Selective inference

Patrick Breheny

April 10, 2025

Introduction

- In this lecture, we will discuss two (somewhat) related approaches:
  - The *covariance test* for testing the significance of additional terms along a covariate path
  - *Selective inference*, or *post-selection inference*, in which we carry out tests/construct confidence intervals conditional on the selected model
- Both approaches are implemented in the R package `selectiveInference`

Motivation

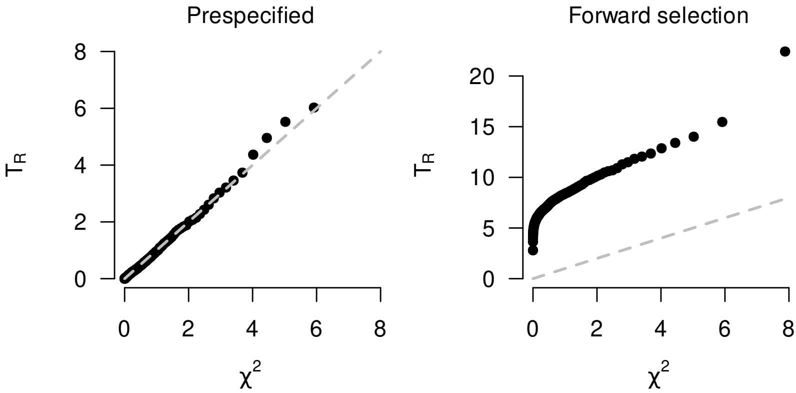
- The classical test for the significance of adding a variable to a model is to calculate the test statistic

$$T_R = \frac{\mathrm{RSS}_0 - \mathrm{RSS}_1}{\sigma^2}$$

  and compare it to a χ^2_1 distribution (for simplicity, let's assume σ^2 is known)
- This is valid, of course, when the variable is prespecified
- In the model selection context, however, it is far too liberal, as we have seen

Failure of χ^2 result

$n = p = 100$, $\boldsymbol{\beta} = \mathbf{0}$:



Covariance test: Motivation

- The goal of the covariance test is to develop a test statistic for an added variable whose distribution can be characterized despite the fact that we searched over a large number of candidate predictors to find it
- Let $\lambda_1 > \lambda_2 \cdots$ denote the values of the regularization parameter at which a new variable enters the model, and let $\mathcal{A}_{k-1}$ denote the active set, not including the variable added at step k
- Now, let us consider two solutions at the same value of λ:
    - $\widehat{\boldsymbol{\beta}}(\lambda_{k+1})$
    - $\widehat{\boldsymbol{\beta}}_{\mathcal{A}_{k-1}}(\lambda_{k+1})$

Covariance test

- Consider, then, the following quantity:

$$T_C = \frac{\mathbf{y}^\top \mathbf{X}\widehat{\boldsymbol{\beta}}(\lambda_{k+1}) - \mathbf{y}^\top \mathbf{X}\widehat{\boldsymbol{\beta}}_{\mathcal{A}_{k-1}}(\lambda_{k+1})}{\sigma^2};$$

  i.e., how much of the covariance between the fitted model and outcome can be attributed to the new predictor, as opposed to the increase in covariance we would get from simply lowering λ without adding any new variables

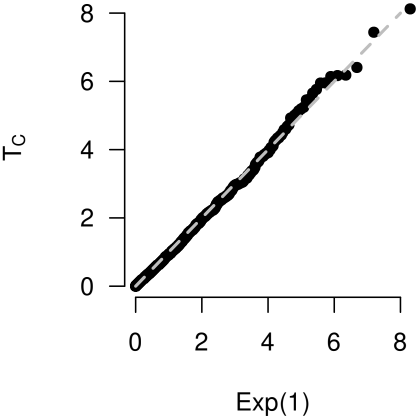
- Under the null hypothesis that all variables with $\beta_j^* \neq 0$ are included in $\mathcal{A}_{k-1}$, it can be shown that

$$T_C \xrightarrow{\mathrm{d}} \mathrm{Exp}(1)$$

  as n and $p \to \infty$

Accuracy of the approximating distribution

As before, $n = p = 100$, $\boldsymbol{\beta} = 0$, first predictor added:



$\mathbb{P}(T > 3.00) = 0.04$

Covariance test in R

- To carry out the covariance test in R, the model must first be fit using the `lar` function:

  ```
  fit <- lar(X, y)
  larInf(fit)
  ```

  the covariance test p-values are reported in the `CovTest` column and returned as `pv.covtest`
- This method requires an estimate of σ^2; the method has a default approach based on cross-validation and available with

  ```
  estimateSigma(X, y)
  ```

  or you can supply your own estimate

Results: Example data

Applying the covariance test to our running example:

Var	TS	p
A1	36.74	< 0.0001
A2	15.14	< 0.0001
A4	8.45	< 0.001
A3	5.25	< 0.01
A6	2.33	0.10
N39	0.44	0.64
A5	6.85	< 0.01
N25	0.07	0.93
N22	1.28	0.28
N9	0.79	0.45

Remarks

- When σ^2 is not known, but estimated, we can substitute $\hat{\sigma}^2$ for σ^2 and presumably the covariance test statistic would follow something like a $F_{2,\mathrm{rdf}}$ distribution, although the residual degrees of freedom (rdf) is not entirely clear in penalized regression
- It is possible to translate the sequential tests into an FDR rule, giving a false discovery rate for the included variables at a given point; for our example, this allows the first 5 variables to enter

Selective inference

- One downside of the covariance test is that we don't truly test individual features, but rather the improvement in the fit of the model as we lower λ
- *Selective inference*, also known as *post-selection inference*, offers a more comprehensive solution, providing post-selection confidence intervals and p-values for all of the terms in the selected model
- Similar to the covariance test, selective inference adjusts for the fact that we have "cherry-picked" the top k variables and eliminated $p - k$ other variables to arrive at the selected model, and conditions on this fact

The lasso selection event

- To proceed, we will need to explicitly characterize the condition that the lasso selects a given model $\mathcal{A}$ and eliminates the variables in set $\mathcal{B} = \mathcal{A}^C$
- **Theorem:** For a fixed value of λ, the event that the lasso sets $\widehat{\beta}_j = 0$ for all $j \in \mathcal{B}$ can be written

$$\|\frac{1}{n}\mathbf{X}_{\mathcal{B}}^{\top}(\mathbf{I} - \mathbf{P}_{\mathcal{A}})\mathbf{y} + \lambda\mathbf{X}_{\mathcal{B}}^{\top}\mathbf{X}_{\mathcal{A}}(\mathbf{X}_{\mathcal{A}}^{\top}\mathbf{X}_{\mathcal{A}})^{-1}\mathbf{s}\|_{\infty} \leq \lambda,$$

where $\mathbf{P}_{\mathcal{A}} = \mathbf{X}_{\mathcal{A}}(\mathbf{X}_{\mathcal{A}}^{\top}\mathbf{X}_{\mathcal{A}})^{-1}\mathbf{X}_{\mathcal{A}}^{\top}$, $\mathbf{s} = \mathrm{sign}(\widehat{\boldsymbol{\beta}}_{\mathcal{A}})$

Remarks

- Thus, the event that the lasso selects a certain model (and assigns its nonzero coefficients certain signs) can be written as a set of linear constraints $\{\mathbf{A}\mathbf{y} \leq \mathbf{b}\}$
- In other words, the set $\{\mathbf{y}|\mathbf{A}\mathbf{y} \leq \mathbf{b}\}$ is the set of random response vectors $\mathbf{y}$ that would yield the same active set and coefficient signs as the model we've selected

Selective inference: Big picture

- Now, suppose that $\mathbf{y} \sim \mathbf{N}(\boldsymbol{\mu}, \sigma^2\mathbf{I})$
- The main idea of selective inference is to make inference on $\boldsymbol{\mu}$, or more generally a linear combination $\theta = \boldsymbol{\eta}^\top\boldsymbol{\mu}$, conditional on the event $\{\mathbf{Ay} \leq \mathbf{b}\}$
- For example, we would likely be interested in $(\mathbf{X}_\mathcal{A}^\top\mathbf{X}_\mathcal{A})^{-1}\mathbf{X}_\mathcal{A}^\top\boldsymbol{\mu}$; to address this question we would set $\boldsymbol{\eta}$ equal to various columns of $\mathbf{X}_\mathcal{A}(\mathbf{X}_\mathcal{A}^\top\mathbf{X}_\mathcal{A})^{-1}$
- Carrying out this conditional inference turns out to be quite a bit easier than one would expect due to a remarkable result known as the *polyhedral lemma*, which we will state on the next slide without proof

Polyhedral lemma

Theorem: The conditional distribution of $\boldsymbol{\eta}^\top \mathbf{y} | \{\mathbf{A}\mathbf{y} \leq \mathbf{b}\}$ is equivalent to distribution of $\boldsymbol{\eta}^\top \mathbf{y}$ given

$$\begin{aligned} \mathcal{V}^-(\mathbf{y}) \leq \boldsymbol{\eta}^\top \mathbf{y} \leq \mathcal{V}^+(\mathbf{y}) \\ \mathcal{V}^0(\mathbf{y}) \geq 0, \end{aligned}$$

where

$$\begin{aligned} \boldsymbol{\alpha} &= \mathbf{A}\boldsymbol{\eta}/\|\boldsymbol{\eta}\|^2 \\ \mathcal{V}^-(\mathbf{y}) &= \max_{j:\alpha_j<0} \frac{b_j - (\mathbf{A}\mathbf{y})_j + \alpha_j \boldsymbol{\eta}^\top \mathbf{y}}{\alpha_j} \\ \mathcal{V}^+(\mathbf{y}) &= \max_{j:\alpha_j>0} \frac{b_j - (\mathbf{A}\mathbf{y})_j + \alpha_j \boldsymbol{\eta}^\top \mathbf{y}}{\alpha_j} \\ \mathcal{V}^0(\mathbf{y}) &= \min_{j:\alpha_j=0} \{b_j - (\mathbf{A}\mathbf{y})_j\} \end{aligned}$$

Using the lemma for inference

- The polyhedral lemma appears complex, but its upshot is actually quite simple: $\boldsymbol{\eta}^\top \mathbf{y}$ would ordinarily follow a normal distribution, but conditional on the model we have selected, it follows a truncated normal distribution with support determined by $\mathbf{A}$ and $\mathbf{b}$
- Thus, letting F denote the CDF of this truncated normal distribution, we can carry out hypothesis tests of $\theta = \boldsymbol{\eta}^\top \boldsymbol{\mu} = 0$ by calculating tail areas with respect to this distribution
- Likewise, we can construct confidence intervals by inverting the above tests, searching for values of θ satisfying $F(\theta) = \alpha/2$ and $F(\theta) = 1 - \alpha/2$

Selective inference in R

- Applying this method in R is slightly complicated; first, you have to standardize the design matrix and fit a model in glmnet:

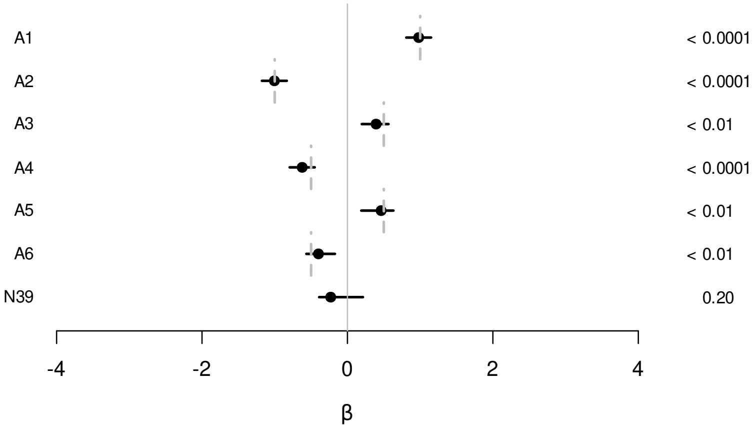
```
x_std <- std(x)
fit <- glmnet(x_std, y, standardize = FALSE)
```

- Then, because glmnet and selectiveInference use different objective functions, you need to rescale λ by n:

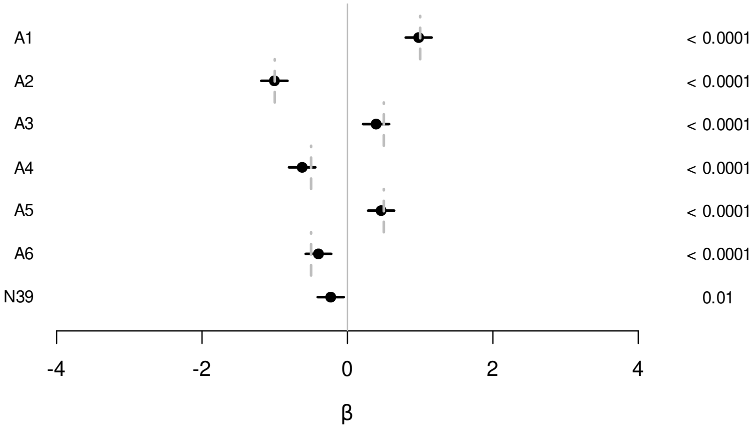
```
lam <- 25
b <- coef(fit, s = lam/n)[-1]
res <- fixedLassoInf(x_std, y, b, lam)
```

- As in larInf, this requires an estimate of σ^2

Application to example data



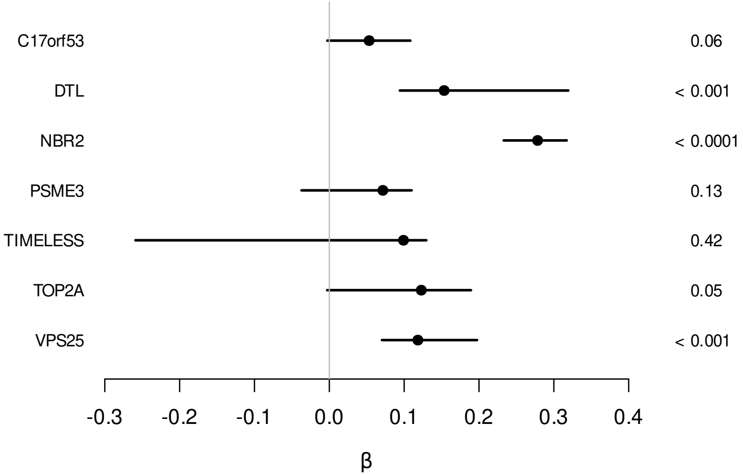
Lasso-OLS hybrid intervals



covTest: TCGA data

Var	TS	p
C17orf53	Inf	< 0.0001
VPS25	3.96	0.02
NBR2	2.82	0.06
DTL	7.15	< 0.001
PSME3	0.68	0.51
TOP2A	2.55	0.08
TIMELESS	1.45	0.23
CDC25C	0.41	0.66
CCDC56	0.69	0.50
CENPK	0.10	0.91

`selectiveInference`: TCGA data



Final remarks

- The covariance test lies somewhere in between the fully conditional and marginal definitions with respect to the error rate it controls
- The covariance test approach scales up reasonably well to high dimensions
- Selective inference is a very promising approach to inference that appears to work very well in the $n > p$ case
- How well it works in the $p > n$ case is debatable; for the TCGA data, once the 8th predictor enters the model, all confidence intervals become infinitely wide