

# Machine Learning

## 1. Motivation + Theorie

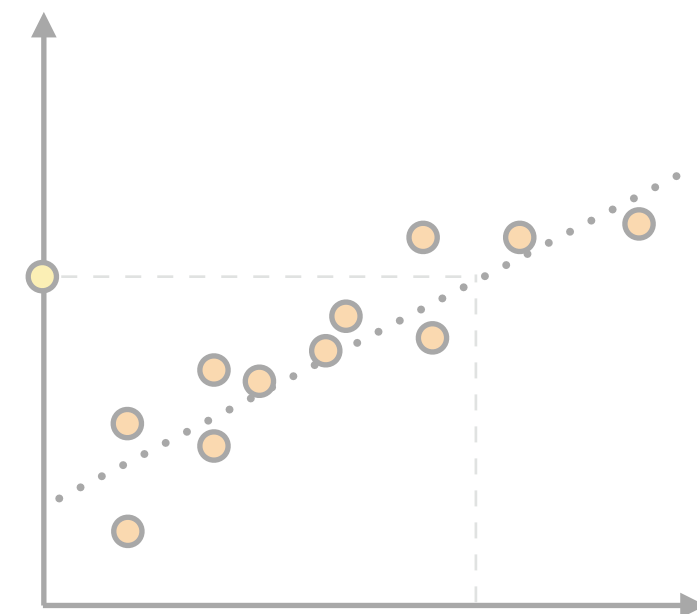
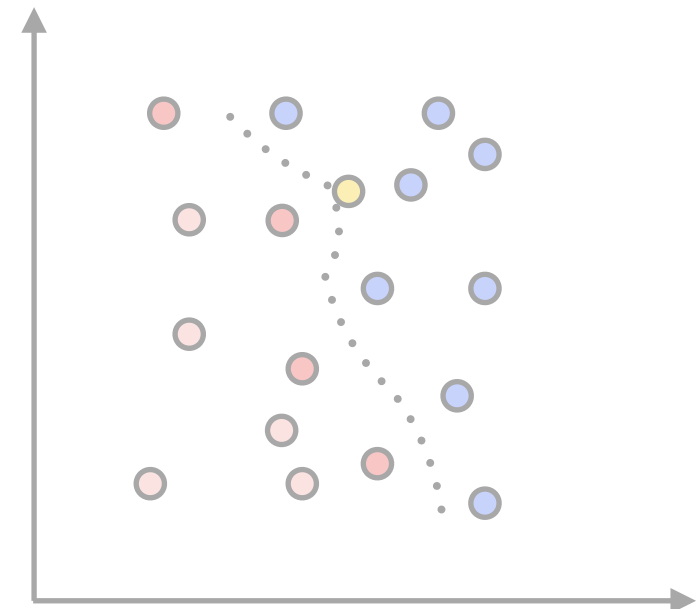
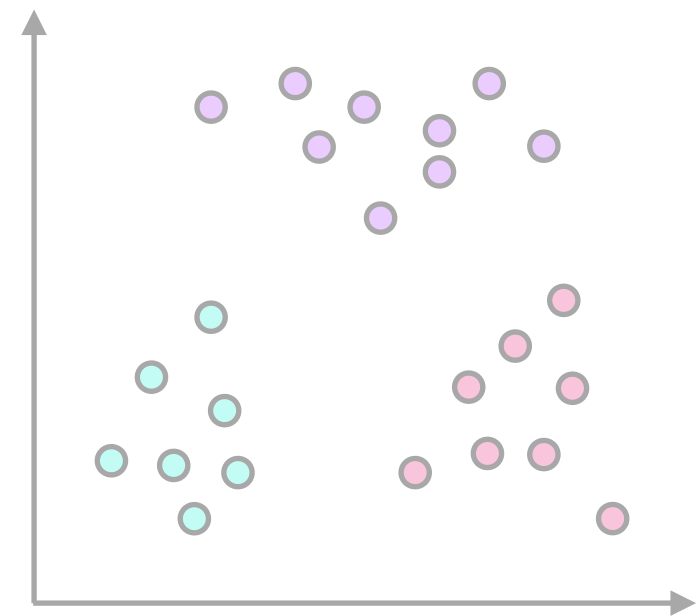
*Siegfried Gessulat*

SAP Health Potsdam

Technical University of Munich  
Chair of Proteomics and and Bioanalytics

[s.gessulat@tum.de](mailto:s.gessulat@tum.de)

FH Ludwigshafen  
2018-10-08



# Course Outline

## Block I Foundations

### Oct 08: Introduction

- Overview machine learning
- Theory: Linear Algebra
- Algorithms: Knn, K-means

### Oct 09: Basics

- Theory: linear regression, logistic regression
- Algorithms: gradient descent

## Block II Best practices

### Oct 29: Neural Networks

- Data cleaning
- Algorithm: Neural Networks

### Oct 30: Best practices

- Theory: Cross validation
- Theory: Regularization

# Course Outline

## **Block III Dark Arts**

### Nov 19: Tricks of the Trade

- Ensembles
- Hyperparameter Search
- Deep Learning Black Magic

### Nov 20: Outlook

- Theory: Dimensionality Reduction

# Outline Today

1. Preliminaries
2. Dataset: MNIST
3. What is Machine Learning?
4. Notation
5. Classification: K nearest neighbors (Knn)
6. Clustering: K-means
7. Theory: Linear Algebra refresher
8. Application: Python Intro, Implementation
9. Dataset: CIFAR-10

# Preliminaries

## 1. Programming Assignments

Teams of two (1 SAP 1 non-SAP), randomly assigned.

## 2. Schedule, Dates & Deadlines

3 blocks. 1 assignment per block.

Help desk: Monday after each block - doodle your slot.

Deadlines: Friday after the help desk.

Results: Friday after the deadline.

<b>Lecture</b>						
Oct 8 <b>Mo</b>	Oct 9 <b>Tue</b>	<b>Wed</b>	<b>Thu</b>	Oct 12 <b>Fr</b>		
<b>Help</b>				<b>Deadline</b>		
Oct 15 <b>Mo</b>	<b>Tue</b>	<b>Wed</b>	<b>Thu</b>	Oct 19 <b>Fr</b>		
				<b>Results</b>		
Oct 22 <b>Mo</b>	<b>Tue</b>	<b>Wed</b>	<b>Thu</b>	Oct 26 <b>Fr</b>		

# Preliminaries

## 1. Programming Assignments

Teams of two (1 SAP 1 non-SAP), randomly assigned.

## 2. Schedule, Dates & Deadlines

3 blocks. 1 assignment per block.

Help desk: Monday after each block

Deadlines: Friday after the help desk.

Results: Friday after the deadline.

**What's your  
schedule Monday?**

## 3. Grades

Assignment I: 30 points

Assignment II: 30 points

Assignment III: 40 points

Late assignments will cost you points.

## 4. Resources

After the end of one block, you will get an email with slides, code demonstrations, assignments and a doodle link to the Help desk.

# Dataset: MNIST

<http://yann.lecun.com/exdb/mnist/>

70,000 samples  
images of handwritten digits  
28x28 grayscale images  
labels are digits from 0-9

**Label**

**Images**

0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9	9	9	9	9	9

image: [wikipedia.com](http://wikipedia.com)



image: [research.fb.com](http://research.fb.com)

**Yann LeCun**

NYU Professor  
facebook Chief AI Scientist  
*Deep Learning (ConvNets)*



image: [di.ku.dk](http://di.ku.dk)

**Corinna Cortes**

Head Google Research NY  
*Support Vector Machines*

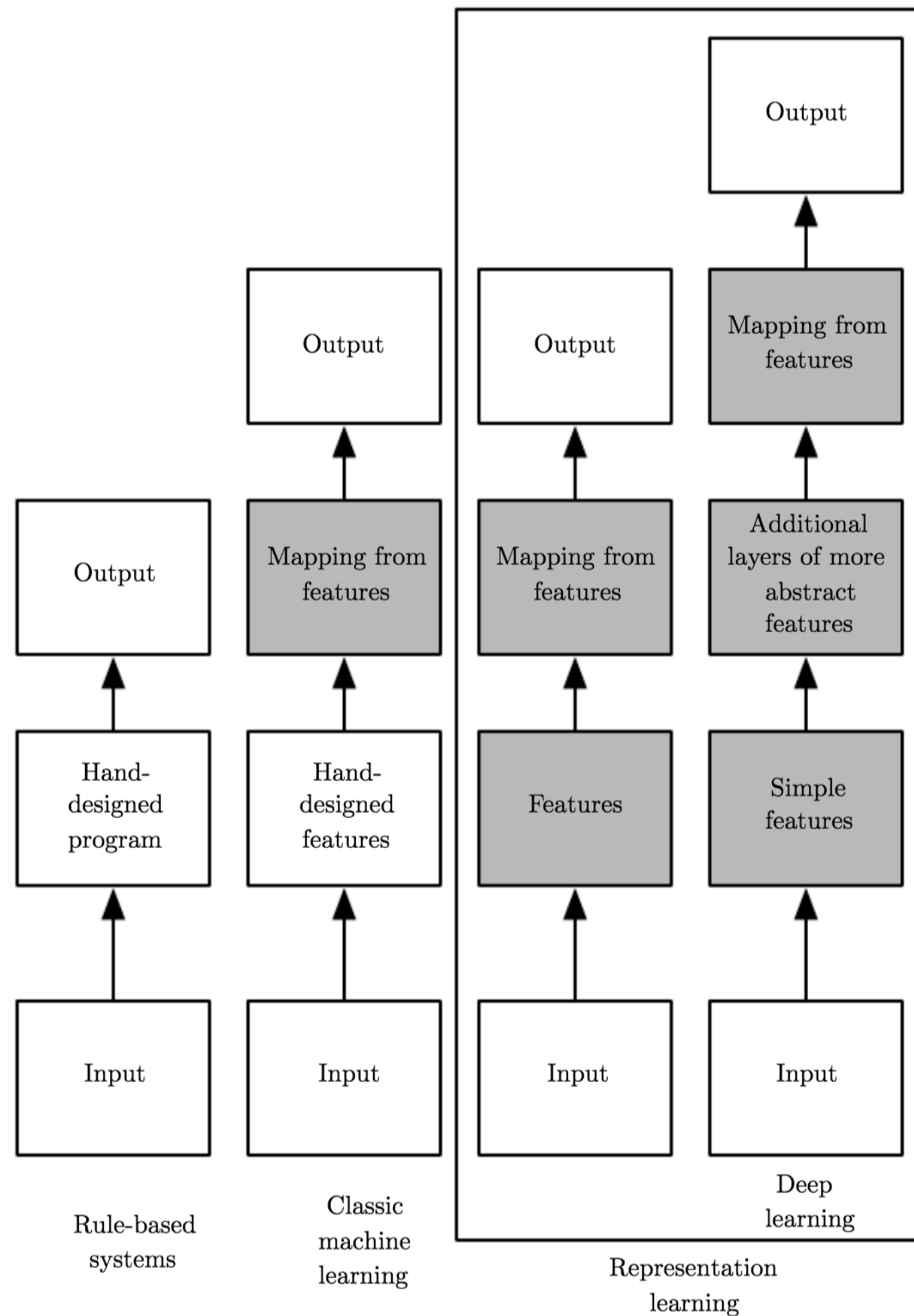


image: [microsoft.com](http://microsoft.com)

**Christopher J.C. Burges**

formerly Microsoft Research  
*Support Vector Machines*

# What is Machine Learning?



graphic: "Deep Learning" Goodfellow, Bengio, Courville

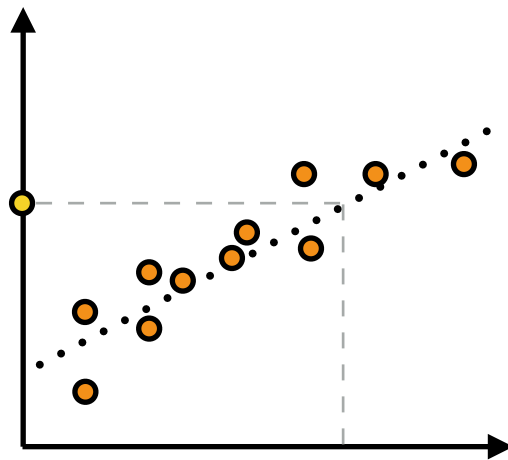


# Machine Learning Overview

Do you have labeled data?

✓  
supervised  
what kind of label?

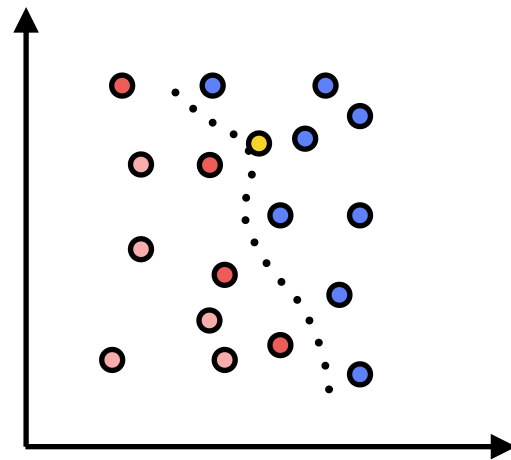
continuous  
regression



Boston Housing

predict real estate price by  
attributes of the property

discrete  
classification

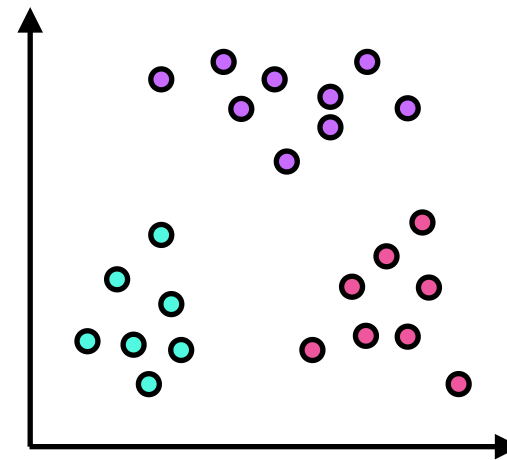


MNIST

classify handwritten  
digit as 0-9

✗  
unsupervised  
what to generate?

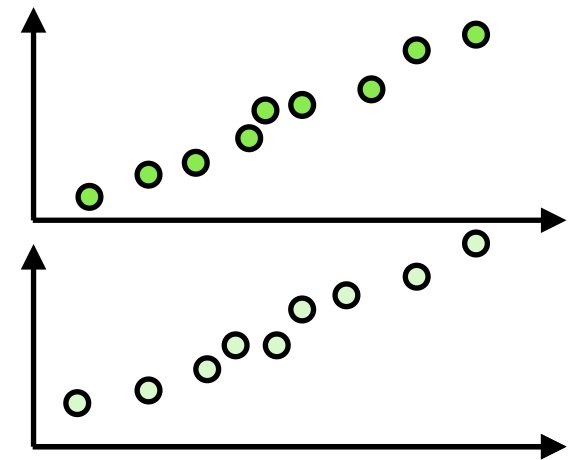
sample → label  
clustering



MNIST

cluster handwritten  
digits in 10 clusters

sample → sample  
generative



MNIST

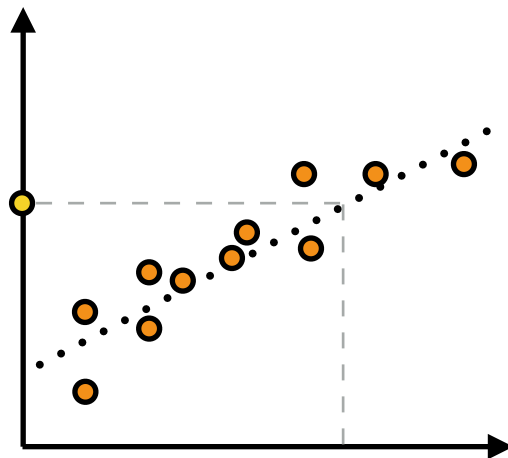
generate images of  
handwritten digits

# Machine Learning Overview

Do you have labeled data?

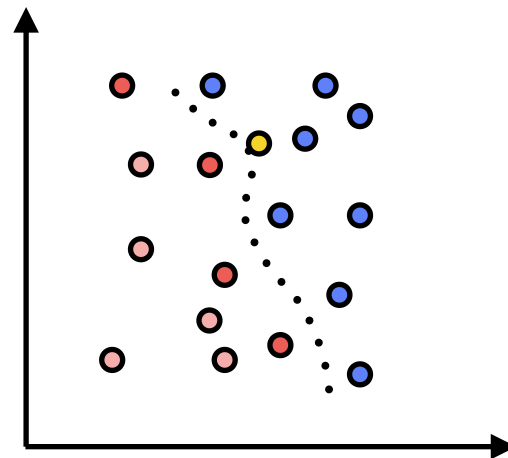
✓  
supervised  
what kind of label?

continuous  
regression



- K-nearest-neighbours
- Linear Regression
- Regression Trees
- Support Vector Regression
- Neural Networks

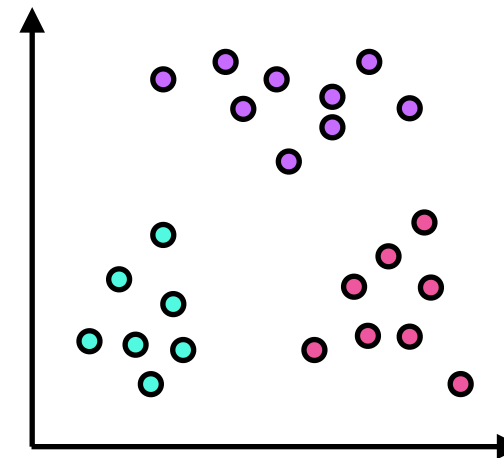
discrete  
classification



- Logistic Regression
- Support Vector Machines
- Decision Trees / Forests
- Neural Networks
- (Gaussian) Mixture Model

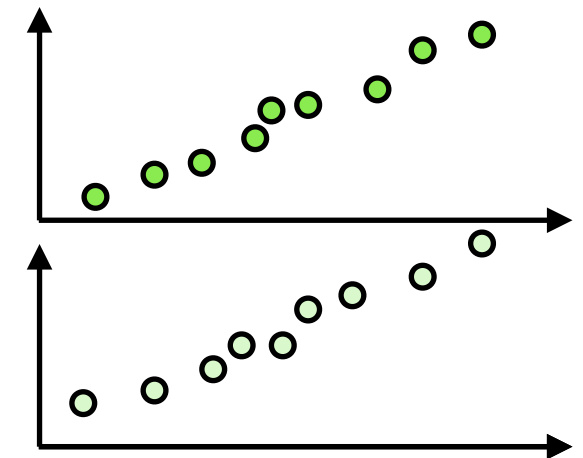
✗  
unsupervised  
what to generate?

sample → label  
clustering



- K-means
- (Gaussian) Mixture Models
- Self-organising maps\*

sample → sample  
generative



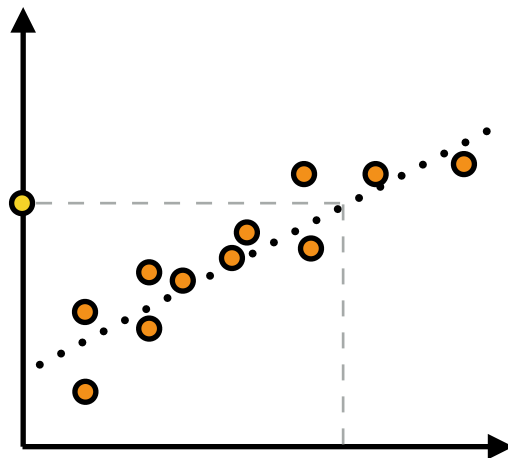
- Markov Chains
- Autoencoder\*
- Generative Adversarial Networks\*

# Machine Learning Overview

Do you have labeled data?

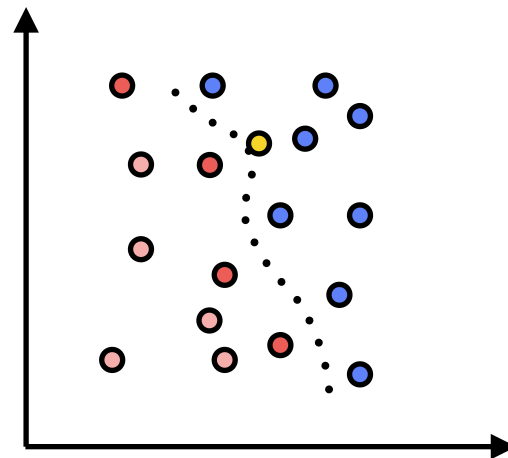
✓  
supervised  
what kind of label?

continuous  
regression



- K-nearest-neighbours
- Linear Regression
- Regression Trees
- Support Vector Regression
- Neural Networks

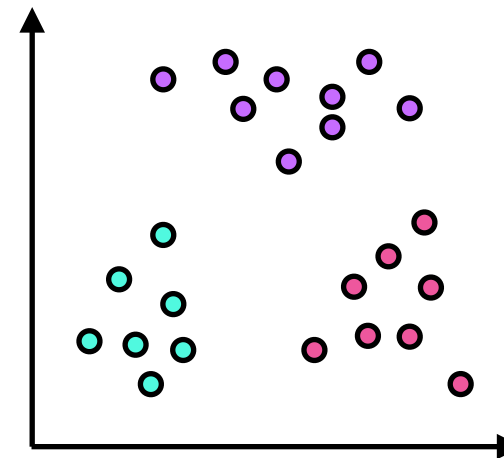
discrete  
classification



- Logistic Regression
- Support Vector Machines
- Decision Trees / Forests
- Neural Networks
- (Gaussian) Mixture Model

✗  
unsupervised  
what to generate?

sample → label  
clustering



- K-means
- (Gaussian) Mixture Models
- Self-organising maps\*

sample → sample  
generative



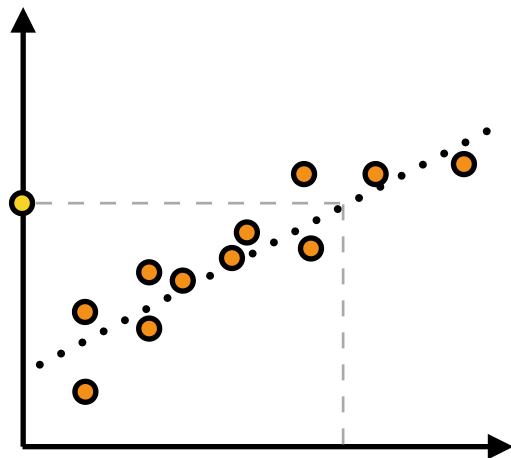
- Markov Chains
- Autoencoder\*
- Generative Adversarial Networks\*

# Machine Learning Overview

Do you have labeled data?

✓  
supervised  
what kind of label?

continuous  
regression



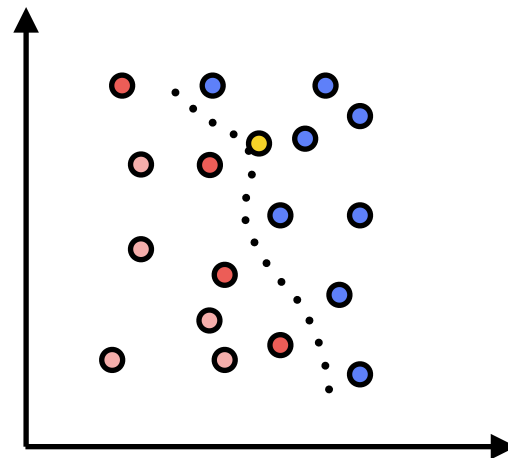
- K-nearest-neighbours

**Oct 08**

- Linear Regression

**Oct 09**

discrete  
classification



- Logistic Regression

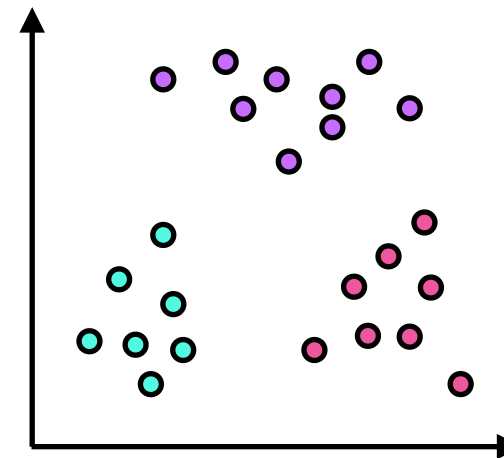
- Support Vector Machines

**Oct 09**

- (Gaussian) Mixture Model

✗  
unsupervised  
what to generate?

sample → label  
clustering



- K-means

**Oct 08**

- Self-organising maps\*

sample → sample  
generative



- Markov Chains

- Autoencoder\*

- Generative Adversarial Networks\*

# Notation

The diagram illustrates the relationship between labels, targets, features, and input/output for a single sample and all samples.

**Labels, Targets or Output:** This column contains the labels for the samples. A single label is shown as 'Y' in a box. The number of samples is denoted as 'n'.

**Features or Input:** This column contains the features for the samples. A single feature is shown as 'x' in a box. The number of features is denoted as 'm'.

**Features of a sample:** This section shows the features of a single sample, labeled 'Y'. The features are represented by handwritten digits 0 through 9, arranged in a grid. The digits are: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. The digits are arranged in a grid where the first row contains 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. The second row contains 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. The third row contains 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. The fourth row contains 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. The fifth row contains 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. The sixth row contains 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. The seventh row contains 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. The eighth row contains 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. The ninth row contains 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. The tenth row contains 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.

**Number of samples: n**

**Number of features: m**

**All labels: Y**

**features of All samples: X**

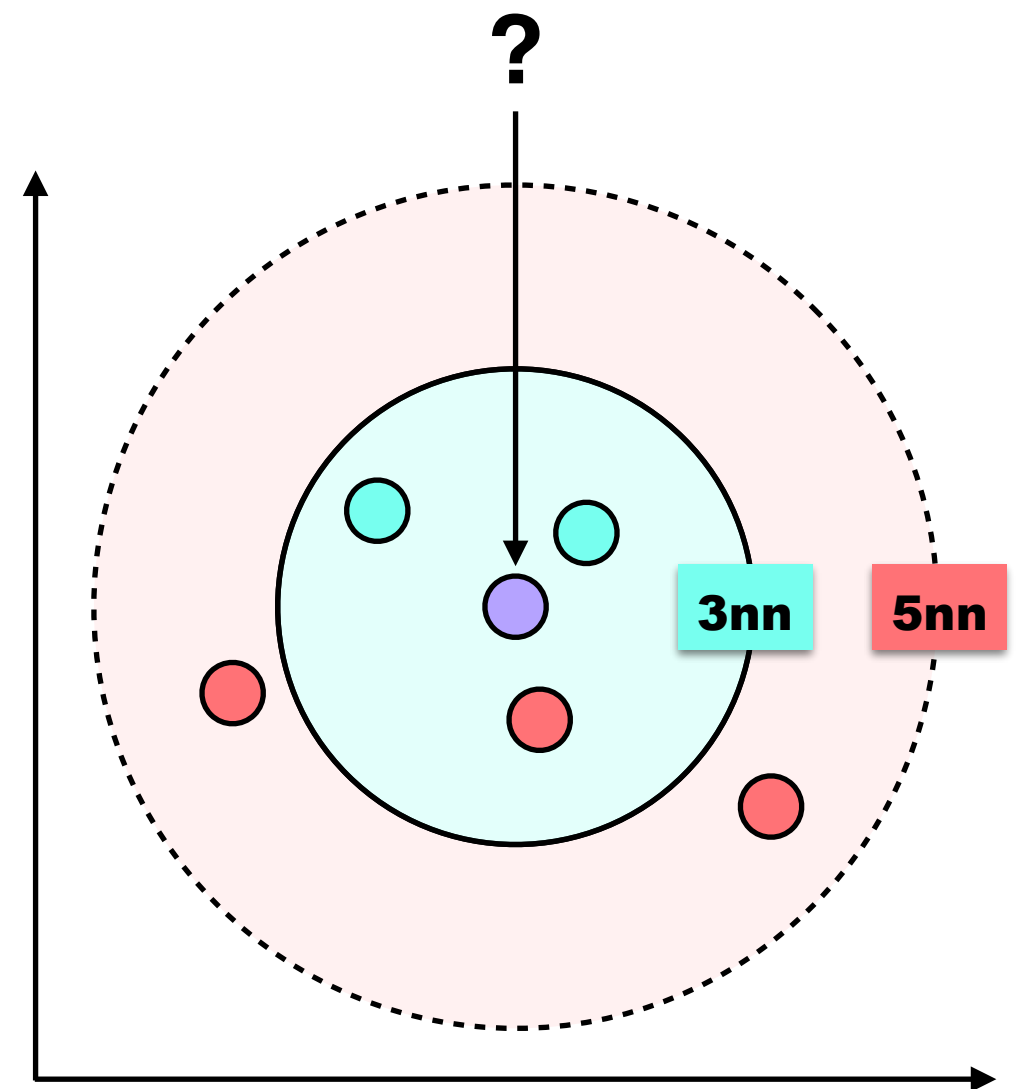
A machine learning model is a function mapping  $X$  to  $Y$ .  
 $\Theta$  is what the model “learned”.

$$f_\theta : X \rightarrow Y$$

# Classification: k-nearest neighbors (knn)

## Intuition

1. Of all neighbors, which are the  $k$  nearest to my sample?
2. What label does the majority of the  $k$  neighbors



# Classification: k-nearest neighbors (knn)

## Algorithm

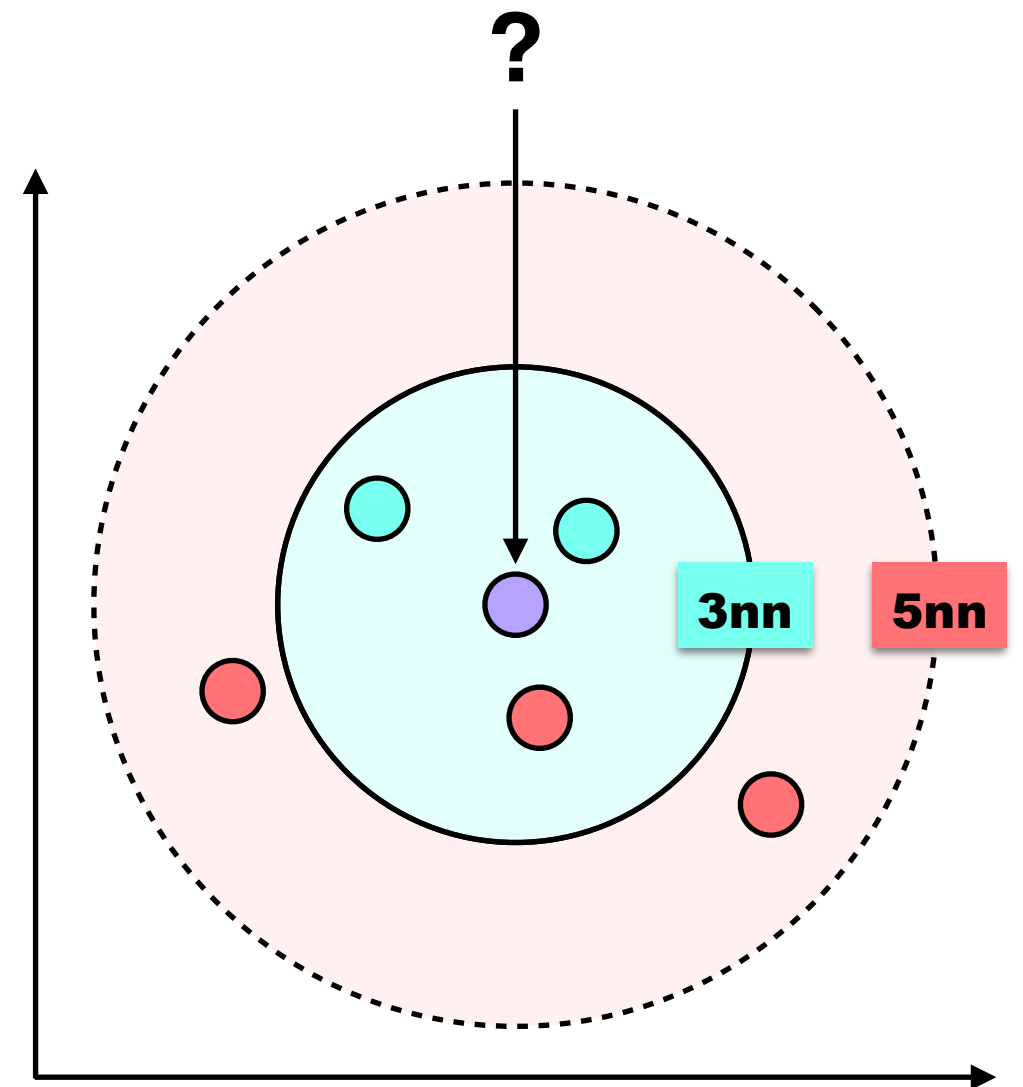
### *Input*

- dataset  $\mathbf{M}$  - matrix of samples (vector, class)
- sample  $\mathbf{v}$  to classify
- number of neighbors  $\mathbf{n}$

### *Output*

- class  $\mathbf{k}$

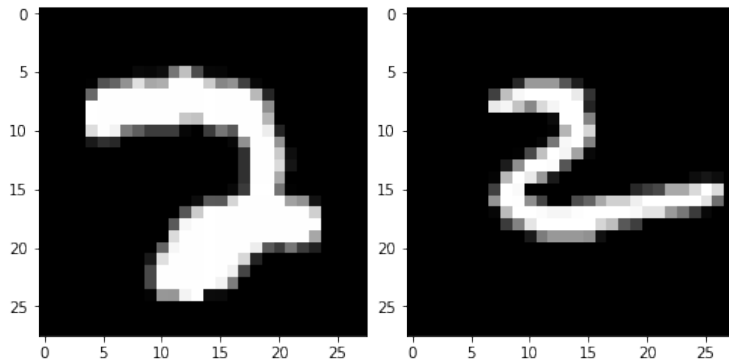
0. Calculate distance of  $\mathbf{v}$  to all samples in  $\mathbf{M}$
1. Select the  $\mathbf{n}$  samples closest to  $\mathbf{v}$
2. choose  $\mathbf{k}$  to be the majority of classes in selection



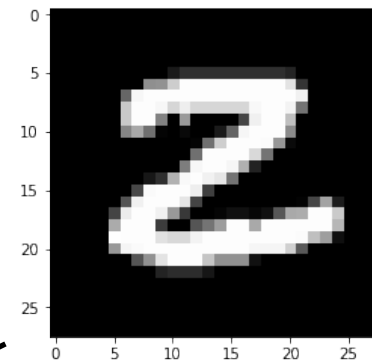
# Classification: k-nearest neighbors (knn)

Data

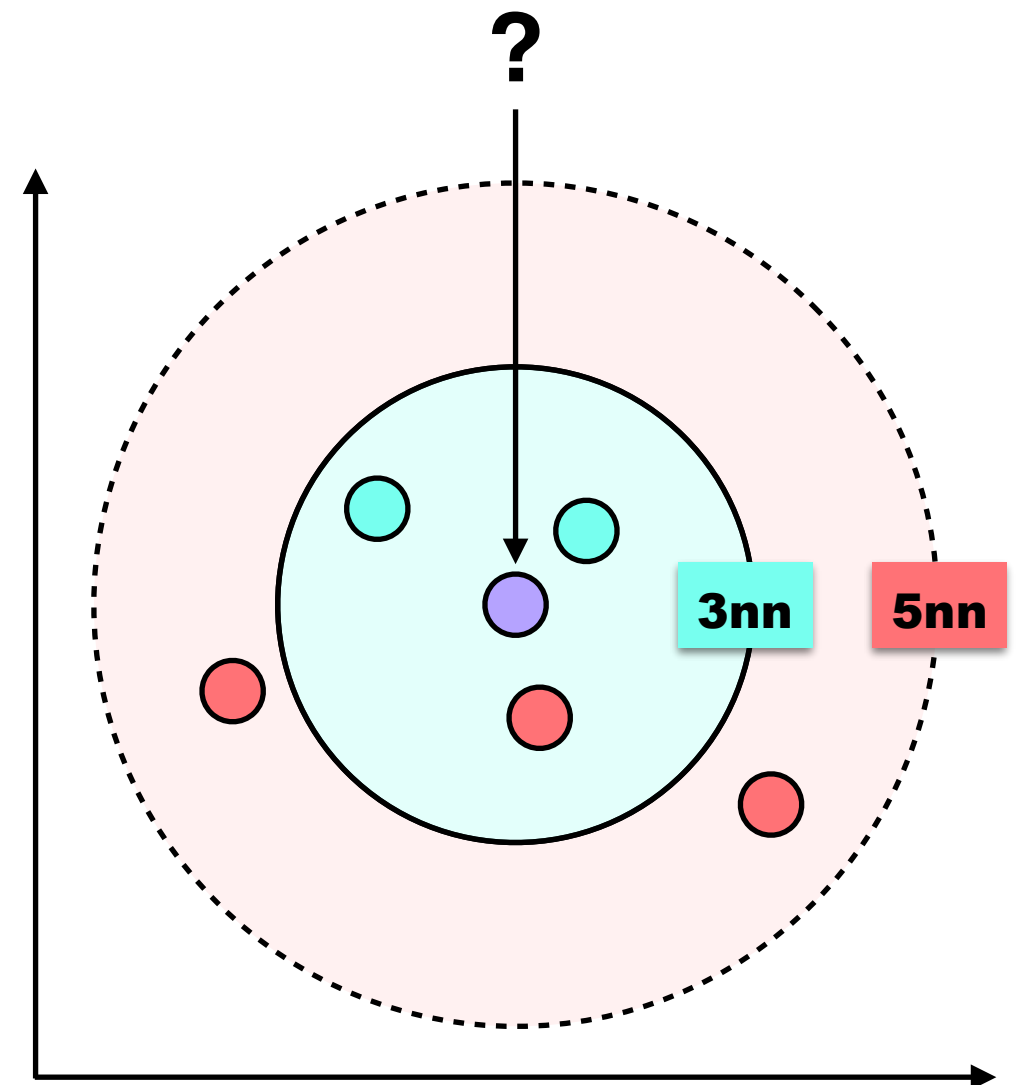
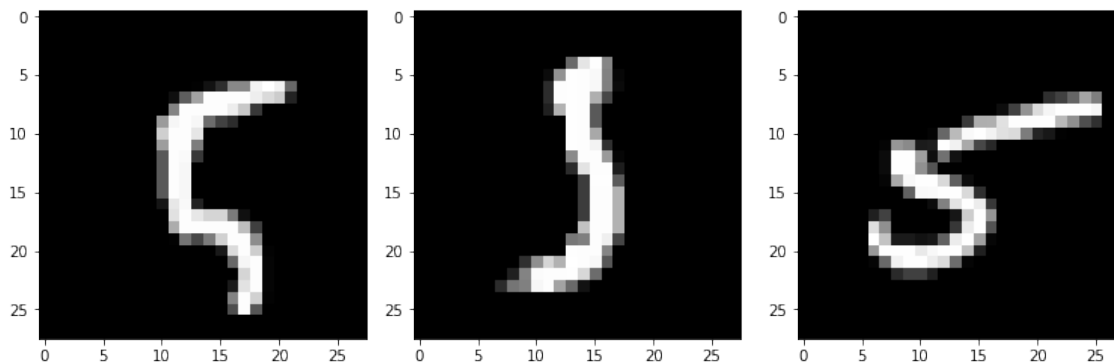
● 2



● ?



● 5



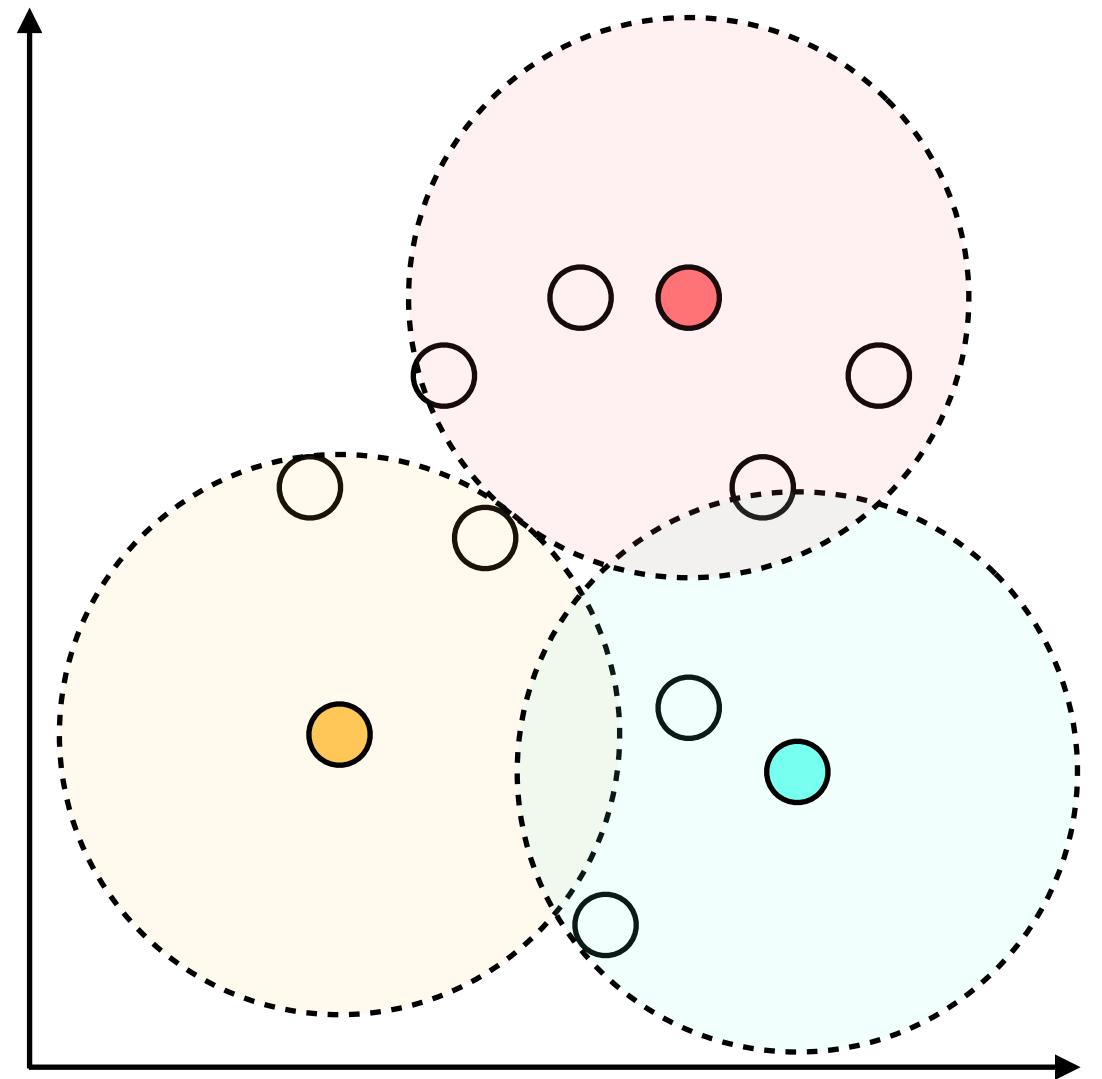


# Clustering: k-means

## Intuition

Initially, select  $k$  random samples, those are your class centroids.

1. assign each sample the class of the closest centroid.
2. determine centroid for each class with smallest distance to each all samples in class
3. if one of the centroids changed, repeat.



# Clustering: k-means

## Algorithm

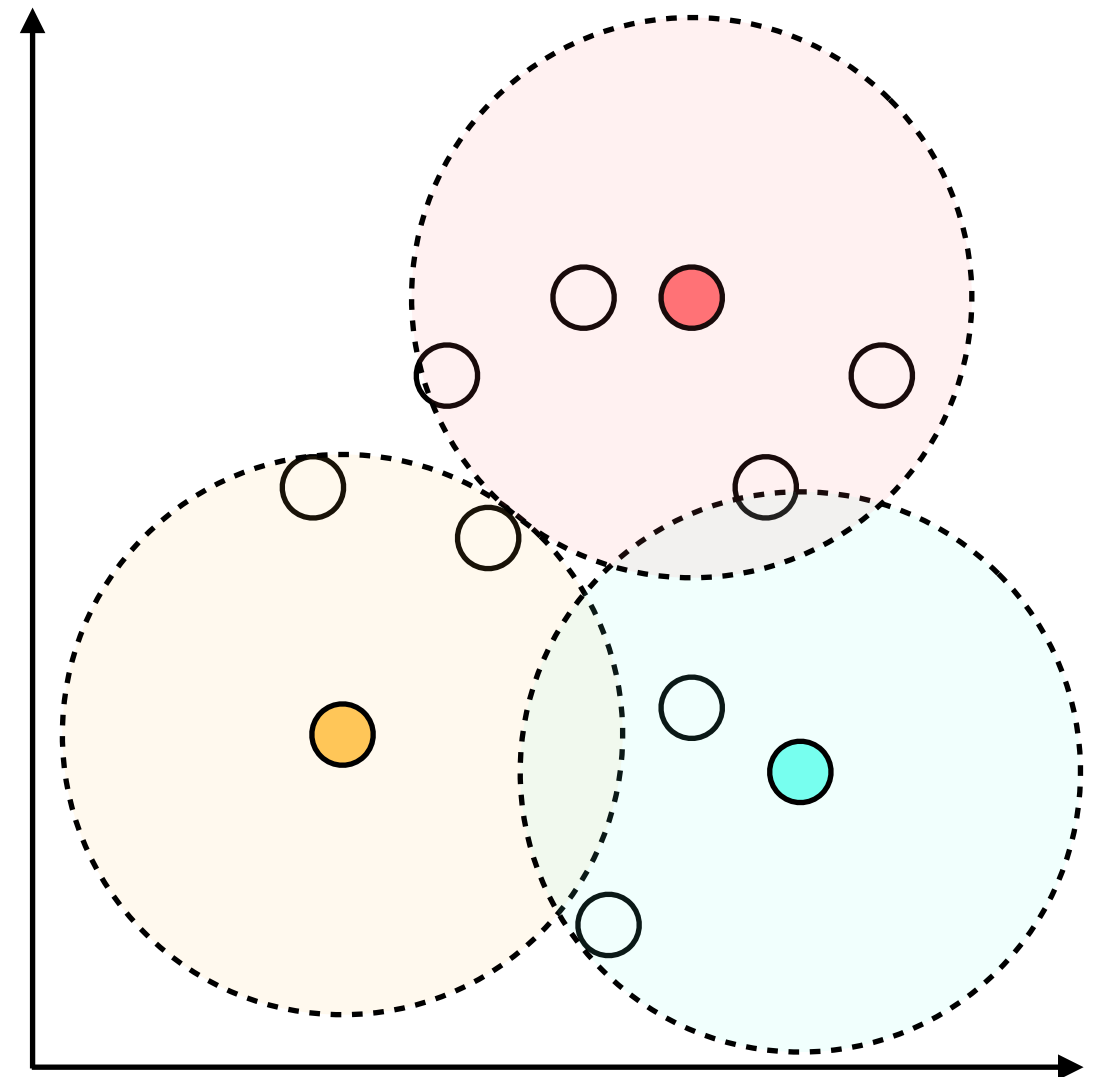
### *Input*

- dataset  $\mathbf{M}$  - matrix of samples (vector)
- number of clusters  $k$

### *Output*

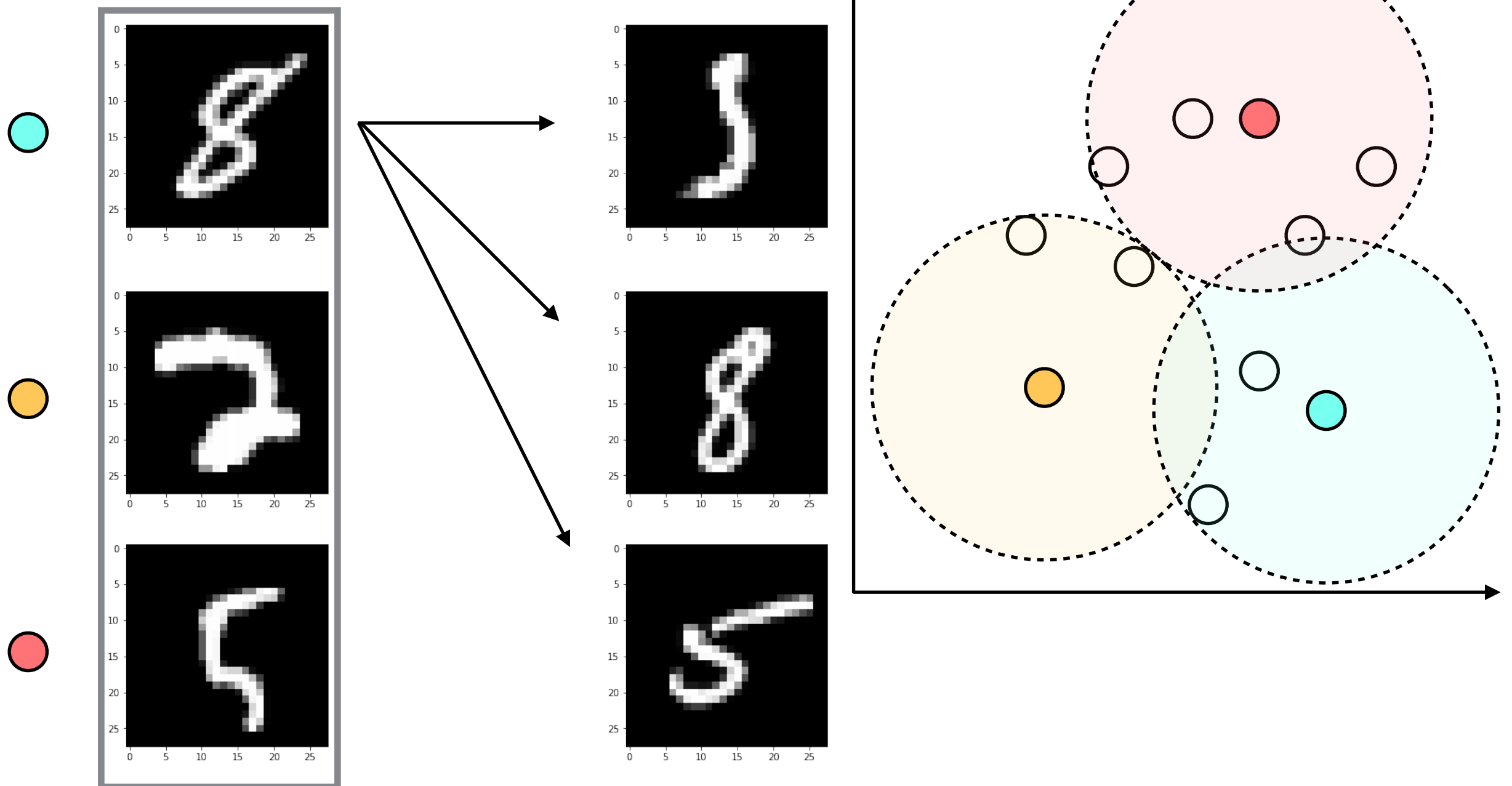
- dataset  $\mathbf{R}$  - matrix of samples (vector, class)

1. calculate all pair-wise distances in  $\mathbf{M}$
2. select  $k$  samples from  $\mathbf{M}$  at random ( $\mathbf{K}$  cluster centroids)
3. assign all samples in  $\mathbf{M}$  to the nearest sample in  $\mathbf{K}$
4. determine sample with smallest distance to all other samples in a cluster for each cluster  $\mathbf{K}'$
5. if  $\mathbf{K} \neq \mathbf{K}'$   
then set  $\mathbf{K} := \mathbf{K}'$ ; Repeat from 3.



# Clustering: k-means

Data



# Theory: Linear Algebra

Euclidean distance

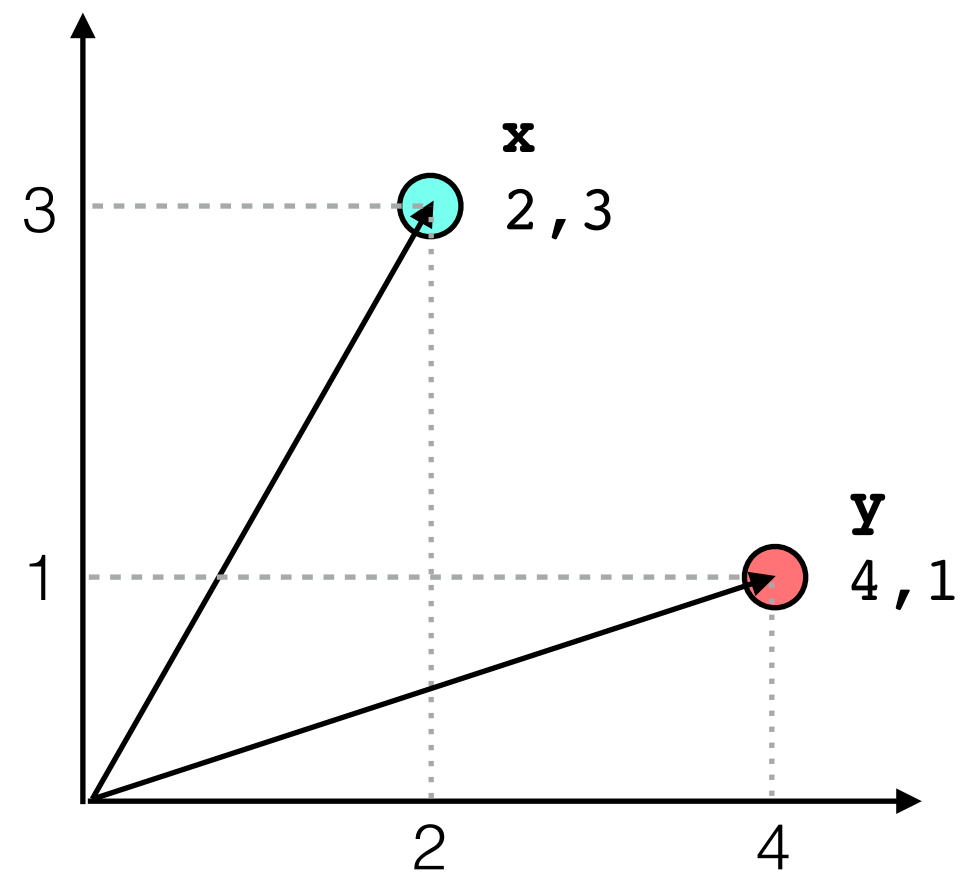
$$\text{dist}(x, y) = \|x - y\|_2 = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

Example

$$\begin{aligned} \sqrt{\sum_{i=1}^2 \left( \begin{bmatrix} 2 \\ 3 \end{bmatrix} - \begin{bmatrix} 4 \\ 1 \end{bmatrix} \right)^2} &= \sqrt{\sum_{i=1}^2 \left( \begin{bmatrix} -2 \\ 2 \end{bmatrix} \right)^2} \\ &= \sqrt{\sum_{i=1}^2 \begin{bmatrix} 4 \\ 4 \end{bmatrix}} \\ &= \sqrt{8} \end{aligned}$$

Code

```
dist = np.linalg.norm(x - y)
```



# Dataset: CIFAR-10

[cs.toronto.edu/~kriz/cifar.html](http://cs.toronto.edu/~kriz/cifar.html)

60,000 samples

images of objects

32x32 color images with labels

labels are 10 categories

airplane  
automobile  
bird  
cat  
deer  
dog  
frog  
horse  
ship  
truck

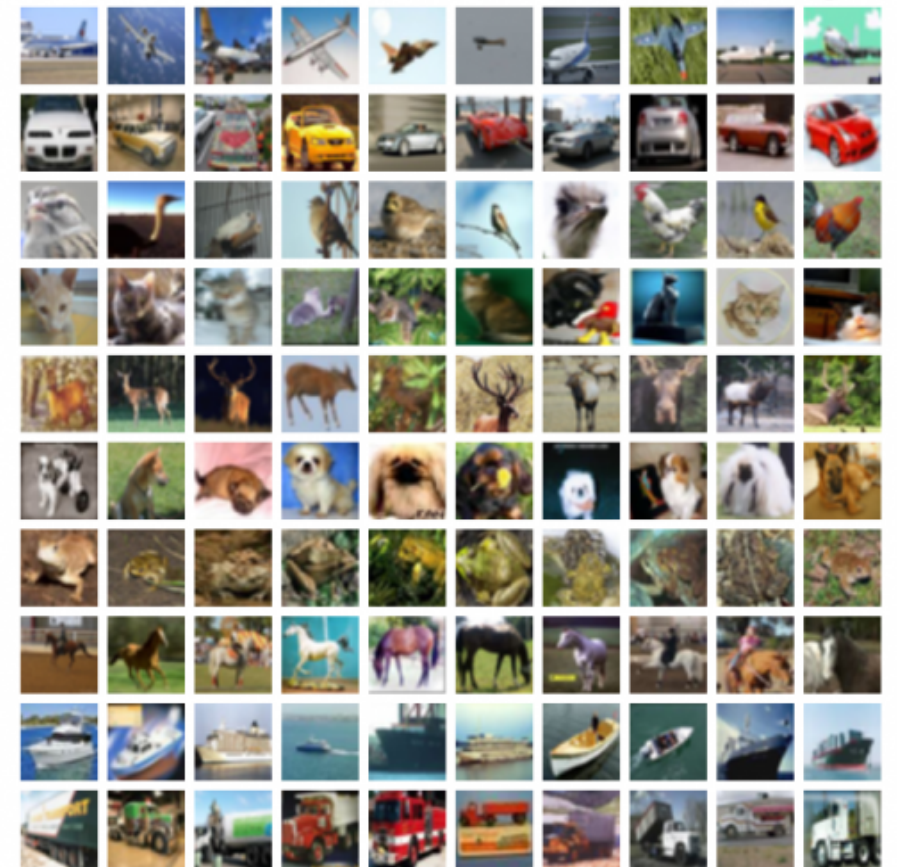


image: kaggle.com

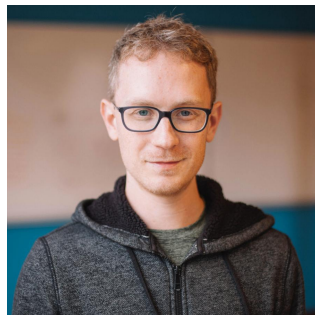


image: qz.com

**Alex Krizhevsky**

Dessa, formerly Google  
*Deep Learning (AlexNet)*



image: cs.toronto.edu

**Vinod Nair**

Yahoo Labs  
*Deep Learning*



image: thestar.com

**Geoffrey Hinton**

U Toronto Professor  
Google Brain  
*Deep Learning*