Redwood Lab: Exploratory Data Analysis (EDA)

February 5, 2025

Today's plan

1 **Review: Data Cleaning**

2 **Exploratory Data Analysis**

Our data cleaning journey: it's been a trek

Where we started:

- Motes info + dates table + three different redwood sensor datasets (all, log, net)
  - All = log + net but without "source" id
  - So many problems...
- Removed duplicates NAs in redwood sensor dataset
  - Can remove rows with NAs without worries since entire row of sensor measurements were NAs

Our data cleaning journey: it's been a trek

After some cleaning/preprocessing:

- Merged into one data frame with "source" id and removed duplicates

Why do we care so much about the "source"?

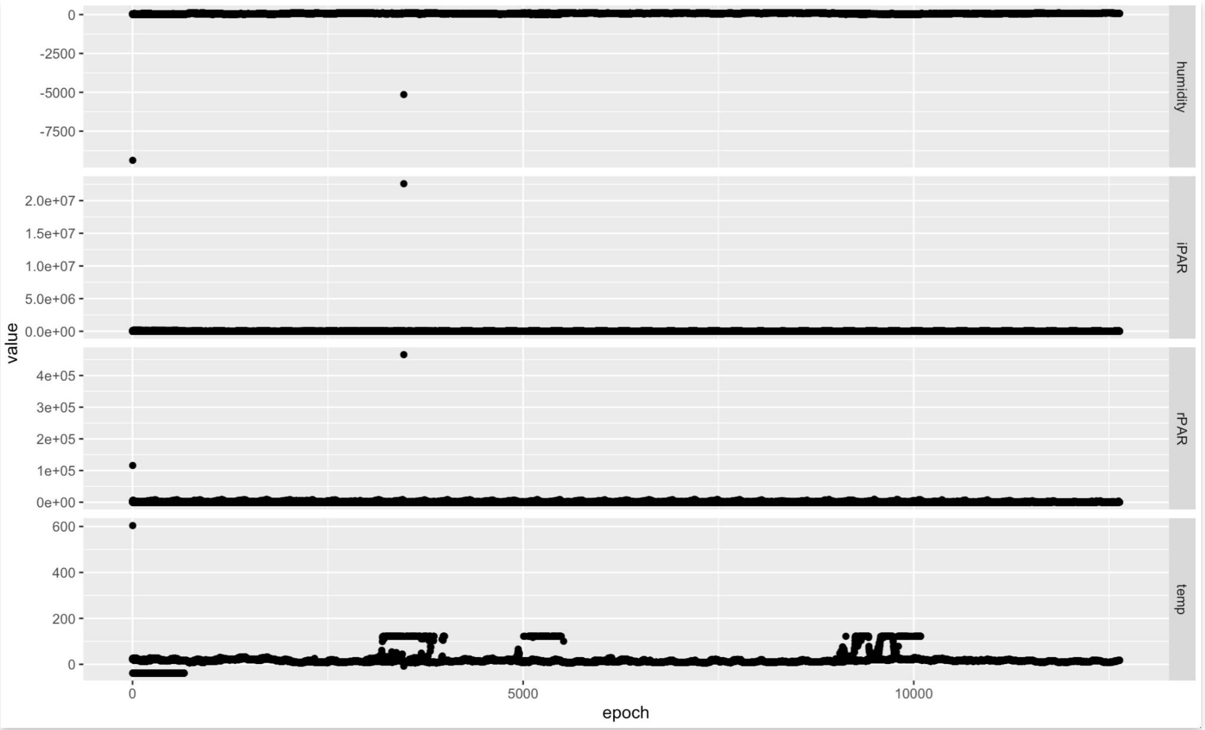
- **The data collection process should guide our data cleaning!**
- Revealed many issues:
    - result_time is constant in log data
    - voltages are different between two sources
    - lots of outliers in network data (also in log)

Why did we take the time to merge all of this data into one data frame?

- So that all of the information is in one data frame and can be readily used for more data cleaning and EDA
    - E.g., plot by source, mote location (height, direction), time of day, ...

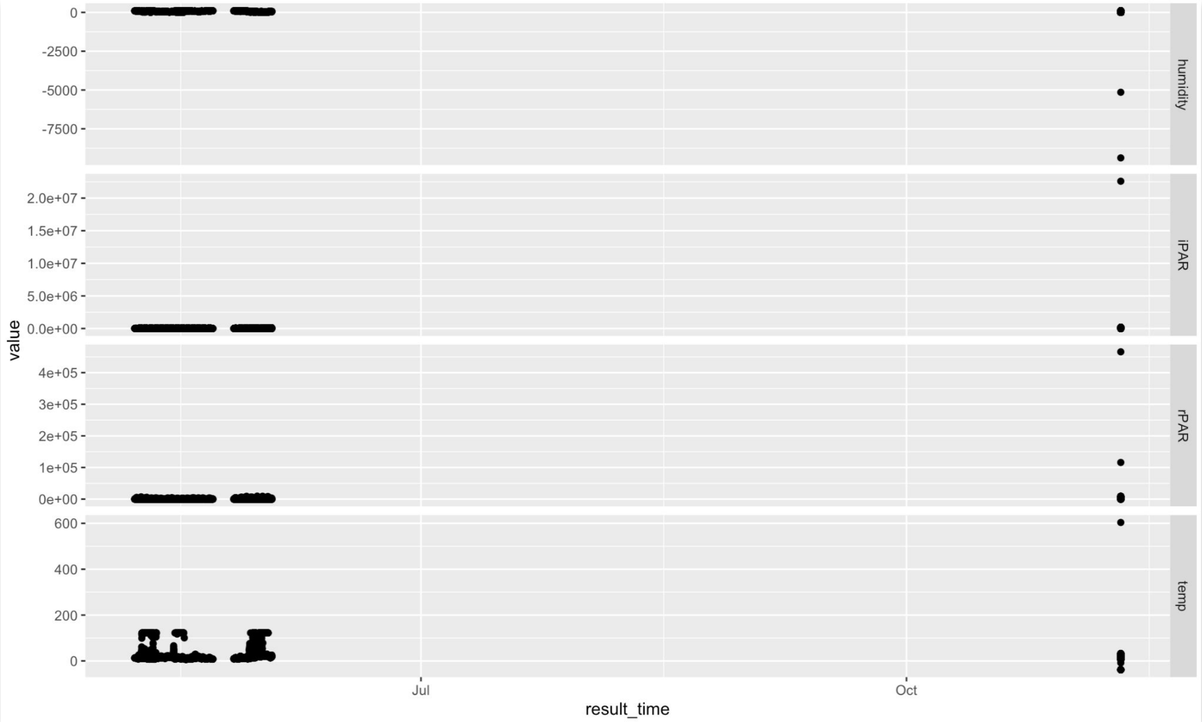
Our data cleaning journey: it's been a trek

Our first glance:



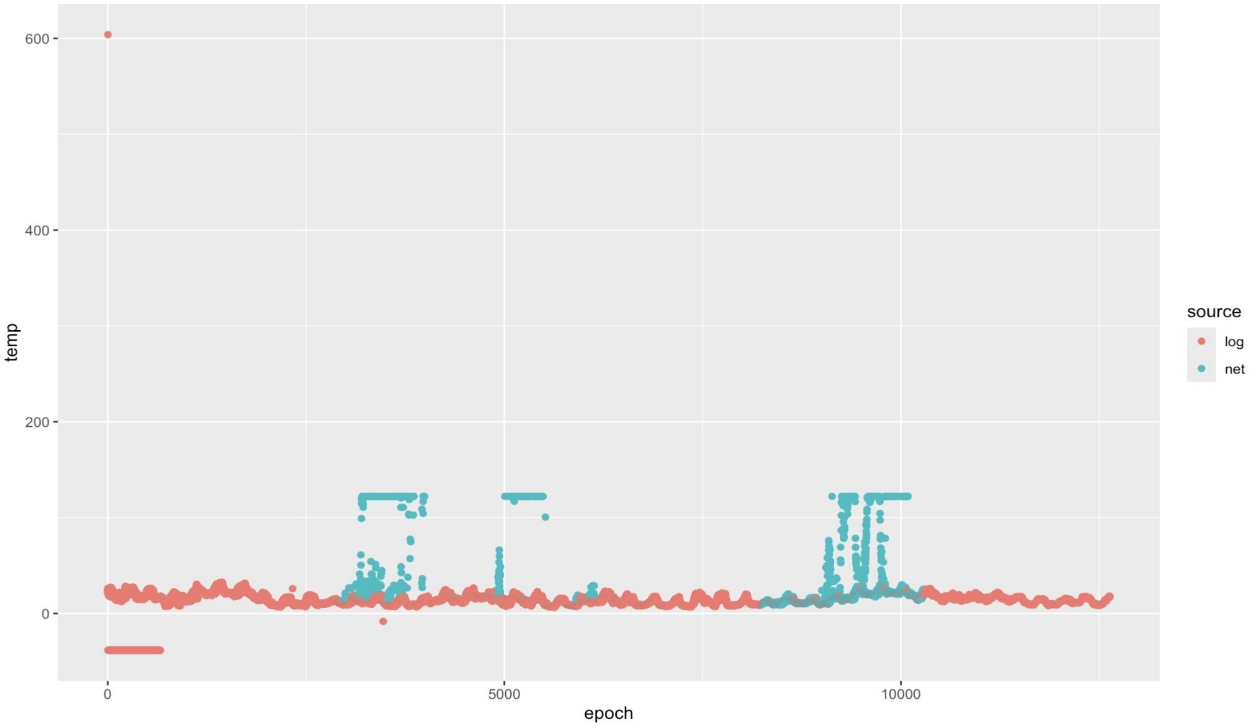
Our data cleaning journey: it's been a trek

What could've gone wrong:



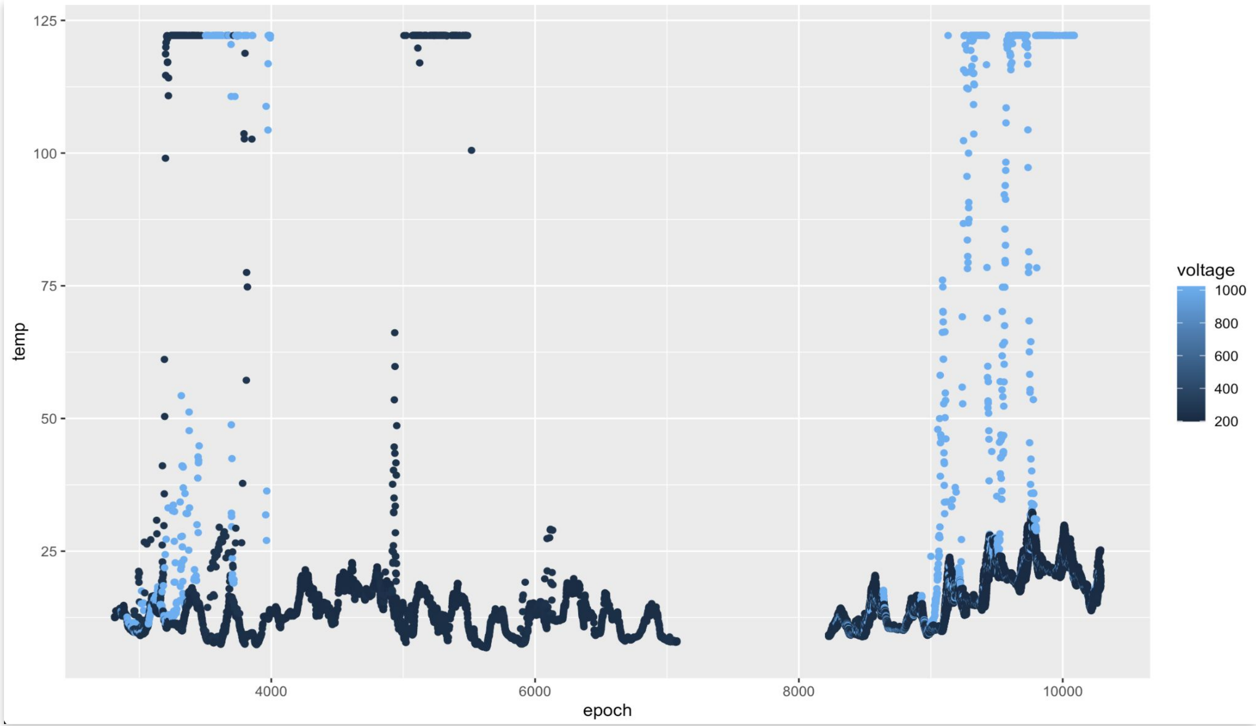
Our data cleaning journey: it's been a trek

Taking a closer look at temperature:



Our data cleaning journey: it's been a trek

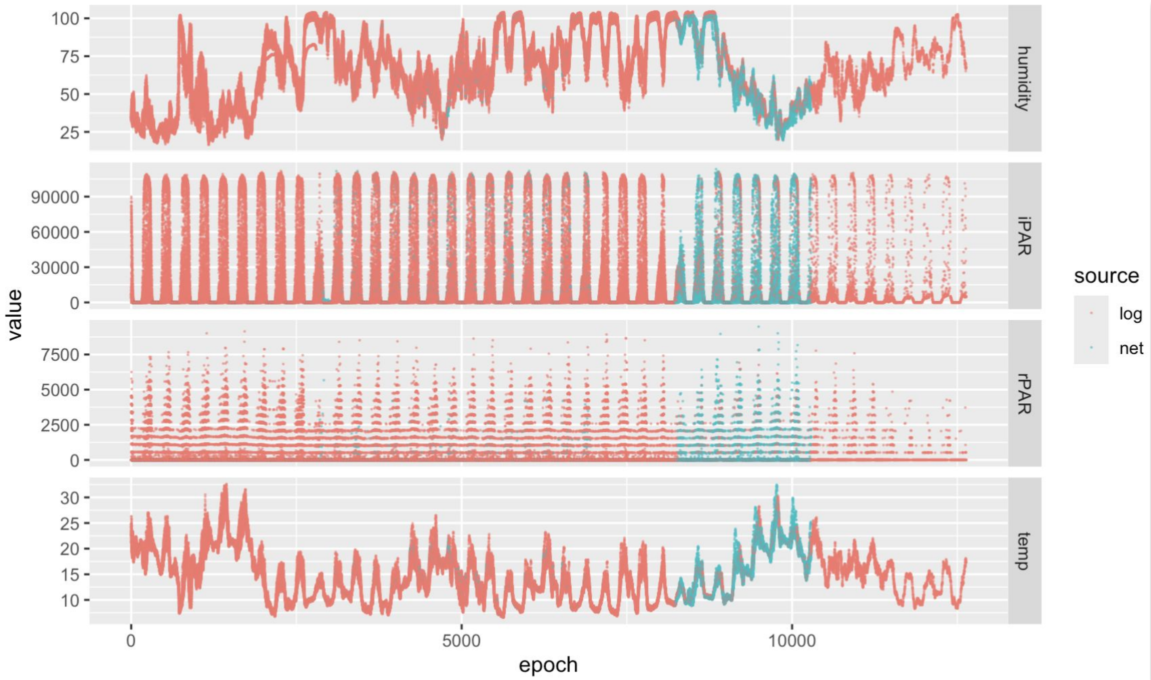
What about the data collection makes the recorded temperatures trail off like that?



Our data cleaning journey: it's been a trek

What next? Use this new information to **iteratively refine** your data cleaning, e.g.,

- Identify one issue
- Fix the issue
- Identify another issue
- Fix the issue
- Do some EDA
- Find another issue
- Fix the issue
- …

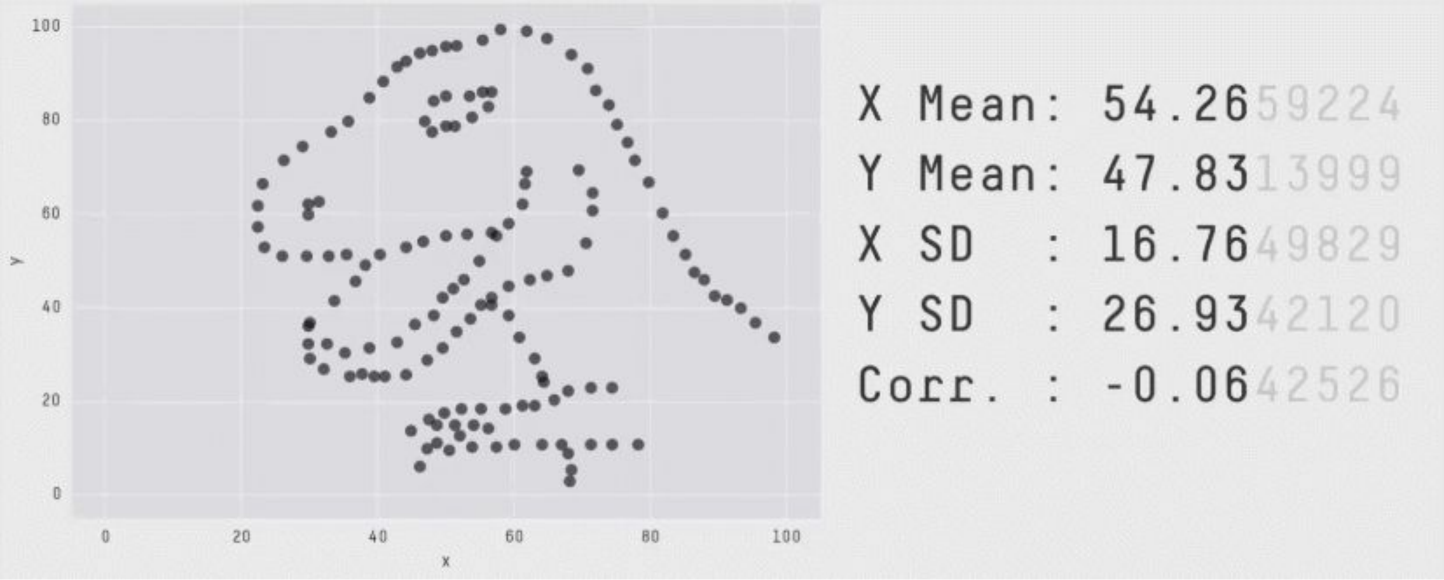


Exploratory Data Analysis (EDA)

Why do we need EDA/visualizations?

Visualizations can tell a more detailed story than numeric summaries

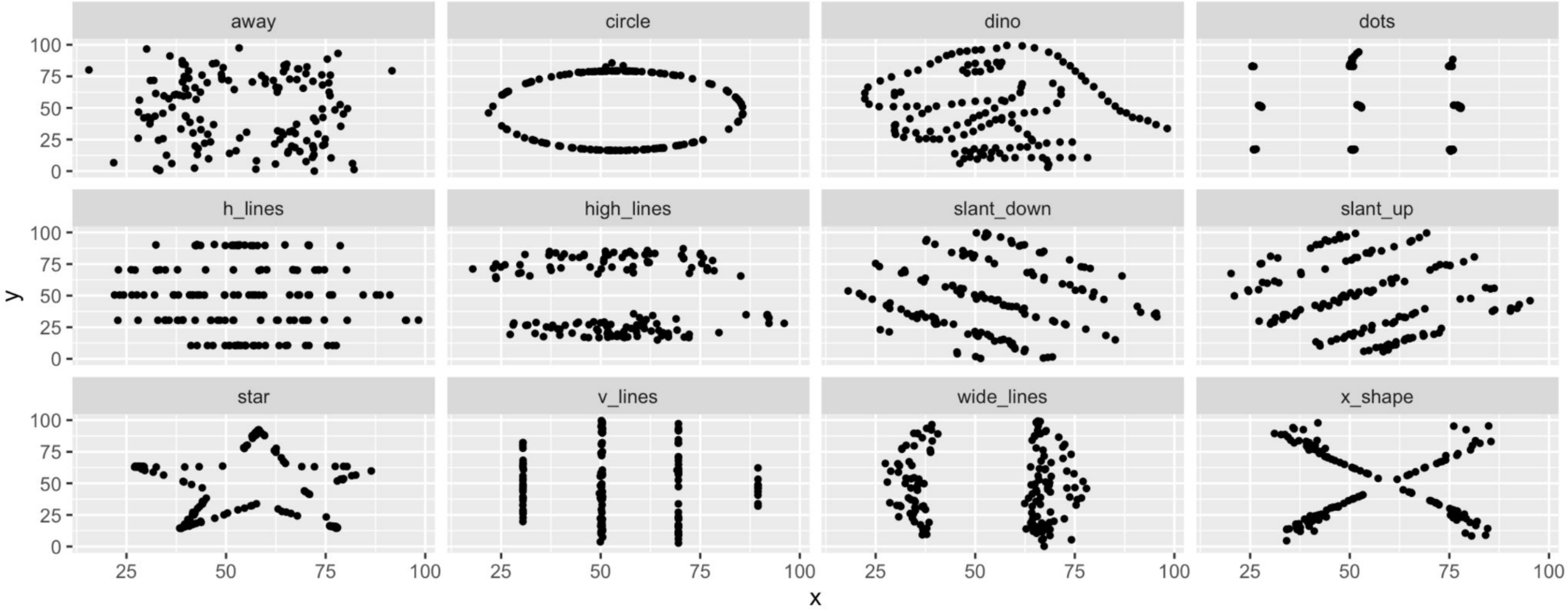
+ Remember this when reporting p-values!



"The Datasaurus Dozen" [Matejka and Fitzmaurice (2017)]

Why do we need EDA/visualizations?

Each of these has the same mean, standard deviation, variance, and correlation



"The Datasaurus Dozen" [Matejka and Fitzmaurice (2017)]

Exploratory Data Analysis (EDA): Purpose

What can we use EDA for?

- To illuminate data oddities and **inform data cleaning**
- To provide insights on the inherent data structure that can **guide modeling**
- To **discover substantively-meaningful patterns** (e.g., unsupervised learning)
- Others?

Two modes of EDA plots

- “Scratchwork”: for internal use
- “Publication-quality”: for public use

"Scratchwork" Plots (for internal use)

"Scratchwork" Mode: Quantity over quality

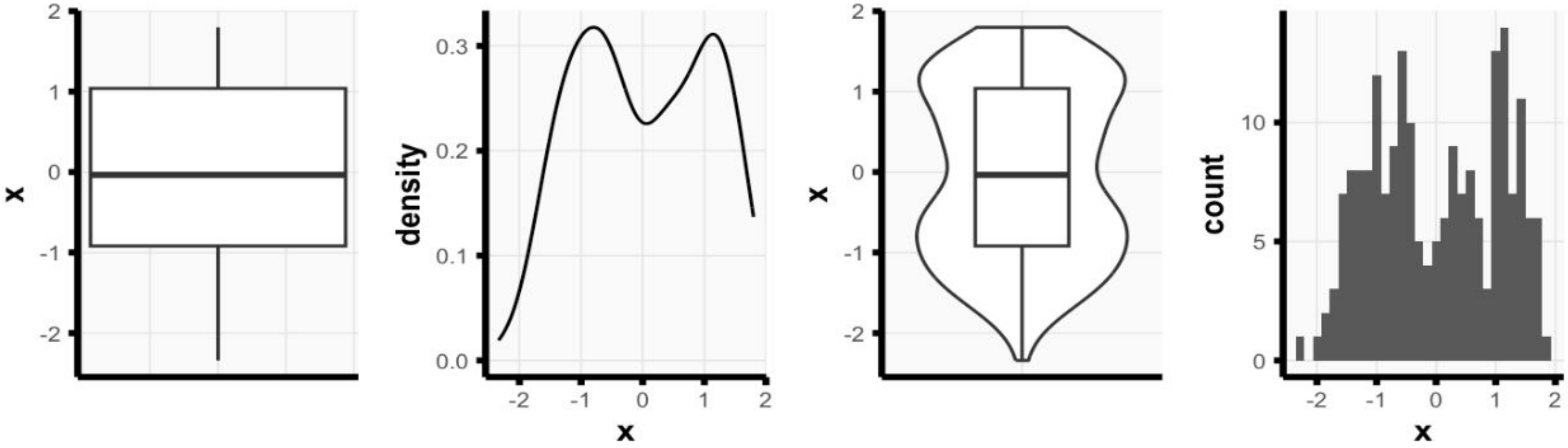
What are some quick plots that you would make when digging into a dataset for the first time?

- Histograms/density/boxplot of the data distribution
- Plots to view the pairwise relationships between variables/covariates/features
  - (Clustered) correlation heatmaps
    [check out `superheat::superheat` (R) and `seaborn.clustermap` (Python)]
  - Scattered pair plots
    [check out `GGally::ggpairs` (R) and `seaborn.PairGrid` (Python)]
- Scatter plots
- 3d plots [check out `plotly` (R and Python)]
- Heatmaps [check out `ggplot::geom_tile` or `superheat::superheat` (R) and `seaborn.heatmap` (Python)]
- Others?

"Scratchwork" Mode

Quantity over quality: Plot the same data in multiple different ways

Example: Four different ways of plotting a data distribution



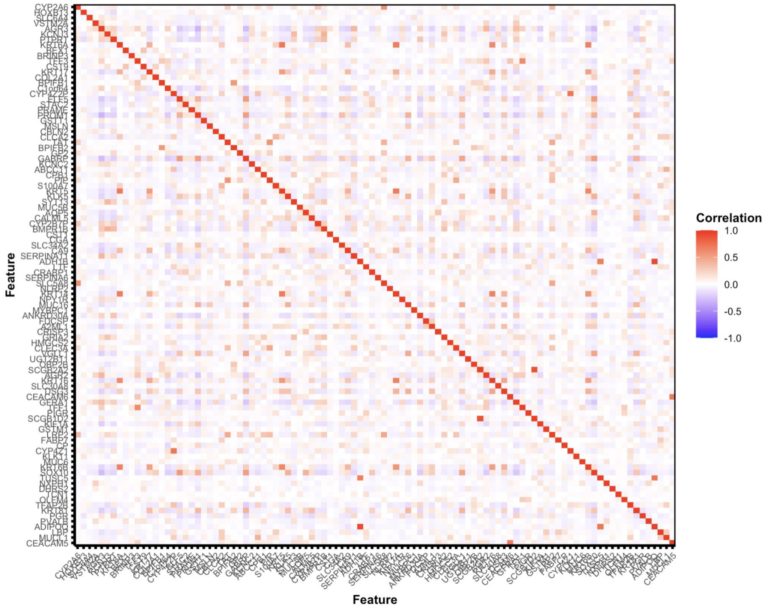
+ different kernel bandwidth, number of histogram bins, etc

"Scratchwork" Mode

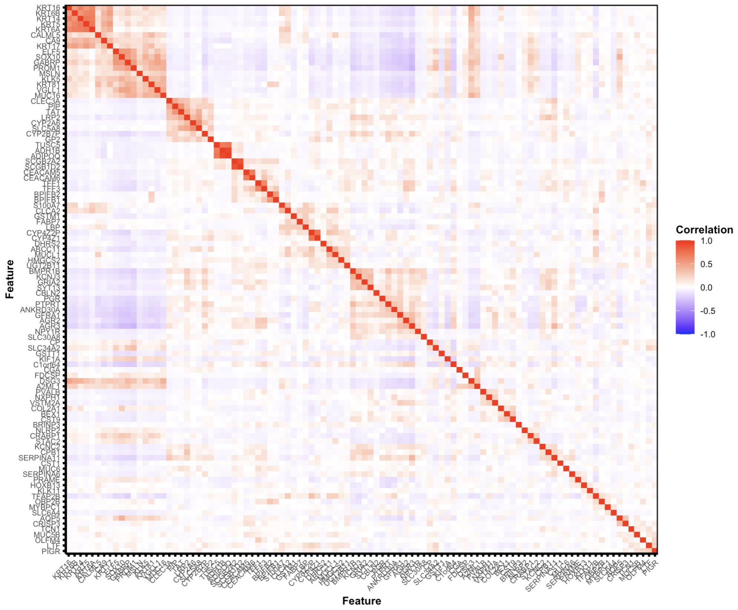
Quantity over quality: Plot the same data in multiple different ways

Example: Two correlation plots of the same data

Unclustered Heatmap



Clustered Heatmap

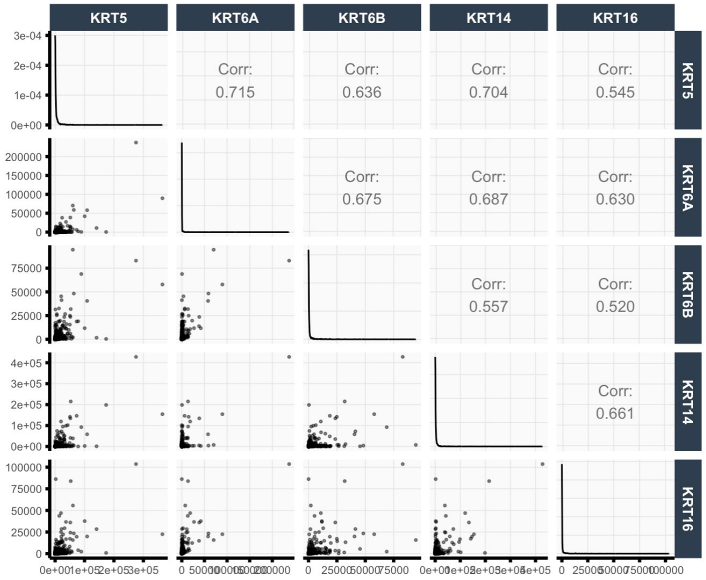


"Scratchwork" Mode

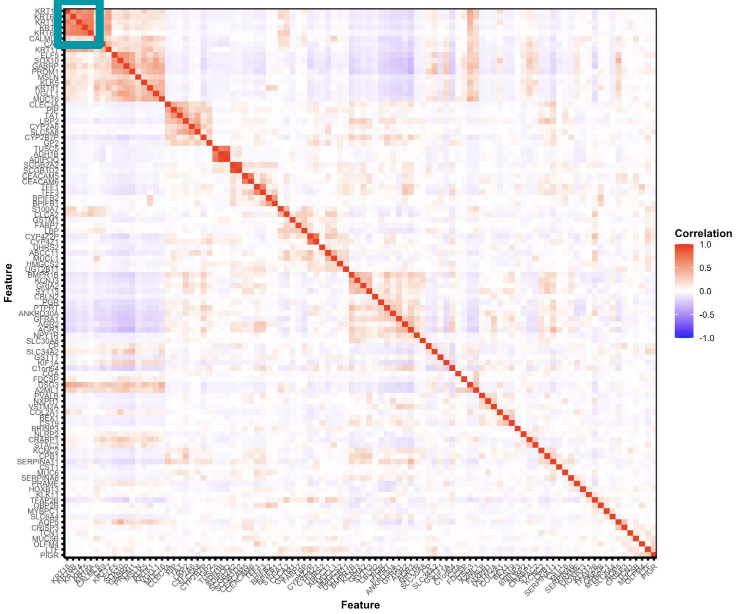
Quantity over quality: Plot the same data in multiple different ways

Example: Two correlation plots of the same data

Pair plot



Clustered Heatmap



"Publication-quality" Plots (for public use)

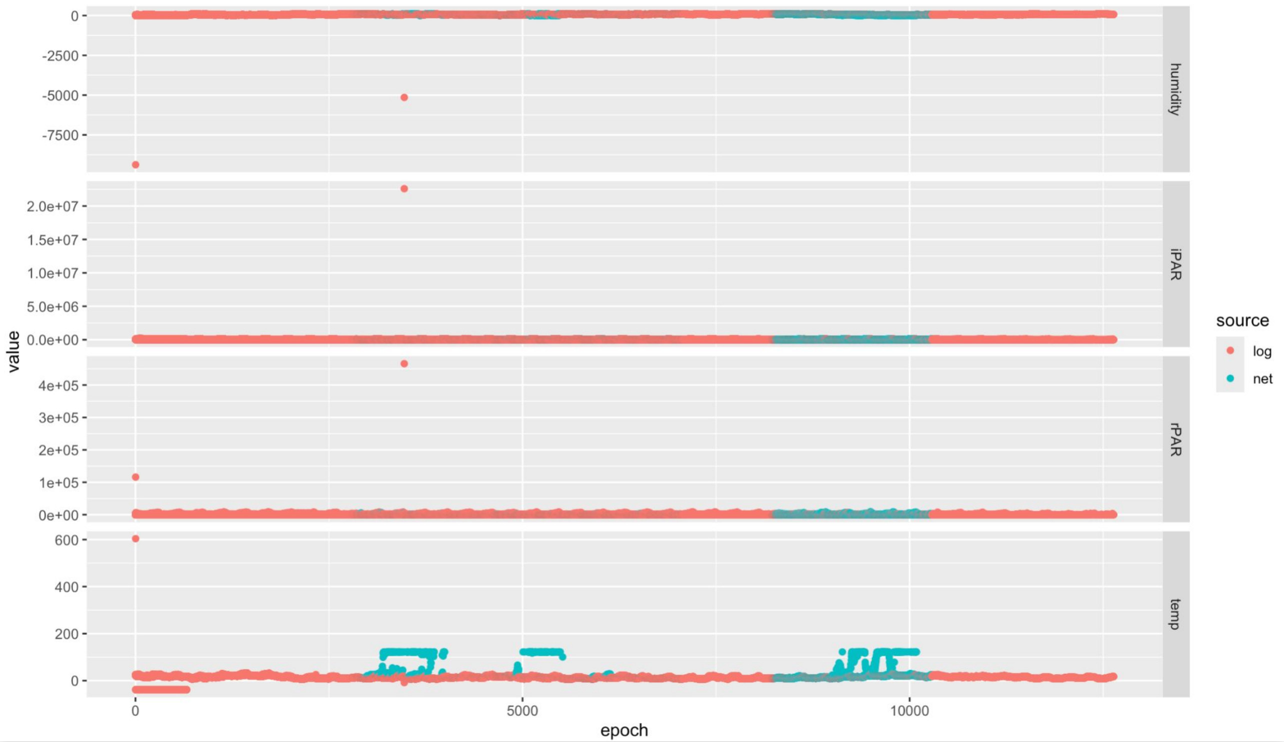
When presenting EDA visualizations to the public...

Remember the #1 rule: First think about your main takeaway.

Then craft the plot to clearly communicate this singular message.

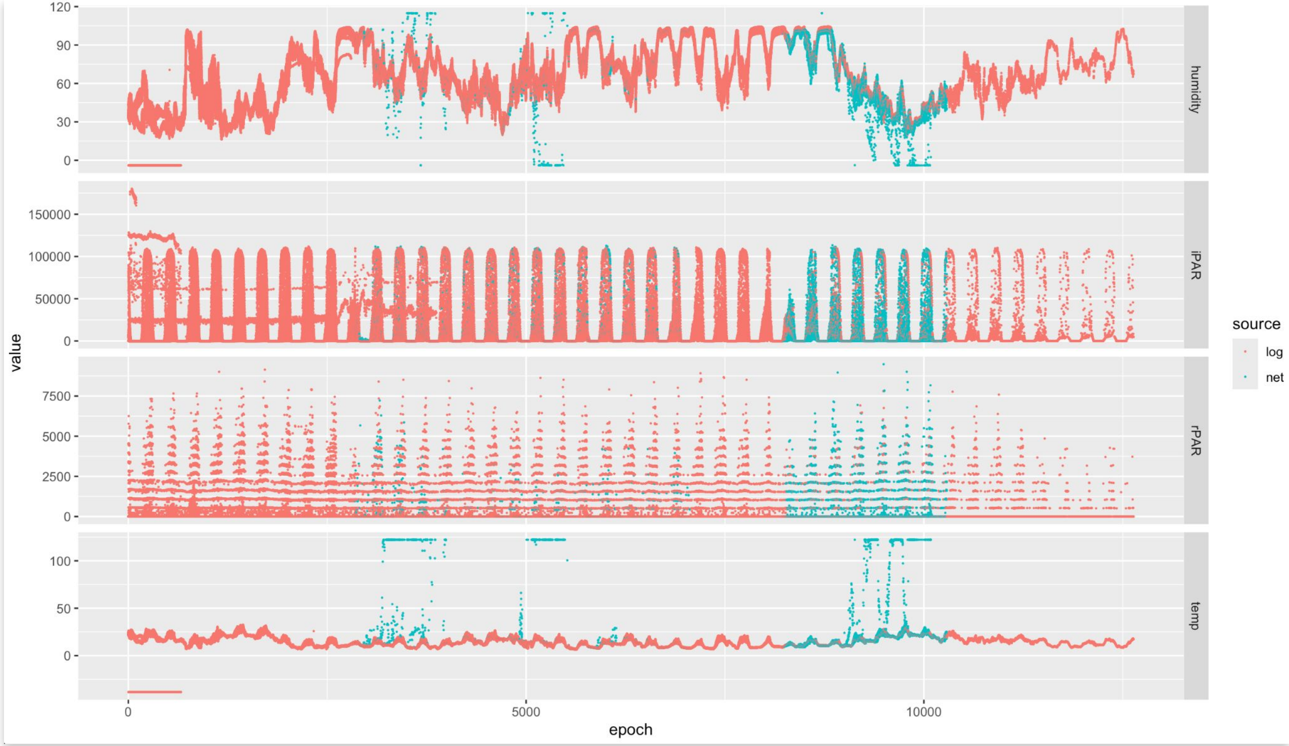
EDA Example: Before

Main message: Outliers generally come from the network data



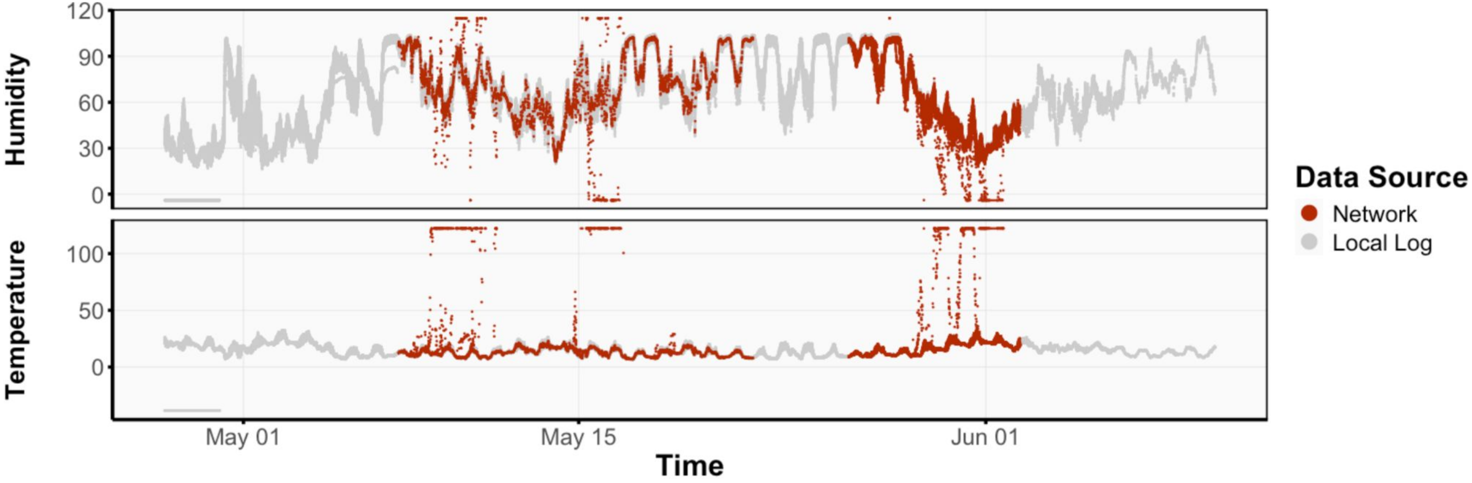
EDA Example: Before

Main message: Outliers generally come from the network data



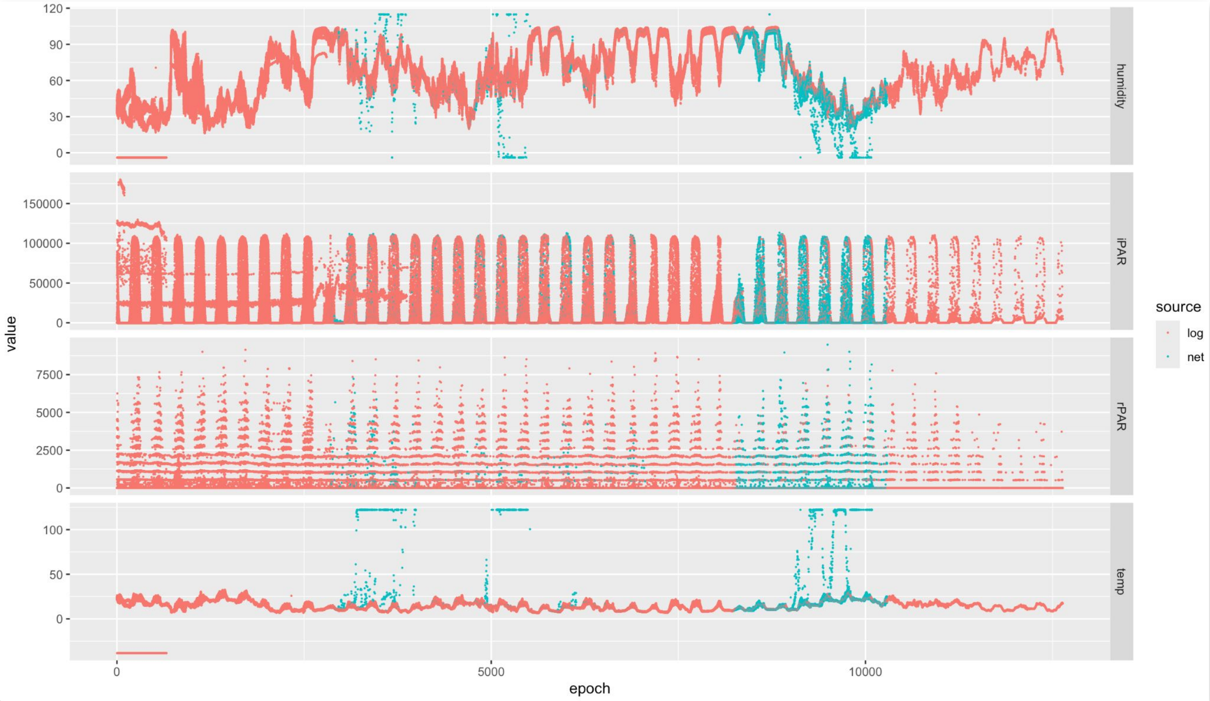
EDA Example: After

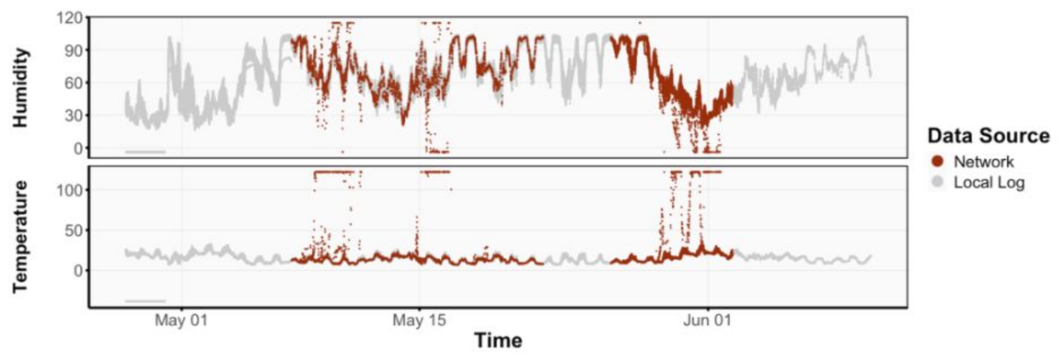
Main message: Outliers generally come from the network data



EDA Example

Spot the differences



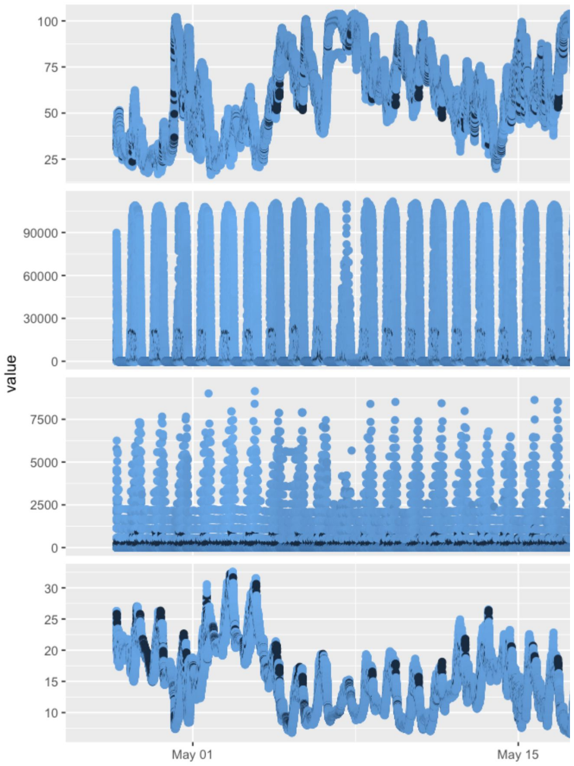


Basic Aesthetics Checklist

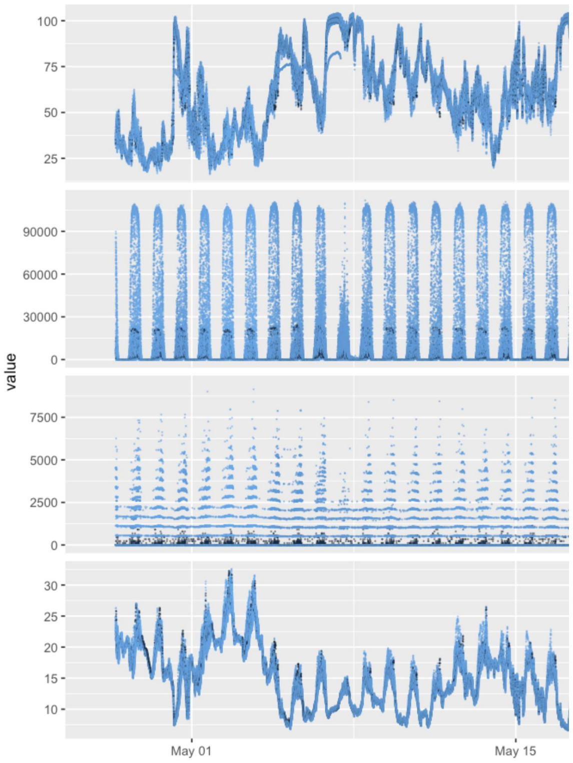
- Labels should be meaningful (not variable names)
  - E.g., plot date/time instead of epochs in redwood lab
- Add labels and capitalize them appropriately
- Text size should be large enough and legible (e.g., in writeups and on slides)
- Legend order matters
- Change the (ggplot) theme
- Choose colors thoughtfully
- Did I overplot?

The biggest pitfall in EDA/visualizations: **Overplotting**

Bad



Better



Strategies to avoid overplotting

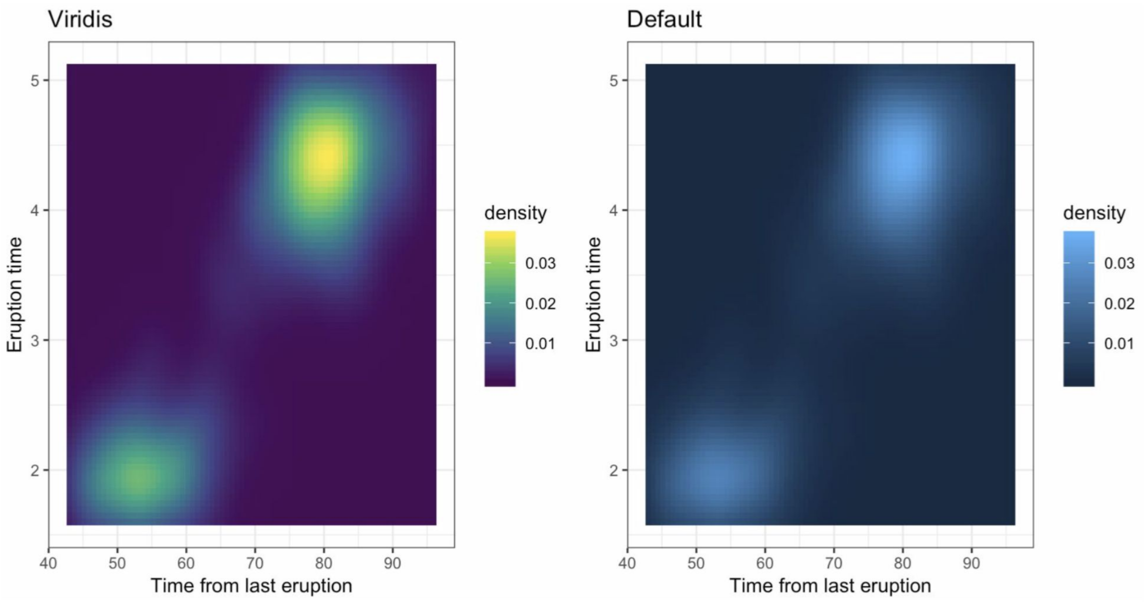
- Use smaller point sizes

  ggplot2: geom_point(size = ...)
  matplotlib: plot.plot(markersize = ...)

- Use transparency (alpha)
- Subsample data points
- Remember to focus on a singular message

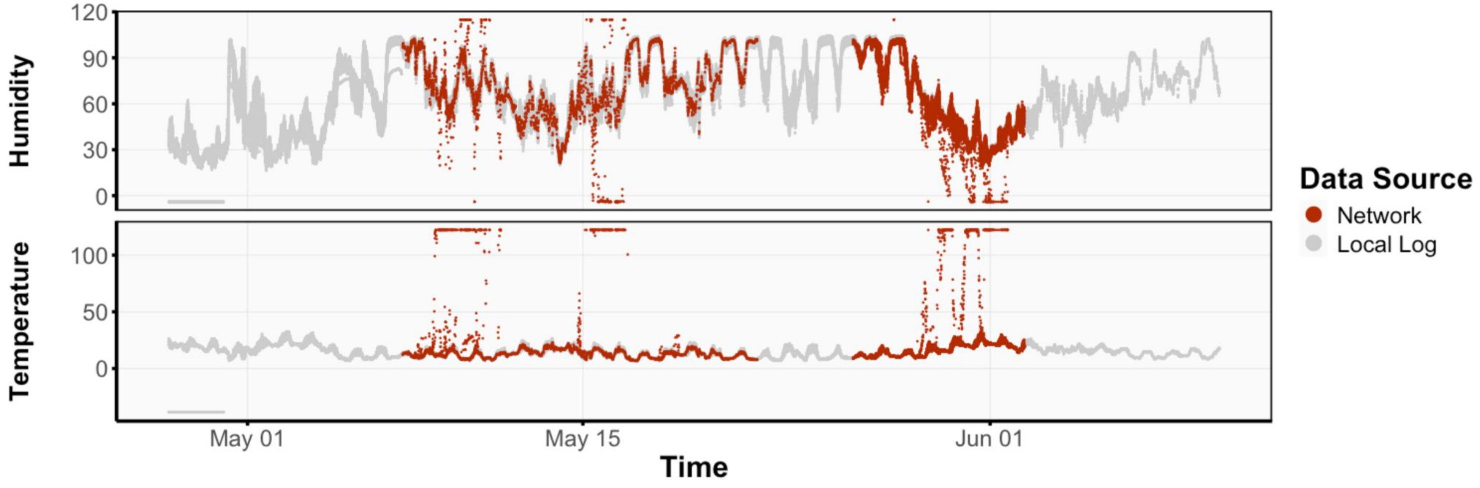
Color matters!

- Color choices can affect the way we perceive the plot



Color matters!

- Color choices can affect the way we perceive the plot



Color matters!

When choosing colors, be considerate of...

- Color blind friendly
  - ~10% of all men are red-green colorblind
- Colors have inherent connotations
  - Red = bad
  - Green = good
  - Gray = ignored
  - Black = bold/draws attention to
- Discrete versus continuous color scales
- Different shades of the same color suggest relatedness

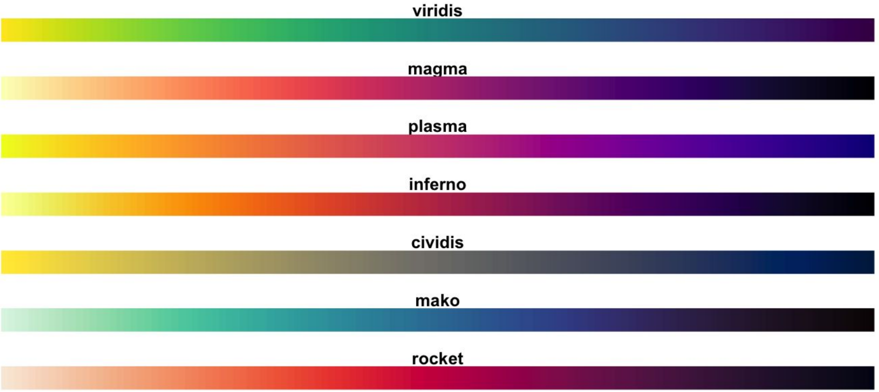
Resources for choosing colors

Color scheme generator: https://coolors.co/

HTML color codes: https://htmlcolorcodes.com/

Encycolorpedia: https://encycolorpedia.com/

Viridis color palette



Exploratory Data Analysis (EDA) tips in a nutshell

- Start with your domain problem
- Explore with "scratchwork" EDA: quantity over quality
- Once you have identified your main finding, think before you plot
  - Your plot should clearly communicate a **singular message**
  - Your main EDA plot should not be a "data cleaning" plot
- Plot type should be an intentional choice
  - Line, scatter, bar, heatmap, ...
- Aesthetics matter
  - Color
  - Point size
  - Transparency
  - Labels
  - Theme
  - Be wary of overplotting
- **Take your time**

Sprucing up your visualizations with interactivity

- Shiny: https://shiny.posit.co/
  - R Tutorial: https://shiny.posit.co/r/getstarted/shiny-basics/lesson1/
  - Python Tutorial: https://shiny.posit.co/py/docs/overview.html
- Plotly
  - R: https://plotly.com/r/ (also see `plotly::ggplotly()`)
  - Python: https://plotly.com/python/

If you need inspiration for visualizations…

NY Times Data Visualizations: https://www.nytimes.com/column/whats-going-on-in-this-graph

- Great for finding new color schemes

Storytelling with Data: https://www.storytellingwithdata.com/

Recap + Next Time

Recap

- **Exploratory data analysis** is a great way to get a feel for the data. [chapter 5 from VDS textbook]
  - "Scratchwork" EDA (internal): quantity over quality
  - "Publication-quality" EDA (public): quality over quantity
    - Think then plot

https://pollev.com/tiffanytang211

Don't forget

- Lab 1 due **Sunday 5pm** submitted to GitHub

Next Time

- Beginning of unsupervised learning unit