

Notebook Contents

- Imports
 - Data
 - Index DataFrame to Get Pitch Types
 - Pitcher Overview
 - General Pitch Data
 - Pitch Usage
 - Pitch Usage by Batter Handedness
 - Velocity by Pitch Type
 - Pitch Usage by Inning
 - Pitcher Stuff
 - Spin Rate by Pitch Type
 - Spin Axis
 - Spin Efficiency
 - HB & VB Axis
 - HB & HB Due to Magnus Force
 - Release Position
 - Release Extension
 - Velocity & Spin Rate
 - Avg. Bauer Units by Pitch Type
 - Count Breakdown
 - Pitch Usage by Count
 - Heatmaps
 - Pitch Location by Pitch Type
 - Hard Hit Summary by Pitch Type
 - Slider Heatmaps
 - 4-Seam Heatmaps
 - Curveball Heatmaps
 - Changeup 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]: kershaw = pd.read_csv('../data/clayton-kershaw.csv')
kershaw.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(kershaw.shape)
kershaw.head(2)
```

```
Out[2]:
```

	pitch_type	game_date	release_speed	release_pos_x	release_pos_z	player_name	batter	pitcher	events	description	zone	
0	SL	2021-05-01	87.5	1.6	6.13	Kershaw, Clayton	592332	477192	strikeout	called_strike	8	
1	FF	2021-05-30	90.4	1.6	6.12	Kershaw, Clayton	592332	477192		N/A	called_strike	6

```
In [3]: gen_dict = kershaw[['pitch_type', 'release_speed', 'release_spin_rate',
    'true_spin', 'spin_eff', 'phi', 'theta', 'pfx_z', 'pfx_x',
    'is_strike', 'release_pos_x', 'release_pos_z', 'bauer_units']]

col_dict = {
    'release_speed': 'velo',
    'release_spin_rate': 'spin',
    'phi': 'spin_axis',
    'theta': 'gyro_spin_deg',
    'pfx_z': 'hb',
    'pfx_x': 'vb',
    'is_strike': 'strike',
    'release_pos_x': 'r_height',
    'release_pos_z': 'r_side'
}

gen_data.rename(columns = col_dict, inplace = True)

hit_labels = [1, 2, 3, 4, 5]
kershaw['hard_hit_summary'] = pd.qcut(kershaw['launch_speed'], [0, .5262, .617, .7283, .8278, 1],
    labels = hit_labels)
```

Index DataFrame to Get Pitch Types

```
In [4]: # kershaw.pitch_type.value_counts(normalize = True)
r_kershaw = kershaw.loc[kershaw['stand'] == 'R']
l_kershaw = kershaw.loc[kershaw['stand'] == 'L']
# all hitters
sl = kershaw.loc[kershaw['pitch_type'] == 'SL']
ff = kershaw.loc[kershaw['pitch_type'] == 'FF']
cu = kershaw.loc[kershaw['pitch_type'] == 'CU']
ch = kershaw.loc[kershaw['pitch_type'] == 'CH']
# spin
r_sl = r_kershaw.loc[r_kershaw['pitch_type'] == 'SL']
r_ff = r_kershaw.loc[r_kershaw['pitch_type'] == 'FF']
r_cu = r_kershaw.loc[r_kershaw['pitch_type'] == 'CU']
r_ch = r_kershaw.loc[r_kershaw['pitch_type'] == 'CH']
# tilt
l_sl = l_kershaw.loc[l_kershaw['pitch_type'] == 'SL']
l_ff = l_kershaw.loc[l_kershaw['pitch_type'] == 'FF']
l_cu = l_kershaw.loc[l_kershaw['pitch_type'] == 'CU']
l_ch = l_kershaw.loc[l_kershaw['pitch_type'] == 'CH']
order = ['sl', 'ff', 'cu', 'ch']

sl_tilt = sl['phi'].mean()
ff_tilt = ff['phi'].mean()
cu_tilt = cu['phi'].mean()
ch_tilt = ch['phi'].mean()
```

Pitcher Overview

General Pitch Data

```
In [5]: gen_data.groupby(['pitch_type'], sort = False).mean()
```

```
Out[5]:
```

pitch_type	velo	spin	true_spin	spin_eff	spin_axis	gyro_spin_deg	hb	vb	strike	r_side
SL	87.075000	2654.045402	501.376237	0.192000	223.276316	78.878947	8.182326	4.712372	0.734884	1.513756
FF	90.636707	2505.065666	1532.492784	0.624588	184.323024	51.099566	19.353897	0.525861	0.641994	1.525317
CU	74.239927	2535.468864	1372.876019	0.558889	345.722022	55.777778	-15.283516	4.782857	0.538462	1.178508
CH	87.045455	2163.272727	1084.480000	0.780000	137.363636	38.000000	15.665455	-12.992727	0.272727	1.790909

Slider: combo sweeping slider/gyro ball movement profile

Fastball: more of a Cutter/4-Seam profile

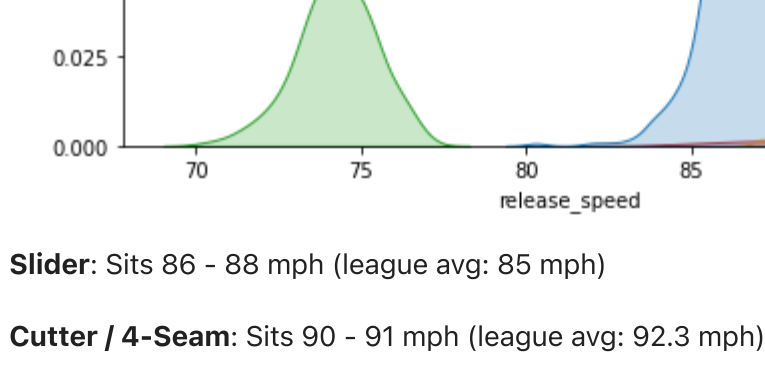
Curveball: True 12-6 banger

Change: straight change

Pitch Usage

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

dist = round(kershaw.pitch_type.value_counts(normalize = True), 2)
color = sns.color_palette('coolwarm_r')
plt.pie(dist, labels = order, colors = color, autopct = '%.0f%',
    ax[0].axis.set_major_formatter(PercentFormatter(1))
plt.title('Distribution of Pitch Types - Clayton Kershaw', fontdict = font_title, pad = 15);
```

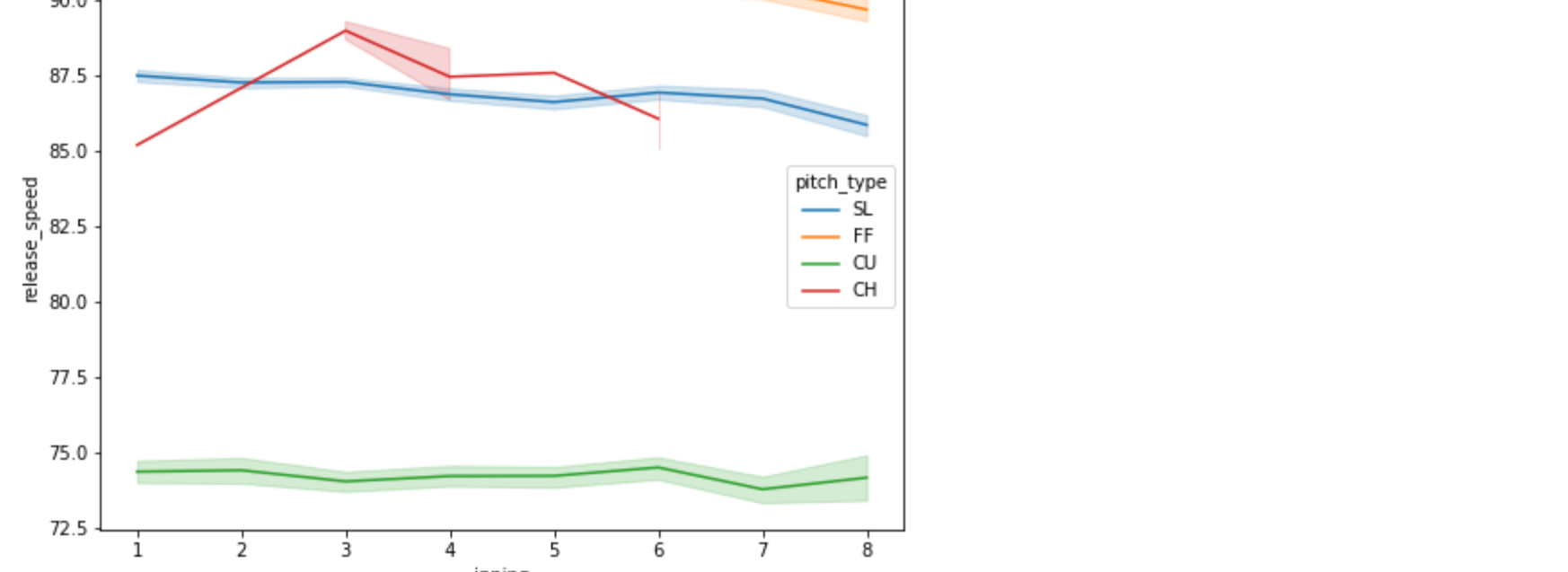


Pitch Usage by Batter Handedness

```
In [7]: blue = '#000072'
red = '#D9534F'
fig, axs = plt.subplots(1, 2, figsize = (20, 6))
fig.suptitle('Pitch Usage by Batter Handedness', fontsize = 16, fontweight = 'bold')

dist_r = r_kershaw.pitch_type.sort_values(ascending = False)
ax[0].hist(dist_r, weights = np.ones(len(dist_r)) / len(dist_r), color = blue)
ax[0].yaxis.set_major_formatter(PercentFormatter(1))
ax[0].set_title('Distribution of Pitch Types - RHH', fontdict = font_title, pad = 15)

dist_l = l_kershaw.pitch_type.sort_values(ascending = False)
ax[1].hist(dist_l, weights = np.ones(len(dist_l)) / len(dist_l), color = red)
ax[1].yaxis.set_major_formatter(PercentFormatter(1))
ax[1].set_title('Distribution of Pitch Types - LHH', fontdict = font_title, pad = 15);
```



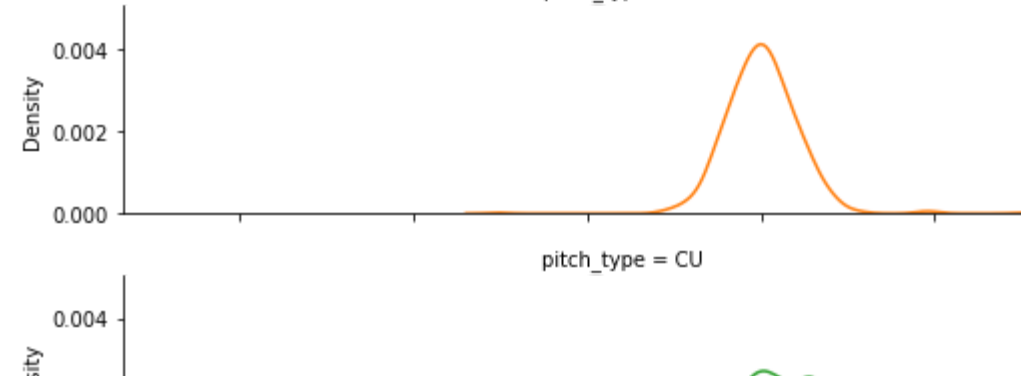
RHH: mainly SL/FF CU mix

LHH: mainly FF/SL CU mix

Velocity by Pitch Type

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

ax = sns.kdeplot(data = kershaw, x = 'release_speed', shade = 'fill', hue = 'pitch_type',
    hue_order = order, palette = 'tab10')
sns.move_legend(ax, 'upper left')
plt.title('Distribution of Velocity by Pitch Type - Clayton Kershaw', fontdict = font_title, pad = 15);
```



Slider: Sits 86 - 88 mph (league avg: 85 mph)

Cutter / 4-Seam: Sits 90 - 91 mph (league avg: 92.3 mph)

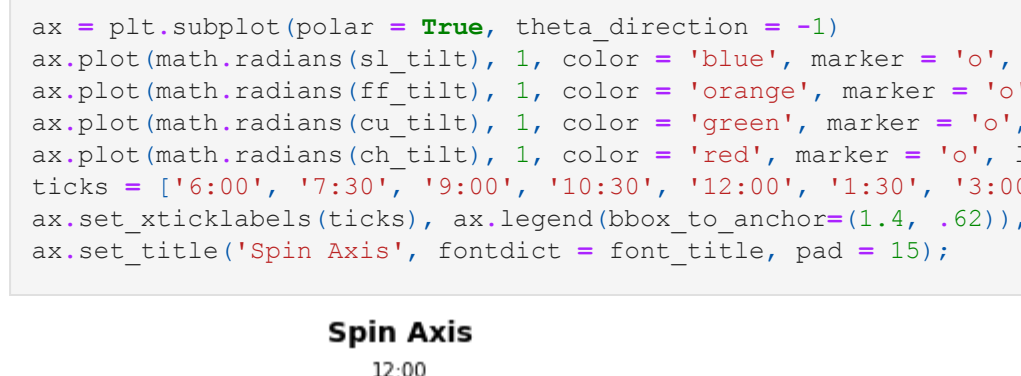
Curveball: Sits 74 - 75 mph (league avg: 77 mph (11 - 18 mph slower compared to FB))

Changeup: Sits 86 - 88 mph (league avg: 85 mph (8 - 12 mph difference from FB))

Pitch Velocity by Inning

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

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

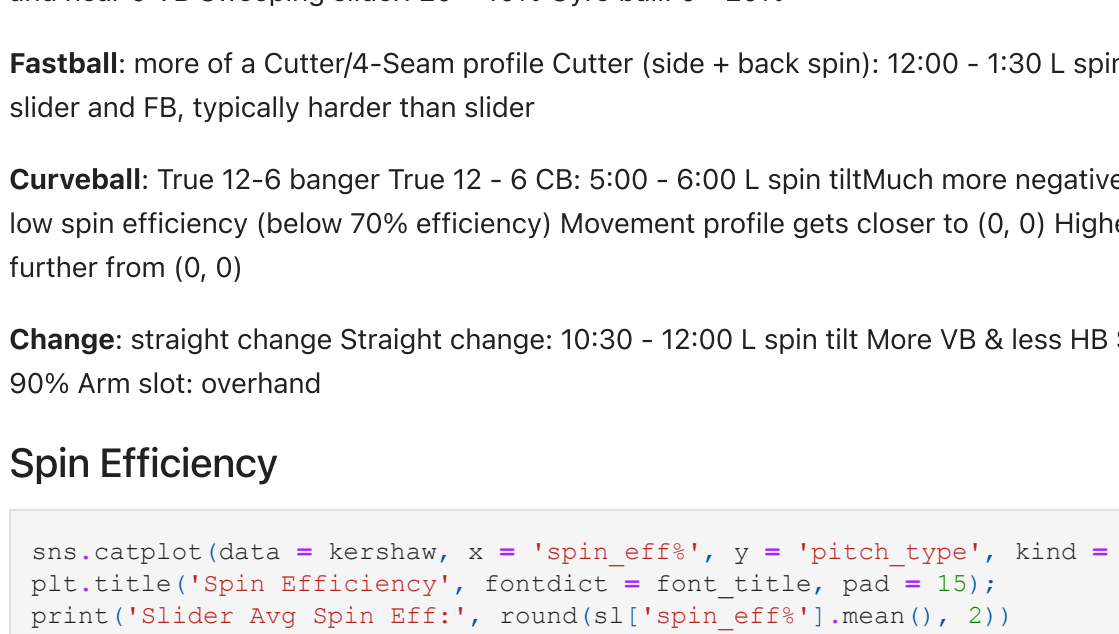


Slight dip in velo over the course of the game

Pitcher Stuff

Spin Rate by Pitch Type

```
In [10]: g = sns.FacetGrid(kershaw, row = 'pitch_type', hue = 'pitch_type', height = 2, aspect = 4, )
g.map(sns.kdeplot, 'release_spin_rate', palette = 'tab10');
```



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

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

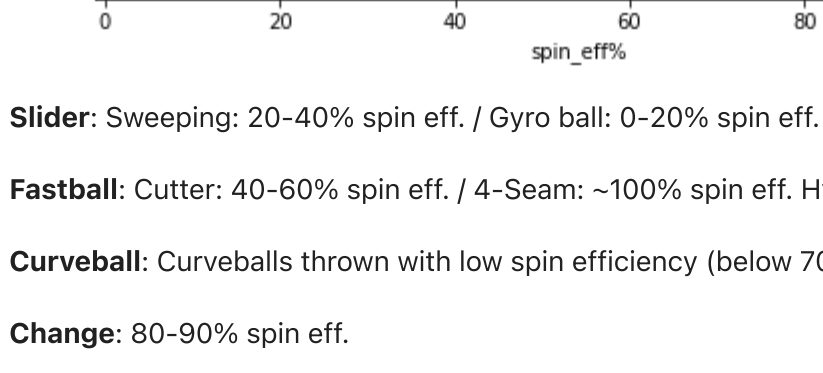
Curveball: Above average spin (league avg: ~2500 rpm)

Changeup: Above average spin (league avg range: ~1200 2400 rpm)

Spin Axis

```
In [11]: ax = plt.figure(figsize = (8, 6))

ax = plt.subplot(polar = True, theta_direction = 'marker')
ax.plot(math.radians(sl_tilt), 1, color = 'blue', marker = 'o', label = 'Slider')
ax.plot(math.radians(ff_tilt), 1, color = 'orange', marker = 'o', label = '4-Seam')
ax.plot(math.radians(cu_tilt), 1, color = 'green', marker = 'o', label = 'Curveball')
ax.plot(math.radians(ch_tilt), 1, color = 'red', marker = 'o', label = 'Changeup')
ax.legend(bbox_to_anchor=(1.4, .62))
ax.set_xticklabels(ticks, ax.legend(bbox_to_anchor=(1.4, .62)), ax.set_theta_zero_location("S"))
ax.set_title('Spin Axis', fontdict = font_title, pad = 15);
```



Pairs: 4-Seam / Curveball pair extremely well. 4-Seam / Slider pair well

Slider: combo sweeping slider/gyro ball movement profile Sweeping slider (side spin): 2:00 - 4:00 L spin tilt Decent amount of HB and near 0 VB Sweeping slider: 20 - 40% Gyro ball: 0 - 20%

Fastball: more of a Cutter/4-Seam profile Cutter (side + back spin): 12:00 - 1:30 L spin tilt Positive VB & Some HB Hybrid between slider and FB, typically harder than slider

Curveball: True 12-6 banger True 12 - 6 CB: 5:00 - 6:00 L spin tilt Much more negative VB and minimal HB Curveballs thrown with low spin efficiency (below 70% efficiency) Movement profile gets closer to (0, 0) Higher spin efficiency Curveballs profile gets further from (0, 0)

Change: straight change Straight change: 10:30 - 12:00 L spin tilt More VB & less HB Spin rate: 1700 - 1800 rpm Spin efficiency: 90% Arm slot: overhand

Spin Efficiency

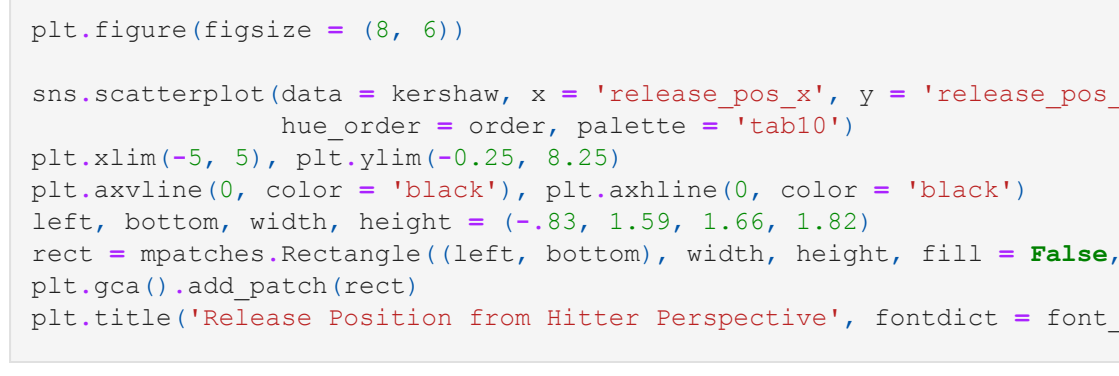
```
In [12]: sns.violinplot(data = kershaw, x = 'spin_eff', y = 'pitch_type', kind = 'violin', aspect = 1.5, palette = 'tab10')
print('Slider Avg Spin Eff:', round(sl['spin_eff'].mean(), 2))
print('4-Seam Avg Spin Eff:', round(ff['spin_eff'].mean(), 2))
print('Curveball Avg Spin Eff:', round(cu['spin_eff'].mean(), 2))
print('Changeup Avg Spin Eff:', round(ch['spin_eff'].mean(), 2));
```

Slider Avg Spin Eff: 19.2

4-Seam Avg Spin Eff: 62.42

Curveball Avg Spin Eff: 55.69

Changeup Avg Spin Eff: 78.0



Slider: Sweeping: 20-40% spin eff. / Gyro ball: 0-20% spin eff.

Fastball: Cutter: 40-60% spin eff. / 4-Seam: ~100% spin eff. Hybrid between slider and FB, typically harder than slider

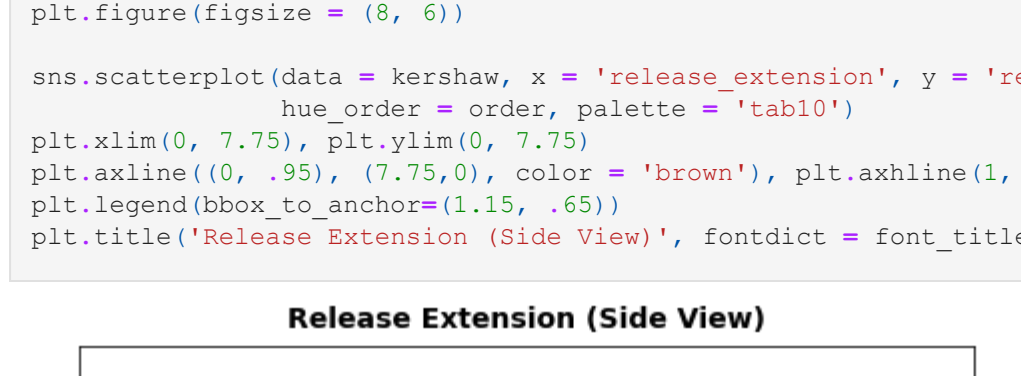
Curveball: Curveballs thrown with low spin efficiency (below 70% efficiency) Movement profile gets closer to (0, 0)

Change: 80-90% spin eff.

Horizontal & Vertical Break Axis

```
In [13]: plt.figure(figsize = (8, 6))

sns.scatterplot(data = kershaw, x = 'pfx_x', y = 'pfx_z', hue = 'pitch_type',
    hue_order = order, palette = 'tab10')
plt.xlim(-30, 30), plt.ylim(-30, 30)
plt.axhline(0, color = 'black'), plt.axvline(0, color = 'black')
plt.title('HB & VB Axis', fontdict = font_title, pad = 15);
```



Slider: Decent amount of HB & minimal VB

Fastball: Ton of HB & near 0 VB

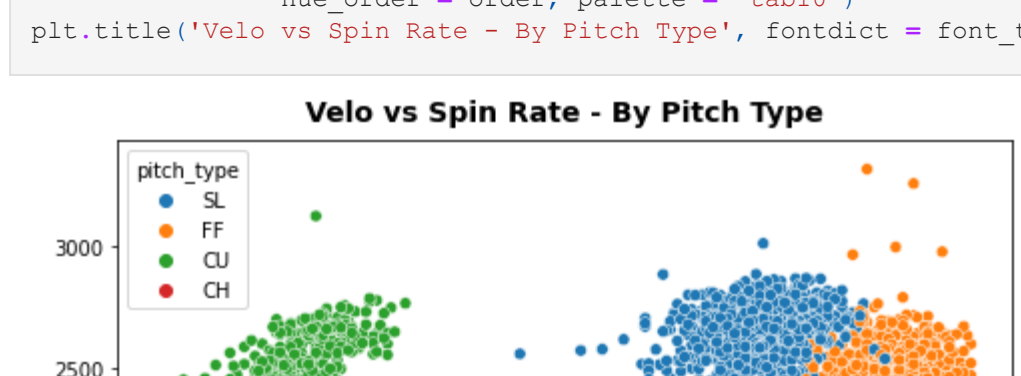
Curveball: Minimal HB & much more negative VB

Change: Decent HB & VB

Horizontal & Vertical Break Due to Magnus Force Axis

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

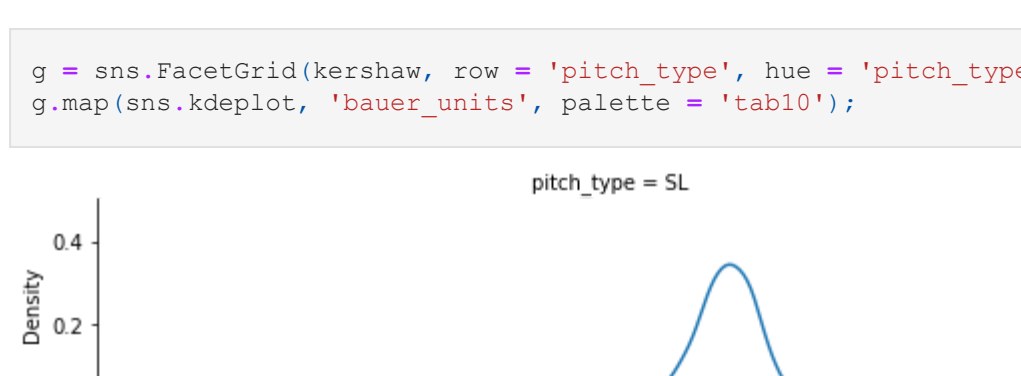
sns.scatterplot(data = kershaw, x = 'Mx', y = 'My', hue = 'pitch_type',
    hue_order = order, palette = 'tab10')
plt.xlim(-30, 30), plt.ylim(-30, 30)
plt.axhline(0, color = 'black'), plt.axvline(0, color = 'black')
plt.title('HB & VB from Magnus Force Axis', fontdict = font_title, pad = 15);
```



Release Postion

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

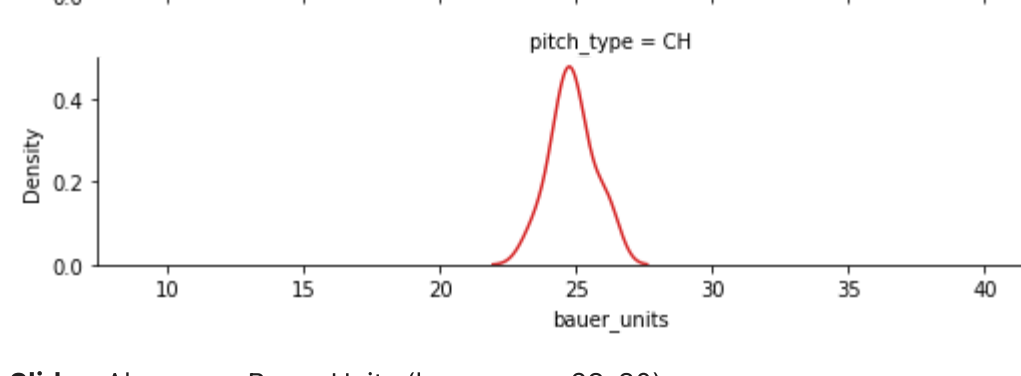
sns.scatterplot(data = kershaw, x = 'release_pos_x', y = 'release_pos_z', hue = 'pitch_type',
    hue_order = order, palette = 'tab10')
plt.xlim(-5, 5), plt.ylim(-5, 5)
plt.axhline(0, color = 'black'), plt.axvline(0, color = 'black')
left, bottom, width, height = (-83, 1.5, 1.66, 1.82)
sns.mplot(Rectangle((left, bottom, width, height), fill = False, color = 'black', linewidth = 2))
plt.gca().add_patch(rect)
plt.title('Release Position from Hitter Perspective', fontdict = font_title, pad = 15);
```



Release Extension (Side)

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

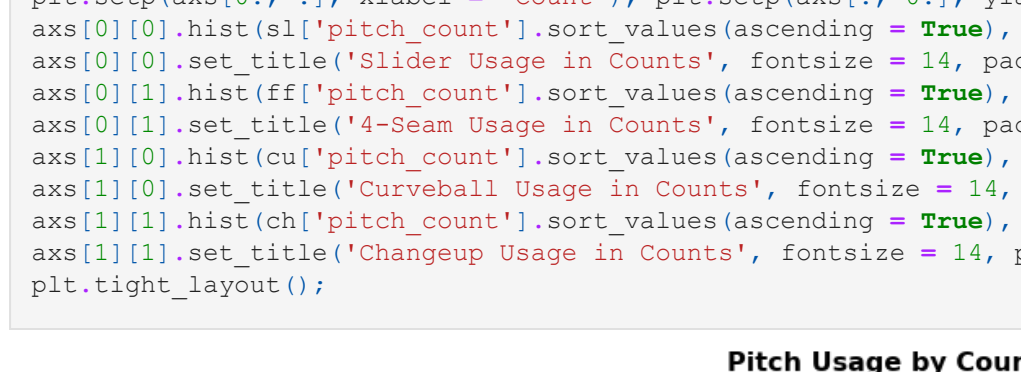
sns.scatterplot(data = kershaw, x = 'release_extension', y = 'release_pos_x', hue = 'pitch_type',
    hue_order = order, palette = 'tab10')
plt.xlim(0, 7.75), plt.ylim(0, 7.75)
plt.axhline(0, color = 'black'), plt.axvline(1, color = 'black')
plt.legend(bbox_to_anchor=(1.4, .62))
plt.title('Release Extension (Side View)', fontdict = font_title, pad = 15);
```



Velocity & Spin Rate

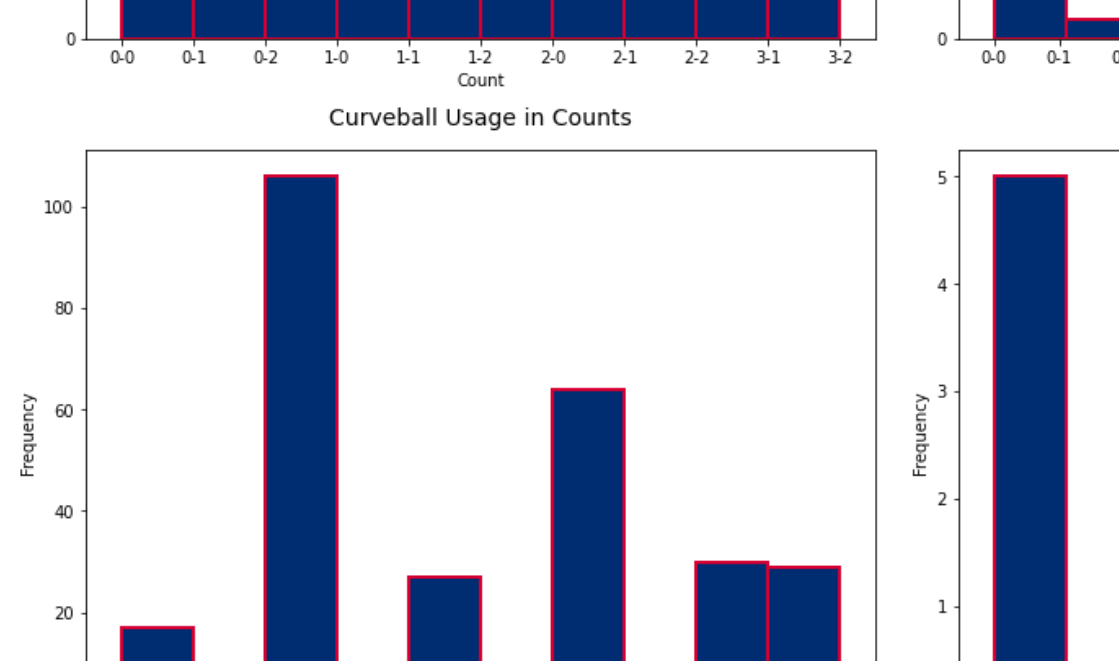
```
In [17]: plt.figure(figsize = (8, 6))

sns.scatterplot(data = kershaw, x = 'release_speed', y = 'release_spin_rate', hue = 'pitch_type',
    hue_order = order, palette = 'tab10')
plt.title('Velo vs Spin Rate - By Pitch Type', fontdict = font_title, pad = 15);
```



Bauer Units

```
In [18]: g = sns.FacetGrid(kershaw, row = 'pitch_type', hue = 'pitch_type', height = 2, aspect = 4, )
g.map(sns.kdeplot, 'bauer_units', palette = 'tab10');
```



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

Fastball: Above avg Bauer Units (league avg: 24)

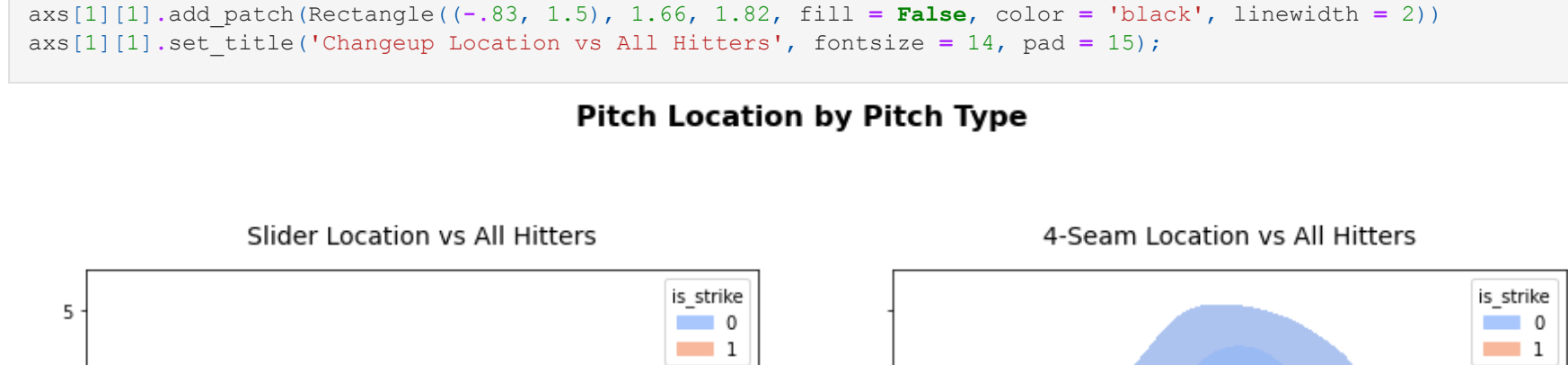
Curveball: Above avg Bauer Units (league avg: 28-30)

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

Count Breakdown

Pitch Usage by Count

```
In [19]: fig, axs = plt.subplots(2, 2, figsize = (15, 12))
fig.suptitle('Pitch Location by Pitch Type', fontsize = 16, fontweight = 'bold')
plt.setp(axs[0, 0].xaxis, label = 'Count', plt.setp(axs[0, 1].xaxis, label = 'Frequency')
axs[0][0].hist(al['pitch_count']).sort_values(ascending = True), color = blue, edgecolor = red, linewidth = 2)
axs[0][1].hist(al['pitch_count']).sort_values(ascending = True), color = blue, edgecolor = red, linewidth = 2)
axs[1][0].hist(cu['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)
plt.tight_layout();
```



• 0-0 Count: Maily 4-Seam (47%) & Slider (19%)

• Does not throw Curveball / Changeup behind in counts

• Heavy Curveball in 0-1 & 1-1 Counts

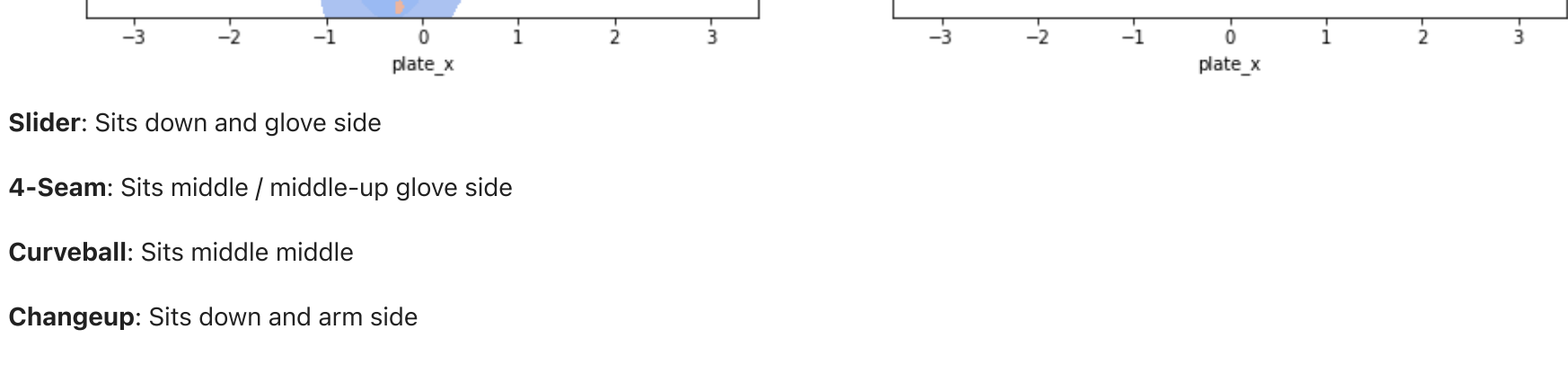
• Mainly 4-Seam & Slider in all other counts

Heatmaps

- All From Hitters' Perspective

Pitch Location by Pitch Type

```
In [20]: fig, axs = plt.subplots(2, 2, figsize = (15, 13), sharex = True, sharey = True)
fig.suptitle('Pitch Location by Pitch Type', fontsize = 16, fontweight = 'bold')
plt.setp(axs[0, 0].xaxis, label = 'plate_x', plt.setp(axs[0, 1].xaxis, label = 'frequency')
axs[0][0].add_patch(Rectangle((-83, 1.5, 1.66, 1.82), fill = False, color = 'black', linewidth = 2))
axs[0][1].set_title('Slider Location vs All Hitters', fontsize = 14, pad = 15)
sns.kdeplot(ax = axs[0][0], data=ff, x='plate_x', y='plate_z', fill = True, hue='is_strike', palette='coolwarm')
axs[0][1].add_patch(Rectangle((-83, 1.5, 1.66, 1.82), fill = False, color = 'black', linewidth = 2))
axs[1][0].add_patch(Rectangle((-83, 1.5, 1.66, 1.82), fill = False, color = 'black', linewidth = 2))
axs[1][1].set_title('4-Seam Location vs All Hitters', fontsize = 14, pad = 15)
sns.kdeplot(ax = axs[1][0], data=cu, x='plate_x', y='plate_z', fill = True, hue='is_strike', palette='coolwarm')
axs[1][1].add_patch(Rectangle((-83, 1.5, 1.66, 1.82), fill = False, color = 'black', linewidth = 2))
axs[2][0].add_patch(Rectangle((-83, 1.5, 1.66, 1.82), fill = False, color = 'black', linewidth = 2))
axs[2][1].set_title('Curveball Location vs All Hitters', fontsize = 14, pad = 15)
sns.kdeplot(ax = axs[2][0], data=ch, x='plate_x', y='plate_z', fill = True, hue='is_strike', palette='coolwarm')
axs[2][1].add_patch(Rectangle((-83, 1.5, 1.66, 1.82), fill = False, color = 'black', linewidth = 2))
axs[3][0].add_patch(Rectangle((-83, 1.5, 1.66, 1.82), fill = False, color = 'black', linewidth = 2))
axs[3][1].set_title('Changeup Location vs All Hitters', fontsize = 14, pad = 15);
```



Slider: Sits down and glove side

4-Seam: Sits middle / middle-up glove side

Curveball: Sits middle middle

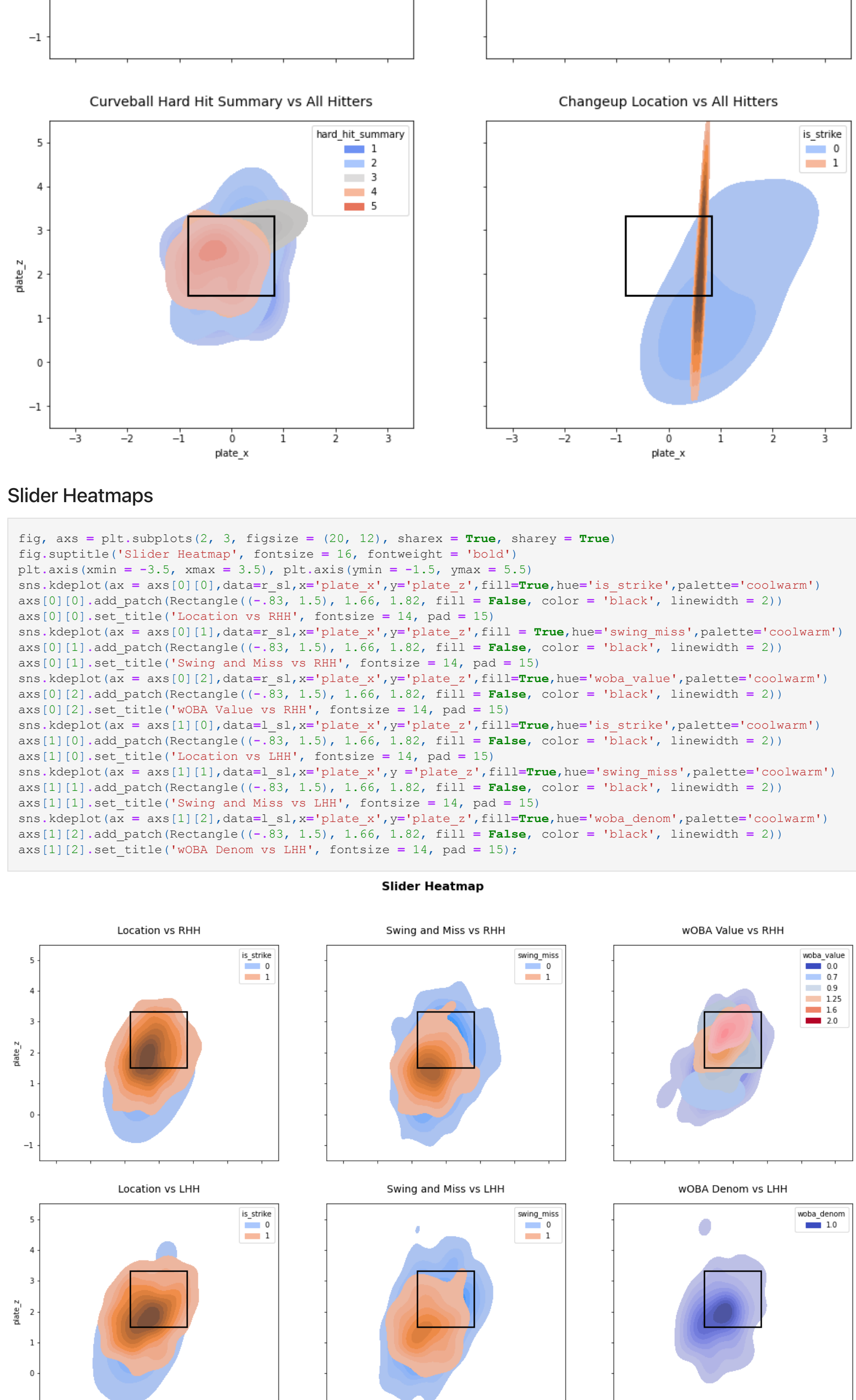
Changeup: Sits down and arm side

Hard Hit Summary by Pitch Type

1. 0-85 EV 2. 85-90 EV 3. 90-95 EV 4. 95-100 EV 5. 100+ EV


```
fig, axes = plt.subplots(2, 3, figsize = (15, 12), sharex = True, sharey = True)
fig.suptitle('Hard Hit Summary by Pitch Type', fontsize = 16, fontweight = 'bold')
plt.axis(xmin = -3.5, xmax = 3.5), plt.axis(ymin = -1.5, ymax = 5.5)
sns.kdeplot(ax=axes[0][0], data=rs, x='plate_x', y='plate_z', fill=True, hue='hard_hit_summary', palette='coolwarm')
axes[0][0].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[0][0].set_title('Slider Hard Hit Summary vs All Hitters', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[0][1], data=rs, x='plate_x', y='plate_z', fill = True, hue='hard_hit_summary', palette='coolwarm')
axes[0][1].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[0][1].set_title('4-Seam Hard Hit Summary vs All Hitters', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[0][2], data=rs, x='plate_x', y='plate_z', fill=True, hue='hard_hit_summary', palette='coolwarm')
axes[0][2].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[0][2].set_title('Curveball Hard Hit Summary vs All Hitters', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[1][0], data=rs, x='plate_x', y='plate_z', fill=True, hue='hard_hit_summary', palette='coolwarm')
axes[1][0].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[1][0].set_title('Changeup Location vs All Hitters', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[1][1], data=rs, x='plate_x', y='plate_z', fill=True, hue='hard_hit_summary', palette='coolwarm')
axes[1][1].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[1][1].set_title('Changeup Location vs All Hitters', fontsize = 14, pad = 15);
```

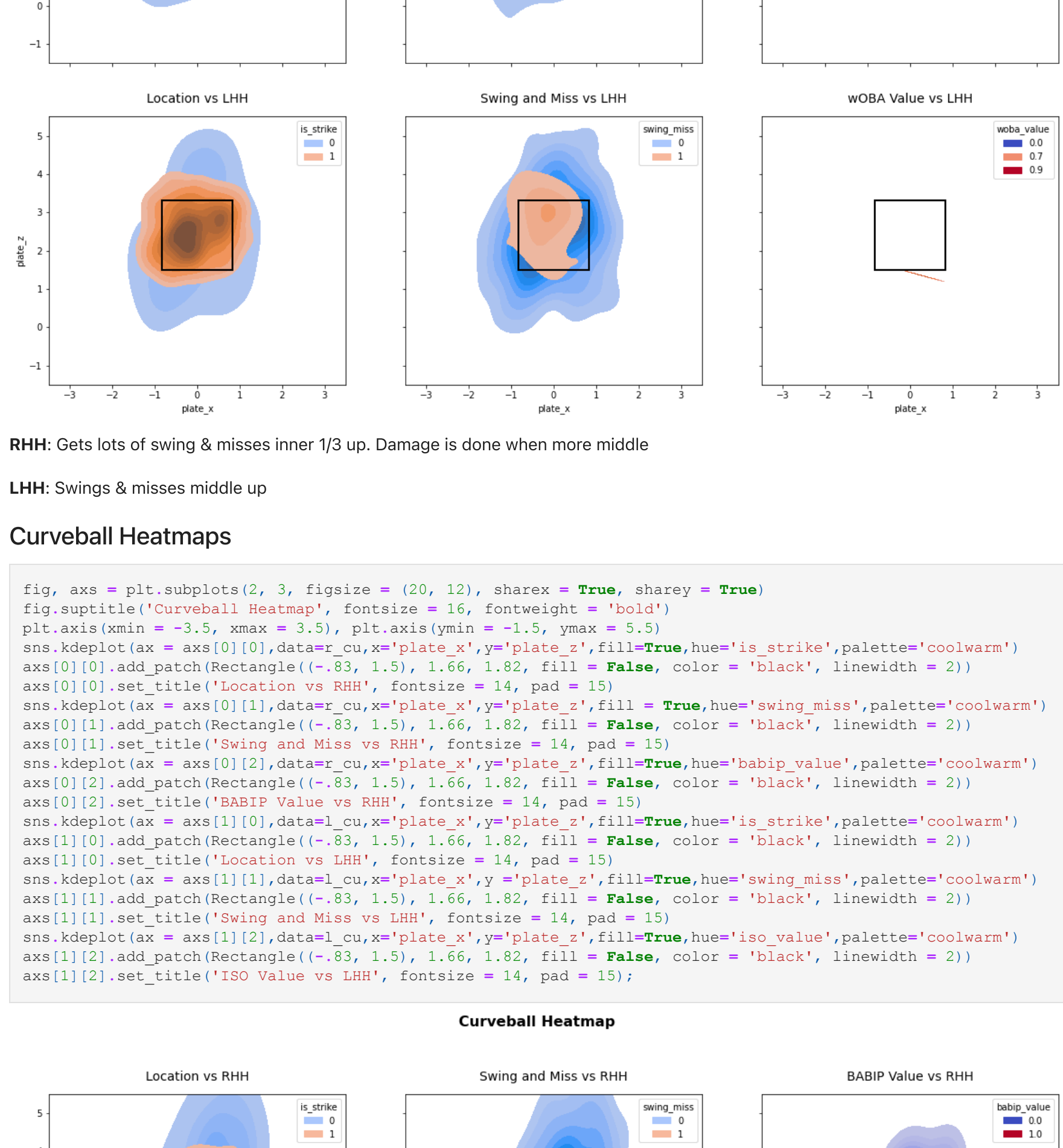
Hard Hit Summary by Pitch Type



Slider Heatmaps

```
fig, axes = plt.subplots(2, 3, figsize = (20, 12), sharex = True, sharey = True)
fig.suptitle('Slider Heatmap', fontsize = 16, fontweight = 'bold')
plt.axis(xmin = -3.5, xmax = 3.5), plt.axis(ymin = -1.5, ymax = 5.5)
sns.kdeplot(ax=axes[0][0], data=rs, x='plate_x', y='plate_z', fill=True, hue='is_strike', palette='coolwarm')
axes[0][0].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[0][0].set_title('Location vs RHH', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[0][1], data=rs, x='plate_x', y='plate_z', fill = True, hue='swing_miss', palette='coolwarm')
axes[0][1].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[0][1].set_title('Swing and Miss vs RHH', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[0][2], data=rs, x='plate_x', y='plate_z', fill=True, hue='woba_value', palette='coolwarm')
axes[0][2].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[0][2].set_title('Woba Value vs RHH', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[1][0], data=rs, x='plate_x', y='plate_z', fill=True, hue='is_strike', palette='coolwarm')
axes[1][0].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[1][0].set_title('Location vs LHH', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[1][1], data=rs, x='plate_x', y='plate_z', fill = True, hue='swing_miss', palette='coolwarm')
axes[1][1].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[1][1].set_title('Swing and Miss vs LHH', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[1][2], data=rs, x='plate_x', y='plate_z', fill=True, hue='woba_value', palette='coolwarm')
axes[1][2].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[1][2].set_title('Woba Denom vs LHH', fontsize = 14, pad = 15);
```

Slider Heatmap



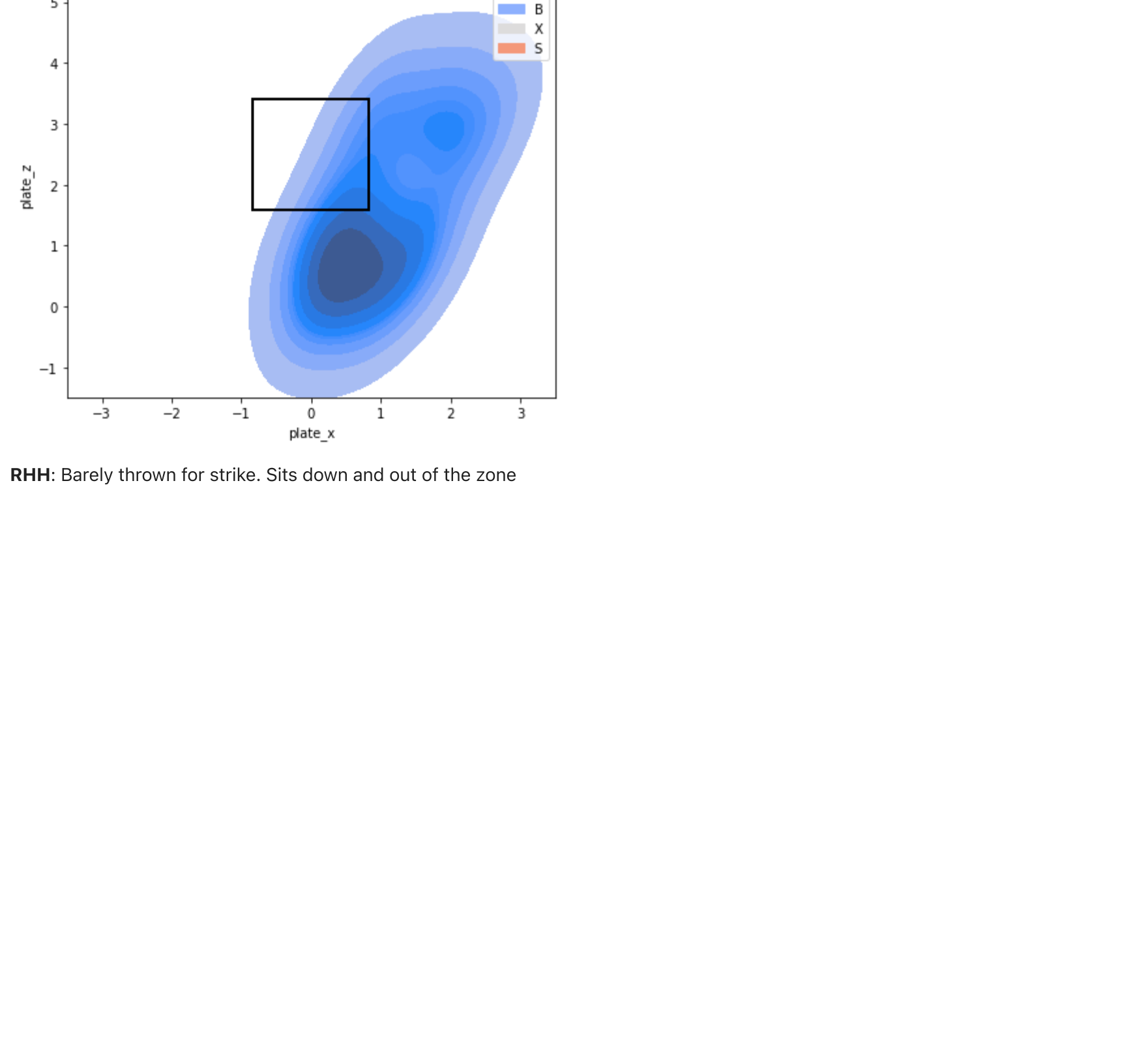
RHH: Gets lots of swing & misses down & in, inside corner. Damage is done when more middle

LHH: Lots of swings & misses down & away. Damage done middle middle

4-Seam Heatmaps

```
fig, axes = plt.subplots(2, 3, figsize = (20, 12), sharex = True, sharey = True)
fig.suptitle('4-Seam Heatmap', fontsize = 16, fontweight = 'bold')
plt.axis(xmin = -3.5, xmax = 3.5), plt.axis(ymin = -1.5, ymax = 5.5)
sns.kdeplot(ax=axes[0][0], data=rs, x='plate_x', y='plate_z', fill=True, hue='is_strike', palette='coolwarm')
axes[0][0].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[0][0].set_title('Location vs RHH', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[0][1], data=rs, x='plate_x', y='plate_z', fill = True, hue='swing_miss', palette='coolwarm')
axes[0][1].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[0][1].set_title('Swing and Miss vs RHH', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[0][2], data=rs, x='plate_x', y='plate_z', fill=True, hue='woba_value', palette='coolwarm')
axes[0][2].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[0][2].set_title('Woba Value vs RHH', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[1][0], data=rs, x='plate_x', y='plate_z', fill=True, hue='is_strike', palette='coolwarm')
axes[1][0].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[1][0].set_title('Location vs LHH', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[1][1], data=rs, x='plate_x', y='plate_z', fill = True, hue='swing_miss', palette='coolwarm')
axes[1][1].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[1][1].set_title('Swing and Miss vs LHH', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[1][2], data=rs, x='plate_x', y='plate_z', fill=True, hue='woba_value', palette='coolwarm')
axes[1][2].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[1][2].set_title('Woba Denom vs LHH', fontsize = 14, pad = 15);
```

4-Seam Heatmap



RHH: Gets lots of swing & misses inner 1/3 up. Damage is done when more middle

LHH: Swings & misses middle up

Curveball Heatmaps

```
fig, axes = plt.subplots(2, 3, figsize = (20, 12), sharex = True, sharey = True)
fig.suptitle('Curveball Heatmap', fontsize = 16, fontweight = 'bold')
plt.axis(xmin = -3.5, xmax = 3.5), plt.axis(ymin = -1.5, ymax = 5.5)
sns.kdeplot(ax=axes[0][0], data=rs, x='plate_x', y='plate_z', fill=True, hue='is_strike', palette='coolwarm')
axes[0][0].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[0][0].set_title('Location vs RHH', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[0][1], data=rs, x='plate_x', y='plate_z', fill = True, hue='swing_miss', palette='coolwarm')
axes[0][1].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[0][1].set_title('Swing and Miss vs RHH', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[0][2], data=rs, x='plate_x', y='plate_z', fill=True, hue='babip_value', palette='coolwarm')
axes[0][2].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[0][2].set_title('BABIP Value vs RHH', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[1][0], data=rs, x='plate_x', y='plate_z', fill=True, hue='is_strike', palette='coolwarm')
axes[1][0].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[1][0].set_title('Location vs LHH', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[1][1], data=rs, x='plate_x', y='plate_z', fill = True, hue='swing_miss', palette='coolwarm')
axes[1][1].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[1][1].set_title('Swing and Miss vs LHH', fontsize = 14, pad = 15)
sns.kdeplot(ax=axes[1][2], data=rs, x='plate_x', y='plate_z', fill=True, hue='iso_value', palette='coolwarm')
axes[1][2].add_patch(Rectangle((-85, 1.5), 1.66, 1.82, fill = False, color = 'black', linewidth = 2))
axes[1][2].set_title('ISO Value vs LHH', fontsize = 14, pad = 15);
```

Curveball Heatmap



RHH: Swing and misses are down & out of the zone. Hit the middle hanger

LHH: Swing and misses are down & out of the zone. Hit the hanger

Changeup Heatmaps

```
plt.figure(figsize = (6, 6))
sns.kdeplot(data = r, ch, x = 'plate_x', y = 'plate_z', fill = True, hue = 'type', palette='coolwarm')
plt.xlim(-3.5, 3.5), plt.ylim(-1.5, 5.5)
plt.title('Changeup Heatmap', fontsize = 16, fontweight = 'bold')
plt.tight_layout();
```

Changeup Heatmap



RHH: Barely thrown for strike. Sits down and out of the zone