**Problem statement 3:**

**Now that you have trained a model, it is time to put it in production (a web app where a customer can apply and get an immediate loan approval or rejection decision). How would you come up with an appropriate cut off that separates those approved and those rejected? We are not necessarily expecting code for this section, just an analytical explanation for how you would solve this problem in a real business setting.**

**App:**

We can upload the data into AWS S3.

EDA, Feature Engineering, and modeling are tested in local machine using Spark, and are then deployed to AWS-EMR.

(Simple way for app implementation)

**Concerns before we proceed:**

unbalanced data leads to decrease in model performance.

**When the model is in production, it’s predicting on unseen data. The main point of model validation is to estimate how the model will generalize to new data. If the decision to put a model into production is based on how it performs on a validation set, it’s critical that oversampling is done correctly.**

**Steps to follow:**

Up-sample the unbalanced dataset after separating the training and validation set and then proceed with model building and evaluation.

**Reference paper for this problem:**

<https://www.stat.berkeley.edu/~breiman/randomforest2001.pdf>