

Notebook Contents

- Imports
- Data
 - Pitcher Overview
 - General Pitch Data
 - Pitch Usage
 - Pitch Usage by Batter Handedness
 - Velocity by Pitch Type
 - Pitch Velocity by Pitch Number
 - Pitcher Stuff
 - Spin Rate by Pitch Type
 - Velocity & Spin Rate
 - Spin Axis
 - Spin Efficiency
 - HB & VB Axis
 - HB & HB Due to Magnus Force
 - Release Position
 - Release Extension
 - Avg. Bauer Units by Pitch Type
 - Count Breakdown
 - Pitch Usage by Count
 - Heatmaps
 - Pitch Location by Pitch Type
 - Quality Contact Heatmap - All Pitches
 - 4-Seam Heatmaps
 - Slider Heatmaps
 - Changeup Heatmaps
 - Curveball Heatmaps
 - Cutter Heatmaps

Imports

```
In [1]: import math
import matplotlib.patches as mpatches
from matplotlib.patches import Rectangle
from matplotlib.ticker import PercentFormatter

import warnings
warnings.filterwarnings('ignore')
```

Data

```
In [2]: scherzer = pd.read_csv('../data/max-scherzer.csv')
scherzer.drop(columns = ['Unnamed: 0'], inplace = True)

# Font Dictionary
font_title = {
    'size': 14,
    'weight': 'bold',
    'verticalalignment': 'center_baseline',
    'horizontalalignment': 'center'
}

pd.set_option('max_columns', None)
print(scherzer.shape)
scherzer.head(2)
```

	pitch_type	game_date	release_speed	release_pos_x	release_pos_z	player_name	batter	pitcher	events	description	zor
0	CU	2021-05-30	78.4	-3.37	5.41	Scherzer, Max	592885	453286	strikeout	called_strike	
1	FF	2021-05-30	94.2	-3.17	5.54	Scherzer, Max	592885	453286		NaN	swinging_strike

```
In [3]: gen_pd = scherzer[['pitch_type', 'release_speed', 'release_spin_rate',
                        'true_spin', 'spin_eff', 'phi', 'pfx_z', 'pfx_x',
                        'is_strike', 'release_pos_x', 'release_pos_z', 'batter_units']]

col_dict = {
    'release_speed': 'velo', 'release_spin_rate': 'spin', 'phi': 'spin_axis', 'pfx_z': 'hb', 'pfx_x': 'vb',
    'is_strike': 'strike', 'release_pos_x': 'r_height', 'release_pos_z': 'r_side'
}
gen_pd.rename(columns = col_dict, inplace = True)
```

```
In [4]: # scherzer.pitch_type.value_counts(normalize=True)
r_scherzer = scherzer.loc[scherzer['stand'] == 'R']
l_scherzer = scherzer.loc[scherzer['stand'] == 'L']
# all batters
ff = scherzer.loc[scherzer['pitch_type'] == 'FF']
sl = scherzer.loc[scherzer['pitch_type'] == 'SL']
ch = scherzer.loc[scherzer['pitch_type'] == 'CH']
cu = scherzer.loc[scherzer['pitch_type'] == 'CU']
fc = scherzer.loc[scherzer['pitch_type'] == 'FC']
# RHH
r_ff = r_scherzer.loc[r_scherzer['pitch_type'] == 'FF']
r_sl = r_scherzer.loc[r_scherzer['pitch_type'] == 'SL']
r_ch = r_scherzer.loc[r_scherzer['pitch_type'] == 'CH']
r_cu = r_scherzer.loc[r_scherzer['pitch_type'] == 'CU']
r_fc = r_scherzer.loc[r_scherzer['pitch_type'] == 'FC']
# LHH
l_ff = l_scherzer.loc[l_scherzer['pitch_type'] == 'FF']
l_sl = l_scherzer.loc[l_scherzer['pitch_type'] == 'SL']
l_ch = l_scherzer.loc[l_scherzer['pitch_type'] == 'CH']
l_cu = l_scherzer.loc[l_scherzer['pitch_type'] == 'CU']
l_fc = l_scherzer.loc[l_scherzer['pitch_type'] == 'FC']
order = ['ff', 'sl', 'ch', 'cu', 'fc']
```

```
In [5]: ff_tilt = ff['phi'].mean()
sl_tilt = sl['phi'].mean()
ch_tilt = ch['phi'].mean()
cu_tilt = cu['phi'].mean()
fc_tilt = fc['phi'].mean()
```

Pitcher Overview

General Pitch Data

```
In [6]: gen_pd.groupby(['pitch_type'], sort = False).mean()
```

pitch_type	velo	spin	true_spin	spin_eff	spin_axis	hb	vb	strike	r_side	r_height	bauer
CU	76.863538	2660.692029	1203.160550	0.454450	411.534031	-7939495	-13.447798	0.613718	-3.232708	5.609061	34
FF	94.298480	2435.912614	1687.460687	0.700331	215.693182	16.034134	10.608974	0.684850	-3.220585	5.560691	2
FC	90.224219	2474.929688	539.441931	0.220207	185.703448	8.931094	1.021406	0.687500	-3.338398	5.400547	21
CH	84.980048	1446.159145	111.533281	0.778594	261.536316	3.952019	15.156200	0.577197	-3.292233	5.437577	17
SL	85.889095	2273.237560	291.186532	0.131012	168.500000	4.634483	-2.915281	0.691470	-3.332087	5.452777	26

4-Seam: Three quarter slot: 2:00 - 2:00 R spin tilt, FB movement profile has similar horizontal & vertical break

Slider: Sweeping slider (side spin): 8:00 - 10:00 R, spin tilt, Decent amount of HB and near 0 VB

Changeup: Circle change: 12:00 - 3 R, 9:00 - 12:00 L spin tilt, Has decent HB & VB, Spin rate: 1700 - 1800 rpm, Tilt: 2:00 - 3:00 R, Spin efficiency: ~65-75%, Arm slot: 3/4

Curveball: Slurve: 7:00 - 8:00 R, 4:00 - 5:00 L spin axis, More negative VB and a little more HB than 12-6 CB

Cutter: Cutter (side + back spin): 10:45 - 12:00 R, 12:00 - 1:30 L spin tilt, Positive VB & Some HB Cutter: 40 - 60%

Pitch Usage

```
In [7]: plt.figure(figsize = (8, 6))

dist = round(scherzer.pitch_type.value_counts(normalize = True), 2)
color = sns.color_palette('coolwarm_r')
plt.pie(dist, labels = order, colors = color, autopct = '%.0f%%')
plt.title('Distribution of Pitch Types - Max Scherzer', fontdict = font_title, pad = 15);
```

Pitch Usage by Batter Handedness

```
In [8]: blue = '#002072'
red = '#D62728'

fig, axs = plt.subplots(1, 2, figsize = (20, 6))
fig.suptitle('Pitch Usage by Batter Handedness', fontsize = 16, fontweight = 'bold')
dist_r = r_scherzer.pitch_type.sort_values(ascending = False)
axs[0].hist(dist_r, weights = np.ones(len(dist_r)) / len(dist_r), color = red)
axs[0].yaxis.set_major_formatter(PercentFormatter(1))
axs[0].set_title('Distribution of Pitch Types - RHH', fontdict = font_title, pad = 15)
dist_l = l_scherzer.pitch_type.sort_values(ascending = False)
axs[1].hist(dist_l, weights = np.ones(len(dist_l)) / len(dist_l), color = blue)
axs[1].set_title('Distribution of Pitch Types - LHH', fontdict = font_title, pad = 15);
```

LHH: Heavy 4-Seam / Change mix Cutter / Curve

Velocity by Pitch Type

```
In [9]: plt.figure(figsize = (8, 6))

sns.kdeplot(data = scherzer, x = 'release_speed', shade = 'fill', hue = 'pitch_type',
            hue_order = order, palette = 'tab10')
plt.title('Distribution of Velocity by Pitch Type - Max Scherzer', fontdict = font_title, pad = 12);
```

4-Seam: Sits 94-95 mph (league avg: 92.3 mph)

Slider: Sits 85-87 mph (league avg: 85 mph)

Changeup: Sits 84-86 mph (league avg: 85 mph (8 - 12 mph difference from FB))

Curveball: Sits 76-78 mph (league avg: 77 mph (11 - 18 mph slower compared to FB))

Cutter: Sits 90-91 mph (league avg: 88 mph)

Pitch Velocity by Pitch Number

```
In [10]: plt.figure(figsize = (8, 6))

sns.lineplot(data = scherzer, x = 'inning', y = 'release_speed', hue = 'pitch_type',
            hue_order = order, palette = 'tab10')
plt.title('Pitch Velocity by Pitch Number', fontdict = font_title, pad = 15);
```

Doesn't have a velo dip deep into games

Pitcher Stuff

Spin Rate by Pitch Type

```
In [11]: g = sns.FacetGrid(scherzer, row = 'pitch_type', hue = 'pitch_type',
                      height = 2, aspect = 4, hue_order = order)
g.map(sns.kdeplot, 'release_spin_rate', palette = 'tab10', hue_order = order);
```

4-Seam: Above avg spin (league avg: ~2200 rpm)

Slider: Below avg spin (league avg: ~2450 rpm)

Changeup: Avg-Below avg, spin (league avg: ~1790 rpm (range from 1200 - 2400))

Curveball: Above avg spin (league avg: ~2500 rpm)

Cutter: Above avg spin (league avg: ~2,185 rpm)

Spin Axis

```
In [12]: ax = plt.figure(figsize = (6, 6))

ax = plt.subplot(polar = True, theta_direction = 'l')
ax.plot(math.radians(ff_tilt), 1, color = 'blue', marker = 'o', label = '4-Seam')
ax.plot(math.radians(sl_tilt), 1, color = 'orange', marker = 'o', label = 'Slider')
ax.plot(math.radians(ch_tilt), 1, color = 'green', marker = 'o', label = 'Changeup')
ax.plot(math.radians(cu_tilt), 1, color = 'red', marker = 'o', label = 'Curveball')
ax.plot(math.radians(fc_tilt), 1, color = 'purple', marker = 'o', label = 'Cutter')
ticks = ['6:00', '7:30', '9:00', '10:30', '12:00', '1:30', '3:00', '4:30']
ax.set_yticklabels(ticks); ax.legend(locat=scherzer(1.4, .64)); ax.set_theta_zero_location('R')
ax.set_title('Spin Axis', fontdict = font_title, pad = 15);
```

Pairs: 4-Seam / Curveball, Slider / Cutter

4-Seam: Three quarter slot: 1:00 - 2:00 spin tilt

Slider: Sweeping slider (side spin): 8:00 - 10:30 spin tilt

Changeup: Circle change: 2:00 - 3:00 spin tilt Arm slot: 3/4

Curveball: Slurve: 7:00 - 8:00 spin axis

Cutter: Cutter (side + back spin): 10:45 - 12:00

Spin Efficiency

```
In [13]: sns.catplot(data = scherzer, x = 'spin_eff', y = 'pitch_type', kind = 'violin', aspect = 1.5, palette = 'tab10')
plt.title('Spin Efficiency', fontdict = font_title, pad = 15);
```

4-Seam: Spin efficiency: ~80%

Slider: Sweeping slider: 20 - 40%

Changeup: Spin efficiency: ~65-75%

Curveball: Curveballs thrown with low spin efficiency (below 70% efficiency) Movement profile gets closer to (0, 0) Higher spin efficiency Movement profile gets further from (0, 0)

Cutter: Spin efficiency: 40 - 60%

Horizontal & Vertical Break Axis

```
In [14]: plt.figure(figsize = (8, 6))

sns.scatterplot(data = scherzer, x = 'pfx_x', y = 'pfx_z', hue = 'pitch_type',
               hue_order = order, palette = 'tab10')
plt.xlim(-30, 30), plt.ylim(-30, 30)
plt.xlabel(0, color = 'black'), plt.ylabel(0, color = 'black')
axs[1][0].set_title('4-Seam Usage in Counts', fontsize = 14, pad = 15)
axs[1][0].hist(ch['pitch_count'].sort_values(ascending = True), color = blue, edgecolor = red, linewidth = 2)
axs[1][1].hist(ch['pitch_count'].sort_values(ascending = True), color = blue, edgecolor = red, linewidth = 2)
axs[1][2].hist(ch['pitch_count'].sort_values(ascending = True), color = blue, edgecolor = red, linewidth = 2)
axs[2][0].set_title('Changeup Usage in Counts', fontsize = 14, pad = 15)
axs[2][0].hist(cu['pitch_count'].sort_values(ascending = True), color = blue, edgecolor = red, linewidth = 2)
axs[2][1].hist(cu['pitch_count'].sort_values(ascending = True), color = blue, edgecolor = red, linewidth = 2)
axs[2][2].set_title('Cutter Usage in Counts', fontsize = 14, pad = 15)
plt.tight_layout();
```

4-Seam: FB movement profile has similar HB & VB

Slider: Decent amount of HB and near 0 VB

Changeup: Has decent HB & VB

Curveball: More negative VB and a little more HB than 12-6 CB

Cutter: Positive VB & Some HB

Horizontal & Vertical Break Due to Magnus Force Axis

```
In [15]: plt.figure(figsize = (8, 6))

sns.scatterplot(data = scherzer, x = 'Mx', y = 'Mz', hue = 'pitch_type',
               hue_order = order, palette = 'tab10')
plt.xlim(-30, 30), plt.ylim(-30, 30)
plt.xlabel(0, color = 'black'), plt.ylabel(0, color = 'black')
axs[1][0].set_title('4-Seam Usage in Counts', fontsize = 14, pad = 15)
axs[1][0].hist(ch['pitch_count'].sort_values(ascending = True), color = blue, edgecolor = red, linewidth = 2)
axs[1][1].hist(ch['pitch_count'].sort_values(ascending = True), color = blue, edgecolor = red, linewidth = 2)
axs[1][2].set_title('Changeup Usage in Counts', fontsize = 14, pad = 15)
axs[2][0].set_title('Changeup Usage in Counts', fontsize = 14, pad = 15)
axs[2][0].hist(cu['pitch_count'].sort_values(ascending = True), color = blue, edgecolor = red, linewidth = 2)
axs[2][1].hist(cu['pitch_count'].sort_values(ascending = True), color = blue, edgecolor = red, linewidth = 2)
axs[2][2].set_title('Cutter Usage in Counts', fontsize = 14, pad = 15)
plt.tight_layout();
```

4-Seam: FB movement profile has similar HB & VB

Slider: Decent amount of HB and near 0 VB

Changeup: Has decent HB & VB

Curveball: More negative VB and a little more HB than 12-6 CB

Cutter: Positive VB & Some HB

Release Postion

```
In [16]: plt.figure(figsize = (8, 6))

sns.scatterplot(data = scherzer, x = 'release_pos_x', y = 'release_pos_z', hue = 'pitch_type',
               hue_order = order, palette = 'tab10')
plt.xlim(-5, 5), plt.ylim(-0.25, 0.25)
plt.xlabel(0, color = 'black'), plt.ylabel(0, color = 'black')
left, bottom, width, height = (-.83, 1.5, 1.66, 1.82), fill = False, color = 'black', linewidth = 2)
rect = mpatches.Rectangle((-0.83, 1.59), 1.66, 1.82, fill = False, color = 'black', linewidth = 2)
plt.gca().add_patch(rect)
axs[1][0].set_title('Release Position from Hitter Perspective', fontdict = font_title, pad = 15);
```

4-Seam: Above avg Bauer Units (league avg: 24)

Slider: Below avg Bauer Units (league avg: 28-30)

Changeup: N/A (league avg: N/A)

Curveball: Above avg Bauer Units (league avg: 28-30)

Cutter: Above avg Bauer Units (league avg: 24-25)

Count Breakdown

Pitch Usage by Count

```
In [20]: fig, axs = plt.subplots(3, 2, figsize = (15, 15))
fig.suptitle('Pitch Usage by Count', fontsize = 16, fontweight = 'bold')
plt.subplots_adjust(wspace = 0.25, hspace = 0.25)
plt.setp(axs[:, 0], xlabel = 'Count')
plt.setp(axs[:, 1], ylabel = 'Frequency')
axs[0][0].hist(ff['pitch_count'].sort_values(ascending = True), color = blue, edgecolor = red, linewidth = 2)
axs[0][1].hist(ff['pitch_count'].sort_values(ascending = True), color = blue, edgecolor = red, linewidth = 2)
axs[1][0].set_title('Slider Usage in Counts', fontsize = 14, pad = 15)
axs[1][0].hist(ch['pitch_count'].sort_values(ascending = True), color = blue, edgecolor = red, linewidth = 2)
axs[1][1].set_title('Slider Usage in Counts', fontsize = 14, pad = 15)
axs[2][0].set_title('Changeup Usage in Counts', fontsize = 14, pad = 15)
axs[2][0].hist(cu['pitch_count'].sort_values(ascending = True), color = blue, edgecolor = red, linewidth = 2)
axs[2][1].set_title('Changeup Usage in Counts', fontsize = 14, pad = 15)
axs[3][0].set_title('Curveball Usage in Counts', fontsize = 14, pad = 15)
axs[3][0].hist(sl['pitch_count'].sort_values(ascending = True), color = blue, edgecolor = red, linewidth = 2)
axs[3][1].set_title('Curveball Usage in Counts', fontsize = 14, pad = 15)
axs[4][0].set_title('Cutter Usage in Counts', fontsize = 14, pad = 15)
axs[4][0].hist(fc['pitch_count'].sort_values(ascending = True), color = blue, edgecolor = red, linewidth = 2)
axs[4][1].set_title('Cutter Usage in Counts', fontsize = 14, pad = 15)
plt.tight_layout();
```

• 0-0 Count: Mally 4-Seam, Slider & Curve

• Heavy Changeup and Curve in plus counts mix some Cutter

• Slider in even counts

Heatmaps

*All From Hitters' Perspective

Pitch Location by Pitch Type

```
In [21]: fig, axs = plt.subplots(3, 2, figsize = (14, 18), sharex = True, sharey = True)
fig.suptitle('Pitch Location by Pitch Type', fontsize = 16, fontweight = 'bold')
plt.subplots_adjust(wspace = -0.5, xmargin = 3.5)
plt.axis(ymin = -1.5, ymax = 5.5)
sns.kdeplot(ax = axs[0][0], data=ff, x='plate_x', y='plate_z', fill=True, hue='is_strike', palette='coolwarm')
axs[0][0].add_patch(Rectangle((-0.83, 1.59), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axs[0][0].set_title('4-Seam Location vs All Hitters', fontsize = 14, pad = 15)
sns.kdeplot(ax = axs[1][0], data=ch, x='plate_x', y='plate_z', fill=True, hue='is_strike', palette='coolwarm')
axs[1][0].add_patch(Rectangle((-0.83, 1.59), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axs[1][0].set_title('Slider Location vs All Hitters', fontsize = 14, pad = 15)
sns.kdeplot(ax = axs[2][0], data=cu, x='plate_x', y='plate_z', fill=True, hue='is_strike', palette='coolwarm')
axs[2][0].add_patch(Rectangle((-0.83, 1.59), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axs[2][0].set_title('Changeup Location vs All Hitters', fontsize = 14, pad = 15)
sns.kdeplot(ax = axs[3][0], data=sl, x='plate_x', y='plate_z', fill=True, hue='is_strike', palette='coolwarm')
axs[3][0].add_patch(Rectangle((-0.83, 1.59), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axs[3][0].set_title('Curveball Location vs All Hitters', fontsize = 14, pad = 15)
sns.kdeplot(ax = axs[4][0], data=fc, x='plate_x', y='plate_z', fill=True, hue='is_strike', palette='coolwarm')
axs[4][0].add_patch(Rectangle((-0.83, 1.59), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axs[4][0].set_title('Cutter Location vs All Hitters', fontsize = 14, pad = 15);
```


