**Assessing Model Fit using AIC and Constructing Generalised Linear Models**

James watt, [James.Watt@ed.ac.uk](mailto:James.Watt@ed.ac.uk)

**Exercise 1: Exploring Model Fit**

Let’s revisit the model with multiple explanatory variables where we looked at how total base saturation and habitat affected soil pH (this is Exercise 5 in the Basic R Stats document). If you recall, our assessment of different statistical models, with and without different explanatory variables including an interaction term, showed that the best statistical model here included habitat and total base saturation, but not an interaction term. Our assessment was based on looking at the variation explained by the model, as measured by the R squared value.

When one has many potential explanatory variables that could be included in an analysis, there are diverse ways of deciding which explanatory variables to include. One approach is to simply include whichever variables seem to be significant using an ANOVA framework or t-tests on the associated parameters (to determine if they are significantly different from zero). However, in large datasets sometimes a variable (or parameter) will be ‘significant’, but it actually does not increase the explanatory value of the model too much. With linear models or general linear models, the R squared value (or better, the adjusted R squared value) can give you an idea of ‘model fit’, or how well the explanatory variables explain the response variable. However, when one moves onto generalised linear models (below), the R squared value can no longer be meaningfully calculated (although there are various ‘pseudo R squared’ metrics). An alternative means for comparing statistical models is an information theoretic approach. The workhorse of such an approach is the Akaike Information Criterion (AIC) and its derivatives (e.g., AICc). The AIC is derived from the probability of the model, i.e. its likelihood. Let’s start off by comparing the various models we generated in the soil pH example with AIC, which can also be used for linear or general linear models.

First, we need to import the data again and create these statistical models.

> soils <- read.csv("Peru\_Soil\_Data.csv", row.names=1, stringsAsFactors=T)

> lm\_Habitat <- lm(Soil\_pH~Habitat,data=soils)

> lm\_TBS <- lm(Soil\_pH~Total\_Base\_Saturation,data=soils)

> lm\_Habitat\_TBS <- lm(Soil\_pH~Habitat + Total\_Base\_Saturation,data=soils)

> lm\_Habitat\_TBS\_Interaction <-lm(Soil\_pH~Habitat\*Total\_Base\_Saturation, data=soils)

Now let’s compare their AIC values

> AIC(lm\_Habitat, lm\_TBS, lm\_Habitat\_TBS, lm\_Habitat\_TBS\_Interaction)

Putting multiple models into the AIC function produces a table with the number of parameters estimated for each model (listed as degrees of freedom) and the respective AIC values. The key thing to remember here is that lower AIC values reflect better models. Thus, the AIC for the model without interaction is the lowest, suggesting it is the best model (similar to the result when consider R-squared values).

Now, how is AIC calculated? It is based on the likelihood, or probability of the model. Higher values mean that the model is more probable or more likely.

AIC = 2k - 2\*log(model likelihood), where k is the number of parameters in the model. Usually the log-likelihood is negative (but not always, as is the case here). Multiplying the log likelihood by a negative number usually makes it positive, but also means that lower values are better than higher values. Lastly, simpler models are generally considered better, so the AIC ‘penalises’ models that have more parameters. In order to validate these calculations, let’s try to calculate the AIC ‘from scratch’ with a log-likelihood value. Note, the log-likelihood also gives you the number of parameters in the model (as degrees of freedom). The best model above model has four parameters, the intercept, the effect of habitat, the effect of base saturation and the error.

> logLik(lm\_Habitat\_TBS) #gives a value of 3.59134

> 2\*4 - 2\*3.59134

An additional point: it is often useful to construct a null model when looking at AIC values, in order to assess if even your simple models are worth considering. This is a quick ‘sniff test’ of whether you have any ability to explain your response variable with the data you have available. To do so in R, you put the number ‘1’ in the explanatory variables part of the model (more on this below).

> lm\_null <- lm(Soil\_pH ~ 1 , data=soils)

> AIC(lm\_null, lm\_Habitat, lm\_TBS, lm\_Habitat\_TBS)

**Exercise 2: Generalised Linear Models (GLMs) with a Poisson Response**

Let’s import another dataset from the same paper, which consists of abundance data for a bunch of different species of the genus Inga in the forest plots from which the soil samples were collected. Those data can also be found online as ‘Supplement\_5\_KGD.csv’at:

<http://www.esapubs.org/archive/mono/M080/009/suppl-1.htm>

They can also be found on Learn as ‘Inga\_abundances.csv’. Let’s start off by importing that file, making sure the rownames match and then combining the two datasets.

> inga <- read.csv("Inga\_abundances.csv",row.names=1,stringsAsFactors=T)

> cbind(rownames(soils),rownames(inga))

Here I have used a function to bind together the rownames of the two dataframes (which are essentially columns, thus cbind rather than rbind, which is for rows). As we can see the rownames match up. Thus we can actually bind these two dataframes together into a single one using the same function.

> combo <- cbind(soils,inga)

Now we can use generalised linear models to assess how the abundance of different species varies with soil characteristics and habitat type. Let’s start with thibaudiana.

> mod1 <- glm(thibaudiana~Habitat,data=combo,family=poisson)

Notice that the structure is identical to that of a linear model, except we have added one argument ‘family’. This allows us to specify what probability distribution we use to model the response variable. As these are count data that cannot be negative, we will use a poisson. Here, and throughout the exercises below, we will use the default link functions. In this case, the default for a poisson distribution is a log-link, which is what prevents us from predicting negative values with our model (there is no logarithm for negative numbers). Now, let’s see what that produced.

> mod1

As with linear models, we get a reminder of our call to the function. We also get the coefficients from the model. As with ANOVA, the coefficients here represent the mean for the first group and the difference between that mean and the mean for other groups. Now you may recall how I emphasised that poisson are used for count data because that do not allow one to predict negative values. Yet, our intercept here (the abundance for floodplain habitat) is given as negative. What gives?? We have to remember that we are using a log link (specifically, a natural log link). If we are to exponentiate that value, we would obtain the mean abundance in floodplain. You can check this.

> exp(-1.946)

> mean(combo$thibaudiana[combo$Habitat=="floodplain"])

You can also make sure this works for uplands

> exp(-1.946+4.518)

> mean(combo$thibaudiana[combo$Habitat=="upland"])

So, what else comes out here? We also get the total degrees of freedom, which is just the number of sample units minus one. Then there is the residual degrees of freedom which is the number of sample units minus the number of parameters we have estimated. We have estimated two parameters in this case. Note that we have not estimated a variance (or error), because poisson assumes the variance to be the same as the mean. Importantly, that is something we also have to check. If the variance is greater than the mean in your data/model, then you have what is called overdispersion. This can quickly be checked here by dividing the residual deviance (also given) by the residual degrees of freedom. If this value is greater than one, then you have overdispersion. Then you need to figure out whether you have too much overdispersion, because if you do, then your model is fitting the data quite poorly and you should not trust the results. In this case, we obtain a ratio greater than two, which indicates a fair bit of overdispersion and that we should consider other models (e.g. a negative binomial response). However, we are going to pretend things are OK and push on. I’ll come back to this later. Now let’s use the summary function on our model object and see what happens.

> summary(mod1)

You will see that R has actually performed some statistical tests on whether the estimated coefficients differ significantly from zero. However, these particular statistical tests assume that your Poisson model was appropriate (i.e., assume a dispersion ‘ratio’ of ~1). This is not the case, so the results of these tests should perhaps not be trusted. In any case, I urge you to use an information theoretic approach to test if this poisson model with habitat fits the data better than a null model. And this brings us onto null models in R. You can create a null model by only estimating the intercept, and this is done as follows:

> mod\_null <- glm(thibaudiana~1,data=combo,family=poisson)

The ‘1’ here is an arbitrary designation for intercept only models in R. There is no sense to the ‘1’. It is just how R ‘speaks’. Now let’s compare the null model to the habitat model with AIC values.

> AIC(mod\_null,mod1)

As we can see the model with habitat is much better than the null model. Thus, even though there are issues with the habitat model, it still fits the data much better than a null model. We know from previous work that the different habitats have different soils, so maybe soil variables provide a better explanation of variation in abundance of Inga thibaudiana than just habitat.

> mod2 <- glm(thibaudiana~Soil\_pH,data=combo,family=poisson)

> summary(mod2)

> AIC(mod\_null,mod1,mod2)

In this case, it looks like habitat provides a better explanation of the variation in abundance than soil pH. What might that mean?

Given that both habitat and soil pH seem to explain variation in abundance of Inga thibaudiana better than a null model, it is sensible to construct multivariate models with both of them and to check how that performs.

> mod3 <- glm(thibaudiana~Habitat + Soil\_pH,data=combo,family=poisson)

> mod4 <- glm(thibaudiana~Habitat \* Soil\_pH,data=combo,family=poisson)

> AIC(mod\_null,mod1,mod2,mod3,mod4)

What we see here is that including both variables results in a better model than including just one. However, including an interaction term actually worsens the model fit (gives a higher AIC value). What this means is that while including the interaction term may have increased the probability of our model slightly (which lowers the log-likelihood), the ‘penalty’ of adding the extra parameter (which results in adding two to the log-likelihood to calculate AIC) outweighed any benefit from including it. Now that we have a good model, let’s conduct a full evaluation of it, to make sure we can trust our results.

> summary(mod3)

The key assumption that is often violated in poisson regression (or poisson GLMs) is that the conditional variance is roughly equal to the conditional mean. We can see that the ratio of the residual deviance to the residual degrees of freedom has decreased, but it is still greater than one. Thus, we still have some overdispersion, which means that the data are more variable than predicted by our model. One approach would be to use a different distribution for the response variable (e.g. a negative binomial). Another approach could be to simplify the data to presence/absence data, which we will do below. Here, we will continue to pretend that everything is OK, to continue going through the motions of how we evaluate the models. Another approach is to plot the predicted versus observed values to see how well they match. Notice that I obtain the predicted values using the ‘fitted’ function rather than the ‘predict’ function, because this quickly gives the predicted values in units of the response variable (after the link function has been applied). To see what I am talking about here, you can check help(predict).

> plot(combo$thibaudiana,fitted(mod3),xlab="Observed",ylab="predicted")

> abline(0,1)

What we can see is that we do not succeed in predicting how abundant Inga thibaudiana can become at high abundances (see points on the far right-hand side of the plot). There may be some other factor that drives Inga thibaudiana to achieve particularly high abundance in certain habitats that we have failed to include in our model. Of course, we also see that we have underpredicted Inga thibaudiana abundance when it is of middle abundance. To further explore this, one could plot the residual values versus our explanatory variables.

> par(mfrow=c(1,2))

> plot(resid(mod3)~Habitat,data=combo)

> plot(resid(mod3)~Soil\_pH,data=combo)

What we see here is a lot of variability around the predicted mean in uplands, but little in floodplain, with the two points where Inga thibaudiana is actually present in floodplain appearing as outliers of sorts (two points between 1 and 2 in floodplain). Meanwhile, we can see that there is a lot of variability in abundance that is not explained by soil pH at low pH values (which is in uplands!). This gives us some insight into the observed overdispersion. Basically, there is something else affecting the abundance of Inga thibaudiana in uplands sites, besides soil pH, that we have failed to account for in our model. Lastly, what we should really do is plot out the raw data and see how our predicted values compare to it, in order to fully understand our analysis.

> combo\_fp <- combo[combo$Habitat=="floodplain",]

> combo\_up <- combo[combo$Habitat=="upland",]

> plot(combo$Soil\_pH,combo$thibaudiana,type="n")

> points(combo\_fp$Soil\_pH,combo\_fp$thibaudiana,pch=21,bg="blue")

> points(combo\_up$Soil\_pH,combo\_up$thibaudiana,pch=24,bg="red")

> points(combo$Soil\_pH,fitted(mod3), col="grey")

This reemphasises what we learned above, that our model is not great for predicting Inga thibaudiana abundance in uplands. But, it is a heck of a lot better than a null model, and we have certainly learned something about what controls Inga thibaudiana abundance. You can check out some other species to see what patterns they show. Some species seem better determine by habitat, others by pH, some by interaction models, etc. Be sure to check for overdispersion though, as most species show overdispersion issues.

**Exercise 3: Generalised Linear Models (GLMs) with a Binomial Response**

As mentioned above, one way to simplify noisy abundance data is to do presence/absence analyses. That is what we will explore here. Of course, many types of data are similar, consisting of yes/no or binary responses, and they would also be suitable for this particular type of binomial statistical model that we will explore here. Inga thibaudiana is always present in upland habitats and nearly always absent in floodplain habitats, so it is not perhaps the most interesting example. Let’s check out Inga auristellae, a species that would have shown exceptional overdispersion issues if you had tried to analyse it with poisson models above.

Let’s first look at the influence of soil pH on whether or not Inga auristellae is present. We must first recode Inga auristellae as a presence/absence variable, which can be done with values of 1 and 0.

> combo$auristellae\_PA <- combo$auristellae

> combo$auristellae\_PA[combo$auristellae\_PA>0] <- 1

> mod1 <- glm(auristellae\_PA~Soil\_pH,data=combo,family=binomial)

> summary(mod1)

A key thing to understand from this summary is the sign of the coefficient for soil pH. As we set the value to one when Inga auristellae is present, this indicates that as soil pH increases, Inga auristellae is less likely to be present (i.e. to switch from a value of 1 to a value of 0). We also see that we have slight overdispersion issues (the simplest binomial model also assumes we have equivalent variance and mean), but the issues are slight, and not nearly as pronounced as if we had used a poisson model. However, in order to see if soil pH significantly affects the presence vs. absence of Inga auristellae, we should compare this model to a null model.

> mod\_null <- glm(auristellae\_PA~1,data=combo,family=binomial)

> AIC(mod\_null, mod1)

As we can see, the soil pH model is more than two AIC units below the null model, indicating that soil pH does seem to affect Inga auristellae presence. Now, let’s compare this to some more models, with habitat and other variables.

> mod2 <- glm(auristellae\_PA~ Habitat,data=combo,family=binomial)

> mod3 <- glm(auristellae\_PA~Soil\_pH + Habitat,data=combo,family=binomial)

> mod4 <- glm(auristellae\_PA~Soil\_pH \* Habitat,data=combo,family=binomial)

> AIC(mod\_null, mod1, mod2, mod3, mod4)

We see that the model with just habitat is almost as good at explaining presence vs. absence of Inga auristellae as soil pH. As the AIC values differ so little, we can consider these two models to have equivalent explanatory power. What is clear is that if we use both variables, we actually produce a worse model, which is due to the penalty for estimating additional parameters (i.e. for adding additional explanatory variables). Essentially, these two variables are communicating the same information and are redundant on each other.

However, as the AIC for soil pH is slightly lower, let’s check out validation for that model.

> plot(auristellae\_PA~Soil\_pH,data=combo,pch=16)

> points(combo$Soil\_pH,fitted(mod1), col="grey")

This graph may seem perplexing at first as the predicted values are never zero or one. A key thing to remember is that the predicted values in this binomial regression are actually the proportion of times the species would be predicted as present for a given soil pH if there were many ‘samples’ or ‘coin flips’ for that soil pH. However, as we only ‘flipped the coin’ once at each site, the observed proportions are either zero or one. This particular case of a binomial generalised linear model (the Bernoulli) is essentially identical to logistic regression (about which you can find lots of info online).

In fact, modelling a binomial response is much more flexible than this and can cover many kinds of data. For example, we might count the species richness in each plot and assess how Inga species richness varies between habitats and with soil properties. Inga species richness essentially represents the outcome of a yes/no process for each species present in the dataset. We could simply sum up all of the ‘yes’ values for each site and model that total number as a binomial response. The formatting of the statistical model differs slightly in this instance. See help(glm) to understand how. Also, in terms of further reading, if you want to try negative binomial models to deal with overdispersion issues, you can install the MASS library and check the glm.nb() function. We will cover how to install optional libraries next week.