Challenges of Machine Learning for Transcriptomics

AI for genomics Bootcamp

Paul Bertin

Mila, Université de Montréal

February 7, 2020
Outline

1 Introduction

2 From real world to input data
   - Dataset biases
   - Acquisition biases
   - Preprocessing

3 The supervised learning pipeline
   - The curse of dimensionality
   - Making the right assumptions: inspiration from Comp. Vis.
   - Which assumptions for transcriptomics?
   - Gene interaction graphs?
   - Parameter sharing among genes?
   - Similar response to perturbation in latent space?

4 Model interpretability
   - Feature importance for deep models
   - Simpson’s paradox

5 Conclusion
Applications of deep learning in genomics.

- Lots of applications of deep learning in genomics
- Today focus on applications to transcriptomics and its challenges

Figure taken from *A primer on deep learning in genomics*
What are transcriptomics?

- The study of an organism’s transcriptome, the sum of all of its RNA transcripts
- We will focus on RNA-seq and single cell RNA-seq

Figure taken from https://www.biocompare.com/Bench-Tips/345311-Single-Cell-Set-Up-Sample-Preparation-Tips/
Lots of different cell types

- NK Cell
- Cytotoxic T Cell
- Helper T Cell
- Follicular Dendritic Cell
- Macrophage
- Treg
- B Cell
- Plasma Cell
- Mast Cell
- Basophil
- Neutrophil
- Eosinophil
More and more data

Figure: Plot of commercial release dates versus machine outputs per run are shown. Numbers inside data points denote current read lengths. Sequencing platforms are color coded.

Figure taken from High-Throughput Sequencing Technologies
Apply modern machine learning techniques?

**Long term goals:**
- Individualized medicine
- Better understanding the biology

**Today’s objective**
Identify and understand the challenges facing machine learning (ML) and deep learning (DL) techniques when applied to transcriptomics

**Why should you care?**
- Be aware of the limitations of usual ML
- Take those limitations into account when you use ML
- Discover fields of research in ML
How can machine learning help?

Example of a pipeline to find better cancer treatment using Machine Learning

Let us study the limitations associated with each step
Outline

1. Introduction
2. From real world to input data
   - Dataset biases
   - Acquisition biases
   - Preprocessing
3. The supervised learning pipeline
   - The curse of dimensionality
   - Making the right assumptions: inspiration from Comp. Vis.
   - Which assumptions for transcriptomics?
   - Gene interaction graphs?
   - Parameter sharing among genes?
   - Similar response to perturbation in latent space?
4. Model interpretability
   - Feature importance for deep models
   - Simpson’s paradox
5. Conclusion
Outline

1 Introduction

2 From real world to input data
   - Dataset biases
   - Acquisition biases
   - Preprocessing

3 The supervised learning pipeline
   - The curse of dimensionality
   - Making the right assumptions: inspiration from Comp. Vis.
   - Which assumptions for transcriptomics?
   - Gene interaction graphs?
   - Parameter sharing among genes?
   - Similar response to perturbation in latent space?

4 Model interpretability
   - Feature importance for deep models
   - Simpson’s paradox

5 Conclusion
Overview

Simplified environment

Tumor → Cancerous cell line → Sequences → Gene Counts → Drug sensitivity prediction → What drug to use given gene expression

- Acquisition biases
- Loss of information
- Imperfect fit
- Confounding effects

Modelize
RNA sequencing
Preprocessing
Machine Learning
Feature importance
Cell lines, a model for tumors

Cell lines as a model

Cell lines are a simplified model of tumors

- Tumor are complex tissues
- Composed of different cell types
- Evolving in a living organism

Figure taken from Biological Pathways Involved in Tumor Angiogenesis and Bevacizumab Based Anti-Angiogenic Therapy with Special References to Ovarian Cancer
Why is this an issue for ML?

Train set
Lab experiments

Test set
Patients
A refresher on supervised learning

- Learn to predict target $Y$ given input $X$
- We model $P_\theta(x|y)$ and learn the parameters $\theta$ based on pairs of examples

Questions: Are there any assumptions on the dataset?
Let’s have a quiz!

What can Supervised Learning do?

A: Replace any domain knowledge
B: Provide explanations for observed patterns
C: Estimate functions from IID samples
D: Reliably generalize to other domains
Machine Learning cannot do everything!

What can Supervised Learning do?

A: Replace any domain knowledge
B: Provide explanations for observed patterns
C: Estimate functions from IID samples
D: Reliably generalize to other domains
The main assumption of Supervised Learning

▶ **Intuition**: Pick balls at random from the same *bag* (and put the ball back before picking another one)

**Independent Identically Distributed**

All samples are independently drawn from a fixed probability distribution

▶ This assumption can be violated in several ways

---

**Figure**: Counterexample where train and test inputs have different distributions
Covariate shift

Figure: X causes Y

- Happens in X causes Y problems
- **Covariate shift**: $P(x)$ changes between train and test but $P(y|x)$ does not change
- At test time, the model will be confronted with parts of the input space that it has not seen during training
Covariate Shift
Prior probability shift

Figure: Y causes X

- Happens in Y causes X problems
- **Prior probability shift**: $P(y)$ changes between train and test but $P(x|y)$ does not change
- Difficult because both the input distribution $P(x)$ and what we model ($P(y|x)$) change
Concept shift

- **Concept shift** in \( X \) causes \( Y \) problems: \( P(x) \) does not change but \( P(y|x) \) changes

- **Concept shift** in \( Y \) causes \( X \) problems: \( P(y) \) does not change but \( P(x|y) \) changes
Different shifts together?

What happens in transcriptomics?

In biology, there are most certainly (very) complicated relationships between inputs and targets. Probably lots of things change together.

▶ Examples: **covariate shift** from one individual to the other, **concept drift** from one cell type to the other.

▶ Quick (imperfect) fix: Normalize data

▶ Warning: Normalizing also means loosing information!

Figure: Z causes X and Y in addition to direct effects.
Selection bias

selected subset
Multiple studies

study 1

study 2

study 3

study 1

study 2

study 3
Towards multi-environment learning and meta-learning?

- Other learning procedures exist and can be adapted to transcriptomics
- **Multi-environment training**: assume that data comes from different environments
- **Meta-learning**: Learn to adapt fast to a new environment

---

The TCGA Meta-Dataset Clinical Benchmark

Mandana Samiei¹  Tobias Würfl²  Tristan Deleu³  Martin Weiss³  
Francis Dutil⁵  Thomas Fevens¹  Geneviève Boucher³,⁴  Sebastien Lemieux³,⁴  
Joseph Paul Cohen³
Outline

1 Introduction

2 From real world to input data
   - Dataset biases
   - Acquisition biases
   - Preprocessing

3 The supervised learning pipeline
   - The curse of dimensionality
   - Making the right assumptions: inspiration from Comp. Vis.
   - Which assumptions for transcriptomics?
   - Gene interaction graphs?
   - Parameter sharing among genes?
   - Similar response to perturbation in latent space?

4 Model interpretability
   - Feature importance for deep models
   - Simpson’s paradox

5 Conclusion
Figure 3 Technical errors and coverage in single-cell sequencing data. (a) Technical errors that occur in single-cell sequencing (SCS) data include: false-positive errors, allelic dropout events and false-negative errors due to insufficient coverage. ‘Pop’ indicates a population of cells. (b) Coverage metrics in SCS data include coverage depth and total physical coverage, or breadth. (c) Coverage uniformity, or ‘eveness’ in SCS data can vary from cell to cell, but is often more uniform in standard genomic DNA sequencing experiments using populations of cells.

Figure taken from Cancer genomics: one cell at a time
.dropout in single cell

Gene counts

0 0 0
0 0 0

Denoising Autoencoder

How to denoise the data?

Extracting and Composing Robust Features with Denoising Autoencoders

Pascal Vincent
Hugo Larochelle
Yoshua Bengio
Pierre-Antoine Manzagol

Université de Montréal, Dept. IRO, CP 6128, Succ. Centre-Ville, Montréal, Quebec, H3C 3J7, Canada
Denoising Autoencoder
ADAGE signature analysis: differential expression analysis with data-defined gene sets

Jie Tan¹, Matthew Huyck²,³, Dongbo Hu², René A. Zelaya², Deborah A. Hogan³ and Casey S. Greene²*
Outline

1. Introduction

2. From real world to input data
   - Dataset biases
   - Acquisition biases
   - Preprocessing

3. The supervised learning pipeline
   - The curse of dimensionality
   - Making the right assumptions: inspiration from Comp. Vis.
   - Which assumptions for transcriptomics?
   - Gene interaction graphs?
   - Parameter sharing among genes?
   - Similar response to perturbation in latent space?

4. Model interpretability
   - Feature importance for deep models
   - Simpson’s paradox

5. Conclusion
Overview

Loss of information

Tumor
- Modelize

Cancerous cell line
- RNA sequencing

Sequences
- Preprocessing

Gene Counts
- Machine Learning

Drug sensitivity prediction
- Feature importance

What drug to use given gene expression

Simplified environment

Acquisition biases

Imperfect fit

Confounding effects
The acquisition process

Figure 3: Technical errors and coverage in single-cell sequencing data. (a) Technical errors that occur in single-cell sequencing (SCS) data include: false-positive errors, allelic dropout events and false-negative errors due to insufficient coverage. ‘Pop’ indicates a population of cells. (b) Coverage metrics in SCS data include coverage depth and total physical coverage, or breadth. (c) Coverage uniformity, or ‘eveness’ in SCS data can vary from cell to cell, but is often more uniform in standard genomic DNA sequencing experiments using populations of cells.

Figure taken from Cancer genomics: one cell at a time
A key step

Preprocessing is a key step that determines what data will be fed to our machine learning model. Each technique comes with limitation and drawbacks

- Need to account for different **total amounts of reads** in the different samples
- Need to account for the **lengths of the genes**
- Classic normalization methods: RPKM (Read Per Kilobase Million), FPKM (Fragment Per Kilobase Million), TPM (Transcripts Per Kilobase Million)
- **Alignment**: different reference genomes can be used from one dataset to the other!
Tremendous information loss!

Towards a more clever preprocessing?

A lot of information is lost during those preprocessing steps, limiting what can be achieved downstream. Moreover, the non-standardized normalization and alignment limit our ability to transfer knowledge from one dataset to the other.

- Could we do better?
- Example: RNA velocity inference using splicing information.
RNA velocity

you can visit https://scvelo.org

Figure taken from RNA velocity of single cells
RNA velocity

Figure taken from *RNA velocity of single cells*
Outline

1. Introduction
2. From real world to input data
   - Dataset biases
   - Acquisition biases
   - Preprocessing
3. The supervised learning pipeline
   - The curse of dimensionality
   - Making the right assumptions: inspiration from Comp. Vis.
   - Which assumptions for transcriptomics?
   - Gene interaction graphs?
   - Parameter sharing among genes?
   - Similar response to perturbation in latent space?
4. Model interpretability
   - Feature importance for deep models
   - Simpson’s paradox
5. Conclusion
Outline

1. Introduction
2. From real world to input data
   - Dataset biases
   - Acquisition biases
   - Preprocessing
3. The supervised learning pipeline
   - The curse of dimensionality
   - Making the right assumptions: inspiration from Comp. Vis.
   - Which assumptions for transcriptomics?
   - Gene interaction graphs?
   - Parameter sharing among genes?
   - Similar response to perturbation in latent space?
4. Model interpretability
   - Feature importance for deep models
   - Simpson’s paradox
5. Conclusion
Fat data

Figure: When data has lots of feature but few examples, data is called fat
Fat data: Beware of spurious correlations!

- **Spurious correlations**: with fat data, features can be highly correlated together out of chance!
- Example: binary, independent features and 2 samples. Some features will have a correlation of 1 out of chance!

**Conclusion**

*In high dimension, you need lots of examples!*
Estimate a function of 1 variable

We want a high density of samples (distance between two sample points $< \frac{1}{N}$) in order to estimate the function reliably.

$N$ samples to estimate the function on $[0, 1]$
Estimate a function of 2 variables

To have the same density of sample in 2 dimensions we need $N^2$ samples.

In dimension 3, we need $N^3$ samples...
And with 20,000 dimensions?
Volumes in high dimension

The volume of a hypercube $[0, a] \times [0, a] \times \ldots \times [0, a]$ of dimension $d$ is:

$$a^d$$

In high dimension, volumes are very big!

- This is why **kNN does not work in high dimension**
- How to estimate a function in dimension $20k$?
- **Machine learning is about making the right assumptions** to overcome the need for many samples.
Outline

1. Introduction
2. From real world to input data
   - Dataset biases
   - Acquisition biases
   - Preprocessing
3. The supervised learning pipeline
   - The curse of dimensionality
   - Making the right assumptions: inspiration from Comp. Vis.
   - Which assumptions for transcriptomics?
   - Gene interaction graphs?
   - Parameter sharing among genes?
   - Similar response to perturbation in latent space?
4. Model interpretability
   - Feature importance for deep models
   - Simpson’s paradox
5. Conclusion
Automatic feature extraction

Figure by Y. LeCun
Comparison with logistic regression

1 hidden layer NN: the features fed to the logistic regression are learnt
Going deeper!
Going deeper!
Example: **Resnet**, a 34 layer network!

- Needs additional trick (residuals) for forward and backward signals to pass through
The bias variance trade-off

- High model complexity: high variance low bias
- Low model complexity: low variance high bias
- You have to choose the right model complexity!
  (regularization, model depth,...)
The bias variance trade-off

![Diagram showing the relationship between model complexity, error, bias, and variance. The graph illustrates how increasing model complexity reduces bias but increases variance, aiming for an optimal model complexity where the total error is minimized.](image-url)
Why does ML work so well in Computer Vision?

People have made simplifying assumptions that hold well in Computer Vision

- Let us dive into the details of CNNs
Fully connected

- $X_L$ and $X_{L+1}$ activations vectors in layers $L$ and $L+1$

- $X_{L+1} = \sigma(W_L X_L + B)$

Figure: matrix $W_L$
Fully connected

\[ x_{L+1} = \sigma(x + B) \]
Convolutional layer

- $X_{L+1} = \sigma(W_L X_L + B)$

- parameter sharing: constraints on $W_L$

**Figure:** matrix $W_L$ with constraints
\[ \sigma(x + B) \]

where \( x \) and \( B \) are matrices of weights and biases, respectively, and \( \sigma \) is the activation function.
Equivariance
Equivariance

$$\sigma(x + B)$$
Equivariance

$\sigma(\mathbf{x} + \mathbf{B})$
1. Introduction

2. From real world to input data
   - Dataset biases
   - Acquisition biases
   - Preprocessing

3. The supervised learning pipeline
   - The curse of dimensionality
   - Making the right assumptions: inspiration from Comp. Vis.
   - Which assumptions for transcriptomics?
     - Gene interaction graphs?
     - Parameter sharing among genes?
     - Similar response to perturbation in latent space?

4. Model interpretability
   - Feature importance for deep models
   - Simpson’s paradox

5. Conclusion
What are the right assumptions for gene expression data?

- We would like to add **prior knowledge** and/or make **biologically grounded assumptions**

<table>
<thead>
<tr>
<th>What are the right assumptions for gene expression data?</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Use gene interaction graphs?</td>
</tr>
<tr>
<td>- Assume similarity of processes between genes?</td>
</tr>
<tr>
<td>- Assume similar perturbation response between individuals/species?</td>
</tr>
</tbody>
</table>
Outline

1. Introduction
2. From real world to input data
   - Dataset biases
   - Acquisition biases
   - Preprocessing
3. The supervised learning pipeline
   - The curse of dimensionality
   - Making the right assumptions: inspiration from Comp. Vis.
   - Which assumptions for transcriptomics?
   - Gene interaction graphs?
   - Parameter sharing among genes?
   - Similar response to perturbation in latent space?
4. Model interpretability
   - Feature importance for deep models
   - Simpson’s paradox
5. Conclusion
Incorporate Graph Prior Knowledge?

Figure: Example of a curated graph: StringDB
Idea: Use gene interaction graphs to **constrain a ML model**

2 questions:

- How to use the graph in a machine Learning model?
- Are curated graphs well suited for gene expression data?
We can represent an undirected graph by its adjacency matrix.

**Adjacency matrix**: the value at coordinates \((i, j)\) is 1 if nodes \(i\) and \(j\) are connected, 0 otherwise.
Constraining the model: a simple example

\[ = \sigma(x + B) \]

This is one simple example

Deep learning with graphs is a dynamic field of research!
It does not seem to work!

Curated graphs do not seem to be well suited for gene expression data when using all genes.

### Analysis of Gene Interaction Graphs as Prior Knowledge for Machine Learning Models

*Paul Bertin*
Mila, Université de Montréal
Montréal, Canada

*Mohammad Hashir*
Mila, Université de Montréal
Montréal, Canada

*Martin Weiss*
Mila, Université de Montréal
Montréal, Canada

*Vincent Frappier*
Mila, Université de Montréal
Montréal, Canada

*Theodore J. Perkins*
Ottawa Hospital Research Institute
University of Ottawa
Ottawa, Canada

*Geneviève Boucher*
Institute for Research in Immunology and Cancer
Université de Montréal
Montréal, Canada

*Joseph Paul Cohen*
Mila, Université de Montréal
Montréal, Canada
A current debate

Graph biased feature selection of genes is better than random for many genes

Jake Crawford *1 Casey S. Greene ††

▶ A current debate: What if you choose the right genes?

What’s next?
Could there be an interplay between graph curation and ML model performance?

▸ Identify genes that hurt performance
Outline

1 Introduction
2 From real world to input data
   - Dataset biases
   - Acquisition biases
   - Preprocessing
3 The supervised learning pipeline
   - The curse of dimensionality
   - Making the right assumptions: inspiration from Comp. Vis.
   - Which assumptions for transcriptomics?
   - Gene interaction graphs?
   - Parameter sharing among genes?
   - Similar response to perturbation in latent space?
4 Model interpretability
   - Feature importance for deep models
   - Simpson’s paradox
5 Conclusion
Parameter sharing among genes?

Low dimensional state of the cell

Gene expressions are far from being independent, the data has a lot of structure.

- Idea: Use **Representation Learning** with low dimensional latent spaces (e.g. dim $\sim 500$)
  - We can perform analysis in the lower dimensional latent space (e.g. fit a prediction model)
  - But we still need a matrix of shape (20k, 500): lots of parameters!

- Lots of things (regulatory processes, effects) might be similar among genes

- We can share parameters among genes $\rightarrow$ Diet Networks
Diet Networks: Thin Parameters for Fat Genomics

Adriana Romero, Pierre Luc Carrier, Akram Erraqabi, Tristan Sylvain, Alex Auvolet, Etienne Dejoie
Montreal Institute for Learning Algorithms
Montreal, Quebec, Canada
firstName.lastName@umontreal.ca, except
adriana.romero.soriano@umontreal.ca
and pierre-luc.carrier@umontreal.ca

Marc-André Legault, Marie-Pierre Dubé
1 University of Montreal, Faculty of Medicine
2 Montreal Heart Institute,
3 Beaulieu-Saucier Pharmacogenomics Centre
Montreal, Quebec, Canada
marc-andre.legault.1@umontreal.ca
marie-pierre.dube@umontreal.ca

Julie G. Hussin
Wellcome Trust Centre for Human Genetics
University of Oxford
Oxford, UK
julieh@well.ox.ac.uk

Yoshua Bengio
Montreal Institute for Learning Algorithms
Montreal, Quebec, Canada
yoshua.umontreal@gmail.com
Diet Networks: Magic trick!
Diet Networks

\[ \hat{Y} \xrightarrow{\text{MLP}} \hat{X} \]

\[ \text{We} \xrightarrow{\text{MLP}} \text{Emb.} \]

\[ \text{Wd} \xrightarrow{\text{MLP}} \text{Emb.} \]
Diet Networks

\[ = \sigma(x + B) \]

hidden units
genes
gene features
Auxiliary network
Diet Networks

- Not yet applied successfully to gene expression data!
- The features that are fed to the auxiliary networks have to contain the relevant information about the task you want to solve!

Open question

What features to use for gene expression data?
Outline

1. Introduction
2. From real world to input data
   - Dataset biases
   - Acquisition biases
   - Preprocessing
3. The supervised learning pipeline
   - The curse of dimensionality
   - Making the right assumptions: inspiration from Comp. Vis.
   - Which assumptions for transcriptomics?
   - Gene interaction graphs?
   - Parameter sharing among genes?
   - Similar response to perturbation in latent space?
4. Model interpretability
   - Feature importance for deep models
   - Simpson’s paradox
5. Conclusion
scGen predicts single-cell perturbation responses

Mohammad Lotfollahi, F. Alexander Wolf* and Fabian J. Theis*
Response to perturbation

Observations: arithmetic in latent space seem to make sense (e.g. Word2Vec)

Assumption: response to perturbation is the same in latent space across species/cell types
Outline

1. Introduction

2. From real world to input data
   - Dataset biases
   - Acquisition biases
   - Preprocessing

3. The supervised learning pipeline
   - The curse of dimensionality
   - Making the right assumptions: inspiration from Comp. Vis.
   - Which assumptions for transcriptomics?
   - Gene interaction graphs?
   - Parameter sharing among genes?
   - Similar response to perturbation in latent space?

4. Model interpretability
   - Feature importance for deep models
   - Simpson’s paradox

5. Conclusion
Overview

Tumor → Cancerous cell line → Sequences → Gene Counts → Drug sensitivity prediction → What drug to use given gene expression

Simplified environment → Acquisition biases → Loss of information → Imperfect fit → Confounding effects
Want to get some explanations from the model?

How to better understand what is happening?

How to know what the model is *looking at*? Let us investigate feature importance techniques and their limitations.
Outline

1. Introduction
2. From real world to input data
   - Dataset biases
   - Acquisition biases
   - Preprocessing
3. The supervised learning pipeline
   - The curse of dimensionality
   - Making the right assumptions: inspiration from Comp. Vis.
   - Which assumptions for transcriptomics?
   - Gene interaction graphs?
   - Parameter sharing among genes?
   - Similar response to perturbation in latent space?
4. Model interpretability
   - Feature importance for deep models
   - Simpson’s paradox
5. Conclusion
Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps

Karen Simonyan  Andrea Vedaldi  Andrew Zisserman
Visual Geometry Group, University of Oxford
{karen, vedaldi, az}@robots.ox.ac.uk
Saliency Maps

Figure taken from
Parametric models

\[ \text{Neural Network} \]

\[ \text{Input parameters} \]

\[ \text{Output} \]

\[ \text{P(\text{owl})} \]

\[ \text{P(\text{peacock})} \]

\[ \text{NN(} , \mathbf{w} \text{)} = \]

0.6
-1.2
0.3
1.4
0.2
-0.7

Neural Network

Input parameters

Output

P(\text{owl})

P(\text{peacock})
Usual training

How is a model usually trained?

Iterate:

1. Compute the error for a given input
2. Compute the gradient of the error w.r.t each parameter \( \frac{\partial E}{\partial w_i} \) using backpropagation
3. Update the parameters in order to lower the error:
   \[
   w_i^{t+1} = w_i^t - \lambda \frac{\partial E}{\partial w_i}
   \]

Note
This is called **Stochastic Gradient Descent**

- \( \lambda \) is called the learning rate
- In practice we use several inputs at once (in a **batch**)
- Other **gradient descent** algorithms exist (e.g. *Adam*)
Back to the computation graph

\[
\frac{1}{1 + e^{-(w_0 x_0 + w_1 x_1 + w_2)}}
\]

Figure by J. P. Cohen
Back to the computation graph

\[
\frac{1}{1 + e^{- (w_0x_0 + w_1x_1 + w_2)}}
\]

Backpropagation

Figure by J. P. Cohen
For a given class probability, e.g. $P(\text{owl})$, compute the gradient with respect to the input $\frac{\partial P(\text{owl})}{\partial x_i}$.

We get a real number for each input feature.

**Interpretation**

$\frac{\partial P(\text{owl})}{\partial x_i}$: how much the class probability $P(\text{owl})$ depends on feature $x_i$. 
Intuition

Make the model dream the input that would maximize a given class probability

- Gradient ascent in the input space to maximize a given class probability
Deep Dream: Gradient Ascent in the input space

Update the input by iterating:

\[
\begin{align*}
X'_1 & = X_1 + \varepsilon \\
X'_2 & = X_2 + \varepsilon \\
X'_3 & = X_3 + \varepsilon \\
X'_4 & = X_4 + \varepsilon \\
X'_5 & = X_5 + \varepsilon \\
X'_6 & = X_6 + \varepsilon 
\end{align*}
\]
The input image has been updated in order to maximize the probability of the *dog* class.
At the end of the lecture

**Figure:** You will learn how to dream 3s from other numbers!

Visit the following notebook: **Google colab**
Limitations of feature importance methods

- Feature importance methods can be very noisy and difficult to interpret for gene expression data.
- **Feature importance does not provide a causal explanation as the prediction can be confounded.**
Outline

1. Introduction
2. From real world to input data
   - Dataset biases
   - Acquisition biases
   - Preprocessing
3. The supervised learning pipeline
   - The curse of dimensionality
   - Making the right assumptions: inspiration from Comp. Vis.
   - Which assumptions for transcriptomics?
   - Gene interaction graphs?
   - Parameter sharing among genes?
   - Similar response to perturbation in latent space?
4. Model interpretability
   - Feature importance for deep models
   - Simpson’s paradox
5. Conclusion
Two treatments for kidney stones
Which one is better?

Figure: The size of kidney stones has an effect on both treatment assignment and outcome
Figure: If we do not take age into account, we may conclude that height has a negative influence on basketball performance!
How to understand the mechanisms of the cell?

Would you like to identify the effect of a gene on another gene?

- Lots of confounders
- Current area of research (module networks...)

Figure taken from https://clincancerres.aacrjournals.org/content/21/22/5047
Simpson’s paradox: Take-away

**Take-away**

If you are provided with data that contains several partitions\(^1\), you may want to **fit a model on a whole data** as well as **separate models for each partition**.

- You can analyse the *story* told by feature importance techniques applied to the different models.
- If all models agree, you have an interpretation that is robust across partitions (but no guarantee that the story is true...)
- If not, you may want to investigate further (lab experiments?)

**In transcriptomics, there are lot of unobserved confounders!**
*\(\text{e.g. non coding parts of the genome}\)*

---

\(^1\) *e.g. cell lines, cell types, expression level of important Transcr. Factors*
Outline

1 Introduction
2 From real world to input data
   ● Dataset biases
   ● Acquisition biases
   ● Preprocessing
3 The supervised learning pipeline
   ● The curse of dimensionality
   ● Making the right assumptions: inspiration from Comp. Vis.
   ● Which assumptions for transcriptomics?
   ● Gene interaction graphs?
   ● Parameter sharing among genes?
   ● Similar response to perturbation in latent space?
4 Model interpretability
   ● Feature importance for deep models
   ● Simpson’s paradox
5 Conclusion
Conclusion

- We investigated several challenges of Machine Learning when it is applied to transcriptomic data.
- **We need to design models making the right assumptions for gene expression data!**
Practice time!

Visit the following notebook: Google colab