Intro to Machine Learning

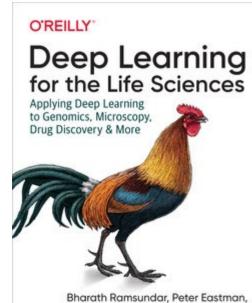
Duke Quantitative Living Systems Center

Justin Savage

1/29/25

Resources

- <u>https://www.deeplearningbook.org</u>
 <u>/</u>
- <u>https://www.oreilly.com/library/vie</u> w/deep-learningfor/9781492039822/



Bharath Ramsundar, Peter Eastman, Patrick Walters & Vijay Pande

DEEP LEARNING

Ian Goodfellow, Yoshua Bengio, and Aaron Courville

What types of problems can computers solve?

What types of problems can computers solve?

Easy for Computers

- Math problems
- Working with lots of data
- Games with formal rules

Hard for Computers

- Recognizing spoken words
- Identifying objects in images
- Understanding data

What types of problems can computers solve?

Easy for Computers

- Math problems
- Working with lots of data
- Games with formal rules

Hard for Computers

- Recognizing spoken words
- Identifying objects in images
- Understanding data

Computers excel at tasks where the key representations of the data are well defined

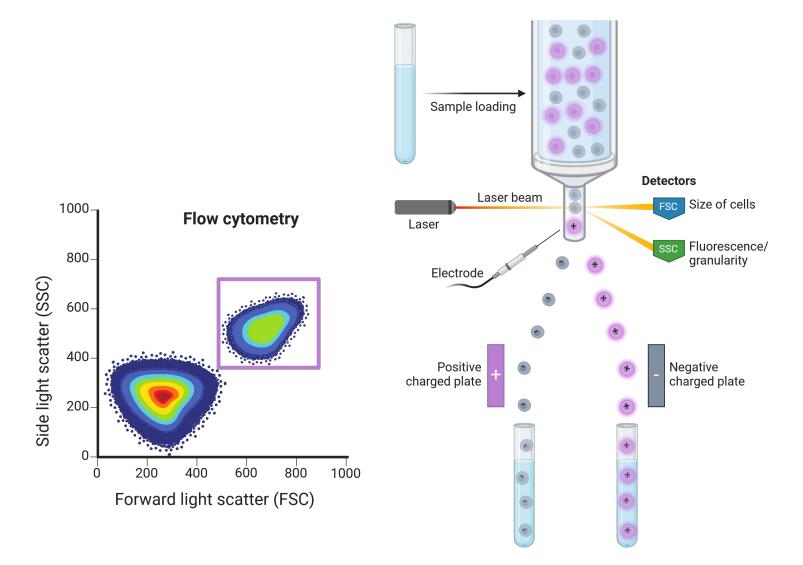
Three flavors of "AI"

- Rule based systems
- Classic Machine Learning
- Representation Learning

Rule based systems

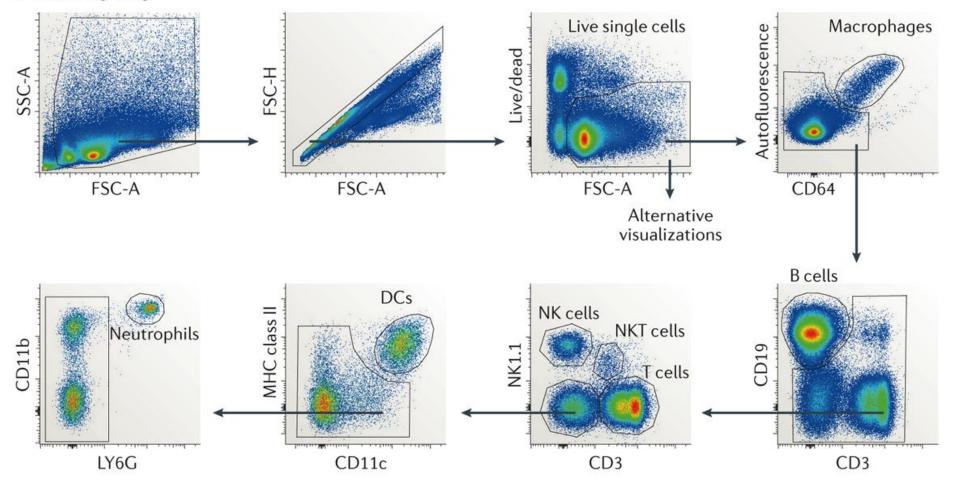
- If condition X then output Y
- Relies on the programmer providing all necessary information for making a decision including:
 - The data in a simple format
 - The exact decision method and boundary values are provided by the programmer
- Example: FACS sorting

Example FACS Sorting



Example FACS Sorting

a Manual gating



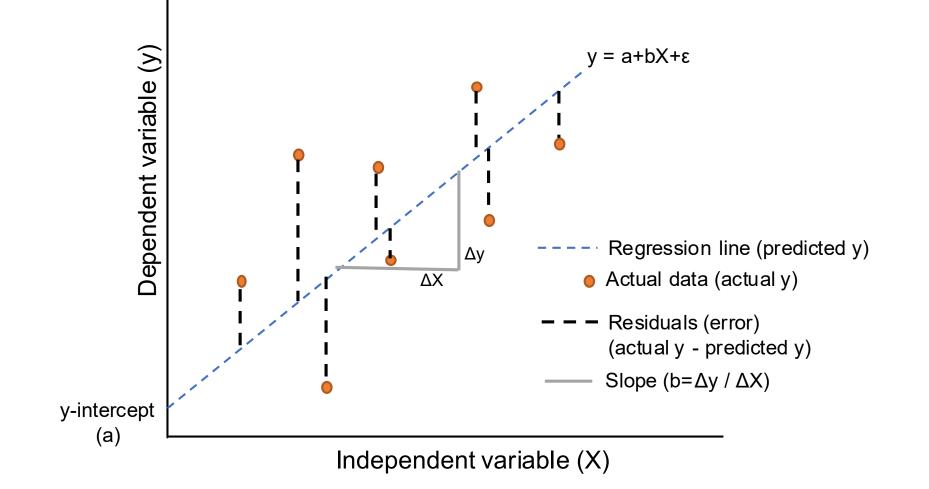
Rule based systems

• "The difficulties faced by systems relying on hard-coded knowledge suggest that AI systems need the ability to acquire their own knowledge, by extracting patterns from raw data. This capability is known as machine learning."

Classical Machine Learning

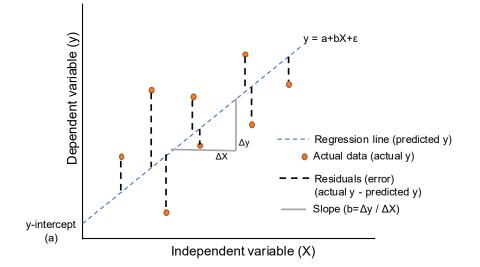
- The program optimizes a small number of parameters (learning)
- Relies on the programmer providing some structure for making a decision including:
 - The data in a simple format
 - A general framework for decision making is provided by the programmer
 - Some feedback mechanism for how far off a prediction is from the truth
- Example: Linear Regression

Linear Regression



Linear Regression

- Program makes a guess for what the parameters (slope and intercept) should be
- Program uses the formula residuals = actual y predicted y for each point to assess how well it predicted
- Program adjusts parameters to minimize the residuals



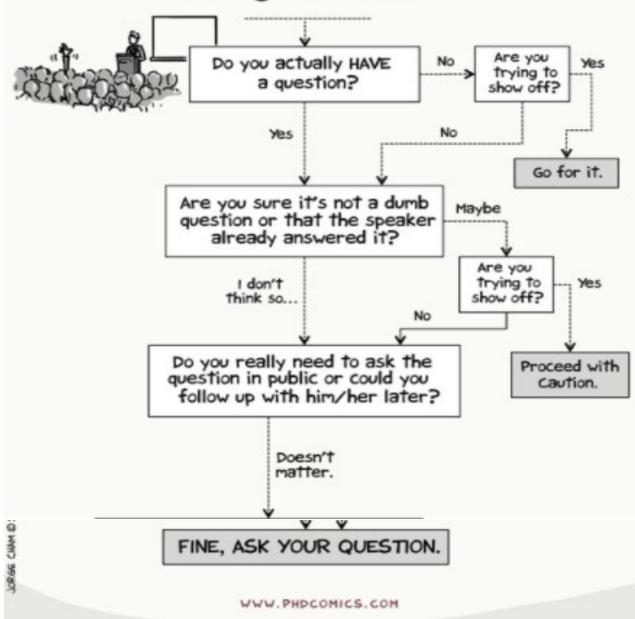
Classical Machine Learning

- The program optimizes a small number of parameters (learning)
- Relies on the programmer providing some structure for making a decision including:
 - The data in a simple format
 - A general framework for decision making is provided by the programmer
 - Some feedback mechanism for how far off a prediction is from the truth
- Example: Linear Regression
- Example: Random Forest (ilastik)

How ilastik works

• Random Forest of Decision Trees

Should you ask a Question during Seminar?



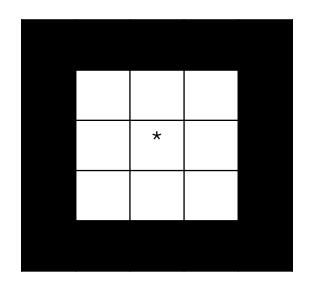
*	

- Trying to figure out if pixel * is a puncta or background
- We know the Intensity of * (range is 0-255)
- We know the position of * and the intensity of all of the pixels around it
- What would be a good first feature to consider to help decide if * is a puncta or background?

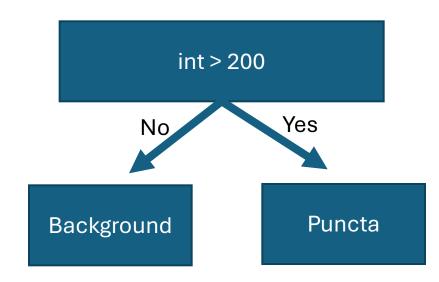
*	

- Trying to figure out if pixel * is a puncta or background
- We know the Intensity of * (range is 0-255)
- We know the position of * and the intensity of all of the pixels around it
- What would be a good first feature to consider to help decide if * is a puncta or background?

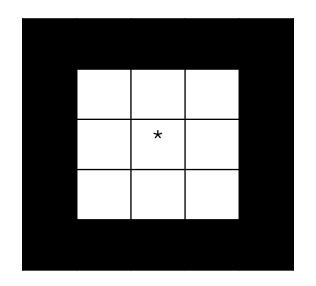
Is * bright enough to be a puncta?

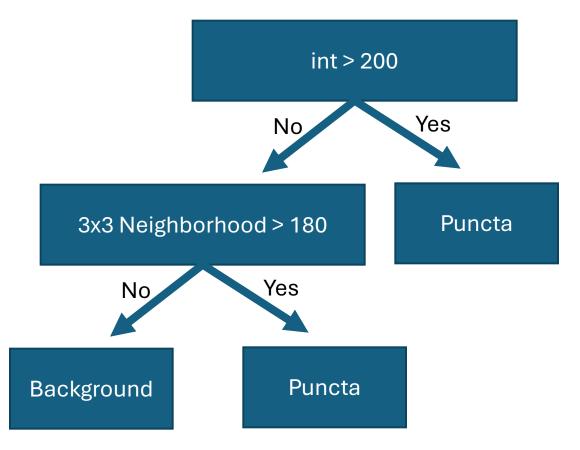


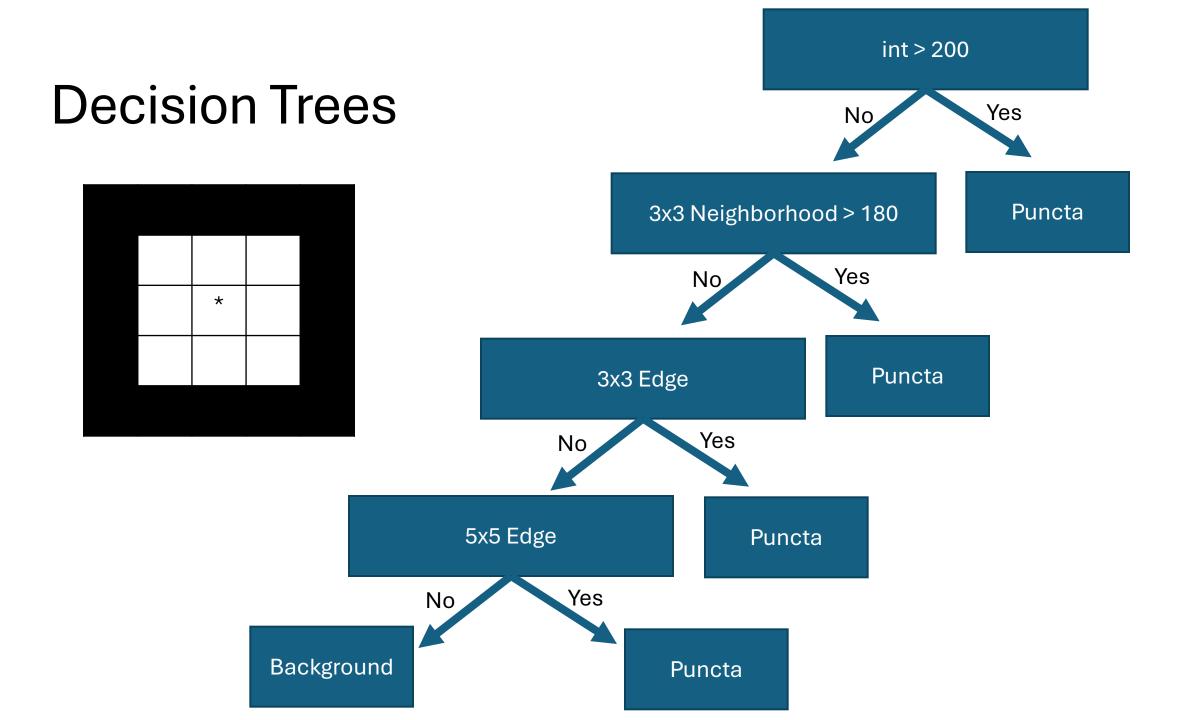
Try an intensity cutoff



This is the same as traditional thresholding, but we can do better







Random Forest

ilastik uses 100 decision trees that are randomly generated to be slightly different from each other
 Each one "votes" for whether a pixel is foreground or background

Viewing Features in Feature Selection Window

ilastik - C:/Users/savag/Desktop/Ilastik_PSD95.ilp - Pixel Classification

- 🗆 🗙

86 % ₫ Δ 🗸

Project Settings View Help > 1. Input Data <mark>ר</mark> ר ✓ 2. Feature Selection (Selected 37 features) Select Features... > 3. Training > 4. Prediction Export 5. Batch Processing Current View: 16772 #3 Vglut1 Image 1-1.tif (red) Features: Raw Data (display only) Gaussian Smoothing (σ =0.3) in 2D Gaussian Smoothing (σ =0.7) in 2D Gaussian Smoothing (σ =1.0) in 2D Gaussian Smoothing (σ =1.6) in 2D Gaussian Smoothing (σ =3.5) in 2D Gaussian Smoothing (σ =5.0) in 2D Gaussian Smoothing (σ =10.0) in 2D Laplacian of Gaussian (σ =0.7) in 2D

5 Y 620 Crosshairs

х

0

Active Requests: 0 Cached Data: 1.0 MB ...

ENG 10:41 AM

US 3/11/2021

Laplacian of Gaussian (σ =1.0) in 2D

🖻 📖 🕮 🕜 🔕 🕲 🐇 🔘 🖬 🖍 🗠 🦛 🕸

Viewing Features in Feature Selection Window

ilastik - C:/Users/savag/Desktop/Ilastik_PSD95.ilp - Pixel Classification X Project Settings View Help n L C > 1. Input Data 86 % 🙆 🛆 🛡 ✓ 2. Feature Selection (Selected 37 features) Select Features... > 3. Training > 4. Prediction Export > 5. Batch Processing Current View: 16772 #3 Vglut1 Image 1-1.tif (red) Features: Laplacian of Gaussian (σ =0.7) in 2D ~ Laplacian of Gaussian (σ =1.0) in 2D Laplacian of Gaussian (σ =1.6) in 2D Laplacian of Gaussian (σ =3.5) in 2D Laplacian of Gaussian (σ =5.0) in 2D Laplacian of Gaussian (σ =10.0) in 2D Gaussian Gradient Magnitude (σ =0.7) in 2D Gaussian Gradient Magnitude (σ =1.0) in 2D Gaussian Gradient Magnitude (σ =1.6) in 2D Gaussian Gradient Magnitude (σ =3.5) in 2D V P P L / F OF OF X 30 Y 576 Crosshairs

0

Active Requests: 0 Cached Data: 1.0 MB ...

10:40 AM

US 3/11/2021

ENG

- C t

Viewing Features in Feature Selection Window

ilastik - C:/Users/savag/Desktop/Ilastik_PSD95.ilp - Pixel Classification

Project Settings View Help

- 🗆 X

ENG

US 3/11/2021

> 1. Input Data	86 % 🙆
✓ 2. Feature Selection	
(Selected 37 features)	
Select Features	
	에는 것은 것은 가슴을 가려면 가슴을 가지 않는 것이 가 바람이 가지 않는 것을 가려야 한다. 가슴을 가 가지 않는 것이 가지 않는 것
> 3. Training	
▶ 4. Prediction Export	
► 5. Batch Processing	
urrent View: 16772 #3 Vglut1 Image 1-1.tif (red) 🔹	
Features:	
Laplacian of Gaussian (σ=0.7) in 2D	
Laplacian of Gaussian (σ=1.0) in 2D	김 것은 것이 같은 것을 하는 것이 가지 않는 것을 하는 것이 같이 많이 많이 봐. 것을 하는 것을 했다.
Laplacian of Gaussian (σ=1.6) in 2D	
Laplacian of Gaussian (σ=3.5) in 2D	
Laplacian of Gaussian (σ=5.0) in 2D	
Laplacian of Gaussian (σ=10.0) in 2D	
Gaussian Gradient Magnitude (σ=0.7) in 2D	
Gaussian Gradient Magnitude (σ=1.0) in 2D	A SUPER SUPERIOR STORES CONCERNING
Gaussian Gradient Magnitude (σ=1.6) in 2D	
Gaussian Gradient Magnitude (σ =3.5) in 2D	all to share a the Busice of the same of the state of the state of the same
	54 Y 721 Crosshairs
	Active Requests: 0 Cached Data: 1.0 M

GAMES

0

₫i

١

0

()

P

🁔 🏠 \land 🗠 🥻 🕸

Classical Machine Learning

- The program optimizes a small number of parameters (learning)
- Relies on the programmer providing some structure for making a decision including:
 - The data in a simple format
 - A general framework for decision making is provided by the programmer
 - Some feedback mechanism for how far off a prediction is from the truth
- Example: Linear Regression
- Example: Random Forest (ilastik)

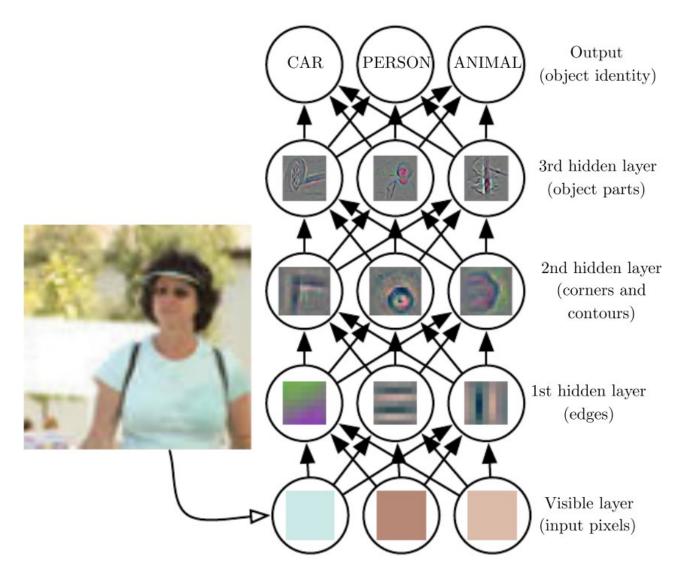
Representation Learning

- For some applications, it is difficult to say which features of a data set are important for a decision
- We may want the algorithm to discover novel combinations of features that could explain the data better than features we have as input
- We may want to eliminate as much of our own bias as possible

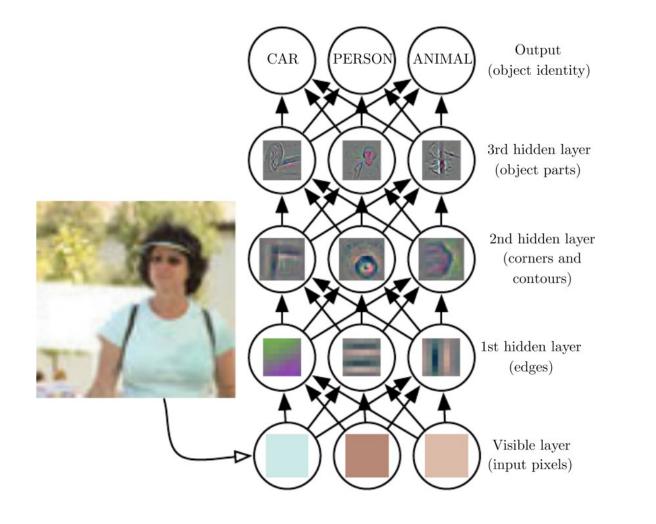
Representation Learning

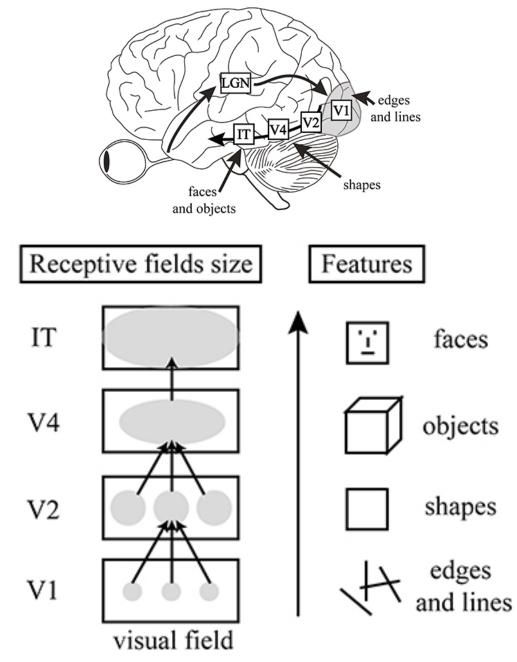
- The program not only learns parameters, but learns which representations of the data are most useful for decision making
- Relies on the programmer providing a minimal framework for making a decision including:
 - The data in a complex, raw format
 - Some structure for how many features to learn and how they should be connected to eachother
 - Some feedback mechanism for how far off a prediction is from the truth
- Example: Artificial Neural Networks

Artificial Neural Networks

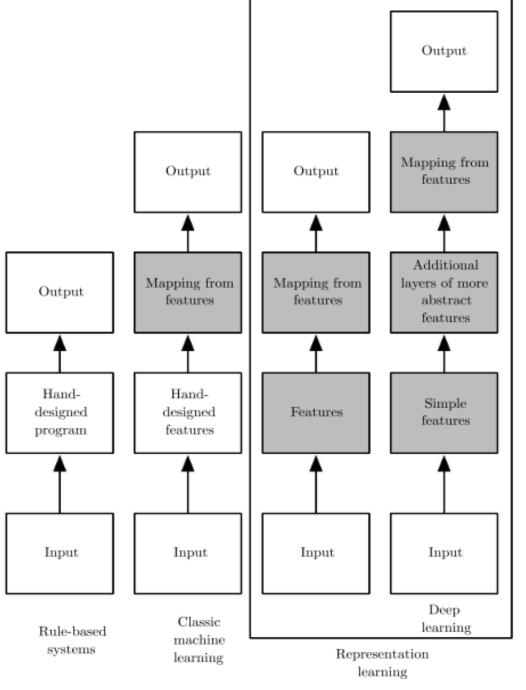


Artificial Neural Networks





Three flavors of "AI"



• If the relevant values and computations are known by the experimenter

- If the relevant values and computations are known by the experimenter
 - Use rule-based systems

- If the relevant values and computations are known by the experimenter
 - Use rule-based systems
- If some relevant features are known, but the best values of the parameters are unknown

- If the relevant values and computations are known by the experimenter
 - Use rule-based systems
- If some relevant features are known, but the best values of the parameters are unknown
 - Classic machine learning

- If the relevant values and computations are known by the experimenter
 - Use rule-based systems
- If some relevant features are known, but the best values of the parameters are unknown
 - Classic machine learning
- If there is no clear set of features that best represent the data

- If the relevant values and computations are known by the experimenter
 - Use rule-based systems
- If some relevant features are known, but the best values of the parameters are unknown
 - Classic machine learning
- If there is no clear set of features that best represent the data
 - Representation learning/ Deep learning

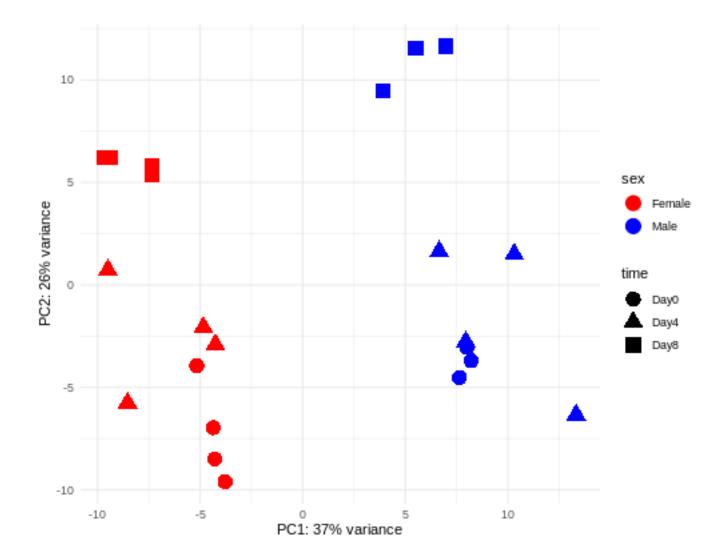
Biological Example: RNAseq data

- RNAseq data from M and F animals collected at different ages
- You want to determine how variable the samples are and if there are effects of sex and/or age
- You've measured thousands of genes and don't know which genes would be changing

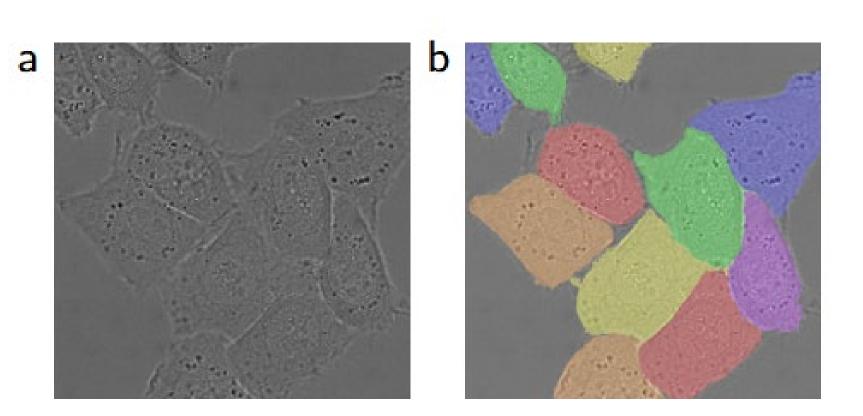
Biological Example: RNAseq data

- RNAseq data from M and F animals collected at different ages
- You want to determine how variable the samples are and if there are effects of sex and/or age
- You've measured thousands of genes and don't know which genes would be changing
- ^We need a better representation of the data to more easily visualize the differences between samples

Biological Example: RNAseq data PCA

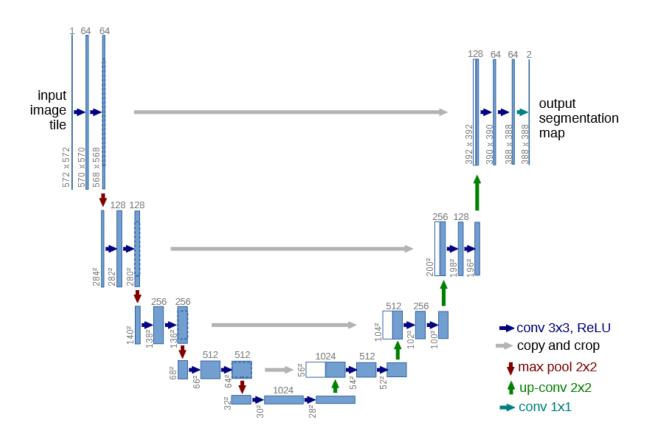


https://carpentries-incubator.github.io/bioc-rnaseq/05-exploratory-qc.html

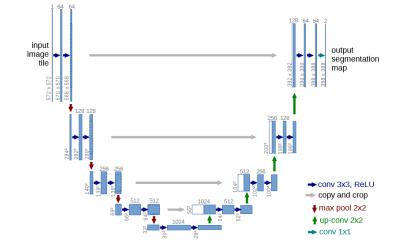


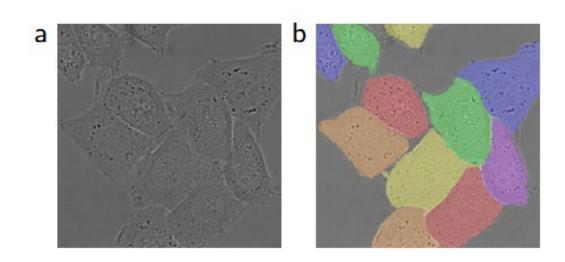
U-net: Convolutional networks for biomedical image segmentation

<u>O Ronneberger</u>, <u>P Fischer</u>, <u>T Brox</u> - ... **image** computing and computer ..., 2015 - Springer ... use of **convolutional networks** is on classification tasks, where the output to an **image** is a single class label. However, in many visual tasks, especially in **biomedical image processing**, ... \therefore Save \mathfrak{D} Cite Cited by 100570 Related articles All 32 versions

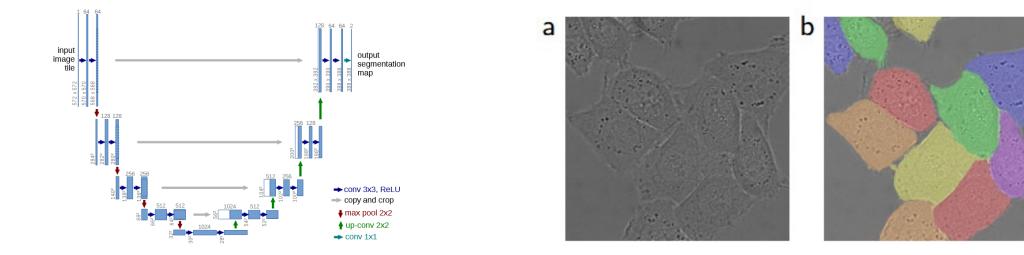


- How does the neural network actually learn?
 - Weights in the network are randomly initialized

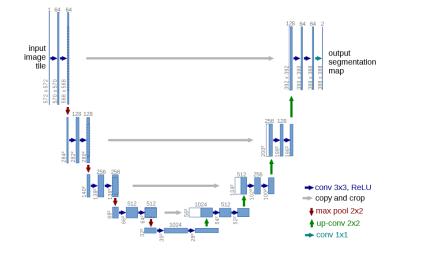


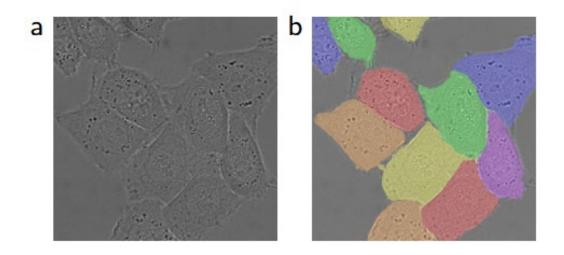


- How does the neural network actually learn?
 - Weights in the network are randomly initialized
 - Loss function compares if each pixel is categorized correctly

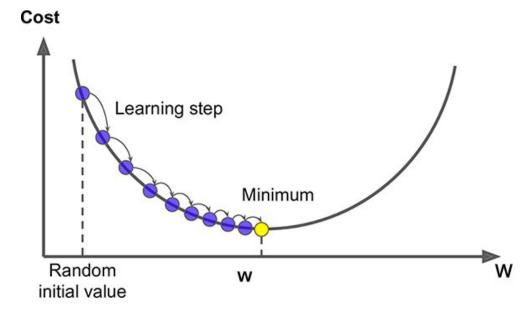


- How does the neural network actually learn?
 - Weights in the network are randomly initialized
 - Loss function compares if each pixel is categorized correctly
 - Gradient descent is used to move the weights in the network towards the correct values





- How does the neural network actually learn?
 - Weights in the network are randomly initialized
 - Loss function compares if each pixel is categorized correctly
 - Gradient descent is used to move the weights in the network towards the correct values



- How does the neural network actually learn?
 - Weights in the network are randomly initialized
 - Loss function compares if each pixel is categorized correctly
 - Gradient descent is used to move the weights in the network towards the correct values

