Q1:
a：
false negative
It is better to predict wrongly than to let go of a fraud.

b：
分割线分开output 0.7-0.8

c:
1（在最左边分割

d:
0（在最右边分割

e:
Nblue，对于ROC_AUC，曲线与右边和下边组合图形的面积越大说明model性能越好，越能代表所有东西

Q2：
a:计算题
b:

- Limitation of Black-Scholes Model
  - The log-normal assumption does not capture extreme movements such as stock market crashes.
  - Volatility and interest rate are not constant throughout the option's life in practice
  - The model is very sensitive to the value of volatility which is difficult to estimate
  - The model normally uses historic volatility for the option price for a future period
  - Application to non-traded assets is questionable (e.g. employee stock options)

Q3:
a:
No, this model fit the data too well, in another way, it means overfit, which means it will do very bad result for the test data, as well as in the real data

b:
1. **Increase training data**: Collecting more training data can help the model learn a more generalized representation of the underlying patterns in the data, reducing the chances of overfitting.

2. **Cross-validation**: Instead of relying solely on a single train-test split, use cross-validation techniques like k-fold cross-validation. This approach divides the data into multiple folds, allowing the model to be trained and evaluated on different subsets of the data. It provides a more reliable estimate of the model's performance and helps mitigate overfitting.

3. **Regularization**: Regularization techniques, such as L1 and L2 regularization, add a penalty term to the loss function to discourage complex models with high coefficients.

Regularization helps prevent overfitting by reducing the model's reliance on individual features and encouraging more generalizable patterns.

4. **Feature selection**: Carefully select relevant features and remove any unnecessary or redundant ones. The goal is to keep the most informative features while reducing the risk of overfitting caused by noise or irrelevant information.

5. **Early stopping**: Monitor the model's performance on a validation set during training and stop training when the performance begins to deteriorate. This prevents the model from over-optimizing on the training data.

6. **Ensemble methods**: Combine the predictions of multiple models, such as using bagging (e.g., Random Forests) or boosting (e.g., Gradient Boosting), to reduce overfitting. Ensemble methods leverage the diversity of multiple models to improve generalization.

7. **Simpler models**: Use simpler models with fewer parameters to reduce the model's capacity to fit the training data too closely. This approach can help avoid overfitting, especially when the data is limited.

8. **Data preprocessing**: Apply techniques like scaling, normalization, or dimensionality reduction (e.g., PCA) to preprocess the data and reduce the risk of overfitting.

9. **Hyperparameter tuning**: Regularize the model by tuning hyperparameters, such as the learning rate, regularization strength, or network architecture, using techniques like grid search or randomized search. This optimization process helps find the best parameter values that balance model complexity and performance.