# Initial Posts

|  |
| --- |
| A scatter plot shows the relationship of multiple variables. Typically it is used to show 2 variables where one variable’s values are plotted on the y-axis and the other on the x-axis. Other variables can be represented however by adding color based on another variable, making the size of the markers different based on another continuous variable, etc. A scatterplot is great at providing a high-level overview of the relationship between variables but does not give a whole lot of detail about the exact nature of the relationship. |
| Some ways to view a relationship between variables:   * **Scatterplot**   + matplotlib.pyplot.scatter   + seaborn.scatterplot   + pandas.DataFrame.plot.scatter      * **Jointplot**   + matplotlib     - gridspec.GridSpec     - pyplot.scatter     - pyplot.hist   + seaborn.jointplot      * **Correlation Heatmap**   + pandas.DataFrame.corr().style.background\_gradient(cmap=’whatever’)   + seabron.heatmap   + matplotlib.pyplot.imshow      * Binning data and plotting the average for each bin for the ‘x’ variable and percentile for each bin for the ‘y’ variable.   + numpy.arange (to make bins)   + numpy.digitize (to create index list to group DataFrame by)   + pandas.DataFrame.groupby (to group DataFrame)   + numpy.unique & numpy.cumsum (to create cdf array)   + bisect.bisect\_left (to get percentile of cdf array) |
| Pearson's r: a statistic that measures the correlation between variables. It varies from -1 to 1 with 0 indicating no correlation, -1 indicating a perfect negative correlation, and 1 indicating a perfect positive correlation. When using Pearson's r, the statistic assumes the data is normally distributed so before using this statistic you would need to determine if this assumption is valid. If not you can utilize a function (log, inverse, etc.) to turn the variables into a normal distribution. If that doesn't work then you would need to perform a non-parametric correlation statistic.  Pearson's r formula:    Spearman's rho: a statistic similar to Pearson's r but first new variables are created from the X & Y variables which calculate the rank of each variable. Pearson's r is then calculated for the ranked variables. Because the variables are ranks, this makes the statistic non-parametric and does not assume the data to be normally distributed.  Spearman's rho formula:    Kendall's tau: a statistic similar to Spearman's rho and is also non-parametric. This statistic goes a step further than just ranking the variables where concordant pairs, discordant pairs, and ties are totaled based on the ranked variables. When it comes to performing non-parametric correlation, Spearman's rho is less computationally complex and is more well known in the Data Science community but Kendall's tau is debatably more accurate.  Kendall's tau steps:  Create new ranked variables from the X & Y variables.  Sort the X variable ascending keeping the X:Y pair intact. This means that Y will arrange based on the sorted X.  Loop through the Y variable to find concordant, discordant, and tie values.  Concordant pair: number of values succeeding the current value in the column that are greater than the current value  Discordant pair: number of values succeeding the current value in the column that are less than the current value  In scipy.stats.kendalltau, ties are accounted for in the variable but are ignored when there is a pair-wise tie.  Kendall's tau formula: |
| **Non-Normal Data Strategies**  Wanted to query the class and professor Shankar Parajulee on ways to address non-normal data. From what I've researched, below are some steps for addressing data that is non-normal.   1. What is the test statistic that you're using? Is it robust to the assumption of normality?    * If so, proceed with the test    * If not, go to step 2 2. What is the sample size?    * If small, you could do a non-parametric test, bootstrapping, or maybe consider a non-central limit theorem (Chebyshev's theorem)    * If medium-large, you could potentially try transforming the data to see if it follows a normal distribution. If that fails you can try a non-parametric test or bootstrapping.   Other thoughts:   * I wondered, why not just always use bootstrapping or non-parametric tests and avoid assumptions made about the distribution? What I have seen from other posts online is that non-parametric testing has less power meaning that it could fail to reject the NULL hypothesis when in reality it should. Vice-versa bootstrapping runs the risk of overestimating the effect and could lead to a NULL hypothesis being rejected when in reality it should have been accepted. This due to the fact that when bootstrapping, you are considering the sample to a be a good representation of the population. The higher the sampling error, the more flawed your bootstrapped estimations will be. * Also, other theorem's like Chebyshev's theorem produce wider confidence intervals for critical values selected. For example, for a normal distribution, 95% of the data falls within 2 standard deviations whereas for Chebyshev's ,which can be used for any distribution, assumes that 75% of the data falls within 2 standard deviations. To get to 95% confidence using Chebyshev's, this would be about 4.5 standard deviations.   Conclusion: Try to use tests robust to the assumption of normality else, try to assume normality if possible or transform the data to a normal one if the sample size is large enough. If that fails, based on the scenario use other methods like non-parametric testing or bootstrapping. |
| **Statistic Estimation**  Estimation is the process of calculating some statistic...or statistic(s) given observed data and then determining how likely it is that the statistic...or statistic(s) for the sample are good estimators for the population. For example:   1. You obtain a sample of data and calculate its median, mean, standard deviation, or whatever statistic you want. 2. Observing the sample data, you think the data looks Normal, Exponential, Gamma, or whatever. You then want to test whether the statistic...or statistic(s) from the sample is a good estimator for the population distribution you think the sample comes from. To test this, we can randomly sample a known Normal, Exponential, Gamma, or whatever using parameters needed to calculate the known distribution. For each random sample created, store the value for the statistic...or statistic(s) that you want to estimate. These values are called a sampling distribution. 3. Let's say we're using the initial sample created from observed values to estimate median for a population. For each random sample created:    * MSE = sum( (random sample median - observed sample mean)^2 ) / number of samples taken    * RMSE (aka Standard Error) = sqrt( MSE )   A high RMSE indicates that the estimator for the sample may not be a good estimate for the population distribution you assumed the sample to come from given the population has the parameters coming from the initial sample. The lower the RMSE, the more confidence we gain that the initial sample estimator is a good estimate for the population distribution you assumed the sample to come from. |

# Replies

|  |
| --- |
| Dan Clayton: exactly! Using different colors for another variable might give insight if there is another variable that is causing an effect on the outcome.  Sameer Nepal: Very true, a scatterplot can also definitely give insight on outliers. Scatterplots are a great first step in EDA. |
| I love Tableau's geographic mapping ability! They do a really good job of making geo mapping look good in reports and also very easy to do. They example graph you mentioned sounds like it would be a perfect way to potentially increase delivery efficiency. |
| Nice Myranda! I just started using Seaborn in this semester and I'm really starting to like it. From what it looks like, Seaborn takes things that are somewhat complex to do in Matplotlib and makes it easier to do. For the Jointplot example, Seaborn behind the scenes is actually utilizing Matplotlib to create create 3 subplots and combine them together. |
| Thanks for clarifying and also providing the cheat sheet. Much appreciated professor Parajulee! |
| Nice article Yousof. I agree with a lot of points made in the article. Specifically where it talks about why visualizing data is so important and useful. I also agree with the points made about not trying to make visualizations that look "cool" but rather try to create visualizations that tell the best story about the data and give the most insight.  I'm sort of an anti-pie-graph guy though lol. So don't agree that a pie-graph should be in the top 21. I get it though, it is a very well-known graph. |