1. Do we have imbalanced data? What do you think is the best way to remedy this (https://imbalanced-learn.org/stable/references/generated/imblearn.over\_sampling.SMOTE.html)
   1. Drop rows?
   2. Reduce size of class 1 data – under-sampling (sklearn – review smoteen as well)
      1. try to reduce it to 3 million Take out last one million rows
      2. Combo over/undersampling
   3. Raise the 0 rows – over-sampling
      1. Boosting and bagging techniques
2. Feature importance

PowerPoint:

## TensorFlow testing:

* Compiling the model
  + Modified crossentropy – result is 20 point drop in accuracy

# model.compile(optimizer='adam', loss='sparse\_categorical\_crossentropy', metrics=['accuracy'])

model.compile(optimizer='adam', loss='binary\_crossentropy', metrics=['accuracy'])

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

## Gradient Boosting Machines (GBMs) with neural network backbones

# Compile the model

model.compile(loss='categorical\_crossentropy', optimizer='adam', metrics=['accuracy'])

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

model.compile(loss='binary\_crossentropy', optimizer='adam', metrics=['accuracy'])

A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

The ROC AUC score (Receiver Operating Characteristic - Area Under the Curve) is a performance metric used for evaluating the ability of a binary classification model to distinguish between positive and negative classes. Here's a breakdown of how to interpret a ROC AUC score of 0.885326903249336:

**Understanding ROC and AUC**

* **ROC Curve**: The ROC curve plots the True Positive Rate (TPR) against the False Positive Rate (FPR) at various threshold settings. It shows the trade-off between sensitivity (recall) and specificity (1 - FPR).
* **AUC**: The area under the ROC curve (AUC) summarizes the ROC curve into a single value, which represents the likelihood that the model will correctly distinguish between a randomly chosen positive instance and a randomly chosen negative instance.

**Interpreting the ROC AUC Score**

* **Range**: The ROC AUC score ranges from 0 to 1.
  + **0.5**: The model's performance is no better than random guessing.
  + **1.0**: The model has perfect discrimination and is able to perfectly separate the positive and negative classes.
  + **< 0.5**: Indicates that the model is performing worse than random guessing, which suggests that the model might be inverted.

**Interpretation of 0.885**

A ROC AUC score of 0.885326903249336 can be interpreted as follows:

* **High Performance**: A score of 0.885 indicates that the model has a high capability of distinguishing between the positive and negative classes.
* **Model Quality**: This score suggests that approximately 88.5% of the time, the model will correctly rank a randomly chosen positive instance higher than a randomly chosen negative instance.
* **Practical Implication**: In practical terms, this is a strong performance, especially in contexts where distinguishing between classes is critical (e.g., medical diagnosis, fraud detection).

**Contextual Consideration**

While a score of 0.885 is generally considered good, it's important to consider the context of the problem:

* **Domain Requirements**: Different domains might have different thresholds for what is considered an acceptable ROC AUC score. For instance, in medical diagnostics, even higher scores might be required.
* **Baseline Comparison**: Compare the ROC AUC score to a baseline model (such as a logistic regression or a simple decision tree) to see if the model provides a significant improvement.
* **Class Imbalance**: In cases of imbalanced datasets, the ROC AUC score can sometimes be misleading. It's important to also look at other metrics such as precision, recall, and F1-score.

**Example Code to Compute ROC AUC Score**

Here is how you can compute the ROC AUC score using Scikit-learn:

python

Copy code

from sklearn.metrics import roc\_auc\_score

# Assuming y\_test are the true labels and y\_pred\_prob are the predicted probabilities for the positive class

roc\_auc = roc\_auc\_score(y\_test, y\_pred\_prob)

print('ROC AUC Score:', roc\_auc)

**Summary**

A ROC AUC score of 0.885326903249336 indicates that your model has a strong ability to discriminate between the positive and negative classes. This is a good sign of model performance, especially if it meets or exceeds domain-specific benchmarks and significantly improves over baseline models.

## Multi-layer Perceptron (MLP) Model

Predict and Evaluate the MLP Model

A screenshot of a computer program

Description automatically generated

Check to see if data is balanced

A screen shot of a computer code

Description automatically generated

TensorFlow / Keras

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated