

# Pressure Patterns code

August 10, 2020

```
[1]: import pandas as pd
import numpy as np
from ast import literal_eval
import plotly.express as px
import plotly.graph_objects as go

from PIL import Image

[2]: nalcs = pd.read_csv("nalcssummer2020.csv")

[3]: eg = nalcs[nalcs['team']=='eg']

[4]: mid_points_b = []
sup_points_b = []
top_points_b = []
adc_points_b = []
move_in_points_b = []
counts_b = np.array([])

mid_points_r = []
sup_points_r = []
top_points_r = []
adc_points_r = []
move_in_points_r = []
counts_r = np.array([])

for n in [1, 11, 16, 21, 26, 31, 36, 41, 46, 51, 56, 61, 66]:
    game = eg.iloc[n]

    if(game['side'] == "red"):
        count = 0
        pressure = False
        for i in range(20,1200):
            try:
                point = literal_eval(game[i])
                if(point[0] > (5/4)*point[1]):
```

```

        if(count > 10 and not pressure):
            pressure = True
            counts_r = np.append(counts_r, count)
            count = 0
        elif(not pressure):
            pressure = not pressure
            count = 0
        elif(pressure):
            count += 1
    else:
        if(pressure and count > 10):
            top_points_r.append(literal_eval(eg.iloc[n-1][i]))
            mid_points_r.append(literal_eval(eg.iloc[n+1][i]))
            adc_points_r.append(literal_eval(eg.iloc[n+2][i]))
            sup_points_r.append(literal_eval(eg.iloc[n+3][i]))
            move_in_points_r.append(point)
            pressure = False
            count = 0
        elif(pressure):
            pressure = not pressure
            count = 0
        elif(not pressure):
            count += 1
    cv2.imshow('minimap', lcsmap)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
except:
    pass
else:
    count = 0
    pressure = False
    for i in range(20, 1200):
        try:
            point = literal_eval(game[i])
            if(point[0] > (5/4)*point[1]-25):
                if(count > 10 and not pressure):
                    if(point[0] < (5/4)*point[1]+30):
                        move_in_points_b.append(point)
                        top_points_b.append(literal_eval(eg.iloc[n-1][i]))
                        mid_points_b.append(literal_eval(eg.iloc[n+1][i]))
                        adc_points_b.append(literal_eval(eg.iloc[n+2][i]))
                        sup_points_b.append(literal_eval(eg.iloc[n+3][i]))
                        pressure = True
                        count = 0
                    elif(pressure):
                        pressure = not pressure
                        count = 0

```

```

        elif(not pressure):
            count += 1
    else:
        if(count > 10 and not pressure):
            pressure = True
            counts_b = np.append(counts_b, count)
            count = 0
        elif(not pressure):
            pressure = not pressure
            count = 0
        elif(pressure):
            count += 1
except:
    pass

```

```

[5]: if(point[0] > (5/4)*point[1]-25):
        if(count > 10 and pressure):
            if(point[0] < (5/4)*point[1]+30):
                move_in_points_b.append(point)
                top_points_b.append(literal_eval(eg.iloc[n-1][i]))
                mid_points_b.append(literal_eval(eg.iloc[n+1][i]))
                adc_points_b.append(literal_eval(eg.iloc[n+2][i]))
                sup_points_b.append(literal_eval(eg.iloc[n+3][i]))
            pressure = False
            count = 0
        elif(pressure):
            pressure = not pressure
            count = 0
        elif(not pressure):
            count += 1

```

```

[6]: mid_points_b_2 = [i for i in mid_points_b if i[0] < 155-i[1]+40 and i[0] >
    ↪155-i[1]-40]
axes_mid = list(zip(*mid_points_b_2))

adc_points_b_2 = [i for i in adc_points_b if i[0] > 155-i[1]+60]
axes_adc = list(zip(*adc_points_b_2))

top_points_b_2 = [i for i in top_points_b if i[0] < 155-i[1]-90]
axes_top = list(zip(*top_points_b_2))

jgl_points_b_2 = [i for i in move_in_points_b if i[0] < i[1]+10]
axes_jgl = list(zip(*jgl_points_b_2))

```

```

[7]: fig = go.Figure()

fig.add_trace(go.Histogram2dContour(

```

```

        x = axes_top[0],
        y = axes_top[1],
        colorscale = ['rgba(0,255,0,0)', 'white'],
        name = "Top"
    ))

fig.add_trace(go.Histogram2dContour(
    x = axes_mid[0],
    y = axes_mid[1],
    colorscale = ['rgba(0,0,255,0)', 'white'],
    name="Mid"
))

fig.add_trace(go.Histogram2dContour(
    x = axes_adc[0],
    y = axes_adc[1],
    colorscale = ['rgba(255,0,0,0)', 'white'],
    name = "ADC"
))

fig.update_xaxes(range=[0,155])
fig.update_yaxes(range=[153,0])

fig.add_layout_image(
    dict(
        source=Image.open("../loltracker/assets/lcs/lcs.png"),
        xref="x",
        yref="y",
        x=0,
        y=0,
        sizex = 155,
        sizey = 153,
        sizing="stretch",
        opacity=0.8,
        layer="below"))

fig.update_layout(
    title = "EG: Positions when jungler pushes past halfway point: Blue side",
    template = "plotly_white",
    xaxis_showgrid = False,
    yaxis_showgrid = False,
    height = 800,
    width = 800
)

fig.update_traces(showlegend=True, showscale=False)

```

```

fig.update_xaxes(showticklabels = False, title_text = "")
fig.update_yaxes(showticklabels = False, title_text = "")

fig.show()

```

```

[8]: fig = go.Figure()

fig.add_trace(go.Histogram2dContour(
    x = axes_jgl[0],
    y = axes_jgl[1],
    colorscale = ['rgba(0,0,255,0)', 'white']
))

fig.update_xaxes(range=[0,155])
fig.update_yaxes(range=[153,0])

fig.add_layout_image(
    dict(
        source=Image.open("../loltracker/assets/lcs/lcs.png"),
        xref="x",
        yref="y",
        x=0,
        y=0,
        sizex = 155,
        sizey = 153,
        sizing="stretch",
        opacity=0.7,
        layer="below"))

fig.update_layout(
    title = "EG: Positions where jungler pushes past halfway point: Blue side",
    template = "plotly_white",
    xaxis_showgrid = False,
    yaxis_showgrid = False,
    height = 800,
    width = 800
)

fig.update_traces(showlegend=False, showscale=False)

fig.update_xaxes(showticklabels = False, title_text = "")
fig.update_yaxes(showticklabels = False, title_text = "")

fig.show()

```

```
[9]: top_points_r_2 = [i for i in top_points_r if i[0] < 155-i[1]-90]
axes_top = list(zip(*top_points_r_2))

jgl_points_r_2 = [i for i in move_in_points_r if i[0] > i[1]-10]
axes_jgl = list(zip(*jgl_points_r_2))

adc_points_r_2 = [i for i in adc_points_r if i[0] > 155-i[1]+60]
axes_adc = list(zip(*adc_points_r_2))

mid_points_r_2 = [i for i in mid_points_r if i[0] < 155-i[1]+40 and i[0] > ↵
↵155-i[1]-40]
axes_mid = list(zip(*mid_points_r_2))
```

```
[10]: fig = go.Figure()

fig.add_trace(go.Histogram2dContour(
    x = axes_top[0],
    y = axes_top[1],
    colorscale = ['rgba(0,255,0,0)', 'white'],
    name = "Top"
))

fig.add_trace(go.Histogram2dContour(
    x = axes_mid[0],
    y = axes_mid[1],
    colorscale = ['rgba(0,0,255,0)', 'white'],
    name="Mid"
))

fig.add_trace(go.Histogram2dContour(
    x = axes_adc[0],
    y = axes_adc[1],
    colorscale = ['rgba(255,0,0,0)', 'white'],
    name = "ADC"
))

fig.update_xaxes(range=[0,155])
fig.update_yaxes(range=[153,0])

fig.add_layout_image(
    dict(
        source=Image.open("../loltracker/assets/lcs/lcs.png"),
        xref="x",
        yref="y",
        x=0,
        y=0,
```

```

        sizex = 155,
        sizey = 153,
        sizing="stretch",
        opacity=0.8,
        layer="below"))

fig.update_layout(
    title = "EG: Positions when jungler pushes past halfway point: Red side",
    template = "plotly_white",
    xaxis_showgrid = False,
    yaxis_showgrid = False,
    height = 800,
    width = 800
)

fig.update_traces(showlegend=True, showscale=False)

fig.update_xaxes(showticklabels = False, title_text = "")
fig.update_yaxes(showticklabels = False, title_text = "")

fig.show()

```