Model Selection I

Bob O'Hara

March 8, 2019

This Week: Model selection

This week you will:

- find out why model selection is needed
- be able to use AIC to compare models
- be able to compare hypotheses with F tests

Why select models?

Simulate some data:

- 100 points,
- up to 90 explanatory variables
- First variable explains ~1% of data, rest nothing

```r
set.seed(25)
N <- 100; P <- 90
x <- matrix(rnorm(N*P), nrow=N)
mu <- 0.1*x[,1] # true R^2 = 0.1^2/(0.1^2 + 1) = 1%
y <- rnorm(N, mu)
```

Why select models?

Use 2, 3, 4. . . explanatory variables (`sapply()` loops over 2:P)

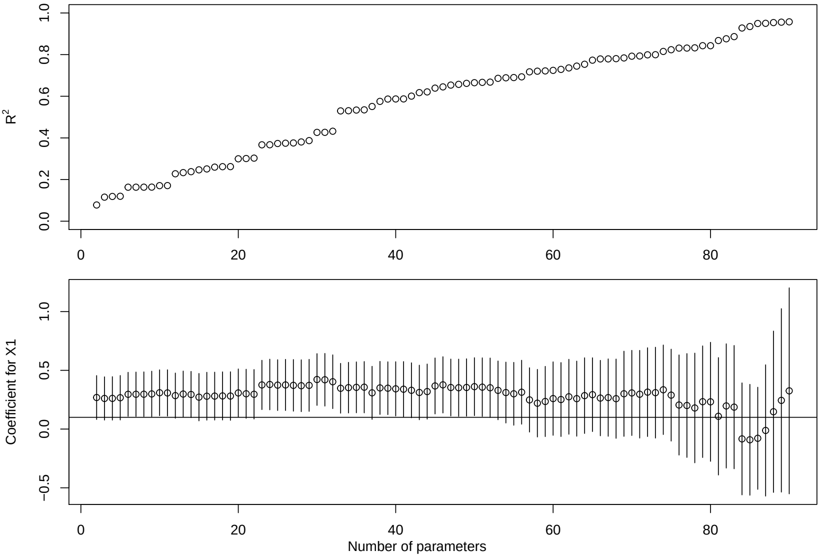
```r
R2 <- sapply(2:P, function(pp, XX, Y) {
  mod <- lm(y ~ XX[,1:pp]) # fit the model
# return coefficient, conf. int. R^2
  c(coef(mod)["XX[, 1:pp]1"],
    confint(mod)["XX[, 1:pp]1",],
    summary(mod)$r.squared)
}, XX=x, Y=y)
```

Why select models?

Plot estimate & R^2

```r
par(mfrow=c(2,1), mar=c(2,4.1,1,1), oma=c(2,0,0,0))
plot(2:P, R2[4,], ylab=expression(R^2), ylim=c(0,1))
plot(2:P, R2[1,], ylim=range(R2[2:3,]),
     ylab="Coefficient for X1")
segments(2:P, R2[2,], 2:P, R2[3,])
abline(h=0.1)
mtext("Number of parameters", 1, outer=TRUE)
```

Plot estimate & R^2



What Happens

We could fit a model with every covariate in it

When we add more variables,

- R^2 increases
- parameter estimates get less precise
- interpretation can become more difficult

So we only want to important variables

Two types of problem: two solutions

Testing a specific hypothesis

- confirmatory

Finding a good model

- exploratory

Question: Which of theseis exploratory & which confirmatory?

Candidate Gene Approach

- does BRCA1 affect the probability of getting cancer?

GWAS

- which of these 30 000 SNPs explains the probability of getting cancer?

Hypothesis Testing

Hypothesis Testing is asymmetrical.

We ask

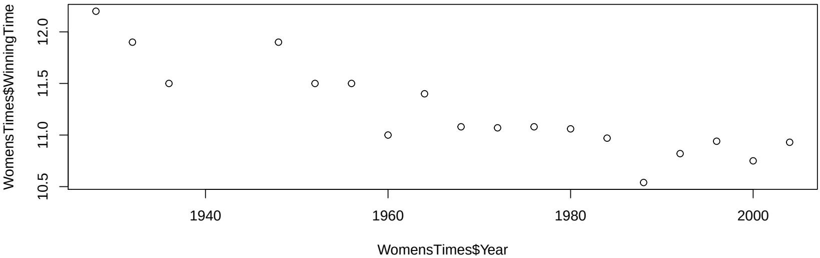
“Is the model without the effect sufficient to explain the data?”

How to Do Statistical Hypothesis Testing

1. get a *null hypothesis* (i.e. without the effect)
2. get an *alternative hypothesis* (i.e. with the effect)
3. Chose a *test statistic* (e.g. the likelihood)
4. calculate the distribution of the test statistic if the null hypothesis was true
5. ask if the observed value of the statistic falls within the null distribution
6. if it does not, declare the null hypothesis wrong

An example

```r
Lk <- "https://www.math.ntnu.no/emner/ST2304/2019v/Week1/Ol
Times100m <- read.csv(Lk)
Use <- Times100m$Sex=="Women" & !is.na(Times100m$WinningTim
WomensTimes <- Times100m[Use,]
plot(WomensTimes$Year, WomensTimes$WinningTime)
```



Are the times decreasing? = Is the slope zero?

An example

get a *null hypothesis*

- The slope is zero

get an *alternative hypothesis*

- The slope is not zero

Chose a *test statistic*

- The slope
- The likelihood

calculate the distribution of the test statistic if the null hypothesis was true

- Your job!

ask if the observed value of the statistic falls within the null distribution

- Your job!

Your job...

Is the likelihood from the data likely if the null hypothesis is true?

Use this code & compare the model.H1 likelihood with the null distribution (`hist()`might help)

```r
# Null Hypothesis
model.H0 <- lm(WinningTime ~ 1, data=WomensTimes)
# Alternative Hypothesis
model.H1 <- lm(WinningTime ~ Year, data=WomensTimes)

SimNullModel <- function(mod, X) {
  Sim <- simulate(mod) # simulate data from model
  model.test <- lm(Sim[,1] ~ X)
  logLik(model.test) # extract log-likelihood
}

Lhood <- replicate(1e3, SimNullModel(mod=model.H0,
                                     X=WomensTimes$Year))
```

Why Use the Likelihood?

It measures model fit

- Pr(Data|parameters)

It has some useful statistical properties

Summarises whole model, not just a parameter

Some statistical theory

In general, we know the distribution for the difference between likelihoods when the models are nested:

$$-2(log(L_1) - log(L_0))$$

follows a χ^2_p distribution, where p is the difference in number of parameters

Nested models: Model A is nested within model B if we can get model A by setting some parameters of model B to zero

Some statistical theory

For the normal distribution, the χ^2 works if we fix σ^2. If we cannot, it adds some extra error. So we use

$$\frac{-2(log(L_1) - log(L_0))}{s^2}$$

where s^2 is the estimate of the residual variance. This also follows a χ^2 distribution

We know from statistical theory that the distribution of the ratio of χ^2s follows an F distribution

The F distribution has 2 parameters, known a "degrees of freedom".

- numerator degrees of freedom: how many extra parameters are in the alternative model
- denominator degrees of freedom: how many parameters are used to estimate $\hat{\sigma}^2$
  - taken from the alternative model

Sums of Squares

The log-likelihood for a normal distributiuon is

$$l(\mathbf{x}|\mu, \sigma^2) = -\frac{n}{2} \log \sigma^2 - \sum_{i=1}^{n} \frac{(x_i - \mu)^2}{2\sigma^2}$$

And the main bit is $\sum_{i=1}^{n} \frac{(x_i - \mu_i)^2}{2\sigma^2}$

which is just a sum of squares, and a variance term

- estimate σ^2 with another sum of squares

Take-home Method

Use sums of squares to calculate log-likelihoods & residual deviance

Degrees of freedom are parameters of the F-distribution that we use to compare the estimated deviance to

In reality, R will do the hard work

With R

We can get R to make the comparison:

```
model.H0 <- lm(WinningTime ~ 1, data=WomensTimes)
model.H1 <- lm(WinningTime ~ Year, data=WomensTimes)

(an <- anova(model.H0, model.H1))
```

```
Analysis of Variance Table

Model 1: WinningTime ~ 1
Model 2: WinningTime ~ Year
  Res.Df    RSS Df Sum of Sq     F    Pr(>F)
1     17 3.3550
2     16 0.7086  1    2.6465 59.76 8.626e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 '
```

The test statistic is the F-ratio

Your Turn

For the Yields data from last week, is there an interaction between treatment and date (i.e. before/after 1970)?

```
Yld <- "https://www.math.ntnu.no/emner/ST2304/2019v/Week8/
Yields <-  read.csv(Yld)
Yields$After1970 <- Yields$StartYear>1969
model.yield.Int <- lm(yield ~ After1970 * Treatment,
                      data=Yields)
model.yield.Main <- lm(yield ~ After1970 + Treatment,
                       data=Yields)
```

Degrees of Freedom

We have N data points. Each is a “degree of freedom” that we can use in the estimation Each df can be spent to estimate one parameter The rest are used to estimate the residual variance

e.g.

$$y_i = \alpha + \beta x_i + \varepsilon_i$$

2 parameters (α and β), so $N - 2$ can be used to estimate σ^2

$N - 2$ is the residual degrees of freedom

Degrees of Freedom

If we compare 2 models, the difference in the residual degrees of freedom is the number of extra parameters in the alternative model

- this is the degrees of freedom.

(the same as used in a χ^2 test)

ANOVA made easier

We have just used anova() to compare 2 models, but it has traditionally been used to compare several:

```r
round(anova(model.yield.Int), 2)
```

```
Analysis of Variance Table

Response: yield
                    Df Sum Sq Mean Sq F value    Pr(>F)
After1970            1  13.06   13.06   56.61 < 2.2e-16 **
Treatment            3  93.87   31.29  135.62 < 2.2e-16 **
After1970:Treatment  3  19.81    6.60   28.62 < 2.2e-16 **
Residuals           64  14.77    0.23
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 '
```

ANOVA made easier

Each row is a test

```
print(xtable::xtable(anova(model.yield.Int)), digits=2,
      comment=FALSE)
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
After1970	1	13.06	13.06	56.61	0.0000
Treatment	3	93.87	31.29	135.62	0.0000
After1970:Treatment	3	19.81	6.60	28.62	0.0000
Residuals	64	14.77	0.23		

It compares a model with the terms above to one including that term e.g. the replicate line compares

```
After1970 + Treatment
```

to

```
After1970 + Treatment + After1970:Treatment
```

Why is ANOVA called ANOVA

ANOVA = Analysis of Variance

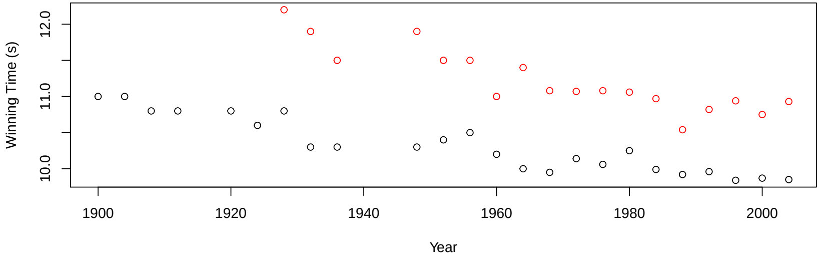
The Mean Sq is the mean square, i.e. Sum Sq/Df

- when the data cooperate, it is an estimate of the variance explained by that effect
- so we can sometimes use the Mean Square to eyeball how important a variable is

For you. . .

Do the mens & womens times improve at different rates?

```r
mod <- lm(WinningTime ~ Sex*Year, data=Times100m)
plot(Times100m$Year, Times100m$WinningTime,
     col=as.numeric(Times100m$Sex),
     xlab="Year", ylab="Winning Time (s)")
```



What we have done today

We can make models that are too big & horrible

If we have specific hypotheses w, we can test them

For regression, we use an ANOVA

- caclulates sums of squares, degrees of freedom, F ratios

We don't yet know how to deal with exploratory problems

- tomorrow!

Exploratory Problems

GWAS

- which genetic marker is correlated to a trait?

Species distributions models

- which environmental covariate explains where a species is?

Exploring for Good Models

Sometimes we don't have strong hypotheses. Instead we might be exploring which variables might have an effect

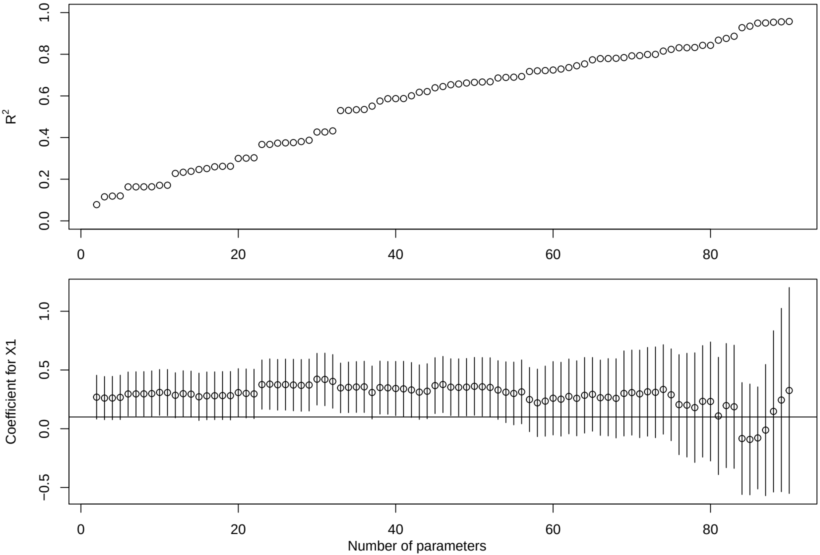
- our aim is to get a good model overall
- e.g. for prediction

What does a good model look like?

Discuss what you would want from a good model

The problem

A model with always fit better if you add a parameter



The problem

Key issue: is adding the extra parameter worth it?

ANOVA answers this by asking if the improvement from the extra parameter can be explained as noise

What does a good model look like?

- Simple
- Fits the data well
- Understandable

We can measure simplicity and fit.

- Fit: likelihood
- Simplicity: number of parameters

The Example

We will only use 20 covariates:

```
set.seed(25)
NSmall <- 100; PSmall <- 20
xSmall <- matrix(rnorm(NSmall*PSmall), nrow=NSmall)
mu <- 0.1*xSmall[,1] # true R^2 = 0.1^2/(0.1^2 + 1) = 1%
ySmall <- rnorm(NSmall, mu)
```

Penalisation

Another way of looking at the problem: we measure model adequacy

- is the model good enough for what we want?

We penalise complicated models

- measure complexity by number of parameters

Find the ‘best’ model as one with optimum between fit & complexity

How to Penalise

There are several ways to penalise. Here I will mention two, which chose different criteria

- *AIC*: Akaike's Information Criterion
- *BIC*: Bayesian Information Criterion

AIC tries to find the model that best predicts the data

BIC trues to find the model most likely to be true

Unfortunately, it's not possible to do both at the same time

AIC

Finds the model that would best predict replicate data

AIC = -2 Likelihood + 2 Number of Parameters

$$AIC = -2log(p(y|\theta)) + 2p$$

BIC

Finds the model which is most likely to be “true”

BIC = -2 Likelihood + log(N) Number of Parameters

$$AIC = -2 log(p(y|\theta)) + \log(n)p$$

- log(n) = log(sample size)
- penalises more than AIC

Extracting AIC

We can use the `AIC()`function:

```
model.null <- lm(ySmall ~ 1)
model.full <- lm(ySmall ~ xSmall)
model.2 <- lm(ySmall ~ x[,2])

AIC(model.null, model.2, model.full)
```

```
           df      AIC
model.null  2 296.1408
model.2     3 294.0873
model.full 22 314.3215
```

A lower value is better, so the null model is better than having all of the variables in it, and the model with variable 2 is slightly better still.

Your task

Fit all of the models with one covariate (i.e. `y~x[,1]`, `y~x[,2]` etc.). Which one gives the best model (i.e. has the lowest AIC)?

```r
model.1 <- lm(ySmall ~ xSmall[,1])
model.2 <- lm(ySmall ~ xSmall[,2])
model.3 <- lm(ySmall ~ xSmall[,3])
# ... up to
model.20 <- lm(ySmall ~ xSmall[,20])

AIC(model.1, model.2, model.3, model.20)
```

Using AIC/BIC

Full Subset Selection

- calculate AIC/BIC for every model
- pick the best (= lowest)

Usually, if the values are within ∼2 of each other, the models are pretty similar.

Fit all of the models

This can get ugly (using the full data set with 80 variables will take too long)

```r
library(bestglm) # might need install.packages("bestglm")
UseData <- data.frame(cbind(xSmall, ySmall))

AllSubsetsAIC <- bestglm(Xy=UseData, IC="AIC")
AllSubsetsBIC <- bestglm(Xy=UseData, IC="BIC")
```

Looking at all of the models

The `bestglm` object has several pieces:

```
names(AllSubsetsAIC)
```

```
## [1] "BestModel"   "BestModels"  "Bestq"       "qTable"
## [6] "Title"       "ModelReport"
```

`Subsets` gives the AIC (or BIC) for the best models:

```
AllSubsetsAIC$BestModels
```

Run this yourselves, but the output is big

Looking at the best model

`BestModel` is the `lm` object for the best model

```
coef(AllSubsetsBIC$BestModel)
```

Run this. Which model is best according to AIC, and which according to BIC?

How good are the models? How do they compare with the truth?

If we have time...

A more relaistic model for reality can be that everything has an effect, but some have a stronger effect than others. We can look at how model selection behaves then:

```r
betas <- 0.3*(0.9^(1:PSmall))
# plot(1:PSmall, betas)
set.seed(25)
NSmall <- 100; PSmall <- 20
xSmall <- matrix(rnorm(NSmall*PSmall), nrow=NSmall)
mu <- sweep(xSmall, 2, betas, '*') # true R^2 = 0.1^2/(0.1
yTaper <- rnorm(NSmall, mu)
TaperData <- data.frame(cbind(xSmall, yTaper))

AllTaperAIC <- bestglm(Xy=TaperData, IC="AIC")
AllTaperBIC <- bestglm(Xy=UseData, IC="BIC")
```

What we've done this week

Learned we sometimes need to compare models to find the best

Use ANOVA to compare models when we have specific hypotheses

Measure model adequacy with AIC or BIC, and compare lots of models to select the best, when we don't have specific hypotheses

Next Week

Generalised Linear Models

- when things aren't normal