Project 1: MLB Team Data

The dataset I will be using comes from this website: https://www.openintro.org/data/index.php?data=mlb_teams

As stated in the website itself:

"A subset of data on Major League Baseball teams from Lahman's Baseball Database. The full data set is available in the Lahman R package."

"A data frame with 2784 rows and 41 variables."

This data is updated to include games from 1876 to 2020. Not to mention several teams in this dataset do not exist or are under a different name.

There are 41 variables. The variables I'm planning to analyze are Year, League ID (American and National League), Team Name, Wins, League Winners, World Series Winner, Runs Scored, and Homeruns.

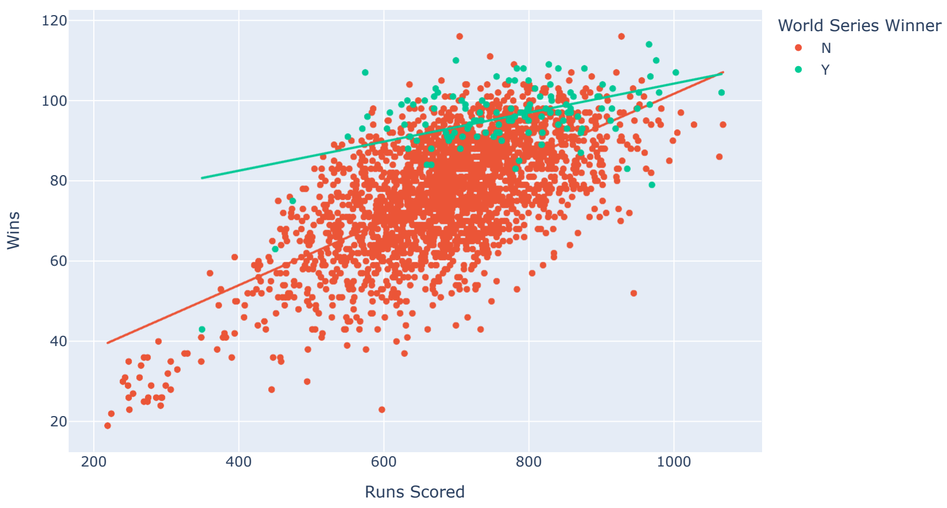
```
import pandas as pd
import plotly.express as px
from google.colab import drive
import plotly.io as pio
pio.renderers.default = "colab"

drive.mount('/content/gdrive')
df_baseball = pd.read_csv("/content/mlb_teams.csv")
```

```
Drive already mounted at /content/gdrive; to attempt to forcibly remount, call drive.mount("/content/gdrive", force_remount=True).
```

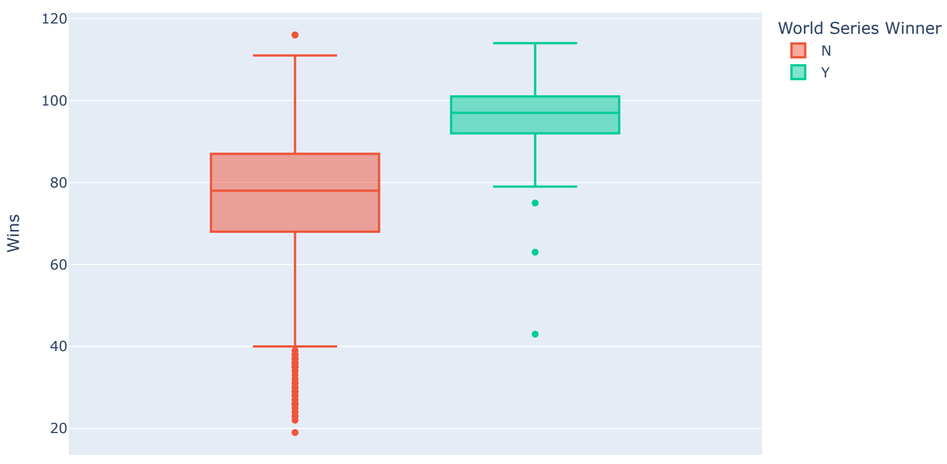
The first thing I planned to do was to find if the number of runs is correlated with wins. Typically a high number of runs is associated with a higher number of wins. This is common sense, but I'd like to see if theres a big difference between World Series winners and non World Series Winners.

```
fig = px.scatter(df_baseball,
                 x = "runs_scored",
                 y = "wins",
                 color = "world_series_winner",
                 trendline = "ols",
                 labels=dict(runs_scored="Runs Scored",
                             wins="Wins",
                             world_series_winner = "World Series Winner"))
fig.show()
```



There is a positive correlation with runs scored and wins. How about looking at the distributions with a box plot?

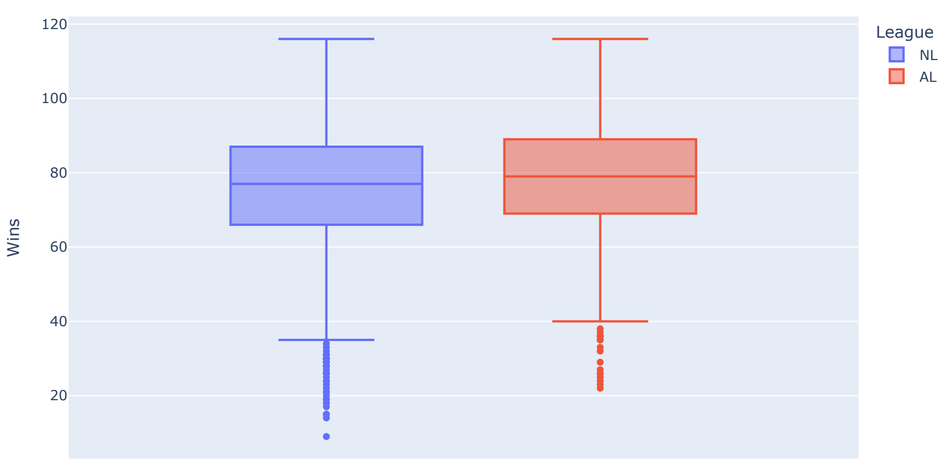
```
fig = px.box(df_baseball,
                 y = "wins",
                 color = "world_series_winner",
                 labels=dict(wins="Wins",
                             world_series_winner = "World Series Winner"))
fig.show()
```



By seperating this data based on teams that won the World Series we find that the teams that didn't win the World Series is skewed to the left (meaning closer to 0 wins) than World Series winners who are are also slightly skewed to the left, but have a noticable higher average final win score. (Expected result but it's nice to visualize)

Next I want to look at the two different leagues. The American League (AL) and the National League (NL). This box plot shouldn't be too exciting since the only real difference between the leagues are the teams in it and designated hitter (DH) rules. They're expected to be the same.

```
fig = px.box(df_baseball,
                 y = "wins",
                 color = "league_id",
                 labels=dict(wins = "Wins",
                             league_id = "League"))
fig.show()
```



Intial Data Questions:

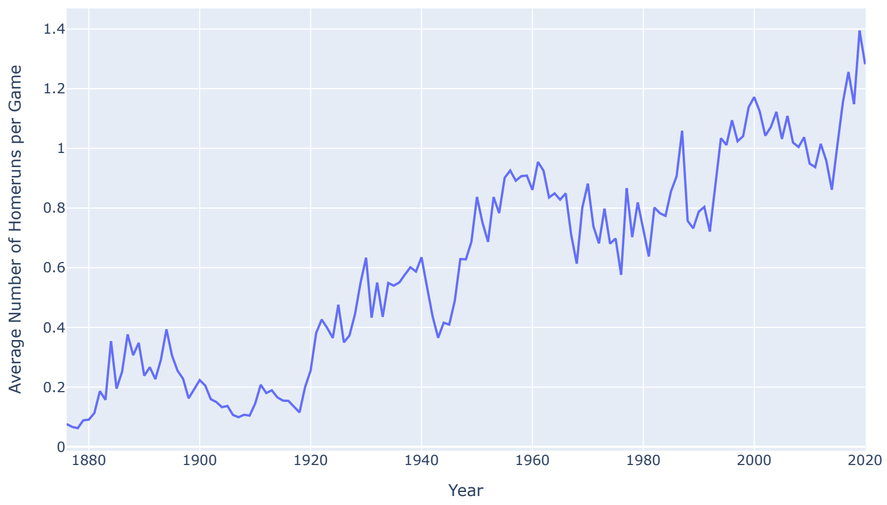
Three big questions I have with the current baseball data; 1. Are the average number of homeruns increasing each year? 2. Are league batting average increasing or decreasing per year? 3. Was the Dodgers 2020 World Series win a fluke? It's a common diss to say that the Dodger 2020 win was a fluke out of pure luck due to the reduced number of games. But was it really?

1. Are the average number of homeruns increasing each year?

```
df_baseball_copy = df_baseball.copy()
df_baseball_copy["avg_homerun"] = df_baseball["homeruns"] / df_baseball["games_played"]

df_graph1 = df_baseball_copy.groupby("year")["avg_homerun"].mean()
df_graph1 = df_graph1.reset_index()
df_graph1["team_name"] = df_baseball_copy["team_name"]

fig = px.line(df_graph1,
              x = "year",
              y = "avg_homerun",
              labels=dict(year="Year", avg_homerun="Average Number of Homeruns per Game"))
fig.show()
```



This graph shows that the average number of homeruns since 1876 has been slowly increasing from 0 homeruns to averaging around 1 homerun per game starting around 2000. (This is nearing the peak of the steroid era and a lot of rule changes occurred at this point)

This trend suggests the number of homeruns are rising every year (including 2020).

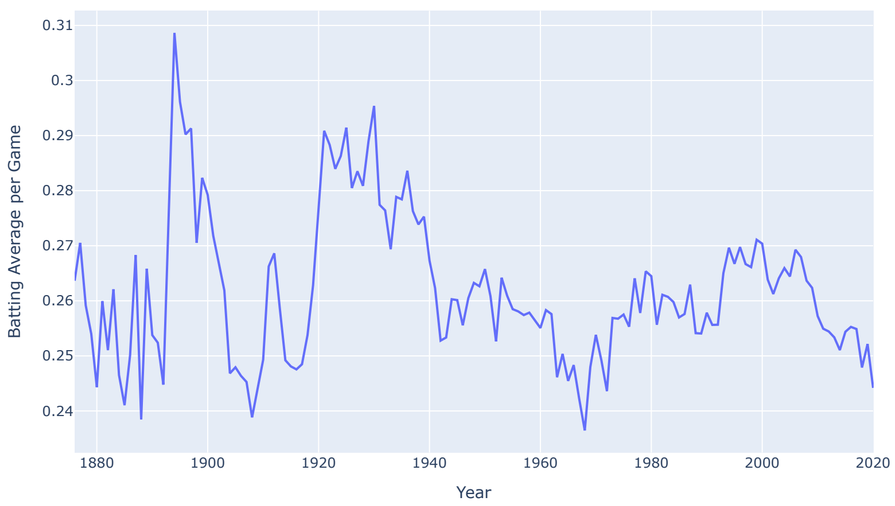
The next question is:

2. Are league batting averages increasing or decreasing per year?

```
df_baseball_copy["bat_avg"] = df_baseball["hits"] / df_baseball["at_bats"]
df_bat_year = df_baseball_copy.groupby("year")["bat_avg"].mean().reset_index()

fig = px.line(df_bat_year,
              x = "year",
              y = "bat_avg",
              labels=dict(year="Year", bat_avg="Batting Average per Game"))

fig.show()
```



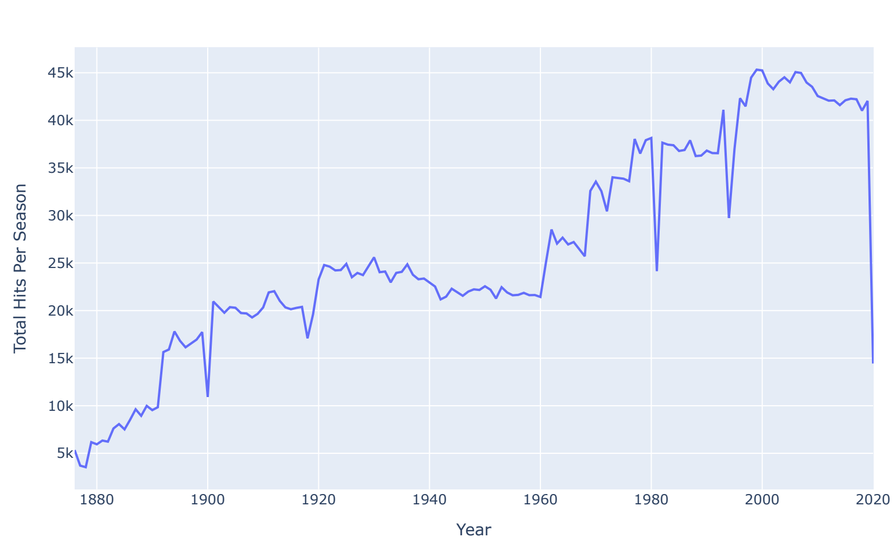
Batting average appears to be decreasing every year. This makes sense as more teams now focus more on defense and pitching has become sophisiticated to confuse pitchers even more.

What about the number of hits? Are they increasing? Not the main question, but it relates. Why are the batting averages going up when the number of hits should be increasing?

```
df_hit_year = df_baseball_copy.groupby("year")["hits"].sum().reset_index()

fig = px.line(df_hit_year,
              x = "year",
              y = "hits",
              labels=dict(year="Year", hits="Total Hits Per Season"))

fig.show()
```



While the batting average in baseball appears to be decreasing, the number of overall hits is increasing. This is likely because teams have focused on defense much more now that batting has become increasingly more difficult. But now there are more teams and at bat opportunities, causing this number to go up.

3. Was the Dodgers 2020 World Series win a fluke? It's a common diss to say that the Dodger 2020 win was a fluke out of pure luck due to the reduced number of games. But was it really out of luck for the Dodgers?

For this analysis, I will be using Euclidean Distances. I want to create new variables to show the batting average, average runs per game, and earned run average (ERA). Additionally I would check what league the team was in and if they won that league for the categorical variables.

While I initially included number of homeruns in the game, I thought that average homeruns would vary too much over the year. Since that's a league wide trend, I will use that variable in the second Euclidean distance.

```
df_world_series = df_baseball[df_baseball["world_series_winner"] == "Y"]
df_world_series = df_world_series.copy()
```

Pandas won't stop yelling at me unless I make this a copy of itself to maniplate columns.

Create all game average variables.

```
df_world_series["bat_avg"] = df_world_series["hits"] / df_world_series["at_bats"]
df_world_series["win_pct"] = df_world_series["wins"] / df_world_series["games_played"]
df_world_series.set_index("year", inplace = True)
df_world_series
```

year	league_id	division_id	rank	games_played	home_games	wins	losses	division_winner	wild_card_winner	league_winner	...	walks_allowed	strikeouts_by_pitchers	errors	double_plays	fielding_percentage	team_name	ball_park	home_attendance	bat_avg	win_pct
1884	NL	NaN	1	114	NaN	84	28	NaN	NaN	Y	...	172	639	398	50	0.918	Providence Grays	Messer Street Grounds	NaN	0.241143	0.736842
1887	NL	NaN	1	127	NaN	79	45	NaN	NaN	Y	...	344	337	394	92	0.925	Detroit Wolverines	Recreation Park	NaN	0.299424	0.622047
1888	NL	NaN	1	137	NaN	84	47	NaN	NaN	Y	...	307	726	432	76	0.924	New York Giants	Polo Grounds I	NaN	0.242048	0.613139
1889	NL	NaN	1	131	NaN	83	43	NaN	NaN	Y	...	524	558	437	90	0.919	New York Giants	Polo Grounds II	NaN	0.282381	0.633588
1903	AL	NaN	1	141	70.0	91	47	NaN	NaN	Y	...	269	579	239	86	0.959	Boston Americans	Huntington Avenue Grounds	379338.0	0.271600	0.645390
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2016	NL	C	1	162	81.0	103	58	Y	N	Y	...	495	1441	101	116	0.983	Chicago Cubs	Wrigley Field	3232420.0	0.256042	0.635802
2017	AL	W	1	162	81.0	101	61	Y	N	Y	...	522	1593	99	153	0.983	Houston Astros	Minute Maid Park	2403671.0	0.281768	0.623457
2018	AL	E	1	162	81.0	108	54	Y	N	Y	...	512	1558	77	106	0.987	Boston Red Sox	Fenway Park II	2895575.0	0.268362	0.666667
2019	NL	E	2	162	81.0	93	69	N	Y	Y	...	517	1511	87	111	0.985	Washington Nationals	Nationals Park	2259781.0	0.264877	0.574074
2020	NL	W	1	60	30.0	43	17	Y	N	Y	...	145	517	40	46	0.982	Los Angeles Dodgers	Dodger Stadium	0.0	0.256121	0.716667

120 rows × 42 columns

All variables are now in the copy, let's use sklearn for the next step. Average homeruns, hits, and runs won't be included because all these numbers increase every year.

```
from sklearn import set_config
from sklearn.preprocessing import StandardScaler
from sklearn.compose import ColumnTransformer
from sklearn.preprocessing import OneHotEncoder
import numpy as np

set_config(transform_output="pandas")

transformer = ColumnTransformer(
    [("Scaled Quantitative", StandardScaler(), ["win_pct", "bat_avg", "earned_run_average"]),
     ("Encoded Categorical", OneHotEncoder(sparse_output = False), ["league_id", "league_winner"])])

transformer.fit(df_world_series)
df_ws_enc = transformer.transform(df_world_series)

df_ws_enc
```

year	Scaled Quantitative__win_pct	Scaled Quantitative__bat_avg	Scaled Quantitative__earned_run_average	Encoded Categorical__league_id_AL	Encoded Categorical__league_id_NL	Encoded Categorical__league_winner_Y
1884	2.754420	-1.810762	-2.783203	0.0	1.0	1.0
1887	0.069843	1.858734	0.967629	0.0	1.0	1.0
1888	-0.138492	-1.753831	-2.222181	0.0	1.0	1.0
1889	0.339728	0.785633	0.198228	0.0	1.0	1.0
1903	0.615735	0.106852	-1.244400	1.0	0.0	1.0
...	...	...	...	...	...	...
2016	0.391521	-0.872701	-0.314707	0.0	1.0	1.0
2017	0.102806	0.747056	1.240126	1.0	0.0	1.0
2018	1.113307	-0.097009	0.647045	1.0	0.0	1.0
2019	-1.052052	-0.316462	1.480564	0.0	1.0	1.0
2020	2.282600	-0.867709	-0.523086	0.0	1.0	1.0

120 rows × 6 columns

Next steps: Generate code with df_ws_enc | View recommended plots | New interactive sheet

```
df_ws_enc = transformer.transform(df_world_series)

x = df_ws_enc.iloc[df_ws_enc.index.get_loc(2020)]

df_ws_enc["distance"] = np.sqrt(((x - df_ws_enc) ** 2).sum(axis = 1))
df_ws_enc["man_distance"] = ((x - df_ws_enc).abs()).sum(axis = 1)

df_ws_top = df_ws_enc.drop(index = 2020)

# Exclude 2020 from the top five.
df_ws_top.sort_values("distance").head(5)
```

year	Scaled Quantitative__win_pct	Scaled Quantitative__bat_avg	Scaled Quantitative__earned_run_average	Encoded Categorical__league_id_AL	Encoded Categorical__league_id_NL	Encoded Categorical__league_winner_Y	distance	man_distance
1986	1.113307	-0.431832	-0.378823	0.0	1.0	1.0	1.256204	1.749433
1942	1.413126	-0.106195	-1.276458	0.0	1.0	1.0	1.379659	2.384361
1919	1.558752	-0.431194	-1.789393	0.0	1.0	1.0	1.522509	2.426670
1909	2.226920	-0.642411	-2.045860	0.0	1.0	1.0	1.540357	1.803753
1975	1.113307	0.097837	0.037936	0.0	1.0	1.0	1.616871	2.695861

A distance of 1.25 from the 1986 World Series winner suggest an irregularity. In 1986 the New York Mets won the World Series. Even then the Eucledian distance being as higher than 1 suggests these games are not that similar.

The most other similar games were in 1942 (Saint Louis Cardinals), 1919 (Cincinnati Reds), 1909 (Pittsburgh Pirates), and 1975 (Cincinnati Reds)

Don't really see too why the 2020 World Series are similar to these specific games.

Let's look at a World Series game that would be similar.

```
df_ws_enc = transformer.transform(df_world_series)

x = df_ws_enc.iloc[df_ws_enc.index.get_loc(1988)]

df_ws_enc["distance"] = np.sqrt(((x - df_ws_enc) ** 2).sum(axis = 1))
df_ws_enc["man_distance"] = ((x - df_ws_enc).abs()).sum(axis = 1)

df_ws_top = df_ws_enc.drop(index = 1988)

# Exclude 1988 from the top five.
df_ws_top.sort_values("distance").head(5)
```

year	Scaled Quantitative__win_pct	Scaled Quantitative__bat_avg	Scaled Quantitative__earned_run_average	Encoded Categorical__league_id_AL	Encoded Categorical__league_id_NL	Encoded Categorical__league_winner_Y	distance	man_distance
1965	-0.474623	-1.569393	-0.859699	0.0	1.0	1.0	0.527046	0.853553
1914	-0.564161	-1.186612	-0.971904	0.0	1.0	1.0	0.532423	0.898914
1963	-0.273585	-1.206732	-0.795583	0.0	1.0	1.0	0.683032	0.993048
2010	-1.196409	-0.805693	0.021907	0.0	1.0	1.0	0.913842	1.513539
1981	-1.083548	-0.476801	-0.539115	0.0	1.0	1.0	0.932788	1.168548

0.52 distance from the 1965 World Series winners with the Dodgers (Again)! Something about the distances between the specific 2020 World Series game was far off. Perhaps maybe it's too difficult to compare World Series winner, espcially because this subset is a lot smaller.

The most other similar games were in 1914 (Boston Braves), 1963 (Los Angeles Dodgers), 2010 (San Francisco Giants), and 1981 (Los Angeles Dodgers). Lots of recurring teams in both Euclidean distances, so maybe something more underlying within each such which is likely winning the division league.

Now let's compare to the overall games with all averaging scores, hits, and win percentages. I decided to use these variables since we can include teams of all years.

```
df_baseball_copy["avg_score"] = df_baseball_copy["runs_scored"] / df_baseball_copy["games_played"]
df_baseball_copy["avg_hits"] = df_baseball_copy["hits"] / df_baseball_copy["games_played"]
df_baseball_copy["win_pct"] = df_baseball_copy["wins"] / df_baseball_copy["games_played"]
df_baseball_copy
```

	year	league_id	division_id	rank	games_played	home_games	wins	losses	division_winner	wild_card_winner	...	double_plays	fielding_percentage	team_name	ball_park	home_attendance	avg_homerun	bat_avg	avg_score	avg_hits	win_pct
0	1876	NL	NaN	4	70	NaN	39	31	NaN	NaN	...	42	0.860	Boston Red Caps	South End Grounds I	NaN	0.128571	0.265614	6.728571	10.328571	0.557143
1	1876	NL	NaN	1	66	NaN	52	14	NaN	NaN	...	33	0.899	Chicago White Stockings	23rd Street Grounds	NaN	0.121212	0.336972	9.454545	14.030303	0.787879
2	1876	NL	NaN	8	65	NaN	9	56	NaN	NaN	...	45	0.841	Cincinnati Reds	Avenue Grounds	NaN	0.061538	0.233980	3.661538	8.538462	0.138462
3	1876	NL	NaN	2	69	NaN	47	21	NaN	NaN	...	27	0.888	Hartford Dark Blues	Hartford Ball Club Grounds	NaN	0.028986	0.266892	6.217391	10.304348	0.681159
4	1876	NL	NaN	5	69	NaN	30	36	NaN	NaN	...	44	0.875	Louisville Grays	Louisville Baseball Park	NaN	0.086957	0.249416	4.057971	9.289855	0.434783
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2779	2020	NL	C	3	58	27.0	30	28	N	Y	...	46	0.983	St. Louis Cardinals	Busch Stadium III	0.0	0.879310	0.234018	4.137931	7.068966	0.517241
2780	2020	AL	E	1	60	29.0	40	20	Y	N	...	52	0.985	Tampa Bay Rays	Tropicana Field	0.0	1.333333	0.237975	4.816667	7.833333	0.666667
2781	2020	AL	W	5	60	30.0	22	38	N	N	...	40	0.981	Texas Rangers	Globe Life Field	0.0	1.033333	0.216942	3.733333	7.000000	0.366667
2782	2020	AL	E	3	60	26.0	32	28	N	Y	...	47	0.982	Toronto Blue Jays	Sahlen Field	0.0	1.466667	0.255067	5.033333	8.600000	0.533333
2783	2020	NL	E	4	60	33.0	26	34	N	N	...	48	0.981	Washington Nationals	Nationals Park	0.0	1.100000	0.263720	4.883333	8.650000	0.433333

2784 rows × 46 columns

Find the 2020 Dodgers Statistics here.

```
df_baseball_copy[(df_baseball_copy['year'] == 2020) & (df_baseball_copy['team_name'] == "Los Angeles Dodgers")]
```

	year	league_id	division_id	rank	games_played	home_games	wins	losses	division_winner	wild_card_winner	...	double_plays	fielding_percentage	team_name	ball_park	home_attendance	avg_homerun	bat_avg	avg_score	avg_hits	win_pct
2767	2020	NL	W	1	60	30.0	43	17	Y	N	...	46	0.982	Los Angeles Dodgers	Dodger Stadium	0.0	1.966667	0.256121	5.816667	8.716667	0.716667

1 rows × 46 columns

Note: It's index 2767. Run the same code.

```
transformer = ColumnTransformer(
    [("Scaled Quantitative", StandardScaler(), ["win_pct", "bat_avg", "avg_hits", "avg_score",
                                                "avg_homerun", "earned_run_average"]),
     ("Encoded Categorical", OneHotEncoder(sparse_output = False), ["league_id"])])

transformer.fit(df_baseball_copy)
df_ws_enc = transformer.transform(df_baseball_copy)

x = df_ws_enc.iloc[2767]

df_ws_enc["distance"] = np.sqrt(((x - df_ws_enc) ** 2).sum(axis = 1))
df_ws_enc["man_distance"] = ((x - df_ws_enc).abs()).sum(axis = 1)

df_ws_top = df_ws_enc.drop(index = 2767)

# Exclude 2020 from the top five.
df_ws_top.sort_values("distance").head(5)
```

	Scaled Quantitative__win_pct	Scaled Quantitative__bat_avg	Scaled Quantitative__avg_hits	Scaled Quantitative__avg_score	Scaled Quantitative__avg_homerun	Scaled Quantitative__earned_run_average	Encoded Categorical__league_id_AL	Encoded Categorical__league_id_NL	distance	man_distance
2737	1.808578	-0.230097	-0.282977	1.244316	2.665955	-0.664207	0.0	1.0	1.167713	2.379223
2776	1.374672	-0.279173	-0.680370	1.176095	2.300982	-0.000982	0.0	1.0	2.012997	4.226273
1200	1.974408	0.090561	0.033248	0.730063	2.009450	-0.542390	1.0	0.0	2.355604	6.142851
2096	1.808578	0.607324	0.340548	0.762758	1.627808	-0.826630	0.0	1.0	2.356564	5.197451
2734	1.879710	0.755201	0.747917	1.517199	2.811945	-0.271686	1.0	0.0	2.381583	6.297931

A distance of 1.167713 is kinda high. We could look at another game. What about the 1986 New York Mets World Series winner.

```
df_baseball_copy.iloc[[2737, 2776, 1200, 2096, 2734]]
```

	year	league_id	division_id	rank	games_played	home_games	wins	losses	division_winner	wild_card_winner	...	double_plays	fielding_percentage	team_name	ball_park	home_attendance	avg_homerun	bat_avg	avg_score	avg_hits	win_pct
2737	2019	NL	W	1	162	81.0	106	56	Y	N	...	117	0.982	Los Angeles Dodgers	Dodger Stadium	3974309.0	1.722222	0.257419	5.469136	8.728395	0.654321
2776	2020	NL	W	2	60	32.0	37	23	N	Y	...	46	0.985	San Diego Padres	Petco Park	0.0	1.583333	0.256592	5.416667	8.433333	0.616667
1200	1961	AL	NaN	1	163	81.0	109	53	NaN	NaN	...	180	0.980	New York Yankees	Yankee Stadium I	1747725.0	1.472393	0.262817	5.073620	8.963190	0.668712
2096	1998	NL	E	1	162	81.0	106	56	Y	N	...	139	0.985	Atlanta Braves	Turner Field	3360860.0	1.327160	0.271517	5.098765	9.191358	0.654321
2734	2019	AL	W	1	162	81.0	107	55	Y	N	...	96	0.988	Houston Astros	Minute Maid Park	2857367.0	1.777778	0.274007	5.679012	9.493827	0.660494

5 rows × 46 columns

So while looking at the entire dataset, the LA Dodgers happen to be most similar to.... the LA Dodgers. Other noteable numbers are the San Diego Padres in 2020, the New York Yankees in 1961, the Atlanta Braves in 1998, and the Houston Astros in 2019. The lowest Euclidean distance is 1.17.

If we notice above, we notice that the either won in the division legaue or was a wildcard winner. Meaning that most, if not all the teams listed participated in the postseason. So at some point, they were all contenders to be the World Series champions.

```
df_baseball_copy[(df_baseball_copy['year'] == 1986) & (df_baseball_copy['team_name'] == "New York Mets")]
```

	year	league_id	division_id	rank	games_played	home_games	wins	losses	division_winner	wild_card_winner	...	double_plays	fielding_percentage	team_name	ball_park	home_attendance	avg_homerun	bat_avg	avg_score	avg_hits	win_pct
1788	1986	NL	E	1	162	81.0	108	54	Y	NaN	...	145	0.978	New York Mets	Shea Stadium	2767601.0	0.91358	0.263044	4.833333	9.024691	0.666667

1 rows × 46 columns

```
transformer = ColumnTransformer(
    [("Scaled Quantitative", StandardScaler(), ["win_pct", "bat_avg", "avg_hits", "avg_score",
```

```
                                          "avg_homerun", "earned_run_average"]),
    ("Encoded Categorical", OneHotEncoder(sparse_output = False), ["league_id"])])

transformer.fit(df_baseball_copy)
df_ws_enc = transformer.transform(df_baseball_copy)

x = df_ws_enc.iloc[1788]

df_ws_enc["distance"] = np.sqrt(((x - df_ws_enc) ** 2).sum(axis = 1))
df_ws_enc["man_distance"] = ((x - df_ws_enc).abs()).sum(axis = 1)

df_ws_top = df_ws_enc.drop(index = 1788)

# Exclude 1986 New York Mets from the top five.
df_ws_top.sort_values("distance").head(5)
```

	Scaled Quantitative__win_pct	Scaled Quantitative__bat_avg	Scaled Quantitative__avg_hits	Scaled Quantitative__avg_score	Scaled Quantitative__avg_homerun	Scaled Quantitative__earned_run_average	Encoded Categorical__league_id_AL	Encoded Categorical__league_id_NL	distance	man_distance
1954	1.666314	0.032014	-0.033567	0.289225	0.881640	-0.975517	0.0	1.0	0.492663	1.015881
2067	1.452917	0.489703	0.348861	0.481848	0.962745	-0.921376	0.0	1.0	0.801131	1.697055
2216	1.497540	-0.084331	-0.092847	-0.149013	0.817058	-0.989052	0.0	1.0	0.826233	1.720400
1477	1.524049	0.630617	0.523448	0.538030	0.395008	-1.205615	0.0	1.0	0.834914	1.816595
1045	1.405648	0.045505	-0.047513	0.634342	0.734185	-0.447644	0.0	1.0	0.857249	1.745705

```
df_baseball_copy.iloc[[1954, 2067, 2216, 1477, 1045]]
```

	year	league_id	division_id	rank	games_played	home_games	wins	losses	division_winner	wild_card_winner	...	double_plays	fielding_percentage	team_name	ball_park	home_attendance	avg_homerun	bat_avg	avg_score	avg_hits	win_pct
1954	1993	NL	W	1	162	81.0	104	58	Y	NaN	...	146	0.983	Atlanta Braves	Atlanta-Fulton County Stadium	3884720.0	1.043210	0.261831	4.734568	8.913580	0.641975
2067	1997	NL	E	1	162	81.0	101	61	Y	N	...	136	0.982	Atlanta Braves	Turner Field	3464488.0	1.074074	0.269537	4.882716	9.197531	0.623457
2216	2002	NL	E	1	161	81.0	101	59	Y	N	...	170	0.982	Atlanta Braves	Turner Field	2603484.0	1.018634	0.259873	4.397516	8.869565	0.627329
1477	1974	NL	W	1	162	81.0	102	60	Y	NaN	...	122	0.975	Los Angeles Dodgers	Dodger Stadium	2632474.0	0.858025	0.271909	4.925926	9.327160	0.629630
1045	1952	NL	NaN	1	155	80.0	96	57	NaN	NaN	...	169	0.982	Brooklyn Dodgers	Ebbets Field	1088704.0	0.987097	0.262058	5.000000	8.903226	0.619355

5 rows × 46 columns

The New York Mets are the most similar to the Atlanta Braves in 1993, 1997, 2002. Other noteable winners are the LA Dodgers in 1974 and Brooklyn Dodgers in 1952. Distances are as low around 0.5

So what is the story I am trying to describe?

First, I established that more games are played each year, resulting in increasing hits, homeruns, and runs scored. The grand question I aimed to answer was whether or not the Dodgers World Series win in 2020 a fluke.

After examining the data I found that the answer leaned towards a "yes", as the unique 2020 season contributed to the Dodgers victory.

The Euclidean Distances calculated from the 2020 year more than double the Euclidean Distance calculated from a normal season match. This difference was most likely caused by shortening the season.

In conclusion, the Dodgers 2020 World Series victory was remarkable different from the previous World Series winners. However, this analysis doesn't disprove the Dodgers ability and skills to win a World Series as a team. However, given the unusual conditions of the 2020 season, their victory involved a larger aspect of chance compared to previous baseball seasons.