**Question 3**

**Question 3:** The bank is clearly interested in minimizing the losses from fraud, but they are also aware that false positives take time to sort out and impact the customer experience.

1. Is the area under the ROC curve the most suitable metric? If so why, and if not, then how should the model be evaluated?

**Answer:** Auc curves tell us the relationship between TPR and FPR and focus on how well a model discriminates between two classes. However, they are not always necessarily a good metric for ml algorithm tuning or as a measure.

For example, a model that predict probability of 0.501 for positive and 0.499 for negative will have a perfect auc score of 1. However, the probability is not helpful and hence other probability difference minimising metrics may also be more useful i.e. Brier's score.

Such probabilistically more accurate models maybe more relevant to the fraud detection rather than simple and plain discriminator models with low discrimination threshold.

Let's see the same demonstration with our empirical example below

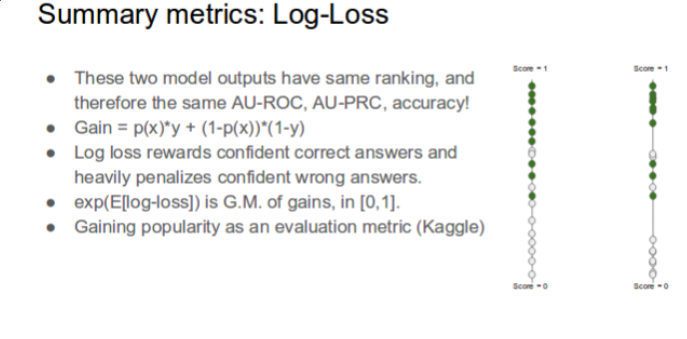
from IPython.display import Image

print('Same AUC score but "MODEL 2" is significantly far better than "MODEL 1"')

Image("images/auc score misleading.png")

# REF: http://cs229.stanford.edu/section/evaluation\_metrics.pdf

Same AUC score but "MODEL 2" is significantly far better than "MODEL 1"



**b) What is the difference between a metric and a loss function?** **Answer:**

1. Metric: Metric is a function used to judge performance of model and is not used in the model training to improve the model.
2. Loss function: Loss function and metric are similar but the only difference is that the loss function is used in the model training to improve model over iterations

For example, we used binary cross entropy as our loss function while using accuracy and auc as metrics for our keras based NN classifier

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

**c) How might your approach in Question 1 change if you had used your desired metric? Answer:** Because metric is not used in the model tuning or learning process as described in earlier question b), it might not have any impact at all.

However, if we use the desired loss function such as Brier score, then it would have significant impact. Brier Score is given by

Brier Score = Sum\_i(observed\_prob - expected\_output)^2

Hence, unlike in the earlier instance where our probabilities were clustering to 0.5 but yielding a good auc score, we would have obtained good probabilistic classifier that would have maximised probabilistic separation i.e. fraud's probability will be clustered towards 1 and vice versa, rather than being

* Predicted Probability -------- vs -------- Actual Fraud Label
* [0.49933839 0.50066161] ----- vs -------- 1
* [0.49972204 0.50027796] ----- vs -------- 1
* [0.49958261 0.50041739] ----- vs -------- 1
* [0.49969233 0.50030767] ----- vs -------- 1
* [0.49959703 0.50040297] ----- vs -------- 1

print('With Brier Score expecting "MODEL B" vs auc tuned "MODEL A"')

Image("images/brier\_score vs auc.png")

With Brier Score expecting "MODEL B" vs auc tuned "MODEL A"



**d) How would you decide at which threshold (probability) to block the transaction?** **Answer :** We can perform threshold scanning taking into accounting the bank's pain threshold for the following

1. Customer Inconvenience i.e. minimise FP or Genuine trans that is tagged Fraud.
2. Fraud loss i.e. minimise FN or Fraud trans that is Unidentified

To decide which threshold to block the transactions at, several plots can be useful.

The plots have error rate on y-axis and threshold on the x-axis with following possible metrics combination, as plotted in the graph

1. Precision vs Recall
   * Precision: maximise: fraction of true frauds vs all predicted frauds
   * Recall: maximise: fraction of predicted frauds vs all true frauds
2. FNR vs FPR i.e. error for fraud trans vs error for non-fraud trans.
   * FNR = FN / (FN + TP) - capture more Fraud Trans (FN)
   * FPR = FP / FP + TN - with low false tagging

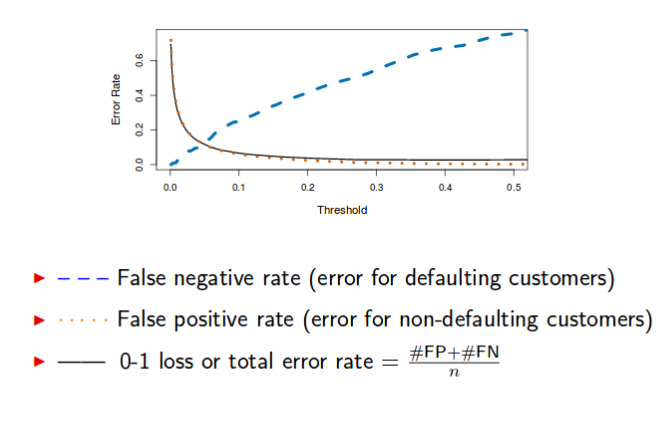
In case, when one of the pain threshold (precision or recall) is hard-fixed, we could optimise for the other attribute along the threshold space, thus reaching to optimal threshold probability.

Now once the plots are done, based on the business stakeholder’s requirement across threshold, we will have to conservatively set the threshold.

print ('Sample example of using plots to finding Threshold. FPR vs FNR plot')

Image ('images/FPR vs FNR.png')

Finding Threshold using plots. Sample FPR vs FNR plot



**To Dos:**

**TD 1: Concept Drift: Sliding Window / Propagate & Forget Strategy**

In this assignment, the momentary priority was on the auc maximisation, while the long-time priority or the ultimate mission was to develop a system capable to withstand concept drift.

To gain the ultimate mission, our currently developed static models would be highly inefficient and hence models able to withstand the concepts drift are required. They can be

**1.0 Sliding window:**

1. Build daily model. (model build timeframe to be optimally determined)
2. Take the latest n models and ensemble them.
3. Put more weight on latest model’s prediction.
4. Perform Prediction

**2.0 Propagate and Forget:**

1. Like sliding window, the only difference is that unlike in sliding window where we disregard the old fraud data, in this approach we carry all those fraud data over any period. As for the non-fraud data only, the latest n periods non- fraud data are considered.

**TD2: Classifier for Alert Fixes:**

Separate classifier needs to be built to deal with manually annotated data because data distribution is different for the manually annotated or corrected datasets than other normal.