Logistic Regression in R

```
library(ggplot2)
library(dominanceanalysis)
library(ggpubr)
library(MuMIn)
library(dplyr)
tropicbird$pres <- factor(tropicbird$pres, labels = c("absent", "present"))
```

Overview

The goal for today is to explore another popular use of a conditional distribution other than Normal. Most real world uses of the `glm` function either use a `binomial` or a `poisson` family. We looked at Poisson models last week (though we found that a negative binomial model was more appropriate - this is often the case). Today we will focus on models with binomial conditional distributions. This material is discussed in Sections 4.1-4.3 of James et al. (2021).

Binomial GLM

A binomial GLM is usually what is called a *logistic regression model*. The terminology comes from the typical link function, the logistic function. The domain of the logistic function is the interval $(0, 1)$ so it makes sense for a link when modeling probabilities. Given a predicted probability $\hat{p}$, the fraction of trials that result in success follows a binomial distribution, so when predicting probabilities, the appropriate conditional distribution is binomial.

The data

There's a nice data set for doing logistic regression in the `dominanceanalysis` package. Install and load that library, and take a look at the `tropicbird` data.

The following description of the dataset is from Soares (2020).

"We explore the distribution of a tropical native bird species inhabiting a small oceanic island. Since human occupation, the island's forests have disappeared from the flat lowland areas, located closer to the coastline. Nowadays, these areas are considered anthropogenic areas, which include cities, agricultural areas (e.g., horticultures), savannas, and oil-palm monocultures.

We use the tropicbird dataset, which is a collection of points distributed across the island (Soares, 2017). In each of these points, a 10-minute count was carried out to record the species presence (assuming 1 if the species was present, or 0 if it was absent). The species' presence/absence is the binary response variable (i.e., dependent variable). Additionally, we characterized all sampled points for each of the following environmental variables (i.e., independent variables, or predictors):

- remoteness (rem) is an index that represents the difficulty of movement through the landscape, with the highest values corresponding to the most remote areas;
- land use (land) is an index that represents the land-use intensification, with the highest values corresponding to the more humanized areas (e.g., cities, agricultural areas, horticultures, oil-palm monocultures);

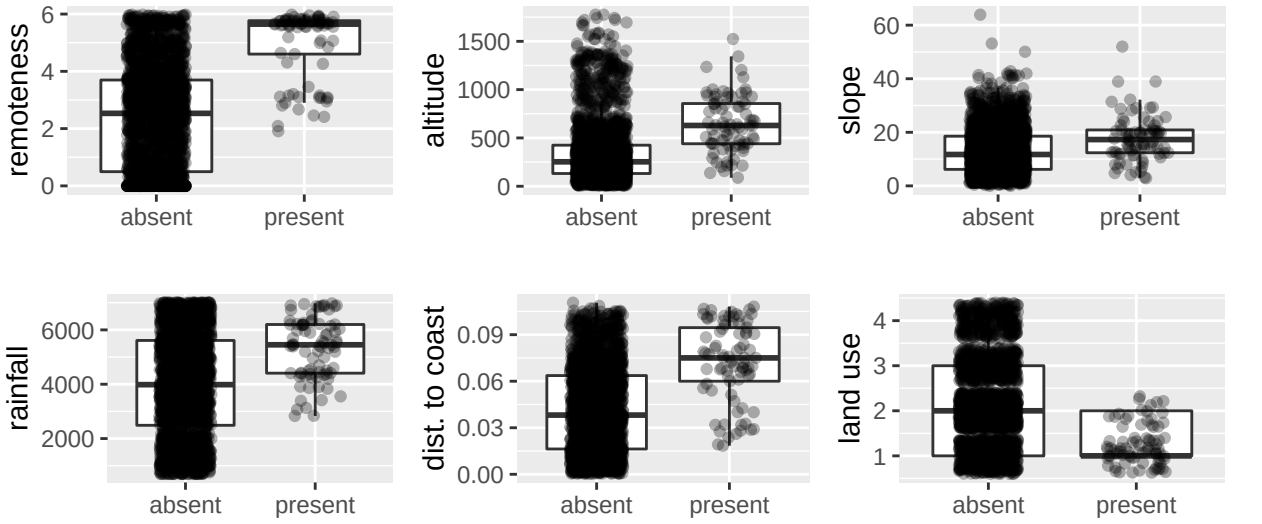
- altitude (alt) is a continuous variable, with the highest values corresponding to the higher altitude areas;
- slope (slo) is a continuous variable, with the highest values corresponding to the steepest areas;
- rainfall (rain) is a continuous variable, with the highest values corresponding to the rainy wet areas;
- distance to the coast (coast) is the minimum linear distance between each point and the coast line, with the highest values corresponding to the points further away from the coastline.

Please note that in this dataset there are no false negatives, hence the bird was always recorded if present. Also, the dataset has no missing values, so it is already prepared for the analysis."

The relationship between each predictor and the response can be visualized by a series of boxplots.

```
a <- ggplot(tropicbird, aes(pres, rem))+
  geom_boxplot(outlier.shape = NA) + geom_jitter(width = 0.2, alpha=0.3)+
  labs(x="", y="remoteness")
b <- ggplot(tropicbird, aes(pres, alt))+
  geom_boxplot(outlier.shape = NA) + geom_jitter(width = 0.2, alpha=0.3)+
  labs(x="", y="altitude")
c <- ggplot(tropicbird, aes(pres, slo))+
  geom_boxplot(outlier.shape = NA) + geom_jitter(width = 0.2, alpha=0.3)+
  labs(x="", y="slope")
d <- ggplot(tropicbird, aes(pres, rain))+
  geom_boxplot(outlier.shape = NA) + geom_jitter(width = 0.2, alpha=0.3)+
  labs(x="", y="rainfall")
e <- ggplot(tropicbird, aes(pres, coast))+
  geom_boxplot(outlier.shape = NA) + geom_jitter(width = 0.2, alpha=0.3)+
  labs(x="", y="dist. to coast")
f <- ggplot(tropicbird, aes(pres, land))+
  geom_boxplot(outlier.shape = NA) + geom_jitter(width = 0.2, alpha=0.3)+
  labs(x="", y="land use")

ggarrange(a,b,c,d,e,f, nrow=2, ncol=3)
```



Model predictions

One of the great things about having a model is that it can be used to predict. In the case of a logistic regression model, we can use the generated probabilities to make presence-absence predictions, which are

especially easy to use because they are either right or wrong.

A one predictor model

Start with a model that only uses remoteness to predict the probability of finding a bird.

```
modpres1 <- glm(pres~rem, data=tropicbird, family=binomial)
summary(modpres1)
```

```
##
## Call:
## glm(formula = pres ~ rem, family = binomial, data = tropicbird)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.7562  -0.2241  -0.1164  -0.0357   3.3419
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -7.6813     0.5626 -13.653   <2e-16 ***
## rem           1.1000     0.1121   9.816   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 708.08  on 2397  degrees of freedom
## Residual deviance: 529.46  on 2396  degrees of freedom
## AIC: 533.46
##
## Number of Fisher Scoring iterations: 8
```

How good is this model? We can look at the p-value for `rem` and see that it is very small and conclude that remoteness is correlated with presence. We will discuss the exact meaning of value 1.1 of the coefficient, but it measures the magnitude of the increase. In particular, since the coefficient is positive, more remoteness is associated with higher probabilities of bird presence.

Another thing that we can do is compare this model to another model using AIC. What other model can we compare to? A model with no predictors will allow us to quantify how useful remoteness is on its own as a predictor of bird presence.

```
modpres0 <- glm(pres~1, data=tropicbird, family=binomial)
model.sel(modpres1, modpres0)
```

```
## Model selection table
##          (Intrc) rem          family df   logLik  AICc  delta weight
## modpres1  -7.681 1.1 binomial(logit)  2 -264.732 533.5   0.00      1
## modpres0  -3.354     binomial(logit)  1 -354.040 710.1 176.61      0
## Models ranked by AICc(x)
```

We can also see how well this model does by comparing predictions to data. First, we calculate the probabilities using the model. Let's do this in three different ways. First, we use the model coefficients directly, and the `plogis` function to invert the logit link. Second, we use the `fitted` command to pull the predictions out of the model object. Third, we use the `predict` command. The `type="response"` is equivalent to the `plogis` command in the first computation. It tells R that we want the predictions on the scale of the response variable. The advantage to using `predict` is that if we want to make predictions on data other than the data that was used to build the model, we just give it appropriate `newdata`.

```
probs1 <- plogis(coef(modpres1)[1]+coef(modpres1)[2]*tropicbird$rem)
probs1a <- fitted(modpres1)
probs1b <- predict(modpres1, type="response")
```

To check that the above three vectors of predictions are the same, we can count the number of times that they differ with the following code. We round to eight decimal places because the numbers produced by the explicit calculation and `plogis` in `probs1` aren't exactly the same as those from `fitted` or `predict` because of rounding issues at the level of machine precision.

```
sum(round(probs1,8)!=round(probs1a,8))
```

```
## [1] 0
```

```
sum(probs1a!=probs1b)
```

```
## [1] 0
```

One of the ways that we can use the `predict` function with new data is to make a common sort of graph to depict this logistic regression model. First, we make a grid of predictor values.

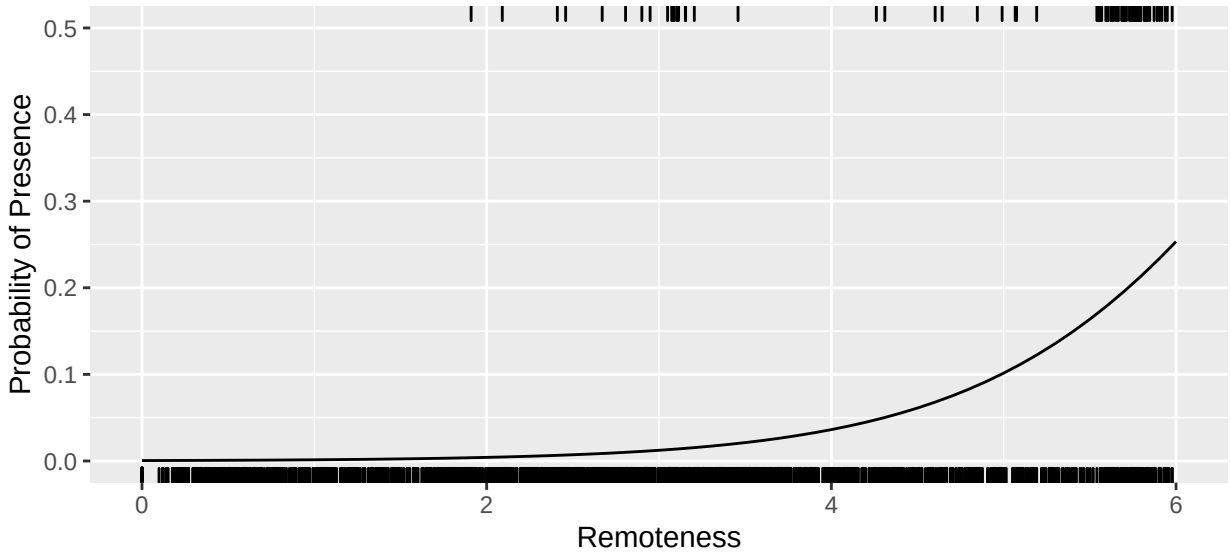
```
newdata <- data.frame(rem=seq(0,6,0.1))
```

Then we use the model to predict responses.

```
newdata$presprob <-predict(modpres1, newdata, type="response")
```

Now we can plot the relationship between predictors and predictions. In addition, this plot adds "rugs" of points on the bottom and the top of the graph to indicate the values of remoteness for sites where birds were in fact found or not.

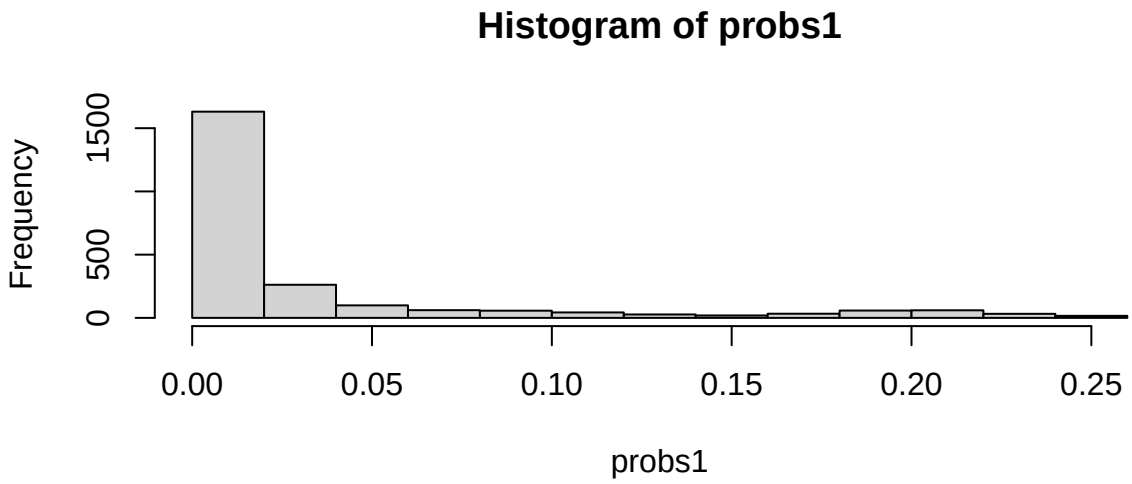
```
ggplot(newdata, aes(rem, presprob)) +
  geom_line() + ylim(0,0.5) + labs(x="Remoteness", y="Probability of Presence") +
  geom_rug(data=tropicbird[tropicbird$pres=="absent",],
           aes(y=as.numeric(pres)), sides = "b") +
  geom_rug(data=tropicbird[tropicbird$pres=="present",],
           aes(y=as.numeric(pres)), sides = "t")
```



Setting a threshold

Now suppose we want to use this model to decide where to go look for birds. We can set a threshold probability and go to the sites where the predicted probabilities are above that probability. Deciding where to set that threshold is a matter of balancing our desire to find a bird with the effort involved in going to look for one that is wasted if there are no birds present. We can use the distribution of the predicted probabilities to guide our decision.

```
hist(probs1)
```



It looks like birds are rather rare, and the model never predicts a probability higher than 25%. Let's set our threshold at 10% - so if the model predicts a greater than 10% chance of finding a bird, we will flag that site for a visit.

```
pred1 <- ifelse(probs1 > 0.1, "yes", "no")
```

We can compare these predictions to reality: were there birds present at the sites we flagged to go and visit?

```
table(pred1,tropicbird$pres)
```

```
##
## pred1 absent present
##   no    2087      23
##   yes    230      58
```

It seems that we can expect to be disappointed and not find any birds 230 out of 288 times, or about 80% of the time, and we'll miss out on 23 of 81 or about 28% of the sites where birds were actually present. We could adjust our threshold probability from 10% to get different results here depending on how we value missing bird sites against spending time looking for nonexistent birds. Alternatively, we could say that we have created a test for bird presence with an estimated sensitivity of 72% and an estimated specificity of 99%.

A model with more predictors

We can follow a very similar process after making a more complete model using all of the predictors.

```
modpres <- glm(pres~rem+land+alt+slo+rain+coast,
               data=tropicbird, family=binomial)
summary(modpres)
```

```
##
## Call:
## glm(formula = pres ~ rem + land + alt + slo + rain + coast, family = binomial,
##     data = tropicbird)
```

```
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2609  -0.2154  -0.0976  -0.0324   3.2903
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.230e+01  1.416e+00  -8.688  < 2e-16 ***
## rem          6.212e-01  1.710e-01   3.633  0.00028 ***
## land         4.330e-01  3.248e-01   1.333  0.18247
## alt          1.536e-03  4.034e-04   3.808  0.00014 ***
## slo          1.068e-02  1.378e-02   0.775  0.43815
## rain         6.041e-04  1.450e-04   4.166 3.09e-05 ***
## coast        3.251e+01  7.363e+00   4.415 1.01e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 708.08  on 2397  degrees of freedom
## Residual deviance: 490.61  on 2391  degrees of freedom
## AIC: 504.61
##
## Number of Fisher Scoring iterations: 8
```

(1) Evaluate `modpres` as a bird locating model using p-values as we did with `modpres1` above. Which predictors appear significant?

Using the `dredge` function from `MuMIn`

We can also use AIC to compare models. Instead of explicitly computing all the models using only subsets of the predictors, we can use `dredge` from the `MuMIn` package. In order to do this, we need to rebuild the model using `na.action=na.fail` because AIC can only compare models built from the same data, and hence missing data causes problems. The output of `dredge` is a table listing all possible models using variables used in `modpres`.

```
modpres <- glm(pres~rem+land+alt+slo+rain+coast,
               data=tropicbird, family=binomial, na.action = na.fail)
model_table <- dredge(modpres)
```

There are $2^6 = 64$ such models, so we'll only look at the first 10 lines of the table.

```
round(model_table[1:10,c(1:7,10:12)],3)
```

```
##    (Intercept)   alt  coast   land  rain   rem   slo    AICc  delta weight
## 28     -10.894 0.002 30.777     NA 0.001 0.525    NA 502.804  0.000  0.374
## 32     -12.129 0.002 32.602  0.413 0.001 0.633    NA 503.243  0.438  0.300
## 60     -10.995 0.001 30.643     NA 0.001 0.510 0.009 504.348  1.543  0.173
## 64     -12.302 0.002 32.509  0.433 0.001 0.621 0.011 504.657  1.853  0.148
## 27      -9.710    NA 31.361     NA 0.000 0.684    NA 515.016 12.211  0.001
## 59      -9.923    NA 31.307     NA 0.000 0.651 0.016 515.691 12.886  0.001
## 12     -11.953 0.002 47.616     NA 0.001    NA    NA 515.817 13.013  0.001
## 44     -12.050 0.002 46.293     NA 0.001    NA 0.019 515.859 13.054  0.001
## 31     -10.577    NA 33.185  0.290 0.000 0.751    NA 516.289 13.484  0.000
## 16     -10.998 0.002 44.463 -0.266 0.001    NA    NA 516.676 13.872  0.000
```

Normalizing predictors

This table tells us that the model that omits `land` and `slo` has the lowest AICc. It also gives us the slopes of the coefficients for these predictors. Unfortunately, we can't use these coefficients to compare magnitudes of the effects, since the scales of the predictors are all different. We can fix this by normalizing the predictors. The `scale` function can be used to modify variables so that all of our predictors have mean zero and standard deviation one. The `mutate_at` function can be used to apply the function to the appropriate subset of columns of the data.

```
scaled_tropicbird <- tropicbird %>%
  mutate_at(c("rem", "alt", "slo", "rain", "coast", "land"), scale)
```

Now when we look at coefficients, they can be more directly compared to one another. Notice that AIC does not change.

```
scaled_modpres <- glm(pres~rem+land+alt+slo+rain+coast,
               data=scaled_tropicbird, family=binomial, na.action = na.fail)
scaled_model_table <- dredge(scaled_modpres)
round(scaled_model_table[1:10,c(1:7,10:12)],3)
```

```
##    (Intercept)   alt coast   land  rain   rem   slo    AICc  delta weight
## 28      -5.364 0.497 0.876     NA 1.105 0.984    NA 502.804  0.000  0.374
## 32      -5.347 0.514 0.928  0.385 1.113 1.186    NA 503.243  0.438  0.300
## 60      -5.362 0.488 0.873     NA 1.119 0.956 0.081 504.348  1.543  0.173
## 64      -5.343 0.505 0.926  0.403 1.131 1.164 0.092 504.657  1.853  0.148
## 27      -5.174    NA 0.893     NA 0.703 1.281    NA 515.016 12.211  0.001
## 59      -5.176    NA 0.891     NA 0.741 1.220 0.137 515.691 12.886  0.001
## 12      -5.282 0.680 1.356     NA 1.810    NA    NA 515.817 13.013  0.001
## 44      -5.266 0.646 1.318     NA 1.791    NA 0.160 515.859 13.054  0.001
## 31      -5.155    NA 0.945  0.270 0.716 1.408    NA 516.289 13.484  0.000
## 16      -5.301 0.644 1.266 -0.248 1.699    NA    NA 516.676 13.872  0.000
```

(2) Discuss the relative effects of the significant predictors on the model's probability of finding a bird.

Predictions from the full model

Two things to note: First, when calculating probabilities, `probs1a` and `probs1b` will translate with almost no modification, but to translate `probs1`, you will have to add the rest of the terms into the linear part of the predictor. Second, making a plot of the model with six predictors is more complicated.

Making a graph using the resulting model is more complicated because there are more variables that need to be provided as new data and represented in the graph. To make a grid of predictor values by building a sequence of points filling out all possible combinations across the ranges of each variable would lead to an absolutely huge `newdata`. Instead we can generate random values within those ranges.

To save some typing, we can define a function `pts` to randomly generate 10000 values within the range of a given variable.

```
pts <- function(x){runif(10000,min(x),max(x))}
```

Then we can use this function to randomly choose 10000 points within the range of the values of the data.

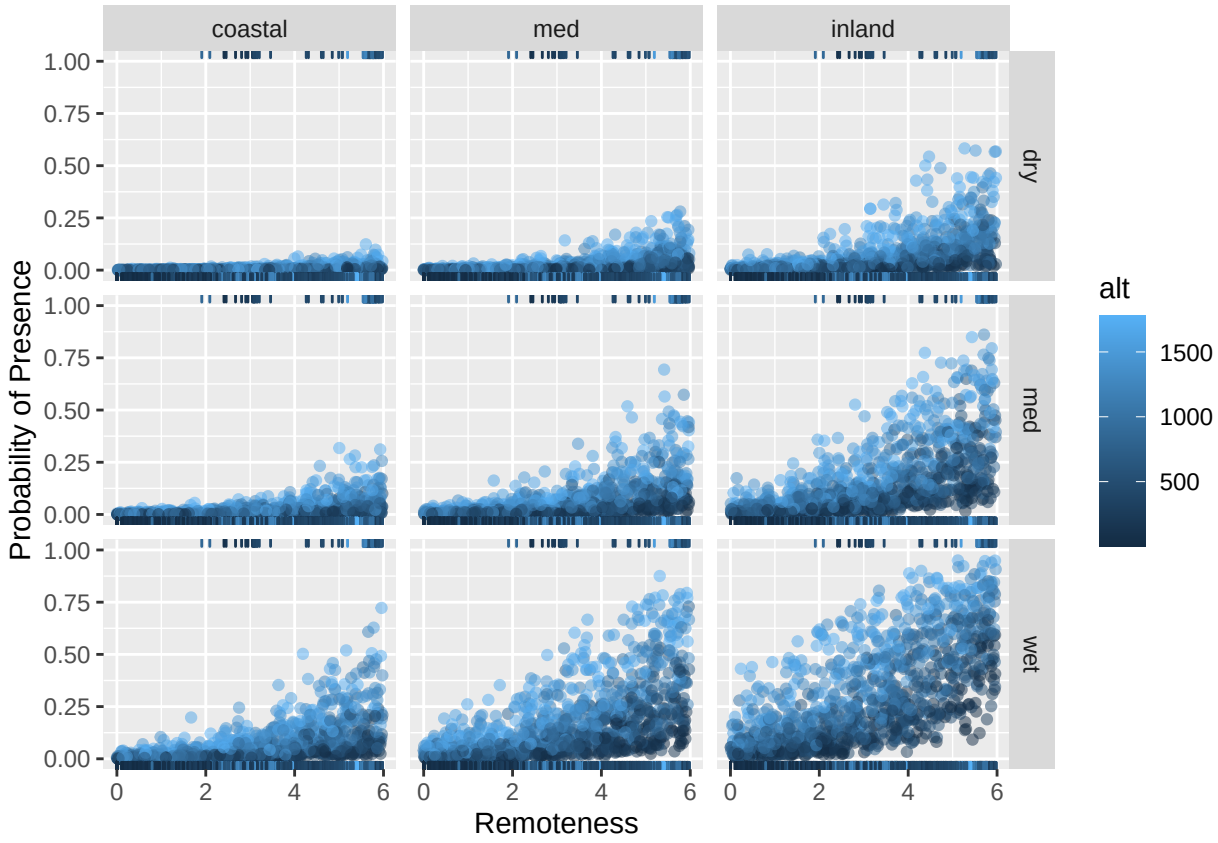
```
newdata <- with(tropicbird, data.frame(rem=pts(rem),
                                       land=pts(land),
                                       alt=pts(alt),
                                       slo=pts(slo),
                                       rain=pts(rain),
                                       coast=pts(coast)))
```

We have six predictors, so we will have to use some creativity and discretion to display appropriate information. Based on the output of `summary(modpres)` we can not bother to display `land` and `slo` since they do not register as statistically significant. We'll keep `rem` as the x variable, and can assign `alt` to color. Faceting only works for categorical variables, but we can use the `cut_number` function to divide `coast` and `rain` into three intervals.

```
newdata$coast_discrete <- cut_number(newdata$coast,3,
                                     labels=c("coastal", "med", "inland"))
newdata$rain_discrete <- cut_number(newdata$rain,3,
                                    labels=c("dry", "med", "wet"))
```

As before, we finally add predictions to the new data and make the plot.

```
newdata$pres <-predict(modpres, newdata, type="response")
ggplot(tropicbird, aes(rem, pres, color=alt)) +
  geom_point(data=newdata, alpha=0.5)+ ylim(0,1) +
  labs(x="Remoteness", y="Probability of Presence") +
  geom_rug(data=tropicbird[tropicbird$pres=="absent",],
           aes(y=as.numeric(pres)), sides = "b") +
  geom_rug(data=tropicbird[tropicbird$pres=="present",],
           aes(y=as.numeric(pres)), sides = "t") +
  facet_grid(rain_discrete~coast_discrete)
```



(3) Remix this graphic. Some things you might try are changing which variable is represented by which aesthetic or by facets, or the number of splits for the faceting variable, or the color palette. You might also experiment with using both x and y for predictors and representing the prediction for presence by color. (If you do this, the `geom_rug` should be removed or repurposed.) What story does your graphic tell about this model?

Coefficients and log-odds

When we looked at linear models or models with a log link, the coefficients had interpretations in terms of rates of change in the response as the various predictors changed. With a logit link, which is the default for binomial glms, there is a similar interpretation but it is a bit more subtle.

The logit function, probability and odds

The formula for the standard logistic function is

$$\sigma(t) = \frac{e^t}{1 + e^t}$$

The inverse function is called the *logit*, and a little algebra tells us that it is defined by the formula

$$\sigma^{-1}(p) = \log\left(\frac{p}{1-p}\right)$$

In a binomial glm, we consider a logistic transform of a linear predictor to predict the probability of a binary response: $p = \sigma(\beta_0 + \beta_1 x)$. This means that

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x$$

Equivalently,

$$\frac{p}{1-p} = \exp\left(\beta_0 + \beta_1 x\right)$$

For a probability p, the expression $\frac{p}{1-p}$ is called the *odds*. For example if the probability of some event is $p = 0.75$ then the odds of that event are

$$\begin{aligned}\frac{0.75}{1-0.75} &= \frac{0.75}{0.25} \\ &= \frac{3}{1}\end{aligned}$$

which are generally referred to as *3 to 1 odds*, meaning that the probability that the event happens is three times greater than the probability that it doesn't happen.

Resource selection functions

What all of this means is that the coefficients of a logistic regression model can be interpreted as fractional changes in the odds of the event whose probability is being predicted, much as the coefficients in a Poisson or negative binomial regression model (with a log link) can be interpreted as fractional changes in the count being predicted.

In a context where the response variable is a presence/ absence, such as in the `tropicbird` data, we can consider each of the predictor variables as *resources* that contribute to this presence or absence, and the coefficients are then indicators of the degree to which these resources are selected for or against by whatever is present or absent.

For example, in `modpres1`, the predictor remoteness (`rem`) has coefficient 1.1. Since this is positive, we can say that remoteness is selected *for* in determining the presence of birds. If we want to be more thorough, we can exponentiate coefficients and interpret them as factors by which the odds change in relation to changes in the resource. Additionally, computing confidence intervals for the coefficients can add to the analysis.

```
exp(coef(modpres1))
```

```
##  (Intercept)          rem
## 0.0004613846 3.0043055168
```

```
exp(confint(modpres1))
```

```
##                    2.5 %      97.5 %
## (Intercept) 0.0001406452 0.001284615
## rem         2.4406327493 3.791457312
```

Thus we can say that a 95% confidence interval for the expected factor by which the odds of a bird being present changes in response to a one unit increase in remoteness is (2.44,3.79). Our point estimate for this factor is $\exp(1.1) \approx 3.00$.

(4) Analyze `scaled_modpres` as a resource selection function. Which resources are significant? What are the point estimates and 95% confidence intervals for the relative effects of each resource, and what do these numbers mean in terms of predicted odds of finding a bird?

References

James, G., D. Witten, T. Hastie, and R. Tibshirani. 2021. *An Introduction to Statistical Learning: With Applications in R*. Springer Texts in Statistics. Springer US. https://books.google.com/books?id=g5gezgEACAAJ.

Soares, Filipa Coutinho. 2020. "Exploring Predictors' Importance in Binomial Logistic Regressions." https://cran.r-project.org/web/packages/dominanceanalysis/vignettes/da-logistic-regression.html.