

Group Project Handout

Scott Thatcher

Introduction

In the group project, you'll be working on a team to produce an interactive data visualization, along with written documentation. Both aspects of the project will be presented at a single web-accessible address, and your group will present its results during the last week of classes.

The sections below detail what I expect to see in the project.

Finding a Data set

Find a data set of interest to your group. This can be the same as a data set used by one of the group members for the first paper. It might also be totally new. The data set should

- be a real-world data set from which you can derive interesting information,
- contain enough data points to make analysis and visualization possible (although it doesn't have to be "big data"),
- be rich enough (often meaning having enough variables) that interesting multivariate visualizations can be created, and
- have enough documentation available that you can write about where the data came from and what the variables represent.

If your data set has more variables than you could possibly use, it would be a good idea to create a paired-down and cleaned data set before starting to make your app. Select only the variables you plan to use in the app.

Talk with me soon if your group has trouble finding a data set.

The Design Process

After finding your data set, your group should explore the data, get to know its variables, then spend some time mocking up a design for your interactive visualization. You should be able to do this before doing much coding. You know what's possible from the tools we've seen. Brainstorm among the group about what sort of visualization would be instructive or useful to the potential consumers of this data.

Coding

As I've suggested before, it may be a good idea to work in stages:

- First write code that creates “static” versions of the graph or graphs you want. (This could also be a way to break up the work among your team.)
- Make sure the code to generate those static versions works.
- After you've done that, you can gradually add in code to make the interactive visualization.
- Add code a bit at a time, and make sure that each bit you add doesn't break anything.

How much interactivity do you need? We've seen several tools that create interactive visualizations: `shiny`, `plotly`, `leaflet`, and `tmap`. Default `plotly` and `leaflet` objects have a built-in amount of interactivity on their own. I expect that you will have more interactivity than that. A `shiny` app is the obvious choice. It's also possible to build more interactivity into a `plotly` object (drop-downs, sliders, etc.), but that's likely not easier. However you choose to do it, include enough interactivity to realize your design and make a useful app.

Documentation

Your app itself should provide enough information (widget labels, graph labels, legends, etc.) to allow the user to use it effectively and interpret the results. Pay attention to detail in the visualizations! I will deduct points for bad or unreadable labels, legends, colors, etc.

Beyond just the app, you should provide a written set of documentation that is accessible from the same link. Providing documentation on a separate tab is an obvious choice, but you may come up with other choices. Shiny apps have a document structure that allows tabbed pages. Your documentation should include the following elements:

- An introduction to the data set. Where did it come from? What population does it study? What does each variable represent? [Note: You need only describe the variables you're actually including with the app.]
- What questions/aspects of the data set motivated the creation of your app? For example, “to allow others to see the weather data” would not be specific enough, but “to allow viewers to see the variation in temperature over time, while also allowing them to see the correlation between temperature and other weather phenomena” would be much better.

- An explanation of how your app allows users to achieve the goals you've set out. This might involve a general explanation of how your visualization addresses the question, as well as any specific instructions necessary to run the app.
- Your Conclusions. You outlined questions that motivated your app in a previous section. Does a story emerge from the data? Are there particular conclusions or answers about your data set that you have discovered as a result of your work? What would you hope users will take away from their interaction with your app?
- Citations. Site the source of your data—provide links. If your work was based on any other sources, online or otherwise, cite those sources as well.

This documentation should be well-written, using a formal academic style. The structure of the paper will be fairly well defined because of the requirements above, but pay attention to organization within each section. Most sections should be (at least) several paragraphs long, which means that your writing should be detailed and not superficial.

Finally, your group should review all the elements of good visual design that we've talked about this semester as it prepares the data visualizations. These should be your best work, based on what you've learned!

Presentation

During the last week of classes, each group will make a presentation where they present their data set and app. Although this presentation need not be too formal, it should be well-practiced. Your presentation should cover all the points in the "Documentation" section above. If each group takes about 10–15 minutes, there will be plenty of time for questions.

It's not necessary to make slides if you feel your app itself stands on its own, but you may make slides if you want to. If you choose to make slides, do it in RStudio for practice!

Due Dates

- **Friday, November 10:** Your group has talked with me and chosen a data set.
- **Friday, November 17:** Peer review in class! Your group should have a short description of your data set and any important variables (about one paragraph), a statement of the question/questions the app will address, and a sketch of your group's idea for an app. Nothing fancy, probably hand-drawn. Bring these to class, where groups will trade and offer feedback.
- **Wednesday, November 29:** Your group has talked with me and shown me your progress on static graphs. You should have at least two main graphs in static form.
- **Wednesday, December 6–Friday, December 8:** Group presentations and completed apps due.