Quantitative and Qualitative Analysis

What's out there and what you need to know.

Devan BeckerPublic Health Agency of Canada; National Microbiology LaboratoryFebruary

Introduction

Land Acknowledgement

UWO exists on the traditional territories of:

- Anishinaabek
- Haudenosaunee
- Lūnaapéewak
- Chonnonton

These lands are connected with the London Township and Sombra Treaties of 1796 and the Dish with One Spoon Covenant Wampum.

I acknowledge that Western University is located on the traditional territories of the Anishinaabek, Haudenosaunee, Lūnaapéewak, and Chonnonton Nations, on lands connected with the London Township and Sombra Treaties of 1796 and the Dish with One Spoon Covenant Wampum. The Dish with One Spoon is a metaphor for the responsibilities that we have when sharing the land; each of us eats from the same bowl and we must always leave enough for the others. It's easy for people like me to think of history books when talking about indigenous peoples, but I recognize that they are contemporary stewards of the land and vital contributors of our society. I also recognize the great diversity among indigenous peoples, both within and between nations. For instance, the Dish With One Spoon Covenant Wampum is not an "Indigenous Thing", it's specific to the Haudenosaunee peoples. As a statistician, I am tied into a particular way of knowing. These land acknowledgements remind me that I should seek out other ways of knowing, especially via consultation with native peoples when doing spatial statistics or public health measures.

About Me

- Education
  - B.Sc. Math (Laurier)
  - M.Sc. Stats/Biostats

  - Ph.D. Stats

- Work
  - Postdoc - SARS-CoV-2 in wastewater
  - Public Health Agency of Canada

- Life
  - Music, reading, outdoorsy stuff
  - Crying about the housing market

A little about me before we begin. Did an undergrad thesis project with Doug Woolford. He introduced me to some people at Western for my master's, then followed me and became my PhD supervisor I defended my PhD thesis exactly one week before the pandemic hit Canada, so I am extremely lucky that I got hired as a Postdoc in the Schulich school. Outside of academia, I listen to and play music (poorly), go camping, hiking, showshoeing, kayaking, and rock climbing, and I like to read a variety of things.

Foreshadowing

Outline

- **Quantitative:** Dealing with numbers
  - Any number in a range
  - Only 0's and 1's (maybe a 2)
  - Things we can turn into numbers

- **Qualitative:** Dealing with descriptions
  - Using your brain
  - Using your computer

- **Meditative:** Dealing with everything
  - How to get started
  - Accessing resources
  - Not being afraid of coding

I'm a statistician by training, so most of this will be spend on quantitative analysis. I'm going to focus on linear regression, which is an analysis technique that can be used when you're modelling something that could take any value in a range. I'll briefly go over a few other techniques that work for numbers in a range. I'll also talk about classification problems, which is when the thing we're modelling can only be one of a handful of numbers. I'll talk about qualitative analysis, but mostly from a data scientist's perspective. Finally, I'll do my best to get you started on learning your own analysis, and hopefully convince you that all this isn't that scary. There are lots of resources on campus and across the internet, but there are some caveats with all of those.

Before we begin

- Interrupt at any time
- All notes/links/resources/R code are on GitHub
- Ask future questions in the PAW Slack chat
- I have allowed myself **ONE** equation.

```
The GitHub version also has my (approximate) script inside
:::notes::: tags, which show up as text in the pdf.
```

I'll add the GitHub link to the chat now, but I'll also have Mihaela send it out after the meeting.

What to watch for

Keep an eye out for the following concepts:

1. Garbage In, Garbage Out (GIGO)
2. Numerical summaries lie - you need plots!
3. Models are models.
4. Models are wrong.

No matter what kind of analysis you are doing, there are some things in common. First is GIGO - your data limit what you can do with your model. Second, looking at numbers alone will only tell you part of the story. Get in the habit of plotting everything you come across. Third is the tautology that all models are models. In most software, spending time learning how one model works will help you understand many other different models. For instance, linear models and neural networks have almost the same syntax in R. Finally, come to terms with the fact that any analysis you do will be imperfect. The most important part of any modelling endeavour is being humble about the results and acknowledging where and why it deviates from the truth.

Regression

Terminology

So let's start with the fun one - regression! Every discipline uses their own terminology, so we'll get that out of the way first. The variable that we're trying to model is called the **target**. You may know this as the dependent variable or the response. Whatever you call it, this is generally what we're trying to make predictions for and usually shows up on the y axis. The features are the information we're trying to incorporate into our model to better predict the target. You may know them as explanatory or independent variables, or maybe as IVs.

- The **Target** could be any number in a range.
  - A.k.a. dependent variable or response.
- The Features could be any data type
  - A.k.a. explantory or independent variables (IVs)

The Data

For this example, we're going to use the Palmer Penguins dataset, which was collected at Palmer station in antarctica. We have their body mass (in grams), which is going to be our target variable. We're trying to determine if penguins are getting better nutrition and/or less competition on different islands. We also know their bill length (mm) and their flipper length (mm). These penguins are one of three species, are found on one of three islands (with some overlap) and are either male or female.

Note to people reading the pdf: the data are from https://allisonhorst.github.io/palmerpenguins/ and all my analyses are done in R.

mass	bill_len	flipper_len	species	island	sex
3750	39.1	181	Adelie	Torgersen	male
3800	35.3	187	Adelie	Biscoe	female
4150	42.0	210	Gentoo	Biscoe	female
5350	48.7	222	Gentoo	Biscoe	male
3725	52.7	197	Chinstrap	Dream	male
3750	51.3	197	Chinstrap	Dream	male
3400	50.1	190	Chinstrap	Dream	female

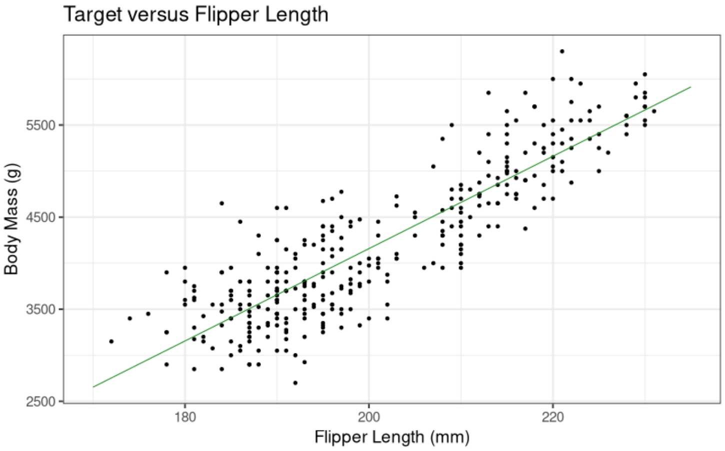
Goal

How does body mass change with flipper length?

- The **slope** quantifies this change

- If the data are good, estimated slope is similar to *population* slope

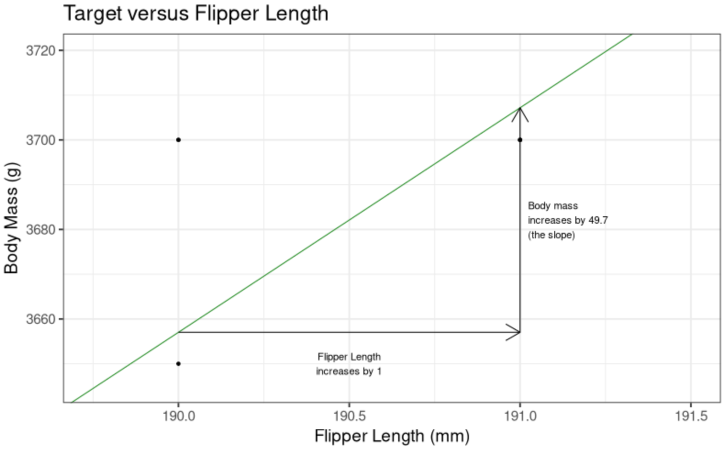
Intro to linear models



- Find the slope and intercept to best fit the cloud of points.
    - Slope: rise over run.
    - Intercept: the body mass when flipper length is 0.

The main goal of linear regression is to find the slope and the intercept that best fit a cloud of points. In this plot, the slope represents how much the body mass increases with flipper length. Generally, this is what we want from a regression - how does the target change with the features? The intercept represents the weight of the penguin when the flipper length is 0. Don't think too much about that - the intercept is just a mathematical requirement for building a line.

Linear models: slopes



While the intercept can be modified so that it is meaningful, the slope is almost always meaningful for any analysis. The value of the slope represents the relationship between the features and the response. Most of the time that we're doing a linear regression, this is what we want to quantify.

Binary Features

Suppose we have a variable that is labelled either 0 or 1.

What does the slope represent?

The slope represents the change in the target for each one unit increase in the feature. Binary features only have one unit in between, so the increase in the target is the difference in the centres of the two groups. This means that t-tests are actually just linear regression in disguise! There's a difference in the variance calculation, but the underlying machinery works the same. As a side note, ANOVA and ANCOVA are also secretly linear models. In fact, fitting ANOVA in R involves fitting a linear model.

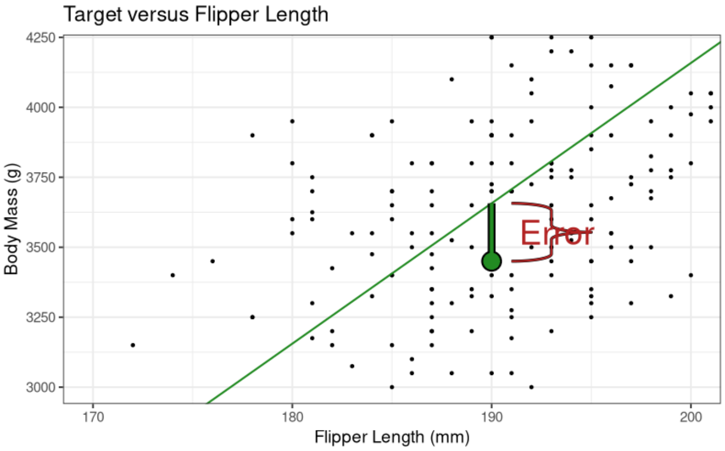
The story so far

- The Intercept is a mathematical necessity
- The Slope answers our questions

But how good is our model?

Computers are really smart, but they're also really dumb. A computer will calculate the intercept and the slope for any data you throw at it - regardless of whether it's a good idea! There are many cases where linear models are not appropriate - such as when the target is categorical - but the computer computes anyway. That's why they call it a computer and not a thinker! Even if a linear model is appropriate, it doesn't mean your results will be good.

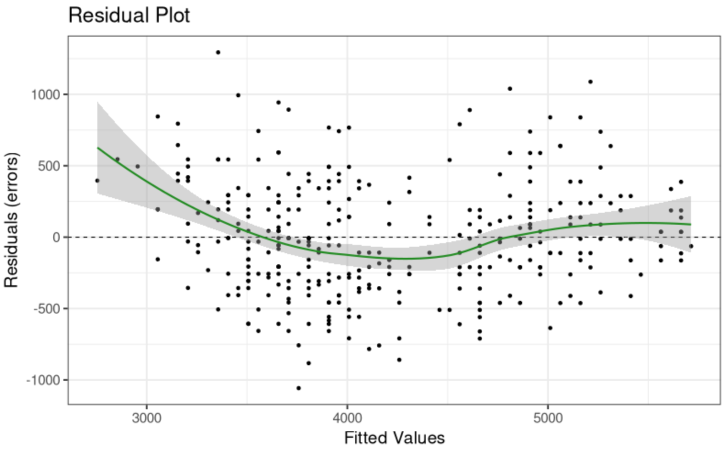
The most important part!



- The line will never go through every point perfectly!
- Know where the model fails can tell you everything!

When we fit a line, we can find the estimated value simply by looking at the height of the line. It would be extremely surprising if the line always hit every point! Instead, there's always going to be some measure of error. Understanding these errors is the most important part of any modelling challenge - no matter what type of model you're using. Later, we'll see some models that use words as their input. Even with these models, knowing about the errors is the most important part!

Residual plots: residuals versus predicted



- A perfect residual plot should show no pattern.
- This plot looks like there's a slight pattern…

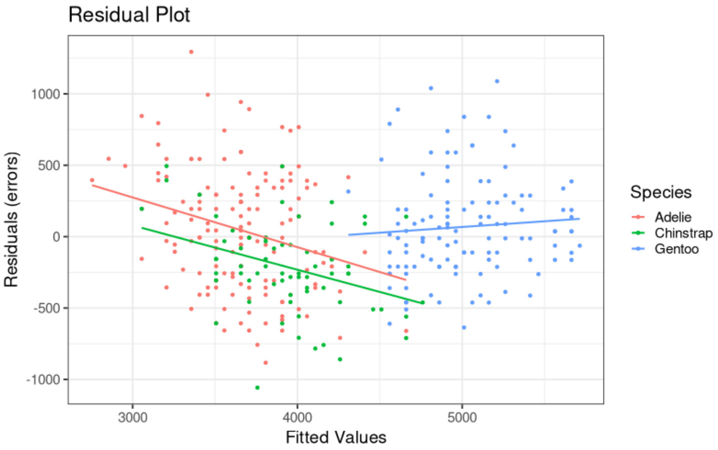
Rather than fitting every point, a perfect model is one that's wrong consistently. That is, the estimated values are not systematically different from the truth. This problem can happen when you have a pattern in the residuals, and the best way to do this is with a residual plot.

The residual plot shows the errors on the y axis and the estimated values on the x axis. This may seem a bit strange; most people would expect to plot the errors against the features. We do it this way for two reasons:

1. If there are a lot of features, it becomes very tedious to check every plot.
2. There might be complicated interactions between features that are hard to see without considering many features.

The pattern

Each species has a slightly different relationship!!!



Here's the problem: there are three different species of penguins, each with different physiology. For this application, we'd need to account for these differences in species. At this point, it's worth working through a textbook or a course on linear models.

Putting it all together

1. Get data
   - Data cleaning is the hardest part.

2. Check data
   - If you haven't plotted it, you're doing it wrong.

3. Fit model

4. Check model

    - If you haven't plotted it, you're doing it wrong.

Let's take a minute and look at what we've done so far. We started with some good clean data, but this is rarely an easy step. In fact, cleaing data is frequently the most time consuming part of an analysis! Furthermore, the quality of the data determines what you can say about the target. We plotted the data and saw that our assumed model form looks more-or-less appropriate, so we went ahead and fit a model. You might notice that I didn't talk too much about actually fitting the model. This is because that step is quite easy once you're comfortable with the software that you use. The hard part is making sure that the model you fit is the model that you want. In this example, I have a long way to go!

Other

- Mixed Models

Machine Learning

What is Machine Learning?

- Statistics, but done by a computer scientist…

–OR–

- Anything algorithm that tries to get information from data!

This includes linear regression!

There are tons opinions on what counts as machine learning. Most definitions have some variation on "getting information from data", but each definition will have different caveats or inclusions.

To help you understand what machine leaning actually is, think back to linear models. When you tell your computer to calculate the slope, that counts as the "machine" "learning" what the slope is based on data. Artificial Intelligence is the process of a machine learning how to think like a human, but even in this case the learning part simply means calculating things from input data.

For this workshop, I'll talk briefly about a few common techniques and then do a bit of a deeper dive into neural networks. Trust me, they're not that scary.

Regression in Machine Learning

- **Lasso Regression**
    - It's like linear regression, but it automatically removes features.
    - Related: Ridge regression, ElasticNet

- **xgBoost**

    - Remember the residual plots? What if we fit a regression to those residuals?

- **Neural Nets**
    - . . .

The first example of machine learning is called LASSO regression. In the linear regression section, we saw that variable selection is hard. Lasso does it for you!

Another issue we had in linear regression was the problem of residuals still being correlated with one of the features. xgBoost does this for you!

There are many other types of machine learning that aren't regression, and we'll cover those in the sections on Classification and Qualitative analysis. For now, let's dive into neural networks.

Neural Nets

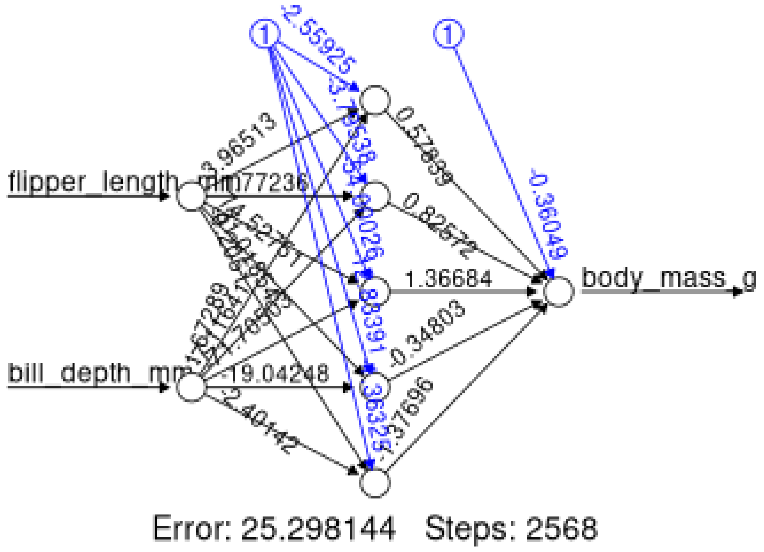
- What most people think of as ML.
    - Deep Learning: fancy neural nets.

- Loosely based on the way synapses work.

- Just a bunch of linear regressions

When most people think of machine learning, they're thinking about neural nets. They see them as a computer having a brain, which is kinda true (but not really). They are inspired by a simplified version of how brains pass information, but a completely disconnected from the way brains learn and think. People also think of them as being hopelessly complex black boxes, but they're actually just a bunch of linear regressions taped together.

Neural Net Setup



- Each node is a linear model
- The target node tells hidden nodes how they were wrong
  - "Backpropagation"; recurrent neural network
- Have to choose number of layers and nodes

So here's a computer brain. There are som inputs, in this case the flipper length and the bill length, and these feed into something called a hidden layer. Basically, these try and combine all of the features in a way that extracts more information about them. To predict the target, the information from the hidden layer is combined.

So why do I call this a linear model? Let's just look at the target. The information is "combined" to predict the target. How is it combined? With a linear model! Each of the "nodes" in the hidden layer is a function that takes in values from all previous nodes and spits out a new value, and these values act as the features. The numbers on the line are just the slopes of a linear model, and the blue "1" is the intercept term. If you learn more about neural networks, you'll learn that these are called loadings and the bias, but that's not important right now.

So the difficult question now is: how do we determine the way this hidden layer works? Each time we try and make a prediction, we're wrong by a certain amount. The target node reports back to all the hidden nodes to tell them them how and why it was wrong, and the hidden nodes adjust their values appropriately. The hidden nodes generally start at a random number and then try and adjust that number to one that would have reduced the error in the prediction. This particular formulation is called a recurrent neural network because of the way the error is propagated backwards through the network.

Choosing the number of nodes in a hidden layer is a challenge, and it's possible to have any number of hidden layers! Just like with linear models, we are stuck trying to decide on a model form.

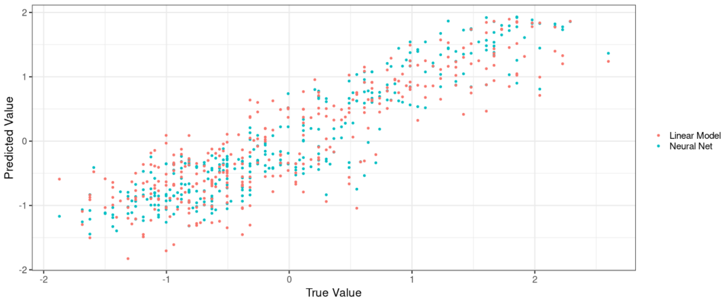
Linear Models Versus Neural Nets

- LM gives interpretable slopes
- NN accounts for complex interactions

- LM is better for **inference**
- NN is better for **prediction**

In general, neural nets tend to give better predictions. However, the interpretation of the parameters of linear models is often much better. For instance, being able to say how much the body mass increases as flipper length increases tells us something about the biological relationships. Neural nets will give similar, if not better, predictions, but we won't know why.

Is NN always better than LM?



No.

The linear model fit the data quite well, depsite not incorporating the species in the model. The neural nets perhaps gave slightly better predictions. I have no idea why, though, and the neural net has no interest in explaining it to me.

The moral here is this: don't be impressed by machine learning; it's a tool to make fantastic yet unexplainable predictions. The moment you see a machine learning model you should immediately ask yourself whether good predictions are worth the lack of interpretability.

ML and Ethics

- ML finds patterns that exist
  - It perpetuates existing patterns, e.g. black recidivism

- ML is hard to audit
  - Is it just looking at peoples' race? Hard to say!
  - Explainable AI (XAI)

- ML doesn't answer email
  - Why did it make a certain decision?
  - Ca't plead your case.

https://delphi.allenai.org/?a1=Using+AI+to+determine+ethics

As a corollary to the lack of interpretation, the model might actually be using data that you don't want it to use. For example, models that try and predict whether a convicted criminal will re-offend (called recidivism. To audit this model, researchers had to look at all of the black cases and all of the white cases, along with the sociological differences between these races. It wasn't enough to just check whether the model gave different predictions for white and black people. Instead, the researchers had to carefully scrutinize cases where the only differences were race **and** features that should not affect recidivism. They found that much of the model's computation was spent predicting whether the person was black, then basing its decision on the fact that, historically, black people are arrested and therefore convicted more for equal offences. The most important thing to know about a model is how it's wrong, especially when it's wrong in a way that perpetuates systematic racism.

Classification

Binary Target

mass	bill_len	flipper_len	species	island	**sex**
4500	46.1	211	Gentoo	Biscoe	female
5150	46.8	215	Gentoo	Biscoe	male
4600	48.2	210	Gentoo	Biscoe	female
5400	48.4	220	Gentoo	Biscoe	male
4200	45.5	210	Gentoo	Biscoe	female
5550	50.4	224	Gentoo	Biscoe	male

Let's return to the penguins data set. The original study intended to quantify the sexual dimorphism, so let's focus on that. In this case, the biosex is either male or female - it can only be one of these two things, which is why it's called binary. It may seem strange to try and predict the biosex of the penguins since that's something we can fairly easily check, but by knowing what factors make better predictions we can learn a lot about the biological, sociological, and environmental factors affecting penguins.

Ethics: biosex versus gender

- Chinstrap penguins have a higher-than-average occurence of homosexual behaviour.
  - Tufts University, Feb 2021: "What's With All the Gay Penguins?"

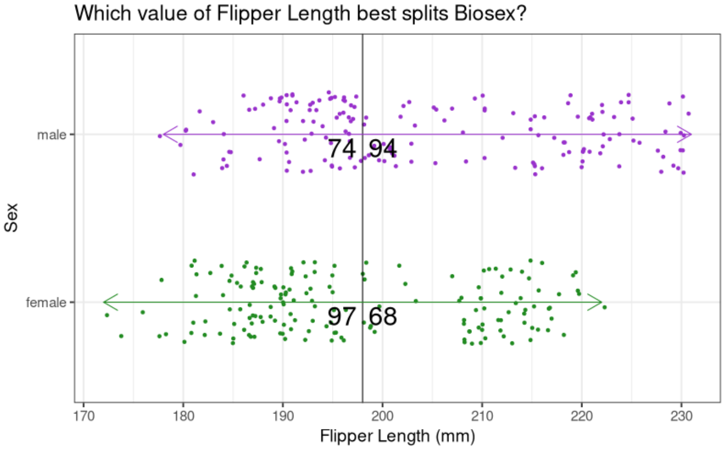
- Gentoo penguins have less rigid gender roles.
  - NBC news, Sept 2019: "Gay penguins at London aquarium are raising 'genderless' chick"

- Adelie penguins of any gender all want to be like Adele
  - <Citation Needed>

"Biosex" is a fundamentally imperfect measurement of gender roles.

In any analysis, you'll likely run into ethical challenges. These are penguins, but that doesn't mean that the difference between biosex and gender isn't present. Understanding the data collection is paramount to a good analysis, and poor interpretations can lead to ethical quandries.

Choosing between two options



- When Flipper Length is below 198, most are female.
- This is called SVM, or Support Vector Machines

Let's start with a simple model where we only use flipper length to try and guess the penguins' biosex. In this plot, I've separated the data into male and female and I added a vertical line at a flipper length of 198. For this model, we're just going to guess any penguin with a longer flipper than 198 is likely female.

But how were we wrong?

If we label any penguin with Flipper < 198 as female:

	Flipper < 198	Flipper ≥ 198
Male	74	94
Female	97	68

- When we label them female, they're actually female 97/(74+97)=56% of the time.
- When they're actually female, we label them female 97/(68+97)=58% of the time.

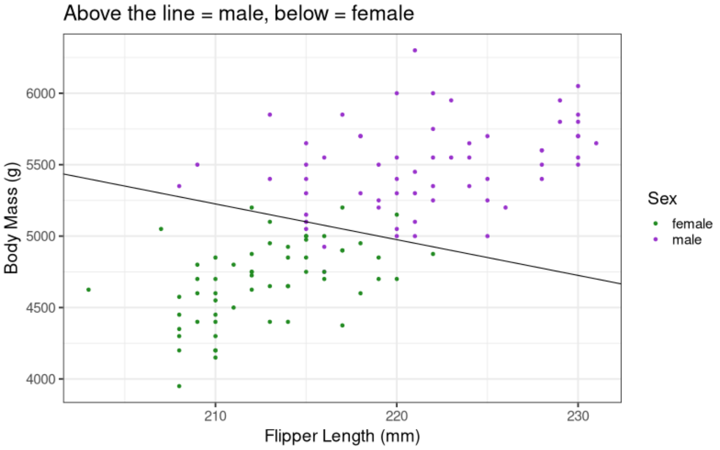
This is a **Confusion Matrix**.

See also: sensitivity, specificity, precision, recall, F1 score, ROC/AUC curves.

This is a whole rabbit hole, so I'm going to try and not spend too much time on it. Again, the most important thing to know about a model is how it's wrong. For classification problems, it's less clear what it means to be wrong. On the one hand, we can simply check how many of the predictions were correct. We can do this just for female and just for male penguins, and in this case our model was correct 56% of the time for female penguins. On the other hand, we can look at how much of the data the model was correctly able to classify, in which case we correctly labelled 58% of the females in the data.

Note the slight change in this: we can look at how many of the predicted females were actual females, and we can look at how many of the actual females were predicted to be female. Which of these definitions is important to you depends on the context, and many volumes have been written on the matter. For our purposes, I'm just going to leave you with some homework: Read up on the sensitivity, specificity, precision, recall, F1 score, ROC/AUC curves.

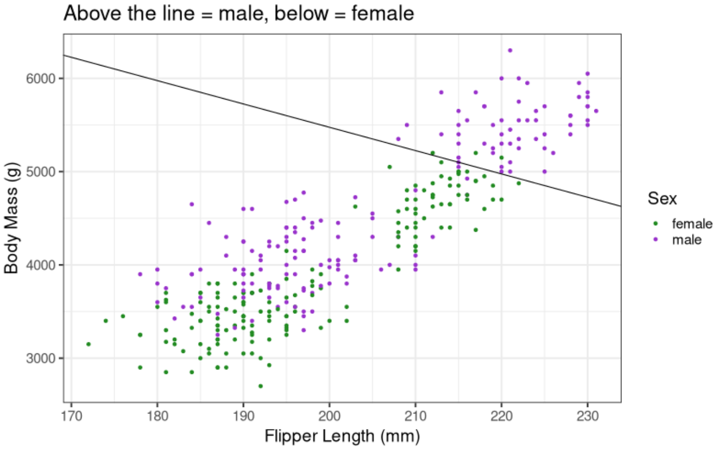
More dimensions!



- With more information we can fit a better model!
- . . .

The single vertical line is obviously too simple. Instead, we could look at two dimensions! Here, it's obvious that our method - Support Vector Machines - is trying to find a line that separate our data. As you might have guessed, this is basically just linear regression. I know it's not flashy, but I highly recommend doing a deep dive into linear regression before trying to learn the fancier models.

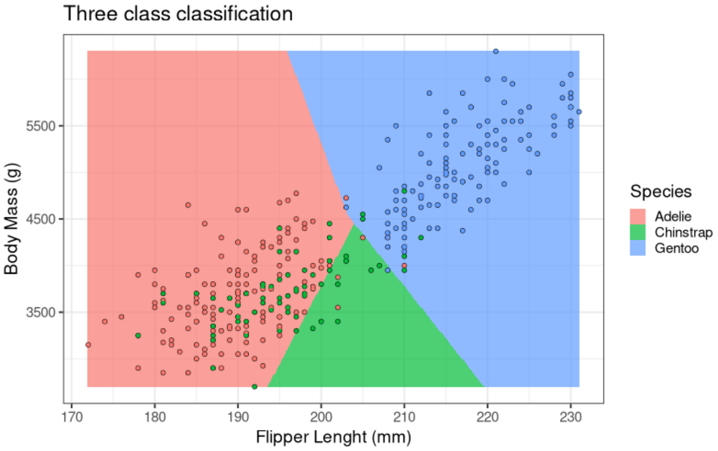
. . . but there's a reason I only used Gentoo



Here's when I show that my simple teaching data sets are sanitized versions of reality. For Gentoo penguins, it's pretty straightforward to draw a line through the data and separate the males from the females. There are three different species in this data set, and the other two species are much harder to separate.

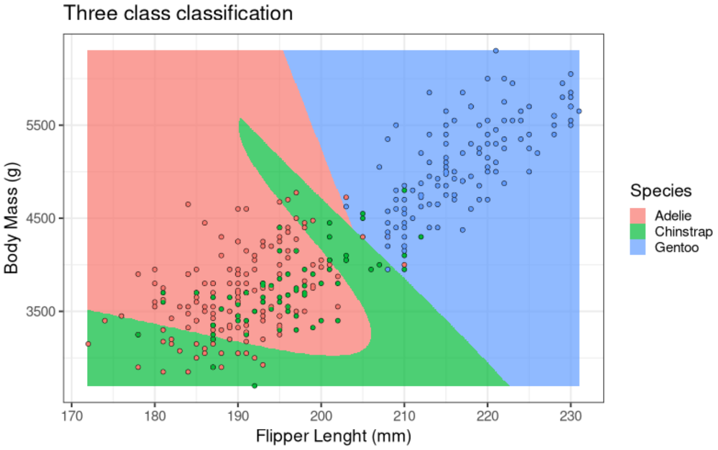
However…

Three categories: Species



We can try to guess the species! With three categories, we can draw a couple lines to separate the data. As you can see, Chinstrap and Adelie are hard to tease apart in this way. However…

It doesn't need to be linear!



We don't have to use straight lines! Obviously something funky happened here, and I need to do more work to tune my model.

Other Classification Models

- **Logistic Regression**

  - Basically, LM for probabilities
    - Interpretable results!
  - For multiclass, Multinomal Regression

- **Decision Trees** and **Random Forests**
  - Very important models that I'm not covering
  - Still a linear model at heart

- Naive Bayes Classifiers

- K-Nearest Neighbours (KNN)

- Neural Nets!

I chose to show you Support Vector Machines (SVM) because the concept is pretty easy to understand (even though the name is pure jargon). However, Logistic Regression is probably the most important for you to learn. It is extremely close to linear regression, except it models the probability that a given observation belongs to a certain class. As with linear regression, it gives you interpretable parameters (except it gives the odds ratio instead of a slope). After learning Logistic, move on to Decision Trees. These models work somewhat similar to support vector machines, but in a much more systematic way. Random Forests are clever collections of Decision Trees, so spend some time with Decision Trees first. Naive Bayes gets very mathy very quickly, but it is often one of the best classifiers (and hardest to implement). KNN is rarely the best classifier for general tasks, but it shines for some very particular problems. Support Vector Machines are great tools for exploring and visualizing your data, but they're rarely useful in practice. SVM suffers from a lack of interpretability in higher dimensions, and in this case you might as well just use a Neural Net!

Unsupervised Learning

Definition

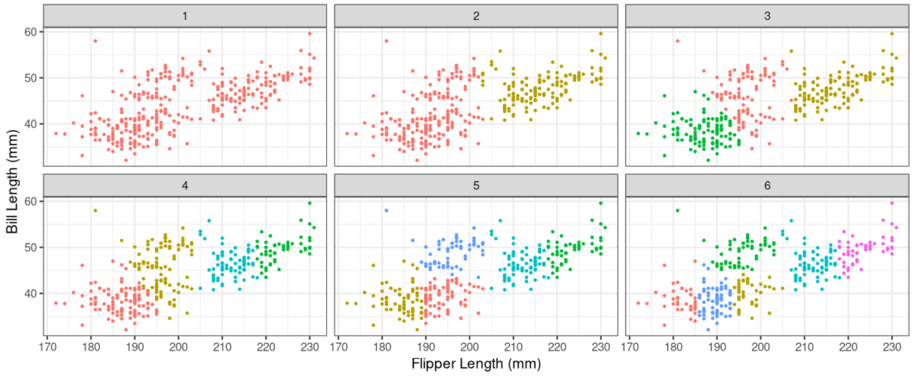
In classification, we're predicting labels *and checking if we're right.*

Unsupervised learning means *we don't know the labels.*

In all the examples we've learned so far, we've known what we're trying to predict. We know that some penguins are male and some are female, and we tried to find out what other differences that entails.

In unsupervised learning, we don't know what we're trying to predict! Instead, we plot our data and see some clusters, and we try and separate the data based on those clusters. Let's do an example.

K-means Clustering



- Pretend that Species info is *NOT* available.
- How many clusters are there?

Kmeans clustering is an unsupervised technique that tries to find groups. If Species information wasn't available, how many groups would you say that there are? Unfortunately, K-means requires us to tell it how many groups we're looking for, so we have to just fit 1 mean, check the plot, fit 2 means, check the plot, fit 3 means, check the plot. This is where the name comes from - it assumes that there are K locations that are the center of the clusters, and tries to choose these locations to minimize the average distance to all of the points in that cluster. With most unsupervised techniques, there's a lot of tweaking to choose the number of groups, and then a lot of investigation afterwards to see what exists in the groups in your data.

Dimension Reduction

Motivation

Why use many features when few features do trick?

By combining features, we might:

- Find out which features have similar effects on the target.

- Find clusters

Another important machine learning technique is dimension reduction. This can be used when you've measured a lot of different features and some of them may be correlated, or it can be used to get different views of your data (lieterally). It's also a good way to find clusters in the data, but please note that this requires an extra step. There are many such techniques, but by far the most popular is Principal Components Analysis.

Principal Components Analysis (PCA)

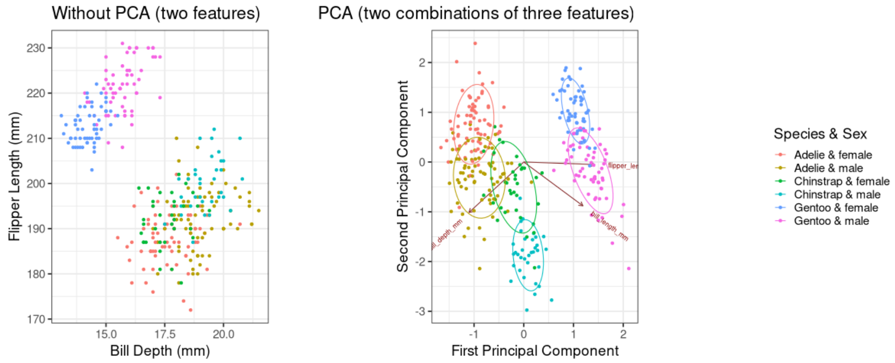
A *Principle Component* is a combination of the features (NOT target).

Each component is unrelated to the others.

$$
\begin{align}
PC1 &= 0.55 * bill\_length - 0.51 * bill\_depth + 0.65 * flipper\_length \tag{1}\\
PC2 &= -0.65 * bill\_length - 0.75 * bill\_depth - 0.03 * flipper\_length \tag{2}\\
&... \tag{3}
\end{align}
$$

Here it is! It's the equation I told you I had! A principal component is a combination of all of the features. Unlike LASSO regression, which tries to remove some features, PCA rotates them so that they're as separated as possible.

Principle Components - clustering



The Components become our new features, so this plot is just like plotting flipper length against bill depth. I've manually labelled the sex and species from the data - this is not clustering! However, it demonstrates how rotating the features so that each component is as different as possible can be used for clustering. The labelled lines represent how much of each feature applies to each component. Flipper length is a big part of the first

When I said PCA rotates the features, I meant literrally. In the repo, there's a script that plots the penguins data in 3d. If you have R installed on your machine, check https://dmurdoch.github.io/rgl/ to install the packages. Once you do that, try to rotate the plot until it looks like this PCA plot!

Qualitative Analysis

Qualitative Data

Quality: the properties/characteristics of a thing (not numbers)

- Survey responses
  - "A lot of people seem to talk about painful things . . . "

- Categories
  - "Registered democrats tend to have these qualities . . . "

- Texts
  - "The grammar in this act is different from Shakespeare's usual style . . . "

- Concepts
  - "These documents could be categorized by their use of . . . "

Usually, qualitative research involves your own personal intelligence. You read through survey responses to get a "sense" of what people are saying, then come up with your own categories. Similarly, you might be doing a literature review and need to organize papers into an introduction section. Alternatively, you might have, say, responses from democrats and republicans and want to know what else is different between the two.

Qualitative Data Analysis

- Fully manual: read everything, pay attention, take notes, compare.
  - I can't help you with this.

- Some computer: search within documents, word clouds, etc.
  - Audio/image/video transcription via neural networks
  - Semantic analysis

- Much computer: **Natural Language Processing**
  - It's machine learning, but for words!

I am not a trained qualitative researcher. However, I know how to do machine learning! In the partially automated scenario, machine learning can transcribe audio interviews into text to make it easier to search. Image recognition is a hot topic in machine learning research right now, and this might help a sociologist looking through historical images. Some advanced work is being done to transcribe videos - that is, the computer watches a video and describes what objects it sees and how they're interacting. Semantic analysis conists of assigning values to the words, such as assigning happier words a higher value, then using this to analyse a document for how happy it is. However, the real fun begins when we get computers to interpret the documents for us!

Natural Language Processing

Code is perfect and English is awful

- It's cold outside, yes?

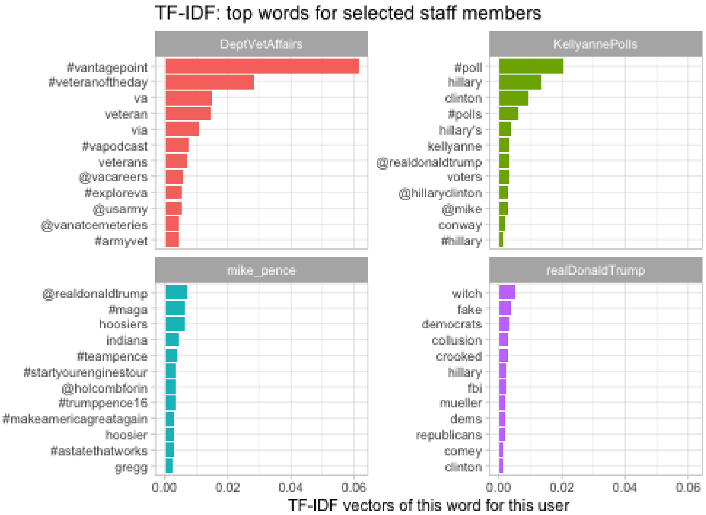
- It’s cold outside, no?

Sometimes, yes and no mean the same thing.

How the heck could a computer have a chance?!?

In this section, I’m going to highlight two NLP applications. Sorry that they are both Trump-related, but he dominated the international conversation and inspired a lot of interesting analyses. In my opinion, both of these applications are neutral in their political tone.

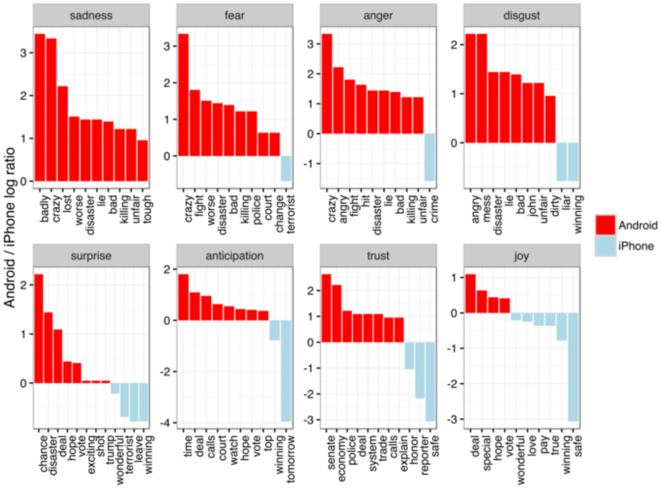
TF-IDF: Who wrote the Op-Ed?



Source: http://varianceexplained.org/r/op-ed-text-analysis/

In this application, an op-ed was written by a anonymous source. That just won’t do - we need to know! The analysis takes us through some interesting data collection and language processing techniques. I love this example because it teaches a lot about how this sort of analysis works, but then the conclusion comes from expert knowledge that NLP wasn’t prepared to handle.

Sentiment Analysis: Trump Uses an Android



Source: http://varianceexplained.org/r/trump-tweets/

This second application shows even more techniques. I would classify this as a confirmatory analysis - they had a hypothesis (Trump tweets from an Android while his staffers use iPhone), and then tried a bunch of methods to see if they all confirm this.

More Advanced Natural Language Processing

- Parts of Speech
  - Nouns, verbs, etc.

- Topic modelling
  - Words that show up in similar sentences prob. have similar topics

- Bag of Words (Word2Vec)
  - How often are words used together?

All of the above can be based on Nueral Nets!

There are many more NLP techniques out there! I want to highlight topic modelling because I find it super interesting. Just like k-means, you tell it how many topics you think there are in a collection of texts. It finds words that tend to be used together and puts them into categories. Yes, this is unsupervised learning for words! What's interesting is that it doesn't have to use all of the words, just the ones that make the strongest coherent topics. It doesn't label the topics, this is your job as a researcher. As I've said many times, the most important thing is to check whether your results make sense.

Meditative

Summary

Same ideas throughout:

1. GIGO
2. Plot everything
3. Learn to code
4. Plot everything

If you take nothing else from this workshop, at least understand these four points.

Learning Path

- Take notes on a basic coding tutorial

- Work through an *easy* passion project
  - Visualize olympic medals (Kaggle)
  - Basic linear model for bitcoin values
- Backpropagate your new knowledge

- Write a tutorial for yourself, share it on GitHub.

- Search Twitter, follow relevant topics/people

If you want to learn how to fit these models and more, you're going to need to learn how to code. There are programs out there that claim to fit these models without needing to code, but you'll end up fighting with them in order to look at the model output in a useful way, or clicking the wrong option and changing everything about the analysis without realizing it.

For the first few steps, I have included resources in the README file for this repo. I highly suggest finding a passion project that you can complete in a matter of days. By analysing data that you're passionate about, you're much more likely to dig further into the results. You'll know the context of the data already, and you'll know immediately if your model results don't make sense.

As you go, take good notes and keep them concise and updated. If you find yourself going back to your notes and only looking at, say, the code, delete everything else! Keep your notes only to the things you for sure won't remember.

Important things we didn't cover

- **Data Cleaning** (don't use Excel)

- Inference versus Prediction

- **Cross Validation**

- Version control and best practices (GitHub!)

- Scrutinizing data

There's a lot more to quantitative and qualitative analysis that I can't cover in one and a half hours! Data cleaning is not an exciting thing to cover, but it's immensely important. I don't recommend using Excel for anything analysis related. With code, you have a record of everything that changed in your data. Several times, I've completed most of an analysis and then received an updated version of the data, and then I forgot what I did.

Another major topic not covered is Cross Validation. This topic is incredibly important for most of the analyses that we covered today. Basically, this gives you another, even better way of evaluating the models. I strongly recommend doing some research on Cross Validation before going forward with your analysis.

R versus Python versus Other

- R is stats focused
  - Python has cutting edge machine learning and general purpose
- R has `dplyr` and `ggplot2`
  - Python teaches/requires better coding skills
- RMarkdown is astounding
  - Black holes were imaged in Jupyter
- Both will work for any analysis
  - Basicaly, use what your colleagues use.

FWIW, I used R for this workshop and code is available.

Thank You!

See you on the slack chat!

Bonus Topic: Bayesian Statistics

I flip a coin and get heads.

What's the probability of heads?

- Frequentist (the usual way): 100%
- Bayesian: I though it was 50%, now I think it's closer to 60% maybe?

Probabilities

- Usual way: long term frequency (e.g., after 100 coin flips, 50 are expected to be heads)
- Bayesian: Uncertainty (e.g., I think the next flip is heads with 50% certainty)

In linear models

- I think the slope is probably 10, on average.
  - There's variance in my belief

- With a small data set, I move my belief closer to what the data says
  - Larger data means estimate is closer to data AND smaller variance.

The *posterior* distribution comes from updating the *prior* with the data (likelihood).

Why Bayesian?

- The posterior is a *distribution*, not a point estimate

- A 95% *credible interval* contains the true mean 95% of the time!

- Much, much, much more flexible models
  - Especially mixed moe