Project Proposal

Project Title: **Airline Delays Analysis and Prediction**

Group Members: **Zizheng Que**, **Lingyun Li**, **Shan Chen**, **Yangyang Jiang**

Project Topic:

Inspired by the Kaggle dataset "2019 Airline Delays w/Weather and Airport Detail", our group decided to further investigate the relationship between airport weather conditions and flight delays/cancellations. Noticed that this Kaggle dataset is made by joining two public dataset, using limited attributes in one single year 2019. From the source datasets, we found that the source dataset contains plenty of relevant attributes, which has rich potential that we can do more research by using more attributes and expanding the target year from one single year to several years.

Datasets:

a) Airline On-Time Performance Data (from Bureau of Transportation Statics) [Link]

b) Airport weather Detail [Link]

Proposed Goals:

Our main goal will be analyzing the airline delay based on weather conditions and other airport details, like airport company, airline departure station.

1. Search qualified datasets such as weather dataset, airport dataset, etc.
2. Process the collected data, clean unnecessary rows and columns. Join data table based on same attributes(factors).
3. Analyze and visualize the factors that have influence on airplane delays.
4. Explore the relationships between different factors and delay rates, create visual charts for factors and delay rates.

Stretch Goals:

1. Predictive analysis: Use past years data to train the model and make predictions according to new factors.
2. Design and create an User Interactive website which allows users to type in relevant data and get the prediction result.

Technologies:

Use **PySpark** and other Python libraries(*e.g.* **Selenium**, **Beautiful Soup**) as needed to process our datasets.

Use **Amazon S3** and **EMR** to store and process our huge datasets.

Use **PyTorch** to implement DNN and training prediction model on our dataset and find the appropriate hyper-parameters.

Use **HTML** and **CSS** to build a front-end interactive website. Use **Flask** framework to build the backend. Use front-end visualization technology to visualize the final result.

More technologies may be added as we work on this project.