Predicting the points scored by a NBA team

Pratik Bhusal, Hrishikesh Inamdar,
Bradley Olbrey, Ron Antonio,
Nishant Gurrapadi

Initial Plans

- Initially wanted to use “stamina” of each player
  - Insight as to how much playtime a player should get, how many subs would be required.
  - Turned out to be too subjective; couldn’t find a fitting description of stamina.
- Chose “team play” instead.
  - Easier to quantify. (No. of passes, steals, etc.)
  - Correlate this with points scored in order to predict each team’s final score by the end of the game.

Challenges

- BIG, BIG Data! **180GB** of it.
- However, not enough data to calculate the total score of each team.
    - Missing points made from foul shots.
- Unmatched data: not every game had both frames files and simple marking files.

Implementation

- Used dictionaries for key-value pairs between events and counts of that event for each player
- {Player id: {Team : home/away, 2PM : count, 2PX : count, PASS : count, etc...}}
- Each row represents a player and their statistics.
  - How many 2 pointers he made
  - Number of turnovers
  - Position on court, etc.

2017-01-17-DAL-CHI.csv

	2PM	2PX	3PM	3PX	PASS	POSS	TO	pos_name	team
2548.0	8	12	0	1	29	52	4	SG	CHI
1627835.0	1	2	1	1	23	25	0	PF	CHI
203487.0	2	2	2	0	41	45	2	PG	CHI
201577.0	10	5	0	0	24	43	3	C	CHI
202710.0	5	6	0	1	59	75	3	SF	CHI
200765.0	0	4	0	0	41	47	1	PG	CHI
203926.0	2	4	1	3	19	26	0	SG	CHI
1626171.0	0	1	0	0	11	12	0	PF	CHI
1626245.0	3	0	0	0	9	13	0	C	CHI
202703.0	0	1	2	1	32	35	0	PF	CHI
1626170.0	0	0	0	1	1	2	0	PG	CHI
1717.0	3	3	0	3	33	35	0	PF	DAL
203084.0	6	8	2	1	37	53	2	SF	DAL
101114.0	4	2	1	4	58	66	4	PG	DAL
202083.0	1	4	3	2	27	33	1	SG	DAL
203552.0	4	1	3	2	25	37	1	PG	DAL
1626257.0	1	1	0	0	17	16	1	C	DAL
2734.0	1	4	0	2	16	27	2	PG	DAL
200826.0	3	1	2	2	40	51	1	PG	DAL
203939.0	3	0	0	1	11	14	0	C	DAL
1627827.0	0	0	0	0	6	5	0	SF	DAL

Implementation

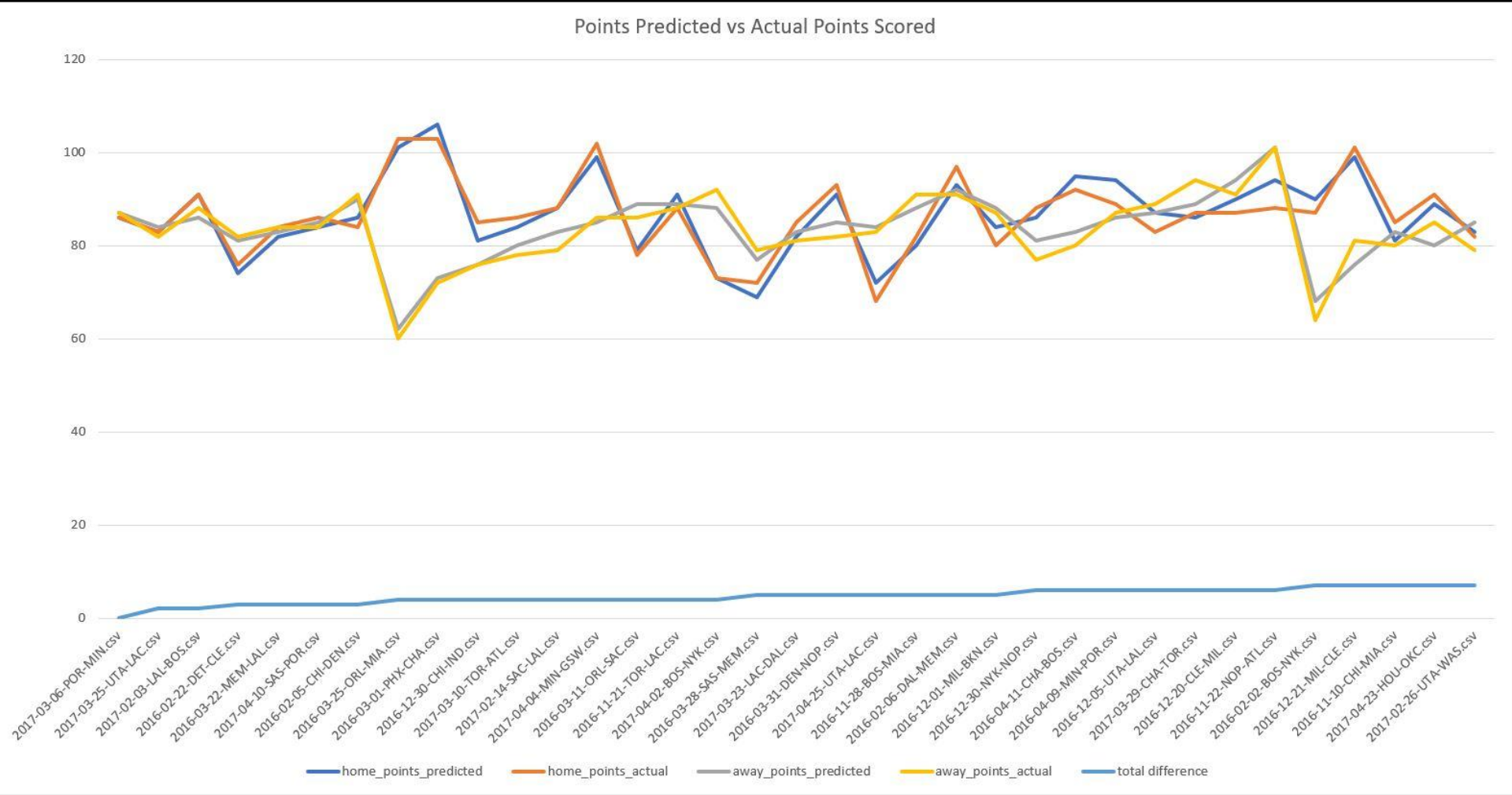
- Needed to find a way to run the program on many many datasets quickly.
  - Whittled down the frames file to just what we needed:
    - Dictionary of all the player ids from the game
  - Significantly reduced the size of the files
    - From ***~30MB*** per file to ***300 bytes***
    - Whole dataset trimmed from ***~180 GIGAbytes*** to a **1.8 MEGAbytes**: 100,000 times smaller!
  - Allows us to process the files much quicker locally, and use significantly less cloud storage and network bandwidth when cloud computing.

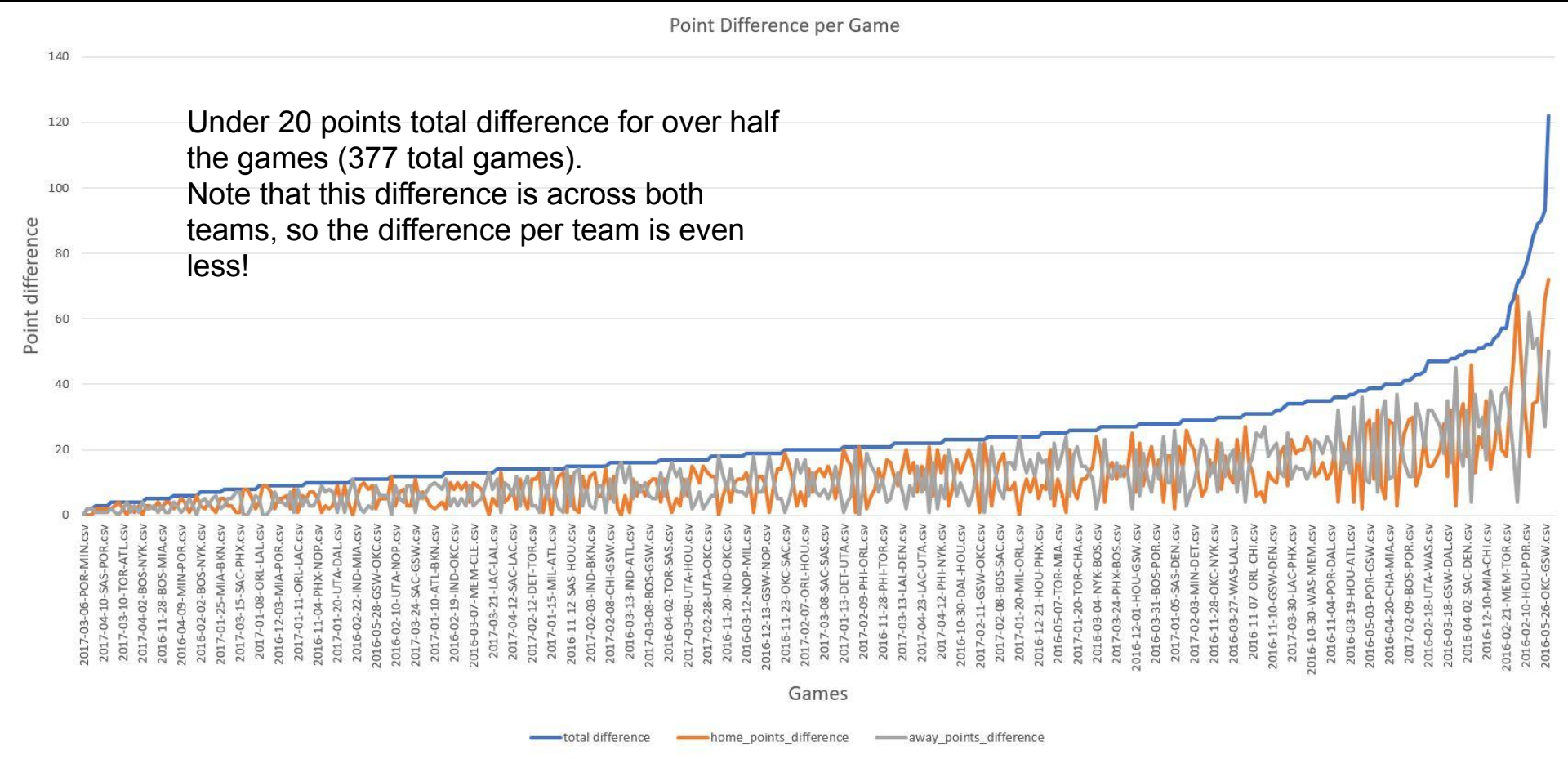
	A	B	C	D	E	F	G	H	I	J
1	game_name	away	away_C_2PM	away_C_2PX	away_C_3PM	away_C_3PX	away_C_PASS	away_C_POSS	away_C_TO	away_PF_2PM
2	2017-03-08-NYK-MIL.csv	NYK	4	3.5	0	0	35.5	45	3.5	1
3	2017-01-17-DAL-CHI.csv	DAL	2	0.5	0	0.5	14	15	0.5	
4	2017-03-10-IND-MIL.csv	IND	2	3.25	0	0.25	18.75	22.5	0.5	
5	2016-11-27-HOU-POR.cs	HOU	6.5	3	0	0	28	31	0.5	
6	2016-03-23-DAL-POR.csv	DAL	4	4.5	0	0	20	25.5	2.5	
7	2017-03-13-CHI-CHA.csv	CHI	1.33333333333333	2	0	0	11.3333333333333	13	0.666666666666667	
8	2017-01-02-WAS-HOU.cs	WAS	6	4	0	0	49	53	4	
9	2017-02-04-CLE-NYK.csv	CLE	6	2	0	0	25	28	1	
10	2016-12-22-ORL-NYK.cs	ORL	2.33333333333333	3.66666666666667	0	0.333333333333333	24.6666666666667	30	0.333333333333333	3
11	2017-03-12-POR-PHX.cs	POR	6	3	0	0	37	38	2	
12	2016-02-21-LAL-CHI.csv	LAL	4.66666666666667	4	0	0.333333333333333	26.3333333333333	31.6666666666667	0.333333333333333	
13	2016-04-01-PHI-CHA.csv	PHI	6.5	4	0	0.5	38.5	43	1	
14	2016-03-08-NYK-DEN.cs	NYK	2.33333333333333	2	0	0.333333333333333	21	23.6666666666667	1	
15	2017-03-11-PHI-LAC.csv	PHI	7	3.5	1	0.5	32.5	48.5	2.5	
16	2016-02-09-WAS-NYK.cs	WAS	2.33333333333333	2.66666666666667	0	0.333333333333333	17.3333333333333	20	1	
17	2016-11-14-MIA-SAS.csv	MIA	9	2	0	0	20	20	4	
18	2016-12-22-SAS-LAC.csv	SAS	3	2.33333333333333	0.333333333333333	0.333333333333333	30.3333333333333	33.3333333333333	0.666666666666667	
19	2016-12-23-DAL-LAC.csv	DAL	1.5	0.5	0	0	18.5	15.5	1	
20	2017-02-28-DEN-CHI.csv	DEN	2	2	1	0	32	32	1.33333333333333	
21	2016-10-26-SAC-PHX.csv	SAC	3.33333333333333	3.66666666666667	0	0.333333333333333	24	29.6666666666667	1.66666666666667	
22	2017-03-15-LAL-HOU.csv	LAL	3.6	1.2	0	0.2	17.8	23.8	0.8	
23	2016-10-31-DEN-TOR.cs	DEN	4	4.33333333333333	0	0.333333333333333	30.3333333333333	36.6666666666667	1.66666666666667	
24	2016-11-02-OKC-LAC.cs	OKC	1	1.8	0.4	0.4	21.4	20.6	1	
25	2016-04-16-IND-TOR.csv	IND	1.75	2.5	0	0.25	14.75	17	0.75	
26	2017-03-25-WAS-CLE.cs	WAS	1.66666666666667	1	0	0	11	12.6666666666667	1	
27	2016-10-31-PHX-LAC.csv	PHX	2	3.33333333333333	0	0	15.6666666666667	19.3333333333333	1.66666666666667	
28	2016-03-30-NOP-SAS.cs	NOP	3.33333333333333	4.33333333333333	0	0	34.6666666666667	37.3333333333333	2.33333333333333	
29	2017-03-27-ORL-TOR.cs	ORL	2	2	0	0.333333333333333	28	31.3333333333333	1.66666666666667	
30	2016-03-13-NYK-LAL.csv	NYK	3.5	1.5	0	0	33.5	37	2	
31	2017-03-21-LAC-LAL.csv	LAC	6	3	0	0	25	27	1	
32	2016-02-18-SAS-LAC.csv	SAS	1.5	2.5	0	0	21	21	1.5	
33	2016-02-21-SAS-PHX.csv	SAS	1	4	0	0	25	29	0	
34	2017-04-10-IND-PHI.csv	IND	5	1.66666666666667	0	0	23.6666666666667	25.3333333333333	1.66666666666667	
35	2017-04-09-DET-MEM.cs	DET	3	3	0	0	16	14	1.5	
36	2017-02-23-POR-ORL.cs	POR	3.5	4	0	0	31	33	2	
37	2016-12-14-SAC-HOU.cs	SAC	5	4.5	0	0	18.5	27.5	0.5	2.66666666666

Machine Learning with XGBoost

- Trained our model to learn and predict the points scored by a team.
- Split data into about 80% training and 20% test.
- Train an xgboost model on training set.
- Ignored the “points scored” attribute because we are trying to predict the number itself.
- Find error at the end with test set.

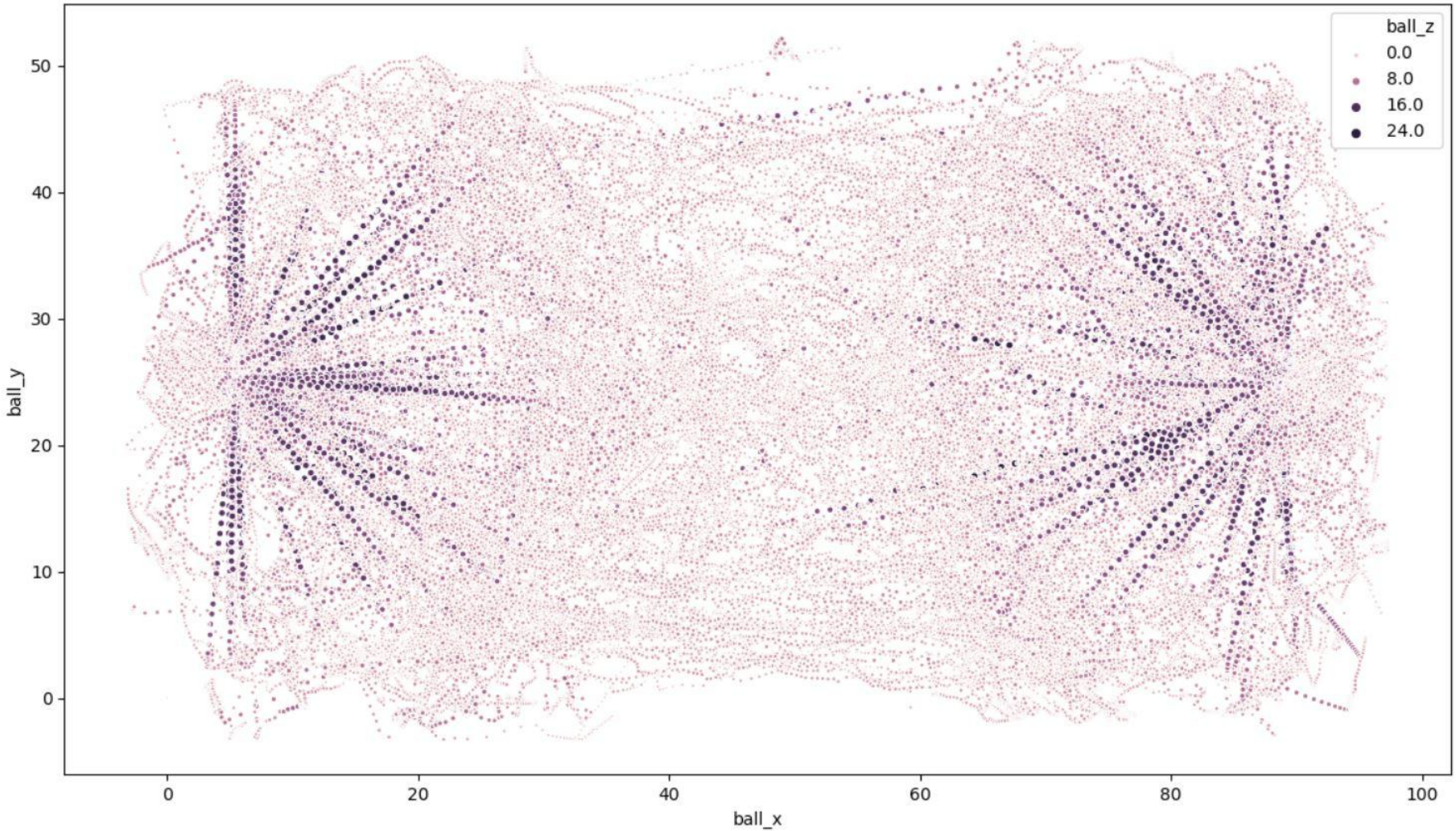
	A	B	C	D	E	F	G
1	game_name	home_points_predicted	home_points_actual	home_loss	away_points_predicted	away_points_actual	away_loss
2	2016-02-02-BOS-NYK.csv	90	87	3	68	64	4
3	2016-02-04-HOU-PHX.csv	84	75	9	88	89	1
4	2016-02-04-LAL-NOP.csv	86	82	4	75	85	10
5	2016-02-04-NYK-DET.csv	78	106	28	73	61	12
6	2016-02-05-CHI-DEN.csv	86	84	2	90	91	1
7	2016-02-06-BKN-PHI.csv	89	86	3	89	75	14
8	2016-02-06-DAL-MEM.csv	93	97	4	92	91	1
9	2016-02-07-ATL-ORL.csv	96	85	11	80	83	3
10	2016-02-10-DEN-DET.csv	93	105	12	65	65	0
11	2016-02-10-HOU-POR.csv	92	122	30	99	53	46
12	2016-02-10-UTA-NOP.csv	85	88	3	80	71	9
13	2016-02-18-UTA-WAS.csv	91	106	15	83	51	32
14	2016-02-19-CHA-MIL.csv	88	79	9	88	83	5
15	2016-02-19-DAL-ORL.csv	76	95	19	88	93	5
16	2016-02-19-HOU-PHX.csv	91	71	20	78	81	3
17	2016-02-19-IND-OKC.csv	81	89	8	82	87	5
18	2016-02-20-WAS-MIA.csv	81	96	15	78	83	5
19	2016-02-21-CHA-BKN.csv	83	93	10	94	72	22
20	2016-02-21-LAL-CHI.csv	97	109	12	75	85	10
21	2016-02-21-MEM-TOR.csv	91	73	18	98	59	39
22	2016-02-21-NOP-DET.csv	91	88	3	73	84	11
23	2016-02-22-DET-CLE.csv	74	76	2	81	82	1
24	2016-02-22-IND-MIA.csv	73	78	5	83	77	6
25	2016-02-22-LAL-MIL.csv	71	93	22	84	68	16
26	2016-02-24-NYK-IND.csv	86	106	20	78	72	6
27	2016-02-24-OKC-DAL.csv	85	79	6	89	98	9
28	2016-02-24-SAS-SAC.csv	81	84	3	87	96	9
29	2016-02-26-MEM-LAL.csv	89	69	20	85	99	14
30	2016-02-27-GSW-OKC.csv	97	101	4	77	104	27
31	2016-02-27-MEM-PHX.csv	82	84	2	83	70	13
32	2016-02-27-POR-CHI.csv	78	85	7	78	81	3
33	2016-02-27-SAS-HOU.csv	69	78	9	84	82	2
34	2016-02-29-OKC-SAC.csv	103	109	6	79	102	23
35	2016-03-01-PHX-CHA.csv	106	103	3	73	72	1
36	2016-03-02-DET-SAS.csv	95	89	6	93	68	25
37	2016-03-02-LAL-DEN.csv	99	88	11	78	81	3


Points Predicted vs Actual Points Scored
120
100
80
60
40
20
0
2017-03-06-POR-MIN.csv
2017-03-25-UTA-IAC.csv
2017-02-03-LAL-BOS.csv
2016-02-22-DET-CLE.csv
2016-03-22-MEM-LAL.csv
2017-04-10-SAS-POR.csv
2016-02-05-CHI-DEN.csv
2016-03-25-ORL-MIA.csv
2016-03-01-PHX-CHA.csv
2016-12-30-CHI-IND.csv
2017-03-10-TOR-ATL.csv
2017-02-14-SAC-LAL.csv
2017-04-04-MIN-GSW.csv
2016-03-11-ORL-SAC.csv
2016-11-21-TOR-IAC.csv
2017-04-02-BOS-NYK.csv
2016-03-28-SAS-MEM.csv
2017-03-23-LAC-DAL.csv
2016-03-31-DEN-NOP.csv
2017-04-25-UTA-IAC.csv
2016-11-28-BOS-MIA.csv
2016-02-06-DAL-MEM.csv
2016-12-01-MIL-BKN.csv
2016-12-30-NYK-NOP.csv
2016-04-11-CHA-BOS.csv
2016-04-09-MIN-POR.csv
2016-12-05-UTA-LAL.csv
2017-03-29-CHA-TOR.csv
2016-12-20-CLE-MIL.csv
2016-11-22-NOP-ATL.csv
2016-02-02-BOS-NYK.csv
2016-12-21-MIL-CLE.csv
2016-11-10-CHI-MIA.csv
2017-04-23-HOU-OKC.csv
2017-02-26-UTA-WAS.csv
home_points_predicted
home_points_actual
away_points_predicted
away_points_actual
total difference


Point Difference per Game
Under 20 points total difference for over half the games (377 total games).
Note that this difference is across both teams, so the difference per team is even less!
Point difference
Games
total difference
home_points_difference
away_points_difference

Cool visuals

- Height map of ball around the field:
  - Shows where shots are taken from by each team.


ball_z
0.0
8.0
16.0
24.0
ball_y
ball_x
50
40
30
20
10
0
0
20
40
60
80
100

Cool visuals

- Height map of ball around the field:
  - Shows where shots are taken from by each team.

Cool visuals

- Height map of ball around the field:
  - Shows where shots are taken from by each team.
- XY coordinate trail-animation of the ball and the closest player to the ball on each the home and away team.
  - Allows you to see the movement of the ball and the players around the field.
  - How much pressure the opposing team is applying to the player with the ball.
  - With some tweaking, could be turned into a beautiful GUI program.
    - Select players to view by ID
    - User-adjustable slider to pick the frame to view
    - Showing players transition into the area of play around the ball

Impact

- We can objectify team performance!
- Predict what areas a team needs to improve on before a single game even starts.
- A curated dataset that is a **100,000 times** smaller than the original.

Future Work

- Right now, our program requires the entire dataset be loaded.
  - However, information it's based on (events for each player) is generated throughout the duration of a game.
  - Could be turned into an **online learning algorithm**, where it predicts the ending score based on the events as the game progresses.