Lecture 10 5/24/2022

Automatic Variable Selection

- Instead of choosing what goes into the model manually, we can find a program/algorithm to do so

- Suppose we have y and q predictors. There are 2^q possible regression models

- A computer can potentially pick a "best" model, but it has to be taught what is meant by "best"

- keep in mind: bias-variance tradeoff for prediction based on learned model

  - bias: the prediction error resulting from miss-specifying the model
  - variance: the prediction error resulting from variations in the data used for fitting

- Suppose we have a model f to describe response y from feature z. Given a new z_{n+1} we may write

$$y_{n+1} = \underbrace{f(z_{n+1})}_{z_{n+1}^T \beta} + \varepsilon_{n+1}$$

- Suppose that we have past date,

this data $((y_i, z_i)_{i=1}^{n})$ we estimate $\hat{f}$ from

In the linear model:

$$\underset{z\in\mathbb{R}^p}{\hat{f}(z)} = z^T\hat{\beta}, \quad \hat{\beta} = (Z^TZ)^{-1}Zy$$

$$Z\in\mathbb{R}^{n\times p}$$

- we have the decomposition:

$$\mathbb{E}\left[(y_{n+1}-\hat{f}(z_{n+1}))^2\right]$$

$$= \mathbb{E}\left[(y_{n+1}-\underline{f(z_{n+1})})^2\right]$$ irreducible error

$$+ \left(\underline{f(z_{n+1})} - \underline{\mathbb{E}[\hat{f}(z_{n+1})]}\right)^2$$ bias^2

$$+ \mathbb{E}\left[\left(\hat{f}(z_{n+1}) - \mathbb{E}[\hat{f}(z_{n+1})]\right)^2\right]$$ variance

$$:= \underset{\substack{\uparrow \\ \text{we cannot control}}}{\sigma^2} + (\text{bias}(\hat{f}))^2 + \text{Var}(\hat{f}(z_{n+1}))$$

Possible tradeoff based on the way we choose $\hat{f}$

- When $f(z) = z^T\beta$ so that

$y_{n+1} = z_{n+1}^T\beta + \varepsilon_{n+1}$, $\mathbb{E}[\varepsilon_{n+1}] = 0$, then

$$\mathbb{E}[\hat{f}(z_{n+1})] = \mathbb{E}[z_{n+1}^T\hat{\beta}] = z_{n+1}^T\mathbb{E}[\hat{\beta}]$$

$$\mathbb{E}[\hat{f}(z_{n+1})] = \mathbb{E}[z_{n+1}^T\hat{\beta}] = z_{n+1}^T\mathbb{E}[\hat{\beta}]$$

$$= z_{n+1}^T \beta = \mathbb{E}[y_{n+1}] = f(z_{n+1})$$

Namely, the bias is zero.

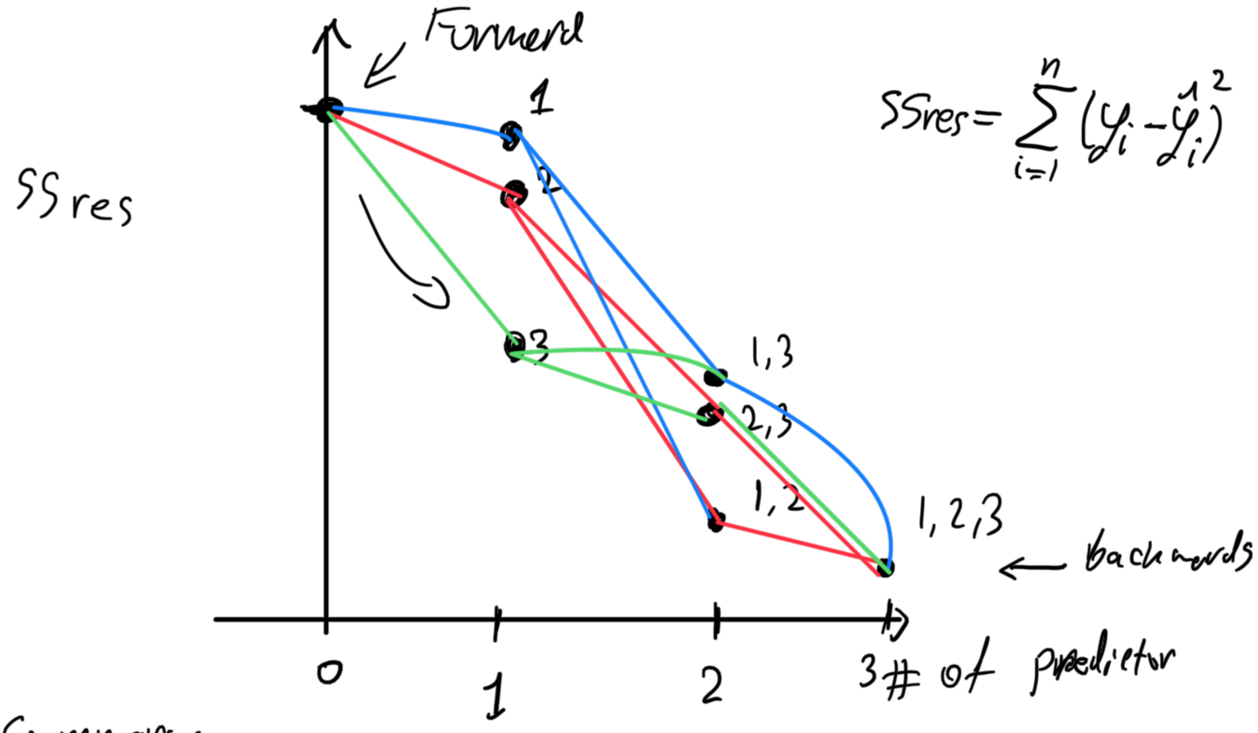
It may not be zero anymore provided:

We don't have/use an unbiased estimator for f. Typically, this can happen because we don't know precisely f

---

All Combinations

Example: 3 predictors



$$SSres = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

Summary:

# pred	best	SS_{res}
0	ϕ	10
1	$\{3\}$	5
2	$\{1, 2\}$	2
3	$\{1, 2, 3\}$	1

- <u>Fornard approach</u>: we start at ϕ and add the best predictor if the new model is statistically significant; we stop otherwise (using F-test for extra sum of squares)

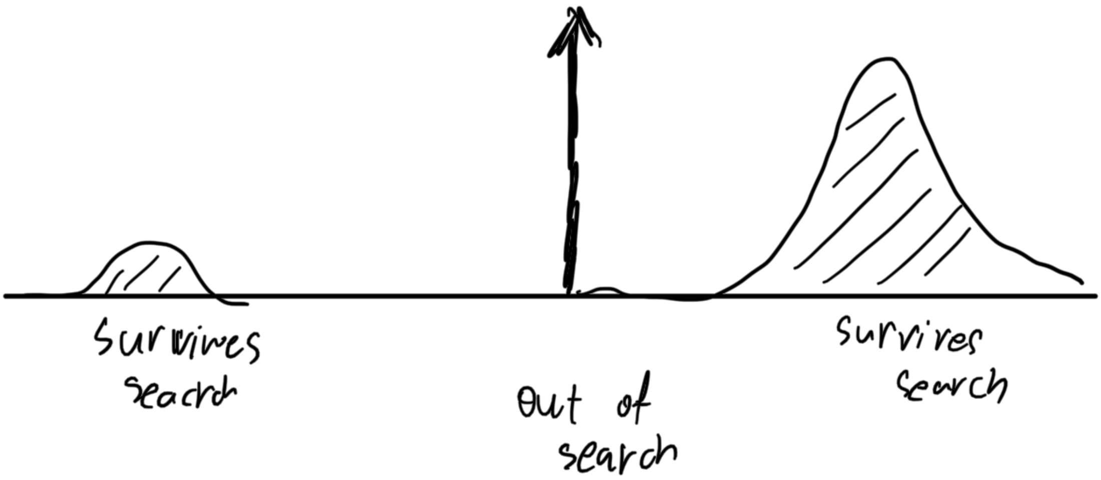
- <u>Backward approach</u>: we start with all predictors, and drop the least significant / the one that leaves you with maximal SS_{res} if the old model is not statistically significantly better than the new one; otherwise, stop

- Issues:
    - these are greedy approaches (a hybrid approach that "looks" into the future also makes sense)

- Both approaches usually disagree on the stoping point

- We cannot use p-values, confidence intervals, t-stats in the post selection model: "the quiet scandal"

- The distribution of $\hat{\beta}_j$ when we take the model selection process into account:



$$Pr(\hat{\beta}_j = 0) > 0$$

"The quiet scandal" [Leo Breiman 92]

- why use stepwise approaches:

  - Models can be <u>cross-validated</u> to measure accuracy

  - Helps leave out variables which

- helps leave out variables which can save time & money

Cross validation

- split the data into k smaller sets; leave one aside; fit a family of models based on the other $k-1$ sets; evaluate accuracy over left-out set; repeat for all $i=1,\dots,k$; average accuracy; pick the model with best averaged accuracy.
- special case $k=n$ called leave-one-out
- Let $\hat{y}^{(i)}$ be LS predictor when z_i is left out:

$$\hat{y}_i^{(i)} = z_i^T \hat{\beta}^{(i)}$$

where $\hat{\beta}^{(i)}$ depends on $\{(z_j, y_j)_{j \neq i}\}$ (fitted based on Z with i-th row removed)

- Cross validation (CV) error:

$$CV(\text{model}) = \sum_{i=1}^{n} (y_i - \hat{y}_i^{(i)})^2$$

- Why use CV:
  - Generally, knocks out models that overfit
  - parallels our goal of predicting new values

- Issues: can computationaly prohibitive

CV in Linear Models

- we have a "short cut"

$$\hat{y}_i = H_{ii} y_i + (1 - H_{ii}) \hat{y}_i^{(i)}$$

$$H = Z(Z^T Z)^{-1} Z^T$$

(proof is based on the Sherman-Morrison formula for the inversion of matrix + rank one mat.)

so $$\hat{y}^{(i)} = \underline{\hat{y}_i - H_{ii} y_i} \over 1 - H_{ii}$$

all left out prediction are evaluated using one regression instead of n

- The residuals:

$$\underline{y_i - \hat{y}_i^{(i)}} = \frac{y_i - \hat{y}_i}{1 - H_{ii}} = \frac{\hat{\varepsilon}_i}{1 - H_{ii}}$$

$$H_{ii} \leq 1$$

- Overall:

$$CV = \sum_{i}^{n} (y_i - \hat{y}_i^{(i)})^2 = \sum_{i=1}^{n} \frac{\hat{\varepsilon}_i^2}{(1 - H_{ii})^2}$$

Penalty on number of variables

- Akaike's Information Criterion (AIC)

$$AIC := n \cdot \log\left(\frac{SS_{res}}{n}\right) + 2p$$

- Baye's Information Criterion (BIC)

$$BIC = n \log \frac{SS_{Res}}{n} + p \log n$$

- we pick the model that minimizes AIC or BIC

- Considerations:

  - AIC & BIC are more "principled" approches than forward/backward

- For $\log(n) > 2$, BIC penalizes predictors more severly.

- BIC is better at getting the "right" model - correctly identities non-zero variables when true model is linear as $n \to \infty$

- AIC is more accurate in making predictions

- We should not use confidence intervals and t-tests after selecting models with AIC or BIC