LassoNet: A Neural Network with Feature Sparsity

Ismael Lemhadri
Joint work with Feng Ruan and Rob Tibshirani

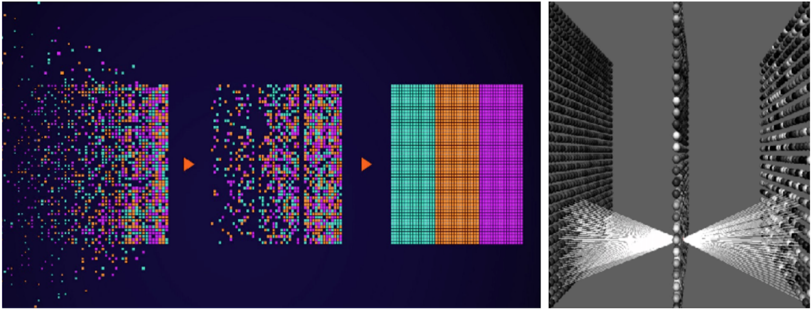
Stanford University

July 9, 2020

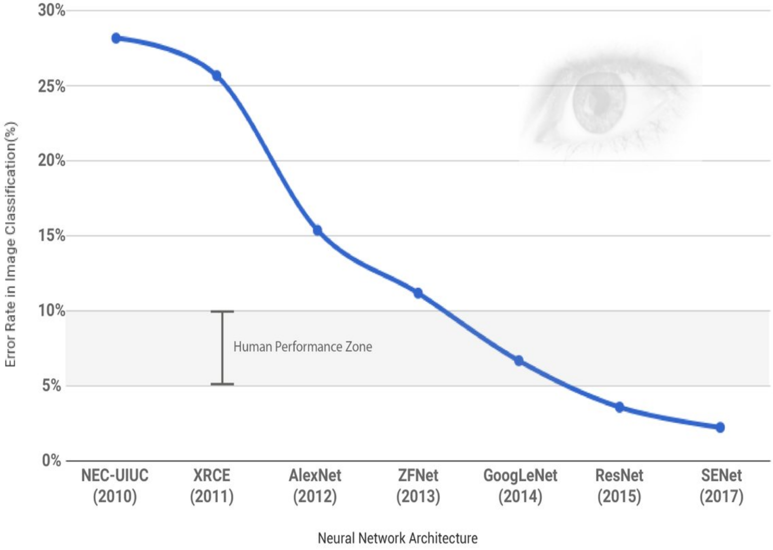
Talk Materials at: `https://tinyurl.com/lassonet`

Modern Machine Learning

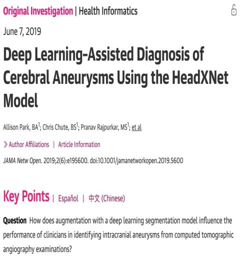
- Large, complex models
- Massive amounts of data



The ILSVRC Competition



deep learning: applications



Original Investigation | Health Informatics

June 7, 2019

Deep Learning-Assisted Diagnosis of Cerebral Aneurysms Using the HeadXNet Model

Allison Park, BA[1]; Chris Chute, BS[1]; Pranav Rajpurkar, MS[1]; et al

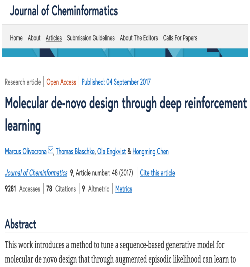
》Author Affiliations | Article Information

JAMA Netw Open. 2019;2(6):e195600. doi:10.1001/jamanetworkopen.2019.5600

Key Points | Español | 中文 (Chinese)

Question How does augmentation with a deep learning segmentation model influence the performance of clinicians in identifying intracranial aneurysms from computed tomographic angiography examinations?

healthcare



Journal of Cheminformatics

Home About Articles Submission Guidelines About The Editors Calls For Papers

Research article | Open Access | Published: 04 September 2017

Molecular de-novo design through deep reinforcement learning

Marcus Olivecrona, Thomas Blaschke, Ola Engkvist & Hongming Chen

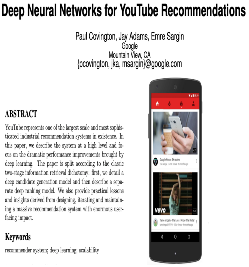
Journal of Cheminformatics 9, Article number: 48 (2017) | Cite this article

9281 Accesses | 78 Citations | 9 Altmetric | Metrics

Abstract

This work introduces a method to tune a sequence-based generative model for molecular de novo design that through augmented episodic likelihood can learn to

drug discovery



Deep Neural Networks for YouTube Recommendations

Paul Covington, Jay Adams, Emre Sargin
Google
Mountain View, CA
{pcovington, jka, msargin}@google.com

ABSTRACT

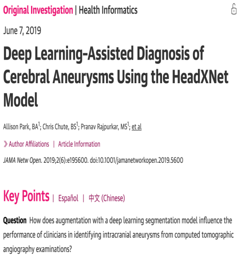
YouTube represents one of the largest scale and most sophisticated industrial recommendation systems in existence. In this paper, we describe the system at a high level and focus on the dramatic performance improvements brought by deep learning. The paper is split according to the classic two-stage information retrieval dichotomy: first, we detail a deep candidate generation model and then describe a separate deep ranking model. We also provide practical lessons and insights derived from designing, iterating and maintaining a massive recommendation system with enormous user-facing impact.

Keywords

recommender system; deep learning; scalability

recommender systems

deep learning: applications


Original Investigation | Health Informatics

June 7, 2019

Deep Learning-Assisted Diagnosis of Cerebral Aneurysms Using the HeadXNet Model

Allison Park, BA[1]; Chris Chute, BS[1]; Pranav Rajpurkar, MS[1]; et al

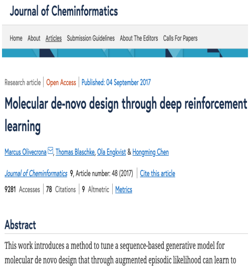
› Author Affiliations | Article Information

JAMA Netw Open. 2019;2(6):e195600. doi:10.1001/jamanetworkopen.2019.5600

Key Points | Español | 中文 (Chinese)

Question How does augmentation with a deep learning segmentation model influence the performance of clinicians in identifying intracranial aneurysms from computed tomographic angiography examinations?

healthcare


Journal of Cheminformatics

Research article | Open Access | Published: 04 September 2017

Molecular de-novo design through deep reinforcement learning

Marcus Olivecrona, Thomas Blaschke, Ola Engkvist & Hongming Chen

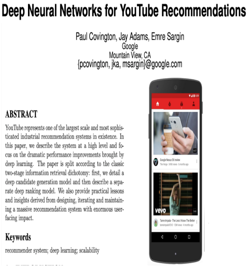
Journal of Cheminformatics 9, Article number: 48 (2017) | Cite this article

9281 Accesses | 78 Citations | 9 Altmetric | Metrics

Abstract

This work introduces a method to tune a sequence-based generative model for molecular de novo design that through augmented episodic likelihood can learn to

drug discovery


Deep Neural Networks for YouTube Recommendations

Paul Covington, Jay Adams, Emre Sargin
Google
Mountain View, CA
{pcovington, jka, msargin}@google.com

ABSTRACT

YouTube represents one of the largest scale and most sophisticated industrial recommendation systems in existence. In this paper, we describe the system at a high level and focus on the dramatic performance improvements brought by deep learning. The paper is split according to the classic two-stage information retrieval dichotomy: first, we detail a deep candidate generation model and then describe a separate deep ranking model. We also provide practical lessons and insights derived from designing, iterating and maintaining a massive recommendation system with enormous user-facing impact.

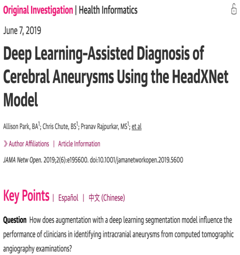
Keywords

recommender system; deep learning; scalability

recommender systems

Also: gene sequencing, advertisement, speech recognition ...

deep learning: applications



Original Investigation | Health Informatics

June 7, 2019

Deep Learning-Assisted Diagnosis of Cerebral Aneurysms Using the HeadXNet Model

Allison Park, BA[1]; Chris Chute, BS[1]; Pranav Rajpurkar, MS[1]; et al

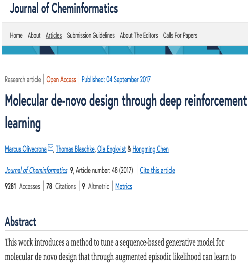
》Author Affiliations | Article Information

JAMA Netw Open. 2019;2(6):e195600. doi:10.1001/jamanetworkopen.2019.5600

Key Points | Español | 中文 (Chinese)

Question How does augmentation with a deep learning segmentation model influence the performance of clinicians in identifying intracranial aneurysms from computed tomographic angiography examinations?

healthcare



Journal of Cheminformatics

Home About Articles Submission Guidelines About The Editors Calls For Papers

Research article | Open Access | Published: 04 September 2017

Molecular de-novo design through deep reinforcement learning

Marcus Olivecrona, Thomas Blaschke, Ola Engkvist & Hongming Chen

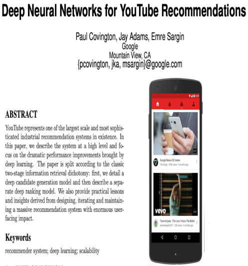
Journal of Cheminformatics 9, Article number: 48 (2017) | Cite this article

9281 Accesses | 78 Citations | 9 Altmetric | Metrics

Abstract

This work introduces a method to tune a sequence-based generative model for molecular de novo design that through augmented episodic likelihood can learn to

drug discovery



Deep Neural Networks for YouTube Recommendations

Paul Covington, Jay Adams, Emre Sargin
Google
Mountain View, CA
{pcovington, jka, msargin}@google.com

ABSTRACT

YouTube represents one of the largest scale and most sophisticated industrial recommendation systems in existence. In this paper, we describe the system at a high level and focus on the dramatic performance improvements brought by deep learning. The paper is split according to the classic two-stage information retrieval dichotomy: first, we detail a deep candidate generation model and then describe a separate deep ranking model. We also provide practical lessons and insights derived from designing, iterating and maintaining a massive recommendation system with enormous user-facing impact.

Keywords

recommender system; deep learning; scalability

recommender systems

Also: gene sequencing, advertisement, speech recognition ...

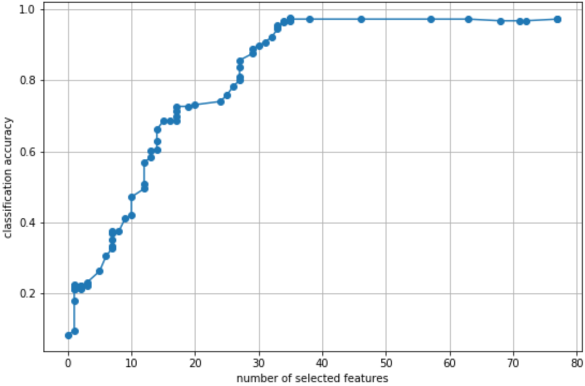
Deep learning pervades data-rich problems

Benefits of feature selection

- reduces overfitting
- improves accuracy
- helps overcome the curse of dimensionality
- allows shorter training time
- aids with interpretability

Mice Protein Data

Find proteins that are discriminant between healthy and trisomic mice. 1080 measurements, 77 proteins.[Higuera et al., 2015]



Best six proteins: AKT, NR2B,TIAM1,nNOS,RRP1,GluR3

Prior art

- Filter and wrapper methods
- Embedded methods

Prior art

- Filter and wrapper methods
  - Individual scores [Fisher score, Laplacian Score, Trace Ratio]
  - Kernel based methods
  - Mutual information based methods [HSIC-Lasso (Yamada et al., 2014), Conditional covariance minimization (Jordan et al., 2018)]
- Embedded methods
  - L1-regularization [Lasso (Tibshirani, 1996) and variants]

Desiderata

- Capture *arbitrary* nonlinearity [nonparametric approach]
- *Achieve* *adaptive* feature selection

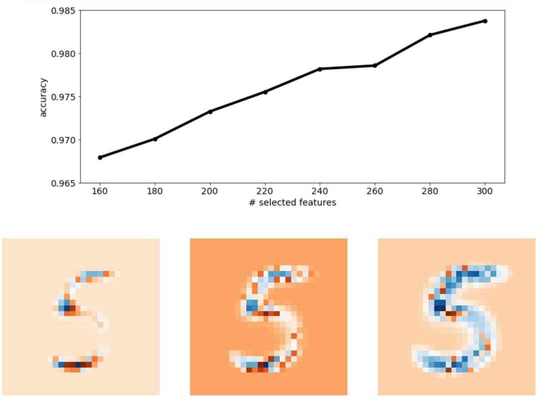
Desiderata

- Capture *arbitrary* nonlinearity [nonparametric approach]
- *Achieve adaptive* feature selection

Today's proposal:

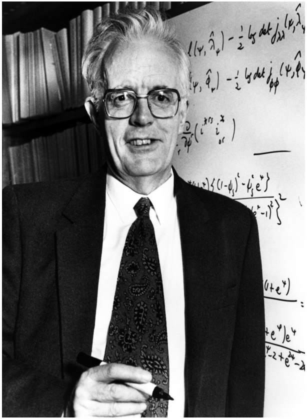
- An embedded method
- Optimizes over a large function class
- Obeys a natural hierarchy principle

Appetizer: results on MNIST



Demonstrating LassoNet on MNIST. Simultaneously selecting informative pixels and classifiyng digit 5 vs. digit 6. **Top:** The classification accuracy by number of selected features. **Bottom:** A sample from the model with 160, 220 and 300 active features out of the 784.

The hierarchy principle

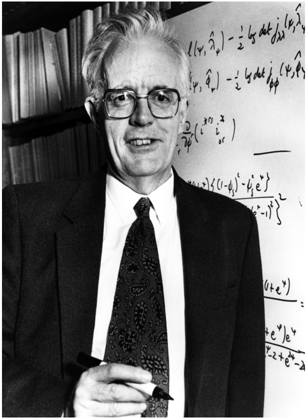


David Cox, 1980

Photo: General Motors Cancer Research Foundation

"Large component **main effects** are more likely to lead to appreciable interactions than small components. Also, the **interactions** corresponding to larger main effects may be in some sense of more practical importance."

The hierarchy principle



David Cox, 1980

Photo: General Motors Cancer Research Foundation

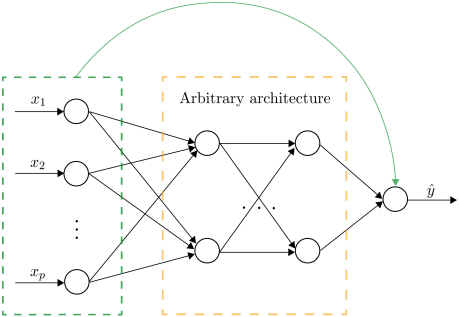
"Large component **main effects** are more likely to lead to appreciable interactions than small components. Also, the **interactions** corresponding to larger main effects may be in some sense of more practical importance."

More recently: Lasso for hierarchical interactions (Bien et al., 2013), reluctant interaction modelling (R.J. Tibshirani, 2019)

Our approach

- An embedded method
- Large function class: residual feedforward neural networks

$$\mathcal{F} = \left\{ f : f(\mathbf{x}) = \theta^T \mathbf{x} + f_W(\mathbf{x}) \right\}$$



LassoNet architecture

LassoNet

- **Objective function:**

$$\underset{\theta, W}{\text{minimize}}\; L(\theta, W) + \lambda \|\theta\|_1$$
$$\text{subject to } \; \|W^{(0)}\|_{j_\infty} \leq M|\theta_j|, \; j = 1, \dots, d.$$

where $W^{(0)}$ denotes the network's input layer.

LassoNet

- **Objective function:**

$$\begin{aligned} &\underset{\theta, W}{\text{minimize}} \ L(\theta, W) + \lambda \|\theta\|_1 \\ &\text{subject to} \ \|W^{(0)}\|_{j_\infty} \leq M|\theta_j|, \ j = 1, \dots, d. \end{aligned}$$

where $W^{(0)}$ denotes the network's input layer.

In particular, $W_j = 0$ as soon as $\theta_j = 0$.

LassoNet

- **Objective function:**

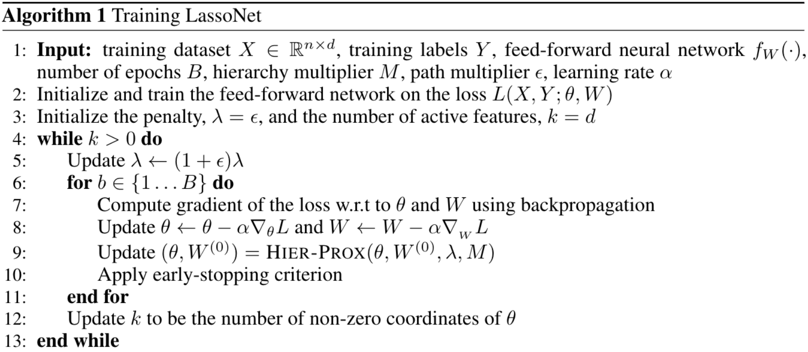
$$\underset{\theta, W}{\text{minimize}}\ L(\theta, W) + \lambda \|\theta\|_1$$
$$\text{subject to}\ \|W^{(0)}\|_{j_\infty} \leq M|\theta_j|,\ j = 1, \ldots, d.$$

where $W^{(0)}$ denotes the network's input layer.

In particular, $W_j = 0$ as soon as $\theta_j = 0$.

- **Hyper-parameters:**
  - ℓ_1 penalty, λ. Higher values of λ encourage sparser models
  - Hierarchy parameter, M. Controls the relative strength of the linear and nonlinear components.

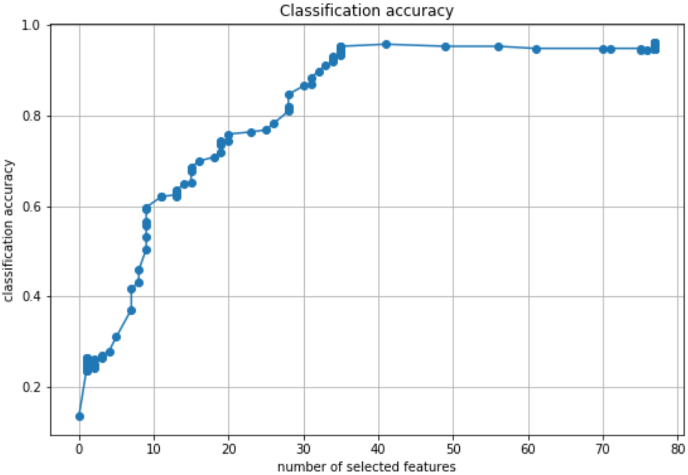
LassoNet Training Loop



Algorithm 1 Training LassoNet

```
Input: training dataset $X \in \mathbb{R}^{n \times d}$, training labels $Y$, feed-forward neural network $f_W(\cdot)$, number of epochs $B$, hierarchy multiplier $M$, path multiplier $\epsilon$, learning rate $\alpha$
Initialize and train the feed-forward network on the loss $L(X, Y; \theta, W)$
Initialize the penalty, $\lambda = \epsilon$, and the number of active features, $k = d$
while $k > 0$ do
    Update $\lambda \leftarrow (1 + \epsilon)\lambda$
    for $b \in \{1 \ldots B\}$ do
        Compute gradient of the loss w.r.t to $\theta$ and $W$ using backpropagation
        Update $\theta \leftarrow \theta - \alpha \nabla_\theta L$ and $W \leftarrow W - \alpha \nabla_W L$
        Update $(\theta, W^{(0)}) = \textsc{Hier-Prox}(\theta, W^{(0)}, \lambda, M)$
        Apply early-stopping criterion
    end for
    Update $k$ to be the number of non-zero coordinates of $\theta$
end while
```

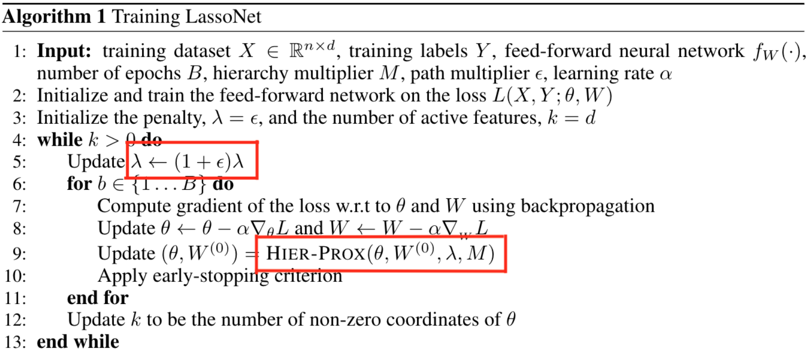
Feature Selection Path



Classification accuracies for LassoNet on a hold-out test-set.

Results on the MICE protein dataset where $n = 864$, $d = 77$.

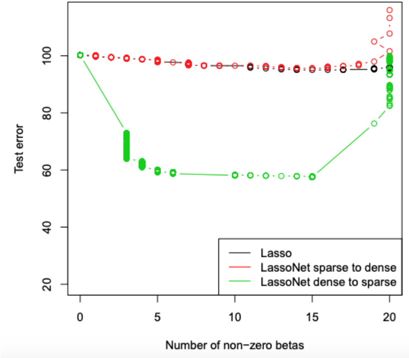
LassoNet Training Loop



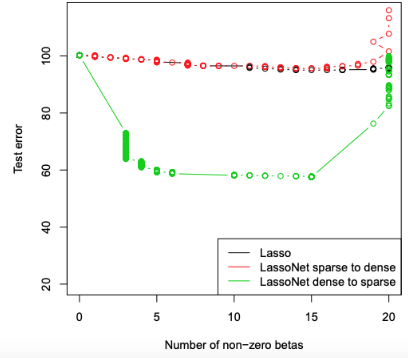
Algorithm 1 Training LassoNet

```
Input: training dataset $X \in \mathbb{R}^{n \times d}$, training labels $Y$, feed-forward neural network $f_W(\cdot)$, number of epochs $B$, hierarchy multiplier $M$, path multiplier $\epsilon$, learning rate $\alpha$
Initialize and train the feed-forward network on the loss $L(X, Y; \theta, W)$
Initialize the penalty, $\lambda = \epsilon$, and the number of active features, $k = d$
while $k > 0$ do
    Update $\boxed{\lambda \leftarrow (1 + \epsilon)\lambda}$
    for $b \in \{1 \dots B\}$ do
        Compute gradient of the loss w.r.t to $\theta$ and $W$ using backpropagation
        Update $\theta \leftarrow \theta - \alpha \nabla_\theta L$ and $W \leftarrow W - \alpha \nabla_W L$
        Update $(\theta, W^{(0)}) = \boxed{\textsc{Hier-Prox}(\theta, W^{(0)}, \lambda, M)}$
        Apply early-stopping criterion
    end for
    Update $k$ to be the number of non-zero coordinates of $\theta$
end while
```

The power of warm starts

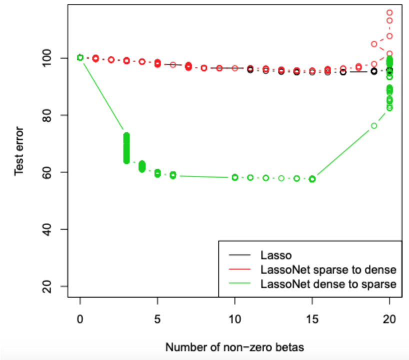


The power of warm starts



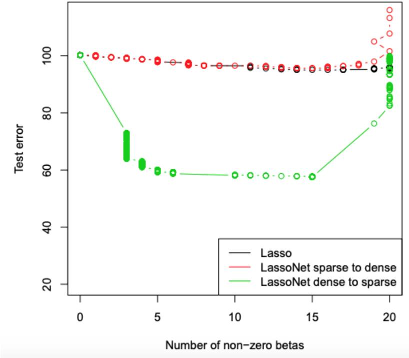
- The sparse to dense optimization along the path efficiently explores the nonconvex landscape.

The power of warm starts



- The sparse to dense optimization along the path efficiently explores the nonconvex landscape.
- Training combines warm starts and early stopping

The power of warm starts



- The sparse to dense optimization along the path efficiently explores the nonconvex landscape.
- Training combines warm starts and early stopping
- The bulk of the computational cost goes to training the dense model.
- This is effectively pruning

The HIER-PROX algorithm

- The hierarchy constraint is separable over the features.
- Objective can be optimized by constrained proximal GD

The HIER-PROX algorithm

- The hierarchy constraint is separable over the features.
- Objective can be optimized by constrained proximal GD
- At its core, LassoNet solves d problems of the form

$$\text{minimize}_{\beta \in \mathbb{R}, W \in \mathbb{R}^K} \ \frac{1}{2}(v - \beta)^2 + \frac{1}{2}\|u - W\|^2 + \lambda\|\beta\|_1$$

$$\text{subject to } \|W\|_\infty \leq M \cdot |\beta|$$

- HIER-PROX: an efficient hierarchical proximal operator

The Hier-Prox operator

- At its core, LassoNet solves d problems of the form

$$\text{minimize}_{\beta\in\mathbb{R},W\in\mathbb{R}^K}\ \frac{1}{2}(v-\beta)^2+\frac{1}{2}\|u-W\|^2+\lambda\|\beta\|_1$$

$$\text{subject to } \|W\|_\infty \leq M\cdot|\beta|$$

- The Hier-Prox operator provides the **global** solution of this **nonconvex** minimization problem

The Hier-Prox operator

- At its core, LassoNet solves d problems of the form

$$\text{minimize}_{\beta \in \mathbb{R}, W \in \mathbb{R}^K} \frac{1}{2}(v - \beta)^2 + \frac{1}{2}\|u - W\|^2 + \lambda\|\beta\|_1$$

$$\text{subject to } \|W\|_\infty \leq M \cdot |\beta|$$

- The Hier-Prox operator provides the **global** solution of this **nonconvex** minimization problem
- Integrates seamlessly with deep learning frameworks PyTorch

The HIER-PROX operator

- At its core, LassoNet solves d problems of the form

$$\text{minimize}_{\beta \in \mathbb{R}, W \in \mathbb{R}^K} \frac{1}{2}(v - \beta)^2 + \frac{1}{2}\|u - W\|^2 + \lambda\|\beta\|_1$$

$$\text{subject to } \|W\|_\infty \leq M \cdot |\beta|$$

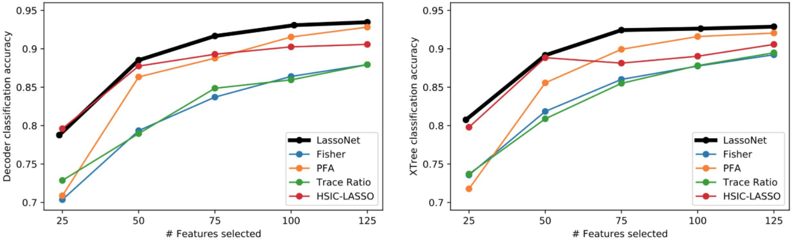
- The HIER-PROX operator provides the **global** solution of this **nonconvex** minimization problem
- Integrates seamlessly with deep learning frameworks PyTorch
- The algorithm has complexity $O(dK \cdot \log(dK))$, where d is the number of features and K the size of the input layer
- Negligible overhead compared to gradient computations

Experimental evaluation

- Most other feature selection methods are not *embedded*
- Plug the selected features into external downstream learners:
    - A feedforward neural network
    - A tree-based classifier
- Systematic evaluation on 6 datasets

Results on the ISOLET dataset

- Letter speech data
- Benchmark data set for feature selection
- $n = 7797, d = 617$

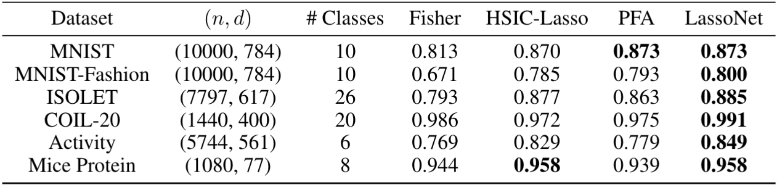


Classification accuracies for feature selection methods

Left: using a one-hidden-layer feedforward neural network. **Right:** using an extremely randomized tree classifier.

Systematic evaluation

Compare the classification accuracies for a fixed number of features, $k = 50$:



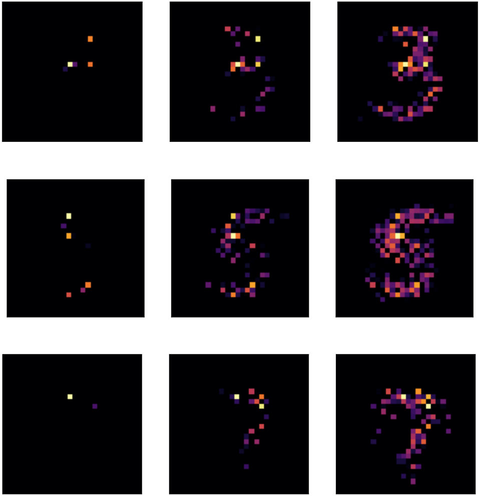
Dataset	(n, d)	# Classes	Fisher	HSIC-Lasso	PFA	LassoNet
MNIST	(10000, 784)	10	0.813	0.870	**0.873**	**0.873**
MNIST-Fashion	(10000, 784)	10	0.671	0.785	0.793	**0.800**
ISOLET	(7797, 617)	26	0.793	0.877	0.863	**0.885**
COIL-20	(1440, 400)	20	0.986	0.972	0.975	**0.991**
Activity	(5744, 561)	6	0.769	0.829	0.779	**0.849**
Mice Protein	(1080, 77)	8	0.944	**0.958**	0.939	**0.958**

Classification accuracies on a hold-out test set, using a one-hidden-layer feedforward neural network.

Summary

Extensions and applications

- Unsupervised learning
  - Reconstruction loss as the objective
  - Related work: Concrete auto-encoder (Abid et al., *ICML* 2019)



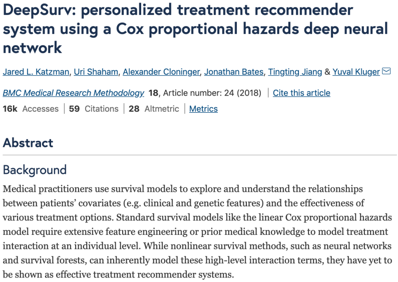
Reconstructing single digit classes of MNIST

Extensions and applications

- Unsupervised Learning
  - Reconstruction loss as the objective
  - Related work: Concrete auto-encoder (Abid et al., *ICML* 2019)
- Sparse PCA Net
  - enforce an interpretable bottleneck layer

Extensions and applications

- Unsupervised Learning
  - Reconstruction loss as the objective
  - Related work: Concrete auto-encoder (Abid et al., *ICML* 2019)
- Sparse PCA Net
  - enforce an interpretable bottleneck layer
- Cox Proportional Hazards Model



DeepSurv: personalized treatment recommender system using a Cox proportional hazards deep neural network

Jared L. Katzman, Uri Shaham, Alexander Cloninger, Jonathan Bates, Tingting Jiang & Yuval Kluger

BMC Medical Research Methodology **18**, Article number: 24 (2018) | Cite this article

16k Accesses | **59** Citations | **28** Altmetric | Metrics

Abstract

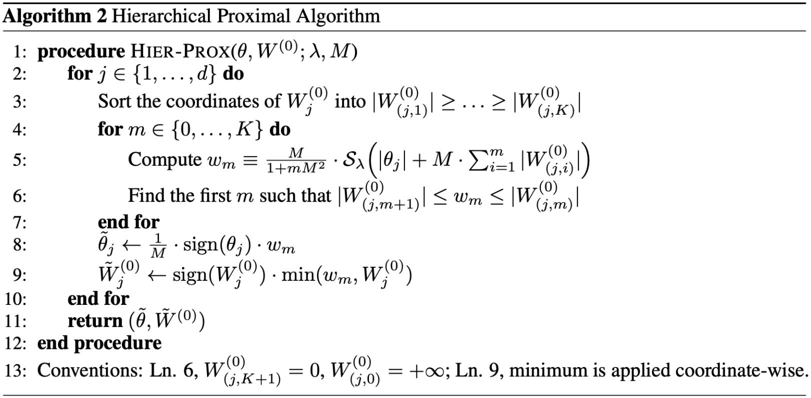
Background

Medical practitioners use survival models to explore and understand the relationships between patients' covariates (e.g. clinical and genetic features) and the effectiveness of various treatment options. Standard survival models like the linear Cox proportional hazards model require extensive feature engineering or prior medical knowledge to model treatment interaction at an individual level. While nonlinear survival methods, such as neural networks and survival forests, can inherently model these high-level interaction terms, they have yet to be shown as effective treatment recommender systems.

Resources

- Talk Materials at: `https://tinyurl.com/lassonet`
- Code at: `https://github.com/ilemhadri/lassonet`
- Thanks:
    - Rob Tibshirani
    - Feng Ruan
    - PyTorch help: Louis Abraham
- **Thank you. Be well!**

The HIER-PROX algorithm

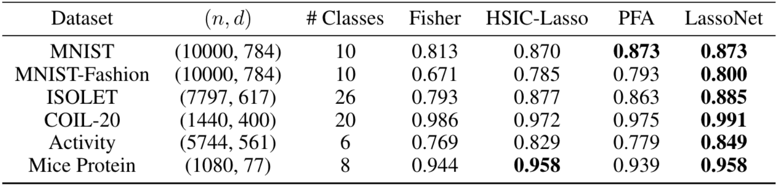


Algorithm 2 Hierarchical Proximal Algorithm

```
1:  procedure HIER-PROX($\theta, W^{(0)}; \lambda, M$)
2:      for $j \in \{1, \ldots, d\}$ do
3:          Sort the coordinates of $W_j^{(0)}$ into $|W_{(j,1)}^{(0)}| \geq \ldots \geq |W_{(j,K)}^{(0)}|$
4:          for $m \in \{0, \ldots, K\}$ do
5:              Compute $w_m \equiv \frac{M}{1+mM^2} \cdot \mathcal{S}_\lambda\left(|\theta_j| + M \cdot \sum_{i=1}^{m} |W_{(j,i)}^{(0)}|\right)$
6:              Find the first $m$ such that $|W_{(j,m+1)}^{(0)}| \leq w_m \leq |W_{(j,m)}^{(0)}|$
7:          end for
8:          $\tilde{\theta}_j \leftarrow \frac{1}{M} \cdot \mathrm{sign}(\theta_j) \cdot w_m$
9:          $\tilde{W}_j^{(0)} \leftarrow \mathrm{sign}(W_j^{(0)}) \cdot \min(w_m, W_j^{(0)})$
10:     end for
11:     return $(\tilde{\theta}, \tilde{W}^{(0)})$
12: end procedure
13: Conventions: Ln. 6, $W_{(j,K+1)}^{(0)} = 0$, $W_{(j,0)}^{(0)} = +\infty$; Ln. 9, minimum is applied coordinate-wise.
```

Systematic evaluation

Compare the classification accuracies for a fixed number of features, $k = 50$:



Dataset	(n, d)	# Classes	Fisher	HSIC-Lasso	PFA	LassoNet
MNIST	(10000, 784)	10	0.813	0.870	**0.873**	**0.873**
MNIST-Fashion	(10000, 784)	10	0.671	0.785	0.793	**0.800**
ISOLET	(7797, 617)	26	0.793	0.877	0.863	**0.885**
COIL-20	(1440, 400)	20	0.986	0.972	0.975	**0.991**
Activity	(5744, 561)	6	0.769	0.829	0.779	**0.849**
Mice Protein	(1080, 77)	8	0.944	**0.958**	0.939	**0.958**

Classification accuracies on a hold-out test set, using Extremely Randomized Tree Classifiers (a variant of random forests).