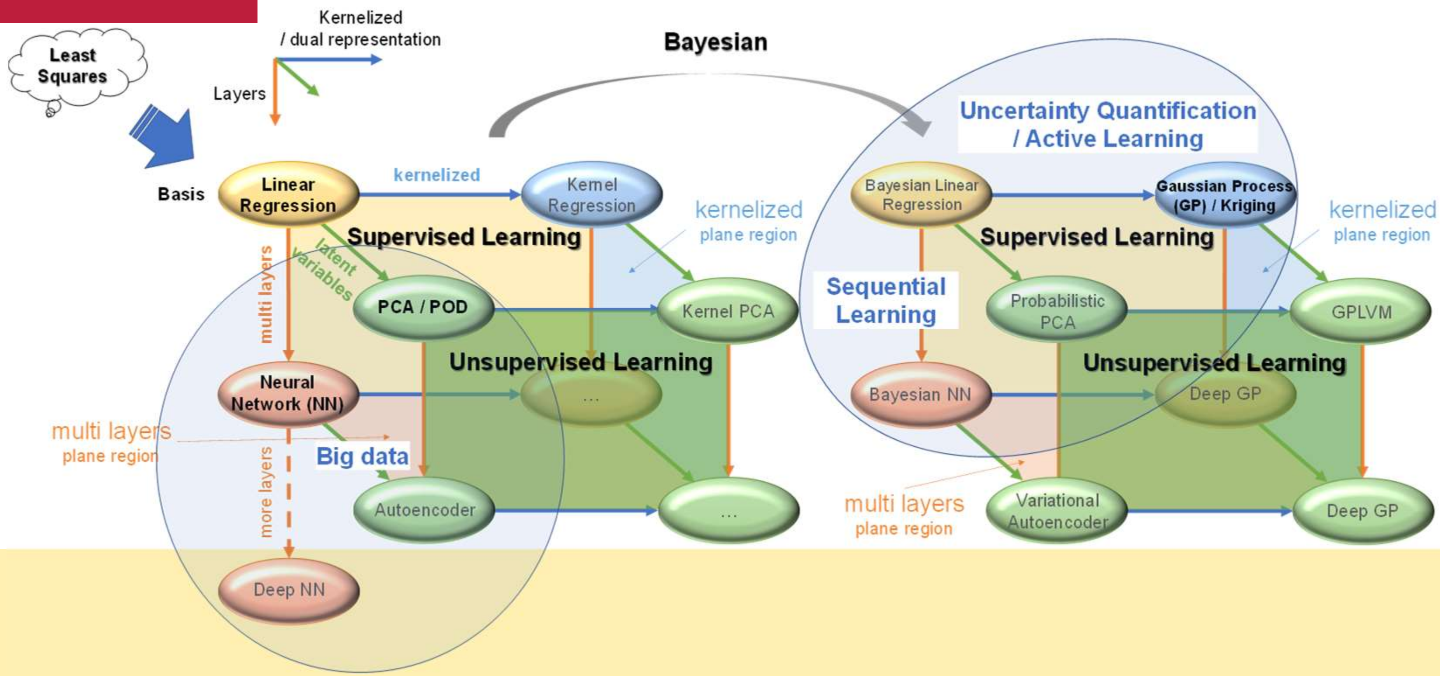


Scientific Machine Learning

Lecture 1: Introduction

Dr. Daigo Maruyama

Prof. Dr. Ali Elham

Self-Introduction

Chair of Overall Aircraft Design / Faculty of Mechanical Engineering

Prof. Dr. Ali Elham

Dr. Daigo Maruyama

- Ph.D. in aerospace engineering in Japan
- 10 years experiences in European aerospace agencies (France and Germany) before joining the team
- Experiences of machine Learning and uncertainty quantification in practice

Lecture: every Tuesday, 15:30-17:00 (13.04.2021-20.07.2021)
A lecture video is uploaded every week just after the allocated lecture time.

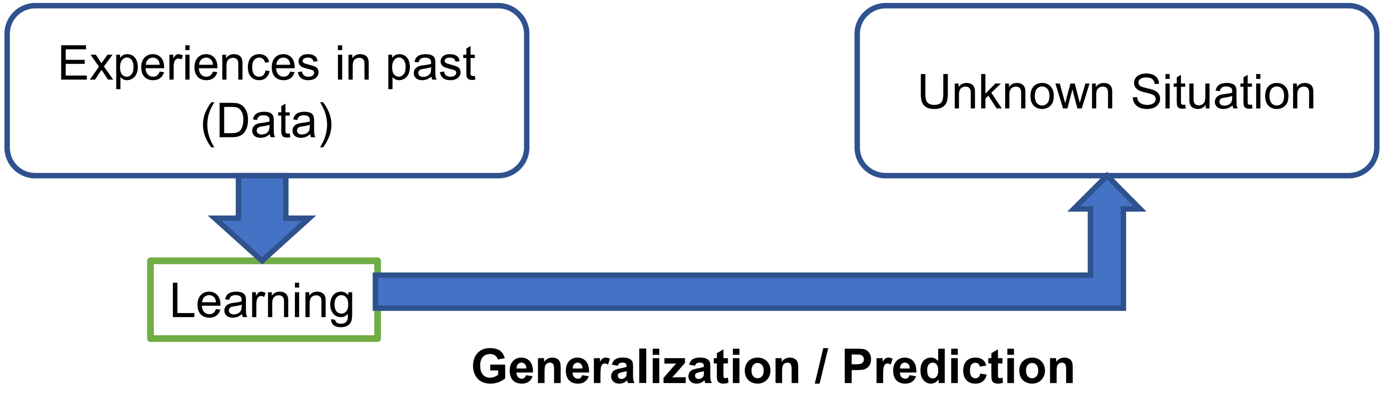
Lecture content

- What is machine learning
- Structures of machine learning techniques from a scientific viewpoint
- Course structure and content
- Course assessment

Objective of Machine Learning

Arthur Samuel, “Field of study that gives computers the ability to learn **without being explicitly programmed**”, 1959

https://en.wikipedia.org/wiki/Arthur_Samuel



Fields of Application of Machine Learning (Examples)

- **Deep Learning**
  - Big data
    - Natural language processing
    - Image recognition (in 2015, it exceeded human)
  - Combination with reinforcement learning

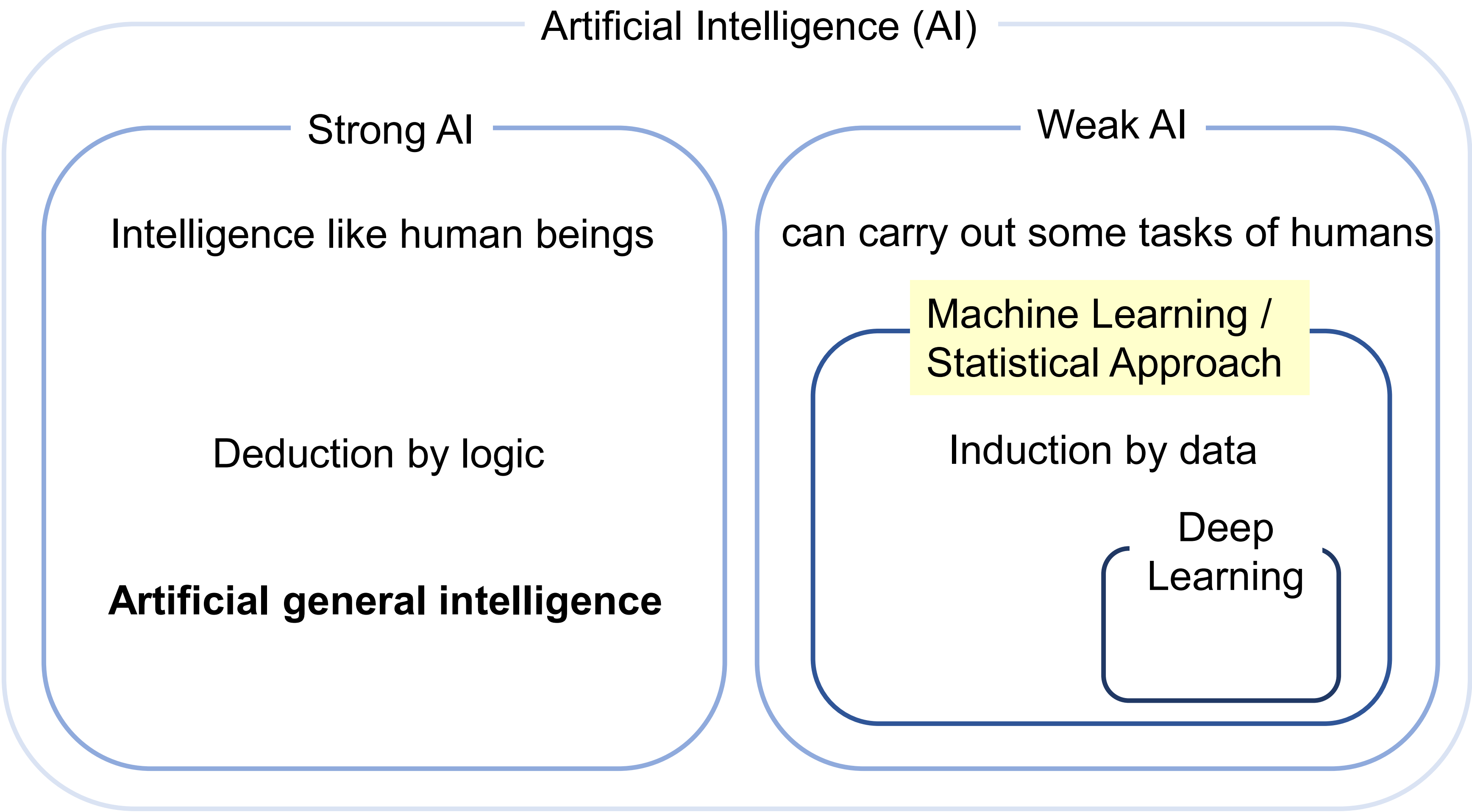
- **Efficient Global Optimization (Bayesian Optimization)**
  - Gaussian process

- **Recursive Bayesian estimation**
  - Data assimilation
  - Karman filter

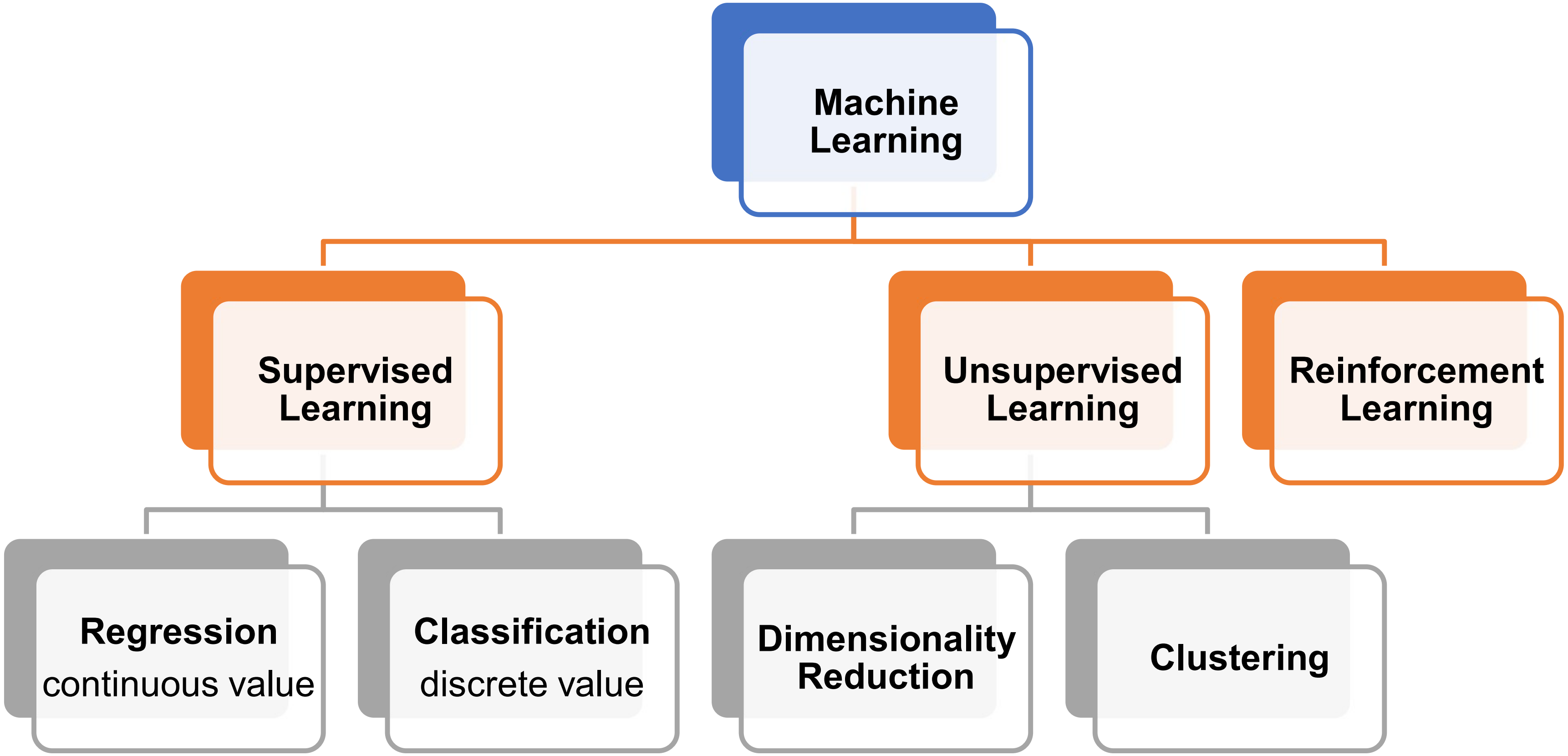
- **Uncertainty Quantification**
  - Robust- / Reliability-based design
  - Parameter inference
  - etc.

Figures from: Brunton, S. L, et. al., “Data-Driven Aerospace Engineering: Reframing the Industry with Machine Learning”

Artificial Intelligence and Machine Learning



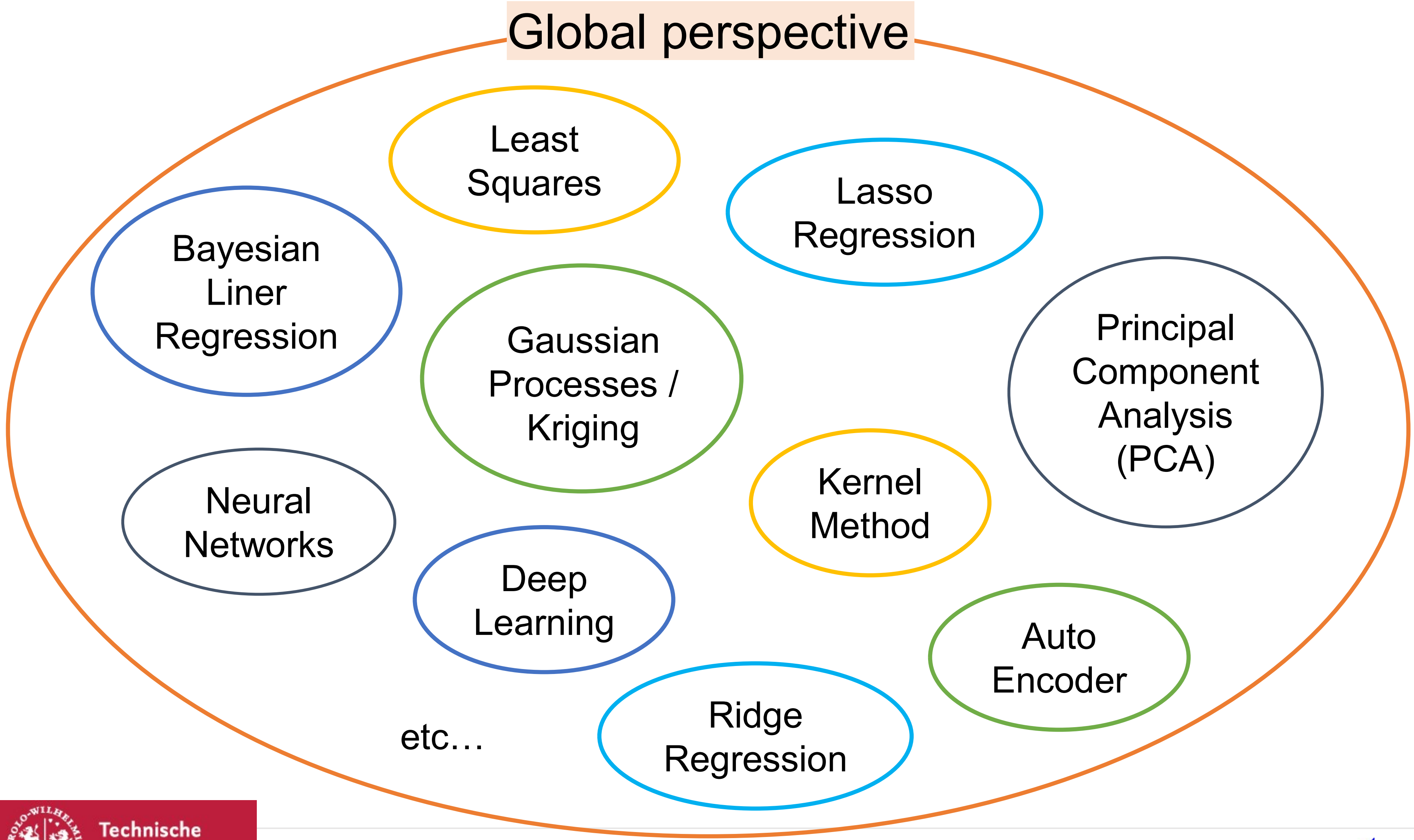
Machine Learning Classification by Use/Application



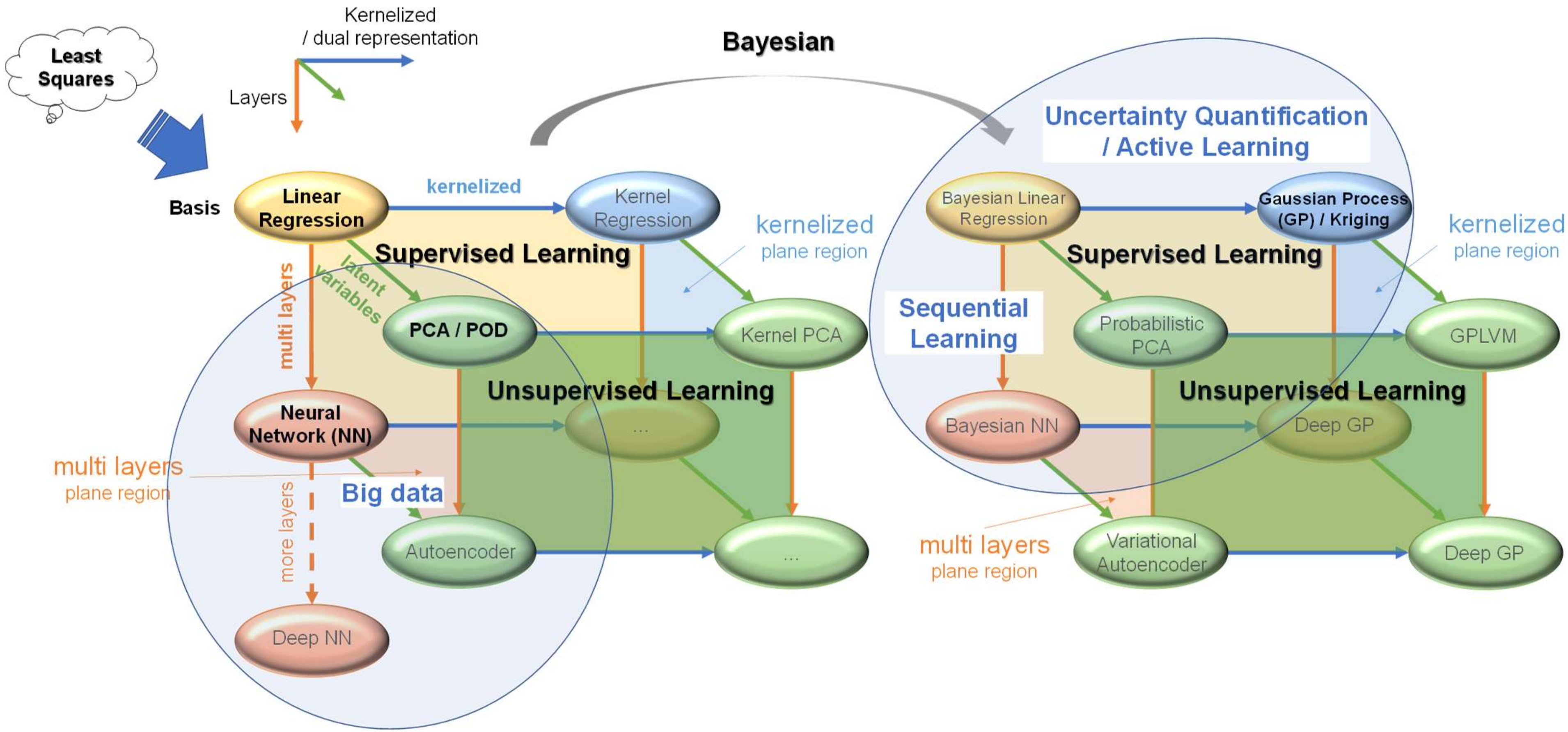
In this course, machine learning classification is done by **methods and their concepts**.

→ Then the use/application is naturally derived/understood.

Various Machine Learning Techniques



Structures of the Techniques/Methods



Big Data

- natural language processing
- Image recognition
- speech recognition

- High dimensional input
- Huge numbers of data

MNIST database

data

new input

Learning

ML device

Prediction

7 ?

Big Data

$\boldsymbol{x}$

y 4 7 7

(output label with probability)

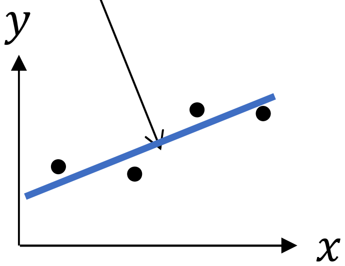
Image data
= color data at each pixel
= **value at each coordinate**

$$\boldsymbol{x} = (x_1, x_2, \cdots, x_{100\times 100})$$

10,000 dimensional input!

Least Squares

$y = a_1 x + a_0$



Learning

$$\hat{\boldsymbol{a}} = \underset{\boldsymbol{a}}{\operatorname{argmin}} \sum_{i=1}^{N} \{y_i - \underbrace{(a_1 x_i + a_0)}_{f(x,\boldsymbol{a})}\}^2$$

Prediction $y_{new} = \hat{a}_1 x_{new} + \hat{a}_0$

$$f(\underbrace{\boldsymbol{x}}_{\text{10,000 dim!}}, \underbrace{\boldsymbol{a}}_{\text{a large number of the parameters!}})$$

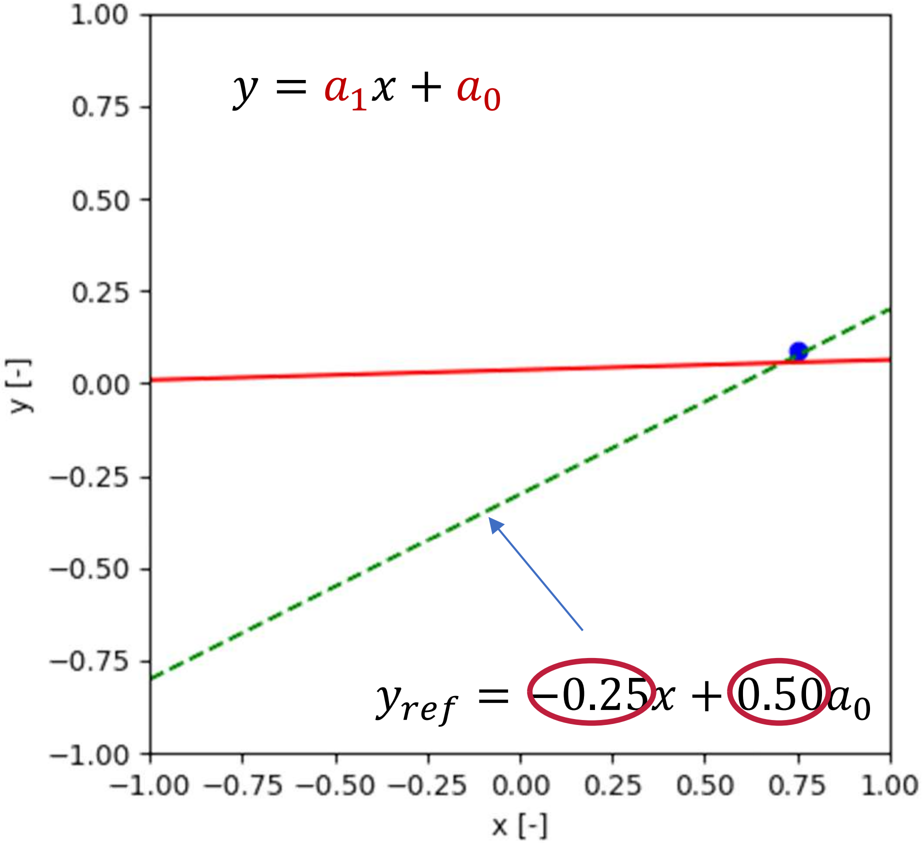
10,000 dim!

a large number of the parameters!

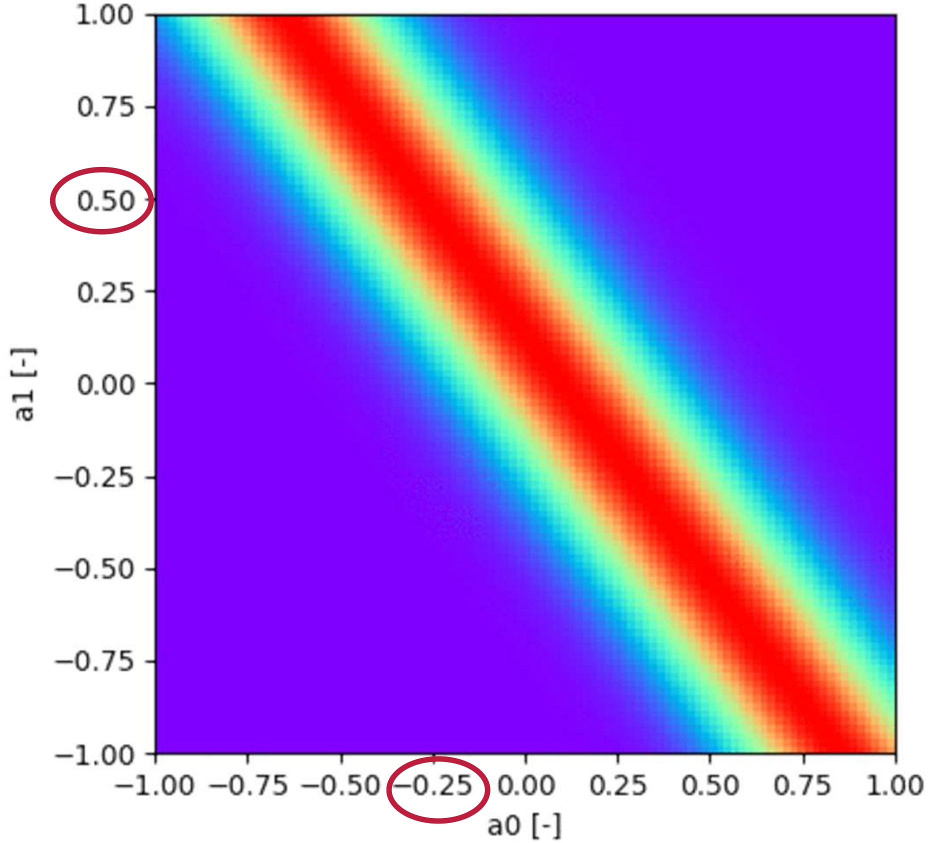
High-flexibility with
fewer parameters
(than the other models)

Bayesian Sequential Learning

input - output space



parameter space



The parameters a_0 and a_1 are being determined to be around (-0.25, 0.50).

Gaussian Process

Problem setting:

- No data is available yet
- Each sample is expensive.

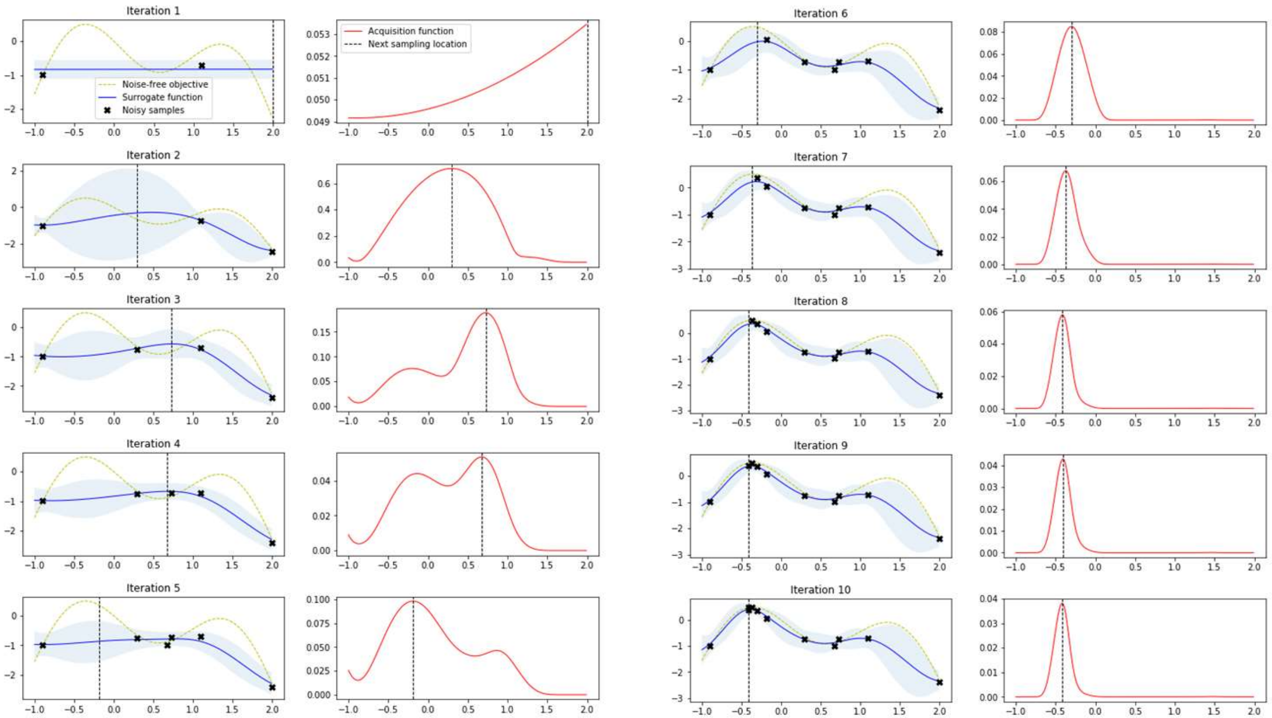
For contrast use to deep learning

However, we are greedy that we want to find the optimum point!

Gaussian Process

uncertainty to find a good sample location to find the max

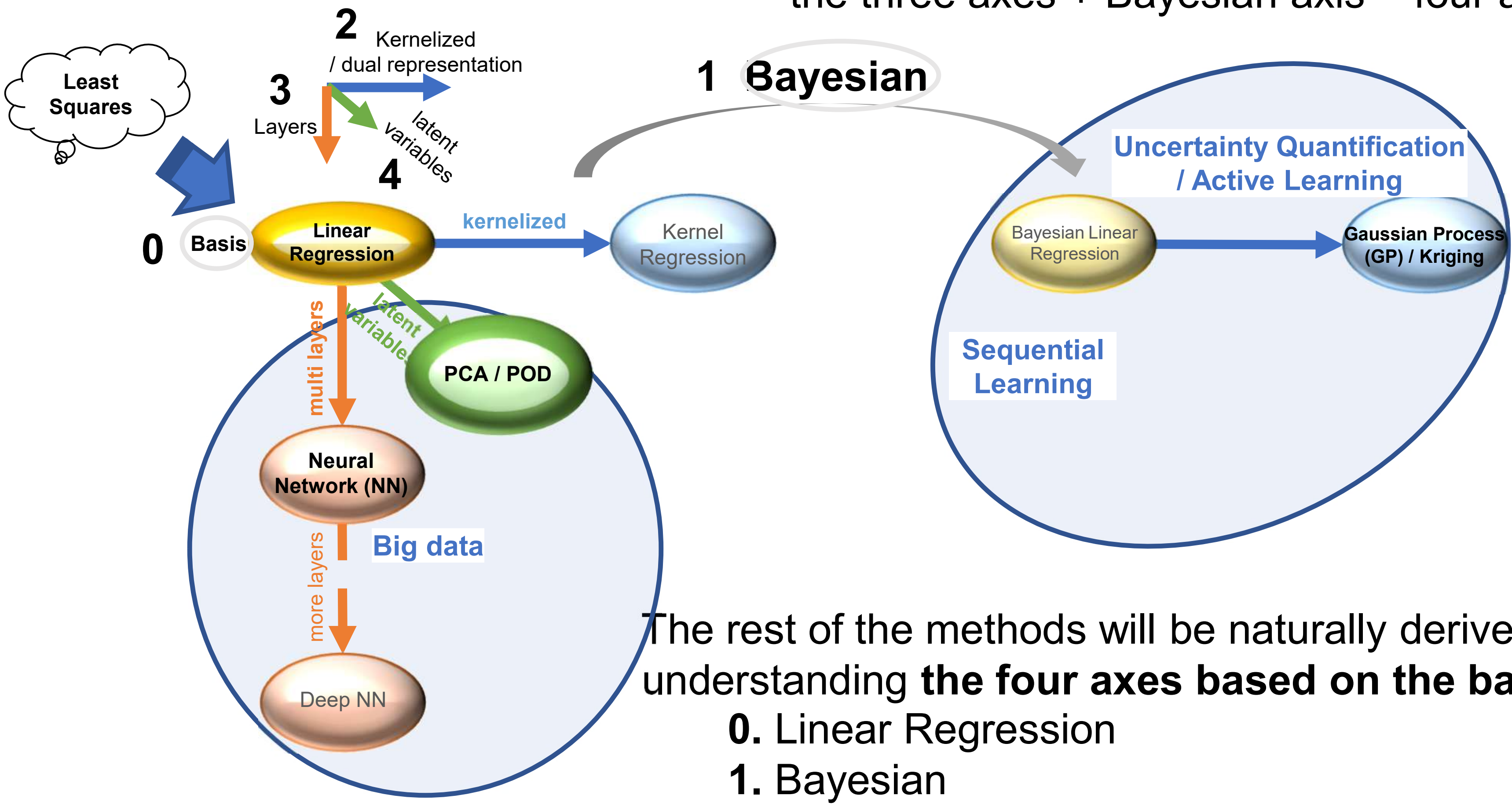
Bayesian optimization



http://krasserm.github.io/2018/03/21/bayesian-optimization/

Key Components

the three axes + Bayesian axis = four axes



The rest of the methods will be naturally derived after understanding **the four axes based on the basis.**

0. Linear Regression
1. Bayesian
2. Kernel / Dual Representation
3. Layers
4. Latent Variables (Unsupervised Learning)

Key Components

0. Linear Regression	Lecture: 4
1. Bayesian	Lecture: 5, 6
2. Kernel / Dual Representation	Lecture: 7
Combination of 1. and 2.	Lecture: 8, 9
3. Layers	Lecture: 10
4. Latent Variables (Unsupervised Learning)	Lecture: 11

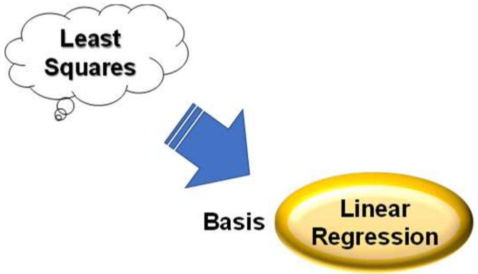
Course Content

Introduction	Lecture 1
Acquisition of Concepts and Tools	Lecture: 2, 3
0. Linear Regression	Lecture: 4
1. Bayesian	Lecture: 5, 6
2. Kernel / Dual Representation	Lecture: 7
Combination of 1. and 2.	Lecture: 8, 9
3. Layers	Lecture: 10
4. Latent Variables (Unsupervised Learning)	Lecture: 11
Advanced Techniques	Lecture: 12, 13, 14

Course Content (the title of each lecture)

1. Introduction
2. Curve Fitting and Probability Theory
3. Probability Distributions and Sampling Methods
4. Linear Regression
5. Bayesian Statistics (1)
6. Bayesian Statistics (2) (Bayesian Linear Regression)
7. Dual Representation (Kernel Method)
8. Gaussian Process (1)
9. Gaussian Process (2) and SVM, Classification/Generalized Linear Model
10. Neural Network
11. Unsupervised Learning
12. Numerical Methods, Bayesian Networks and Clustering
13. Mixture Model and Hierarchical Bayesian Model
14. Other Methods

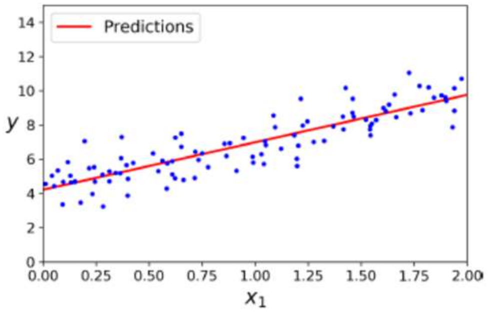
0. Linear Regression (Lectures 4)

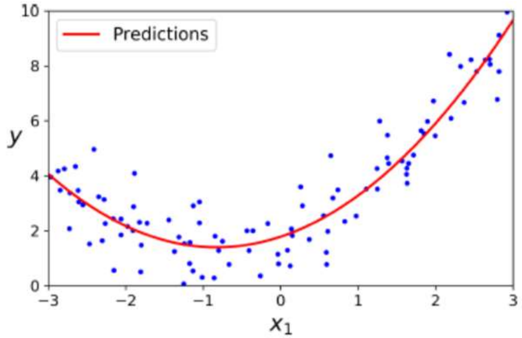


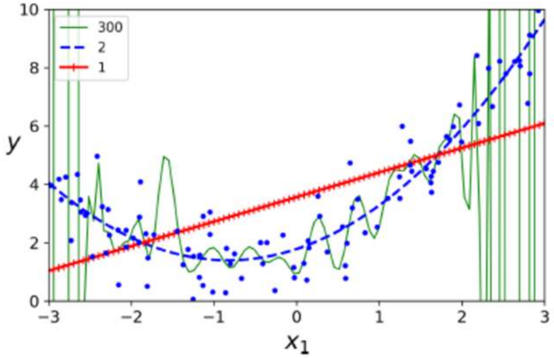
Basis for all the derivations

- However, this is a bridge to Bayesian perspective to get familiar with probability theory and fundamental machine learning techniques.
- At first extended from the least square method.

Overfitting

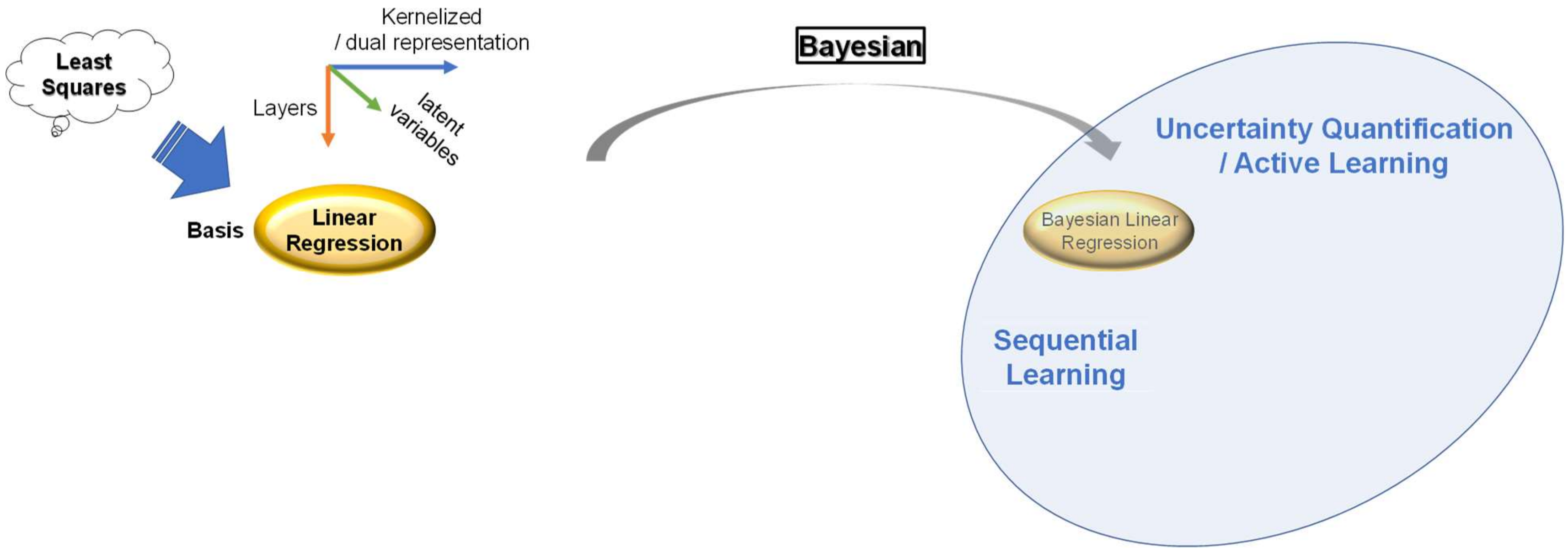






Aurélien Géron, “hands-on machine learning with scikit-learn keras and tensorflow”

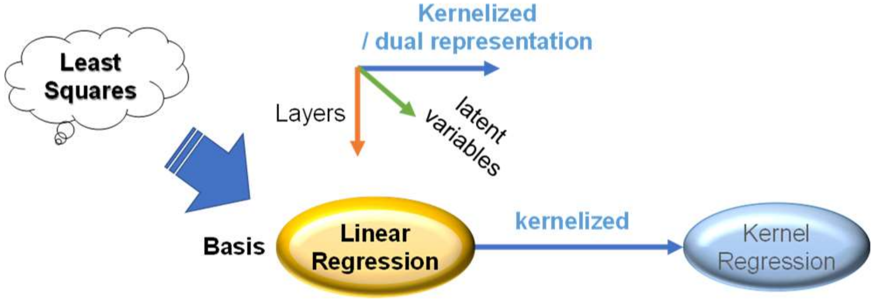
1. Bayesian Model (Lectures 5 and 6)



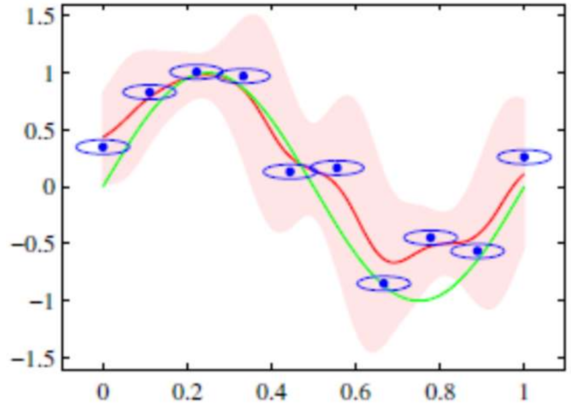
Very important:

- Consistent and coherent
- After introducing this perspective, basically all the theories are viewed in Bayesian statistics.

2. Kernel Methods / Dual Representation (Lecture 7)



Christopher M. Bishop "Pattern Recognition And Machine Learning"

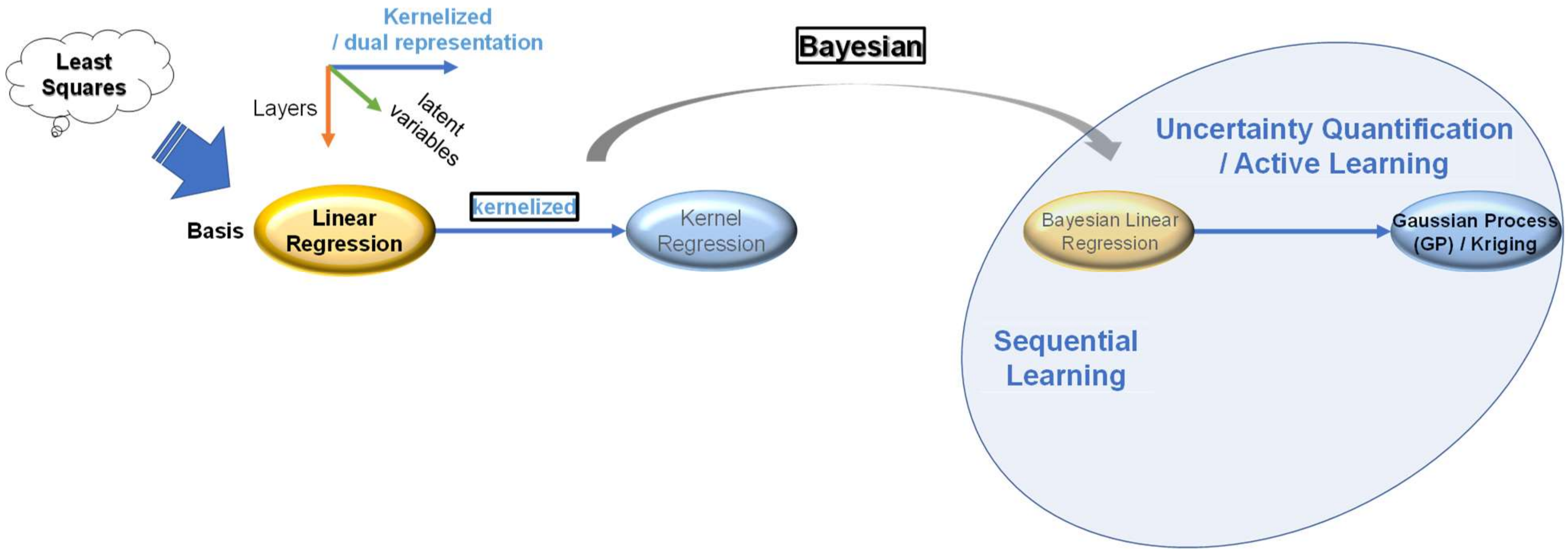


Nonparametric model

Another expression of the linear regression models

- Parametric models to Nonparametric models
- The number of the parameter can be controlled by another expression (pros and cons).

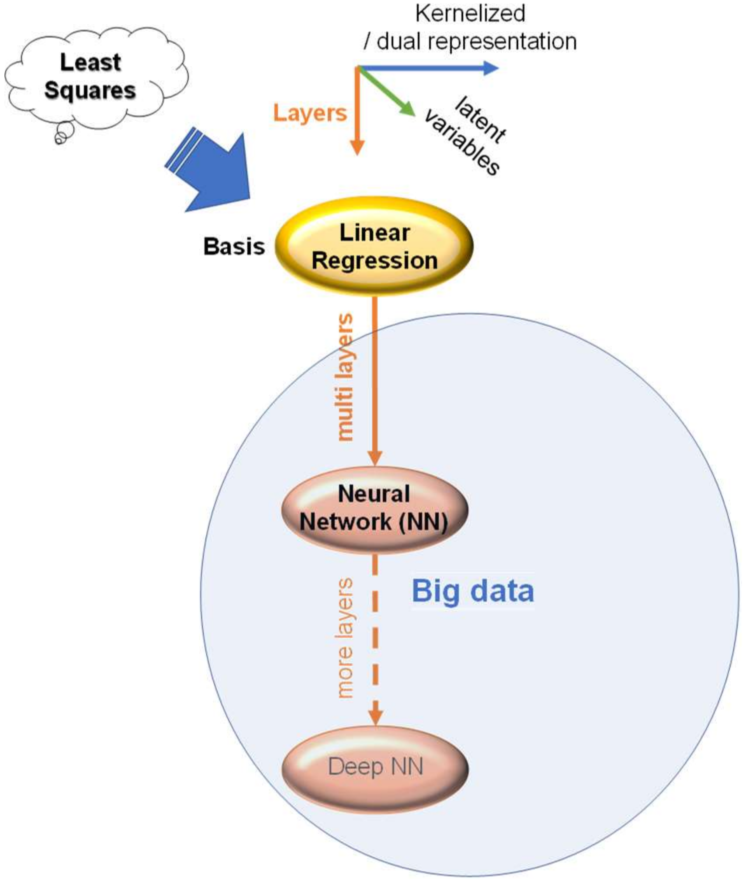
1+2. Combination of Kernel and Bayesian Approaches (Lectures 8 and 9)



Gaussian Process (GP) is one of the most important models, especially handling the uncertainty due to lack of data.

- Good Models to express complex functions with uncertainty.

3. Multi Layers (Lectures 10)



Deep NN (Deep Learning) is one of the most important models especially for handling big data.

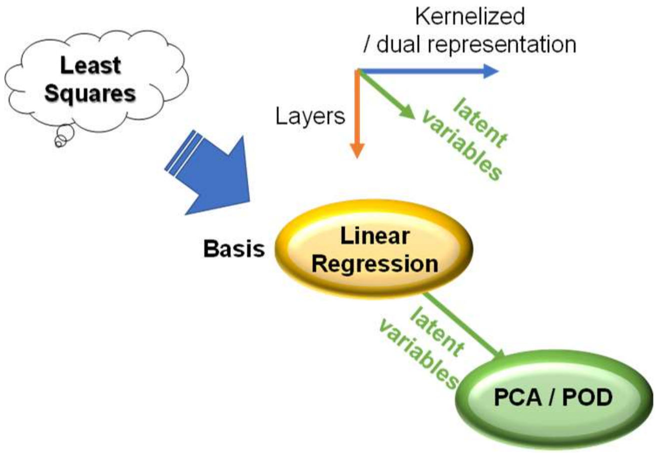
- Good models to express complex functions.
- The variety of the functions is wide.

Due to its nice properties and the state-of-the-art techniques, "big data" (billions of sample size) with relatively small numbers of the parameter – still order millions

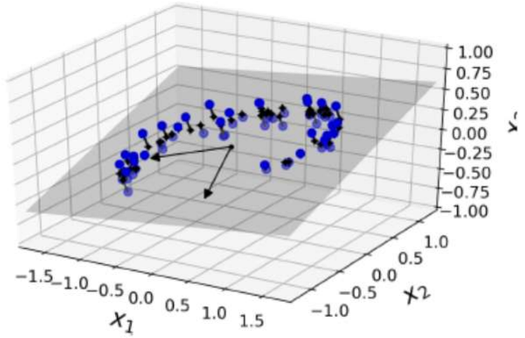
Key words:
- Linear model
- Nonlinear function
- Differentiable

➡ Suppress increase of the parameters

4. Latent Variables – Unsupervised Learning (Lecture 11)

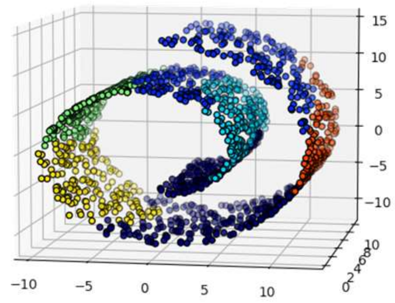


Aurélien Géron, "hands-on machine learning with scikit-learn keras and tensorflow"



by Principal Component Analysis (PCA)

scikit-learn example



The data points are inherently lie on a lower dimensional space.

3D→2D

by Manifold Learning

Unsupervised Learning

- Dimensionality reduction
  - Computational cost
  - Visualization
- Feature extraction

We already know some solutions by the previous lectures till there.

Concepts and Tools: Probability Theory and Statistics

Machine Learning $\in$ Statistical Inference

= Approximation theory/method of probability measure

Essentially…

The Rules of Probability

sum rule $p(x) = \int p(x,y)\,\mathrm{d}y$

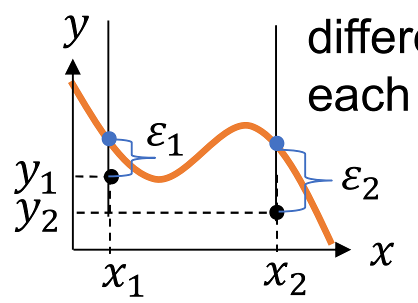
product rule $p(x,y) = p(x|y)p(y)$

… and probability distributions

Training to get familiar with the rule of probability

Machine Learning Modeling

Why we consider probability?



different ε for each case?

$$y = f(x) + \varepsilon$$

ε

ε: noise

Imagine the butterfly effect…

We pick up one of the observable quantities.

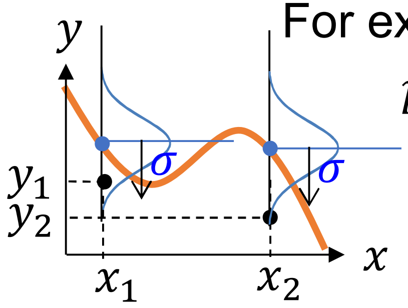
1000 dim~ $x \longrightarrow \boxed{f(x)} \longrightarrow y$ 1 dim

x: input

y: output

If we know all the components which describe the output, the output can be deterministic.

Why we consider probability?



For example, Gaussian distribution

$$p(y|x, \sigma) = \mathcal{N}(f(x)|\sigma^2)$$

$$y \sim p(y|x)$$

Probabilistic model

$$p(y|x, \boldsymbol{a}, \sigma) = \mathcal{N}(f(x, \boldsymbol{a})|\sigma^2)$$

where, $f(x, \boldsymbol{a}) = a_3 x^3 + a_2 x^2 + a_1 x + a_0$

$\boldsymbol{a}, \sigma$ are learned by the data.

We pick up one of the observable quantities.

1000 dim~ $x \longrightarrow \boxed{p(y|x)} \longrightarrow y$ 1 dim

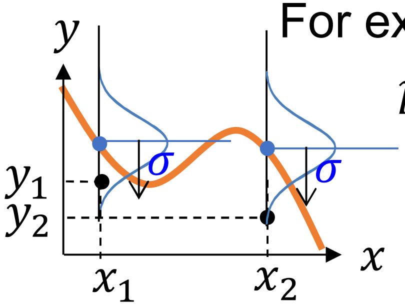
x: input

y: output

If we know all the components which describe the output, the output can be deterministic.

Machine Learning Modeling

Why we consider probability?



For example, Gaussian distribution

$$p(y|x, \sigma) = \mathcal{N}(f(x)|\sigma^2)$$

$$y \sim p(y|x)$$

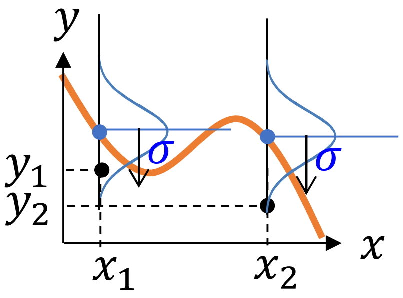
Probabilistic model

$$p(y|x, \boldsymbol{a}, \sigma) = \mathcal{N}(f(x, \boldsymbol{a})|\sigma^2)$$

where, $f(x, \boldsymbol{a}) = a_3x^3 + a_2x^2 + a_1x + a_0$

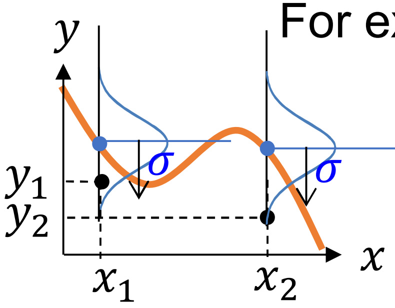
$\boldsymbol{a}, \sigma$ are learned by the data.

Physics perspectives



Machine Learning Modeling

Why we consider probability?



For example, Gaussian distribution

$$p(y|x,\sigma) = \mathcal{N}(f(x)|\sigma^2)$$

$$y \sim p(y|x)$$

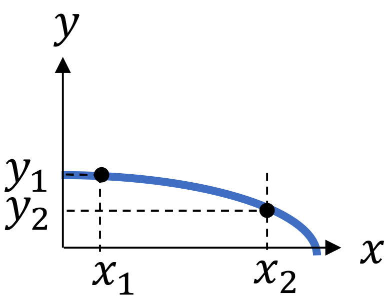
Probabilistic model

$$p(y|x,\boldsymbol{a},\sigma) = \mathcal{N}(f(x,\boldsymbol{a})|\sigma^2)$$

where, $f(x,\boldsymbol{a}) = a_3x^3 + a_2x^2 + a_1x + a_0$

$\boldsymbol{a}, \sigma$ are learned by the data.

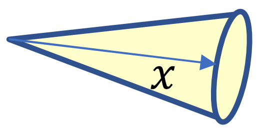
Physics perspectives



Intensity of light

x: distance

$$f(x,\boldsymbol{a}) = ax^{-2}$$



However, if we use flexible but complex functions with large amounts of data, the discovered complex function can imitate real physics behavior (discovered only by image data)

Accelerating Eulerian Fluid Simulation with Convolutional Networks (Google, DeepMind, ICML2017)

Machine Learning Modeling

Probabilistic model is hypothesis.

"**All models are wrong, but some are useful**.“
The aphorism from George Box*

*George E. P. Box “Science and Statistics“, *Journal of the American Statistical Association*, 71(791799), 1976.

https://en.wikipedia.org/wiki/George_E._P._Box

We cannot inquire which is correct, input error or output error.

Other Related Fields

The fields below are **clearly distinguished with the theory of machine learning**.

(This is very important also for better understanding the theory since some of the optimization techniques explained in books are classical optimization techniques themselves.)

- Optimization (except for the methods below)
  - Gradient method
  - Lagrange multiplier
  - (Gradient-free method)
- Linear Algebra
  - Some properties of specific matrices (positive semi definite, symmetric, etc.)
  - Some techniques of solving linear system
    - LU decomposition
    - Cholesky decomposition
- Calculus
  - Chain rule

Basically only usability issues will be explained.

Teaching Assistant

Nils Bock, M.Sc.

Every Tuesday (from the 2nd lecture time): 17:15-18:00

Note:

- Python is used for program execution/writing

Study Materials

Course main study materials

- Christopher M. Bishop "Pattern Recognition And Machine Learning" Springer-Verlag, 2006 (doesn't include all the subjects of the course)

- Lecture notes and some scientific papers

Additional suggested study materials (basically for Gaussian Processes: Lectures 8, 9)

- Forrester, Alexander, Sobester, Andras, Keane, Andy, "Engineering Design via Surrogate Modelling: A Practical Guide" (no pdf available)
- Carl Edward Rasmussen. Christopher K. I. Williams, "Gaussian Processes for Machine Learning", The MIT Press (not for beginner)

Assessment

- No final written exam

- Take home assignments:

  - Assignment
    - A presentation based on the take-home assignments
    - 100%
    - 10 mins presentation + 10 mins questions answering

Who are you?

1. Your Study track?

2. Do you have basic programming skills with Python?

3. Additionally, if you have any experiences of machine learning…