1 Recap

In the previous classes, we considered the following general linear model

\[ y = X\mu + z \]  

(1)

where the noise, \( z \sim \mathcal{N}(0, \sigma^2 I) \). Given the observations \( y \), we had to estimate \( \mu \), where \( \mu \in \mathbb{R}^p \).

We proved that that soft-threshold estimator with threshold \( \lambda_p \approx 8\left(1 + \sqrt{\log p}\right) \), achieves an error within \( \log p \) factor of a genie-aided estimator. We also stated that this is the best estimator in the minimax sense.

2 Outline

In today’s class we will consider the above problem (1) when \( \mu \) is sparse. We begin by motivating the special case of sparsity. Thereafter we show that a threshold estimator is optimal (in minimax sense), if the sparsity of \( \mu \) is known. We then introduce FDR-threshold estimator, which is oblivious to \( \mu \)'s sparsity and is yet optimal. We then draw parallel between this problem and multiple-hypothesis testing problem with few non-nulls.

The material as well as the figures are taken from [1].

2.1 Agenda

- FDR - thresholding
- Sorted L-One Penalized Estimation (SLOPE)

Regime of interest

- Sparse \( \mu \).
- Asymptotic in the size of \( \mu \).
3 Motivation for the sparsity assumption

The underlying assumption of the learning, data-compression community is - Most of the data is sparse when represented in the right feature space. For example - Images and videos on the internet have compression ratio > 5. Consider the following time domain signal, in an orthonormal basis (say wavelet).

\[ y(t) = f(t) + \sigma z(t) \]

The representation of \( y(t) \) in orthonormal basis is given by

\[ \langle y, \psi_i \rangle = \langle f, \psi_i \rangle + \sigma \langle z, \psi_i \rangle \]

\[ y_i = \theta_i + \sigma z_i \]

If the signal \( y(t) \) is sparse in the \( \Psi \) basis, then we expect \( \theta_i \) to be sparse. Therefore the threshold-estimator for \( \theta_i \)'s will correspond to the best estimator for \( \hat{f}(t) \)
. The idea is represented in the pipeline below, where \( U \) denotes the transform operator.

\[ y \xrightarrow{\Psi} \{ U^T y = \theta \} \xrightarrow{\text{Threshold}} \hat{\theta} \xrightarrow{\Psi^{-1}} \{ U^T \hat{\theta} = \hat{f} \} \]

Since the transformation \( U^T \) is orthonormal, the error in the time domain is equal to the error in the \( \Psi \) domain. That is

\[ \mathbb{E}[||f - \hat{f}||^2] = \mathbb{E}[||\theta - \hat{\theta}||^2] \]

Example of the pipeline \( [2] \) is shown in the below figure.
4 Parallels with multiple-hypothesis testing problem

The thresholding estimators (soft and hard) considered in the last lecture and the previous section, estimate the parameter $\mu_i$ only using 'local' information $y_i$. In other words they are not taking global information into consideration. One could potentially obtain a 'better' estimator by using global information. Note that we observed a similar phenomenon in the multiple-hypothesis testing problem where the FDR-test (which used global info) was 'better' than benforroni test, which used only local information. Motivated by this line of reasoning consider the following FDR-hard-threshold estimator

$$\hat{u}_{(i)} = \begin{cases} y_{(i)} & |y_{(i)}| > t_{FDR} \\ 0 & \text{otherwise} \end{cases}$$

The estimator (3) is less conservative than the hard-threshold estimator because the threshold $t_{FDR}$ is lower than the Bonferroni threshold, which implies that this estimator has higher number of non-zero $\mu_i$'s. Therefore, this reduces the contribution of bias term in the error but increases the variance. The question is this: does the reduction in error due to the decrease in bias over-compensate the increase in error due to higher variance? Before jumping into the question, we shall first define few notions of sparsity.
5 Notions of sparsity

1. $\ell_0(\epsilon) = \{ \mu \in \mathbb{R}^n : \|\mu\|_0 \leq \epsilon n \}$ - The number of non-zero nulls is $\epsilon$ fraction of the number of parameters ($n$).

2. $\ell_p(\epsilon) = \{ \mu \in \mathbb{R}^n : \|\mu\|_p \leq c \}$ - The $p$-norm of the parameters is less than a constant.

3. $|\mu|_{(i)} \leq Ci^{-\alpha}$ - The $|\mu|_{(i)}$’s follow a power law.

In these notes, we shall only consider the $\ell_0(\epsilon)$ notion of sparsity.

5.1 Estimators with knowledge sparsity

Let $\epsilon = n^{-\beta}$, which implies that $\|\mu\|_0 = n^{1-\beta}$. For a given $\beta$, a hard-threshold estimator, $\hat{\mu}$, with threshold $\lambda_n(\epsilon) = \sigma \sqrt{2/\beta \log n}$, achieves the following guarantees

$$\sup_{\mu \in \ell_0(\epsilon)} \mathbb{E}\left[\|\hat{\mu} - \mu\|^2\right] = (1 + o_n(1)) R^*(\ell_0(\epsilon)),$$

where $R^*(\ell_0(\epsilon)) = \inf_{\hat{\mu}} \sup_{\mu \in \ell_0(\epsilon)} \mathbb{E}\left[\|\hat{\mu} - \mu\|^2\right]$.

Even though the above result looks pretty satisfying, the fact that the estimator $\hat{\mu}$ requires the knowledge of $\beta$ is a major concern. In practice the sparsity knowledge, $\beta$ is not available. The figure below gives an example where the above estimator $\hat{\mu}$ performs poorly by using an incorrect $\beta$ for thresholding.
Figure 2: Gaussian shift model with $n = 10,000$ and $\sigma_n = 1$. There are $n_0 = n^{1/4} = 10$ nonzero components $\mu_i = \mu_0 = 5.21$. Thus $\beta = 3/4$. Stars show ordered data $|y|_{(k)}$ and solid circles the corresponding true means. Dotted horizontal line is “correct” threshold $t_{3/4} = \sqrt{2^{3/4} \log n} = 3.72$, and dotted vertical lines show magnitude of the error committed with $t_{1/4}$. Solid horizontal line is a “misspecified” threshold $t_{1/2} = \sqrt{2^{1/2} \log n} = 3.03$ which would be the appropriate choice for $n_0 = n^{1/2} = 100$ nonzero components. Solid vertical lines show the additional absolute error suffered by using this misspecified threshold. Quantitatively, the absolute error $||\hat{\mu} - \mu||_1$ using the right threshold is 14.4 versus 70.0 for the wrong threshold. For $l_2$ error $||\hat{\mu} - \mu||_2^2$, the right threshold has error 38.8 and the wrong one has error 221.1.

5.2 FDR Thresholding

As see in the example in figure (2), the 'simple' thresholding estimator is not robust i.e if the estimator does not know the sparsity parameter $\beta$, it performs poorly.

It turns out that the FDR estimator (3) achieves near optimal guarantees without using the knowledge of $\beta$. In [1], the authors prove the following theorem:

**Theorem 1** ([1]). Under $\mu \in \ell_0(\epsilon_n)$, $\epsilon_n \in \left[\frac{1}{\log n}, n^{-\delta}\right]$, when $y \sim \mathcal{N}(\mu, \sigma^2 I)$, the FDR estimator (3), $\hat{\mu}$ has the following guarantees

$$
\sup_{\mu \in \ell_0(\epsilon)} \mathbb{E}\left[||\hat{\mu} - \mu||^2\right] = \left(1 + \frac{2q - 1}{2} + o_n(1)\right) R^*(\ell_0(\epsilon)).
$$

**Implications**

1. For $q < 1/2$ (and $q$ not too close to zero), FDR estimator (3) achieves the minimax error rate.

2. The threshold $t_{\text{FDR}}$ adapts to the unknown sparsity level.
Figure (3) illustrates the performance of the FDR-threshold estimator

![Graph](image)

Figure 3: (a) Gaussian shift model with $n = 10,000$ and $\sigma_n = 1$. There are $n_o = n^{1/4} = 10$ nonzero components $\mu_i = \mu_0 = 5.21$ and $\mu_i = 0$ for $i = \{11, 12, \ldots, n\}$. Solid line: ordered data $|y|(k)$. Solid circles: true unobserved mean value $\mu_i$ corresponding to observed $|y|(k)$. Dashed line: FDR quantile boundary $t_k = z(qk/2n), q = 0.05$. Last crossing at $k^F = 12$ producing threshold $t^F = 4.02$. Thus $|y|(10)$ and $|y|(12)$ are false discoveries out of a total of $k^F = 12$ discoveries. The empirical false discovery rate $\hat{FDR} = 2/12$. (b) 100 out of 10,000. $\mu_i = \mu_0 = 4.52$ for $i = 1, \ldots, n_0 = 100$; otherwise zero. Same FDR quantile boundary, $q = 0.05$. Now there are $k^F = 84$ discoveries, yielding $t^F = 3.54$ and $\hat{FDR} = 5/84$.

5.3 Connection with penalized estimation

Consider the following expression

$$\min_{\hat{\mu}} \frac{1}{2} \|y - \hat{\mu}\|^2 + \text{Pen}(\hat{\mu}). \quad (4)$$

In (4), the $\text{Pen}(\hat{\mu})$ is the measure of the estimator’s complexity. If

$$\text{Pen}(\hat{\mu}) = \sum_{j=1}^{||\hat{\mu}||_0} t_j^2 \quad t_j = \phi^{-1}(1 - q_j/2n) \approx \sqrt{2\log n/k}$$

in (4), we obtain

$$\arg\min_{\hat{\mu}} \frac{1}{2} \|y - \hat{\mu}\|^2 + \sum_{j=1}^{||\hat{\mu}||_0} t_j^2 \quad (5)$$

It turns out that the estimator given by (5) is close to the FDR-thresholding estimate. web
Figure 4: Comparison of FDR and power between the different methods. Sparsity is the number of non-zero $\beta_j$ regression coefficients.

6 Adaptivity of SLOPE

We first examine the problem of variable selection in linear models from the perspective of multiple testing. In this case, we have the following definitions of power and FDR.

\[ \text{power} = \frac{\# \text{ selected true variables}}{\# \text{ true variables}} \]

\[ \text{FDR} = \mathbb{E} \left[ \frac{\# \text{ selected false variables}}{\# \text{ selected variables} \lor 1} \right] \]

We examine the numerical example in Figure 4 in which the standard Lasso, cross-validated Lasso and SLOPE are compared. Our model is \( y \sim N(\beta, \sigma^2 I_p) \). We test at level \( q = 0.1 \) with \( n = p = 5000 \) and \( \sigma = 1 \). The non-zero regression coefficients are set to be \( \sqrt{2 \log p} \) in the spirit of the Bonferroni threshold. This setup makes the signal hard to distinguish from the noise.

Recall that in the case of an orthogonal design matrix, the Lasso applies the same soft-thresholding rule to all coordinates of the least squares estimator, which is just \( y \) in this case. Drawing an analogy with multiple testing, this is equivalent to comparing all p-values to a fixed threshold, which we know from Bonferroni, results in a very conservative method. Hence, as we expect, when we attempt to control the FDR with the Lasso by setting \( \lambda \) to reflect the Bonferroni threshold, we make very few discoveries and have low power. In the case of cross-validated Lasso, we choose \( \lambda_{cv} \) to minimize an estimate of the prediction error. We know that prediction error increases if important variables are omitted or their coefficient magnitudes are poorly estimated, or if irrelevant variables are selected. Hence, it turns out that \( \lambda_{cv} \) is very small so we end up with higher power, but also high FDR. On the other hand, SLOPE has better performance with reasonable power while maintaining FDR control. This is because of its adaptive nature which we will discuss next.
Recall that BHq is adaptive in the sense that the rejection threshold depends on the sparsity and magnitude of the true signals. If there are many large $\beta_j$s, then the last selected $\beta_j$ needs to pass a less stringent threshold. If the $\beta_j$ are all essentially zero, then the last selected $\beta_j$ needs to pass a more stringent threshold. This is illustrated in Figure 5.

SLOPE is inspired by BHq’s approach to FDR control in that larger coefficients are penalized more. With SLOPE we seek a solution to

$$\min_{\beta} \frac{1}{2} \|y - X\beta\|_2^2 + \sum_{j=1}^{p} \lambda_j |\beta|_{(j)}$$

where $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0$ and $|\beta|_{(1)} \geq |\beta|_{(2)} \geq \cdots \geq |\beta|_{(p)}$ are the ordered components of $|\beta|$. In particular, a la BHq, we may take $\lambda_{BH}(j) = \sigma \Phi^{-1}(1 - qj/2p)$. In the case of orthogonal design, this procedure is shown to control the FDR at level $q$ by Bogdan, van den Berg, Sabatti, Su and Candes (2015). Using these weights, we have the following numerical example in Figure 3 which illustrates how SLOPE is able to adapt its rejection threshold to the strength of signals.

**Question:** Under orthogonal design, why not do $\beta_i = (y_i - \lambda_i)_+$ instead?

1. Not clear how to move to general regression setup
2. It is also statistically bad because the map $y_i \rightarrow (y_i - \lambda_i)_+$ does not preserve the ordering of $y_i$.

It is interesting to point out that while the $\lambda_{BH}(j)$ are set with reference to BHq, the procedure is different from both the BHq method (step-up) and its step-down variant. In fact, the method is sandwiched between the step-up and step-down approaches: rejecting at most as many hypotheses...
Figure 6: Adaptivity of SLOPE to weak and strong signals. Red: selected variables. Blue: not selected.

as the step-up procedure and at least as many as the step-down procedure, while maintaining FDR control. This is shown in Figure 7 below. As the 2015 paper points out, control of the FDR is not a trivial consequence of this sandwiching behavior. This is a good point to remember procedures rejecting less than BHq do not control the FDR in general.
Figure 7: SLOPE is sandwiched between step-up and step-down version of BHq. Labeled are the SLOPE rejection point, and the BHq step-up and step-down points. Red: selected variables. Blue: not selected. Dashed line: SLOPE rejection threshold.

7 Asymptotic minimaxity

7.1 Orthogonal design

We consider the class of nearly black objects

$$\ell_0(k) = \{ \beta \in \mathbb{R}^p \mid \| \beta \|_0 \leq k \}$$

In the orthogonal design setting ($X'X = I_p$) we may assume $y \sim N(\beta, \sigma^2 I_p)$. As $p$ grows to infinity, if $k/p \to 0$ the minimax risk over $\ell_0(k)$ is known to satisfy

$$R(k) = \inf_{\beta} \sup_{\beta \in \ell_0(k)} \mathbb{E} \left\| \hat{\beta} - \beta \right\|_2^2 = 2\sigma^2 k \log \left( \frac{p}{k} \right) \cdot (1 + o(1))$$

and this asymptotic minimax risk is attained by both soft and hard thresholding at level $2\sigma \sqrt{\log(p/k)}$. However, performing this thresholding operation is impractical since $k$ is never known. In contrast, when the coefficients of SLOPE are chosen properly, it will attain the asymptotic minimax risk for every $k$, without prior knowledge of $k$. This is formalized as follows,

**Theorem 2** (Su and Candes ’15). In the orthogonal design, with $k/p \to 0$ as $p \to \infty$, if $\lambda_i = \sigma \Phi^{-1} (1 - qi/2p)$ for some $q \in (0, 1)$, then

$$\sup_{\beta \in \ell_0(k)} \mathbb{E} \left\| \hat{\beta}_{\text{SLOPE}} - \beta \right\|_2^2 = R(k) \cdot (1 + o(1)) = 2\sigma^2 k \log(p/k) \cdot (1 + o(1))$$

In the previous lecture we have seen that selecting the coefficients of $\beta$ according to BHq also attains the minimax risk without knowledge of $k$. However, the conditions there were more limiting: $q$ had to be in the range $(0, 0.5)$ and $k$ could be neither to small or too large (had to satisfy
\[ \log^5 p \leq k \leq p^{1-\delta} \text{ for some } \delta > 0 \). The inherent advantage of SLOPE over “testimation” with BHq is that SLOPE performs shrinkage, and hence pays far less than BHq for the false discoveries it makes, especially close to its adaptively chosen cutoff. More importantly, it is not clear how to use BHq for efficient estimation for general design; simply applying it on the least squares estimate of \( \beta \) is equivalent to BHq on correlated noise, which suffers from high variability, as discussed above. In contrast, SLOPE is readily applicable for general design and has good performance empirically, as well as some guarantees as explained below.

### 7.2 Random Design

Currently no minimax guarantees are available for SLOPE on general fixed designs. However, for random Gaussian designs, the following result is known,

**Theorem 3** (Su and Candes ’15, sketch). For \( X_{ij} \sim \mathcal{N}(0, 1/n) \) i.i.d., if \( k/p \to 0 \) and \( k \log p/n \to 0 \) as \( p \to \infty \), and if \( \lambda_i \approx \sigma \Phi^{-1}(1 - q_i/2p) \) for some \( q \in (0, 1) \), then for every \( \epsilon > 0 \),

\[
\sup_{\beta \in \ell_0(k)} \mathbb{P} \left( \frac{\| \beta_{\text{SLOPE}} - \beta \|_2^2}{2\sigma^2k \log(p/k)} > 1 + \epsilon \right) \to 0
\]

The probability above is taken with respect to both the design matrix \( X \) and the noise added to \( X\beta \). It is interesting to note that this result depends on the sample size \( n \) only through the condition \( k \log p/n \to 0 \). In particular, \( p \) might be much greater than \( n \). This guarantee is essentially optimal, due to the following converse result,

**Theorem 4** (Su and Candes ’15). For \( X_{ij} \sim \mathcal{N}(0, 1/n) \) i.i.d., if \( k/p \to 0 \) and \( k \log p/n \to 0 \) as \( p \to \infty \), then for every \( \epsilon > 0 \),

\[
\inf_{\beta} \sup_{\beta \in \ell_0(k)} \mathbb{P} \left( \frac{\| \beta - \beta \|_2^2}{2\sigma^2k \log(p/k)} > 1 - \epsilon \right) \to 1
\]

That is, under Gaussian design, for any estimator \( \hat{\beta}(y) \) there will be at least one \( \beta \) for which it has no chance of doing significantly better \( 2\sigma^2k \log(p/k) \). Since by Theorem 3 SLOPE never does significantly worse than \( 2\sigma^2k \log(p/k) \), it is essentially optimal. SLOPE is currently the only known estimator with that property. In particular, there is no known method to choose a \( \lambda \) for Lasso obtains this performance.

These results can be extended to prediction error — Theorem 3 continues to hold if \( \| \beta_{\text{SLOPE}} - \beta \|_2^2 \) is replaced by \( \| X\beta_{\text{SLOPE}} - X\beta \|_2^2 \) and similarly Theorem 4 holds when \( \| \beta - \beta \|_2^2 \) is replaced by \( \| X\beta - X\beta \|_2^2 \). For further reading on this topic, see **SLOPE is adaptive to unknown sparsity and asymptotically minimax** by Su and Candes (2015).
8 Computing SLOPE

The SLOPE estimator is a solution to the optimization problem

\[
\min_{\beta \in \mathbb{R}^p} \left\{ \frac{1}{2} \| y - X\beta \|_2^2 + \sum_{i=1}^{p} \lambda_i |\beta|_{(i)} \right\}
\]

This problem is convex (strongly convex for \( X'X > 0 \)), since \( \sum_{i=1}^{p} \lambda_i |\beta|_{(i)} \) is a norm (whenever \( \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0 \)), and every norm is a convex function. For example, for \( p = 2 \) we have

\[
\lambda_1 |\beta|_{(1)} + \lambda_2 |\beta|_{(2)} = \lambda_2 \| \beta \|_1 + (\lambda_2 - \lambda_1) \| \beta \|_{\infty}
\]

which is evidently a norm since \( \lambda_2, \lambda_1 - \lambda_2 \geq 0 \). For general \( p \) we may use the fact that \( \| \beta \|_{(j)} = \sum_{i=1}^{j} |\beta|_{(j)} \) is a norm to immediately verify that \( \sum_{i=1}^{p} \lambda_i |\beta|_{(i)} = \sum_{i=1}^{p-1} (\lambda_i - \lambda_{i+1}) \| \beta \|_{(i)} + \lambda_p \| \beta \|_1 \) is a norm. Alternatively, we may use

\[
\sum_{i=1}^{p} \lambda_i |\beta|_{(i)} = \max_{\pi \in S_p} \sum_{i=1}^{p} \lambda_{\pi(i)} |\beta_i|
\]

where \( S_p \) is the set of all permutations of \( p \) elements, and verify directly that the sorted \( L_1 \) norm indeed satisfies all the requirements of a norm.

However, the fact that the problem is convex does not mean it can be efficiently solved. In particular, for SLOPE to be applicable to the largest datasets, its complexity can be at most quasi-linear in \( \max \{ n, p \} \). Even quadratic complexity becomes prohibitive when the data size exceeds \( 10^5 \).

8.1 Orthogonal design

For orthogonal design, the optimization problem becomes

\[
\hat{\beta}_{\text{SLOPE}}(y) = \arg \min_{\beta \in \mathbb{R}^{p}} \left\{ \frac{1}{2} \| y - \beta \|_2^2 + \sum_{i=1}^{p} \lambda_i |\beta|_{(i)} \right\}
\]

since as usual we may assume without loss of generality \( X'y = y \). Clearly, \( \text{sign}[\hat{\beta}_{\text{SLOPE}}(y)]_i = \text{sign} y_i \) while \( |[\hat{\beta}_{\text{SLOPE}}(y)]_i| \) depends only on \( |y_1|, \ldots, |y_p| \). Therefore, we may replace \( y_i \) with \( |y_i| \) and then modify the sign of the resulting estimate accordingly. Since we may also sort the elements of \( y \) and then permute the result back to the original order, we can assume without loss of generality \( y_1 \geq y_2 \geq \cdots \geq y_p \geq 0 \). Under this condition, \( \hat{\beta}_{\text{SLOPE}} \) must also have decreasing and positive elements, since otherwise swapping elements and/or changing signs (neither affecting the sorted \( L_1 \) norm) would decrease the cost. Therefore, in the orthogonal design computation of SLOPE is equivalent to the quadratic program

\[
\min_{\beta \in \mathbb{R}^{p}} \sum_{i=1}^{p} \frac{1}{2} (y_i - \beta_i)^2 + \lambda_i \beta_i \\
\text{s.t. } \beta_1 \geq \beta_2 \geq \cdots \geq \beta_p \geq 0
\]

While convex quadratic programs can be solved in polynomial time via generic convex optimization methods, this will not yield anything close to the linear time solution we aim for. Fortunately, the
above problem is solved by the following simple method (where \((x)_+\) denotes the element-wise positive part of vector \(x\)):

1. If \(y - \lambda\) has non-increasing entries, set \(\hat{\beta}_{\text{SLOPE}} = (y - \lambda)_+\) and terminate.

2. Otherwise, for every increasing segment of \(y - \lambda\), replace the entries of \(y\) and \(\lambda\) belonging to that segment by their means. Go to step 1.

This method is guaranteed to converge to a solution for \(\text{SLOPE}\), and if implemented carefully it runs in linear time. With this implementation, the run time of the above procedure is several orders of magnitude less than the preprocessing time required to obtain \(y_1 \geq y_2 \geq \cdots \geq y_p \geq 0\).

The problem \(\text{SLOPE}\) is also equivalent to

\[
\min_{\beta \in \mathbb{R}^p} \sum_{i=1}^{p} \frac{1}{2} (y_i - \beta_i - \lambda_i)^2 \quad \text{s.t. } \beta_1 \geq \beta_2 \geq \cdots \geq \beta_p \geq 0
\]

which is the isotonic regression problem. In this context, the efficient solution discussed above is known as the Pooled Adjacent Violaters Algorithm (PAVA).

9 Concluding remarks

SLOPE was heavily inspired by the Benjamini-Hochberg procedure, and was originally designed as a method to control the FDR. The fact that both BHq and SLOPE can achieve the asymptotic minimax risk of estimating \(k\)-sparse Gaussian means that FDR controls offers much more than a type I error guarantee for multiple testing. Model selection according to the \(C_p\) criterion tends to treat too many non-signal components as signal when \(k = o(p)\), resulting in estimates with low bias but high variance. Attempts to mend the \(C_p\) method by using a higher, universal, threshold are akin to controlling the FWER. These methods tend to throw away too many signal components when \(k = \Omega(1)\), resulting in estimates with low variance but high bias. By choosing a threshold based on the observed data, BHq-like schemes select a fixed fraction of the \(k\) true signal components, while keeping the amount of noise selected below a fixed fraction of \(k\). Asymptotically, this strategy achieves the optimal tradeoff between bias and variance, leading to adaptive minimax estimation.

References
