ETL Project – FIFA World Cup Data

Ryan Bandrowski, Kyle Moretti, Neel Pendyala, Tyler Ratajczyk

1. Project Setup
   1. Import dependencies: pandas, numpy, and sqlalchemy
   2. Load in csv files to dataframes

Our data set came from 3 separate csv files. WorldCups.csv contained high-level data for each World Cup event from 1930 through 2014. Its columns were Year, Country, Winner, Runners-Up, Third, Fourth, Goals Scored, Qualified Teams, Matches Played, and Attendance.

WorldCupMatches.csv contained data on every match that occurred in each World Cup in the data set. Its columns were Year, Datetime, Stage, Stadium, City, Home Team Name, Home Team Goals , Away Team Goals , Away Team Name , Win conditions , Attendance , Half-time Home Goals , Half-time Away Goals , Referee , Assistant 1 , Assistant 2 , RoundID , MatchID , Home Team Initials , and Away Team Initials.

WorldCupPlayers.csv contained data on ever player in every match in the data set. Its columns were RoundID, MatchID, Team Initials, Coach Name, Line-up, Shirt Number, Player Name, Position, and Event.

1. Data Cleaning
   1. World Cups Data

We used a string replace to remove decimals in the attendance columns which were being used as columns and change the data type to integer. Changed the name of the Runners-Up column to ‘Second’ for continuity with the others.

* 1. World Cup Matches

NA’s were dropped because there were many extra rows with no data as well as one match that did not have attendance data. We dropped duplicates by Match ID because many of the matches in 2014 were duplicated. We applied pd.numeric to the dataframe, ignoring errors to skip strings and downcasting to integers to get the numerical data into the desired format. The datetime format for the time of each match was not in a useable format. We split them into separate columns for date and time and ran some for loops to on the columns to convert them to data and time formats respectively. Team initials were used in the players data, so we move the team initials to the location where the country names were and deleted the country name columns, renaming the initial columns to what the country columns were named. MatchID would be the key so we renamed the original column, placed a new one in the first position copying the data from the original and then deleted the original.

* 1. World Cup Players

First, we deleted some data that was incomplete or that we would not need, this included Coach Name, Line-up, Shirt Number, and Position. We deleted rows with the Match ID 300186460 as this was the one with no attendance data from the World Cup Matches dataframe. Many players did not have some sort of event in many games so that column was N/A, we filled N/A’s with an empty string. To make the event data useful, we parsed the events out into new columns called Goals, Own\_Goals, Yellow\_Card, Red\_Card, Second\_Yellow\_Card, Penalty, and Missed\_Penalty. We then ran a loop over the rows of the player data. It would extract the data from the Event column, splitting at spaces to account for multiple events in a match. It would then loop through the events for that player. It checks to make sure that the event list is not empty, and then runs a series of if statements to check the first one or two positions of the string, if it matched “G” or “MP” for example, then the corresponding column for that event would have 1 added to it. We then deleted the original events column and were left with a more traditional and useful display of player stats. We also deleted Round ID as this was rather useless. We inserted a column at the first position called Player\_MatchID, which combined Player Name and Match ID, separated by an underscore, to be used as the unique identifier for this table since both players and match ID’s would have many instances. We dropped duplicates by Player\_MatchID to test it and found no change in the number of rows.

* 1. Unique Players DataFrame

We grouped the World Cup Players dataframe by player name and removed the match ID, summing the stats columns. This gave us a unique list of players who’ve competed in the World Cup as well as the total of their stats for their appearances.

1. Loading Data to SQL
   1. In PostgreSQL, we created 4 tables with the names world\_cups, wc\_matches, unique\_players, and wc\_players with all of the dataframe’s columns outlined in the tables with the proper data types.
      1. For world\_cups, the primary key was the year of the World Cup.
      2. The table, ­­wc\_matches, had Match ID as the primary key, and the year was a foreign key linking back to the world\_cups table.
      3. Next we created unique\_players since it’s primary key of Player\_Name would point to wc\_players and had to be created first.
      4. Finally, wc\_players would use our created Player\_MatchID column as it’s primary key. It would then have 2 foreign keys: MatchID and Player\_Name, pointing to wc\_matches and unique\_players, respectively.
2. SQL Queries
   1. Our first query created a master table by joining some relevant match data with the wc\_players table like year, date, stage, attendance, and referee.
   2. Our second and third queries were grouped by team, counted the number of matches, and the average attendance they received. We ordered one by number of matches and one by average attendance to help give a more complete picture since some of the teams with the highest averages played in a small number of matches compared to the leaders.
   3. Our fourth and fifth queries followed the same concept as the last two, but they were grouped by player name. Number of matches and average attendance were aggregated again, and they were ordered by match count and average attendance again to help get a better idea.
   4. Our sixth and final query was inspired by the idea that referees call games differently. We pulled in referee name and the count of their matches from wc\_matches and joined in the yellow cards, red cards, and penalties by match ID. We also calculated the rate at which these calls were made on a per game basis to get a better picture as a few of the outliers for yellow cards per match were from referees who officiated far fewer matches than many others.