Data Science, Statistics, and Health

with a focus on Statistical Learning and Sparsity

Rob Tibshirani
Departments of Biomedical Data Science & Statistics
Stanford University

University of Padua, 2022

Outline

1. Some general thoughts about data science and health
2. General advice for data scientists
3. Some comments about supervised learning and **sparsity**
4. Example: large scale **GWAS** (Genome-wide association studies]
5. **The elephant in the room:** deep learning
6. If you can't beat'em... join em.: **LassoNet**

The last time I heard Frank Harrell speak

(Stanford 2019)

He was "grumpy" - critical of "big data" analyses & machine learning! But it led to the paper:

Cross-validation: what does it estimate and how well does it do it?

Stephen Bates, Trevor Hastie, Robert Tibshirani

Derives proper standard errors and confidence intervals for cross-validation error.

arXiv

There is a lot of excitement about Data Science and health

- Artificial intelligence, predictive analytics, precision medicine are all hot areas with huge potential
- A wealth of data is now available in every area of public health and medicine, from new machines and assays, smart phones, smart watches ....
- Already, there have been good successes in data science in pathology, radiology, and other diagnostic specialties.

For Statisticians: 15 minutes of fame

- 2009: " I keep saying the sexy job in the next ten years will be statisticians." Hal Varian, Chief Economist Google

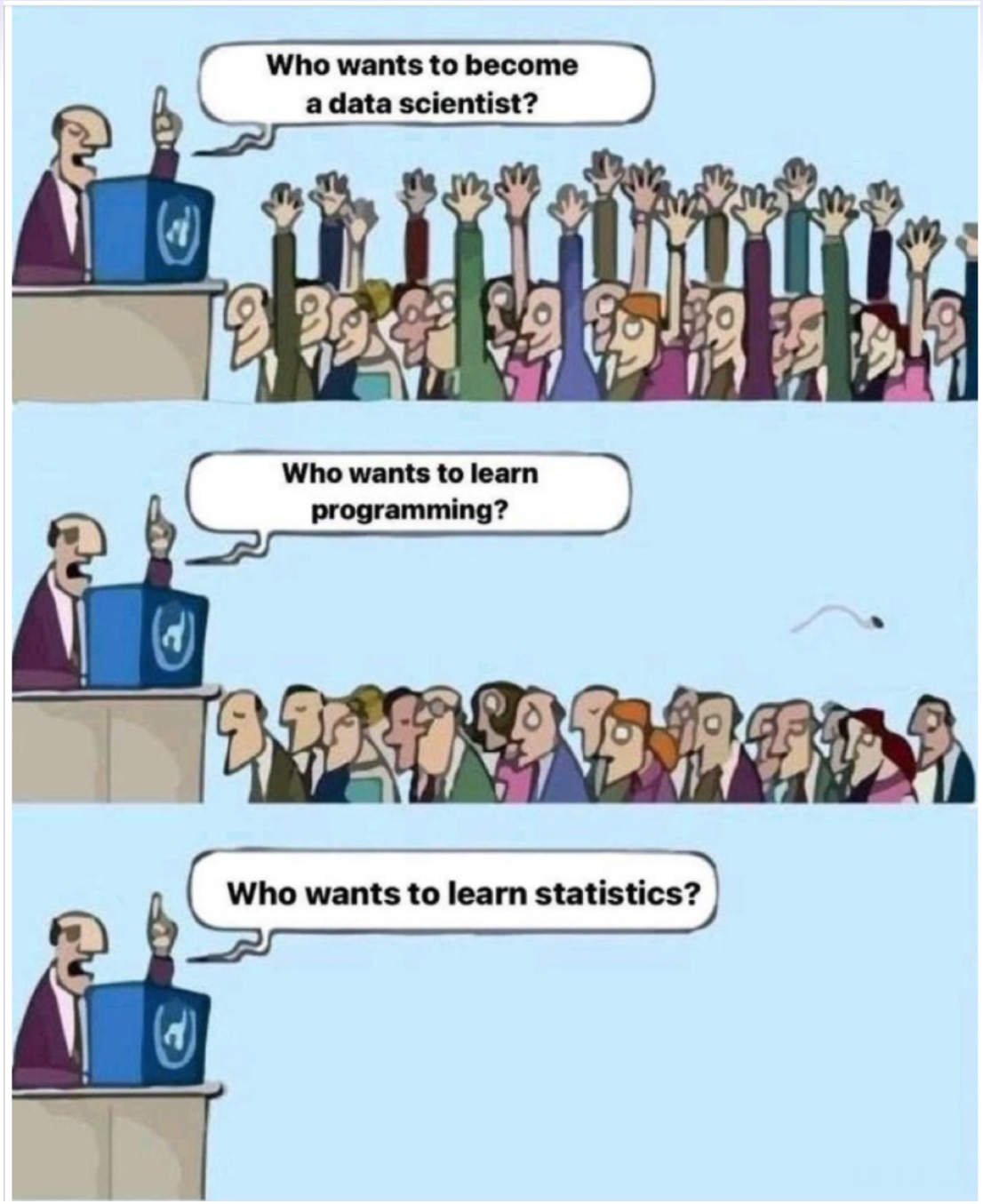
- 2012 "Data Scientist: The Sexiest Job of the 21st Century" Harvard Business Review
- 2021 (??; we need a new quote)

Some obstacles to this success

- **Data siloing** is a problem. In the US health care system researchers tend not to share data openly. Access to large, representative datasets is essential for this work.
- **The bar is higher in health.** There is more more at stake in the health area than in say a recommendation system for movies. Errors are much more costly
- The public may be **less tolerant** of data science/machine errors than human errors (analogous to self-driving cars).
- NEW for 2020-2021! Building trust in (data) science and communicating its results are, sadly, much harder than previously thought

High level comments about Data Science, Statistics and Machine learning

- Data Science and Statistics involve **much more** than simply running a **machine learning algorithm** on data
- For example:
  - What is the question of interest? How can I collect data or run a experiment to address this question?
  - What inferences can be drawn from the data?
  - What action should I take as a result of what I've learned?
  - Do I need to worry about bias, confounding, generalizability, concept drift ...?
- $\rightarrow$**Statistical concepts and training are essential**


Who wants to become a data scientist?
Who wants to learn programming?
Who wants to learn statistics?

Important points for data science applications in health

- Models should be kept as **simple as possible**, as they are easier to understand. *And we can more easily anticipate how/when they might break.*
- **Uncertainly quantification** is essential. We must build trust in our models.
- **Algorithmic fairness** is important
- Example of a key unsolved problem: in classification, with high dimensional features, how can we arrange for our classifier to sometimes "**abstain**"? (because it is extrapolating).
- Estimation of **heterogeneous treatment effects (HTE)**, eg for personalized medicine, is a very important area in need of research. A simple unsolved problem: how can we do internal cross-validation of a model for HTE?

General advice for Data Scientists

Details matter!

Q: *Have you tried method X?*
A: I tried that last year and it didn't work very well!

Q: *What do you mean? What exactly did you try? How did you measure performance?*
A: I can't remember.

General tips

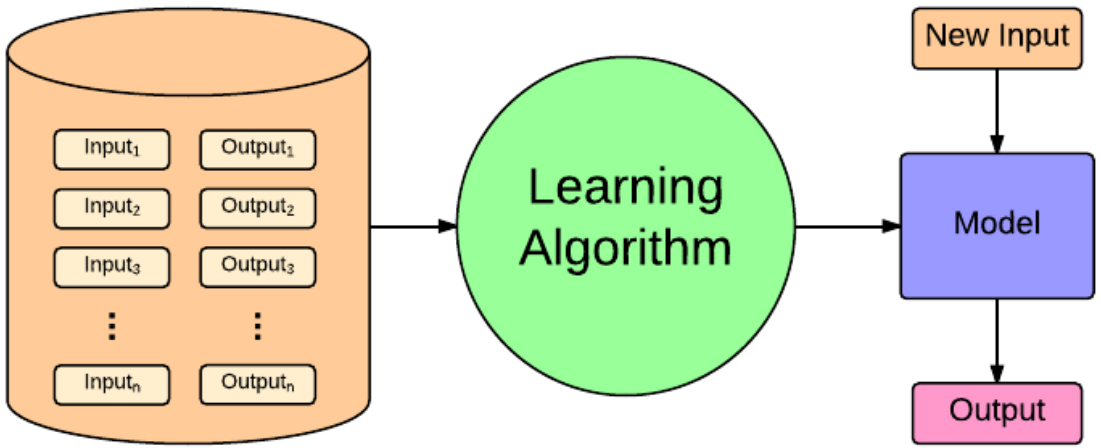
- Try a number of methods and use **internal validation** or **cross-validation** to tune and compare models (our "lab" is the computer- experiments are quick and free)
- Be **systematic** when you run methods and make comparisons: compare methods on the same training and test sets.
- Keep **carefully documented scripts** and data archives (where practical).
- Your work should be **reproducible** by **you and others**, even a few years from now!

In Praise of Simplicity

'Simplicity is the ultimate sophistication" — Leonardo Da Vinci

- Many times I have been asked to review a data analysis by a biology postdoc or a company employee. Almost every time, they are unnecessarily complicated. Multiple steps, each one poorly justified.
- Why? I think we all like to justify– internally and externally— our advanced degrees. And then there's the "hit everything with deep learning" problem
- Suggestion: Always try simple methods first. Move on to more complex methods, only if necessary

The Supervising Learning Paradigm



Training Data Fitting Prediction

Traditional statistics: domain experts work for 10 years to learn good features; they bring the statistician a small clean dataset

Today's approach: we start with a large dataset with many features, and use a machine learning algorithm to find the as good ones. A huge change.

Many problems involve supervised learning: building models from data for predicting an outcome using a collection of input features.

Big data vary in *shape*. These call for different approaches.

Wide Data

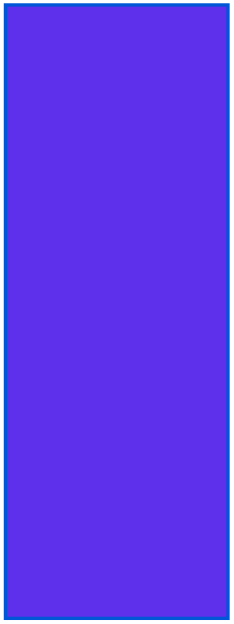
Thousands / Millions of Variables

Hundreds of Samples

Lasso & Elastic Net

We have too many variables; prone to overfitting.
Lasso fits linear models to the data that are *sparse* in the variables.
Does automatic variable selection.

Tall Data



Tens / Hundreds of Variables

Thousands / Tens of Thousands of Samples

Random Forests &
Gradient Boosting

Sometimes simple models (linear) don't suffice.
We have enough samples to fit nonlinear models with many interactions, and not too many variables.
A Random Forest is an automatic and powerful way to do this.

The Lasso

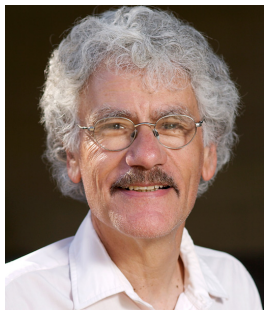
The **Lasso** is an estimator defined by the following optimization problem:

$$\underset{\beta_0,\beta}{\text{minimize}}\ \frac{1}{2}\sum_i (y_i - \beta_0 - \sum_j x_{ij}\beta_j)^2 \qquad \text{subject to} \quad \sum |\beta_j| \leq s$$

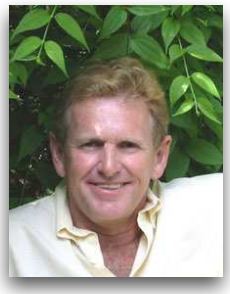
- Penalty $\implies$ sparsity (feature selection)
- Convex problem (good for computation and theory)

The glmnet package in R

- Our lab has written a open-source R language package called `glmnet` for fitting lasso models. Numerics in FORTRAN(!)
- Many clever computational tricks were used to achieve it's impressive speed.
- 3.5 million downloads as of July 2021
- lasso software also available in Python (e.g. scikit.learn)



Jerry Friedman



Trevor Hastie

Balasubramanian Narasimhan

Noah Simon

Ken Tay

Features of the current version (glmnet 4.1)

- Gaussian, binomial, multinomial, poisson and user-defined "family" objects
- grouped lasso for multi-response Gaussian family
- support for sparse matrices
- feature filtering within cross-validation
- full treatment of survival analysis: "start/stop", TD covariates, interval censoring, etc.

Beyond Glmnet

SNPnet: Lasso and elastic net for GWAS: *Estimation of Polygenic risk scores*

- with current software (eg **glmnet**), lasso and elastic net cannot be applied to data of size say 500K (patients) by 800K (SNPs)
- we have developed a new approach using the idea of strong screening rules (Tibs et al JRSSB 2012), that successfully carries out this computation in hours (exact, within machine precision)
- Joint work with PhD students **Junyang Qian**, Yosuke Tanigawa, Trevor Hastie and the Manny Rivas group at Stanford DBDS
- "A Fast and Flexible Algorithm for Solving the Lasso in Large-scale and Ultrahigh-dimensional Problems" , Qian et al bioRxiv 2019

Strong rules: general idea

- Lasso computes solutions over a path of tuning parameter values λ, from sparse to dense;
  $\lambda_1 = \lambda_{max} > \lambda_2, \ > \lambda_3, \ \cdots > \lambda_K$
- Having solved the problem for some λ_k, can make a very good guess as to which subset of predictors might be active at the next value λ_{k+1}
- We check for violations of the KKT conditions at each step (these are rare), so resulting strategy is not an approximation and delivers the set of exact solutions.

Strong Rules; more details

For lasso fit, with *active set* $\mathcal{A}$:

$$\left|\langle \mathbf{x}_j, \mathbf{y} - \mathbf{X}\hat{\beta}(\lambda)\rangle\right| \quad \begin{array}{ll} = \lambda & \forall j \in \mathcal{A} \\ \leq \lambda & \forall j \notin \mathcal{A} \end{array}$$

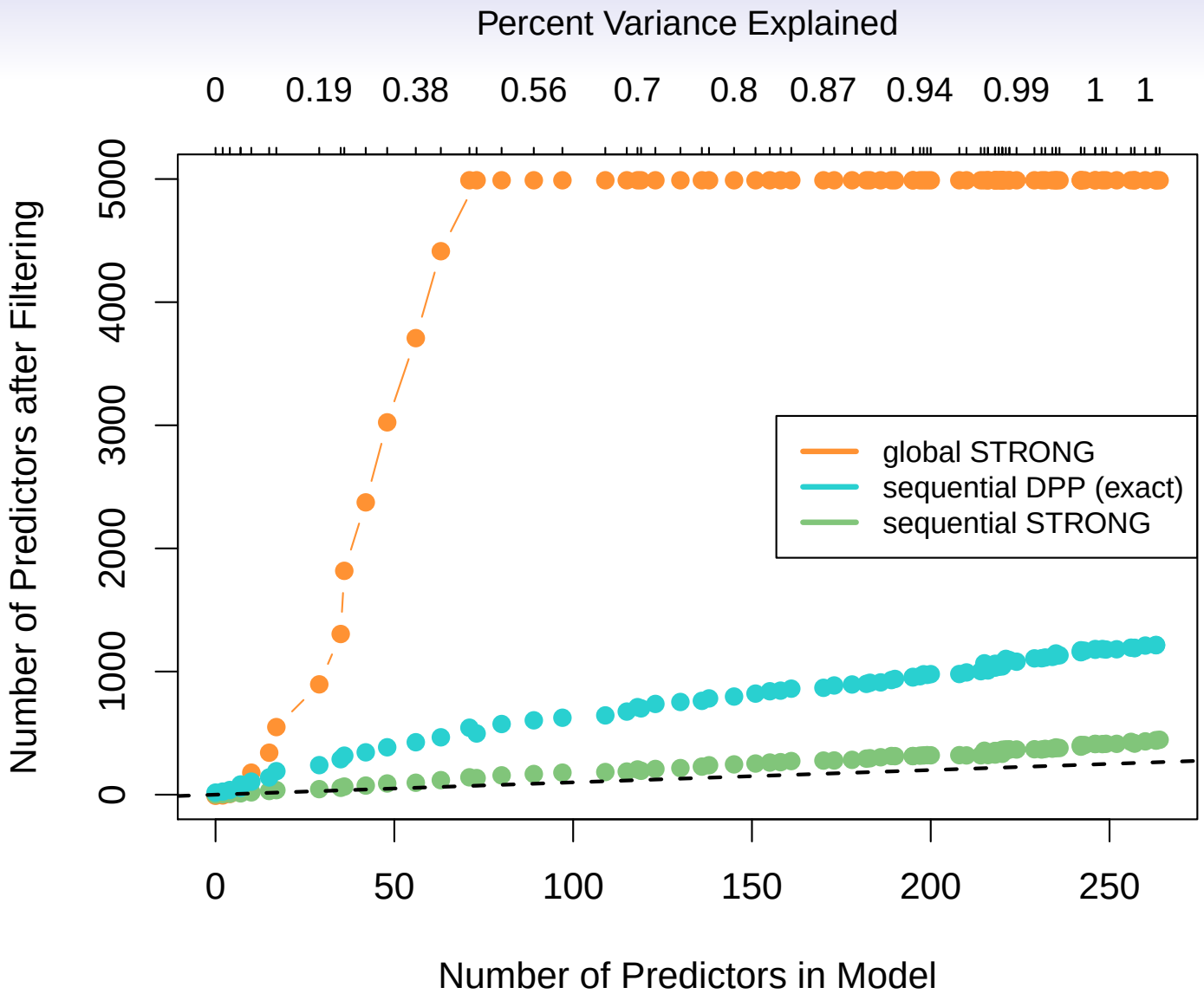
So variables *nearly in* $\mathcal{A}$ will have inner-products with the residuals *nearly* equal to λ.

Suppose fit at λ_ℓ is $\mathbf{X}\hat{\beta}(\lambda_\ell)$, and we want to compute the fit at $\lambda_{\ell+1} < \lambda_\ell$.
Strong rules gamble on the set

$$\mathcal{S}_{\ell+1} = \left\{ j : |\langle \mathbf{x}_j, \mathbf{y} - \mathbf{X}\hat{\beta}(\lambda_\ell)\rangle| > \lambda_{\ell+1} - (\lambda_\ell - \lambda_{\ell+1}) \right\}$$

This strategy is used by GLMNET: it screens at every λ step, and after convergence, checks if any violations. Mostly $\mathcal{A}_{\ell+1} \subseteq \mathcal{S}_{\ell+1}$.

* Tibshirani, Bien, Friedman, Hastie, Simon, Taylor, Ryan Tibshirani (JRSSB 2012)



Strong rules inspired by El Ghaoui, Viallon and Rabbani (2010)

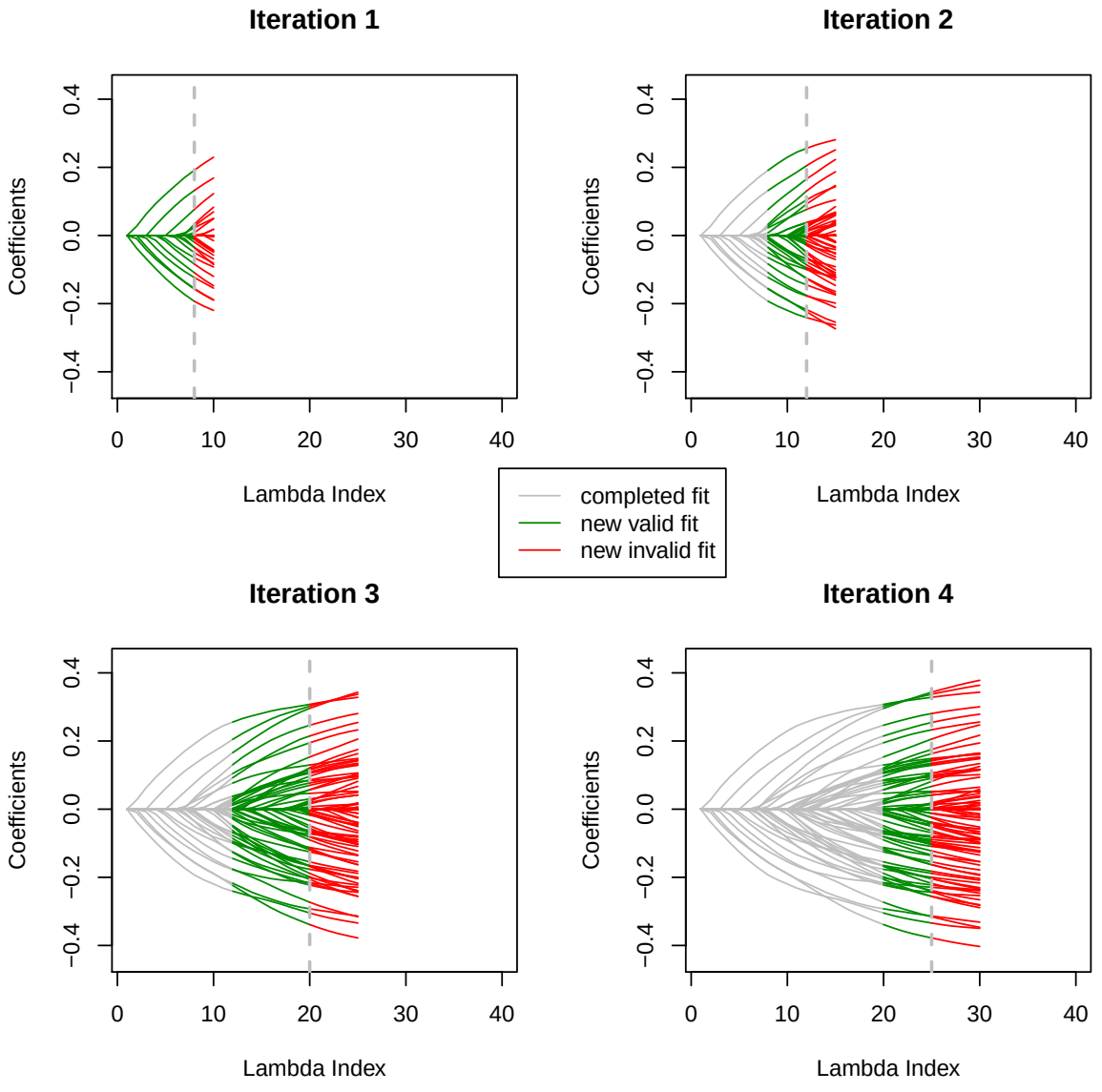
Sequential DPP due to Wang, Lin, Gong, Wonka and Ye (2013)

SNPnet

1. Initialize active variables A_0= empty set, $\lambda = \lambda_{max}$
2. for $k = 1, 2, \ldots$
   - Screening: At current λ_{max}, form the strong set $S_k = A_k \cup E_k$ where A_k is set of M variables with largest inner products with the residual and E_k are the ever active variables
   - Fitting: Solve the lasso using just the strong set of predictors S_k
   - Checking: find the smallest λ_{k+1} so that KKT conditions are satisfied (ie. no predictors have been omitted that should be present)

Typically $M = 1000$; Screening and Checking require the full dataset, and are done efficiently using memory-mapped I/O. (**bigmemory** R package)

"SNPnet" in pictures

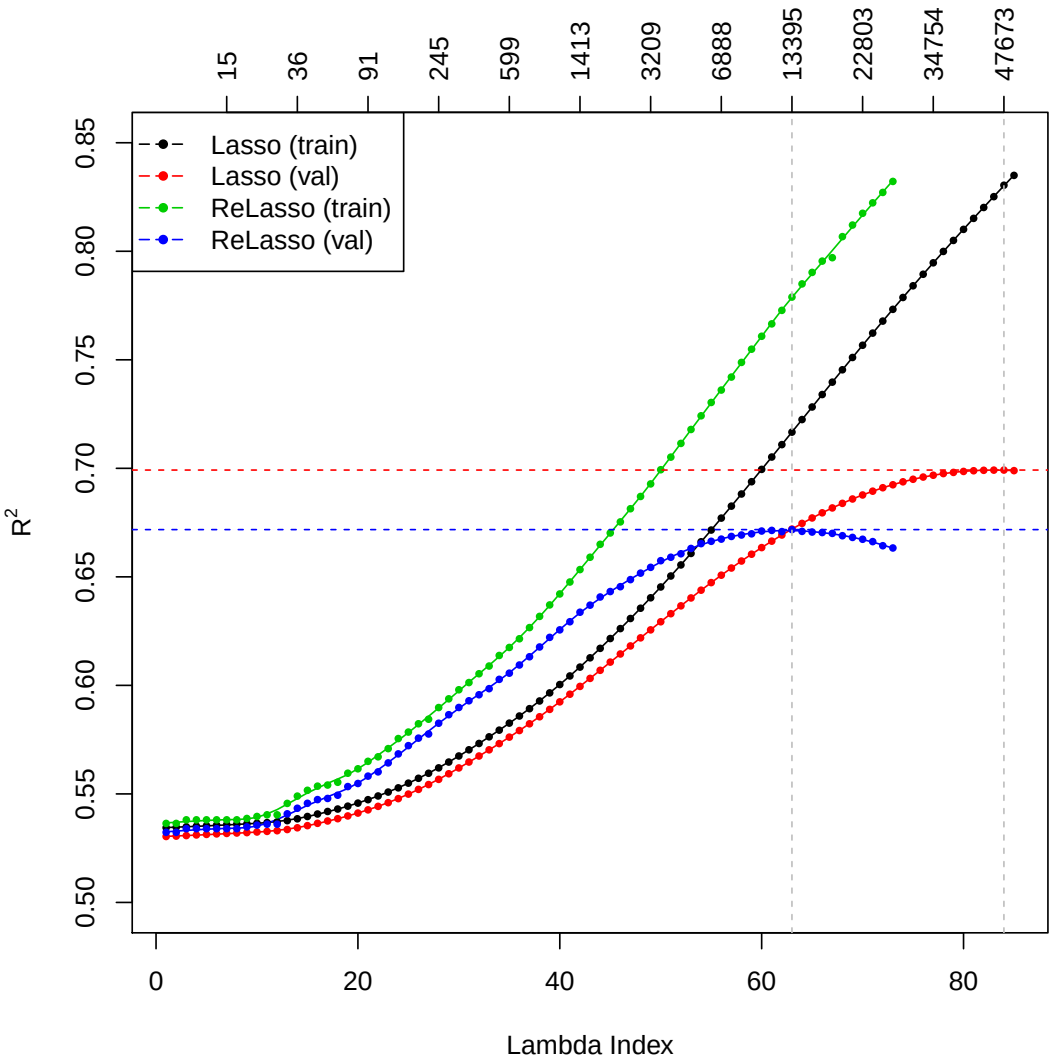


GWAS example

- Large cohort of 500K British adults from UK BioBank
- Each individual genotyped at 805K locations (AA, Aa, aa or NA)
- 100s of phenotypes measured on each subject
- We looked at white British subset of 337K, and illustrate with `height` phenotype
- Divided the data 60% training, 20% validation, 20% test.
- computation took a few hours (128GB memory and 16 cores)

Package SNPNET available on Github (link on Hastie's website, and to 2019 report)

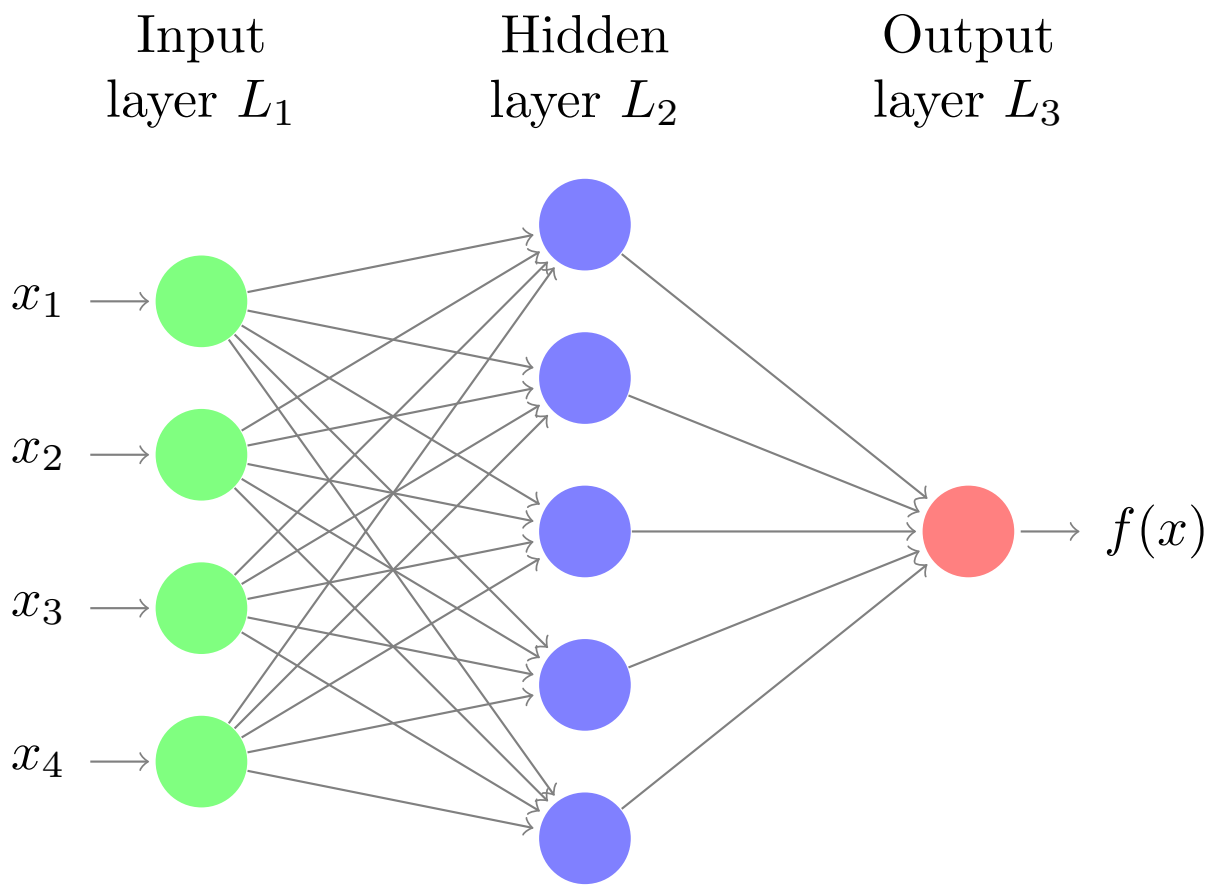
Lasso fit to `height`



The Elephant in the Room: DEEP LEARNING

Will it eat the lasso and other statistical models?

Deep Nets/Deep Learning



Neural network diagram with a single hidden layer. The hidden layer derives transformations of the inputs — nonlinear transformations of linear combinations — which are then used to model the output

What has changed since the invention of Neural nets?

- Much bigger training sets and faster computation (especially GPUs)
- Many clever ideas: convolution filters, stochastic gradient descent, input distortion ...
- Use of **multiple layers** (the "Deep" part): Tommy Poggio says that the universal approximation theorems for single layer NNs in the 1980s **set the field back 30 years**
- **Confession:** I was Geoff Hinton's colleague at Univ. of Toronto (1985-1998) and didn't appreciate the potential of Neural Networks!

What makes Deep Nets so powerful

(and challenging to analyze!)

It's not one "mathematical model" but a **customizable framework**– a set of **engineering tools** that can exploit the special aspects of the problem (weight-sharing, convolution, recurrent NNs ...)

Will Deep Nets eat the lasso and other statistical models?

Deep Nets are especially powerful when the features have some spatial or temporal organization (signals, images), and SNR is high

But they may not be a good approach when

- we have moderate #obs or wide data (#obs $<$ #features),
- SNR is low, or
- interpretability is important

It's difficult to find examples where Deep Nets beat lasso or boosting in low SNR settings, with "generic " features

LassoNet: "If you can't beat'em ....

Feature sparse neural networks

Lemhadri, Ruan, Abraham, Tibshirani, JMLR 2021

LassoNet in two minutes
Click for video

LassoNet in detail

- We assume the model

$$y_i = \beta_0 + \sum_j x_{ij}\beta_j + \sum_{k=1}^{K}[\alpha_k + \gamma_k \cdot f(\theta_k^T x_i)] + \epsilon_i \qquad (1)$$

  with $\epsilon \sim (0, \sigma^2)$. Here f is a monotone, nonlinear function such as a sigmoid or rectified linear unit, and each $\theta_k = (\theta_{1k}, \ldots \theta_{pk})$ is a p-vector.
- Our objective is to minimize

$$\begin{aligned} J(\beta, \Theta, \alpha, \gamma) \quad = \quad & \frac{1}{2}\sum_i (y_i - \hat{y}_i)^2 + \lambda \sum_j |\beta_j| + \bar{\lambda} \sum_{jk} |\theta_{jk}| \\ & \textbf{subject to } |\theta_{jk}| \leq M \cdot |\beta_j| \ \ \forall j, k. \quad \leftarrow \textbf{secret sauce} \end{aligned}$$

In a nutshell

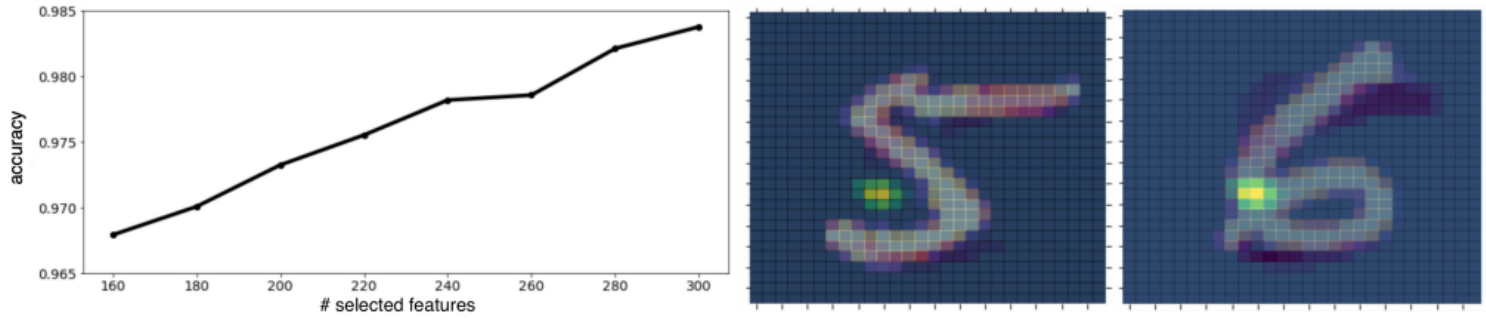
"*Features can participate in the hidden layer only if they have non-zero main effects*"

- A neural network is a complex model, but is made simpler if the model is a function of only a subset of the features
- Eg a protein-based blood test for a disease: if only 10 proteins out of 1000 need to be measured, then the complicated neural net mapping of these 10 proteins is still acceptable to scientists

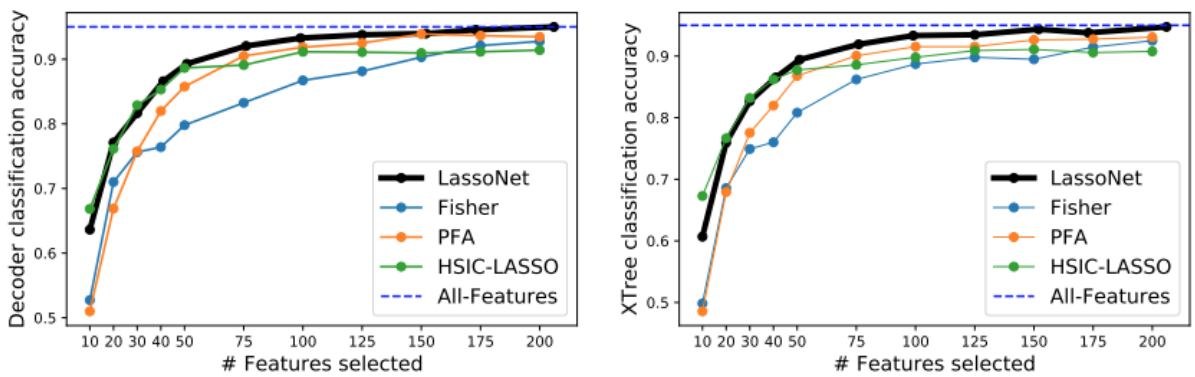
Computational strategy

- We seek the solution over a path of λ values, using current solutions as warm starts. We use a projected proximal gradient procedure at each λ
- Unlike Lasso, objective is **non-convex**. Makes optimization much harder. Eg the starting values for (β, θ) matter!
- In lasso (our glmnet package), we compute a path of solutions from sparse (λ large) to dense (λ small)
- This worked badly for LassoNet; we tried many tricks; finally we tried going from Dense to Sparse. **Worked!**

Examples



Differentiating 5s from 6s



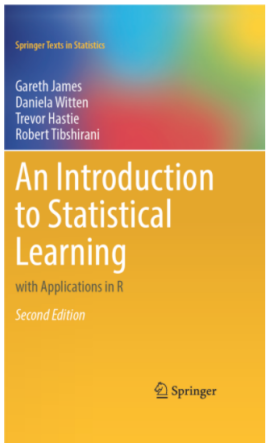
ISOLET (26 classes); LassoNet vs other feature selection methods; two different classifiers

Ongoing work on LassoNet

Now working with **Benjamin Haibe-Kains**, of the Princess Margaret Cancer Centre, his students **Michal Kazmierski**, and **Petr Smirnov** and former stanford student **Feng Ruan**:

Applying LassoNet to head & neck cancer images and clinical features

Just published: 2nd edition!



James, Witten, Hastie, Tibshirani.

New chapters on Deep Learning, Survival analysis, Multiple testing

Book pdf is available online for free at
`https://www.statlearning.com`

Free online course at
`https://www.edx.org/course/statistical-learning`

Includes recent interviews with David Cox, Geoff Hinton and Yoav Benjamini