

In [20]:

```
import pandas as pd
from shapely.geometry import Point
from shapely.geometry.polygon import Polygon
import numpy as np
```

In [21]:

```
polygon = Polygon([(-1735, 250), (-2024, 398), (-2806, 742), (-2472, 1233), (-1565, 580)
```

In [22]:

```
p = Point(-1310, -3332)
```

In [23]:

```
polygon.contains(p)
```

Out[23]:

False

In [24]:

```
pd.set_option('display.max_colwidth', None)
```

In [25]:

```
data = pd.read_parquet('C:\\Users\\vedbh\\Downloads\\Evil Genuises\\data', engine = 'pya
```

In [26]:

```
data.columns
```

Out[26]:

```
Index(['round_num', 'tick', 'side', 'team', 'hp', 'armor', 'is_alive',
      'x',
      'y', 'z', 'inventory', 'total_utility',
      'equipment_value_freezetime_end', 'area_name', 'seconds', 'clock_ti
me',
      't_alive', 'ct_alive', 'bomb_planted', 'map_name', 'utility_used',
      'player'],
      dtype='object')
```

In [27]:

```
data.head()
```

Out[27]:

	round_num	tick	side	team	hp	armor	is_alive	x	y	z	...	equipment_va
0	1	2511	T	Team1	100	100	True	-1310	-3333	294	...	
1	1	2527	T	Team1	100	100	True	-1303	-3322	294	...	
2	1	2543	T	Team1	100	100	True	-1290	-3303	292	...	
3	1	2559	T	Team1	100	100	True	-1275	-3280	290	...	
4	1	2575	T	Team1	100	100	True	-1254	-3262	288	...	

5 rows × 22 columns

In [28]:

```
team2T = data[(data['team'] == 'Team2') & (data['side'] == "T") & (data['area_name'] ==  
team2TW = team2T[(~team2T['inventory'].isnull())]
```

In [29]:

```
team2TW
```

Out[29]:

	round_num	tick	side	team	hp	armor	is_alive	x	y	z	...	equipme
77956	16	293384	T	Team2	100	100	True	-724	-3	50	...	
77957	16	293400	T	Team2	100	100	True	-736	25	56	...	
77958	16	293416	T	Team2	100	100	True	-746	53	61	...	
77959	16	293432	T	Team2	100	100	True	-756	82	66	...	
77960	16	293448	T	Team2	100	100	True	-769	109	73	...	
...	
220038	28	484524	T	Team2	72	100	True	-1131	320	51	...	
220039	28	484540	T	Team2	72	100	True	-1143	321	45	...	
220045	28	484636	T	Team2	72	100	True	-1142	331	45	...	
220046	28	484652	T	Team2	72	100	True	-1137	352	48	...	
220047	28	484668	T	Team2	72	100	True	-1132	375	54	...	

859 rows × 22 columns

In [34]:

carrying

Out[34]:

[]

In [37]:

```

class ProcessGameState:
    def __init__(self, dataPath):
        self.data = pd.read_parquet(path = dataPath, engine = 'pyarrow')
#         self.preProcess()
        self.weaponsData = None
        self.weapons = set()
        self.getWeapons()
        self.blue = Polygon([(-1735, 250), (-2024, 398), (-2806, 742), (-2472, 1233), (-
        self.z = (285, 481)

    def getWeapons(self):
        self.weaponsData = self.data[~self.data['inventory'].isnull()]
        for i in self.weaponsData['inventory']:
            for j in i:
                self.weapons.add(j.get('weapon_class', None))

    def getWeaponsClass(self):
        return self.weapons

    def strategy(self):
        self.team2T = self.data[(self.data['team'] == 'Team2') & (self.data['side'] == "
        c = 0
        for (r, (x, y, z)) in self.team2T[['x', 'y', 'z']].iterrows():
            p = Point(x, y)
            if self.blue.contains(p) and self.z[0] <= z <= self.z[1]:
                c += 1
        return c / self.team2T.shape[0] > 0.5

    def getTime(self):
        self.team2TW = self.team2T[self.team2T['area_name'] == 'BombsiteB']
        self.team2TW = self.team2TW[~self.team2TW['inventory'].isnull()]
        carrying = []
        for i, v in self.team2TW.groupby('round_num'):
            rc, smgc = 0, 0
            c = []
            for j, z in v.iterrows():
                c.append(j)
                for w in z['inventory']:
                    if w['weapon_class'] == 'Rifle':
                        rc += 1
                    elif w['weapon_class'] == 'SMG':
                        smgc += 1
            if rc > 1 or smgc > 1:
                carrying.extend(c)
        self.filteredT2 = self.team2TW.loc[carrying]
        return self.filteredT2['seconds'].mean()

```

In [38]:

```
pgs = ProcessGameState('C:\\Users\\vedbh\\Downloads\\Evil Genuises\\data')
```

In [39]:

```
# 1c  
pgs.getWeaponsClass()
```

Out[39]:

```
{'Grenade', 'Pistols', 'Rifle', 'SMG'}
```

In [40]:

```
# 2a  
pgs.strategy()
```

Out[40]:

```
False
```

In [41]:

```
# 2b  
pgs.getTime()
```

Out[41]:

```
24.95777777777778
```