

```
In [1]: #Import libraries
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib
import matplotlib.pyplot as plt
import folium
import webbrowser
from IPython.display import set_matplotlib_formats
from matplotlib import rcParams
import warnings
```

```
In [2]: warnings.filterwarnings('ignore')
```

```
In [3]: df_costco=pd.read_csv("costcos-geocoded.csv")
df_ppg=pd.read_csv("ppg2008.csv",index_col=0)
```

```
In [4]: df_costco.head()
```

Out[4]:

	Address	City	State	Zip Code	Latitude	Longitude
0	1205 N. Memorial Parkway	Huntsville	Alabama	35801-5930	34.743095	-86.600955
1	3650 Galleria Circle	Hoover	Alabama	35244-2346	33.377649	-86.812420
2	8251 Eastchase Parkway	Montgomery	Alabama	36117	32.363889	-86.150884
3	5225 Commercial Boulevard	Juneau	Alaska	99801-7210	58.359200	-134.483000
4	330 West Dimond Blvd	Anchorage	Alaska	99515-1950	61.143266	-149.884217

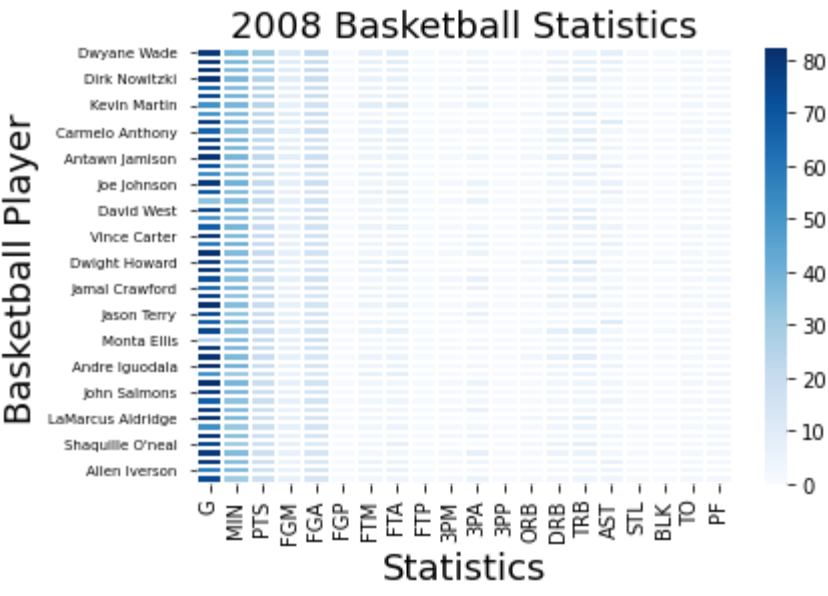
```
In [5]: df_ppg.head()
```

Out[5]:

	G	MIN	PTS	FGM	FGA	FGP	FTM	FTA	FTP	3PM	3PA	3PP	ORB	DRB	TRB	AST	STL	BLK	TO	PF
Name																				
Dwyane Wade	79	38.6	30.2	10.8	22.0	0.491	7.5	9.8	0.765	1.1	3.5	0.317	1.1	3.9	5.0	7.5	2.2	1.3	3.4	2.3
LeBron James	81	37.7	28.4	9.7	19.9	0.489	7.3	9.4	0.780	1.6	4.7	0.344	1.3	6.3	7.6	7.2	1.7	1.1	3.0	1.7
Kobe Bryant	82	36.2	26.8	9.8	20.9	0.467	5.9	6.9	0.856	1.4	4.1	0.351	1.1	4.1	5.2	4.9	1.5	0.5	2.6	2.3
Dirk Nowitzki	81	37.7	25.9	9.6	20.0	0.479	6.0	6.7	0.890	0.8	2.1	0.359	1.1	7.3	8.4	2.4	0.8	0.8	1.9	2.2
Danny Granger	67	36.2	25.8	8.5	19.1	0.447	6.0	6.9	0.878	2.7	6.7	0.404	0.7	4.4	5.1	2.7	1.0	1.4	2.5	3.1

Heat Map

```
In [6]: sns.heatmap(df_ppg,cmap='Blues',linewidth=0.30)
plt.xticks(rotation=90)
plt.yticks(fontsize=7)
plt.xlabel("Statistics",size=18)
plt.ylabel("Basketball Player",size=18)
plt.title("2008 Basketball Statistics",size=18)
plt.show()
```



Spatial Plot

```
In [7]: map=folium.Map(prefer_canvas=True)

def plot(point):
    folium.CircleMarker(location=[point.Latitude,point.Longitude],
                        radius=2, popup=point.Address+" "+point.City+" "+point.State+" "+point["Zip Code"],
                        fill_color=point.State,
                        fill=True,
                        fill_opacity=0.7,
                        weight=5).add_to(map)
```

```
In [8]: df_costco.apply(plot,axis=1)
map.fit_bounds(map.get_bounds())
output_file="Map_Python.html"
map.save(output_file)
webbrowser.open(output_file,new=2)
```

```
Out[8]: True
```

Contour Plot

```
In [9]: Z=df_ppg.pivot_table(index='FGA',columns='PTS',values='FGP').T.values
x_unique=np.sort(df_ppg.FGA.unique())
y_unique=np.sort(df_ppg.PTS.unique())
```

```
In [10]: X,Y=np.meshgrid(x_unique,y_unique)
Z[np.isnan(Z)]=0
pd.DataFrame(Z).round(3)
pd.DataFrame(Y).round(3)
pd.DataFrame(X).round(3)
```

Out[10]:

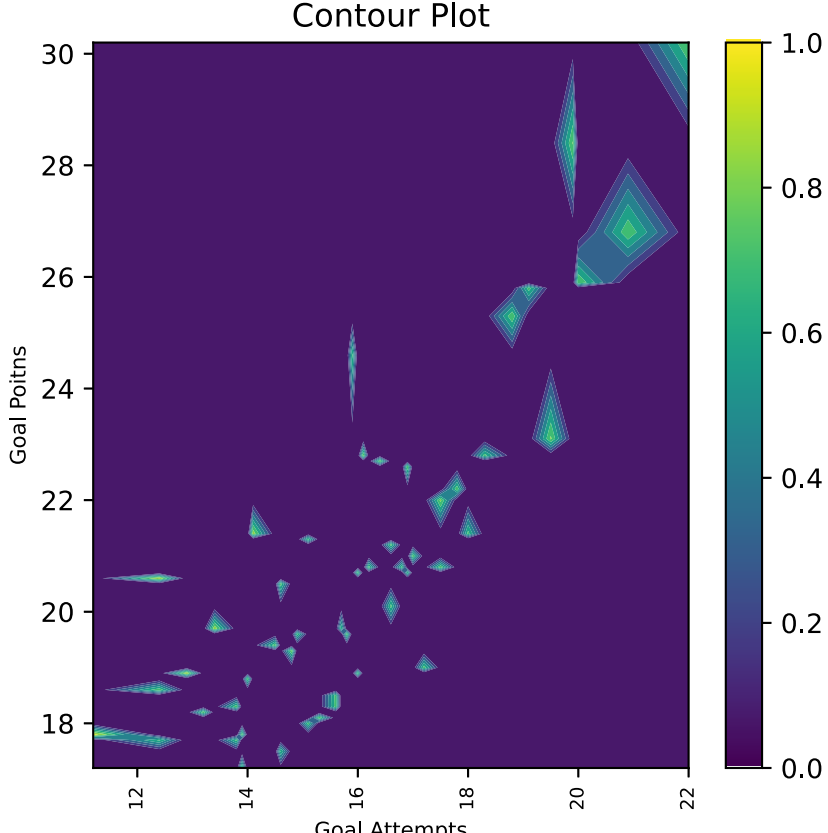
	0	1	2	3	4	5	6	7	8	9	...	29	30	31	32	33	34	35	36	37	38
0	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
1	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
2	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
3	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
4	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
5	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
6	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
7	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
8	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
9	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
10	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
11	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
12	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
13	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
14	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
15	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
16	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
17	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
18	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
19	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
20	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
21	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
22	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
23	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
24	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
25	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
26	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
27	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
28	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
29	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
30	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
31	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
32	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
33	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
34	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
35	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
36	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
37	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0
38	11.2	12.4	12.9	13.2	13.4	13.8	13.9	14.0	14.1	14.5	...	17.8	18.0	18.3	18.8	19.1	19.5	19.9	20.0	20.9	22.0

39 rows x 39 columns

```
In [11]: set_matplotlib_formats('svg')
rcParams['figure.figsize']=5,5
fig=plt.figure()
ax=fig.add_subplot(111)

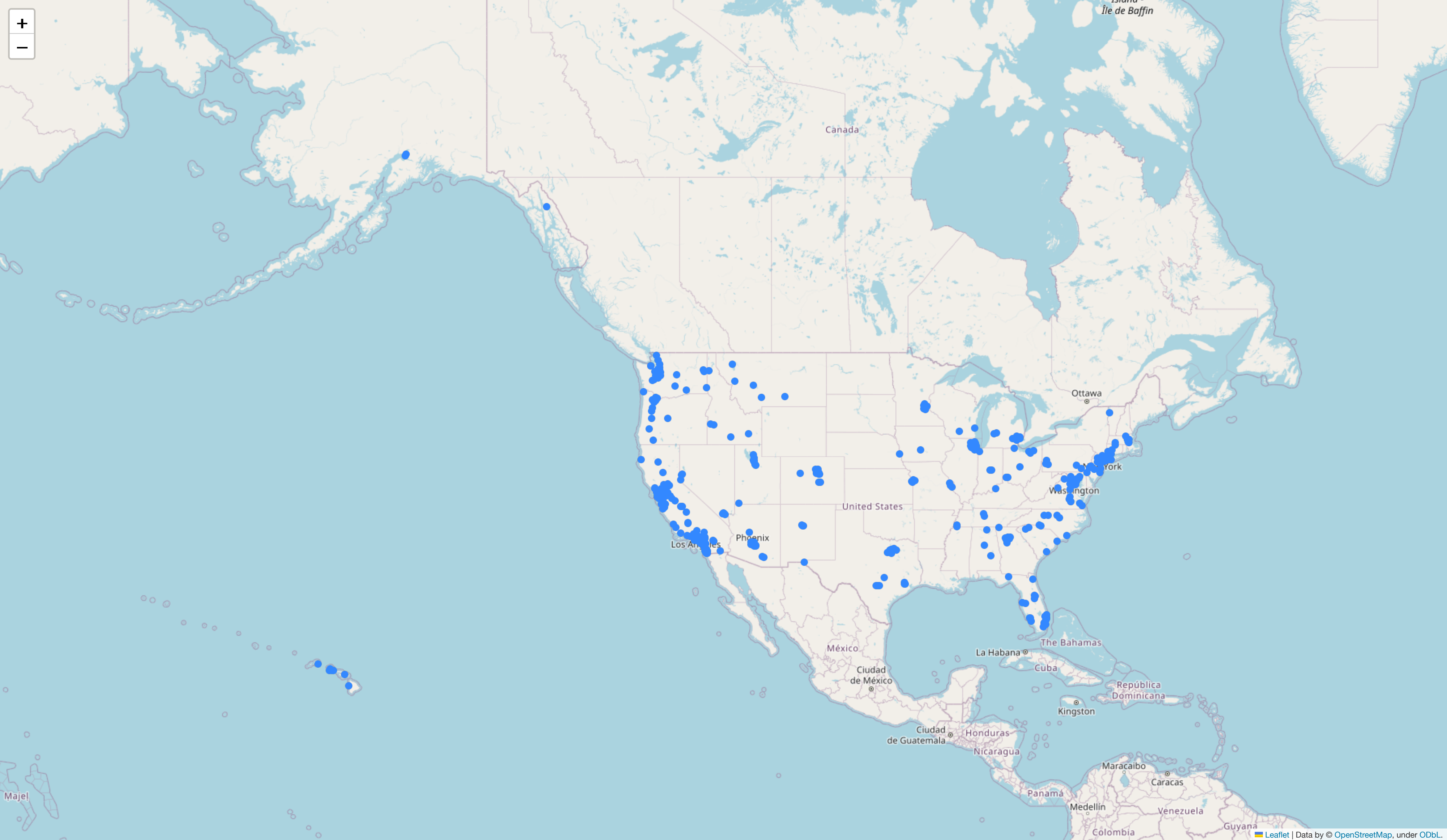
cp=ax.contourf(X,Y,Z)
ax=plt.gca()
PCM=ax.get_children()[2]
plt.colorbar(PCM,ax=ax)

plt.xticks(rotation=90)
plt.xticks(fontsize=7)
plt.xlabel('Goal Attempts',size=8)
plt.ylabel('Goal Poitns',size=8)
plt.title("Contour Plot")
plt.show()
```



```
In [ ]:
```

```
In [ ]:
```



Assignment_5.2_Vayuvegula_Soma_Shekar_R

Soma Shekar Vayuvegula

02/18/2023

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##   filter, lag
```

```
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

```
## — Attaching packages ————— tidyverse 1.3.2 —
## ✓ tibble 3.1.7 ✓ purrr 0.3.4
## ✓ tidyr 1.2.0 ✓ stringr 1.4.0
## ✓ readr 2.1.2 ✓ forcats 0.5.2
## — Conflicts ————— tidyverse_conflicts() —
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag() masks stats::lag()
##
## Attaching package: 'reshape2'
##
## The following object is masked from 'package:tidyr':
##
##   smiths
##
## Attaching package: 'data.table'
##
## The following objects are masked from 'package:reshape2':
##
##   dcast, melt
##
## The following object is masked from 'package:purrr':
##
##   transpose
##
## The following objects are masked from 'package:dplyr':
##
##   between, first, last
##
## Attaching package: 'plotly'
##
## The following object is masked from 'package:ggplot2':
##
##   last_plot
##
## The following object is masked from 'package:stats':
##
##   filter
##
## The following object is masked from 'package:graphics':
##
##   layout
##
## Attaching package: 'reshape'
##
## The following object is masked from 'package:plotly':
##
##   rename
##
## The following object is masked from 'package:data.table':
##
##   melt
##
## The following objects are masked from 'package:reshape2':
##
##   colsplit, melt, recast
##
## The following objects are masked from 'package:tidyr':
##
##   expand, smiths
##
## The following object is masked from 'package:dplyr':
##
##   rename
##
## -----
##
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
## -----
##
## Attaching package: 'plyr'
##
## The following objects are masked from 'package:reshape':
##
##   rename, round_any
##
## The following objects are masked from 'package:plotly':
##
##   arrange, mutate, rename, summarise
##
## The following object is masked from 'package:purrr':
##
##   compact
##
## The following objects are masked from 'package:dplyr':
##
##   arrange, count, desc, failwith, id, mutate, rename, summarise,
##   summarize
```

```
df_ppg<-read.csv("ppg2008.csv")
head(df_ppg,5)
```

	Name	G	MIN	PTS	FGM	FGA	FGP	FTM	FTA
	<chr>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	Dwyane Wade	79	38.6	30.2	10.8	22.0	0.491	7.5	9.8
2	LeBron James	81	37.7	28.4	9.7	19.9	0.489	7.3	9.4
3	Kobe Bryant	82	36.2	26.8	9.8	20.9	0.467	5.9	6.9
4	Dirk Nowitzki	81	37.7	25.9	9.6	20.0	0.479	6.0	6.7
5	Danny Granger	67	36.2	25.8	8.5	19.1	0.447	6.0	6.9

5 rows | 1-10 of 22 columns

```
df_costco<-read.csv("costcos-geocoded.csv")
head(df_costco,5)
```

	Address	City	State	Zip.Code	Latitude	Longitude
	<chr>	<chr>	<chr>	<chr>	<dbl>	<dbl>
1	11205 N. Memorial Parkway	Huntsville	Alabama	35801-5930	34.74309	-86.60096
2	3650 Galleria Circle	Hoover	Alabama	35244-2346	33.37765	-86.81242
3	8251 Eastchase Parkway	Montgomery	Alabama	36117	32.36389	-86.15088
4	5225 Commercial Boulevard	Juneau	Alaska	99801-7210	58.35920	-134.48300
5	330 West Dimond Blvd	Anchorage	Alaska	99515-1950	61.14327	-149.88422

5 rows

```
library(scales)
```

```
##
## Attaching package: 'scales'
```

```
## The following object is masked from 'package:purrr':
##
##   discard
```

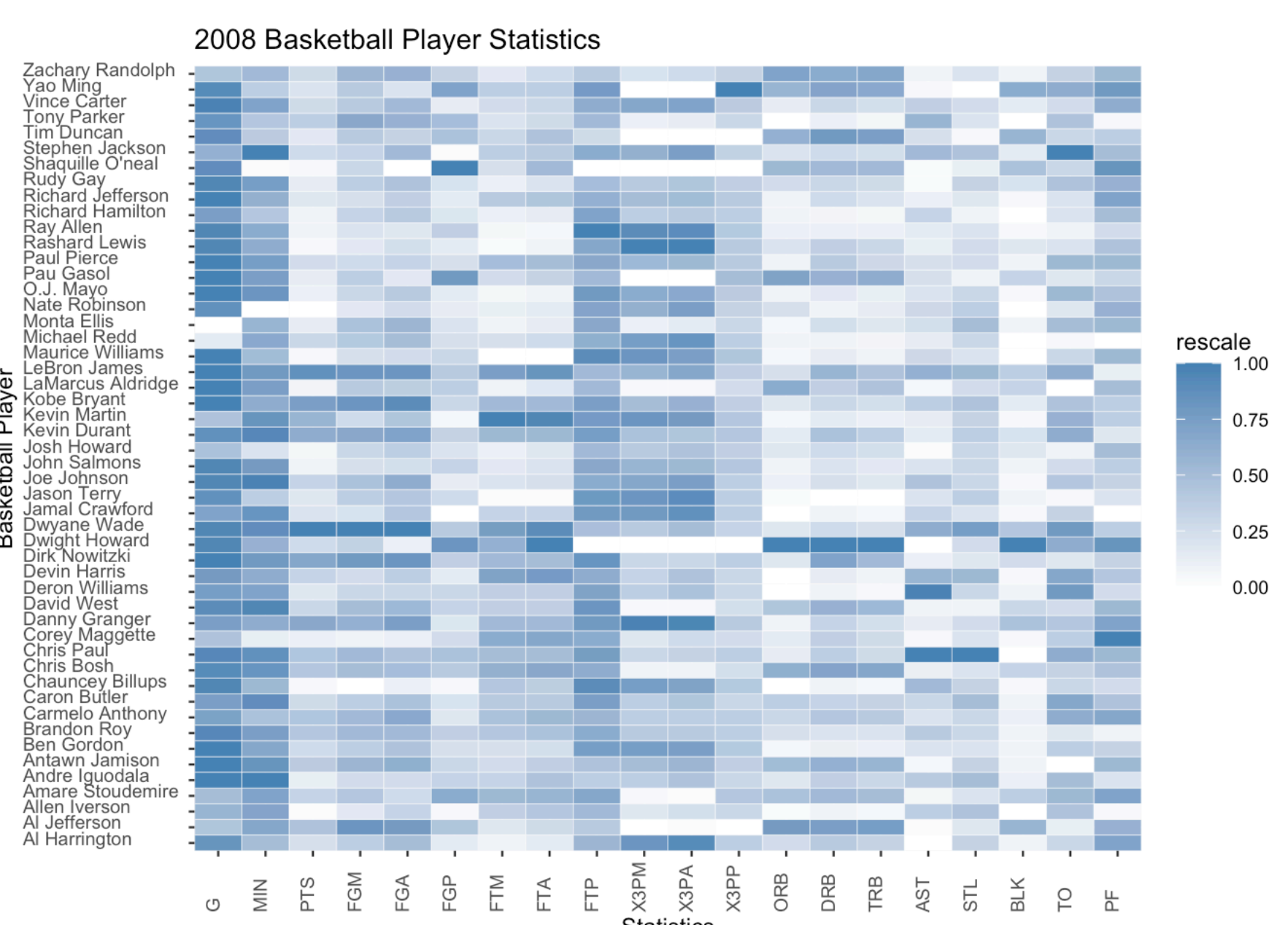
```
## The following object is masked from 'package:readr':
##
##   col_factor
```

```
df_melt<-melt(df_ppg)
```

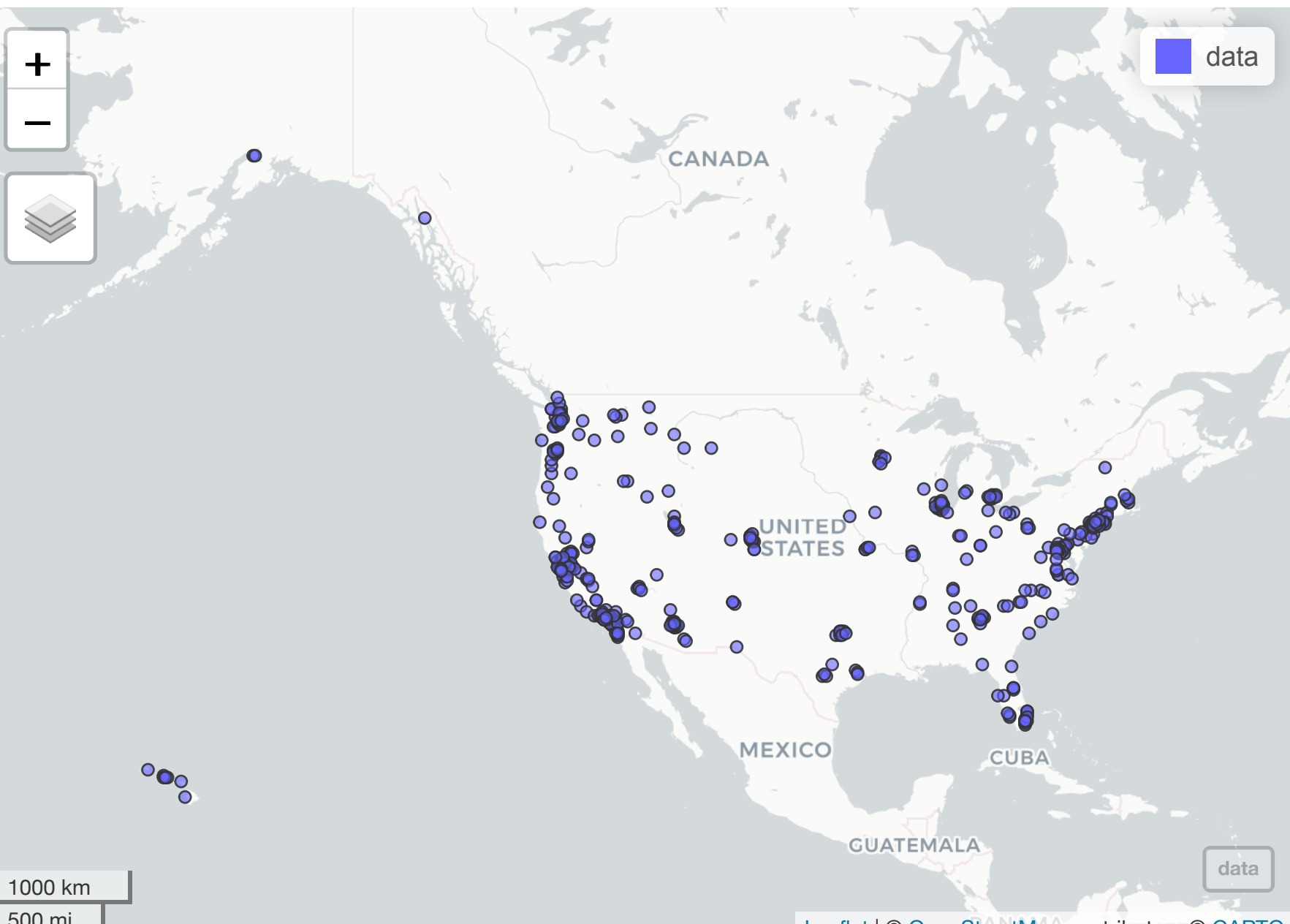
```
## Using Name as id variables
```

```
df_melt<-ddply(df_melt, .(variable),transform,rescale=rescale(value))
base_size<-9

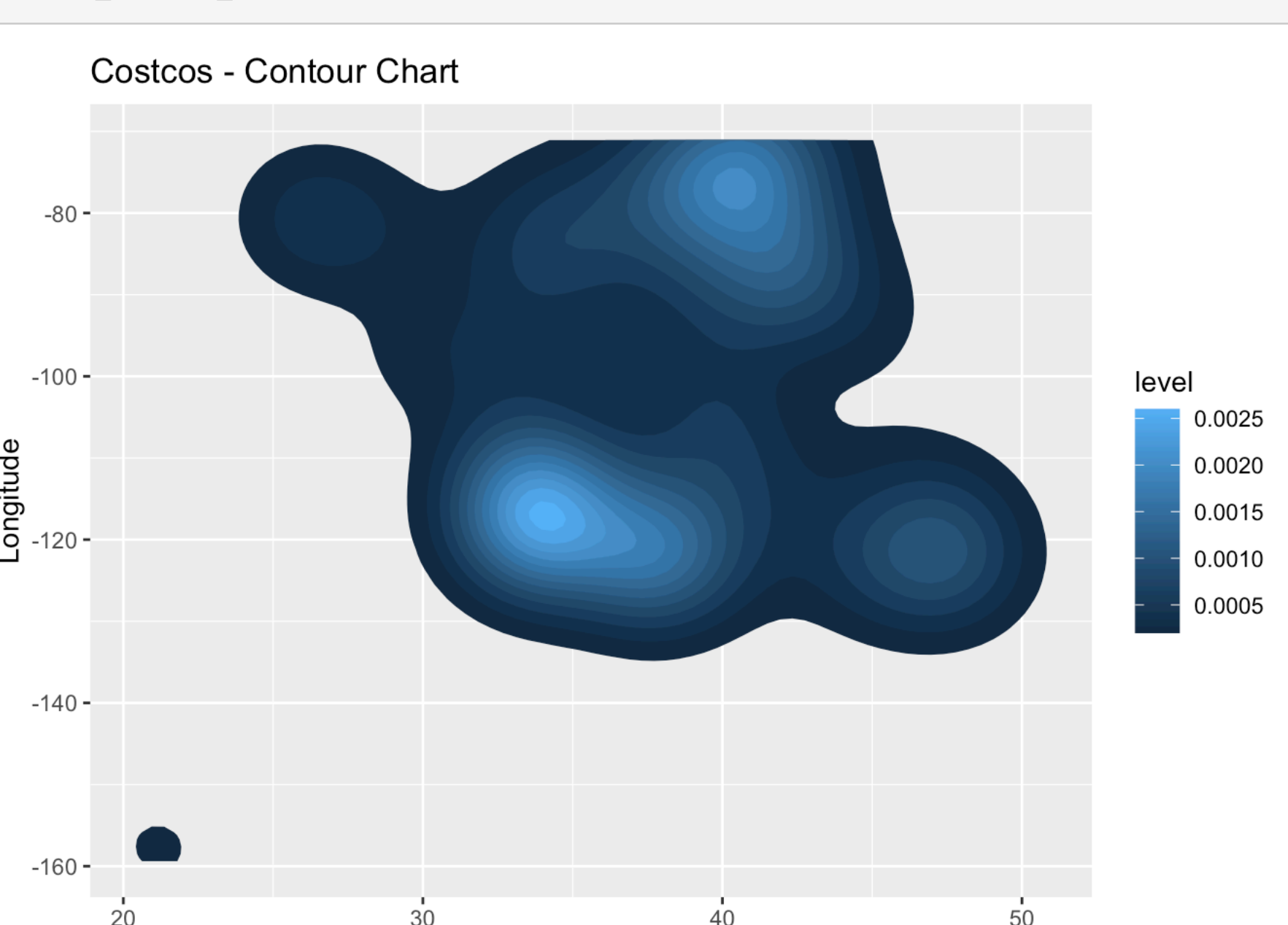
ggp <- ggplot(df_melt, aes(variable,Name))+geom_tile(aes(fill=rescale),color='white')+scale_fill_gradient(low="white",high = "steelblue")+ theme_gray(base_size = base_size) + labs(x = "", y = "") + scale_x_discrete(expand = c(0, 0)) +
  scale_y_discrete(expand = c(0, 0)) +
  theme(axis.text.x=element_text(angle=90, hjust=0, vjust= 0.1)) +
  theme(axis.text.y=element_text(hjust=0, vjust= 0.1)) +
  ggtitle("2008 Basketball Player Statistics") +
  xlab("Statistics") +
  ylab("Basketball Player")
g9p
```



```
mapview(df_costco, xcol = "Longitude", ycol = "Latitude",cex = 3,crs = 4269, grid = FALSE)
```



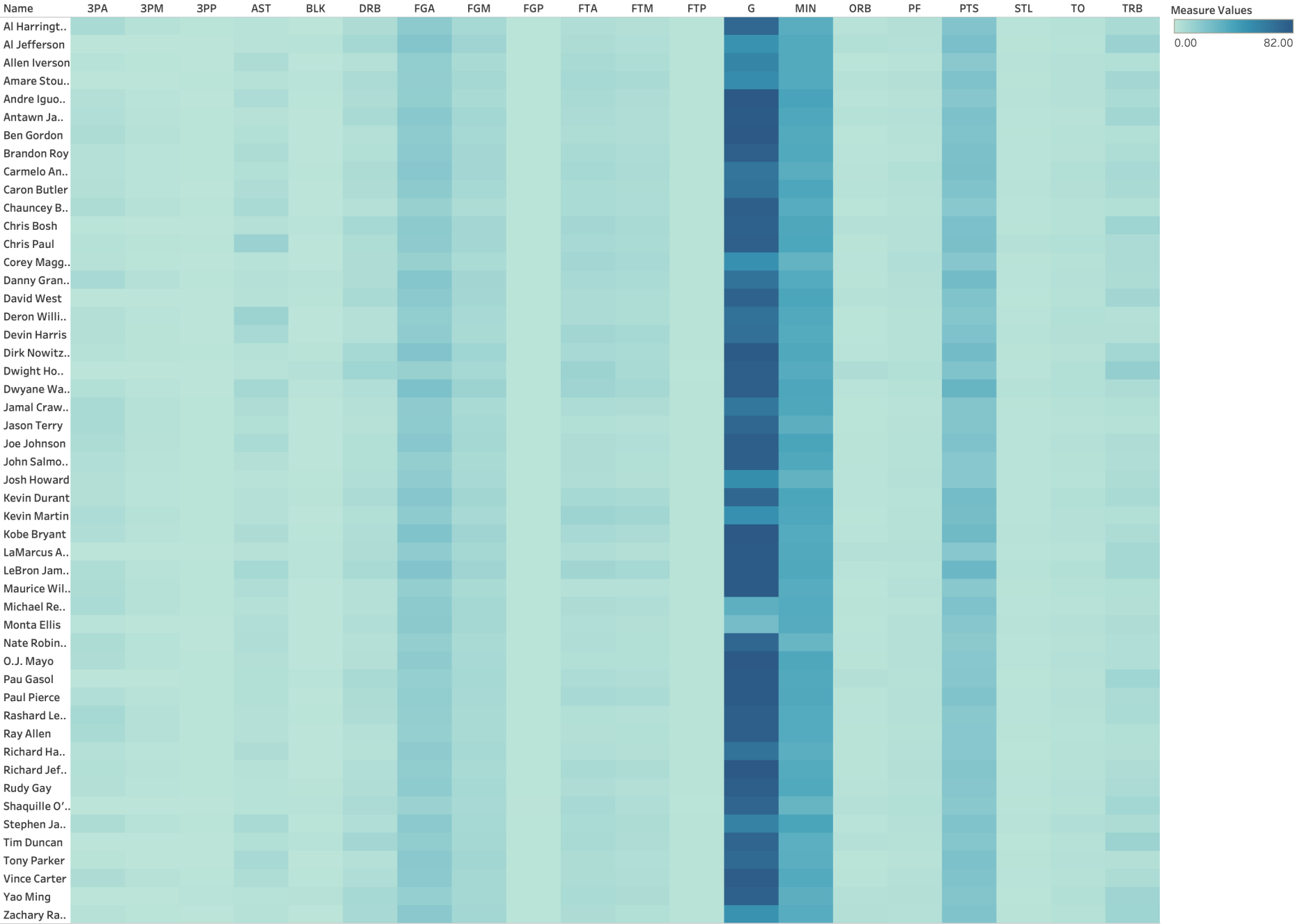
```
ggplot(df_costco, aes(x = Longitude, y = Longitude, fill = ..level..)) +
  stat_density_2d(geom = "polygon") + ggtitle("Costcos - Contour Chart")
```



Assignment_5.2_Vayuvegula _Soma_Shekar_Tableau

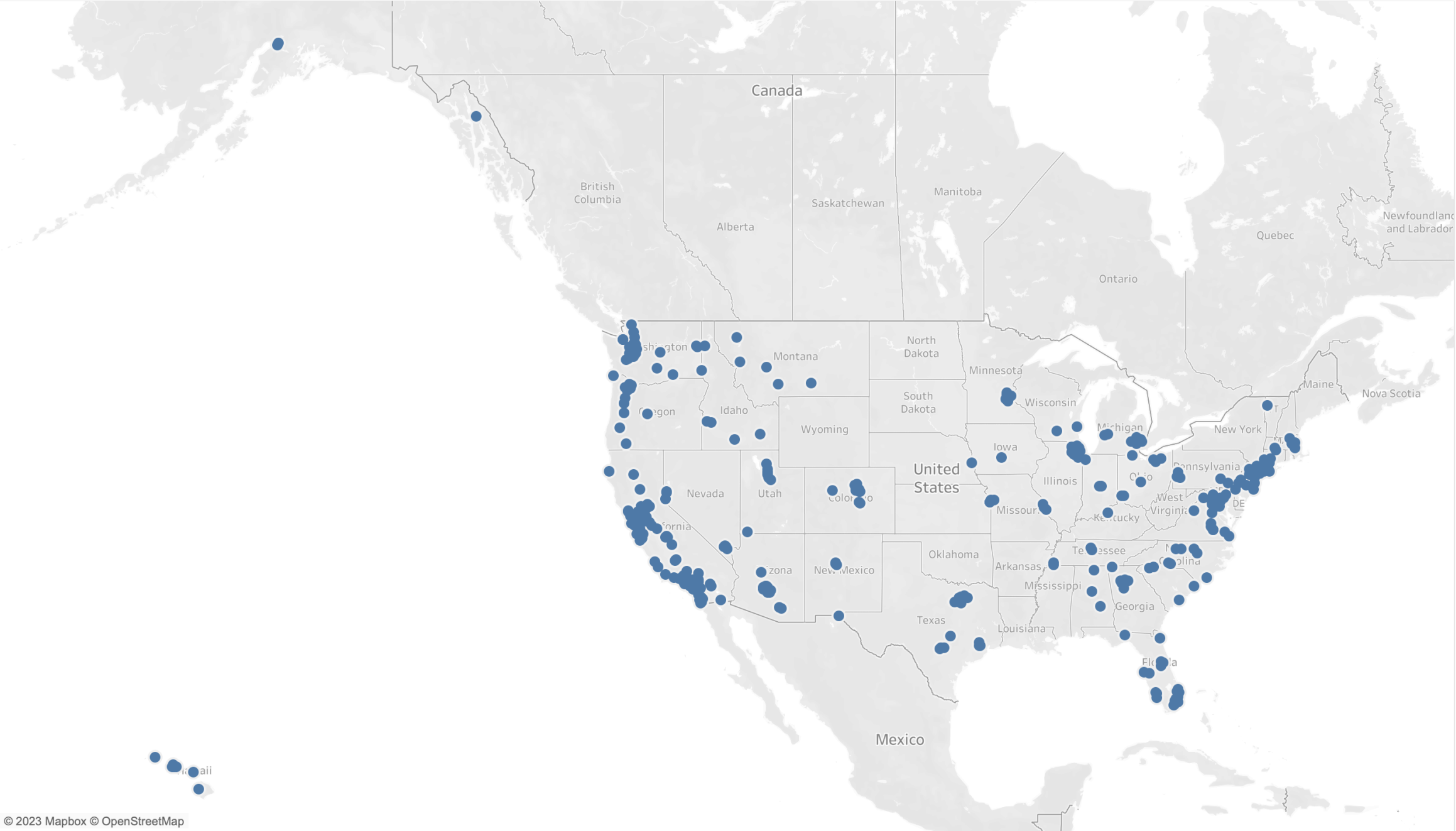
File created on: 2/19/23 2:58:01 AM CST

Heat Map

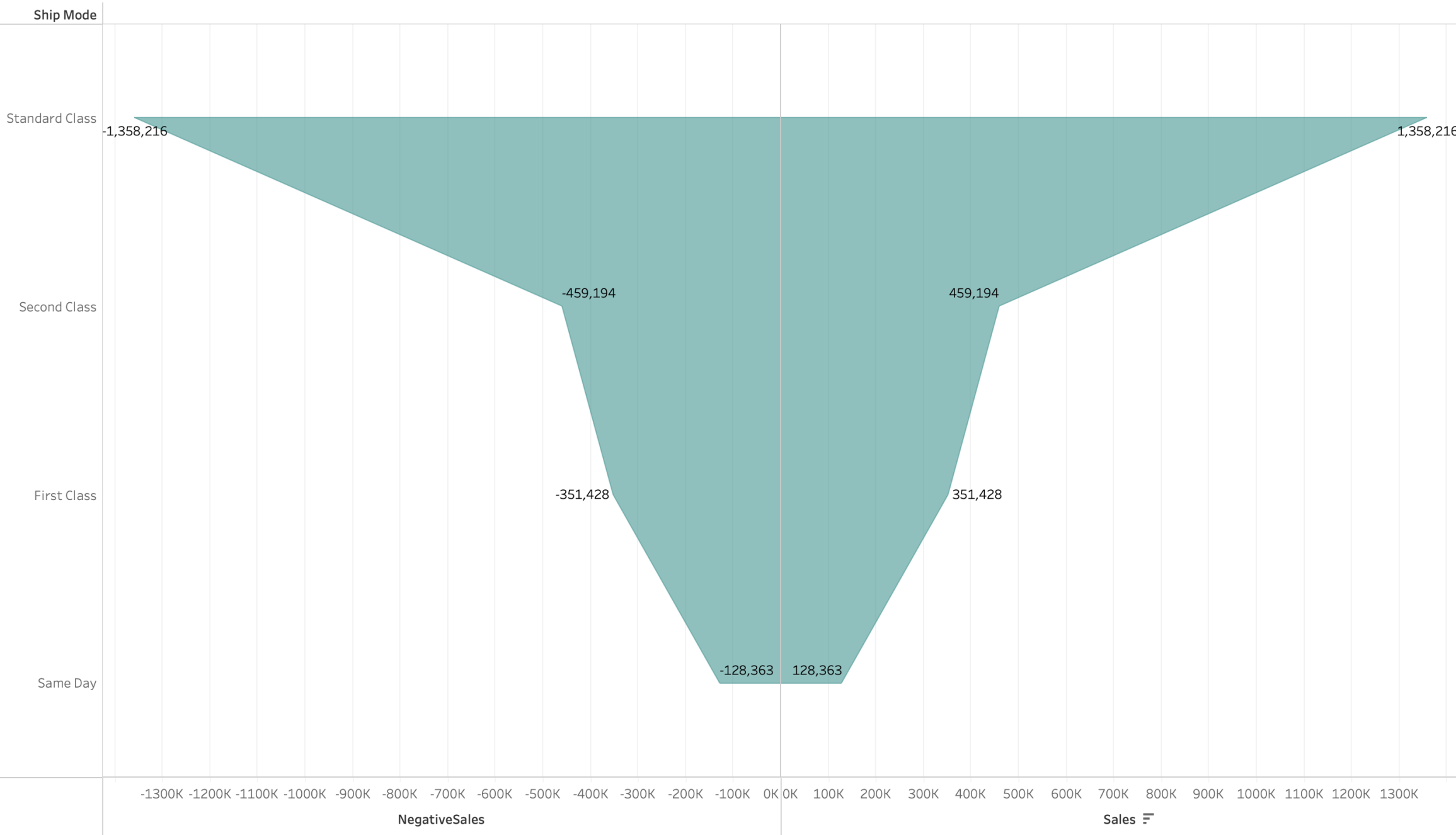


3PA, 3PM, 3PP, AST, BLK, DRB, FGA, FGM, FGP, FTA, FTM, FTP, G, MIN, ORB, PF, PTS, STL, TO and TRB (color) broken down by Name.

Spatial Chart



Funnel Chart



Sum of NegativeSales and sum of Sales for each Ship Mode.