

Lab 14 – Model Selection and Multimodel Inference

FANR 6750

Richard Chandler and Bob Cooper

① MODEL FITTING

② MODEL SELECTION

③ MULTI-MODEL INFERENCE

SWISS DATA

```
swissData <- read.csv("swissData.csv")
head(swissData, n=11)
```

##	elevation	forest	water	sppRichness
## 1	450	3	No	35
## 2	450	21	No	51
## 3	1050	32	No	46
## 4	950	9	Yes	31
## 5	1150	35	Yes	50
## 6	550	2	No	43
## 7	750	6	No	37
## 8	650	60	Yes	47
## 9	550	5	Yes	37
## 10	550	13	No	43
## 11	1150	50	No	52

FOUR LINEAR MODELS

```
fm1 <- lm(sppRichness ~ forest, data=swissData)
fm2 <- lm(sppRichness ~ elevation, data=swissData)
fm3 <- lm(sppRichness ~ forest + elevation +
          water, data=swissData)
fm4 <- lm(sppRichness ~ forest + elevation +
          I(elevation^2) + water, data=swissData)
```

```
summary(fm4)

##
## Call:
## lm(formula = sppRichness ~ forest + elevation + I(elevation^2) +
##     water, data = swissData)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.314  -3.205  -0.377   3.334  15.082
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.518e+01  1.286e+00  35.137 < 2e-16 ***
## forest       2.311e-01  1.276e-02  18.111 < 2e-16 ***
## elevation    -1.016e-02  2.572e-03  -3.951  0.0001 ***
## I(elevation^2) 6.103e-08  9.661e-07  0.063  0.9497
## waterYes     -3.013e+00  6.821e-01  -4.418 1.46e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.954 on 262 degrees of freedom
## Multiple R-squared:  0.7929, Adjusted R-squared:  0.7897
## F-statistic: 250.8 on 4 and 262 DF, p-value: < 2.2e-16
```

```
summary.aov(fm4)

##              Df Sum Sq Mean Sq F value    Pr(>F)
## forest         1  13311   13311   542.40 < 2e-16 ***
## elevation       1  10820   10820   440.89 < 2e-16 ***
## I(elevation^2)  1     7      7     0.27  0.604
## water          1    479     479   19.52 1.46e-05 ***
## Residuals      262   6430      25
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We could compute AIC using the equation $AIC = n \log(RSS/n) + 2K$, where RSS is the residual sum-of-squares.

However, we will use the more general formula: $AIC = -2\mathcal{L}(\hat{\theta}; \mathbf{y}) + 2K$.

OUTLINE

1 MODEL FITTING

2 MODEL SELECTION

3 MULTI-MODEL INFERENCE

COMPUTE AIC FOR EACH MODEL

Sample size

```
n <- nrow(swissData)
```

log-likelihood for each model

```
logL <- c(logLik(fm1), logLik(fm2), logLik(fm3), logLik(fm4))
```

Number of parameters

```
K <- c(3, 3, 5, 6)
```

AIC

```
AIC <- -2*logL + 2*K
```

ΔAIC

```
delta <- AIC - min(AIC)
```

AIC Weights

```
w <- exp(-0.5*delta)/sum(exp(-0.5*delta))
```

Put vectors in data.frame

```
ms <- data.frame(logL, K, AIC, delta, w)
rownames(ms) <- c("fm1", "fm2", "fm3", "fm4")
round(ms, digits=2)
```

```
##      logL K      AIC delta  w
## fm1 -939.03 3 1884.06 266.90 0.00
## fm2 -934.07 3 1874.15 256.99 0.00
## fm3 -803.58 5 1617.16   0.00 0.73
## fm4 -803.58 6 1619.15   2.00 0.27
```

Sort data.frame based on AIC values

```
ms <- ms[order(ms$AIC),]
round(ms, digits=2)
```

```
##      logL K      AIC delta  w
## fm3 -803.58 5 1617.16   0.00 0.73
## fm4 -803.58 6 1619.15   2.00 0.27
## fm2 -934.07 3 1874.15 256.99 0.00
## fm1 -939.03 3 1884.06 266.90 0.00
```

```
AIC(fm1, fm2, fm3, fm4)
```

```
##      df      AIC
## fm1   3 1884.057
## fm2   3 1874.146
## fm3   5 1617.157
## fm4   6 1619.153
```

Notes

- If we had used the residual sums-of-squares instead of the log-likelihoods, the AIC values would have been different, but the ΔAIC values would have been the same
- Either approach is fine with linear models, but log-likelihoods must be used with GLMs and other models fit using maximum likelihood

OUTLINE

1 MODEL FITTING

2 MODEL SELECTION

3 MULTI-MODEL INFERENCE

MODEL-SPECIFIC PREDICTIONS

Expected number of species at 1000m elevation, 25% forest cover, and no water, **for each model**

```
predData1 <- data.frame(elevation=1000, forest=25, water="No")
```

```
E1 <- predict(fm1, newdata=predData1, type="response")
as.numeric(E1) # remove names (optional)
```

```
## [1] 37.90222
```

```
E2 <- predict(fm2, newdata=predData1, type="response")
as.numeric(E2)
```

```
## [1] 42.53368
```

```
E3 <- predict(fm3, newdata=predData1, type="response")
as.numeric(E3)
```

```
## [1] 40.88604
```

```
E4 <- predict(fm4, newdata=predData1, type="response")
as.numeric(E4)
```

```
## [1] 40.86092
```

MODEL-AVERAGED PREDICTION

Expected number of species at 1000m, 25% forest cover, and no water, **averaged over all 4 models**

```
E1*w[1] + E2*w[2] + E3*w[3] + E4*w[4]
```

```
##          1
## 40.87927
```

MODEL-AVERAGED REGRESSION LINES

Predict species richness over range of forest cover, for each model

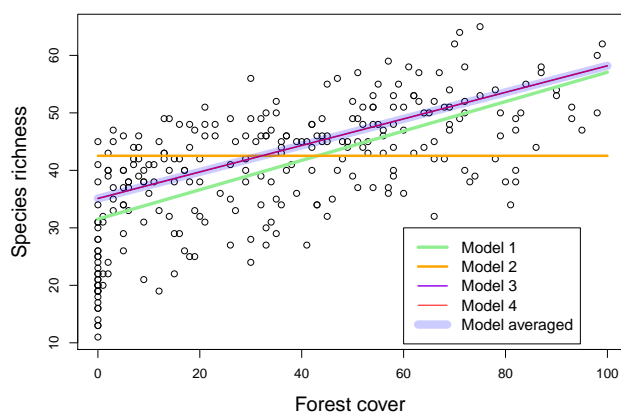
```
predData2 <- data.frame(forest=seq(0, 100, length=50),
                        elevation=1000, water="No")
E1 <- predict(fm1, newdata=predData2)
E2 <- predict(fm2, newdata=predData2)
E3 <- predict(fm3, newdata=predData2)
E4 <- predict(fm4, newdata=predData2)
Emat <- cbind(E1, E2, E3, E4)
```

How do we model-average these vectors?

```
Evec <- Emat %*% w
```

MODEL-AVERAGED REGRESSION LINE

```
plot(sppRichness~forest, data=swissData, xlab="Forest cover", ylab="Species richness", cex.lab=1.5)
lines(E1 ~ forest, predData2, col="lightgreen", lwd=4)
lines(E2 ~ forest, predData2, col="orange", lwd=3)
lines(E3 ~ forest, predData2, col="purple", lwd=2)
lines(E4 ~ forest, predData2, col="red", lwd=1)
lines(Evec ~ forest, predData2, col=rgb(0,0,1,0.2), lwd=10)
legend(60, 30, c("Model 1", "Model 2", "Model 3", "Model 4", "Model averaged"), lty=1, cex=1.2,
      lwd=c(4,3,2,1,10), col=c("lightgreen", "orange", "purple", "red", rgb(0,0,1,0.2)))
```



ASSIGNMENT

Use the jayData from lab 12 to do the following:

- (1) Fit four linear models of jay abundance. Include elevation all four models. Include at least one interaction, and one quadratic term for elevation.
- (2) Create AIC table by hand, not using R's AIC function
- (3) Model-average regression lines of jay abundance and elevation. Plot the averaged regression line along with the regression lines from each model.
- (4) Bonus problem with 10 extra points: Create a map showing the model-averaged estimates of jay abundance on Santa Cruz Island. Hint: use predict with cruzData supplied as the newdata argument.