[Week 4 - R Markdown-Scientific Thinking and Big Data 3](#_Toc31832717)

[During this final module, you'll learn to use R Markdown and get an introduction to three concepts that are incredibly important to every successful data scientist: asking good questions, experimental design, and big data. 3](#_Toc31832718)

[W4 1 - R Markdown 3](#_Toc31832719)

[What is R Markdown? 3](#_Toc31832720)

[Why use R Markdown? 3](#_Toc31832721)

[Installation 4](#_Toc31832722)

[Getting started with R Markdown 4](#_Toc31832723)

[“Knitting” documents 6](#_Toc31832724)

[What are some easy Markdown commands? 7](#_Toc31832725)

[Summary 8](#_Toc31832726)

[W4 2 - Types of data science questions 10](#_Toc31832727)

[The main divisions of data science questions 10](#_Toc31832728)

[1. Descriptive analysis 10](#_Toc31832729)

[2. Exploratory analysis 11](#_Toc31832730)

[3. Inferential analysis 12](#_Toc31832731)

[4. Predictive analysis 13](#_Toc31832732)

[5. Causal analysis 14](#_Toc31832733)

[6. Mechanistic analysis 15](#_Toc31832734)

[Summary 15](#_Toc31832735)

[W4 3 - Experimental Design 15](#_Toc31832736)

[What does experimental design mean? 15](#_Toc31832737)

[Why should you care? 16](#_Toc31832738)

[Principles of experimental design 17](#_Toc31832739)

[Sharing data 23](#_Toc31832740)

[Beware p-hacking! 23](#_Toc31832741)

[Summary 24](#_Toc31832742)

[W4 4 - Big Data 24](#_Toc31832743)

[What is big data? 24](#_Toc31832744)

[What is structured data? What is unstructured data? 25](#_Toc31832745)

[Challenges of working with big data 26](#_Toc31832746)

[Benefits to working with big data 27](#_Toc31832747)

[Will big data solve all our problems? 28](#_Toc31832748)

[Summary 28](#_Toc31832749)

# Week 4 - R Markdown-Scientific Thinking and Big Data

During this final module, you'll learn to use R Markdown and get an introduction to three concepts that are incredibly important to every successful data scientist: asking good questions, experimental design, and big data.

Key Concepts

* Use R Markdown
* Propose data science questions
* Discuss experimental design and its importance for data science
* Summarize the qualities of big data

## W4 1 - R Markdown

We’ve spent a lot of time getting R and RStudio working, learning about projects and version control - you are practically an expert at this! There is one last major functionality of R/RStudio that we would be remiss to not include in your introduction to R - [Markdown!](http://rmarkdown.rstudio.com/)

### What is R Markdown?

R Markdown is a way of creating fully reproducible documents, in which both text and code can be combined. In fact, these lessons are written using R Markdown! That’s how we make things:

* bullets
* **bold**
* italics
* [links](https://en.wikipedia.org/wiki/Rickrolling)
* or run inline r code

And by the end of this lesson, you should be able to do each of those things too, and more!

Despite these documents all starting as plain text, you can render them into HTML pages, or PDFs, or Word documents, or slides! The symbols you use to signal, for example, **bold** or italics is compatible with all of those formats.

### Why use R Markdown?

One of the main benefits is the reproducibility of using R Markdown. Since you can easily combine text and code chunks in one document, you can easily integrate introductions, hypotheses, your code that you are running, the results of that code and your conclusions all in one document. Sharing what you did, why you did it and how it turned out becomes so simple - and that person you share it with can re-run your code and get the exact same answers you got. That’s what we mean about reproducibility. But also, sometimes you will be working on a project that takes many weeks to complete; you want to be able to see what you did a long time ago (and perhaps be reminded exactly why you were doing this) and you can see exactly what you ran AND the results of that code - and R Markdown documents allow you to do that.

Another major benefit to R Markdown is that since it is plain text, it works very well with version control systems. It is easy to track what character changes occur between commits; unlike other formats that aren’t plain text. For example, in one version of this lesson, I may have forgotten to bold **this** word. When I catch my mistake, I can make the plain text changes to signal I would like that word bolded, and in the commit, you can see the exact character changes that occurred to now make the word bold.

Check out [this video](https://vimeo.com/178485416) that the RStudio developers have released about R Markdown and what it is!

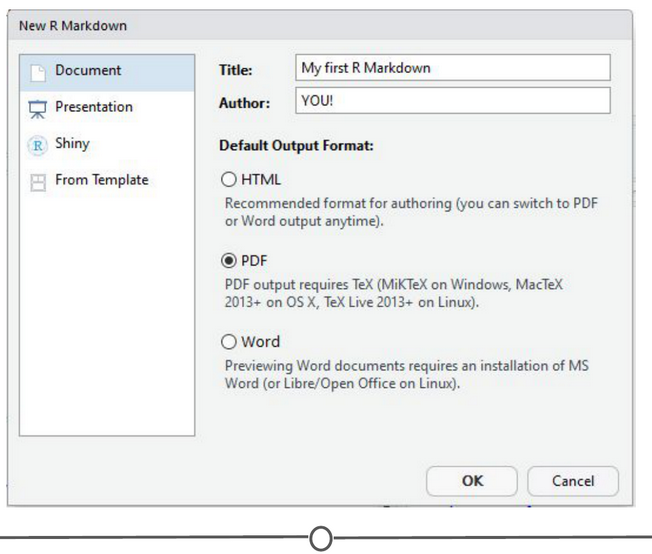
### Installation

Another (selfish) benefit of R Markdown is how easy it is to use! Like everything in R, this extended functionality comes from an R package - “rmarkdown.” All you need to do to install it is run install.packages("rmarkdown")

And that’s it, you are ready to go.

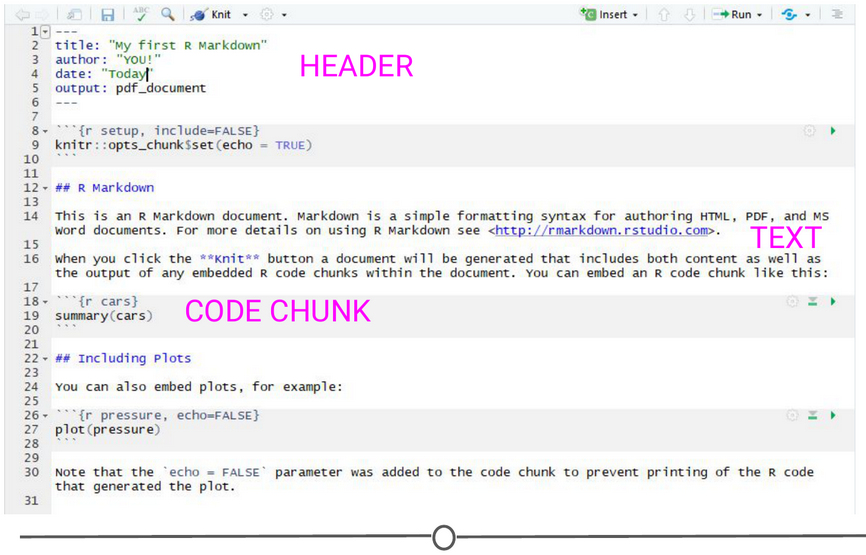
### Getting started with R Markdown

To create an R Markdown document, in R Studio, go to File > New File > R Markdown. You will be presented with the following window:



**Initiating an R Markdown file**

I’ve filled in a title and an author and switched the output format to a PDF. Explore around this window and the tabs along the left to see all the different formats that you can output to. When you are done, click OK, and a new window should open with a little explanation on R Markdown files.



**The default template for R Markdown files**

There are three main sections of an R Markdown document. The first is the **header** at the top, bounded by the three dashes. This is where you can specify details like the title, your name, the date, and what kind of document you want output. If you filled in the blanks in the window earlier, these should be filled out for you.

Also on this page, you can see **text sections**, for example, one section starts with “## R Markdown” - We’ll talk more about what this means in a second, but this section will render as text when you produce the PDF of this file - and all of the formatting you will learn generally applies to this section.

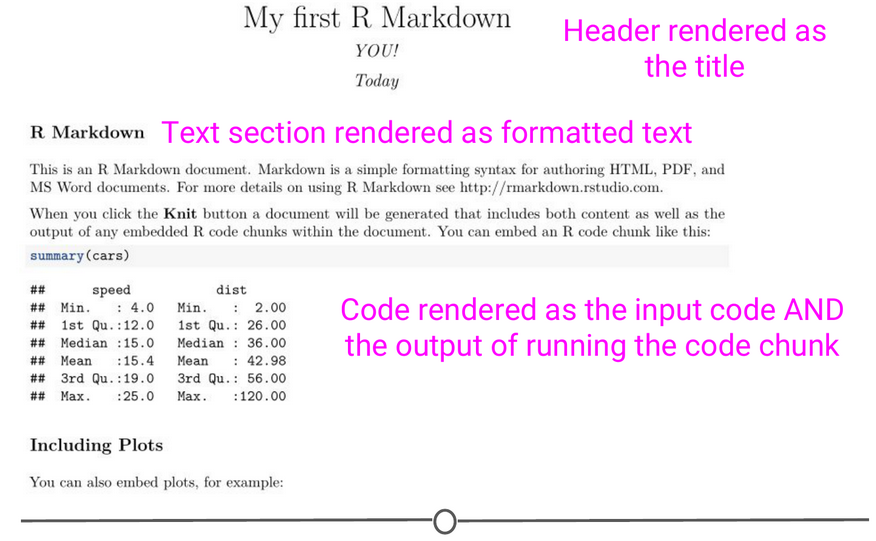
And finally, you will see **code chunks**. These are bounded by the triple backticks. These are pieces of R code (“chunks”) that you can run right from within your document - and the output of this code will be included in the PDF when you create it.

The easiest way to see how each of these sections behave is to produce the PDF!

### “Knitting” documents

When you are done with a document, in R Markdown, you are said to **“knit”** your plain text and code into your final document. To do so, click on the “Knit” button along the top of the source panel. When you do so, it will prompt you to save the document as an RMD file. Do so.

You should see a document like this:



**The rendered PDF you created by knitting your markdown file**

So here you can see that the content of the header was rendered into a title, followed by your name and the date. The text chunks produced a section header called “R Markdown” which is followed by two paragraphs of text. Following this, you can see the R code summary(cars), importantly, followed by the output of running that code. And further down you will see code that ran to produce a plot, and then that plot. This is one of the huge benefits of R Markdown - rendering the results to code inline.

Go back to the R Markdown file that produced this PDF and see if you can see how you signify you want text bolded. (Hint: Look at the word “Knit” and see what it is surrounded by).

### What are some easy Markdown commands?

At this point, I hope we’ve convinced you that R Markdown is a useful way to keep your code/data and have set you up to be able to play around with it. To get you started, we’ll practice some of the formatting that is inherent to R Markdown documents.

To start, let’s look at bolding and italicising text. To bold text, you surround it by two asterisks on either side. Similarly, to italicise text, you surround the word with a single asterisk on either side. \*\*bold\*\* and \*italics\* respectively.

We’ve also seen from the default document that you can make section headers. To do this, you put a series of hash marks (#). The number of hash marks determines what level of heading it is. One hash is the highest level and will make the largest text (see the first line of this lecture), two hashes is the next highest level and so on. Play around with this formatting and make a series of headers, like so:

# Header level 1  
## Header level 2  
### Header level 3...

The other thing we’ve seen so far is code chunks. To make an R code chunk, you can type the three backticks, followed by the curly brackets surrounding a lower case R, put your code on a new line and end the chunk with three more backticks. Thankfully, RStudio recognized you’d be doing this a lot and there are short cuts, namely Ctrl+Alt+I (Windows) or Cmd + Option + I (Mac). Additionally, along the top of the source quadrant, there is the “Insert” button, that will also produce an empty code chunk. Try making an empty code chunk. Inside it, type the code print("Hello world"). When you knit your document, you will see this code chunk and the (admittedly simplistic) output of that chunk.

If you aren’t ready to knit your document yet, but want to see the output of your code, select the line of code you want to run and use Ctrl+Enter or hit the “Run” button along the top of your source window. The text “Hello world” should be output in your console window. If you have multiple lines of code in a chunk and you want to run them all in one go, you can run the entire chunk by using Ctrl+Shift+Enter OR hitting the green arrow button on the right side of the chunk OR going to the Run menu and selecting Run current chunk.

One final thing we will go into detail on is making bulleted lists, like the one at the top of this lesson. Lists are easily created by preceding each prospective bullet point by a single dash, followed by a space. Importantly, at the end of each bullet’s line, end with TWO spaces. This is a quirk of R Markdown that will cause spacing problems if not included.

* Try
* Making
* Your
* Own
* Bullet
* List!

This is a great starting point and there is so much more you can do with R Markdown. Thankfully, RStudio developers have produced an [“R Markdown cheatsheet”](http://www.rstudio.com/wp-content/uploads/2016/03/rmarkdown-cheatsheet-2.0.pdf) that we urge you to go check out and see everything you can do with R Markdown! The sky is the limit!

### Summary

In this lesson we’ve delved into R Markdown, starting with what it is and why you might want to use it. We hopefully got you started with R Markdown, first by installing it, and then by generating and knitting our first R Markdown document. We then looked at some of the various formatting options available to you and practiced generating code and running it within the R Studio interface.

## W4 2 - Types of data science questions

In this lesson, we’re going to be a little more conceptual and look at some of the types of analyses data scientists employ to answer questions in data science.

### The main divisions of data science questions

There are, broadly speaking, six categories in which data analyses fall. In the approximate order of difficulty, they are:

1. Descriptive
2. Exploratory
3. Inferential
4. Predictive
5. Causal
6. Mechanistic

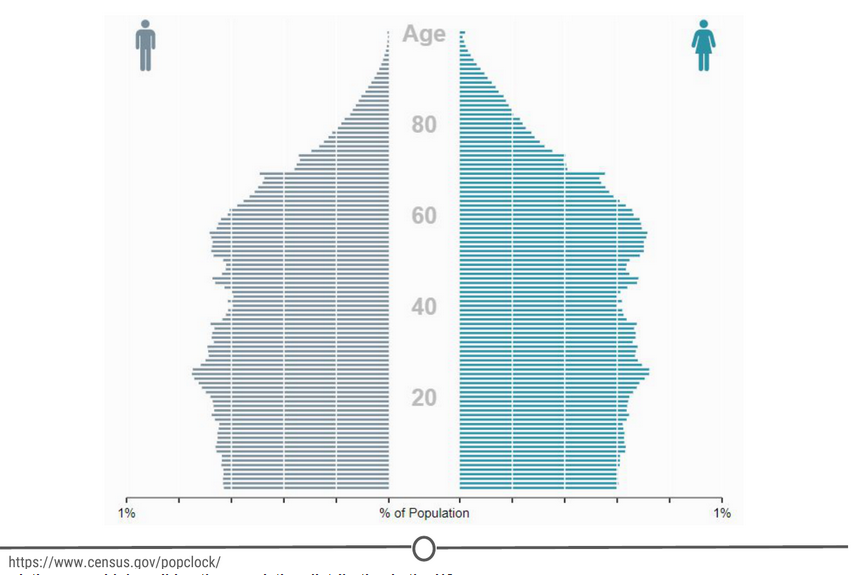
Let’s explore the goals of each of these types and look at some examples of each analysis!

### 1. Descriptive analysis

The goal of descriptive analysis is to **describe** or **summarize** a set of data. Whenever you get a new dataset to examine, this is usually the first kind of analysis you will perform. Descriptive analysis will generate simple summaries about the samples and their measurements. You may be familiar with common descriptive statistics: measures of central tendency (eg: mean, median, mode) or measures of variability (eg: range, standard deviations or variance).

This type of analysis is aimed at summarizing your sample – not for generalizing the results of the analysis to a larger population or trying to make conclusions. Description of data is separated from making interpretations; generalizations and interpretations require additional statistical steps.

Some examples of purely descriptive analysis can be seen in censuses. Here, the government collects a series of measurements on all of the country’s citizens, which can then be summarized. Here, you are being shown the age distribution in the US, stratified by sex. The goal of this is just to describe the distribution. There is no inferences about what this means or predictions on how the data might trend in the future. It is just to show you a summary of the data collected.



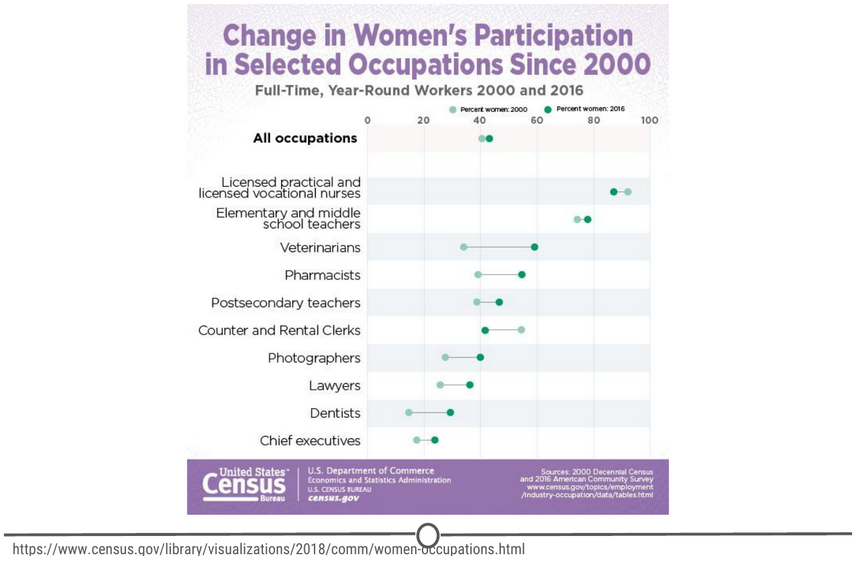
**A population pyramid describing the population distribution in the US**

### 2. Exploratory analysis

The goal of exploratory analysis is to examine or **explore** the data and find **relationships** that weren’t previously known. Exploratory analyses explore how different measures might be related to each other but do not confirm that relationship as causitive. You’ve probably heard the phrase “Correlation does not imply causation” and exploratory analyses lie at the root of this saying. Just because you observe a relationship between two variables during exploratory analysis, it does not mean that one necessarily causes the other.

Because of this, exploratory analyses, while useful for discovering new connections, should not be the final say in answering a question! It can allow you to formulate hypotheses and drive the design of future studies and data collection, but exploratory analysis alone should never be used as the final say on why or how data might be related to each other.

Going back to the census example from above, rather than just summarizing the data points within a single variable, we can look at how two or more variables might be related to each other. In the plot below, we can see the percent of the workforce that is made up of women in various sectors and how that has changed between 2000 and 2016. Exploring this data, we can see quite a few relationships. Looking just at the top row of the data, we can see that women make up a vast majority of nurses and that it has slightly decreased in 16 years. While these are interesting relationships to note, the causes of these relationships is not apparent from this analysis. All exploratory analysis can tell us is that a relationship exists, not the cause.



**Exploring the relationships between the percentage of women in the workforce in various sectors between 2000 and 2016**

### 3. Inferential analysis

The goal of inferential analyses is to use a relatively **small sample** of data to **infer** or say something about the **population** at large. Inferential analysis is commonly the goal of statistical modelling, where you have a small amount of information to extrapolate and generalize that information to a larger group.

Inferential analysis typically involves using the data you have to estimate that value in the population and then give a measure of your uncertainty about your estimate. Since you are moving from a small amount of data and trying to generalize to a larger population, your ability to accurately infer information about the larger population depends heavily on your sampling scheme - if the data you collect is not from a representative sample of the population, the generalizations you infer won’t be accurate for the population.

Unlike in our previous examples, we shouldn’t be using census data in inferential analysis - a census already collects information on (functionally) the entire population, there is nobody left to infer to; and inferring data from the US census to another country would not be a good idea because the US isn’t necessarily representative of another country that we are trying to infer knowledge about. Instead, a better example of inferential analysis is [a study](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3521092/) in which a subset of the US population was assayed for their life expectancy given the level of air pollution they experienced. This study uses the data they collected from a sample of the US population to infer how air pollution might be impacting life expectancy in the entire US.

### 4. Predictive analysis

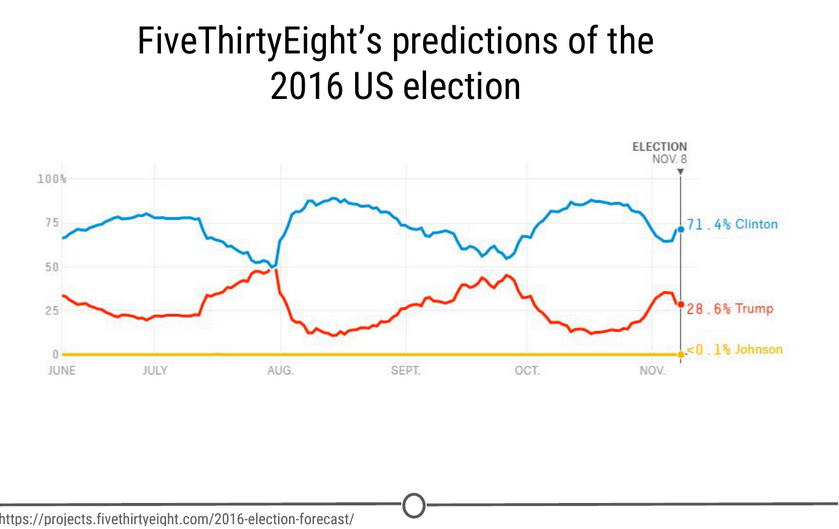
The goal of predictive analysis is to use **current** data to make **predictions** about **future** data. Essentially, you are using current and historical data to find patterns and predict the likelihood of future outcomes.

Like in inferential analysis, your accuracy in predictions is dependent on measuring the right variables. If you aren’t measuring the right variables to predict an outcome, your predictions aren’t going to be accurate. Additionally, there are many ways to build up prediction models with some being better or worse for specific cases, but in general, having more data and a simple model generally performs well at predicting future outcomes.

All this being said, much like in exploratory analysis, just because one variable may predict another, it does not mean that one causes the other; you are just capitalizing on this observed relationship to predict the second variable.

A common saying is that prediction is hard, especially about the future. There aren’t easy ways to gauge how well you are going to predict an event until that event has come to pass; so evaluating different approaches or models is a challenge.

We spend a lot of time trying to predict things - the upcoming weather, the outcomes of sports events, and in the example we’ll explore here, the outcomes of elections. We’ve previously mentioned Nate Silver of [FiveThirtyEight](http://fivethirtyeight.com/), where they try and predict the outcomes of U.S. elections (and sports matches, too!). Using historical polling data and trends and current polling, FiveThirtyEight builds models to predict the outcomes in the next US Presidential vote - and has been fairly accurate at doing so! FiveThirtyEight’s models accurately predicted the 2008 and 2012 elections and was widely considered an outlier in the 2016 US elections, as it was one of the few models to suggest Donald Trump at having a chance of winning.



**FiveThirtyEight’s predictions over time for the winner of the US 2016 election**

### 5. Causal analysis

The caveat to a lot of the analyses we’ve looked at so far is that we can only see correlations and can’t get at the cause of the relationships we observe. Causal analysis fills that gap; the goal of causal analysis is to see what happens to one variable when we manipulate another variable - looking at the **cause** and **effect** of a **relationship**.

Generally, causal analyses are fairly complicated to do with observed data alone; there will always be questions as to whether it is correlation driving your conclusions or that the assumptions underlying your analysis are valid. More often, causal analyses are applied to the results of randomized studies that were designed to identify causation. Causal analysis is often considered the gold standard in data analysis, and is seen frequently in scientific studies where scientists are trying to identify the cause of a phenomenon, but often getting appropriate data for doing a causal analysis is a challenge.

One thing to note about causal analysis is that the data is usually analysed in aggregate and observed relationships are usually average effects; so, while on average giving a certain population a drug may alleviate the symptoms of a disease, this causal relationship may not hold true for every single affected individual.

As we’ve said, many scientific studies allow for causal analyses. Randomized control trials for drugs are a prime example of this. For example, [one randomized control trial](http://www.nejm.org/doi/full/10.1056/NEJMoa1702752) examined the effects of a new drug on treating infants with spinal muscular atrophy. Comparing a sample of infants receiving the drug versus a sample receiving a mock control, they measure various clinical outcomes in the babies and look at how the drug affects the outcomes.

### 6. Mechanistic analysis

Mechanistic analyses are not nearly as commonly used as the previous analyses - the goal of mechanistic analysis is to understand the **exact changes in variables** that lead to **exact changes in other variables**. These analyses are exceedingly hard to use to infer much, except in simple situations or in those that are nicely modeled by deterministic equations. Given this description, it might be clear to see how mechanistic analyses are most commonly applied to physical or engineering sciences; biological sciences, for example, are far too noisy of data sets to use mechanistic analysis. Often, when these analyses are applied, the only noise in the data is measurement error, which can be accounted for.

You can generally find examples of mechanistic analysis in material science experiments. [Here](https://www.sciencedirect.com/science/article/pii/S0142941817303422), we have a study on biocomposites (essentially, making biodegradable plastics) that was examining how biocarbon particle size, functional polymer type and concentration affected mechanical properties of the resulting “plastic.” They are able to do mechanistic analyses through a careful balance of controlling and manipulating variables with very accurate measures of both those variables and the desired outcome.

### Summary

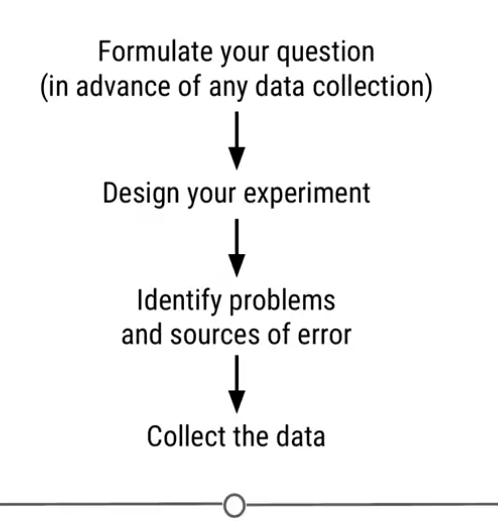
In this lesson we’ve covered the various types of data analysis, their goals, and looked at a few examples of each to demonstrate what each analysis is capable of (and importantly, what it is not).

## W4 3 - Experimental Design

Now that we’ve looked at the different types of data science questions, we are going to spend some time looking at experimental design concepts. As a data scientist, you are a scientist and as such, need to have the ability to design proper experiments to best answer your data science questions!

### What does experimental design mean?

Experimental design is organizing an experiment so that you have the correct data (and enough of it!) to clearly and effectively answer your data science question. This process involves clearly formulating your question in advance of any data collection, designing the best set-up possible to gather the data to answer your question, identifying problems or sources of error in your design, and only then, collecting the appropriate data.

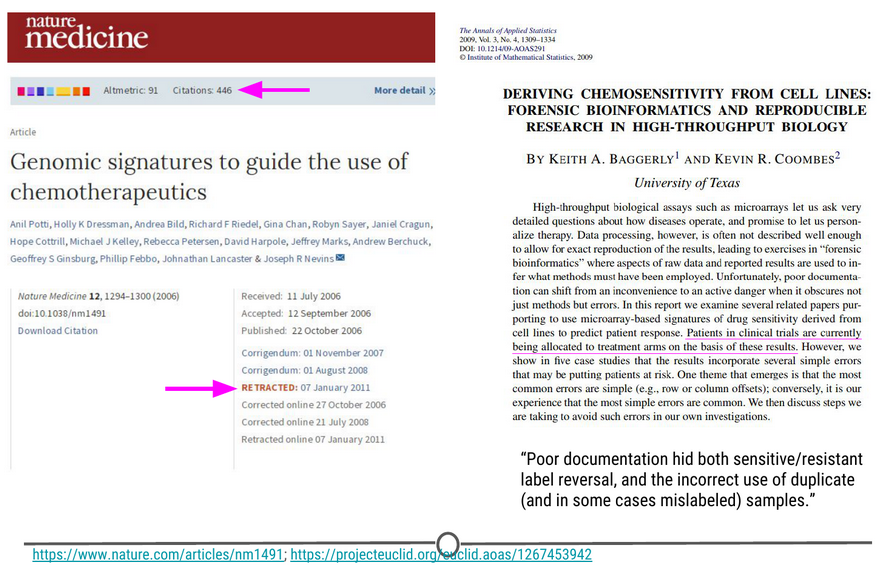


### Why should you care?

Going into an analysis, you need to have a plan in advance of what you are going to do and how you are going to analyse the data. If you do the wrong analysis, you can come to the wrong conclusions!

We’ve seen many examples of this exact scenario play out in the scientific community over the years - there’s an entire website, [Retraction Watch](https://retractionwatch.com/), dedicated to identifying papers that have been retracted, or removed from the literature, as a result of poor scientific practices. And sometimes, those poor practices are a result of poor experimental design and analysis.

Occasionally, these erroneous conclusions can have sweeping effects; particularly in the field of human health. For example, [here](https://www.nature.com/articles/nm1491) we have a paper that was trying to predict the effects of a person’s genome on their response to different chemotherapies, to guide which patient receives which drugs to best treat their cancer. As you can see, this paper was retracted, over 4 years after it was initially published. In that time, this data, which was later shown to have numerous problems in their set-up and cleaning, was cited in nearly 450 other papers that may have used these erroneous results to bolster their own research plans. On top of this, this wrongly analysed data was used in clinical trials to determine cancer patient treatment plans. When the stakes are this high, experimental design is paramount.



**A retracted paper and the forensic analysis of what went wrong**

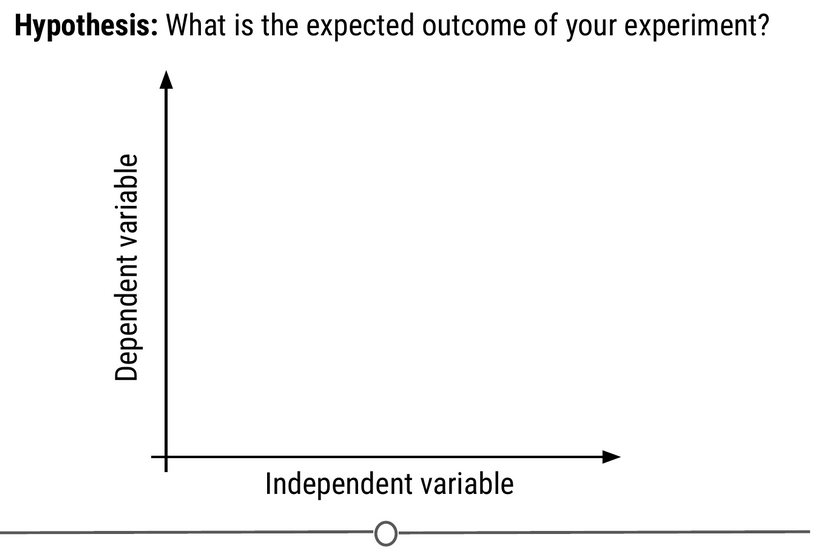
### Principles of experimental design

There are a lot of concepts and terms inherent to experimental design. Let’s go over some of these now!

**Independent variable (AKA factor):** The variable that the experimenter manipulates; it does not depend on other variables being measured. Often displayed on the x-axis.

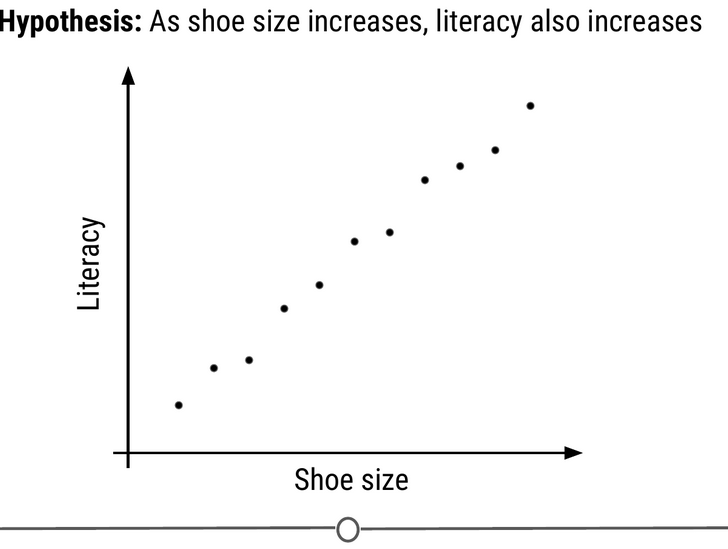
**Dependent variable:** The variable that is expected to change as a result of changes in the independent variable. Often displayed on the y-axis, so that changes in X, the independent variable, effect changes in Y.

So when you are designing an experiment, you have to decide what variables you will measure, and which you will manipulate to effect changes in other measured variables. Additionally, you must develop your **hypothesis**, essentially an educated guess as to the relationship between your variables and the outcome of your experiment.

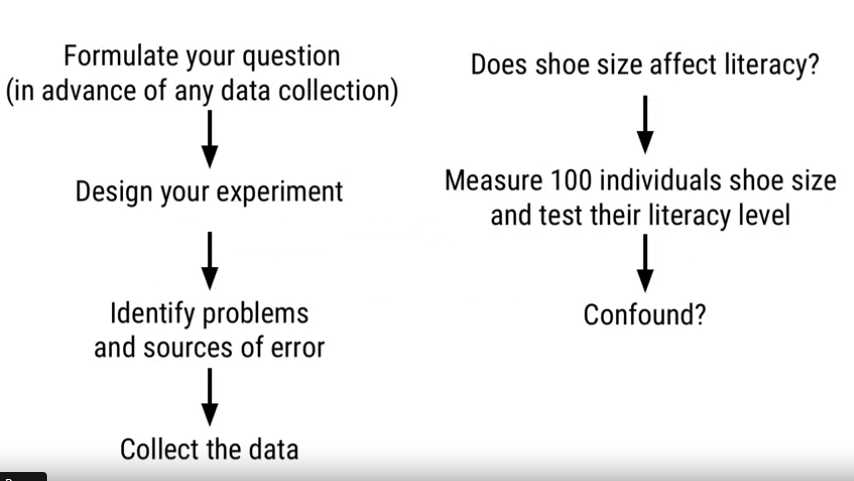


**How hypotheses, independent, and dependent variables are related to each other**

Let’s do an example experiment now! Let’s say for example that I have a hypothesis that as shoe size increases, literacy also increases. In this case, designing my experiment, I would choose a measure of literacy (eg: reading fluency) as my variable that depends on an individual’s shoe size.



**My experimental set-up: I hypothesize that literacy level depends on shoe size**

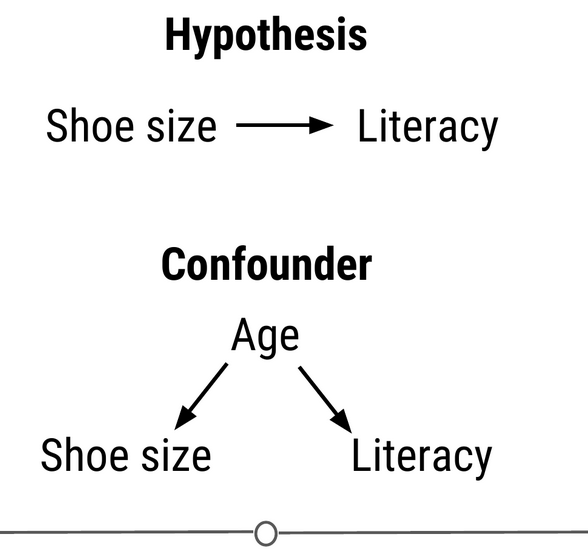


To answer this question, I will design an experiment in which I measure the shoe size and literacy level of 100 individuals. **Sample size** is the number of experimental subjects you will include in your experiment. There are ways to pick an optimal sample size, that you will cover in later courses. Before I collect my data though, I need to consider if there are problems with this experiment that might cause an erroneous result. In this case, my experiment may be fatally flawed by a **confounder**.

**Confounder:** An extraneous variable that may affect the relationship between the dependent and independent variables.

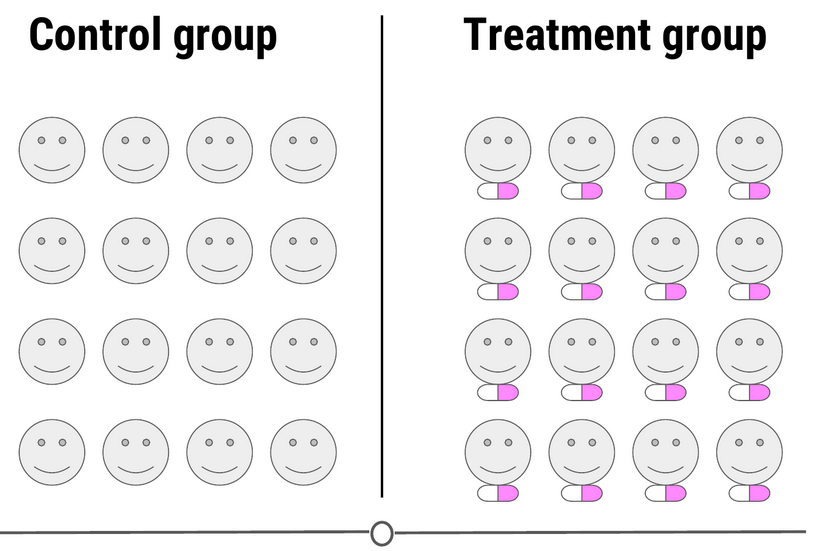
In our example, since age affects foot size and literacy is affected by age, if we see any relationship between shoe size and literacy, the relationship may actually be due to age – age is “confounding” our experimental design!

To **control** for this, we can make sure we also measure the age of each individual so that we can take into account the effects of age on literacy, as well. Another way we could **control** for age’s effect on literacy would be to **fix** the age of all participants. If everyone we study is the same age, then we have removed the possible effect of age on literacy.



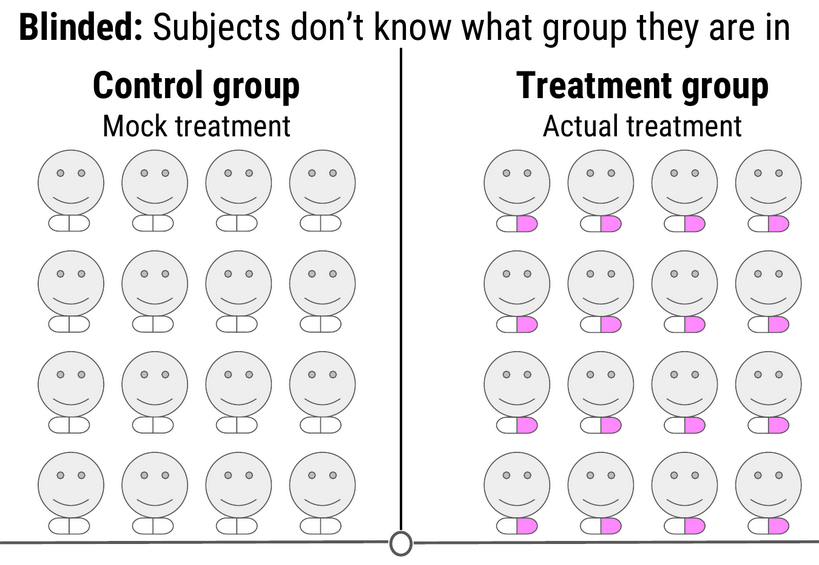
**Age is confounding my experimental design! We need to control for this**

In other experimental design paradigms, a **control group** may be appropriate. This is when you have a group of experimental subjects that are not manipulated. So if you were studying the effect of a drug on survival, you would have a group that received the drug (**treatment**) and a group that did not (**control**). This way, you can compare the effects of the drug in the treatment versus control group.



**A control group is a group of subjects that do not receive the treatment, but still have their dependent variables measured**

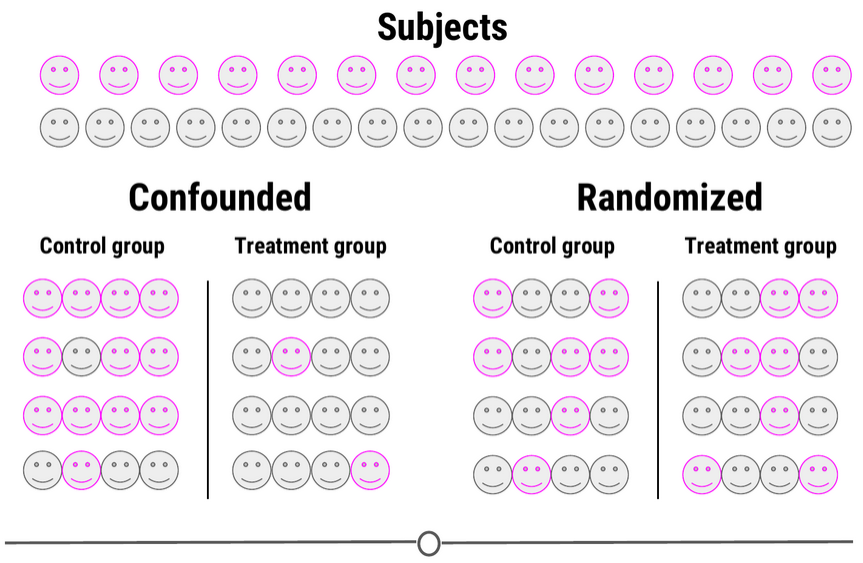
In these study designs, there are other strategies we can use to control for confounding effects. One, we can **blind** the subjects to their assigned treatment group. Sometimes, when a subject knows that they are in the treatment group (eg: receiving the experimental drug), they can feel better, not from the drug itself, but from knowing they are receiving treatment. This is known as the **placebo effect**. To combat this, often participants are blinded to the treatment group they are in; this is usually achieved by giving the control group a mock treatment (eg: given a sugar pill they are told is the drug). In this way, if the placebo effect is causing a problem with your experiment, both groups should experience it equally.



**Blinding your study means that your subjects don’t know what group they belong to - all participants receive a “treatment”**

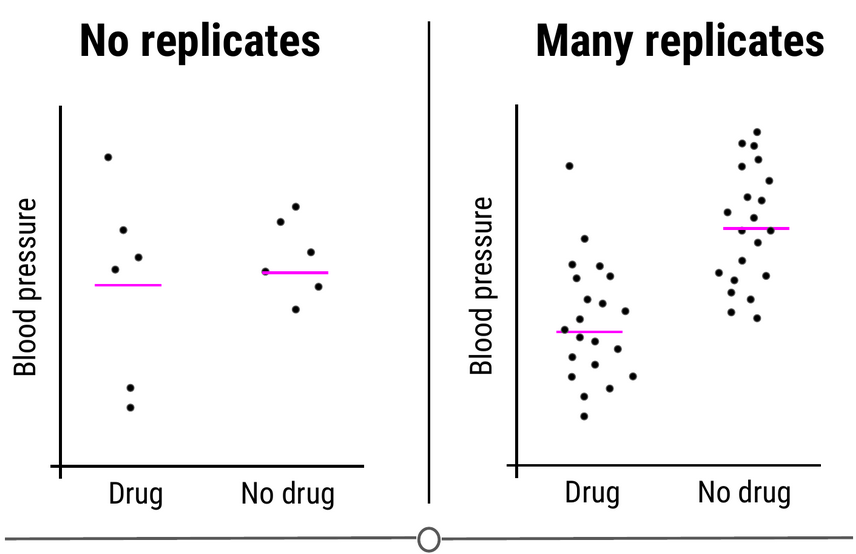
And this strategy is at the heart of many of these studies; spreading any possible confounding effects equally across the groups being compared. For example, if you think age is a possible confounding effect, making sure that both groups have similar ages and age ranges will help to mitigate any effect age may be having on your dependent variable - the effect of age is equal between your two groups.

This “balancing” of confounders is often achieved by **randomization**. Generally, we don’t know what will be a confounder beforehand; to help lessen the risk of accidentally biasing one group to be enriched for a confounder, you can randomly assign individuals to each of your groups. This means that any potential confounding variables should be distributed between each group roughly equally, to help eliminate/reduce systematic errors.



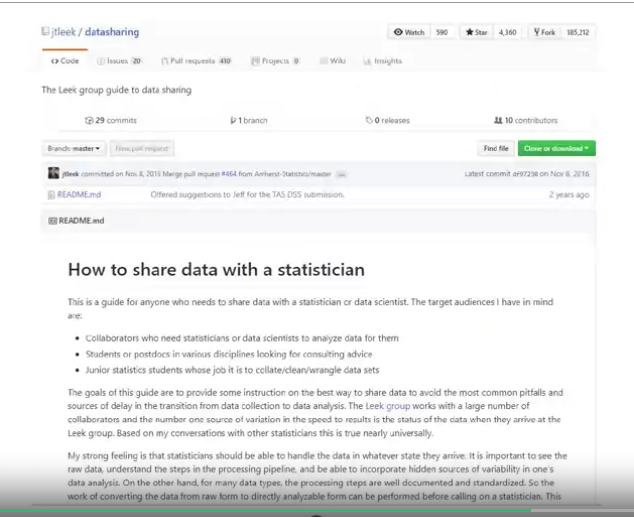
**Randomizing subjects to either the control or treatment group is a great strategy to reduce confounders’ effects**

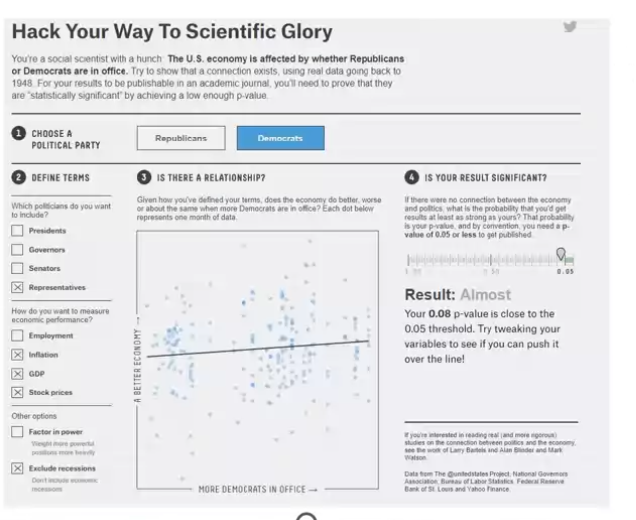
There is one final concept of experimental design that we need to cover in this lesson, and that is **replication**. Replication is pretty much what it sounds like, repeating an experiment with different experimental subjects. A single experiment’s results may have occured by chance; a confounder was unevenly distributed across your groups, there was a systematic error in the data collection, there were some outliers, etc. However, if you can repeat the experiment and collect a whole new set of data and still come to the same conclusion, your study is much stronger. Also at the heart of replication is that it allows you to measure the **variability** of your data more accurately, which allows you to better assess whether any differences you see in your data are significant.



**Replication studies are a great way to bolster your experimental results and get measures of variability in your data**

### Sharing data

Once you’ve collected and analysed your data, one of the next steps of being a good citizen scientist is to share your data and code for analysis. Now that you have a GitHub account and we’ve shown you how to keep your version controlled data and analyses on GitHub, this is a great place to share your code! 



In fact, hosted on GitHub, our group, [the Leek group](https://github.com/jtleek/datasharing), has developed a guide that has great advice for how to best share data!

### Beware p-hacking!

One of the many things often reported in experiments is a value called the **p-value**. This is a value that tells you the probability that the results of your experiment were observed by chance. This is a very important concept in statistics that we won’t be covering in depth here, if you want to know more, check out [this](https://www.youtube.com/watch?v=UsU-O2Z1rAs) video explaining more about p-values.

What you need to look out for is when you manipulate p-values towards your own end. Often, when your p-value is less than 0.05 (in other words, there is a 5 percent chance that the differences you saw were observed by chance), a result is considered [significant](https://xkcd.com/1478/). But if you do 20 tests, by chance, you would expect one of the twenty (5%) to be significant. In the age of big data, testing twenty hypotheses is a very easy proposition. And this is where the term [p-hacking](https://en.wikipedia.org/wiki/Data_dredging) comes from: This is when you exhaustively search a data set to find patterns and correlations that appear statistically significant by virtue of the sheer number of tests you have performed. These spurious correlations can be reported as significant and if you perform enough tests, you can find a data set and analysis that will show you what you wanted to see.

Check out this [FiveThirtyEight](https://projects.fivethirtyeight.com/p-hacking/) activity where you can manipulate and filter data and perform a series of tests such that you can get the data to find whatever relationship you want!

[XKCD](https://xkcd.com/882/) mocks this concept in a comic testing the link between jelly beans and acne - clearly there is no link there, but if you test enough jelly bean colours, eventually, one of them will be correlated with acne at p-value < 0.05!

### Summary

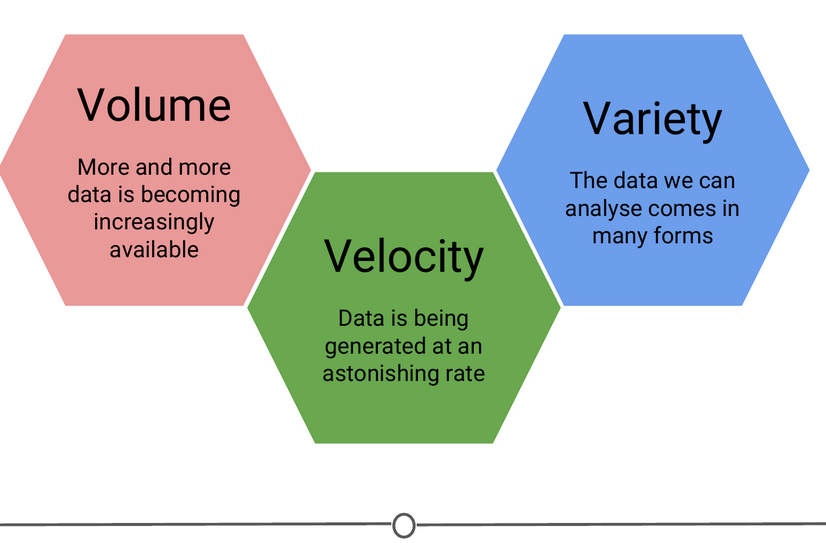
In this lesson we covered what experimental design is and why good experimental design matters. We then looked in depth to the principles of experimental design and defined some of the common terms you need to consider when designing an experiment. Next, we detoured a bit to see how you should share your data and code for analysis. And finally, we looked at the dangers of p-hacking and manipulating data to achieve significance.

## W4 4 - Big Data

A term you may have heard of before this course is “Big Data” - there have always been large datasets, but it seems like lately, this has become a buzzword in data science. But what does it mean?

### What is big data?

We talked a little about big data in the very first lecture of this course. As the name suggests, big data are very large data sets. We previously discussed three qualities that are commonly attributed to big data sets: Volume, Velocity, Variety. From these three adjectives, we can see that big data involves large data sets of diverse data types that are being generated very rapidly.



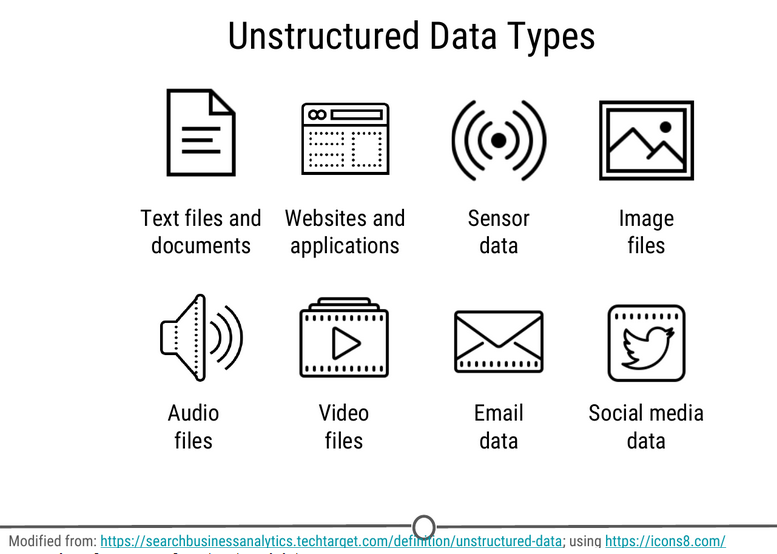
**Three qualities of big data**

So none of these qualities seem particulary new - why has the concept of big data been so recently popularized? In part, as technology and data storage has evolved to be able to hold larger and larger data sets, the definition of “big” has evolved too. Also, our ability to collect and record data has improved with time such that the speed with which data is collected is unprecedented. Finally, what is considered “data” has evolved, so that there is now more than ever - companies have recognized the benefits to collecting different sorts of information, and the rise of the internet and technology have allowed different and varied data sets to be more easily collected and available for analysis. One of the main shifts in data science has been moving from structured data sets to tackling unstructured data.

### What is structured data? What is unstructured data?

Structured data is what you traditionally might think of data; long tables, spreadsheets, or databases with columns and rows of information that you can sum or average or analyse however you like within those confines. Unfortunately, this is rarely how data is presented to you in this day and age. The data sets we commonly encounter are much messier, and it is our job to extract the information we want and corral it into something tidy and structured.

With the digital age and the advance of the internet, many pieces of information that weren’t traditionally collected were suddenly able to be translated into a format that a computer could record, store, search, and analyse. And once this was appreciated, there was a proliferation of this unstructured data being collected from all of our digital interactions: emails, Facebook and other social media interactions, text messages, shopping habits, smartphones (and their GPS tracking), websites you visit, how long you are on that website and what you look at, CCTV cameras and other video sources, etc. The amount of data and the various sources that can record and transmit data has exploded.



**Some examples of sources of unstructured data sources**

It is because of this explosion in the volume, velocity, and variety of data that “big data” has become so salient a concept; these data sets are now so large and complex that we need new tools and approaches to make the most of them. As you can guess given the variety of data types and sources, very rarely is the data stored in a neat, ordered spreadsheet, that traditional methods for cleaning and analysis can be applied to!

### Challenges of working with big data

Given some of the qualities of big data above, you can already start seeing some of the challenges that may be associated with working with big data.

1. It is big: there is a lot of raw data that you need to be able to store and analyse;
2. It is constantly changing and updating: By the time you finish your analysis, there is even more new data you could incorporate into your analysis! Every second you are analysing, is another second of data you haven’t used!
3. The variety can be overwhelming: There are so many sources of information that it can sometimes be difficult to determine what source of data may be best suited to answer your data science question! And finally,
4. It is messy: You don’t have neat data tables to quickly analyse - you have messy data. Before you can start looking for answers, you need to turn your unstructured data into a format that you can analyse!

### Benefits to working with big data

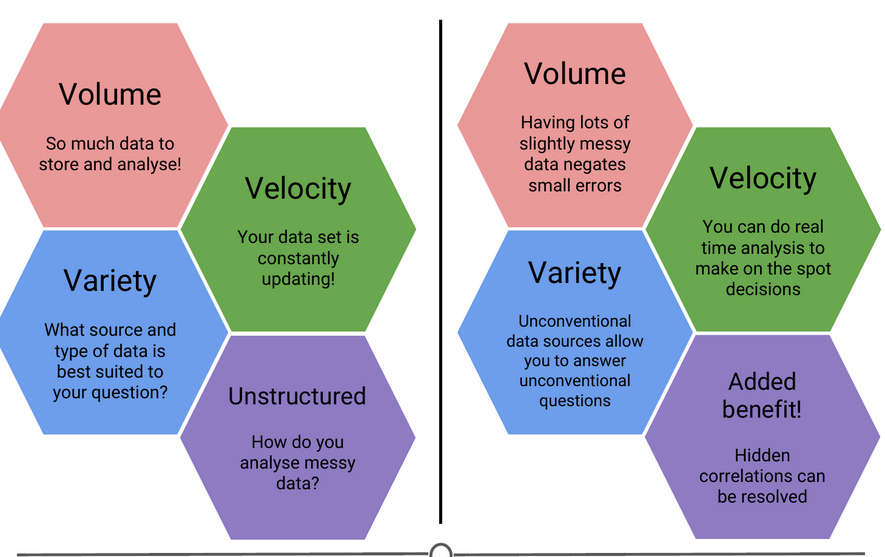
So with all of these challenges, why don’t we just stick to analysing smaller, more manageable, curated datasets and arriving at our answers that way?

Sometimes questions are best addressed using these smaller datasets, but many questions benefit from having lots and lots of data, and if there is some messiness or inaccuracies in this data, the sheer volume of it negates the effect of these small errors. So we are able to get closer to the truth even with these messier datasets.

Additionally, when you have data that is constantly updating, while this can be a challenge to analyse, the ability to have real time, up to date information allows you to do analyses that are accurate to the current state and make on the spot, rapid, informed predictions and decisions.

One of the benefits of having all these new sources of information is that questions that weren’t previously able to be answered due to lack of information, suddenly have many more sources to glean information from and new connections and discoveries are now able to be made! Questions that previously were inaccessible now have newer, unconventional data sources that may allow you to answer these formerly unfeasible questions.

Another benefit to using big data is that it can identify hidden correlations. Since we can collect data on a myriad of qualities on any one subject, we can look for qualities that may not be obviously related to our outcome variable, but the big data can identify a correlation there - instead of trying to understand precisely why an engine breaks down or why a drug’s side effect disappears, researchers can instead collect and analyze massive quantities of information about such events and everything that is associated with them, looking for patterns that might help predict future occurrences. Big data helps answer what, not why, and often that’s good enough.



**Comparing the challenges and benefits to working with big data**

### Will big data solve all our problems?

Big data has now made it possible to collect vast amounts of data, very rapidly, from a variety of sources (and improvements in technology have made it cheaper to collect, store and analyse) - but the question remains, how much of this data explosion is useful for answering questions you care about?

Regardless of the size of the data, you need the right data to answer a question. A famous statistician, John Tukey, said in 1986, “The combination of some data and an aching desire for an answer does not ensure that a reasonable answer can be extracted from a given body of data.” Essentially, any given data set may not be suited for your question, even if you really wanted it to; and big data does not fix this. Even the largest data sets around might not be big enough to be able to answer your question if it’s not the right data.

### Summary

In this lesson, we went over some qualities that characterize big data: volume, velocity, and variety. We compared structured and unstructured data, and examined some of the new sources of unstructured data. Then we turned to looking at the challenges and benefits of working with these big data sets. And finally, we came back to the idea that data science is question driven science and even the largest of data sets may not be appropriate for your case.