Understanding LM, GLM and GLMM

A worksheet for Data Science. Written by Hannah Wauchope

2nd November 2024

Welcome! Today we're going to be working with a dataset of Hummingbird Counts taken from the Christmas Bird Count. Though this data is freely available online, you do have to make a request to access, which I have done in the past for a separate research project. We should technically request again for this class, so instead I've just scrambled the years and site names a bit.

In the Christmas Bird Count, volunteers go out once a year and count all the birds they see within a set area. It happens all across North America and has been going over 100 years - so is an incredible data resource. Here, I've taken data on counts of five Hummingbird species across six sites.

Our question for today: "Are hummingbird numbers, in general, increasing or decreasing through time?"

Part 1: Data exploration and prep

Exercise 1 - Visualise

Take some time to think about your data, and have a look at it. What columns do we have? What data do they contain?

Exercise 2 - Plot

Now, I'd like you to plot your data to get a better handle on it. Can you make a plot that looks at how count changes through time?

HINT: You're going to want to use the `ggplot` function, and the `geom_point()` call within that. Remember that anything that changes (E.g. Count, Year) should go within `aes` .

Hmm. This plot is a bit of a mess. It also doesn't really represent our data that well, as our data has groups.

Can you adjust this plot so it shows how count changes through time *per species, per site?*

HINT add `facet_grid(Site ~ Species)` as a new line on your ggplot call

That's interesting, but some sites seem to have *lots* of Hummingbirds and some only a few. It's hard to see what's going on. Add `,scales="free"` inside your `facet_grid` call and see it that helps.

Okay, we've had a good look.

Exercise 3 - Year

A final thing before we do: the models are going to get a bit weird if we keep Year as it is. With any continuous variable, the model intercept will give you the estimate for that variable at zero. In this case that would be the year of the birth of Jesus! Not ideal. Let's adjust so that the first year we have data is 'Year 1'. That'll make more sense.

Run the following code:

```
Humming$Year <- Humming$Year - min(Humming$Year)
```

Check your output. Cool? Okay - let's build some models to answer our question!

Part 1: Simple linear models

Exercise 1 – Building an LM

Let's first build a simple linear model. Make a model that compares Count with Year (ignore Species and Site for now). Check the powerpoint for the model call you should use, and remember that we express as model as "response ~ predictor". Give your model a sensible name (e.g. LMModel)

Exercise 2 – Model output

Look at the model summary, and check the estimate and p value for "Year". What does this tell us about how hummingbird counts change with year? (We'll look at interpreting this in more detail next week)

Exercise 3 - Variance

How much variance does this model explain? (In other words, if we make a prediction from this model, how likely are we to get it right?). There's an easy way to check (hint run `summary(YourModel)` and look for the R^2). Write it in big here.

… what do you think of that value?

Exercise 4 – Assumptions.

Okay. So far, so not great. What about assumptions? There are three assumptions of linear models, write them here:

1:

2:

3:

Do you think our model violates any of these? How might you check?

HINT: Plotting your model (`plot(YourModel`)) and running a shapiro test might be a good way to go! You'll note a bartlett test doesn't work on this data, don't worry, we can check that later. And ask me (Hannah) if you're wondering why.

Exercise 5 - Check Assumptions

Hopefully you've seen that we've violated two, probably 3 assumptions. Ah.

Let's focus on the assumption of normality first. It doesn't look like our residuals are normally distributed (run `hist(resid(YourModel))` for really conclusive evidence of that).

Why is this? What might we do to fix it?

Part 2 - Generalised Linear Models

You guessed it, we should use a glm!

Exercise 1 - Building a GLM

Our response variable is count data. What distribution does count data take? And therefore what link function will we use in our model? (hint… they're the same)

With this information, build a generalised linear model to consider how abundance correlates with year (again, ignore species and site for now). Again, check the powerpoint for model calls, and name your model something sensible

Exercise 2 - Model output

Look at the summary of the model. Has our understanding changed about how abundance changes with year? (You may notice the actual value for the estimate is pretty different now - all will be revealed next week!)

Exercise 3 - Variance

Are we doing any better of a job at explaining variance now we've adjusted our model? Alas GLMs don't just output an R^2, but it's just a very quick bit of maths to get that value.

Run `summary(Your_GLM)` . At the bottom it gives estimates of null and residual deviance. Write them here.

To get the approx R2, calculate (Null - Residual)/Null. And write it in big here.

How much variance have we explained (as a percentage?)? Is it any better than the simple lm?

Exercise 4 - Check assumptions

So its better, but still explains very little variance. Before we get carried away, let's check our assumptions. Are our residuals homogenous and normally distributed? Checking gets trickier for GLMs - but thankfully a wonderful German Professor wrote an R package that creates residuals plots that are readily interpretable. Let's install the package

```
install.packages("DHARMa")
library(DHARMa)
```

Now, run these two lines to extract and plot residuals:

```
ModResiduals <- simulateResiduals(fittedModel = Your_GLM_Model, plot = F)
plot(ModResiduals)
```

Have a look at the plots. In the first, the triangles should pretty much follow the red line, and the text should be black (red means significant violation of assumptions). In the second, the thick lines should roughly follow the dotted lines, and text should be black.

How are we looking?

Exercise 5 - Now what

Oh my. We are not looking great. Why is that? What are we missing? (Think back to the three assumptions of linear models in Part 1)

Part 3 - Linear Mixed Models

Yes! We must use a mixed model! (Aka hierarchical model, aka mixed effects models)

Exercise 1 - Identifying grouped data

So. The reason to use a mixed model is if your data has groups (that you don't really care about understanding the difference between). What are our groups in this case?

Do our groups have enough 'levels' to be used as random factors?

Exercise 2 - Building a GLMM

Now, code up a glmm. We need the 'g' (for generalised) part because we have count data, so will use a poisson distribution. And the extra 'm' is for mixed. Use the powerpoint to see what call you need, and how to include the random factors.

Exercise 3 - Model output

Have a look at the model output again (by running `summary(Your_GLMM)`). Has our understanding changed?

Exercise 4 - Variance

Not really, but let's be careful before reading into the model output before we're happy with the model fit.

Annoyingly we need *another* package to check the R2 for how much variance this model explains. It's cos it's weird to calculate, but thankfully Tim Newbold down in London made a package. Let's install it.

```
remotes::install_github("timnewbold/StatisticalModels")
library(StatisticalModels)
```

Now run `R2GLMER(Your_GLMM)`. We get two numbers: the R2 of the whole model ("Conditional") and the R2 for just the fixed effects, in our case 'Year' ("Marginal"). What do you think?

On the whole, the model is explaining a lot more variance! But how much is explained by Year relative to the whole? What does that mean for how we interpret things?

Exercise 5 - Check Assumptions

Okay great, we're doing better. But are we violating assumptions? Run the same code as from Exercise 4 of Part 2 (but change the name of the model to Your_GLMM)

How are we looking? Not so bad!

You'll note that it's still not perfect. Welcome to data science, and the real world. Data *often* will always violate some assumptions. But in my books this model is looking a *lot* better than that first lm we built. We should still be cautious about interpreting model output as its not perfect, but I'd be pretty happy to make conclusions from this! Also, the creator of DHARMa notes that when you have big datasets (like here, we have 200 data points) - you're almost always going to get red significance. So look at the plots, and use your judgement.

Next week:

So we've built a model that explains a decent amount of variance and doesn't horrifically violate model assumptions. But what does it actually tell us?! How are the Hummingbirds doing? Stay tuned for next week!