A practical evaluation of recent methods in high-dimensional inference

Charles Zheng

Stanford University

May 6, 2015

Problem and motivation

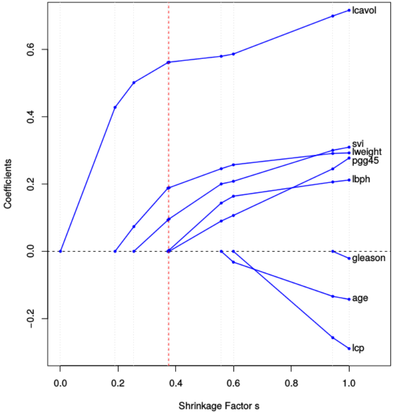
- $x \in \mathbb{R}^p, y \in \mathbb{R}$ have a joint distribution P where $y|x \sim N(x^T\beta, \sigma^2)$
- Observe $X = (x_1, \ldots, x_n)^T$, $Y = (y_1, \ldots, y_n)$ iid
- Problem: test $H_i : \beta_0 = i$ for $i = 1, \ldots, p$
- Motivation: x are SNPs (mutations), y is phenotype

Methods

	Control	$p > n$
Classical inference (Pearson 1930)	Marginal	No
Covariance test (Lockhart et al. 2014)	FWER?	Yes
Debiased lasso (Javanmard et al. 2014)	Marginal	Yes
Knockoffs (Barber et al. 2014)	FDR	?

The LASSO path

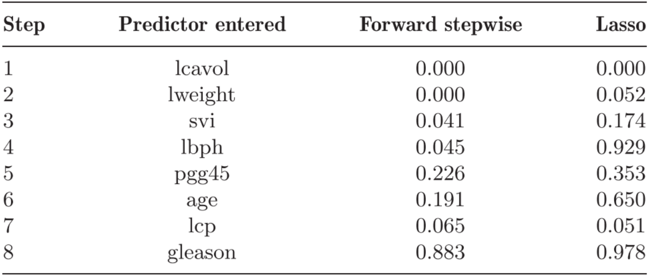
$$\hat{\beta}_\lambda = \text{argmin}_\beta \frac{1}{2}||X\beta - Y||^2 + \lambda||\beta||_1$$



(Image credit: ??)

Covariance test

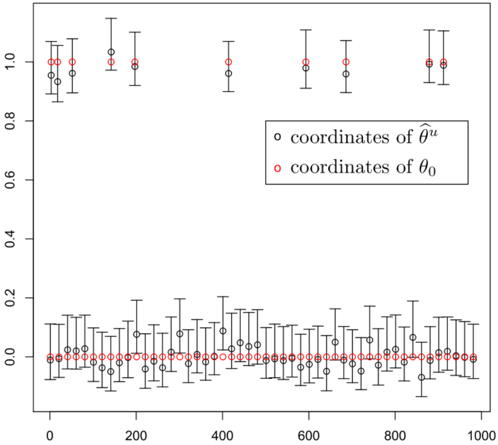
- (2014) Lockhart, Taylor, Tibshirani (x 2)
- Standard assumptions $Y \sim N(X\beta, \sigma^2 I)$ + large p asymptotics
- *See also* non-asymptotic exact test (Lee, Sun x 2, Taylor 2015)



Step	Predictor entered	Forward stepwise	Lasso
1	lcavol	0.000	0.000
2	lweight	0.000	0.052
3	svi	0.041	0.174
4	lbph	0.045	0.929
5	pgg45	0.226	0.353
6	age	0.191	0.650
7	lcp	0.065	0.051
8	gleason	0.883	0.978

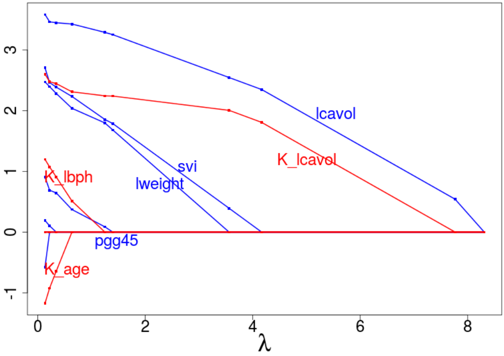
Debiased regularized M-estimators

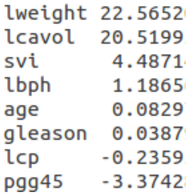
- (2014) Javanmard and Montanari
- Standard assumptions + sparsity condition on β + large *n* and *p* asymptotics



Knockoff filter

- (2014) Barber and Candés
- *Finite sample* $Y \sim N(X\beta, \sigma^2 I)$, $n \leq p$, control FDR
- Extension to $p > n$, FWER control, etc. forthcoming...





```
lweight  22.5652
lcavol   20.5199
svi       4.4871
lbph      1.1865
age       0.0829
gleason   0.0387
lcp      -0.2359
pgg45    -3.3742
```

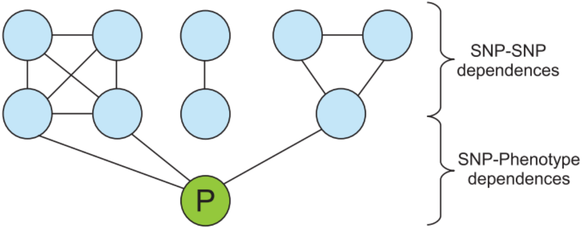
Methods

But what's actually used in practice?

	Control	$p > n$
Classical inference (Pearson 1930)	Marginal	No
Covariance test (Lockhart et al. 2014)	FWER?	Yes
Debiased lasso (Javanmard et al. 2014)	Marginal	Yes
Knockoffs (Barber et al. 2014)	FDR	?
Marginal screening	???	Yes

Regression vs Marginal Screening

Testing $H_i : \beta_i = 0$ is better than testing $H_i : \text{Cov}(X_i, Y) = 0$ when you are looking for X_i *directly* linked to Y



$\beta_i = 0$

$\beta_i \neq 0$

(Adapted from *Mourad 2012*)

Statistical Validation

- These procedures are derived under strong assumptions (linearity, gasusianity, homoscedasticty)
- How well do they work in real data where these assumptions are violated?
- We could validate inference procedures in real data if only we knew the '*true*' β, defined as

$$\beta = \mathbf{E}[xx^T]^{-1}\mathbf{E}[yx]$$

Statistical Validation

- These procedures are derived under strong assumptions (linearity, gasusianity, homoscedasticty)
- How well do they work in real data where these assumptions are violated?
- We could validate inference procedures in real data if only we knew the '*true*' β, defined as

$$\beta = \mathbf{E}[xx^T]^{-1}\mathbf{E}[yx]$$

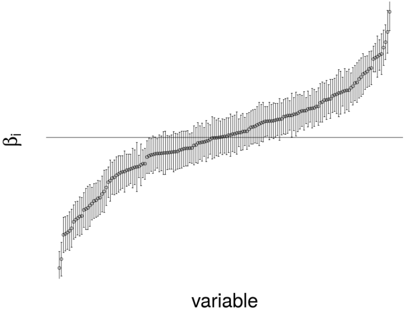
- Possibility: take a dataset with large p and *humongous* n, so we can get an extremely precise estimate of β using OLS. Then test the high-dimensional inference procedures on subsamples of size $n_0 \leq p < n$ of the data

Example: personality data

- Data with $p = 163$ survey questions from an online personality test, $n = 49086$ (after processing)
- Predict self-reported age of respondent, y, from their responses
- Is n large enough for us to confidently say which $\beta_i = 0$ (for use as ground truth?)

Example: personality data

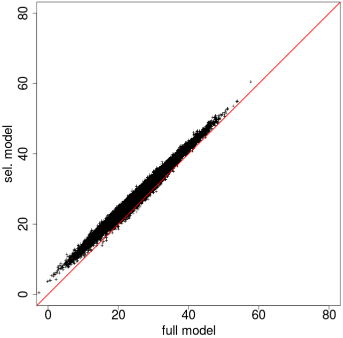
Coefficient estimates ± 3 sd



Consider declaring all variables whose intervals cross 0 to be null. Then $p_1 = 105$ (out of 163)

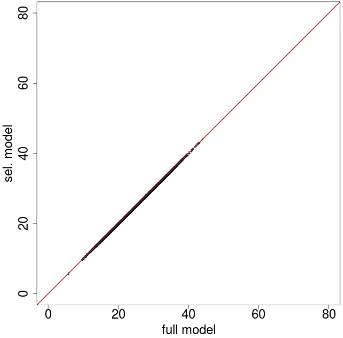
Example: personality data

- If n were large enough, then for the selected model S we should have $\hat{y} = \sum_{i=1}^{p} X\hat{\beta}_i$ close to $\hat{y}_S = \sum_{i\in S} X_i\hat{\beta}_i$
- But...



Example: personality data

- Here n is not large enough for $p = 163$
- If we reduce the dimensionality to 15 by subsampling columns, it looks more convincing that we selected the correct 10 variables



Dillemma

- It is by no means *impossible* to get large enough data to estimate high-dimensional β, with say, $p > 100$
- But if were *easy* to get such large n data... we wouldn't need these new inference techniques in the first place!

Why not use simulations?

- Simulations can be used to test robustness of the procedure
- In simulations, we can add all the nonlinearities, nongaussianity, etc. that we want

Why not use simulations?

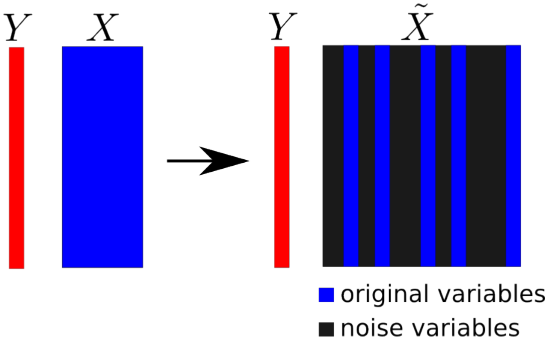
- Advantage: In simulations, we not only know β, but exactly how the data is generated
- Advantage: We can vary simulation parameters and get a lot of insight about the procedure being tested

Why not use simulations?

- Advantage: In simulations, we not only know β, but exactly how the data is generated
- Advantage: We can vary simulation parameters and get a lot of insight about the procedure being tested
- **Disadvantage:** Are these simulations relevant? How can we tell the simulated models are realistic?

Idea

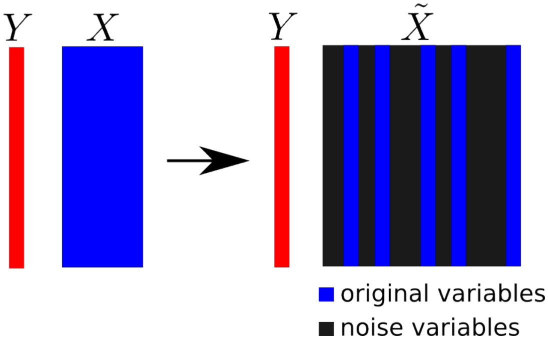
I give you real data *mixed in* with noise variables



- Can you identify the original columns from the noise columns?

Idea

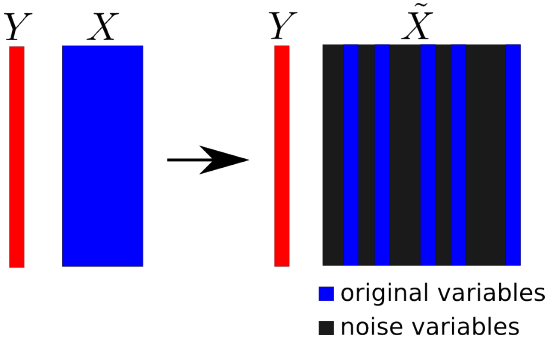
I give you real data *mixed in* with noise variables



- Can you identify the original columns from the noise columns?
- I can test your procedure this way, because I know the ground truth!

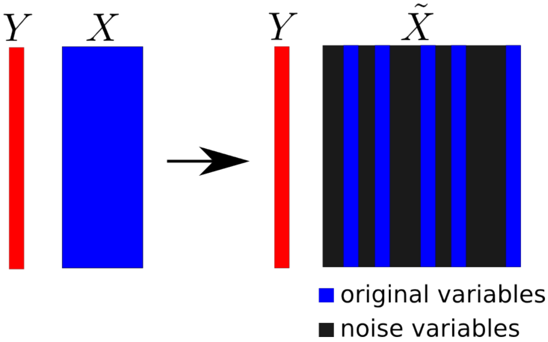
Idea

I give you real data *mixed in* with noise variables



- Can you identify the original columns from the noise columns?
- I can test your procedure this way, because I know the ground truth!
- **Caveat**: this test is unrealistically ‘easy’ (due to lack of correlations)

Synthetic Negative Controls



- Synthetic negative controls (SNCs) are artificial columns *which are correlated* to X, yet still have zero (population) regression coefficients
- Suppose I give you real data + SNCs, then you apply high-dimensional inference. If you reject any SNCs, we know these are errors!
- This gives us some measure of performance on “real” data (maybe?)

Synthetic Negative Controls

- Given random vector $x \in \mathbb{R}^p$, let e be noise in $\mathbb{R}^p$ independent of x.
- Let Γ be a fixed $p \times q$ matrix. *Define* synthetic negative controls $z \in \mathbb{R}^q$ by by

$$z = x'\Gamma + e$$

and let $\tilde{x} = (x, z)$, so that

$$\tilde{x}_1 = x_1, \ldots, \tilde{x}_p = x_p$$

$$\tilde{x}_{p+1} = z_1, \ldots, \tilde{x}_{p+q} = z_q$$

Synthetic Negative Controls

- Given random vector $x \in \mathbb{R}^p$, let e be noise in $\mathbb{R}^p$ independent of x.
- Let Γ be a fixed $p \times q$ matrix. *Define* synthetic negative controls $z \in \mathbb{R}^q$ by by
$$z = x'\Gamma + e$$
and let $\tilde{x} = (x, z)$, so that
$$\tilde{x}_1 = x_1, \dots, \tilde{x}_p = x_p$$
$$\tilde{x}_{p+1} = z_1, \dots, \tilde{x}_{p+q} = z_q$$
- Let
$$\beta = \mathbf{E}[xx^T]^{-1}\mathbf{E}[yx], \quad \tilde{\beta} = \mathbf{E}[\tilde{x}\tilde{x}^T]^{-1}\mathbf{E}[y\tilde{x}]$$

Synthetic Negative Controls

- Given random vector $x \in \mathbb{R}^p$, let e be noise in $\mathbb{R}^p$ independent of x.
- Let Γ be a fixed $p \times q$ matrix. *Define* synthetic negative controls $z \in \mathbb{R}^q$ by by

  $$z = x'\Gamma + e$$

  and let $\tilde{x} = (x, z)$, so that

  $$\tilde{x}_1 = x_1, \ldots, \tilde{x}_p = x_p$$

  $$\tilde{x}_{p+1} = z_1, \ldots, \tilde{x}_{p+q} = z_q$$

- Let

  $$\beta = \mathbf{E}[xx^T]^{-1}\mathbf{E}[yx], \quad \tilde{\beta} = \mathbf{E}[\tilde{x}\tilde{x}^T]^{-1}\mathbf{E}[y\tilde{x}]$$

- Then

  $$\forall i \in \{1, \ldots, p\} : \beta_i = \tilde{\beta}_i$$

  $$\forall i \in \{p+1, \ldots, p+q\} : \tilde{\beta}_i = 0$$

Why is this...?

- Recall that $\hat{\beta}_i$ is the *univariate regression* coefficient of Y on $X_{i|-i}$, where $X_{i|-i}$ is the *residual of* X_i after X_i is regressed on the other columns..
- Population version: $\beta_i = 0$ if the projection of X_i on the null space of the other covariates is uncorrelated with Y

Why is this...?

- Population version: $\beta_i = 0$ if the projection of X_i on the null space of the other covariates is uncorrelated with Y
- For $i = 1, \ldots, q$, we have

$$\tilde{X}_{p+i} = x'\Gamma_i + E_i$$

where here $\tilde{X}_{p+1}$ denotes the random variable (not the column of the design matrix)

Why is this...?

- Population version: $\beta_i = 0$ if the projection of X_i on the null space of the other covariates is uncorrelated with Y
- For $i = 1, \ldots, q$, we have

  $$\tilde{X}_{p+i} = x'\Gamma_i + E_i$$

  where here $\tilde{X}_{p+1}$ denotes the random variable (not the column of the design matrix)
- The orthogonal projection $P_X^\perp$ of $\tilde{X}_{p+1}$ is

  $$P_X^\perp \tilde{X} = P_X^\perp X\Gamma_i + P_X^\perp E_i = 0 + E_i$$

  since $P_X^\perp X = 0$; meanwhile since $E_i \perp X$, $P_X^\perp E_i = E_i$.

Why is this...?

- Population version: $\beta_i = 0$ if the projection of X_i on the null space of the other covariates is uncorrelated with Y

- The orthogonal projection $P_X^\perp$ of $\tilde{X}_{p+1}$ is
$$P_X^\perp \tilde{X} = P_X^\perp X \Gamma_i + P_X^\perp E_i = 0 + E_i$$
since $P_X^\perp X = 0$; meanwhile since $E_i \perp X$, $P_X^\perp E_i = E_i$.
- Since $E_i \perp y$, we have $\text{Cor}(P_X^\perp \tilde{X}_{p+i}, y) = 0$, hence $\tilde{\beta}_{p+i} = 0$

Why is this...?

- Population version: $\beta_i = 0$ if the projection of X_i on the null space of the other covariates is uncorrelated with Y

- Since $E_i \perp y$, we have $\text{Cor}(P_X^\perp \tilde{X}_{p+i}, y) = 0$, hence $\tilde{\beta}_{p+i} = 0$
- And since $\tilde{\beta}_j = 0$ for all the added variables $j = p+1, \ldots p+q$, it follows that $\tilde{\beta}_i$ is unchanged for $i = 1, \ldots, p$.

Using SNCs to evaluate procedures

- Take low-dimensional real data mixed with SNCs (synthetic negative controls), apply inference procedure
- *Proxy for Type I error:* Rejected SNCs
- *Proxy for Power:* Rejected original variables

A step-by-step tutorial (in R)

1. Take the prostate data

```
> data(prostate)
> x <- prostate[, 1:8]
> y <- prostate[, 9]
> colnames(x)
[1] "lcavol"  "lweight" "age"     "lbph"    "svi"
    "lcp"     "gleason" "pgg45"
> dim(x)
[1] 97 8
```

A step-by-step tutorial

2. Construct 20 synthetic negative controls

```
> GAMMA <- matrix(rnorm(8 * 20), 8, 20)
> E <- matrix(rnorm(97 * 20), 97, 20)
> sncs <- as.matrix(x) %*% GAMMA + 2 * E
> sncs <- data.frame(sncs)
> colnames(sncs)
 [1] "X1"  "X2"  "X3"  "X4"  "X5"  "X6" ...
[19] "X19" "X20"
```

3. Create combined design matrix

```
> x2 <- cbind(x, sncs)
```

A step-by-step tutorial

4. Try marginal screening

```
> cors <- cor(x2, y)
> cors[order(-abs(cors)), , drop = F]
              [,1]
lcavol   0.7344603
svi      0.5662182
lcp      0.5488132
X6      -0.4591506
X16      0.4482263
lweight  0.4333194
X4      -0.4326898
```

A step-by-step tutorial

5. Try covariance test

```
> library(covTest)
> covTest(lars(as.matrix(x2), y), as.matrix(x2), y)
$results
 Predictor_Number Drop_in_covariance P-value
                1            69.0292  0.0000
                5             1.5390  0.2219
                2             6.8094  0.0020
               11             0.8559  0.4294
```

(Numbers 1, 5, 2 are original, 11 is a SNC)

A step-by-step tutorial

6. Try debiased lasso (code at http://web.stanford.edu/ montanar/sslasso/)

```
> res <- SSLasso(as.matrix(x2), y)
[1] "10% done"
...
[1] "90% done"
> rej <- (res$up < 0) | (res$low > 0)
> names(x2)[rej]
[1] "lcavol"  "lweight" "svi"
```

A step-by-step tutorial

7. Try knockoffs

```
> library(knockoff)
> knockoff.filter(x2, y)
Call:
knockoff.filter(X = x2, y = y)

Selected variables:
lweight      X7
      2      15
```

Disclaimer!

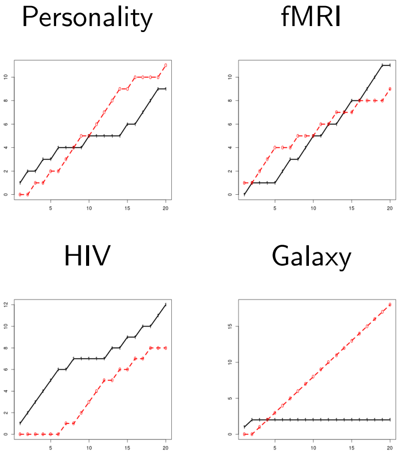
- I am *not* proposing SNCs as a methodology for *inference*
- There is a danger of inferring *that Type I error has been controlled* from lack of rejection of SNCs. There are no formal guarantees of this!
- One should interpret results from experiments with SNCs in the same way one interprets simulation results with purely synthetic data

More Experiments!

Data	n	p_1	Linear?	Gaussian?	Constant σ^2?
Personality	3000	163	No	No	No
fMRI	1750	53	No	OK	No
HIV	842	207	No	Yes?	OK?
Galaxy	323	4	No	OK	No

- We add $n/2 - p_1$ synthetic negative controls
- X is scaled, Γ is a gaussian matrix, $Var(E)$ is chosen to yield 'interesting' results
- Personality data is subsampled

Marginal Screening



Personality

fMRI

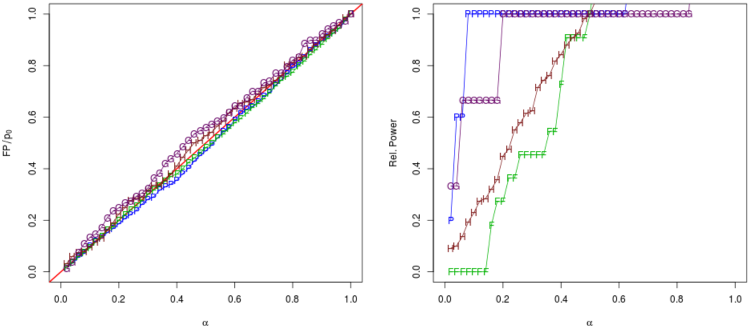
HIV

Galaxy

Legend: 0 = False positives, 1 = True positives

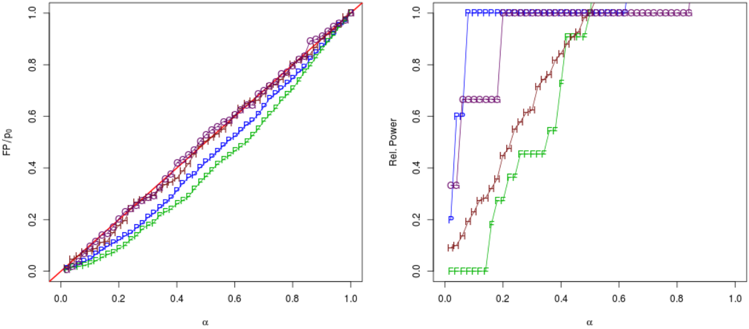
Ordinary Least Squares

“Rel. power” $=$ TP/(max number of TPs at $\alpha = 0.5$ for any method)



Legend: P $=$ Personality, F $=$ fMRI, H $=$ HIV, G $=$ Galaxy

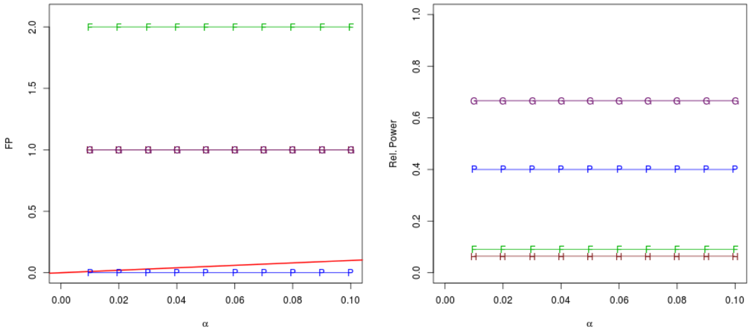
Debiased Lasso



Legend: P = Personality, F = fMRI, H = HIV, G = Galaxy

Covariance Test

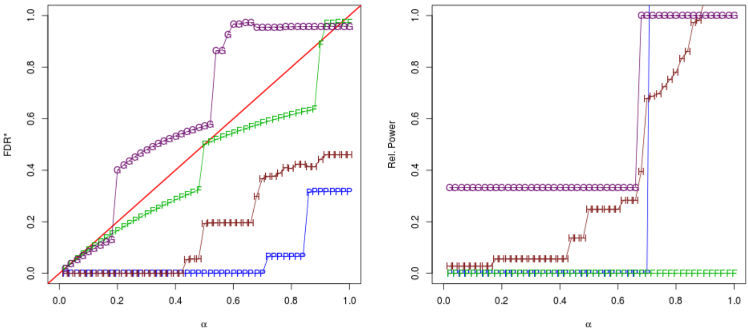
Procedure: continue rejecting until $p > \alpha$, then accept all other variables



Legend: P = Personality, F = fMRI, H = HIV, G = Galaxy

Knockoffs

Note: $FDR^* = \mathbf{E}[FP/(FP + TP + 1/\alpha)]$

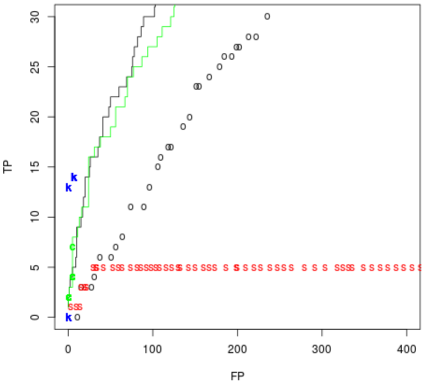


Legend: P = Personality, F = fMRI, H = HIV, G = Galaxy

Variable Ranking Criteria

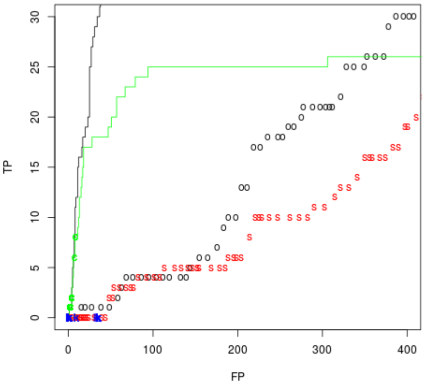
- Forget about Type I error for a second...
- Use procedures to *rank* variables by p-value
- Easy to compare procedures with different Type I criteria and also non-inference variable selection
- (Optional) score by Area Under Curve (AUC), etc.

Variable Ranking: Personality



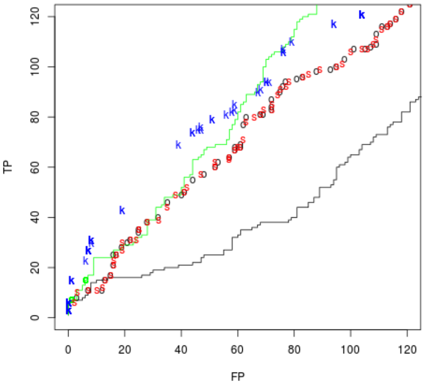
Legend: o = OLS, c = covariance test, k = knockoff, s = debiased lasso, (line) = marginal screening, (line) = lasso path

Variable Ranking: fMRI



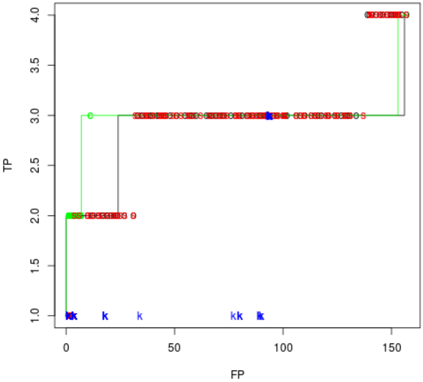
Legend: o = OLS, c = covariance test, k = knockoff, s = debiased lasso, (line) = marginal screening, (line) = lasso path

Variable Ranking: HIV



Legend: o = OLS, c = covariance test, k = knockoff, s = debiased lasso, (line) = marginal screening, (line) = lasso path

Variable Ranking: Galaxy



Legend: o = OLS, c = covariance test, k = knockoff, s = debiased lasso, (line) = marginal screening, (line) = lasso path

Commentary

- We should not conclude too much from four experiments with rather arbitrary generation parameters...
- Both OLS and debiased lasso have balanced Type I error
- Hard to interpret covariance test, but it clearly doesn't control FWER here
- Knockoffs do very well on the HIV data! (not surprising?)
- Marginal screening remains annoyingly effective...

How could this be useful?

- There are no formal guarantees... so what could we gain from using SNC experiments?

How could this be useful?

- There are no formal guarantees... so what could we gain from using SNC experiments?
- We can complement *theoretical guarantees* with *application-specific benchmarks*

How could this be useful?

- There are no formal guarantees... so what could we gain from using SNC experiments?
- We can complement *theoretical guarantees* with *application-specific benchmarks*
- Poor performance on benchmarks would tell us where our methods need improvement
    - Failure to control Type I error on benchmarks indicates a need for methods derived under weaker assumptions
    - Overly conservative Type I error control indicates a need for methods which are more adaptive to ‘easy’ cases

How could this be useful?

- There are no formal guarantees... so what could we gain from using SNC experiments?
- We can complement *theoretical guarantees* with *application-specific benchmarks*
- Poor performance on benchmarks would tell us where our methods need improvement
  - Failure to control Type I error on benchmarks indicates a need for methods derived under weaker assumptions
  - Overly conservative Type I error control indicates a need for methods which are more adaptive to 'easy' cases
- Possible to run a Kaggle-style competition for *inference* rather than prediction

How could this be useful?

- There are no formal guarantees... so what could we gain from using SNC experiments?
- We can complement *theoretical guarantees* with *application-specific benchmarks*
- Poor performance on benchmarks would tell us where our methods need improvement
  - Failure to control Type I error on benchmarks indicates a need for methods derived under weaker assumptions
  - Overly conservative Type I error control indicates a need for methods which are more adaptive to 'easy' cases
- Possible to run a Kaggle-style competition for *inference* rather than prediction
- Recognizing that different procedures can have differing strengths creates room for a diversity of approaches

Do SNCs reduce the need for scientific validation?

- Short answer: no
- Long answer: SNCs make sense if the true β is sparse. If true β is not sparse, hypothesis testing for regression doesn't make sense, and we need to reformulate the problem.

Do SNCs reduce the need for scientific validation?

- Short answer: no
- Long answer: SNCs make sense if the true β is sparse. If true β is not sparse, hypothesis testing for regression doesn't make sense, and we need to reformulate the problem.
- Ultimately, we need the practitioner to measure the value of the results we obtain from the inference procedure. That is the only way to check that we have the right formulation of the problem.

Closing thoughts

" Both the client and the statistician... must base their thinking on a recognition that their assumptions will always require review and reappraisal... "

– John Tukey

Acknowledgements

Thanks to Will Fithian for useful discussions.

References

- Barber, R., and Candes, E. (2014). Controlling the False Discovery Rate via Knockoffs. arXiv Preprint arXiv:1404.5609, 127. Retrieved from http://arxiv.org/abs/1404.5609
- Javanmard, A., and Montanari, A. (2014). Confidence intervals and hypothesis testing for high-dimensional regression. The Journal of Machine Learning Research, 15, 28692909. Retrieved from http://dl.acm.org/citation.cfm?id=2697057
- Lockhart, R., Taylor, J., Tibshirani, R. J., and Tibshirani, R. (2014). a Significance Test for the Lasso. Annals of Statistics, 42(2), 413468. doi:10.1214/13-AOS1175