Visualization

Prof. Murillo
Computational Mathematics, Science and Engineering
Michigan State University

I am away October 30 – November 3, but class will continue normally.

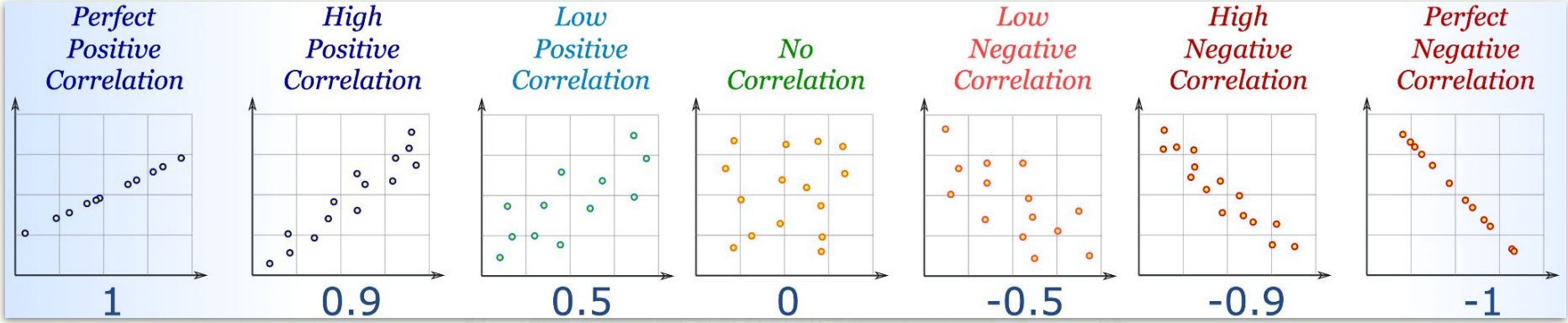
But First....Three Topics From Last Time

- impact of mean imputation on correlations
- stochastic regression
- diagnosing missingness
- visualization

But First....Three Topics From Last Time

- impact of mean imputation on correlations
- stochastic regression
- diagnosing missingness
- visualization

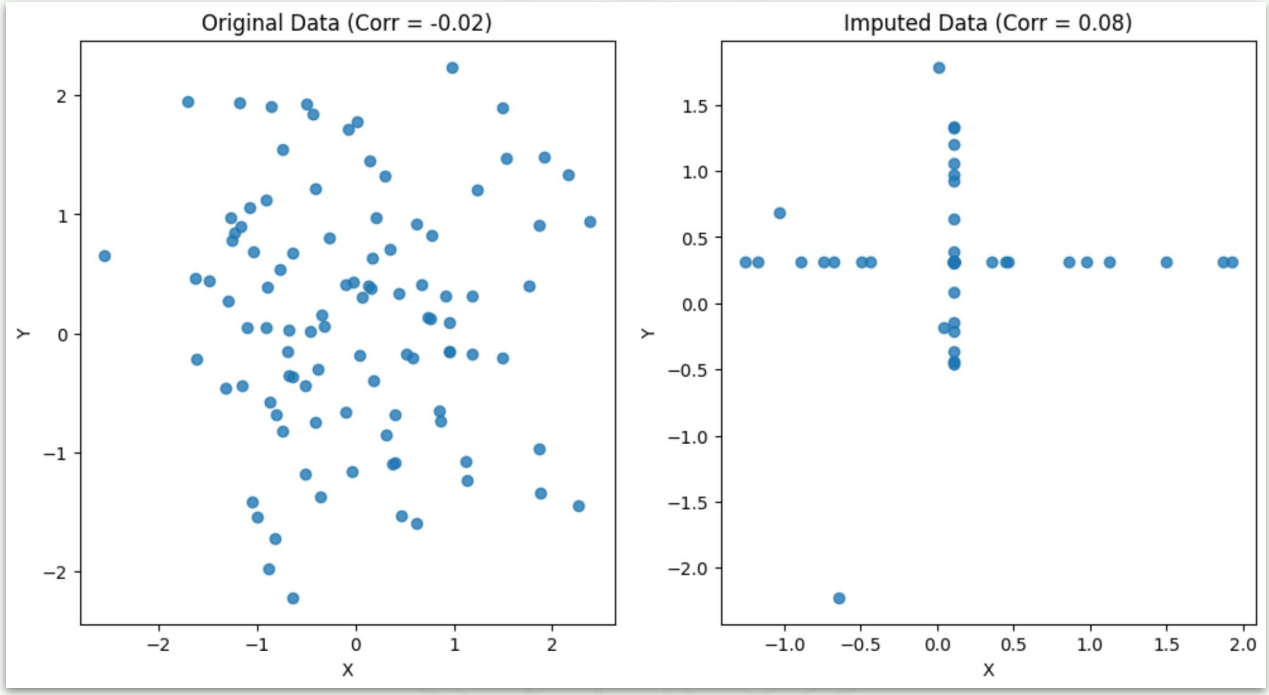
Loss of Correlations from Mean Imputation



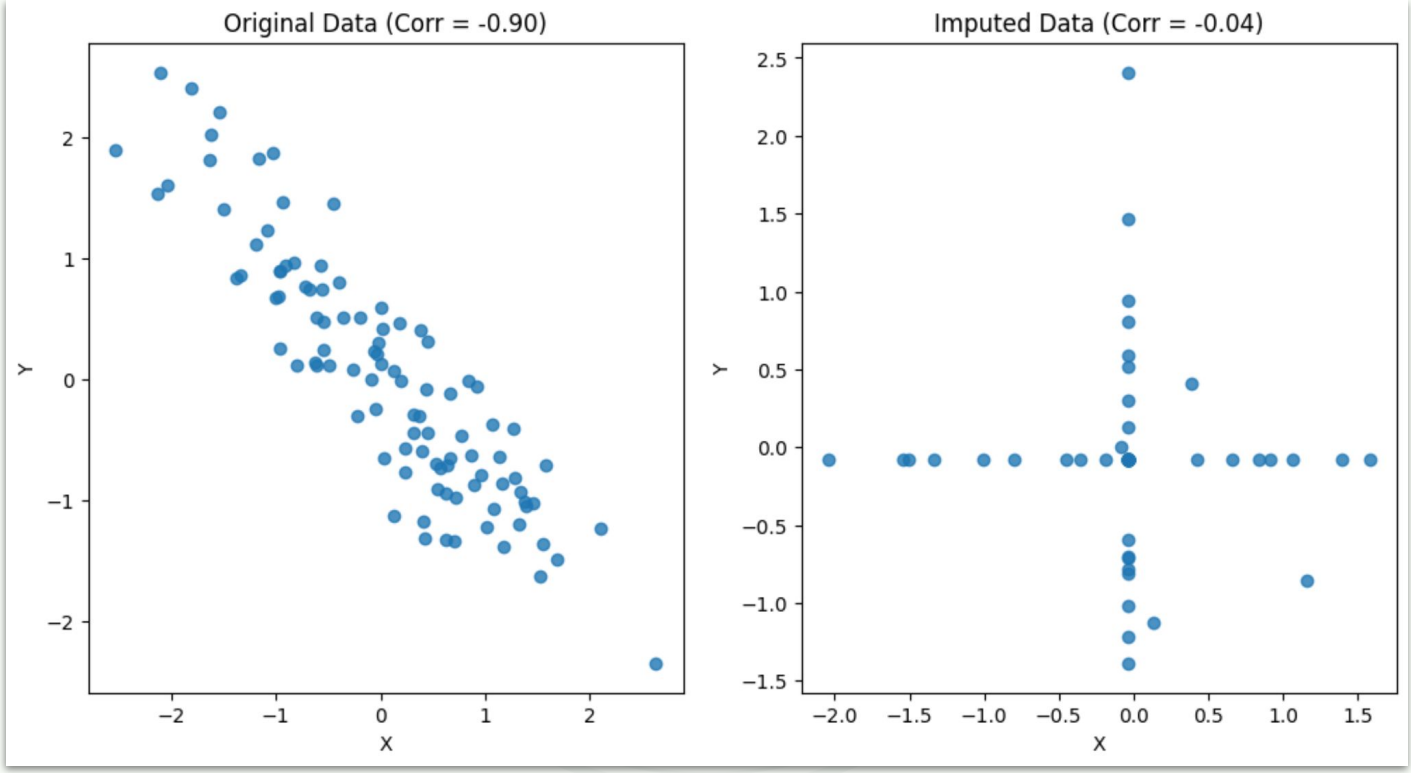
$$\rho_{X,Y} = \frac{\mathbb{E}[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y}$$

Loss of Correlations from Mean Imputation: Examples

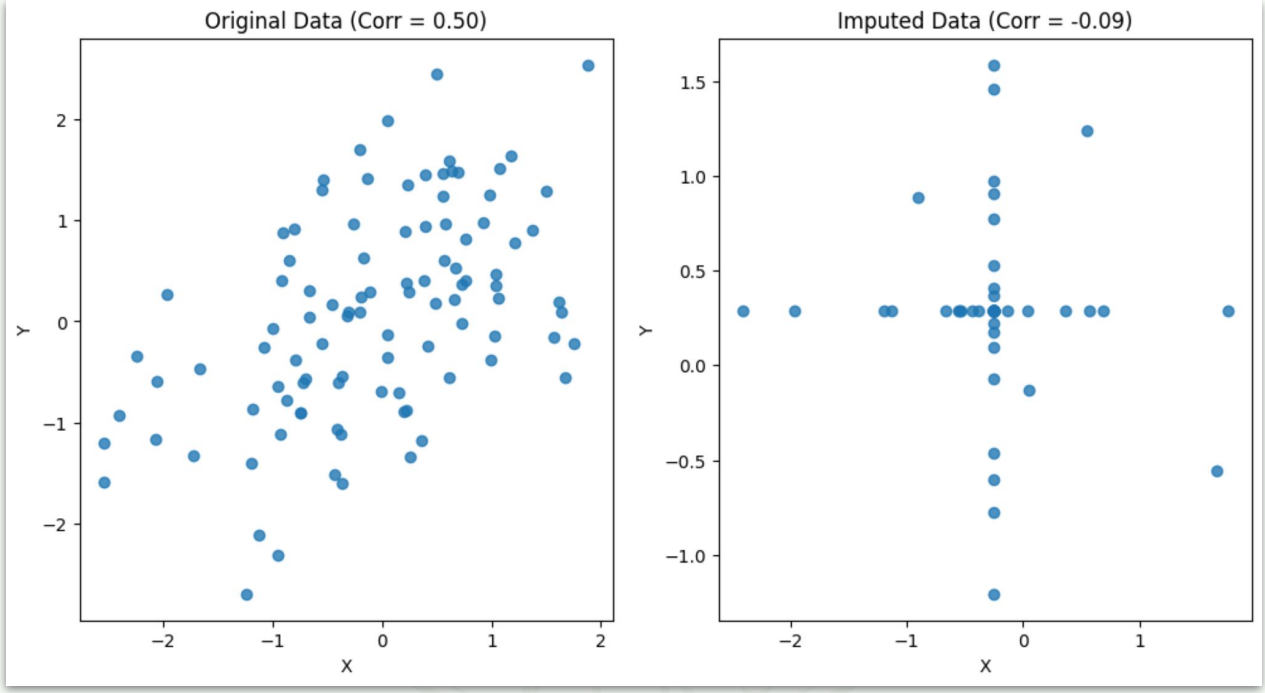
missing rate = 0.8



Loss of Correlations from Mean Imputation: Examples



Loss of Correlations from Mean Imputation: Examples



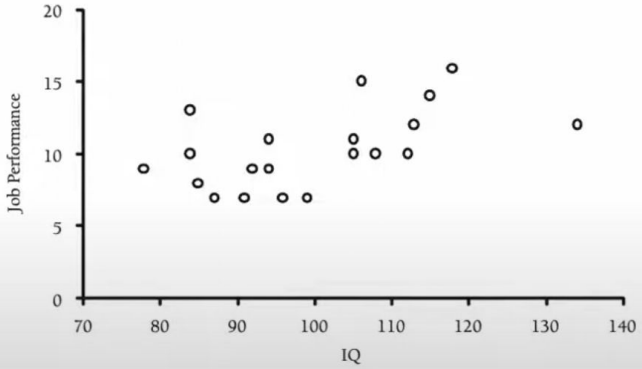
But First....Three Topics From Last Time

- impact of mean imputation on correlations
- stochastic regression
- diagnosing missingness
- visualization

Stochastic Regression: What is the Goal?

Here is the raw data:

Example dataset



2.1. Complete-data scatterplot of the IQ and job performance scores from Table 2.1.

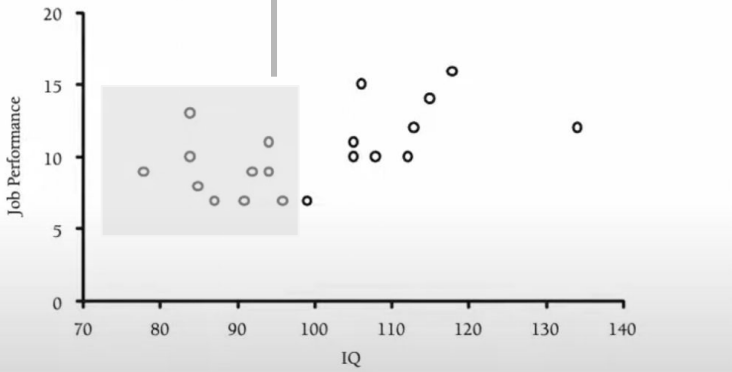
TABLE 2.1. Employee Selection Data Set

Complete data		Missing data
IQ	Job performance	Job Performance
78	9	—
84	13	—
84	10	—
85	8	—
87	7	—
91	7	—
92	9	—
94	9	—
94	11	—
96	7	—
99	7	7
105	10	10
105	11	11
106	15	15
108	10	10
112	10	10
113	12	12
115	14	14
118	16	16
134	12	12

Stochastic Regression: What is the Goal?

Here is the synthetic data:

Example dataset

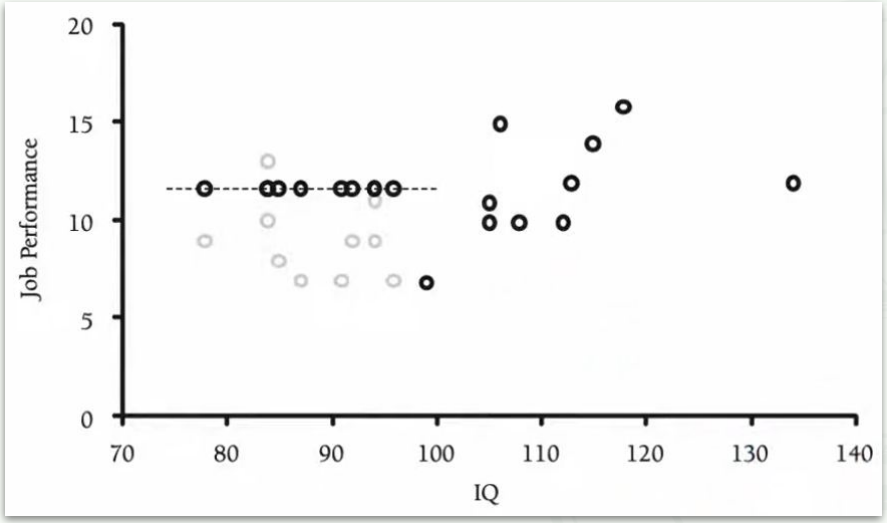


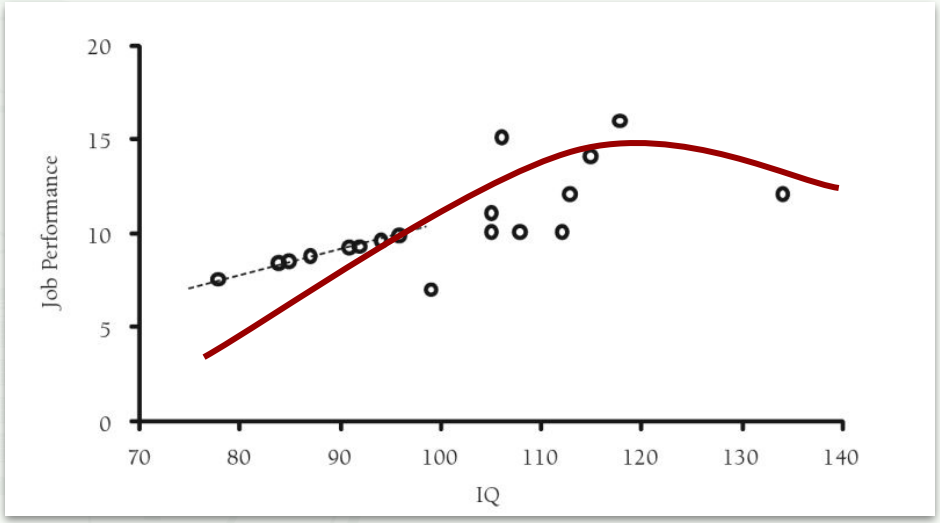
2.1. Complete-data scatterplot of the IQ and job performance scores from Table 2.1.

TABLE 2.1. Employee Selection Data Set

Complete data		Missing data
IQ	Job performance	Job Performance
78	9	—
84	13	—
84	10	—
85	8	—
87	7	—
91	7	—
92	9	—
94	9	—
94	11	—
96	7	—
99	7	7
105	10	10
105	11	11
106	15	15
108	10	10
112	10	10
113	12	12
115	14	14
118	16	16
134	12	12

Fit Data and Extrapolate



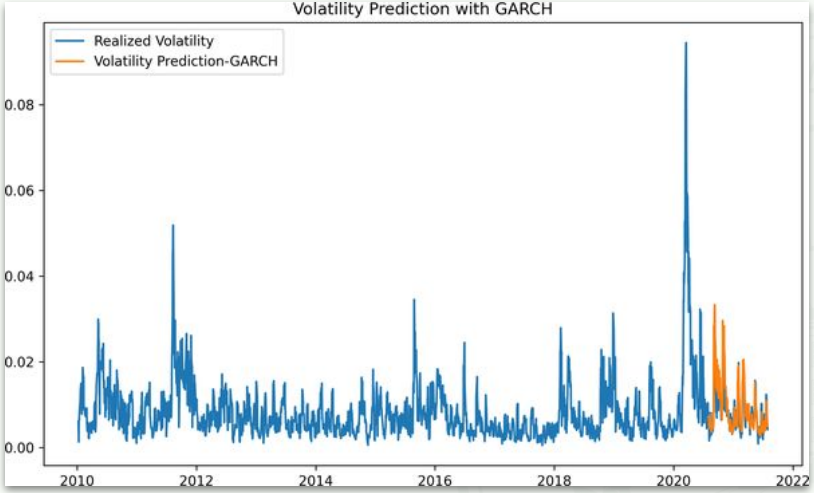


Point #1: mean imputation should “never” be done

Point #2: fitting is better, and perhaps good enough?

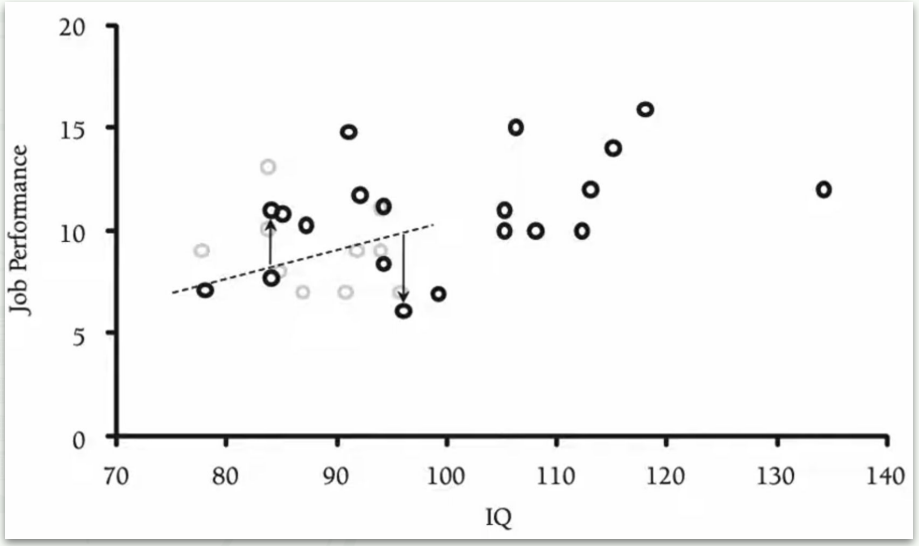
Point #3: don’t need to use a line (first-order polynomial)

Variance/Volatility



Very often we want to extrapolate the "volatility" in our data.

Sometimes, we only have "volatility".



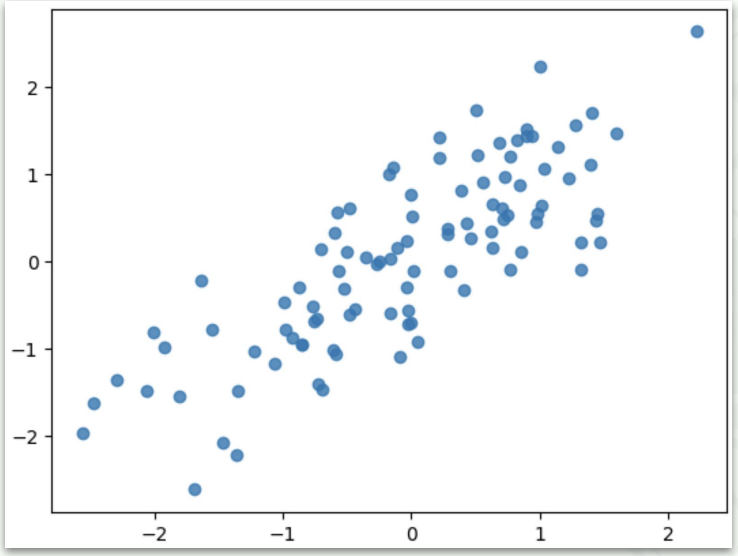
Point #4: when we impute, we want to preserve the mean, trend and variance

multiple imputation can change the conclusion!

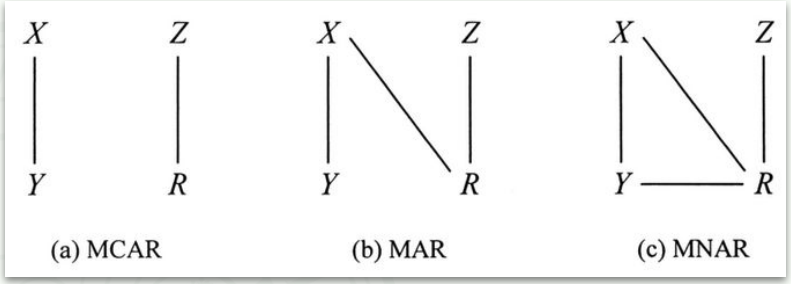
But First....Three Topics From Last Time

- impact of mean imputation on correlations
- stochastic regression
- diagnosing missingness
- visualization

More Details on Missingness



original




```
19 # Introduce MCAR missingness
20 missing_rate = 0.2  # Proportion of missing values
21 df_mcar = df_original.copy()
22 for col in df_mcar.columns:
23     missing_indices = random.sample(range(n), int(missing_rate * n))
24     df_mcar.loc[missing_indices, col] = np.nan
25
26 # Introduce MAR missingness
27 df_mar = df_original.copy()
28 missing_indices_mar = df_original[df_original['X'] > df_original['X'].quantile(0.75)].index
29 df_mar.loc[missing_indices_mar, 'Y'] = np.nan
30
31 # Introduce MNAR missingness
32 df_mnar = df_original.copy()
33 missing_indices_mnar = df_original[df_original['Y'] > 0].index
34 df_mnar.loc[missing_indices_mnar, 'Y'] = np.nan
```

add missingness to the data.

Missingness in Terms of Conditional Probabilities

MCAR:

$$P(\text{missing}|\text{complete}) = P(\text{missing})$$

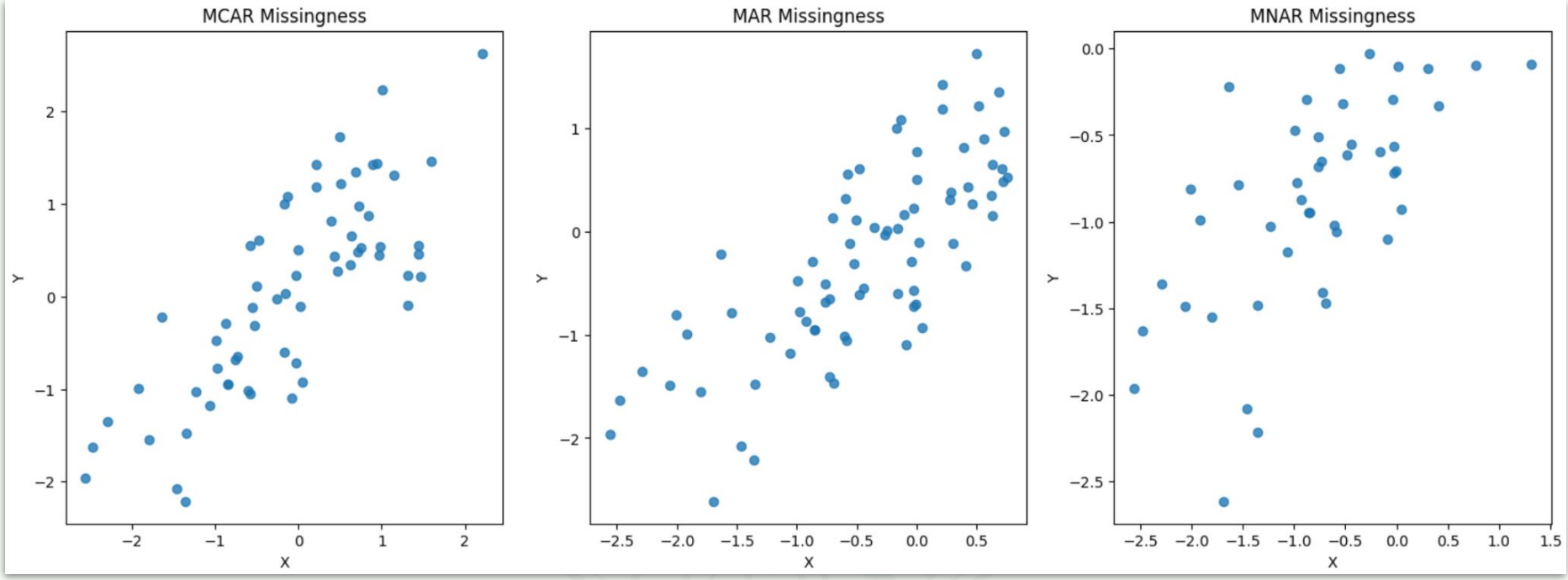
MAR:

$$P(\text{missing}|\text{complete}) = P(\text{missing}|\text{observed})$$

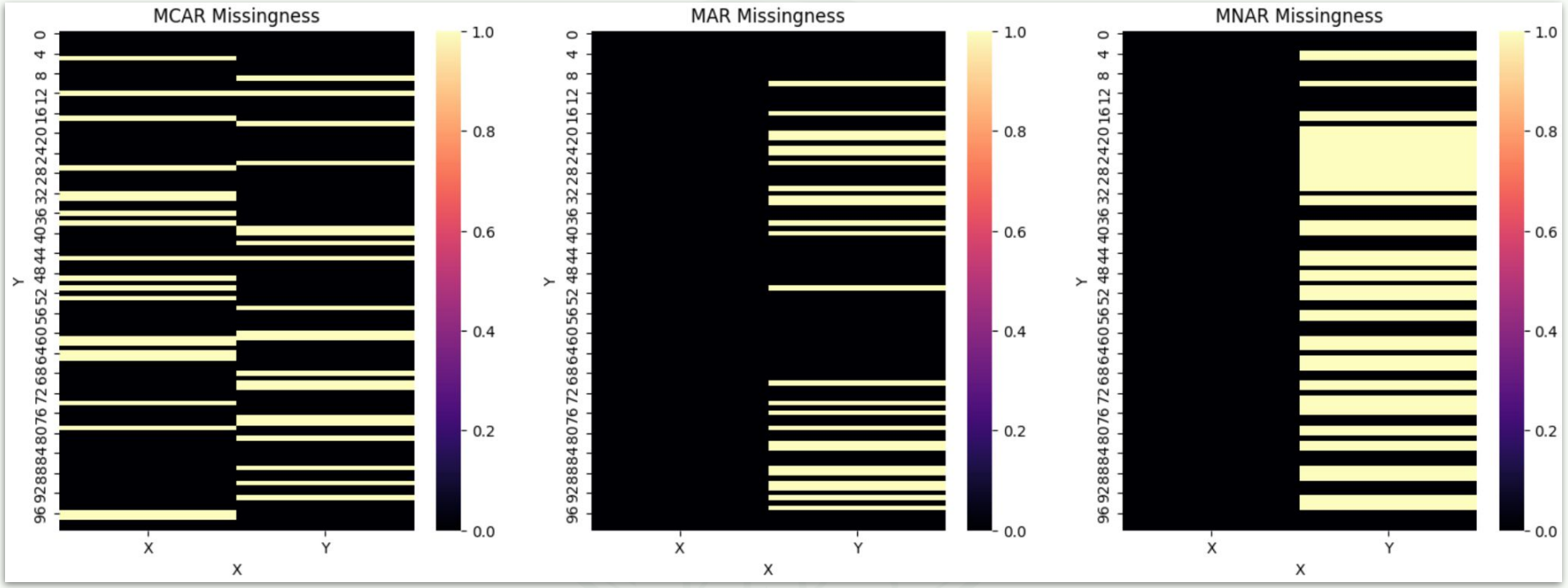
MNAR:

$$P(\text{missing}|\text{complete}) \neq P(\text{missing}|\text{observed})$$

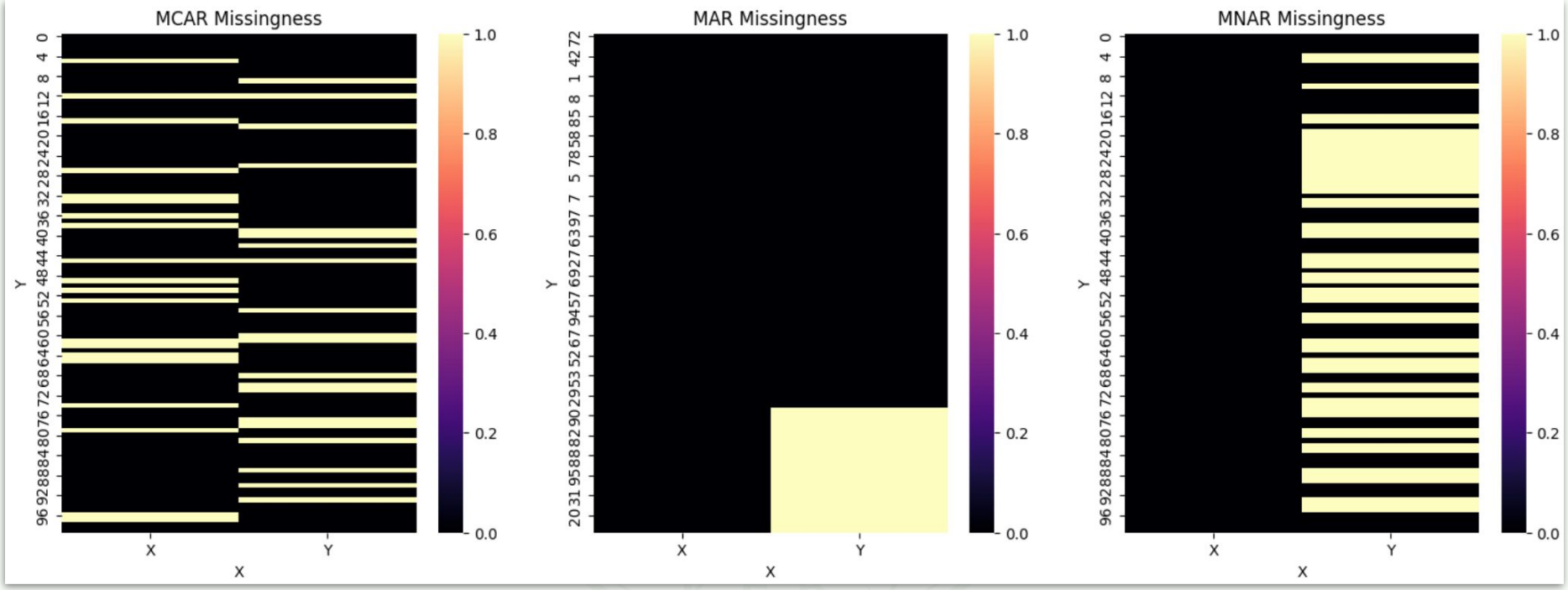
More Details on Missingness



Visualizing Synthetic Missingness



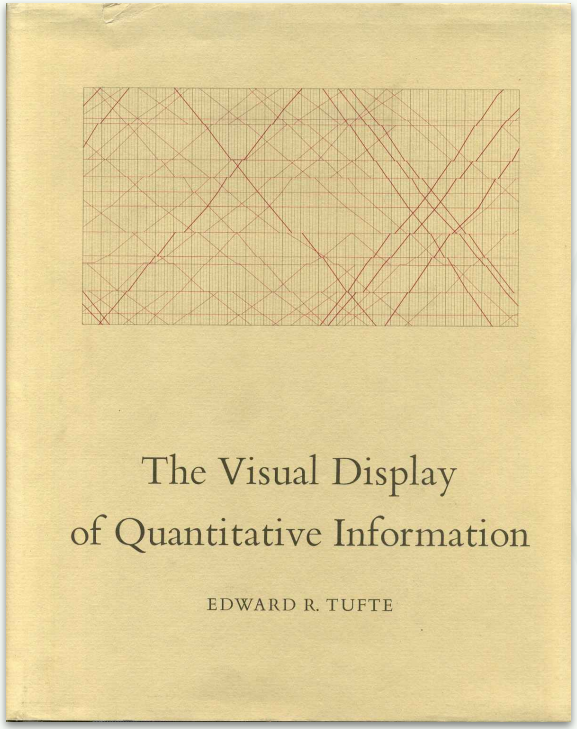
Sorted by X Value



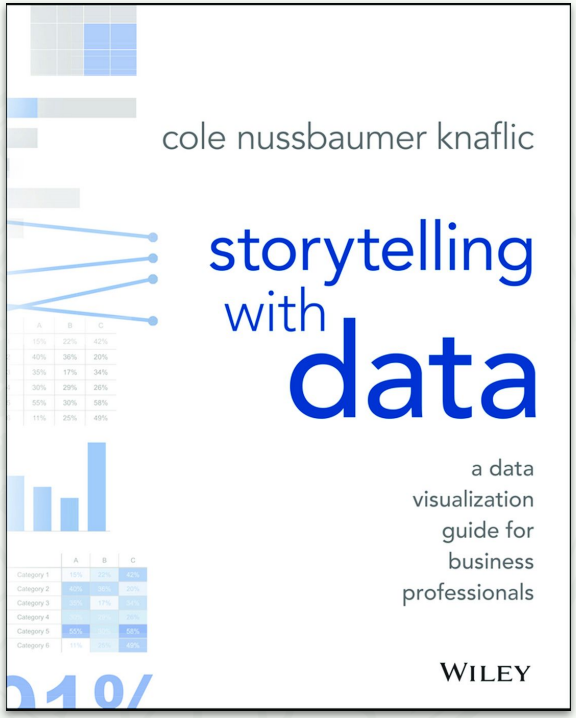
But First....Three Topics From Last Time

- impact of mean imputation on correlations
- stochastic regression
- diagnosing missingness
- visualization

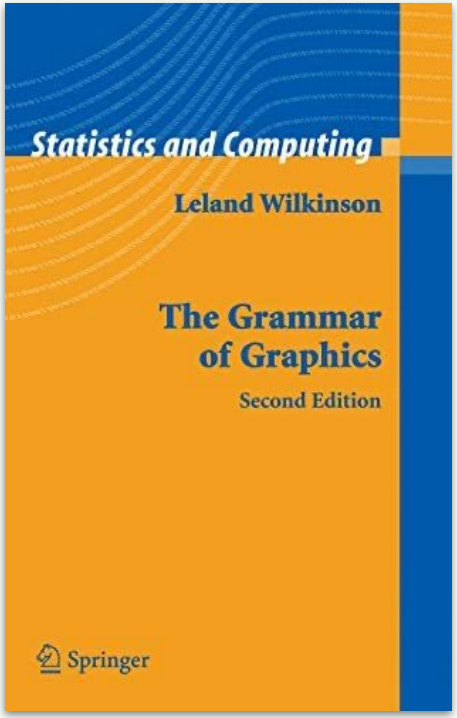
Three Classics You May Wish To Own



classic on good practice

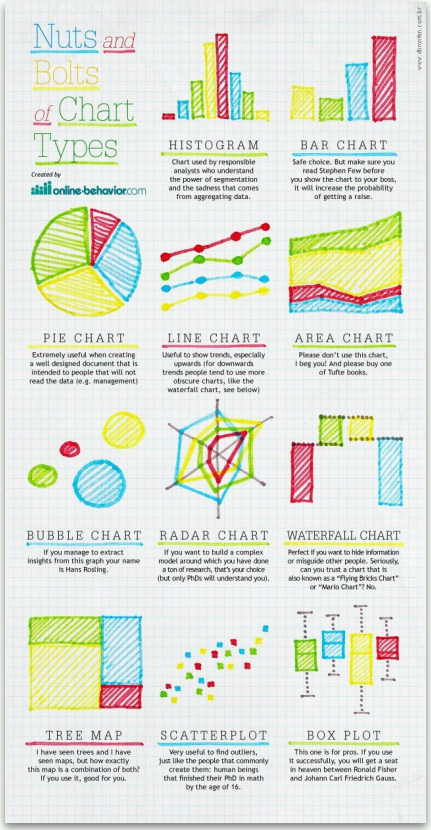


focuses on the message



technical approach to graphics

Plotting By Type

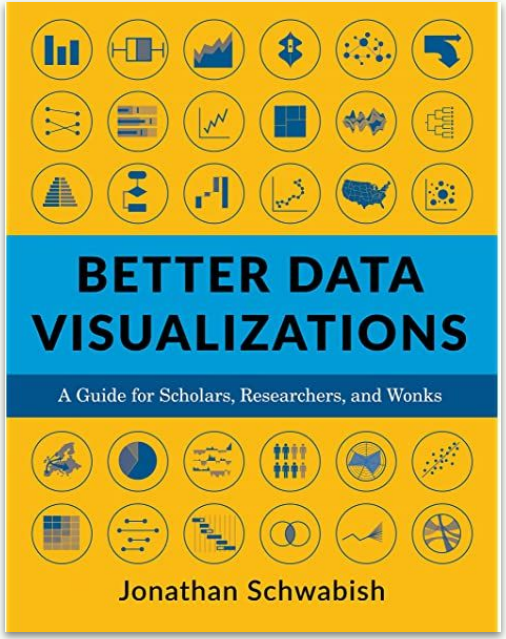


There are many standard types of plots in wide use.

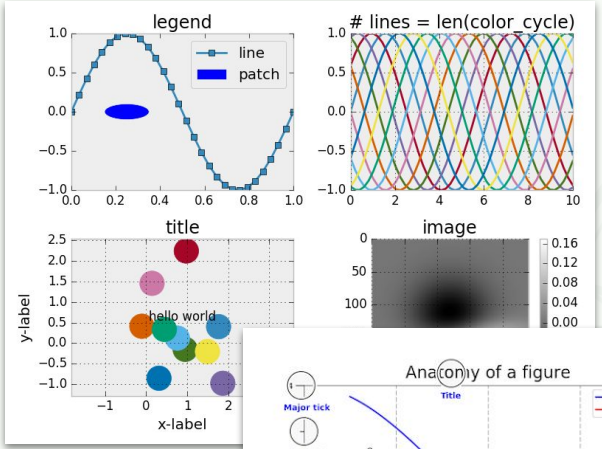
Most of these are built into matplotlib.

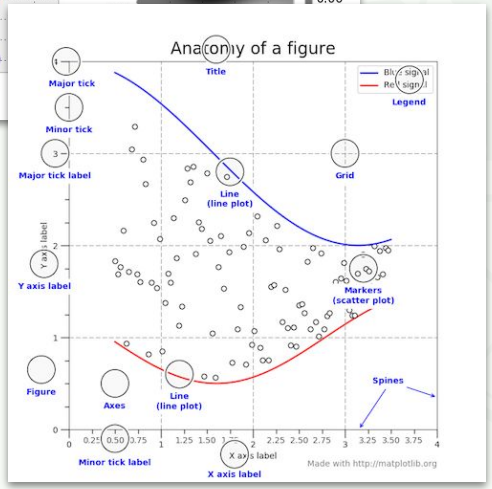
You can quickly make these plots with very few lines of code.

For most plotting tasks, these standard plots are all you need!

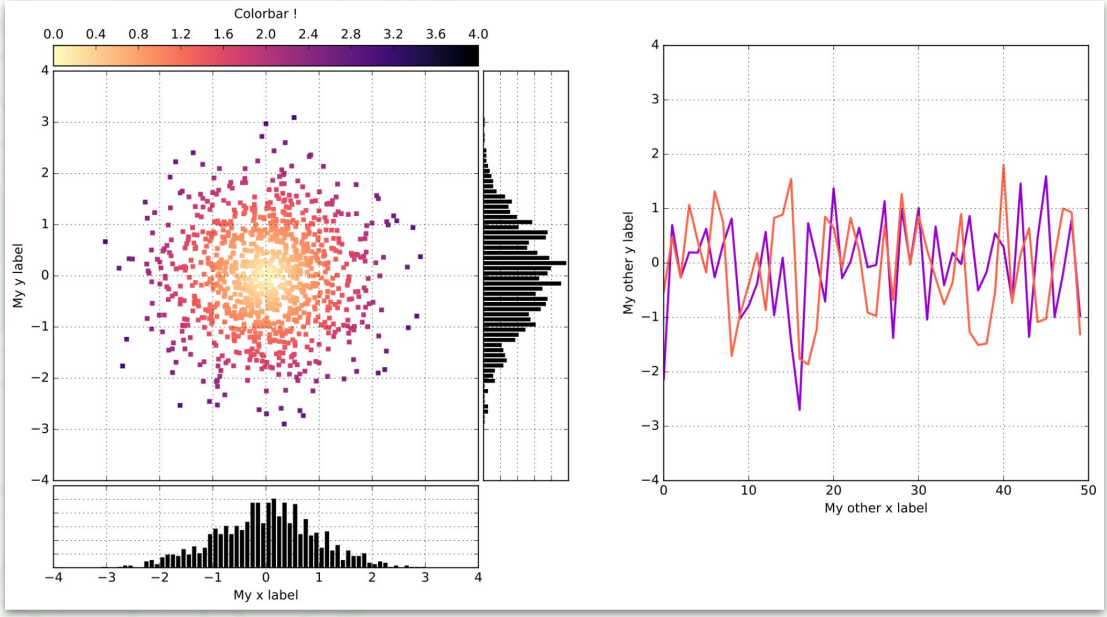


Custom Plotting





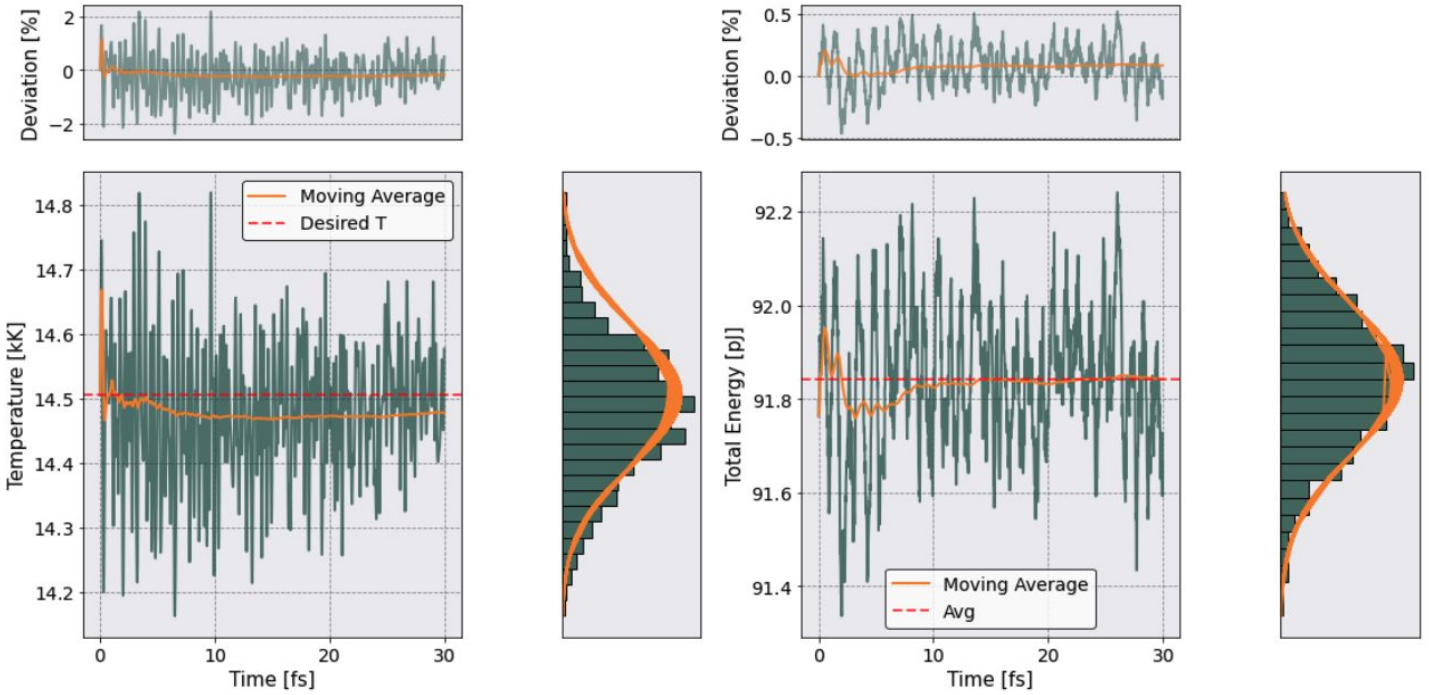
You want complete control over what your plot looks like.



Data From A Simulation

Job ID: yocp
Phase: Production
No. of species = 1
Species 1 : H
  No. of particles = 10000
  Temperature = 14.51 [kK]

Total N = 10000
Thermostat: berendsen
  Berendsen rate = 1.00
  Equilibration cycles = 167
Potential: yukawa
  Coupling Const = 1.01e+02
  Tot Force Error = 6.44e-06
Integrator: verlet
  Δt = 2.00 [as]
Completed steps = 15000
Total steps = 15000
100.00 % Completed
Production time = 30.00 [fs]
Production cycles = 502



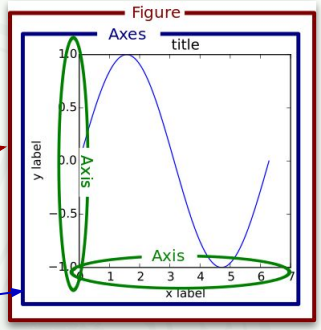
Plot Structure

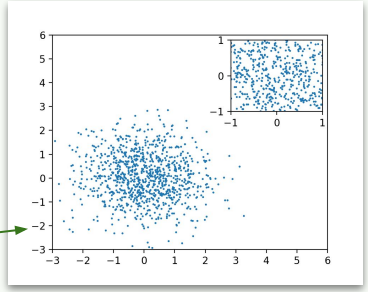
Plots are typically organized around three concepts:

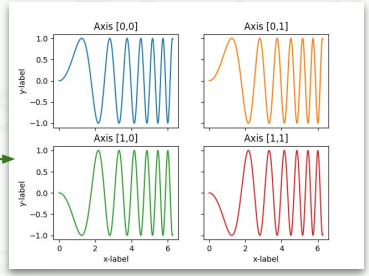
Figure: this is the frame. You can have many of these, and they form separate entities.

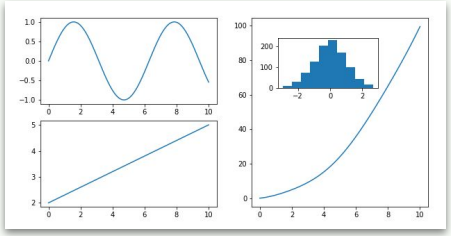
Axis: this is the plot itself, with its axes. It has no position - it is free floating in the frame somewhere. You can have many of these within a single figure (frame).

Subplot: these are axes in some sort of predetermined arrangement.









example of using both

Trends in Visualization For Data Science: GoG

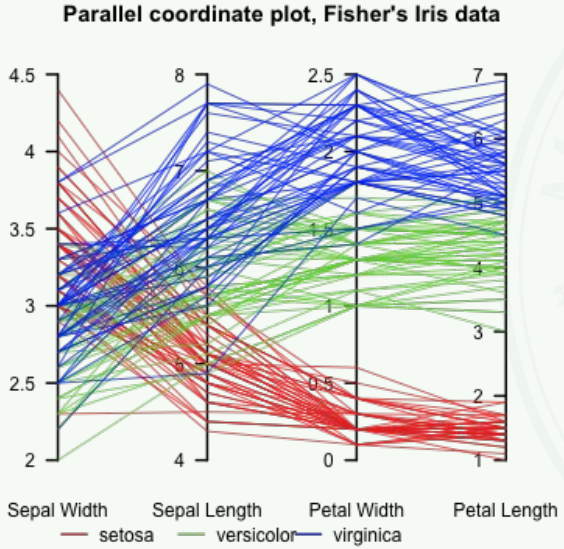
```python
import altair as alt
import pandas as pd

data = pd.DataFrame({'x': ['A', 'B', 'C', 'D', 'E'],
                     'y': [5, 3, 6, 7, 2]})
alt.Chart(data).mark_bar().encode(
    x='x',
    y='y',
)
```

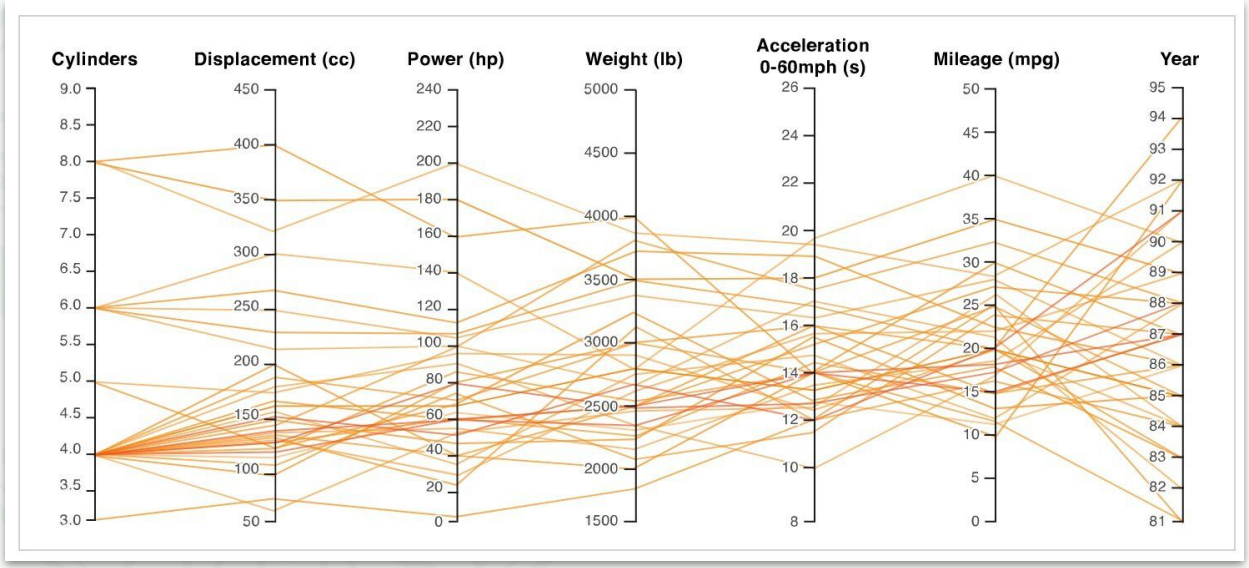
The seaborn.objects interface

The `seaborn.objects` namespace was introduced in version 0.12 as a completely new interface for making seaborn plots. It offers a more consistent and flexible API, comprising a collection of composable classes for transforming and plotting data. In contrast to the existing `seaborn` functions, the new interface aims to support end-to-end plot specification and customization without dropping down to matplotlib (although it will remain possible to do so if necessary).

Trends in Visualization For Data Science: Types

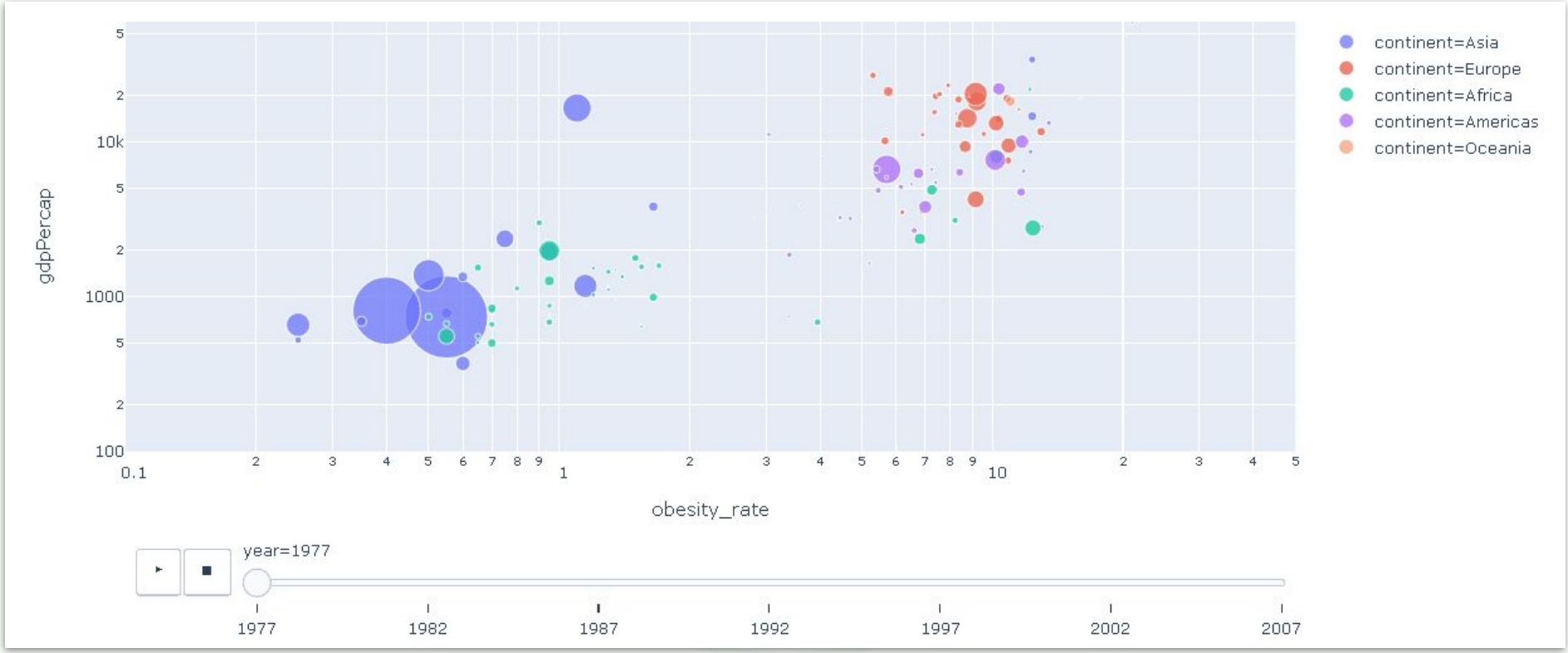
Important one: **parallel plot**.





Useful for high-dimensional data.

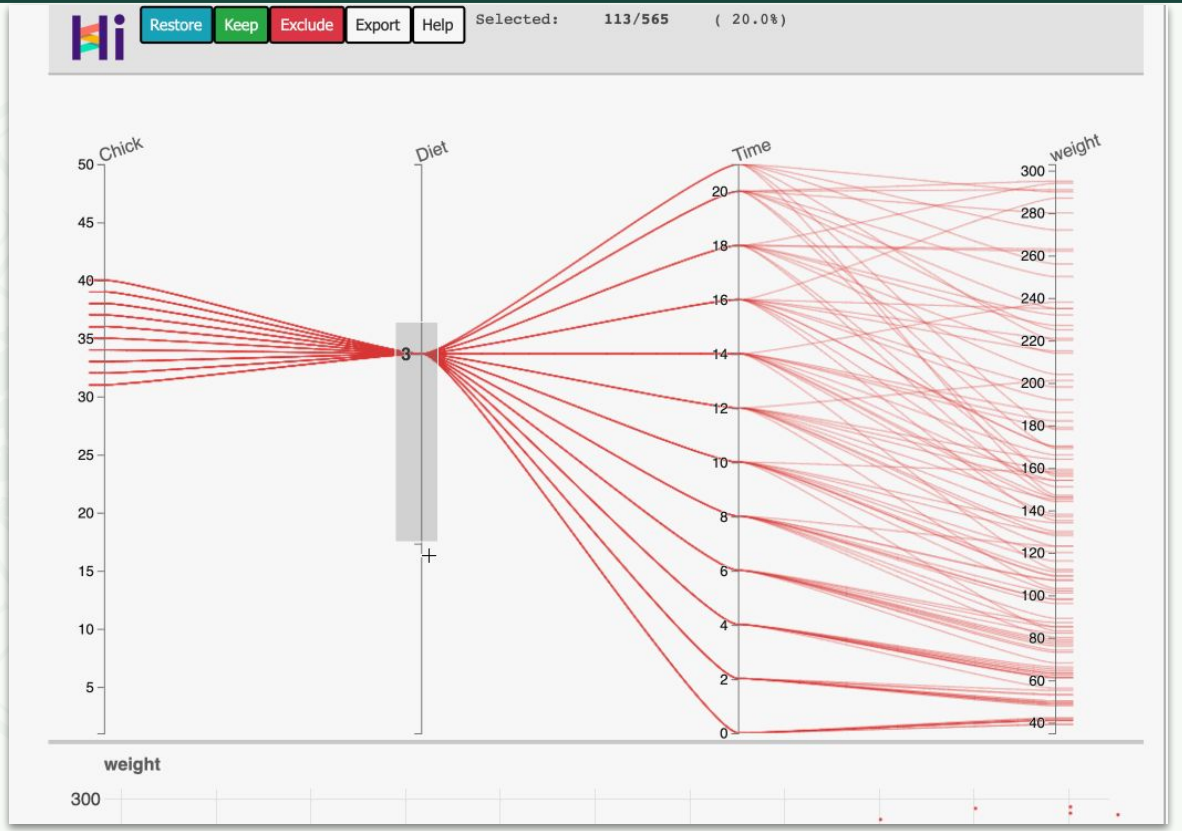
Trends in Visualization For Data Science: Interactivity



Trends in Visualization For Data Science: Web Interactivity

Hi-Plot: Interactive Parallel Plot

HiPlot



Communicating Your Message

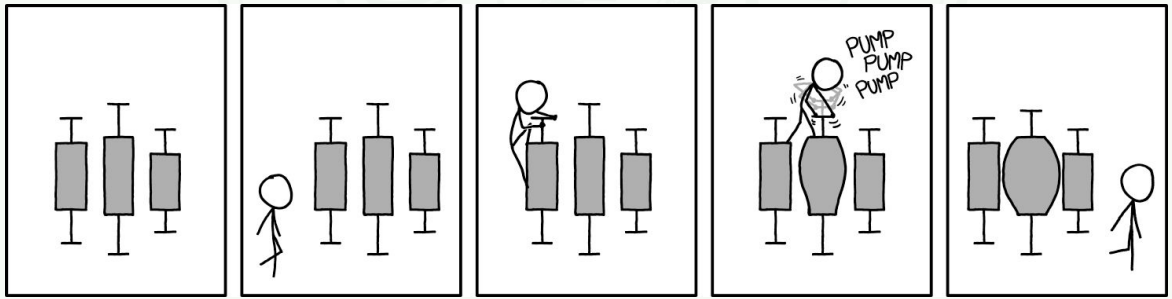
You now know the mechanics of making visualizations, including very rich data science charts.

But, are you telling the right story?

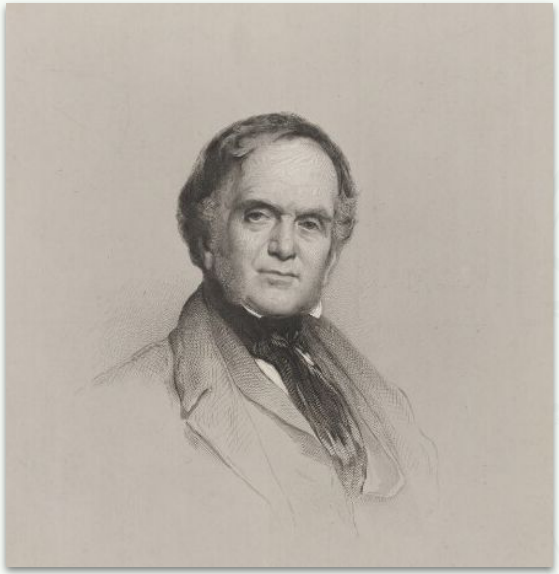
Are you communicating and connecting with your audience?

In this next segment, we will examine how to efficiently tell the most accurate and compelling story.

Parts of what is to come are somewhat subjective - use the ideas as guides for good practice, but break the rules when it works for you.

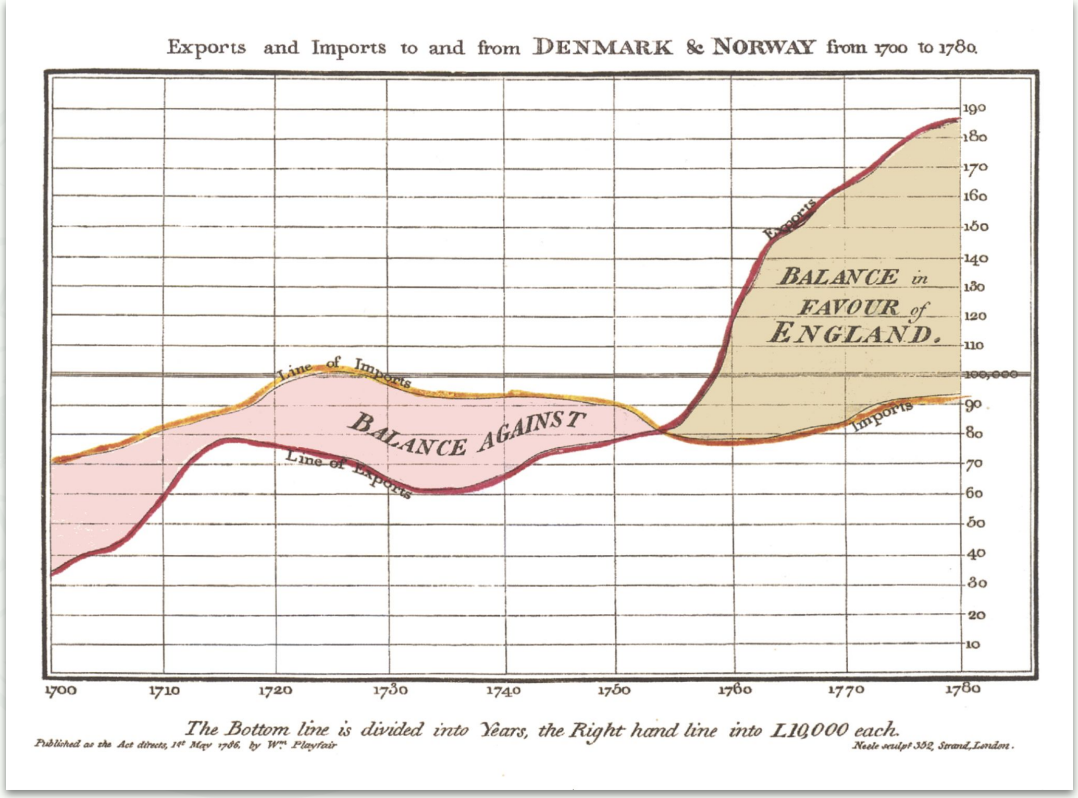


Visualization is An Old Science



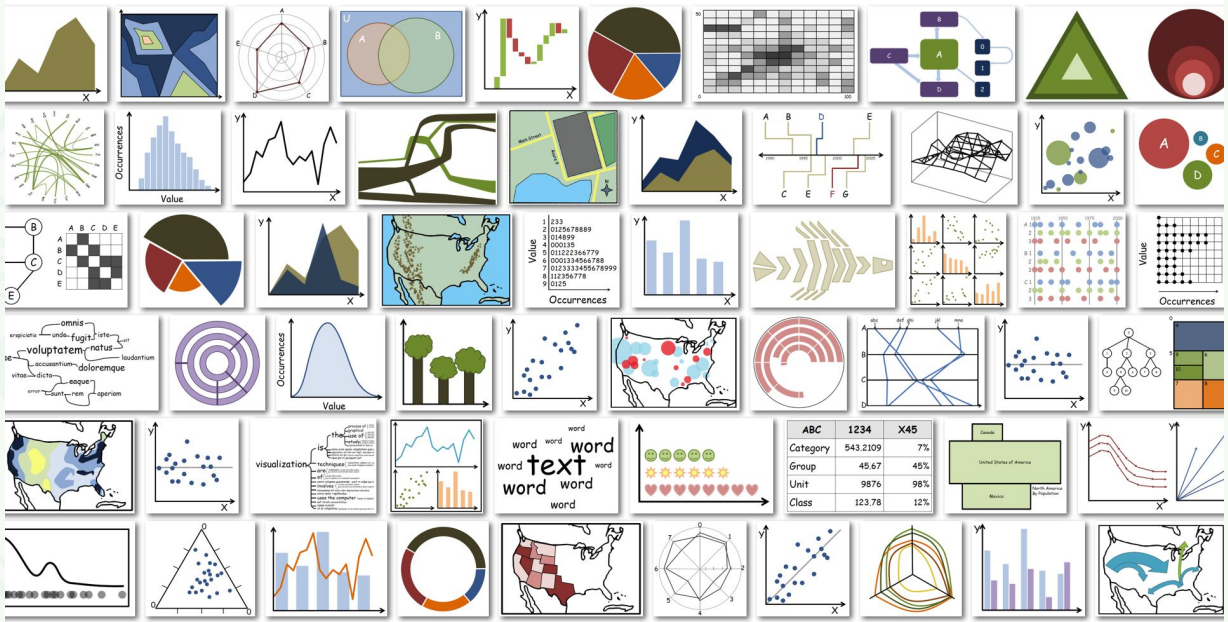
William Playfair
1759 - 1823

Considered the “father of visualization”.



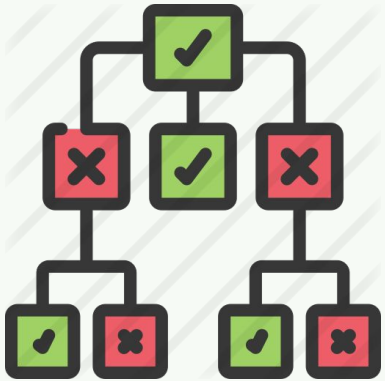
Visualization is Communication

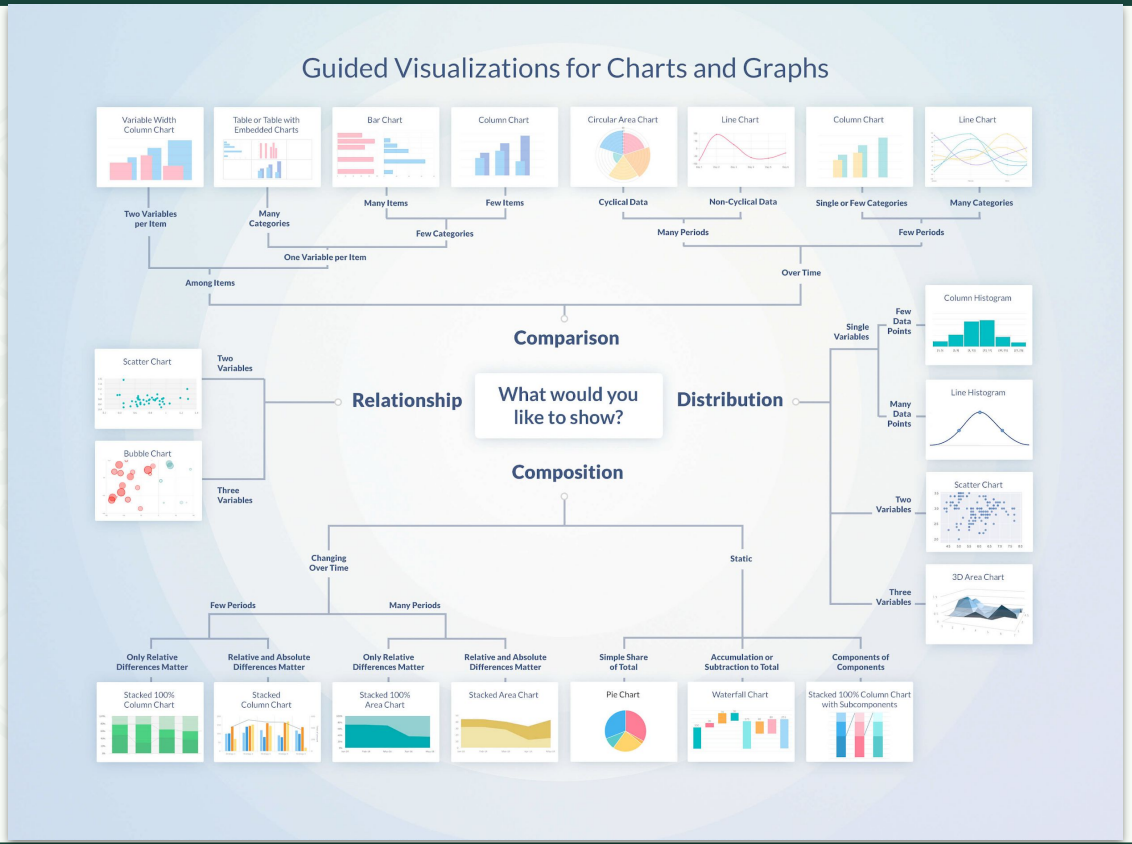
Choose the best visualization format to communicate your message.



Visualizations come in various types.

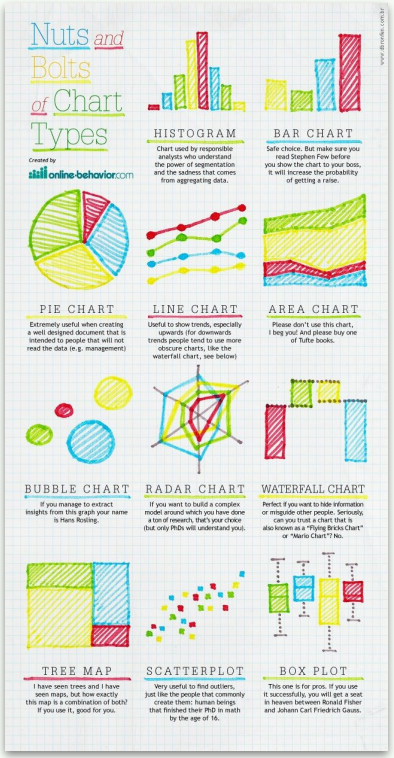
Think about your goal in terms of a decision tree of choices from the most basic to the detailed.





Use Standard Types, Or Invent Your Own

plotting by standard type



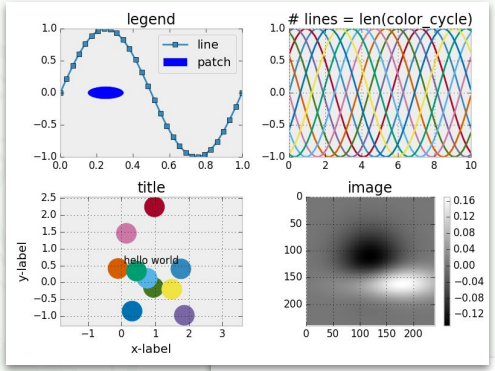
There are many standard types of plots in wide use.

Most of these are built into matplotlib.

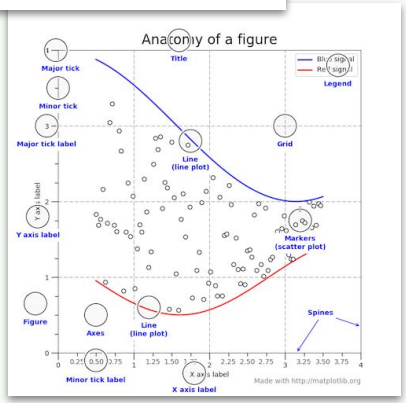
You can quickly make these plots with very few lines of code.

For most plotting tasks, these standard plots are all you need!

custom plotting

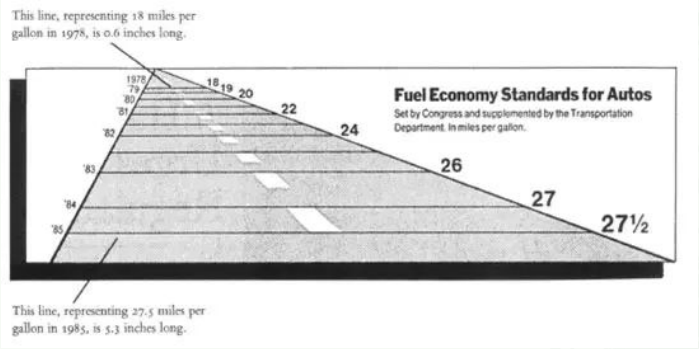


In some cases you want to take complete control over what your plot looks like.



Three* Rules From Edward Tufte

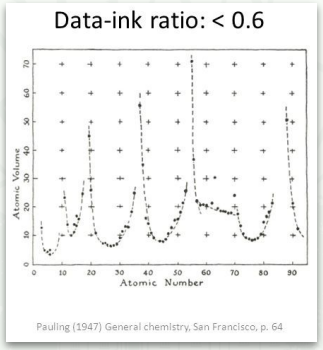
graphical integrity

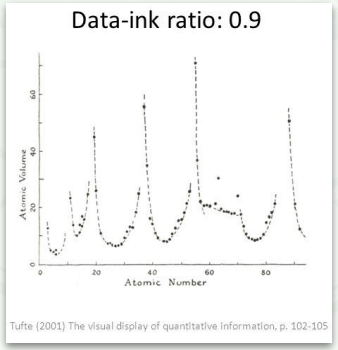


This chart has what Tufte calls a large "Lie Factor": the scales in the plot mislead the viewer because they are not scaled to the data.

1. Make your plots honest.
2. Remove ink that adds no value.
3. Don't use gimmicks that make your chart difficult to understand.

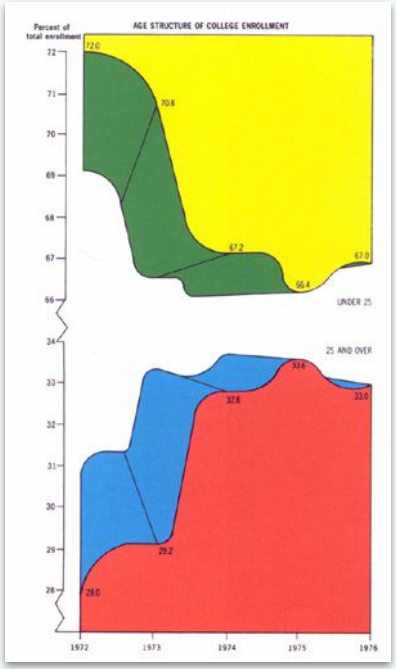
data ink





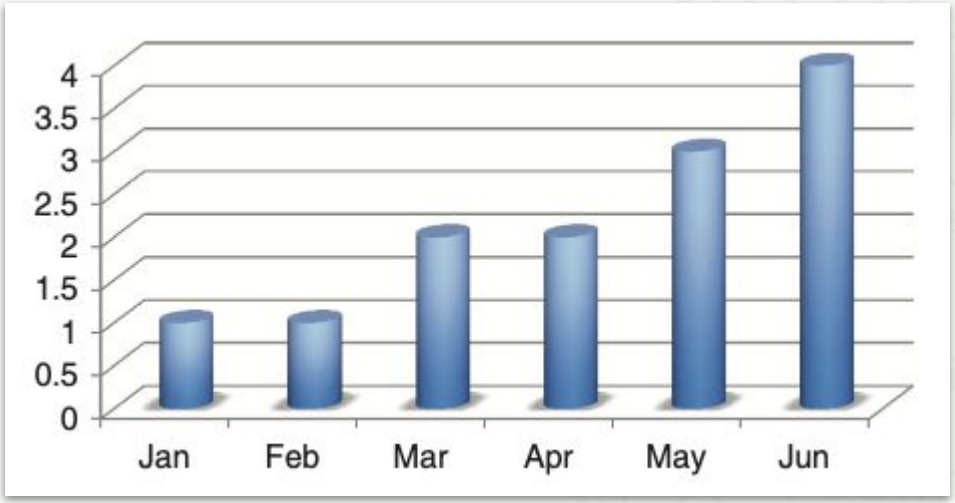
Is your extra ink making the plot less easy to read?

chartjunk



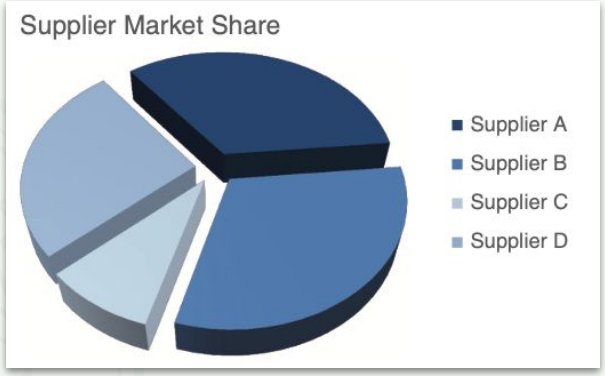
Can you decipher what this is supposed to be showing?

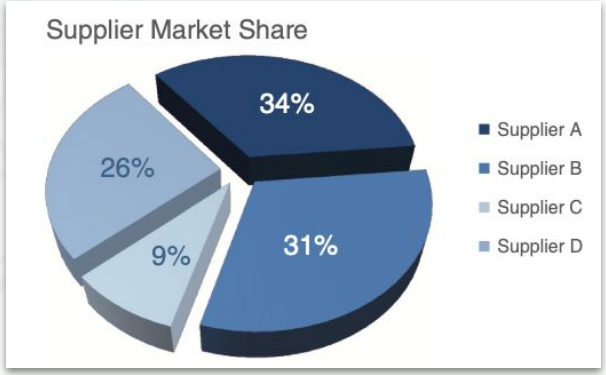
Fancy Plots Can Mislead



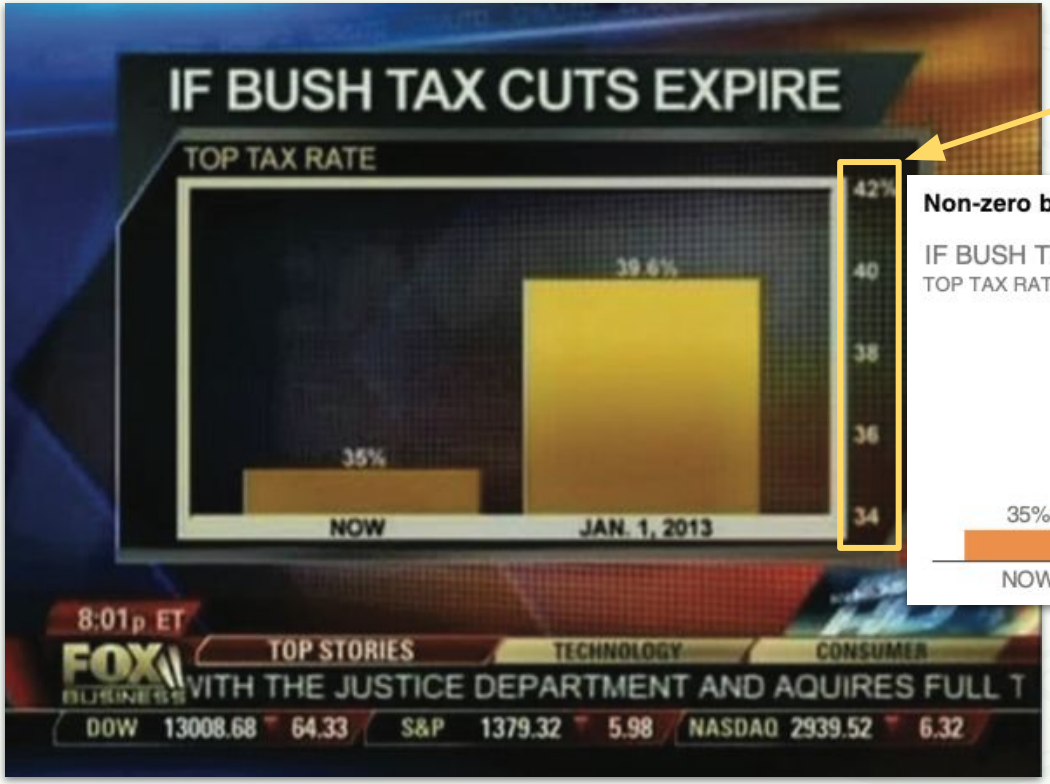
How precisely can you write down the actual values?

For example: is the value for Jan less than or more than 1?

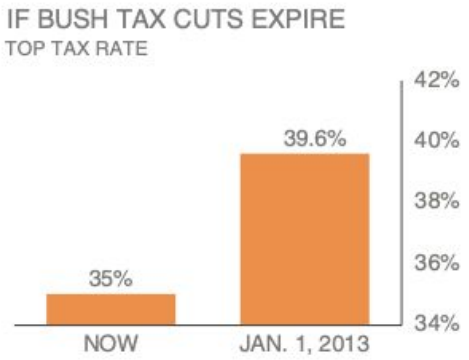




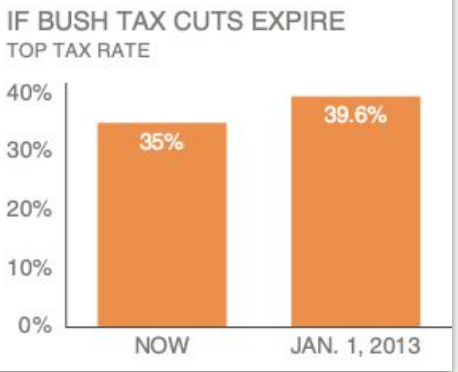
In The News: Example of Misleading Plot



Non-zero baseline: as originally graphed



Zero baseline: as it should be graphed



Overuse Of Color Can Mislead

This creates a “puzzle” for the viewer, which is to keep in their mind the ordering of colors in the rainbow as they look over the chart.

Worse, the viewer has to “turn off” social biases, such as “red = bad”, or bright colors are more important (e.g., yellow).

Country Level Sales Rank Top 5 Drugs

Rainbow distribution in color indicates sales rank in given country from #1 (red) to #10 or higher (dark purple)

Country	A	B	C	D	E
AUS	1	2	3	6	7
BRA	1	3	4	5	6
CAN	2	3	6	12	8
CHI	1	2	8	4	7
FRA	3	2	4	8	10
GER	3	1	6	5	4
IND	4	1	8	10	5
ITA	2	4	10	9	8
MEX	1	5	4	6	3
RUS	4	3	7	9	12
SPA	2	3	4	5	11
TUR	7	2	3	4	8
UK	1	2	3	6	7
US	1	2	4	3	5

Top 5 drugs: country-level sales rank

RANK	1	2	3	4	5+
COUNTRY | DRUG	A	B	C	D	E
Australia	1	2	3	6	7
Brazil	1	3	4	5	6
Canada	2	3	6	12	8
China	1	2	8	4	7
France	3	2	4	8	10
Germany	3	1	6	5	4
India	4	1	8	10	5
Italy	2	4	10	9	8
Mexico	1	5	4	6	3
Russia	4	3	7	9	12
Spain	2	3	4	5	11
Turkey	7	2	3	4	8
United Kingdom	1	2	3	6	7
United States	1	2	4	3	5

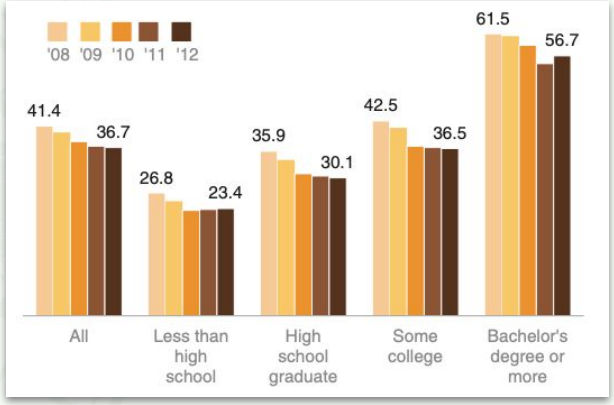
remake

Walkthrough of Plot Design

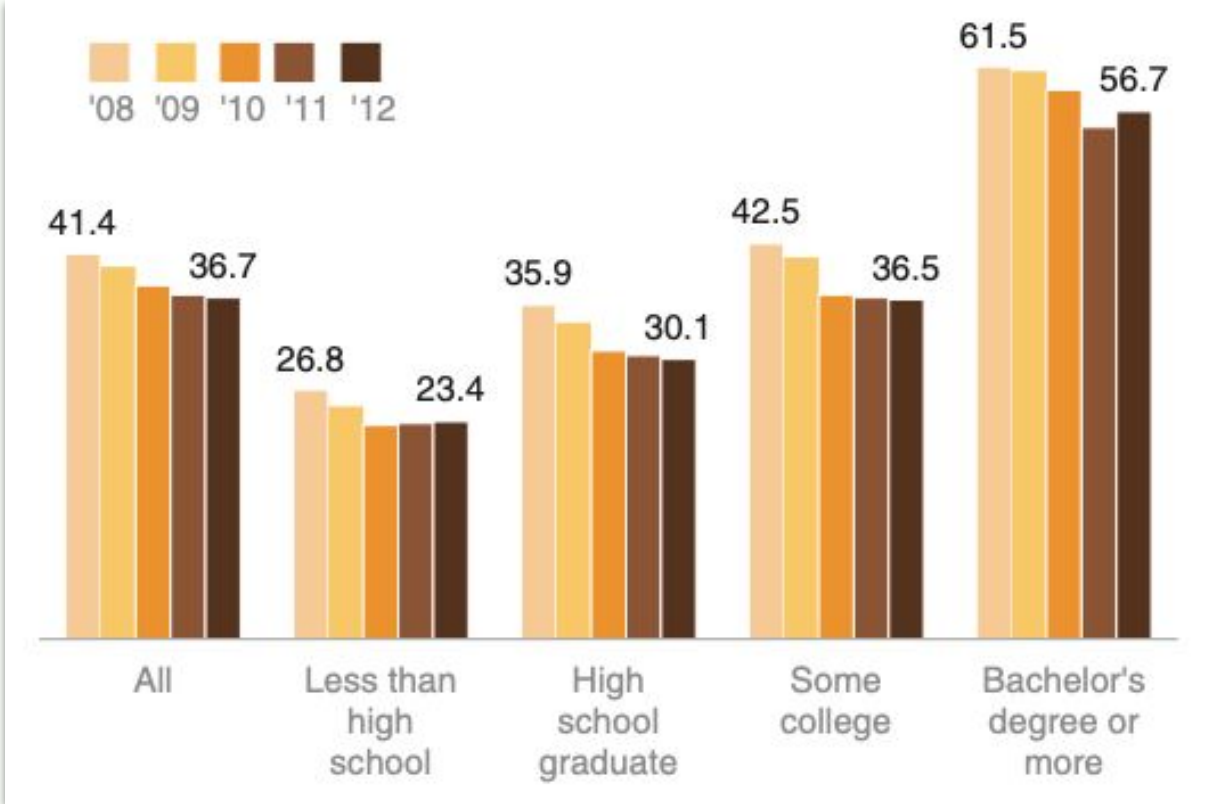
Let's take some reasonable plots and see if we can apply some of these ideas to improve them.

The most important goal is to be able to communicate what our point is, not just throw data at the viewer to try to figure out.

Tell your story.

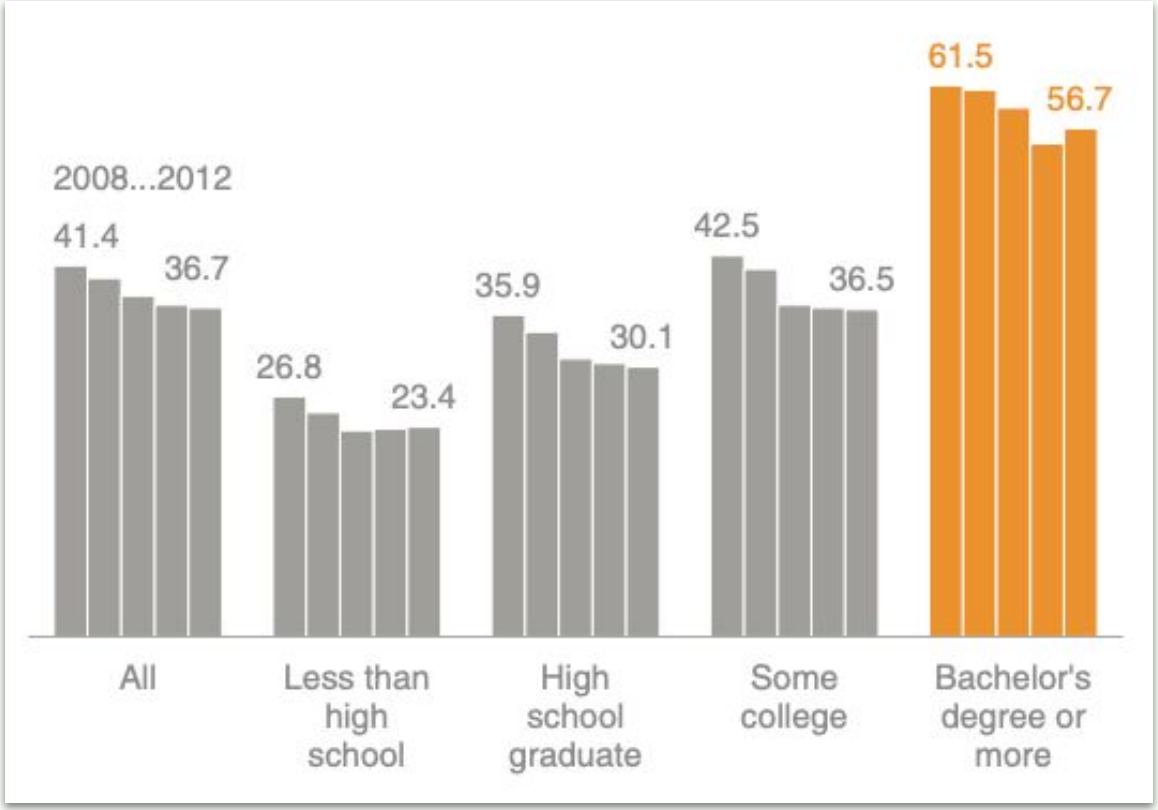


First Version - nice!

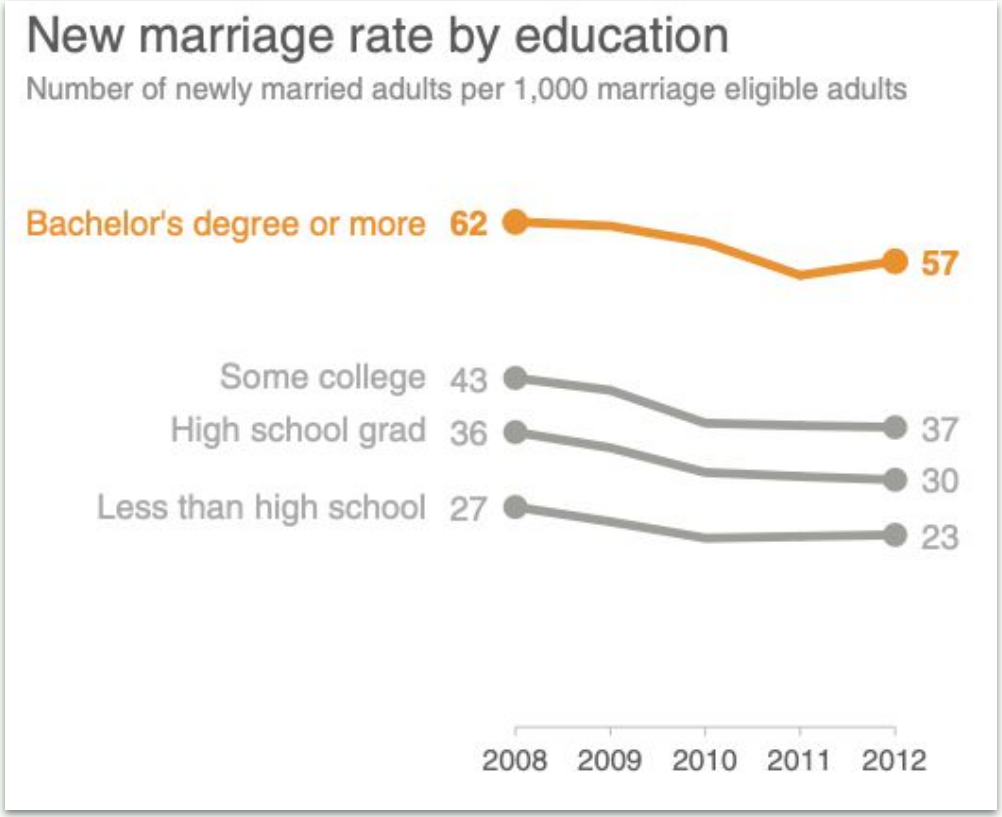


Redirect Attention To Your Story

What you might do at this step greatly depends on the story you wish to tell.



Think Outside The Box

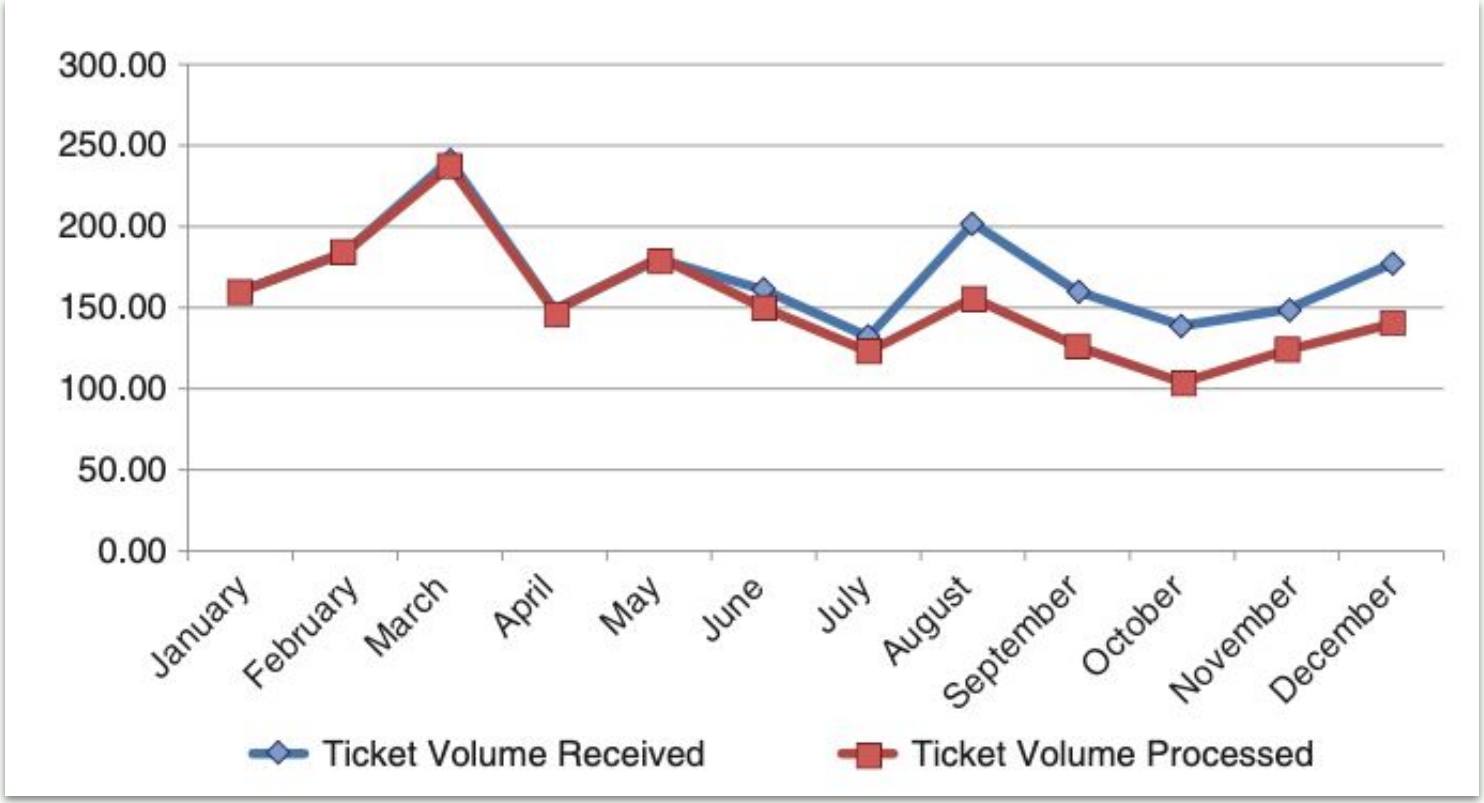


Is this an easier way to make your point?

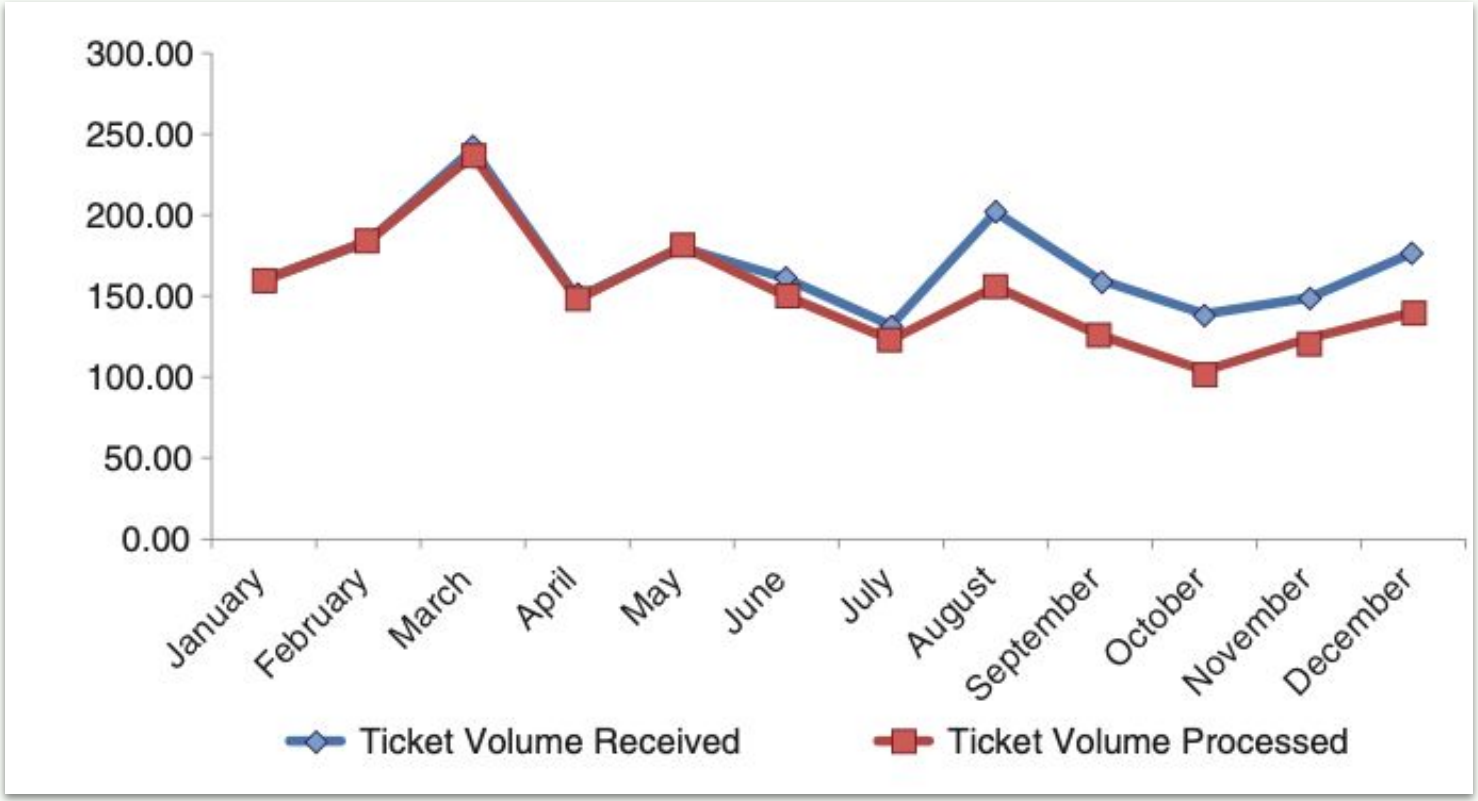
Walkthrough of Plot Design

Let's do another one in **nine** steps.

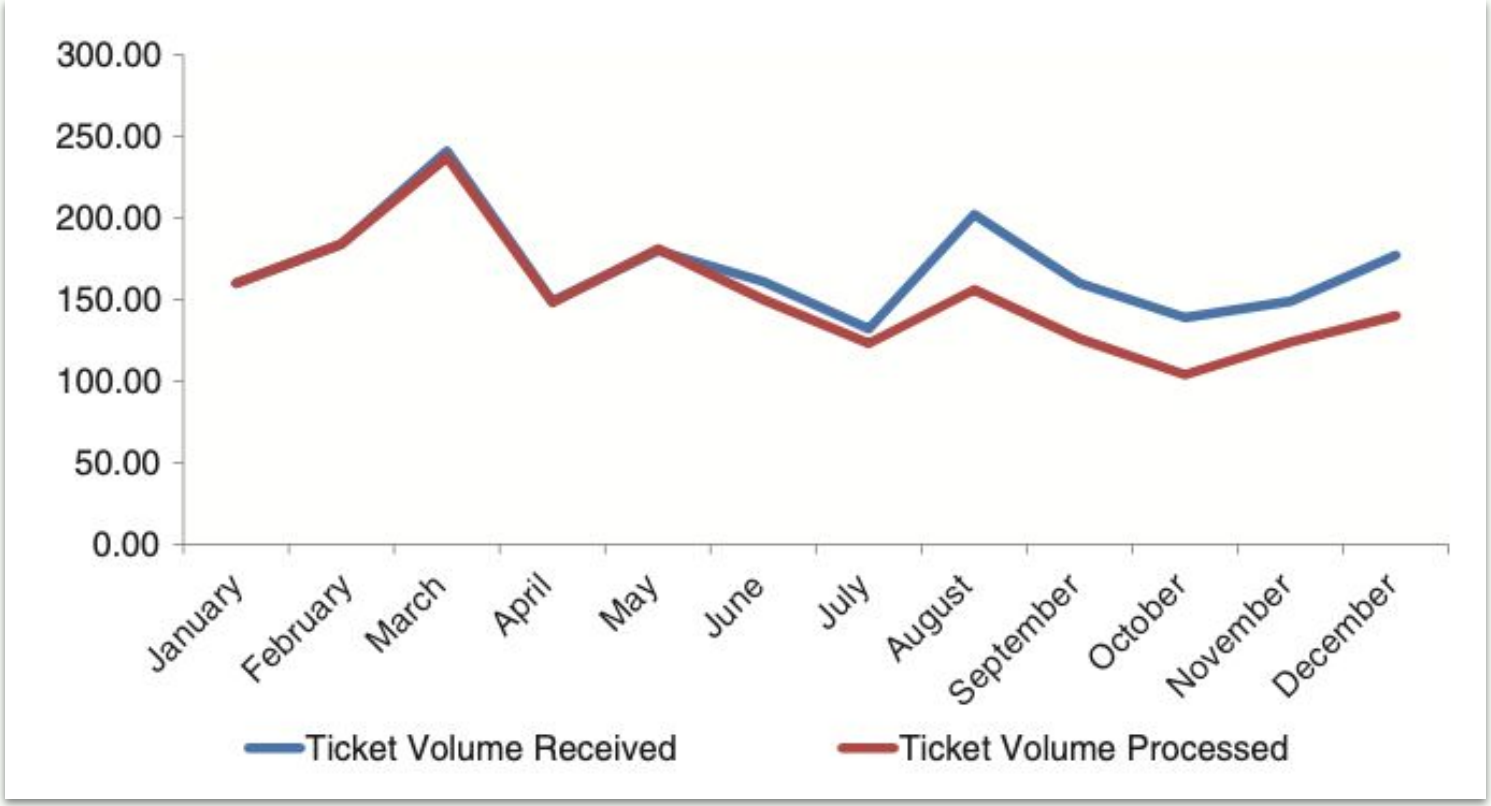
Starting Point (Not Bad!)



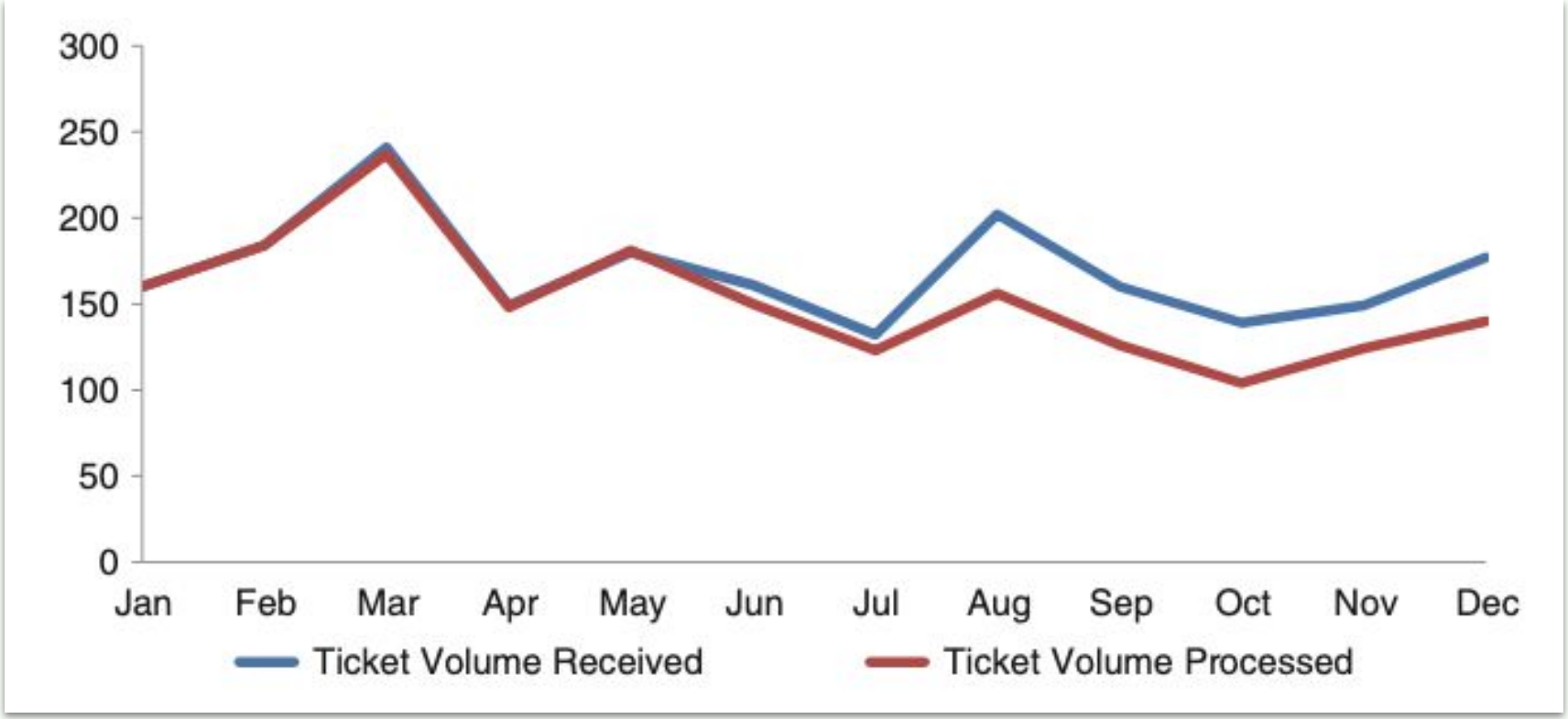
Step 2: Do We Need Grid Lines?



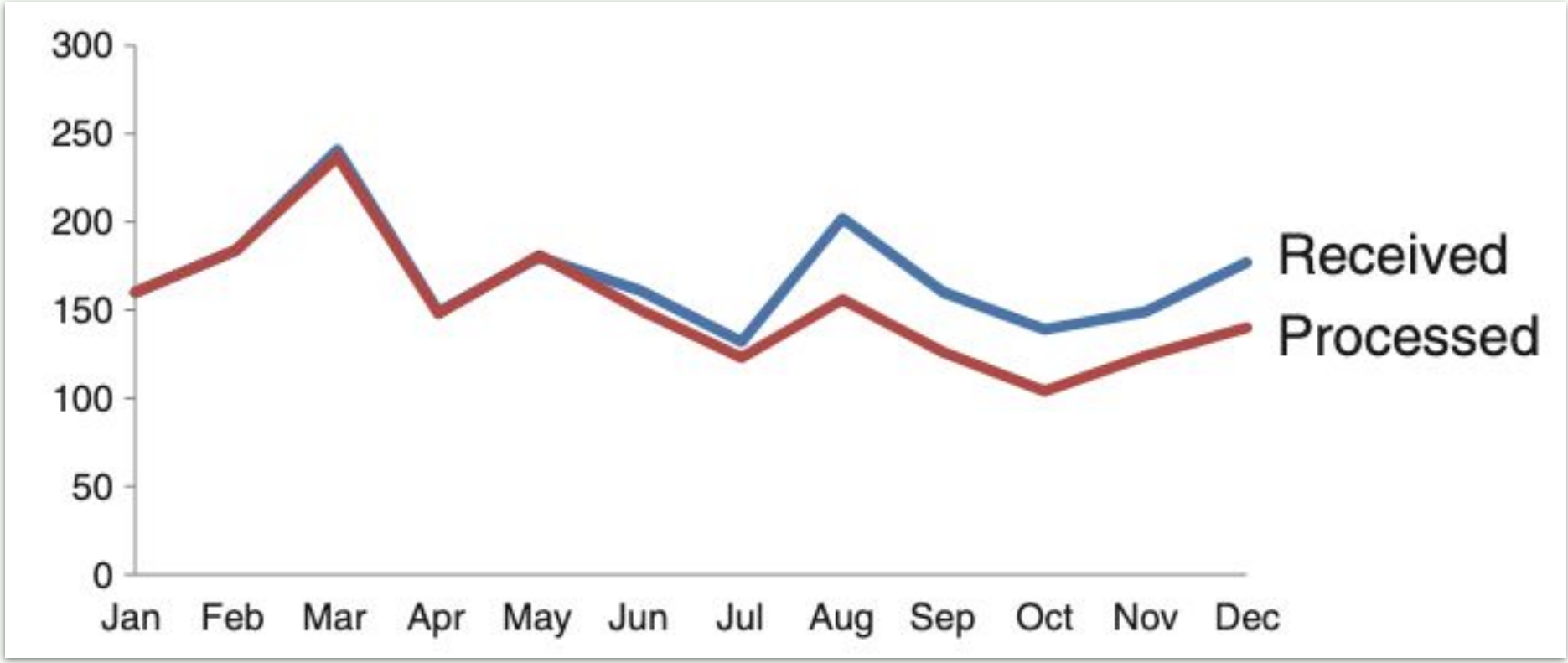
Step 3: Markers Are Redundant?



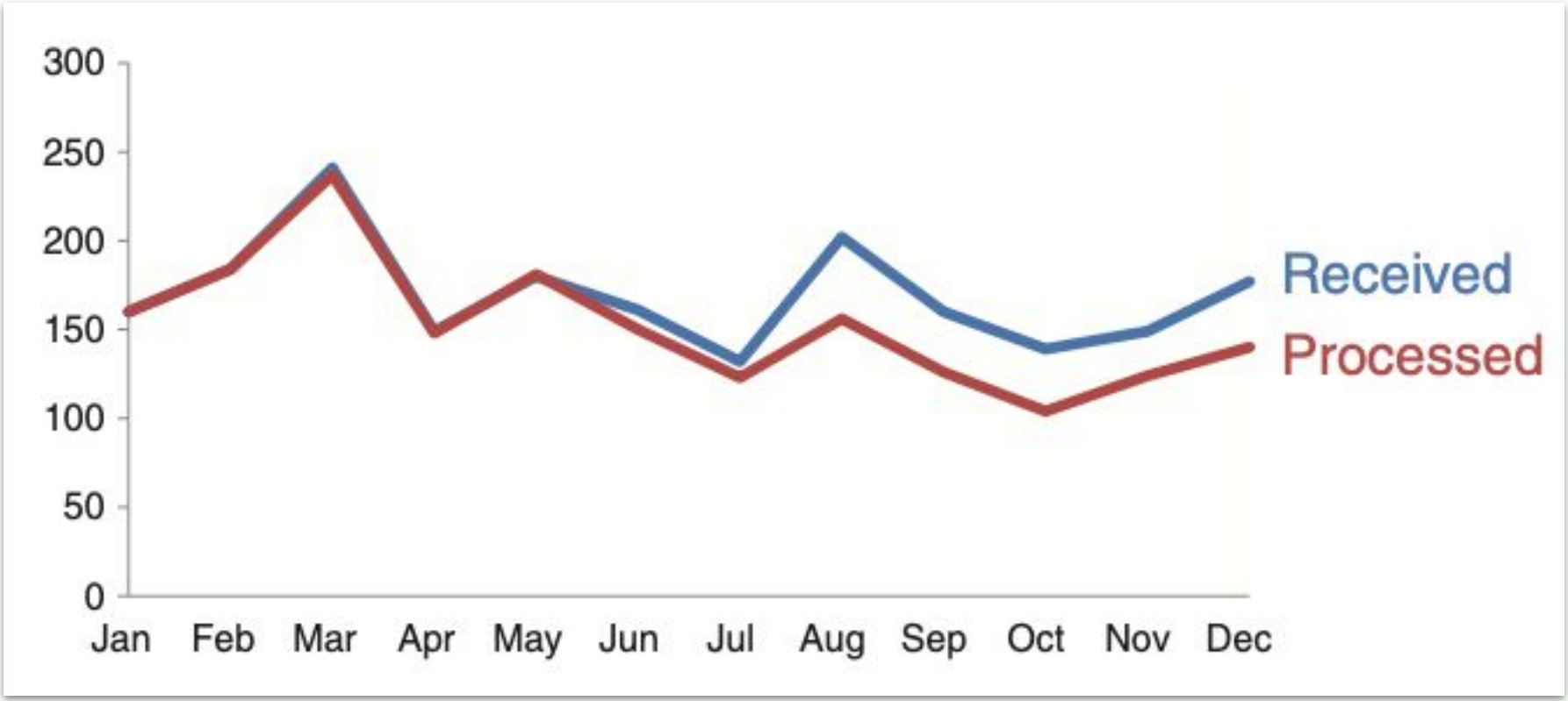
Step 4: Too Much Axis-Label Clutter



Step 5: Move Legend To Data

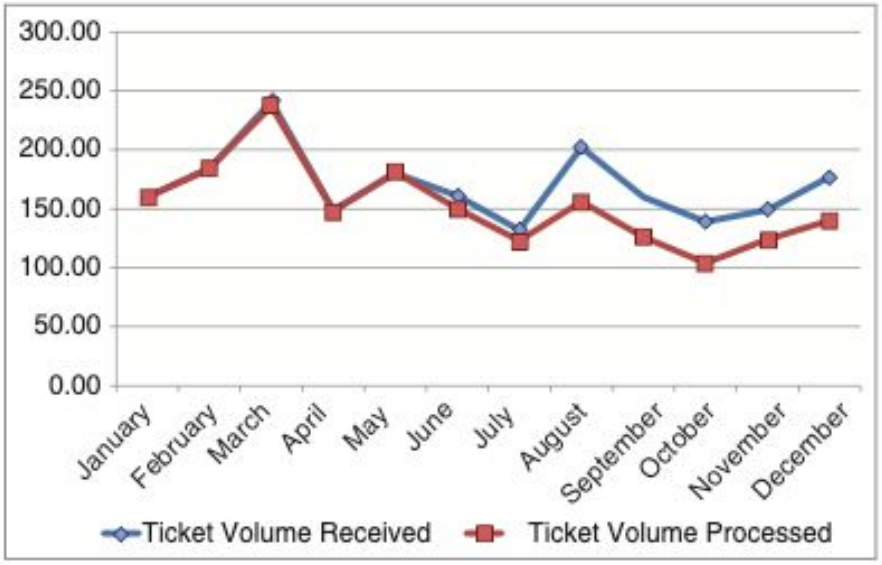


Step 6: Color Code Labels To Data

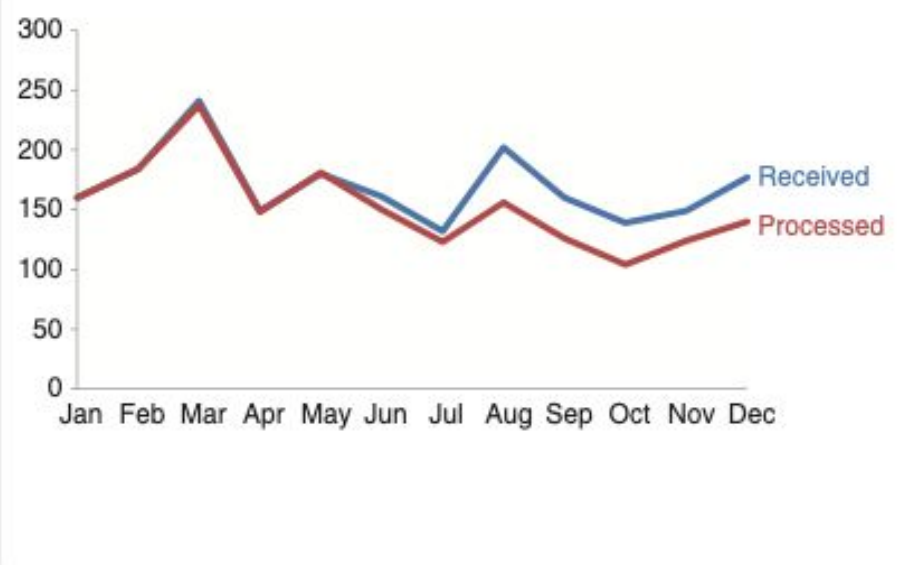


Summary So Far

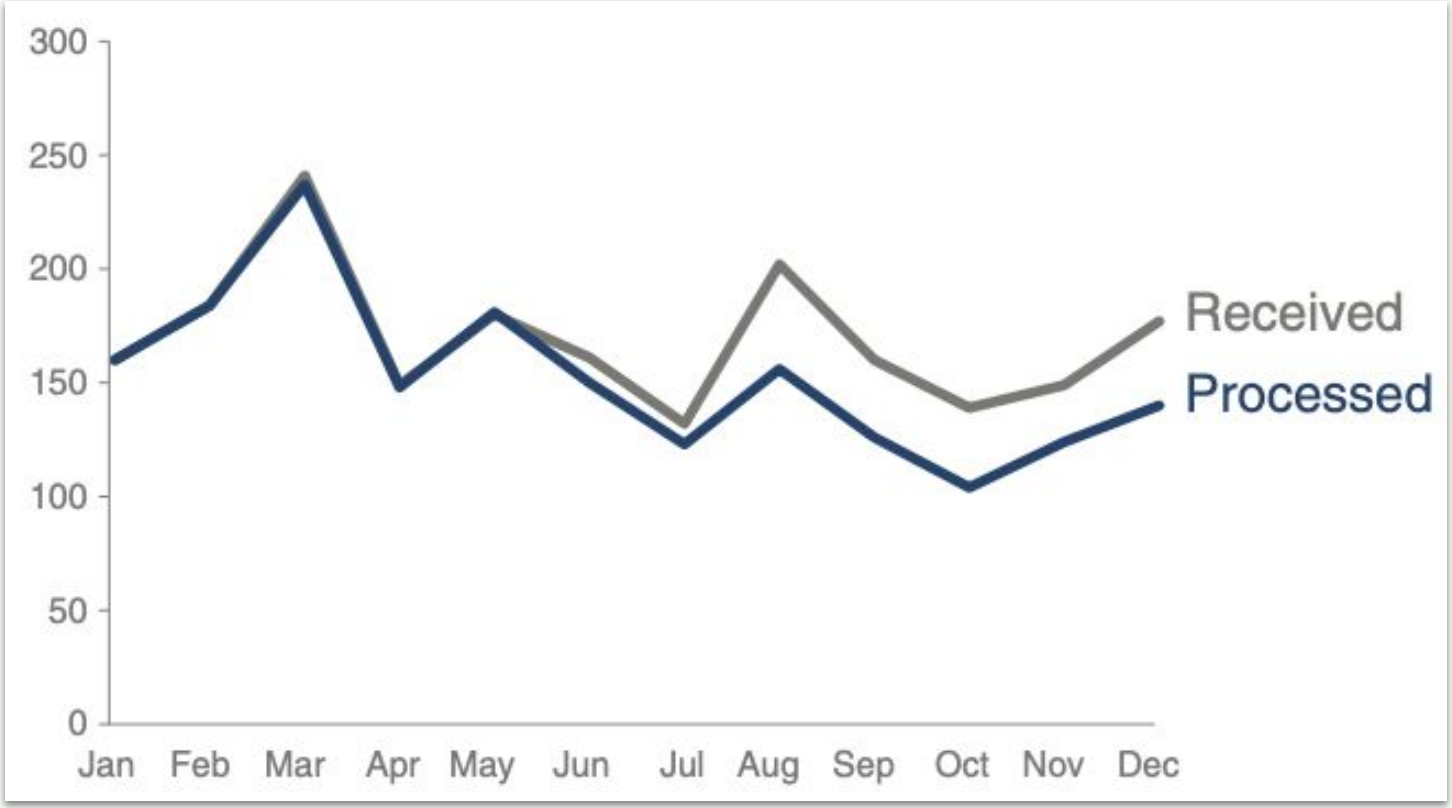
starting point



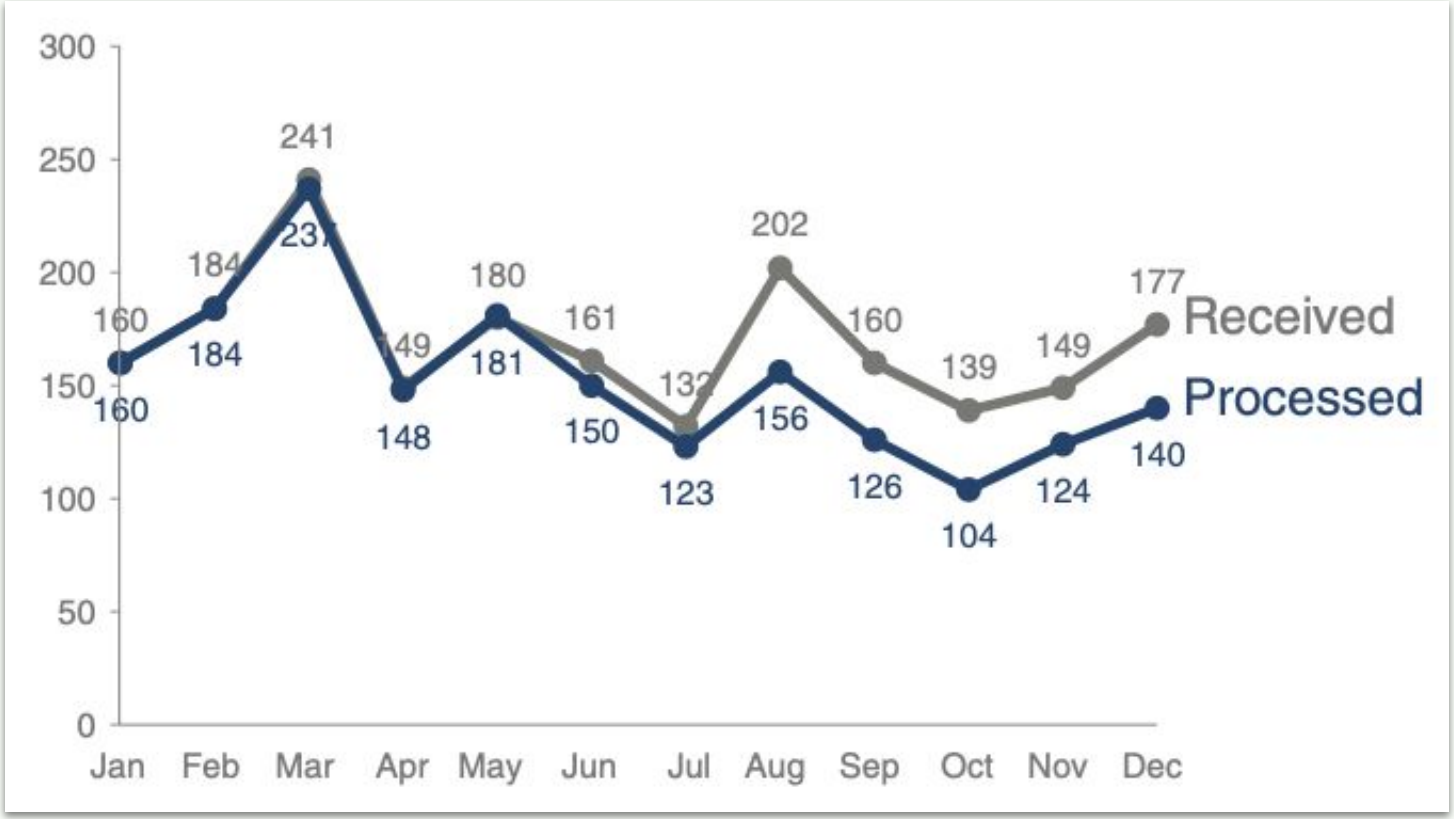
clean, quick to comprehend!



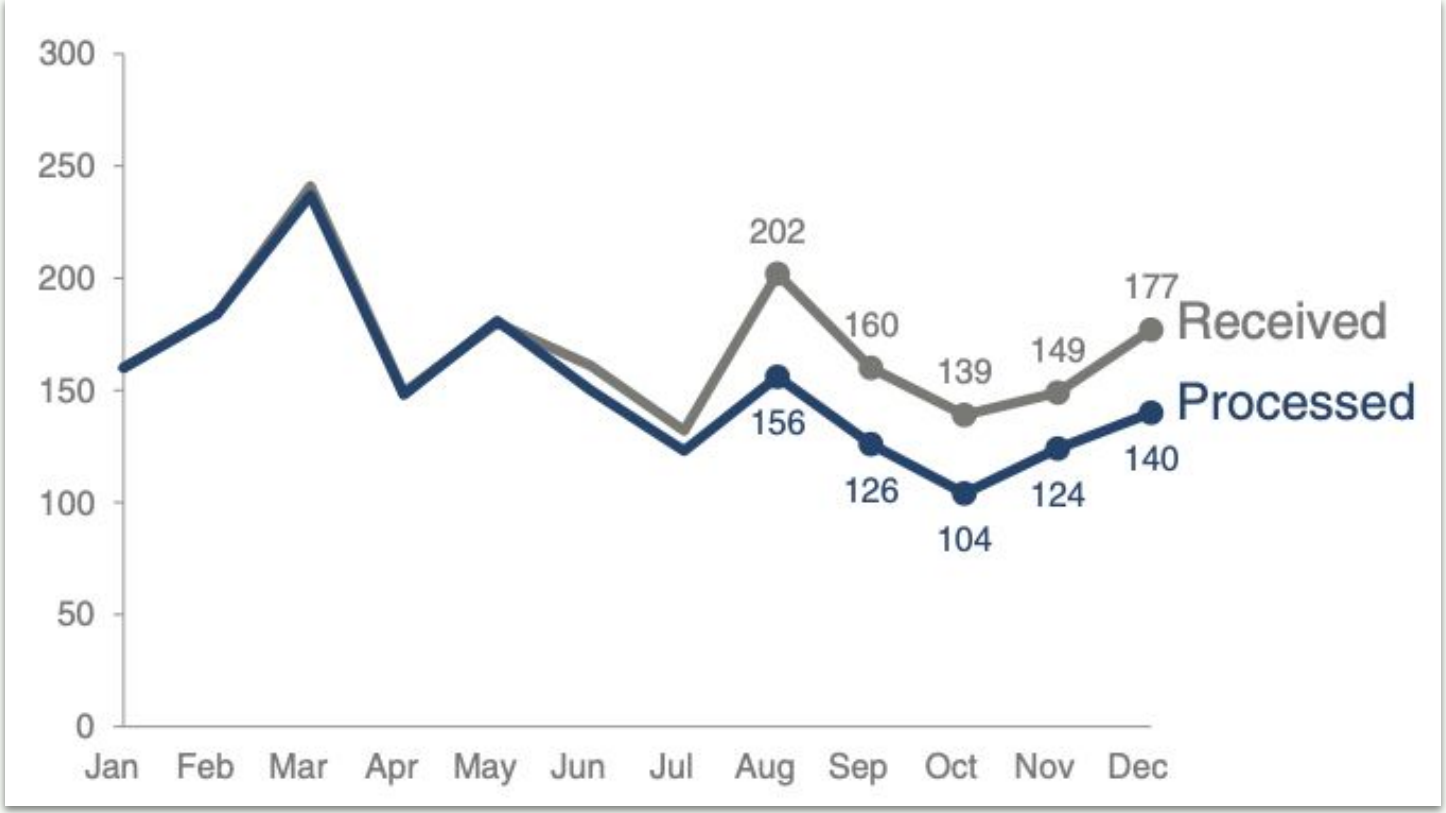
Step 7: Gray-Out Less Important Story



Step 8: Highlight Specific Values

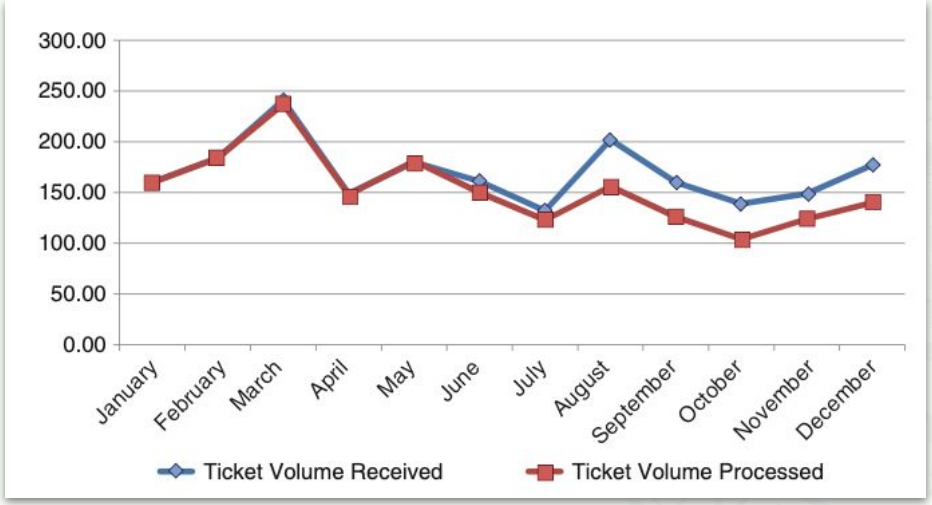


Step 9: But, Only Where The Story Is!



We're Done!

starting point



A story has emerged...