Name: Khanh Tran
INFO 250-001
Date: July 4, 2020

Assignment 1

The Dataset

RAW is a simple-to-use online tool that I was introduced to in this course. For this particular assignment, I chose the *Movies* dataset. Even though there were plenty more options but since the requirement was to use 4 graphing methods out of the 21 options available, and each one of them was designed for a specific type of statistical purpose or type of the dataset , I wanted to choose a dataset on which I could make the best use of the graphs. RAW was very helpful in supporting me with my decision-making process as it already categorized its datasets with the purposes and data types that corresponded to the categories of the graphing methods. This way, I would know exactly which methods will fit which datasets. Again, I needed to use 4 out of 21 graphing options, so I chose the *Movies* dataset – a *Dispersion* dataset – as there were 5 visualizing options for *Dispersion*. I did not go for the other datasets because of the lack of corresponding graphing methods for their types.

Before jumping into the visualizing part, I used Python to have a look at the dataset in the form of a *dataframe*.

Out[2]:

	Movie	Genre	Production Budget (millions)	Box Office (millions)	ROI	Rating IMDB
0	Avatar	Action	237	2784	11.7	8.0
1	The Blind Side	Drama	29	309	10.7	7.6
2	The Chronicles of Narnia: The Lion, the Witch ...	Adventure	180	745	4.1	6.9
3	The Dark Knight	Action	185	1005	5.4	9.0
4	ET: The Extra-Terrestrial	Drama	11	793	75.5	7.9
5	Finding Nemo	Adventure	94	940	10.0	8.1
6	Ghostbusters	Comedy	144	229	1.6	7.8
7	The Hunger Games	Thriller/Suspense	78	649	8.3	7.2
8	Iron Man 3	Action	178	1215	6.8	7.6
9	Jurassic Park	Action	53	1030	19.4	8.0
10	King Kong	Adventure	207	551	2.7	7.3
11	The Lion King	Adventure	45	968	21.5	8.4
12	Monsters, Inc.	Adventure	115	577	5.0	8.0
13	The Twilight Saga: New Moon	Drama	50	710	14.2	4.5
14	Oz the Great and Powerful	Adventure	160	493	3.1	6.6
15	Pirates of the Caribbean: Dead Man's Chest	Adventure	225	1066	4.7	7.3
16	Quantum of Solace	Action	200	586	2.9	6.7
17	Raiders of the Lost Ark	Adventure	18	390	21.7	8.7
18	Star Wars Ep. I: The Phantom Menace	Adventure	115	1027	8.9	6.5
19	Titanic	Thriller/Suspense	200	2187	10.9	7.6
20	Up	Adventure	175	735	4.2	8.3
21	The Vow	Drama	30	196	6.5	6.7
22	The War of the Worlds	Action	132	704	5.3	6.5
23	X-Men: The Last Stand	Action	210	459	2.2	6.8
24	You've Got Mail	Drama	65	251	3.9	6.3
25	Zookeeper	Romantic Comedy	80	170	2.1	5.0

Figure 1. Movies dataset

Look at Figure 1, this dataset has 26 rows and 6 columns, or in other words, there are 26 records and 6 variables in the dataset. *Movie* and *Genre* consist of strings while the rest of the variables are made up of numerical data as they are either money or rating points.

Visualization and Analytics

This part will be divided into two sections; one is about the process of mapping and customizing, and the another will be about comparing the pros and cons of the chosen methods.

- **Visualization**

    - The first choice for the visualization is what is called a *Convex Hull.* I had never heard of this type of graph before I started working on this assignment. According to the short introduction on RAW, *"In mathematics, the convex hull is the smallest convex shape containing a set of points. Applied to a scatterplot, it is useful to identify points belonging to the same category"*. So, the main idea is that this graph will be useful in visualizing the categorization of the data. The figure below is my version of the *Convex Hull* graph.

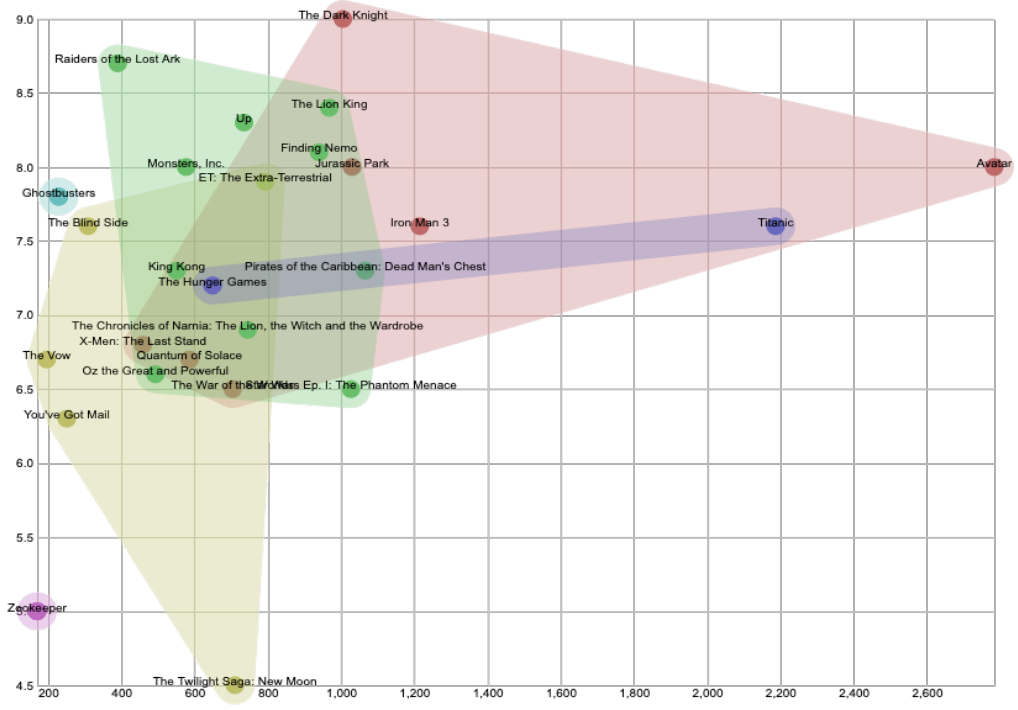

Figure 2. Convex Hull Graph

To produce Figure 2, I was required to use 4 variables for an *X Axis,* an *Y Axis, label, and Group* . I used *Box Office (millions)* as the x axis, *Rating IMDB* as the y axis, *Movie* for the labels, and *Genre* for the grouping. The basic idea was to categorize the movies into groups of genres on a 2-D dimension graph with film grossing on the horizontal axis and rating on the vertical axis. For this graph, I did not alter any of the customization parameters except for the *Dots Diameter.* The default size was 6, and to me it didn't specify the data point clearly enough. There were overlaps, especially in the 7-ish in IMDB rating, 700-ish-million-dollar area, where it was hard to figure out what movies fall into what groups, so I thought increasing the diameter would help prevent this issue.

- The second graph that I included was a *Contour Plot* graph. Again, this was a new term to me, and according to RAW, a *Contour Plot "shows the estimated density of point clouds, which is especially useful to avoid overplotting in large datasets"*. I did some research on my own and to my knowledge, a contour plot helped visualize a multiple-variable dataset in a 2-D dimension plot. Usually, it was used to visualize 3 variables in a 2-dimensional plane, and that was also what I did with RAW. The tool asked for an *X Axis,* an *Y Axis*, and *Label.* In order to have create a sense of similarity among different visualizing approach for the sake of graphs comparison later on, I kept *Box Office (millions), Rating IMDB,* and *Movie* for x axis, y axis, and label respectively.

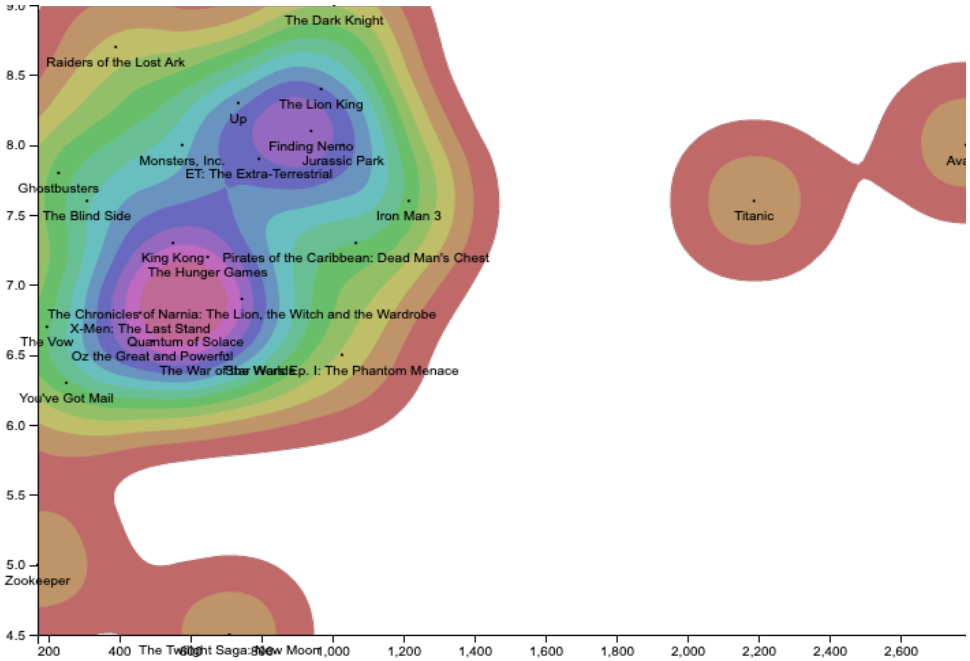

Figure 3. Contour Plot

I kept the default customization settings but did change *Colors Applied To* parameter from *"Stroke"* to *"Fill".* To me, the color-filled areas in the graph emphasized more efficiently the density of the cloud and did a better job in illustrating what movie titles falling into what level of density.

- The next method that I used was *Voronoi Tessellation.* This particular kind of graph had a very fancy name that I doubted it was even English. Obviously, I had never heard of it before using it for this assignment. However, its basic idea was not too difficult to understand. As RAW stated on their website, a *Voronoi Tessellation "creates the minimum area around each point defined by two variables. When applied to a scatterplot, it is useful to show the distance between points",* this type of visualization was useful to illustrate how far data points were from each other.

  The 3 variables required for this graph were the same as those in the two earlier approaches, and my picks were also the same. The customization setting was kept in default values.

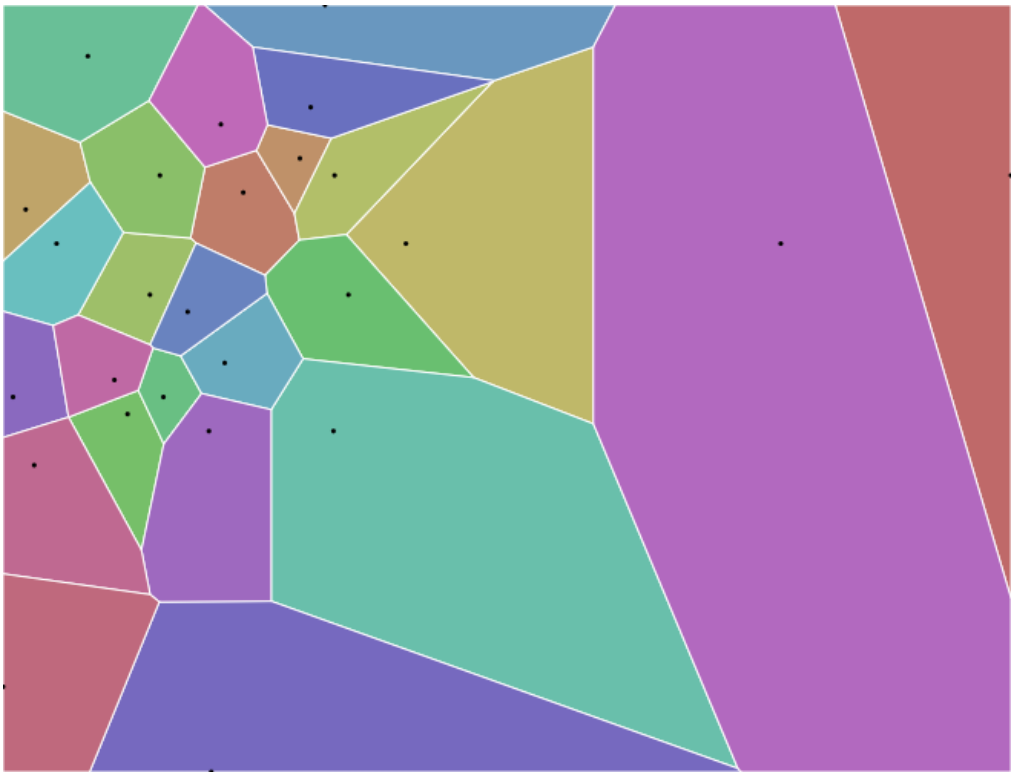

Figure 4. Voronoi Tessellation

- The last approach was to graph a *Scatter Plot.* This kind of graph sounded a lot more familiar to me as it was a popular plotting method in describing both 2-variable and multi-variable datasets. In this case, RAW required 5 variables: *X Axis, Y Axis, Size, Color,* and *Label.* Basically, the requirement resembled that of the *Convex Hull* approach but with an extra parameter, *Size.* As the result, for this particular approach, apart from using the same set of variables used in *Convex Hull*, I added *ROI* (Return on Investment) for sizing up the data points. Every other customization parameter was kept in default values.

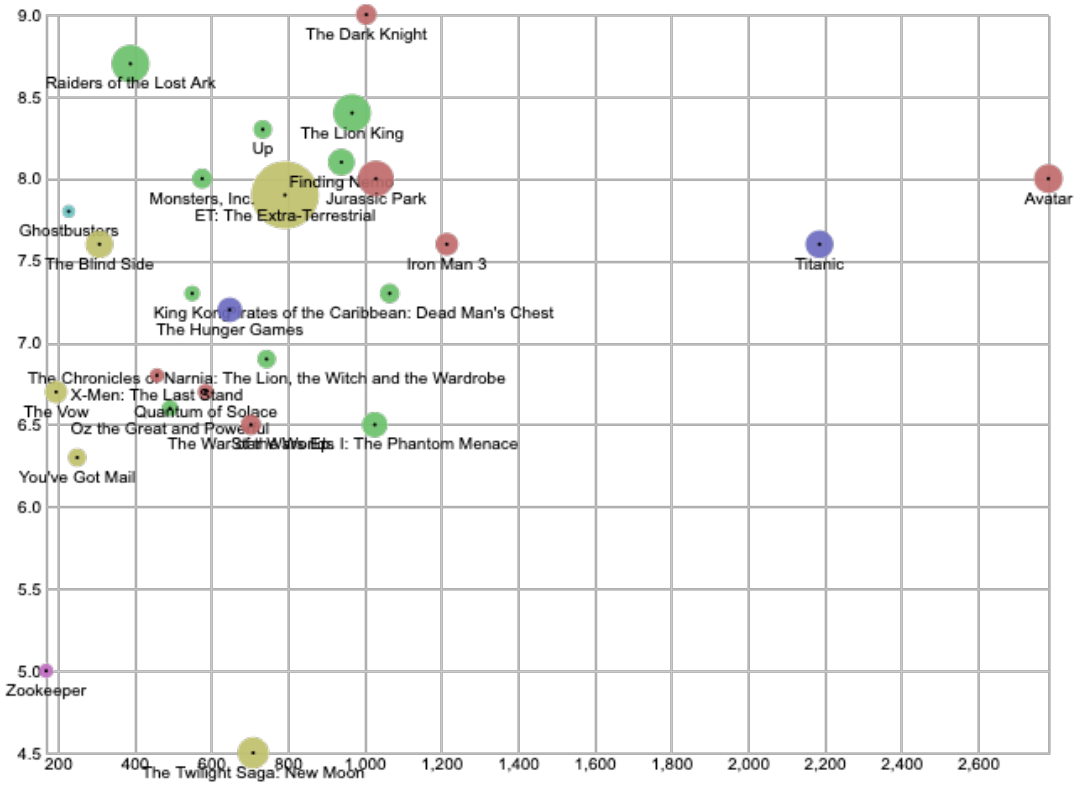

Figure 5. Scatter Plot

- **Analytics**

Graph / Criteria	Convex Hull	Contour Plot	Voronoi Tessellation	Scatter Plot
Visual patterns supported	- The dots: These dots represent the data points, which are the movie titles, on the 2-	- The dots: These dots represent the data points, which are the movie titles, on the 2-	- The dots: These dots represent the data points, which are the movie titles, on	- The dots: These dots represent the data points on the 2-dimensional plane.

	dimensional plane. The diameters can be manually set by users. - The colored areas: These areas are the area covered by dots of the same group (in this case, movie genre) connected together. - Color: Each color represents a group or category of data points.	dimensional plane. The diameters cannot be manually set by users. - The contours: These colored contours represent the density of the data points in a specific area. - Color: Each color represents a different level of density.	the 2-dimensional plane. The diameters cannot be manually set by users. - The colored area: These are the areas/distance from data points to data points. In Figure 4, they are automatically generated. - Color: Each color represents a label.	- The circles: The radiuses represent another variable of the dataset, which is *ROI* in Figure 5. The radiuses can be manually set by users (the ratios of different circles' sizes remain unchanged). - Color: Each color represents a group or category of data points.
Data type supported for each pattern	- The dots support numerical and datetime data. In Figure 2, they support numerical data. On a side note, the labels in the figure are strings. - The colored areas support the same type of data as the dots do because they basically just cover the dots. - Color: The groups or categories can be numbers, strings, or datetimes. In Figure 1, they are strings (movie genres).	- The dots support numerical and datetime data. In Figure 3, they support numerical data. On a side note, the labels in the figure are strings. - The contours can support numerical data which scale the density level. They also support strings in other cases as data points on the same "circle" of color have the same label. - Color: The colors support numerical data and strings.	- The dots support numerical and datetime data. The labels are not available along with the dots. - The colored areas support numerical data since they show distances. - The colors support numbers, strings, and dates as they represent the labels.	- The dots support numerical and datetime data. The labels are not available along with the dots. - The circles, or their radiuses, support numerical data. - The colors support numbers, strings, or datetimes. In Figure 5, they are strings (movie genres).

What data relationship can the readers see?	- The graph illustrates the area that the data points of the same group cover. The bigger the area, the more variance there is in the group. Data points that stay close to each other will create a small area. - Outliers can also be the cause to an instant increase in sizes of the areas. Therefore, readers can spot outliers if the area tends to extend sharply in a direction far from its cluster.	- The readers can draw a conclusion on the location of the clusters where the density levels are the highest to lowest. - Outliers can also be detected according to the color.	- The readers can see very clearly the minimum area around each data point. This helps defining the “borders” of these points. - Clusters can be spotted in areas with a high density of small patches. - Outliers can be spotted in big patches.	- The readers can see the clustering areas. - The plot allows readers to see the relationship of 4 different variables of each record, which is more than what the other methods offer. - Outliers can be detected. - Big-sized circles can be easily spotted as they will stand out. - Has visualization on categorization.
Pros	- Has a good illustration of the categorizations. - With a small number of data points, it can be a very clear visualization with the use of colors. - Show the variance of data points in the same group. - Easy to understand and draw quick conclusions. - Shows non-linear pattern.	- Great visualization on how to detect clusters with a high density of data points. - Shows what data points fall into the clusters. - Easy to understand and draw quick conclusions. - Can handle a large dataset. - Shows non-linear pattern.	- Very clear to read and understand. - Shows positions of each data point in the plane. - Is useful when users want to specify the location of individual data points. - Shows non-linear pattern.	- Clear visualization. - Observation and reading are straightforward. - A wide range of quick conclusions can be drawn from the plot: locating clusters, outliers; have visualization on categorizations; etc. - Shows non-linear pattern.
Cons	- A large number of records will make the graph	- Does not illustrate the data categorization.	- Does not illustrate the data categorization.	- A large number of labels can hurt the visualization as the

	extremely less clear to read as there will be too many data points in the overlap areas. - A large number of groups will create too many overlaps, which will also hurt the readability.	- A short of data may not make use of the feature of this type of graph.	- Too many records can hurt the visualization since the colors can be confusing to tell apart. - The area of the patches can be misunderstood as the size of the data points.	color range can grow to an extreme length that causes confusion. - Does not have a good illustration of the categorizations. - Does not describe the distance and spaces taken by data points. - Does not describe the "depth" of the clusters.

Conclusion

Each of the graphing methods has its own strengths and weaknesses. However, for me, the *Scatter Plot* stands out the most. Each of the other three methods have shown excellent visualization on different aspects of the dataset, but none of them can offer a wide range of conclusions like the *Scatter Plot*. Even though these conclusions, in cases, can be much shallower than those drawn from the other methods, such as *Scatter Plot* surely can neither describe the distance as powerfully as the *Voronoi Tessellation* nor illustrate the clustering areas as in detail as the *Contour Plot*, but for users who want to have a look at the big picture of the dataset, the *Scatter Plot* serves as a perfect option. Usually, analysts would want to visualize their dataset at the beginning of the project with a scatter plot because they prefer a simple-looking yet informative mapping that provides them with quick insights of the features and variables. The other three graphs are suitable for specific tasks when it comes down to digging deep into each aspect of the data. Without such specific tasks, scatter plots are will come in handy most of the time.