Group 2 Project Proposal

Waleed Almousa, Samuel Handel, Thomas Huspeni, Colin Macy, Teagen Williams

Dataset:

We are using a custom dataset that consists of data pulled from MLB's statcast database. It consists of data from their catcher throwing, pitch tempo, sprint speed, and pitcher running game leaderboards. Additionally we accessed the data for each stolen base attempt using an API call to baseballsavant.mlb.com which hosts their statcast data.

Code:

```
import requests
import pandas as pd
import pybaseball



catcher_throwing = pd.read_csv("catcher_throwing.csv")
pitch_tempo = pd.read_csv("pitch_tempo.csv").dropna()
sprint_speed = pd.read_csv("sprint_speed.csv")
pitcher_running_game = pd.read_csv("pitcher_running_game.csv")

statcast_2023 = pb.statcast("2023-03-29","2023-10-02") #comment out after run once
%store statcast_2023 #comment out after run once
%store -r statcast_2023

catcher_ids = list(set(statcast_2023.fielder_2))



sb_columns = [
   'game_pk',
   'at_bat_number',
   'runner_id',
   'pitcher_id',
   'catcher_id',
   'is_runner_sb2'
]

total_sb = pd.DataFrame(columns = sb_columns)

for id in catcher_ids:
```

```
    url =
f'https://baseballsavant.mlb.com/leaderboard/services/catcher-throwing/{id}?game_typ
e=Regular&n=q&season_end=2023&season_start=2023&split=no&team=&type=Cat&with_team_on
ly=1'

    response = requests.get(url)

    data = response.json().get('data', [])
    data = pd.json_normalize(data)

    if len(data) > 0:
        total_sb = pd.concat([total_sb, data[sb_columns]], ignore_index=True)


total_sb = total_sb.rename(columns = {'is_runner_sb2': 'successful_sb'})
total_sb = pd.merge(total_sb,
sprint_speed[["player_id","sprint_speed"]].rename(columns = {"player_id" :
"runner_id", "sprint_speed" : "runner_sprint_speed"}), how = "left", on =
"runner_id")

total_sb = pd.merge(total_sb,
pitch_tempo[["entity_id","median_seconds_empty"]].rename(columns = {"entity_id" :
"pitcher_id", "median_seconds_empty" : "pitch_tempo"}), how = "left", on =
"pitcher_id")

total_sb = pd.merge(total_sb,
pitcher_running_game[["player_id",'rate_sb2cs2',"pos11_r_sec_minus_prim_lead","pos11
_r_sec_minus_prim_lead_sb2cs2"]].rename(columns = {"player_id" : "pitcher_id",
"rate_sb2cs2":"sb_att_rate","pos11_r_sec_minus_prim_lead":"lead_dist_gained_opp","po
s11_r_sec_minus_prim_lead_sb2cs2":"lead_dist_gained_att"}), how = "left", on =
"pitcher_id")

total_sb = pd.merge(total_sb,
catcher_throwing[["player_id","pop_time","exchange_time","arm_strength","cs_aa_per_t
hrow"]].rename(columns = {"player_id" : "catcher_id"}), how = "left", on =
"catcher_id")

total_sb = total_sb.dropna().reset_index(drop = True)

total_sb.to_csv("stolen_base_features.csv")
```

Potential Questions:

Can we predict the success of a stolen base attempt based on other factors, such as runner sprint speed, pitch tempo, and lead distance gained? Which variables influence the success of a stolen base attempt the most?

- A Logistic regression model would likely work best for this question, since the type of variable we are trying to predict is binary
- Multiple regression may also work, where we use multiple variables to try to predict the success of a stolen base attempt

Is there a relationship between the success of a stolen base attempt and the current game condition(i.e. Inning and score difference)?

- Logistic regression

How does lead distance gained relate to the success of a stolen base attempt? What is the threshold for lead_distance_gained that correlates to a higher stolen base success rate?

- A decision tree may work well for answering this question
- We can group the data by different ranges of lead_distance_gained and their respective stolen base attempt success rates, and see if there's any range where the success rate is significantly higher

Is there any placebo effect where a specific range of player_id's have a higher stolen base attempt success rate?

- A similar method to the previous question would work for this

How much does the skill of the pitcher(i.e pitch_tempo, arm strength), or the skill of the runners(runner_sprint_speed) influence the success rate of stolen base attempts? Which matters more, the runner's sprint speed or the skill of the pitcher?

- Logistic regression

Variables:

Variable	Explanation
game_pk	Game ID

at_bat_number	At bat number from game.
runner_id	Runner ID
pitcher_id	Pitcher ID
catcher_id	Catcher ID
successful_sb	Indicator of whether or not stolen base attempt was successful.
runner_sprint_speed	Average sprint speed for runner in ft/sec.
pitch_tempo	Measures the time between pitch releases: starting the clock as soon as the pitcher releases the previous pitch and ending when the pitcher releases the next pitch.
sb_att_rate	Stolen Base Attempt Percent: Stolen base attempts per opportunity
lead_dist_gained_opp	Lead Distance Gained: Distance a runner has advanced from the start of delivery to pitch release (in feet). All stolen base opportunities.
lead_dist_gained_att	Lead Distance Gained: Distance a runner has advanced from the start of delivery to pitch release (in feet). Only stolen base attempts.
pop_time	The time it takes from catcher receiving the pitch to the middle infielder catching the throw at second.
exchange_time	The average of exchange times in seconds. Exchange is the time between the catcher receiving the pitch and releasing his throw.
arm_strength	The average arm strength of the catcher's throws to second.
cs_aa_per_throw	The CS Above Avg. gained per throw compared to the expectation of an average catcher. Calculated by CS Above Avg. / Throw Opportunities.

Methods:

Logistic regression would work best for answering most of these questions, since the type of variable we are trying to predict is binary.We can also use decision tree classification to answer some questions, to determine the splits based on different variables for stolen base success.