Quantum Algorithms for the Pathwise Lasso

Tushar Vaidya

Joint work with: *Joao Doriguello, Debbie Lim, Chi Seng Pun and Patrick Rebentrost*

High dimensional regression

$$Y = X\beta + \epsilon$$

here X is $n \times d$ matrix, where $n \ll d$. More features, assets than observations. Thus traditional OLS solution $\beta = (X^\top X)^{-1} X^\top Y$ is not feasible. Sparsity is thus desired: $(X^\top X)$ is rank deficient.

What is Lasso?

β

Regressor Selection

$$\tilde{\beta} = \underset{\beta \in \mathbb{R}^d \setminus \{0\}}{\operatorname{argmin}} \|Y - X\beta\|_2^2 + \lambda \|\beta\|_1 \quad \text{for all } \lambda > 0.$$

We will use this set up. There are two cases to consider.

1. We seek a sparse portfolio to emulate Y such as in index tracking. The ℓ_1 penalty captures the transaction costs.
2. In an mean-variance portfolio, we can set $Y = 0$, thus our problem becomes

$$\tilde{\beta} = \underset{\beta \in \mathbb{R}^d \setminus \{0\}}{\operatorname{argmin}} \|X\beta\|_2^2 + \lambda \|\beta\|_1 \quad \text{for all } \lambda > 0.$$

This is the same as a standard mean-variance problem without the constraint of summing to 1.

Portfolio Selection - Sparse

We desire a sparse solution.

1. Not all assets are included in the portfolio so $\beta_{\mathcal{I}} = 0$ for some inactive set of assets.
2. Introducing the ℓ_1 norm induces sparsity but we are still left with the problem of choosing λ
3. Perhaps the answer is homotopy methods, where we seek a regularized path $\beta(\lambda)$. So the weights are a function of varying λ allowing for greater flexibility in choice.
4. LARS or Homotopy algorithm method gives us such a regularized path.

Classical Algorithm

Algorithm 1: Classical LARS algorithm for the pathwise Lasso

Input: Vector $\mathbf{y} \in \mathbb{R}^n$ and matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$

Output: Exact regularisation path $\mathcal{P}$

Initialise $\mathcal{A} = \operatorname{argmax}_{j \in [d]} |\mathbf{X}_j^\top \mathbf{y}|$, $\mathcal{I} = [d] \backslash \mathcal{A}$, $\lambda_0 = \|\mathbf{X}^\top \mathbf{y}\|_\infty$, $\hat{\boldsymbol{\beta}}(\lambda_0) = \mathbf{0}$, $t = 0$

while $\mathcal{I} \neq \varnothing$ **do**

  $\boldsymbol{\eta}_\mathcal{A} \leftarrow \operatorname{sign}\left(\mathbf{X}_\mathcal{A}^\top (\mathbf{y} - \mathbf{X}_\mathcal{A} \hat{\boldsymbol{\beta}}_\mathcal{A}(\lambda_t))\right)$

  $\boldsymbol{\mu} \leftarrow \mathbf{X}_\mathcal{A}^+ \mathbf{y}$ // $\boldsymbol{\mu} \in \mathbb{R}^{|\mathcal{A}|}$

  $\boldsymbol{\theta} \leftarrow (\mathbf{X}_\mathcal{A}^\top \mathbf{X}_\mathcal{A})^+ \boldsymbol{\eta}_\mathcal{A}$ // $\boldsymbol{\theta} \in \mathbb{R}^{|\mathcal{A}|}$

  $\Lambda_i^{\text{cross}} \leftarrow \frac{\mu_i}{\theta_i} \cdot \mathbf{1}\left[\frac{\mu_i}{\theta_i} \leqslant \lambda_t\right] \quad \forall i \in \mathcal{A}$

  $\Lambda_i^{\text{join}} \leftarrow \frac{\mathbf{X}_i^\top (\mathbf{y} - \mathbf{X}_\mathcal{A} \boldsymbol{\mu})}{\pm 1 - \mathbf{X}_i^\top \mathbf{X}_\mathcal{A} \boldsymbol{\theta}} \quad \forall i \in \mathcal{I}$

  $i_{t+1}^{\text{cross}} \leftarrow \operatorname{argmax}_{i \in \mathcal{A}} \{\Lambda_i^{\text{cross}}\}$ and $\lambda_{t+1}^{\text{cross}} \leftarrow \max_{i \in \mathcal{A}} \{\Lambda_i^{\text{cross}}\}$

  $i_{t+1}^{\text{join}} \leftarrow \operatorname{argmax}_{i \in \mathcal{I}} \{\Lambda_i^{\text{join}}\}$ and $\lambda_{t+1}^{\text{join}} \leftarrow \max_{i \in \mathcal{I}} \{\Lambda_i^{\text{join}}\}$

  $\lambda_{t+1} \leftarrow \max\{\lambda_{t+1}^{\text{join}}, \lambda_{t+1}^{\text{cross}}\}$

  $\hat{\boldsymbol{\beta}}_\mathcal{A}(\lambda_{t+1}) \leftarrow \boldsymbol{\mu} - \lambda_{t+1} \boldsymbol{\theta}$

  if $\lambda_{t+1} = \lambda_{t+1}^{\text{cross}}$ **then**

    Move i_{t+1}^{cross} from $\mathcal{A}$ to $\mathcal{I}$

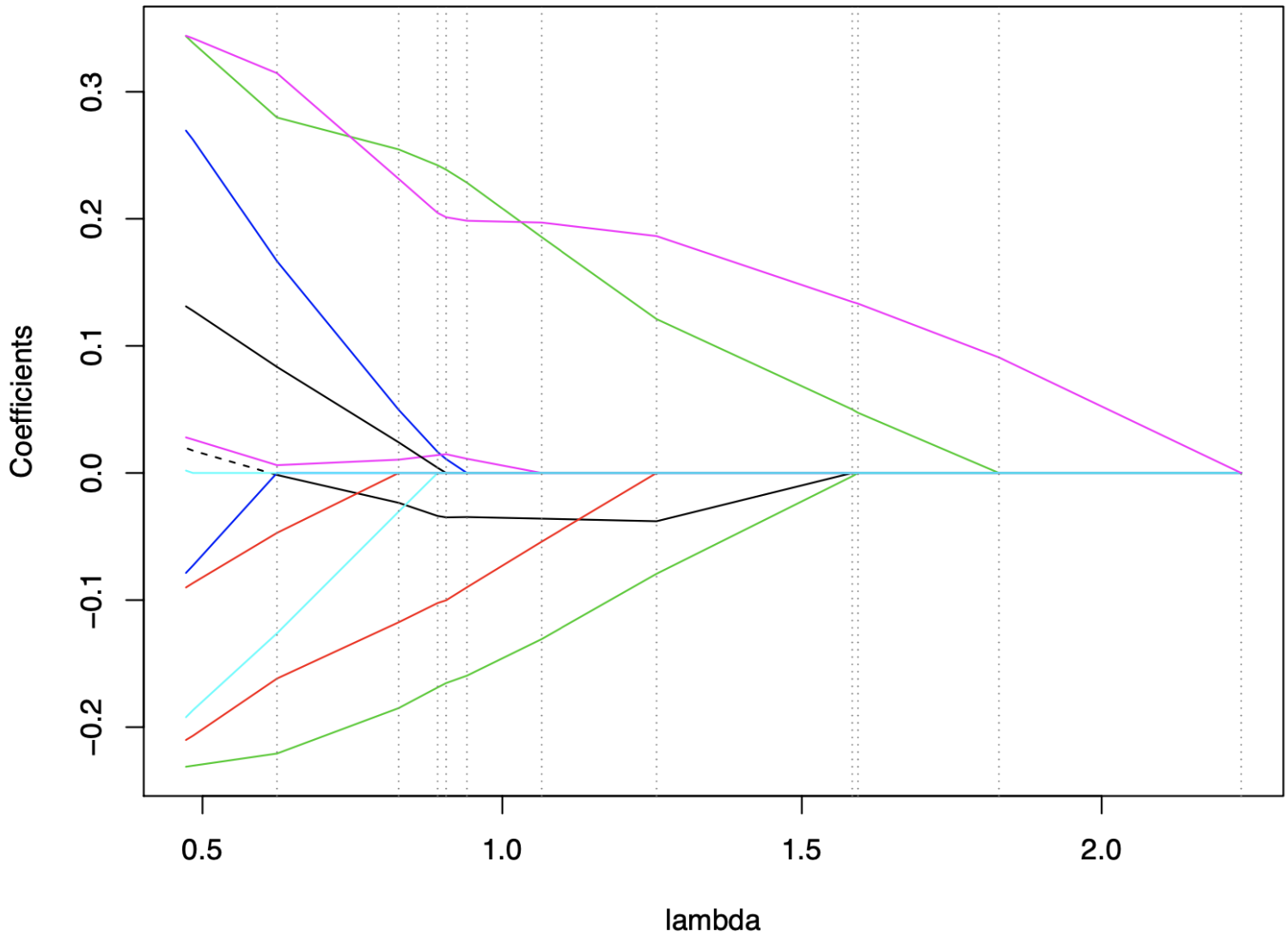
  else

    Move i_{t+1}^{join} from $\mathcal{I}$ to $\mathcal{A}$

  $t \leftarrow t + 1$

return coefficients $[(\lambda_0, \hat{\boldsymbol{\beta}}(\lambda_0)), (\lambda_1, \hat{\boldsymbol{\beta}}(\lambda_1)), \ldots]$

Lasso Path



Why Quantum

Evaluating a regularized path is highly non-trivial and in fact an NP-hard problem. How to determine τ for the the next λ.
Quantizing parts of the classical algorithm may help speed up.
As complexity of the classical algorithm is
We will analyse this algorithm and its variants.

How our work is new and different

Evaluating a regularized path is highly non-trivial and in fact an NP-hard problem. How to determine τ for the the next λ. Existing work focuses on a fixed λ

$$\tilde{\beta} = \underset{\beta \in \mathbb{R}^d}{\operatorname{argmin}} \|Y - X\beta\|_2^2 + \lambda \|\beta\|_1 \quad \text{fixed } \lambda > 0.$$

Definition

Given $\epsilon \geq 0$, we say that a vector $\tilde{\boldsymbol{\beta}} \in \mathbb{R}^d$ is an approximate Lasso solution with error $\epsilon(\lambda)$ with $\lambda > 0$ if

$$\frac{1}{2}\|\mathbf{y} - \mathbf{X}\widetilde{\boldsymbol{\beta}}\|_2^2 + \lambda\|\widetilde{\boldsymbol{\beta}}\|_1 - \left(\min_{\boldsymbol{\beta} \in \mathbb{R}^d} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda\|\boldsymbol{\beta}\|_1\right) \leq \epsilon(\lambda)$$

Aprroximate quantum LARS

Algorithm 3: Approximate quantum LARS algorithm for the pathwise Lasso

Input: $T \in \mathbb{N}$, $\delta \in (0,1)$, $\epsilon > 0$, $\mathbf{y} \in \mathbb{R}^n$, $\mathbf{X} \in \mathbb{R}^{n \times d}$, and $\{\alpha_{\mathcal{A}} \geqslant \|\mathbf{X}_{\mathcal{A}}^{+}\mathbf{X}_{\mathcal{A}^c}\|_1\}_{\mathcal{A} \subseteq [d]:|\mathcal{A}| \leqslant T}$

Output: Regularisation path $\widetilde{\mathcal{P}}$ with T kinks and error $\lambda\epsilon\|\widetilde{\boldsymbol{\beta}}(\lambda)\|_1$ with probability $\geqslant 1-\delta$

Initialise $\mathcal{A} = \operatorname{argmax}_{j \in [d]} |\mathbf{X}_j^\top \mathbf{y}|$, $\mathcal{I} = [d] \backslash \mathcal{A}$, $\lambda_0 = \|\mathbf{X}^\top \mathbf{y}\|_\infty$, $\widetilde{\boldsymbol{\beta}}(\lambda_0) = 0$, $t = 0$

while $\mathcal{I} \neq \varnothing$, $t \leqslant T$ **do**

  $\boldsymbol{\eta}_{\mathcal{A}} \leftarrow \frac{1}{\lambda_t}\mathbf{X}_{\mathcal{A}}^\top(\mathbf{y} - \mathbf{X}_{\mathcal{A}}\widetilde{\boldsymbol{\beta}}_{\mathcal{A}}(\lambda_t))$

  $\boldsymbol{\mu} \leftarrow \mathbf{X}_{\mathcal{A}}^{+}\mathbf{y}$ // $\boldsymbol{\mu} \in \mathbb{R}^{|\mathcal{A}|}$

  $\boldsymbol{\theta} \leftarrow (\mathbf{X}_{\mathcal{A}}^\top \mathbf{X}_{\mathcal{A}})^{+}\boldsymbol{\eta}_{\mathcal{A}}$ // $\boldsymbol{\theta} \in \mathbb{R}^{|\mathcal{A}|}$

  Classically compute $\mathbf{y} - \mathbf{X}_{\mathcal{A}}\boldsymbol{\mu}$ and $\mathbf{X}_{\mathcal{A}}\boldsymbol{\theta}$ and input them into QRAMs and KP-trees

  Define $\Lambda_i^{\text{join}} \triangleq \frac{\mathbf{X}_i^\top(\mathbf{y}-\mathbf{X}_{\mathcal{A}}\boldsymbol{\mu})}{\pm 1 - \mathbf{X}_i^\top \mathbf{X}_{\mathcal{A}}\boldsymbol{\theta}}$ and $\Lambda_i^{\text{cross}} \triangleq \frac{\mu_i}{\theta_i} \cdot \mathbf{1}\left[\frac{\mu_i}{\theta_i} \leqslant \lambda_t\right]$

  $i_{t+1}^{\text{cross}} \leftarrow \operatorname{argmax}_{i \in \mathcal{A}}\{\Lambda_i^{\text{cross}}\}$ and $\lambda_{t+1}^{\text{cross}} \leftarrow \max_{i \in \mathcal{A}}\{\Lambda_i^{\text{cross}}\}$

  Obtain $\widetilde{i}_{t+1}^{\text{join}} \in \left\{j \in \mathcal{I} : \Lambda_j^{\text{join}} \geqslant \left(1 - \frac{\epsilon}{1+\alpha_{\mathcal{A}}}\right)\max_{i \in \mathcal{I}}\{\Lambda_i^{\text{join}}\}\right\}$ with failure probability $\frac{\delta}{T}$ (Fact 14 and Lemma 16)

  $\widetilde{\lambda}_{t+1}^{\text{join}} \leftarrow \left(1 - \frac{\epsilon}{1+\alpha_{\mathcal{A}}}\right)^{-1}\Lambda_{\widetilde{i}_{t+1}^{\text{join}}}^{\text{join}}$

  $\lambda_{t+1} \leftarrow \min\{\lambda_t, \max\{\lambda_{t+1}^{\text{cross}}, \widetilde{\lambda}_{t+1}^{\text{join}}\}\}$

  $\widetilde{\boldsymbol{\beta}}_{\mathcal{A}}(\lambda_{t+1}) \leftarrow \boldsymbol{\mu} - \lambda_{t+1}\boldsymbol{\theta}$

  if $\lambda_{t+1} = \lambda_{t+1}^{\text{cross}}$ **then**

    Move i_{t+1}^{cross} from $\mathcal{A}$ to $\mathcal{I}$

  else

    Move $\widetilde{i}_{t+1}^{\text{join}}$ from $\mathcal{I}$ to $\mathcal{A}$

  $t \leftarrow t+1$

return coefficients $[(\lambda_0, \widetilde{\boldsymbol{\beta}}(\lambda_0)), (\lambda_1, \widetilde{\boldsymbol{\beta}}(\lambda_1)), \ldots]$

Lasso Complexity

Lemma 3 ([10]). *For any Y and X, the Lasso path algorithm performs at most*

$$\sum_{k=0}^{d} \binom{d}{k} 2^k = 3^d$$

iterations before termination.

Proof. Actually, we give a simple proof. This is just a simple application of the Binomial theorem

$$\sum_{k=0}^{d} \binom{d}{k} x^k = (1+x)^d$$

where $x = 2$. Hence, the identity holds. □

This can be reduced by putting a hard constraint on the number of regressors $m < \min(n, d)$ but we still have an exponential worst-case complexity.

Some Bounds - Fixed Design Matrix

Algorithm	**Error**	**Time complexity per iteration**
Classical LARS 1	None	$O(n\lvert\mathcal{I}\rvert + n\lvert\mathcal{A}\rvert + \lvert\mathcal{A}\rvert^2)$
Simple quantum LARS 2	None	$\widetilde{O}(n\sqrt{\lvert\mathcal{I}\rvert} + n\lvert\mathcal{A}\rvert + \lvert\mathcal{A}\rvert^2)$
Approximate classical LARS 4	$\lambda\epsilon\lVert\widetilde{\boldsymbol{\beta}}\rVert_1$	$\widetilde{O}\left(\frac{\gamma_{\mathcal{A}}^{-2}+n\lVert\mathbf{X}\rVert_{\max}^2\lVert\mathbf{X}_{\mathcal{A}}^{+}\rVert_2^2}{(1-\alpha_{\mathcal{A}})^2\epsilon^2}\lvert\mathcal{I}\rvert + n\lvert\mathcal{A}\rvert + \lvert\mathcal{A}\rvert^2\right)$
Approximate quantum LARS 3	$\lambda\epsilon\lVert\widetilde{\boldsymbol{\beta}}\rVert_1$	$\widetilde{O}\left(\frac{\gamma_{\mathcal{A}}^{-1}+\sqrt{n}\lVert\mathbf{X}\rVert_{\max}\lVert\mathbf{X}_{\mathcal{A}}^{+}\rVert_2}{(1-\alpha_{\mathcal{A}})\epsilon}\sqrt{\lvert\mathcal{I}\rvert} + n\lvert\mathcal{A}\rvert + \lvert\mathcal{A}\rvert^2\right)$

Table 1: Summary of results. Throughout this work, n is the number of observations, d is the number of features, $\mathcal{A}$ is the active set, and $\mathcal{I}$ is the inactive set. In addition, $\epsilon > 0$ is the approximation error, $\delta \in (0,1)$ is the failure probability, T is the number of kinks in the regularisation path $\mathcal{P}$, $\mathbf{X}_{\mathcal{A}} = [\mathbf{X}_i]_{i\in\mathcal{A}} \in \mathbb{R}^{n\times|\mathcal{A}|}$ is the matrix formed by the columns of $\mathbf{X}$ in $\mathcal{A}$, $\mathbf{X}_{\mathcal{A}}^{+}$ is the Moore–Penrose inverse of $\mathbf{X}_{\mathcal{A}}$, and the parameters $\alpha_{\mathcal{A}}, \gamma_{\mathcal{A}} \in (0,1)$ are defined in Result 3. The notation $\widetilde{O}(\cdot)$ omits $\operatorname{poly}\log$ terms in n, d, T, and δ.

Random Matrix

Algorithm	**Error**	**Time complexity per iteration**
Classical LARS 1	None	$O(nd)$
Simple quantum LARS 2	None	$\widetilde{O}(n\sqrt{d} + n\lvert\mathcal{A}\rvert)$
Approximate classical LARS 4	$\lambda\epsilon\lVert\widetilde{\boldsymbol{\beta}}\rVert_1$	$\widetilde{O}(d/\epsilon^2 + n\lvert\mathcal{A}\rvert)$
Approximate quantum LARS 3	$\lambda\epsilon\lVert\widetilde{\boldsymbol{\beta}}\rVert_1$	$\widetilde{O}(\sqrt{d}/\epsilon + n\lvert\mathcal{A}\rvert)$

Table 2: Summary of results for the case when the design matrix $\mathbf{X} \in \mathbb{R}^{n\times d}$ is a random Gaussian matrix with rows sampled i.i.d. from $\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$ with well-behaved covariance matrix $\boldsymbol{\Sigma} \succ \mathbf{0} \in \mathbb{R}^{d\times d}$ and the vector of observations $\mathbf{y} \in \mathbb{R}^n$ is such that $\lVert\mathbf{y}\rVert_1/\lVert\mathbf{y}\rVert_2 = \operatorname{poly}\log n$. These results are valid for active and inactive sets $\mathcal{A}$ and $\mathcal{I}$ of size $\lvert\mathcal{A}\rvert = O(n/\log d)$ and $\lvert\mathcal{I}\rvert = \Theta(d)$. The notation $\widetilde{O}(\cdot)$ omits $\operatorname{poly}\log$ terms in n, d, and the failure probability δ.

NUS and NTU joint work - Thank you

1. Errors accumulate with quantizing subroutines.
2. Finding kinks optimally.
3. Difficulties in choosing classical algorithms and then quantizing.
4. Links to compressed sensing - open problems.
5. Paper discusses fast and slow rates of convergence.

References

- Tibshirani RJ. The lasso problem and uniqueness. Electronic Journal of Statistics. 2013;7:1456-90.
- Kerenidis I, Prakash A. Quantum recommendation systems. arXiv preprint arXiv:1603.08675. 2016 Mar 29.
- V., Doriguello J, Lim D, Rebentrost P, Pun CS. Quantum Lasso (Current work) (in preparation).
- Chen Y, de Wolf R. Quantum algorithms and lower bounds for linear regression with norm constraints. arXiv preprint arXiv:2110.13086. 2021 Oct 25.