Machine Learning for Computer Vision

PD Dr. Rudolph Triebel
Lecturers

- PD Dr. Rudolph Triebel
  - rudolph.triebel@in.tum.de
  - Room number 02.09.059 (Fridays)
  - Main lecture

- MSc. Ioannis “John” Chiotellis
  - ioannis.chiotellis@gmail.com
  - Room number 02.09.059
  - Assistance and exercises
Lecturers

• PD Dr. Rudolph Triebel
• rudolph.triebel@in.tum.de
• Room number 02.09.059 (Fridays)
• Main lecture

Main affiliation (Mo - Thur):
Head of department for Perception and Cognition
Institute of Robotics and Mechatronics, DLR
rudolph.triebel@dlr.de
Aim of this Class

• Give a major **overview** of the most important machine learning methods

• Present relations to **current research** applications for most learning methods

• Explain some of the more **basic** techniques in more detail, others in less detail

• Provide a **complement** to other machine learning classes
Topics Covered

• Introduction (today)
• Regression
• Graphical Models (directed and undirected)
• Clustering
• Boosting and Bagging
• Metric Learning
• Convolutional Neural Networks and Deep Learning
• Kernel Methods
• Gaussian Processes
• Learning of Sequential Data
• Sampling Methods
• Variational Inference
• Online Learning
Literature

Recommended textbook for the lecture: Christopher M. Bishop: “Pattern Recognition and Machine Learning”

More detailed:

- “Gaussian Processes for Machine Learning” Rasmussen/Williams
- “Machine Learning - A Probabilistic Perspective” Murphy
The Tutorials

- **Bi-weekly** tutorial classes
- So far: one tutorial class, but we are trying to establish a second one
- Participation in tutorial classes and submission of solved assignment sheets is **free**
- In class, you have the opportunity to present your solution
- Assignments will be theoretical and practical problems
- First tutorial class: May 16
The Exam

- No “qualification” necessary for the final exam
- It will be a **written** exam
- So far, the date is not fixed yet, it will be announced within the next weeks
- In the exam, there will be more assignments than needed to reach the highest grade
Class Webpage

https://vision.in.tum.de/teaching/ss2017/mlcv17

- Contains the slides and assignments for download
- Also used for communication, in addition to email list
- Some further material will be developed in class
- Material from earlier semesters also available
- Video lectures from an earlier semester on YouTube
Why Machine Learning?
Typical Problems in Computer Vision

Image Segmentation

Object Classification

![Image Segmentation Diagram]

![Object Classification Diagram]
Typical Problems in Computer Vision

3D Shape Analysis, e.g. Shape Retrieval

Optical Character Recognition

“qnnivm”
Typical Problems in Computer Vision

Image compression

Noise reduction

... and many others, e.g.: optical flow, scene flow, 3D reconstruction, stereo matching, ...
Some Applications in Robotics

Detection of cars and pedestrians for autonomous cars

Semantic Mapping
What Makes These Problems Hard?

- It is very hard to express the relation from input to output with a mathematical model.
- Even if there was such a model, how should the parameters be set?
- A hand-crafted model is not general enough, it can not be used again in similar applications.
- There is often no one-to-one mapping from input to output.

**Idea:** extract the needed information from a data set of input - output pairs by optimizing an objective function.
Example Application of Learning in Robotics

• Most objects in the environment can be classified, e.g. with respect to their size, functionality, dynamic properties, etc.

• Robots need to interact with the objects (move around, manipulate, inspect, etc.) and with humans

• For all these tasks it is necessary that the robot knows to which class an object belongs
Learning = Optimization

• A natural way to do object classification is to first find a mapping from input data to object labels (“learning”) and then infer from the learned data a possible class for a new object.

• The area of machine learning deals with the formulation and investigates methods to do the learning automatically.

• It is essentially based on optimization methods

• Machine learning algorithms are widely used in robotics and computer vision
Mathematical Formulation

Suppose we are given a set \( \mathcal{X} \) of objects and a set \( \mathcal{Y} \) of object categories (classes). In the learning task we search for a mapping \( \varphi : \mathcal{X} \rightarrow \mathcal{Y} \) such that similar elements in \( \mathcal{X} \) are mapped to similar elements in \( \mathcal{Y} \).

Examples:

- Object classification: chairs, tables, etc.
- Optical character recognition
- Speech recognition

Important problem: Measure of similarity!
Categories of Learning

Unsupervised Learning
- clustering, density estimation

Supervised Learning
- learning from a training data set, inference on the test data

Reinforcement Learning
- no supervision, but a reward function

Discriminant Function
- no prob. formulation, learns a function from objects $\mathbf{x}$ to labels $\mathbf{y}$

Discriminative Model
- estimates the posterior $p(y_k | \mathbf{x})$ for each class

Generative Model
- est. the likelihoods $p(\mathbf{x} | y_k)$ and use Bayes rule for the post.
Supervised Learning is the main topic of this lecture!

Methods used in Computer Vision include:

- Regression
- Conditional Random Fields
- Boosting
- Deep Neural Networks
- Gaussian Processes
- Hidden Markov Models
In unsupervised learning, there is no **ground truth** information given.

Most Unsupervised Learning methods are based on Clustering.
Categories of Learning

Unsupervised Learning
- clustering, density estimation

Supervised Learning
- learning from a training data set, inference on the test data

Reinforcement Learning
- no supervision, but a reward function

Reinforcement Learning requires an action
- the reward defines the quality of an action
- mostly used in robotics (e.g. manipulation)
- can be dangerous, actions need to be “tried out”
- not handled in this course
Categories of Learning

Further distinctions are:

• online vs offline learning (both for supervised and unsupervised methods)
• semi-supervised learning (a combination of supervised and unsupervised learning)
• multiple instance / single instance learning
• multi-task / single-task learning
• …
Generative Model: Example

Nearest-neighbor classification:

• Given: data points \((x_1, t_1), (x_2, t_2), \ldots\)
• Rule: Each new data point is assigned to the class of its nearest neighbor in feature space

1. Training instances in feature space

\[ \phi(x_1), \phi(x_2), \ldots \]
Generative Model: Example

Nearest-neighbor classification:

- Given: data points $(x_1, t_1), (x_2, t_2), \ldots$
- Rule: Each new data point is assigned to the class of its nearest neighbor in feature space

2. Map new data point into feature space
Generative Model: Example

Nearest-neighbor classification:

• Given: data points \((x_1, t_1), (x_2, t_2), \ldots\)

• Rule: Each new data point is assigned to the class of its nearest neighbor in feature space

3. Compute the distances to the neighbors
Generative Model: Example

Nearest-neighbor classification:

• Given: data points \((x_1, t_1), (x_2, t_2), \ldots\)

• Rule: Each new data point is assigned to the class of its nearest neighbor in feature space

4. Assign the label of the nearest training instance
Generative Model: Example

Nearest-neighbor classification:

• General case: $K$ nearest neighbors
• We consider a sphere around each training instance that has a fixed volume $V$.

$K_k$: Number of points from class $k$ inside sphere

$N_k$: Number of all points from class $k$
Generative Model: Example

Nearest-neighbor classification:

- General case: $K$ nearest neighbors
- We consider a sphere around a training / test sample that has a fixed volume $V$.

- With this we can estimate:

$$p(x \mid y = k) = \frac{K_k}{N_k V}$$

“likelihood”

# points in sphere

- and likewise:

$$p(x) = \frac{K}{NV}$$

“uncond. prob.”

# all points

- using Bayes rule:

$$p(y = k \mid x) = \frac{p(x \mid y = k)p(y = k)}{p(x)} = \frac{K_k}{K}$$

“posterior”
Generative Model: Example

Nearest-neighbor classification:

• General case: $K$ nearest neighbors

$$p(y = k \mid x) = \frac{p(x \mid y = k)p(y = k)}{p(x)} = \frac{K_k}{K}$$

• To classify the new data point $x$ we compute the posterior for each class $k = 1,2,\ldots$ and assign the label that maximizes the posterior (MAP).

$$t := \arg \max_k p(y = k \mid x)$$
Summary

- Learning is usually a two-step process consisting in a \textit{training} and an \textit{inference} step.
- Learning is useful to extract \textit{semantic} information, e.g. about the objects in an environment.
- There are three main categories of learning: \textit{unsupervised}, \textit{supervised} and \textit{reinforcement} learning.
- Supervised learning can be split into \textit{discriminant function}, \textit{discriminant model}, and \textit{generative model} learning.
- An example for a generative model is \textit{nearest neighbor classification}.


Introduction to Probabilistic Reasoning
Motivation

Suppose a robot stops in front of a door. It has a sensor (e.g. a camera) to measure the state of the door (open or closed). **Problem**: the sensor may fail.
Motivation

**Question**: How can we obtain knowledge about the environment from sensors that may return incorrect results?

**Using Probabilities!**
Basics of Probability Theory

Definition 1.1: A sample space \( S \) is a set of outcomes of a given experiment.

Examples:

a) Coin toss experiment: \( S = \{H, T\} \)
   
b) Distance measurement: \( S = \mathbb{R}^+ \)

Definition 1.2: A random variable \( X \) is a function that assigns a real number to each element of \( S \).

Example: Coin toss experiment: \( H = 1, T = 0 \)

Values of random variables are denoted with small letters, e.g.: \( X = x \)
Discrete and Continuous

If $\mathcal{S}$ is countable then $X$ is a \textit{discrete} random variable, else it is a \textit{continuous} random variable.

The probability that $X$ takes on a certain value $x$ is a real number between 0 and 1. It holds:

$$\sum_{x} p(X = x) = 1$$

\text{Discrete case}

$$\int p(X = x) \, dx = 1$$

\text{Continuous case}
A Discrete Random Variable

Suppose a robot knows that it is in a room, but it does not know in \textit{which} room. There are 4 possibilities:

\textbf{Kitchen, Office, Bathroom, Living room}

Then the random variable $Room$ is discrete, because it can take on one of four values. The probabilities are, for example:

\[
\begin{align*}
P(\text{Room} = \text{kitchen}) &= 0.7 \\
P(\text{Room} = \text{office}) &= 0.2 \\
P(\text{Room} = \text{bathroom}) &= 0.08 \\
P(\text{Room} = \text{living room}) &= 0.02
\end{align*}
\]
A Continuous Random Variable

Suppose a robot travels 5 meters forward from a given start point. Its position $X$ is a continuous random variable with a *Normal distribution*:

$$p(X = x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2} \frac{(x-5)^2}{\sigma^2}}$$

**Shorthand:**

$$\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2} \frac{(x-\mu)^2}{\sigma^2}}$$

$N(x; \mu, \sigma^2)$
Joint and Conditional Probability

The **joint probability** of two random variables $X$ and $Y$ is the probability that the events $X = x$ and $Y = y$ occur at the same time:

$$p(X = x \text{ and } Y = y)$$

**Shorthand:**

- $p(X = x) \rightarrow p(x)$
- $p(X = x \text{ and } Y = y) \rightarrow p(x, y)$

**Definition 1.3:** The **conditional probability** of $X$ given $Y$ is defined as:

$$p(X = x \mid Y = y) = p(x \mid y) := \frac{p(x, y)}{p(y)}$$
Independency, Sum and Product Rule

Definition 1.4: Two random variables $X$ and $Y$ are independent iff:

$$p(x, y) = p(x)p(y)$$

For independent random variables $X$ and $Y$ we have:

$$p(x \mid y) = \frac{p(x, y)}{p(y)} = \frac{p(x)p(y)}{p(y)} = p(x)$$

Furthermore, it holds:

$$p(x) = \sum_{y} p(x, y) \quad p(x, y) = p(y \mid x)p(x)$$

"Sum Rule" \quad "Product Rule"
Law of Total Probability

**Theorem 1.1:** For two random variables \( X \) and \( Y \) it holds:

\[
p(x) = \sum_y p(x \mid y)p(y) \quad p(x) = \int p(x \mid y)p(y)\,dy
\]

Discrete case  
Continuous case

The process of obtaining \( p(x) \) from \( p(x, y) \) by summing or integrating over all values of \( y \) is called **Marginalisation**
Bayes Rule

**Theorem 1.2:** For two random variables $X$ and $Y$ it holds:

$$p(x \mid y) = \frac{p(y \mid x)p(x)}{p(y)}$$

**Proof:**

I. $p(x \mid y) = \frac{p(x, y)}{p(y)}$ (definition)

II. $p(y \mid x) = \frac{p(x, y)}{p(x)}$ (definition)

III. $p(x, y) = p(y \mid x)p(x)$ (from II.)
Bayes Rule: Background Knowledge

For \( p(y \mid z) \neq 0 \) it holds:

\[
p(x \mid y, z) = \frac{p(y \mid x, z)p(x \mid z)}{p(y \mid z)}
\]

Shorthand:

\[
p(y \mid z)^{-1} \rightarrow \eta
\]

“Normalizer”

\[
p(x \mid y, z) = \eta \ p(y \mid x, z)p(x \mid z)
\]
Computing the Normalizer

\[ p(x \mid y) = \frac{p(y \mid x)p(x)}{p(y)} \]

Bayes rule

\[ p(y) = \sum_x p(y \mid x)p(x) \]

Total probability

\[ p(x \mid y) = \frac{p(y \mid x)p(x)}{\sum_{x'} p(y \mid x')p(x')} \]

\( p(x \mid y) \) can be computed without knowing \( p(y) \)
Conditional Independence

**Definition 1.5:** Two random variables $X$ and $Y$ are *conditional independent* given a third random variable $Z$ iff:

$$p(x, y \mid z) = p(x \mid z)p(y \mid z)$$

This is equivalent to:

$$p(x \mid z) = p(x \mid y, z) \quad \text{and} \quad p(y \mid z) = p(y \mid x, z)$$
Expectation and Covariance

**Definition 1.6:** The *expectation* of a random variable $X$ is defined as:

$$E[X] = \sum_{x} x \ p(x) \quad \text{(discrete case)}$$

$$E[X] = \int x \ p(x) \ dx \quad \text{(continuous case)}$$

**Definition 1.7:** The *covariance* of a random variable $X$ is defined as:

Mathematical Formulation of Our Example

We define two binary random variables: \( z \) and open, where \( z \) is “light on” or “light off”. Our question is: What is \( p(\text{open} \mid z) \)?
Causal vs. Diagnostic Reasoning

- Searching for $p(\text{open} \mid z)$ is called \textit{diagnostic reasoning}.
- Searching for $p(z \mid \text{open})$ is called \textit{causal reasoning}.
- Often causal knowledge is easier to obtain.
- Bayes rule allows us to use causal knowledge:

$$p(\text{open} \mid z) = \frac{p(z \mid \text{open})p(\text{open})}{p(z)} = \frac{p(z \mid \text{open})p(\text{open})}{p(z \mid \text{open})p(\text{open}) + p(z \mid \neg \text{open})p(\neg \text{open})}$$
Example with Numbers

Assume we have this sensor model:

\[
p(z \mid \text{open}) = 0.6 \quad \quad p(z \mid \neg \text{open}) = 0.3
\]

and:

\[
p(\text{open}) = p(\neg \text{open}) = 0.5 \quad \quad \text{“Prior prob.”}
\]

then:

\[
p(\text{open} \mid z) = \frac{p(z \mid \text{open})p(\text{open})}{p(z \mid \text{open})p(\text{open}) + p(z \mid \neg \text{open})p(\neg \text{open})} = \frac{0.6 \cdot 0.5}{0.6 \cdot 0.5 + 0.3 \cdot 0.5} = \frac{2}{3} = 0.67
\]

“\(z\) raises the probability that the door is open”
Summary

• **Probabilistic reasoning** is necessary to deal with uncertain information, e.g. sensor measurements

• Using **Bayes rule**, we can do diagnostic reasoning based on causal knowledge

• This is used to infer knowledge from imprecise (“noisy”) data input