BIG DATA

Class Project: FPL

K V Subramaniam
Computer Science and Engineering

BIG DATA

Class Project: FPL

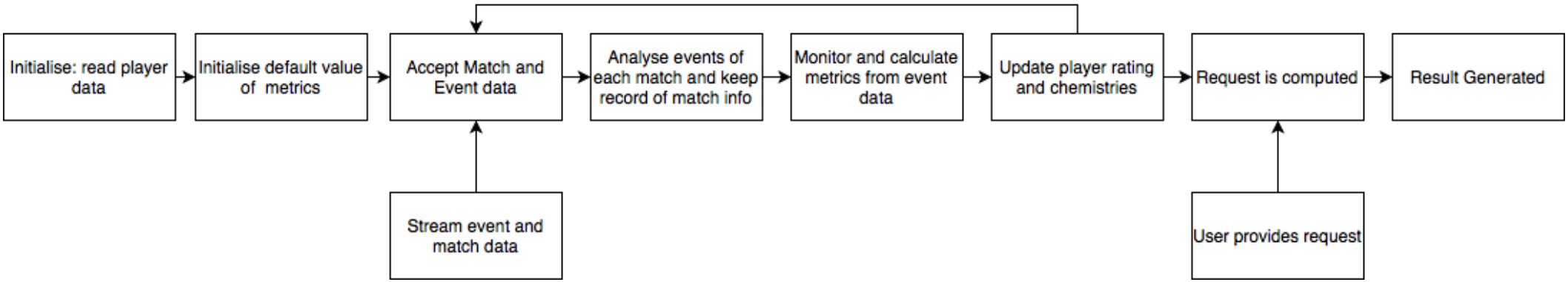
K V Subramaniam
Computer Science and Engineering

BIG DATA

Introduciton

Aim

- Rate and rank players
- Real time analysis of match event data
- Compute strength of team
- Predict rating of player



Requirements

- Use only Spark to do this assignment
- PySpark library can be used
- Use Spark Mllib for clustering and regression
- No native python code will be allowed, all code must be compatible with PySpark and executed on Spark
- Use Streaming spark to accept the streamed match and event data

Initialisation

- Players.csv will provide the background info for all players, this info will be a part of player profile
- A streaming file will be provided that will stream match data on port 6100
- Read match event data from 6100
- At the beginning of a match a match JSON object is sent and then events JSON objects are sent for that match.
- Not all data streamed will be useful, be careful while processing and discard data not required for your use case

BIG DATA

Match JSON

Field	Description
competitionId	the identifier of the competition to which the match belongs to. It is a integer and refers to the field "wyId" of the *competition* document;
date and **dateutc**	the former specifies date and time when the match starts in explicit format (e.g., May 20, 2018 at 8:45:00 PM GMT+2), the latter contains the same information but in the compact format YYYY-MM-DD hh:mm:ss;
duration	the duration of the match. It can be "Regular", "ExtraTime", "Penalties"
gameweek	the week of the league, starting from the beginning of the league;
roundID	indicates the match-day of the competition to which the match belongs to.
seasonId	indicates the season of the match;
status	it can be "Played", "Cancelled", "Postponed" or "Suspended";
venue	the stadium where the match was held (e.g., "Stadio Olimpico");
winner	the identifier of the team which won the game, or 0 if the match ended with a draw;
wyId	the identifier of the match

BIG DATA

Match JSON Contd.

teamsData: it contains several subfields describing information about each team that is playing that match: such as lineup, bench composition, list of substitutions, coach and scores:

Sub-field	Description
hasFormation	it has value 0 if no formation (lineups and benches) is present, and 1 otherwise;
score	the number of goals scored by the team during the match (not counting penalties);
scoreET	the number of goals scored by the team during the match, including the extra time (not counting penalties);
scoreHT	the number of goals scored by the team during the first half of the match;
scoreP	the total number of goals scored by the team after the penalties;
side	the team side in the match (it can be "home" or "away");
teamId	the identifier of the team;
coachId	the identifier of the team's coach;
bench	the list of the team's players that started the match in the bench and some basic statistics about their performance during the match (goals, own goals, cards);
lineup	the list of the team's players in the starting lineup and some basic statistics about their performance during the match (goals, own goals, cards);
substitutions	the list of team's substitutions during the match, describing the players involved and the minute of the substitution.

BIG DATA

Events JSON

Field	Description
eventId	the identifier of the event's type. Each eventId is associated with an event name (see next point);
eventName	name of the event's type. There are seven types of events: pass, foul, shot, duel, free kick, offside and touch;
subEventId	the identifier of the subevent's type. Each subEventId is associated with a subevent name (see next point);
subEventName	the name of the subevent's type. Each event type is associated with a different set of subevent types;
tags	a list of event tags, each one describes additional information about the event (e.g., accurate). Each event type is associated with a different set of tags;
eventSec	the time when the event occurs (in seconds since the beginning of the current half of the match);
id	a unique identifier of the event;
matchId	the identifier of the match the event refers to
matchPeriod	the period of the match. It can be "1H" (first half of the match), "2H" (second half of the match), "E1" (first extra time), "E2" (second extra time) or "P" (penalties time);
playerId	the identifier of the player who generated the event.
positions	the origin and destination positions associated with the event.
teamId	the identifier of the player's team.

BIG DATA

Processing

Step 1

- Read players.csv file
- Compute the following form event data for each player, as events are streamed:
  - Pass accuracy
  - Duel Effectiveness
  - Free kick effectiveness
  - Shots on target
  - Foul loss
  - Own Goals

Pass Accuracy

- Pass is signified by eventId = 8
- Value must be bound between 0 and 1
- For each pass if tags have,
  - 'id' = 1801 it is an accurate pass
  - 'id' = 1802 it not an accurate pass
  - 'id' = 302, it is a key pass
- Pass accuracy will be a moving average with double weightage for key passes:

$$Pass\ Accuracy = \frac{(number\ of\ accurate\ normal\ passes + (number\ of\ accurate\ key\ passes\ *\ 2))}{(number\ of\ normal\ passes + (number\ of\ key\ passes\ *2))}$$

Duel Effectiveness

- Duel is signified by eventId = 1
- It is calculated according to how good a player is at retaining the ball
- Value must be bound between 0 and 1
- For each duel if tags have,
  - ‘id’ = 701, duel is lost
  - ‘id’ = 702 , duel is neutral
  - ‘id’ = 703, duel is won
- Pass accuracy will be a moving average with double weightage for key passes:

$$Duel\ Effectiveness = \frac{(Number\ of\ duels\ won + (Number\ of\ neutral\ duels * 0.5))}{Total\ number\ of\ duels}$$

Free Kick Effectiveness

- Free Kick is signified by eventId = 3
- Value must be bound between 0 and 1
- For each free kick if tags have,
  - ‘id’ = 1801, free kick is effective
  - ‘id’ = 1802, free kick is not effective
- If subEventId = 35, the free kick is a penalty and in such a case if tags has Id = 101, the penalty was a goal. Some penalties can be effective but may not be a goal.

$$Free\ Kick\ Effectiveness = \frac{(Number\ of\ effective\ free\ kicks + number\ of\ penalties\ scored)}{Total\ number\ of\ free\ kicks}$$

Shots Effectiveness

- Shot is signified by eventId = 10
- Value must be bound between 0 and 1
- For each shot if tags have,
  - 'id' = 1801, shot is on target
  - 'id' = 1802, shot is not on target
  - 'id' = 101, shot was a goal

$$Shot\ Effectiveness = \frac{(Shots\ on\ target\ and\ goals + (Shots\ on\ target\ but\ not\ gaols * 0.5))}{Total\ shots}$$

Shots on Target

- Shot is signified by eventId = 10
- Value must be bound between 0 and 1
- For each shot if tags have,
  - ‘id’ = 1801, shot is on target
  - ‘id’ = 1802, shot is not on target
  - ‘id’ = 101, shot was a goal

$$Shot\ Effectiveness = \frac{(Shots\ on\ target\ and\ goals + (Shots\ on\ target\ but\ not\ gaols * 0.5))}{Total\ shots}$$

Foul Loss

- Foul is signified by eventId = 2
- Count number of fouls committed by each player
- Player contribution will be penalised for number of fouls committed

Own Goal

- If for any event, the tags has id = 102, it is an own goal
- Count number of own goals scored by each player
- Player contribution will be penalised for number of own goals scored

Player Contribution

- At the end of each match, calculate the following for each player who played at any time during the match
- Player contribution is calculated by the given formula
- Player contribution is normalised by multiplying with proportion of match played
  - For players who were never substituted in or out contribution is multiplied by 1.05
  - For all other players it is multiplied by $\frac{minutes\ played}{90}$

$$Player\ contribution = \frac{(pass\ accuracy + duel\ effectiveness + free\ kick\ effectiveness + shots\ on\ target\ )}{4}$$

Player Rating

- After computing player contribution, calculate player rating
- Initially, the rating of every player is 0.5
- Player performance is calculated by reducing the contribution by 0.5% for every foul committed by the player and 5% for every own goal
- Player rating is calculated by the given formula

$$Player\ rating = \frac{(player\ performance + existing\ player\ rating)}{2}$$

Chemistry between Pairs of Players

- Chemistry is to be calculated for each pair of players in the dataset
- Chemistry between each pair of players is initially 0.5
- After every match update the chemistry of all pairs of players who played
- Value must be bound between 0 and 1
- After every match,
  - for a pair of a players in the same team, the change in chemistry is absolute value of the average of the change in rating of both players. If rating of both players increased or decreased then chemistry increases but if rating of one player increased while that of other decreased, then chemistry decreases by same value.
  - for a pair of a players in the opposing teams, the change in chemistry is absolute value of the average of the change in rating of both players. If rating of both players increased or decreased then chemistry decreases but if rating of one player increased while that of other decreased, then chemistry increases by same value.

Examples

- If Rashford and Salah are playing on opposite sides and their ratings go up by 0.02 and 0.06 respectively, then their chemistry will go down by 0.04
- If Bale and Ozil are playing on the same team and their ratings go up by 0.07 and down by 0.03 respectively, then their chemistry will go down by 0.02
- If Pogba and Kane are playing on opposing teams and their ratings go up by 0.07 and down by 0.03 respectively, then their chemistry will go up by 0.02

Examples

- If Rashford and Salah are playing on opposite sides and their ratings go up by 0.02 and 0.06 respectively, then their chemistry will go down by 0.04
- If Bale and Ozil are playing on the same team and their ratings go up by 0.07 and down by 0.03 respectively, then their chemistry will go down by 0.02
- If Pogba and Kane are playing on opposing teams and their ratings go up by 0.07 and down by 0.03 respectively, then their chemistry will go up by 0.02

Player Profile

- Maintain profile of each player
- All data from the players dataset must be available on the profile
- Profile should also have the following data, as of at the end training data:
  - Number of Fouls
  - Number of Goals
  - Number of Own Goals
  - Pass Accuracy
  - Shots on Target

Predict Winning Chances

- For a pair of teams, comprising 11 players each predict the winning chance:
  - For each player of a team, average of all chemistry coefficients with all other members of that team
  - Multiply this consolidated chemistry coefficient with the rating of that player to give player strength
  - Find the team's overall strength by averaging the player strength of the 11 players
  - Predict winning chance from team overall strength from given formula

$$chance\ of\ A\ winning = \left(0.5 + strength\ of\ A\ - \frac{Strength\ of\ A\ + Strength\ of\ B}{2}\right) * 100\%$$

$$chance\ of\ B\ winning = 100\ - chance\ of\ A\ winning$$

Problems

- Some players may not have played enough matches in the given data to accurately assess their ratings
- Ratings of players are a function of their age and thus will change according to when match is held

Solution: Clustering

- Cluster players into 5 groups to approximate rating and chemistry coefficient for players who have played less than 5 matches in training data
- Cluster based on the profile of the players
- Take average rating and chemistry coefficient of cluster for players who have played less than 5 matches

Solution: Regression

- Use non linear (quadratic) regression to model the change in player's rating with age
- When a user provides teams for a match and match date, use regression to get the player's rating at that date and the compute winning chance

User Interface

- Users can provide a date of a match and 2 teams of 11 players each
- System must ensure that the list of players have:
  - 1 Goalkeeper
  - At least 2 defenders
  - At least 2 Mid fielders
  - And at least 1 Forward
  - Return Invalid team in case the above conditions are not met
- After regression make sure that the rating of any player is not below 0.2, if any player has a rating of less than 0.2, return that the player has retired
- If all conditions are met, return chance of either team winning the match
- GUI is not required, file can be read from command like and written to a destination

User Interface

- User can request the profile of each player
- User can request details of any match from training data by providing match label and date
- The system should read a JSON file for the request from the user and write back a JSON file as result

BIG DATA

Marks Break up

Task	Marks
Accepting Streamed data on streaming spark	2
Compute metrics for each match	2
Maintaining player profile	2
Clustering	3
Regression	3
Prediction of winning chances	1
Processing request and giving response	3 (1 Each)
Report	2
Viva and Demo	2

Other Info

- Due Date: 01/12/2020
- Final Report: the submission has to be accompanied by a final report which should outline your inferences and findings from the data and problems and difficulties faced while doing the project.

THANK YOU

K V Subramaniam
Computer Science and Engineering

subramaniamkv@pes.edu

+91 80 6666 3333 Extn 877