Machine Learning – Lecture 1

Introduction

09.10.2019

Bastian Leibe
RWTH Aachen
http://www.vision.rwth-aachen.de/
leibe@vision.rwth-aachen.de

Organization

• Lecturer
  > Prof. Bastian Leibe (leibe@vision.rwth-aachen.de)

• Assistants
  > Ali Athar (athar@vision.rwth-aachen.de)
  > Sabarinath Mahadevan (mahadevan@vision.rwth-aachen.de)

• Course webpage
  > http://www.vision.rwth-aachen.de/courses/
  > Slides will be made available on the webpage and in moodle
  > Lecture recordings as screencasts will be available via moodle

• Please subscribe to the lecture in rwth online!
  > Important to get email announcements and moodle access!

Language

• Official course language will be English
  > If at least one English-speaking student is present.
  > If not... you can choose.

• However...
  > Please tell me when I'm talking too fast or when I should repeat
  something in German for better understanding!
  > You may at any time ask questions in German!
  > You may turn in your exercises in German.
  > You may answer exam questions in German.

Exercises and Supplementary Material

• Exercises
  > Typically 1 exercise sheet every 2 weeks.
  > Pen & paper and programming exercises
    > Python for first exercise slots
    > TensorFlow for Deep Learning part
  > Hands-on experience with the algorithms from the lecture.
  > Send your solutions the night before the exercise class.
  > Need to reach > 50% of the points to qualify for the exam!

• Teams are encouraged!
  > You can form teams of up to 4 people for the exercises.
  > Each team should only turn in one solution via L2P.
  > But list the names of all team members in the submission.

Course Webpage

http://www.vision.rwth-aachen.de/courses/
Textbooks

- The first half of the lecture is covered in Bishop’s book.
- For Deep Learning, we will use Goodfellow & Bengio.

Machine Learning

- Statistical Machine Learning
  - Principles, methods, and algorithms for learning and prediction on the basis of past evidence
- Already everywhere
  - Speech recognition (e.g. Siri)
  - Machine translation (e.g. Google Translate)
  - Computer vision (e.g. Face detection)
  - Text filtering (e.g. Email spam filters)
  - Operation systems (e.g. Caching)
  - Fraud detection (e.g. Credit cards)
  - Game playing (e.g. Alpha Go)
  - Robotics (everywhere)

How to Find Us

- Office:
  - UMIC Research Centre
  - Mies-van-der-Rohe-Strasse 15, room 124
- Office hours
  - If you have questions about the lecture, contact Paul or Sabarinath.
  - My regular office hours will be announced (additional slots are available upon request)
  - Send us an email before to confirm a time slot.

Questions are welcome!

What Is Machine Learning Useful For?

- Automatic Speech Recognition
- Computer Vision (Object Recognition, Segmentation, Scene Understanding)
- Information Retrieval (Retrieval, Categorization, Clustering, ...)

Machine Learning

• Statistical Machine Learning
  • Principles, methods, and algorithms for learning and prediction on the basis of past evidence
• Already everywhere
  • Speech recognition (e.g. Siri)
  • Machine translation (e.g. Google Translate)
  • Computer vision (e.g. Face detection)
  • Text filtering (e.g. Email spam filters)
  • Operation systems (e.g. Caching)
  • Fraud detection (e.g. Credit cards)
  • Game playing (e.g. Alpha Go)
  • Robotics (everywhere)
What Is Machine Learning Useful For?

- Financial Prediction (Time series analysis, ...)
- Medical Diagnosis (Inference from partial observations)
- Bioinformatics (Modelling gene microarray data, ...)
- Autonomous Driving (DARPA Grand Challenge, ...)

And you might have heard of…

Deep Learning

Machine Learning

- Goal
  - Machines that learn to perform a task from experience

- Why?
  - Crucial component of every intelligent/autonomous system
  - Important for a system’s adaptability
  - Important for a system’s generalization capabilities
  - Attempt to understand human learning

Slide adapted from Zoubin Gharamani
Image from Kevin Murphy
Machine Learning: Core Questions

• **Learning to perform a task from experience**

  • Learning
    - Most important part here!
    - We do not want to encode the knowledge ourselves.
    - The machine should learn the relevant criteria automatically from past observations and adapt to the given situation.

  • Tools
    - Statistics
    - Probability theory
    - Decision theory
    - Information theory
    - Optimization theory

  • Task
    - Can often be expressed through a mathematical function
    - \( y = f(x, w) \)
      - \( x \): Input
      - \( y \): Output
      - \( w \): Parameters
    - Classification vs. Regression
      - Regression: continuous \( y \)
      - Classification: discrete \( y \)
        - E.g. class membership, sometimes also posterior probability

  • Classification vs. Regression
    - Continuous output
    - Discrete output

Example: Regression

• Automatic control of a vehicle

\[ f(x; w) \]

Examples: Classification

• Email filtering
  \( x \in \{a-z\} \rightarrow y \in \{\text{important, spam}\} \)

• Character recognition

• Speech recognition

Machine Learning: Core Problems

• Input \( x \):

  • Features
    - Invariance to irrelevant input variations
    - Selecting the “right” features is crucial
    - Encoding and use of “domain knowledge”
    - Higher-dimensional features are more discriminative.

• Curse of dimensionality
  - Complexity increases exponentially with number of dimensions.
Machine Learning: Core Questions

- **Learning to perform a task from experience**

- Performance: “99% correct classification”
  - Of what???
  - Characters? Words? Sentences?
  - Speaker/writer independent?
  - Over what data set?
  - ...

- “The car drives without human intervention 99% of the time on country roads”

Machine Learning: Core Questions

- **Learning is optimization of** \( y = f(x; w) \)
  - \( w \): characterizes the family of functions
  - \( w \): indexes the space of hypotheses
  - \( w \): vector, connection matrix, graph, ...

Course Outline

- **Fundamentals**
  - Bayes Decision Theory
  - Probability Density Estimation

- **Classification Approaches**
  - Linear Discriminants
  - Support Vector Machines
  - Ensemble Methods & Boosting

- **Deep Learning**
  - Foundations
  - Convolutional Neural Networks
  - Recurrent Neural Networks

Topics of This Lecture

- **Review: Probability Theory**
  - Probabilities
  - Probability densities
  - Expectations and covariances

- **Bayes Decision Theory**
  - Basic concepts
  - Minimizing the misclassification rate
  - Minimizing the expected loss
  - Discriminant functions
Probability Theory

"Probability theory is nothing but common sense reduced to calculation."

Pierre-Simon de Laplace, 1749-1827

Example: apples and oranges
- We have two boxes to pick from.
- Each box contains both types of fruit.
- What is the probability of picking an apple?

Formalization
- Let $B \in \{r, b\}$ be a random variable for the box we pick.
- Let $F \in \{a, o\}$ be a random variable for the type of fruit we get.
- Suppose we pick the red box 40% of the time. We write this as $p(B = r) = 0.4$
- The probability of picking an apple given a choice for the box is $p(F = a \mid B = r) = 0.25$
- What is the probability of picking an apple?
- $p(F = a)$?

More general case
- Consider two random variables $X \in \{x_i\}$ and $Y \in \{y_j\}$.
- Consider $N$ trials and let:
  - $n_{ij} = \#\{X = x_i \land Y = y_j\}$
  - $c_i = \#\{X = x_i\}$
  - $r_j = \#\{Y = y_j\}$
- Then we can derive:
  - Joint probability:
    - $p(X = x_i, Y = y_j) = \frac{n_{ij}}{N}$
  - Marginal probability:
    - $p(X = x_i) = \frac{c_i}{N}$
    - $p(Y = y_j) = \frac{r_j}{N}$
  - Conditional probability:
    - $p(Y = y_j \mid X = x_i) = \frac{n_{ij}}{c_i}$

The Rules of Probability

Thus we have
- Sum rule
  - $p(X) = \sum_{Y} p(X, Y)$
- Product rule
  - $p(X, Y) = p(Y \mid X) p(X)$

From those, we can derive
- Bayes’ theorem:
  - $p(Y \mid X) = \frac{p(X, Y)}{p(X)}$
  - where
    - $p(X) = \sum_{Y} p(X, Y)$

Probability Densities

- Probabilities over continuous variables are defined over their probability density function (pdf) $p(x)$
  - $p(x) \in (a, b)$ $= \int_{a}^{b} p(x) \, dx$

- The probability that $x$ lies in the interval $(-\infty, z)$ is given by the cumulative distribution function
  - $P(z) = \int_{-\infty}^{z} p(x) \, dx$
**Expectations**

- The average value of some function $f(x)$ under a probability distribution $p(x)$ is called its expectation.
  \[
  \mathbb{E}[f] = \sum_x p(x) f(x) \quad \text{discrete case} \\
  \mathbb{E}[f] = \int p(x) f(x) \, dx \quad \text{continuous case}
  \]

- If we have a finite number $N$ of samples drawn from a pdf, the expectation can be approximated by
  \[
  \mathbb{E}[f] \approx \frac{1}{N} \sum_{n=1}^{N} f(x_n)
  \]

- We can also consider a conditional expectation
  \[
  \mathbb{E}_x[f] = \sum_x p(x|y) f(x)
  \]

**Variances and Covariances**

- The variance provides a measure how much variability there is in $f(x)$ around its mean value.
  \[
  \text{var}[f] = \mathbb{E} \left[ (f(x) - \mathbb{E}[f(x)])^2 \right] = \mathbb{E}[f(x)^2] - \mathbb{E}[f(x)]^2
  \]

- For two random variables $x$ and $y$, the covariance is defined by
  \[
  \text{cov}[x,y] = \mathbb{E}_{x,y} \left[ (x - \mathbb{E}[x])(y - \mathbb{E}[y]) \right] = \mathbb{E}[xy] - \mathbb{E}[x] \mathbb{E}[y]
  \]

- If $x$ and $y$ are vectors, the result is a covariance matrix
  \[
  \text{cov}[x,y] = \mathbb{E}_{x,y} \left[ (x - \mathbb{E}[x])(y^T - \mathbb{E}[y])^T \right]
  \]

- $\text{cov}[x,y] = \mathbb{E}_{x,y} \left[ xy^T - \mathbb{E}[x] \mathbb{E}[y]^T \right]
  \]

**Bayes Decision Theory**

- Example: handwritten character recognition

- Goal:
  - Classify a new letter such that the probability of misclassification is minimized.

**Bayes Decision Theory**

- **Concept 1: Priors (a priori probabilities)**
  - What we can tell about the probability before seeing the data.
  - Example:
    \[
    \begin{align*}
    C_1 &= a & p(C_1) &= 0.75 \\
    C_2 &= b & p(C_2) &= 0.25
    \end{align*}
    \]
  - In general:
    \[
    \sum_i p(C_i) = 1
    \]

- **Concept 2: Conditional probabilities**
  - Let $x$ be a feature vector.
  - $x$ measures/describes certain properties of the input.
    - E.g. number of black pixels, aspect ratio, ...
  - $p(x|C_i)$ describes its likelihood for class $C_i$. 
Bayes Decision Theory

- Example:
  \[
p(x|a) \quad p(x|b)
  \]
  \[x = 15\]

- Question:
  \[\text{Which class?} \]
  \[\text{Since } p(x|b) \text{ is much smaller than } p(x|a) \text{ the decision should be 'a' here.}\]

- Question:
  \[\text{Which class?} \]
  \[\text{Since } p(x|a) \text{ is much smaller than } p(x|b) \text{, the decision should be 'b' here.}\]

Bayes Decision Theory

- Concept 3: Posterior probabilities
  \[p(C_k|x)\]
  We are typically interested in the \textit{a posteriori} probability, i.e. the probability of class \(C_k\) given the measurement vector \(x\).

- Bayes’ Theorem:
  \[p(C_k|x) = \frac{p(x|C_k)p(C_k)}{p(x)} = \frac{p(x|C_k)p(C_k)}{\sum_{i} p(x|C_i)p(C_i)}\]

- Interpretation
  \[\text{Posterior} = \frac{\text{Likelihood} \times \text{Prior}}{\text{Normalization Factor}}\]

Bayesian Decision Theory

- Goal: Minimize the probability of a misclassification
  \[p(\text{mistake}) = p(x \in R_1, C_2) + p(x \in R_2, C_1)\]
  \[= \int_{R_1} p(x|C_2)dx + \int_{R_2} p(x|C_1)dx\]
  \[= \int_{R_1} p(C_2|x)p(x)dx + \int_{R_2} p(C_1|x)p(x)dx\]
Bayes Decision Theory

- Optimal decision rule
  - Decide for $C_1$ if
    \[ p(C_1 | x) > p(C_2 | x) \]
  - This is equivalent to
    \[ p(x | C_1) p(C_1) > p(x | C_2) p(C_2) \]
  - Which is again equivalent to (Likelihood-Ratio test)
    \[ \frac{p(x | C_1)}{p(x | C_2)} > \frac{p(C_2)}{p(C_1)} \]
  - Decision threshold $\theta$

Generalization to More Than 2 Classes

- Decide for class $k$ whenever it has the greatest posterior probability of all classes:
  \[ p(C_k | x) > p(C_j | x) \quad \forall j \neq k \]
  \[ p(x | C_k) p(C_k) > p(x | C_j) p(C_j) \quad \forall j \neq k \]
- Likelihood-ratio test
  \[ \frac{p(x | C_k)}{p(x | C_j)} > \frac{p(C_j)}{p(C_k)} \quad \forall j \neq k \]

Classifying with Loss Functions

- Generalization to decisions with a loss function
  - Differentiate between the possible decisions and the possible true classes.
  - Example: medical diagnosis
    - Decisions: sick or healthy (or, further examination necessary)
    - Classes: patient is sick or healthy
  - The cost may be asymmetric:
    \[ \text{loss}(\text{decision} = \text{healthy} | \text{patient} = \text{sick}) > \text{loss}(\text{decision} = \text{sick} | \text{patient} = \text{healthy}) \]

Classifying with Loss Functions

- Loss functions may be different for different actors.
  - Example:
    \[ L_{\text{stocktrader}}(\text{subprime}) = \begin{pmatrix} -\frac{1}{2} \text{gain} & 0 \\ 0 & 0 \end{pmatrix} \]
    \[ L_{\text{bank}}(\text{subprime}) = \begin{pmatrix} -\frac{1}{2} \text{gain} & 0 \\ \infty & 0 \end{pmatrix} \]
  - Different loss functions may lead to different Bayes optimal strategies.

Minimizing the Expected Loss

- Optimal solution is the one that minimizes the loss.
  - But: loss function depends on the true class, which is unknown.
- Solution: Minimize the expected loss
  \[ E[L] = \sum_k \sum_j \int_{R_j} L_{kj} p(x, C_k) \, dx \]
  - This can be done by choosing the regions $R_j$ such that
    \[ E[L] = \sum_k L_{kj} p(C_k | x) \]
    which is easy to do once we know the posterior class probabilities $p(C_k | x)$
Minimizing the Expected Loss

- **Example:**
  - 2 Classes: $C_1, C_2$
  - 2 Decision: $\alpha_1, \alpha_2$
  - Loss function: $L(\alpha_j|C_k) = L_k$
  - Expected loss (= risk $R$) for the two decisions:
    $E_{\alpha_1}[L] = R(\alpha_1|x) = L_{11}p(C_1|x) + L_{21}p(C_2|x)$
    $E_{\alpha_2}[L] = R(\alpha_2|x) = L_{12}p(C_1|x) + L_{22}p(C_2|x)$

- **Goal:** Decide such that expected loss is minimized
  - i.e. decide $\alpha_i$ if $R(\alpha_2|x) > R(\alpha_1|x)$

The Reject Option

- Classification errors arise from regions where the largest posterior probability $p(C_k|x)$ is significantly less than 1.
  - These are the regions where we are relatively uncertain about class membership.
  - For some applications, it may be better to reject the automatic decision entirely in such a case and e.g. consult a human expert.

Different Views on the Decision Problem

- $y_k(x) \propto p(x|C_k)p(C_k)$
  - First determine the class-conditional densities for each class individually and separately infer the prior class probabilities.
  - Then use Bayes’ theorem to determine class membership.
  - **Generative methods**
- $y_k(x) = p(C_k|x)$
  - First solve the inference problem of determining the posterior class probabilities.
  - Then use decision theory to assign each new $x$ to its class.
  - **Discriminative methods**
- **Alternative**
  - Directly find a discriminant function $y_k(x)$ which maps each input $x$ directly onto a class label.
References and Further Reading

- More information, including a short review of Probability theory and a good introduction in Bayes Decision Theory can be found in Chapters 1.1, 1.2 and 1.5 of

  Christopher M. Bishop
  Pattern Recognition and Machine Learning
  Springer, 2006