PlayerUnknown's Battlegrounds Data Analysis Using Spark

Team 6: Beiwen Guo
Zhenyu Zhu

About PUBG :


Zharki
Severny
Stalber
Yasnaya Polyana
Georgopol
Rozhok
Lipovka
Gatka
Pochinki
Safe Zone
Mylta
Mylta Power
Primorsk
Sosnovka Island
Novorepnoye
Sosnovka Military Base

Our Goals

- Processing and visualizing matches data from the popular video game PUBG to determine if specific strategies or behaviors are related to wining the game.
- Gain more hands-on experience of coding with Scala.
- Learn to use Spark to build big data engineer project.
- With the assistance of Spark and Scala, train our skills of ingesting and processing data with cloud resources.

Use Cases

Business users:

- By uploading a number of matches data to our system, they can cluster all players from these matches by play style/ strategy. Corresponding plots will be provided.

- By inputing corresponding values of players' performances in one single match, to predict if such players can be winners in this match by using trained classification model our project will build.

Methodology

Filtering data	Discard data which will not be used and ingest data into a useable form .
Spark MLlib	Building and training model using Spark MLlib.
AWS	Using AWS resource to process large scale data.
Apache Zeppelin	Using Apache Zeppelin to generate visualization output.
Hortonworks Data Cloud	With assistance of HDCloud, configure computing resources we need.

Data Source

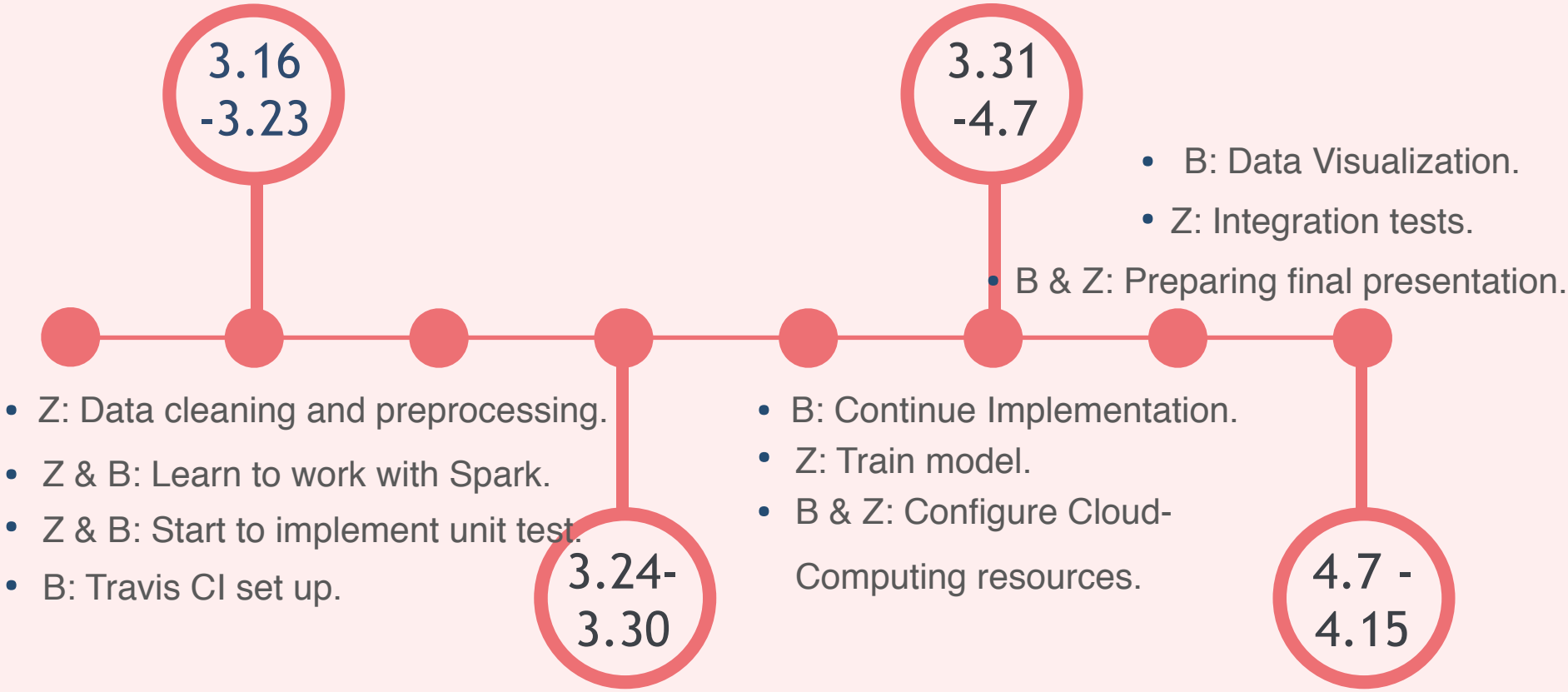
All data are from kaggle called PUBG Match Deaths and Statistics. There are two zip files composed of 10 .csv files. Approximately 10 million rows and 20 GB in all.

Reference:

https://www.kaggle.com/skihikingkevin/pubg-match-deaths

Milestones/sprints

B stands for Beiwen Guo
Z stands for Zhenyu Zhu



- Z & B: Spark MLlib(Clustering&Classification) self-learning.
- Z & B: Starting implementation of ingesting raw data into Spark Dataset.
- B: Zeppelin self-learning.

Programing in Scala

- **Cleaning, parsing and ingesting data**
- **Building ML Model**
- **Unit Test**

Our repository

https://github.com/beiwen/CSYE7200_FinalProject_Team06

Acceptance Criteria

- The speed of our system to finish a visualization work should be faster than 20s/GB.
- Classification model can determine if the user is a winner. The accuracy should be at least 70%.

Thank you!