Max Scherzer

Notebook Contents

- Imports
- Data
  - Index DataFrame for Each Pitch Type
- Pitcher Overview
  - General Pitch Data
  - Pitch Usage
  - Pitch Usage by Batter Handedness
  - Velocity by Pitch Type
  - Spin Axis by Pitch Type
  - Spin Efficiency
  - Horizontal & Vertical Break Axis
  - Release Position
  - Release Extension
  - Velocity & Spin Rate
  - Bauer Units
  - Count Breakdown
- Count Breakdown
  - Pitch Usage by Count
- Heatmaps
  - Pitch Location by Pitch Type
  - Pitch Usage and Count by Pitch Type
  - Count Heatmaps
  - Pitch Usage Heatmaps
  - Changeup Heatmaps
  - Cutter Heatmaps
  - Curveball Heatmaps

Imports

```
import pandas as pd
import matplotlib.pyplot as plt, matplotlib
from matplotlib.ticker import (MultipleLocator)
from matplotlib.patches import Patch
from matplotlib.lines import Line2D
%matplotlib inline
%config InlineBackend.figure_format = 'retina'
```

Data

```
df = pd.read_csv('./data/max_scherzer_statcast_2021.csv')
df['game_date'] = pd.to_datetime(df['game_date'], infer_datetime_format = True)
df.shape, df.dtypes
```

```
(2804, 92)
pitch_type                 object
game_date          datetime64[ns]
release_speed              float64
release_pos_x              float64
release_pos_z              float64
                        ...
fielder_7                  float64
fielder_8                  float64
fielder_9                  float64
release_extension          float64
sv_id                       object
Length: 92, dtype: object
```

```
df[['pitch_type', 'game_date', 'release_speed', 'release_pos_x', 'release_pos_z', 'pfx_x', 'pfx_z', 'plate_x', 'plate_z', 'vx0', 'vy0', 'vz0', 'ax', 'ay', 'az', 'sz_top', 'sz_bot', 'effective_speed', 'release_spin_rate', 'spin_axis', 'pitcher', 'events', 'description', 'zone', 'des', 'game_type', 'stand', 'p_throws', 'home_team', 'away_team', 'type', 'bb_type', 'balls', 'strikes', 'pitch_name', 'at_bat_number', 'pitch_number', 'pitcher_at_bat_number', 'pitch_number_appearance', 'arm_angle', 'vx0', 'vy0', 'vz0', 'ax', 'ay', 'az', 'sz_top', 'sz_bot', 'hit_location', 'bb_type', 'on_3b', 'on_2b', 'on_1b', 'outs_when_up', 'inning', 'inning_topbot']].sample(1, random_state = 7)
```

	pitch_type	game_date	release_speed	release_pos_x	release_pos_z	pfx_x	pfx_z	plate_x	plate_z	vx0	vy0	vz0	ax	ay	az	sz_top	sz_bot	effective_speed	release_spin_rate	spin_axis	pitcher	events	description	zone	des	game_type	stand	p_throws	home_team	away_team	type	bb_type	balls	strikes	pitch_name	at_bat_number	pitch_number	pitcher_at_bat_number	pitch_number_appearance	arm_angle	vx0	vy0	vz0	ax	ay	az	sz_top	sz_bot	hit_location	bb_type	on_3b	on_2b	on_1b	outs_when_up	inning	inning_topbot
0	FF	2021-04-21	92.9	-2.1	5.9	-1.4	1.5	-0.4	2.4	5.6	-138.4	-4.6	22.8	30.8	4.7	3.4	1.6	93.4	2556	203.0	592332	NaN	swinging_strike	13	swinging_strike	R	R	R	NYM	WSH	strike	NaN	0	2	Four-Seam Fastball	4	6	1	6	5.0	5.6	-138.4	-4.6	22.8	30.8	4.7	3.4	1.6	NaN	NaN	NaN	NaN	NaN	2	Top

Index DataFrame for Get Pitch Types

```
df[['pitch_type', 'pitch_name']].sort_values(['pitch_type', 'pitch_name'], ignore_index = True).drop_duplicates(ignore_index = True)
```

	pitch_type	pitch_name
0	CU	Curveball
1	FC	Cutter
2	FF	Four-Seam Fastball
3	FS	Splitter
4	SI	Sinker
5	SL	Slider
6	ST	Sweeper
7	SV	Slow Curve

Pitcher Overview

General Pitch Data

```
df[['pitch_type', 'release_speed', 'pfx_x', 'pfx_z', 'plate_x', 'plate_z', 'vx0', 'vy0', 'vz0', 'ax', 'ay', 'az', 'sz_top', 'sz_bot', 'effective_speed', 'release_spin_rate', 'spin_axis']].describe.T.round(2)
```

	count	mean	std	min	25%	50%	75%	max
release_speed	2804.0	92.37	2.57	78.0	91.0	92.9	94.4	99.1
pfx_x	2804.0	-0.14	0.67	-2.42	-0.62	-0.14	0.34	2.42
pfx_z	2804.0	1.51	0.49	-0.42	1.19	1.54	1.87	3.29
plate_x	2804.0	-0.02	0.61	-2.29	-0.42	-0.04	0.37	2.29
plate_z	2804.0	2.42	0.57	0.52	2.04	2.42	2.78	4.65
vx0	2804.0	5.57	5.61	-138.4	5.4	5.6	5.8	134.0
vy0	2804.0	-138.41	5.58	-154.0	-140.1	-138.4	-136.7	-96.0
vz0	2804.0	-4.59	3.41	-22.8	-6.7	-4.6	-2.4	19.0
ax	2804.0	30.79	2.46	13.2	29.4	30.8	32.2	41.2
ay	2804.0	3.41	1.59	-3.4	2.4	3.4	4.4	9.9
az	2804.0	1.61	0.27	0.7	1.4	1.6	1.8	2.5
sz_top	2804.0	3.40	0.22	2.6	3.3	3.4	3.5	4.1
sz_bot	2804.0	1.69	0.09	1.4	1.6	1.7	1.8	2.0
effective_speed	2804.0	92.16	2.56	78.1	90.8	92.7	94.2	98.8
release_spin_rate	2804.0	2556.00	203.00	1316.0	2432.0	2556.0	2683.0	3337.0
spin_axis	2804.0	203.00	101.00	1.0	197.0	218.0	232.0	359.0

Changeup: Three quarter arm (100 - 2318 RPM avg), 18 movement profile has similar horizontal & vertical break

Slider: Sweeping slider with sweep (2200 - 2616 RPM avg), 84 mph average velocity

Curveball: Slow curve (1500 - 2318 RPM avg), 78 movement profile has similar horizontal & vertical break

Cutter: Cutter (slide / cleanly) (2300 - 2616 RPM avg), 88 mph average velocity

Four-Seam Fastball: Four-seam fastball (2400 - 2616 RPM avg), 94 mph average velocity

Pitch Usage

```
df[['pitch_type', 'release_speed']].groupby('pitch_type', as_index = False).agg({'release_speed': ['count', 'mean']}).round(2).sort_values(('release_speed', 'count'), ignore_index = True)
```

	pitch_type	release_speed	
		count	mean
0	CU	172	78.0
1	FC	280	88.0
2	FF	1344	94.0
3	FS	1	78.1
4	SI	1	91.0
5	SL	1	84.0
6	ST	1	84.0
7	SV	1	78.0

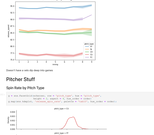
Note: 4-seam fastball into extra Orange

Left: Hole in Batter's Strike zone Cutter

Right: Hole in Batter's Strike zone Curve

Velocity by Pitch Type

```
df['game_date'] = df['game_date']
sns.set(rc = {'figure.figsize': (7, 5)})
sns.set(style = "whitegrid", palette = "muted")
sns.boxplot(data = df[df['release_speed' > 70][['game_date', 'release_speed', 'pitch_name']].sort_values(['pitch_name', 'game_date'], ignore_index = True).sample(frac = 1.0, random_state = 7).round(2).reset_index(drop = True)
plt.suptitle('Distribution of Velocity by Pitch Type', fontsize = 18)
plt.title('Max Scherzer', fontsize = 14)
plt.xlabel('Release Speed', fontsize = 12)
plt.ylabel('Pitch Name', fontsize = 12)
plt.show()
```



Note: 2 seam fastball has some Orange

Left: Hole in Batter's Strike zone Cutter

Right: Hole in Batter's Strike zone Curve

Velocity by Pitch Time

```
df['game_date'] = df['game_date']
sns.set(rc = {'figure.figsize': (7, 5)})
sns.set(style = "whitegrid", palette = "muted")
sns.boxplot(data = df[df['release_speed' > 70][['game_date', 'release_speed', 'pitch_name']].sort_values(['game_date', 'pitch_name'], ignore_index = True).sample(frac = 1.0, random_state = 7).round(2).reset_index(drop = True)
plt.suptitle('Pitch Velocity by Inning', fontsize = 18)
plt.title('Max Scherzer', fontsize = 14)
plt.xlabel('Release Speed', fontsize = 12)
plt.ylabel('Game Date', fontsize = 12)
plt.show()
```

Doesn't have a lot+ day-to-day his games

Pitcher Stuff

Spin Axis by Pitch Type

```
df.sort_values(['release_speed', 'game_date'], ignore_index = True).groupby('pitch_type').sample(1, random_state = 7).round(2).reset_index(drop = True)[['pitch_type', 'game_date', 'release_speed', 'release_pos_x', 'release_pos_z', 'spin_axis']]
```

	pitch_type	game_date	release_speed	release_pos_x	release_pos_z	spin_axis
0	CU	2021-09-24	78.0	-2.3	5.8	197.0
1	FC	2021-04-21	88.0	-2.1	5.9	232.0
2	FF	2021-09-24	94.0	-2.1	5.9	218.0
3	FS	2021-04-21	78.1	-2.1	5.9	1.0
4	SI	2021-09-24	91.0	-2.3	5.8	359.0
5	SL	2021-09-24	84.0	-2.3	5.8	232.0
6	ST	2021-09-24	84.0	-2.3	5.8	232.0
7	SV	2021-09-24	78.0	-2.3	5.8	197.0

Fastball: Cut-fastball thrown with low spin efficiency (below 70% efficiency) Movement profile goes down to (0.3) higher spin efficiency (fastball) profile goes further down to (1)

Cutter: Cutter (slide / cleanly) (2300 - 2616 rpm) with 88 mph avg

Curveball: Slow curve (1500 - 2318 rpm) with 78 mph avg

Slider: Sweeping slider (2200 - 2616 rpm) with 84 mph avg

Changeup: Three quarter arm (100 - 2318 rpm) with 78 mph avg

Sinker: Sinker (2300 - 2616 rpm) with 91 mph avg

Sweeper: Sweeping slider with sweep (2200 - 2616 rpm) with 84 mph avg

Slow Curve: Slow curve (1500 - 2318 rpm) with 78 mph avg

Spin Efficiency

```
df['game_date'] = df['game_date']
sns.set(rc = {'figure.figsize': (7, 5)})
sns.set(style = "whitegrid", palette = "muted")
sns.boxplot(data = df[df['release_spin_rate' > 1000][['game_date', 'release_spin_rate', 'pitch_name']].sort_values(['pitch_name', 'game_date'], ignore_index = True).sample(frac = 1.0, random_state = 7).round(2).reset_index(drop = True)
plt.suptitle('Spin Efficiency', fontsize = 18)
plt.title('Max Scherzer', fontsize = 14)
plt.xlabel('Release Spin Rate', fontsize = 12)
plt.ylabel('Pitch Name', fontsize = 12)
plt.show
```

Fastball: Cut-fastball thrown with low spin efficiency (below 70% efficiency) Movement profile goes down to (0.3) higher spin efficiency (fastball) profile goes further down to (1)

Slow Curve: Slow curve (1500 - 2318 rpm) with 78 mph avg

Sinker: Sinker (2300 - 2616 rpm) with 91 mph avg

Cutter: Cutter (slide / cleanly) (2300 - 2616 rpm) with 88 mph avg

Slider: Sweeping slider (2200 - 2616 rpm) with 84 mph avg

Sweeper: Sweeping slider with sweep (2200 - 2616 rpm) with 84 mph avg

Changeup: Three quarter arm (100 - 2318 rpm) with 78 mph avg

Curveball: Slow curve (1500 - 2318 rpm) with 78 mph avg

Horizontal & Vertical Break Axis

```
df['game_date'] = df['game_date']
sns.set(rc = {'figure.figsize': (7, 5)})
sns.set(style = "whitegrid", palette = "muted")
sns.boxplot(data = df[df['release_spin_rate' > 1000][['game_date', 'release_spin_rate', 'pitch_name']].sort_values(['pitch_name', 'game_date'], ignore_index = True).sample(frac = 1.0, random_state = 7).round(2).reset_index(drop = True)
plt.suptitle('Spin Efficiency', fontsize = 18)
plt.title('Max Scherzer', fontsize = 14)
plt.xlabel('Release Spin Rate', fontsize = 12)
plt.ylabel('Pitch Name', fontsize = 12)
plt.show
```

Fastball: Cut-fastball thrown with low spin efficiency (below 70% efficiency) Movement profile goes down to (0.3) higher spin efficiency (fastball) profile goes further down to (1)

Slow Curve: Slow curve (1500 - 2318 rpm) with 78 mph avg

Sinker: Sinker (2300 - 2616 rpm) with 91 mph avg

Cutter: Cutter (slide / cleanly) (2300 - 2616 rpm) with 88 mph avg

Slider: Sweeping slider (2200 - 2616 rpm) with 84 mph avg

Sweeper: Sweeping slider with sweep (2200 - 2616 rpm) with 84 mph avg

Changeup: Three quarter arm (100 - 2318 rpm) with 78 mph avg

Curveball: Slow curve (1500 - 2318 rpm) with 78 mph avg

Horizontal & Vertical Break Due to Magnus Force Axis

```
df['game_date'] = df['game_date']
sns.set(rc = {'figure.figsize': (7, 5)})
sns.set(style = "whitegrid", palette = "muted")
sns.boxplot(data = df[df['release_spin_rate' > 1000][['game_date', 'release_spin_rate', 'pitch_name']].sort_values(['pitch_name', 'game_date'], ignore_index = True).sample(frac = 1.0, random_state = 7).round(2).reset_index(drop = True)
plt.suptitle('Spin Efficiency', fontsize = 18)
plt.title('Max Scherzer', fontsize = 14)
plt.xlabel('Release Spin Rate', fontsize = 12)
plt.ylabel('Pitch Name', fontsize = 12)
plt.show
```

Release Position

```
df['game_date'] = df['game_date']
sns.set(rc = {'figure.figsize': (7, 5)})
sns.set(style = "whitegrid", palette = "muted")
sns.boxplot(data = df[df['release_spin_rate' > 1000][['game_date', 'release_spin_rate', 'pitch_name']].sort_values(['pitch_name', 'game_date'], ignore_index = True).sample(frac = 1.0, random_state = 7).round(2).reset_index(drop = True)
plt.suptitle('Spin Efficiency', fontsize = 18)
plt.title('Max Scherzer', fontsize = 14)
plt.xlabel('Release Spin Rate', fontsize = 12)
plt.ylabel('Pitch Name', fontsize = 12)
plt.show
```

Release Extension - Side View

```
df['game_date'] = df['game_date']
sns.set(rc = {'figure.figsize': (7, 5)})
sns.set(style = "whitegrid", palette = "muted")
sns.boxplot(data = df[df['release_spin_rate' > 1000][['game_date', 'release_spin_rate', 'pitch_name']].sort_values(['pitch_name', 'game_date'], ignore_index = True).sample(frac = 1.0, random_state = 7).round(2).reset_index(drop = True)
plt.suptitle('Spin Efficiency', fontsize = 18)
plt.title('Max Scherzer', fontsize = 14)
plt.xlabel('Release Spin Rate', fontsize = 12)
plt.ylabel('Pitch Name', fontsize = 12)
plt.show
```


Velocity & Spin Rate

```
df['game_date'] = df['game_date']
sns.set(rc = {'figure.figsize': (7, 5)})
sns.set(style = "whitegrid", palette = "muted")
sns.boxplot(data = df[df['release_spin_rate' > 1000][['game_date', 'release_spin_rate', 'pitch_name']].sort_values(['pitch_name', 'game_date'], ignore_index = True).sample(frac = 1.0, random_state = 7).round(2).reset_index(drop = True)
plt.suptitle('Spin Efficiency', fontsize = 18)
plt.title('Max Scherzer', fontsize = 14)
plt.xlabel('Release Spin Rate', fontsize = 12)
plt.ylabel('Pitch Name', fontsize = 12)
plt.show
```


Bauer Units

```
df['game_date'] = df['game_date']
sns.set(rc = {'figure.figsize': (7, 5)})
sns.set(style = "whitegrid", palette = "muted")
sns.boxplot(data = df[df['release_spin_rate' > 1000][['game_date', 'release_spin_rate', 'pitch_name']].sort_values(['pitch_name', 'game_date'], ignore_index = True).sample(frac = 1.0, random_state = 7).round(2).reset_index(drop = True)
plt.suptitle('Spin Efficiency', fontsize = 18)
plt.title('Max Scherzer', fontsize = 14)
plt.xlabel('Release Spin Rate', fontsize = 12)
plt.ylabel('Pitch Name', fontsize = 12)
plt.show
```


Fastball: Sinker (spin efficiency) (2300 - 2616 rpm) with 91 mph avg

Sinker: Sinker (spin efficiency) (2300 - 2616 rpm) with 91 mph avg

Cutter: Cutter (slide / cleanly) (2300 - 2616 rpm) with 88 mph avg

Curveball: Slow curve (1500 - 2318 rpm) with 78 mph avg

Slider: Sweeping slider (2200 - 2616 rpm) with 84 mph avg

Sweeper: Sweeping slider with sweep (2200 - 2616 rpm) with 84 mph avg

Changeup: Three quarter arm (100 - 2318 rpm) with 78 mph avg

Count Breakdown

Pitch Usage by Count

```
df['pitch_count'] = df['balls'].astype(str) + '-' + df['strikes'].astype(str)
sns.set(rc = {'figure.figsize': (7, 5)})
sns.set(style = "whitegrid", palette = "muted")
sns.countplot(data = df[['pitch_count', 'pitch_name', 'release_speed']].sort_values(['pitch_count', 'pitch_name'], ignore_index = True).sample(frac = 1.0, random_state = 7).round(2).reset_index(drop = True)
plt.suptitle('Pitch Usage by Count', fontsize = 18)
plt.title('Max Scherzer', fontsize = 14)
plt.xlabel('Pitch Count', fontsize = 12)
plt.ylabel('Pitch Usage', fontsize = 12)
plt.legend(loc = 'upper right', ncol = 1, fancybox = True, shadow = True, title = 'Pitch Name', fontsize = 10)
plt.show
```


- 0-0 Count Heat & Sweet, Slow Curve
- Heavy Changeup and Curve in ahs counts for some Cutter
- 2-0/3-0 Curve counts

Heatmaps

Pitch Location by Pitch Type

```
df['pitch_count'] = df['balls'].astype(str) + '-' + df['strikes'].astype(str)
sns.set(rc = {'figure.figsize': (7, 5)})
sns.set(style = "whitegrid", palette = "muted")
sns.scatterplot(data = df[df['pitch_name' == 'Four-Seam Fastball'][['plate_x', 'plate_z', 'release_speed']].round(2).reset_index(drop = True)
plt.suptitle('Pitch Location by Pitch Type', fontsize = 18)
plt.title('Max Scherzer', fontsize = 14)
plt.xlabel('Plate_x', fontsize = 12)
plt.ylabel('Plate_z', fontsize = 12)
plt.legend(loc = 'upper right', ncol = 1, fancybox = True, shadow = True, title = 'Release Speed', fontsize = 10)
plt.show
```

```
df['pitch_count'] = df['balls'].astype(str) + '-' + df['strikes'].astype(str)
sns.set(rc = {'figure.figsize': (7, 5)})
sns.set(style = "whitegrid", palette = "muted")
sns.scatterplot(data = df[df['pitch_name' == 'Cutter'][['plate_x', 'plate_z', 'release_speed']].round(2).reset_index(drop = True)
plt.suptitle('Pitch Location by Pitch Type', fontsize = 18)
plt.title('Max Scherzer', fontsize = 14)
plt.xlabel('Plate_x', fontsize = 12)
plt.ylabel('Plate_z', fontsize = 12)
plt.legend(loc = 'upper right', ncol = 1, fancybox = True, shadow = True, title = 'Release Speed', fontsize = 10)
plt.show
```

```
df['pitch_count'] = df['balls'].astype(str) + '-' + df['strikes'].astype(str)
sns.set(rc = {'figure.figsize': (7, 5)})
sns.set(style = "whitegrid", palette = "muted")
sns.scatterplot(data = df[df['pitch_name' == 'Curveball'][['plate_x', 'plate_z', 'release_speed']].round(2).reset_index(drop = True)
plt.suptitle('Pitch Location by Pitch Type', fontsize = 18)
plt.title('Max Scherzer', fontsize = 14)
plt.xlabel('Plate_x', fontsize = 12)
plt.ylabel('Plate_z', fontsize = 12)
plt.legend(loc = 'upper right', ncol = 1, fancybox = True, shadow = True, title = 'Release Speed', fontsize = 10)
plt.show
```

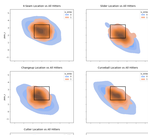
Pitch Usage and Count by Pitch Type

```
df['pitch_count'] = df['balls'].astype(str) + '-' + df['strikes'].astype(str)
sns.set(rc = {'figure.figsize': (7, 5)})
sns.set(style = "whitegrid", palette = "muted")
sns.scatterplot(data = df[df['pitch_name' == 'Four-Seam Fastball'][['plate_x', 'plate_z', 'release_speed']].round(2).reset_index(drop = True)
plt.suptitle('Pitch Location by Pitch Type', fontsize = 18)
plt.title('Max Scherzer', fontsize = 14)
plt.xlabel('Plate_x', fontsize = 12)
plt.ylabel('Plate_z', fontsize = 12)
plt.legend(loc = 'upper right', ncol = 1, fancybox = True, shadow = True, title = 'Release Speed', fontsize = 10)
plt.show
```

```
df['pitch_count'] = df['balls'].astype(str) + '-' + df['strikes'].astype(str)
sns.set(rc = {'figure.figsize': (7, 5)})
sns.set(style = "whitegrid", palette = "muted")
sns.scatterplot(data = df[df['pitch_name' == 'Cutter'][['plate_x', 'plate_z', 'release_speed']].round(2).reset_index(drop = True)
plt.suptitle('Pitch Location by Pitch Type', fontsize = 18)
plt.title('Max Scherzer', fontsize = 14)
plt.xlabel('Plate_x', fontsize = 12)
plt.ylabel('Plate_z', fontsize = 12)
plt.legend(loc = 'upper right', ncol = 1, fancybox = True, shadow = True, title = 'Release Speed', fontsize = 10)
plt.show
```

```
df['pitch_count'] = df['balls'].astype(str) + '-' + df['strikes'].astype(str)
sns.set(rc = {'figure.figsize': (7, 5)})
sns.set(style = "whitegrid", palette = "muted")
sns.scatterplot(data = df[df['pitch_name' == 'Curveball'][['plate_x', 'plate_z', 'release_speed']].round(2).reset_index(drop = True)
plt.suptitle('Pitch Location by Pitch Type', fontsize = 18)
plt.title('Max Scherzer', fontsize = 14)
plt.xlabel('Plate_x', fontsize = 12)
plt.ylabel('Plate_z', fontsize = 12)
plt.legend(loc = 'upper right', ncol = 1, fancybox = True, shadow = True, title = 'Release Speed', fontsize = 10)
plt.show
```

Pitch Location by Pitch Type



Hard Hit Summary by Pitch Type

```
1 0.00(0) 0.00(0) 0.00(0) 0.00(0) 0.00(0) 0.00(0) 0.00(0)
```

```
Fig. set 2.07: ggplot(data=d1, X_Plate_loc_ft, Y_Plate_loc_ft, shape = Throw, alpha = Throw)
ggplot(data=d1, X_Plate_loc_ft, Y_Plate_loc_ft, shape = Throw, alpha = Throw)
geom_point(size=2.5, fill=NA, color=NA)
geom_rect(data=Strike Zone, Left, Bottom, Right, Top, fill=NA, color="black", parameter="strike")
geom_rect(data=Strike Zone, Left, Bottom, Right, Top, fill=NA, color="black", parameter="strike")
geom_rect(data=Strike Zone, Left, Bottom, Right, Top, fill=NA, color="black", parameter="strike")
geom_rect(data=Hard Hit Summary
```