Data and Donuts
Data organization
Notes

Slide 1: Hi, and welcome to Data and Donuts. I'm Tobin Magle, the Data Management Specialist at the Morgan Library at Colorado State University. Have you ever tried looking at someone else's files to find a piece of information? Have you ever spent too much time trying to reformat data to make a table or a graph? Proper data organization can help with these problems.

Slide 2: To review, the research cycle generally goes as follows

- Come up with a hypothesis
- Design experiments, and hopefully write a data management plan
- Collect and analyze data
- And finally Publish these data, which can be used to generate new hypothesis

Data organization is initially important during the collection phase and through analysis, but will also make analysis archiving and sharing easier.

Slide 3: We're going to be talking about data organization in the context of 3 topics:

1. hierarchical organization
2. file naming, and
3. “tidy data” in spreadsheets.

Let’s start with hierarchical organization.

Slide 4: The simplest way to think of hierarchical organization is by thinking about nested folders in an operating system. Your computer's file structure is inherently hierarchical, or arranged in the order of rank. Or, put another way, the file system is organized as "folders inside folders". A good way to approach this is to separate your files by the most important attribute and further separate things inside those folders by less important attributes.

Slide 5: To organize research data, we recommend having one folder per project, and subfolders for distinct aspects of that project. For example, you could have folders for data, notes, protocols and manuscripts. Then within manuscripts, you could have folders for each paper, and then sections of that paper.

Slide 6: When thinking about how to organize your data, it’s a good idea to go back to the data inventory you created when writing your data management plan. This document describes the type of data you’re going to collect and lists any other research outputs. So, as you're looking through your data inventory, ask yourself

- What kinds of files do I have?
- How can I group these things?
- Of these attributes, what are the most important ones?

You'll base your folder structure on the answers to these questions.

Slide 7: To illustrate these points, let's look at a hypothetical grad student named Lou. He's trying to use data from someone who left the lab, who collected weight and cytokine data from 16 mice. Half of these mice were infected with a parasite.
Exercise 1: Now answer
* What are the attributes for this project?
* Which seem like the most important ones?
Pause the video and take a moment to answer these questions.

Slide 8: One attribute of this experiment is time, as the data are being collected longitudinally over the course of the year.

Slide 9: Another attribute is infection status, because some of the data are from infected mice, and some from uninfected

Slide 10: Additionally, we have two types of data here: weight and cytokine data. Are there any others I haven't identified? How would you rank the attributes we've identified? Which is the most important?

Slide 11:
Exercise 2:
Now that we've identified the relevant attributes, let's look at the data to decide their relative importance and organize Lou's files into a hierarchical structure.

Download the files from the link on the slide. Using the attributes you identified in the last exercise, create a file structure for Lou. If you have time, you can describe your strategy in Lou's README.txt file. For bonus points, think about your own project and decide how you should organize your files.

Demo 1:
To me, the most important attribute is probably data type. So in this case

- Make a folder for weight
- Add all of the weight files
- Make a folder for cytokines
- Add all the cytokine files
- Leave the README and the mouse inventory in the main folder.

Don't worry if you didn't choose the same attribute. If you thought time was most important, you could have made a folder for each month, and put the weight and cytokine data for each month in the same folder. This would be a better call if you were planning on plotting cytokine levels against weight, and not analyzing the same data types across months.

Slide 12: Another tool you can use to hierarchically organize your data is the open science framework. The OSF is a free, open source service of the Center for Open Science that was created as a public good to aid research. Instead of folders, OSF uses "components" to organize aspects of your project. Other OSF features include

- Add-ons, which allow you to link cloud services to your project
- contributors, which allow you to give others access, and
- a wiki, which lets you take notes on your project

Slide 13: OSF has its own organizational guidelines that you can read about at the link on this slide.

- First, they advocate consistency. Organizing all your projects in a similar way helps you find what you need.
- Second, only include one directory, or folder, per project. Too much content without context can get confusing
- Also, make components in your projects for raw data, processed data, code and output.
- Finally, include readme files that describes what is in each directory.

That said, these are only guidelines, and may not be appropriate for every project. You know your data better than anyone else, so trust your instincts.

Slide 14: Now, let's talk about components. One way to think of them is as "sub projects" or folders. Each component has separate privacy settings, wiki, add ons and files. Here are some examples of how to separate your content into components.

Demo 2:
In this first example,

- The project is for the MacDougall lab
- The components are for projects within that lab, like a barcoding project and a insect food web project, and a project about the effect of chocolate on grad student happiness.
- If we click on the **barcoding project**, we can see that there are sub-sub components for Photos, lab notebooks, and data collection sheets.
- The insect food web project has components for identification keys and field samples

Demo 3:
This next example is for a clinical research project. Thus, the components are for standard parts of a clinical trial, such as clinical protocol, consent forms, IRB forms and analysis. Notice that there is no data component because of privacy concerns. None of these components have sub components.

Demo 4:
This example is a project associated with a manuscript.

- You can see that a link in the wiki takes you to the full manuscript within OSF
- Each component represents a manuscript section, like Analysis scripts and output, data, tables, and figures.

Demo 5:
This last example is a project for a large collaborative effort by many contributors to replicate psychological research. The components are fairly eclectic, with sections for

analysis, resources, presentations, commends, and papers that they are trying to replicate. Each of these study component has subcomponents for
Demo5

Slide 15: Let's go into Open Science Framework to create a project.

Demo 5:

- Create a new OSF project: Lou's data
- Add components
  - Weight data
  - Cytokine data
- Add postdoc files to the component
  - Upload data to OSF storage

Slide 16: If you're starting to be overwhelmed by the possibilities, don't worry. There's no one right answer. The fact that you're thinking about it is putting you ahead of the curve. Make sure to be consistent and document what you're doing in README files.

Slide 17: Even if you have good folder organization, bad file naming can make your data hard to use and interpret. The goal of file naming best practices is to make the file names human and machine interpretable.

Slide 18: The first rule of good file naming is to use descriptive names. These details help human readability because the name says something about the content.

- Bad file names like file.txt are not recommended, even if they're in a well-described folder, because they lose this information if they're moved.
- Better names include collection dates and some text to indicate the contents of the file. The "ok" file name, tells me the dates collected and that the data come from mice.
- The best file names, however, get really specific about the contents and format of the file. The "good" file name tells me the date collected, and that the file contains mouse weights in tab separated format.

Slide 19: The second rule is to name files from general to specific. Let's use the example of gene expression experiments that have several replicates.

- First, separate out dates and content. Within the date, put the year first, then month, then day, from general to specific. This allows the computer to sort the files in a meaningful way.
- Then let's look at the content. The replicate number is more specific than the type of data, so it should go last. That way the sorting would be meaningful if there was non-gene expression data in the folder.

Slide 20: The third rule for file naming is to avoid abbreviations, because your abbreviations might not be intuitive to other people. Unless they are widely known, avoid them completely. Another option is to include a detailed readme file for what your abbreviations mean.

Slide 21: The fourth rule for file naming is to avoid spaces. Spaces are a bad idea because computers use spaces to separate out file names, so typing a file name with a space in it in the unix terminal will likely make the computer think there's two different files.

- Alternatives are dashes, underscores, or
- camel case, which is capitalization of each word in the file name, to delineate words and make it more human readable.

Slide 22: The fifth rule for file naming is to avoid special characters, like the ones listed on this slide. They are best avoided because certain programming languages have special meanings associated with these symbols.

- For example the tilde tells the unix shell to return to the home directory.
- Alternatives to these symbols are underscores and dashes, as with the space.

Slide 23: The final and most important rule is to be consistent with your file naming conventions. This will allow your data to be more findable. Even better, try to establish common conventions for your research group so you can work together better.

Slide 24: You probably aren't starting from scratch on your research project, and already have files that are inconsistently named. Luckily, there are programs out there that will automate file renaming. Let's look at PSRenamer.

Slide 25:
Exercise 3: Let's get back to helping Lou. Now that you've separated Lou's files into folders, rename his files with descriptive names that adhere to best practices described here.

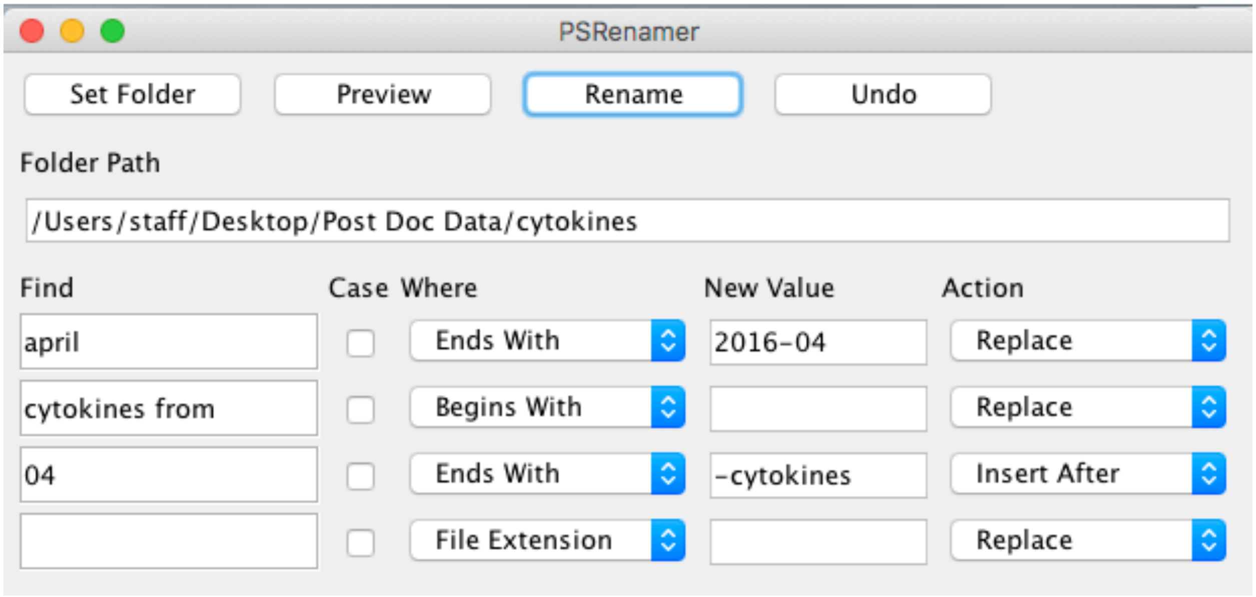
Let's come up with a general strategy to name Lou's files

- Let's start with weight
- Start with the most general: the date
  - YYYY_MM_DD
- Then what's in the file: mouse_weights
- Make sure to use _ instead of spaces.

Slide 26: Since you probably aren't starting from scratch in your research project, you already have files that are inconsistently named. Luckily, there are programs out there that automate file renaming. Let's look at PS renamer.

Demo 6: PSrenamer

- Set folder to cytokines
- Settings to rename the April File

- 

Folder Path

/Users/staff/Desktop/Post Doc Data/cytokines

Find	Case	Where	New Value	Action
april	☐	Ends With	2016-04	Replace
cytokines from	☐	Begins With		Replace
04	☐	Ends With	-cytokines	Insert After
	☐	File Extension		Replace

- Then the file extension
- Use underscores instead of spaces
- The same strategy will work for weight

Slide 26: Now that our files are in folder and named properly, let's work on the interior of the files, or how to organize your data efficiently in spreadsheets.

Slide 27: Many people use their spreadsheet programs like a lab notebook. In addition to data, these spreadsheets contain color coding and formatting that often contain information about the data. They also have notes, calculations, graphs and tables. This is a very human readable and intuitive way to use a spreadsheet.

Slide 28: However, this type of organization has its downsides.

- For one, computers are dumb, so they won't understand the meaning of notes or color coding or formatting
- Also, making graphs and tables in spreadsheet programs is inefficient and can be done quicker using scripts
- Thus, taking the time to use spreadsheets wisely will save you time in the long run

Slide 29: To use spreadsheets wisely, here are some tips

- **One table per sheet** - computers can't tell where one ends and another begins
- **Don't use multiple sheets** - computers only read one at a time
- **Use descriptive variable names** - so it's easy to understand what they mean
- **Don't mix notes and data** - not machine readable

Slide 30: Even if you follow all those recommendations, there's still one more step to make your data machine readable: tidy it up. There are really only 2 rules to making your data tidy:

1. Each column is a variable: Make sure not to combine multiple variables in one column
2. Each row is an observation, or one measured value.

Slide 31:
Exercise 4: Let's look back at Lou's spreadsheets to see how tidy the data are.

1. Open MouseInventory.xls.
   - Is he using spreadsheets wisely?
   - Is each column a variable?
2. Now look at the January data for both weight and cytokines.
   - What variables are being measured?
   - What are the observations?

4.1: In the mouse inventory file, we have one sheet, which is good, but there is some room for improvement:

- The highlighting in the Mouse column indicates that there is a comment in that row. To fix this, we can remove the highlighting and move the comment to a new column called "Comment"
- Additionally, the location column seems to have 2 distinct variables: Rack and Cage. So to fix this, we can separate location into Rack and Cage Columns.
- Finally, the Birth date column is formatted improperly. We can change the format to DD-MM-YYYY. Then, you can split the column into Year, Month, and Day Columns. And Reorder them

4.2: In the cytokines file, you can see that we have 2 tabs, one with data from infected mice, one with data from uninfected mice. Each row is a mouse, and each column is a cytokine. There's also some color coding. Let's fix it so that these two sheets are combined into one tidy data set

- Make a new tab called tidy so that the original data remain intact.
- Think about what variables are being recorded: Mouse #, cytokine, cytokine type (pro- or anti-inflammatory) and the value you're measuring.
- That makes each time you measure one cytokine in one mouse an observation
- The way the data are stored with cytokine across the top is called "wide" data. The description above is called "long" data. There are advantages to either depending on what you're going to do with the data, but long data tends to be more versatile for machine readability.
- So, instead of having cytokines across the top, we can create a column called "cytokine" to record the cytokine name, and one called value to hold the number.
  - Copy, paste special -> transpose for the cytokines in the cytokine names colum (8x)
  - Paste transpose each mouse (6x, one for each cytokine)
- Instead of having color coding for pro and anti-inflammatory, add a new column called Cytokine type.

- Then, to combine the tabs, add a new column called infection_status to differentiate

4.3: Weight

- Here we have Days as rows, and mouse number across the top.
- Though there is only 1 tab, we have two separate tables here: one for infected, one for uninfected.
- We can fix this by having a column for date (or one each for month, day, and year), A column for mouse number and a column for infection_status, and one for comments
- The treated Average column is derived, so it's better to delete it
- Also the graph should not be in the table. Either move it to its own tab, or better yet, learn R and write a script to make the graph!

Slide 32: Let's look at an example from the Data Carpentry Ecology Lesson. These are data from a field study that were collected by 2 separate field techs over 2 years. How can we combine all of these data tables into 1 large table?

Exercise 5:

- **Variables**: year, month, day, plot, sex, species, weight, comments
- **Observations**: variables above for each animal they trapped

Make a new tab. Put the variables as columns. Copy/paste data.

Slide 33: Now let's look at some examples of real data posted to FigShare, which houses real shared data. The first is supplemental data from a manuscript. This record contains high performance liquid chromatography data, which separates out molecules in solution to get an idea of how big they are. Primarily, they are measuring retention time, which is how long a molecule stays on the column. Let's look at how these data are organized.

Demo 7: https://figshare.com/articles/Supplemental_data_1_xls/4055544

- Rt = Retention time
- + = presence of a peak
- - = absence of a peak
- **Variables**: Sample Code, Species, Sampling area, Altitude, Lat, Lon, Retention time
- **Observation**, whether there is a peak (+/-) at the combination of the variables above
- Lots of "extra" data. Could assume that no data = absence of a peak.
- Make new tab called tidy
- Put variables across the top
    - Copy A1 – A5
    - Paste transpose
    - Add retention time

- Instead of having a series of +/- for each combo of variables and each retention time, make a row for where each + sign is
  - Each combo of variables can have more than one record.
  - Hard to do by hand: scan for + , add records manually
  - Easier to do in R.
  - Results in way fewer data points -> efficient storage.

Slide 34: This next dataset contains cytokine data from a CCK-8 assay and ELISA. It's hard to say what all of these data mean, because the metadata are insufficient, but we'll do our best.

Demo 8: https://figshare.com/articles/cck8_xls/3505772

- First, you can see they have multiple sheets, but sheets 2 and 3 are empty, so they can be deleted
- It also looks like they have 2 separate assays (CCK-8 and ELISA). I would make a separate sheet for each assay.
- Let's start with the CCK8 data:
  - **Variables**: Concentration (uC/ml), time, treatment, value
  - **Observations**: value at a certain treatment, concentration and time
- For ELISA:
  - **Variables**: Cytokine, treatment, value
  - **Observation**: level of a specified cytokine given the treatment

Slide 35: Thanks for listening. I hope you found this data organization session to be helpful. Please email me at the address on this slide if you need help. Also, check out our data management pages for more information. If you want to learn more about data organization and automation, check out the lessons from Data and Software carpentry. I have used both of these lessons for inspiration for Data and Donuts.