DDA4210 Advanced Machine Learning
Lecture 01 Introduction and Review

Jicong Fan

School of Data Science, CUHK-Shenzhen

January 04

Overview

Logistics

- Instructor: Jicong Fan (Session I)
- Email: fanjicong@cuhk.edu.cn
- Office: Daoyuan Building 502a
- Office hours: Monday 3:30-4:30pm
- Mixed teaching mode:
  - onsite: Zhi Xin Bldg 109 (after Lunar new year)
  - online: ZOOM 214 182 2367

Logistics

- Instructor: Jicong Fan (Session I)
- Email: fanjicong@cuhk.edu.cn
- Office: Daoyuan Building 502a
- Office hours: Monday 3:30-4:30pm
- Mixed teaching mode:
  - onsite: Zhi Xin Bldg 109 (after Lunar new year)
  - online: ZOOM 214 182 2367
- Teaching assistants:
  - Dong Qiao 221019079@link.cuhk.edu.cn
  - Fangchen Yu 220019040@link.cuhk.edu.cn
  - Zhengyang Tang 222010059@link.cuhk.edu.cn
  - Ziwei Zhu 221049028@link.cuhk.edu.cn
- Course website: `https://tongxin.me/DDA4210/`

Evaluation

- Homework (29%)
  - Three assignments (tri-weekly)
  - Involves theory, analysis, and computation
- Two course projects (40%)
  - Python programming for advanced machine learning
  - Work individually or in a team (<=4 members)
  - Project 1 (15%): a competition (partially designated)
    - evaluation: ranking (70%)+ code & report (30%)
  - Project 2 (25%): determined by yourself
    - evaluation: report (25%)+ presentation
      (75% $=$ 15%peer $+$ 25%TA $+$ 35%instructor)
- Final exam (30%)
  - Single-choice questions, True or False, and other questions

Evaluation

- Homework (29%)
  - Three assignments (tri-weekly)
  - Involves theory, analysis, and computation
- Two course projects (40%)
  - Python programming for advanced machine learning
  - Work individually or in a team (<=4 members)
  - Project 1 (15%): a competition (partially designated)
    - evaluation: ranking (70%)+ code & report (30%)
  - Project 2 (25%): determined by yourself
    - evaluation: report (25%)+ presentation
      (75% $=$ 15%peer $+$ 25%TA $+$ 35%instructor)
- Final exam (30%)
  - Single-choice questions, True or False, and other questions
- Participation in Course&Teaching Evaluation (CTE) (1%)
  - Your feedback and evaluation are very important!
  - This is a new course and could be improved according to your feedback and evaluation.

Some remarks

- Plagiarism violates the university policy of “Academic Integrity"
  - Plagiarism in homework assignments, course projects, and final exam will be dealt with severity.
  - For example, assignments with plagiarism will be graded as zero.
  - Repeated plagiarism will lead to an "F" for the entire course.
- Attendance requirement
  - Attending lectures and tutorials online or onsite are highly encouraged.
  - Please answer or raise questions actively.
  - Let the instructor/TAs be able to recognize you as a student in the class.

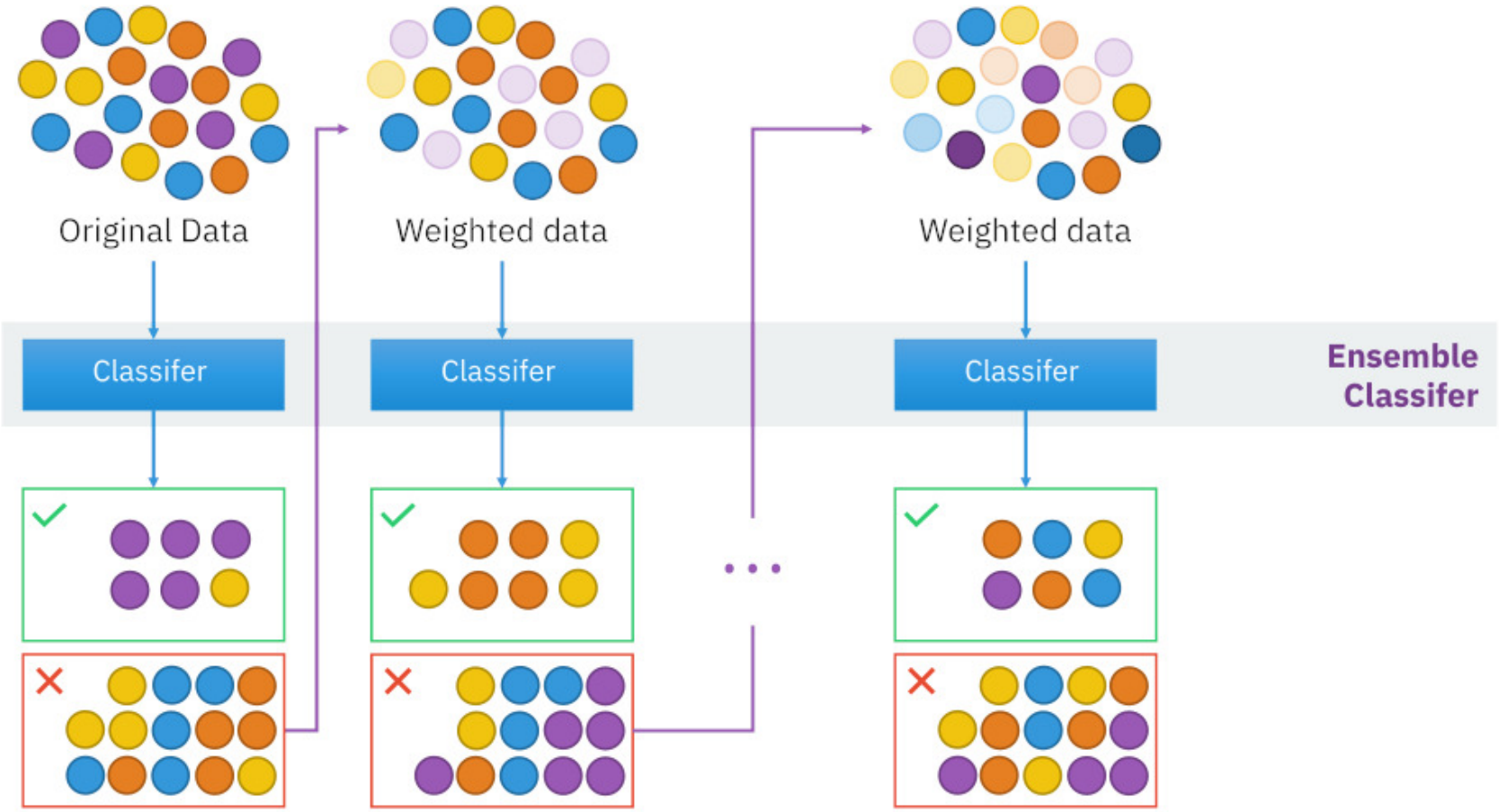
Syllabus

1. Review of basic machine learning methods
2. Advanced ensemble learning
3. Learning theory
4. Advanced applications: recommendation and search
5. Spectral clustering and semi-supervised learning
6. Graph neural networks
7. Nonlinear dimensionality reduction and data visualization
8. Generative models (VAE, GAN, diffusion model)
9. Causal machine learning
10. Privacy in machine learning
11. Safety and fairness
12. Interpretability and explainability
13. Adversarial machine learning, robustness of neural networks
14. Course project presentation and review

Basic machine learning methods

- Linear regression and classification
- K-nearest neighbor method
- Decision tree, bagging, and random forest
- Support vector machine
- Neural networks (MLP, CNN, and RNN)
- K-means and Gaussian mixture models
- Principal component analysis

Advanced Machine Learning: Boosting



Boosting is an ensemble meta-algorithm for primarily reducing bias, and also variance in supervised learning, and a family of machine learning algorithms that convert weak learners to strong ones.—-Wikipedia

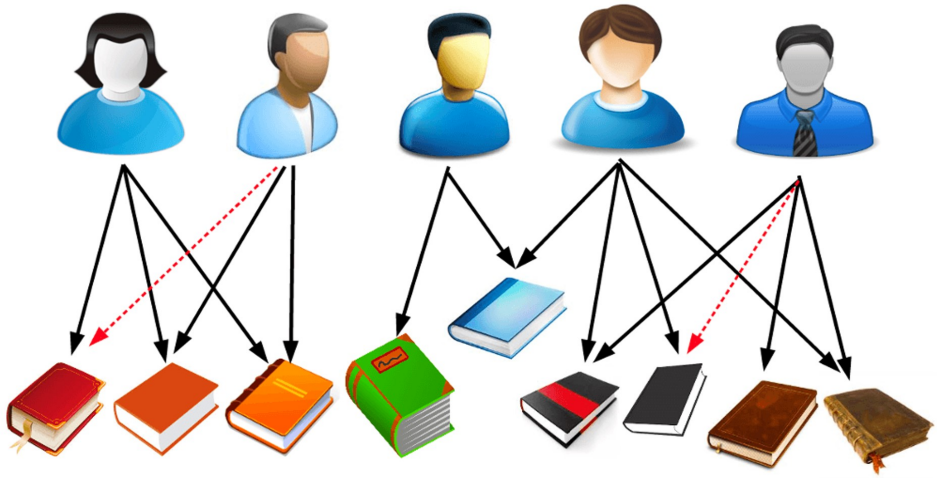
Advanced Machine Learning: Learning Theory

Machine Learning Theory[1]

- Also known as *Computational Learning Theory*
- Aims to understand the fundamental principles of learning as a computational process and combines tools from Computer Science and Statistics
  - Creating mathematical models that capture key aspects of machine learning, in which one can analyze the inherent ease or difficulty of different types of learning problems.
  - Proving guarantees for algorithms (under what conditions will they succeed, how much data and computation time is needed) and developing machine learning algorithms that provably meet desired criteria.
  - Mathematically analyzing general issues, such as: "When can one be confident about predictions made from limited data?", "What kinds of methods can learn even in the presence of large quantities of distracting information?", etc.

[1] https://www.cs.cmu.edu/~avrim/Talks/mlt.pdf

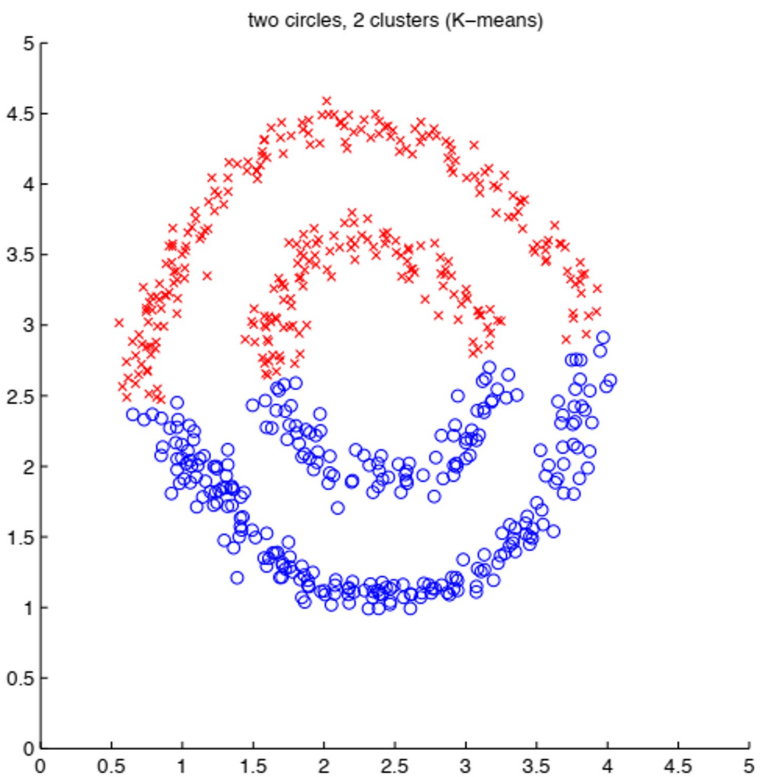
Advanced Machine Learning: Recommendation System



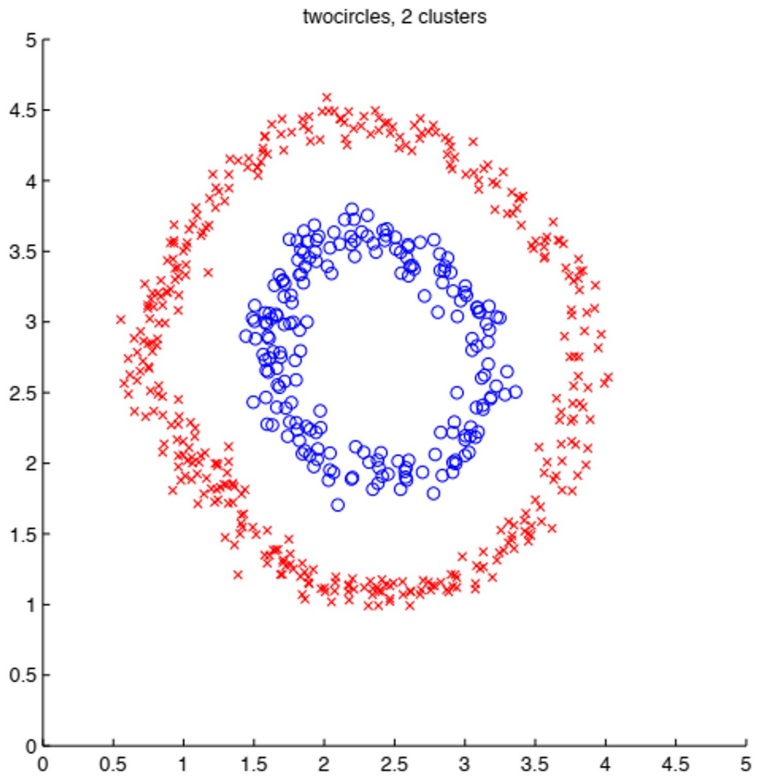
- Collaborative filtering methods
- Content-based methods
- Hybrid methods

Advanced Machine Learning: Spectral Clustering

K-means

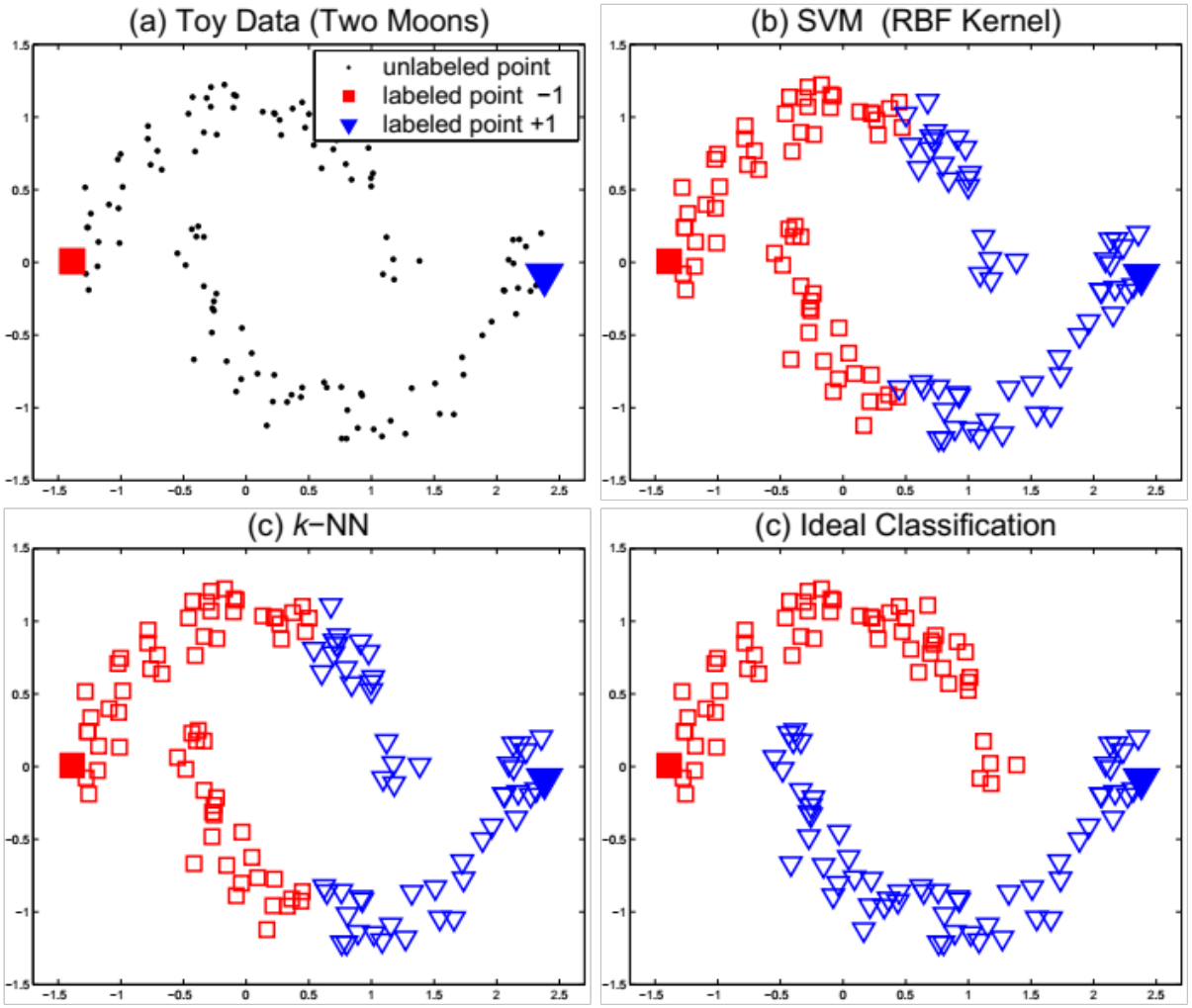


Spectral clustering



Why Semi-Supervised Learning?

Classification on the two moons pattern [Zhou et al. 04]:
(a) two labeled points; (b) SVM with an RBF kernel; (c) k-NN with k = 1.

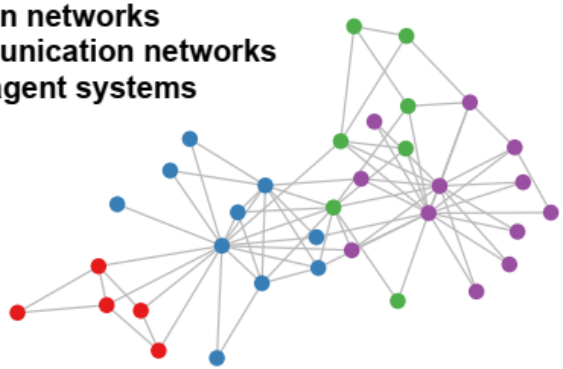


Advanced Machine Learning: Graph Neural Networks

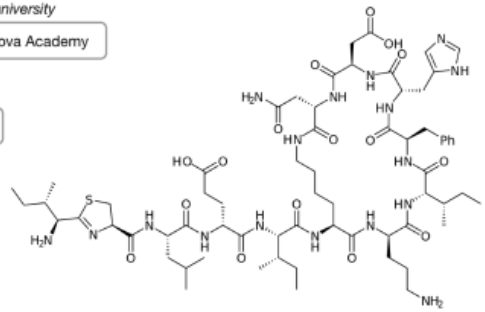
Graph-Structured data cannot be well handled by conventional neural networks!

A lot of real-world data does not "live" on grids

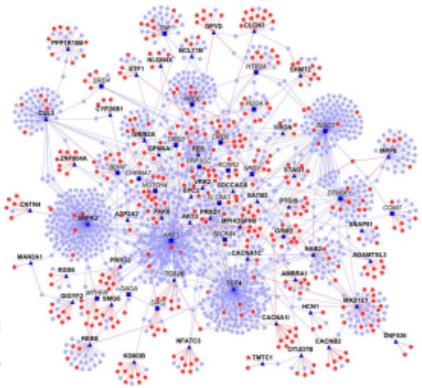
Social networks
Citation networks
Communication networks
Multi-agent systems



Knowledge graphs



Molecules



Protein interaction networks

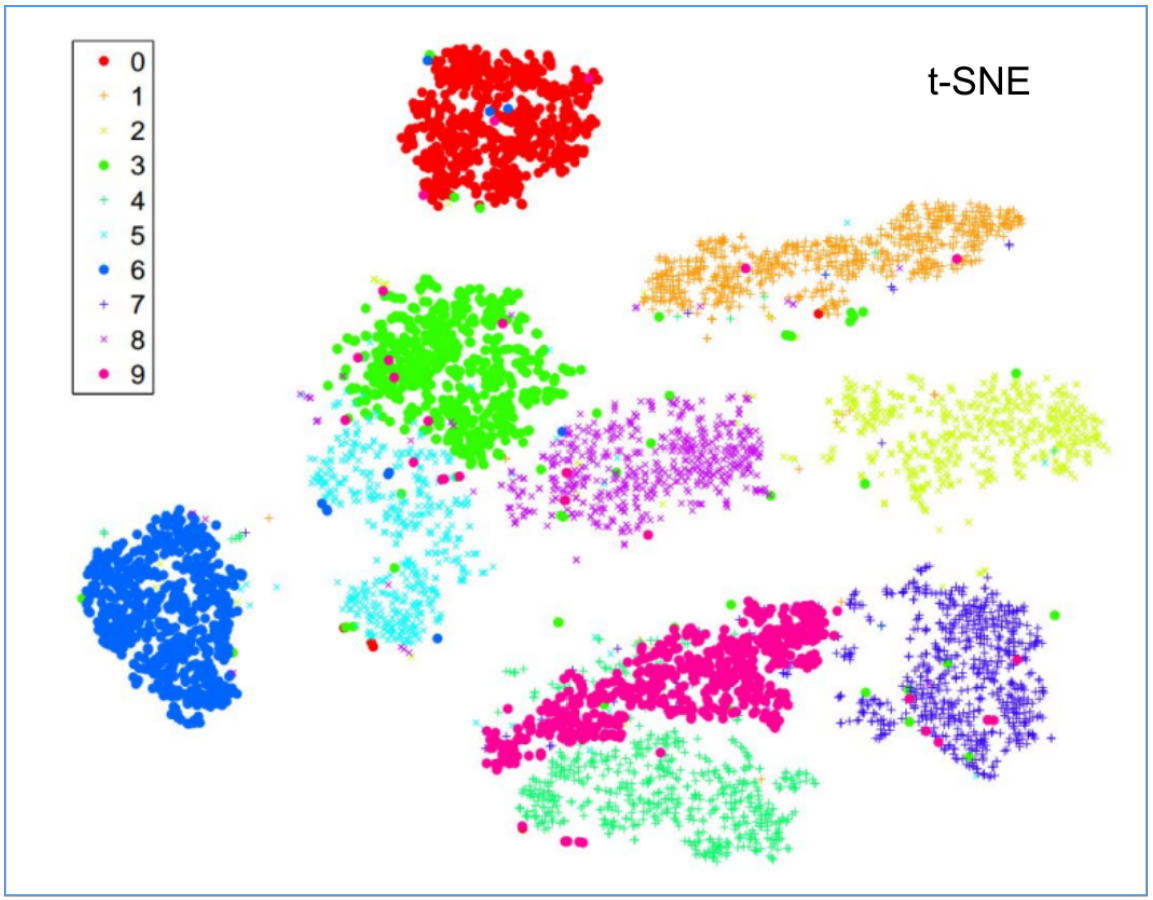
Road maps

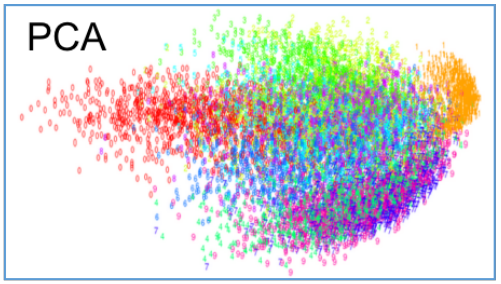
Standard deep learning architectures like CNNs and RNNs don't work here!

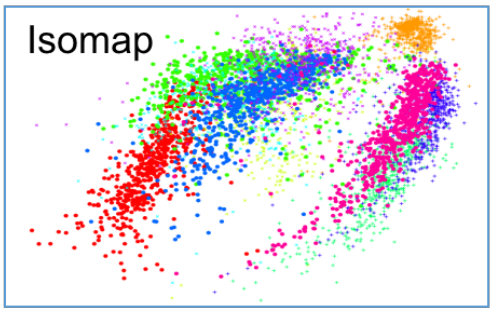
The image is from Thomas Kipf.

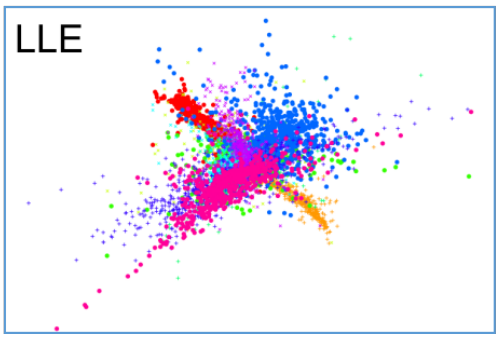
Advanced Machine Learning: NLDR

NLDR: nonlinear Dimensionality reduction
Example: visualizing MNIST handwritten digits (10 classes)





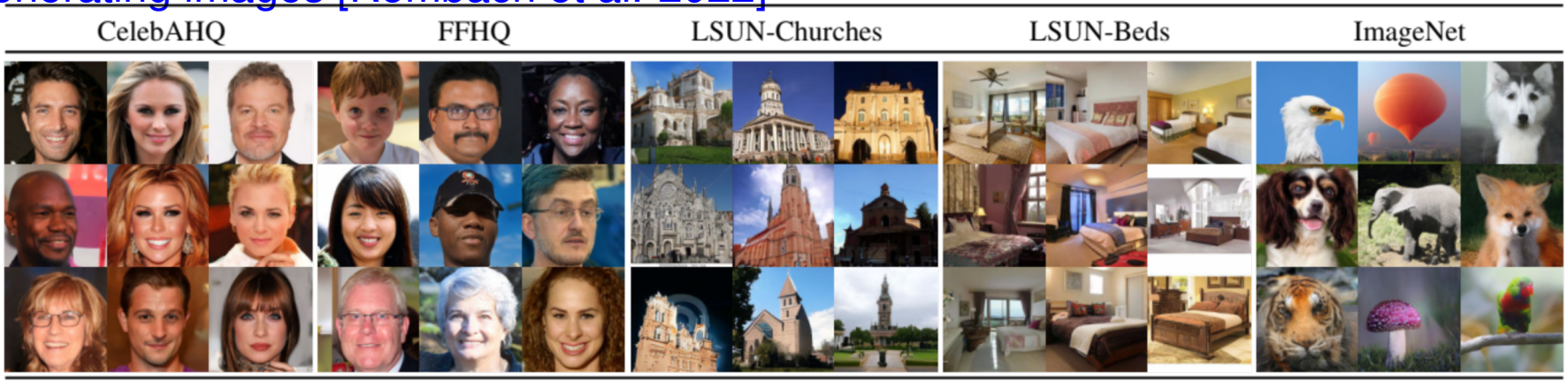




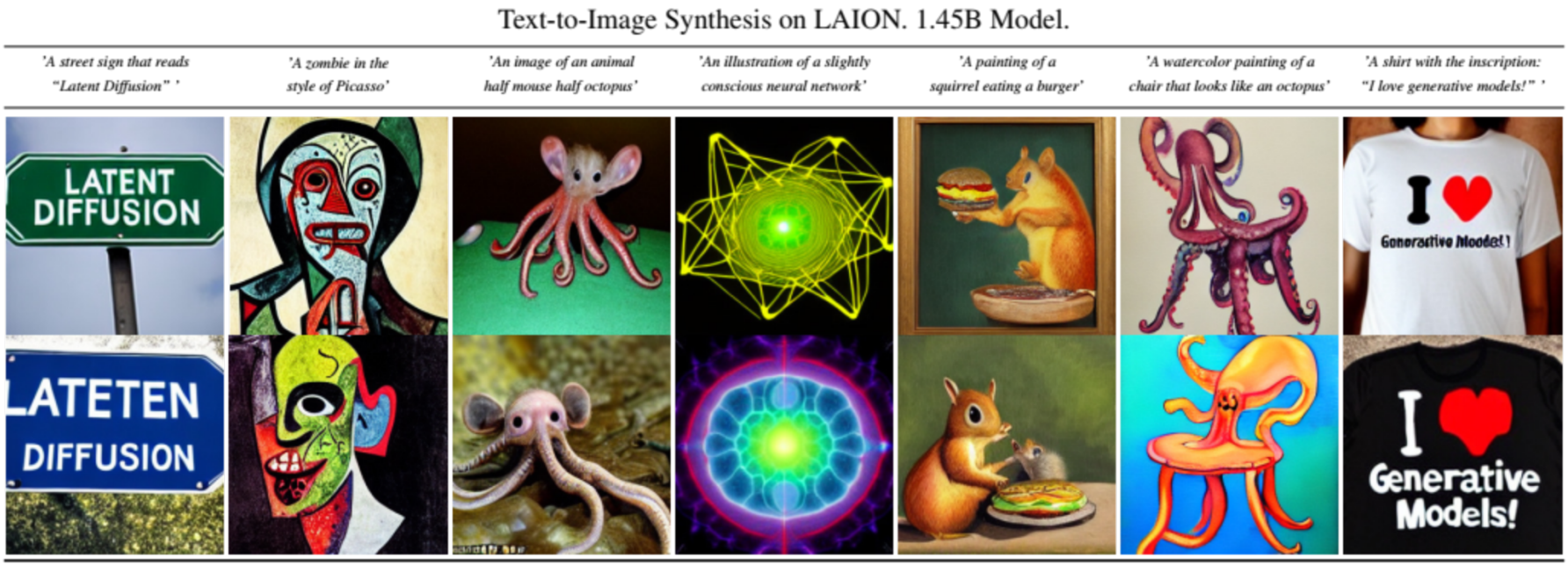
Advanced Machine Learning: Generative Models

Use the model trained on training data to generate new data

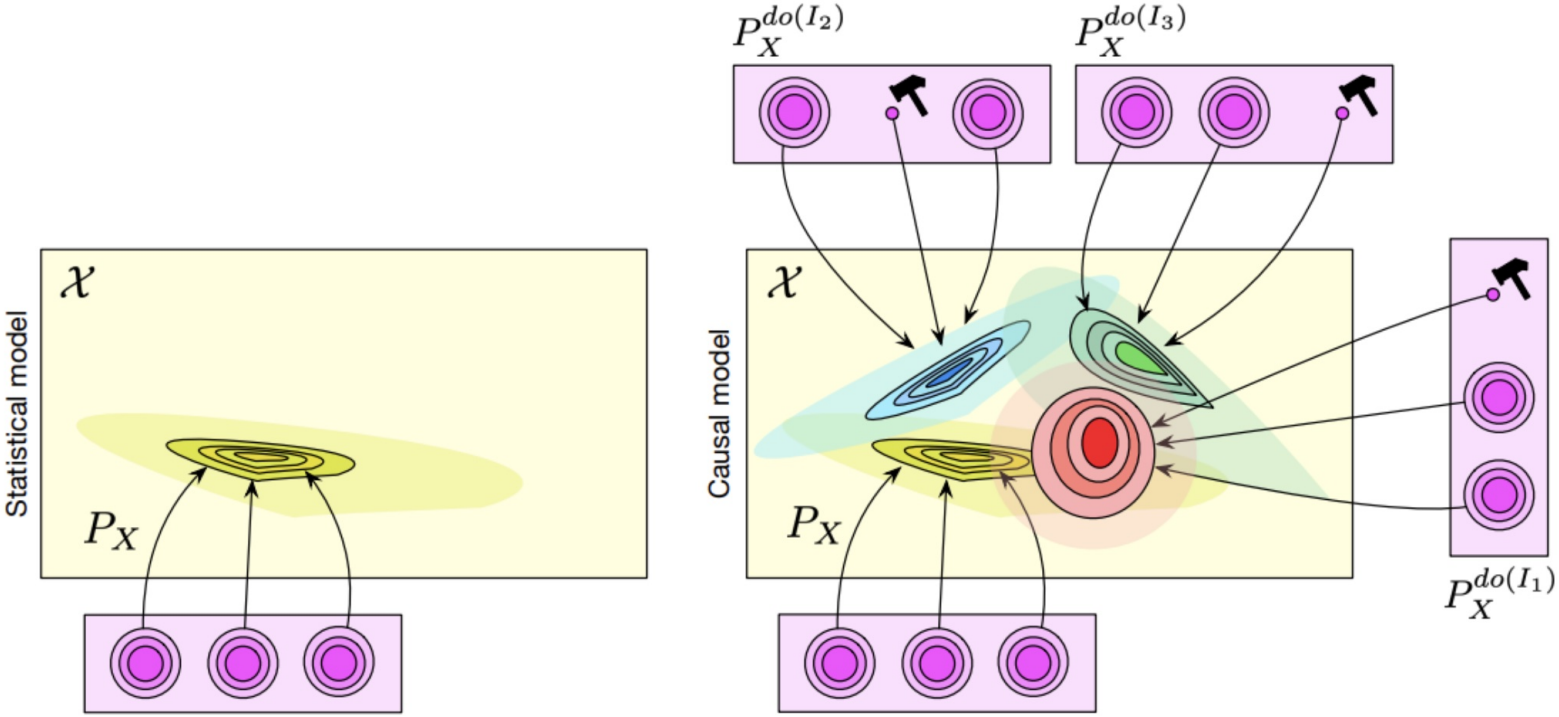
Generating images [Rombach et al. 2022]



Text to image [Rombach et al. 2022]



Understanding and generalization beyond the training distribution [Schölkopf et al. 2021]

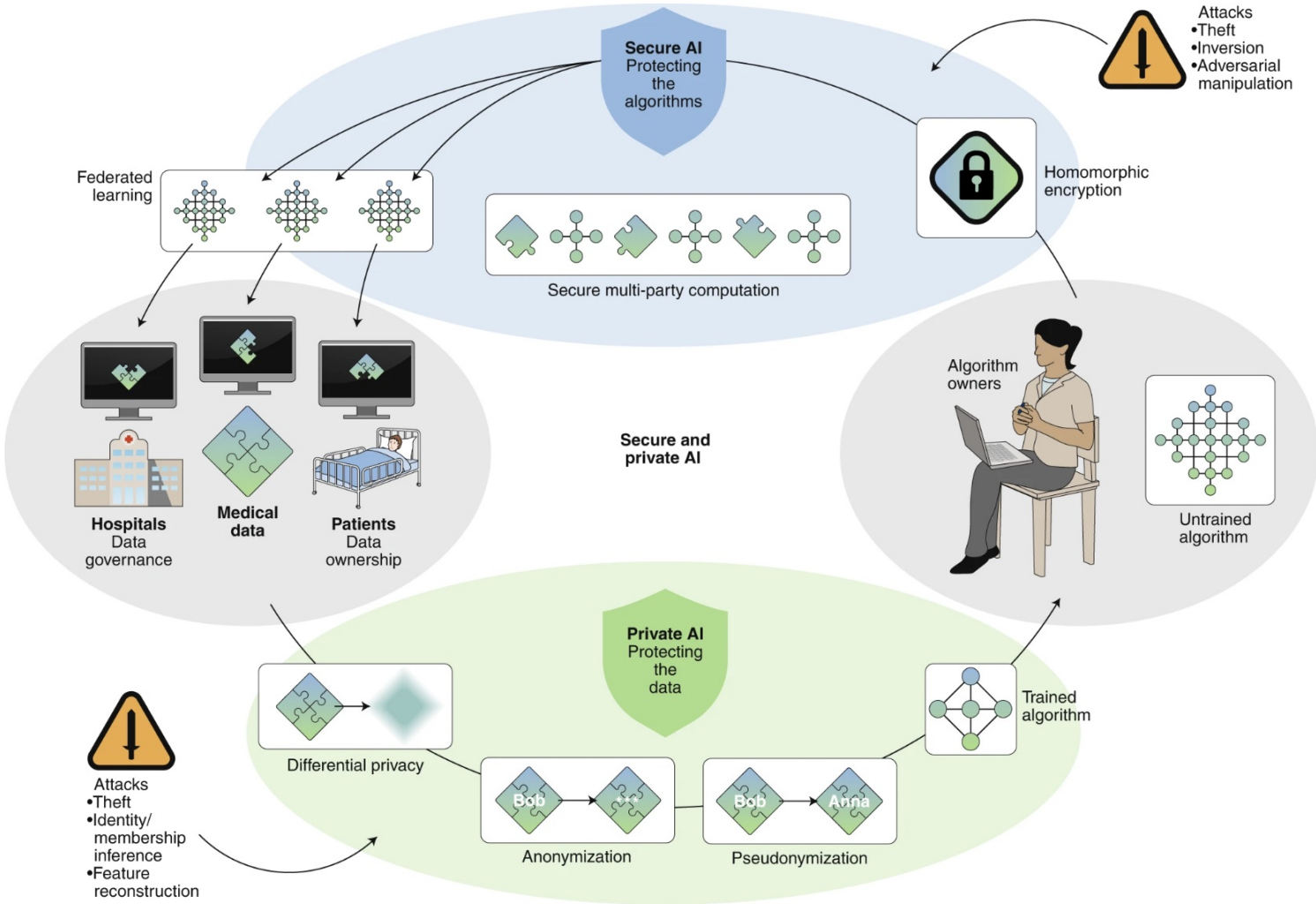


Level of modelling in physical systems

Model	Predict in i.i.d. setting	Predict under distr. shift/intervention	Answer counter-factual questions	Obtain physical insight	Learn from data
Mechanistic/physical	yes	yes	yes	yes	?
Structural causal	yes	yes	yes	?	?
Causal graphical	yes	yes	no	?	?

Advanced Machine Learning: Privacy and Safety

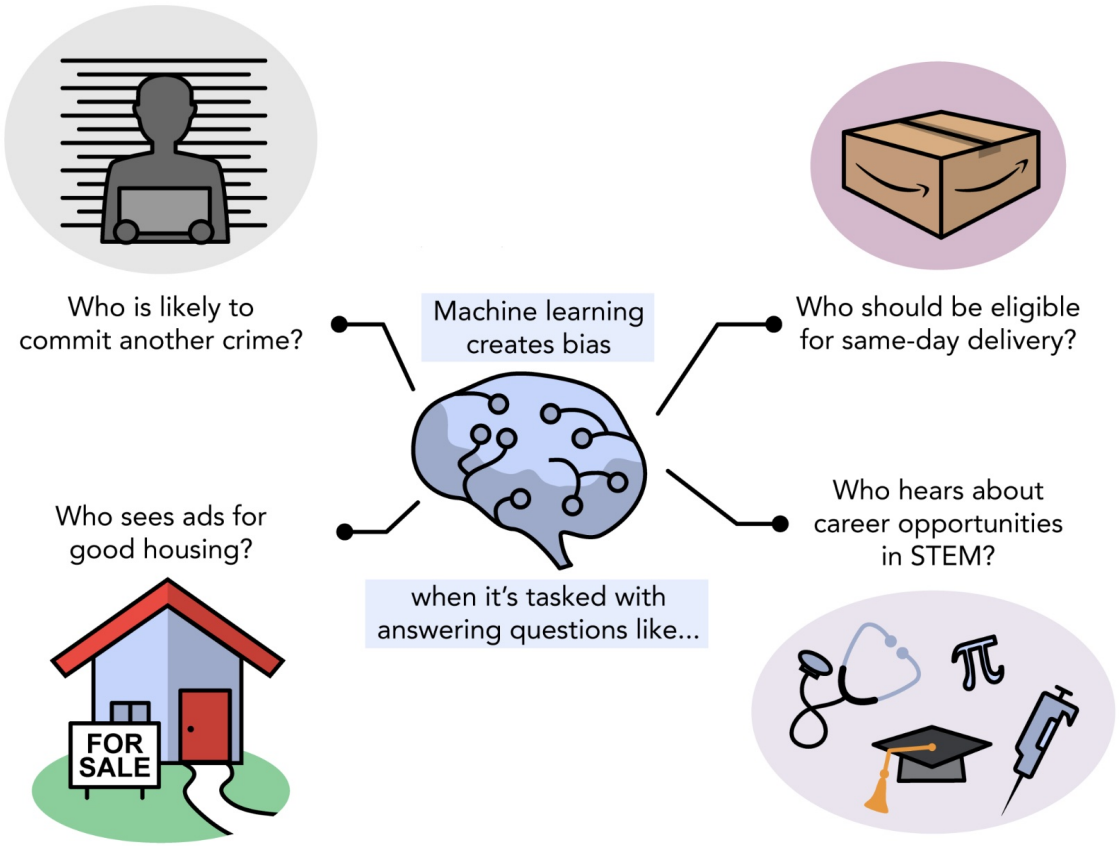
Schematic overview of the relationships and interactions between data, algorithms, actors and techniques in the field of secure and private AI [Kaissis et al. 2020]:



Advanced Machine Learning: Fairness

Where does the unfairness in machine learning algorithms come from?
How can we address the unfairness?

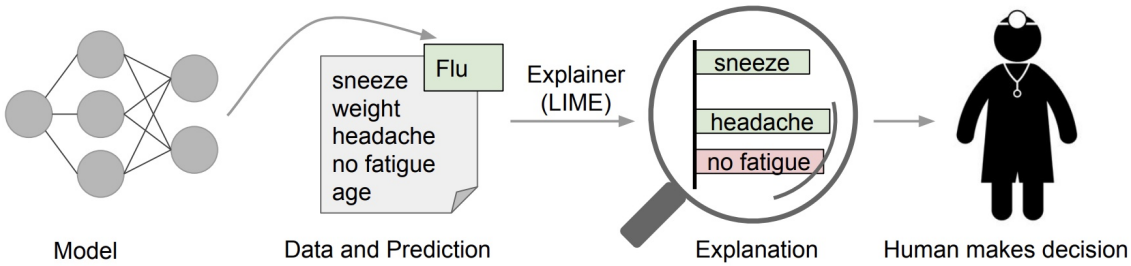
Examples of how bias in machine learning can affect our daily lives [Grabski et al. 2020]:



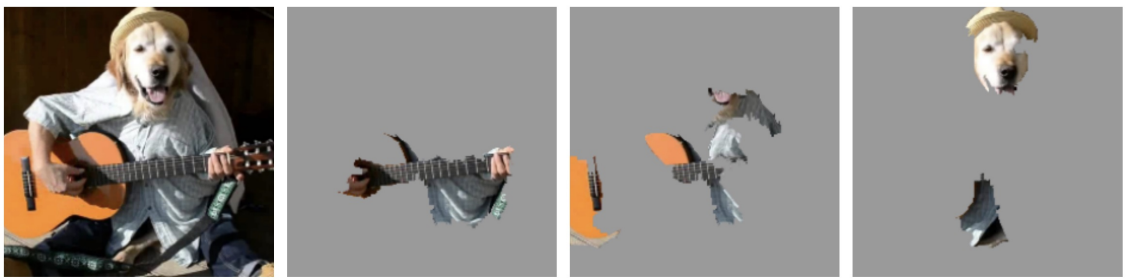
Advanced Machine Learning: Interpretability

Understanding the reasons behind decisions made by black-box machine learning models

Explaining individual outputs of a model that predicts that a patient has the flu [Ribeiro et al. 2016]:



Explaining an image classification prediction made by Google's Inception neural network [Ribeiro et al. 2016]:



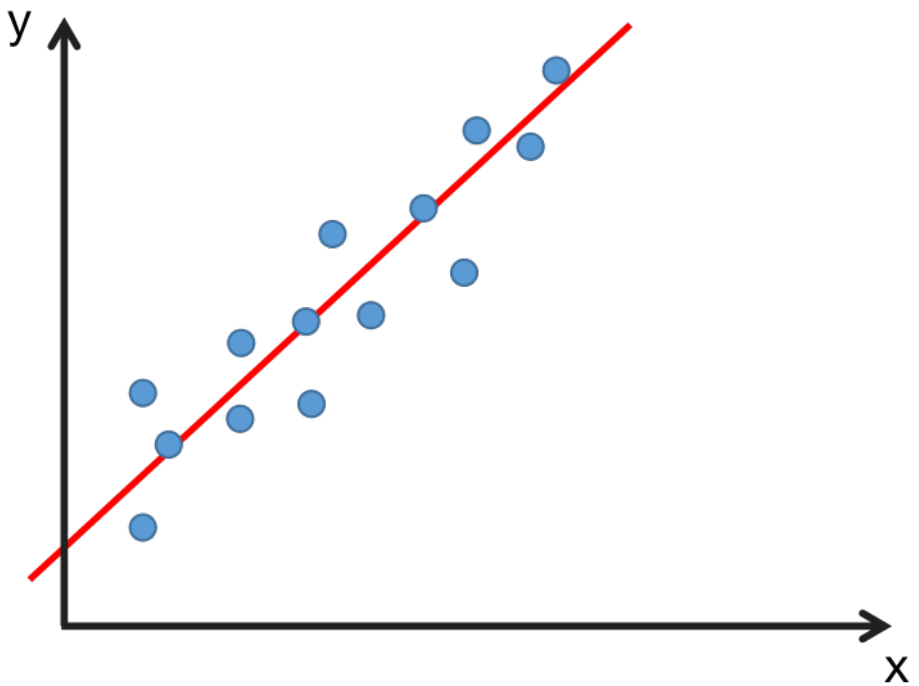
(a) Original Image (b) Explaining *Electric guitar* (c) Explaining *Acoustic guitar* (d) Explaining *Labrador*

Review for basic machine learning methods

- Linear regression and classification
- K-nearest neighbor method
- Decision tree, bagging, and random forest
- Support vector machine
- Neural networks (MLP, CNN, and RNN)
- K-means and Gaussian mixture models
- Principal component analysis

Review: Linear Regression

- Training data: $\mathcal{D} = \{(\mathbf{x}_1, \mathbf{y}_1), (\mathbf{x}_2, \mathbf{y}_2), \ldots (\mathbf{x}_N, \mathbf{y}_N)\}$
  - $\mathbf{x}_i \in \mathbb{R}^D$, $\mathbf{y}_i \in \mathbb{R}^K$, $i = 1, 2, \ldots, N$
  - with i.i.d assumption usually
- Learn a linear function $f_{\mathbf{W},\mathbf{b}}(\mathbf{x}) = \mathbf{W}\mathbf{x} + \mathbf{b}$ from $\mathcal{D}$
  - $\mathbf{W} \in \mathbb{R}^{K \times D}$, $\mathbf{b} \in \mathbb{R}^K$



Review: Linear Regression

- Linear regression (least squares)

$$\min_{\mathbf{W},\mathbf{b}} \sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{W}\mathbf{x}_i - \mathbf{b}\|^2 \tag{1}$$

- Ridge regression

$$\min_{\mathbf{W},\mathbf{b}} \frac{1}{2} \sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{W}\mathbf{x}_i - \mathbf{b}\|^2 + \frac{\lambda}{2} \|\mathbf{W}\|_F^2 \tag{2}$$

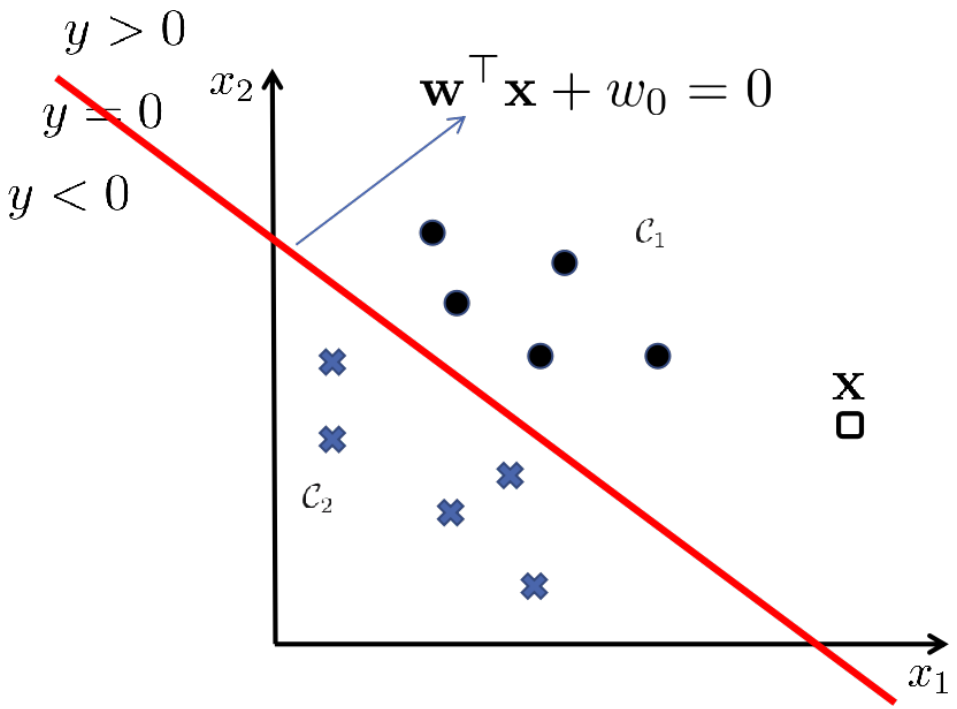
- LASSO

$$\min_{\mathbf{W},\mathbf{b}} \frac{1}{2} \sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{W}\mathbf{x}_i - \mathbf{b}\|^2 + \lambda \|\mathbf{W}\|_1 \tag{3}$$

* $\|\mathbf{W}\|_F = \sqrt{\sum_i \sum_j w_{ij}^2}$, $\|\mathbf{W}\|_1 = \sum_i \sum_j |w_{ij}|$

Review: Linear Classification

- Training data: $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots (\mathbf{x}_N, y_N)\}$
  - $\mathbf{x}_i \in \mathbb{R}^D$, $y_i \in \{+1, -1\}$, $i = 1, 2, \ldots, N$
  - with i.i.d assumption usually
- Learn a linear classifier $f_{\mathbf{w},b}(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + b$ from $\mathcal{D}$



Review: Linear Classification

- Logistic regression (binary classification, $y \in \{0, 1\}$)

$$f_{\mathbf{w},b}(\mathbf{x}) = \sigma(\mathbf{w}^\top \mathbf{x} + b) = \frac{1}{1 + \exp(-\mathbf{w}^\top \mathbf{x} - b)} \tag{4}$$

$$\min_{\mathbf{w},b} \; -\frac{1}{N} \sum_{i=1}^{N} \left(y_i \log f_{\mathbf{w},b}(\mathbf{x}_i) + (1 - y_i) \log(1 - f_{\mathbf{w},b}(\mathbf{x}_i)) \right) \tag{5}$$

- Softmax regression (multi-class classification, $\mathbf{y} \in \{0, 1\}^K$)

$$f^{(j)}_{\mathbf{W},\mathbf{b}}(\mathbf{x}) = \frac{\exp(\mathbf{w}_j^\top \mathbf{x} + b_j)}{\sum_{c=1}^{K} \exp(\mathbf{w}_c^\top \mathbf{x} + b_c)} \tag{6}$$

$$\min_{\mathbf{w},\mathbf{b}} \; -\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{K} y_{ij} \log f^{(j)}_{\mathbf{w},b}(\mathbf{x}_i) \tag{7}$$

Review: K-Nearest Neighbor Method

- k-NN: a nonlinear regression or classification model
- Determine the following beforehand
  - distance metric (ℓ_2 or ℓ_1 norms, etc)
  - number (k) of nearest neighbors
- k-NN is a non-parametric model

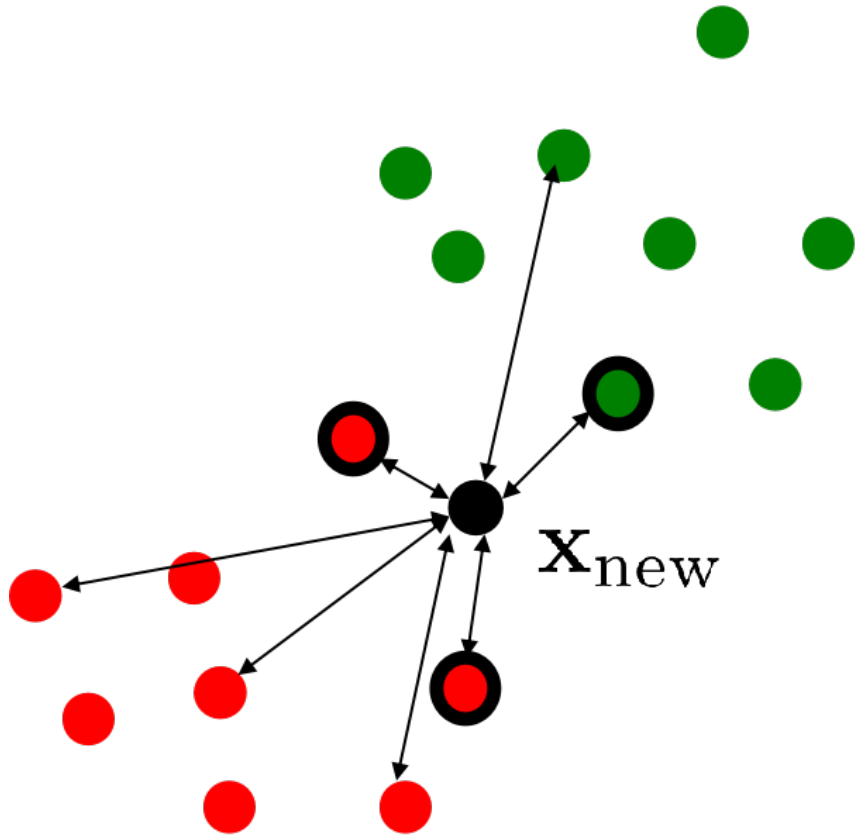
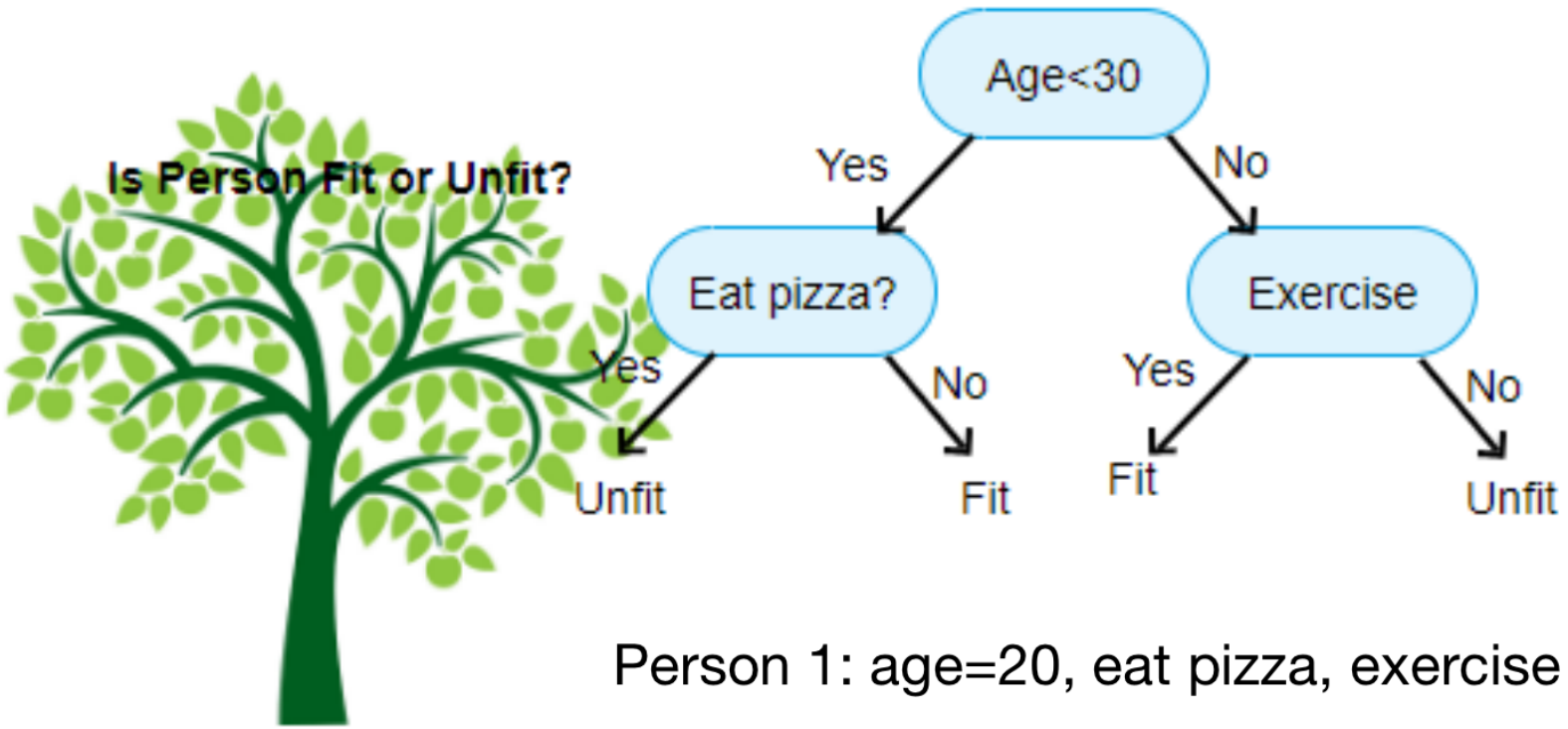


Figure: A toy example (k=3)

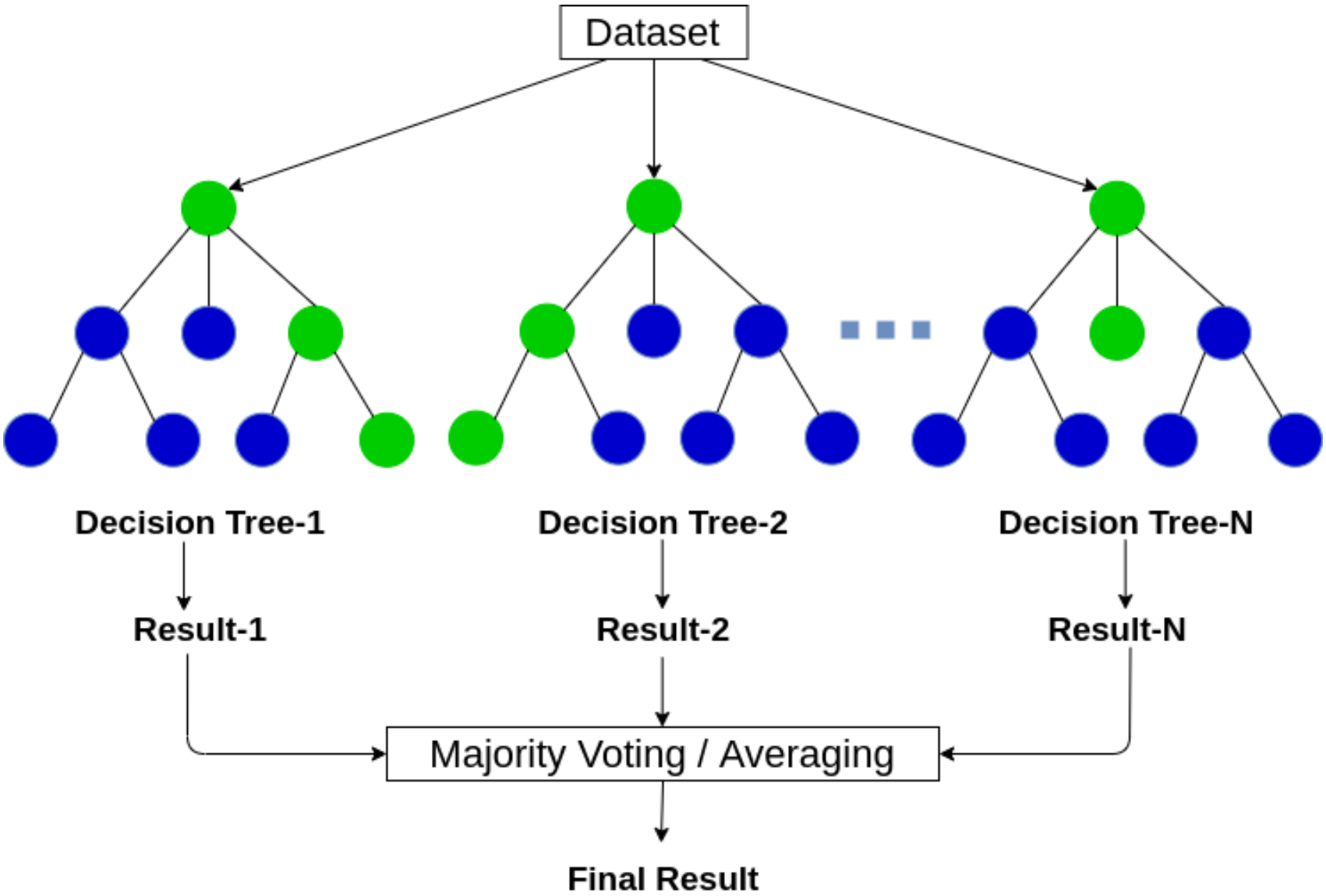
Review: Decision Tree



Person 1: age=20, eat pizza, exercise

Person 2: age=40, no eat pizza, exercise

Review: Random Forest



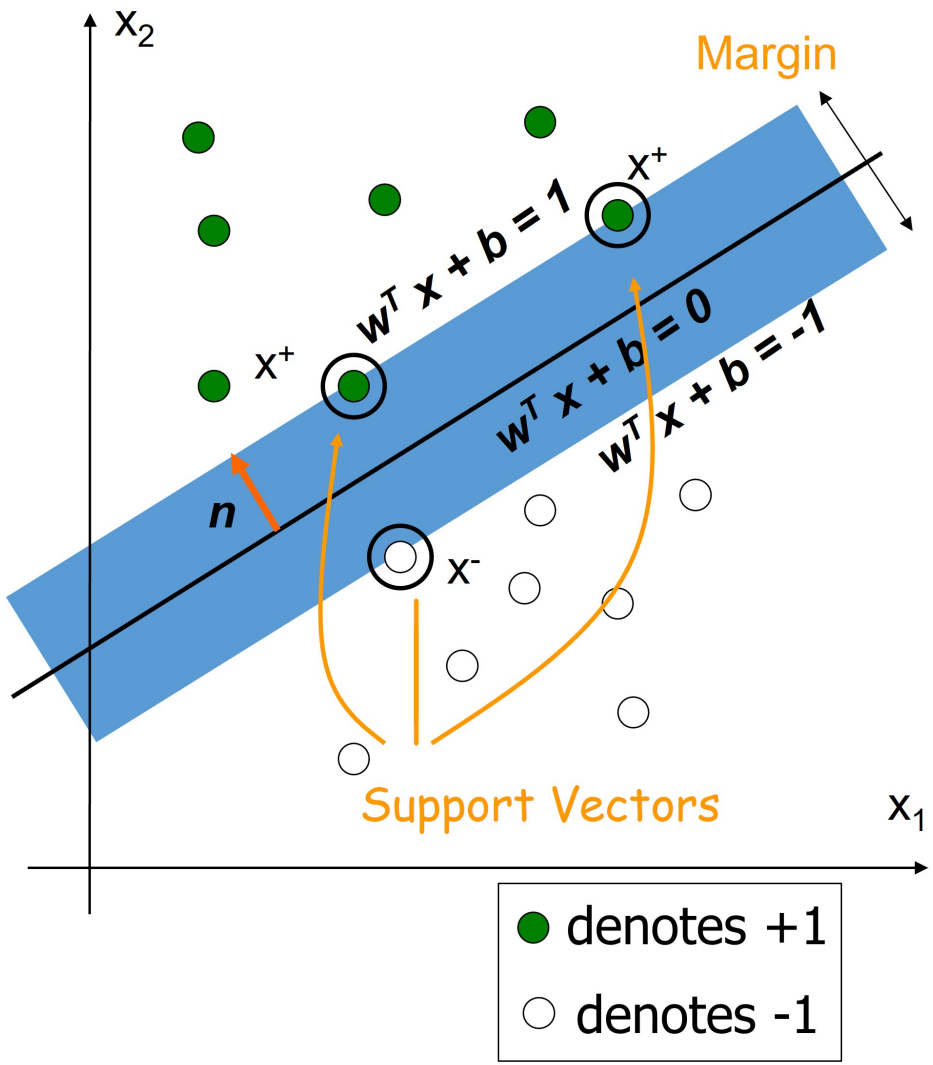
Review: Support Vector Machine

- Margin width:

$$M = (\mathbf{x}^+ - \mathbf{x}^-) \cdot \mathbf{n}$$
$$= (\mathbf{x}^+ - \mathbf{x}^-) \cdot \frac{\mathbf{w}}{\|\mathbf{w}\|} = \frac{2}{\|\mathbf{w}\|}$$

- Maximum margin classifier

$$\min_{\mathbf{w},\, b} \ \frac{1}{2}\|\mathbf{w}\|^2 \quad \text{s.t.} \ \ y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1, \forall i \tag{8}$$



Review: Support Vector Machine

- Dual problem

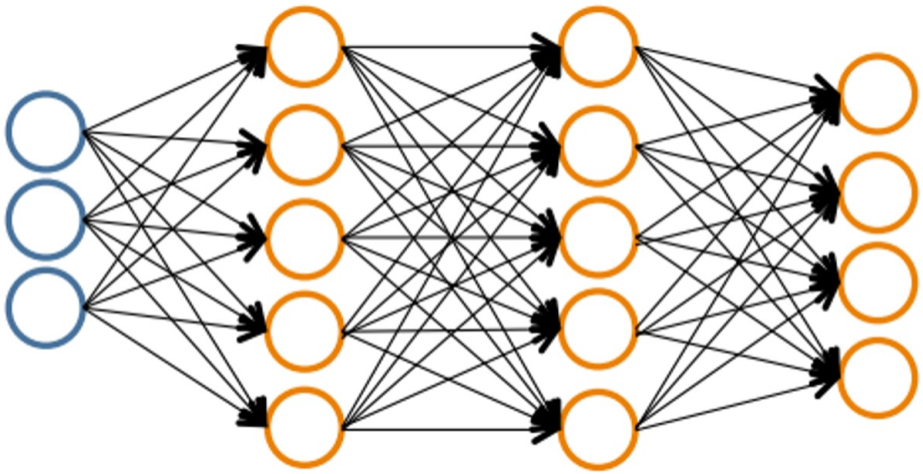
$$\begin{aligned} \max_{\alpha} \ & \mathcal{L}_{\mathcal{D}}(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^\top \mathbf{x}_j \\ \text{s.t.} \ & \sum_{i=1}^{N} \alpha_i y_i = 0, \ \alpha_i \geq 0, i = 1, \ldots, N \end{aligned} \tag{9}$$

- Kernel SVM
  - replace $\mathbf{x}$ with $\phi(\mathbf{x})$
  - $\phi(\mathbf{x}_i)^\top \phi(\mathbf{x}_j) = k(\mathbf{x}_i, \mathbf{x}_j)$
  - $k(\mathbf{x}_i, \mathbf{x}_j)$ is a kernel function, e.g, $k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2)$
- Slacked SVM

$$\begin{aligned} \min_{\mathbf{w}, b, \xi} \ & \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^{N} \xi_i \\ \text{s.t.} \ & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i, \ \ \xi_i \geq 0, \ \ i = 1, \ldots, N \end{aligned} \tag{10}$$

Review: Neural Networks

- Fully connected feedforward network (multi-layer perceptron, MLP)

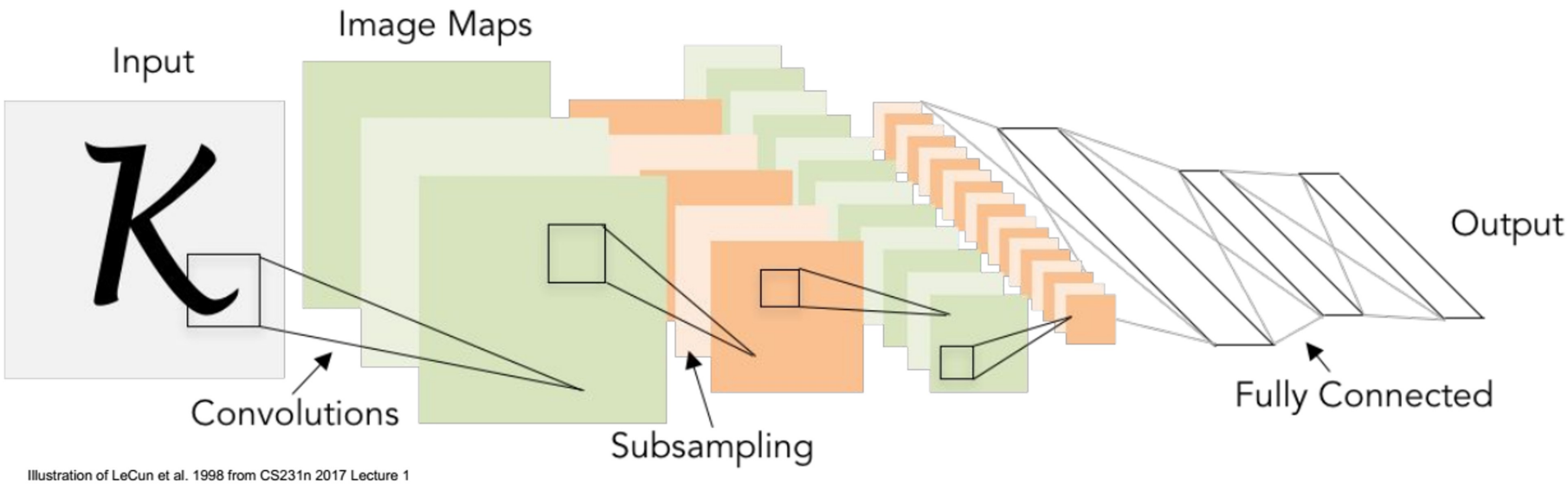


$$\mathbf{h}^{(1)} = f^{(1)}(\mathbf{x}) \quad \mathbf{h}^{(2)} = f^{(2)}(\mathbf{h}^{(1)}) \quad \dots \quad \mathbf{y} = f^{(L)}(\mathbf{h}^{(L-1)})$$

$$\text{Or} \qquad \mathbf{y} = f^{(L)} \circ \dots \circ f^{(1)}(\mathbf{x})$$

Review: Neural Networks

- Convolutional neural network (CNN)



LeNet-5

Review: Neural Networks

- Recurrent neural network (RNN)

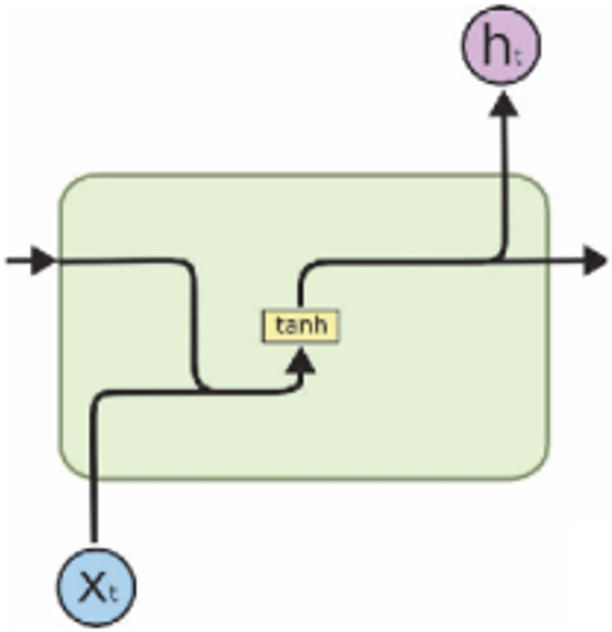
$$h_t = f_W(h_{t-1}, x_t)$$

new state

old state

input vector at some time step

some function with parameters W



Review: Classification on Real Data

Classification on MNIST handwritten digits dataset

`http://yann.lecun.com/exdb/mnist/`

Figure: Samples of MNIST (28 $\times$ 28 gray-scale images, 60k for training, 10 k for testing)

classifier	test error rate (%)
linear classifier (least squares)	12.0
k-nearest-neighbors	5.0
generalized linear classifier (Gaussian basis 1000)	3.6
neural network (MLP) 500-300 HU, softmax	1.53
CNN LeNet-5	0.95
SVM (Gaussian kernel)	1.4

Review: Classification on Real Data

Classification on Fashion-MNIST dataset

https://cloudxlab.com/blog/fashion-mnist-using-machine-learning/

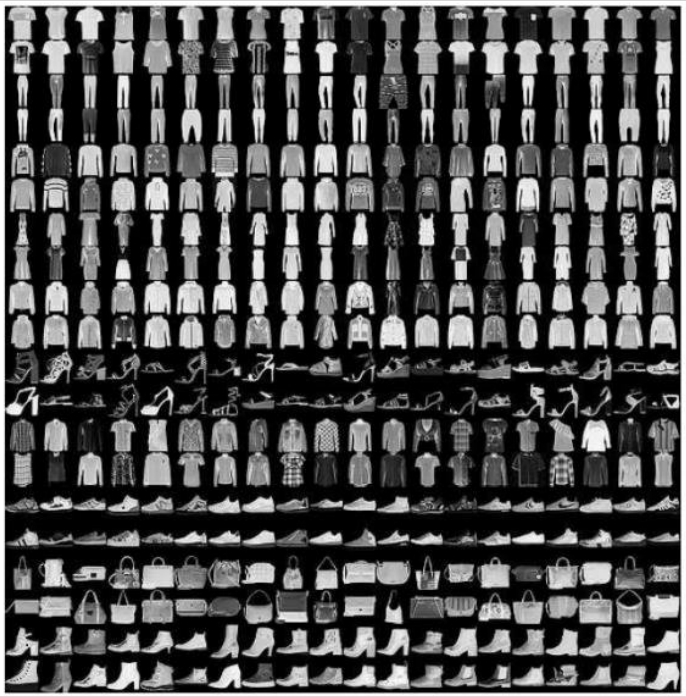
Label	Description	Examples
0	T-Shirt/Top	
1	Trouser	
2	Pullover	
3	Dress	
4	Coat	
5	Sandals	
6	Shirt	
7	Sneaker	
8	Bag	
9	Ankle boots	

Figure: Samples of Fashion-MNIST (28×28 gray-scale images, 60k for training, 10 k for testing)

classifier	test error rate (%)
softmax	15.3
decision tree	21.06
random forest	15.18
neural network (MLP) (256-128-100 HU)	12.6
CNN	<8
HOG+SVM	7.4
Google AutoML	6.1

More results are at https://github.com/zalandoresearch/fashion-mnist

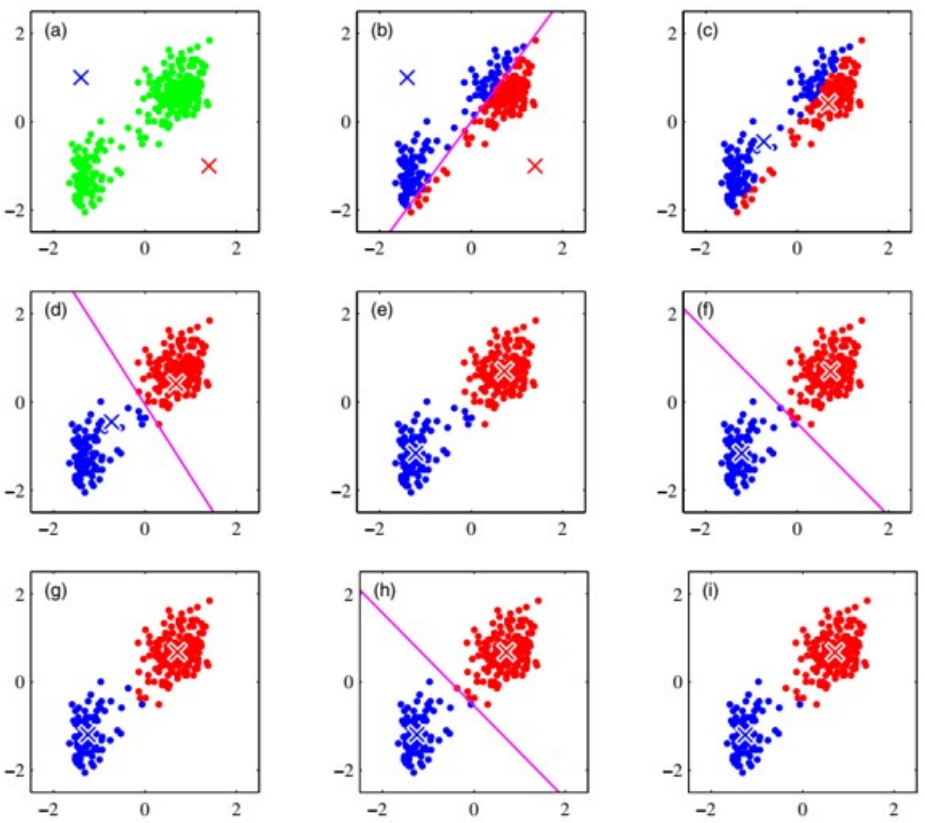
Review: K-Means Clustering

- Clustering (unsupervised learning): given a set of D-dimensional data $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$, partition them into K clusters such that each data point is similar to the data in the same cluster and dissimilar to the data in different clusters.

- Denote cluster j by $\mathcal{C}_j$ and let $\boldsymbol{\mu}_j$ be the centroid of $\mathcal{C}_j$.
  K-means clustering minimizes

$$J(\mu) = \sum_{j=1}^{K} \sum_{\mathbf{x} \in \mathcal{C}_j} \|\mathbf{x} - \boldsymbol{\mu}_j\|^2 \tag{11}$$

- Algorithm (alternate)
  1. Assign each data point to the closest center
  2. Update the cluster center



Review: Gaussian Mixture Models

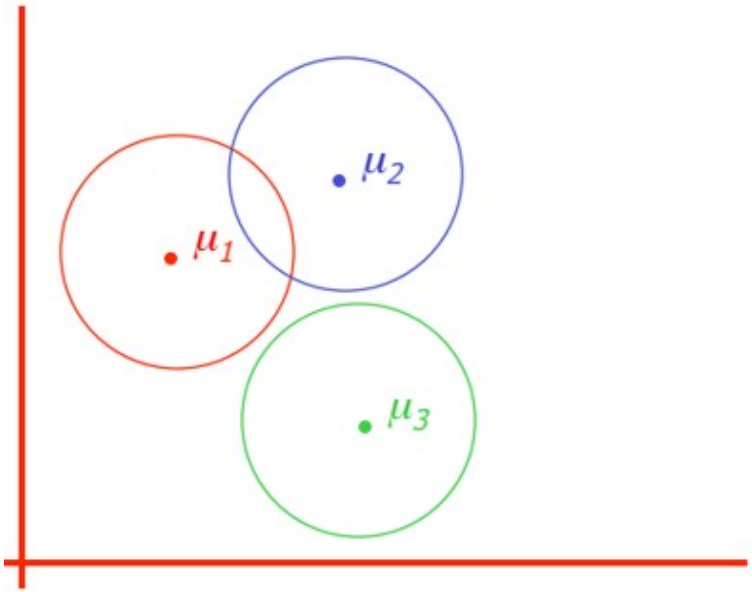
- Multivariate Gaussian distribution

$$p(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^D |\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)$$

- Gaussian mixture distribution

$$p(\mathbf{x}) = \sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_j, \Sigma_j)$$

  - K different Gaussian distributions
  - $\{\pi_j\}$: mixing coefficients
  - $\sum_{j=1}^{K} \pi_j = 1, \quad 0 \le \pi_j \le 1$

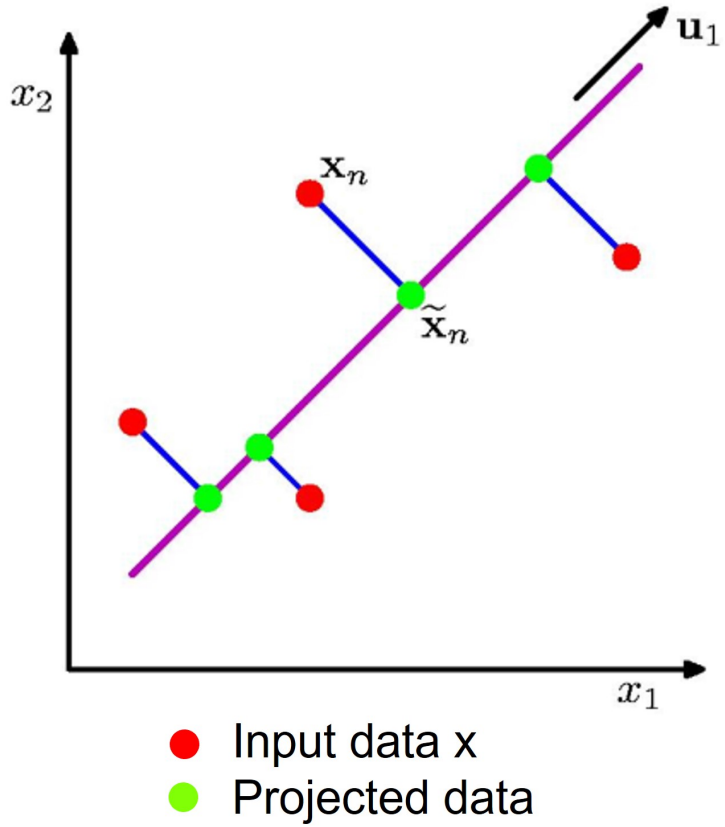


- Algorithm: Expectation-Maximization

Review: Principal Component Analysis

- PCA: find the orthogonal projection of data onto a lower-dimensional subspace that
  - maximizes the variance of projected data
  - or minimizes the reconstruction error, i.e.,

$$
\begin{aligned}
J &= \frac{1}{N}\sum_{i=1}^{N}\|\mathbf{x}_i - \tilde{\mathbf{x}}_i\|^2 \\
&= \frac{1}{N}\sum_{i=1}^{N}\left\|\mathbf{x}_i - \mathbf{U}\mathbf{U}^\top\mathbf{x}_i\right\|^2
\end{aligned}
\tag{12}
$$



  * $\boldsymbol{x} \in \mathbb{R}^D, \quad \boldsymbol{U} \in \mathbb{R}^{D\times d}$

- Solution of PCA: eigenvalue decomposition or singular value decomposition

Review: More Topics (optional)

- Bayes' theorem, maximum likelihood estimation (MLE), maximum a posteriori estimation (MAP)
- Classification evaluation metrics
  - Precision, recall, accuracy, F1-score, AUC (TPR/FPR)
- Cross-validation
- Over-/under-fitting and bias-variance trade-off
- Expectation maximization
- Kernel density estimation
- Clustering evaluation metrics