ETH zürich

Machine Learning for Chemoinformatics

An introduction

Francesca Grisoni

University of Milano-Bicocca, Dept. of Earth and Environmental Sciences, Milan, Italy

ETH Zurich, Dept. of Chemistry and Applied Biosciences, Zurich, Switzerland

francesca.grisoni@unimib.it

UNIVERSITÀ DEGLI STUDI DI MILANO BICOCCA

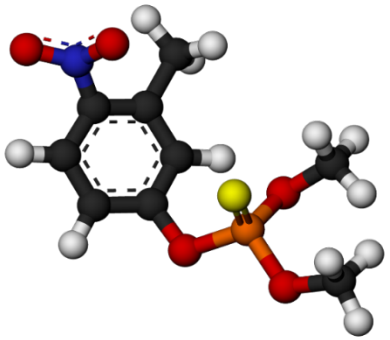
MODLAB

Presentation Outline

- Introduction
  - Definition
  - Elements of Machine Learning
- Additional Considerations
  - The NFL theorem
  - Validation
  - Applicability
- Machine Learning approaches: some examples
  - Local methods
  - Tree-like approaches
  - Neural Networks

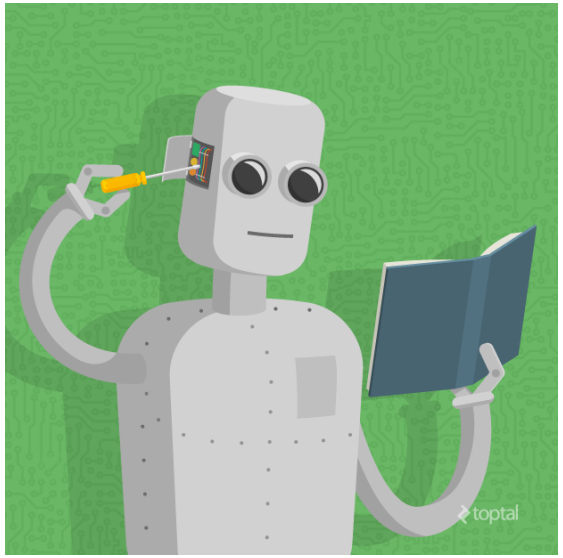
Machine learning in chemoinformatics

- Biological activity prediction
- Toxicity
- Physico-chemical properties
- Multi-objective optimization
- Rational drug design

$P = f($  $)$

Machine Learning (ML)

"Machine Learning is the field of study that gives computers the ability to learn without being explicitly programmed."[1]



https://www.toptal.com/machine-learning/machine-learning-theory-an

"A computer program is said to **learn from** **experience** *E* with respect to **some task** *T* and some **performance measure** P, if its performance on *T*, as measured by *P*, improves with experience *E*."[2]

(1) Data

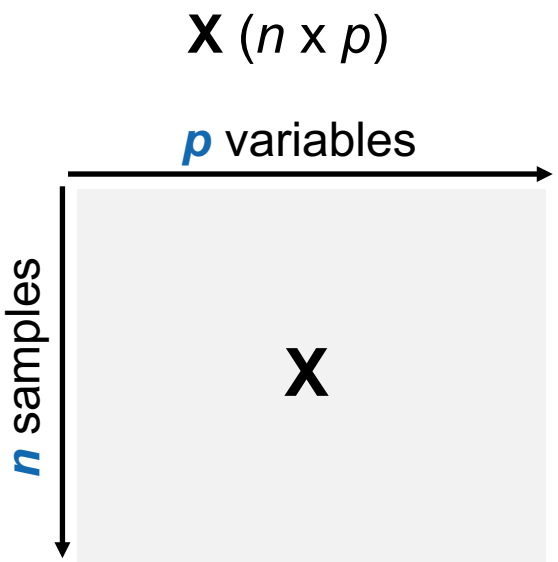
(2) Task

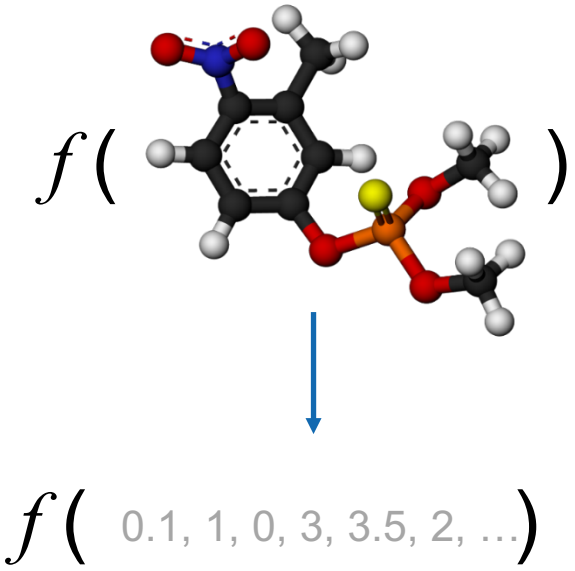
(3) Performance

[1] Samuel, A. L. (1959). "Some studies in machine learning using the game of checkers". IBM Journal of research and development, 3(3), 210-229.

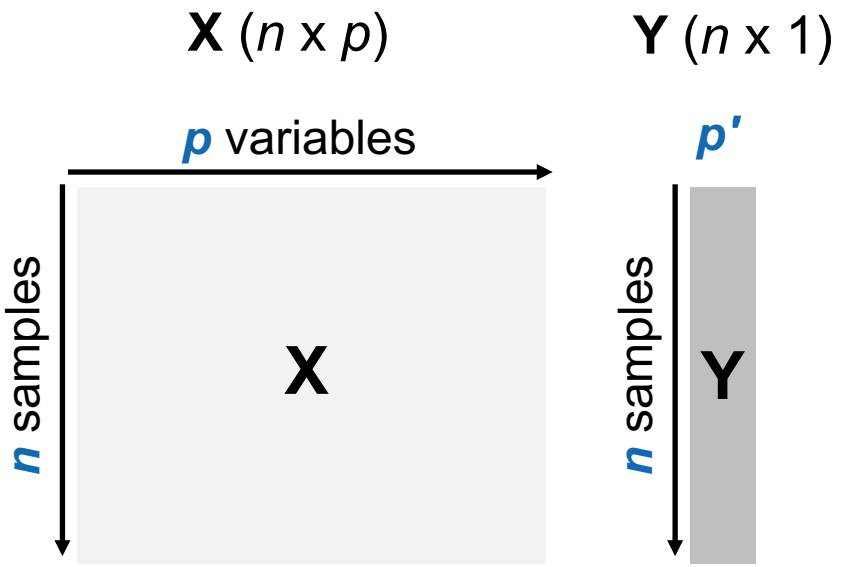
[2] Mitchell, T. M. (1997). Machine learning. 1997. Burr Ridge, IL: McGraw Hill, 45(37), 870-877.

(1) The data and the "*G-I-G-O*" principle

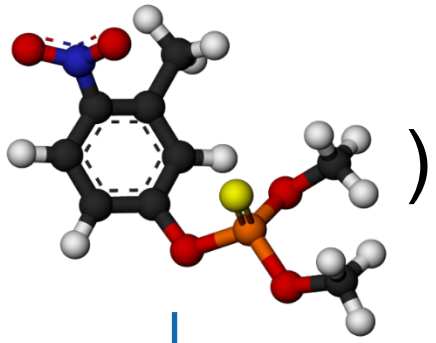




(1) The data and the "*G-I-G-O*" principle

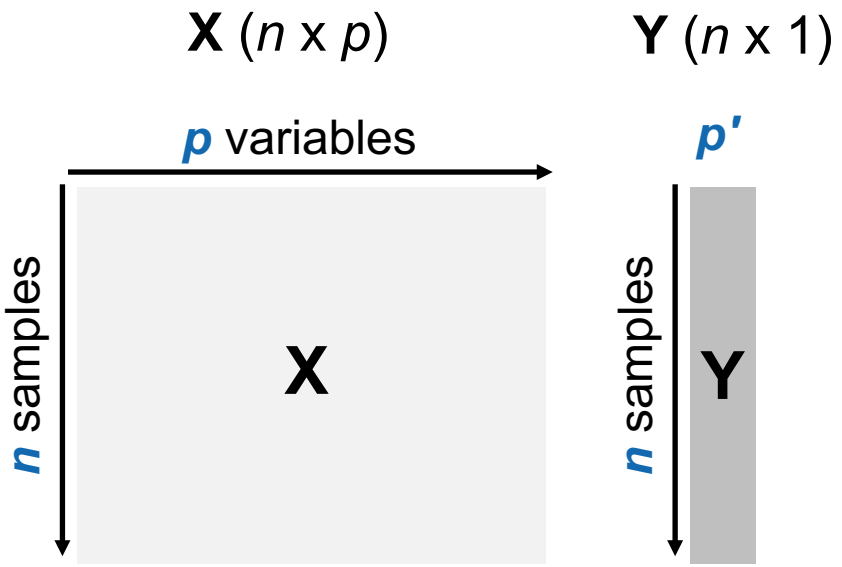


$$P = f\left(\right)$$

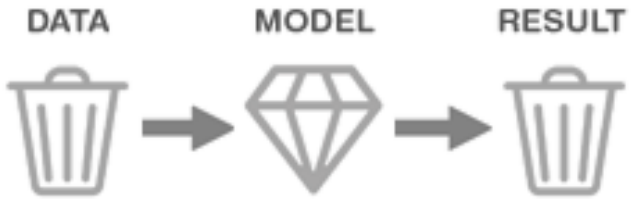


$$P = f(\ 0.1,\ 1,\ 0,\ 3,\ 3.5,\ 2,\ \ldots)$$

(1) The data and the "*G-I-G-O*" principle



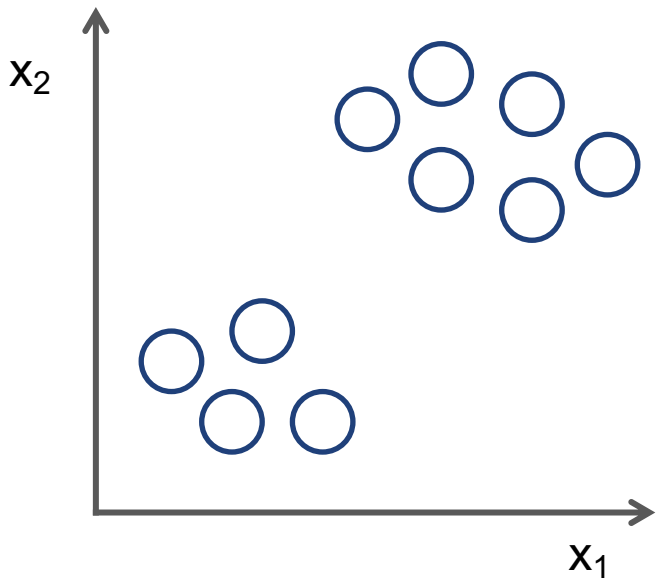
Garbage **I**n = **G**arbage **O**ut



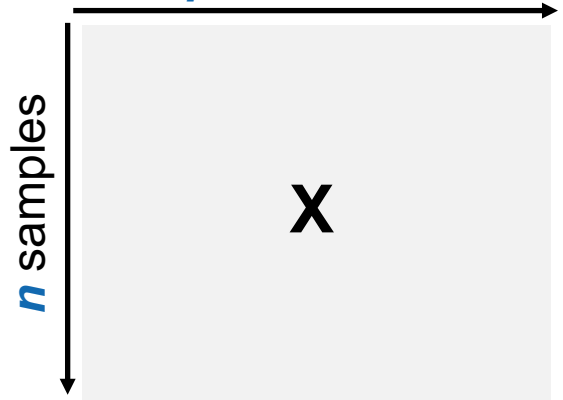
- Structures
- Experimental Responses

(2) Machine Learning Tasks

Unsupervised Learning

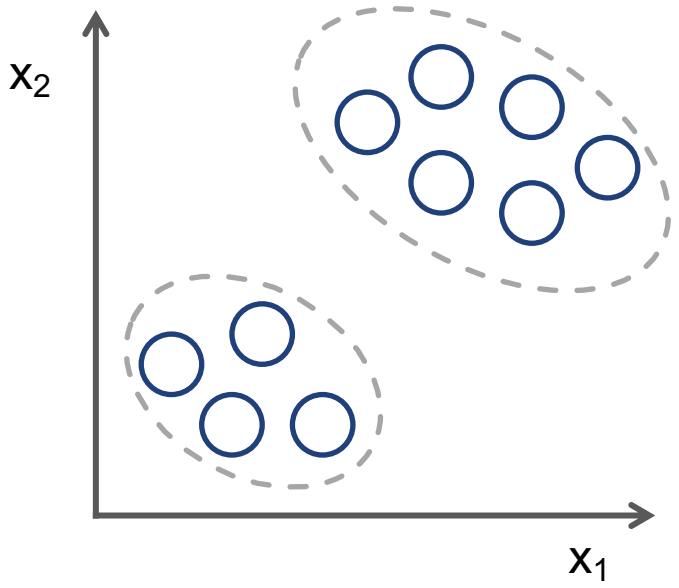


X (*n* x *p*)

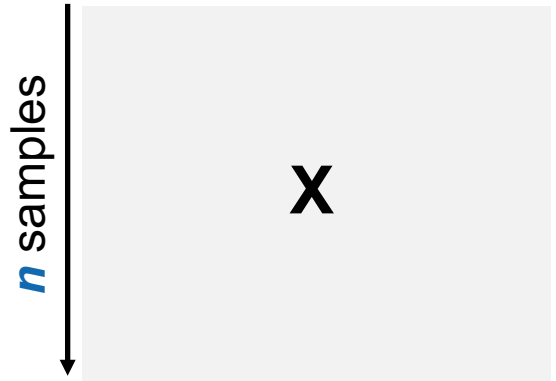


(2) Machine Learning Tasks

Unsupervised Learning

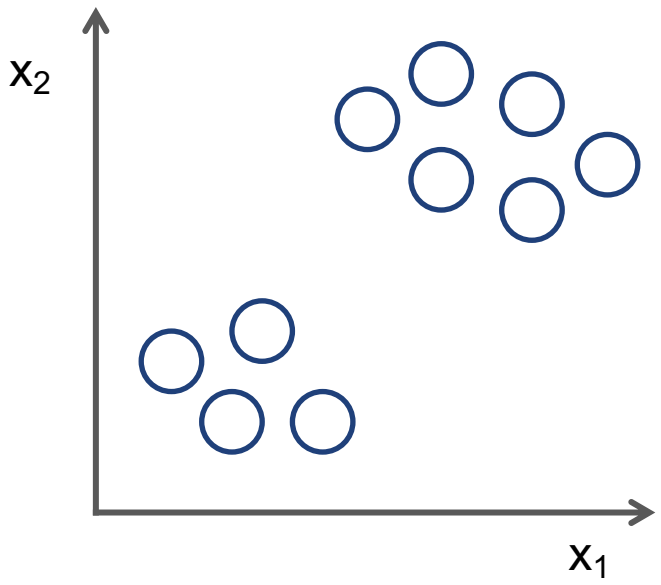


X (*n* x *p*)

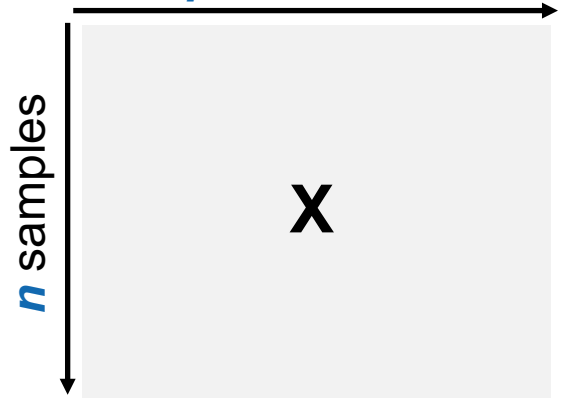


(2) Machine Learning Tasks

Supervised Learning

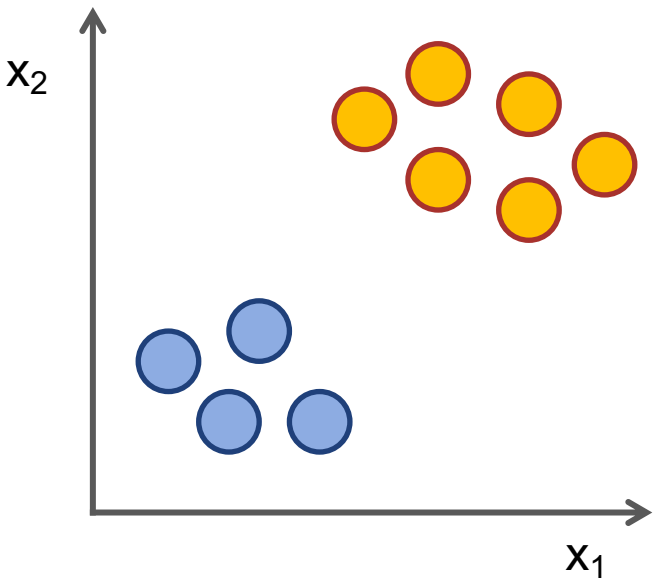


X (*n* x *p*)

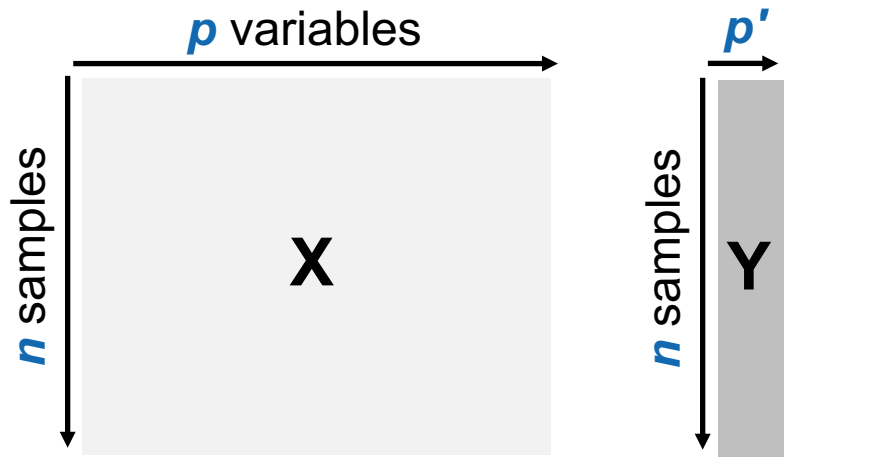


(2) Machine Learning Tasks

Supervised Learning

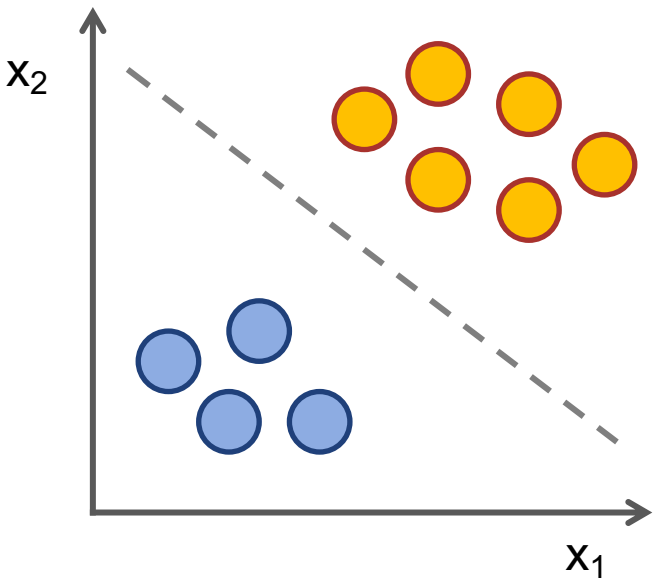


X (*n* x *p*) + **Y** (*n* x 1)

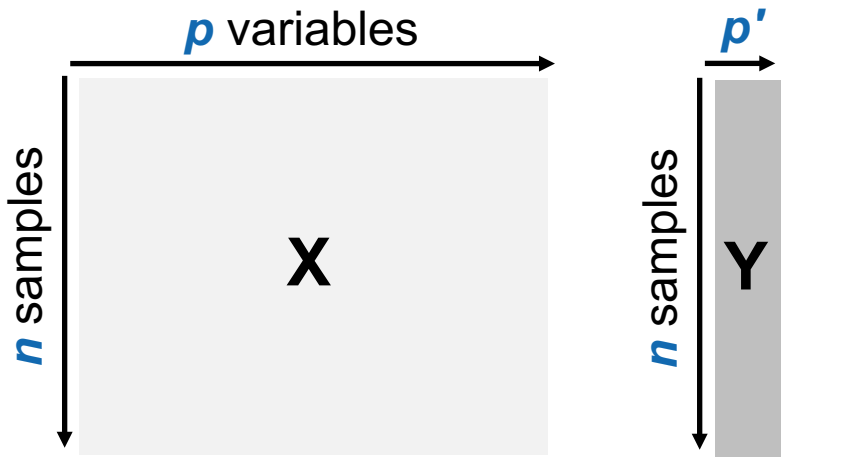


(2) Machine Learning Tasks

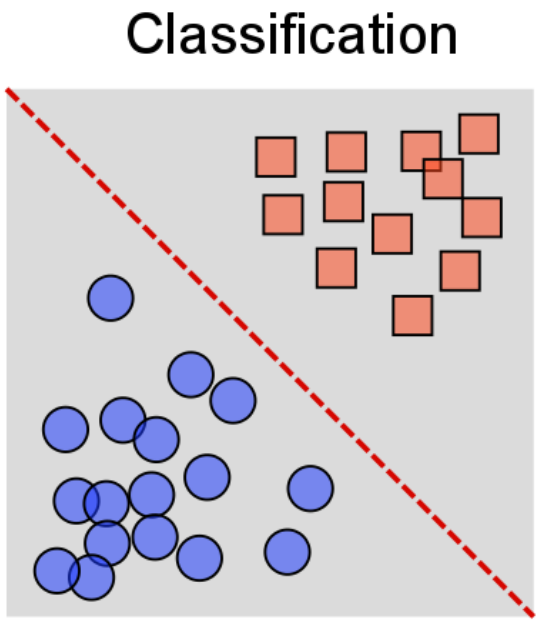
Supervised Learning

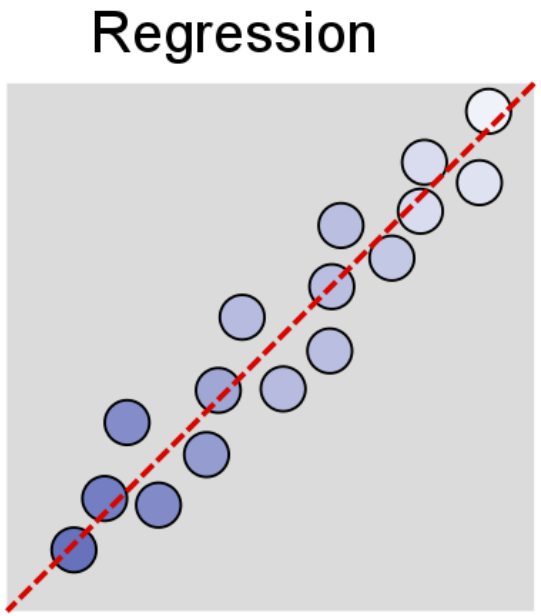


X (*n* x *p*) + **Y** (*n* x 1)

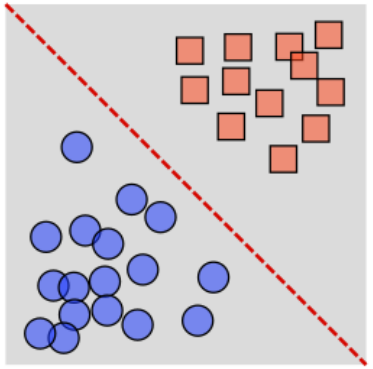
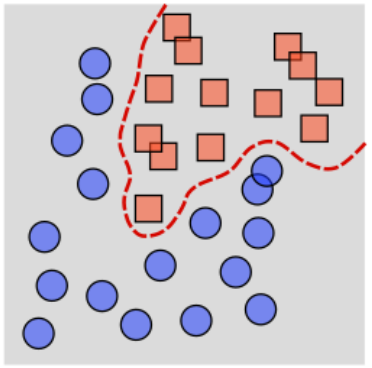
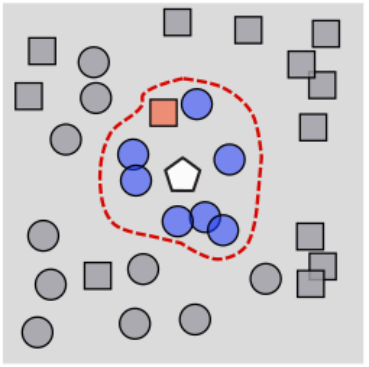
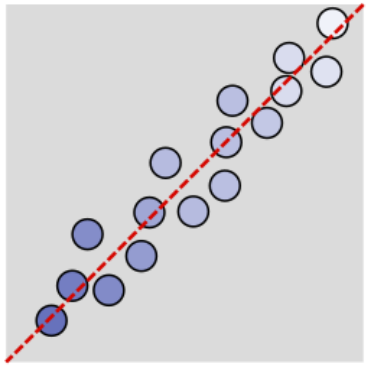
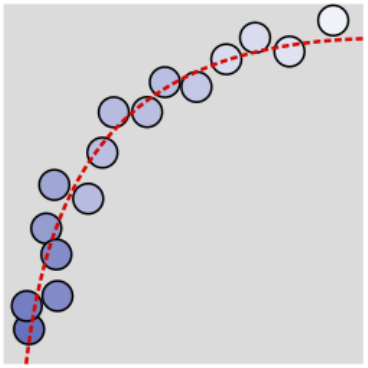
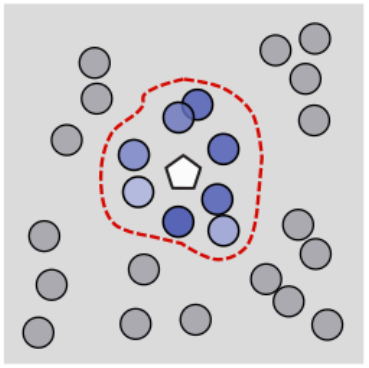


(2) Machine Learning Tasks





(2) Machine Learning Tasks

	Linear	Non-linear	Local
Classification			
Regression			

(3) Performance

Classification

	N	P
N	TN	FP
P	FN	TP

Single class

- **Sensitivity** or True Positive rate (TPR)

$$Sn_P \equiv TPR = \frac{TP}{TP + FN} = Sp_N$$

- **Specificity** or True Negative rate (TNR)

$$Sp_P \equiv TNR = \frac{TN}{TN + FP} = Sn_N$$

- **Precision**

$$Pr_P = \frac{TP}{TP + FP}$$

(3) Performance

Classification

	N	P
N	TN	FP
P	FN	TP

Global Performance

- **Non-Error Rate** or Balanced-Accuracy ϵ [0,1]

$$NER = \frac{Sn_P + Sn_N}{2}$$

- **Matthews Correlation Coefficient** (MCC) ϵ [-1,1]

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FN) \cdot (TP + FP) \cdot (TN + FP) \cdot (TN + FN)}}$$

- **Accuracy** ϵ [0,1]

$$Acc = \frac{TP + TN}{n}$$

(3) Performance

Classification

	N	P
N	TN	FP
P	FN	TP

N = 990; P = 10
TN = 990 (100%)
TP = 0 (0%)

Global Performance

- **Non-Error Rate** or Balanced-Accuracy ϵ [0,1]

$$NER = \frac{Sn_P + Sn_N}{2}$$

- **Matthews Correlation Coefficient** (MCC) ϵ [-1,1]

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FN)\cdot(TP+FP)\cdot(TN+FP)\cdot(TN+FN)}}$$

- **Accuracy** ϵ [0,1]

$$Acc = \frac{TP + TN}{n}$$

(3) Performance

Classification

	N	P
N	TN	FP
P	FN	TP

N = 990; P = 10
TN = 990 (100%)
TP = 0 (0%)

Sn_P = 0/10 = 0%
Sn_N = 990/990 = 100%

Global Performance

- **Non-Error Rate** or Balanced-Accuracy ϵ [0,1]

$$NER = \frac{Sn_P + Sn_N}{2}$$

- **Matthews Correlation Coefficient** (MCC) ϵ [-1,1]

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FN)\cdot(TP+FP)\cdot(TN+FP)\cdot(TN+FN)}}$$

- **Accuracy** ϵ [0,1]

$$Acc = \frac{TP + TN}{n}$$

(3) Performance

Classification

	N	P
N	TN	FP
P	FN	TP

N = 990; P = 10
TN = 990 (100%)
TP = 0 (0%)

Sn_P = 0/10 = 0%
Sn_N = 990/990 = 100%

NER = 50%

Global Performance

- **Non-Error Rate** or Balanced-Accuracy ϵ [0,1]

$$NER = \frac{Sn_P + Sn_N}{2}$$

- **Matthews Correlation Coefficient** (MCC) ϵ [-1,1]

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FN)\cdot(TP+FP)\cdot(TN+FP)\cdot(TN+FN)}}$$

- **Accuracy** ϵ [0,1]

$$Acc = \frac{TP + TN}{n}$$

(3) Performance

Classification

	N	P
N	TN	FP
P	FN	TP

N = 990; P = 10
TN = 990 (100%)
TP = 0 (0%)

Sn_P = 0/10 = 0%
Sn_N = 990/990 = 100%

NER = 50%
Acc = 99%

Global Performance

- **Non-Error Rate** or Balanced-Accuracy ϵ [0,1]

$$NER = \frac{Sn_P + Sn_N}{2}$$

- **Matthews Correlation Coefficient** (MCC) ϵ [-1,1]

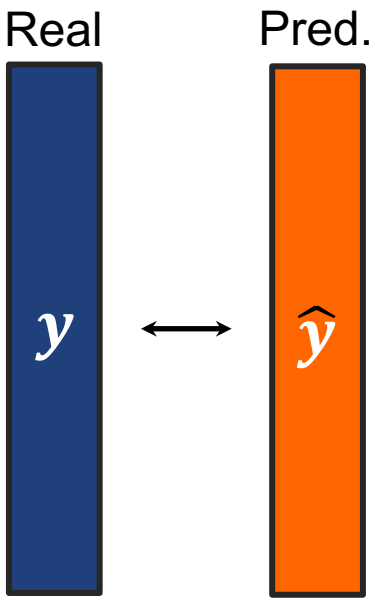
$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FN)\cdot(TP+FP)\cdot(TN+FP)\cdot(TN+FN)}}$$

- **Accuracy** ϵ [0,1]

$$Acc = \frac{TP + TN}{n}$$

(3) Performance

Regression



- **Root Mean Squared Error in Prediction** (RMSEP)

$$RMSEP = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}}$$

Additional Considerations

Additional Considerations

1. Choice of the learner

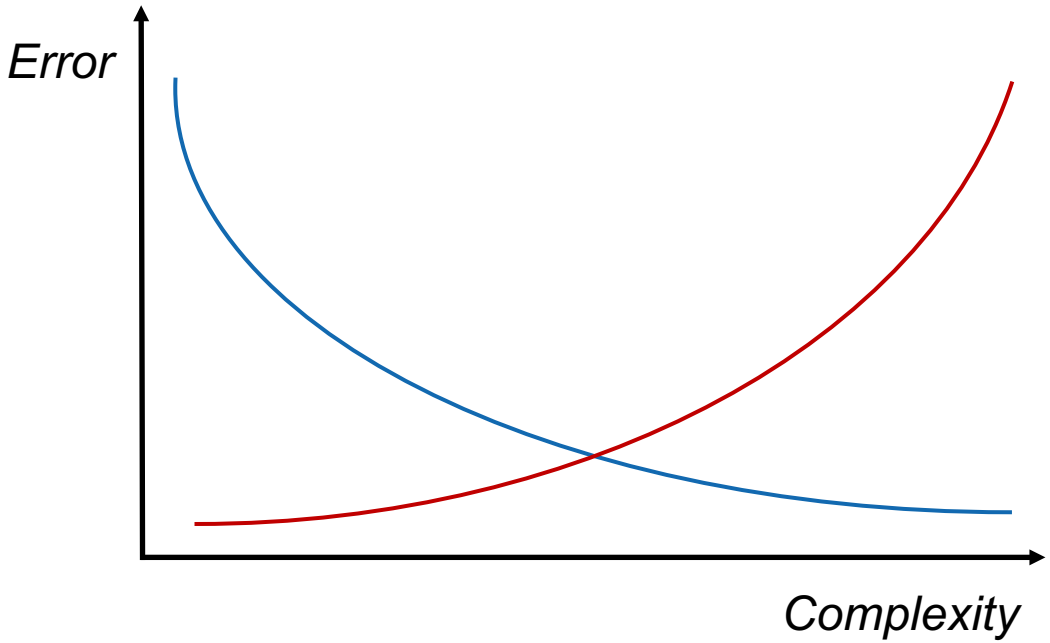
"No Free Lunch" Theorem:

"For every learner, there exists a task on which it fails, even though that task can be successfully learned by another learner."[1]

[1] Shalev-Shwartz, S., & Ben-David, S. (2014). "*Understanding machine learning: From theory to algorithms*." Cambridge university press.

Additional Considerations

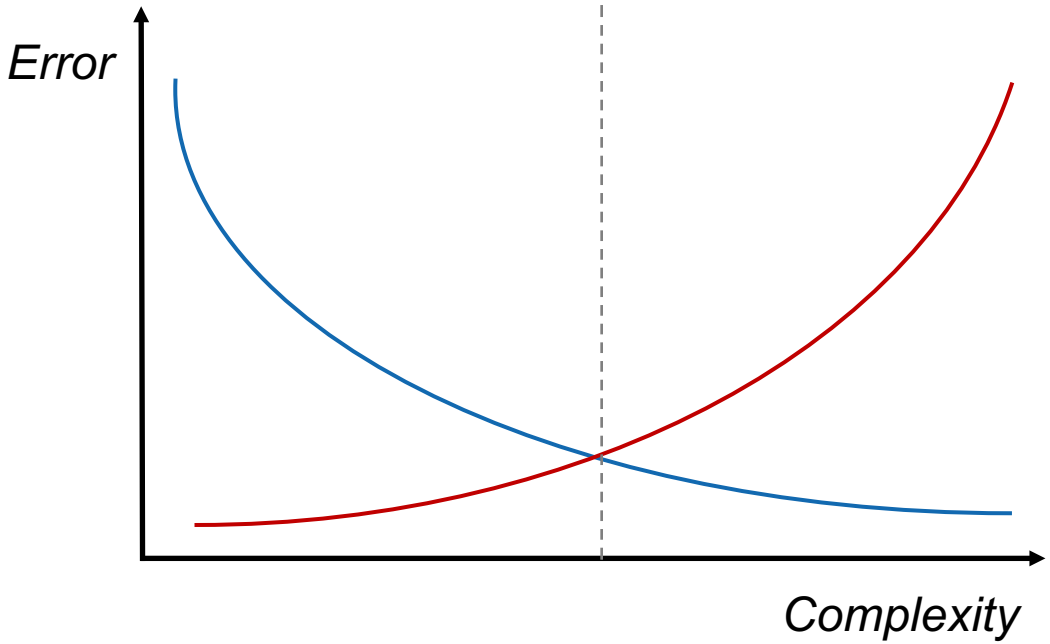
2. Bias-Variance Trade-off



- Bias → generalization (underfitting)
- Variance → descriptive ability (overfitting)

Additional Considerations

2. Bias-Variance Trade-off



- Bias → generalization (underfitting)
- Variance → descriptive ability (overfitting)

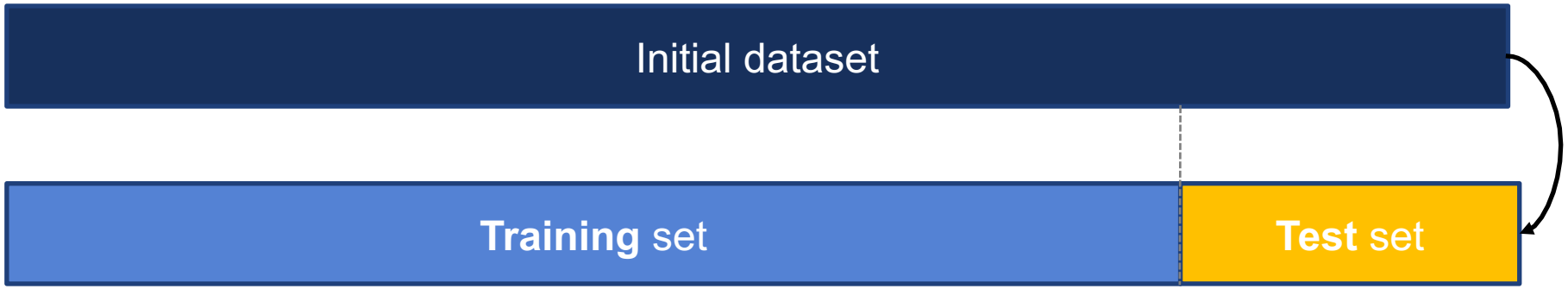
Additional Considerations

3. Validation

Initial dataset

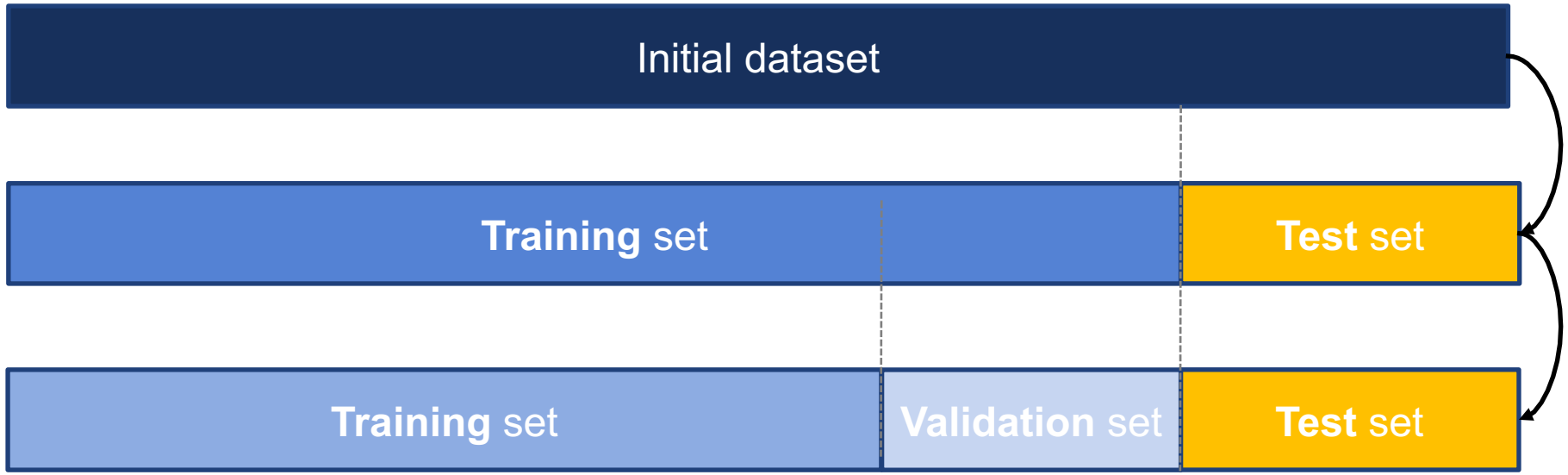
Additional Considerations

3. Validation



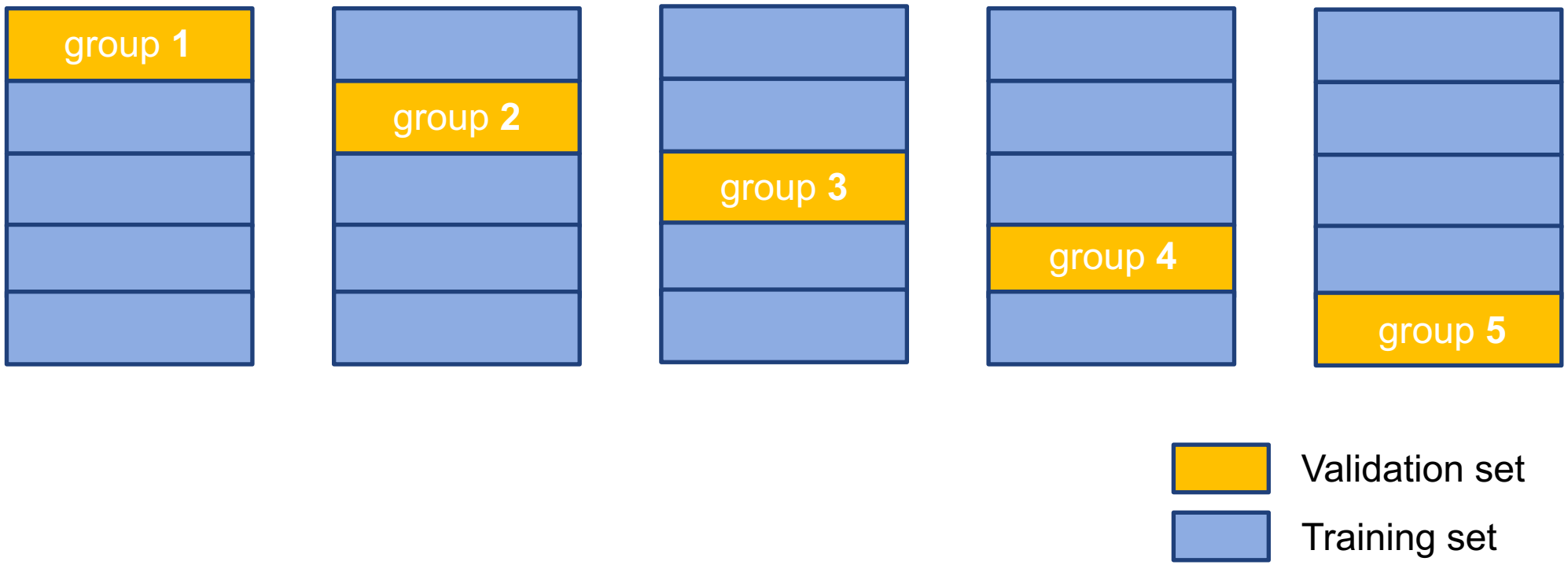
Additional Considerations

3. Validation



Additional Considerations

3. Validation



Additional Considerations

4. Applicability

The “No Free Dessert (either!)” theorem

Machine learning models → **Reductionist**

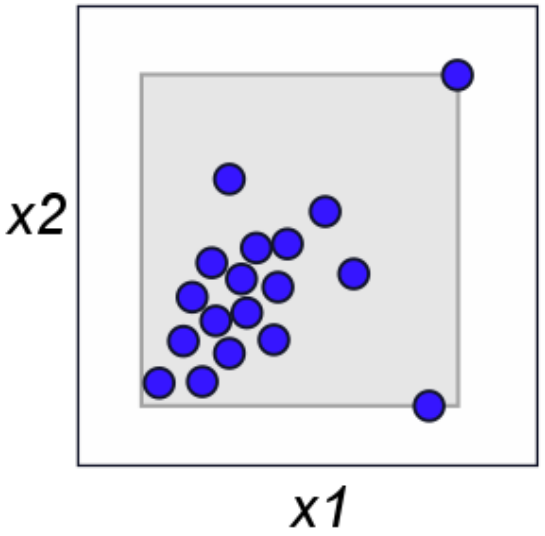
- Types of chemical structures
- Physicochemical properties
- Mechanisms of action considered

Applicability Domain: Chemical space where the property can be reliably predicted

Additional Considerations

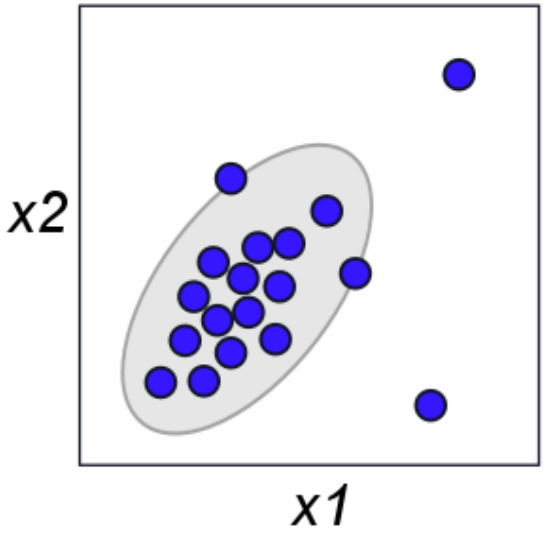
4. Applicability

Bounding-box



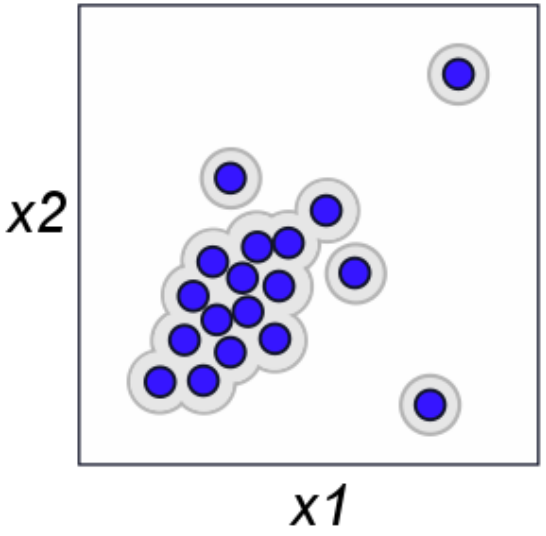
$$\min(x_1),\ \max(x_1)$$
$$\min(x_2),\ \max(x_2)$$

Leverage



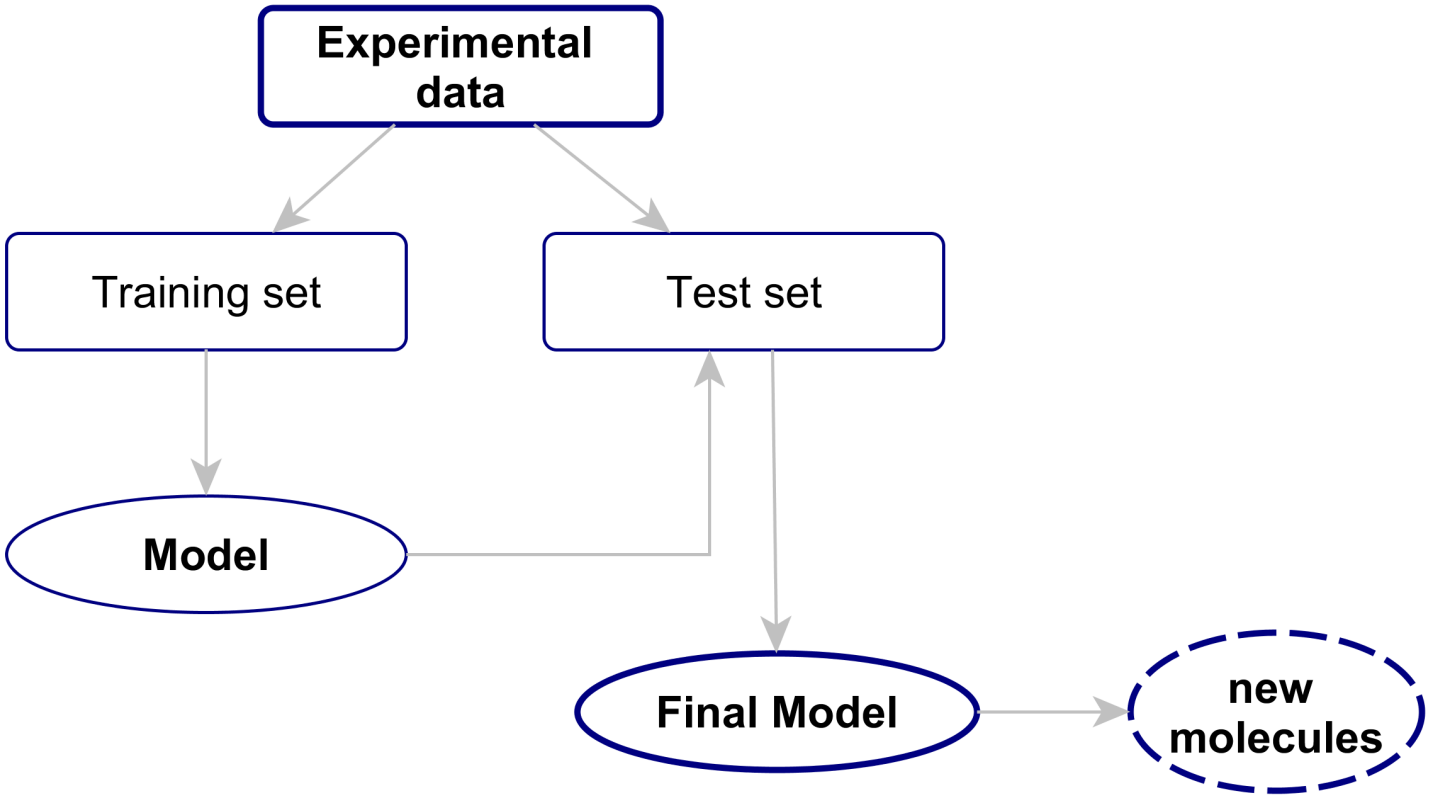
$$H = X \cdot (X^T X)^{-1} \cdot X$$

Distance-based



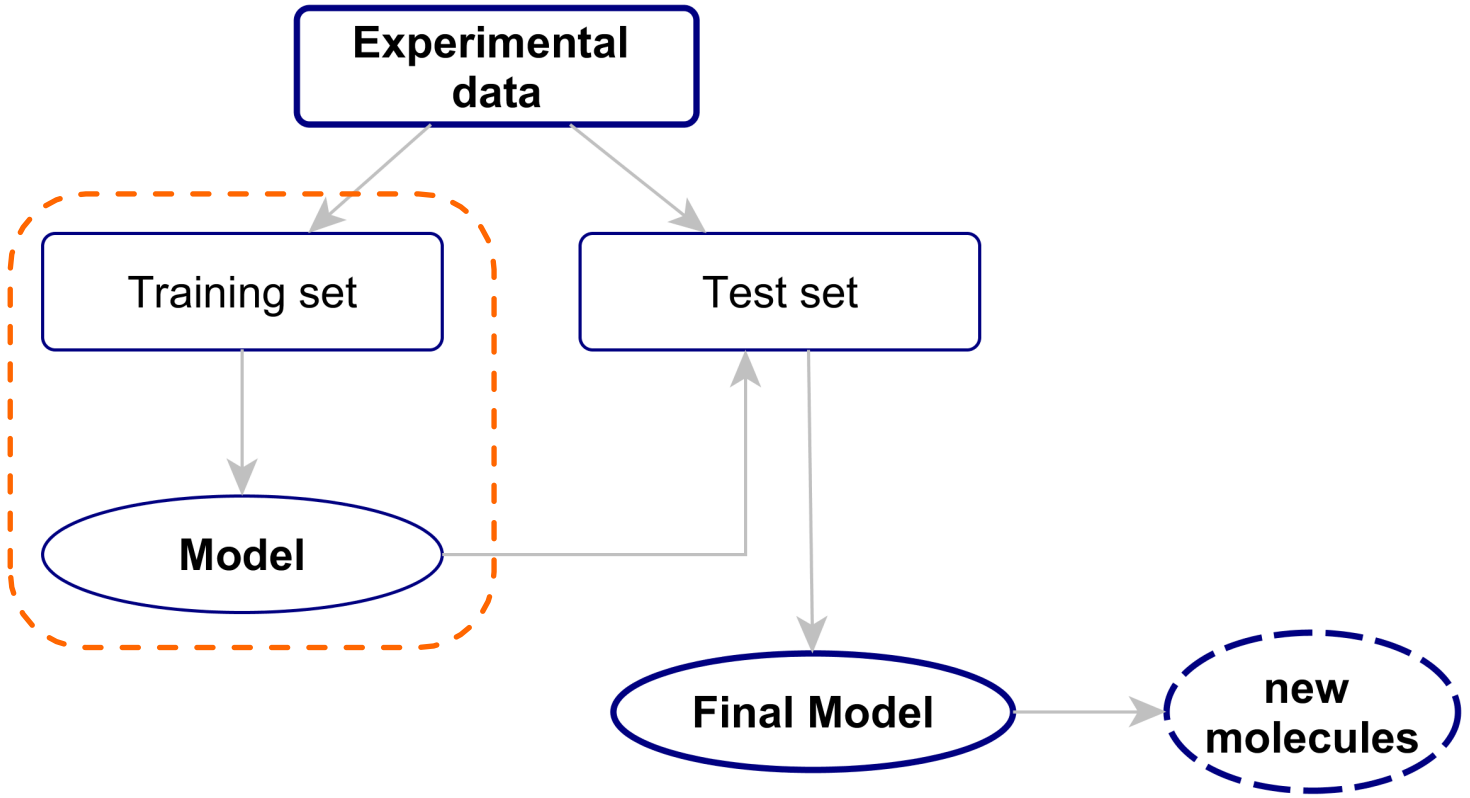
$$D_{xy}^{\text{Man}} = \sum_{j=1}^{p} \left| x_j - y_j \right|$$

Standard Machine Learning workflow (in chemoinformatics)

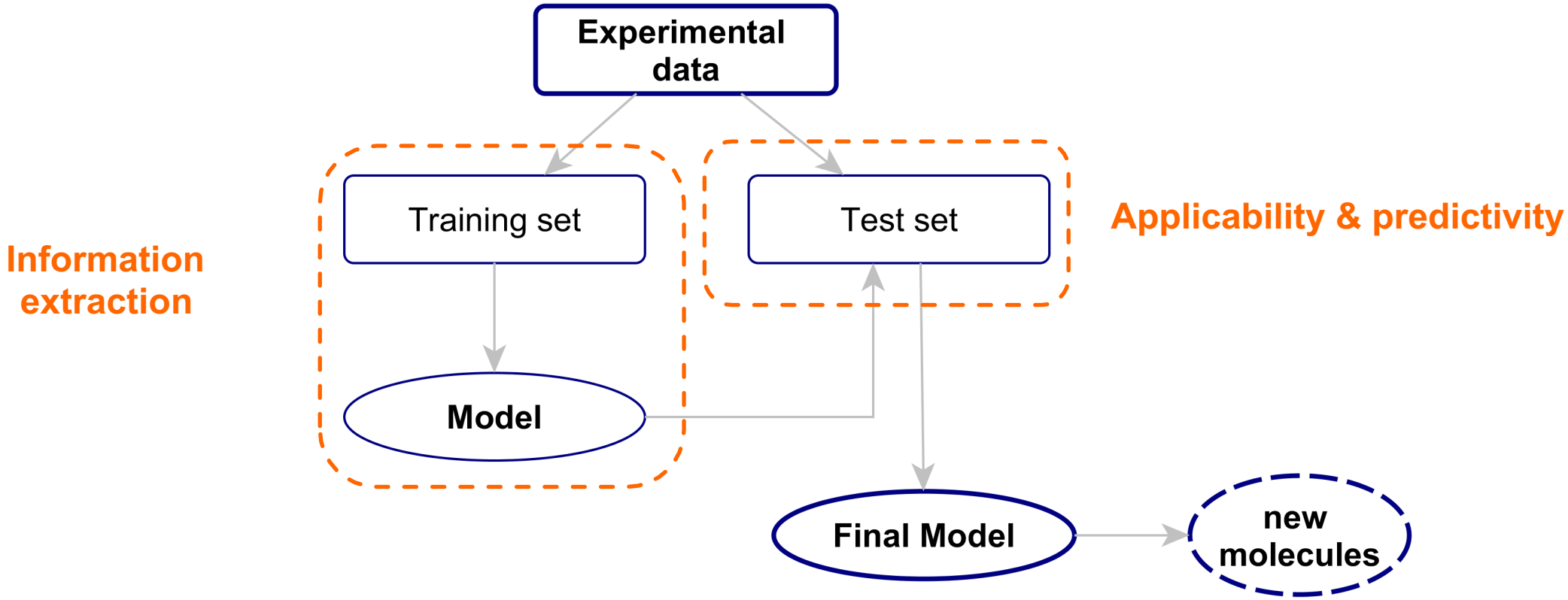


Standard Machine Learning workflow (in chemoinformatics)

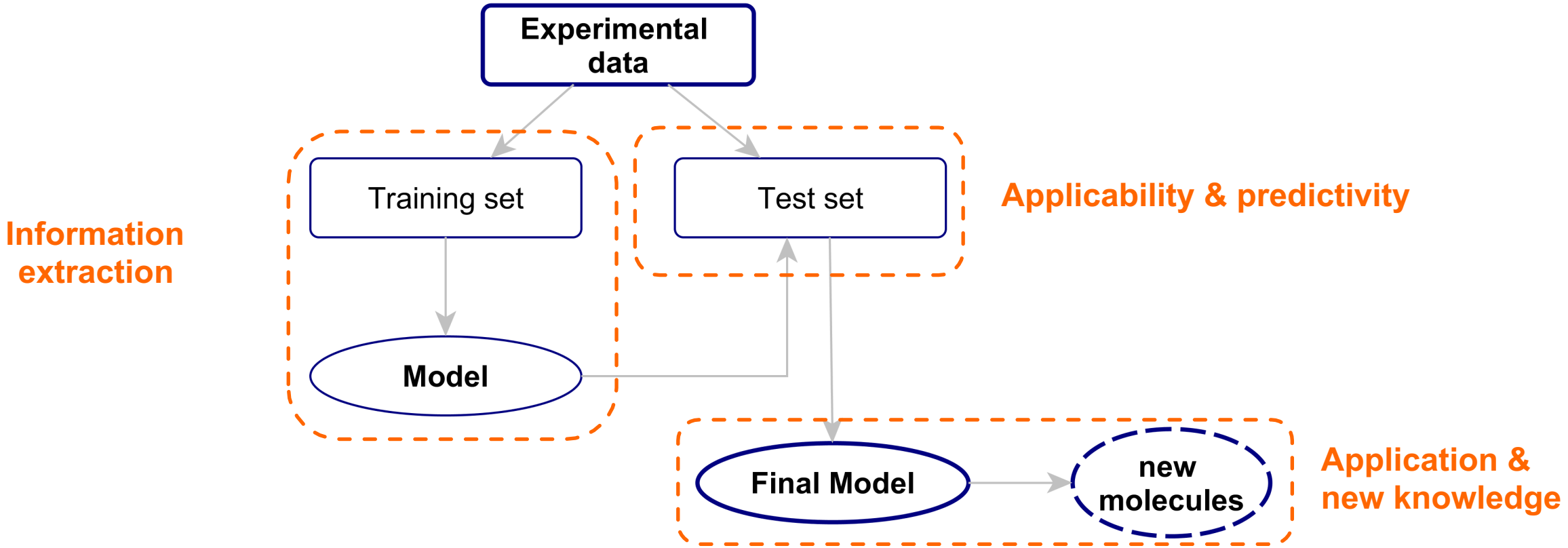
Information extraction



Standard Machine Learning workflow (in chemoinformatics)



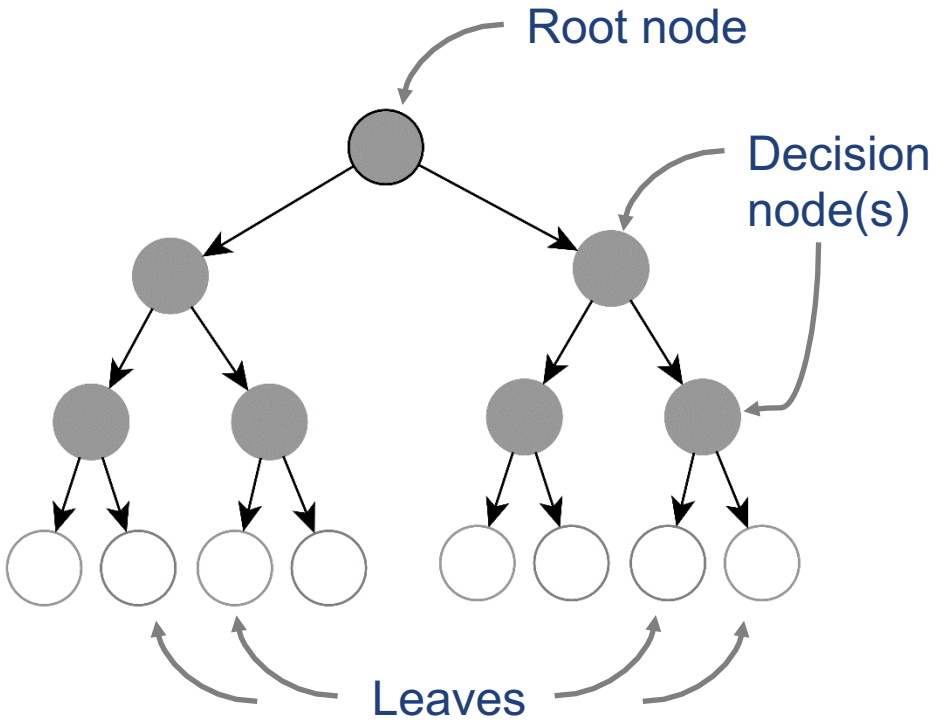
Standard Machine Learning workflow (in chemoinformatics)



Machine Learning methods
(overview)

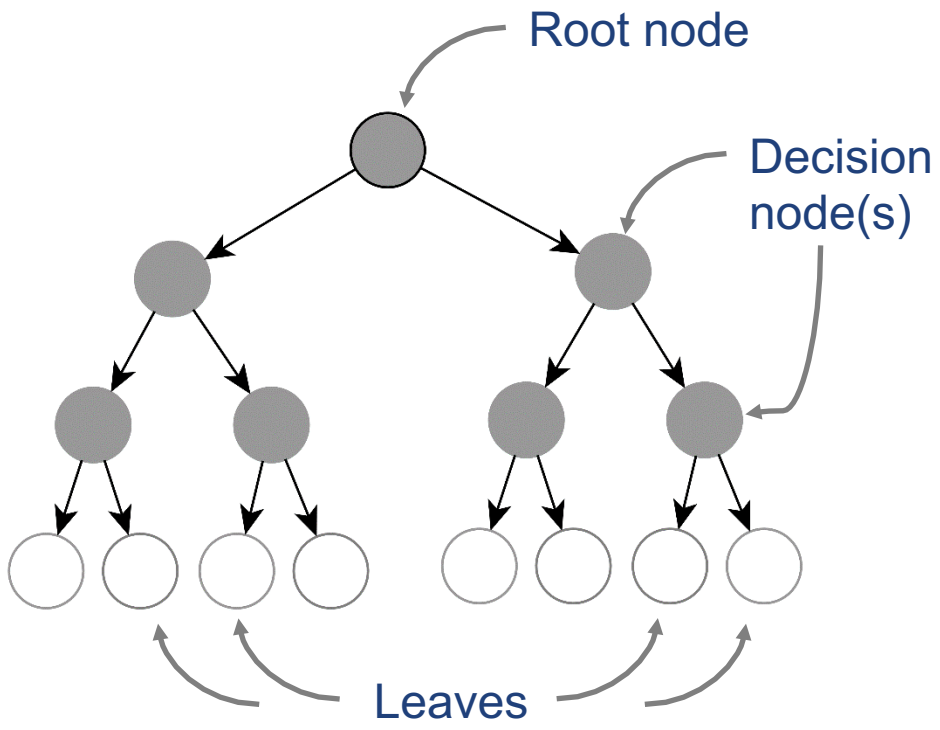
1. Decision Tree-based learning
    - Decision Trees
    - Random Forest
2. Local Methods
    - *k*-means algorithm
    - *k*-NN algorithm
3. Artificial Neural Networks
    - Feed-Forward NN
    - Kohonen Maps

(1) Decision Tree Learning



(1) Decision Tree Learning

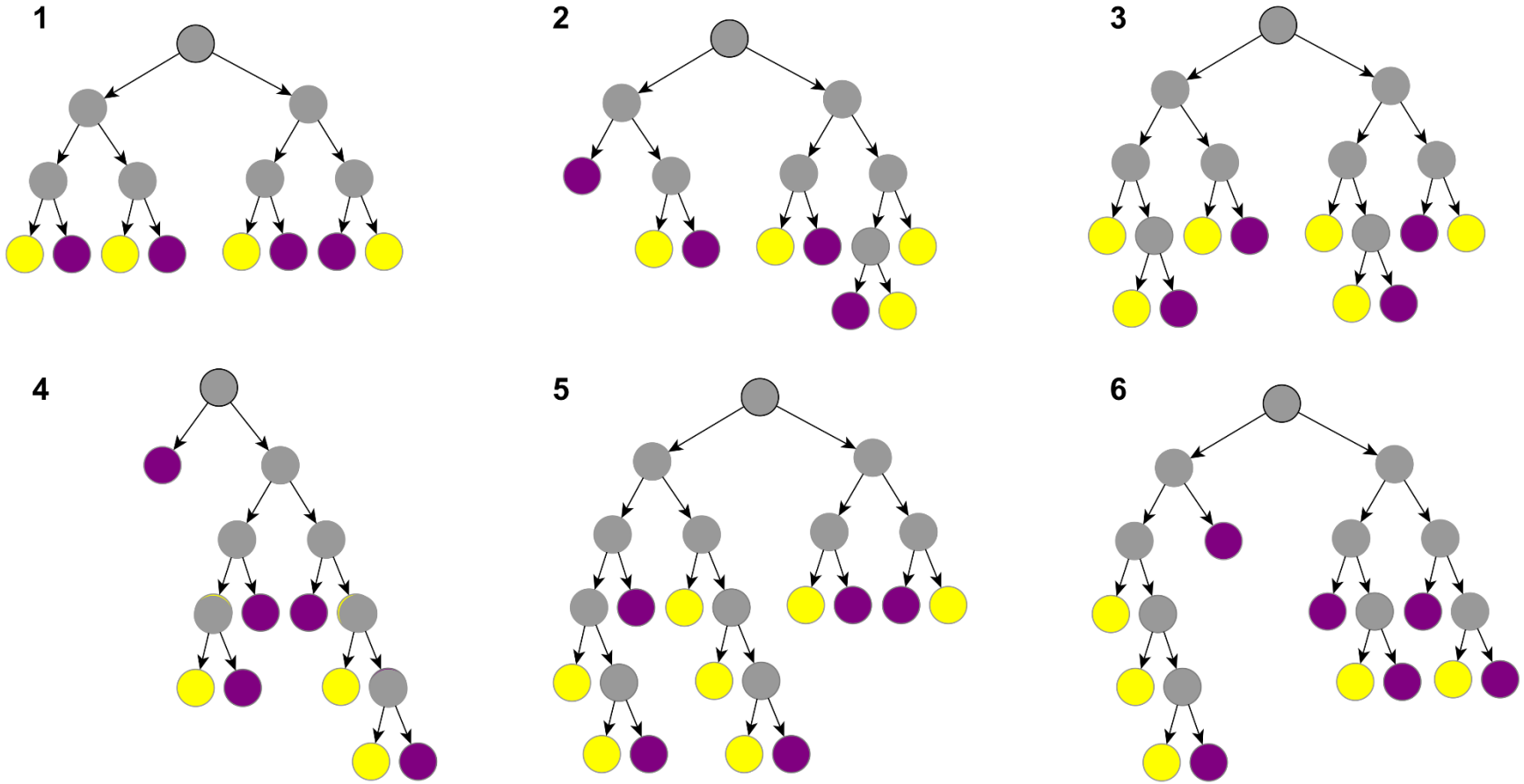
1. Easy to interpret
2. No data pretreatment
3. Numerical/categorical variables
4. Classification and regression
5. Non parametric
6. Automatic variable selection



(1) Decision Tree Learning

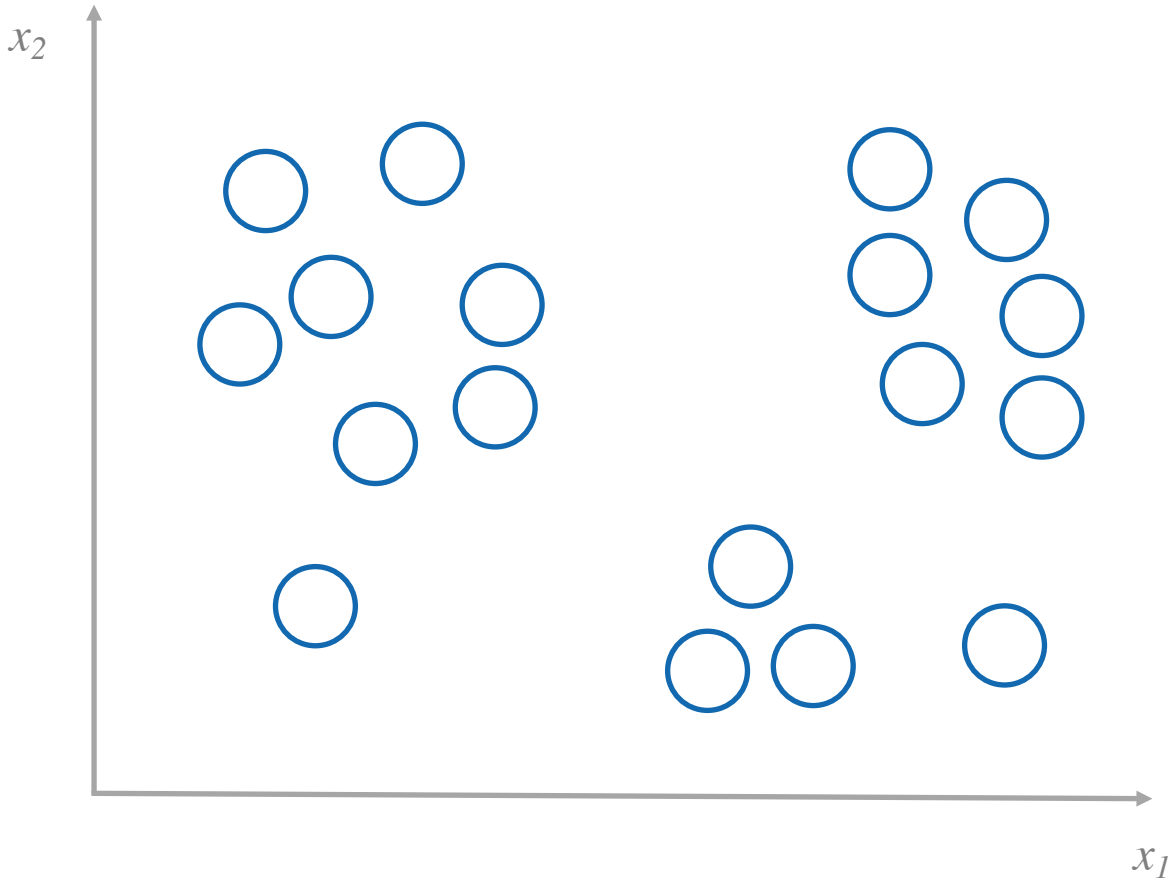
Random Forest

Bagging (**B**ootstrap **Agg**regat**ing**) = “the power of the crowd”



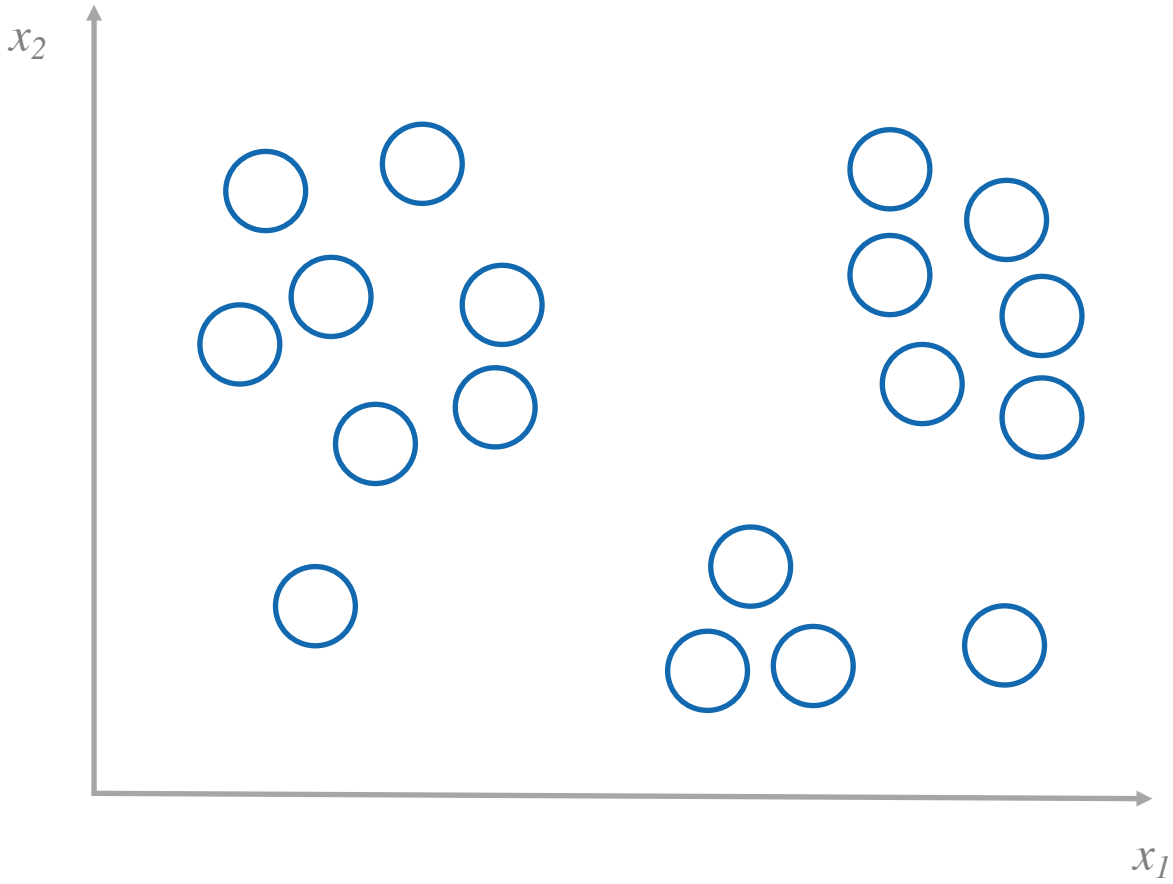
(2) Local approaches

k-Means clustering



(2) Local approaches

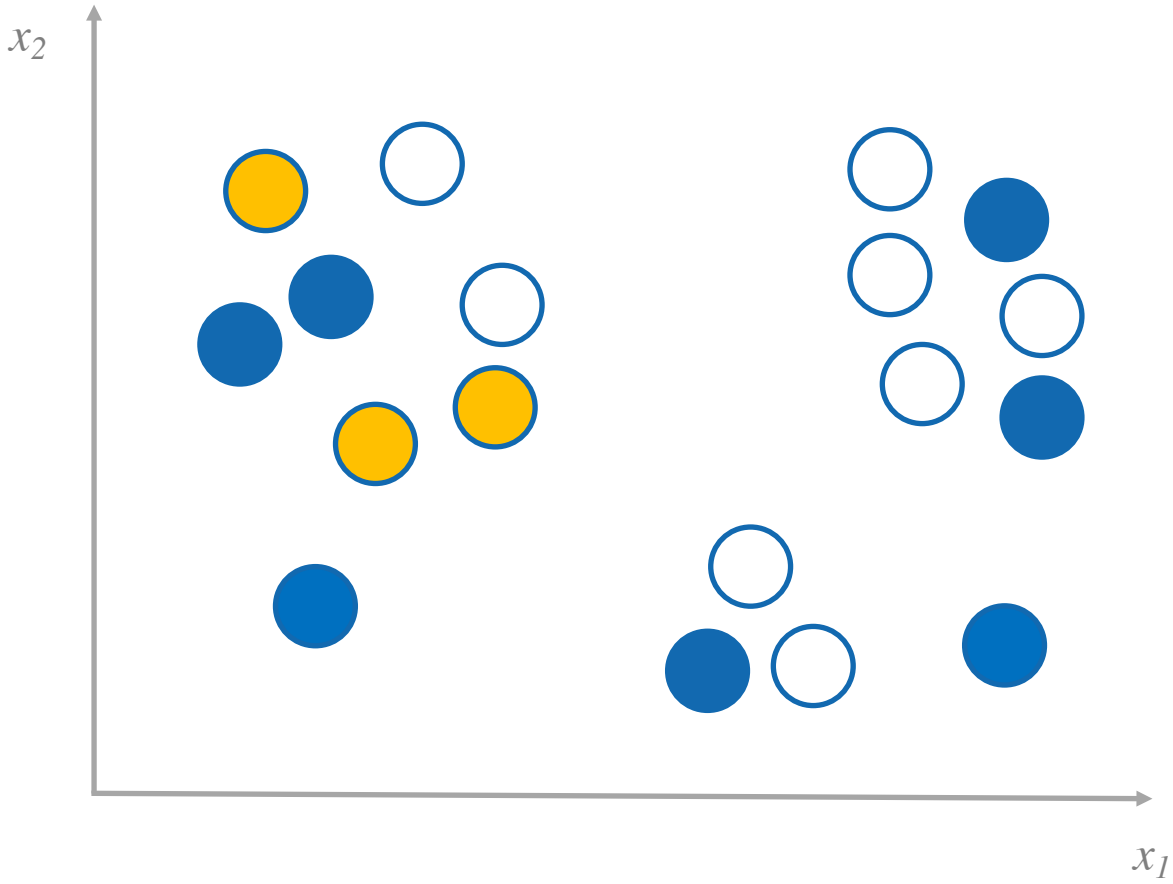
k-Means clustering



1. Select a *k* (**3**)

(2) Local approaches

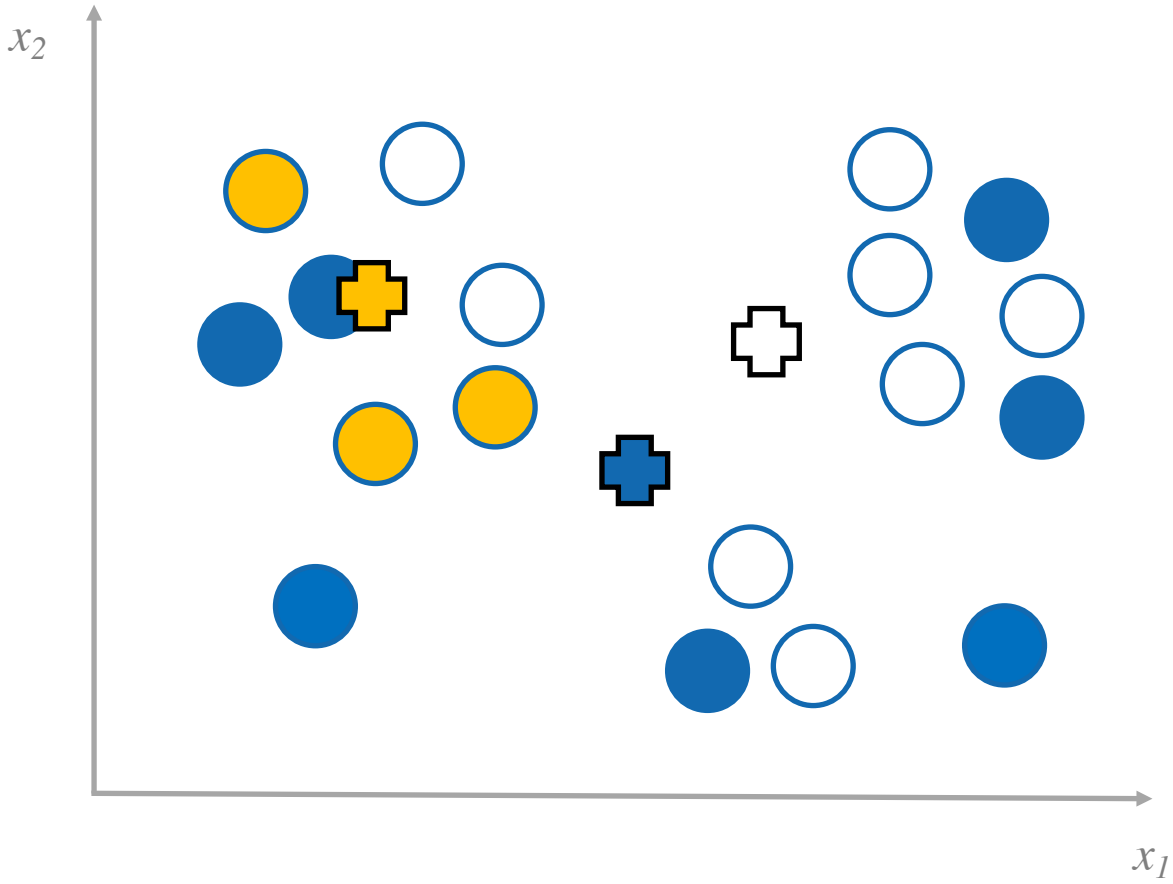
k-Means clustering



1. Select a *k* (**3**)
2. Random assignment

(2) Local approaches

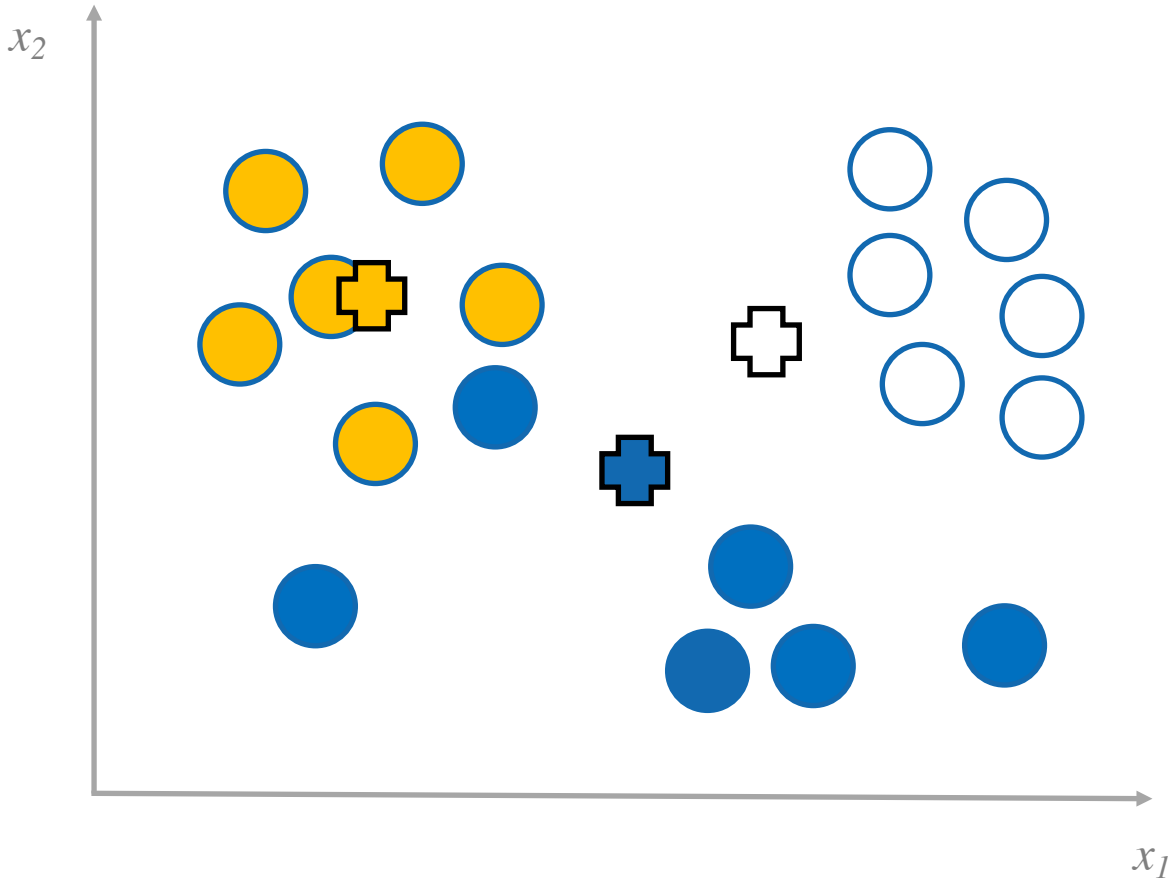
k-Means clustering



1. Select a *k* (**3**)
2. Random assignment
3. Centroid calculation

(2) Local approaches

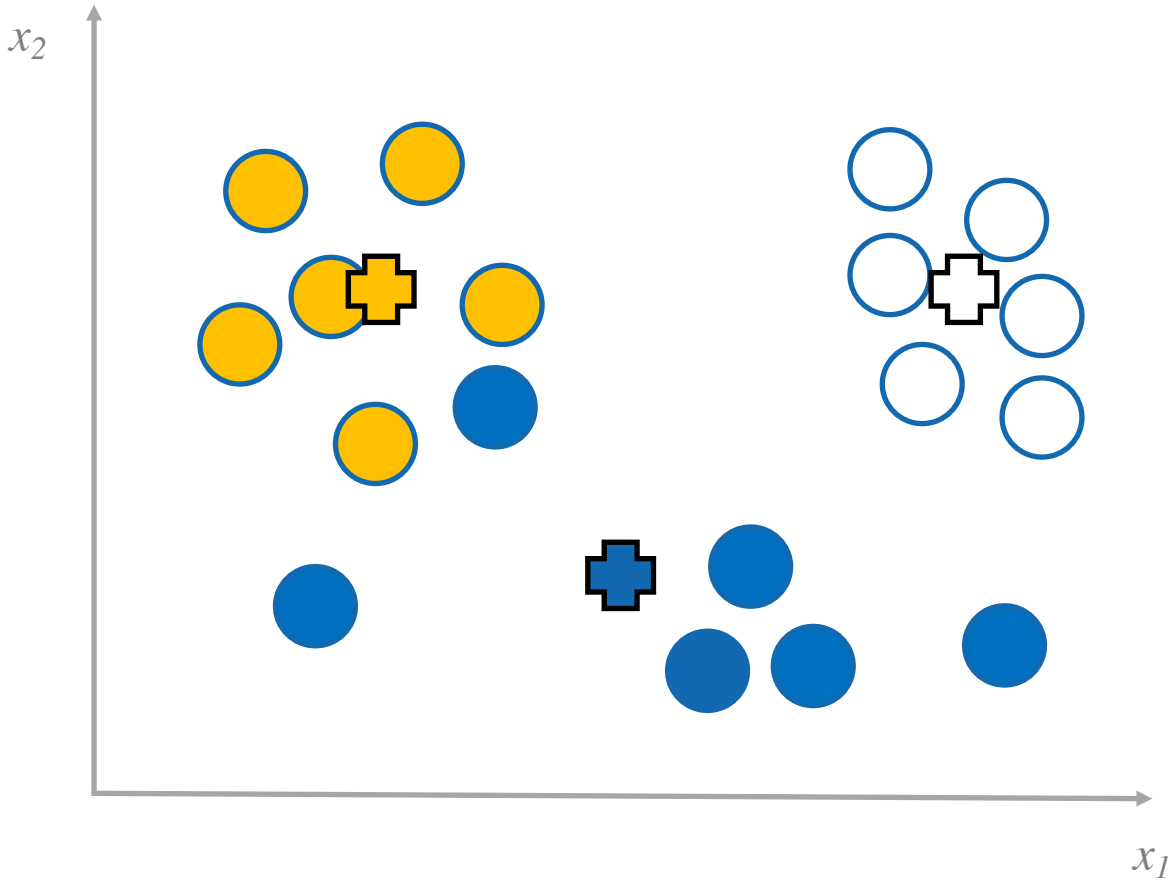
k-Means clustering



1. Select a *k* (**3**)
2. Random assignment
3. Centroid calculation
4. Closest centroid

(2) Local approaches

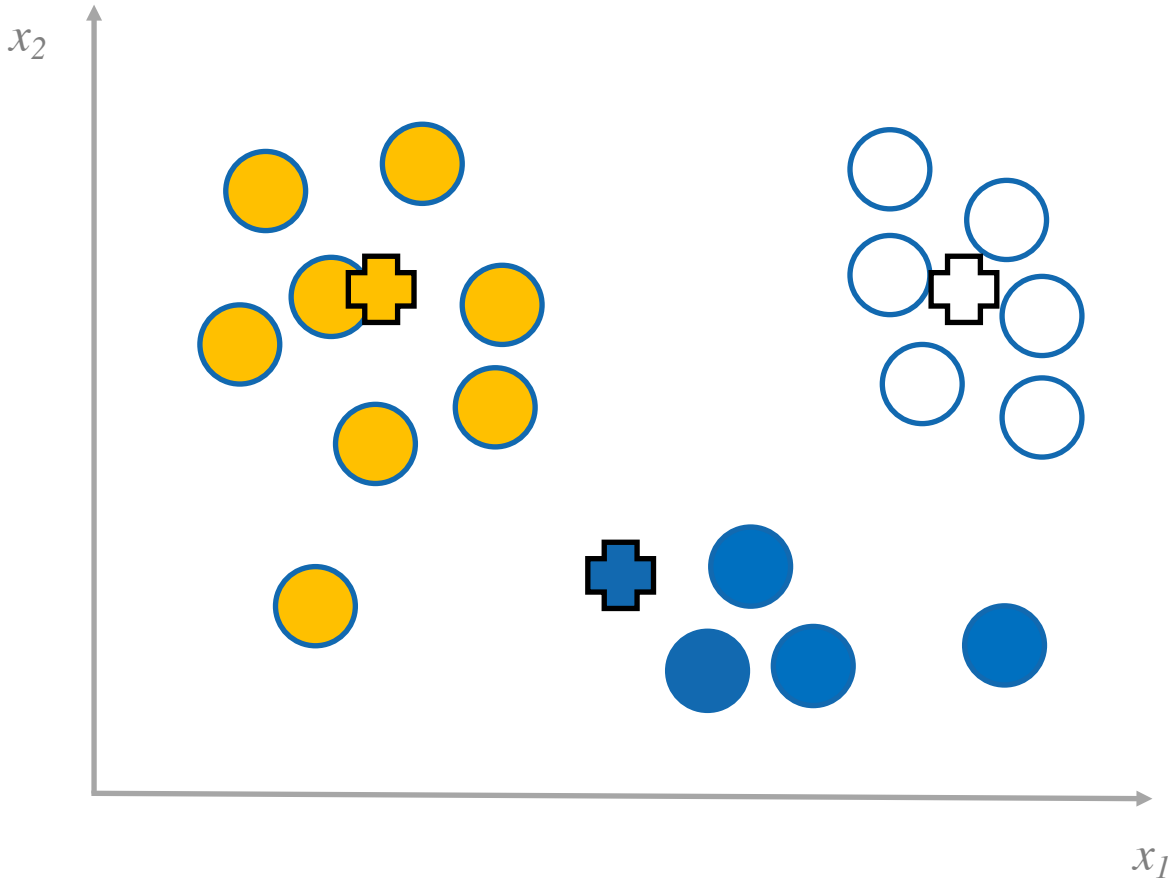
k-Means clustering



1. Select a *k* (**3**)
2. Random assignment
3. **Centroid calculation**
4. Closest centroid

(2) Local approaches

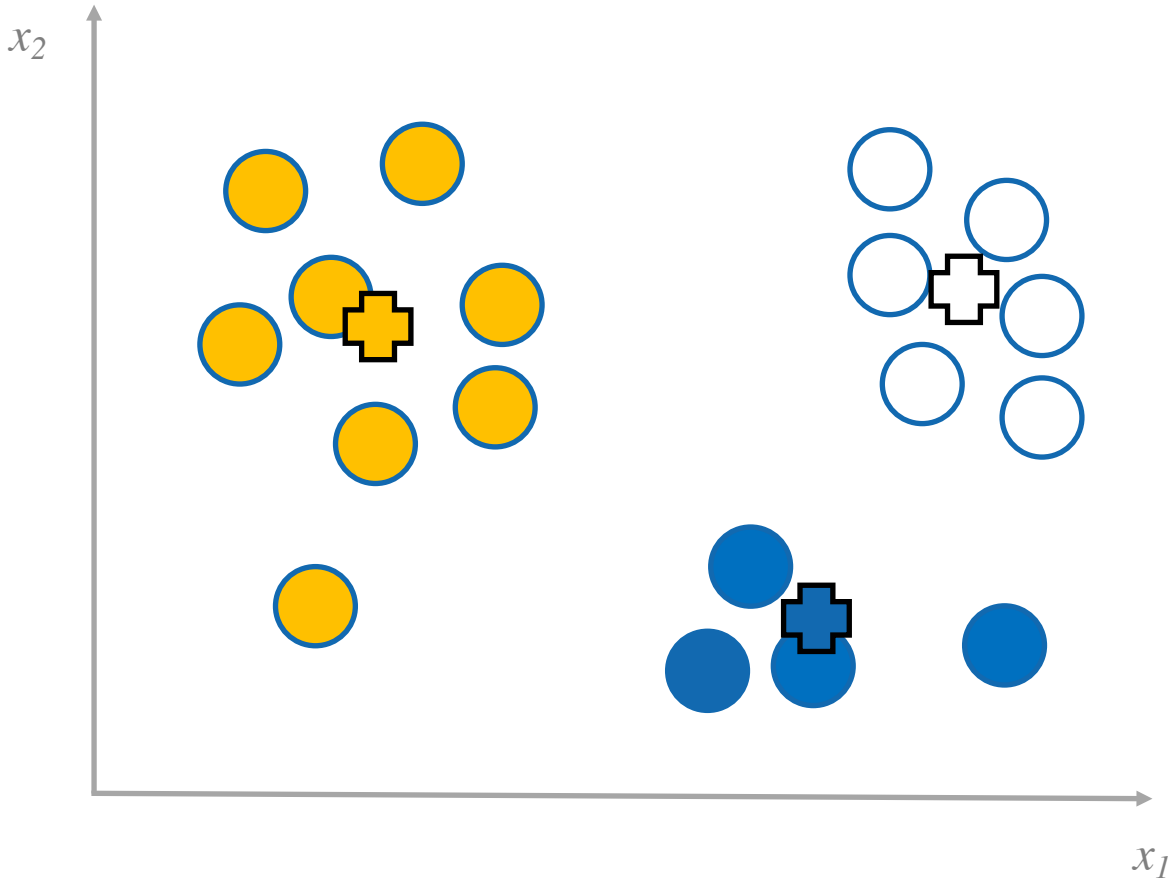
k-Means clustering



1. Select a *k* (**3**)
2. Random assignment
3. Centroid calculation
4. **Closest centroid**

(2) Local approaches

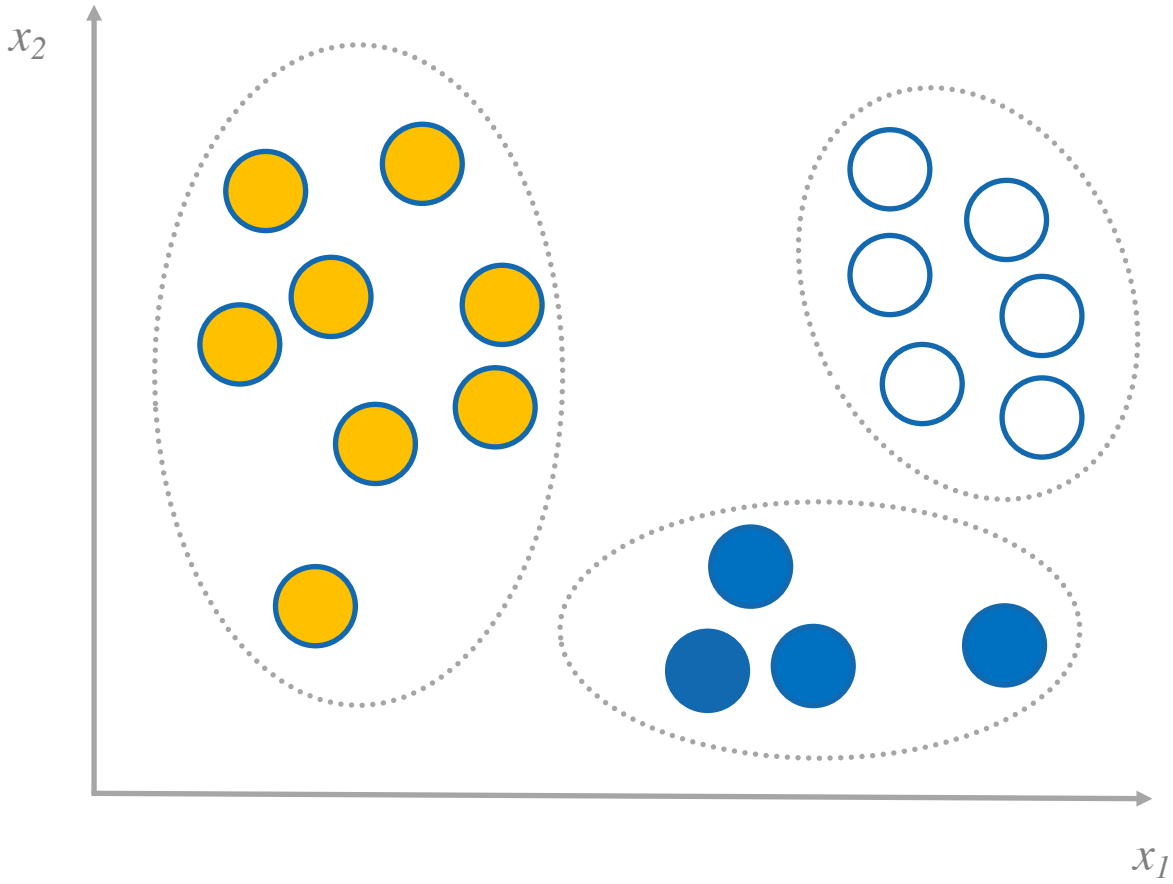
k-Means clustering



1. Select a *k* (**3**)
2. Random assignment
3. **Centroid calculation**
4. Closest centroid

(2) Local approaches

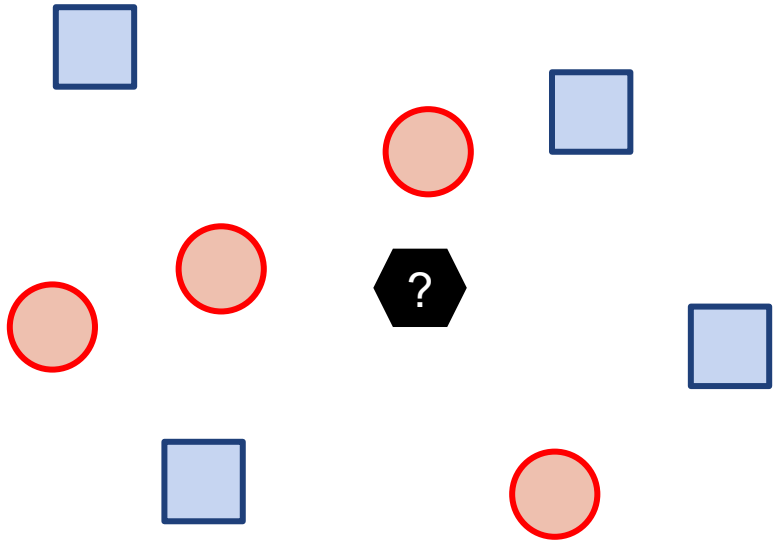
k-Means clustering



1. Select a *k* (**3**)
2. Random assignment
3. Centroid calculation
4. Closest centroid
5. End

(2) Local approaches

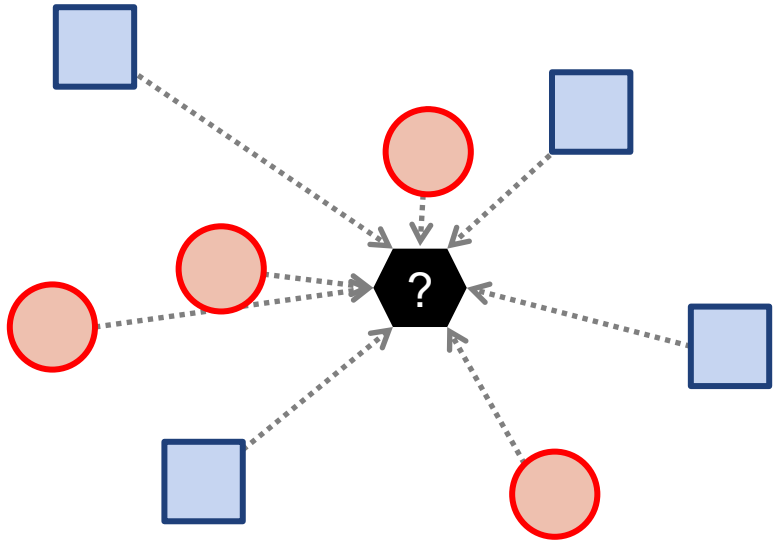
k-Nearest Neighbor (kNN)



(2) Local approaches

k-Nearest Neighbor (kNN)

1. Calculate a distance (n_{train} times)

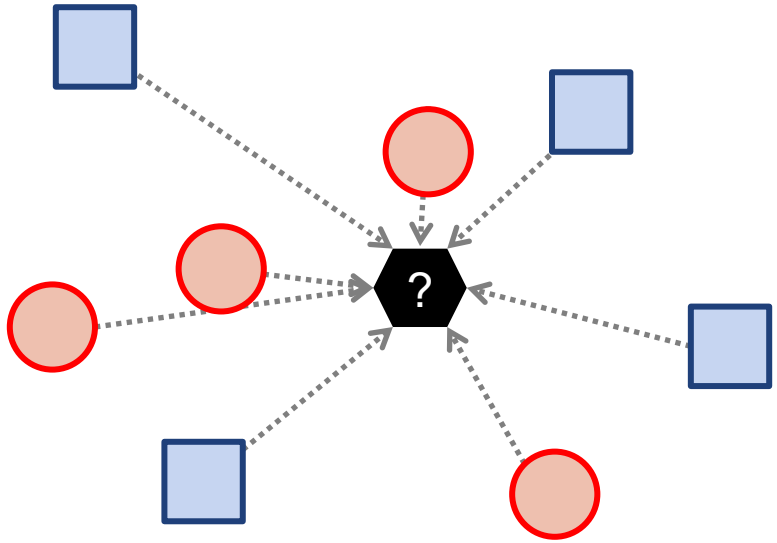


(2) Local approaches

k-Nearest Neighbor (kNN)

1. Calculate a distance (n_{train} times)

2. Select a number of neighbors (***k***) to predict the response

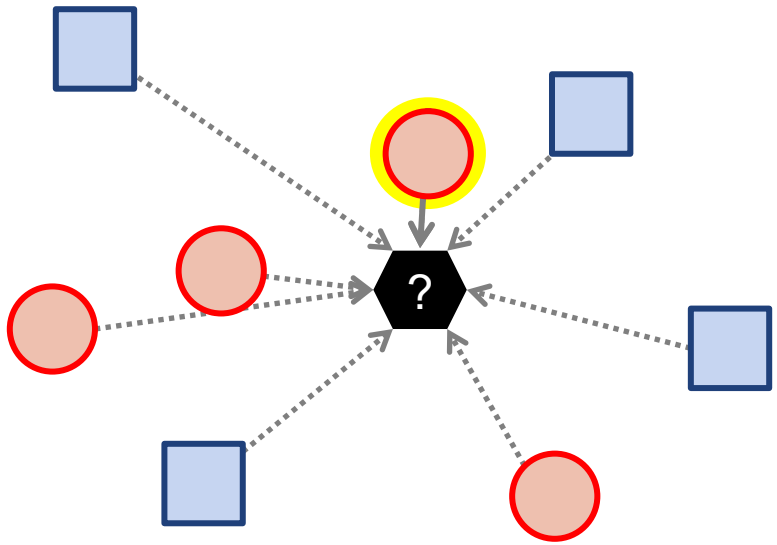


(2) Local approaches

k-Nearest Neighbor (kNN)

1. Calculate a distance (n_{train} times)

2. Select a number of neighbors (***k***) to predict the response



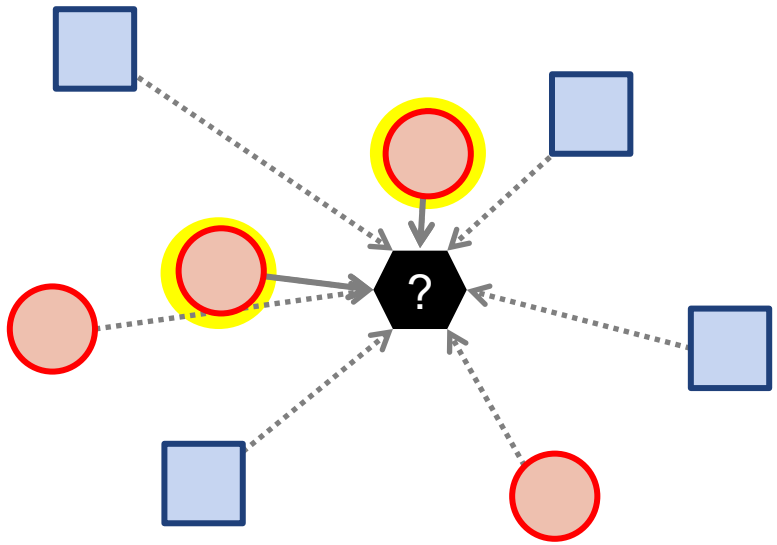
k = 1

(2) Local approaches

k-Nearest Neighbor (kNN)

1. Calculate a distance (n_{train} times)

2. Select a number of neighbors (***k***) to predict the response



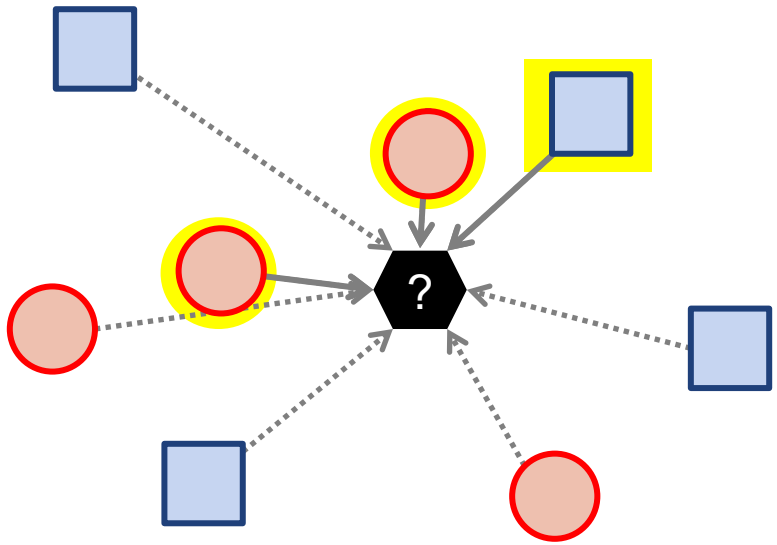
k = 2

(2) Local approaches

k-Nearest Neighbor (kNN)

1. Calculate a distance (n_{train} times)

2. Select a number of neighbors (***k***) to predict the response



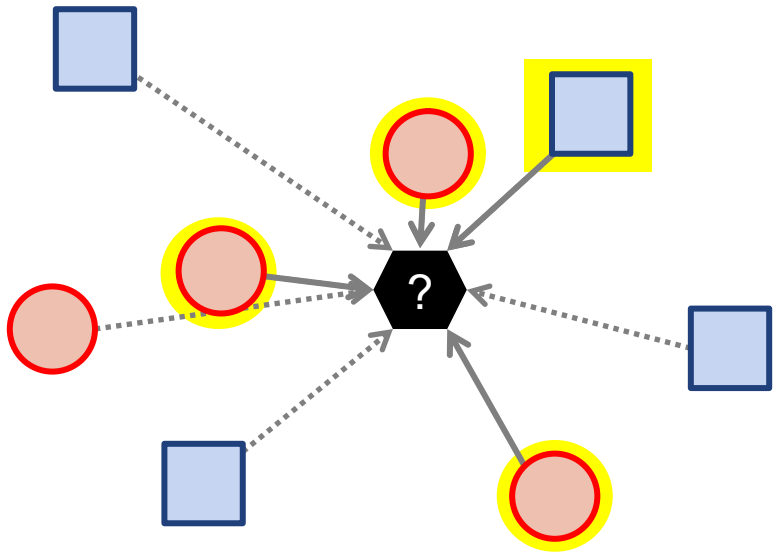
k = 3

(2) Local approaches

k-Nearest Neighbor (kNN)

1. Calculate a distance (n_{train} times)

2. Select a number of neighbors (***k***) to predict the response



k = 4

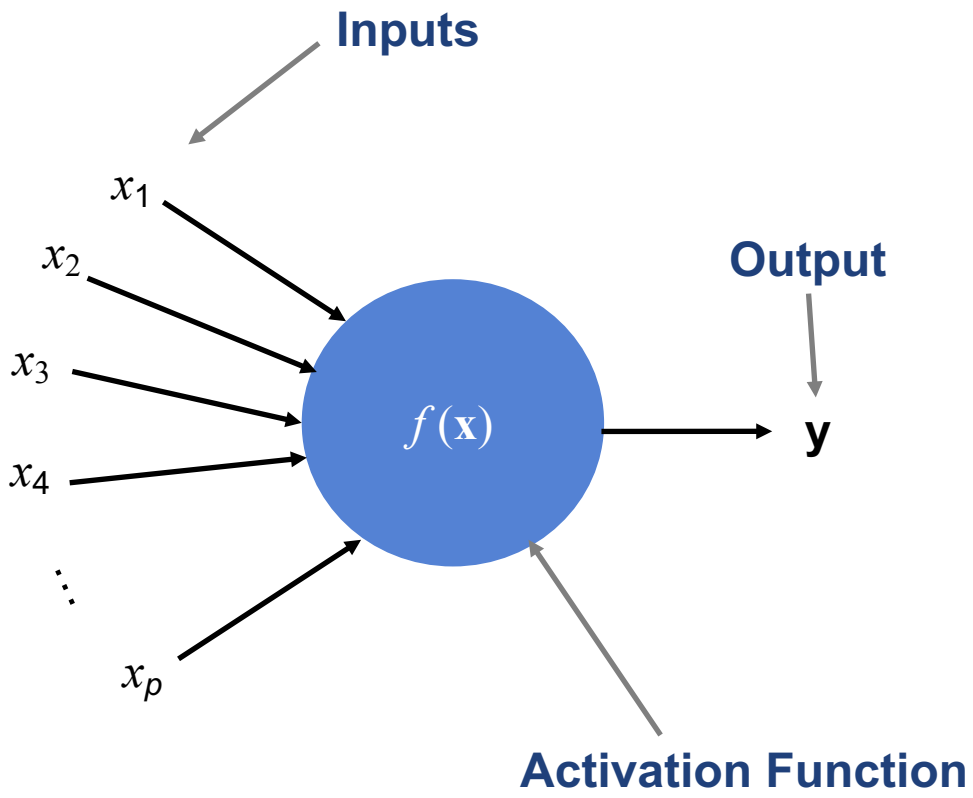
(2) Local approaches

k-Nearest Neighbor (kNN)

1. Good for large training set with “localized” differences
2. Difficult to be interpreted
3. Which *k?*
4. Which distance measure?
5. Curse of dimensionality → Variable selection

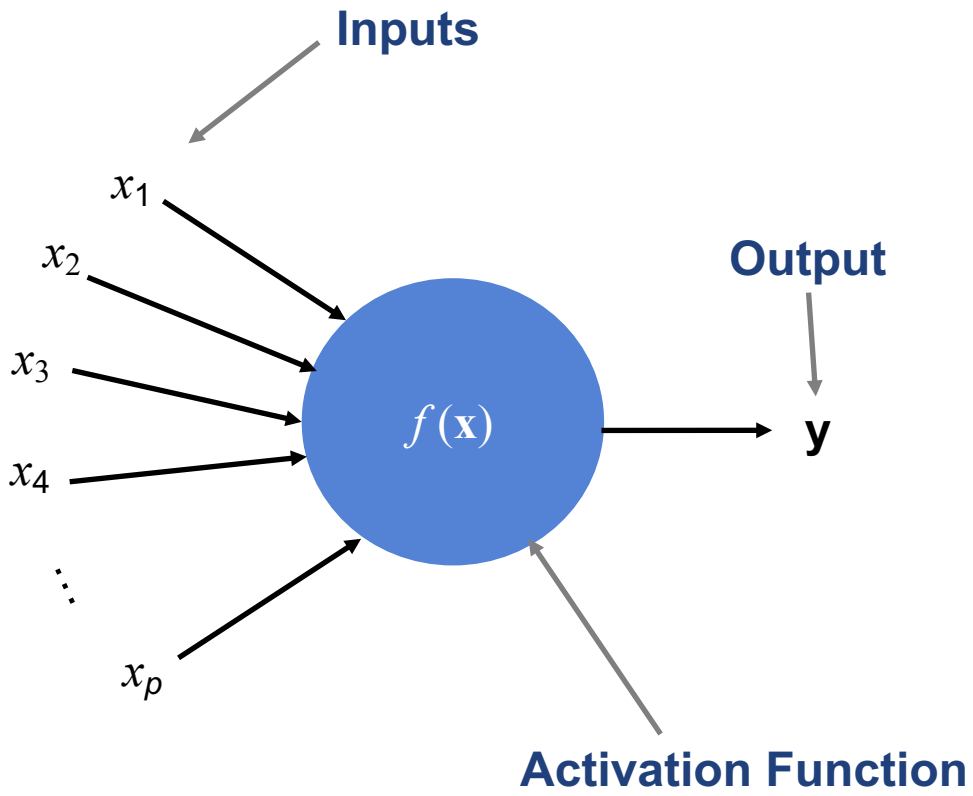
(3) Neural Networks

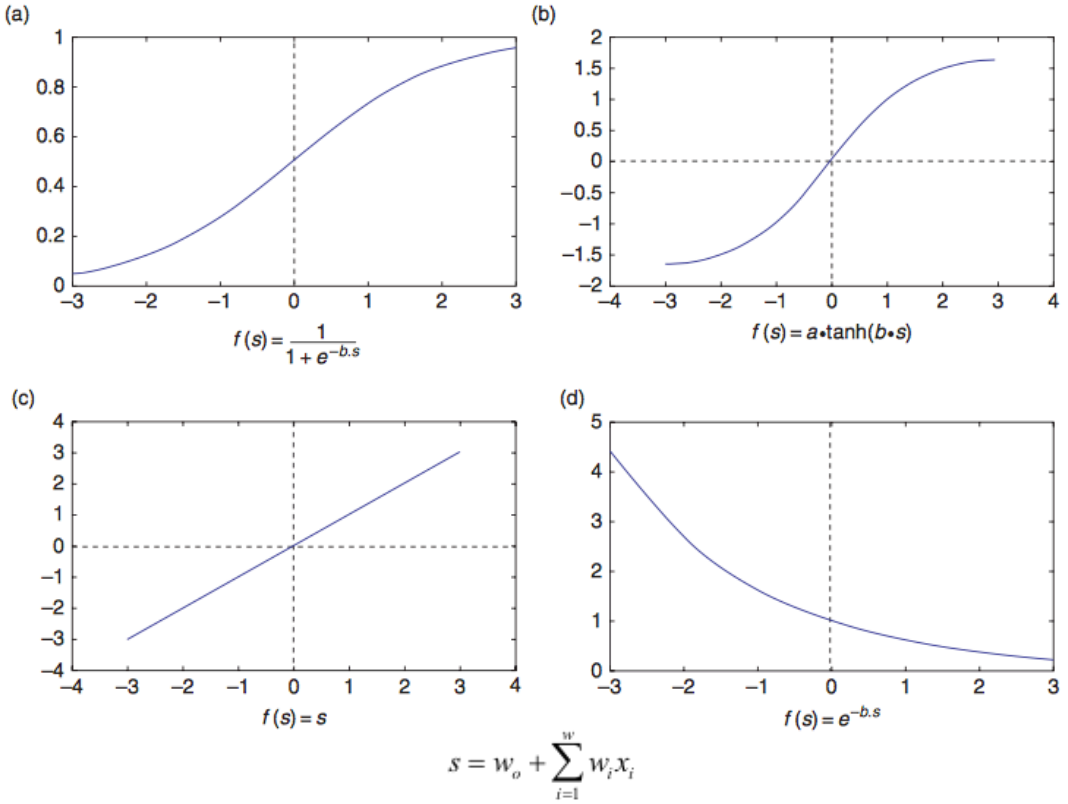
Artificial Neurons



(3) Neural Networks

Artificial Neurons

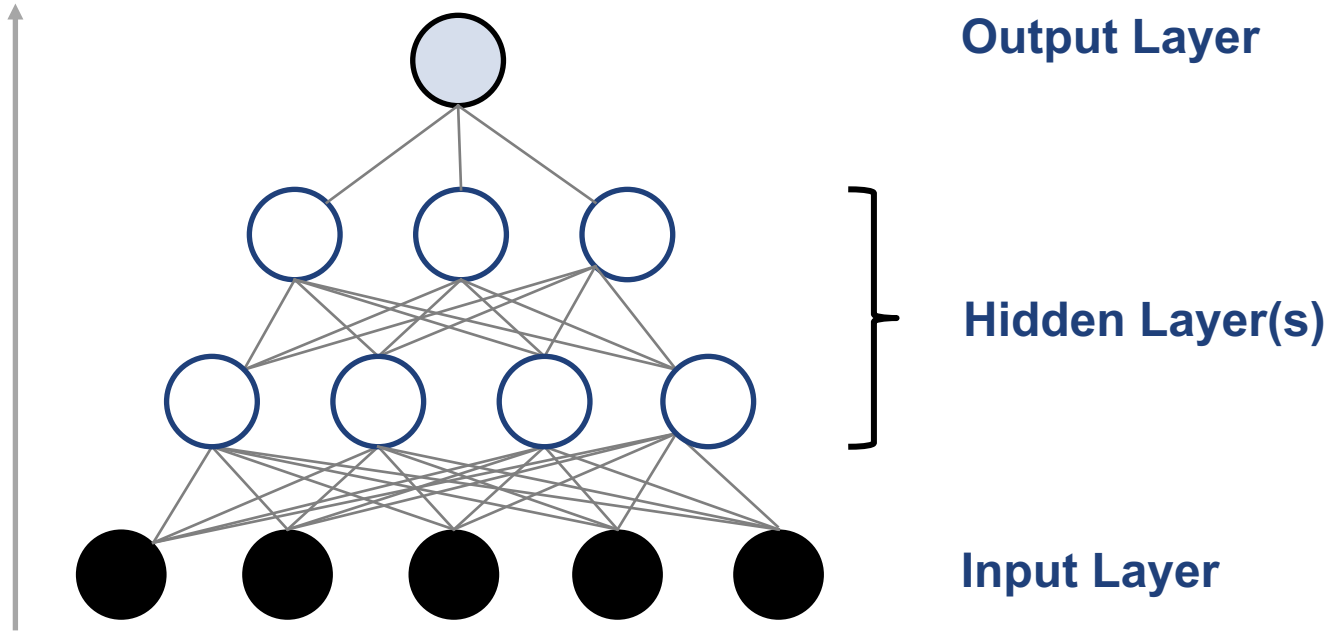




Neural networks. Comprehensive Chemometrics: Chemical and Biochemical Data Analysis Vol, 3.

(3) Neural Networks

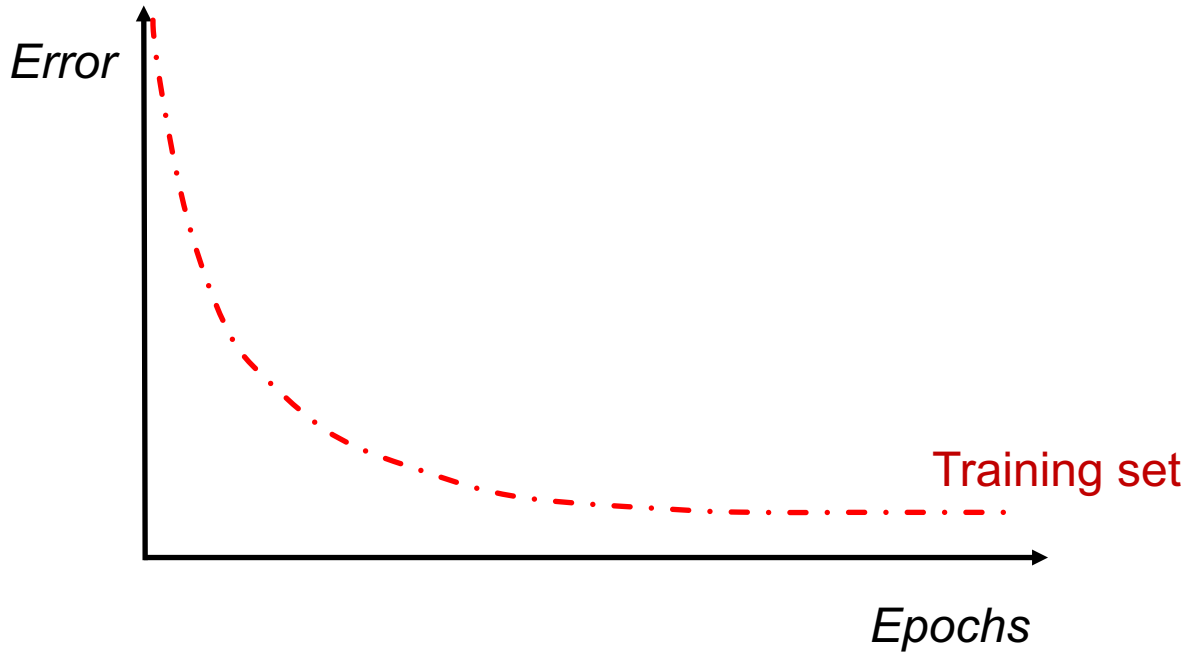
Feed-Forward NN



1. Untrained Network
2. Compute the outcome
3. Compute the error
4. Back propagation learning
5. Repeat until stop criterion

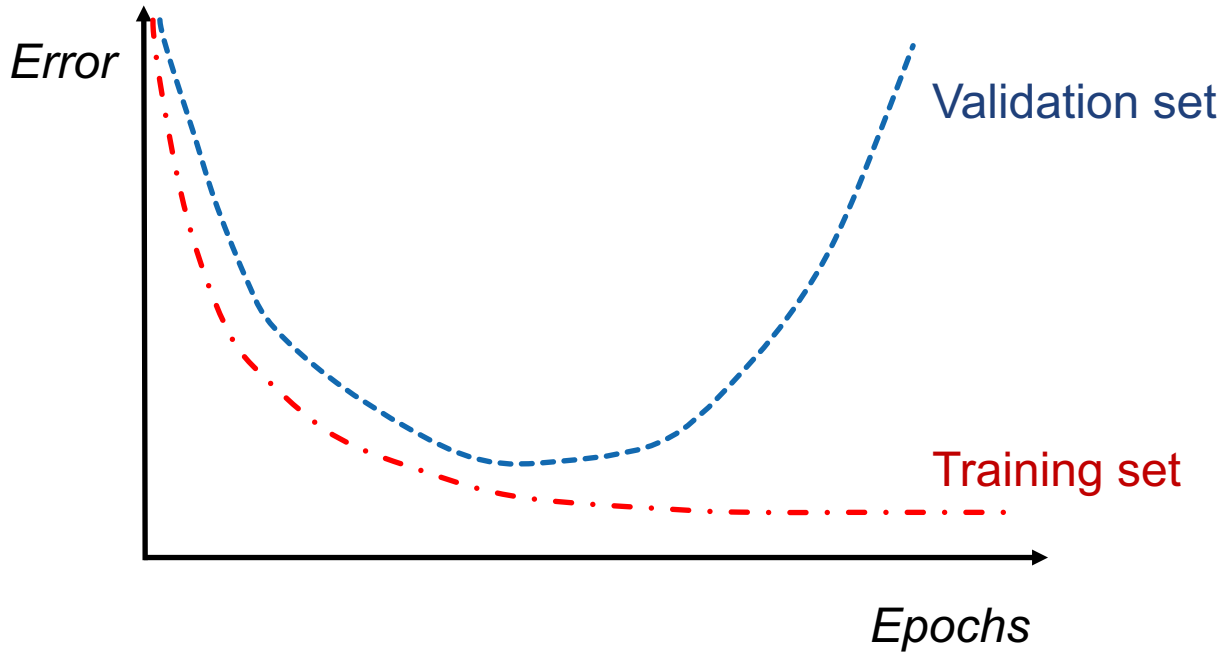
(3) Neural Networks

Feed-Forward NN



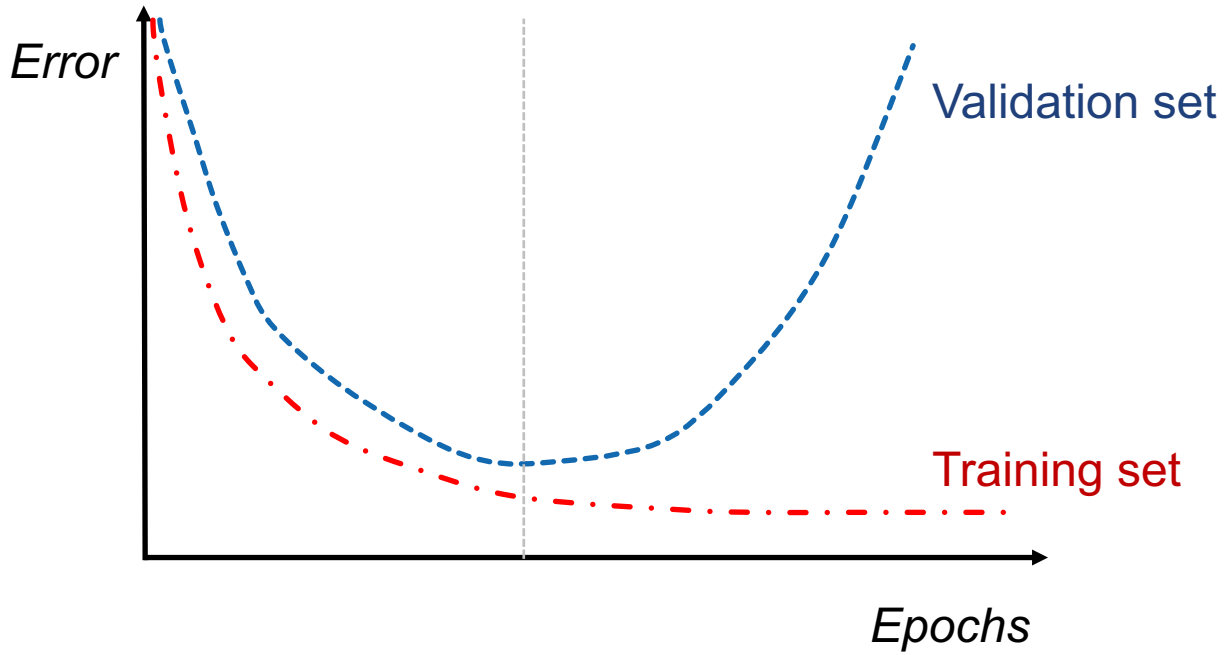
(3) Neural Networks

Feed-Forward NN



(3) Neural Networks

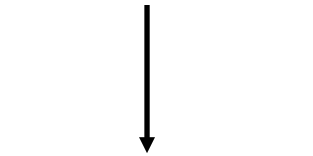
Feed-Forward NN



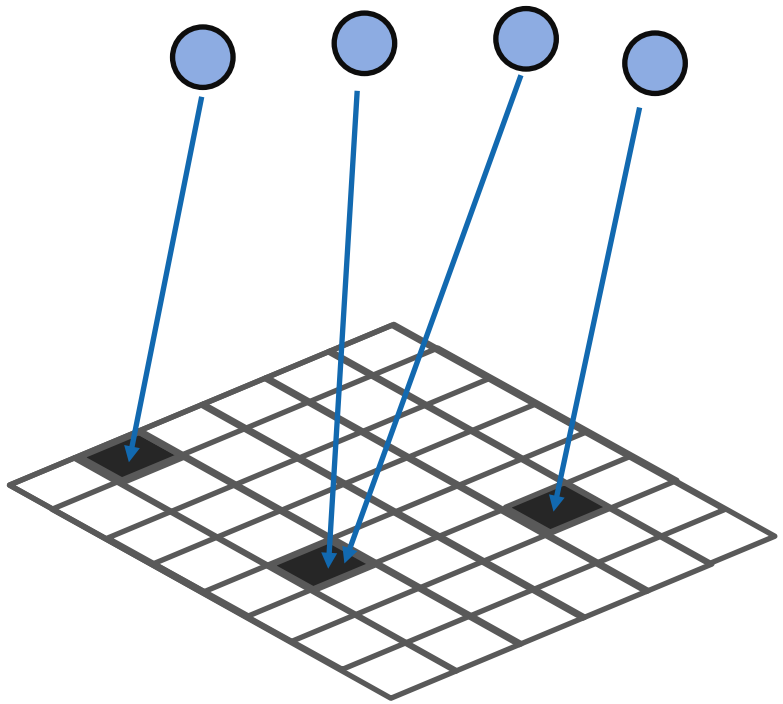
(3) Neural Networks

Kohonen Maps

p dimensional



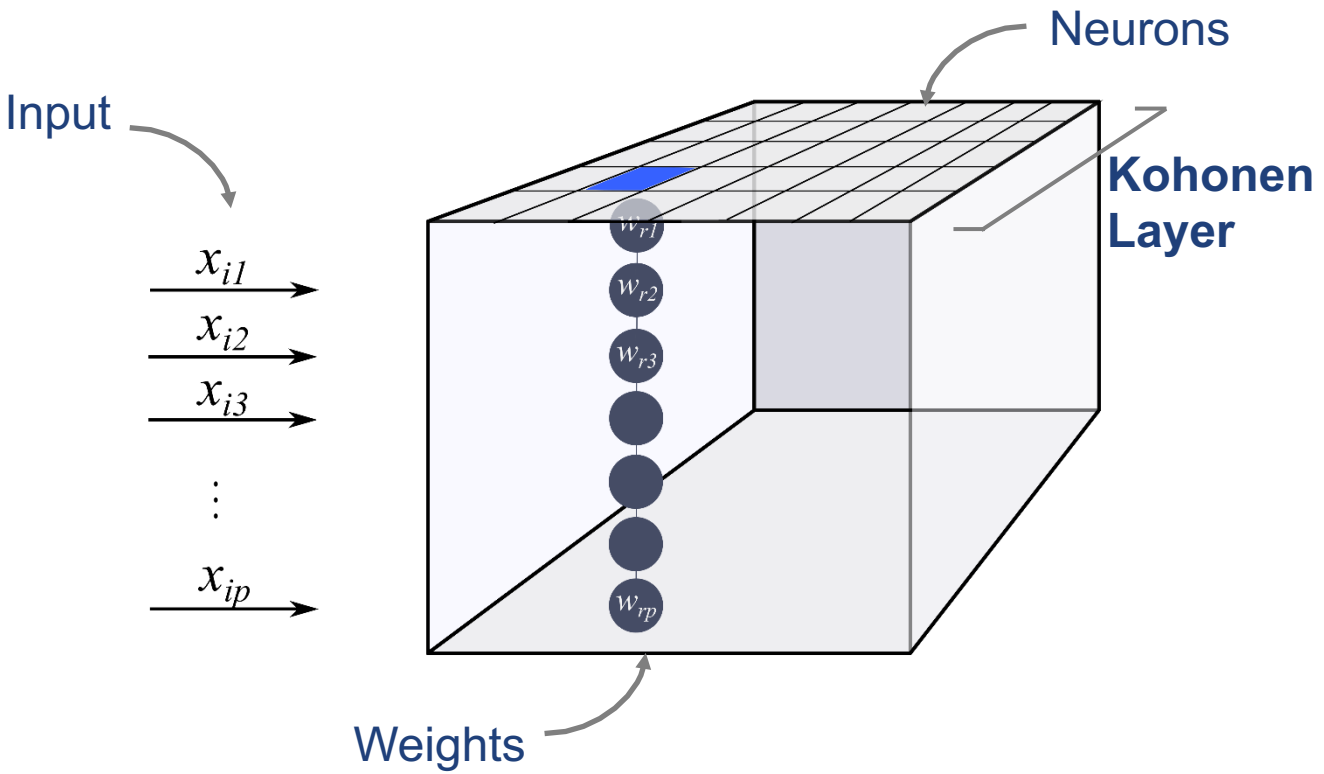
2 dimensional



- Unsupervised non-linear mapping
- Topology preserving map

(3) Neural Networks

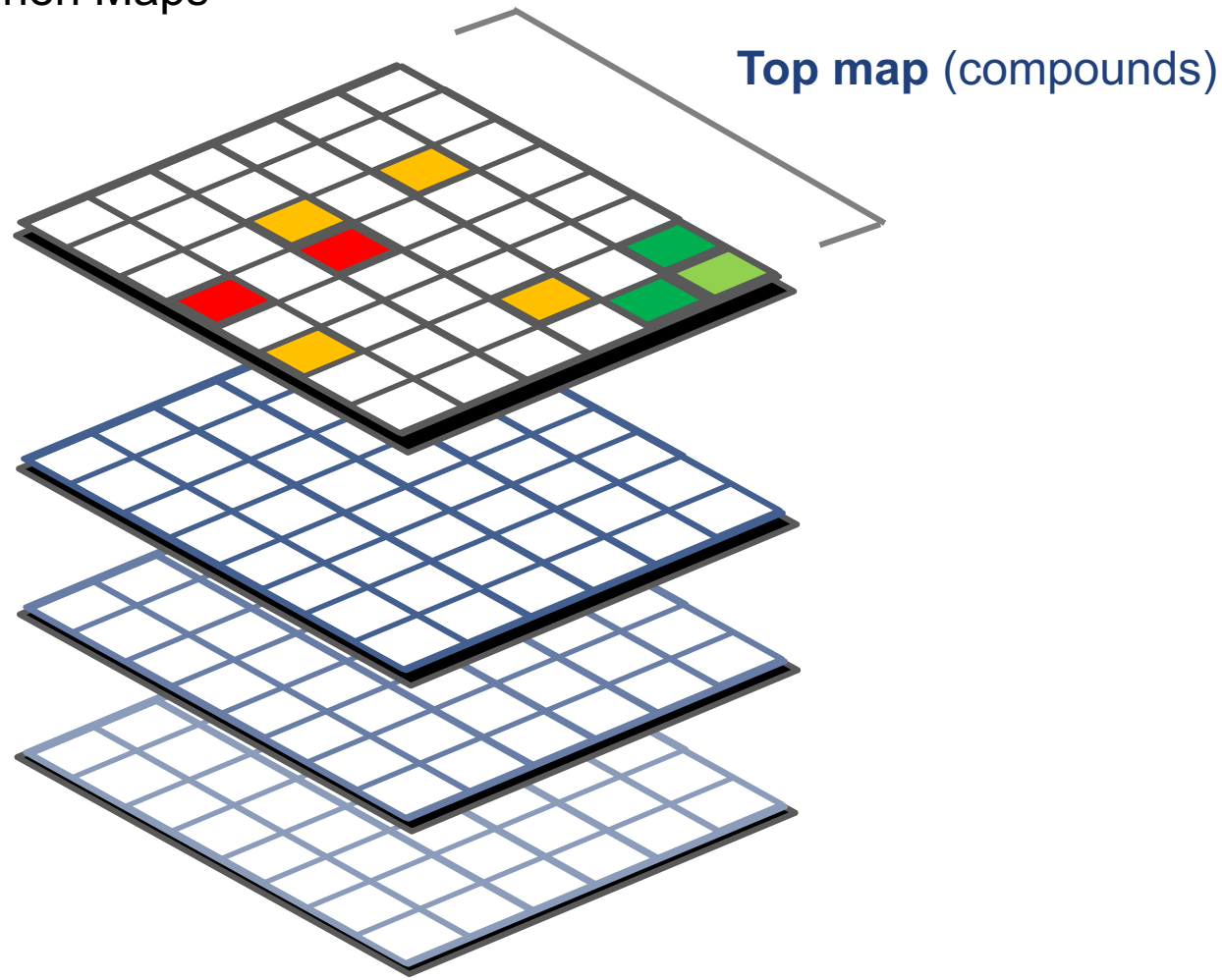
Kohonen Maps



1. **Competitive Learning**
   - Similarity to each neuron
   - "Winner takes all"
2. **Collaborative Learning**
   - Winning neuron update
   - Update of close neurons

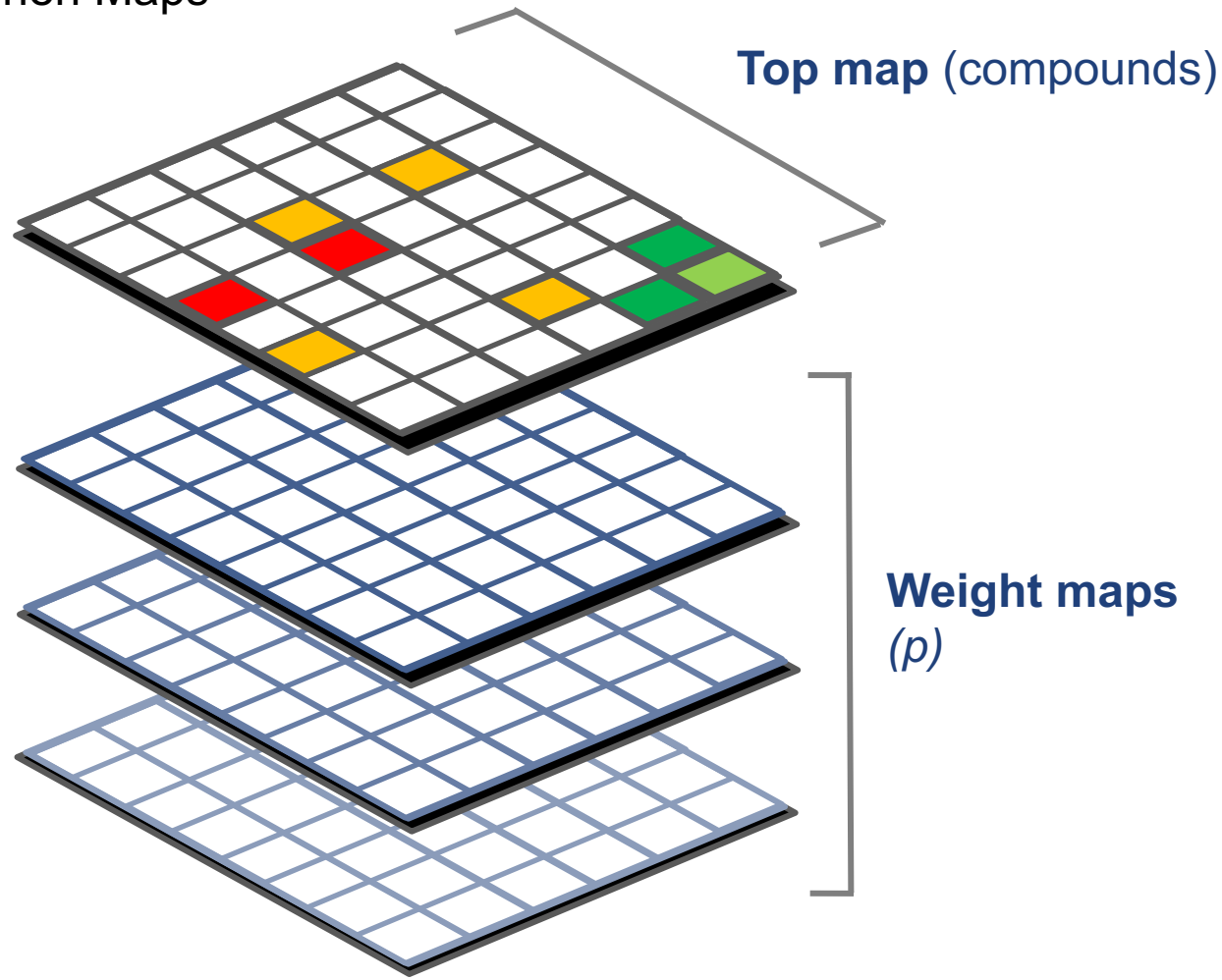
(3) Neural Networks

Kohonen Maps



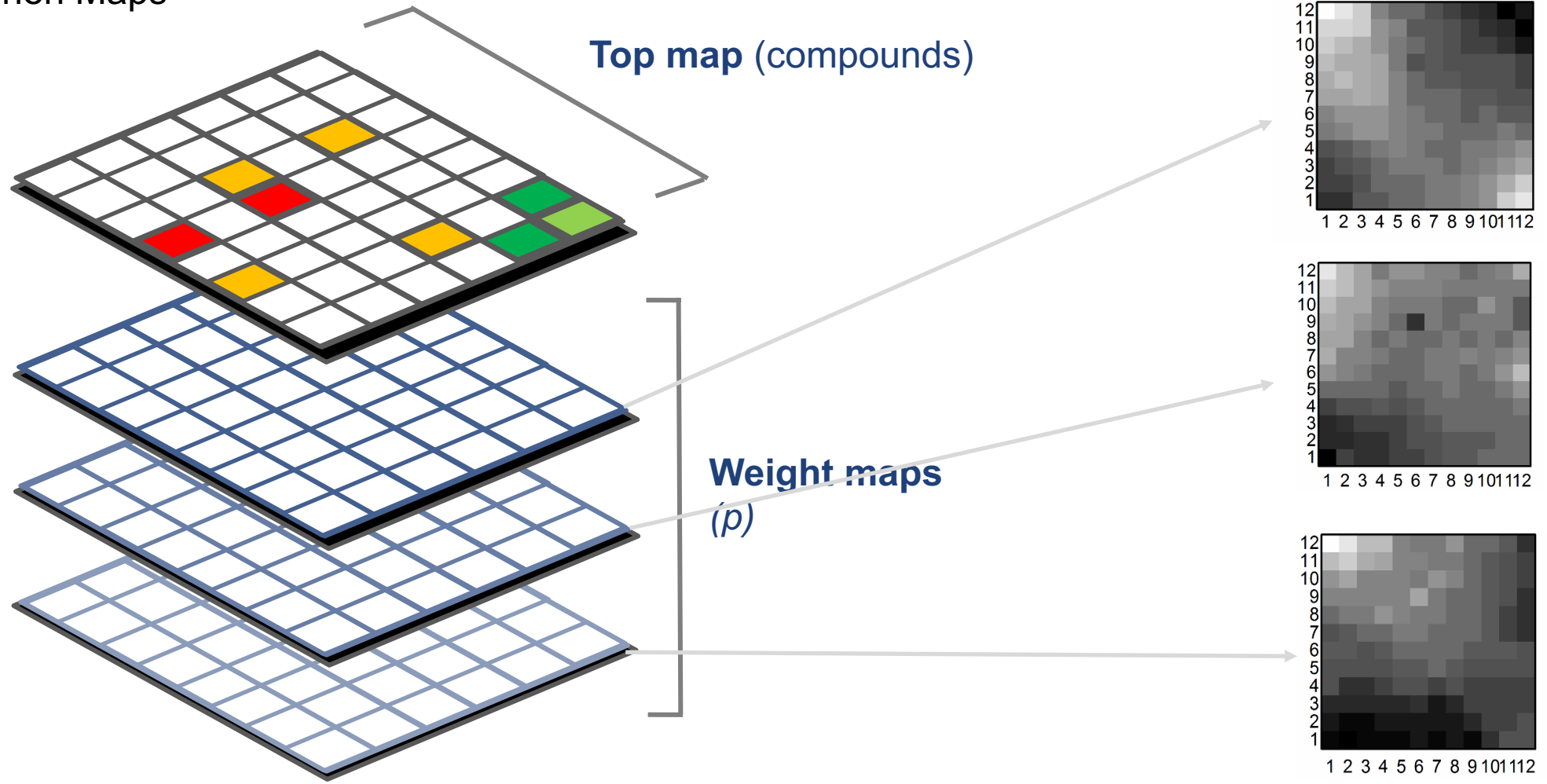
(3) Neural Networks

Kohonen Maps



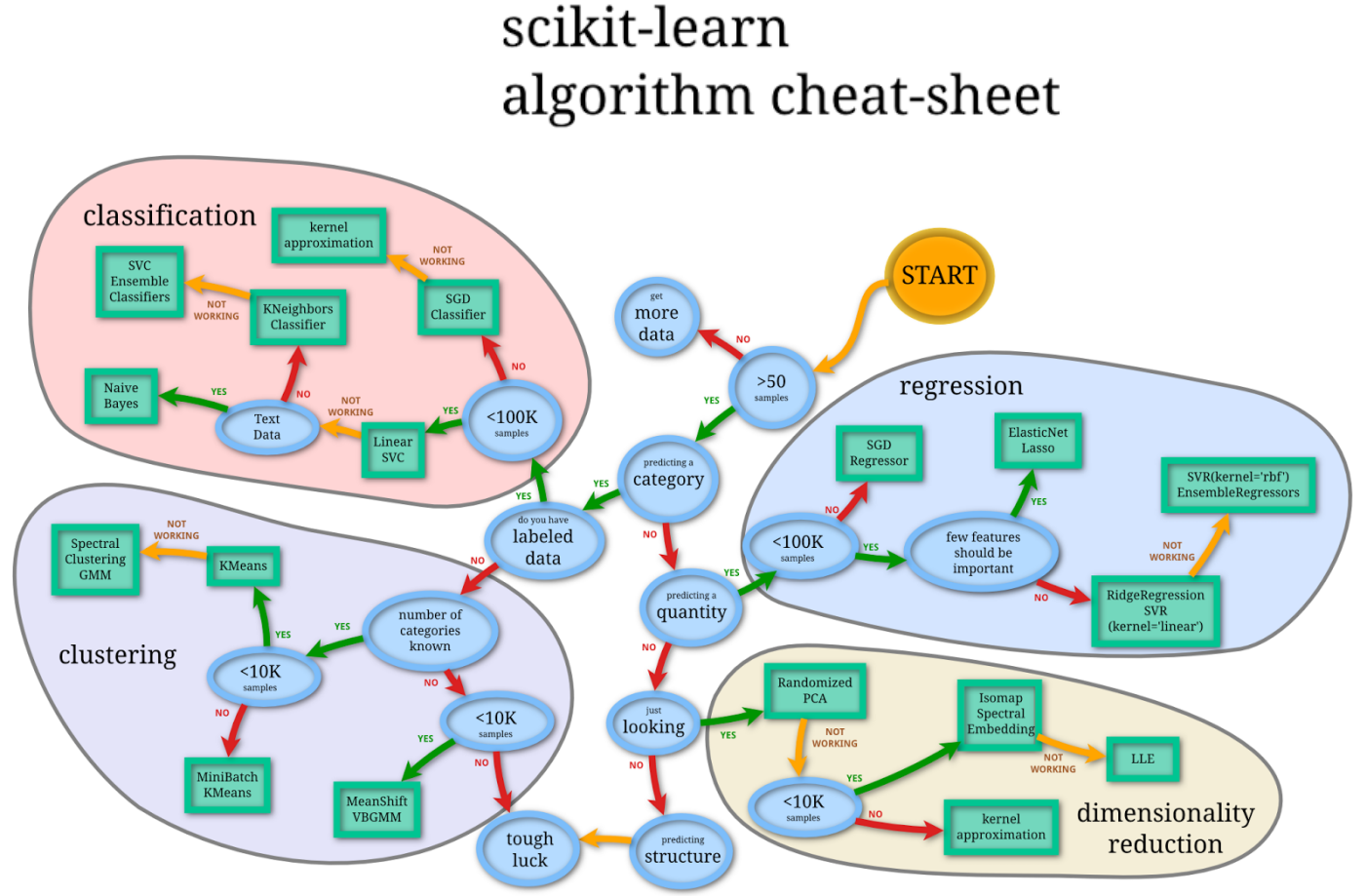
(3) Neural Networks

Kohonen Maps



Which ML algorithm?

- Purpose (clustering, regression, classification)
- Performance *vs* interpretability
- Covered chemical space (*e.g.*, AD)
- Types of included variables
- …



http://scikit-learn.org/stable/tutorial/machine_learning_map/

Summary

- Machines can learn from our data
- No ML algorithm always outperforms the others
- Validation and Applicability Domain assessment are crucial
- Pay attention to what the performance metric is telling you!

Supplementary reading

Theory and Algorithms

- Shalev-Shwartz, S., & Ben-David, S. (2014). “Understanding machine learning: From theory to algorithms.” Cambridge university press.
- Marini, F. (2009). “Neural networks”. In: *Comprehensive Chemometrics: Chemical and Biochemical Data Analysis* - Vol, 3.

Online resources

- [Coursera] Ng, A. Machine Learning, Stanford University. https://www.coursera.org/learn/machine-learning
- [Online Book] Neural Networks and Deep Learning. http://neuralnetworksanddeeplearning.com/