

CSc 496, Homework #7: My team is worse than your team.

Due date: Thursday, November 21st, 2024. **No late assignments will be accepted.**

In this assignment you will write a program to determine the percentage of games in which each team loses a game in which they are likely to win.

You are to create play-by-play data (one CSV file per game) for seasons 2001–2023. This data should include at least the win probability, team on offense, team on defense, score differential after the play is over, and winner. These correspond to fields `wp`, `posteam_score`, `defteam_score`, and `score_differential_post`. The “winner” field is to be one if the team on offense won the game and zero otherwise. Unfortunately, the winner field is not a column in the data. You will need to figure out who won the game by looking at the last play of the game (think about how to do this). You will need to buffer all of the fields so that when you output them, you can put the winner field in for *each* line. The CSV files should be placed into a subdirectory (from where the program is invoked) called `NFLCSV`; you need to create this directory if it does not yet exist. Note that for debugging purposes you may want to put more fields in the CSV files; that is fine but not required.

Next, you will write your “blown game” program, which will take three command line parameters. The first is a directory where all of your generated CSV files reside. The second and third are thresholds, `lowerWPThreshold` and `upperWPThreshold`, which are probabilities between 0.0 and 1.0 (`lowerWPThreshold < upperWPThreshold`).

The program iterates through all the CSV files. For each game, if either team ever (at any point during the game) had a win probability between `lowerWPThreshold` and `upperWPThreshold`, you will add one to `gamesBetweenThresholds`. If the team won the game, then you also add one to `gamesWon`. After you have processed all the CSV files, compute the ratio R of `gamesWon` to `gamesBetweenThresholds` for each team. You must output a header and then 32 lines; each line has the team name, `gamesBetweenThresholds`, `gamesWon`, and R . Sort the 32 lines in descending order of R .

Note that if the lower and upper thresholds are 0.0 and 1.0, respectively, the results just degenerate into every team’s win-loss record. On the other hand, if the thresholds are 0.9 and 1.0, then the results show which team won the most games for which it had (at some point during the game) a probability of winning of 0.9. For seasons 2014 to 2023, with thresholds 0.9 and 1.0, here is the output I get:

```
Team, Games Between Thresholds, Games Won, Winning Percentage
GB, 101, 100, 99.01%
NE, 113, 110, 97.35%
HOU, 74, 72, 97.30%
MIA, 73, 71, 97.26%
DAL, 93, 90, 96.77%
WAS, 58, 56, 96.55%
BUF, 94, 90, 95.74%
BAL, 93, 89, 95.70%
MIN, 88, 84, 95.45%
LV, 64, 61, 95.31%
NO, 100, 95, 95.00%
PHI, 96, 91, 94.79%
CAR, 73, 69, 94.52%
```

```
KC, 127, 120, 94.49%
ARI, 70, 66, 94.29%
PIT, 103, 97, 94.17%
SEA, 101, 95, 94.06%
LA, 95, 89, 93.68%
TEN, 74, 69, 93.24%
CIN, 83, 77, 92.77%
TB, 79, 72, 91.14%
JAX, 53, 48, 90.57%
DEN, 81, 73, 90.12%
IND, 86, 77, 89.53%
NYJ, 57, 51, 89.47%
DET, 73, 65, 89.04%
LAC, 70, 62, 88.57%
SF, 92, 81, 88.04%
ATL, 77, 67, 87.01%
CLE, 61, 53, 86.89%
CHI, 66, 57, 86.36%
NYG, 64, 53, 82.81%
```

Submit your python files on lectura using the `turnin` command; for this program, use the assignment name `csc496-f24-hw7`. Your programs should be named `createCSV.py` and `calculateBlownGames.py`.