Post-Selection Inference with R

Amit Meir

May 28, 2018

Department of Statistics, The University of Washington

- Motivation: The Lasso
- A Crash Course in Post-Selection Inference
- Software Packages
- Code Example
- Conclusion

Motivation: The Lasso

The Lasso

The Lasso is a regularized regression/model selection method,

$$\hat{\beta}_{\text{lasso}} = \arg\min_{\beta} \frac{1}{2n} \|y - X\beta\|_2^2 + \lambda \|\beta\|_1$$

- The ℓ_1 penalty induces sparsity (sets many coefficients to zero).
- Has many good properties.
    - Consistent for the true model under some conditions.
    - Consistent in ℓ_2 norm $\|\hat{\beta}_{\text{lasso}} - \beta\|_2^2 \to^{P} 0$ under mild conditions.
    - Computationally efficient!
- **Very widely used!**

The Lasso: Statistical Problems

1. Statistical significance?
2. Confidence intervals?
3. Efficient estimation?

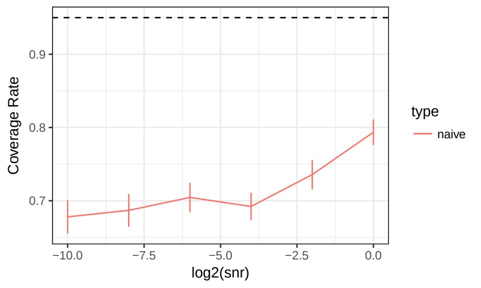
The Lasso: Statistical Problems

1. Statistical significance?
2. Confidence intervals?
3. Efficient estimation?

Whats wrong with the gaussian confidence intervals?

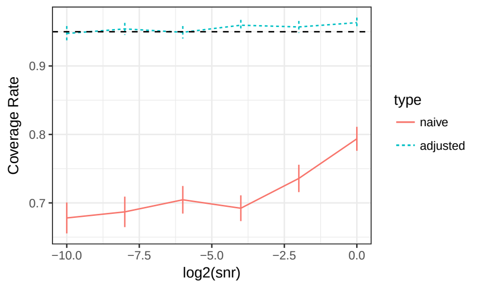
$$(\hat{\beta}_j - \sigma_j z_{1-\alpha/2}, \hat{\beta}_j + \sigma_j z_{1-\alpha/2})$$

Coverage Rate after Model Selection



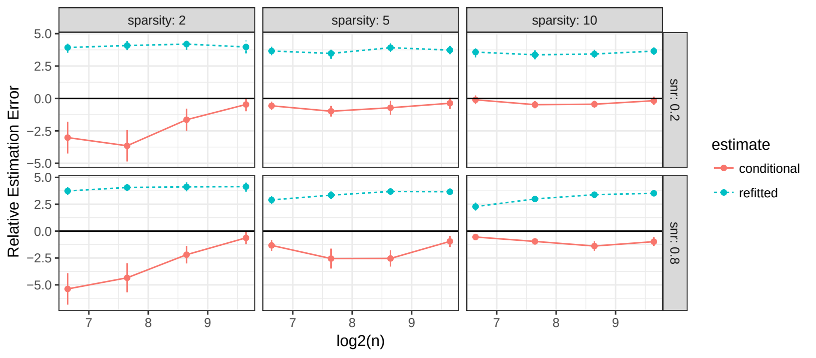
- The fact that we selected a model based on the data invalidated our confidence intervals

Coverage Rate after Model Selection



- The fact that we selected a model based on the data invalidated our confidence intervals
- But we can adjust for selection to get valid confidence intervals!

Estimation Error after Model Selection



$$\frac{1}{|M|}\sum_{j\in M}\log_2(\hat{\beta}_j-\beta_j)^2-\log_2(\hat{\beta}_j^{\text{lasso}}-\beta_j)^2$$

A Crash Course in Post-Selection Inference

Conditional Inference: Estimation with Testing

Suppose that $y \sim N(\mu, 1)$ and estimate μ only if:

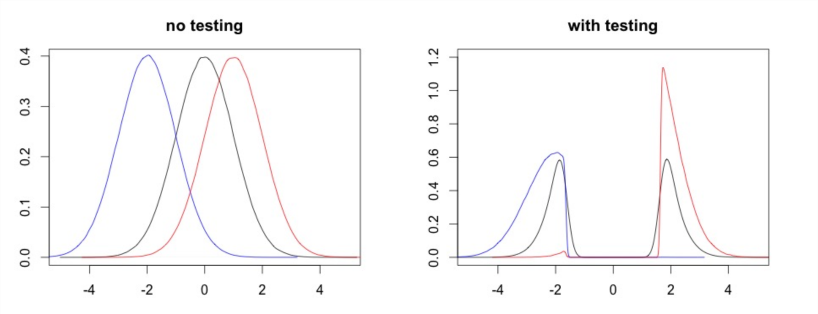
$$|y_i| \geq c > 0$$

Conditional Inference: Estimation with Testing

Suppose that $y \sim N(\mu, 1)$ and estimate μ only if:

$$|y_i| \geq c > 0$$

If $0 < \mu < c$ **we will always overestimate** μ **if we use the standard MLE,** y **itself.**



We assumed a distribution $y \sim N(\mu, 1)$. But If we only observe $|y| > c$ the actual observed distribution is a **Truncated Normal**.

The Univariate Conditional MLE

- The standard MLE maximizes a misspecified Likelihood.

The Univariate Conditional MLE

- The standard MLE maximizes a misspecified Likelihood.
- The correct likelihood is that of a truncated normal distribution:

$$L(\mu | \{|y| > c) = \frac{\varphi(y; \mu, 1)}{P(|y| > c)} I\{|y| > c\}$$

The Univariate Conditional MLE

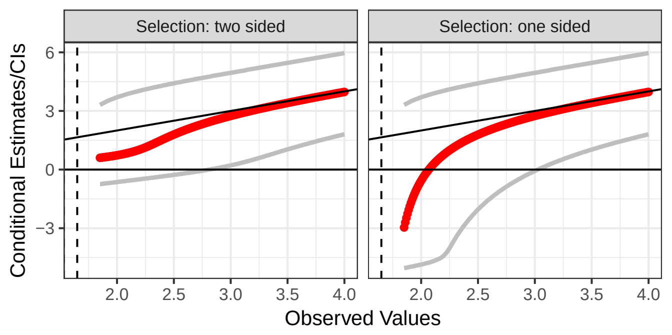
- The standard MLE maximizes a misspecified Likelihood.
- The correct likelihood is that of a truncated normal distribution:

$$L(\mu|\{|y| > c) = \frac{\varphi(y;\mu,1)}{P(|y| > c)} I\{|y| > c\}$$

- We can obtain a correct MLE by maximizing the conditional likelihood:

$$\hat{\mu} = \arg\max_{\mu} L(\mu|\{|y| > c\})$$

  We can also compute CIs based on the conditional likelihood.



Conditional estimates are adaptive shrinkage estimators:

- Apply shrinkage when observed value is close to the threshold.
- Report ‘naive’ estimates when observed values are far away from the threshold.

Software Packages

Available Software Packages

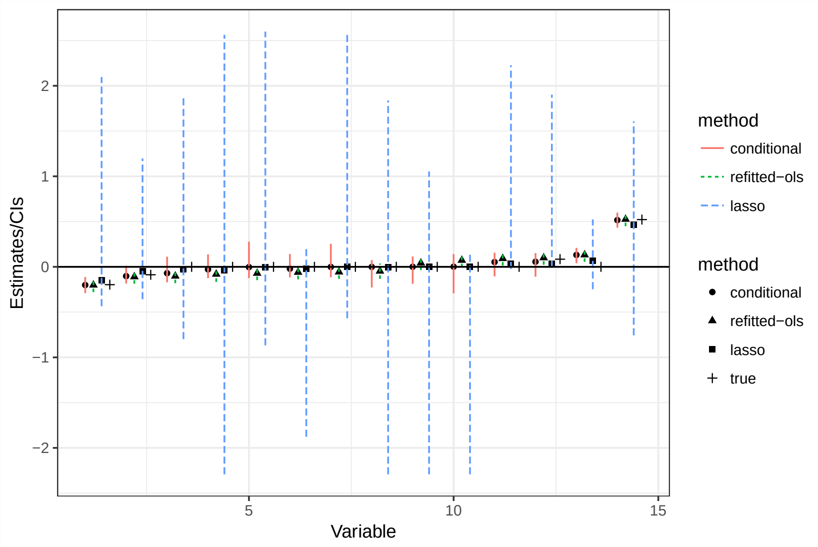
- **selectiveInference:** Post-selection inference based on the Polyhedral Lemma (Tibshiriani et al.).

- **selectiveMLE:** Computation of the conditional MLE for the Lasso + CIs based on a quadratic approximation (AM).

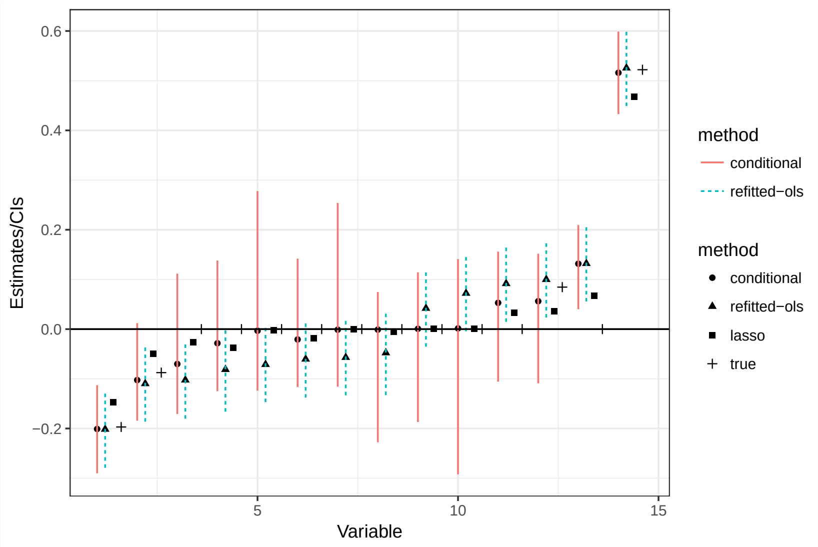
  !!Work in progress!! Overhaul planned:
  - More reliable/faster sampler.
  - CIs based on a modified bootstrap procedure.

- **PSAT:** Post-selection inference following aggregate testing based on the Polyhedral Lemma and other more efficient methods (AM & Ruth Heller).

Comparison of Post-Selection Inference Methods



Comparison of Post-Selection Inference Methods



Code Example

Loading Dependencies

```r
# devtools::install_github("ammeir2/selectiveMLE")
library(selectiveInference)
library(selectiveMLE)
library(ggplot2)
library(magrittr)
library(dplyr)
library(reshape2)

generate_sqrt_Sigma <- function(p, rho, sigsq = 1) {...}

generate_regression_data <- function(n, sqrtSigma, numberNonzero,
                                     snr = 2, ysig = 1) {...}
```

Generating Data

```
# Parameters ----------------
n <- 400
n <- 400
p <- 400
snrEta <- 0.5
numberNonzero <- 4
rho <- 0.5

# Generating Data ----------------------
set.seed(123)
Xsqrtsig <- generate_sqrt_Sigma(p, rho, sigsq = 1)$sqrt
s <- 0
while(s < 2 | s > n / 4) {
  regData <- generate_regression_data(n, Xsqrtsig, numberNonzero, snr = snr, ysig = 1)
  X <- regData$X[1:n, ]
  X <- apply(X, 2, function(x) (x - mean(x)) / sd(x))
  obsMu <- regData$mu[1:n]
  ysig <- sqrt(var(obsMu) / snrEta)
  y <- rnorm(n, mean = obsMu, sd = ysig)
  y <- y - mean(y)

  ysd <- sd(y)
  y <- y / ysd
  lassoFit <- cv.glmnet(X, y, standardize = FALSE, intercept = FALSE)
  lambda <- n * lassoFit$lambda.min
  lassoBeta <- as.vector(coef(lassoFit, s = lambda / n))[-1]
  selected <- lassoBeta != 0
  s <- sum(selected)
  yoracle <- regData$mu[1:n] + rnorm(n, sd = ysig)
  yoracle <- yoracle / ysd
}
```

SelectiveInference Package: Running Analysis

```r
# selectiveInference (Polyhedral CIs) ------------------
Xm <- X[, selected]
lassoysig <- sd((y - Xm %*% lassoBeta[lassoBeta != 0]))
selectiveysig <- lassoysig * sqrt(n / (n - s - 1))
selectiveFit <- fixedLassoInf(X, y, alpha = 0.05,
                              beta = coef(lassoFit, s = "lambda.min")[-1],
                              lambda = lassoFit$lambda.min * n,
                              sigma = selectiveysig)
selectiveFit
```

SelectiveInference Package: Output

```
Standard deviation of noise (specified or estimated) sigma = 0.835

Testing results at lambda = 30.776, with alpha = 0.050
```

Var	Coef	Z-score	P-value	LowConfPt	UpConfPt	LowTailArea	UpTailArea
10	-0.144	-3.348	0.624	-0.202	1.053	0.024	0.025
20	-0.080	-1.874	0.312	-0.687	0.393	0.025	0.025
24	-0.040	-0.920	0.793	-0.108	0.937	0.025	0.025
97	0.111	2.556	0.092	-0.110	0.875	0.025	0.025
105	0.044	0.859	0.964	-Inf	0.099	0.000	0.025
106	0.057	1.101	0.078	-0.367	Inf	0.000	0.000
118	0.093	2.151	0.507	-0.613	0.399	0.025	0.025
134	0.106	2.361	0.655	-0.924	0.179	0.025	0.025
136	0.090	1.705	0.031	-0.011	1.336	0.025	0.025
137	0.085	1.678	0.462	-0.400	0.241	0.025	0.025
154	0.119	2.753	0.200	-0.164	0.202	0.025	0.024
169	-0.412	-7.848	0.000	-0.795	-0.274	0.025	0.024
170	-0.065	-1.228	0.617	-0.225	0.624	0.025	0.025
192	-0.073	-1.706	0.355	-0.264	0.283	0.025	0.000
211	0.112	2.586	0.333	-0.304	0.304	0.025	0.025
235	0.133	3.076	0.129	-0.111	0.226	0.025	0.025
256	0.108	2.504	0.100	-0.077	0.363	0.025	0.025
277	-0.053	-1.240	0.637	-0.201	0.577	0.025	0.025
304	-0.078	-1.807	0.581	-1.234	1.931	0.025	0.025

selectiveMLE: Function Call

```r
# Conditional MLE ----------------
system.time(mle <- lassoMLE(y, X, lambda = "lambda.min",
                            ysig = NULL, lassoFit = lassoFit,
                            delay = 20, optimSteps = 1000,
                            sampSteps = 3000, stepRate = 0.8,
                            method = "selected",
                            verbose = TRUE))
```

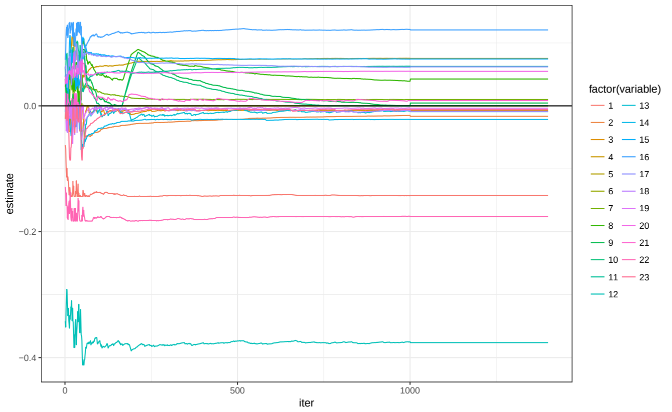
selectiveMLE: Function Call

```
> system.time(mle <- lassoMLE(y, X, lambda = "lambda.min",
+                             lassoFit = lassoFit,
+                             method = "selected",
+                             verbose = TRUE))
5% 10% 15% 20% 25% 30% 35% 40% 45% 50% 55% 60% 65% 70% 75% 80% 85% 90% 95% 100%
   user  system elapsed
  1.152   0.018   1.203
> system.time(exact <- lassoMLE(y, X, lambda = "lambda.min",
+                             ysig = NULL, lassoFit = lassoFit,
+                             method = "exact",
+                             verbose = TRUE))
5% 10% 15% 20% 25% 30% 35% 40% 45% 50% 55% 60% 65% 70% 75% 80% 85% 90% 95% 100%
   user  system elapsed
 41.027   0.149  41.665
```

selectiveMLE: Results

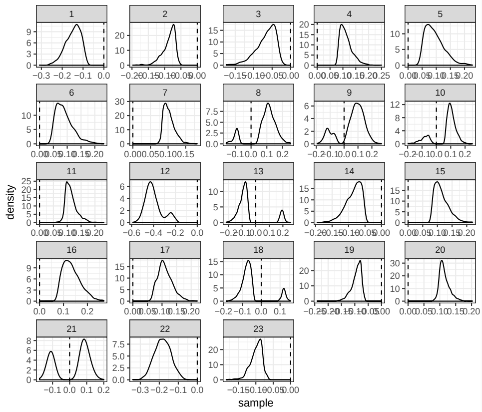
```
> mle$conditionalBeta[1:5] %>% round(3)
[1] -0.142 -0.017 -0.007  0.075  0.000
> exact$conditionalBeta[1:5] %>% round(3)
[1] -0.135 -0.012 -0.008  0.095  0.000
> exact$lassoBeta[1:5] %>% round(3)
[1] -0.062 -0.014 -0.007  0.029  0.001
> mle$wald_CI[1:5, ]
           [,1]        [,2]
[1,] -0.2347781 -0.02925365
[2,] -0.1310234  0.18836851
[3,] -0.1257605  0.16920867
[4,] -0.1078374  0.18816108
[5,] -0.2308295  0.20955257
```

selectiveMLE: Diagnostics




```r
# Solution Path -------------------------
solutionPath <- mle$solution_path[1:assumeConvergence, ]
solutionPath <- melt(solutionPath)
names(solutionPath) <- c("iter", "variable", "estimate")
ggplot(solutionPath) +
  geom_line(aes(x = iter, y = estimate, col = factor(variable))) +
  theme_bw() +
  geom_hline(yintercept = 0)
```

selectiveMLE: Diagnostics




```r
# Truncated Samples -------------------------
samples <- mle$coef_sample
samples <- samples[(assumeConvergence):(nrow(samples)), ]
samples <- melt(samples)
names(samples) <- c("replicate", "variable", "sample")
ggplot(samples) +
  geom_density(aes(x = sample, y = ..density..)) +
  facet_wrap(~ variable, scales = "free") +
  theme_bw() + geom_vline(xintercept = 0, linetype = 2)
```

Conclusion

Conclusion

1. Model selection invalidates standard inferential methods!
2. Solutions now exist which allow for model selection and inference to be conducted on the same dataset (with no data splitting).
3. **selectiveInference:** Is a great, easy to use software package.
4. **selectiveMLE:** Maximum likelihood estimation, and more efficient CIs - Soon[1]

[1] or at least, by the time the revision on the paper is due

Conclusion

1. Model selection invalidates standard inferential methods!
2. Solutions now exist which allow for model selection and inference to be conducted on the same dataset (with no data splitting).
3. **selectiveInference:** Is a great, easy to use software package.
4. **selectiveMLE:** Maximum likelihood estimation, and more efficient CIs - Soon[1]

Thank You! Questions?

[1] or at least, by the time the revision on the paper is due

References

Benjamini, Yoav, and Amit Meir. ”Selective Correlations-the conditional estimators.” arXiv preprint arXiv:1412.3242 (2014).

Meir, Amit, and Mathias Drton. ”Tractable Post-Selection Maximum Likelihood Inference of the Lasso.” arXiv preprint arXiv:1705.09417 (2017).

Heller, Ruth, Amit Meir, and Nilanjan Chatterjee. ”Post-selection estimation and testing following aggregated association tests.” arXiv preprint arXiv:1711.00497 (2017)

Lee, J. D., Sun, D. L., Sun, T., and Taylor, J. E. (2016). Exact post-selection inference, with application to the lasso. Annals of Statistics., 44(3):907-927