

STAT406- Methods of Statistical Learning Lecture 4

Matias Salibian-Barrera

UBC - Sep / Dec 2017

Proper use of CV

- An example of the importance and relevance of what we discussed in our last class:

Ambroise, C. and McLachlan, G.J.
Selection bias in gene extraction on
the basis of microarray
gene-expression data, PNAS, 2002, 99
(10), 6562-6566.

<https://doi.org/10.1073/pnas.102102699>

Discussion points

- Why? Why would anybody want to **not** use all available features?
- “Somewhat obvious”: model parsimony, identify features that are relevant for the process under study.
- “Not so obvious?”: does prediction suffer if we use fewer variables? how much variability is induced by the feature selection step?



Model / feature selection

- Simple example:

```
set.seed(123)
x1 <- rnorm(506)
x2 <- rnorm(506, mean=2, sd=1)
x3 <- rexp(506, rate=1)
x4 <- x2 + rnorm(506, sd=.1)
x5 <- x1 + rnorm(506, sd=.1)
x6 <- x1 - x2 + rnorm(506, sd=.1)
x7 <- x1 + x3 + rnorm(506, sd=.1)
y <- x1*3 + x2/3 + rnorm(506, sd=2.2)
```

- Variables X_1 and X_2 are clearly important. But they are also highly correlated to X_4 , X_5 , X_6 and X_7 .

Model / feature selection

- However, nothing is significant?

```
> summary(lm(y~., data=x))
```

Call:

```
lm(formula = y ~ ., data = x)
```

Residuals:

| Min | 1Q | Median | 3Q | Max |
|--------|--------|--------|-------|-------|
| -6.882 | -1.474 | -0.033 | 1.415 | 5.823 |

Coefficients:

| | Estimate | Std. Error | t value | Pr(> t) |
|-------------|----------|------------|---------|----------|
| (Intercept) | 0.03457 | 0.23018 | 0.150 | 0.8807 |
| x1 | 3.22612 | 1.68088 | 1.919 | 0.0555 . |
| x2 | 0.23867 | 1.39355 | 0.171 | 0.8641 |
| x3 | -0.35926 | 0.98680 | -0.364 | 0.7160 |
| x4 | -0.69359 | 0.99025 | -0.700 | 0.4840 |
| x5 | 0.09271 | 0.91162 | 0.102 | 0.9190 |
| x6 | -0.73887 | 1.01114 | -0.731 | 0.4653 |
| x7 | 0.31651 | 0.98610 | 0.321 | 0.7484 |

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.148 on 498 degrees of freedom

Multiple R-squared: 0.6353, Adjusted R-squared: 0.6302

F-statistic: 123.9 on 7 and 498 DF, p-value: < 2.2e-16

Model / feature selection

- But...

```
> summary(lm(y~x1+x2, data=x))
```

Call:

```
lm(formula = y ~ x1 + x2, data = x)
```

Residuals:

| | Min | 1Q | Median | 3Q | Max |
|--|---------|---------|---------|--------|--------|
| | -6.9303 | -1.5736 | -0.0068 | 1.3840 | 5.9567 |

Coefficients:

| | Estimate | Std. Error | t value | Pr(> t) |
|-------------|----------|------------|---------|-------------|
| (Intercept) | 0.00733 | 0.20900 | 0.035 | 0.97204 |
| x1 | 2.89168 | 0.09806 | 29.490 | < 2e-16 *** |
| x2 | 0.27903 | 0.09249 | 3.017 | 0.00268 ** |

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.141 on 503 degrees of freedom

Multiple R-squared: 0.6343, Adjusted R-squared: 0.6328

F-statistic: 436.2 on 2 and 503 DF, p-value: < 2.2e-16

Model / feature selection

- Even worse...

```
> summary(lm(y~x1+x2+x4, data=x))
```

Call:

```
lm(formula = y ~ x1 + x2 + x4, data = x)
```

Residuals:

| | Min | 1Q | Median | 3Q | Max |
|--|---------|---------|---------|--------|--------|
| | -6.8064 | -1.5229 | -0.0308 | 1.4226 | 5.8861 |

Coefficients:

| | Estimate | Std. Error | t value | Pr(> t) |
|-------------|------------|------------|---------|------------|
| (Intercept) | 0.0001127 | 0.2093588 | 0.001 | 1.000 |
| x1 | 2.8964461 | 0.0983390 | 29.454 | <2e-16 *** |
| x2 | 0.9740807 | 0.9917783 | 0.982 | 0.326 |
| x4 | -0.6934442 | 0.9851714 | -0.704 | 0.482 |

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.142 on 502 degrees of freedom

Multiple R-squared: 0.6347, Adjusted R-squared: 0.6325

F-statistic: 290.7 on 3 and 502 DF, p-value: < 2.2e-16

Model / feature selection

- If we use AIC

```
> st <- stepAIC(null,  
  scope=list(lower=null, upper=full))  
> st
```

Call:

```
lm(formula = y ~ x1 + x6, data = x)
```

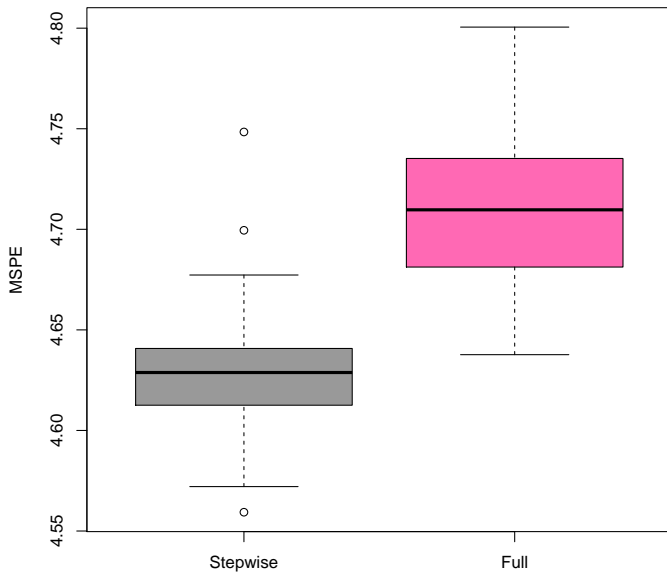
Coefficients:

| (Intercept) | x1 | x6 |
|-------------|----------|-----------|
| -0.000706 | 3.175239 | -0.282906 |

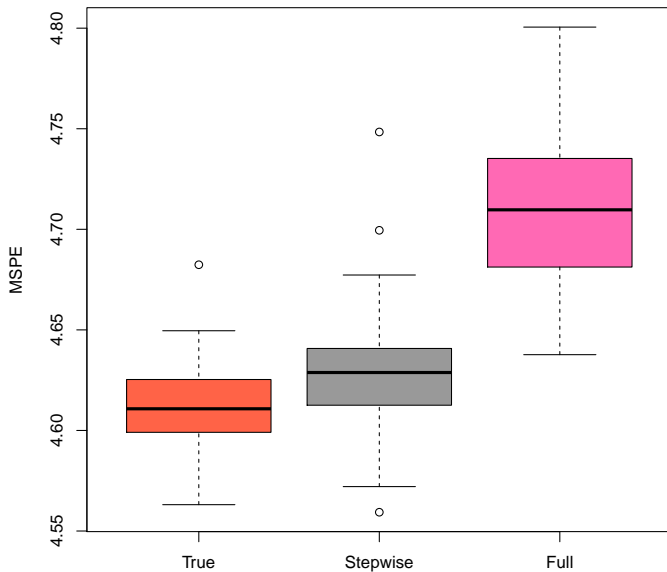
Discussion points

- Modeling problem (important variables may be missed)
- Prediction? Does stepwise give “the best” predicting model?

MSPEs



MSPEs



Discussion points

- Correlated covariates have become prevalent
- Researchers can (and do) collect data “blindly”
- Data are collected without a specific question in mind

Discussion points

- Correlated covariates:
- Mask each other when included simultaneously in a model
- May reduce prediction accuracy

Model / feature selection

One strategy:

- (1): Select models to be considered
- (2): Select a quantitative criterion to compare them (e.g. AIC, C_p , CV-based $\widehat{\text{MSPE}}$)
- (3): Choose a strategy to explore the models under consideration

Model / feature selection

For example:

- (1): Consider all possible models
 - (2): Use AIC to compare them
 - (3): Best subset search (2^p fits!)
 - (3'): Stepwise search
-
- Is this strategy prediction-based?

AIC?

Why not compare models using residual sum of squares, or R^2 ?

LS vs MLE

Note that, if we assume that the error distribution is Gaussian, then a least squares fit for a linear regression model is the same as the MLE fit

... or is it?

Comparing models

- Comparing likelihoods / residuals isn't very useful
- More complex models have higher likelihoods (smaller residuals)
- The Akaike Information Criterion provides a way to compare models with different number of parameters
- There are many different ways to motivate it

Comparing models

- We can measure the “distance” between the true distribution of the data ($f_0(y)$) and our model $f(y, \theta)$

$$\begin{aligned} d(\theta, f_0) &= E_0[-2 \ell(y, \theta)] = \\ &\int -2 \ell(y, \theta) f_0(y) dy = \\ &2 \left[\mathcal{K}(\theta, f_0) - \int \log(f_0(y)) f_0(y) dy \right] \end{aligned}$$

Comparing models

- Given our estimator $\hat{\theta}_n$ we could use

$$d(\hat{\theta}_n, f_0) = E_0[-2 \ell(y, \theta)]_{\theta=\hat{\theta}_n}$$

to see “how far” our model-based estimator is from the true distribution

- However, we can't compute $d(\hat{\theta}_n, f_0)$
- Can we use $-2 \ell(y, \hat{\theta}_n)$ to estimate $d(\hat{\theta}_n, f_0)$?

Comparing models

- Yes, but this estimator is biased

$$E_0 \left[-2 \ell(y, \hat{\theta}_n) \right] = \\ E_0 \left\{ E_0 \left[-2 \ell(y, \theta) \right]_{\theta = \hat{\theta}_n} \right\} - 2p + o(1)$$

Comparing models

- In other words

$$E_0 [\text{AIC}] \approx E_0 \left[d(\hat{\theta}_n, f_0) \right]$$

where

$$\text{AIC} = -2\ell(y, \hat{\theta}_n) + 2p$$

Comparing models

- For Gaussian errors we have

$$\text{AIC} = n \log \left(\frac{\text{RSS}}{n} \right) + 2p + \text{constant}$$

where

$$\text{RSS} = \sum_{i=1}^n r_i^2,$$

the **constant** depends on n , not on p

Comparing models

- However, many times we find

$$\text{AIC} = \frac{1}{n} \frac{1}{\hat{\sigma}^2} \left(\text{RSS} + 2p\hat{\sigma}^2 \right) + \text{constant}$$

(e.g. [JWHT13])

Where does this expression come from?

Comparing models

- Regularity assumptions are needed
 - This is an asymptotic approximation, n should be large
 - One of the models should include truth
 - $\theta_1 \neq \theta_2 \Rightarrow f(y, \theta_1) \neq f(y, \theta_2)$
 - Standard large-sample MLE assumptions to obtain asymptotic normality

Comparing models

- Air pollution example in \mathbb{R}
- Synthetic data example

Sometimes...

- Selecting variables is not always necessary in terms of prediction accuracy.
- Doing so may in fact yield worse results.
- One such an example is discussed on Github.
- Read it carefully.