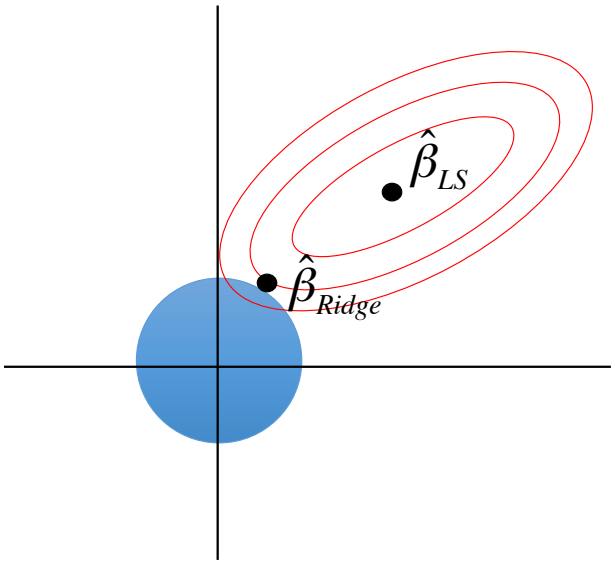
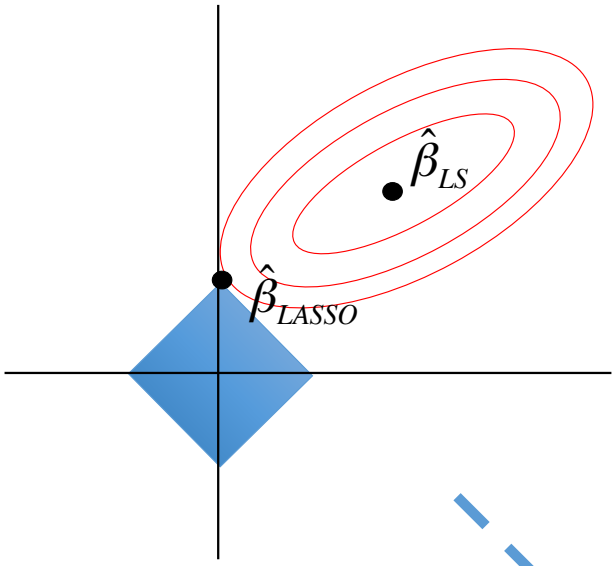
Outline: Linear Models in Large Feature Spaces, traditional Feature Selection (F. Chiaromonte)

Introduction to Statistical Learning

Chapter 6 Sections 1 and 3





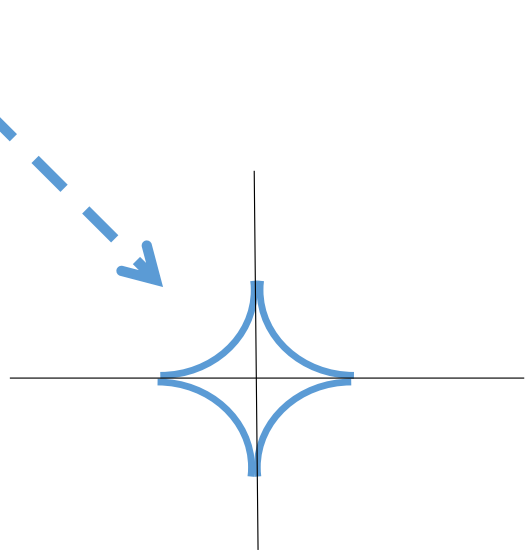
CONVEX constrained optimization
very efficient computational approaches.

Estimates are biased (even when the LS for the full model is not)… but **more stable**, and **sparse** (L1).

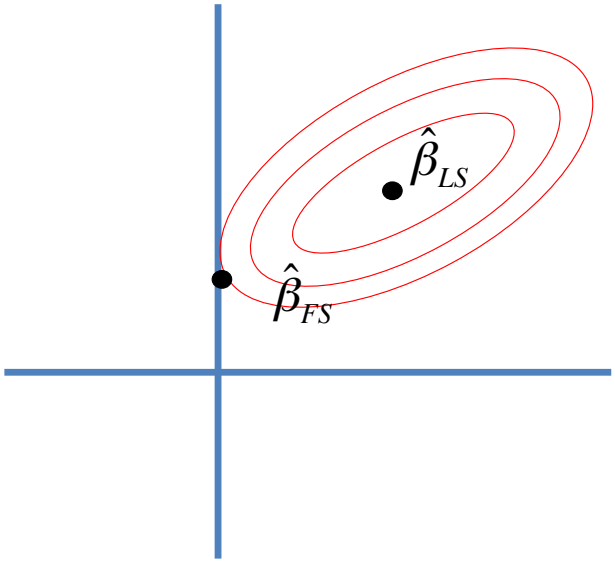
Very broad literature, lots of variants, including those to **incorporate group or order structure** for features.

Some references:

- Hastie T., Tibshirani R., Friedman J. (2009). Elements of Statistical learning 2nd ed. Springer.
- Tibshirani, R. (1996) Regression shrinkage and selection via the lasso. JRSS B, 58(1) 267-288.
- Zou H. Hastie T. (2005) Regularization and variable selection via the elastic net. JRSS B, 67(2) 301–320.
- Tibshirani R., Saunders M. (2005) Sparsity and smoothness via the fused lasso. JRSS B, 67(1) 91–108
- Yuan M., Lin Y. (2006) Model selection and estimation in regression with grouped variables. JRSS B, 68(1) 49–67.
- Fan J. Li R. (2001) Variable selection via nonconcave penalized likelihood and its oracle properties. JASA, 96(456) 1348-1360



… abandon CONVEXITY
reduced bias, but harder computational problem.



Traditional ("hard") feature selection: (most) **NON-CONVEX constrained optimization**

L0 norm counts non-0 coefficients, but size of each non-0 coefficient is unconstrained.

Much harder, previously not computationally viable; now *Mixed Integer Optimization*.

Some (non-traditional) references:

- Bertsimas D., King A., Mazumder R. (2016) Best subset selection via a modern optimization lens. AOS 44(2) 813-852.
- Kenney A., Chiaromonte F., Felici G. (2020) MIP-BOOST: Efficient and Effective L_0 Feature Selection for Linear Regression. JCGS.

$$\| \underline{Y} - \underline{X}\beta \|^2 = \min_{\beta \in R^p}$$

$$\sum_{j=1}^{p} Ind(\beta_j \neq 0) \leq c$$ size constraint

SIZE CONSTRAINT TUNED BY CROSS VALIDATION
… the traditional **Best Subset Selection** problem

Important: can also add further "integer" constraints to capture structure.

Back to traditional statistical approaches:

- **Feature (Subset) Selection**
  - Best Subset Selection
  - Step-wise Selection

  Followed by LS fit of the (smaller) model comprising the selected features.

- **Dimension Reduction**
  - Principal Components (unsupervised reduction)
  - Sufficient Dimension Reduction (supervised reduction)

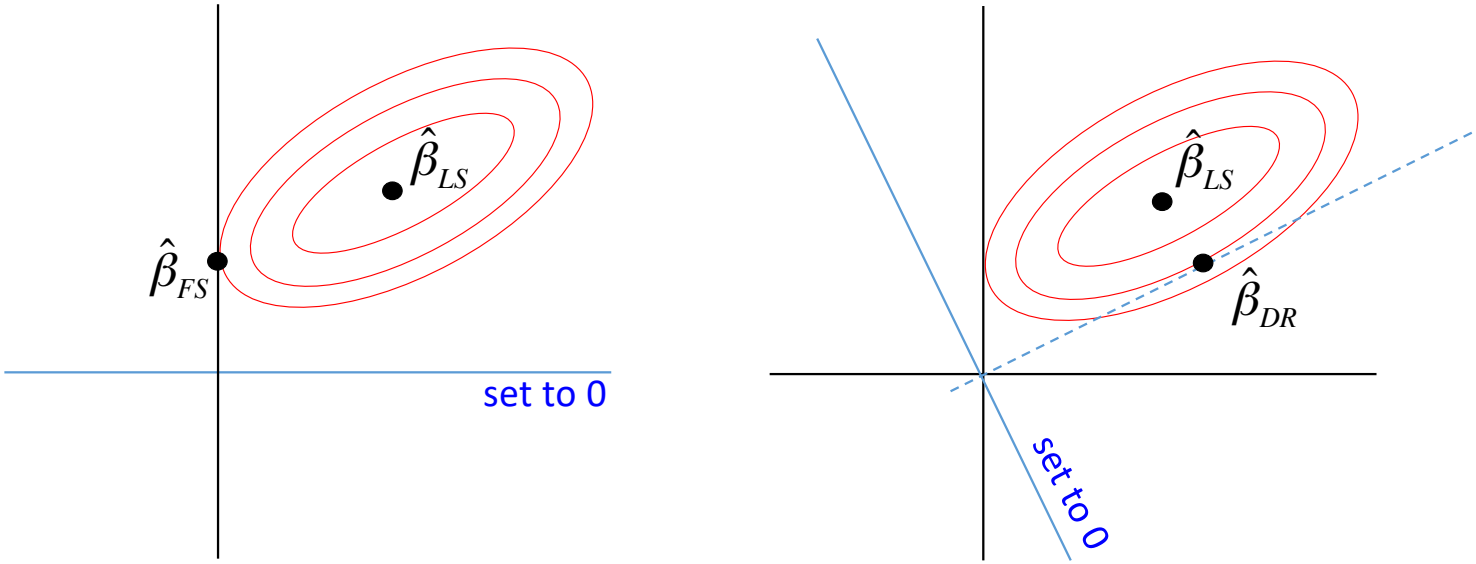
  Followed by LS fit of the (smaller) model comprising the selected linear combinations

Also these can be though of in the framework of constrained LS: *Linear constraints* to force β in a coordinate space, or a generic linear subspace, of R^p. ***BUT WE NEED AN ADDITIONAL "INGREDIENT"!***

==========

Dimension Reduction: ISLR also describes Partial Least Squares. But does not describe Sufficient Dimension Reduction techniques.

Cartoons with p=2 and c=1 (d=1)



$$\| \underline{Y} - \underline{X}\beta \|^2 = \min_{\beta \in R^p}$$

$$H_{FS}\beta = 0_{(p-c)}$$

linear constraints

$$\| \underline{Y} - \underline{X}\beta \|^2 = \min_{\beta \in R^p}$$

$$V_{DR}\beta = 0_{(p-d)}$$

(maximal) **COUNT** OF FEATURES c OR **DIMENSION** d FIXED… SO IS **(O.N.) BASIS** IN THE FEATURE SPACE FROM WHICH WE FORM THE (p-c) or (p-d) CONSTRAINTS

Feature Selection: How do we produce a pxp matrix H expressing an O.N. basis in the feature space?

(we then focus on its last (p-c) rows to create the linear constraints with H_{FS})

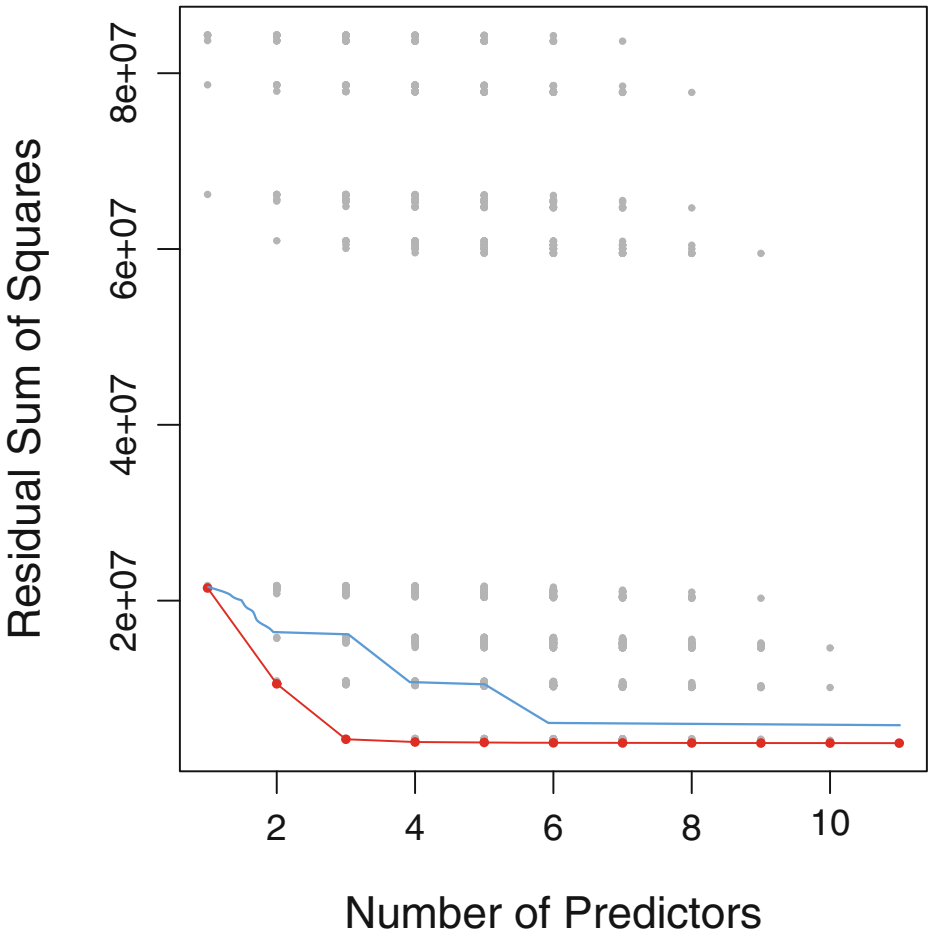
Feature (subset) Selection:

Make a sequence of nested subsets adding one feature at a time, i.e. make an ordering of the features.

This corresponds to a permutation of the elements of the Canonical O.N. basis $\{e_1,... e_p\}$, and provides

H (a permutation matrix)

Rigorous for feature selection implemented through **stepwise methods**, not for **best subset methods** which find the best subset of each size – in principle the subsets may not be nested (but the intuition still works).



The RSS of 2^p possible models (subsets of features).

For each size c, there are $\binom{p}{c}$ possible models.

On the lower boundary (red) are the best models of each size; best subsets. May or may not be a nested sequence.

The blue path represents a nested sequence of "good" models obtained stepwise. May depart from the lower boundary.

Algorithm 6.1 *Best subset selection*

1. Let $\mathcal{M}_0$ denote the *null model*, which contains no predictors. This model simply predicts the sample mean for each observation.
2. For $k = 1, 2, \ldots p$:
   (a) Fit all $\binom{p}{k}$ models that contain exactly k predictors.
   (b) Pick the best among these $\binom{p}{k}$ models, and call it $\mathcal{M}_k$. Here *best* is defined as having the smallest RSS, or equivalently largest R^2.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2.

Best Subset approach requires fitting ALL possible models ("brute" enumeration).

Algorithm 6.2 *Forward stepwise selection*

1. Let $\mathcal{M}_0$ denote the *null* model, which contains no predictors.
2. For $k = 0, \ldots, p-1$:
   (a) Consider all $p - k$ models that augment the predictors in $\mathcal{M}_k$ with one additional predictor.
   (b) Choose the *best* among these $p - k$ models, and call it $\mathcal{M}_{k+1}$. Here *best* is defined as having smallest RSS or highest R^2.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2.

Stepwise approaches do not; they explore fewer models. Computationally leaner.

Hybrid stepwise approaches mix forwards and backward progression.

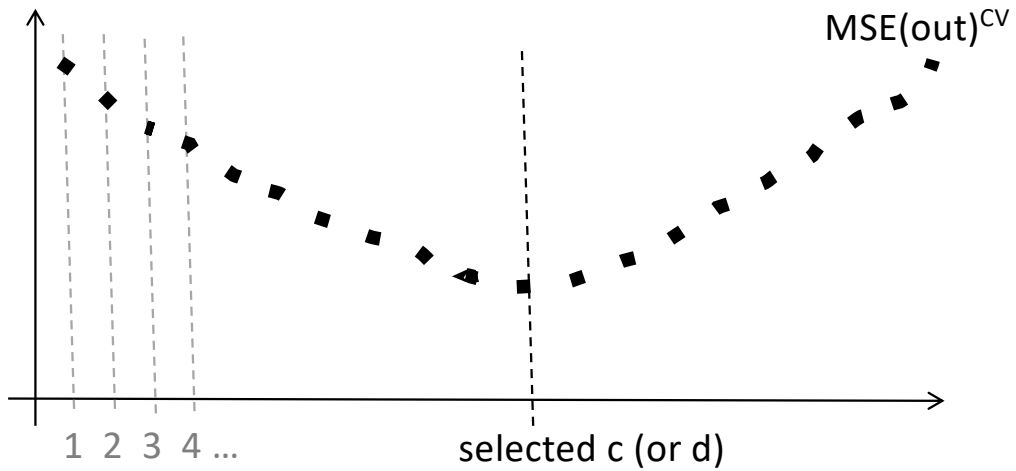
Algorithm 6.3 *Backward stepwise selection*

1. Let $\mathcal{M}_p$ denote the *full* model, which contains all p predictors.
2. For $k = p, p-1, \ldots, 1$:
   (a) Consider all k models that contain all but one of the predictors in $\mathcal{M}_k$, for a total of $k - 1$ predictors.
   (b) Choose the *best* among these k models, and call it $\mathcal{M}_{k-1}$. Here *best* is defined as having smallest RSS or highest R^2.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2.

(Step 3 in each algorithm; next…)

How do we select the (maximal) count c, or dimension d, along an ordered sequence of features or linear combinations?

Of course once H (or V) are fixed we could perform the selection estimating the out-of-sample MSE associated to each value of c, or d, by cross-validation (same as for the penalty parameter in Ridge or LASSO).



Historically though other (non-computation based) approaches have been used.

Feature (subset) Selection: the number of features c (**= d** if formulae below) is selected optimizing a criterion that approximates an estimate of the out-of-sample performance – but indirectly, not directly as in cross-validation; no intensive computation required.

$$\text{Adjusted } R^2 = 1 - \frac{\text{RSS}/(n-d-1)}{\text{TSS}/(n-1)}.$$

Simplest adjustment, each sum of squares is divided by its degrees of freedom.

$$C_p = \frac{1}{n}\left(\text{RSS} + 2d\hat{\sigma}^2\right)$$

Formula in the book, proportional to original formula.
If $\hat{\sigma}^2$ is an unbiased estimate of the error variance, C_p is an unbiased estimate of the out-of-sample MSE (can take the s^2 estimate from the full model, least likely to carry bias.

Strike a balance between RSS (error on the training data) and d (size, complexity of the model)

Akaike and Bayes Information Criteria: can be used for model selection whenever models are fitted by Maximum Likelihood.

For linear models with additive Gaussian errors their expressions become:

$$\mathrm{AIC} = \frac{1}{n\hat{\sigma}^2}\left(\mathrm{RSS} + 2d\hat{\sigma}^2\right)$$

Behaves exactly like C_p here!

$$\mathrm{BIC} = \frac{1}{n}\left(\mathrm{RSS} + \log(n)d\hat{\sigma}^2\right)$$

The weight of the uses log(n), depending on sample size, instead of 2. For n>7 BIC has a steeper penalty than AIC (C_p), it favors smaller models.

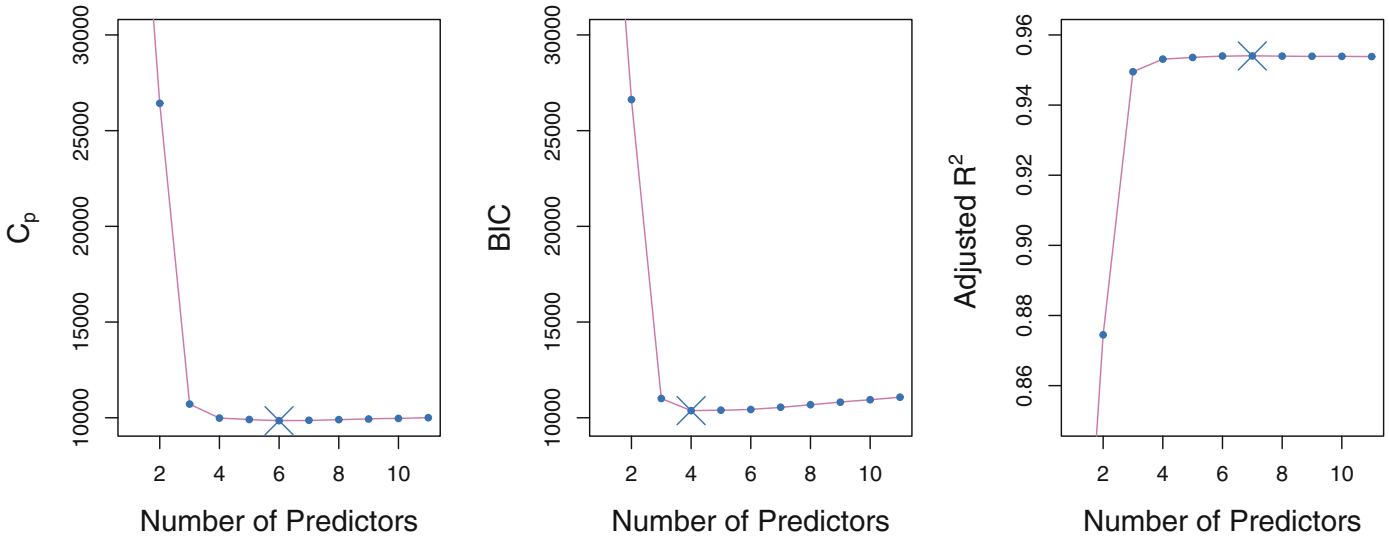
General equations for Information Criteria:

$$AIC = -2\log L(\hat{\vartheta}_{ML}) + 2d$$

$$BIC = -2\log L(\hat{\vartheta}_{ML}) + \log(n)d$$

$L(\hat{\vartheta}_{ML})$ = optimal likelihood value on MLE of parameter vector

d = # of free parameters being estimated



This behavior is rather common; even though the criteria are NOT monotone and ought to have a minimum/maximum, in many applications they do not clearly discriminate the number of features (the model) to be used – rather flat ranges.

Would you pick 3 instead of the “technical” optima, which btw differ across criteria?