VISUALLY REPRESENTING HOCKEY SHOT DATA

KAITLEN MARTIN

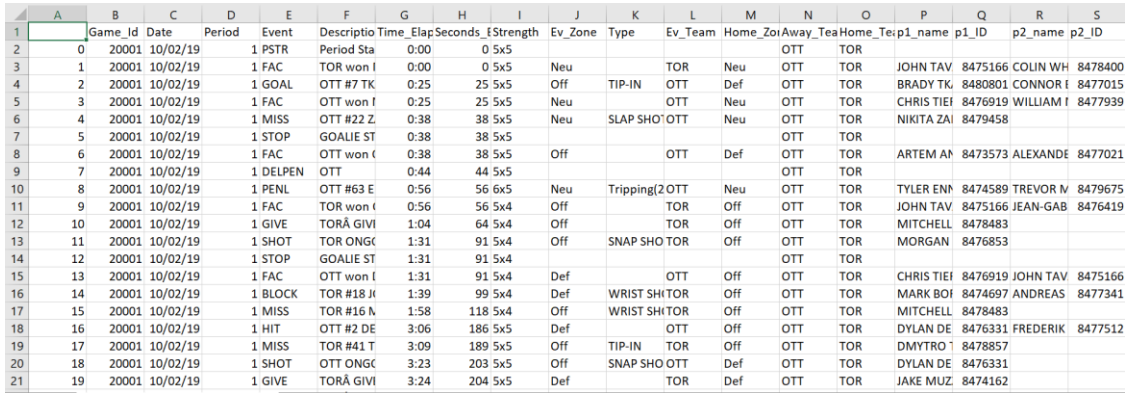
THE TORONTO MAPLE LEAFS ARE BAD AT HOCKEY. I'M BAD AT PYTHON.

Me working on this project

- Objective: Create a python script that could:
  - 1) Pull data from the NHL API
  - 2) Take this data and sort through to only find shots made by the Toronto Maple Leafs
  - 3) Display this data on a shapefile of a standard NHL rink.

```
import hockey_scraper
##Source:https://hockey-scraper.readthedocs.io/en/latest/index.html

hockey_scraper.scrape_date_range('2019-10-02', '2019-11-16', True)
##Collecting all the game data for the 2019-2020 season,
    ##including individual shift data, into an Excel spreadsheet
```



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1		Game_Id	Date	Period	Event	Descriptio	Time_Elap	Seconds_E	Strength	Ev_Zone	Type	Ev_Team	Home_Zor	Away_Tea	Home_Tea	p1_name	p1_ID	p2_name	p2_ID
2	0	20001	10/02/19	1	PSTR	Period Sta	0:00	0	5x5					OTT	TOR				
3	1	20001	10/02/19	1	FAC	TOR won I	0:00	0	5x5	Neu		TOR	Neu	OTT	TOR	JOHN TAV	8475166	COLIN WH	8478400
4	2	20001	10/02/19	1	GOAL	OTT #7 TK	0:25	25	5x5	Off	TIP-IN	OTT	Def	OTT	TOR	BRADY TK	8480801	CONNOR I	8477015
5	3	20001	10/02/19	1	FAC	OTT won I	0:25	25	5x5	Neu		OTT	Neu	OTT	TOR	CHRIS TIEI	8476919	WILLIAM I	8477939
6	4	20001	10/02/19	1	MISS	OTT #22 Z	0:38	38	5x5	Neu	SLAP SHO	OTT	Neu	OTT	TOR	NIKITA ZA	8479458		
7	5	20001	10/02/19	1	STOP	GOALIE ST	0:38	38	5x5					OTT	TOR				
8	6	20001	10/02/19	1	FAC	OTT won (	0:38	38	5x5	Off		OTT	Def	OTT	TOR	ARTEM AN	8473573	ALEXANDE	8477021
9	7	20001	10/02/19	1	DELPEN	OTT	0:44	44	5x5					OTT	TOR				
10	8	20001	10/02/19	1	PENL	OTT #63 E	0:56	56	6x5	Neu	Tripping(2	OTT	Neu	OTT	TOR	TYLER ENN	8474589	TREVOR M	8479675
11	9	20001	10/02/19	1	FAC	TOR won (	0:56	56	5x4	Off		TOR	Off	OTT	TOR	JOHN TAV	8475166	JEAN-GAB	8476419
12	10	20001	10/02/19	1	GIVE	TORÅ GIVI	1:04	64	5x4	Off		TOR	Off	OTT	TOR	MITCHELL	8478483		
13	11	20001	10/02/19	1	SHOT	TOR ONG(	1:31	91	5x4	Off	SNAP SHO	TOR	Off	OTT	TOR	MORGAN	8476853		
14	12	20001	10/02/19	1	STOP	GOALIE ST	1:31	91	5x4					OTT	TOR				
15	13	20001	10/02/19	1	FAC	OTT won I	1:31	91	5x4	Def		OTT	Off	OTT	TOR	CHRIS TIEI	8476919	JOHN TAV	8475166
16	14	20001	10/02/19	1	BLOCK	TOR #18 J(	1:39	99	5x4	Def	WRIST SH(	TOR	Off	OTT	TOR	MARK BOI	8474697	ANDREAS	8477341
17	15	20001	10/02/19	1	MISS	TOR #16 N	1:58	118	5x4	Off	WRIST SH(	TOR	Off	OTT	TOR	MITCHELL	8478483		
18	16	20001	10/02/19	1	HIT	OTT #2 DE	3:06	186	5x5	Def		OTT	Off	OTT	TOR	DYLAN DE	8476331	FREDERIK	8477512
19	17	20001	10/02/19	1	MISS	TOR #41 T	3:09	189	5x5	Off	TIP-IN	TOR	Off	OTT	TOR	DMYTRO T	8478857		
20	18	20001	10/02/19	1	SHOT	OTT ONG(	3:23	203	5x5	Off	SNAP SHO	OTT	Def	OTT	TOR	DYLAN DE	8476331		
21	19	20001	10/02/19	1	GIVE	TORÅ GIVI	3:24	204	5x5	Def		TOR	Def	OTT	TOR	JAKE MUZ	8474162		

PULLING DATA FROM THE NHL API

- Original plan: using the nhlscrapi package
  - This didn't work out, it only runs on Python 2.7
- Instead: the hockey-scraper package was used.
- Input the dates requested and receive a CSV of all the play by play data.

ADDING THE NHL RINK

- NHL Rink Layer file came from gisdummy on ArcGIS Online.
- Using geopandas and matplotlib, the rink was displayed.

DEALING WITH THAT MASSIVE SPREADSHEET

	Game_Id	Event	Ev_Team	Type	pl_name	xC	yC
11	20001	SHOT	TOR	SNAP SHOT	MORGAN RIELLY	-32.0	-2.0
21	20001	SHOT	TOR	WRIST SHOT	MORGAN RIELLY	-59.0	-20.0
26	20001	SHOT	TOR	SLAP SHOT	TYSON BARRIE	-42.0	-29.0
27	20001	SHOT	TOR	SLAP SHOT	TYSON BARRIE	-52.0	-7.0
29	20001	SHOT	TOR	WRIST SHOT	CODY CECI	-38.0	38.0
...	...	...	...	...	...	...	...
97033	20308	SHOT	TOR	WRIST SHOT	DMYTRO TIMASHOV	-6.0	-4.0
97041	20308	SHOT	TOR	WRIST SHOT	TRAVIS DERMOTT	-35.0	-28.0
97042	20308	SHOT	TOR	WRIST SHOT	FREDERIK GAUTHIER	-51.0	5.0
97045	20308	SHOT	TOR	SNAP SHOT	JOHN TAVARES	-68.0	-14.0
97053	20308	SHOT	TOR	WRIST SHOT	ANDREAS JOHNSSON	-53.0	-24.0

[671 rows x 7 columns]

```
data = pd.read_excel ('pbp.xlsx')
df = pd.DataFrame(data, columns= ['Game_Id', 'Event', 'Ev_Team', 'Type',
                                  'pl_name', 'xC', 'yC'])
```

- Columns I wanted:
  - Game ID, Event, Event Type, Event Team, Player Name, X and Y data
- Using pandas, the excel sheet was imported and filtered.

```
geometry = [Point(xy) for xy in zip(df_filtered['xC'], df_filtered['yC'])]
crs = {'epsg':'4269'}
shots = gpd.GeoDataFrame(df_filtered,
                         geometry=geometry,
                         crs=crs)
```

```
       Game_Id Event Ev_Team  ...     xC    yC                        geometry
11       20001  SHOT     TOR  ... -32.0  -2.0   POINT (-32.00000 -2.00000)
21       20001  SHOT     TOR  ... -59.0 -20.0  POINT (-59.00000 -20.00000)
26       20001  SHOT     TOR  ... -42.0 -29.0  POINT (-42.00000 -29.00000)
27       20001  SHOT     TOR  ... -52.0  -7.0   POINT (-52.00000 -7.00000)
29       20001  SHOT     TOR  ... -38.0  38.0   POINT (-38.00000 38.00000)
...        ...   ...     ...  ...   ...   ...                          ...
97033    20308  SHOT     TOR  ...  -6.0  -4.0    POINT (-6.00000 -4.00000)
97041    20308  SHOT     TOR  ... -35.0 -28.0  POINT (-35.00000 -28.00000)
97042    20308  SHOT     TOR  ... -51.0   5.0    POINT (-51.00000 5.00000)
97045    20308  SHOT     TOR  ... -68.0 -14.0  POINT (-68.00000 -14.00000)
97053    20308  SHOT     TOR  ... -53.0 -24.0  POINT (-53.00000 -24.00000)

[671 rows x 8 columns]
```

CONVERTING TO GEOMETRY

- Used Shapely to take the columns of x and y data to add geometry.

WE DID IT KIDS