CHAPTER 9: Storytelling and Data Communication with Visualization
Written by Dr. Ami Gates

Introduction

Data scientists have many different responsibilities. These responsibilities include understanding probability and statistics, creating models and methods (such as those involving machine learning and data mining), performing data wrangling (such as cleaning, formatting, and transformation), managing big data and advanced computation, as well as a plethora of other technical steps and methods. However, the technical aspects of data science and analytics are only a part of the full data science process.

Data scientists must also clearly and successfully communicate the information derived from data so that it can be understood and utilized by non-technical stakeholders. Information derived from data, and models or methods generated through data, may be applied to affect public health and medicine, public policy and government, civil liberties and equality, business decisions and marketing, sustainability and environment, and an unlimited number of other areas. To enable data to be applied, and to become actionable, results and conclusions must be presented, illustrated, and visually communicated to decision-makers. For this reason, critical and invaluable aspects of data science and analytics are data communication, storytelling, and the use of visual narratives.

What is storytelling?

Storytelling is both an art and a science. A data science story should have a clear and familiar flow – with a beginning (introduction), a center (context and problem description), and a happy ending (actionable and clear conclusions). Building a story is a creative activity and can depend highly on the audience, the goals, and the topic. Storytelling brings the viewer on a journey through the information contained within the data (data communication) without necessarily overwhelming them with techniques or technical details. A data-based story can be persuasive or informative, or both. A story can assist with decision making, can provide options for improvements, can illustrate a current state and how to attain a preferred state, and can offer interesting discovery.

The most important element of any data science story (including a presentation, report, or narrative) is that it is inclusive. In other words, it enables anyone (technical or non-technical) to understand the results and key conclusions, to benefit from the information, to utilize the knowledge, and to follow-up with questions. It should not be necessary nor expected for clients, stakeholders, or managers to have advanced degrees in STEM to understand a data story or to use and value and information derived from the data.

Consider the following real example. A team of data scientists made an effort to explain models and methods to a group of stakeholders. The team of data scientists used words such as “precision”, “k-means clustering”, “supervised modeling”, “sigmoid kernels”, etc. As one may imagine, the group of stakeholders did not fully understand and so became impatient and disconcerted. The stakeholders

wanted to focus on conclusions and actionable items, while the data scientists were more focused on the methods and models. This disconnect caused two months of non-productive meetings during which the team of data scientists finally suggested that the stakeholders should be trained in statistics and in machine learning (ML).

What went wrong here?

Should the stakeholders need to be trained in statistics and ML techniques?

Take a step back for a moment. Imagine that a car owner takes their vehicle into a car mechanic to have it checked and serviced if needed. Suppose the mechanic evaluates the car and then tells the owner the following. The six-cylinder engine appears to be generating a clicking sound, and in some cases may not crank. The battery may need to be discharged, but could also have corroded battery cables. There may be a faulty fuel pump or blocked fuel filter. The ignition switch is demonstrating evidence of a possible breakdown, and there may be a failure of in starter motor relay. There may also be worn gaskets, damaged rings, a poor crankcase, and an incorrect oil grade.

Here again, what went wrong?

Would it be more appropriate for the car mechanic to simply say that the engine and starter have a few small issues that can be easily corrected? In most cases, the answer here is to offer the simple conclusions first. Most car owners do not wish to learn car mechanics, nor should they be expected to. Of course, if the owner wants to know deeper details, they can ask.

In both examples above, the clients were expected to understand more than they may need to. While clients and stakeholders may certainly choose to request technical details, it should not be expected that they will. Similarly, clients and stakeholders are likely to ask many questions that focus on use-cases, actionable items, and decision-support conclusions. The nature of the client-based questions will likely reflect on their knowledge, background, understanding, goals, and area of focus or interest. Therefore, the data story, subsequent conclusions, and answers to questions should focus on the goals, and should not obfuscate the actionable results.

In summary, it not necessary for a person to be an auto-mechanic to get their car fixed, it is not necessary for a person to be a medical doctor to get healthcare, and it is not necessary for a person to be a data scientists or statistician to understand, value, and utilize the information gained from data analytics.

Communicating data-based conclusions and information clearly and through storytelling, narratives, presentation, and visualizations is the job of the data scientist.

Storytelling as a Visual Narrative

Creating a cohesive, coherent, inclusive, and informative story (about data) starts with the creation of a storyline, or a plan. It is common to consider questions such as whether a story is intended to inform, to persuade, to convince, to enable, to promote, or some combination. One option for telling a story is to create a solution or conclusions-set first, and then build the story around the conclusions. This method

has the positive affect of leading the viewers in the desired direction and showing them what to think. Of course, this method also can have the equivalent negative affect of prejudicing the viewer and telling them what to think.

It has been suggested that decisions are more frequently made through the use of emotionalism rather than logic or data evaluation. Storytelling can either take advantage of this human characteristic (by conjuring emotion through directed narratives and visuals), or storytelling can combat this propensity by expressing information cohesively through unbiased and informative narratives and visuals. A story can inform, can support decision-making, can motivate action, and can be very powerful in effecting change. In all cases, stories can be generalized into three core elements.

Elements of Storytelling

Introducing the Topic

Stories begin with an introduction; a clarification of and motivation for the situation. An introduction will present the topic of interest, its background, possible stakeholders, those directly or indirectly affected, a history, and a foundation for understanding its nature, importance, and value. Ideally, a successful introduction will be inclusive and will enable all viewers, with or without technical backgrounds, to understand and appreciate the topic, the goals, and the basis. Weak introductions that do not properly motivate a topic may not retain the audience. Introductions that are too technical or too detail-oriented may lose an otherwise interested audience. As such, the beginning of the story should invite, intrigue, engage, and involve the audience.

Reflect on this idea. Recall a time when you were listening to a presentation and it “lost you” or “bored you” and as a result, you stopped listening.

Motivating the Problem

The next step in telling a story is to define the problem set in a way that all viewers can understand. If the definition of the problem is too technical, the audience may not fully understand the problem itself and so will not be able to appreciate the solutions presented. If the problem definition is not thorough enough or is not engaging, the audience may not invest concern, and therefore their focus and interest into the problem and subsequent solutions may not develop. Like the introduction, the definition of the problem should balance clarity, technicality, and personalization.

Reflect on this idea. Can you describe the problem or question set in 3 – 5 sentences? Do you fully understand it?

Explaining the Solutions

Once the area has been introduced and the problem set has been clearly defined, the solution set can be presented. Solution presentation can be particularly challenging as great effort must be made to avoid becoming overly technical. For example, many viewers may be more interested in a useable solution rather than analytics jargon about how the solution was reached. As with the introduction and the problem definition, the explanation of the solution should be clear, inclusion, actionable, and useable.

The Art Within the Science

While storytelling can be described generally as: (1) offering a solid and clear introduction, (2) motivating and expressing the problem set, and (3) explaining and describing the solutions, there are several interspersed elements that can make a story better, more convincing, more actionable, more inclusive, and more engaging.

Finding a functional balance between simplicity and detail can make the difference between an interested audience and a bored and disconnected audience. It has been suggested that data scientists, statisticians, and mathematicians have a great love for the details. This is perhaps true. However, most viewers do not. In many cases, the audience wants to understand fully, but does not want to be bogged down in why the data scientists are smart or what the nitty gritty details might have been.

Successful data stories often focus on critical and important elements at an appropriate audience level. This way, the core points will find the audience and the superfluous details will not lose them.

Motivating Action

One of the goals of storytelling is motivating people into action. A data story introduces a topic, presents a problem, and suggests solutions. However, this is only a fraction of what a good story must do. A story should draw the audience into the topic on a personal and so emotional level. A good story will clearly answer the following questions:

(1) Why does the topic or area matter?
(2) Who does the topic or area affect?
(3) Why does the area directly affect the viewer?

Stories that offer actionable solutions while also stimulating personalized and emotionalized concern in the viewers are more likely to result in decision-making and perhaps policy-making. To this end, showing an audience numbers, tables, or highly technical results may have less of an effective than showing visualizations that express results and encourage action. A good story will illustrate the positive effects of putting the data-based conclusions into action and making related decisions. If an audience is emotionally motivated by a story, they are more likely to continue forward with the suggestions and results.

Stories with Happy Endings

Storytelling in data science is very similar to common storytelling in “real life”. When listening to or reading a story, it is common to expect to understand the context, to understand the problems, and then to understand the solution or ending. It is rarely the goal of a story to ask or require the viewer to do the work. Instead, stories will generally deliver the information to the viewer, starting from the introduction and context, traveling through the relevant observations, and ending with a conclusion that enables action. Precision is stories includes leaving out unneeded and unnecessary details, erring on the side of simplicity, and enabling all members of the audience to understand the conclusions. Data science stories that offer actionable conclusions can be thought of as having a happy ending. In other wrods, such stories identify and define a problem or issue and then (based on data science and analytics results) offer a solution.

It's Not About You!

As a final note, always remember that stories and visual narratives, as well as professional papers and presentations, are about the topic, the story, and the audience. They are not about the authors or the research team. In other words, it's never about you! Too often, stories tend to focus on the experiences of the writer or the research team, their trials and challenges in data collection and analysis, and their personalized experiences. To avoid this misstep, words like, "I", "we", and "us" should rarely appear. Direct the focus away from you and your team and toward the topic and the audience.

Storytelling with Visual Narratives

A visual narrative is a story that relies mostly and heavily on the use of visual media to convey the elements: the introduction, the context and problem, and the conclusions. Visual media can include static visualizations, interactive visualizations, animations, voice, audio, video, and sound. This chapter will focus on static and interactive visualizations only.

A visual (a graphic or image) can move a person to tears, can evoke laughter, and can make a point without the use of words. The power of visualization cannot be overemphasized and so visualizations should be used often and with accuracy and caution. While all stories can benefit from the use of visualizations, a visual narrative relies more fully on visualizations to guide the audience through the message and results.

Because visualizations leverage dimensions such as color, shading, transparency, size, shape, positioning, labeling, and in some cases choice, input, and interaction, they are able to convey an enormous amount of information within a single two-dimensional page.

When creating or building a story that is a visual narrative, a good rule is to create the story with only visuals first, and then add minimal text. Like all stories, visual narratives will flow through the elements of the story and will show the audience the topic, the context, the problems, and the solutions. The following illustration in Figure XXX shows a simple but poignant example of how a visualization can communicate with more efficiency and effectiveness than plain text. Which option is likely to work better in practice, Box A or Box B?

Box A	Box B
At the end of this road is an intersection. Other cars may or may not be driving in the opposite direction. For safety, please come to a complete stop at the end of the street. Look both ways, determine that it is safe to pass, and then proceed.	

Figure 1: Two options for communicating information. Which is most effective and efficient?

Key Elements to Visual Stories and Narratives

Both storytelling and data visualization are a blend of art and science, creativity and content, and clarity and information sharing. Therefore, a developer's mindset matters. When creating a set of visualizations that tell a story, the first element to consider is how to best *show* the information (rather than telling it with text). While it may initially be a challenge not to use and rely on text, the goal of visual narratives is to share more than what can be offered with words, and to do so with greater richness. There are several benefits of using visualizations, rather than wordy explanations, in the communication of information. This includes the fact that visualizations may be understood by people who speak any language (or who are hearing impaired), visualizations can reduce the ambiguity that is often found in verbal communication, and visualizations are significantly faster in communicating ideas and concepts. Smart visualizations can often be universally understood all over the world.

The second core element to storytelling with visual narratives is the creation of a clear context. The longer it takes for the audience to understand what the story is about, the less likely it is to retain their attention and interest. What is context? In general, context combines the background, the ideas, the goals, the relevance, and the basis. Context not only enables the audience to understand, but also to empathize and engage.

Once the context and basis are conveyed, the next step is to illustrate the problem set. What is the problem? What are the sub-problems that support or are related to the core problem? Who or what do these problems effect? How can solving the problems help or improve? Who can benefit and how?

For example, telling (with words) a group of people that exponential population growth may affect sustainability and may result in eventual critical shortages of resources may not have as strong of an effect as showing (visually illustrating) these facts with clear, interactive, and temporal visualizations. It may be suggested that most people have a tendency to respond more quickly and more deeply to what they see, rather than to what they hear.

Instruction, interest, and impact can work in parallel in a visual story. The greater the interest, personalization, and engagement, the more likely the audience will become motivated and involved. As noted above, stories should focus on the topic and the audience (rather than on the storytellers). This goal can extend to the idea that a story can empower an audience by provoking a feeling of ability, of control, and of being part of the solution.

The identification of potential obstacles (the bad guys in the story) can also increase audience interest and empathy for the topic and the solutions. For example, suppose the goal of a visual narrative is to prohibit the use of gas-powered and toxic lawn care methods due to excessive noise, egregious pollution, potential for cancer, and a lack of real value. Think about the elements of this narrative. The introduction will contain visual evidence of the significant pollution and noise generated by "lawn care". Another visualization might show data on the level of noise and how such noise levels may adversely affect children. A visualization may illustrate increases in cancers (especially in highly lawn-maintained areas). Further visualization may illustrate pollution comparisons, such as increases in air pollution, with

respect to measure of lawn-care methods. In creating these visualizations, the story invites the viewer to learn about the issue and to begin to personalize and empathize with the related concerns.

The next collection of visualizations in this story may further and more technically illustrate the problems and the context. Such visualizations might include maps of noise and pollution ranges, densities and ages of those affected, increases in cancer rates near active lawn-care areas, and maps of locations of children's parks including the amount of chemicals deposited on those grounds.

Once the problem has been clearly and visually communicated, the next stage in the visual narrative (the story) is to provide conclusions and actionable solutions. In some cases, conclusions can begin with asking and answering questions that the audience may have. For example, what is the value of using toxin chemicals such as herbicides and pesticides on lawns and in parks? Do the risks outweigh the benefits? What is the benefit of choosing grass over other more sustainable options? Is the constant removal of leaves more important to citizens than a noise-free environment? Are leaves more bothersome than air pollution?

From here, further visualizations might invite the audience to critically question the value toxic lawn and park care versus the value of human health and quality of life. Visualizations might also illustrate the number of complaints, petitions, studies, research, and requests from citizens to prohibit the use of such toxic lawn care methods.

Upon reflection, and primarily through the use of visualizations, the topic has been well motived, the problems have been explained, questions have been addressed, and who is being affected has been described.

The final set of visualizations will illustrate a collection of conclusions and solutions. Solutions might include options such as the use of clean, green, noise-free, gas-free, and non-toxic tools and methods and proof of their equivalent effectiveness.

Visual narratives tell a story. The goals are to inform, instruct, support, motivate, direct, convince, describe, and promote action.

In general, visualizations can include static graphics, interactive graphics, and other visual media, such as image, video, and sound. While this first example was intended to be persuasive, it is not necessary for a visual story to convince or sway. Visual narratives may also be used to instruct, inform, or support decision making (such as with business analytics).

Data Science topics can focus on any area for which data can be collected. This makes the applications of data science and data visualization nearly infinite. Data science and analytics are often utilized in economics and finance, marketing and advertising, engineering and science, medicine and health, astronomy and physics, business and sales, psychology and sociology, public policy and government, environment and sustainability, and so on. Each area will tell its own stories, will have its own goals, and will have a specific audience in mind. Stories must reflect these criteria and should engage, instruct, and respond to the specific audience and topics.

Types of Visualizations

Static Data Visualization

By using creative, skilled, and thoughtful storytelling and visual narratives, a data scientist can motivate and realize a topic, can accurately and clearly communicate data-based information, can include technical aspects as part of the story, and can explain results and how they might be understood and utilized. It is not uncommon for data scientists to focus so thoroughly on the data itself, and its analysis, that the underlying goals, the general topic, the clients and stakeholders, and the conclusions are often overlooked or under emphasized.

A recent and strong trend in all areas of data science, analytics, machine learning, data mining, and the like is to shift the focus away from purely technical analytics into a blend of analytics and information communication. This shift does not devalue or reduce any of the critical data science techniques, but rather it shifts the focus to communication. Paramount to this shift, and to data communication and storytelling, is the use of visualizations.

Therefore, and before moving deeper into storytelling methods with visual narratives and examples, it is first necessary to discuss data visualization itself. On the surface, data visualization is the process of building graphics (static or interactive) from data.

As an interesting experiment, 50 students were given a clean dataset and 15 minutes of time. They were asked to visualize the data using R. As a result, 95% of the students used a bar graph and a scatterplot – both static. Some students tried to use color. Some students labeled and titled the visualizations. After the experiment deadline of 15 minutes, students were asked about why they chose the visualizations they selected. The most common answers were, “I am not really sure and was not sure what to look at.”, and, “I am familiar with bar graphs and how to create and understand them.”

This important experiment reveals a critical and yet often overlooked aspect of data visualization. Data visualization is a tool. Like any tool, the user must start with a goal and a clear understanding of the project in order to properly use the tool. However, if one is not certain what they wish to build, the tool may not be as useful and can prove to be dangerous. In fact, the first steps in data visualization (and data science) including thinking, planning, and conception.

The thinking part involves investigation, clarification, research, goals, clients or audience, topics, and the nature of the data. If possible, background, collection methods, use-cases, and questions from the clients can be of great value. While a full and complete basis (background and meaning of the data) is not always possible, initial steps can be taken to gather information about the data so as to begin to generate questions.

Question generation is a critical initial step in data analysis. In many cases, initial questions can be investigated and even answered using data visualization. Question generation is iterative. As questions are formed, they lead to other questions. As questions are answered, they lead to deeper questions.

Where does data visualization fit in and what are all the different aspects of visualizing data? This compound question is actually quite involved. Topically, data visualization is the process of creating visuals or graphics that describe elements of a dataset. Data visualizations might be static or interactive. They may be used for discovery, explanation, or deeper investigation. Visualizations can be simple and

two dimensional or can be complex, compound, and highly dimensional with several user-selected and interactive options. In some cases, a visual may be constructed to determine the underling likely population distribution of a single variable in a dataset. In other cases, several variables may be visualized together to determine possible relationships or to discover points of interest. Data visualization may also be used to teach or instruct, to pursued or convince, or to communicate huge amounts of data quickly and in a manner that is universally clear. In short, data visualization can and should fit into all aspects of data science and analytics; especially storytelling.

The following subsections will discuss and investigate several aspects, types, and methods for visualizing data. To support and motivate the following discussion and examples, consider the dataset in the FIGURE 2. This dataset is fictitious, but was created from real data so as to conform to expectations. This dataset has already been thoroughly cleaned and is record-type data. As an aside, this chapter will later consider the visualization of several data format types, including network data and transaction data.

	A	B	C	D	E	F	G
1	Label	Gender	Cholesterol	MaritalStatus	Weight	Height	StressLevel
2	Risk	M	251	S	267	70	5
3	NoRisk	F	105	M	103	62	1
4	Medium	M	156	S	193	72	3
5	NoRisk	F	109	M	100	63	2
6	Risk	M	198	S	210	70	4
7	Risk	F	189	S	189	64	3
8	NoRisk	F	121	S	105	65	1
9	Medium	F	134	M	125	60	2
10	Risk	M	250	S	156	69	5
11	NoRisk	M	118	M	190	71	3
12	Risk	F	290	M	300	62	4
13	NoRisk	F	156	M	119	69	1
14	NoRisk	F	145	S	112	68	2
15	Risk	M	178	S	177	68	3

FIGURE 2: Simple record-type dataset to support and motive data visualization examples that follow. This data is fictitious. This data has 7 variables. The first is “Label” as this is labeled data. The “Label” represents the risk of getting heart disease. This figure only represents a fraction of the entire dataset. Dataset:
https://drive.google.com/file/d/14wEkB8eSTG8HRiALNN3WB5mcyBV0WCeQ/view?usp=sharing

The dataset in FIGURE 2 was created to emulate expected data for heart health risk factors. The dataset is clean and is labeled. The label of the data, “Label”, represents one of three categories: Risk, Medium, NoRisk. Each represents a risk-group for getting heart disease. Therefore, the first row (or person) in the dataset is at the highest risk of the three categories while the second person is at the lowest risk category. Medium is in the middle of low and high risk. The dataset contains two nominal and qualitative variables, “Gender”, and “MaritalStatus”. If using the programming language, R, the types of these variables must be set to “factor”, as must “Label”. The variable, “StressLevel” is ordinal and qualitative (ordered factor in R). A stress level of “1” is the lowest stress and a stress level of “5” is the highest stress. Finally, the variables, “Cholesterol”, “Weight”, and “Height” are quantitative (numeric in R).

As a Case Study example and to support the process and discussion of visualization methods, assume that a team of data scientists has been given a large dataset with the above structure and content type. The general goals are to understand the information contained in the data, to generate health-related recommendations, to communicate key elements to patients and other people, and to create support documents and pamphlets. A broader goal might be to update and effect public policy.

With this in mind, it can be surmised that this data will be used for information, for prediction and classification, and for persuasion. The report that will be generated will be a story about the data that will enable and support this health-related effort, and will also inform and support patients, doctors, and others.

How might the team begin?

While there are many ways to start, one method is to start with discovery. As a team, the data science group might brainstorm and generate 10 – 20 questions about the data. It is OK if not all the questions prove to be critical. It is OK if some of the questions may not be able to be answered. It is also OK if some or most of the questions lead to deeper and more involved questions and analysis.

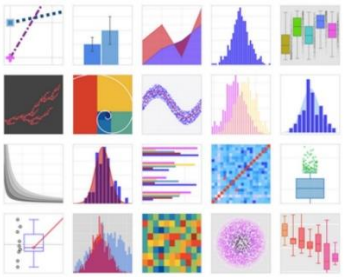
Fifteen Possible Questions:

1) Are women or men at greater risk of heart disease?
2) Is stress level correlated with risk?
3) Is cholesterol correlated with risk or stress or both?
4) Is marital status correlated with risk or stress of both?
5) Are weight and height correlated with each other?
6) Is weight correlated with risk and if so, is height connected in some way?
7) What are the joint or paired correlations between all variables? Based on this, does anything stand out that was not initially investigated in the above questions?
8) Is cholesterol normally distributed and if not can the underlying distribution be identified?
9) Is the data balanced? In other words, is there an even number of rows representing all three categories of risk or all categories or gender?
10) Is there a significant different in risk factor between the three stress levels?
11) Is there a significant difference in risk factor or stress level between the two genders?
12) Can this data be clustered if the label is eliminated and if so do the clusters match or closely match the current labels?
13) What are the min, max, variance, IQR, and median for the numeric variables?
14) Can this data be used to predict a person's current risk of heart disease?
15) Does association rule mining reveal strong relationships between certain variable values?

Creating questions is part of discovery. As questions are created, other questions will be thought of. This is an iterative process and the above 15 questions are just the beginning – the brainstorming part – of that process. However, these 15 questions can direct the next step, which is exploratory data analysis (EDA) using visualization tools. To begin the process of exploration, static visualization options are often employed first. The following sections will investigate several families of visualization types, when and why they are used, and what types of information they may reveal.

Static Visualization Families

A static visualization is a graphic that cannot be altered or affected by the user or viewer. Unlike interactive visual options (which will be discussed later), static visualizations are used to illustrate a finite amount of information with clarity and ease. Can you think of 20 static visualization options? Most people think of pie graphs, bar-type graphs, scatterplots, and boxplots. These are excellent for discovery. They are generally easy to create, easy to interpret, and easy to share with others. However, they are not the only choices. In fact, there are thousands of options, and being aware of the wide range of data visualization graph choices should be a part of every data scientist's toolbox. The following several figures illustrate only a small collection of all the different types of static visual families that are available. A visualization family (or category) is a group of visualizations that share similar attributes but may vary slightly in form. For example, the bar-graphic family may contain vertical bar graphs, horizontal bar graphs, stacked bar graphs, side-by-side bar graphs, multiple or compound bar graphs, etc.



https://help.plot.ly/citations/
Figure 3: This image is borrowed from: https://pypi.org/project/plotly/

For the following sections, it is assumed that data has been collected, cleaned, and formatted. It is also assumed that the goals, clients, basis, and other important information about the data are being actively considered.

Choosing the Best Visualization Option

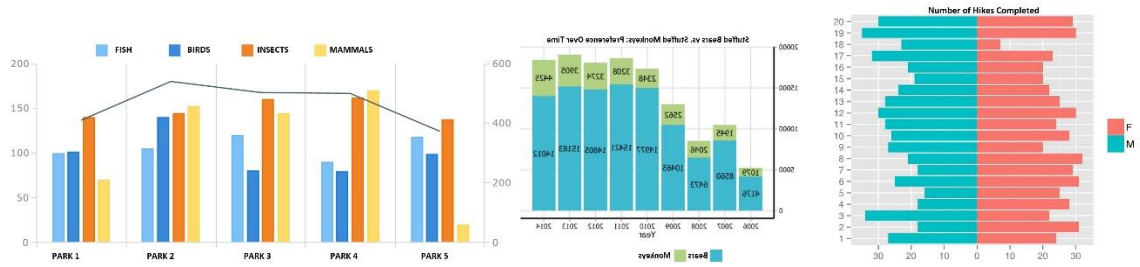
1) A visualization should illustrate, share, and convey accurate information without misrepresenting or misleading the viewer.
2) A visualization should be as simple as it can be, given (1) above. Visualizations should not be needlessly complex nor should they require the viewer to try to figure out what is being communicated.
3) Visualizations should wisely utilize aesthetics options, such as color, shading, sizes, shapes, etc. so as to elegantly communicate information without obfuscation or distortion.
4) Visualizations should never be too busy or too difficult for the viewer to clearly comprehend.

The task of choosing the most appropriate visual option can seem daunting. For this reason, bar graphs remain quite popular. However, a productive approach can include first becoming familiar with common families of graphic options and then selecting options that best (and most clearly and accurately) convey and communicate the intended information. There are several families of graphics, including the bar-graphs (vertical, horizontal, stacked, grouped), the box-plots (including the violin), the scatterplots

(including bubble graphs and heat maps), the radial graphics (such as rose plots and pie graphs), and several others such as time series, geospatial and maps, and networks. Because graphic options and types may be combined in an almost limitless number of ways, there are an infinite number of graphs (visualizations) that can be created. While it may not be feasible to become familiar with every graph type, it is very possible, and a good idea, to become knowledgeable about common graph families and when and how they are best utilized.
The following will discuss and illustrate several common static graphics families. Interactive, compound, and advanced visualizations will be discussed later. As each family of static visualizations are discussed, important considerations such as the use of color, size, space, axes, labels, titles, and shapes should be considered. However, because there are an infinite number of possibilities and combinations, the following examples should be viewed as an introduction rather than a complete set.

The building of a visualization, especially one that accurately and clearly represents the data or variable(s) of interest, is a blend of art, science, programming, analysis, and human psychology. As a simple example, most people are comfortable with the color green as representing “good” or “go”, and red as representing “stop” or “bad”. Similarly, the size of an element may automatically be presumed to represent quantity, level, or volume. Shading and shapes must also be carefully considered. When using shading, consider issues such as the fact that shades that are too close in color may be indistinguishable from each other. Similarly, consider that a common assumption is that deeper or darker shades tend to imply concepts of “more” or “worse” in some cases. The following will introduce and discuss several common static graphics families.

The Bar Family



A “bar” can be thought of as a geometric rectangular structure. Bars can represent count, relative or absolute frequency, or quantity of any kind. However, bars cannot illustrate measures such as variation, direction, order, or location. Bar graphs can be used for qualitative or quantitative variables. By combining bar graphs, multiple variables can be illustrated.

To represent qualitative data using a bar-type graphic, it is necessary to identify a finite (and relatively) small set of categories (contained within the variable of interest) to be represented. From there, the number of occurrences of each category can be “counted”. For example, the variable StressLevel in the Heart Health Dataset (Figure 2) contains five categories that are labeled with the numbers 1 through 5. Remember that while numbers can be used to label data, these numbers are not numeric. They are qualitative and ordinal.

Figure 4 below (on the upper left) illustrates the variable, “StressLevel”, where each of the five bars (one for each category of stress level) will then represent the count (frequency of occurrence) of each stress

level in the data. For example, stress level "1" occurs a total of five times in the dataset and stress level of "2" occurs eight times in the dataset. (Recall that Figure 2 above represents only a small portion of the dataset and that the entire dataset can be downloaded from https://drive.google.com/file/d/14wEkB8eSTG8HRiALNN3WB5mcyBV0WCeQ/view?usp=sharing)

Using a bar graph to represent a qualitative variable such as StressLevel offers the viewer a fast, effective, and easy way to determine (see) the following:

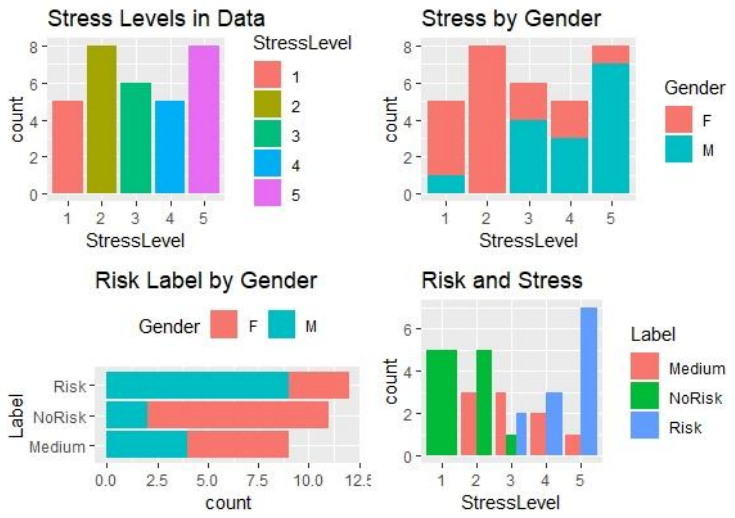
1) A comparison between the stress level counts for each stress level. Here, we can see that stress levels 2 and 5 occur the most relatively.
2) A determination as to whether one or more of the stress levels is much more frequent (or infrequent) as compared to the others. Here, while some stress levels occur a bit more than others, there does not appear to be any extreme differences.
3) A determination as to whether each stress level is fairly represented; if the dataset is balanced with respect to stress level. In this case, each stress level is represented and none have a zero count.

This type of bar graph is excellent for comparing natural categories within a given variable. As an aside, a "natural category" is one that is self-defined by the nature of the data. For example, the possible values for StressLevel are 1, 2, 3, 4, or 5. Therefore, there are five naturally occurring categories. In some cases, one must first categorize or discretize the data before applying it to a bar graph.

Examples of general (or every-day) variables for which a bar graph can be of value may include gender, political or religious orientation, marital status, admissions status, stress level, letter grades from a class, etc. Alternatively, bar graphs tend to be inappropriate for representing numerical variables such as age, height, weight, or cholesterol level. While there are no set rules, it is fair to suggest that a bar graph with too many bars can become difficult to view or to understand. For example, imagine using a bar graph to represent an AGE variable. Suppose there are 90 different ages, such as 3, 17, 35, or 67. By definition, the bar graph would then contain a bar for every age; so 90 bars in this case. Think about why a bar graph with 90 bars may not assist in illustrating data or conveying information. Can you think of how you could successfully use a bar graphic for data that represents ages (Hint: Discretization)?

Bar graphs may be vertical, horizontal, stacked, or grouped, and may represent both positive and negative values. Bar graphs are quick and simple, and often do a good job of illustrating basic ideas about variables. They are also often easy for anyone to view and understand, and can be effective for initial exploratory data analysis.

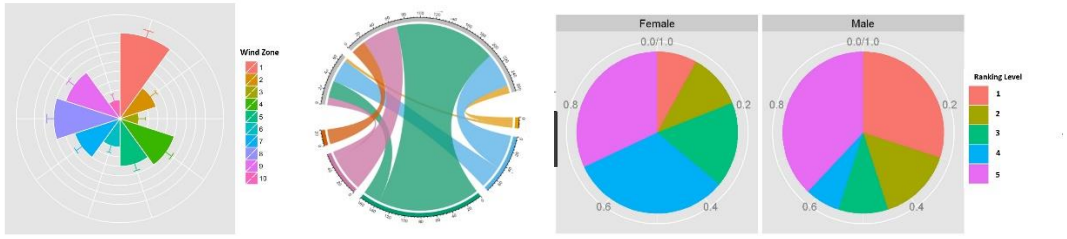
Bar graphs offer many options and attributes. These include the width of each bar, the number of bars, the colors used (both fill and outline), bar labels, bar stacking, bar grouping, and so on. The following figure will illustrate a few bar graph options. The Heart Health Risk dataset was used to generated these graphs, and the R code (ggplot2) is included below. There are an infinite number of possibilities when creating a bar graph. The key to creating an excellent visualization is to first identify the information to be conveyed and then to determine the best (and clearest) way to illustrate it.




```
BaseGraph <- ggplot(RiskDF)
## Basic
(G1<-BaseGraph + geom_bar(aes(StressLevel, fill = StressLevel)) + ggtitle("Stress Levels in Data"))
## Stacked
(G2<-BaseGraph + geom_bar(aes(StressLevel, fill = Gender)) + ggtitle("Stress by Gender"))
## Horizontal and theme
(G3<-BaseGraph + geom_bar(aes(Label, fill = Gender)) + ggtitle("Risk Label by Gender")+
  coord_flip() + theme(legend.position = "top"))
## Grouped
(G4<-BaseGraph + geom_bar(aes(StressLevel, fill = Label), position="dodge")+ggtitle("Risk and Stress"))

library(gridExtra)
grid.arrange(G1, G2, G3, G4, nrow = 2)
```

Figure 5: Starting from the upper left, this figure represents a subplot containing four bar-type plots: standard bar, stacked bar, horizontal stacked, and grouped. To the right of the graphics collection is the R code which uses ggplot2.

The Radial Family



Radial or circular visualizations include the well-known pie graph, as well as many others such as the radial bar, radial line, radial tree, radar, and the nightingale rose. Bar graphs, line plots, area plots, and scatterplots can all be represented radially. In addition, and as with all other graphics, attributes such as color, size, and shape can determine how clearly and effectively the contained information is conveyed.

Why represent a variable (or variables) using a radial or circular graph rather than a more standard "rectangular" graphic? This type of question is important to ask and often depends on the nature of the variables and the goals of the visualization. In other words, one might first determine what information the visualization should illustrate and then determine the most appropriate graphic. For example,

suppose “order” is of importance and needs to be retained as part of the visualization. In this case, a bar graph will not offer the desired outcome. However, circular graphs can. For example, a radial plot may include geographical directions of North, East, South, and West which may be used to better illustrate things like wind speed and direction. The following figure illustrates (left) a basic bar graph showing wind direction and (right) a radial bar graph illustrating speed and direction. Notice that the radial graph not is not only conveys the information with greater ease, but it also permits a notions of continuity, order, and direction which the conventional bar graph (left) does not.

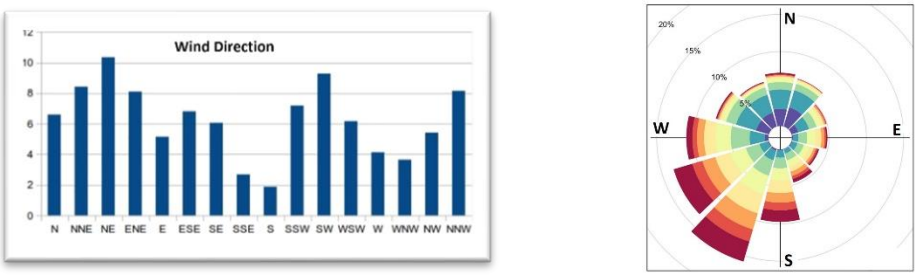
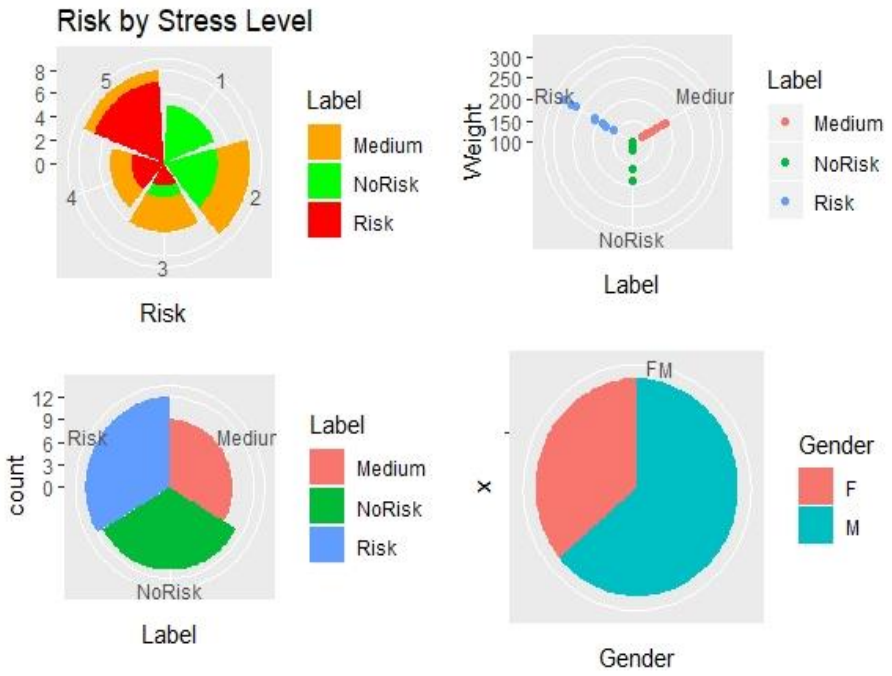


Figure 6: On the left is a simple bar graph that represents wind direction. On the right is a radial bar that represents wind direction and speed.

Because circular graphics can represent rankings in order around a circle, they are often more appropriate for ordinal data. This additional attribute offers an extra “dimension” of information communication. Circular graphs can also “save space” as they utilize space differently. This space-saving allows circular graphs to convey more information in a smaller overall area.
Each radial graphic offers unique attributes or dimensions that may be appropriate for representing certain types of data or information. The common pie graph offers the extra attribute of “whole” or “100%” and so is more intuitive for representing percentages or fractions of data. Similarly, the Nightingale Rose plot retains all of the value of a stacked bar chart and adds to it dimensions of direction and order. Figure 6 illustrates several radial plots.

The following visualization offers four radial graphic examples for representing data from the Heart Health Dataset.




```
(R1<-
ggplot(data=RiskDF,aes(x=StressLevel,
fill=Label))+
 geom_bar()+ coord_polar()+
ggtitle("Risk by Stress Level")+
  scale_fill_manual(values=c("orange",
"green", "red"))+xlab("Risk")+ylab(""))
(R2<-ggplot(data=RiskDF)+
geom_point(data=RiskDF,aes(x=Weight
,y=Label, col=Label)) +xlab("Weight")+
   coord_polar("y"))
(R3<-ggplot(data=RiskDF)+
   geom_pie(data=RiskDF,aes(x=Label,
col=Label))+ggtitle("Risk by Weight"))
(R3 <- ggplot(RiskDF, aes(x = Label, fill =
Label)) +
 geom_bar(width = 1) +
 coord_polar(theta = "x"))
(R4<- ggplot(RiskDF, aes(x="",
y=Gender, fill=Gender))+
 geom_bar(width = 1, stat =
"identity")+
   coord_polar(theta = "y"))
grid.arrange(R1, R2, R3, R4, nrow = 2)
```

Figure 7: Starting from the upper left, this figure represents a subplot containing four radial-type plots: To the right of the graphics collection is the R code which uses ggplot2. Upper left: Nightingale Rose. Upper Right: Radial Dot. Lower Left: Radial Bar. Lower Right: Standard Pie.

As a final example, the following creative visualization, published in Science (Quantifying Global International Migration Flows, Guy J. Abel*, Nikola Sander*, et. al., Science 28 Mar 2014:, Vol. 343, Issue 6178, pp. 1520-1522) illustrates the blend of radial, network, color, shape, size, and direction. This is, in fact, a radial Sankey type plot.

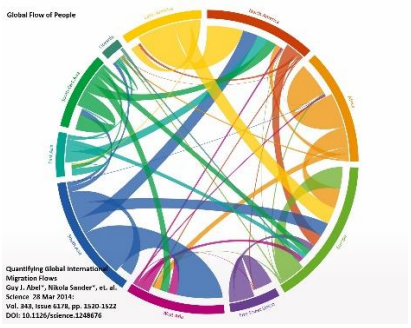
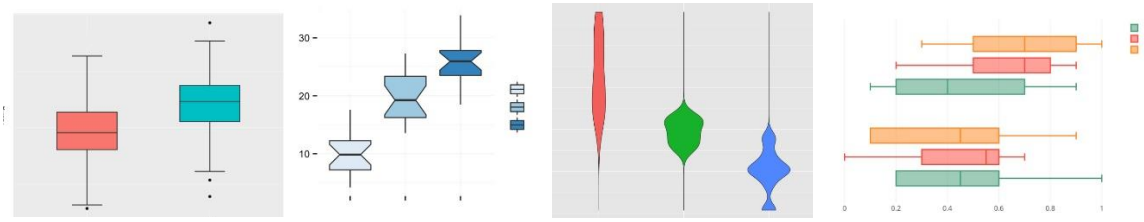


Figure 8: Published in Science (Quantifying Global International Migration Flows, Guy J. Abel*, Nikola Sander*, et. al., Science 28 Mar 2014:, Vol. 343, Issue 6178, pp. 1520-1522.

Consider the amount of information being easily shared through the visualization above in Figure XXX. While there are often many visual options that can be used to describe any variable or collection of variables, the art of data visualization is the process of creating an ideal visualization that truly captures and conveys the information in the data; without error or prejudice.
https://datavizcatalogue.com/

The Boxplot Family



The Boxplot Family contains the standard boxplot, the violin plot, the jitter plot, the notched boxplot, and grouped boxplots. Boxplots may also be horizontal or vertical. In fact, the only real limitation is one's creativity.

Elements and attributes that differentiate the boxplot and its variations from other plots such as bars or pies is that it includes measures of center and variation. The standard boxplot includes and illustrates the measures of the data minimum, the first quartile (Q1), the median (Q2), the third quartile (Q3), the interquartile range (IQR), and the maximum. Figure XXX illustrates the anatomy of a standard boxplot.

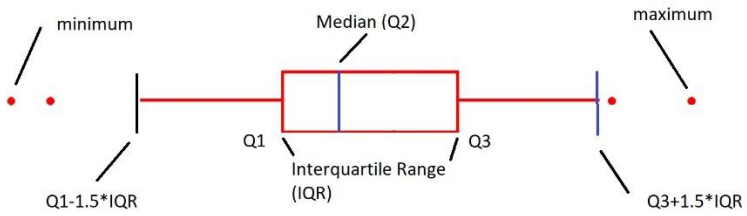


Figure 9: Anatomy of a standard boxplot, including the minimum, the quartiles, and the maximum.

Boxplots may be used to illustrate quantitative or numeric data. However, they cannot be used to visualize qualitative data. If this is not clear, think about how you might calculate the median or Q1 for hair color. The measures that define the boxplot can only be applied to truly numeric (quantitative) data.

Boxplots are also excellent for investigating and visualizing possible outliers. There are several variations of boxplots. These include the notched boxplot, the violin plot, and the jitter plot. The notched boxplot has all the same attributes as the standard boxplot, with the addition of a "notch" that illustrates the 95% confidence interval about the median. Figure XXX illustrates a notched boxplot anatomy.

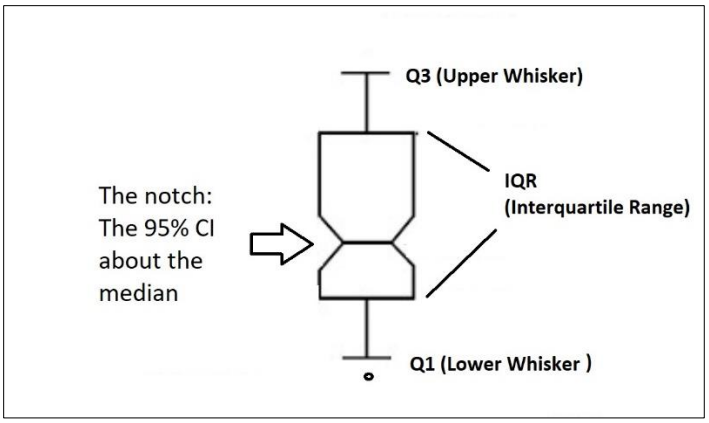
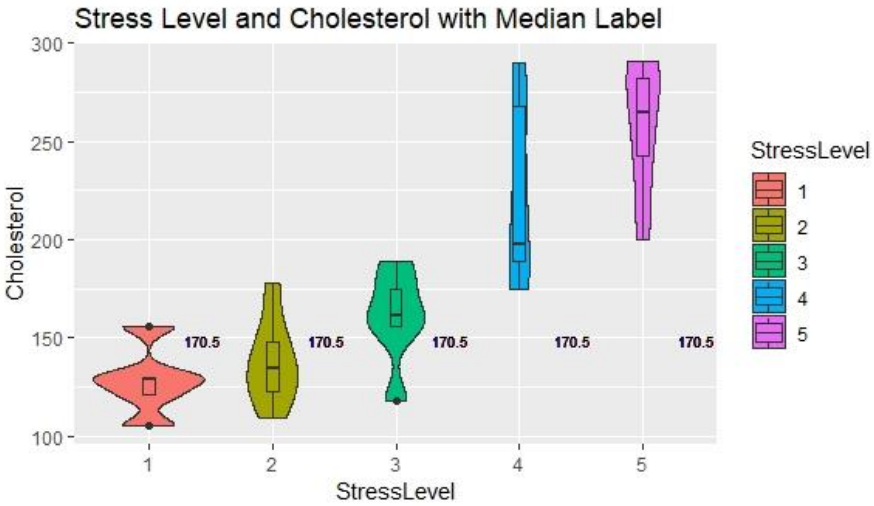


Figure 10: Notched boxplot with the notch representing the 95% confidence interval (CI) about the median (Q2).

The violin plot is a variation of the boxplot that also represents the distribution of the data; the probability density distribution. A violin plot can be thought of as the blending of a boxplot and a density plot. The boxplot portion of the violin plot continues to illustrate potential outliers, the min and max, the quartiles, and the interquartile range. The addition of the density (the width of the plot) enables the inclusion of the "shape" or distribution of the data. Violin plots, because they include data density, also show modality, such as whether data is unimodal, bimodal, uniform, etc.

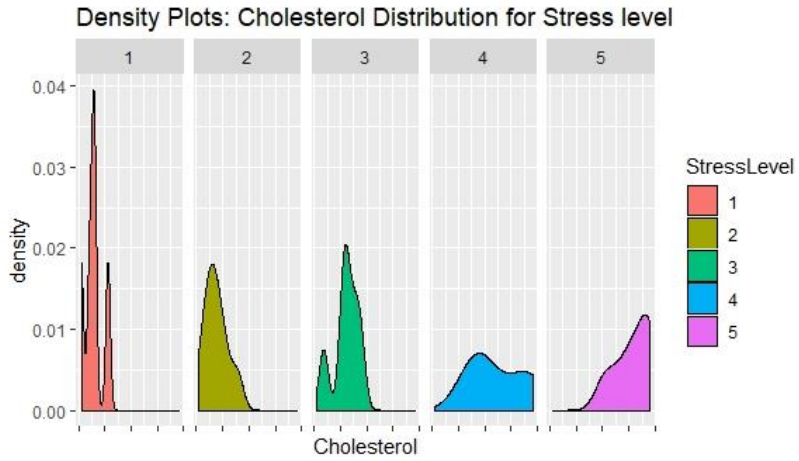
Figure 10 (above) illustrates the anatomy of a violin plot. Figure 1 (below) offers a set of violin plots, with each plot representing a stress level category with respect to Cholesterol level from the Heart Health Dataset. Notice that while a stress level of 4 (fairly high stress) is largely uniformly distributed between cholesterol levels ranging from 175 through275, a stress level of 5 (the highest stress) is skewed toward higher cholesterol. In addition, a stress level of 1 (lowest stress) is tri-model with the most pronounced mode in the center and near to a cholesterol level of 125.




```
medC=median(RiskDF$Cholesterol)
(V1 <- ggplot(RiskDF,
aes(x=StressLevel, y=Cholesterol,
fill=StressLevel)) +
  geom_violin(trim=TRUE)+
geom_boxplot(width=0.1)+
   geom_text(aes(y = medC, label =
medC),
   size = 2.5, vjust = 4, hjust=-1)+
   ggtitle("Stress Level and
Cholesterol with Median Label"))
```

Figure 11: Violin plot illustrating data range and distribution for cholesterol level and grouped by stress level. The R code using ggplot2 is included.

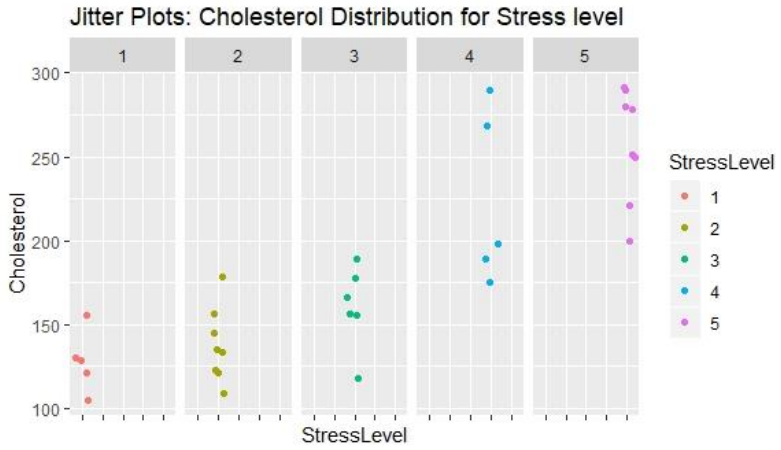
Figure 11 above suggests that the distribution for cholesterol level with respect to stress level varies quite a bit. To further investigate this finding, a more specific density-type of visualization can be created to visualize each density distribution. Figure 12 (below) illustrates an area density distribution for each stress level with respect to cholesterol. Consider the stress level of 1 (lowest stress). The density plots below, for stress level 1, confirm the sharp and tri-modal distribution of the data as well as a right skew due to the lower cholesterol values. Alternatively, the stress level of 5 (the highest stress level) is clearly skewed with higher stress being more likely with higher cholesterol.




```
(H1<-
ggplot(RiskDF,aes(x=Cholesterol, fill=StressLevel))+
geom_density()+

facet_grid(~StressLevel)+theme(axis.text.x=element_blank()) +
   ggtitle("Density Plots: Cholesterol Distribution for Stress level"))
```

Figure 12: Density visualization of cholesterol distribution for stress level.

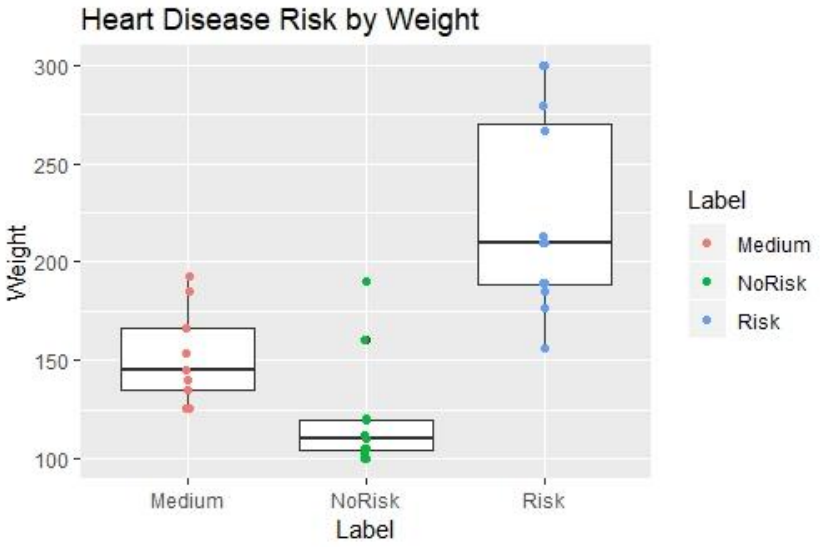
The boxplot family also includes the notched box and the jitter plot. The notched boxplot retains all the attributes of the standard boxplot with the addition of a "notch" that offers the 95% confidence interval about the median. Figure 12 above describes the anatomy of the notch boxplot. The jitter plot emulates the structure of a boxplot in that is shows spread. However, it is distinct because is displays each data point and does not display quartiles. Figure 13 below shows an example of a jitter plot. "Jitter" itself is a pseudo-random value that is assigned to the dots so as to separate them. This allows each data point (dot) to be viewed and prohibits points to be plotted on top of each other.




```
(J1<-ggplot(RiskDF,aes(x=StressLevel, y=Cholesterol, col=StressLevel))+
   geom_jitter(stat = "identity",position = "jitter")+

facet_grid(~StressLevel)+theme(axis.text.x=element_blank()) +
   ggtitle("Jitter Plots: Cholesterol Distribution for Stress level"))
```

Figure 13: Jitter plot illustrating the range and distribution of cholesterol data for each stress level.

In many cases, more than one visual option can be used together. This is known as layering. Figure XXX below illustrates a jitter plot layered on a boxplot in ggplot2.



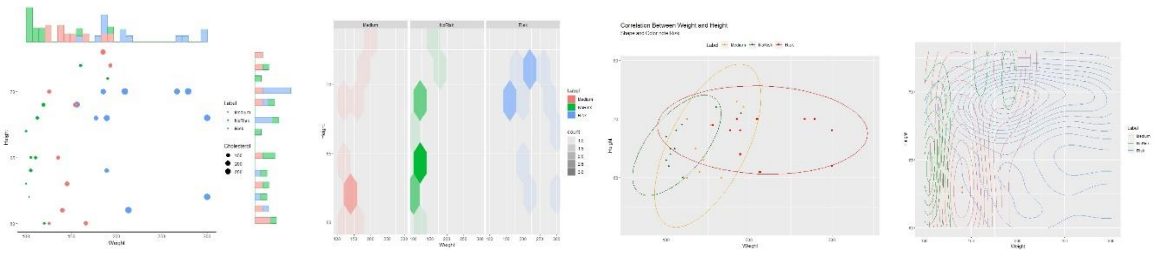

```
(L1<-ggplot(RiskDF,
aes(x=Label, y=Weight))+
  geom_boxplot()+

geom_jitter(position=position
_jitter(.01), aes(color=Label))+
    ggtitle("Heart Disease Risk
by Weight"))
```

Figure 14: Layered visualization using boxplots and jitter plots. Code is to the right and uses ggplot2.

Blending and layering visualization options can often significantly improve the information both contained and communicated. For example, by observing Figure 14 above, there is a visual suggestion (that can be confirmed statistically) that there is likely a statistically significant difference between weights that are associated with the label of “Risk” (highest heart risk) as opposed to those associated with the labels of “Medium” or “NoRisk”. While it is never appropriate to guess at statistical significance visually, the visual graphic can encourage appropriate statistical hypothesis testing.

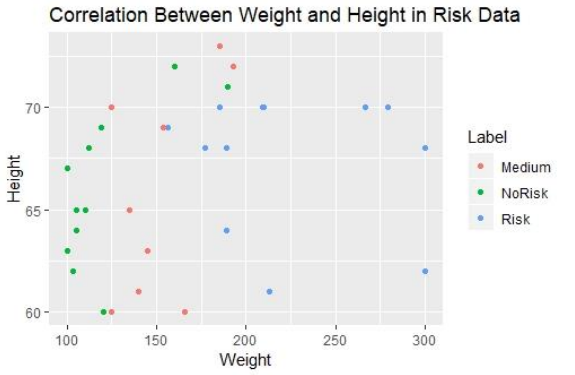
In general, the family of box-plot options can offer considerable insight into distribution, quartiles, and variation. Again, box plots are ideal for quantitative (numeric) data. They can also be employed to investigate potential outliers, skew, and modality.

The Scatter Family

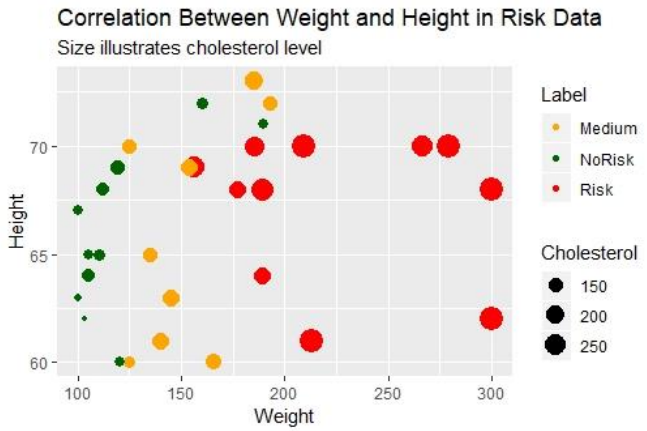


Standard scatterplots are great for visualizing two variables at once and for visualizing possible relationships between those variables. Scatterplots are often used to assess linear and non-linear relationships. Within the scatter family, there are other variations and blends that each add or focus on further dimensions or attributes. The bubble plot, for example, is very similar to the standard scatterplot, but adds another dimension – the size of the data point or bubble. Figure 15 below illustrates a standard scatterplot (top) and a bubble plot option (bottom). The x axis is the “Weight” variable from the Heart Health Dataset, the y axis is the “Height” variable. Color is also used as an

additional dimension and represents the “Label” of the heart health risk: NoRisk, Medium, or Risk. By adding this extra “bubble size” dimension, the visualization reveals that weight and height may be more strongly linearly correlated for risk categories of NoRisk and Medium. This observation may encourage further visual and statistical analysis.




```
(S0<-ggplot(RiskDF, aes(x=Weight,
y=Height, color=Label))+
  geom_point()+
  ggtitle("Correlation Between
Weight and Height in Risk Data"))
```



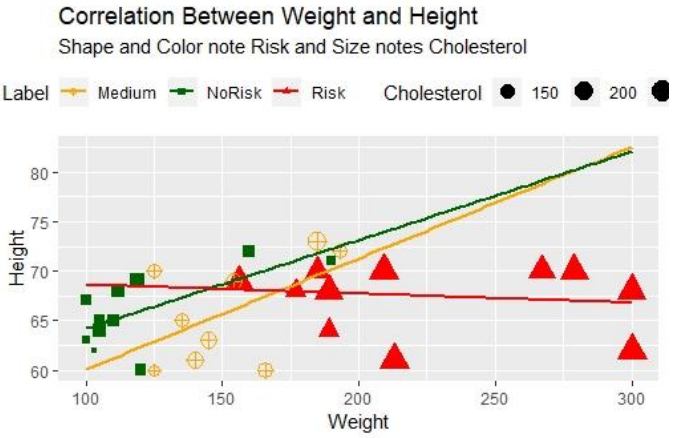

```
(S1<-ggplot(RiskDF, aes(x=Weight, y=Height,
color=Label))+
   geom_point(aes(size=Cholesterol))+

scale_color_manual(values=c('orange','dark
green', 'red'))+
   labs(title="Correlation Between Weight
and Height in Risk Data",
        subtitle="Size illustrates cholesterol
level"))
```

Figure 15: Top: Standard scatterplot, color-coded for Risk. Bottom: Example of bubble chart with added dimensions of color (for Risk Label) and size (for Cholesterol).

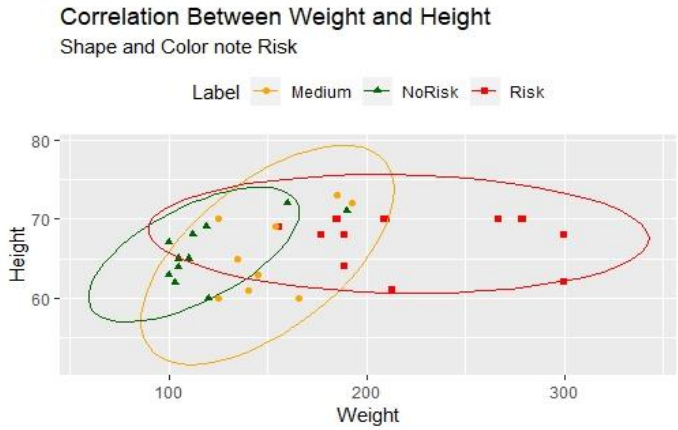
Visual illustrations, such as scatter-type plots, may contain additional layers as well as additional elements. By adding further layers and elements, a greater number of dimensions can be simultaneously visualized (still in our confined two-dimensional graph space).

So far, by adding the size attribute to a standard scatter plot, the number of variables represented increased from two (Weight and Height) to three (Weight, Height, and Cholesterol level). My using color, a fourth dimension has been included offering Weight, Height, Cholesterol level, and the Risk Label. Consider the amount of extra information conveyed within the same two-dimensional space when size and color attributes are added. From here, one may choose to increase the information and dimensionality of the visualization by also adding shapes and layers.

Figure 15 (top) illustrates a scatter plot with the addition of size, color, shape, and regression lines. Figure 15 (bottom) offers ellipses rather than regression lines.



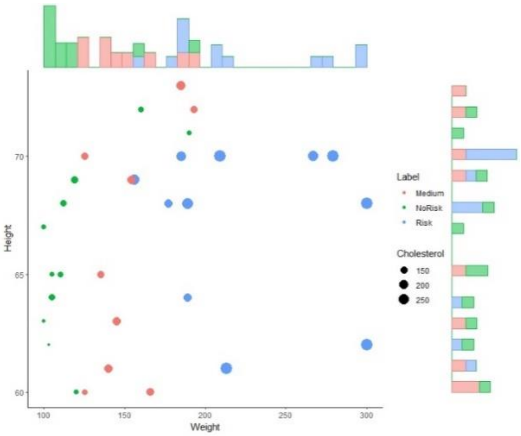

```
(S1<-ggplot(RiskDF, aes(x=Weight, y=Height,
color=Label, shape=Label))+
  geom_point(aes(size=Cholesterol)) +
  geom_smooth(method=lm, se=FALSE,
fullrange=TRUE)+
  scale_shape_manual(values=c(10, 15, 17))+
  scale_color_manual(values=c('orange','dark
green', 'red'))+
  theme(legend.position="top")+
  labs(title="Correlation Between Weight and
Height",
       subtitle="Shape and Color note Risk and Size
notes Cholesterol"))
```



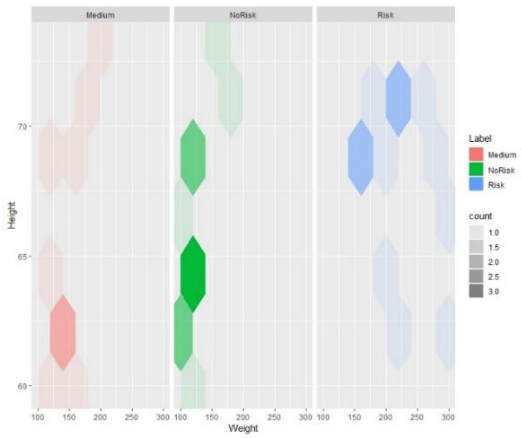

```
(SE1<-ggplot(RiskDF, aes(x=Weight, y=Height,
color=Label, shape=Label))+
   geom_point() +
   stat_ellipse()+
   #geom_smooth(method=lm, se=FALSE,
fullrange=TRUE)+
   #scale_shape_manual(values=c(10, 15, 17))+
   scale_color_manual(values=c('orange','dark green',
'red'))+
   theme(legend.position="top")+
   labs(title="Correlation Between Weight and
Height",
        subtitle="Shape and Color note Risk"))
```

Figure 16: Top: Scatterplot with linear regression lines. Shape, size, and color are added dimensions. Bottom: Ellipses added as an informational element to a scatterplot.

Next, the notion of density (or other measures) may be combined with or utilized in conjunction with Scatter visualizations. Figure 17 (below) illustrates two such options. The first combines a bubble plot with histograms and the second offers a unique density-based scatter method with color depth as an attribute.



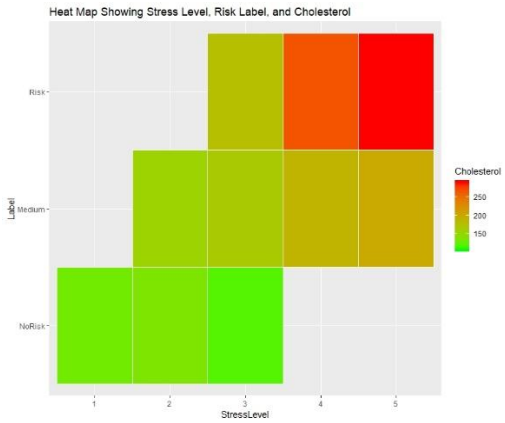

```
library(ggExtra)
##
https://www.rdocumentation.org/packages/ggExtra
/versions/0.9/topics/ggMarginal
SH<-ggplot(RiskDF, aes(x=Weight, y=Height,
color=Label)) +
 geom_point(aes(size=Cholesterol)) +
 theme_classic()
# add marginal histograms
ggExtra::ggMarginal(SH, type = "histogram",
groupColour=TRUE, groupFill = TRUE)
```




```
(SD2<-ggplot (RiskDF, aes(x=Weight, y=Height,
fill=Label)) +
  stat_binhex (bins=5, aes (alpha = ..count..)) +
facet_grid (. ~ Label))
```

Figure 17: Top: Bubble plot with stacked histogram. Bottom: binhex density scatter.

Another member of the scatter family for visualizing two or more dimensions is the heat map. A heat map can be thought of as a matrix in which each cell or location represents a measure of quantity. Color shades and representative colors are often used to visually differentiate between relative differences. Unlike scatterplots, heat maps can be applied to qualitative variables. For example, to create a standard scatterplot, before adding attributes for size, shape, and others, two quantitative variables must be used. These might be weight and height, or cholesterol and BMI, etc. However, a heat map may be applied to two qualitative variables, such as StressLevel and Risk Label, or Political Label and Religious Label, etc. Figure 18 illustrates a heat map for StressLevel, Risk Label, and Cholesterol.



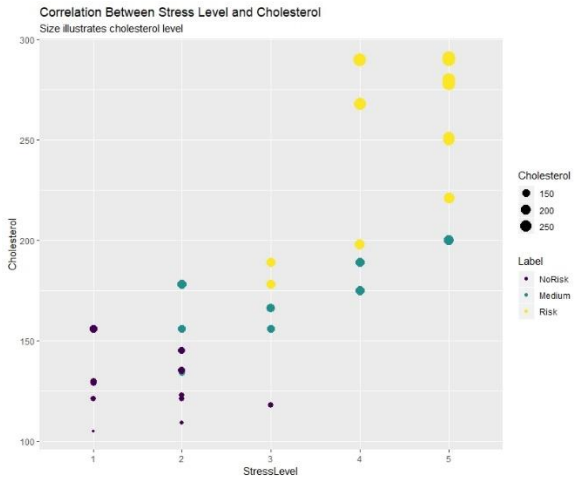

```
(S1_1<-ggplot(RiskDF, aes(x=StressLevel,
y=Cholesterol, color=Label))+
   geom_point(aes(size=Cholesterol))+
   labs(title="Correlation Between Stress Level
and Cholesterol",
        subtitle="Size illustrates cholesterol
level"))
```

Figure 18: Heat map to illustrate StressLevel (x), Risk Label (y), and Cholesterol (color).

A common question is whether a standard scatterplot can be “forced” to use values that are represented with numbers, but that are not actually quantitative. For example, can StressLevel (1 – 5) be used in a scatterplot? StressLevel is actually a qualitative variable and its five possible categories are 1, 2, 3, 4, and 5. While numbers are used to represent the five possible categories, they are not actually numeric values. In fact, they could have been called, “Lowest”, “Low”, “Medium”, “High”, and “Highest”. However, because programming languages cannot know this, it is possible to force the code to create a

scatter plot. Figure 19 shows a scatterplot generated from StressLevel (actually qualitative but represented with numbers), and Cholesterol. Here, because StressLevel is ordered, it offers a sense of "increasing" from left to right.

However, it can be observed that this scatter is actually discrete and that no points can occur between any of the five distinct stress levels. Further, if this same method is applied to qualitative and nominal data (with no order), the results may be misleading and inconclusive. When creating plots, the coder (you) must always understand what is being done, what information is being illustrated, and if that information is being conveyed clearly, accurately, and fairly. To better understand why using qualitative variables as numbers (when they are not), think about the idea that a 4 for one person may be a 3 for another. Qualitative measures, such as rankings, may be subjective and so cannot be compared in the same way that true numbers can.



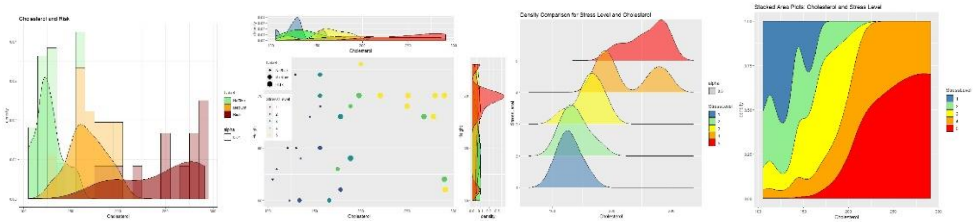

```
(S1_1<-ggplot(RiskDF, aes(x=StressLevel,
y=Cholesterol, color=Label))+
   geom_point(aes(size=Cholesterol))+
   labs(title="Correlation Between Stress Level
and Cholesterol",
        subtitle="Size illustrates cholesterol
level"))
```

Figure 19: Illustration of using a scatter bubble plot on a qualitative and ordinal variable (StressLevel).

Another type of scatterplot is the Lag Plot. Lag Plots plot two variables (x and y) such that the "y" variable is "lagged" (temporally) behind the "x" variable. This concept is the beginning of another type of visualization family know as Time Series plots. Therefore, the discussion of Lag Plots will be included in the Time Series section.

In general, the scatter family, and related additional options, can offer very rich analyses. Scatter plots can reveal and illustrate relationships between variables, both linear and non-linear. In addition, a scatter plot may be used to show many dimensions at once, through the use of size, shape, color, and layering. While investigating and illustrating relationships is often informative, other cases may require a deeper look into the distribution or density of specific variables. The next section will discuss the Density Family.

The Density Family



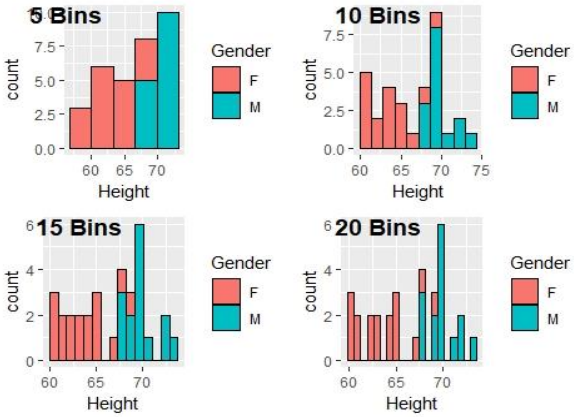
Density plots are appropriate for quantitative data and are excellent for illustrating the "shape" (or distribution) of the data. The shape or distribution of data can offer many insights into the population from which the sample data was collected, as well as other data traits, such as whether the data might be a mixture of distributions. For example, by observing a density plot, it may be possible to determine if the data is normal, uniform, skewed, beta, gamma, bimodal, etc. If a density plot appears to be bell-shaped, this may imply that the underlying population of the data may be normally distributed. Density plots also illustrate the center and variation of the data.

Density plots include several common plots such as histograms and density area plots, as well as advanced plots such as stacked area graphs, density ridge plots, density kernel plots, and blended or compound plots. Several density plots, one for each variable, may also be plotted on the same graphic using a transparency (alpha) option that permits visualizations to overlap.

Histograms are constructed by binning quantitative data into a selected number of groups or categories, and then representing the relative frequency of each bin. Bars are used to represent each bin and so the greater the number of groups (bins or categories) selected, the greater the number of bars in the histogram. Binning is a discretization method, which means that it can be used to categorize quantitative data into a finite number of groups. In other words, binning can be used to create a new qualitative variable from a quantitative variable. The following example will illustrate binning, as well as a histogram that displays the distribution of the Height variable from the Heart Risk dataset. Recall that the Height variable is quantitative (numeric).

Choosing the number of bins or categories (and so the number of bars in the histogram) is not an easy decision and often experimentation with various numbers of bins can assist with not only discerning more about the data, but also in improving the visualization.

Figure 20 shows four histograms, each with a different number of bins selected. Notice that the "fill" (or color-coding) is filled by gender. What extra information does this reveal? Which number of bins seems best? For example, the choice of five bins does not seem to show the interesting fact that this data variable (Height) is a mixture of distributions. The bin choice of ten (or fifteen) shows enough separation and detail to illustrate the idea that gender is affecting the distribution and that male and female heights each have their own distribution, mean, and variance. The use of color and bin number both work together to enable this view. The bin choice of twenty is fine, but does not appear to add further insight. What information might have been overlooked if color had not been used to fill in the bars by gender?




```
library(ggpubr)
plist<-list()
for(i in c(5, 10, 15, 20)){

  (plist[[i]]<- ggplot(RiskDF, aes(Height))+
    geom_histogram(aes(fill=Gender), bins=i, col="black"))
}
(figure <- ggarrange(plist[[5]], plist[[10]], plist[[15]],plist[[20]],
                     labels = c("5 Bins", "10 Bins", "15 Bins", "20 Bins"),
                ncol = 2, nrow = 2))
```

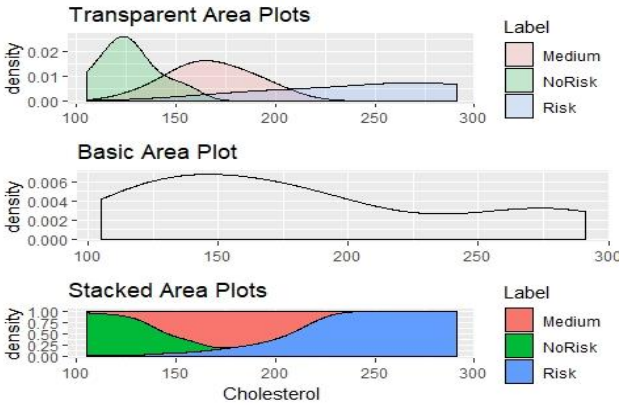
Figure 20: Four histogram options with different bin numbers. R code from ggplot2 is to the right.

Area plots are based on a line graph that represent the density and distribution of the data variable being visualized. Area Plots offer several options, such as a basic plot that shows the overall distribution of a single variable, overlapping area plots from a single variable but colored with respect to a selected qualitative variable, stacked plots, and combinations.

Ares plots are excellent for discovering and illustrating that a single variable may be a mixture or blend of one or more populations. For example, Figure 21 below, illustrates three Area Plot options to describe Cholesterol and Risk from the Heart Health Dataset. First, notice the middle graphic in the Figure. The middle graphic is a very basic area plot for Cholesterol (only) and shows a slight skew in the data and a possible bimodal distribution.

Next, consider the top graph. It is also an area plot for Cholesterol. However, by adding the color and transparency options, a basic area plot can transform into a more robust plot that reveals the potential for multiple distributions. The first plot illustrates that there may be different cholesterol level populations, each for a given risk factor.

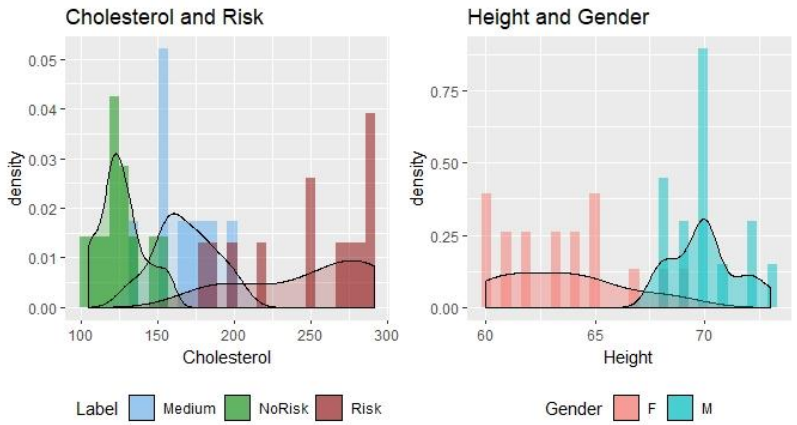
The third plot in Figure 21 illustrates a Stacked Area Plot that is colored by Risk label. The stacked plot shows that the lowest risk group (NoRisk) has its highest distribution of data closest to cholesterol levels between 100 and 125 and then it begins to reduce. The Medium Risk group has its peak between cholesterol levels of 160 – 180, and the highest risk group has a skewed distribution that is most prominent between 225 and 300.




```
(A1<-ggplot(RiskDF,aes(x=Cholesterol, fill=Label)) +
  geom_density(adjust=1.5 , alpha=0.2)+
   labs(title="Transparent Area Plots")+
    theme(axis.title.x = element_blank()))
(A2<-ggplot(RiskDF, aes(x=Cholesterol))+
    geom_density()+
    labs(title="Basic Area Plot")+
    theme(axis.title.x = element_blank()))
(A3<-ggplot(data=RiskDF,aes(x=Cholesterol,
group=Label, fill=Label)) +
    geom_density(adjust=1.5, position="fill")+
    labs(title="Stacked Area Plots"))
(figure2 <- ggarrange(A1, A2, A3, ncol = 1, nrow = 3))
```

Figure 21: Examples of three different types of area plots.

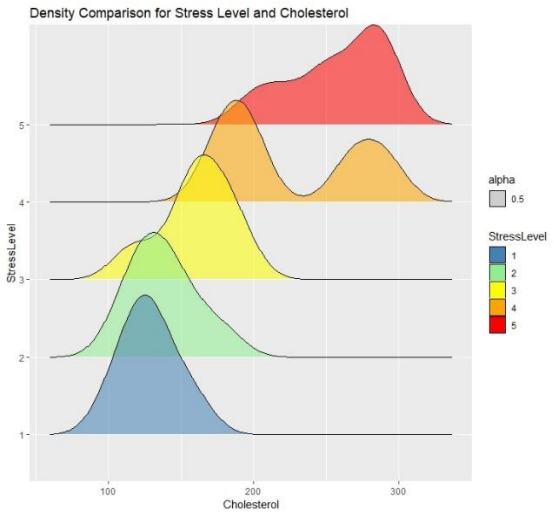
Another type of area plot is the Kernel Density Plot. Kernel density plots produce a smooth curve that estimates the probability density function of a continuous variable from a sample that is likely to comprise some error (Howell, 2013). Kernel plots can be used to better understand and illustrate the distribution of the data. Figure 22 below shows two histograms, each with an added layer that is the kernel density plot.




```
(HD1<-ggplot(RiskDF, aes(x=Cholesterol,  fill=Label))
+
 geom_histogram(aes(y=..density..), alpha=0.5,
position="identity")+
   geom_density(alpha=.2)+

scale_fill_manual(values=c('steelblue2','green4',"red
4"))+
   theme(legend.position="bottom")+
   labs(title="Cholesterol and Risk"))

(HD2<-ggplot(RiskDF, aes(x=Height,  fill=Gender)) +
   geom_histogram(aes(y=..density..), alpha=0.5,
position="identity")+
   geom_density(alpha=.4)+
   theme(legend.position="bottom")+
   labs(title="Height and Gender"))

(figure3 <- ggarrange(HD1, HD2, ncol = 2, nrow = 1))
```

Figure 22: Histograms with density kernel smooth estimate layer.

The Density Ridge Plot is another excellent example of a collection of plots that visualize qualitative categories with respect to a quantitative variable. The following example shows the distributions for all five stress levels with respect to cholesterol level.




```
### Density Ridge
#install.packages("ggridges")
library(ggridges)
(DR<-ggplot(RiskDF, aes(x = Cholesterol, y =
StressLevel)) +
  geom_density_ridges(aes(fill = StressLevel,
alpha=.5)) +
    ggtitle("Density Comparison for Stress Level and
Cholesterol")+
  scale_fill_manual(values = c("steelblue",
"lightgreen", "yellow", "orange", "red")))
```

Figure 23: Density Ridge Plot illustrating the distribution of cholesterol levels per each Stress level.

In general, area-type plots can be very powerful in illustrating data variable shapes (distributions), relationships, and mixtures. While area plots can only be used directly on quantitative data, they can be augmented with qualitative data. For example, Cholesterol is quantitative, but the Risk level that it can be color-coded with, is qualitative.

Graph types can be combined in any creative way that maintains accuracy for the information being conveyed. For example, one may wish to create a color-coded Bubble-type scatter plot that is accompanied by two area-density plots. Figure 24 illustrates this option.

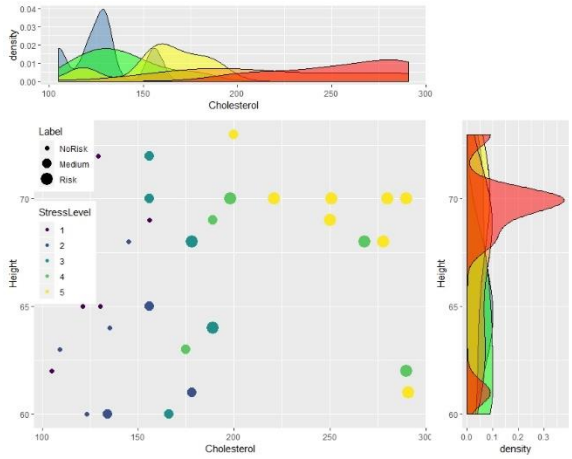
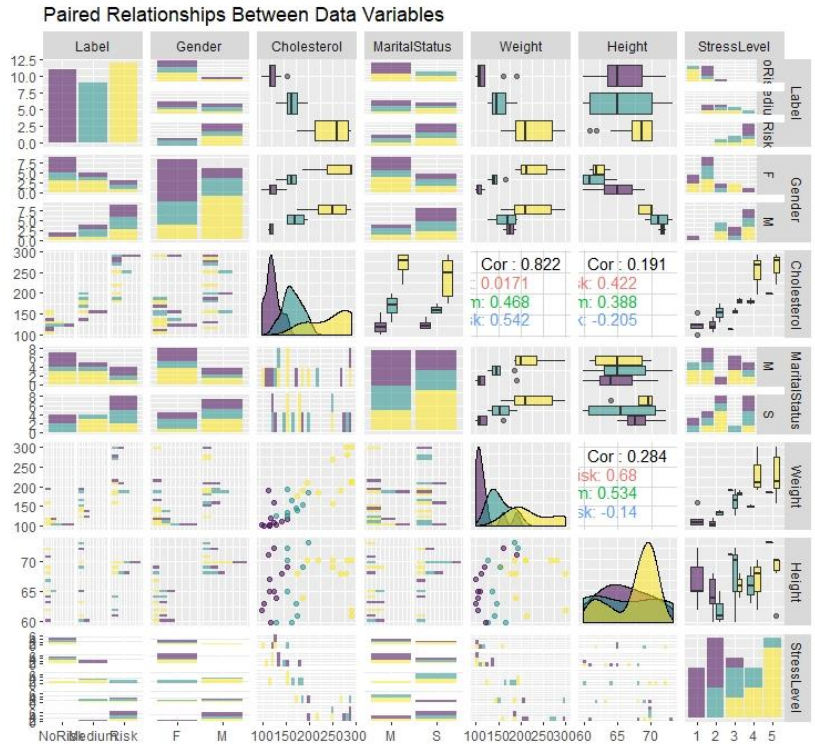


Figure 24: Bubble-Scatter combined with two color-coded area density plots. The code for this plot is large and so will be available on the website.

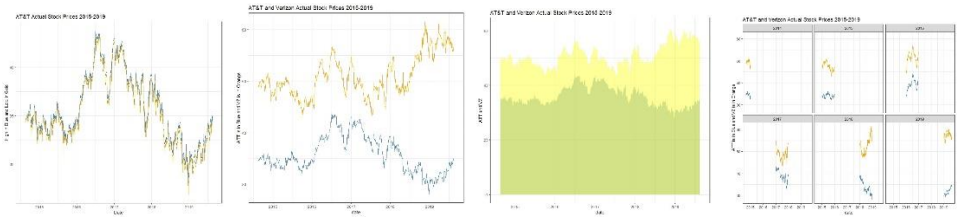
Before considering a few more complicated visualization families, it is worth noting that programming languages, such as R, sometimes offer options for paired visualization. In other words, it is possible to quickly take a glance at a visualization for each pair of variables (qualitative or quantitative) within a data. Use caution with this method. There the number of variables is too large, the result may not be useful. Figure 25 below illustrates an example of this option.




```
#install.packages("GGally")
library("GGally")
(NewDF<-RiskDF[,-c(1,2,4,7)])
(str(NewDF))
(GGP<-ggpairs(RiskDF,mapping
= ggplot2::aes(color =
RiskDF$Label, alpha=.5))+
   ggtitle("Paired Relationships
Between Data Variables"))
```

Figure 25: A quick visualization of the relationships between each pair of variables in the Heart Heath Dataset.

The Time Series Family



Up to this point, all of the visualization types and blends have been appropriate for record datasets. Recall that data in "record format" has rows and columns, where each row represents an instance or observation and each column represents a variable. The Heart Heath Dataset used for all of the above examples is a record-format dataset.

Consider the following sample of two other datasets. The first is AT&T stock data and the second is Verizon stock data. Both dataset have the same exact (and case-sensitive) variable names and the same exact dates. This stock data was downloaded directly from https://finance.yahoo.com and spans all business days between August 18, 2014 and Aug 16, 2019.

date	stockname	open	high	low	close	volume
8/18/2014	ATT	34.87	34.9	34.57	34.65	18496100
8/19/2014	ATT	34.69	34.69	34.33	34.48	20478400
8/20/2014	ATT	34.54	34.56	34.44	34.53	12549500
8/21/2014	ATT	34.53	34.75	34.51	34.64	15931900
8/22/2014	ATT	34.56	34.63	34.35	34.5	14285400
8/25/2014	ATT	34.52	34.66	34.45	34.51	17321500
8/26/2014	ATT	34.6	34.65	34.46	34.5	14805600
8/27/2014	ATT	34.59	34.79	34.54	34.75	14853400
8/28/2014	ATT	34.66	34.75	34.56	34.74	10539700
8/29/2014	ATT	34.73	34.96	34.62	34.96	12709000
9/2/2014	ATT	34.93	35	34.7	34.84	12691200
9/3/2014	ATT	34.95	35	34.82	34.97	13026600
9/4/2014	ATT	34.98	35	34.82	34.94	12483900
9/5/2014	ATT	34.95	35.26	34.88	35.15	17709000

date	stockname	open	high	low	close	volume
8/18/2014	Verizon	49.05	49.14	48.64	48.78	12682100
8/19/2014	Verizon	49.03	49.04	48.27	48.7	17348700
8/20/2014	Verizon	48.82	48.97	48.6	48.82	10685300
8/21/2014	Verizon	48.86	49.15	48.73	48.87	10188900
8/22/2014	Verizon	48.83	48.98	48.52	48.64	9991400
8/25/2014	Verizon	48.68	49.16	48.68	49.15	9913500
8/26/2014	Verizon	49.21	49.29	49.02	49.25	10363400
8/27/2014	Verizon	49.29	49.46	49.21	49.43	11500800
8/28/2014	Verizon	49.38	49.5	49.2	49.41	8018800
8/29/2014	Verizon	49.43	49.82	49.4	49.82	11259700
9/2/2014	Verizon	49.82	50	49.5	49.77	8724400
9/3/2014	Verizon	49.94	50	49.69	49.88	9100900
9/4/2014	Verizon	49.93	49.94	49.51	49.72	9472900
9/5/2014	Verizon	49.73	50.03	49.56	49.94	11920900

Figure XXX: Actual historical stock prices for AT&T and Verizon collected from https://finance.yahoo.com. Spans all business days between August 18, 2014 and Aug 16, 2019

Like the Heart Heath Dataset, the above stock datasets are also in record format. Each row represents one day. The columns represent the variables of date, open (the amount that the stock was when the stock market opened), high (the highest price of the stock on that day), low (the lowest stock price on that day), close (the price that stock closed at when the market closed on that day), and volume (the number of shared bought and sold – traded – on that day.)

A key and critical difference between the stock datasets and the Heart Health Dataset is that the stock datasets both contain a variable for the date. The "date" variable enables temporal (time) analysis and visualization to be performed on this dataset. The existence of a temporal variable, "date" in this case, also allows for the notion of order. The row on 8/18/2014 occurred before the row on 8/22/2014, and so on. This type of timed and ordered data is known as Time Series data.

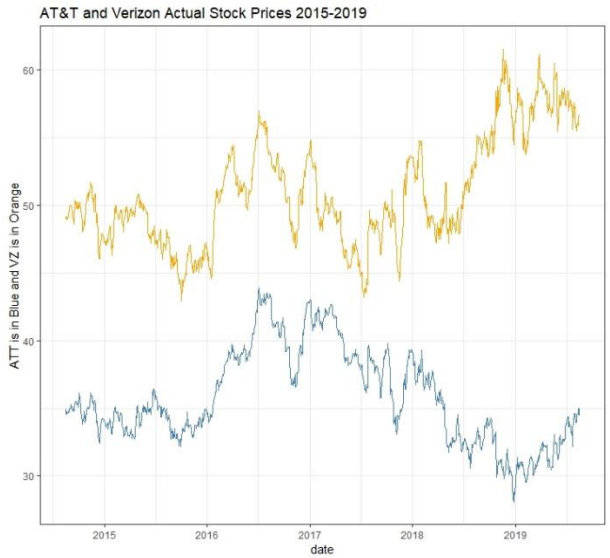
Time Series data can be visualized with any graphic that it appropriate for quantitative data. This might include the histogram, area or density plots, etc. However, unlike other non-time-series datasets, it can also be plotted over time and in order.

Having temporal data allows for visualizations that can investigate data trends and changes over time that may reveal significant information. Before building Time Series visualizations, it is important to assure that the temporal variable(s) in the datasets are properly typed as a Date data type. Recall that understanding (and correcting if necessary) the data types in a dataset is critical. Missing this important step can cause odd and unexpected outcomes. Figure 26 shows an example of reading in a temporal dataset for AT&T stock data from 2014 – 2019 and then changing the column called "Date" to an actual temporal variable type.

```
filename="ATT_5year.csv"
StockDF_ATT <- read.csv(filename, header = TRUE)
(head(StockDF_ATT))
StockDF_ATT$date<-as.Date(StockDF_ATT$date, format="%m/%d/%Y")
(str(StockDF_ATT))
```

Figure 26: Small example in R that updates a column called Date to a true Date data type.

Because temporal datasets (datasets with time or date based variables) offer order and progression, they can be visualized using "time" as the x-axis; values over time. Figure XXX below illustrates the AT&T Stock Dataset values for the stock high and low prices over time.




```
(ATT_VZ<-ggplot()+
   geom_line(data=StockDF_ATT, aes(x =
date,y=StockDF_ATT$high), color="steelblue") +
   geom_line(data=StockDF_VZ, aes(x = date,
y=StockDF_VZ$high), color="orange") +
   ylab("ATT is in Blue and VZ is in Orange")+
   theme_bw()+
   ggtitle("AT&T and Verizon Actual Stock Prices
2015-2019"))
```

Figure 27: Time Series plot for AT&T and Verizon stock prices (highs) from 2014 – 2019 (from Yahoo Finance open data)

Time Series plots can be created using Line Plots, and can also be created using other graphic options such as Area Plots. As with other graphic options, it is also possible to combine, to layer, and to facet. Figure 28 will illustrate a few examples. Think about the differences and potential benefits of each.

Figure 28: Left: Temporal stacked Area Plot. Right: Facetted

As with all visualization families, there are thousands of options and many can become very complex. For example, the StreamGraph and the Gantt Chart are two excellent graphics options for illustrating temporal data. The Stream Graph is very much like a stacked area plot that progresses (changes) over a time variable. Therefore, the x-axis represents a change over time, and the y-axis represents the area or density of each group over that time period. A Gantt Chart is a temporal horizontal bar chart that illustrates the progression of a set of tasks or items over time. The Gantt chart was developed in 1917 by Henry L. Gantt, an American engineer and social scientist. Figure 28 shows examples of each. However, because the code for this chart types is beyond the scope or intention of this chapter, the reader is invited to begin to develop code and to become self-supported as a visual designer.

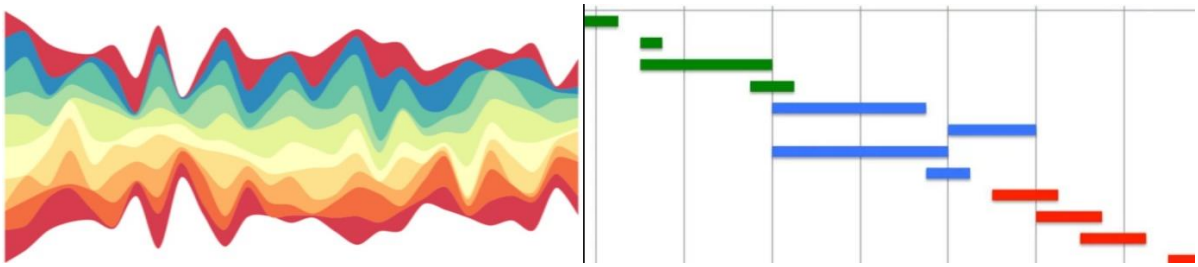
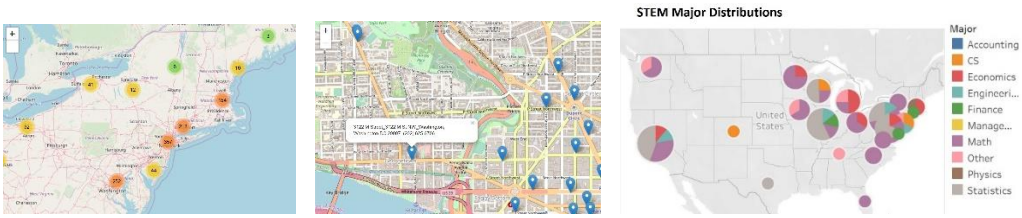


Figure 28: Stream and Gantt basic examples.

The study of Time Series data is also beyond the scope of this chapter. However, it is interesting to observe that if data contains a temporal variable (such as the date), not only can trends or movement over time be illustrated, but other features such as differences over time can be considered. For example, one might wonder if a stock tends to increase more often than it deceases over time. Think about how you might update a dataset to include a new variable that can then be visualized to answer this question.

The most common visualization families for record or temporal data have been discussed. However, not all data is in record format and not all visualizations are limited to an x-y axis. Further families include the Geospatial or Maps Family and the Networks Family. The Maps Family includes any type of visualization that illustrates location. The Networks Family includes any types of illustrations that connect data points to other data points via relationships. The Networks family includes examples such as the basic node and edge graph-based network (directed or undirected), the Tree Map, and also hierarchical visualizations such as the dendrogram. The follow sections will introduce that discuss these families.

The Maps Family



Geospatial visualizations are any image, static or interactive, that offers an illustration of location-based data. Examples are physical maps, Google Maps, town, city, county, state, country, and world maps. Geospatial data (location data) can be combined and layered with any information that relates to location. For example, adding date or time data can create Geospatial Temporal visualizations, such as the movement of ground forces over a section of disputed land, changes in volcanic activity over time, changes in population growth or density over time in any locations, and so on. The possible applications are limitless. Geospatial data may also be combined with other visualizations, such as placing pie graphs as a layer of all states in the US to visualize the percentage of types of college majors for graduate students in STEM areas.

Like time series data, geospatial data can be in record format, but it must contain geospatial locators. Without a location variable, one cannot have geospatial data. There are several options for creating (or gathering) Geospatial data location variables. Common geospatial data locators are latitude and longitude. Others include zip code (for some but not all areas), complete address, and geocodes such as FIPS (Federal Information Processing Standards.) For this chapter, the focus will be on either zip code or latitude and longitude.

Geospatial (or map data from this point forward) also includes other attributes such as scale or scalability, area measure options, direction, distance, and angles. In addition, GIS systems are often employed for more involved and complex Geospatial data visualization and analysis. GIS stands for geographic information system. GIS is a system designed to collect, store, manipulate, analyze, manage, and present spatial or geographic data. GIS systems and methods also include special applications that can be used for related analysis. While GIS is beyond the scope of this chapter and book, it is important to be aware of that it stands for. GIS systems and applications are not necessary for mapping data or for creating layered and geospatial visualizations. Geospatial analysis and GIS often also involve advanced areas such as remote sensing, hyperspectral imaging, and other land sensing.

There are three core types of geospatial data, vector data, raster data, and tabular (or record) data. Vector data uses points, lines, and other polygons (shapes) to represent and visualize locations. Raster data uses scanned images (such as those captured from airplane, ships, and ground), as well as other scanned photographs to render and represent locations. Record data comes from census collections, state, county, and country data, or other such collected data methods. Record data can be more easily combined and layered with other types of data. We will focus on only record data that contains geospatial variables (such as latitude and longitude).

To this point, all examples have been created using R and ggplot2. However, it is often important to choose the right tool for the job. While there are many applications and languages from GoogleVis, to Plotly, to Tableau, to Leaflet for illustrating geospatial and geotemporal data, this section will focus on two. The first is Leaflet (in R) and the second is Tableau. Because maps are collections of polygons (counties, states, and countries to humans), it is often necessary to download and utilize shapefiles. There are also many other complexities associated with creating map-type visualizations.

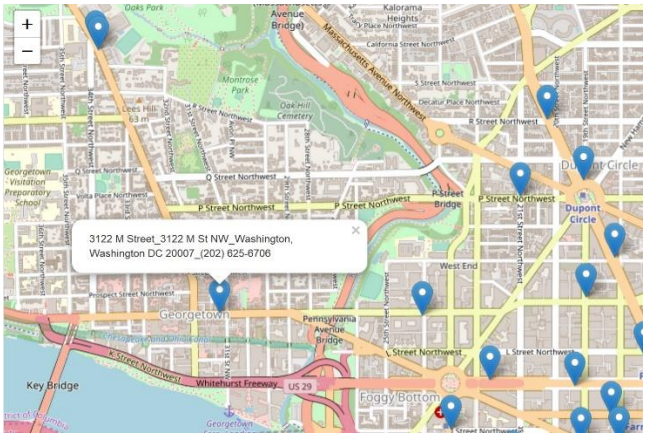
Leaflet is an open source JavaScript library (and can work in R). It is excellent for building interactive, layered, and advanced geospatial visualizations. Tableau is very different from Leaflet. Tableau is not open source and is an application rather than a library. While Leaflet is “coded” in R, Tableau is used as a point-and-click desktop application. Because Tableau is free for students and faculty and can render professional visualizations with no coding requirements, it is also an interesting option.

The following dataset contains variables for latitude and longitude and represents Geospatial data that can be mapped.

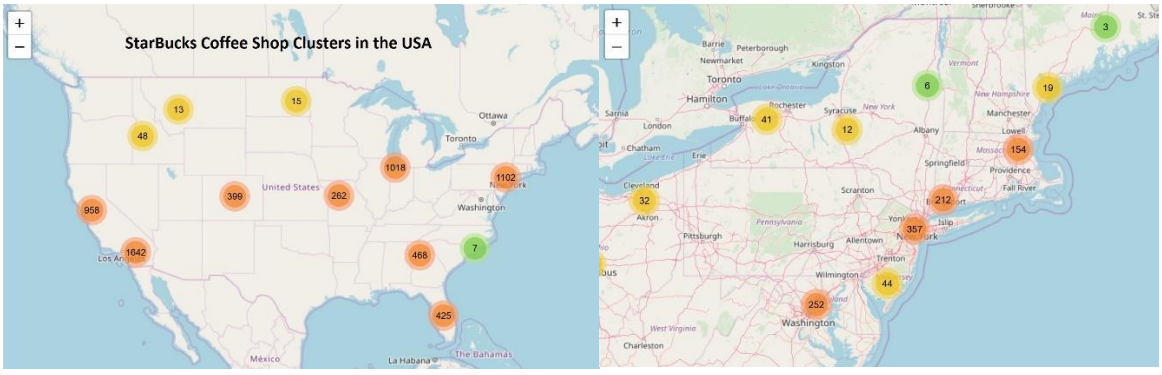
long	lat	name	address
-149.894	61.21759	Starbucks	601 West Street_601 West 5th Avenue_Anchorage, Alaska 99501_907-277-2477
-149.905	61.19534	Starbucks	Carrs-Anchorage #1805_1650 W Northern Lights Blvd_Anchorage, Alaska 99503_907-339-05(
-149.752	61.2297	Starbucks	Elmendorf AFB_Bldg 5800 Westover Avenue_Anchorage, Alaska 99506
-149.864	61.19525	Starbucks	Fred Meyer - Anchorage #11_1000 E Northern Lights Blvd_Anchorage, Alaska 995084283_90
-149.838	61.13751	Starbucks	Fred Meyer - Anchorage #656_2300 Abbott Road_Anchorage, Alaska 99507_907-365-2000
-149.909	61.13995	Starbucks	Fred Meyer - Anchorage (Dimond) #71_2000 W Dimond Blvd_Anchorage, Alaska 995151400_
-149.736	61.19533	Starbucks	Safeway-Anchorage #1817_7731 E Northern Lights Blvd_Anchorage, Alaska 99504_907-331-1
-149.821	61.2156	Starbucks	Safeway - Anchorage #520_3101 PENLAND PKWY._Anchorage, Alaska 99508
-149.845	61.13806	Starbucks	Safeway-Anchorage #2628_1725 Abbott Rd_Anchorage, Alaska 99507_907-339-2800
-149.973	61.17669	Starbucks	ANC Anchorage_5000 W. Int'l Airport Rd._Anchorage, Alaska 99502_907-243-4331
-149.864	61.14473	Starbucks	Old Seward & Dimond_1005 E Dimond Blvd_Anchorage, Alaska 99515_907-344-4160
-149.836	61.18082	Starbucks	Tudor & Lake Otis- Anchorage_2421 East Tudor Road_Anchorage, Alaska 99507_907-561-164

Figure 29: Open sources StarBucks locations around the US. The Geospatial variables are Longitude (long) and Latitude (Lat).

Example in Leaflet and R




```
library(leaflet)
StarBucksDF<-
read.csv("StarBucksLocationsUS_Lat_Long.csv")
head(StarBucksDF)
str(StarBucksDF)
DC_Bucks <- subset(StarBucksDF, ((long > -77.1 &
long < -76.9 ) & (lat>38.86  & lat < 38.95 )))
head(DC_Bucks)

(SBMap1 <- leaflet() %>%
  addTiles() %>%
  addMarkers(data = DC_Bucks, lng = ~ long, lat = ~
lat, popup = DC_Bucks$address))
```




```
(SBMapUS <- leaflet() %>%
   addTiles() %>%
   setView(-90.07, 40.92, zoom = 4) %>%
   addCircleMarkers(data =StarBucksDF, lng = ~ long, lat = ~ lat, radius = 3,
                    #color = ~ ifelse(Some$Category == 'something', 'red', 'blue'),
                    clusterOptions = markerClusterOptions()))
```

Figure 30: Leaflet/R for geo data.