for regression models or neural networks to run effectively:

First, Handling Missing Values: The dataset's missing values must be considered if they exist. The missing values can be imputed using methods like mean, median, or interpolation, or they can be removed from the rows or columns that contain them.

Encoding Categorical Variables: The dataset's categorical variables must be converted into numerical values if they exist. Depending on the structure and cardinality of the category variables, methods like Ordinal encoding or one-hot encoding can be used to accomplish this.

Scaling/Normalization: Numerical features should ideally be scaled or normalized to a similar range. This aids in keeping characteristics with higher magnitudes from controlling how the model learns. Standard scaling techniques include standardization or min-max scaling.

Handling Outliers: If outliers are present, they can significantly affect neural networks or regression models. It's critical to recognize and deal with outliers correctly. This can entail capping and eliminating outliers based on statistical measurements like Z-score or Isolation Forest.

Feature Selection: If the dataset has many features, it may be beneficial to perform feature selection techniques to reduce the complexity of the model and improve performance. Techniques such as VarianceThreshold and SelectKBest can be used for this purpose.

Specific challenges include:

Imbalanced Data: Biased models may result from imbalanced classes, in which one class significantly outweighs the others. Strategies like undersampling, oversampling, or employing synthetic data-generating techniques can be used to balance the data.

Feature Engineering: During exploration, specific patterns or relationships between variables are identified that can be captured by creating new features. This involves creating derived features or combining existing features to provide more meaningful information to the model. So the performance or the model can be enhanced.

Handling Multicollinearity: A high correlation between predictor variables can cause problems with multicollinearity and impact model performance. One can employ methods like variance inflation factor (VIF) analysis or correlation analysis to detect and manage multicollinearity.

Q7:

We can impute mean, median, mode, and constant with zero to the numerical features during the Imputation. But for the categories features, we can only apply the mode and constant imputer cause we can calculate the string’s mean or median.

Impact on feature distribution and model performance:

Mean Imputation:

* Impact on Feature Distribution: The mean of the available values is used to fill in the missing variables by using mean imputation. This method maintains the feature mean and, barring extreme outliers, has little effect on the distribution.
* Model Performance: Mean imputation is appropriate when the feature has a continuous numerical distribution, and the missing values are assumed to be missing at random (MAR). Besides, features without extreme outliers or a skewed distribution do well with it.

Median Imputation:

* Impact on Feature Distribution: Median imputation replaces missing values with the median of the available values. This strategy is robust to outliers and does not affect the median of the feature.
* Model Performance: Median imputation is suitable when the missing values are assumed to be missing, not at random (MNAR), and the feature has a skewed distribution or outliers. It offers a reliable estimate for missing values and performs well with continuous numerical features.

Mode Imputation:

* Impact on Feature Distribution: Mode imputation replaces missing values with the most frequently available values. So that can keep the mode of the feature and does not affect the shape of the distribution.
* Model Performance: Mode imputation works well with nominal or categorical features. It helps to maintain the feature's frequency distribution and functions effectively when missing values arise in variables with a small number of dominant categories.

Constant Imputation:

* Impact on Feature Distribution: Constant imputation replaces missing values with a predefined constant value, which does not show the original data distribution. But introduces a new category or level for the missing values.
* Model Performance: Constant imputation is suitable when the missing values are missing completely at random (MCAR) or when the missingness is informative. It is frequently utilized when missing values must be handled as a distinct category.

Judge what imputation should be used and when

* Mean imputation can be applied to continuous numerical features with skewed distributions or no outliers.
* Median imputation is reliable for continuous numerical characteristics with skewed distributions or outliers.
* Mode imputation is ideal for features that are nominal or categorical.
* When missing values must be handled differently or when the missingness provides information, constant imputation can be applied.

Q8 There are the first ten outputs from the first numerical feature before and after different processing.

文本

描述已自动生成

StandardScaler:

This tool divides the data by the standard deviation and centers the data by removing the mean. It is ideal when the data has a Gaussian distribution or when you don't know anything about the data distribution beforehand. It maintains the distribution's shape but does not provide a range guarantee.

MinMaxScaler:

This function scales the data within a given range, usually from 0 to 1. Most appropriate for non-Gaussian or uncertain distribution data. It retains the distribution's structure and is outlier-sensitive.

RobustScaler:

This tool uses statistics like the median and interquartile range, which are resistant to outliers, to scale the data, which is appropriate when the data has a non-Gaussian distribution or contains outliers. Unlike StandardScaler, it maintains the distribution's form and is less impacted by outliers.

Q9 Encoding Categorical Variables:

There are the first ten outputs from the first categorical feature before and after different processing 文本

描述已自动生成

OneHot Encoding:

Scenario: When dealing with categorical features with no inherent order or hierarchy and when the number of unique categories is relatively small.

Every distinct category will be transformed into a binary column, with a value of 1 denoting the category's presence and 0 indicating its absence. This approach is recommended when there isn't any natural hierarchy or sorting among the categories, and there aren't too many distinct categories. Single-hot coding is beneficial in cases where the categorical features lack meaning.

Ordinal Encoding:

Scenario: When dealing with categorical features with an inherent order or hierarchy.

A numerical value is allocated to each distinct category based on its rank or order. The value assigned may follow a logical order or be determined arbitrarily. This approach is favored when there is a natural order or rank to the categorical traits. Ordinal features, for instance, like rating (low, medium, high) or education level (high school, bachelor's, master's, etc.) are examples of such features.

Q10 Feature selection

Feature selection impacts the model in the following way:

Enhanced Model Efficiency: Feature selection can lower the dataset's dimensionality by choosing a subset of pertinent features. This feature reduction may result in quicker training and inference times because there are fewer parameters for the model to learn and calculations to do.   
  
Reducing Overfitting: Feature selection lowers the possibility of overfitting, a condition where a model is overly intricate and performs well on training data but badly on unobserved data. Feature selection helps the model focus on the most informative aspects by eliminating noisy or irrelevant features. This lowers the likelihood of overfitting and improves generalization to new data.

Removed features:

For the Variance threshold, I set the threshold value to 0.2. Any feature with a variance lower than 0.2 will be removed.

Suppose we apply the numerical data after StandardScaler into it. The result will be all true. No numerical features will be removed. StandardScaler scales the feature to make the variance unit variance, but VarianceThreshold throws features away based on the variance.

Besides, the removed features for the categorical after using the VaricaneThreshold are ‘C4’, ‘C9’, ‘C10’, ‘C11’, ‘C13’, and ‘C15’. Because they had low variance, meaning they had little or no variability in their values across the dataset.

For the SelectKBest, I set the score\_func of categorical to chi2 and the score\_func of numerical to f\_regression with int k. So, the top K feature will be kept because they have the statistical relationship between each feature and the target variable.

Because I set the k to be five here, the numerical feature will keep the ‘C6’,’ C18’,’ C20’, and’C25’.’ C26’ columns; the other will be removed.

For the categorical feature,’ C1’,’ C3’,’ C10’,’ C12’, and’ C13’ will be kept, and the rest of the columns will be removed since these five columns are the most correlated with the target.

Q11

If I plan to use linear regression, creating interaction features will be one of the suitable feature engineering.

Interaction features are derived by combining two or more existing features through multiplication or other mathematical operations. Interaction features are being created to capture possible non-linear correlations and intricate interactions between the features.

Adding interaction features to the linear regression model can aid in better-capturing pattern combinations observed in the non-linear or interactive relationships between the target variable and specific feature.

Identify potentially meaningful combinations of features based on domain knowledge or data exploration. For instance, if we have feature1 and feature2, we can create a feature interaction. Which is feature = feature1 \* feature2. Then, include the newly created interaction features and the original features in the feature matrix. Fit the linear regression model using the augmented feature matrix.

In this way, the linear regression model can find out the relationship between the data and interaction, which can improve the performance of the linear regression model.

Q12

The R2 score measures how well the regression models fit the data. It represents the proportion of the variance in the target variable that can be explained by the features used in the model. A higher R2 score indicates a better fit, suggesting a stronger relationship between the features and the regression target.

Here are the R2 scores for each model:

Model 0: R2 score is 0.3457983455467899

Model 1: R2 score is -0.0009205773022631369

Model 4: R2 score is 0.01785195834079767

Model 10: R2 score is 0.010488645157375132

Model 12: R2 score is -0.007736001857362451

Model 14: R2 score is 0.0025504461422434233

Combined Model: R2 score is 0.009222254707064947

The R2 scores obtained from the different models suggest varying degrees of relationship between the selected features and the regression target. Model 0 seems to have the highest R2 score, indicating a relatively stronger relationship, while Model 1 has an R2 score close to zero, suggesting a weaker or no relationship. Models 4, 10, 12, and 14 show relatively small positive or negative R2 scores, indicating a more fragile relationship than Model 0.

Combined Model: This model combines the effects of all the selected features from the individual models. We can see the R2 score in combination is not significantly improved; the R2 score is relatively low compared to the unique models, suggesting that combining all the selected features does not significantly improve the model's ability to explain the variance in the target variable.

Q13 MSE

The MSE measures the average squared difference between the predicted and actual values.

Model 0: MSE is 0.0031823076103367383  
Model 1: MSE is 0.004868891952212952  
Model 4: MSE is 0.004777574569207919  
Model 10: MSE is 0.004813392772083733  
Model 12: MSE is 0.0049020449980387015  
Model 14: MSE is 0.004852007457580571  
Combined Model: MSE is 0.17350390869738402

Lower values of MSE indicate better model performance, as it represents the average squared difference between the predicted and actual values.

Hence, when we compare the R2 score and Mean Squared Error, we can find that Model 0 has the highest value of 0.3457983455467899, which performs the best in explaining the variance in the variable of interest. Model 4 has the second-highest R2 score, followed by Model 10 and Model 14. Model 1 and Model 12 get negative R2 scores, meaning poor performance correlates with the target variable. The combined model has a relatively low R2 score of 0.009222254707064947, which is not as good as the individual model.

In terms of MSE, Model 0 has the lowest value of 0.0031823076103367383, indicating the best performance in terms of prediction accuracy. Model 1 has a little higher MSE than Models 4, 10, 12, and 14. The MSE values of these models are comparable. The combined model has a significantly higher MSE of 0.17350390869738402, indicating poorer prediction accuracy than the individual models.

In conclusion, Model 0 performs best when looking at the R2 score and MSE. Compared to the individual models, the combined model's R2 score is relatively low, and its MSE is much greater, indicating that incorporating all of the chosen features does not enhance prediction performance or accuracy.

Q14

The mathematical meaning of weight represents the change in the regression target associated with a shift from the reference category to the other category. It quantifies the effect of the binary categorical variable on the target variable while holding all other variables constant.

When dealing with a definite feature with more than two possible values, it can be formulated as the independent variable of a linear model using a one-hot encoding technique. One-hot encoding converts each category into a binary variable, indicating whether a particular category is present for each data point. These binary variables are then used as independent variables in the linear regression model.

A binary categorical independent variable can be encoded as binary, typically using one-hot encoding. In this encoding scheme, the binary variable takes a value of 0 or 1 to represent the absence or presence of a particular category respectively. The weight represents the average change in the target variable when the corresponding binary variable changes from 0 to 1.

When dealing with a definite feature with more than two possible values. We may apply a Label encoder to give different values to a unique label to represent it. For example, in the C0, there are ‘married,’’singler,’’ divorced,’ and so on, and the label encoder will separate them into a unique label.

Q15

This model set consists of a neural network with three hidden layers, each with the number of units (H) specified by the input parameter "hidden\_unit." The activation function used in all hidden layers is ReLU, and the output layer has 1 unit with a linear activation function. The model is trained using the Adam optimizer and mean squared error loss. Use early stopping with a patience of 10 to prevent overfitting. This model is trained for a maximum of 100 epochs. The training process is repeated "Repeat\_time" times, and each repetition's R2 score and training time are recorded. Finally, the average and variance are calculated based on the three R2 scores and training time.

The performance of the model is in the following, which shows the training time, R2 score, mean, and standard deviation.

文本

描述已自动生成

Q16

The following are two figures about the R2 score VS hidden units number and the training time VS hidden units number.

图表, 折线图

描述已自动生成

**The blue color line represents the mean of the R2 score or training time. The orange one will define the std of R2 score or training time.**

图表, 折线图

描述已自动生成

We can find that the eight hidden units will get the highest R2 score with the longest training time. One hidden unit got the lowest R2 score in performance with a shorter training time. The hidden unit increased to 32 and 128, and the performance decreased slightly; the overfitting may have caused the R2 score to drop. And for the training time, hidden units 32 and 128 are short. This is because the model setting is about an early stop; when the model performance gets worse with more profound training, the model will stop the training. So, the training time will be small.

Q17-18:

This model set includes a logistic regression model with up to 100 iterations and a "sag" solver. The regularization strength is determined by multiplying the input parameter "learning rate" by C = 1/learning rate. The model is trained "repeat count" times, set to 3 by the question requirement. The model will calculate the F1\_score and accuracy using the accuracy score function and f1\_score function. The average training time, accuracy, and F1 score are recorded for each "learning rate" value. The standard deviations of these metrics are also calculated and recorded. The results are stored in a dictionary called "results."

The following is the output when using the learning rate to be 10.

图形用户界面, 文本

描述已自动生成

The following figure is the ROC curve with the abovementioned setting with learning rate = 10

图表, 折线图

描述已自动生成

The ROC curve depicts the trade-off between the actual positive rate (sensitivity) and the false positive rate (specificity - 1) at various classification thresholds. We can see how the threshold for differentiating between positive and negative cases affects the model's sensitivity and specificity. When the cost of false positives and false negatives varies, this knowledge is crucial because it enables us to choose the best threshold that balances the two rates according to requirements.

Q19

The learning rate array = [10,1,0.1]

The performance of the model in the following, which shows the training time, F1 score Accuracy mena, and standard deviaction:

The learning rate comes to convergence at 0.1文本

描述已自动生成

The following figure is the RPC and AUC of different learning rates.

图表, 折线图

描述已自动生成

Q20

The input parameter "hidden\_unit" specifies the number of units for each of the three hidden layers in the multi-layer perceptron (MLP) classifier that is part of the model set. Overfitting is avoided by early quitting. After the "num\_runs" of training, the model's accuracy, F1 score, and average training time are noted for every value of "hidden\_unit." These measures' standard deviation is also computed and noted. Lists named "training\_times," "neural\_accuracy," and "neural\_f1\_scores" are used to hold the training times, accuracies, and F1 scores for each condition. The results provide the neural network's model setup, training duration, and performance for each value of "hidden\_unit."

The following figures show the different hidden units with training time, accuracy, f1 score mean, and std.

文本

描述已自动生成

Q21

图表, 折线图

描述已自动生成

Regarding the gap between accuracy and F1 score, the F1 score is always higher than the accuracy for all values ​​of hidden units. This may be due to the category imbalance in the data set, and we only selected the most suitable fears based on KBestSelect. F1 score is more sensitive to minority class performance than accuracy. Therefore, the F1 score is higher than the accuracy, forming a gap.

Q22

According to the data, the best neural network model beats the logistic regression model with any learning rate regarding accuracy and F1 score. On the other hand, compared to the neural network model with 128 hidden units, the logistic regression model requires substantially less training time.

The optimal neural network model with 128 hidden units performs better than the logistic regression model with any learning rate in accuracy and F1 score. This indicates that logistic regression is less appropriate for this classification problem than the neural network model.

Regarding training time, the neural network model with 128 hidden units requires a significantly more extended training period than the logistic regression model, which would be a better option.

Q23

文本

描述已自动生成

The output data indicates a tendency that occurs when the hidden layer size rises from 1 to 128. In other words, as the number of remote units rises, so do the neural network model's accuracy and F1 score. This demonstrates that adding more hidden units can boost the model's capacity for learning and, as a result, enhance its capacity for accurately classifying data.

However, when the number of hidden units rises, neural network models' training times likewise do. Because there will be more parameters to optimize during the training process if there are more hidden units, more computations will be involved, which will raise the computing cost of training the model.

Overall, the trend shows that increasing the number of hidden units improves the performance of the neural network model but at the cost of longer training times.

Q24

图形用户界面, 文本, 应用程序

描述已自动生成

CombinationA imputes the value with mean, using the skip.StandardScaler(), and use the OneHot encoder to deal with the categorical features.

Based on the above figure, the MLP classifier with three hidden layers and 128 hidden units per layer achieved a mean accuracy of 0.8865 and a mean F1 score of 0.9231 on the transformed dataset (combination), with a mean training time of 0.3201 seconds. The standard deviations for accuracy and F1 score are relatively small, indicating that the model's performance is consistent.

These findings imply that the pipeline and column transformer classes' pre-processing actions enhanced the dataset's capacity for accurate sample classification by the model. The model may be improved to better capture the underlying patterns in the data by using one-hot encoding for categorical features and standardization for numerical characteristics.

Q25

文本

描述已自动生成

Combination B imputes the value with a constant using the skip.MinMaxScaler(), and separate the categorical features to use different encoders. Ordinal features use an Ordinal encoder; the rest use a OneHot encoder.

From the above figure, the MLP classifier with three hidden layers and 128 hidden units per layer achieved a mean accuracy of 0.8653 and a mean F1 score of 0.9072 on the transformed dataset (CombinationB), with a mean training time of 0.5237 seconds.

Compared to CombinationA, CombinationB has a lower mean accuracy and F1 score, but the difference is not significant. This suggests that the pre-processing steps applied to CombinationB may not be as practical as those applied to CombinationA in capturing the underlying patterns in the data.

Q26

文本

描述已自动生成

Combination C uses feature selection and imputes the value with the median using the skip.RobustScaler(), encoded by OneHotEncoder and the

From the above figure, the MLP classifier trained on CombinationC features achieved an average accuracy of 0.8917 and an average F1 score of 0.9251 over three runs. The standard deviations for both metrics were also reported as 0.00133 for accuracy and 0.000129 for F1 score. The mean training time was 0.430 seconds, with a standard deviation of 0.0847 seconds.

Compared to CombinationA and CombinationB, the MLP classifier trained on CombinationC features achieved a higher average accuracy of 0.8917 and a higher average F1 score of 0.9251 over three runs. However, the model's performance may depend on the separate test set so that the code may have a different performance each time.

Q27  
The best five combinations of the hyperparameter setting are in the following figure. The figures also show the **mean and std scores.**

文本, 信件

描述已自动生成

The best combination is the following:

文本

描述已自动生成

We can see that the first four combinations have the same mean and std score, but the best is the first one. The grid search algorithm may choose the first combination that gets those scores because it encounters them first during the search.