Model Selection

-Introduction to Data Science-

ISL 6.1

Lecturer: Darren Homrighausen, PhD

Preamble:

- Armed with our risk estimators, we need ways to sift through the possible models
- Available techniques vary with the absolute and relative sizes of n and p
- Like most statistical techniques, model selection comes down to optimization...

Brief optimization and convexity detour

Optimization

An optimization problem (program) can be generally formulated as

$$\text{minimize } F(x) \tag{1}$$
$$\text{subject to } f_j(x) \leq 0 \text{ for } j = 1, \ldots, m \tag{2}$$
$$h_k(x) = 0 \text{ for } k = 1, \ldots, q \tag{3}$$

Here

- $x = (x_1, \ldots, x_n)^\top$ are the parameters
- $F : \mathbb{R}^n \to \mathbb{R}$ is the objective function
- $f_j, h_k : \mathbb{R}^n \to \mathbb{R}$ are constraint functions

The optimal solution x^* is such that $F(x^*) \leq F(x)$ for any x^*, x that satisfies equations (2) and (3).

Convexity

The main dichotomy of optimization programs is convex vs. nonconvex

A convex program is one in which the objective and constraint functions are all convex

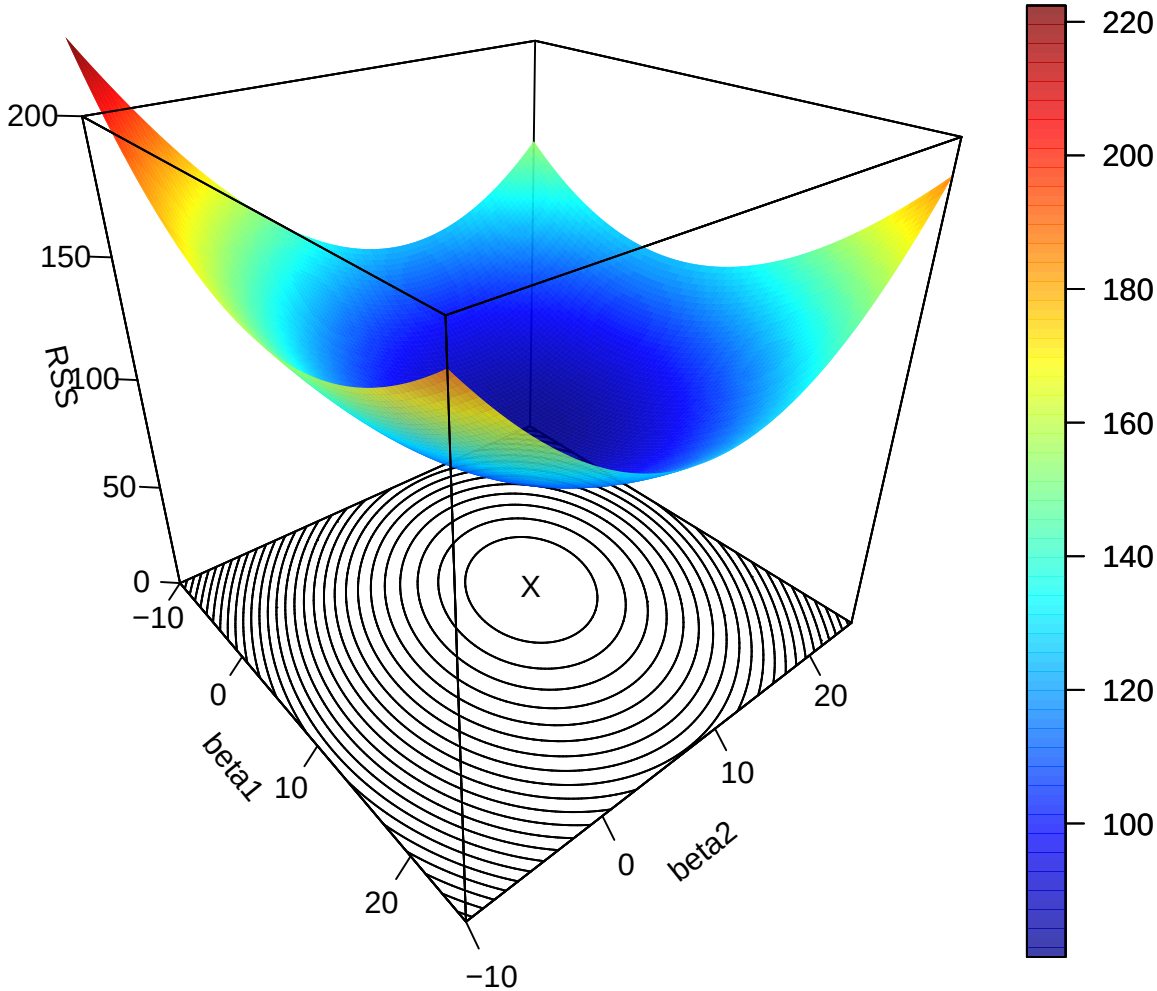
$$f(tx + (1-t)x') \leq tf(x) + (1-t)f(x') \qquad \text{for any } t \in [0,1]$$

This can be thought of (for smooth enough f)

$$f(x') \geq f(x) + (\nabla f|_x)^\top (x' - x)$$

Intuition: This means that the function values at a point x' are above the supporting hyperplane given by the tangent space at any point x

Convexity example



With $RSS = ||Y - \mathbb{X}\beta||_2^2$ for $p = 2$

Convexity

Methods for convex optimization programs are (roughly) always global and fast

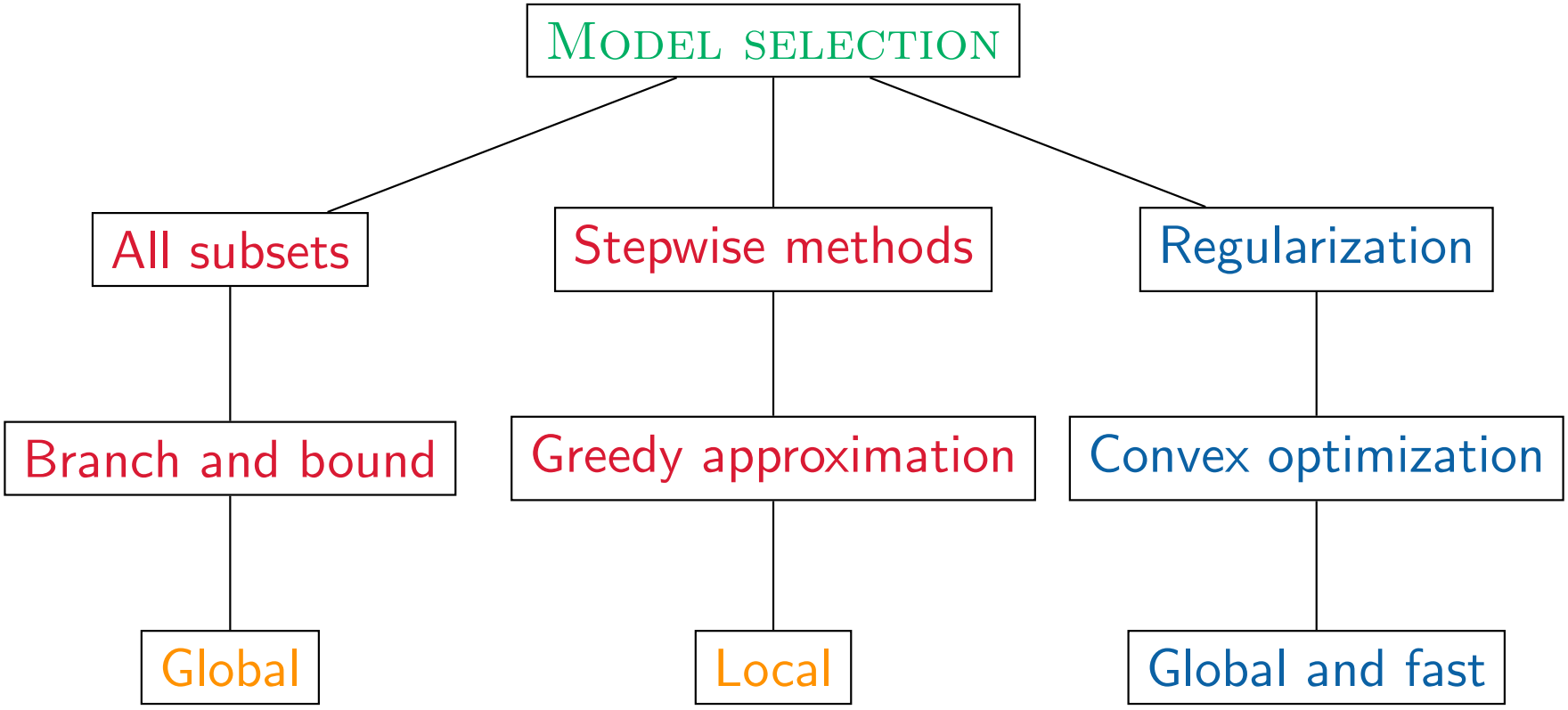
For general nonconvex problems, we have to give up one of these:

- Local optimization methods that are fast, but need not find global solution

  (So called greedy approximations)
- Global optimization methods that find global solutions, but are not always fast (indeed, are often slow)

  (Usually exhaustive search type approaches)

Model selection



Some comments:

- Non convex programs
- Can be seen as a convex relaxation of the nonconvex program giving all subsets

All Subsets Regression

First, identify all considered features and transformations and put them in the feature matrix $\mathbb{X} \in \mathbb{R}^{n \times p}$

Best subset selection algorithm: For $k = 1, \ldots, p$

1. Find $\hat{R}$ for the $\binom{p}{k}$ models of size k
2. Save the model that minimizes $\hat{R}$

(This can be found in Algorithm 6.1 in ISL)

Now, report the model that minimizes a risk estimation method over these p models

(Such as AIC, BIC, Mallows Cp, ...)

In general, this is a nonconvex problem, though some shortcuts can be taken

(A general idea known as "Branch and Bound")

All Subsets Regression in R

We can use the function regsubsets in the package leaps

The syntax and associated objects look like:

```
allsubsets.out = regsubsets(Y~.,data=X,nvmax=pmax)
> summary(allsubsets.out)
[1] "which"  "req"  "rss"  "adjr2"  "cp"  "bic  "outmat"  "obj"
```

- The nvmax = pmax controls the max size of models considered. The default is 8 and that is usually far too small.
- Now, we can pick among the pmax models that minimize $\hat{R}$ for a given model size using BIC or Cp

All Subsets Regression: A Big Problem (Literally)

If there are p features then there are 2^p possible models

If $p = 40$ (which is considered a modest problem these days), then the number of possible models is

$$2^{40} \approx 1,099,512,000,000 \Rightarrow \text{More than 1 trillion!}$$

If $p = 265$, then the number of possible models is more than the number of atoms in the universe[1]

We must sift through the models in a computationally feasible way

[1] It is estimated there are 10^{80} atoms in the universe.

Greedy approximations

Forward selection

In the likely event that 2^p is too large to be searched over exhaustively, a common greedy approximation is the following

1. Find $\mathrm{GIC}(\emptyset)$: The GIC of the intercept only model
2. Search over all p single feature models, computing GIC for each one. Say including X_j minimizes GIC with a value $\mathrm{GIC}(X_j)$. If $\mathrm{GIC}(X_j) < \mathrm{GIC}(\emptyset)$, add X_j to the model and continue. Otherwise terminate
3. Now search over all $p-1$ models that contain X_j and find the $X_{j'}$ that minimizes GIC. If $\mathrm{GIC}(X_j, X_{j'}) < \mathrm{GIC}(X_j)$, add $X_{j'}$ to the model and continue. Otherwise terminate
4. $\ldots$

(See Algorithm 6.2 in ISL)

Forward selection

```
regsubsets(Y~.,data=X,nvmax=pmax,method='forward')
```

Pros:

- This approach can be used effectively in either the Big Data or High Dimensional regimes
- It tends to produce sensible answers that are not too different from all-subsets

Cons:

- Can get trapped in a poor local minimum

General stepwise selection

This algorithm can can adapted to..

- start with the full model and stepwise remove features. This is known as backward selection

  ```
  regsubsets(Y~.,data=X,nvmax=pmax,method='backward')
  ```

  (useful if the full model isn't too large and a superset of the important features is desired)
- consider both adding and removing features at each step. This is known as stepwise selection

  ```
  regsubsets(Y~.,data=X,nvmax=pmax,method='seqrep')
  ```

Important comments

After using any of these model selection approaches, we produce estimates $\hat{\beta}$ and predictions $\hat{Y} = \hat{\beta}^\top X_{\text{select}}$ where X_{select} includes only the selected features

This can be interpreted as these features are most important for predicting Y from the features included in $X \in \mathbb{R}^p$

(The usual caveats apply: linearity (correlation), there are surely some important coefficients left out/unimportant ones included)

If we run out = lm(Y$\sim \mathbb{X}_{\text{select}}$), then summary(out) will produce the usual significance tests:

$\rightarrow$ these are not valid after model selection

Important comments

- If we want to be sure to include all the important features, then we can use AIC or Cp + backward selection
- If we want to be sure to only include important features, then we can use BIC + forward selection
- If we want to do predictions, use AIC or Cp, but it isn't clear what selection method is the best

Important: As stated in "Building multiple regression models interactively" (1981) *Biometrics*

"The data analyst knows more than the computer...

failure to use that knowledge produces inadequate data analysis"

Important comments

Some famous comments about stepwise variable selection:

(From Frank Harrel, author of "Regression Modeling Strategies")

- The F and chi-squared tests quoted next to each variable on the printout do not have the claimed distribution.
- The method yields confidence intervals for effects and predicted values that are falsely narrow
- It yields p-values that do not have the proper meaning, and the proper correction for them is a difficult problem.
- It gives biased regression coefficients that need shrinkage
- It has severe problems in the presence of collinearity.
- It is based on methods (e.g., F tests for nested models) that were intended to be used to test prespecified hypotheses.
- Increasing the sample size does not help very much
- It allows us to not think about the problem.

(As an aside, all subsets regression solves none of these problems)

Big data

Selection methods can be used on very large data sets

In R, some pseudo-code for the first step:

```
AIC_star   = Inf
j_star = 0
for(j in 1:p){
  grabVec = rep('NULL',p)
  grabVec[j] = NA
  X = read.csv('bigDataSet.csv', colClasses=grabVec)
  AIC_x = AIC(X) #Get the AIC value for this vector
  if(AIC_x < AIC_star){
    AIC_star = AIC_x
    j_star = j
  }
}
```

(Note: the ideas from branch and bound can be used here as well)

Some more advanced/recent developments

There has been a lot of recent research about these topics:

- All-subsets:
  - There are more modern approaches to all-subsets than "branch and bound" based on mixed-integer programming. This opens all-subsets up to noticeably larger problems
    (See `https://projecteuclid.org/euclid.aos/1458245736`)
  - Often, the perspective is that if we could all-subsets it would lead to the best results. This isn't necessarily the case and is only true for "easy" problems
    (See `http://www.stat.cmu.edu/~ryantibs/papers/bestsubset.pdf` for a very readable paper)
- Post model selection inference:
  - As noted, the reported p-values after model-selection are not to be trusted. There are ways that p-values can be computed after model selection
    (See `http://www.stat.cmu.edu/~ryantibs/papers/lassoinf.pdf`)