Evolution of Right and Left Handed Pitchers of the MLB 2019-2023

Cindy Chen, JT Herren, Niha Suravarjjala

June 11th, 2024

1 Introduction

A standard MLB season involves each team playing 162 games starting in late March to late September. A majority of these games are played within a team's league for example the American and National League while some are played between the different leagues. For our purposes, we will be investigating the pitching data of multiple different pitchers from 2019 to 2023. The most notable deviation from the standard MLB season is that 2020 only included 60 games due to COVID-19.

To narrow down our analysis we intend to focus on 4 different types of pitches that were common amongst our pitchers of interest. Fastball is characterized by its quick speed and limited lateral movement, a curve ball is known for its unique downward trajectory and lower speed, the slider is known for its horizontal and quick movement, and change-up which is meant to appear like a fastball but arrives to the plate at a slower rate.

By investigating these different pitches and pitchers, we intend to analyze any changes that occur with the techniques by examining different motion characteristics, using the different seasons and dates of each game as the main response variables. Our main goal is to determine any significant evolutions in their styles.

2 Data Scraping & Exploration

We chose four pitchers to analyze: Clayton Kershaw, Max Scherzer, Jacob DeGrom, and Max Fried. We started our analysis with Clayton Kershaw, a left-handed pitcher who has been pitching for the LA Dodgers since 2008. To form a basis for comparison, we selected Max Scherzer, a right-handed pitcher for the Texas Rangers with a similar career length to Kershaw, and Jacob DeGrom, a right-handed pitcher also with the Texas Rangers but with fewer seasons than the other two. We wanted to compare a second left-handed pitcher, so we selected Max Fried from the Atlanta Braves. We also chose these pitchers because they primarily had the four types of pitches we discussed above.

We scraped pitch data for the three pitchers from Statcast using the `pybaseball` package in Python. We focused on the most recent five-year period, 2019-2023. We further filtered the data to look only at fastball, slider, curveball, and change-up pitches.

Statcast provides us with many different metrics for each pitch, but we chose a subset of explanatory variables to explore based on what was discussed in class and in the Physica A paper (Hsieh, 2022). Through our exploratory data analysis, we then identified six variables which would allow us to differentiate between different pitch types, and help us see changes in pitching over time: release speed, pfx_x, pfx_z, x acceleration, y acceleration, and release spin rate (Figure 1).

Finally, we filtered our data by removing NA values, since we found that the number of rows with NAs was insignificant compared to the overall scope of the data. We considered filtering out pitches that were classified as a strike, but we decided that a complete dataset would better suit our question of interest, which focuses more on how each pitcher changes their pitch over time rather than the result of the pitch. After filtering the data, each of our four data sets had 8 columns: the 6 variables from Figure 1 plus game date and pitch type. The number of rows for each data set is displayed in Figure 2.

Variables	Description
`release_speed`	speed of pitch (mph) when released
`pfx_x`	horizontal movement of pitch
`pfx_z`	vertical movement of pitch
`ax`	x-coordinate of pitch acceleration
`ay`	y-coordinate of pitch acceleration
`release_spin_rate`	spin rate

Figure 1: Summary of variables and their meaning.

Pitcher	Number of Rows
Clayton Kershaw (LH)	9758
Max Fried (LH)	10067
Max Scherzer (RH)	11658
Jacob DeGrom (RH)	7104

Figure 2: Summary of data set dimensions.

3 Methodology

For our initial exploration of the data, we focused on 15 specific variables that we split into 3 groups. Group 1 was composed of variables related to the speed and position of the ball: release speed, effective speed, release position x, release position y, and release position z. Group 2 was composed of variables related to the lateral movement and velocity of the ball: pfx_x (horizontal movement of pitch), pfx_z (vertical movement of pitch), vx0, vy0, and vz0. Group 3 was composed of variables related to how the ball spun and its acceleration: release spin rate, spin axis, ax, ay, az.

To best visualize the data we tried multiple different kinds of plots to better understand the variables over our chosen 5 year period. Within each group, we first produced pair plots to better visualize patterns between each variable. We color-coded the data points in the plots using the pitch types in hopes of spotting some initial differences. It was evident that clusters were forming from the data points of each pitching type. To better visualize this information over a specific period we decided to create line plots. We created line plots for each variable where the y-axis is the variable and the x-axis is the year. The plot includes a line for each pitch type based on the average value of each variable for a given year. This made it much simpler to view the patterns of the variables individually. From these plots, we chose 6 specific variables to focus on based on fluctuation. These variables were release speed, pfx_x, pfx_z, ax, ay, and release spin rate.

For the 6 specific variables, we made a few additional plots to compare the trends in these variables for different pitchers. In specific, we compared the two left-handed pitchers (Kershaw and Fried) and the two right-handed pitchers (Scherzer and DeGrom) by combining their line plots, but by setting one pitcher to a different line type. These plots allowed us to see the pitchers' trends for a variable, such as release speed, across the years 2019-2023, by pitch type.

4 Results & Discussion

4.1 Clayton Kershaw (LH)

In our line plots, we noticed the pitch that varied the most over time was Kershaw's change-up, which makes sense as this was by far his least frequent pitch (Figure 3). The other pitch types were fairly consistent in release speed, movement, and x acceleration. There were some notable trends in y acceleration: all four of his pitches had faster y acceleration in 2020, which could be skewed due to the fact it was a shorter MLB season. The y acceleration also dipped in 2022 for all pitch types. Interestingly, Kershaw's release spin rate had a continual decline over 5 years, across all four pitch types, but especially noticeable in his change-up.

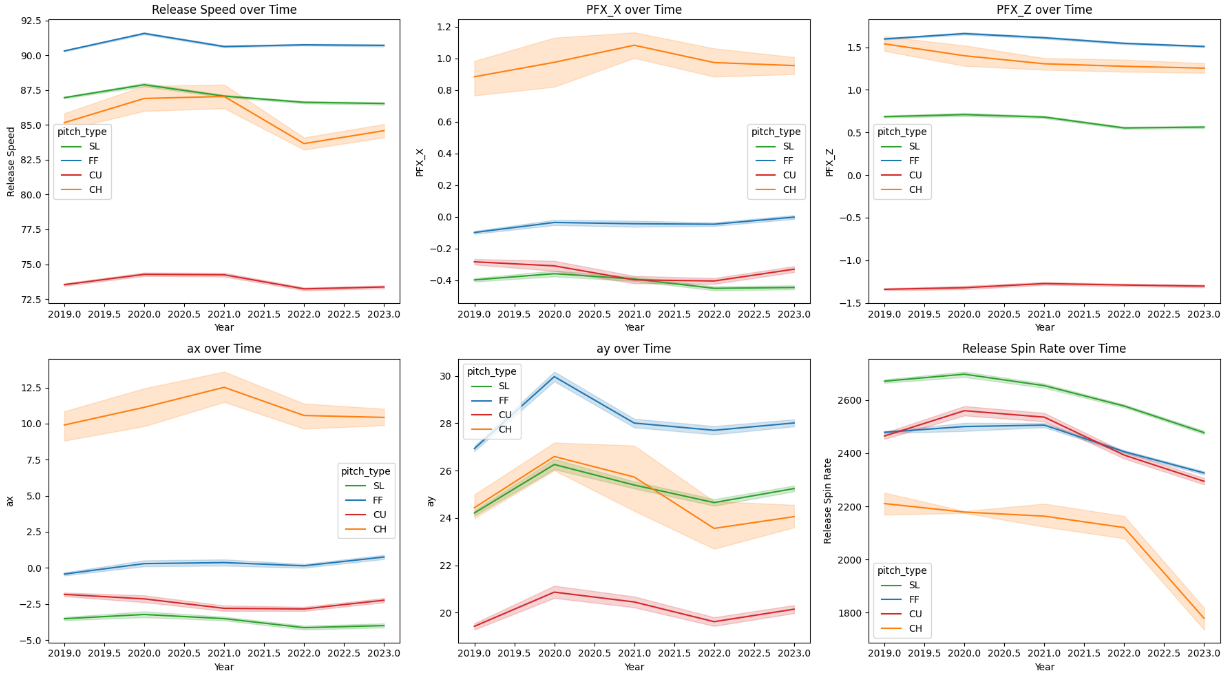


Figure 3: Clayton Kershaw: Lineplots of pitch variables over time, by pitch type.

4.2 Max Fried (LH)

Max Fried saw a lot of change in his slider pitch over time (Figure 4). In particular, for his slider, the pitch movement increased in both x and y, as well as the acceleration in x and y. The release speed for his slider fluctuates and peaks in 2022. His other pitch types appear to stay relatively consistent in release speed, movement and x acceleration. There is an increasing trend in the y acceleration of the fastball and change-up pitches. For release spin rate, there is a slight decreasing trend for the slider, but a slight increase for the fastball pitch type. His curveball pitch appears fairly consistent in trend across all metrics since 2019, despite having some fluctuations.

4.3 Max Scherzer (RH)

All four of Max Scherzer's pitches experienced a continual decline in release speed over time, with his fastball and curveball showing the most noticeable change (Figure 5). For pitch movement, Scherzer's curveball had the most fluctuation, with little variation in the other pitch types. For acceleration, there was more change in the y direction than the x direction, for all four pitch types. Specifically, the y acceleration appears to increase for his fastball and curveball. Finally, for spin rate, his fastball and change-up decreased in rate over time, while the slider and change-up had some variation but overall had less of a decline. Scherzer displays interesting trends as his acceleration in y increases but yet the overall release speed and spin rate has a decreasing pattern.

Figure 4: Max Fried: Lineplots of pitch variables over time, by pitch type.

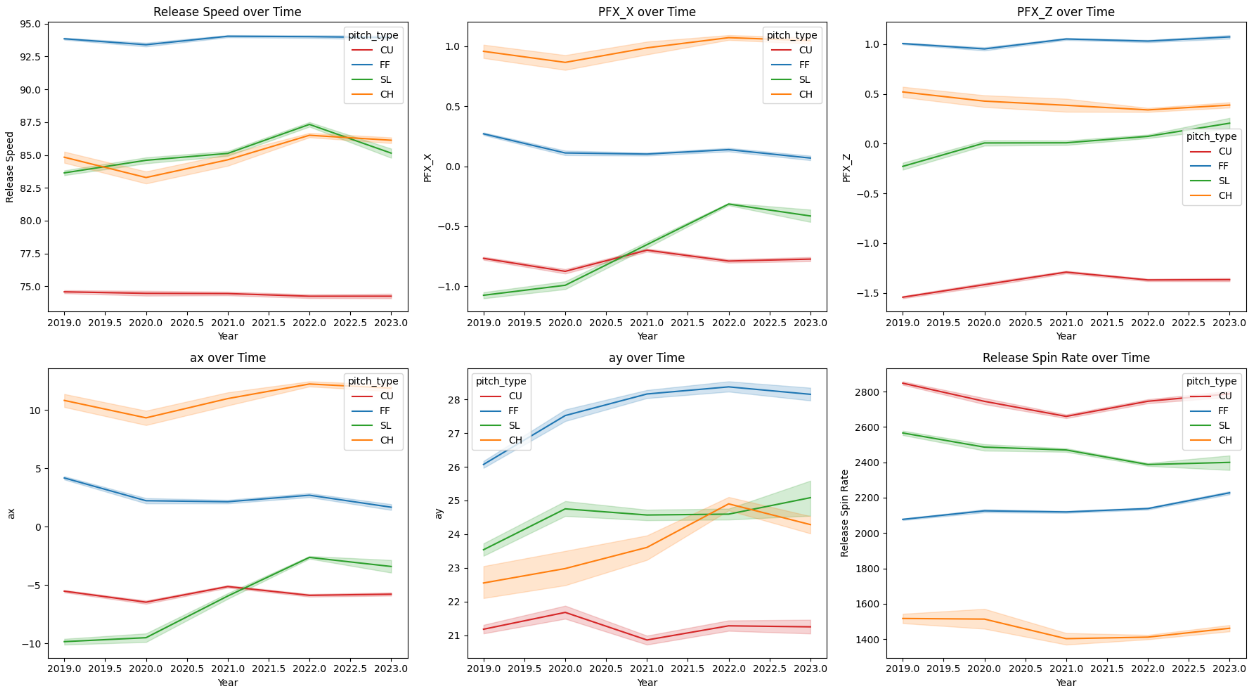
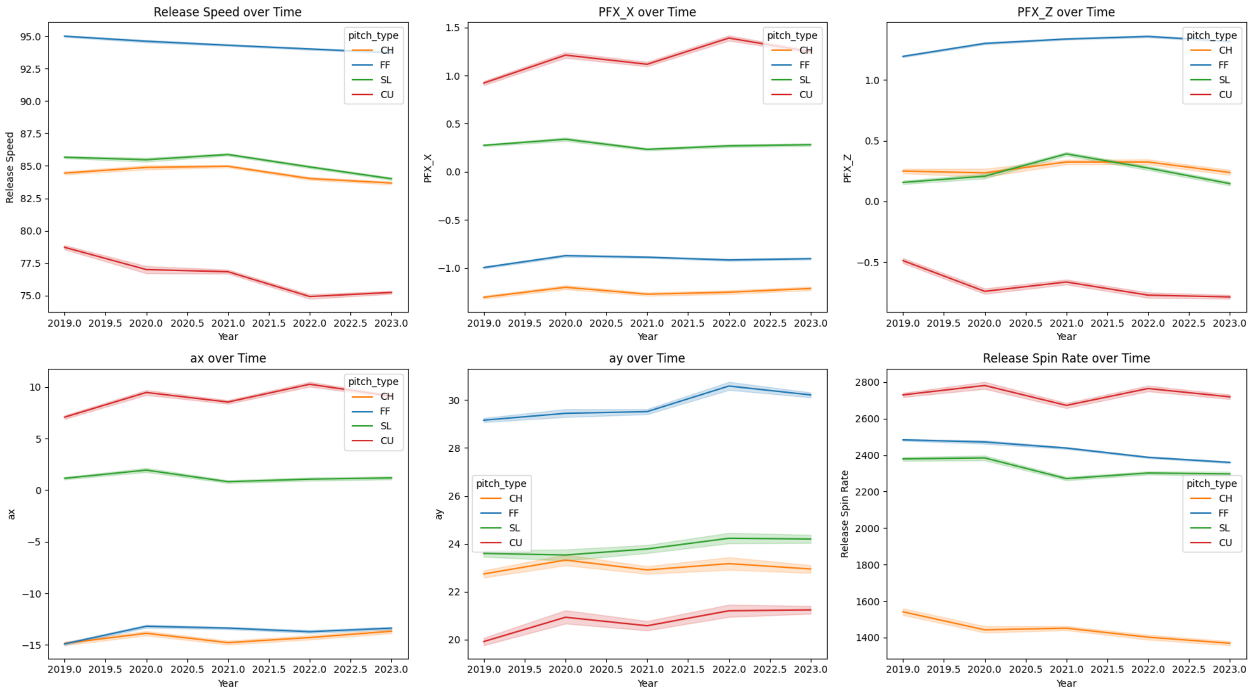


Figure 5: Max Scherzer: Lineplots of pitch variables over time, by pitch type.



4.4 Jacob DeGrom (RH)

For Release speed there appears to be minimal fluctuation with a slight decrease and increase in speed for his curveball and slider however they normalize back to the initial speeds (Figure 6). For release spin rate all pitches except for change up appear to have a slight increase overtime. The lateral movement variables show some change with fastball's horizontal movement declining however vertical movement remains relatively unchanged. The other pitches remain relatively unchanged in horizontal movement. However, when it comes to the other pitches, they follow a similar trend of decreasing before spiking and then decreasing again For the acceleration variables all of the pitches' horizontal acceleration follow a similar trend to the horizontal acceleration. Contrasting this similarity between horizontal acceleration and movement vertical acceleration has a gradual increase for all pitches contrasting the differing patterns in vertical movement. The overall spread of the variables seems wider for curveball than any other pitch.

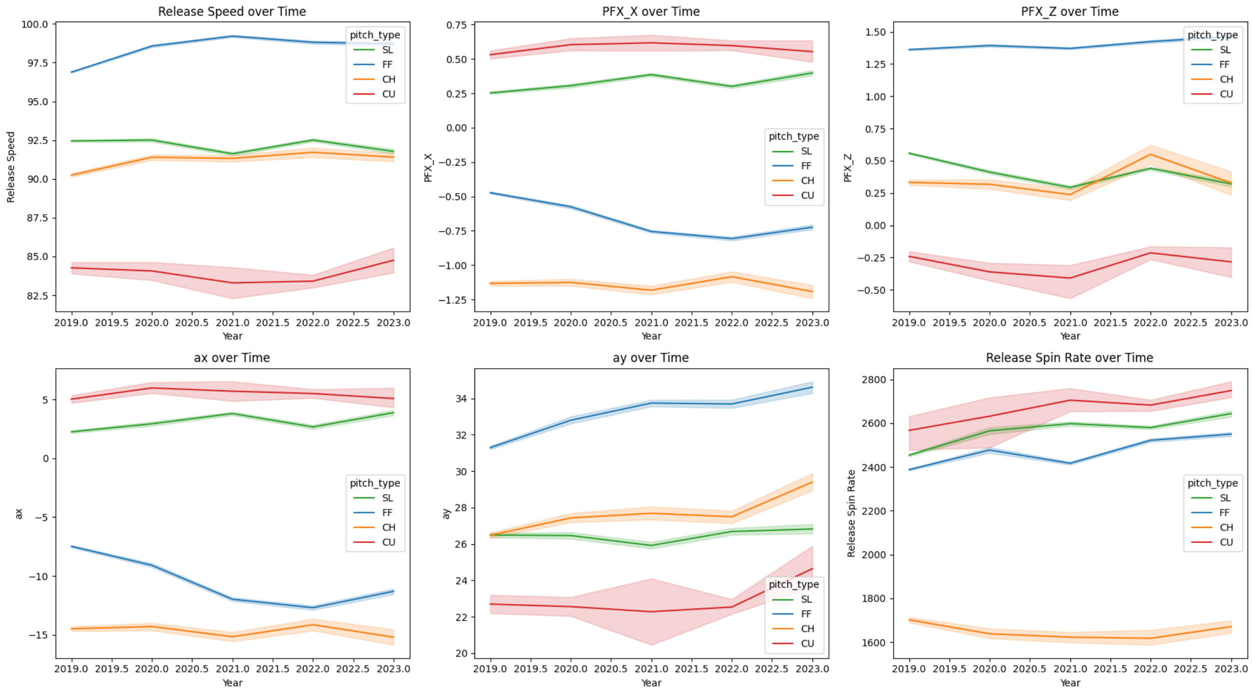


Figure 6: Jacob DeGrom: Lineplots of pitch variables over time, by pitch type.

4.5 Comparison of Pitchers

4.5.1 Left-handed Pitchers (Kershaw & Fried)

Kershaw and Fried, the two left-handed pitchers, have very similar release speed for their pitches, especially the curveball, but Fried has a quicker fastball (Figure 7). Interestingly, while Fried's change-up dips in speed in 2020 and then increases, Kershaw has the opposite trend, with faster change-ups before 2021. For movement, both pitchers stay fairly consistent in their four pitch types across time, except for Fried's slider as discussed previously. For acceleration, Fried's fastballs have more acceleration than Kershaw's, although all three of of the other pitch types are similar. An interesting comparison point is release spin rate, where there is a significant difference in the spin rate of the change-up. Fried has a much slower spin rate for his change-up compared to Kershaw (Figure 8). For spin rate, it appears Fried has a wider spread across his different pitches, whereas Kershaw's spin rates fall in a smaller range. The spin rate seems to converge for both of the pitchers as time goes on, with the exception of the curveball - Fried's increases in spin rate in 2023 while Kershaw's decreases in the opposite direction.

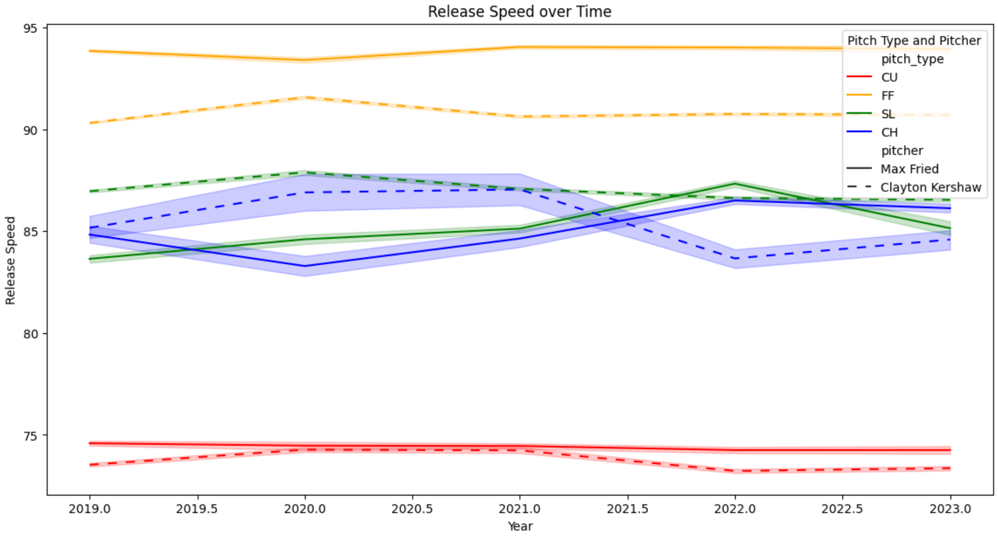


Figure 7: Comparison of release speed for left-handed pitchers (Kershaw and Fried).

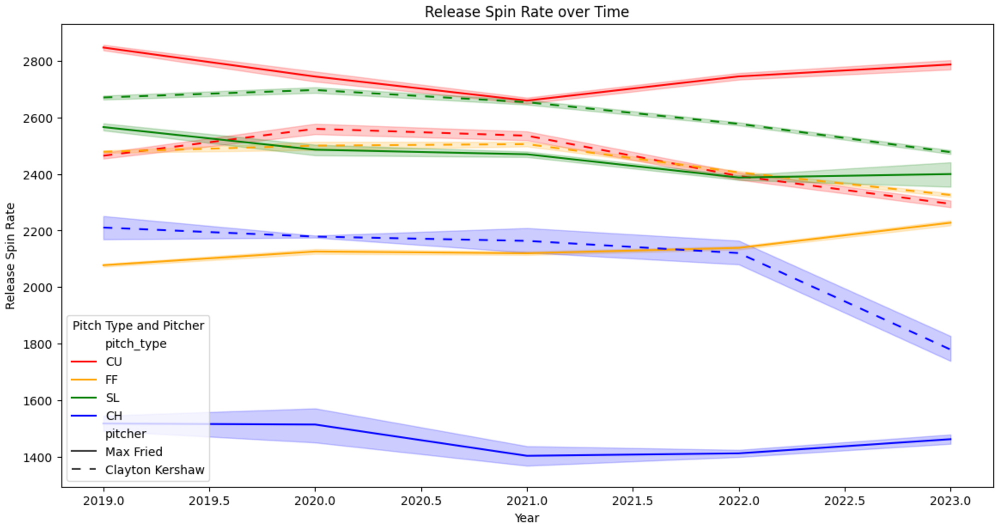


Figure 8: Comparison of release spin rate for left-handed pitchers (Kershaw and Fried).

4.5.2 Right-handed Pitchers (Scherzer & DeGrom)

Degrom and Scherzer differ when it comes to pitching speed with Degrom's technique seeming to remain the same while Scherzer seems to be gradually increasing release speed for all pitches (Figure 9). For release speed, Degrom also displays consistently higher speeds for all 4 pitch types compared to Scherzer. When it comes to horizontal position it appears that Degrom and Scherzer each change one pitch the most, which are fastball and curveball respectively. For vertical movement all of their pitches follow relatively similar trends and values. However, curveball does differ between the two with Scherzer's gradually decreasing and Degrom's remains relatively the same. For release spin rate their techniques seem to contrast with Degrom's rate gradually increasing for all but change-up and all of Scherzer's pitches appear to gradually increase. Both of their horizontal accelerations increase as with Degrom's rate of increase being slightly higher. Their vertical accelerations have some slight fluctuations but both return to where they started except for Degrom's fastball which gradually decreases.

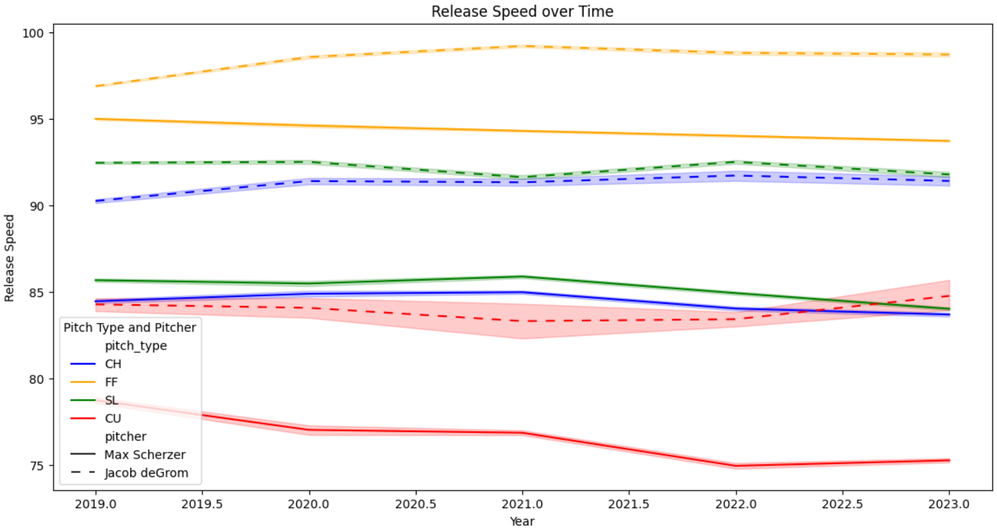


Figure 9: Comparison of release speed for right-handed pitchers (Degrom and Scherzer).

5 Conclusion

The pitchers have very different pitching characteristics, even for the same pitch types. Additionally, each individual pitcher's evolution over time differs from the other pitchers. For example, in our comparison of two left-handed starting pitchers, we saw that Clayton Kershaw has a decreasing trend in release spin rate and a unique pattern for acceleration that differs from the patterns that Max Fried has. There are also some patterns that are evident across the entire time span: for example, although Max Scherzer and Jacob Degrom are both right-handed, Degrom has faster release speed for every single pitch type compared to Scherzer. Although only a subset of our exploratory and comparison plots are displayed in this report, it is evident that pitch movement, acceleration, release speed, and release spin rate are important features that vary over time for each pitcher. These six variables can be used to compare different pitchers as well as differentiate between pitch types.

There is a lot of room for further exploration in the future. Specifically, we would like to research what factors could potentially explain the trends that we see, like whether these pitchers may have had injuries or surgeries that affect how they pitch. Additionally, some of the pitchers we looked at, like Max Scherzer, went through multiple MLB teams since 2019, so it could be interesting to see how that might be affecting his pitch evolution as well. Finally, we could look into methods and metrics to quantify how similar or how different two pitchers are, and then we could track whether two pitchers became more similar or more different over time.

6 References

Albert, J., Baumer, B. S., & Marchi, M. (n.d.). *Appendix C — Statcast Data Reference.* Analyzing Baseball Data with R (3e) https://beanumber.github.io/abdwr3e/C_statcast.html

Dhakar, Lokesh (n.d.). *Baseball pitches illustrated.* Baseball pitches illustrated — Lokesh Dhakar. https://lokeshdhakar.com/baseball-pitches-illustrated/

Fushing, H., Chou, E. P., & Chen, T.-L. (2023). *Multiscale major factor selections for complex system data with structural dependency and heterogeneity.* Physica A: Statistical Mechanics and Its Applications, 630, 129227. https://doi.org/10.1016/j.physa.2023.129227

7 Appendix

```
Automatically generated by Colab.

Original file is located at
    https://colab.research.google.com/drive/1
       qvPMtnTKlCyv_YWsQDgQf3Cwys1W6sQg
”””
pip install pybaseball

”””# Clayton Kershaw”””

from pybaseball import statcast_pitcher, playerid_lookup
import pandas as pd

def get_pitcher_data(last_name, first_name, start_year, end_year):
    # Find the player's MLBAM ID
    player_info = playerid_lookup(last_name, first_name)
    player_id = player_info.iloc[0]['key_mlbam']

    # Function to get data for multiple seasons
    def get_pitcher_data_season(player_id, year):
        data = statcast_pitcher(f”{year}−01−01”, f”{year}−12−31”,
            player_id)
        data['year'] = year
        return data

    # Get data for each year in the specified range
    all_data = [get_pitcher_data_season(player_id, year) for year in
       range(start_year, end_year + 1)]
    pitcher_data = pd.concat(all_data, ignore_index=True)

    return pitcher_data

# Gather Clayton Kershaw's pitching data for the desired five−year
   range
kershaw_data = get_pitcher_data('kershaw', 'clayton', 2019, 2023)

# Display the first few rows of the DataFrame
print(”Clayton Kershaw's Pitching Data:”)
print(kershaw_data.head())

# Filter by pitch type
pitch_types = ['FF', 'CU', 'SL', 'CH']
kershaw_pitch = kershaw_data[kershaw_data['pitch_type'].isin(
   pitch_types)]

# Define the columns to include in the subset
subset = ['pitch_type', 'game_date', 'release_speed', 'effective_speed
   ', 'release_pos_x', 'release_pos_y', 'release_pos_z', 'pfx_x', '
   pfx_z', 'vx0', 'vy0', 'vz0', 'ax', 'ay', 'az', 'release_spin_rate',
   'spin_axis']
```

```
# Create the new subset by filtering the DataFrame
kershaw_filtered = kershaw_pitch[subset]

# Drop rows with missing values
kershaw_dropna = kershaw_filtered.dropna()

#final subset with primary vars of interest
new_subset = ['pitch_type', 'game_date','release_speed', 'pfx_x', '
   pfx_z', 'ax', 'ay', 'release_spin_rate']
kershaw_data = kershaw_dropna[new_subset]

# Check data dimensions
kershaw_data.shape

"""# Clayton Kershaw plots"""

import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

# Convert game_date to datetime
kershaw_data['game_date'] = pd.to_datetime(kershaw_data['game_date'])

# Extract year and month
kershaw_data['year'] = kershaw_data['game_date'].dt.year

# Create a custom color palette for the pitch types
pitch_palette = {
    'FF': '#1f77b4', # Blue
    'CH': '#ff7f0e', # Orange
    'SL': '#2ca02c', # Green
    'CU': '#d62728'  # Red
}

# Create a 2x3 grid for subplots
fig, axes = plt.subplots(2, 3, figsize=(18, 10))

# Line plot for Release Speed over Time
sns.lineplot(data=kershaw_data, x='year', y='release_speed', hue='
   pitch_type', ax=axes[0, 0], palette=pitch_palette)
axes[0, 0].set_title('Release Speed over Time')
axes[0, 0].set_xlabel('Year')
axes[0, 0].set_ylabel('Release Speed')

# Line plot for PFX_X over Time
sns.lineplot(data=kershaw_data, x='year', y='pfx_x', hue='pitch_type',
   ax=axes[0, 1], palette=pitch_palette)
axes[0, 1].set_title('PFX_X over Time')
axes[0, 1].set_xlabel('Year')
axes[0, 1].set_ylabel('PFX_X')
```

```
# Line plot for PFX_Z over Time
sns.lineplot(data=kershaw_data, x='year', y='pfx_z', hue='pitch_type',
    ax=axes[0, 2],palette=pitch_palette)
axes[0, 2].set_title('PFX_Z over Time')
axes[0, 2].set_xlabel('Year')
axes[0, 2].set_ylabel('PFX_Z')

# Line plot for ax over Time
sns.lineplot(data=kershaw_data, x='year', y='ax', hue='pitch_type', ax=
    axes[1, 0],palette=pitch_palette)
axes[1, 0].set_title('ax over Time')
axes[1, 0].set_xlabel('Year')
axes[1, 0].set_ylabel('ax')

# Line plot for ay over Time
sns.lineplot(data=kershaw_data, x='year', y='ay', hue='pitch_type', ax=
    axes[1, 1],palette=pitch_palette)
axes[1, 1].set_title('ay over Time')
axes[1, 1].set_xlabel('Year')
axes[1, 1].set_ylabel('ay')

# Line plot for Release Spin Rate over Time
sns.lineplot(data=kershaw_data, x='year', y='release_spin_rate', hue='
    pitch_type', ax=axes[1, 2],palette=pitch_palette)
axes[1, 2].set_title('Release Spin Rate over Time')
axes[1, 2].set_xlabel('Year')
axes[1, 2].set_ylabel('Release Spin Rate')

# Adjust layout
plt.tight_layout()
plt.show()

# Define the list of variables
variables = ['release_speed', 'pfx_x', 'pfx_z', 'ax', 'ay', '
    release_spin_rate']

# Create a 2x3 grid of subplots
fig, axes = plt.subplots(2, 3, figsize=(20, 10))

# Flatten the axes array for easier iteration
axes = axes.flatten()

# Iterate over each variable and create scatter plots
for i, var in enumerate(variables):
    # Scatter plot for the current variable
    for idx, pitch_type in enumerate(kershaw_data['pitch_type'].unique
        ()):
        data_filtered = kershaw_data[kershaw_data['pitch_type'] ==
            pitch_type]
        sns.scatterplot(data=data_filtered, x='game_date', y=var, ax=
            axes[i], color=pitch_palette[pitch_type], label=pitch_type)
    axes[i].set_title(f'{var} vs Game Date')
    axes[i].set_xlabel('Game Date')
    axes[i].set_ylabel(var)
    axes[i].legend(title='Pitch Type')

# Adjust layout
plt.tight_layout()
plt.show()

# Loop over each variable
for var in variables:
    # Filter data for the current variable
    var_data = kershaw_data[['game_date', var, 'year']]

    # Plot scatterplot with all points black
    plt.figure(figsize=(14, 7))
    sns.scatterplot(data=var_data, x='game_date', y=var, color='black',
         alpha=0.5)

    # Plot lineplot for average value per year with a brighter color
       scheme
    sns.lineplot(data=var_data, x='game_date', y=var, hue='year',
       estimator='mean', ci=None, lw=2, palette='bright')

    # Set title and labels
    plt.title(f'{var} Over Time')
    plt.xlabel('Date')
    plt.ylabel(var)
    plt.legend(title='Year')
    plt.show()

# plot pfx_x and pfx_z against each other, for years 2019-2023
years = [2019,2020,2021,2022,2023]

num_cols = 3
num_rows = (len(years) + num_cols - 1) // num_cols

fig, axes = plt.subplots(num_rows, num_cols, figsize=(15, 10))
axes = axes.flatten()

for i, year in enumerate(years):
    ax = axes[i]
    data = kershaw_data[kershaw_data['year'] == year]
    sns.scatterplot(data = data, x= 'pfx_x', y = 'pfx_z', hue='
       pitch_type',
                    hue_order=['FF','SL','CU','CH',], ax=ax)
    ax.set_title('PFX_X and PFX_Z in '+str(year))
    ax.set(xlim=(-2, 2), ylim=(-2, 2.5))
plt.tight_layout()
plt.show()

# plot ax and ay against each other, for years 2019-2023
years = [2019,2020,2021,2022,2023]
```

```
num_cols = 3
num_rows = (len(years) + num_cols - 1) // num_cols

fig, axes = plt.subplots(num_rows, num_cols, figsize=(15, 10))
axes = axes.flatten()

for i, year in enumerate(years):
    ax = axes[i]
    data = kershaw_data[kershaw_data['year'] == year]
    sns.scatterplot(data = data, x= 'ax', y = 'ay', hue='pitch_type',
                    hue_order=['FF','SL','CU','CH',], ax=ax)
    ax.set_title('ax and ay in '+str(year))
    ax.set(xlim=(-25, 15), ylim=(0, 50))
plt.tight_layout()
plt.show()

"""# Max Fried"""

# Gather Max Fried's pitching data for the desired five-year range
fried_data = get_pitcher_data('fried', 'max', 2019, 2023)

# Display the first few rows of the DataFrame
print("Max Fried's Pitching Data:")
print(fried_data.head())

# Filter by pitch type
pitch_types = ['FF', 'CU', 'SL', 'CH']
fried_pitch = fried_data[fried_data['pitch_type'].isin(pitch_types)]

# Define the columns to include in the subset
subset = ['pitch_type', 'game_date', 'release_speed', 'effective_speed
   ', 'release_pos_x', 'release_pos_y', 'release_pos_z', 'pfx_x', '
   pfx_z', 'vx0', 'vy0', 'vz0', 'ax', 'ay', 'az', 'release_spin_rate',
   'spin_axis']

# Create the new subset by filtering the DataFrame
fried_filtered = fried_pitch[subset]

# Drop rows with missing values
fried_dropna = fried_filtered.dropna()

#final subset with primary vars of interest
new_subset = ['pitch_type', 'game_date','release_speed', 'pfx_x', '
   pfx_z', 'ax', 'ay', 'release_spin_rate']
fried_data = fried_dropna[new_subset]

# Check data dimensions
fried_data.shape

"""# Max Fried Plots"""
```

```
# Convert game_date to datetime
fried_data['game_date'] = pd.to_datetime(fried_data['game_date'])

# Extract year and month
fried_data['year'] = fried_data['game_date'].dt.year

# Create a custom color palette for the pitch types
pitch_palette = {
    'FF': '#1f77b4',  # Blue
    'CH': '#ff7f0e',  # Orange
    'SL': '#2ca02c',  # Green
    'CU': '#d62728'   # Red
}

# Create a 2x3 grid for subplots
fig, axes = plt.subplots(2, 3, figsize=(18, 10))

# Line plot for Release Speed over Time
sns.lineplot(data=fried_data, x='year', y='release_speed', hue='
    pitch_type', ax=axes[0, 0], palette=pitch_palette)
axes[0, 0].set_title('Release Speed over Time')
axes[0, 0].set_xlabel('Year')
axes[0, 0].set_ylabel('Release Speed')

# Line plot for PFX_X over Time
sns.lineplot(data=fried_data, x='year', y='pfx_x', hue='pitch_type', ax
   =axes[0, 1],palette=pitch_palette)
axes[0, 1].set_title('PFX_X over Time')
axes[0, 1].set_xlabel('Year')
axes[0, 1].set_ylabel('PFX_X')

# Line plot for PFX_Z over Time
sns.lineplot(data=fried_data, x='year', y='pfx_z', hue='pitch_type', ax
   =axes[0, 2],palette=pitch_palette)
axes[0, 2].set_title('PFX_Z over Time')
axes[0, 2].set_xlabel('Year')
axes[0, 2].set_ylabel('PFX_Z')

# Line plot for ax over Time
sns.lineplot(data=fried_data, x='year', y='ax', hue='pitch_type', ax=
    axes[1, 0],palette=pitch_palette)
axes[1, 0].set_title('ax over Time')
axes[1, 0].set_xlabel('Year')
axes[1, 0].set_ylabel('ax')

# Line plot for ay over Time
sns.lineplot(data=fried_data, x='year', y='ay', hue='pitch_type', ax=
    axes[1, 1],palette=pitch_palette)
axes[1, 1].set_title('ay over Time')
axes[1, 1].set_xlabel('Year')
axes[1, 1].set_ylabel('ay')
```

```
# Line plot for Release Spin Rate over Time
sns.lineplot(data=fried_data, x='year', y='release_spin_rate', hue='
   pitch_type', ax=axes[1, 2],palette=pitch_palette)
axes[1, 2].set_title('Release Spin Rate over Time')
axes[1, 2].set_xlabel('Year')
axes[1, 2].set_ylabel('Release Spin Rate')

# Adjust layout
plt.tight_layout()
plt.show()

# Define the list of variables
variables = ['release_speed', 'pfx_x', 'pfx_z', 'ax', 'ay', '
   release_spin_rate']

# Create a 2x3 grid of subplots
fig, axes = plt.subplots(2, 3, figsize=(20, 10))

# Flatten the axes array for easier iteration
axes = axes.flatten()

# Iterate over each variable and create scatter plots
for i, var in enumerate(variables):
    # Scatter plot for the current variable
    for idx, pitch_type in enumerate(fried_data['pitch_type'].unique())
       :
        data_filtered = fried_data[fried_data['pitch_type'] ==
           pitch_type]
        sns.scatterplot(data=data_filtered, x='game_date', y=var, ax=
           axes[i], color=pitch_palette[pitch_type], label=pitch_type)
    axes[i].set_title(f'{var} vs Game Date')
    axes[i].set_xlabel('Game Date')
    axes[i].set_ylabel(var)
    axes[i].legend(title='Pitch Type')

# Adjust layout
plt.tight_layout()
plt.show()

# Loop over each variable
for var in variables:
    # Filter data for the current variable
    var_data = fried_data[['game_date', var, 'year']]

    # Plot scatterplot with all points black
    plt.figure(figsize=(14, 7))
    sns.scatterplot(data=var_data, x='game_date', y=var, color='black',
        alpha=0.5)

    # Plot lineplot for average value per year with a brighter color
       scheme
    sns.lineplot(data=var_data, x='game_date', y=var, hue='year',
       estimator='mean', ci=None, lw=2, palette='bright')

    # Set title and labels
    plt.title(f'{var} Over Time')
    plt.xlabel('Date')
    plt.ylabel(var)
    plt.legend(title='Year')
    plt.show()

# plot pfx_x and pfx_z against each other, for years 2019-2023
years = [2019,2020,2021,2022,2023]

num_cols = 3
num_rows = (len(years) + num_cols - 1) // num_cols

fig, axes = plt.subplots(num_rows, num_cols, figsize=(15, 10))
axes = axes.flatten()

for i, year in enumerate(years):
    ax = axes[i]
    data = fried_data[fried_data['year'] == year]
    sns.scatterplot(data = data, x= 'pfx_x', y = 'pfx_z', hue='
       pitch_type',
                    hue_order=['FF','SL','CU','CH',], ax=ax)
    ax.set_title('PFX_X and PFX_Z in '+str(year))
    ax.set(xlim=(-2, 2), ylim=(-2, 2.5))
plt.tight_layout()
plt.show()

# plot ax and ay against each other, for years 2019-2023
years = [2019,2020,2021,2022,2023]

num_cols = 3
num_rows = (len(years) + num_cols - 1) // num_cols

fig, axes = plt.subplots(num_rows, num_cols, figsize=(15, 10))
axes = axes.flatten()

for i, year in enumerate(years):
    ax = axes[i]
    data = fried_data[fried_data['year'] == year]
    sns.scatterplot(data = data, x= 'ax', y = 'ay', hue='pitch_type',
                    hue_order=['FF','SL','CU','CH',], ax=ax)
    ax.set_title('ax and ay in '+str(year))
    ax.set(xlim=(-25, 15), ylim=(0, 50))
plt.tight_layout()
plt.show()

"""# Comparison plots"""

import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Convert game_date to datetime
fried_data['game_date'] = pd.to_datetime(fried_data['game_date'])
kershaw_data['game_date'] = pd.to_datetime(kershaw_data['game_date'])

# Extract year
fried_data['year'] = fried_data['game_date'].dt.year
kershaw_data['year'] = kershaw_data['game_date'].dt.year

# Add a pitcher column to each dataset
fried_data['pitcher'] = 'Max Fried'
kershaw_data['pitcher'] = 'Clayton Kershaw'

# Combine the datasets
combined_data = pd.concat([fried_data, kershaw_data])

# Define the custom color palette
custom_palette = {
    'CH': 'blue',
    'FF': 'orange',
    'SL': 'green',
    'CU': 'red'
}

# Plot Release Speed over Time
plt.figure(figsize=(14, 7))
sns.lineplot(data=combined_data, x='year', y='release_speed', hue='
   pitch_type', style='pitcher',
            dashes={'Max Fried': '', 'Clayton Kershaw': (5, 5)},
               palette=custom_palette)
plt.title('Release Speed over Time')
plt.xlabel('Year')
plt.ylabel('Release Speed')
plt.legend(title='Pitch Type and Pitcher')
plt.show()

# Plot PFX_X over Time
plt.figure(figsize=(14, 7))
sns.lineplot(data=combined_data, x='year', y='pfx_x', hue='pitch_type',
    style='pitcher',
            dashes={'Max Fried': '', 'Clayton Kershaw': (5, 5)},
               palette=custom_palette)
plt.title('PFX_X over Time')
plt.xlabel('Year')
plt.ylabel('PFX_X')
plt.legend(title='Pitch Type and Pitcher')
plt.show()

# Plot PFX_Z over Time
plt.figure(figsize=(14, 7))
sns.lineplot(data=combined_data, x='year', y='pfx_z', hue='pitch_type',
    style='pitcher',
             dashes={'Max Fried': '', 'Clayton Kershaw': (5, 5)},
                 palette=custom_palette)
plt.title('PFX_Z over Time')
plt.xlabel('Year')
plt.ylabel('PFX_Z')
plt.legend(title='Pitch Type and Pitcher')
plt.show()

# Plot AX over Time
plt.figure(figsize=(14, 7))
sns.lineplot(data=combined_data, x='year', y='ax', hue='pitch_type',
    style='pitcher',
             dashes={'Max Fried': '', 'Clayton Kershaw': (5, 5)},
                 palette=custom_palette)
plt.title('Acceleration (x) over Time')
plt.xlabel('Year')
plt.ylabel('Acceleration (x)')
plt.legend(title='Pitch Type and Pitcher')
plt.show()

# Plot AY over Time
plt.figure(figsize=(14, 7))
sns.lineplot(data=combined_data, x='year', y='release_speed', hue='
    pitch_type', style='pitcher',
             dashes={'Max Fried': '', 'Clayton Kershaw': (5, 5)},
                 palette=custom_palette)
plt.title('Acceleration (y) over Time')
plt.xlabel('Year')
plt.ylabel('Acceleration (y)')
plt.legend(title='Pitch Type and Pitcher')
plt.show()

# Plot Release Spin Rate over Time
plt.figure(figsize=(14, 7))
sns.lineplot(data=combined_data, x='year', y='release_spin_rate', hue='
    pitch_type', style='pitcher',
             dashes={'Max Fried': '', 'Clayton Kershaw': (5, 5)},
                 palette=custom_palette)
plt.title('Release Spin Rate over Time')
plt.xlabel('Year')
plt.ylabel('Release Spin Rate')
plt.legend(title='Pitch Type and Pitcher')
plt.show()

"""righthanded_pitchers.ipynb

Automatically generated by Colab.

Original file is located at
    https://colab.research.google.com/drive/1P07Zi4Ll-ti_Wr8Px-
        UraTjjdb1y80ZO
```

```
”””

”””# Max Scherzer”””

# Gather Max Scherzer’s pitching data for the desired five−year range
scherzer_data = get_pitcher_data(’scherzer’, ’max’, 2019, 2023)

# Display the first few rows of the DataFrame
print(”Max Scherzer’s Pitching Data:”)
print(scherzer_data.head())

# Filter by pitch type
pitch_types = [’FF’, ’CU’, ’SL’, ’CH’]
scherzer_pitch = scherzer_data[scherzer_data[’pitch_type’].isin(
   pitch_types)]

# Define the columns to include in the subset
subset = [’pitch_type’, ’game_date’, ’release_speed’, ’effective_speed
   ’, ’release_pos_x’, ’release_pos_y’, ’release_pos_z’, ’pfx_x’, ’
   pfx_z’, ’vx0’, ’vy0’, ’vz0’, ’ax’, ’ay’, ’az’, ’release_spin_rate’,
   ’spin_axis’]

# Create the new subset by filtering the DataFrame
scherzer_filtered = scherzer_pitch[subset]

# Drop rows with missing values
scherzer_dropna = scherzer_filtered.dropna()

#final subset with primary vars of interest
new_subset = [’pitch_type’, ’game_date’,’release_speed’, ’pfx_x’, ’
   pfx_z’, ’ax’, ’ay’, ’release_spin_rate’]
scherzer_data = scherzer_dropna[new_subset]

# Check data dimensions
scherzer_data.shape

”””# Max Scherzer plots”””

import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Custom color palette for the pitch types
pitch_palette = {
    ’FF’: ’#1f77b4’,  # Blue
    ’CH’: ’#ff7f0e’,  # Orange
    ’SL’: ’#2ca02c’,  # Green
    ’CU’: ’#d62728’   # Red
}

# Convert game_date to datetime
scherzer_data[’game_date’] = pd.to_datetime(scherzer_data[’game_date’])
```

```
# Extract year
scherzer_data['year'] = scherzer_data['game_date'].dt.year

# Create a 2x3 grid for subplots
fig, axes = plt.subplots(2, 3, figsize=(18, 10))

# Line plot for Release Speed over Time
sns.lineplot(data=scherzer_data, x='year', y='release_speed', hue='
   pitch_type', ax=axes[0, 0], palette=pitch_palette)
axes[0, 0].set_title('Release Speed over Time')
axes[0, 0].set_xlabel('Year')
axes[0, 0].set_ylabel('Release Speed')

# Line plot for PFX_X over Time
sns.lineplot(data=scherzer_data, x='year', y='pfx_x', hue='pitch_type',
    ax=axes[0, 1], palette=pitch_palette)
axes[0, 1].set_title('PFX_X over Time')
axes[0, 1].set_xlabel('Year')
axes[0, 1].set_ylabel('PFX_X')

# Line plot for PFX_Z over Time
sns.lineplot(data=scherzer_data, x='year', y='pfx_z', hue='pitch_type',
    ax=axes[0, 2], palette=pitch_palette)
axes[0, 2].set_title('PFX_Z over Time')
axes[0, 2].set_xlabel('Year')
axes[0, 2].set_ylabel('PFX_Z')

# Line plot for ax over Time
sns.lineplot(data=scherzer_data, x='year', y='ax', hue='pitch_type', ax
   =axes[1, 0], palette=pitch_palette)
axes[1, 0].set_title('ax over Time')
axes[1, 0].set_xlabel('Year')
axes[1, 0].set_ylabel('ax')

# Line plot for ay over Time
sns.lineplot(data=scherzer_data, x='year', y='ay', hue='pitch_type', ax
   =axes[1, 1], palette=pitch_palette)
axes[1, 1].set_title('ay over Time')
axes[1, 1].set_xlabel('Year')
axes[1, 1].set_ylabel('ay')

# Line plot for Release Spin Rate over Time
sns.lineplot(data=scherzer_data, x='year', y='release_spin_rate', hue='
   pitch_type', ax=axes[1, 2], palette=pitch_palette)
axes[1, 2].set_title('Release Spin Rate over Time')
axes[1, 2].set_xlabel('Year')
axes[1, 2].set_ylabel('Release Spin Rate')

# Adjust layout
plt.tight_layout()
plt.show()
```

```
# Define the list of variables
variables = ['release_speed', 'pfx_x', 'pfx_z', 'ax', 'ay', '
    release_spin_rate']

# Create a 2x3 grid of subplots
fig, axes = plt.subplots(2, 3, figsize=(20, 10))

# Flatten the axes array for easier iteration
axes = axes.flatten()

# Iterate over each variable and create scatter plots
for i, var in enumerate(variables):
    # Scatter plot for the current variable
    for pitch_type, color in pitch_palette.items():
        data_filtered = scherzer_data[scherzer_data['pitch_type'] ==
            pitch_type]
        sns.scatterplot(data=data_filtered, x='game_date', y=var, ax=
            axes[i], color=color, label=pitch_type)
    axes[i].set_title(f'{var} vs Game Date')
    axes[i].set_xlabel('Game Date')
    axes[i].set_ylabel(var)
    axes[i].legend(title='Pitch Type')

# Adjust layout
plt.tight_layout()
plt.show()

# Loop over each variable
for var in variables:
    # Filter data for the current variable
    var_data = scherzer_data[['game_date', var, 'year']]

    # Plot scatterplot with all points black
    plt.figure(figsize=(14, 7))
    sns.scatterplot(data=var_data, x='game_date', y=var, color='black',
        alpha=0.5)

    # Plot lineplot for average value per year with a brighter color
       scheme and wider line
    sns.lineplot(data=var_data, x='game_date', y=var, hue='year',
       estimator='mean', ci=None, lw=2, palette='bright')

    # Set title and labels
    plt.title(f'{var} Over Time')
    plt.xlabel('Date')
    plt.ylabel(var)
    plt.legend(title='Year')
    plt.show()

# Define the list of years
years = [2019, 2020, 2021, 2022, 2023]
```

```
# Calculate the number of rows and columns for subplots
num_cols = 3
num_rows = (len(years) + num_cols − 1) // num_cols

# Create subplots
fig, axes = plt.subplots(num_rows, num_cols, figsize=(15, 10))
axes = axes.flatten()

# Iterate over each year and create scatter plots
for i, year in enumerate(years):
    ax = axes[i]
    data = scherzer_data[scherzer_data[’year’] == year]
    sns.scatterplot(data=data, x=’pfx_x’, y=’pfx_z’, hue=’pitch_type’,
                    hue_order=[’FF’, ’SL’, ’CU’, ’CH’], ax=ax, palette=
                        pitch_palette)
    ax.set_title(’PFX_X and PFX_Z in ’ + str(year))
    ax.set(xlim=(−2, 2), ylim=(−2, 2))  # Adjust axis limits

# Adjust layout
plt.tight_layout()
plt.show()

# Calculate the number of rows and columns for subplots
num_cols = 3
num_rows = (len(years) + num_cols − 1) // num_cols

# Create subplots
fig, axes = plt.subplots(num_rows, num_cols, figsize=(15, 10))
axes = axes.flatten()

# Iterate over each year and create scatter plots
for i, year in enumerate(years):
    ax = axes[i]
    data = scherzer_data[scherzer_data[’year’] == year]
    sns.scatterplot(data=data, x=’ax’, y=’ay’, hue=’pitch_type’,
                    hue_order=[’FF’, ’SL’, ’CU’, ’CH’], ax=ax, palette=
                        pitch_palette)
    ax.set_title(’ax and ay in ’ + str(year))
    ax.set(xlim=(−25, 15), ylim=(0, 45))  # Adjust axis limits

# Adjust layout
plt.tight_layout()
plt.show()

”””# Jacob Degrom”””

# Gather Jacob Degrom’s pitching data for the desired five−year range
degrom_data = get_pitcher_data(’degrom’, ’jacob’, 2019, 2023)

# Display the first few rows of the DataFrame
print(”Jacob Degrom’s Pitching Data:”)
print(degrom_data.head())

# Filter by pitch type
pitch_types = ['FF', 'CU', 'SL', 'CH']
degrom_pitch = degrom_data[degrom_data['pitch_type'].isin(pitch_types)]

# Define the columns to include in the subset
subset = ['pitch_type', 'game_date', 'release_speed', 'effective_speed
    ', 'release_pos_x', 'release_pos_y', 'release_pos_z', 'pfx_x', '
    pfx_z', 'vx0', 'vy0', 'vz0', 'ax', 'ay', 'az', 'release_spin_rate',
    'spin_axis']

# Create the new subset by filtering the DataFrame
degrom_filtered = degrom_pitch[subset]

# Drop rows with missing values
degrom_dropna = degrom_filtered.dropna()

#final subset with primary vars of interest
new_subset = ['pitch_type', 'game_date','release_speed', 'pfx_x', '
    pfx_z', 'ax', 'ay', 'release_spin_rate']
degrom_data = degrom_dropna[new_subset]

# Check data dimensions
degrom_data.shape

"""# Jacob Degrom plots"""

# Convert game_date to datetime
degrom_data['game_date'] = pd.to_datetime(degrom_data['game_date'])

# Extract year
degrom_data['year'] = degrom_data['game_date'].dt.year

# Create a custom color palette for the pitch types
pitch_palette = {
    'FF': '#1f77b4',  # Blue
    'CH': '#ff7f0e',  # Orange
    'SL': '#2ca02c',  # Green
    'CU': '#d62728'   # Red
}

# Create a 2x3 grid for subplots
fig, axes = plt.subplots(2, 3, figsize=(18, 10))

# Line plot for Release Speed over Time
sns.lineplot(data=degrom_data, x='year', y='release_speed', hue='
    pitch_type', ax=axes[0, 0], palette=pitch_palette)
axes[0, 0].set_title('Release Speed over Time')
axes[0, 0].set_xlabel('Year')
axes[0, 0].set_ylabel('Release Speed')
```

```
# Line plot for PFX_X over Time
sns.lineplot(data=degrom_data, x='year', y='pfx_x', hue='pitch_type',
   ax=axes[0, 1], palette=pitch_palette)
axes[0, 1].set_title('PFX_X over Time')
axes[0, 1].set_xlabel('Year')
axes[0, 1].set_ylabel('PFX_X')

# Line plot for PFX_Z over Time
sns.lineplot(data=degrom_data, x='year', y='pfx_z', hue='pitch_type',
   ax=axes[0, 2], palette=pitch_palette)
axes[0, 2].set_title('PFX_Z over Time')
axes[0, 2].set_xlabel('Year')
axes[0, 2].set_ylabel('PFX_Z')

# Line plot for ax over Time
sns.lineplot(data=degrom_data, x='year', y='ax', hue='pitch_type', ax=
   axes[1, 0], palette=pitch_palette)
axes[1, 0].set_title('ax over Time')
axes[1, 0].set_xlabel('Year')
axes[1, 0].set_ylabel('ax')

# Line plot for ay over Time
sns.lineplot(data=degrom_data, x='year', y='ay', hue='pitch_type', ax=
   axes[1, 1], palette=pitch_palette)
axes[1, 1].set_title('ay over Time')
axes[1, 1].set_xlabel('Year')
axes[1, 1].set_ylabel('ay')

# Line plot for Release Spin Rate over Time
sns.lineplot(data=degrom_data, x='year', y='release_spin_rate', hue='
   pitch_type', ax=axes[1, 2], palette=pitch_palette)
axes[1, 2].set_title('Release Spin Rate over Time')
axes[1, 2].set_xlabel('Year')
axes[1, 2].set_ylabel('Release Spin Rate')

# Adjust layout
plt.tight_layout()
plt.show()

import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

# Define the list of variables
variables = ['release_speed', 'pfx_x', 'pfx_z', 'ax', 'ay', '
   release_spin_rate']

# Create a 2x3 grid of subplots
fig, axes = plt.subplots(2, 3, figsize=(20, 10))

# Flatten the axes array for easier iteration
axes = axes.flatten()
```

```
# Iterate over each variable and create scatter plots
for i, var in enumerate(variables):
    # Scatter plot for the current variable
    for pitch_type, color in pitch_palette.items():
        data_filtered = degrom_data[degrom_data['pitch_type'] ==
           pitch_type]
        sns.scatterplot(data=data_filtered, x='game_date', y=var, ax=
           axes[i], color=color, label=pitch_type)
    axes[i].set_title(f'{var} vs Game Date')
    axes[i].set_xlabel('Game Date')
    axes[i].set_ylabel(var)
    axes[i].legend(title='Pitch Type')

# Adjust layout
plt.tight_layout()
plt.show()

import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Variables to plot
variables = ['release_speed', 'pfx_x', 'pfx_z', 'ax', 'ay', '
   release_spin_rate']

# Custom color palette for the years
year_palette = {
    2019: 'blue',
    2020: 'orange',
    2021: 'green',
    2022: 'red',
    2023: 'purple'
}

# Loop over each variable
for var in variables:
    # Filter data for the current variable
    var_data = degrom_data[['game_date', var, 'year']]

    # Plot scatterplot with all points black
    plt.figure(figsize=(14, 7))
    sns.scatterplot(data=var_data, x='game_date', y=var, color='black',
        alpha=0.5)

    # Plot lineplot for average value per year with a custom color
       scheme
    sns.lineplot(data=var_data, x='game_date', y=var, hue='year',
       estimator='mean', ci=None, lw=2, palette=year_palette)

    # Set title and labels
    plt.title(f'{var} Over Time')
    plt.xlabel('Date')
    plt.ylabel(var)
    plt.legend(title='Year')
    plt.show()

import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Custom color palette
custom_palette = {
    'CH': 'blue',
    'FF': 'orange',
    'SL': 'green',
    'CU': 'red'
}

# Define the years
years = [2019, 2020, 2021, 2022, 2023]

# Determine the number of rows and columns for the subplots
num_cols = 3
num_rows = (len(years) + num_cols - 1) // num_cols

# Create subplots
fig, axes = plt.subplots(num_rows, num_cols, figsize=(15, 10))
axes = axes.flatten()

# Plot pfx_x and pfx_z for each year
for i, year in enumerate(years):
    ax = axes[i]
    data = degrom_data[degrom_data['year'] == year]
    sns.scatterplot(data=data, x='pfx_x', y='pfx_z', hue='pitch_type',
                    hue_order=['FF', 'SL', 'CU', 'CH'], palette=
                       custom_palette, ax=ax)
    ax.set_title('PFX_X and PFX_Z in ' + str(year))
    ax.set_xlabel('PFX_X')
    ax.set_ylabel('PFX_Z')
    ax.set(xlim=(-2, 2), ylim=(-2, 2.5))

# Remove any empty subplots
for j in range(len(years), len(axes)):
    fig.delaxes(axes[j])

# Adjust layout
plt.tight_layout()
plt.show()

# Custom color palette
custom_palette = {
    'CH': 'blue',
    'FF': 'orange',
    'SL': 'green',
    'CU': 'red'
}

# Define the years
years = [2019, 2020, 2021, 2022, 2023]

# Determine the number of rows and columns for the subplots
num_cols = 3
num_rows = (len(years) + num_cols - 1) // num_cols

# Create subplots
fig, axes = plt.subplots(num_rows, num_cols, figsize=(15, 10))
axes = axes.flatten()

# Plot ax and ay for each year
for i, year in enumerate(years):
    ax = axes[i]
    data = degrom_data[degrom_data['year'] == year]
    sns.scatterplot(data=data, x='ax', y='ay', hue='pitch_type',
                    hue_order=['FF', 'SL', 'CU', 'CH'], palette=
                       custom_palette, ax=ax)
    ax.set_title('ax and ay in ' + str(year))
    ax.set_xlabel('ax')
    ax.set_ylabel('ay')

# Remove any empty subplots
for j in range(len(years), len(axes)):
    fig.delaxes(axes[j])

# Adjust layout
plt.tight_layout()
plt.show()

"""# Comparison plots"""

import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Convert game_date to datetime
scherzer_data['game_date'] = pd.to_datetime(scherzer_data['game_date'])
degrom_data['game_date'] = pd.to_datetime(degrom_data['game_date'])

# Extract year
scherzer_data['year'] = scherzer_data['game_date'].dt.year
degrom_data['year'] = degrom_data['game_date'].dt.year

# Add a pitcher column to each dataset
scherzer_data['pitcher'] = 'Max Scherzer'
degrom_data['pitcher'] = 'Jacob deGrom'
```

```
# Combine the datasets
combined_data = pd.concat([scherzer_data, degrom_data])

# Define the custom color palette
custom_palette = {
    'CH': 'blue',
    'FF': 'orange',
    'SL': 'green',
    'CU': 'red'
}

# Plot Release Speed over Time
plt.figure(figsize=(14, 7))
sns.lineplot(data=combined_data, x='year', y='release_speed', hue='
   pitch_type', style='pitcher',
            dashes={'Max Scherzer': '', 'Jacob deGrom': (5, 5)},
               palette=custom_palette)
plt.title('Release Speed over Time')
plt.xlabel('Year')
plt.ylabel('Release Speed')
plt.legend(title='Pitch Type and Pitcher')
plt.show()

# Plot PFX_X over Time
plt.figure(figsize=(14, 7))
sns.lineplot(data=combined_data, x='year', y='pfx_x', hue='pitch_type',
    style='pitcher',
            dashes={'Max Scherzer': '', 'Jacob deGrom': (5, 5)},
               palette=custom_palette)
plt.title('PFX_X over Time')
plt.xlabel('Year')
plt.ylabel('PFX_X')
plt.legend(title='Pitch Type and Pitcher')
plt.show()

# Plot PFX_Z over Time
plt.figure(figsize=(14, 7))
sns.lineplot(data=combined_data, x='year', y='pfx_z', hue='pitch_type',
    style='pitcher',
            dashes={'Max Scherzer': '', 'Jacob deGrom': (5, 5)},
               palette=custom_palette)
plt.title('PFX_Z over Time')
plt.xlabel('Year')
plt.ylabel('PFX_Z')
plt.legend(title='Pitch Type and Pitcher')
plt.show()

# Plot AX over Time
plt.figure(figsize=(14, 7))
sns.lineplot(data=combined_data, x='year', y='ax', hue='pitch_type',
   style='pitcher',
            dashes={'Max Scherzer': '', 'Jacob deGrom': (5, 5)},
                palette=custom_palette)
plt.title('Acceleration (x) over Time')
plt.xlabel('Year')
plt.ylabel('Acceleration (x)')
plt.legend(title='Pitch Type and Pitcher')
plt.show()

# Plot AY over Time
plt.figure(figsize=(14, 7))
sns.lineplot(data=combined_data, x='year', y='ay', hue='pitch_type',
    style='pitcher',
            dashes={'Max Scherzer': '', 'Jacob deGrom': (5, 5)},
                palette=custom_palette)
plt.title('Acceleration (y) over Time')
plt.xlabel('Year')
plt.ylabel('Acceleration (y)')
plt.legend(title='Pitch Type and Pitcher')
plt.show()

# Plot Release Spin Rate over Time
plt.figure(figsize=(14, 7))
sns.lineplot(data=combined_data, x='year', y='release_spin_rate', hue='
    pitch_type', style='pitcher',
            dashes={'Max Scherzer': '', 'Jacob deGrom': (5, 5)},
                palette=custom_palette)
plt.title('Release Spin Rate over Time')
plt.xlabel('Year')
plt.ylabel('Release Spin Rate')
plt.legend(title='Pitch Type and Pitcher')
plt.show()
```