Machine Learning

Introduction

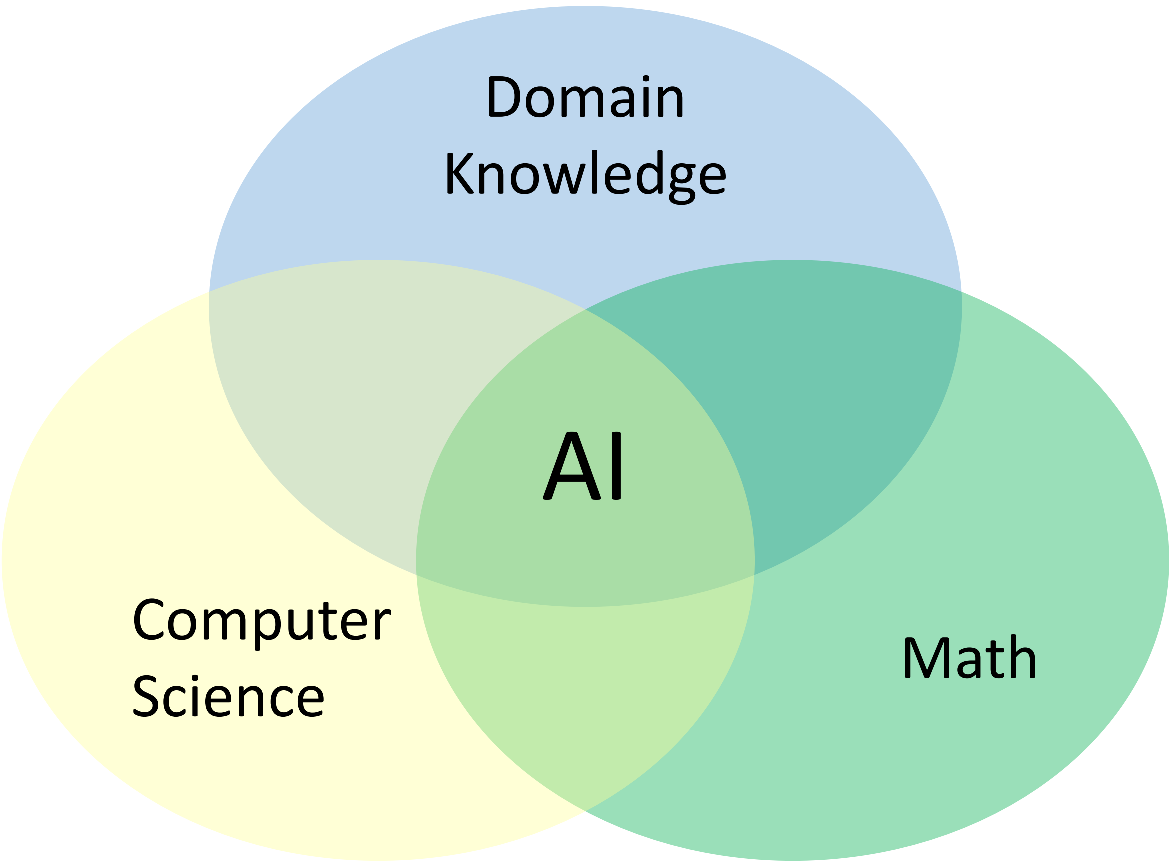
Prof. Chang-Chieh Cheng
Information Technology Service Center
National Chiao Tung University

What is AI?

- **AI, artificial intelligence**, is an area of computer science that emphasizes the creation of intelligent machines that work and react like humans.
  - For example, an AI machine is designed for
    - Speech recognition
    - Learning
    - Planning
    - Problem solving
- Two primary subareas of AI
  - **Machine learning**
  - **Artificial neural networks**

Artificial Intelligence Fundamentals

- Three primary fundamentals:



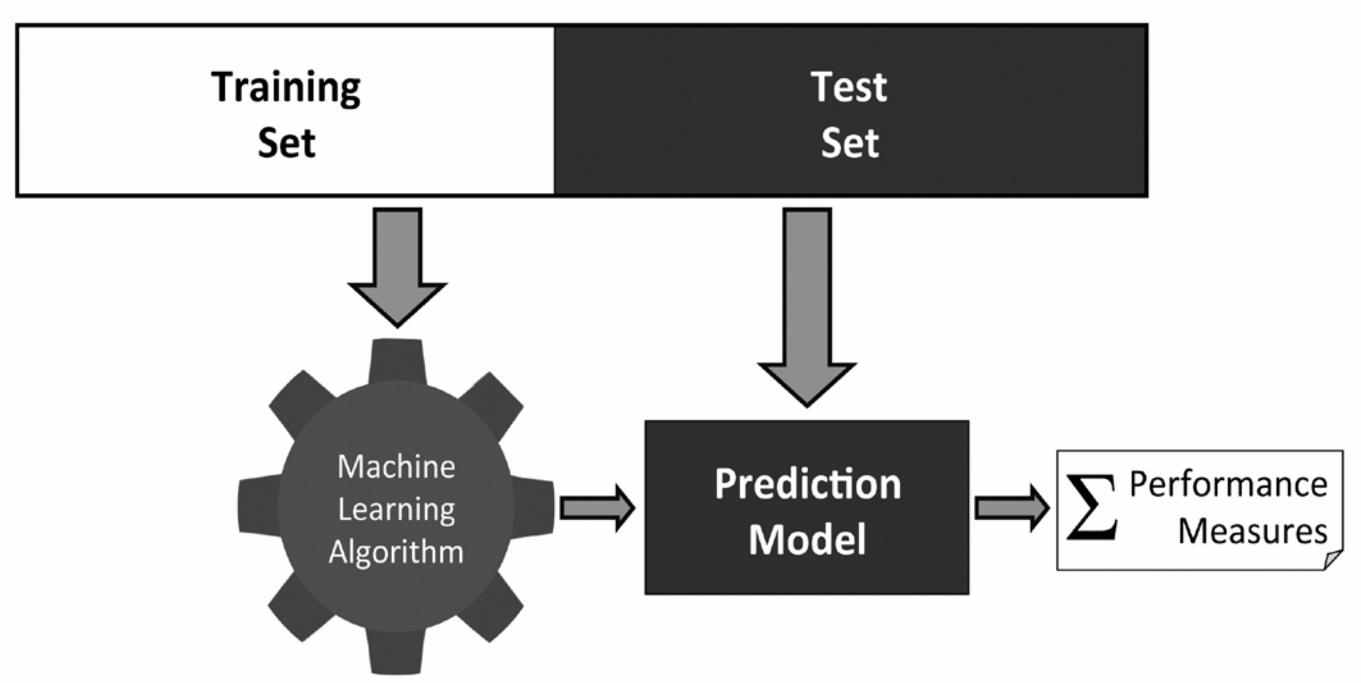
What is Machine Learning?

- A **computer program** can learn something from **experiences**, **observations**, or **historic data instances** to **predict an event**, **make a decision**, or **improve a behavior**.
- **Learning model**
  - A learning algorithm, a computer program
- **Prediction Model**
  - A trained learning model
- **Training data and Test Data**
  - experiences, observations, or historic data instances
  - **Training data**
    - It is used to train a learning model
  - **Test data**
    - It is used to verify the accuracy and performance of a learning model

Data Mining vs. Machine Learning

- **Data Mining**
  - A cross-disciplinary field that focuses on **discovering properties of data sets**.
  - Example:
    - Costco analyzed their point-of-sale data with data mining techniques they would be able to determine sales trends, develop marketing campaigns, and customer loyalty.
- **Machine Learning**
  - Designing algorithms that can **learn** and **make predictions from the data**.
  - Example:
    - Costco analyzed their point-of-sale data with a machine learning model to predict the sales of next season, new product, and new location.

A Simple Machine Learning Model



Learning Types

- Supervised learning
  - Learning from labeled training data
    - The targets of training data are known
- Unsupervised learning
  - Learning from unlabeled training data
    - The targets of training data are unknown
- Reinforcement learning
  - How a machine ought to take actions in an environment so as to maximize some notion of cumulative reward.
  - Alpha GO

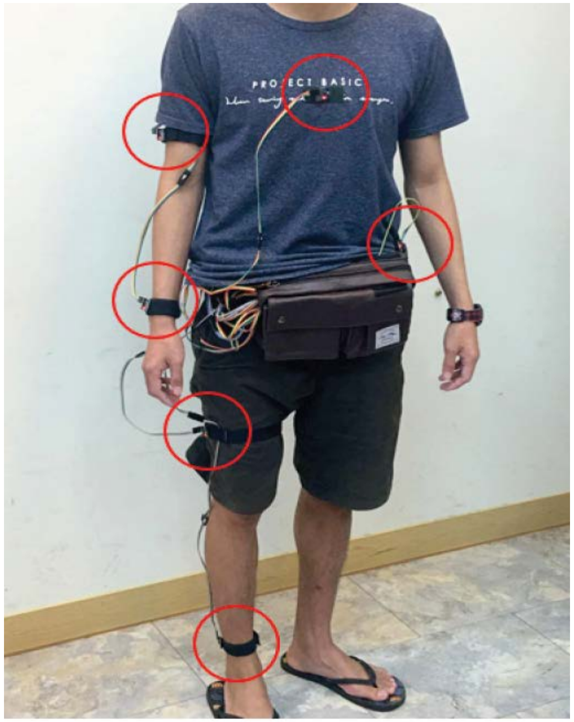
not bad decision but not garentee that you can win the game

Machine Learning Models

- Decision tree
- K-nearest neighbors KD tree
- Linear regression
- Logistic classifier
- SVM, support vector machine
- Naive Bayes classifier

Applications of Machine Learning

- Human activities classification
  - 6 G-sensor (MPU6050)
    - 3-axis accelerometer and gyroscope
    - 32 features including 3-axis linear accelerations and 3-axis angular velocities.
  - Using SVM to classify 8 activities
    - standing, sitting, lying, walking, running, going upstairs, drinking water, and dumbbelling.
  - Estimating the importance of each G-sensor to reduce the number of G-sensors.
  - Two sensors directly relate the movement of whole foot.
    1. Ankle
    2. Thigh

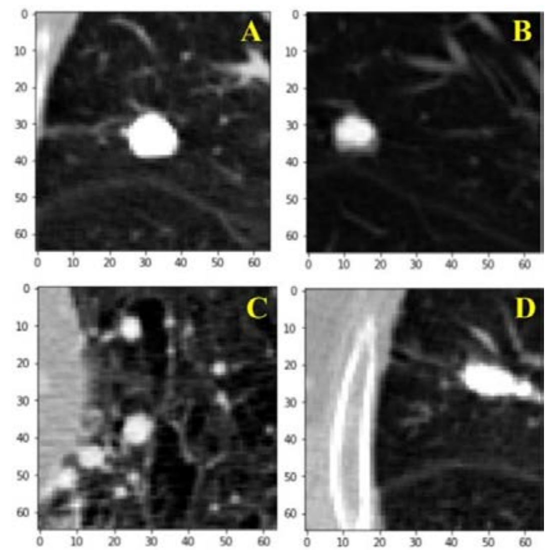


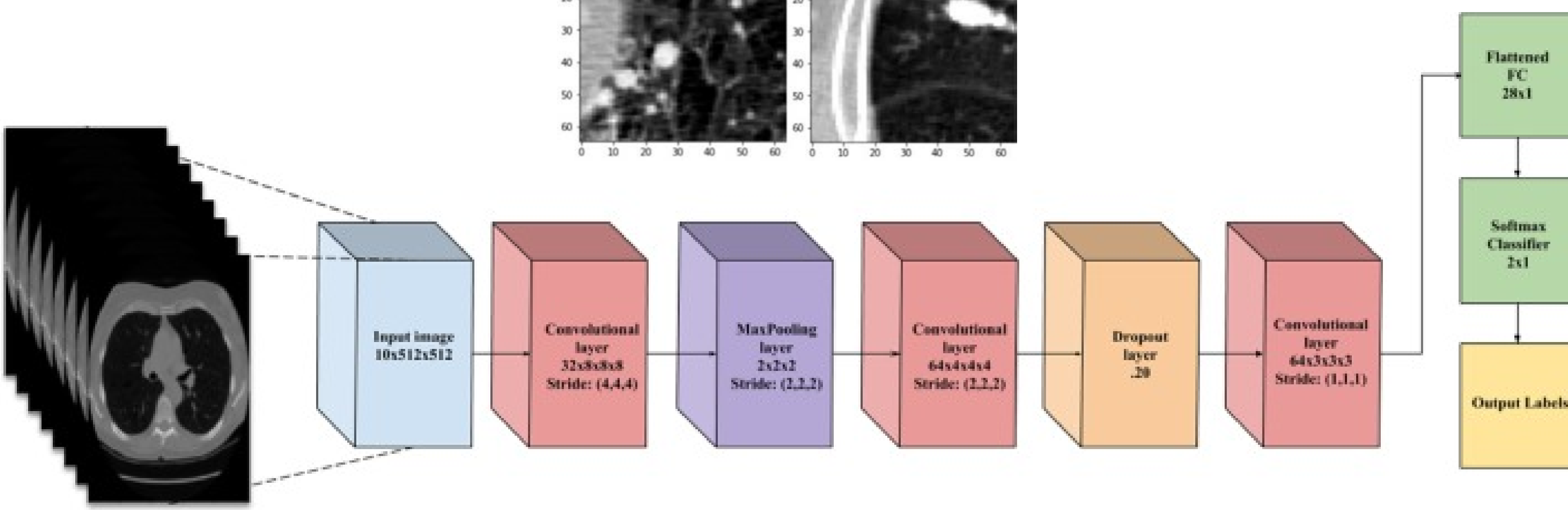
Applications of Machine Learning

- Handwritten signature verification
  - Using a single G-sensor to gather a period of pen movement signal.
  - Verifying a signature by the SVM classifier.
  - Accuracy:
    - True positive: 95%
    - True negative: 92%

Application of Neural Network

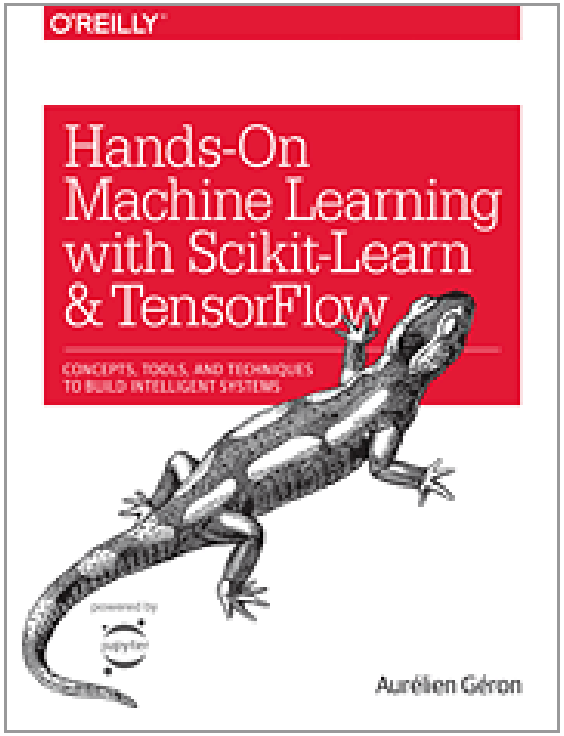
- Lung nodule detection from low-dose CT
  - Ali, Issa et al. “Lung Nodule Detection via Deep Reinforcement Learning.” *Frontiers in Oncology* 8 (2018): 108. PMC. Web. 23 Aug. 2018.





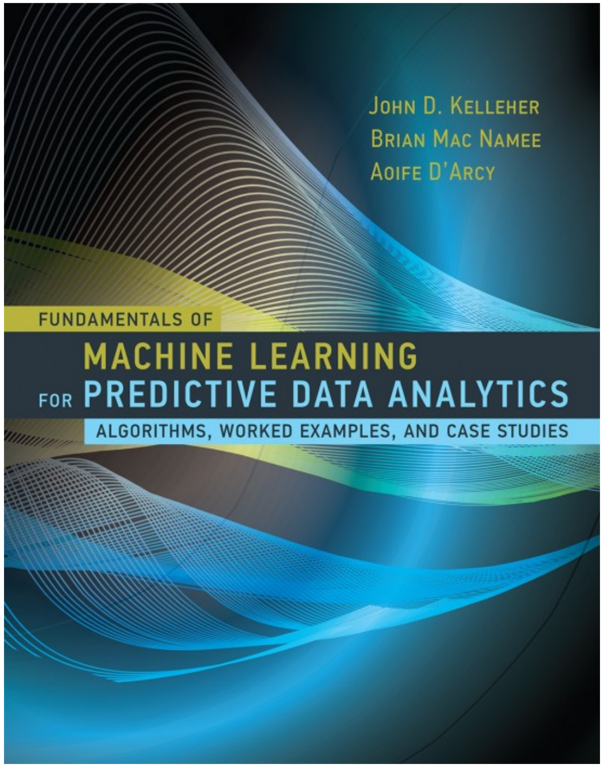
Reference Books

- Hands-On Machine Learning with Scikit-Learn and TensorFlow - Concepts, Tools, and Techniques to Build Intelligent Systems
  - Aurélien Géron
  - O'Reilly Media
  - 2017



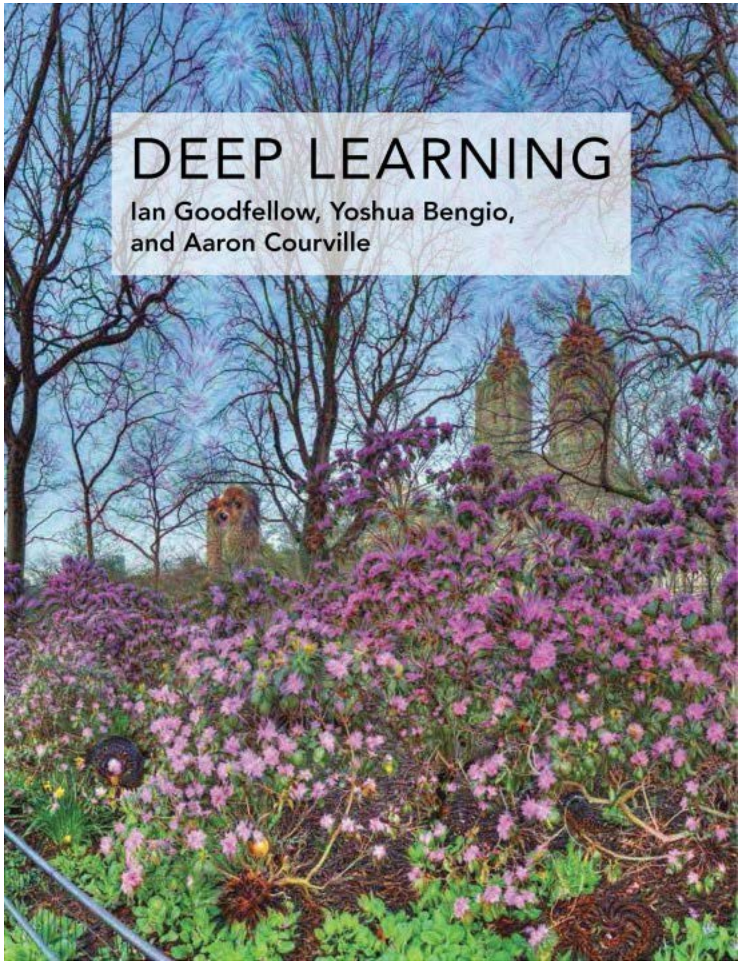
Reference Books

- Fundamentals of Machine Learning for Predictive Data Analytics - Algorithms, Worked Examples, and Case Studies
  - John D. Kelleher, Brian Mac Namee, and Aoife D'Arcy
  - MIT Press
  - 2015



Reference Books

- Deep Learning
  - Ian Goodfellow, Yoshua Bengio, and Aaron Courville
  - MIT Press
  - 2016



Machine Learning with Python

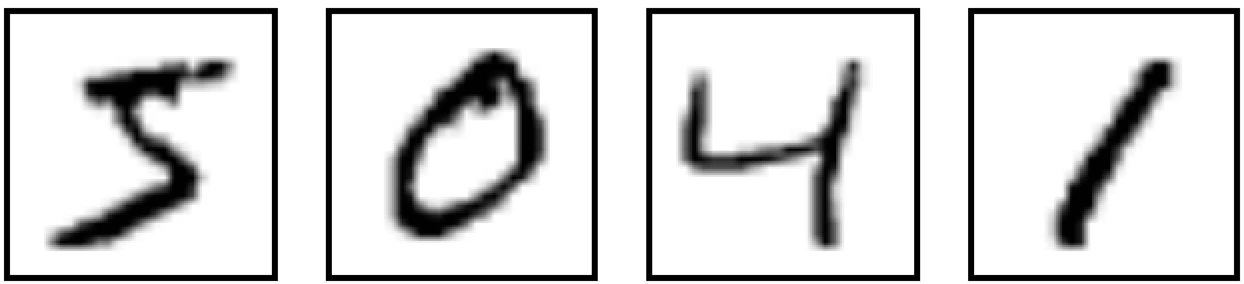
- scikit-learn
  - A python library for data mining and data analysis
  - Built on NumPy, SciPy, and matplotlib
- TensorFlow
  - An open-source library for artificial intelligence applications including mathematics, machine learning, and artificial neural network.
  - Developed by Google Brain Team
  - API: Python (the most complete and the easiest to use), C++, JAVA, and Go.
  - It supports GPU (CUDA)

Online Resource

- Kaggle
  - https://www.kaggle.com/
  - An open platform for machine learning
  - Open datasets
  - Open software and source code
  - Google Cloud team

Popular Datasets for Research

- MNIST
  - A simple computer vision dataset.
  - It consists of images of handwritten digits
  - Each image has 28 x 28 = 784 pixels
  - Each pixel has a single value in [0, 1]
  - Each image has a labels. For example, the labels for the above images are 5, 0, 4, and 1.



Popular Datasets for Research

- IRIS
  - A data set that consists of 50 samples from each of three species of Iris (setosa, versicolor and virginica).
  - Four features were measured from each sample: the length and the width of the sepals and petals, in centimeters.
  - It introduced by the British statistician and biologist Ronald Fisher in 1936.

setosa

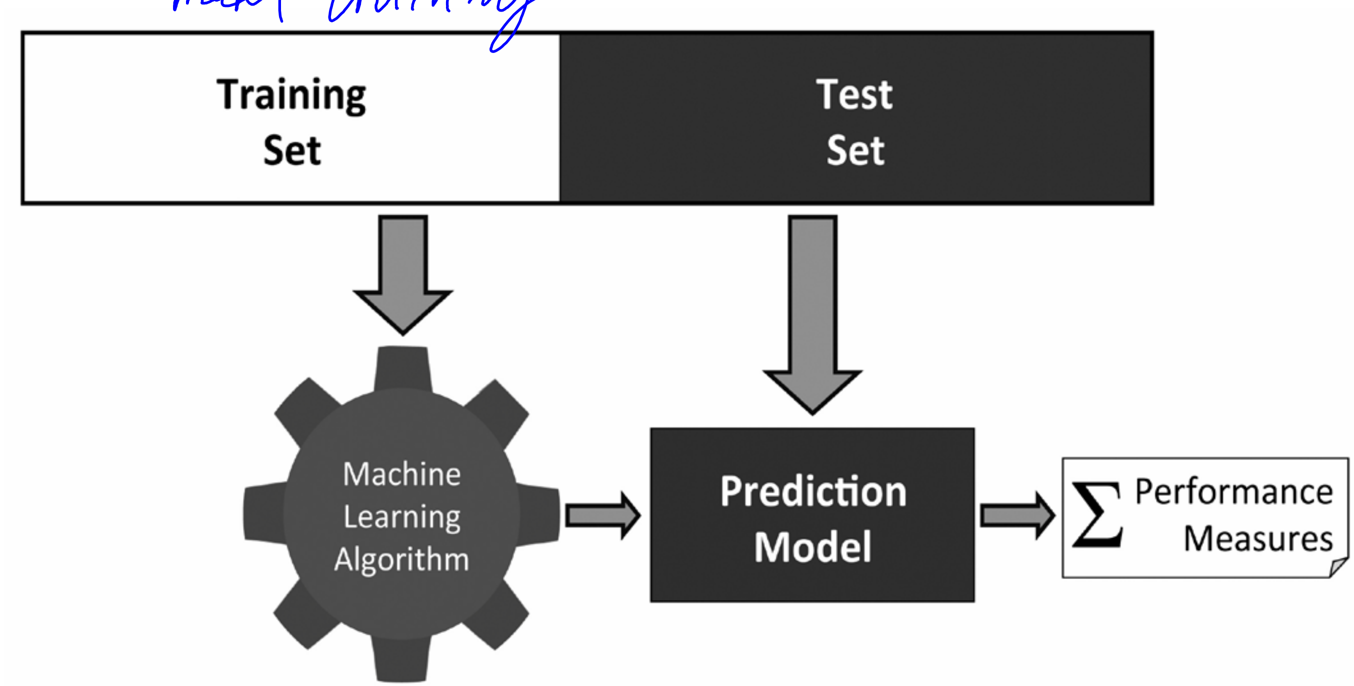
versicolor

virginica

Training and validation

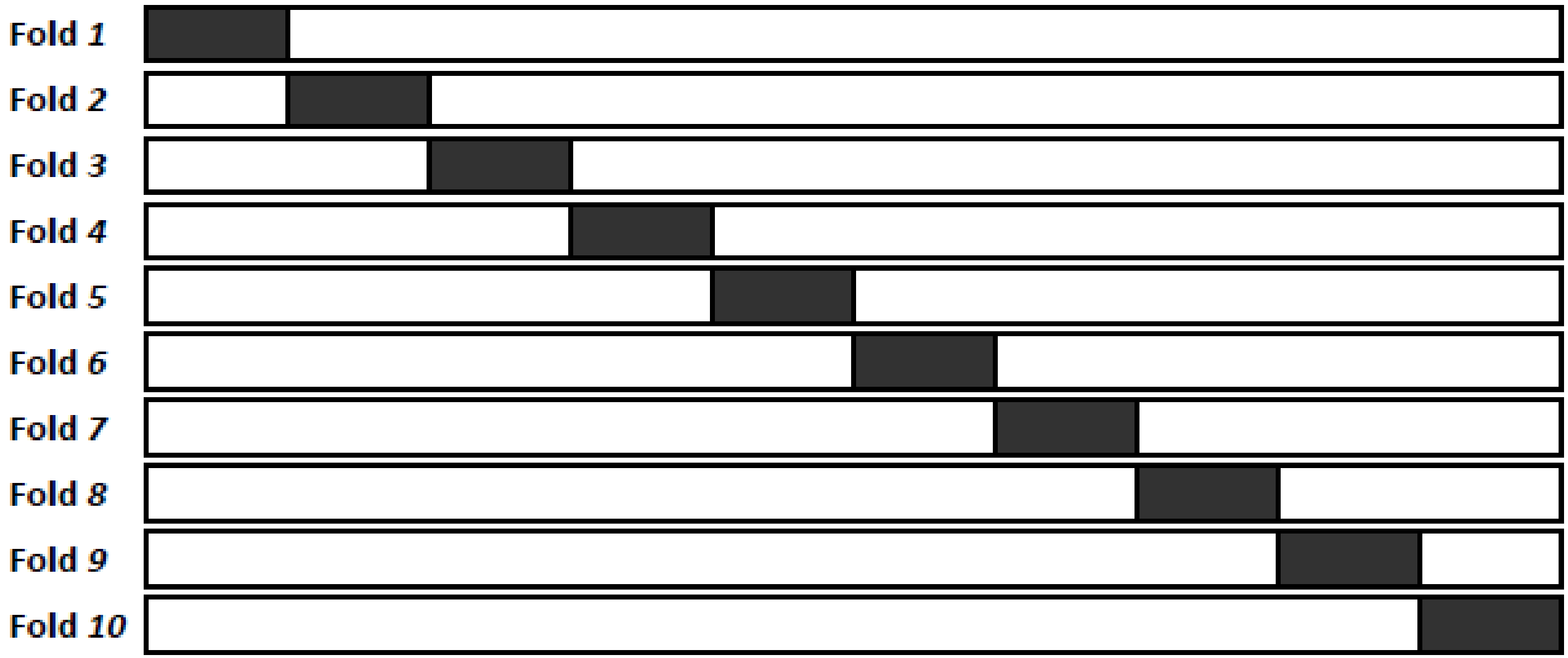
- Data
  - Training dataset
  - Test dataset (validation dataset)

model training



Validation

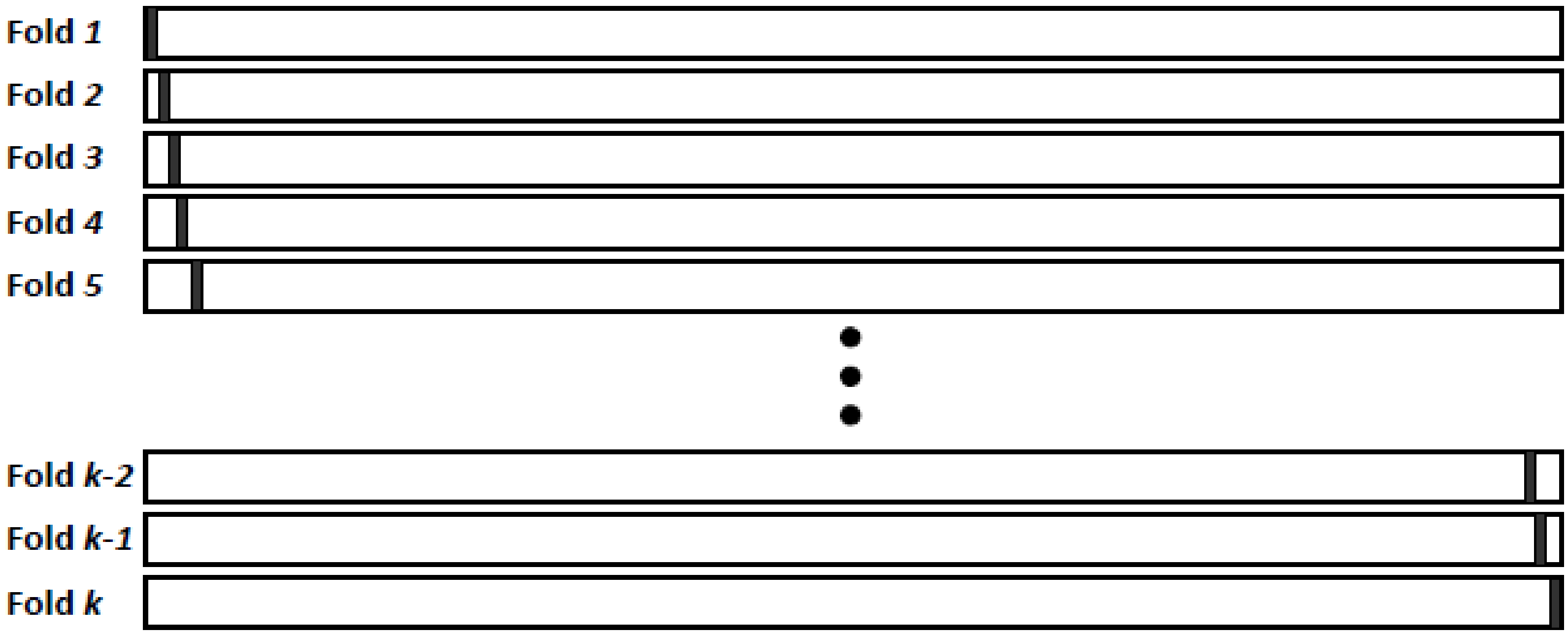
- K-ford cross validation 確認



Black rectangles indicate test data,
and white spaces indicate training data.

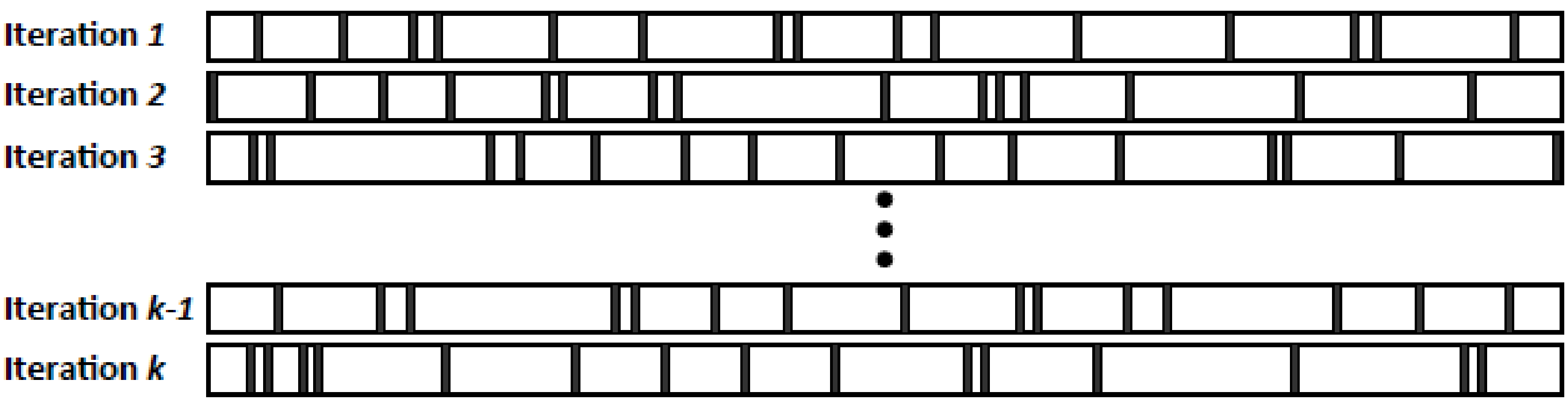
Validation

- Leave-one-out cross validation



Validation

- ϵ0 bootstrap process
  - **bootstrapping**: a self-starting process
  - k iterations
  - Each iteration randomly select *m* instances as training set



Black rectangles indicate test data,
and white spaces indicate training data.

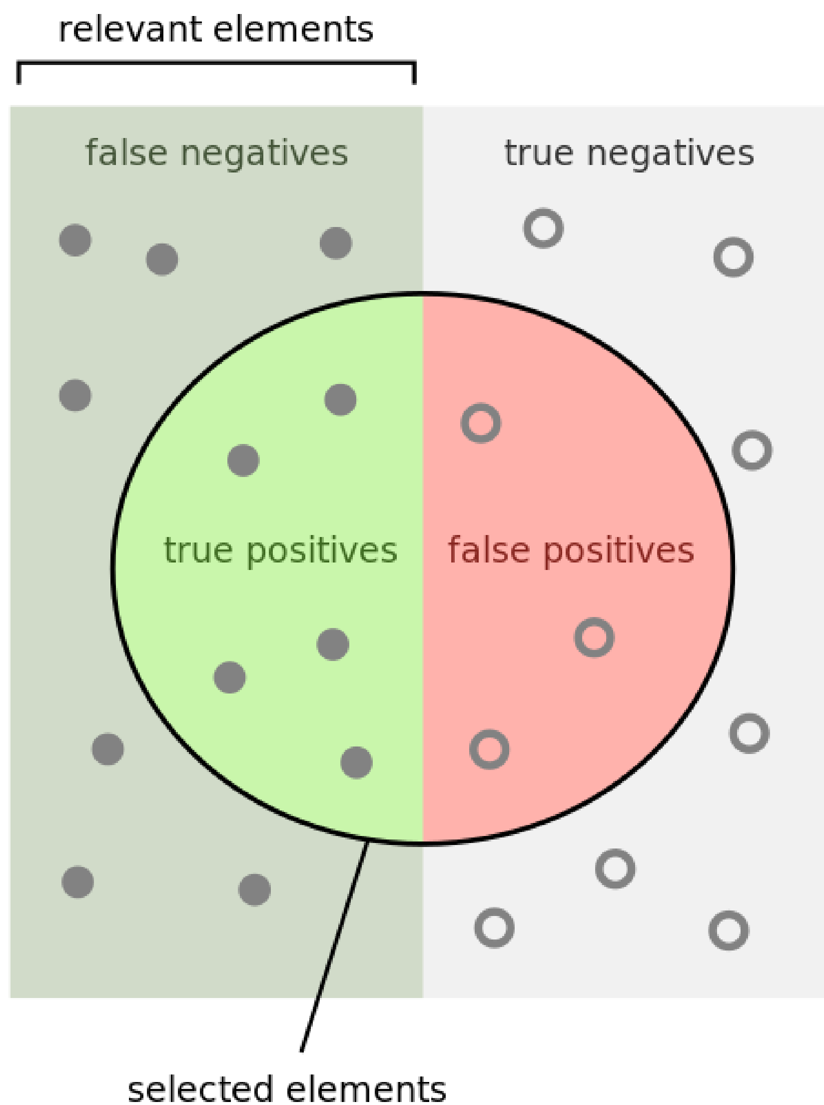
Evaluation

- Basic evaluation

$$misclassification\ rate = \frac{number\ incorrect\ predictions}{total\ predictions}$$

Evaluation

- Four possible outcomes
  - True Positive (TP)
  - True Negative (TN)
  - False Positive (FP)
  - False Negative(FN)



Evaluation

- Confusion matrix

		Prediction	
		positive	**negative**
Target	**positive**	*TP*	*FN*
	negative	*FP*	*TN*

Evaluation

- Confusion matrix
  - Example: spam/harmful email classification

ID	Target	Pred.	Outcome	ID	Target	Pred.	Outcome
1	spam	ham	FN	11	ham	ham	TN
2	spam	ham	FN	12	spam	ham	FN
3	ham	ham	TN	13	ham	ham	TN
4	spam	spam	TP	14	ham	ham	TN
5	ham	ham	TN	15	ham	ham	TN
6	spam	spam	TP	16	ham	ham	TN
7	ham	ham	TN	17	ham	spam	FP
8	spam	spam	TP	18	spam	spam	TP
9	spam	spam	TP	19	ham	ham	TN
10	spam	spam	TP	20	ham	spam	FP

		Prediction	
		'spam'	***'ham'***
Target	***'spam'***	6	3
	'ham'	2	9

Evaluation

- Misclassification accuracy

$$\frac{(FP + FN)}{(TP + TN + FP + FN)}$$

$$\frac{(2+3)}{(6+9+2+3)} = 0.25$$

- Classification accuracy

$$\frac{(TP + TN)}{(TP + TN + FP + FN)}$$

$$\frac{(6+9)}{(6+9+2+3)} = 0.75$$

Evaluation

- TP rate (TPR) $\frac{TP}{(TP + FN)}$
- TN rate (TNR) $\frac{TN}{(TN + FP)}$
- FP rate (FPR) $\frac{FP}{(TN + FP)}$
- FN rate (FNR) $\frac{FN}{(TP + FN)}$

Evaluation

- For example

		Prediction	
		'spam'	*'ham'*
Target	*'spam'*	6	3
	'ham'	2	9

TP	FN
FP	TN

$$
\begin{aligned}
\text{TPR} &= \frac{6}{(6+3)} &= 0.667 \\
\text{TNR} &= \frac{9}{(9+2)} &= 0.818 \\
\text{FPR} &= \frac{2}{(9+2)} &= 0.182 \\
\text{FNR} &= \frac{3}{(6+3)} &= 0.333
\end{aligned}
$$

Evaluation

- Precision

$$\frac{TP}{(TP + FP)}$$

- Recall

$$\frac{TP}{(TP + FN)}$$

Evaluation

- For example

		Prediction	
		'spam'	*'ham'*
Target	*'spam'*	6	3
	'ham'	2	9

TP	*FN*
FP	*TN*

$$\text{precision} = \frac{6}{(6+2)} = 0.75$$

$$\text{recall} = \frac{6}{(6+3)} = 0.667$$

Evaluation

- A confusion matrix for a k-NN model trained on a churn prediction problem.

		Prediction	
		'non-churn'	*'churn'*
Target	*'non-churn'*	90	0
	'churn'	9	1

Target

$$Recall_{nc} = \frac{90}{90+0} = 1.0$$

$$Recall_{c} = \frac{1}{9+1} = 0.1$$

- A confusion matrix for a naive Bayes model trained on a churn prediction problem.

		Prediction	
		'non-churn'	*'churn'*
Target	*'non-churn'*	70	20
	'churn'	2	8

$$Recall_{nc} = \frac{70}{70+20} = 0.778$$

$$Recall_{c} = \frac{8}{2+8} = 0.8$$