8th Curie Course on Computational Systems Biology of Cancer

Machine learning for genomics

Chloé-Agathe Azencott

Centre for Computational Biology (CBIO)
Mines Paris – PSL, Institut Curie & INSERM U1331
PR[AI]RIE-PSAI

September 22, 2025

http://cazencott.info chloe-agathe.azencott@minesparis.psl.eu @cazencott@lipn.info

MINES PARIS | PSL

PR[AI]RIE Paris School

institut Curie

Inserm

Machine learning

- Learn/build/define a **statistical model** using **data**
- **Model:** a function of input variables

$$f : \mathbb{R}^p \to \mathbb{R}$$
$$\vec{x} \mapsto \dots$$

```
def model(x):
    ...
    return ...
```

Supervised machine learning problems

- **Supervised** machine learning: learn a **predictive model**

- Example 1 (**classification**): Predict whether a DNA sequence is an **enhancer or not**
- Example 2 (**regression**): Predict **plant yield** from the expression of genes

Unsupervised machine learning problems

- **Unsupervised** machine learning: **data exploration**

- Example 1 (**dimensionality reduction**): **project** SNP data on principal components
- Example 2 (**clustering**): find **groups** of cells with **similar** scRNA-seq patterns
- Example 3 (**generative modeling/density estimation**): generate plausible DNA sequences

Why use **supervised** machine learning in genomics?

- For the **predictions**
- For the **interpretation**
  - Example 1: Predict whether a sample is a case or a control
  - Example 2: Predict the residual tumor size after treatment

How machine learning works

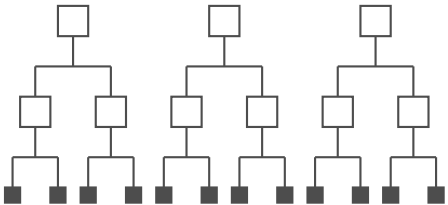
(A very simplified view)

– Choose a **family of models**

How machine learning works

(A very simplified view)

– Choose a **family of models**

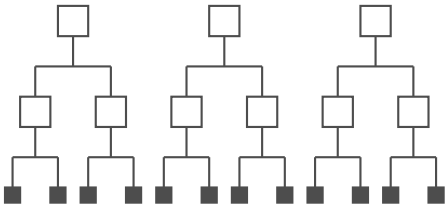


Random forest

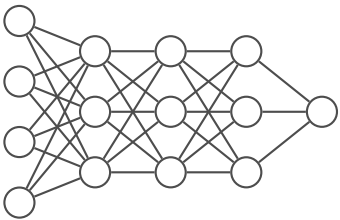
How machine learning works

(A very simplified view)

– Choose a **family of models**



Random forest

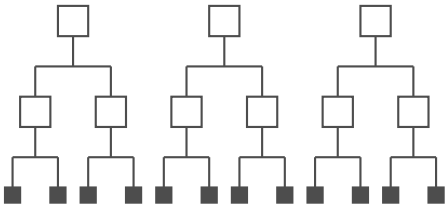


Neural network

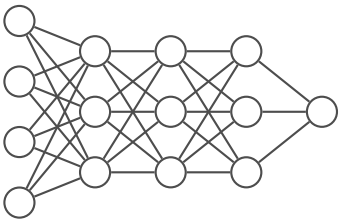
How machine learning works

(A very simplified view)

– Choose a **family of models**



Random forest



Neural network

– **Empirical risk minimization:** Use the data to find, in this family, a model with **minimal error.**

Machine learning works best …

… when the data is **really big**

- ImageNet: 14 million images
- Llama4 training set: 30 trillion tokens

… when the **nature of the data** is well understood

$\Rightarrow$ good representations/modeling/architecture

… when the **nature of the problem** is well understood

humans can do it

… for **making predictions** rather than **explaining how** they were made

Genomics does not fit this picture very well!

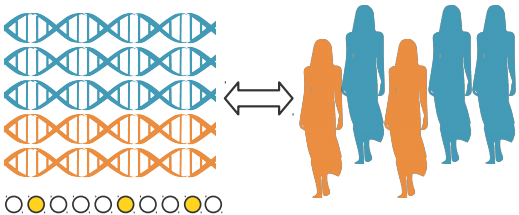
Talk outline

I. Many features, few samples: the example of genotype-to-phenotype studies

II. Good representations of genomic data

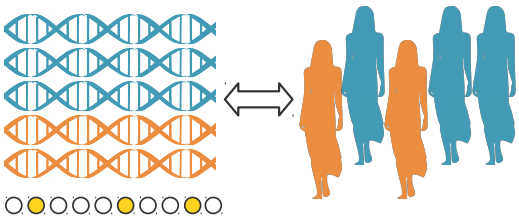
I. Many features, few samples

Genotype-to-phenotype studies



Which genomic features explain the phenotype?

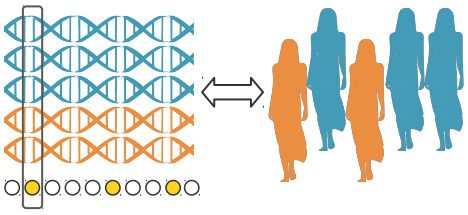
Genotype-to-phenotype studies



Which genomic features explain the phenotype?

– Typically **fewer samples** than **genomic features** (gene expressions, SNPs, etc)

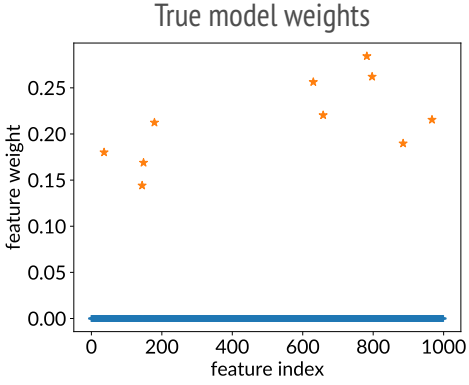
State of the art: Statistical tests



Perform a **statistical test of association** between **each feature** and the phenotype.

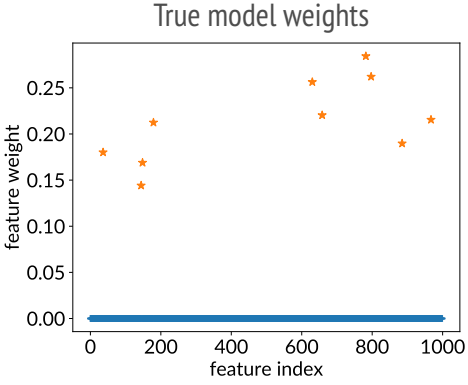
Simulation

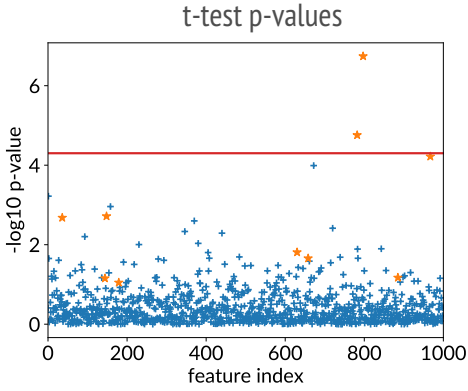
- 100 **samples** 1 000 **features** 10 of which **influence** the phenotype
- $y = \sum_{j=1}^{1000} w_j x_j + \varepsilon$



Simulation

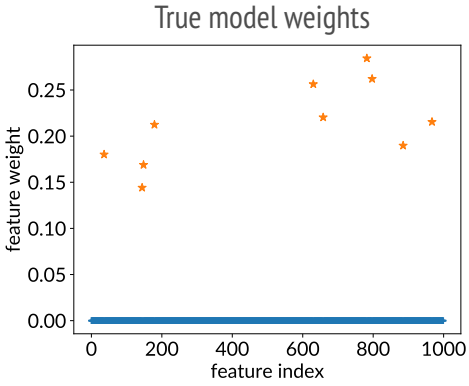
- 100 **samples** 1 000 **features** 10 of which **influence** the phenotype
- $y = \sum_{j=1}^{1000} w_j x_j + \varepsilon$





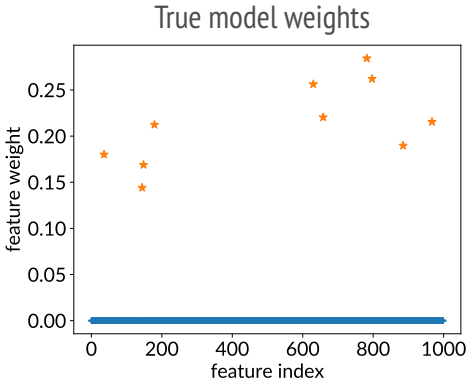
Simulation: linear regression

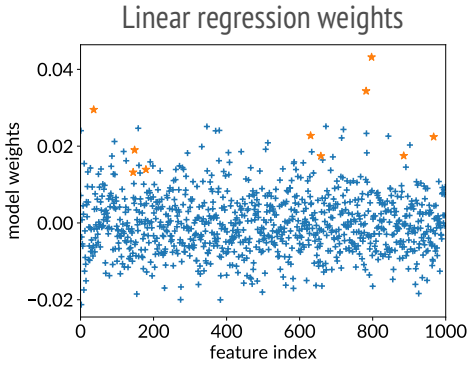
- 100 **samples** 1 000 **features** 10 of which **influence** the phenotype
- $y = \sum_{j=1}^{1000} w_j x_j + \varepsilon$



Simulation: linear regression

- 100 **samples** 1 000 **features** 10 of which **influence** the phenotype
- $y = \sum_{j=1}^{1000} w_j x_j + \varepsilon$





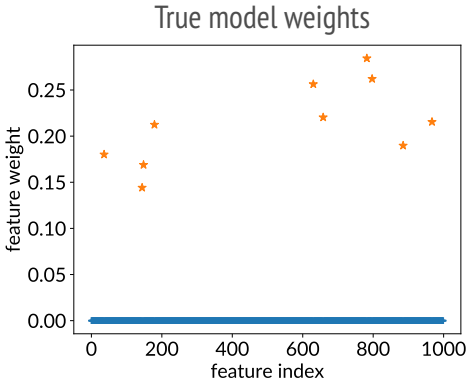
Regularization

- **Empirical risk minimization:** find a model with minimal error on the training data
- **Regularization:** force the model to respect some additional constraints
  - **Weight decay**: don't allow the model parameters to take large values
    (or ridge/Tikhonov/ℓ_2 regularization)
  - **Sparsity**: don't allow too many of the model parameters to have non-zero values
    E.g.: Lasso (or ℓ_1 regularization)

V. Tozzo et al. **Where do we stand in regularization for life science studies?** Journal of Computational Biology 2022

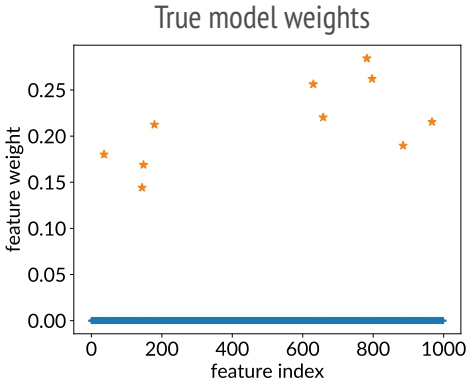
Simulation: Lasso

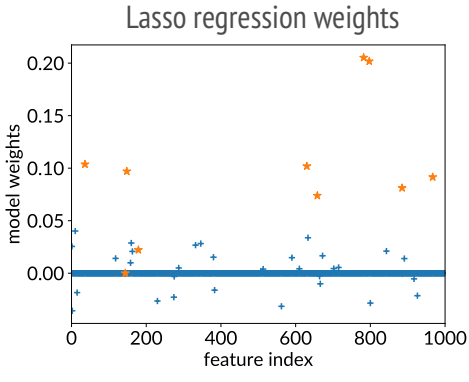
- 100 **samples** 1 000 **features** 10 of which **influence** the phenotype
- $y = \sum_{j=1}^{1000} w_j x_j + \varepsilon$



Simulation: Lasso

- 100 **samples** 1 000 **features** 10 of which **influence** the phenotype
- $y = \sum_{j=1}^{1000} w_j x_j + \varepsilon$



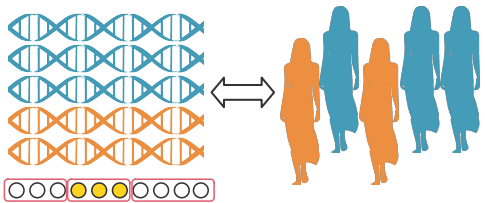


Regularization to integrate prior biological knowledge

- **Goals:**
  - Make the model **consistant** with previously established knowledge
  - Help find a **good model**
  - Increase **interpretability**
- Prior biological knowledge has **structure:**
  - **Groups:** genes belonging to the same pathway / regulated by the same transcription factor; SNPs belonging to the same LD block
  - **Graphs:** biological networks

V. Tozzo et al. **Where do we stand in regularization for life science studies?** Journal of Computational Biology 2022

Group-based regularization



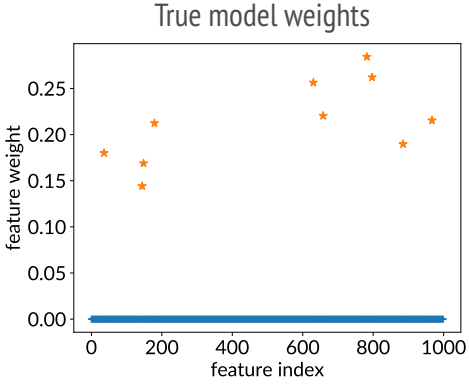
Variants of the **Lasso** encourage the sparsity pattern to **respect a given groups structure**: features that belong to the same provided group will tend to be selected together

- Group Lasso [YL05]
- Overlapping Group Lasso [JOV09]

Simulation: Group Lasso

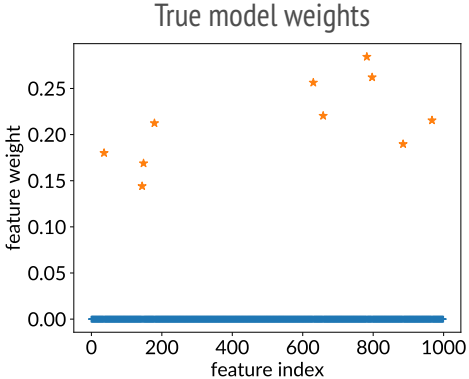
- 100 **samples**
  1 000 **features** 10 of which **influence** the phenotype and **form two of the provided groups**

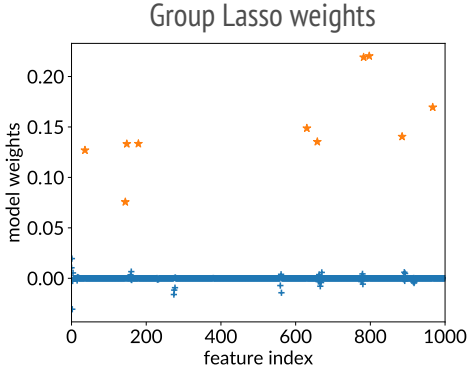
- $y = \sum_{j=1}^{1000} w_j x_j + \varepsilon$



Simulation: Group Lasso

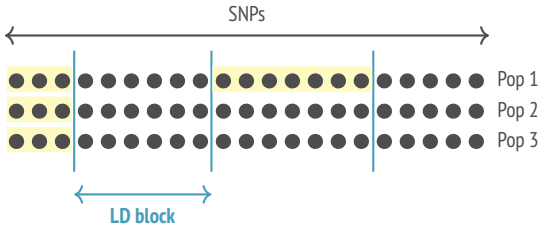
- 100 **samples**
  1 000 **features** 10 of which **influence** the phenotype and **form two of the provided groups**
- $y = \sum_{j=1}^{1000} w_j x_j + \varepsilon$





SMuGLasso for GWAS in diverse populations

- **G = Group:** Group SNPs by **linkage disequilibrium blocks** [DAN15]
- Split samples by genetically homogeneous population (PCA + clustering) $\rightarrow$ **tasks**
  - **Mu = Multitask:** same blocks are selected across tasks [OTJ09]
  - **S = Sparse:** some blocks are task-specific

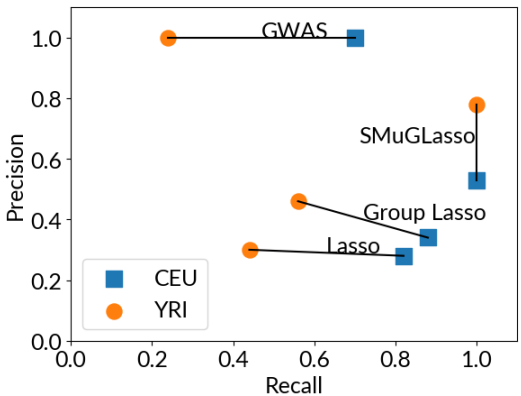


A. Nouira & C.-A. Azencott. **Sparse Multitask group Lasso for genome-wide association studies**. PLoS Comp Bio 2025

SMuGLasso has better recall than other methods

Simulation with GWAsimulator [LL07]

- 2 populations from HapMap3:
  - **CEU** (1300 cases, 1700 controls)
  - **YRI** (400 cases, 600 controls)
- 50 000 SNPs
  - 200 disease-causing SNPs
  - 50 **CEU-specific** SNPs
  - 50 **YRI-specific** SNPs

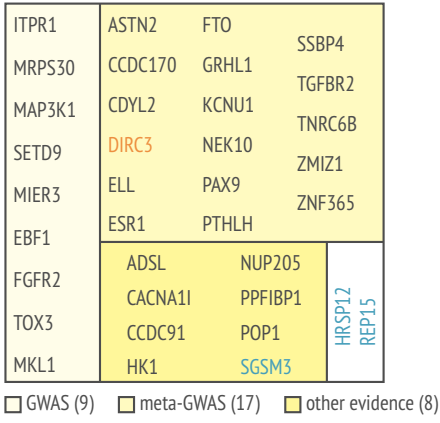


A. Nouira & C.-A. Azencott. **Sparse Multitask group Lasso for genome-wide association studies**. PLoS Comp Bio 2025

SMuGLasso identifies disease genes

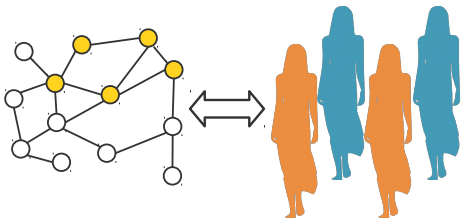
DRIVE dataset [Hun+10]

- 13 846 breast cancer cases, 14 435 controls
- 312 237 SNPs after quality control
- PCA + kmeans $\rightarrow$ 2 populations:
  - **Pop1** (USA, Australia, Denmark)
  - **Pop2** (USA, Cameroon, Nigeria, Uganda)



A. Nouira & C.-A. Azencott. **Sparse Multitask group Lasso for genome-wide association studies**. PLoS Comp Bio 2025

Graph-based regularization



Variants of the **Lasso** encourage the sparsity pattern to **respect the structure of a given graph**: features that are connected on the provided graph will tend to be selected together

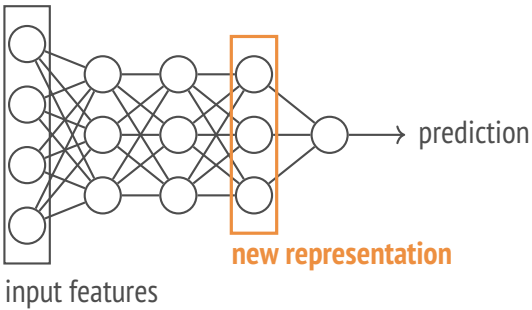
- Network-constrained Lasso [LL08]
- Graph Lasso [JOV09]
- Graph-guided fused Lasso [KSX09]

H. Climente-González et al. **A network-guided protocol to discover susceptibility genes in genome-wide association studies**. STAR Prot 2023
H. Climente-González et al. **Boosting GWAS using biological networks: A study on susceptibility to familial breast cancer**. PLoS Comp Bio 2021
C.-A. Azencott. **Network-guided biomarker discovery**. Lecture Notes in Computer Science 2016

II. Good representations

Representation learning

- **Good representations** = features from which learning is “easy”
- If we cannot **handcraft** good features using domain knowledge, can we **learn** them?



Foundation models

- **LOTS** of **broad** data
  - LAION-5B: 5.85 billion image-text pairs
  - GPT-3 was trained on 570 GB of text
- **self-supervision:**
  - Masked language modeling, next sentence prediction
  - Reconstructing a blurred, partially erased or scrambled image
- **Fine-tuning:** learned representations can then be used for any downstream task

Foundation models in genomics

- Pre-training = masked language modeling

- **NucleotideTransformer** [DT+24]
  - trained on 4k genomes (300B 6bp tokens)
  - 50M to 2.5B parameters
  - 12 kbp context length
  - trained on 16 $\times$8 A100 GPUs ($\sim$ 20 000 €)
  - Try it out: https://hclimente.eu/blog/hf-transformers/

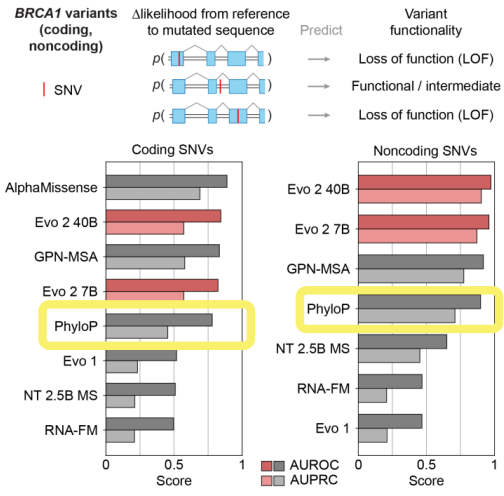
- **Evo2** [Bri+25]
  - trained on up 8.8 Tbp (1 token = 1bp)
  - 7 to 40 B parameters
  - 1 million bp context length
  - training took 2.25 $\times 10^{24}$ FLOPS (on par with Llama 3.1)

Variant pathogenicity prediction

- **Evo2** predicts BRAC1 variant pathogenicity
  - without training!
  - “unnatural” sequence = pathogenic

Variant pathogenicity prediction

- **Evo2** predicts BRAC1 variant pathogenicity
  - without training!
  - "unnatural" sequence = pathogenic
- So does **PhyloP** [Pol+09]
  - conservation score
  - number of parameters: 2
- much better than **NucleotideTransformer**



Closing remarks

When **evaluating** a machine learning model, question

- Whether the **evaluation data sets** are appropriate
- Whether the **evaluation metrics** are appropriate
- Whether the gain in performance is **good enough**
  - appropriate baselines and comparison partners
  - worth the effort/resources

Keep the use case in mind!

References I

[Aze+13] Chloé-Agathe Azencott et al. "Efficient network-guided multi-locus association mapping with graph cuts". In: *Bioinformatics* 29.13 (2013), pp. i171i179. ISSN: 1367-4811. DOI: 10.1093/bioinformatics/btt238.

[Aze16] Chloé-Agathe Azencott. "Network-guided biomarker discovery". In: *Machine Learning for Health Informatics*. Springer International Publishing, 2016, pp. 319336. DOI: 10.1007/978-3-319-50478-0_16.

[Bri+25] Garyk Brixi et al. "Genome modeling and design across all domains of life with Evo 2". In: (2025). DOI: 10.1101/2025.02.18.638918.

[CG+21] Héctor Climente-González et al. "Boosting GWAS using biological networks: A study on susceptibility to familial breast cancer". In: *PLOS Computational Biology* 17.3 (2021), e1008819. DOI: 10.1371/journal.pcbi.1008819.

[CGAY23] Héctor Climente-González, Chloé-Agathe Azencott, and Makoto Yamada. "A network-guided protocol to discover susceptibility genes in genome-wide association studies using stability selection". In: *STAR Protocols* 4.1 (2023). DOI: 10.1016/j.xpro.2022.101998.

[DAN15] Alia Dehman, Christophe Ambroise, and Pierre Neuvial. "Performance of a blockwise approach in variable selection using linkage disequilibrium information". In: *BMC Bioinformatics* 16.1 (2015). DOI: 10.1186/s12859-015-0556-6.

[DT+24] Hugo Dalla-Torre et al. "Nucleotide Transformer: building and evaluating robust foundation models for human genomics". In: *Nature Methods* 22.2 (2024), pp. 287297. DOI: 10.1038/s41592-024-02523-z.

References II

[Hun+10] David J. Hunter et al. *Discovery, Biology, and Risk of Inherited Variants in Breast Cancer (DRIVE) Oncoarray Genotypes.* 2010. URL: https://www.ncbi.nlm.nih.gov/projects/gap/cgi-bin/study.cgi?study_id=phs001265.v1.p1.

[JOV09] Laurent Jacob, Guillaume Obozinski, and Jean-Philippe Vert. "Group lasso with overlap and graph lasso". In: *Proceedings of the 26th Annual International Conference on Machine Learning*. ICML 09. ACM, 2009, pp. 433440. DOI: 10.1145/1553374.1553431.

[KSX09] Seyoung Kim, Kyung-Ah Sohn, and Eric P. Xing. "A multivariate regression approach to association analysis of a quantitative trait network". In: *Bioinformatics* 25.12 (2009), pp. i204i212. DOI: 10.1093/bioinformatics/btp218.

[LL07] Chun Li and Mingyao Li. "GWAsimulator: a rapid whole-genome simulation program". In: *Bioinformatics* 24.1 (2007), pp. 140142. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btm549.

[LL08] Caiyan Li and Hongzhe Li. "Network-constrained regularization and variable selection for analysis of genomic data". In: *Bioinformatics* 24.9 (2008), pp. 11751182. DOI: 10.1093/bioinformatics/btn081.

[NA] Asma Nouira and Chloé-Agathe Azencott. "Multitask group Lasso for Genome Wide association Studies in diverse populations". In: *Pacific Symposium on Biocomputing 2022*, pp. 163–174. DOI: 10.1142/9789811250477_0016.

[NA25] Asma Nouira and Chloé-Agathe Azencott. "Sparse multitask group Lasso for genome-wide association studies". In: *PLOS Computational Biology* 21.9 (2025), e1012734. DOI: 10.1371/journal.pcbi.1012734.

References III

[OTJ09] Guillaume Obozinski, Ben Taskar, and Michael I. Jordan. "Joint covariate selection and joint subspace selection for multiple classification problems". In: *Statistics and Computing* 20.2 (2009), pp. 231252. DOI: 10.1007/s11222-008-9111-x.

[Pol+09] Katherine S. Pollard et al. "Detection of nonneutral substitution rates on mammalian phylogenies". In: *Genome Research* 20.1 (2009), pp. 110121. DOI: 10.1101/gr.097857.109.

[Sug+14] Mahito Sugiyama et al. "Multi-task feature selection on multiple networks via maximum flows". In: *Proceedings of the 2014 SIAM International Conference on Data Mining*. 2014, pp. 199207. DOI: 10.1137/1.9781611973440.23.

[Tib96] Robert Tibshirani. "Regression Shrinkage and Selection Via the Lasso". In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 58.1 (1996), pp. 267288. ISSN: 1467-9868. DOI: 10.1111/j.2517-6161.1996.tb02080.x.

[Toz+22] Veronica Tozzo et al. "Where do we stand in regularization for life science studies?" In: *Journal of Computational Biology* 29.3 (2022), pp. 213232. DOI: 10.1089/cmb.2019.0371.

[YL05] Ming Yuan and Yi Lin. "Model Selection and Estimation in Regression with Grouped Variables". In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 68.1 (2005), pp. 4967. DOI: 10.1111/j.1467-9868.2005.00532.x.

Empirical risk minimization

– The idea behind (most) supervised machine learning algorithms:

Find a model f in the **hypothesis space** $\mathcal{F}$ that **minimizes** the **empirical risk**.

$$\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \underbrace{\mathcal{L}(y_i, f(\vec{x}_i))}_{\text{loss}}$$

Empirical risk minimization

- The idea behind (most) supervised machine learning algorithms:

  Find a model f in the **hypothesis space** $\mathcal{F}$ that **minimizes** the **empirical risk**.

$$\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \underbrace{\mathcal{L}(y_i, f(\vec{x}_i))}_{\text{loss}}$$

- Examples of **losses:**
  - For a regression problem, the **quadratic loss**

$$\mathcal{L}(y, f(\vec{x})) = (y - f(\vec{x}))^2$$

  - For a binary classification problem, the **logistic loss**

$$\mathcal{L}(y, f(\vec{x})) = -y \log(f(\vec{x})) - (1 - y) \log(1 - f(\vec{x}))$$

Regularized empirical risk minimization

- Idea: impose **a priori constraints** on the solution of the empirical risk minimization problem
- **Parametric** models: $\mathcal{F} = \{f_{\vec{w}}; \vec{w} \in \mathbb{R}^d\}$

$$\min_{\vec{w} \in \mathbb{R}^d} \underbrace{\frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(y_i, f_{\vec{w}}(\vec{x}_i))}_{\text{loss}} + \lambda \underbrace{\Omega(\vec{w})}_{\text{regularizer}}$$

Example: Lasso

- **Linear model:** $f_{\vec{w}}(\vec{x}) = \langle \vec{w}, \vec{x} \rangle = w_0 + \sum_{j=1}^{p} w_j x_j$
- Regularized empirical risk minimization

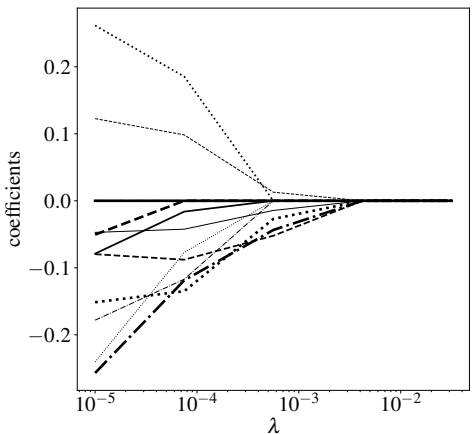
$$\min_{\vec{w} \in \mathbb{R}^p} \underbrace{\frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(y_i, \langle \vec{w}, \vec{x_i} \rangle)}_{\text{loss}} + \lambda \underbrace{\Omega(\vec{w})}_{\text{regularizer}}$$

- **Prior knowledge / a priori constraints:** few features are relevant.
- **Lasso:** $\Omega(\vec{w}) = ||\vec{w}||_1 = \sum_{j=0}^{p} |w_j|$ [Tib96]
- **Sparsity:** many features are assigned a weight of 0. They can be removed from the model.

Regularization coefficient λ

$$\min_{\vec{w}\in\mathbb{R}^p} \underbrace{\frac{1}{n}\sum_{i=1}^{n}\mathcal{L}(y_i, \langle \vec{w}, \vec{x}_i\rangle)}_{\text{loss}} + \lambda \underbrace{\Omega(\vec{w})}_{\text{regularizer}}$$

- λ controls the amount of regularization
- Typically set by **grid search** + **cross-validation:** {number of folds} $\times$ {number of values on the grid} experiments
- For the lasso, efficient ways to get the entire regularization path $\{\vec{w}_\lambda$ for $\lambda \in \{\lambda_{\text{min}}, \ldots, \lambda_{\text{max}}\}\}$



Group-based regularization

$$\min_{\vec{w}\in\mathbb{R}^p} \underbrace{\frac{1}{n}\sum_{i=1}^{n}\mathcal{L}(y_i, \langle \vec{w}, \vec{x}_i\rangle)}_{\text{loss}} + \lambda \underbrace{\Omega_{\text{group}}(\vec{w})}_{\text{group-level regularizer}}$$

- Given a way of grouping the p features in G groups $\mathcal{G}_1, \ldots, \mathcal{G}_G$, each of size p_g, define Ω_{group} to encourage **the selection of only a few groups**

- **Group Lasso** [YL05]

$$\Omega_{\text{group}}(\vec{w}) = \sum_{g=1}^{G} \sqrt{p_g} \sum_{j\in\mathcal{G}_g} w_j^2$$

- **Overlapping Group Lasso** [JOV09]

Multitask regularization

$$\min_{\vec{w}\in\mathbb{R}^p} \sum_{t=1}^{T} \underbrace{\frac{1}{n_t} \sum_{i=1}^{n_t} \mathcal{L}(y_i^{(t)}, \langle \vec{w}^{(t)}, \vec{x}_i^{(t)} \rangle)}_{\text{loss}} + \lambda \underbrace{\Omega_{\text{tasks}}(\vec{w}^{(1)}, \ldots, \vec{w}^{(T)})}_{\text{task regularizer}}$$

- Given T **related tasks,** define Ω_{tasks} so as to solve the T empirical risk minimization problems in such a way that **the same features are selected across tasks.**

- **Multitask Lasso** [OTJ09]

$$\Omega_{\text{tasks}}(\vec{w}^{(1)}, \ldots, \vec{w}^{(T)}) = \sum_{t=1}^{T} \sum_{j=1}^{p} \left(w_j^{(t)} \right)^2$$

MuGLasso

Multitask **G**roup Lasso:

- **multitask group**-level sparsity
- the same groups are selected for all tasks

$$\min_{\vec{w}\in\mathbb{R}^p} \underbrace{\sum_{t=1}^{T} \frac{1}{n_t} \sum_{i=1}^{n_t} \mathcal{L}(y_i^{(t)}, \langle \vec{w}^{(t)}, \vec{x}_i^{(t)} \rangle)}_{\text{loss}} + \lambda \underbrace{\sum_{g=1}^{G} \sqrt{p_g} \sum_{j\in\mathcal{G}_g} \sum_{t=1}^{T} \left(w_j^{(t)} \right)^2}_{\text{mutltitask group-level sparsity}}$$

- If $T = 1 \rightarrow$ group Lasso
- If $G = p$ and $\mathcal{G}_1, \ldots, \mathcal{G}_p = \{1\}, \ldots, \{p\} \rightarrow$ multitask Lasso

A. Nouira & C.-A. Azencott. **Multitask group Lasso for genome-wide association studies in diverse populations**. PSB 2022

SMuGLasso

Sparse **Mu**ltitask **G**roup Lasso:

- the same groups are selected for all tasks
- among those, some groups can be task-specific

$$\min_{\vec{w}\in\mathbb{R}^p} \sum_{t=1}^{T} \underbrace{\frac{1}{n_t}\sum_{i=1}^{n_t} \mathcal{L}(y_i^{(t)}, \langle \vec{w}^{(t)}, \vec{x}_i^{(t)} \rangle)}_{\text{loss}} + \lambda \underbrace{\sum_{g=1}^{G} \sqrt{p_g} \sum_{j\in\mathcal{G}_g} \sum_{t=1}^{T} \left(w_j^{(t)}\right)^2}_{\text{mutltitask group-level sparsity}} + \lambda_2 \underbrace{\sum_{g=1}^{G} \sqrt{p_g} \sum_{j\in\mathcal{G}_g} \sum_{t=1}^{T} \left|w_j^{(t)}\right|}_{\text{task-level sparsity}}$$

A. Nouira & C.-A. Azencott. **Sparse Multitask group Lasso for genome-wide association studies**. PLoS Comp Bio 2025

Graph-based regularization

$$\min_{\vec{w} \in \mathbb{R}^p} \underbrace{\frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(y_i, \langle \vec{w}, \vec{x}_i \rangle)}_{\text{loss}} + \lambda_s \underbrace{||\vec{w}||_1}_{\text{sparsity}} + \lambda_g \underbrace{\Omega_{\text{graph}}(\vec{w})}_{\text{connectivity}}$$

– Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ of p nodes over the features, define Ω_{graph} to encourage the sparsity pattern to **respect the structure of** $\mathcal{G}$.

Graph-based regularization

$$\min_{\vec{w}\in\mathbb{R}^p} \underbrace{\frac{1}{n}\sum_{i=1}^{n}\mathcal{L}(y_i,\langle\vec{w},\vec{x}_i\rangle)}_{\text{loss}} + \lambda_s \underbrace{||\vec{w}||_1}_{\text{sparsity}} + \lambda_g \underbrace{\Omega_{\text{graph}}(\vec{w})}_{\text{connectivity}}$$

- Given a graph $\mathcal{G}=(\mathcal{V},\mathcal{E})$ of p nodes over the features, define Ω_{graph} to encourage the sparsity pattern to **respect the structure of $\mathcal{G}$**.
- **Graph-fused Lasso** [KSX09]

$$\Omega_{\text{graph}}(\vec{w}) = \sum_{(v_j,v_k)\in\mathcal{E}} |w_j - w_k|$$

- **Network-constrained Lasso** [LL08]

$$\Omega_{\text{graph}}(\vec{w}) = \vec{w}^\top L\vec{w} = \sum_{(v_j,v_k)\in\mathcal{E}} A_{jk}(w_j - w_k)^2$$

Graph-based regularization

$$\min_{\vec{w}\in\mathbb{R}^p} \underbrace{\frac{1}{n}\sum_{i=1}^{n}\mathcal{L}(y_i,\langle\vec{w},\vec{x}_i\rangle)}_{\text{loss}} + \lambda_s \underbrace{||\vec{w}||_1}_{\text{sparsity}} + \lambda_g \underbrace{\Omega_{\text{graph}}(\vec{w})}_{\text{connectivity}}$$

- Given a graph $\mathcal{G}=(\mathcal{V},\mathcal{E})$ of p nodes over the features, define Ω_{graph} to encourage the sparsity pattern to **respect the structure of** $\mathcal{G}$.
- **Graph Lasso:** overlapping group lasso with edges as groups [JOV09]

$$\Omega_{\text{graph}}(\vec{w}) = \inf_{(\vec{\beta}_1,\ldots,\vec{\beta}_{\mathcal{E}}):\vec{w}=\sum_{k=1}^{|\mathcal{E}|}\beta_k} \sum_{k=1}^{|\mathcal{E}|}||\beta_k||_2^2 \qquad \vec{\beta}_k\in\mathbb{R}^p \text{ s.t. } \vec{\beta}_{kj}\neq 0 \text{ iff node } j \text{ in edge } k$$

Network-constrained Lasso

$$\min_{\vec{w}\in\mathbb{R}^p} \underbrace{\frac{1}{n}\sum_{i=1}^{n}\mathcal{L}(y_i,\langle\vec{w},\vec{x}_i\rangle)}_{\text{loss}} + \lambda_s \underbrace{||\vec{w}||_1}_{\text{sparsity}} + \lambda_g \underbrace{\vec{w}^\top L\vec{w}}_{\text{connectivity}}$$

Can be solved as a **Lasso on transformed data**

$$X^* = \frac{1}{\sqrt{1+\lambda_g}}\begin{pmatrix} X \\ \sqrt{\lambda_g}S^\top \end{pmatrix} \in \mathbb{R}^{(n+m)\times p} \qquad \vec{y}^* = \begin{pmatrix} \vec{y} \\ 0 \end{pmatrix} \in \mathcal{Y}^{n+m}$$

where $S \in \mathbb{R}^{m\times p}$ such that $L = SS^\top$

with regularization parameter $\frac{\lambda_s}{\sqrt{1+\lambda_g}}$ and then $\vec{w} = \sqrt{1+\lambda_g}\vec{w}^*$

Network-constrained Lasso

$$\min_{\vec{w}\in\mathbb{R}^p} \underbrace{\frac{1}{n}\sum_{i=1}^{n}\mathcal{L}(y_i,\langle\vec{w},\vec{x_i}\rangle)}_{\text{loss}} + \lambda_s \underbrace{||\vec{w}||_1}_{\text{sparsity}} + \lambda_g \underbrace{\vec{w}^\top L\vec{w}}_{\text{connectivity}}$$

Can be solved as a **Lasso on transformed data**

$$X^* = \frac{1}{\sqrt{1+\lambda_g}}\begin{pmatrix} X \\ \sqrt{\lambda_g}S^\top \end{pmatrix} \in \mathbb{R}^{(n+m)\times p} \qquad \vec{y}^* = \begin{pmatrix} \vec{y} \\ 0 \end{pmatrix} \in \mathcal{Y}^{n+m}$$

where $S \in \mathbb{R}^{m\times p}$ such that $L = SS^\top$

with regularization parameter $\frac{\lambda_s}{\sqrt{1+\lambda_g}}$ and then $\vec{w} = \sqrt{1+\lambda_g}\vec{w}^*$

- Defining S:
  - Option 1 ($m = p$) and $S = U\Lambda^{1/2}$ with $L = U\Lambda U^\top \rightarrow$ runtime issues ☹
  - Option 2 ($m = |\mathcal{E}|$) and S is the incidence matrix of $\mathcal{G} \rightarrow$ memory issues ☹