Proposal: Titanic: Machine Learning from Disaster

1. Domain background

I plan to tackle the “Titanic: Machine Learning from Disaster” challenge on Kaggle!

From Kaggle: https://www.kaggle.com/c/titanic/overview

> “The sinking of the Titanic is one of the most infamous shipwrecks in history. On April 15, 1912, during her maiden voyage, the widely considered “unsinkable” RMS Titanic sank after colliding with an iceberg. Unfortunately, there weren’t enough lifeboats for everyone onboard, resulting in the death of 1502 out of 2224 passengers and crew.
> While there was some element of luck involved in surviving, it seems some groups of people were more likely to survive than others.
> In this challenge, we ask you to build a predictive model that answers the question: “what sorts of people were more likely to survive?” using passenger data (ie name, age, gender, socio-economic class, etc).”

This is very interesting to me as I would like to focus more on this area of using Machine Learning to make predictions. Also, I would like to get involved with competitions on Kaggle.

The gist of the challenge seems to be binary classification.

2. Problem statement

Given passenger data, I will predict whether a passenger would or would not have survived the crash of the titanic. As Kaggle puts it: “what sorts of people were more likely to survive?”

This can be solved using binary classification, which we have experimented with in this very course.

3. Datasets and inputs

The data sets are provided by Kaggle, both a “train.csv” and a “test.csv”. This data was gathered by Kaggle for this use case.

From Kaggle:

> “In this competition, you’ll gain access to two similar datasets that include passenger information like name, age, gender, socio-economic class, etc. One dataset is titled `train.csv` and the other is titled `test.csv`.
>
> Train.csv will contain the details of a subset of the passengers on board (891 to be exact) and importantly, will reveal whether they survived or not, also known as the “ground truth”.”

4. Solution statement

The solution to the problem is to accurately predict survival rates for the "unseen" data. I am able to use the given features and features that I can engineer myself.

From Kaggle:

> "The training set should be used to build your machine learning models. For the training set, we provide the outcome (also known as the "ground truth") for each passenger. Your model will be based on "features" like passengers' gender and class. You can also use feature engineering to create new features.
>
> The test set should be used to see how well your model performs on unseen data. For the test set, we do not provide the ground truth for each passenger. It is your job to predict these outcomes. For each passenger in the test set, use the model you trained to predict whether or not they survived the sinking of the Titanic."
>
> "The `test.csv` dataset contains similar information but does not disclose the "ground truth" for each passenger. It's your job to predict these outcomes.
>
> Using the patterns you find in the train.csv data, predict whether the other 418 passengers on board (found in test.csv) survived."

5. Benchmark model

The benchmark model is provided by Kaggle as part of the challenge process, a leaderboard and score is provided.

We can also consider that the sinking of the titanic was a real passed event, in which about 1,500 people died and about 700 survived. The model should not deviate from this statistic.

6. Evaluation metrics

From Kaggle:

> "Your score is the percentage of passengers you correctly predict. This is known as accuracy."

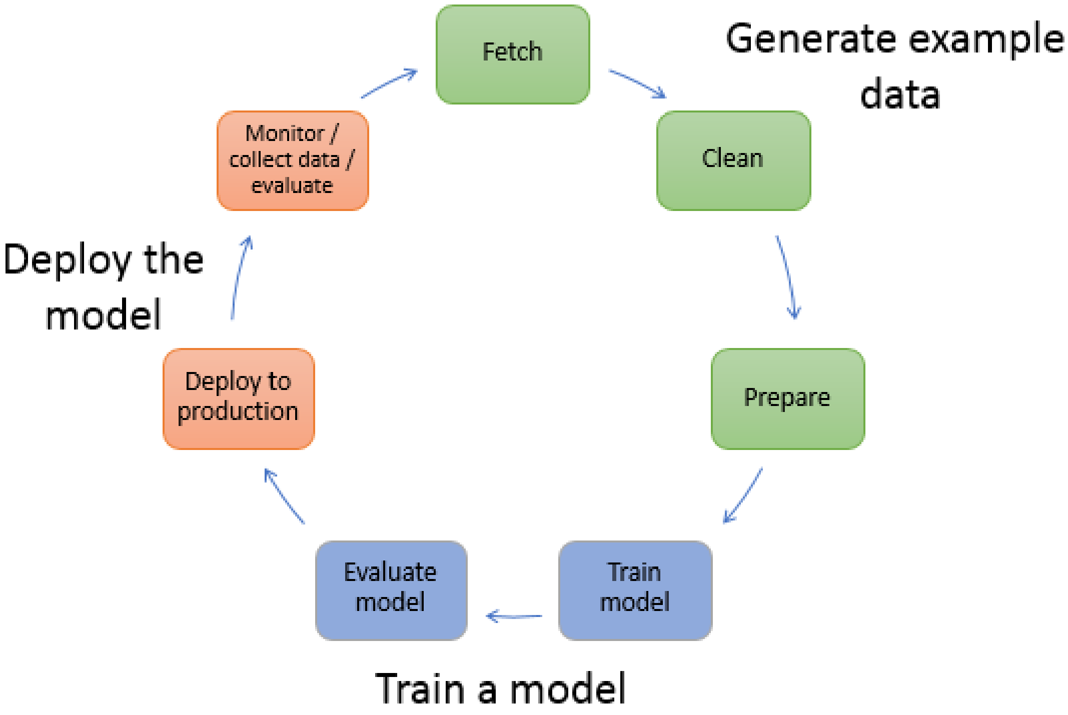
As this is binary classification, the accuracy metric that we have used throughout this course is the same used in this project, `accuracy = (tp + tn) / (tp + fp + tn + fn)`

7. Project design

The workflow can be broken down into steps represented by the aws machine learning workflow:

- Clean (and explore)
- Prepare and process
- Build and Train (the model)
- Evaluate (the model)

- Deploy (Due to the nature of Kaggle, this step will not be a deploy to production, but rather a figurative deploy, i.e "submission of the model")
- Evaluate (performance)
- Repeat



*https://docs.aws.amazon.com/sagemaker/latest/dg/how-it-works-mlconcepts.html