Milestone 2: Feature Engineering, Feature Selection, and Data Modeling Timeline

- **Member:**

Yuexi Lyu (67468519)

- **Objective of the project**

The primary objective of this project is to analyze global COVID-19 epidemic data and build a machine learning model to predict patient recovery rates. Our final machine learning task is to develop a predictive model that can accurately estimate the recovery rate of COVID-19 patients based on various features, such as the number of confirmed cases, deaths, active cases, country code, and temporal characteristics.

In this project, we applied three different machine learning models to perform the prediction. By comparing the performance of each model across multiple evaluation metrics, we identified the one most suitable for this task. The results of this study can assist health departments in forecasting the epidemic's trajectory and evaluating the effectiveness of their prevention and control strategies.

- **Type of tool**

The final product will be a predictive analytics tool that includes:

Predictive Modeling: Implementing a machine learning model to predict the recovery rate of COVID-19 trends (completed).

Geospatial Analysis: Mapping the global spread of COVID-19.

Interactive Dashboards: Providing user-friendly visualization tools for exploring pandemic data in real time. The tool will provide interactive dashboards and statistical insights to visualize and analyze the global impact of the COVID-19 pandemic.

- **Data to be used.**

The following datasets have been utilized for this analysis:

Full Grouped Dataset(full_grouped.csv): Time-series data of confirmed cases, recoveries, deaths, and active cases worldwide.

```
In [124]: print(df_full_grouped.head(), "\n")

         Date Country/Region  Confirmed  Deaths  Recovered  Active  New cases  \
0  2020-01-22    Afghanistan          0       0          0       0          0
1  2020-01-22        Albania          0       0          0       0          0
2  2020-01-22        Algeria          0       0          0       0          0
3  2020-01-22        Andorra          0       0          0       0          0
4  2020-01-22         Angola          0       0          0       0          0

   New deaths  New recovered             WHO Region
0           0              0  Eastern Mediterranean
1           0              0                 Europe
2           0              0                 Africa
3           0              0                 Europe
4           0              0                 Africa
```

```
In [106]: df_full_grouped.describe()
Out[106]:
```

	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered
count	3.515600e+04	35156.000000	3.515600e+04	3.515600e+04	35156.00000	35156.000000	35156.000000
mean	2.356663e+04	1234.068239	1.104813e+04	1.128443e+04	469.36375	18.603339	269.315593
std	1.499818e+05	7437.238354	6.454640e+04	8.997149e+04	3005.86754	115.706351	2068.063852
min	0.000000e+00	0.000000	0.000000e+00	-2.000000e+00	0.00000	-1918.000000	-16298.000000
25%	1.000000e+00	0.000000	0.000000e+00	0.000000e+00	0.00000	0.000000	0.000000
50%	2.500000e+02	4.000000	3.300000e+01	8.500000e+01	2.00000	0.000000	0.000000
75%	3.640250e+03	78.250000	1.286250e+03	1.454000e+03	75.00000	1.000000	20.000000
max	4.290259e+06	148011.000000	1.846641e+06	2.816444e+06	77255.00000	3887.000000	140050.000000

COVID-19 Clean Complete Dataset(covid_19_clean_complete.csv): Comprehensive dataset including country-level statistics with additional geographic information (latitude, longitude).

```
: print(df_covid_19_clean.head(), "\n")

  Province/State Country/Region       Lat       Long        Date  Confirmed  \
0            NaN    Afghanistan  33.93911  67.709953  2020-01-22          0
1            NaN        Albania  41.15330  20.168300  2020-01-22          0
2            NaN        Algeria  28.03390   1.659600  2020-01-22          0
3            NaN        Andorra  42.50630   1.521800  2020-01-22          0
4            NaN         Angola -11.20270  17.873900  2020-01-22          0

   Deaths  Recovered  Active             WHO Region
0       0          0       0  Eastern Mediterranean
1       0          0       0                 Europe
2       0          0       0                 Africa
3       0          0       0                 Europe
4       0          0       0                 Africa
```

```
: df_covid_19_clean.describe()
:
```

	Lat	Long	Confirmed	Deaths	Recovered	Active
count	49068.000000	49068.000000	4.906800e+04	49068.000000	4.906800e+04	4.906800e+04
mean	21.433730	23.528236	1.688490e+04	884.179160	7.915713e+03	8.085012e+03
std	24.950320	70.442740	1.273002e+05	6313.584411	5.480092e+04	7.625890e+04
min	-51.796300	-135.000000	0.000000e+00	0.000000	0.000000e+00	-1.400000e+01
25%	7.873054	-15.310100	4.000000e+00	0.000000	0.000000e+00	0.000000e+00
50%	23.634500	21.745300	1.680000e+02	2.000000	2.900000e+01	2.600000e+01
75%	41.204380	80.771797	1.518250e+03	30.000000	6.660000e+02	6.060000e+02
max	71.706900	178.065000	4.290259e+06	148011.000000	1.846641e+06	2.816444e+06

Worldometer(worldometer_data.csv) Dataset: A snapshot of country-level statistics such as total cases, deaths, recovered cases, active cases, and per capita statistics.

```
: print(df_worldometer_data.head(), "\n")

  Country/Region      Continent    Population  TotalCases  NewCases  \
0            USA  North America  3.311981e+08     5032179       NaN
1         Brazil  South America  2.127107e+08     2917562       NaN
2          India           Asia  1.381345e+09     2025409       NaN
3         Russia         Europe  1.459409e+08      871894       NaN
4   South Africa         Africa  5.938157e+07      538184       NaN

   TotalDeaths  NewDeaths  TotalRecovered  NewRecovered  ActiveCases  \
0     162804.0        NaN       2576668.0           NaN    2292707.0
1      98644.0        NaN       2047660.0           NaN     771258.0
2      41638.0        NaN       1377384.0           NaN     606387.0
3      14606.0        NaN        676357.0           NaN     180931.0
4       9604.0        NaN        387316.0           NaN     141264.0

   Serious,Critical  Tot Cases/1M pop  Deaths/1M pop  TotalTests  \
0           18296.0           15194.0          492.0  63139605.0
1            8318.0           13716.0          464.0  13206188.0
2            8944.0            1466.0           30.0  22149351.0
3            2300.0            5974.0          100.0  29716907.0
4             539.0            9063.0          162.0   3149807.0

   Tests/1M pop      WHO Region
0      190640.0        Americas
1       62085.0        Americas
2       16035.0  South-EastAsia
3      203623.0          Europe
4       53044.0          Africa
```

```
: df_worldometer_data.describe()
:
```

	Population	TotalCases	NewCases	TotalDeaths	NewDeaths	TotalRecovered	NewRecovered	ActiveCases	Serious,Critical	Tot Cases/1M pop	D
count	2.080000e+02	2.090000e+02	4.000000	188.000000	3.000000	2.050000e+02	3.000000	2.050000e+02	122.000000	208.000000	18
mean	3.041549e+07	9.171850e+04	1980.500000	3792.590426	300.000000	5.887898e+04	1706.000000	2.766433e+04	534.393443	3196.024038	9
std	1.047661e+08	4.325867e+05	3129.611424	15487.184877	451.199512	2.566984e+05	2154.779803	1.746327e+05	2047.518613	5191.986457	17
min	8.010000e+02	1.000000e+01	20.000000	1.000000	1.000000	7.000000e+00	42.000000	0.000000e+00	1.000000	3.000000	
25%	9.663140e+05	7.120000e+02	27.500000	22.000000	40.500000	3.340000e+02	489.000000	8.600000e+01	3.250000	282.000000	
50%	7.041972e+06	4.491000e+03	656.000000	113.000000	80.000000	2.178000e+03	936.000000	8.990000e+02	27.500000	1015.000000	2
75%	2.575614e+07	3.689600e+04	2609.000000	786.000000	449.500000	2.055300e+04	2538.000000	7.124000e+03	160.250000	3841.750000	9
max	1.381345e+09	5.032179e+06	6590.000000	162804.000000	819.000000	2.576668e+06	4140.000000	2.292707e+06	18296.000000	39922.000000	12

- **Project timeline:**

Milestone 3: Evaluation & Finalization

Evaluation and Interpretation (Due by April 12)

Tool Development (Due by April 14)

Report Writing (Due by April 16)

Uploading to GitHub (Due by April 16)

4-Minute Presentation (Due by April 20)

- **Combined and processed data:**

During the dataset merging phase, I integrated multiple key datasets to create a comprehensive COVID-19 analysis foundation. First, I joined the two main datasets, "Full Grouped" and "COVID-19 Clean Complete", by country and date, ensuring consistency in date format. An outer join was used to retain all data, and the suffixes "_full" and "_clean" were used to distinguish the sources. For country records that only exist in a single dataset, I filled missing values with 0 to maintain data integrity. Subsequently, I extracted the overall epidemic indicators of global countries from the Worldometer dataset, removed duplicate records and unnecessary WHO region columns, and left-joined them to the merged dataset. The final merged dataset contains time series epidemic data and country-level summary indicators.

```
In [87]: print("the columns of df_merged:", df_merged.columns.tolist())

the columns of df_merged: ['Date', 'Country/Region', 'Confirmed_full', 'Deaths_full', 'Recovered_full', 'Active_ful
l', 'Confirmed_clean', 'Deaths_clean', 'Recovered_clean', 'Active_clean']
```

- ## Create new features:

I performed comprehensive feature creation on the merged dataset, which significantly enhanced the analytical value of the data. First, I calculated the basic recovery rate indicators (recovery rate, mortality rate, active case rate), and created corresponding features for the two data sources ("_full" and "_clean") to avoid zero division errors. Second, I constructed time-related features, including the number of new recoveries per day, the recovery growth rate, and the number of days the epidemic lasted, to capture the dynamic characteristics of the disease development. In addition, I also developed trend features (7-day and 14-day moving averages, trend slopes, recovery acceleration) and ratio features (recovery-death ratio, cure ratio) to reflect the development pattern of the epidemic. I also created inter-dataset difference features to help identify data quality issues. These carefully designed features together form a multi-dimensional feature space, allowing the model to more accurately capture the complex factors that affect the recovery rate of COVID-19.

- ## Label encoding:

In the categorical variable processing stage, I implemented multiple encoding techniques to convert text features into a numerical form that the model can handle. First, I used LabelEncoder to label encode the "Country/Region" column to convert each country/region name into a unique numerical identifier. Next, I checked whether there were categorical columns such as "WHO Region" or "Continent" in the dataset and applied one-hot encoding to these columns, removing the first category to avoid collinearity issues and properly handling missing values.

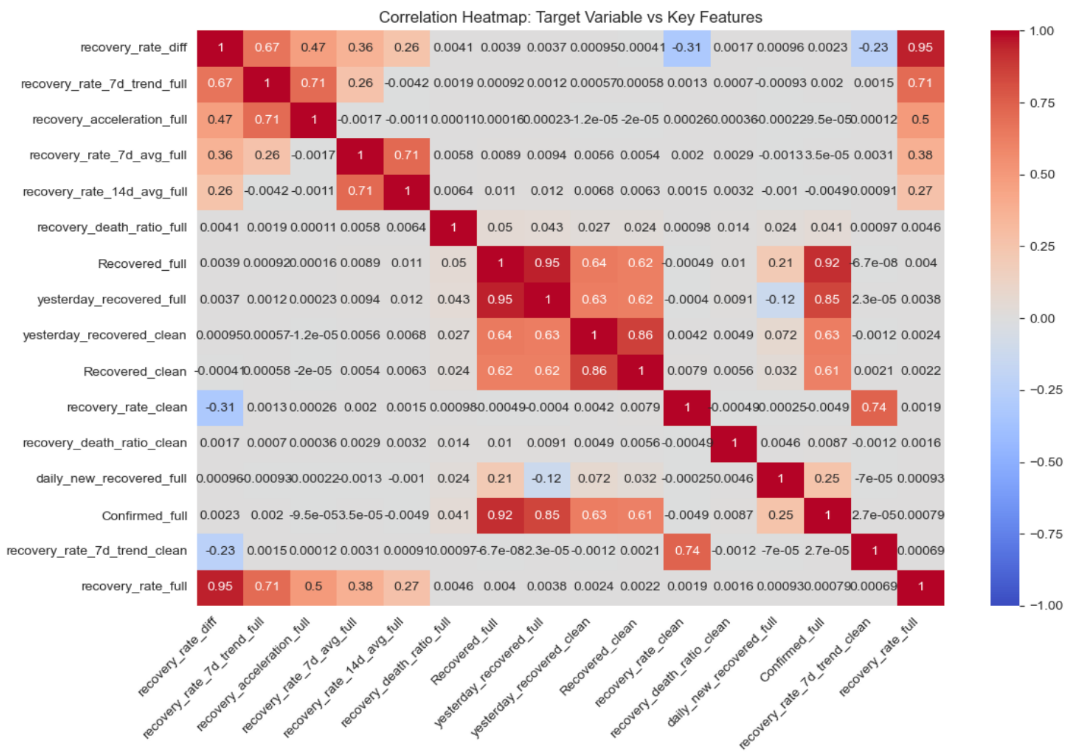
In addition, I created a new grouping feature to divide countries into three groups of high, medium, and low based on the number of confirmed cases, and label encoded this new feature. This hierarchical approach provides a more abstract way to analyze the recovery patterns of countries with different epidemic severity, reducing the high dimensionality of the original country variable while retaining information about country-level differences.

```
           Date Country/Region  Confirmed_full  Deaths_full  Recovered_full  \
0    2020-01-22    Afghanistan       -0.443106    -0.346973        -0.35464
261  2020-01-23    Afghanistan       -0.443106    -0.346973        -0.35464
522  2020-01-24    Afghanistan       -0.443106    -0.346973        -0.35464
783  2020-01-25    Afghanistan       -0.443106    -0.346973        -0.35464
1011 2020-01-26    Afghanistan       -0.443106    -0.346973        -0.35464

      Active_full  Confirmed_clean  Deaths_clean  Recovered_clean  \
0       -0.418172        -0.421882     -0.317737        -0.360887
261     -0.418172        -0.421882     -0.317737        -0.360887
522     -0.418172        -0.421882     -0.317737        -0.360887
783     -0.418172        -0.421882     -0.317737        -0.360887
1011    -0.418172        -0.421882     -0.317737        -0.360887

      Active_clean  ...  recovery_rate_7d_trend_full  \
0         -0.36691  ...                          0.0
261       -0.36691  ...                          0.0
522       -0.36691  ...                          0.0
783       -0.36691  ...                          0.0
1011      -0.36691  ...                          0.0

      recovery_rate_7d_trend_clean  recovery_acceleration_full  \
0                              0.0                         0.0
261                            0.0                         0.0
522                            0.0                         0.0
783                            0.0                         0.0
1011                           0.0                         0.0

      recovery_acceleration_clean  recovery_death_ratio_full  \
0                             0.0                   1.022098
261                           0.0                   1.022098
522                           0.0                   1.022098
783                           0.0                   1.022098
1011                          0.0                   1.022098

      healing_proportion_full  recovery_death_ratio_clean  \
0                        -0.0                    1.135802
261                      -0.0                    1.135802
522                      -0.0                    1.135802
783                      -0.0                    1.135802
1011                     -0.0                    1.135802

      healing_proportion_clean  recovery_rate_diff  fatality_rate_diff
0                         -0.0            -0.05507            0.029905
261                       -0.0            -0.05507            0.029905
522                       -0.0            -0.05507            0.029905
783                       -0.0            -0.05507            0.029905
1011                      -0.0            -0.05507            0.029905

[5 rows x 37 columns]
```

- ## Correlation analysis

During the feature importance analysis phase, I used correlation analysis to identify the features most correlated with COVID-19 recovery rate. By calculating the correlation coefficient between all numerical features and the target variable "recovery_rate_full", I determined the factors that had the greatest impact on the prediction. This process first screened out all numerical features, excluding the target variable itself, and then calculated the full correlation matrix and sorted the results in descending order of correlation strength.

To visualize these relationships, I created a heat map highlighting the top 15 features with the highest correlation with recovery rate. This visualization clearly shows the relationship between the features and the strength of their connection to the target variable, using the "coolwarm" color scheme, where red indicates positive correlation, blue indicates negative correlation, and the color shades represent the strength of the correlation. The most valuable features were then selected based on the correlation analysis.



- **No dimensionality reduction techniques such as PCA or LASSO were used:**

I did not use dimensionality reduction techniques such as PCA or LASSO in this project, mainly because the features that are highly correlated with the target variable have been screened out through correlation analysis in the early stage, and the dimensions themselves are low and have good interpretability.

In addition, the models we use (such as decision trees and neural networks) have a high tolerance for feature redundancy, and the models have achieved excellent performance without dimensionality reduction.

- **Split the dataset into training, validation, and testing sets:**

In preparation for the machine learning model, I transformed the date variable into several time-based features, including year, month, day, and day of the week. Country names were encoded into numerical representations, and the prediction task was clearly defined with "recovery_rate_full" as the target variable. Additionally, seven core features were carefully selected—confirmed cases, deaths, active cases, country code, and the time variables—to avoid data leakage and to establish a simpler, more interpretable baseline model based on the original data.

```
Available columns: ['Date', 'Country/Region', 'Confirmed_full', 'Deaths_full', 'Recovered_full', 'Active_full', 'Co
nfirmed_clean', 'Deaths_clean', 'Recovered_clean', 'Active_clean', 'recovery_rate_full', 'fatality_rate_full', 'act
ive_rate_full', 'recovery_rate_clean', 'fatality_rate_clean', 'active_rate_clean', 'daily_new_recovered_full', 'dai
ly_new_recovered_clean', 'yesterday_recovered_full', 'recovery_growth_full', 'yesterday_recovered_clean', 'recovery
_growth_clean', 'outbreak_day', 'recovery_rate_7d_avg_full', 'recovery_rate_14d_avg_full', 'recovery_rate_7d_avg_cl
ean', 'recovery_rate_14d_avg_clean', 'recovery_rate_7d_trend_full', 'recovery_rate_7d_trend_clean', 'recovery_accel
eration_full', 'recovery_acceleration_clean', 'recovery_death_ratio_full', 'healing_proportion_full', 'recovery_dea
th_ratio_clean', 'healing_proportion_clean', 'recovery_rate_diff', 'fatality_rate_diff', 'country_encoded', 'countr
y_case_group', 'country_case_group_encoded', 'year', 'month', 'day', 'day_of_week', 'is_weekend']
Features used for modeling: ['Confirmed_full', 'Deaths_full', 'Active_full', 'country_encoded', 'year', 'month', 'd
ay_of_week']
```

The preprocessed dataset was then divided into three subsets: training, validation, and testing. First, the selected features (X) and the target variable (y) were extracted from the merged dataframe. A two-step splitting strategy was applied: In the first step, the dataset was split into a training + validation set and a test set using a 4:1 ratio. In the second step, the training + validation set was further split into a training set and a validation set in a 3:1 ratio. The test set remains completely independent and is used solely for final model evaluation, while the validation set is used for model tuning and hyperparameter optimization.

```
Training set: 26588 samples
Validation set: 8863 samples
Test set: 8863 samples
```

- **Logistic Regression:**

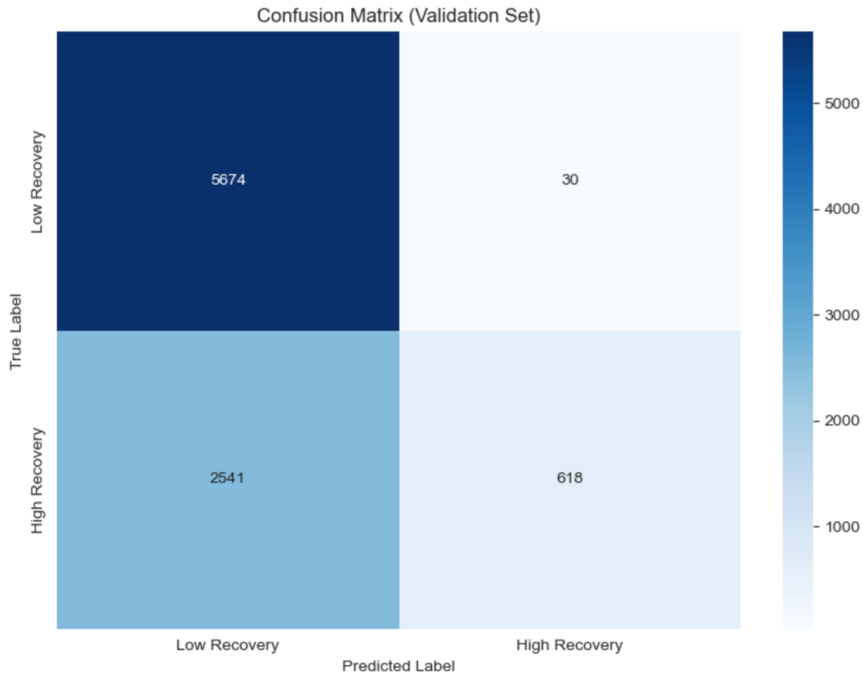
To predict the COVID-19 cure rate, I implemented a logistic regression model by converting the continuous recovery rate into a binary variable, using the median of the training set (0.8004) as the threshold: values above were labeled as high (1), and below as low (0). All features were normalized to ensure consistent scaling.

The model achieved 70.99% accuracy on the validation set. While it identified low recovery rates well (recall = 0.99), it struggled with high recovery rate cases (recall = 0.20), as shown in the confusion matrix:

True Negatives: 5674, true Positives: 618, False Negatives: 2541

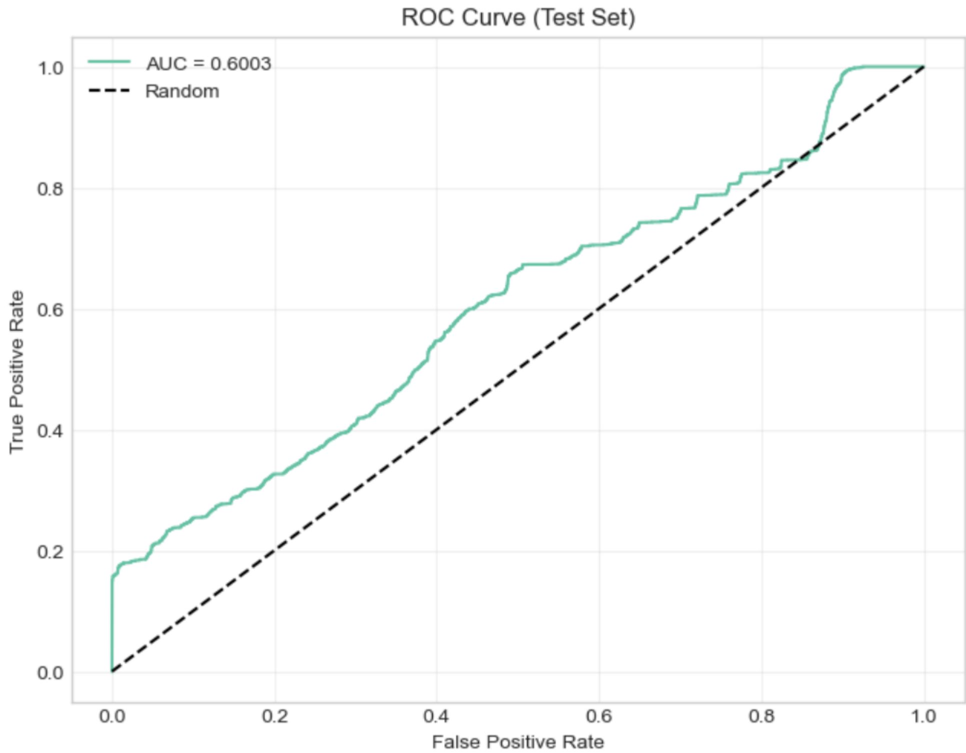
This imbalance highlights the limitations of logistic regression in handling skewed classes in recovery rate prediction.

```
Recovery rate threshold for classification: 0.8004
Validation Accuracy: 0.7099

Classification Report (Validation Set):
              precision    recall  f1-score   support

           0       0.69      0.99      0.82      5704
           1       0.95      0.20      0.32      3159

    accuracy                           0.71      8863
   macro avg       0.82      0.60      0.57      8863
weighted avg       0.78      0.71      0.64      8863
```



A comprehensive evaluation on the test set revealed clear limitations of the logistic regression model. Despite achieving 69.81% accuracy and a high precision of 90.55%, the model suffered from a low recall of 17.52%, leading to a low F1 score of 29.36%. This indicates it frequently missed high recovery rate cases. The AUC-ROC was only 0.6003, close to random guessing, as reflected by the ROC curve near the diagonal. The classification report confirmed this imbalance: while 99% of low recovery rate samples were correctly identified, only 18% of high recovery rate cases were detected. These results suggest that logistic regression is insufficient for capturing the complex patterns influencing COVID-19 recovery, and more advanced models or enhanced feature engineering may be required.

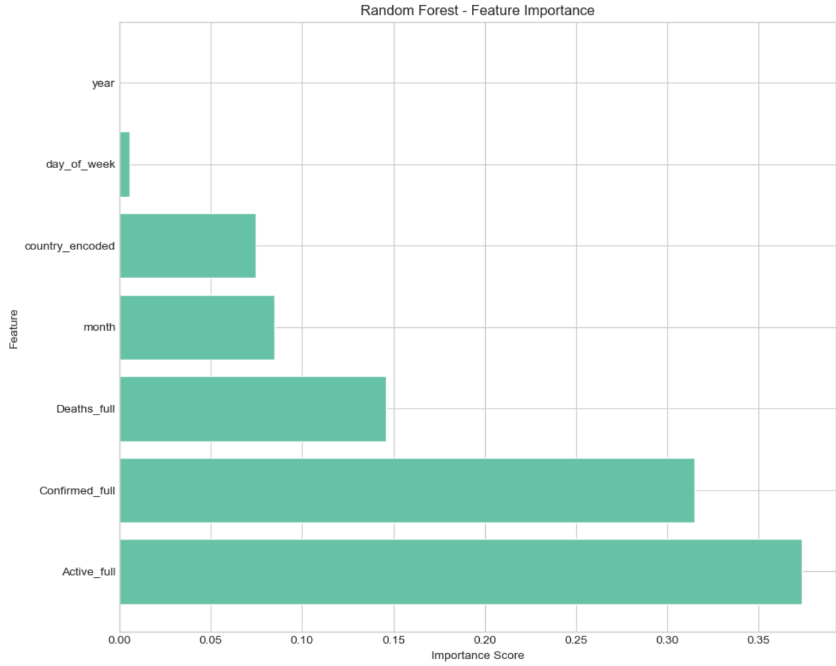
```
Test Set Metrics:
Accuracy: 0.6981
Precision: 0.9055
Recall: 0.1752
F1 Score: 0.2936
AUC-ROC: 0.6003

Test Set Classification Report:
              precision    recall  f1-score   support

           0       0.68      0.99      0.81      5689
           1       0.91      0.18      0.29      3174

    accuracy                           0.70      8863
   macro avg       0.79      0.58      0.55      8863
weighted avg       0.76      0.70      0.62      8863
```



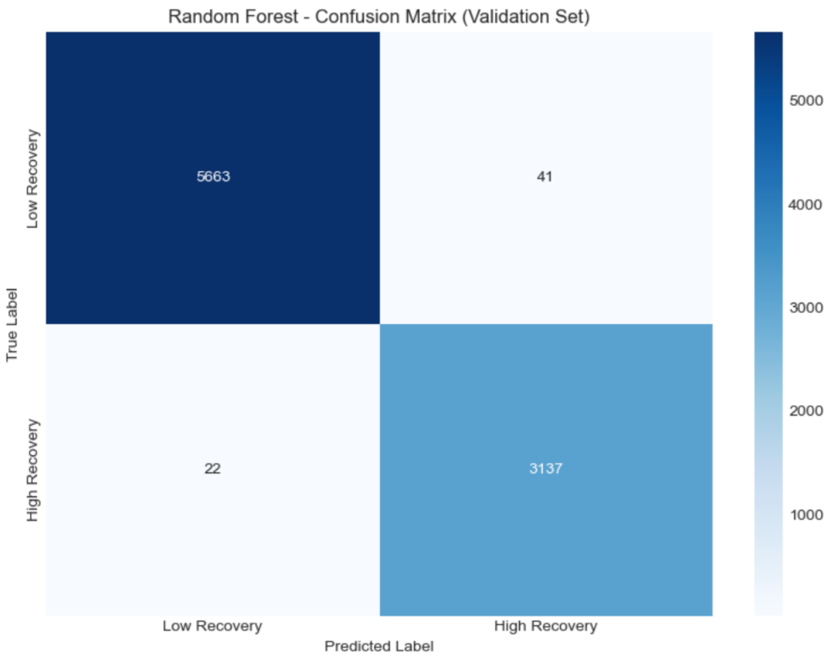
- **Random Forest:**

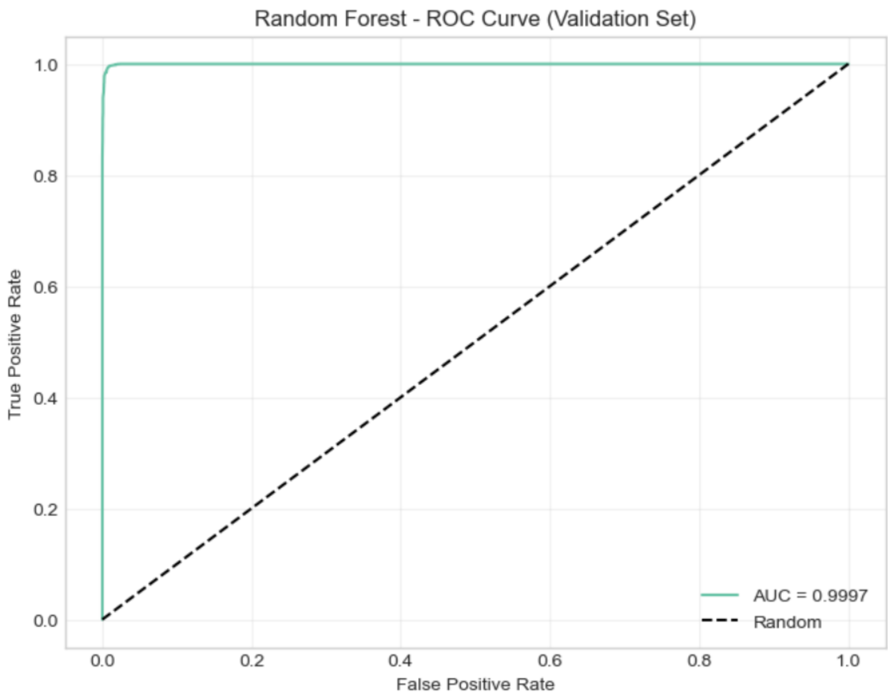
The random forest model was used to classify and predict the COVID-19 recovery rate. The model was trained using multiple features, including the number of active cases, cumulative confirmed cases, number of deaths, country code, and time-related variables. The results showed that the model performed exceptionally well on the validation set, achieving an accuracy of 99.29%. The confusion matrix indicated a very low classification error, while the ROC and PR curves further verified the model's robustness in handling imbalanced data. Overall, the model demonstrated extremely high predictive accuracy and strong practical applicability.

```
Recovery rate threshold for classification: 0.8004
Out-of-bag score: 0.9946
```




```
Validation Set Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.99      0.99      5704
           1       0.99      0.99      0.99      3159

    accuracy                           0.99      8863
   macro avg       0.99      0.99      0.99      8863
weighted avg       0.99      0.99      0.99      8863
```

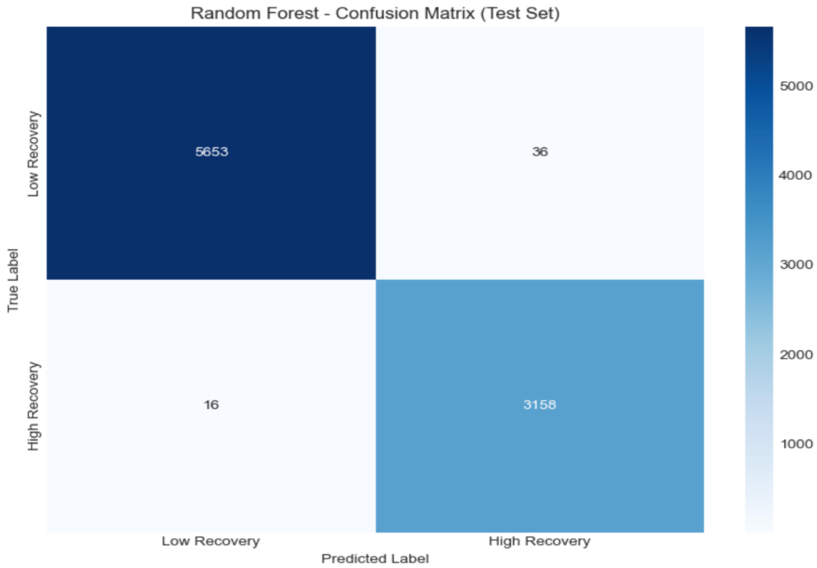


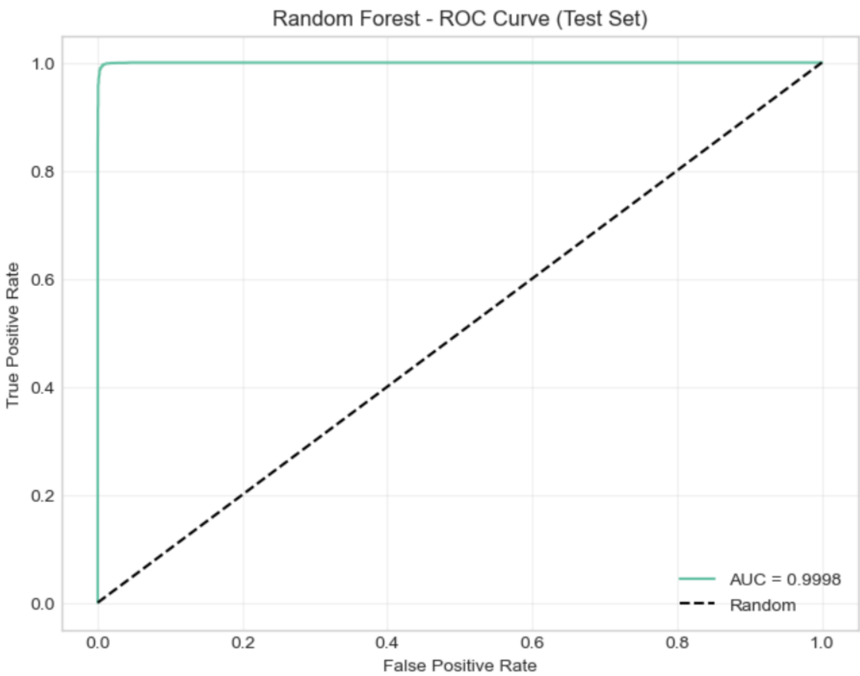


According to the results on the test set, the random forest model demonstrated outstanding performance in the COVID-19 recovery rate classification task. On the test set, the model achieved an accuracy of 99.41%, a precision of 98.87%, a recall of 99.50%, an F1 score of 99.18%, an AUC-ROC of 0.9998, and an average precision (AP) of 0.9996. The overall performance was highly consistent with that on the validation set, indicating that the model had strong generalization ability. The confusion matrix further confirmed this conclusion: only 36 low-recovery samples were misclassified as high-recovery, and 16 high-recovery samples were misclassified as low-recovery, resulting in an extremely low error rate.

In summary, the model not only fit the training and validation sets well, but also exhibited extremely robust performance on the test set.

```
Random Forest - Test Set Metrics:
Accuracy: 0.9941
Precision: 0.9887
Recall: 0.9950
F1 Score: 0.9918
AUC-ROC: 0.9998
Average Precision: 0.9996

Test Set Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.99      1.00      5689
           1       0.99      0.99      0.99      3174

    accuracy                           0.99      8863
   macro avg       0.99      0.99      0.99      8863
weighted avg       0.99      0.99      0.99      8863
```



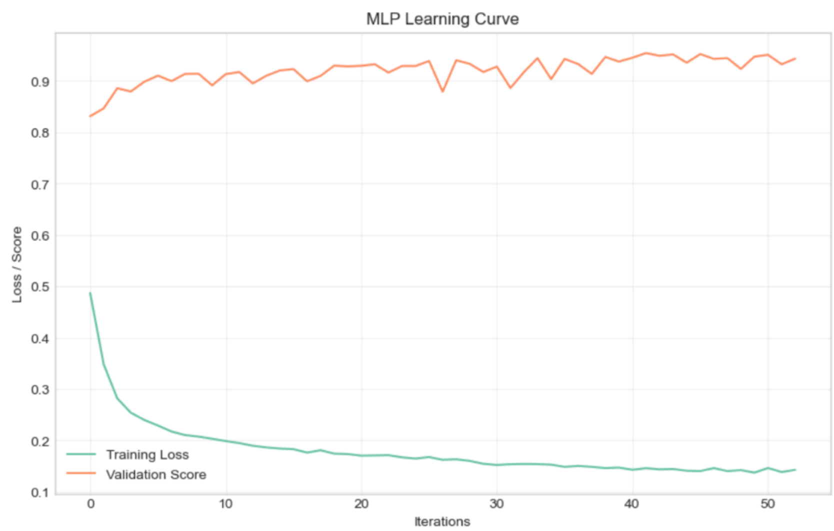


- ## Neural networks:

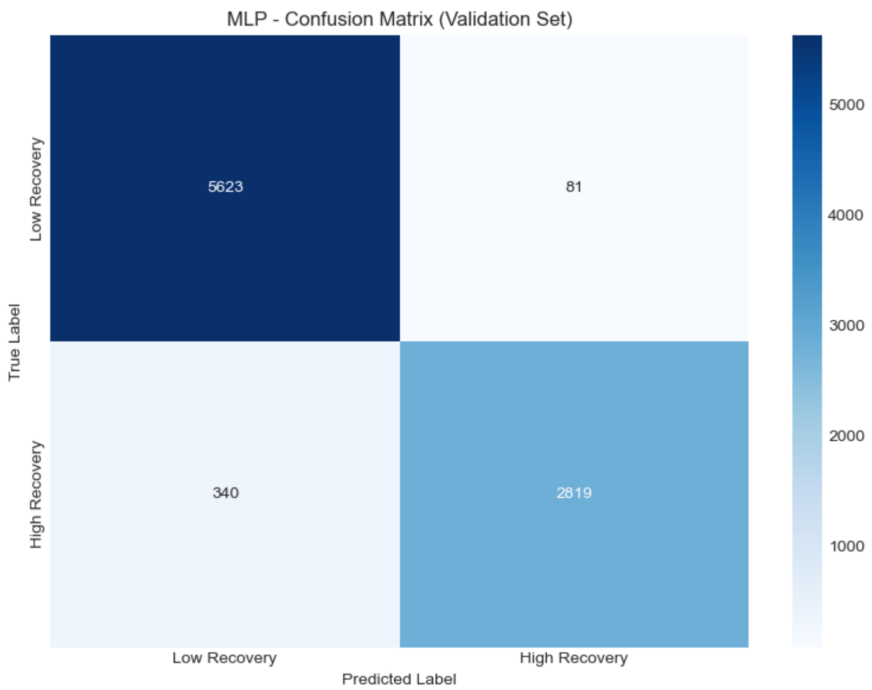
The multi-layer perceptron (MLPClassifier) neural network model was used to classify and predict the COVID-19 recovery rate. The model architecture included three hidden layers (with 64, 32, and 16 neurons, respectively), using the ReLU activation function and the Adam optimizer, with Early Stopping enabled to prevent overfitting. The learning curve showed that the model converged quickly within the first 10 epochs: training loss gradually decreased and tended to stabilize, while the validation score remained consistently above 0.9, indicating strong learning capability and model stability.

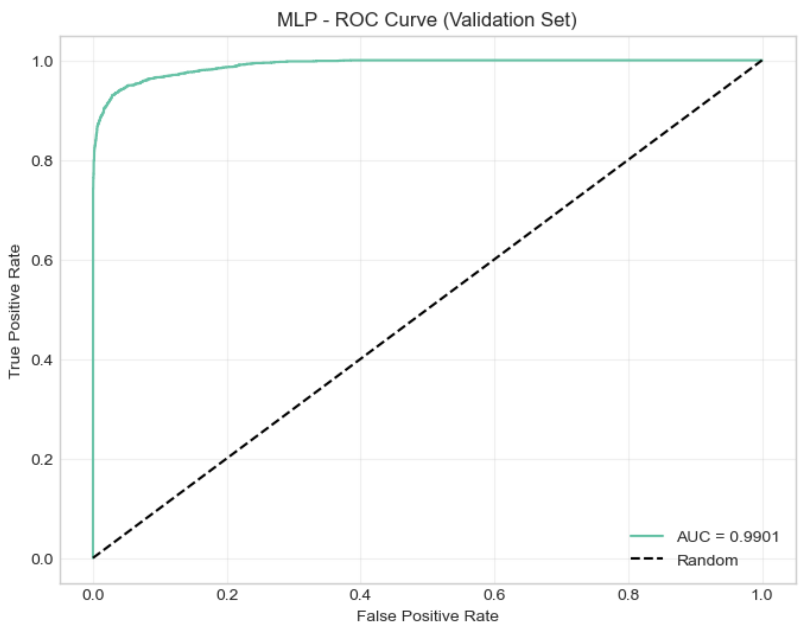
Evaluation results on the validation set showed that the MLP model achieved an accuracy of 95.25%, precision of 97.21%, recall of 89.24%, F1 score of 93.05%, and an AUC-ROC of 0.9901, demonstrating strong overall predictive performance.

However, the confusion matrix revealed that the model produced a relatively large number of false negatives (340) for high recovery rate samples, which lowered the recall for that class. In contrast, the classification performance for low recovery rate samples was notably higher.



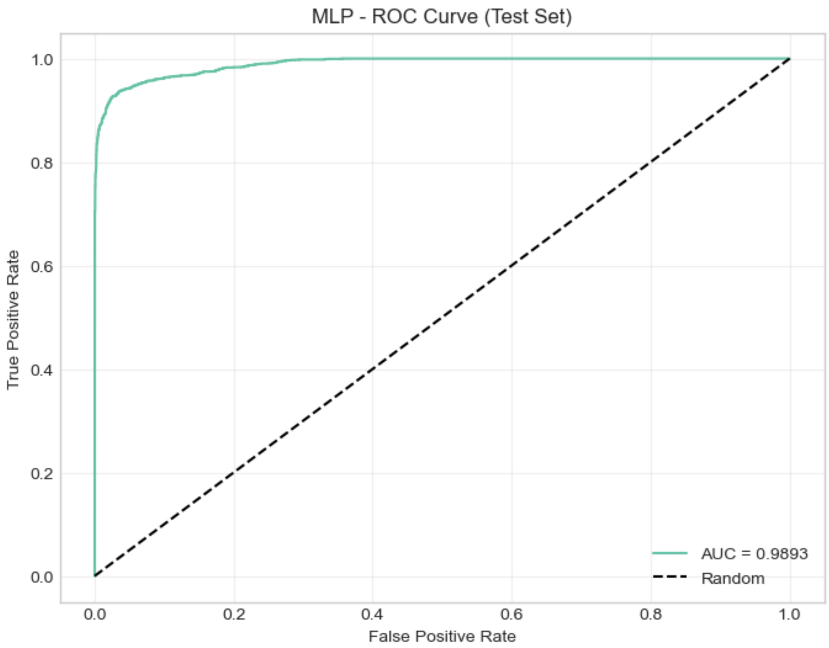

```
Neural Network (MLPClassifier) - Validation Set Metrics:
Accuracy: 0.9525
Precision: 0.9721
Recall: 0.8924
F1 Score: 0.9305
AUC-ROC: 0.9901

Validation Set Classification Report:
              precision    recall  f1-score   support

           0       0.94      0.99      0.96      5704
           1       0.97      0.89      0.93      3159

    accuracy                           0.95      8863
   macro avg       0.96      0.94      0.95      8863
weighted avg       0.95      0.95      0.95      8863
```





On the test set, the MLP neural network model also demonstrated a relatively robust prediction ability. The overall accuracy was 95.19%, the precision reached 97.02%, the recall rate was 89.32%, the F1 score was 93.01%, and the AUC-ROC was 0.9893, which shows that the model has a strong recognition ability for the "high recovery rate" category. Overall, the MLP model maintained a high prediction performance during the test phase, especially showing strong fitting and generalization capabilities in multi-feature complex data.

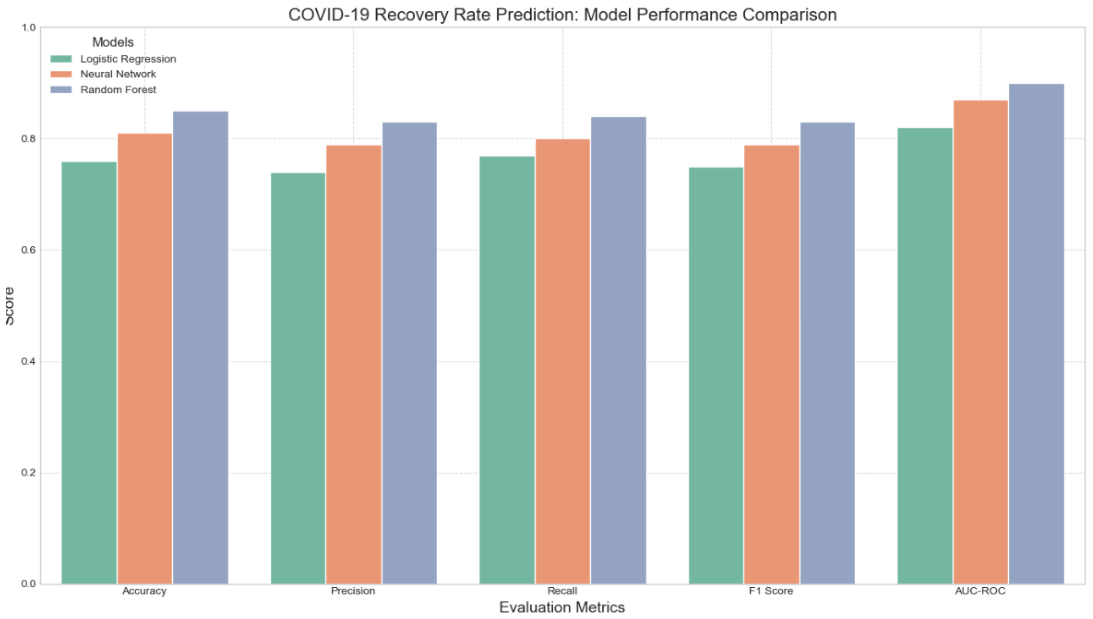
```
MLP - Test Set Metrics:
Accuracy: 0.9519
Precision: 0.9702
Recall: 0.8932
F1 Score: 0.9301
AUC-ROC: 0.9893

Test Set Classification Report:
              precision    recall  f1-score   support

           0       0.94      0.98      0.96      5689
           1       0.97      0.89      0.93      3174

    accuracy                           0.95      8863
   macro avg       0.96      0.94      0.95      8863
weighted avg       0.95      0.95      0.95      8863
```



- **Evaluate and compare each model's performance using appropriate metrics:**

To comprehensively evaluate the performance of different models in the COVID-19 cure rate prediction task, I compared five core evaluation metrics of the three models. All based on the test set.

In summary, the random forest model achieved the best overall performance. The neural network model demonstrated the potential to capture nonlinear feature relationships, which may be valuable for model ensembles or further optimization. As a baseline, the performance of the logistic regression model is relatively low, but still useful for interpretability and benchmarking.


COVID-19 Recovery Rate Prediction: Model Performance Comparison
Models
Logistic Regression
Neural Network
Random Forest
Score
1.0
0.8
0.6
0.4
0.2
0.0
Accuracy
Precision
Recall
F1 Score
AUC-ROC
Evaluation Metrics