Machine Learning - Lecture 3

Probability Density Estimation II

21.04.2015

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

leibe@vision.rwth-aachen.de

Many slides adapted from B. Schiele
Course Outline

• Fundamentals (2 weeks)
  ➢ Bayes Decision Theory
  ➢ Probability Density Estimation

• Discriminative Approaches (5 weeks)
  ➢ Linear Discriminant Functions
  ➢ Support Vector Machines
  ➢ Ensemble Methods & Boosting
  ➢ Randomized Trees, Forests & Ferns

• Generative Models (4 weeks)
  ➢ Bayesian Networks
  ➢ Markov Random Fields
Topics of This Lecture

• Recap: Bayes Decision Theory

• Parametric Methods
  - Recap: Maximum Likelihood approach
  - Bayesian Learning

• Non-Parametric Methods
  - Histograms
  - Kernel density estimation
  - K-Nearest Neighbors
  - k-NN for Classification
  - Bias-Variance tradeoff
Recap: 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 $\theta$

Slide credit: Bernt Schiele
Recap: Bayes Decision Theory

- Decision regions: $R_1, R_2, R_3, \ldots$
Recap: Classifying with Loss Functions

- We can formalize the intuition that different decisions have different weights by introducing a loss matrix $L_{kj}$

$$L_{kj} = \text{loss for decision } C_j \text{ if truth is } C_k.$$ 

- Example: cancer diagnosis

$$L_{\text{cancer diagnosis}} = \begin{pmatrix} \text{cancer} \\ \text{normal} \end{pmatrix} \begin{pmatrix} 0 & 1000 \\ 1 & 0 \end{pmatrix}$$
Recap: 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

  \[ \mathbb{E}[L] = \sum_{k} \sum_{j} \int_{R_j} L_{kj} p(x, C_k) \, dx \]

• This can be done by choosing the regions \( R_j \) such that

  \[ \mathbb{E}[L] = \sum_{k} L_{kj} p(C_k | x) \]

\[ \Rightarrow \text{Adapted decision rule:} \]

\[ \frac{p(x | C_1)}{p(x | C_2)} > \frac{(L_{21} - L_{22}) p(C_2)}{(L_{12} - L_{11}) p(C_1)} \]

B. Leibe
Recap: Gaussian (or Normal) Distribution

- **One-dimensional case**
  - Mean $\mu$
  - Variance $\sigma^2$

$$N(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma}} \exp \left\{ -\frac{(x - \mu)^2}{2\sigma^2} \right\}$$

- **Multi-dimensional case**
  - Mean $\mu$
  - Covariance $\Sigma$

$$N(x|\mu, \Sigma) = \frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} \exp \left\{ -\frac{1}{2}(x - \mu)^T \Sigma^{-1} (x - \mu) \right\}$$

Image source: C.M. Bishop, 2006
Recap: Maximum Likelihood Approach

• Computation of the likelihood
  - Single data point: \( p(x_n|\theta) \)
  - Assumption: all data points \( X = \{x_1, \ldots, x_n\} \) are independent
  \[
  L(\theta) = p(X|\theta) = \prod_{n=1}^{N} p(x_n|\theta)
  \]
  - Log-likelihood
  \[
  E(\theta) = -\ln L(\theta) = -\sum_{n=1}^{N} \ln p(x_n|\theta)
  \]

• Estimation of the parameters \( \theta \) (Learning)
  - Maximize the likelihood (=minimize the negative log-likelihood)
  \[ \Rightarrow \text{Take the derivative and set it to zero.} \]
  \[
  \frac{\partial}{\partial \theta} E(\theta) = -\sum_{n=1}^{N} \frac{\partial}{\partial \theta} p(x_n|\theta) \frac{p(x_n|\theta)}{p(x_n|\theta)} = 0
  \]
Topics of This Lecture

• Recap: Bayes Decision Theory

• Parametric Methods
  ➢ Recap: Maximum Likelihood approach
  ➢ Bayesian Learning

• Non-Parametric Methods
  ➢ Histograms
  ➢ Kernel density estimation
  ➢ K-Nearest Neighbors
  ➢ k-NN for Classification
  ➢ Bias-Variance tradeoff
Recap: Maximum Likelihood - Limitations

- Maximum Likelihood has several significant limitations
  - It systematically underestimates the variance of the distribution!
  - E.g. consider the case
    \[ N = 1, X = \{x_1\} \]

  \[ \Rightarrow \text{Maximum-likelihood estimate:} \]
  \[ \hat{\sigma} = 0! \]

  - We say ML *overfits to the observed data*.
  - We will still often use ML, but it is important to know about this effect.
Deeper Reason

• Maximum Likelihood is a **Frequentist** concept
  - In the **Frequentist view**, probabilities are the frequencies of random, repeatable events.
  - These frequencies are fixed, but can be estimated more precisely when more data is available.

• This is in contrast to the **Bayesian** interpretation
  - In the **Bayesian view**, probabilities quantify the uncertainty about certain states or events.
  - This uncertainty can be revised in the light of new evidence.

• Bayesians and Frequentists do not like each other too well...
Bayesian vs. Frequentist View

• To see the difference...
  - Suppose we want to estimate the uncertainty whether the Arctic ice cap will have disappeared by the end of the century.
  - This question makes no sense in a Frequentist view, since the event cannot be repeated numerous times.
  - In the Bayesian view, we generally have a prior, e.g. from calculations how fast the polar ice is melting.
  - If we now get fresh evidence, e.g. from a new satellite, we may revise our opinion and update the uncertainty from the prior.

\[ \text{Posterior} \propto \text{Likelihood} \times \text{Prior} \]

- This generally allows to get better uncertainty estimates for many situations.

• Main Frequentist criticism
  - The prior has to come from somewhere and if it is wrong, the result will be worse.
Bayesian Approach to Parameter Learning

- Conceptual shift
  - Maximum Likelihood views the true parameter vector $\theta$ to be unknown, but fixed.
  - In Bayesian learning, we consider $\theta$ to be a random variable.

- This allows us to use knowledge about the parameters $\theta$
  - i.e., to use a prior for $\theta$
  - Training data then converts this prior distribution on $\theta$ into a posterior probability density.

  - The prior thus encodes knowledge we have about the type of distribution we expect to see for $\theta$. 

Slide adapted from Bernt Schiele
Bayesian Learning Approach

- Bayesian view:
  - Consider the parameter vector $\theta$ as a random variable.
  - When estimating the parameters from a dataset $X$, we compute

$$p(x|X) = \int p(x, \theta|X)d\theta$$

$$p(x, \theta|X) = p(x|\theta, X)p(\theta|X)$$

Assumption: given $\theta$, this doesn’t depend on $X$ anymore

$$p(x|X) = \int p(x|\theta)p(\theta|X)d\theta$$

This is entirely determined by the parameter $\theta$
(i.e., by the parametric form of the pdf).

Slide adapted from Bernt Schiele
Bayesian Learning Approach

\[ p(x|X) = \int p(x|\theta)p(\theta|X)d\theta \]

\[ p(\theta|X) = \frac{p(X|\theta)p(\theta)}{p(X)} = \frac{p(\theta)}{p(X)}L(\theta) \]

\[ p(X) = \int p(X|\theta)p(\theta)d\theta = \int L(\theta)p(\theta)d\theta \]

• Inserting this above, we obtain

\[ p(x|X) = \int \frac{p(x|\theta)L(\theta)p(\theta)}{p(X)}d\theta = \int \frac{p(x|\theta)L(\theta)p(\theta)}{\int L(\theta)p(\theta)d\theta}d\theta \]
Bayesian Learning Approach

Discussion

If we now plug in a (suitable) prior \( p(\theta) \), we can estimate \( p(x|X) \) from the data set \( X \).
Bayesian Density Estimation

- Discussion

\[ p(x|X) = \int p(x|\theta)p(\theta|X)d\theta = \int \frac{p(x|\theta)L(\theta)p(\theta)}{\int L(\theta)p(\theta)d\theta}d\theta \]

- The probability \( p(\theta|X) \) makes the dependency of the estimate on the data explicit.

- If \( p(\theta|X) \) is very small everywhere, but is large for one \( \hat{\theta} \), then

\[ p(x|X) \approx p(x|\hat{\theta}) \]

\[ \Rightarrow \] In this case, the estimate is determined entirely by \( \hat{\theta} \).

\[ \Rightarrow \] The more uncertain we are about \( \theta \), the more we average over all parameter values.

Slide credit: Bernt Schiele
Bayesian Density Estimation

• Problem
  - In the general case, the integration over $\theta$ is not possible (or only possible stochastically).

• Example where an analytical solution is possible
  - Normal distribution for the data, $\sigma^2$ assumed known and fixed.
  - Estimate the distribution of the mean:
    \[
    p(\mu|X) = \frac{p(X|\mu)p(\mu)}{p(X)}
    \]
  - Prior: We assume a Gaussian prior over $\mu$,
    \[
    p(\mu) = \mathcal{N}(\mu|\mu_0, \sigma_0^2).
    \]
Bayesian Learning Approach

- Sample mean: \( \bar{x} = \frac{1}{N} \sum_{n=1}^{N} x_n \)

- Bayes estimate:
  \[
  \mu_N = \frac{\sigma^2 \mu_0 + N \sigma_0^2 \bar{x}}{\sigma^2 + N \sigma_0^2}
  \]
  \[
  \frac{1}{\sigma^2_N} = \frac{1}{\sigma_0^2} + \frac{N}{\sigma^2}
  \]

- Note:
<table>
<thead>
<tr>
<th>( N = 0 )</th>
<th>( N \to \infty )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \mu_N )</td>
<td>( \mu_0 )</td>
</tr>
<tr>
<td>( \sigma^2_N )</td>
<td>( \sigma_0^2 )</td>
</tr>
</tbody>
</table>

Image source: C.M. Bishop, 2006
Summary: ML vs. Bayesian Learning

- **Maximum Likelihood**
  - Simple approach, often analytically possible.
  - Problem: estimation is biased, tends to overfit to the data.
    - $\Rightarrow$ Often needs some correction or regularization.
  - But:
    - Approximation gets accurate for $N \to \infty$.

- **Bayesian Learning**
  - General approach, avoids the estimation bias through a prior.
  - Problems:
    - Need to choose a suitable prior (not always obvious).
    - Integral over $\theta$ often not analytically feasible anymore.
  - But:
    - Efficient stochastic sampling techniques available.

*In this lecture, we’ll use both concepts wherever appropriate*
Topics of This Lecture

- Recap: Bayes Decision Theory

- Parametric Methods
  - Recap: Maximum Likelihood approach
  - Bayesian Learning

- Non-Parametric Methods
  - Histograms
  - Kernel density estimation
  - K-Nearest Neighbors
  - k-NN for Classification
  - Bias-Variance tradeoff
Non-Parametric Methods

- Non-parametric representations
  - Often the functional form of the distribution is unknown

- Estimate probability density from data
  - Histograms
  - Kernel density estimation (Parzen window / Gaussian kernels)
  - k-Nearest-Neighbor

Slide credit: Bernt Schiele
Histograms

- Basic idea:
  - Partition the data space into distinct bins with widths \( \Delta_i \) and count the number of observations, \( n_i \), in each bin.
  
  \[
  p_i = \frac{n_i}{N \Delta_i}
  \]

  - Often, the same width is used for all bins, \( \Delta_i = \Delta \).
  - This can be done, in principle, for any dimensionality \( D \)...

...but the required number of bins grows exponentially with \( D \)!
Histograms

- The bin width $\Delta$ acts as a smoothing factor.

- Not smooth enough

- About OK

- Too smooth

Image source: C.M. Bishop, 2006
Summary: Histograms

• Properties
  - Very general. In the limit \( N \to \infty \), every probability density can be represented.
  - No need to store the data points once histogram is computed.
  - Rather brute-force

• Problems
  - High-dimensional feature spaces
    - \( D \)-dimensional space with \( M \) bins/dimension will require \( M^D \) bins!
      \( \Rightarrow \) Requires an exponentially growing number of data points
      \( \Rightarrow \) “Curse of dimensionality”
  - Discontinuities at bin edges
  - Bin size?
    - too large: too much smoothing
    - too small: too much noise
Statistically Better-Founded Approach

- Data point $x$ comes from pdf $p(x)$
  - Probability that $x$ falls into small region $\mathcal{R}$
    \[ P = \int_{\mathcal{R}} p(y) dy \]

- If $\mathcal{R}$ is sufficiently small, $p(x)$ is roughly constant
  - Let $V$ be the volume of $\mathcal{R}$
    \[ P = \int_{\mathcal{R}} p(y) dy \approx p(x)V \]

- If the number $N$ of samples is sufficiently large, we can estimate $P$ as
  \[ P = \frac{K}{N} \quad \Rightarrow \quad p(x) \approx \frac{K}{NV} \]

Slide credit: Bernt Schiele
Statistically Better-Founded Approach

\[ p(x) \approx \frac{K}{NV} \]

- Kernel Methods
- K-Nearest Neighbor

**Kernel methods**
- Example: Determine the number \( K \) of data points inside a fixed window...

Slide credit: Bernt Schiele
Kernel Methods

- Parzen Window
  - Hypercube of dimension $D$ with edge length $h$:

  $$k(u) = \begin{cases} 
  1, & |u_i| \leq \frac{1}{2}, \quad i = 1, \ldots, D \\
  0, & \text{else} 
  \end{cases}$$

  “Kernel function”

  $$K = \sum_{n=1}^{N} k(\frac{x - x_n}{h})$$

  $$V = \int k(u) du = h^d$$

  - Probability density estimate:

  $$p(x) \approx \frac{K}{NV} = \frac{1}{Nh^D} \sum_{n=1}^{N} k(\frac{x - x_n}{h})$$

  Slide credit: Bernt Schiele
Kernel Methods: Parzen Window

- Interpretations
  1. We place a kernel window $k$ at location $x$ and count how many data points fall inside it.
  2. We place a kernel window $k$ around each data point $x_n$ and sum up their influences at location $x$.
     \[ \Rightarrow \text{Direct visualization of the density.} \]

- Still, we have artificial discontinuities at the cube boundaries...
  - We can obtain a smoother density model if we choose a smoother kernel function, e.g. a Gaussian
Kernel Methods: Gaussian Kernel

- Gaussian kernel
  - Kernel function
    \[ k(u) = \frac{1}{(2\pi h^2)^{1/2}} \exp \left\{ -\frac{u^2}{2h^2} \right\} \]

  \[ K = \sum_{n=1}^{N} k(x - x_n) \]
  \[ V = \int k(u) du = 1 \]

- Probability density estimate
  \[ p(x) \approx \frac{K}{NV} = \frac{1}{N} \sum_{n=1}^{N} \frac{1}{(2\pi)^{D/2}h} \exp \left\{ -\frac{\|x - x_n\|^2}{2h^2} \right\} \]
Gauss Kernel: Examples

not smooth enough

about OK

too smooth

$h$ acts as a smoother.

Image source: C.M. Bishop, 2006
Kernel Methods

• In general
  - Any kernel such that
    \[ k(u) \geq 0, \quad \int k(u) \, du = 1 \]
    can be used. Then
    \[
    K = \sum_{n=1}^{N} k(x - x_n)
    \]
    And we get the probability density estimate
    \[
    p(x) \approx \frac{K}{NV} = \frac{1}{N} \sum_{n=1}^{N} k(x - x_n)
    \]

Slide adapted from Bernt Schiele
Statistically Better-Founded Approach

\[ p(x) \approx \frac{K}{NV} \]

- **Kernel Methods**
  - fixed \( V \)
  - determine \( K \)

- **K-Nearest Neighbor**
  - fixed \( K \)
  - determine \( V \)

- **K-Nearest Neighbor**
  - Increase the volume \( V \) until the \( K \) next data points are found.
K-Nearest Neighbor

- Nearest-Neighbor density estimation
  - Fix $K$, estimate $V$ from the data.
  - Consider a hypersphere centred on $x$ and let it grow to a volume $V^*$ that includes $K$ of the given $N$ data points.
  - Then
    \[ p(x) \sim \frac{K}{NV^*}. \]

- Side note
  - Strictly speaking, the model produced by K-NN is not a true density model, because the integral over all space diverges.
  - E.g. consider $K = 1$ and a sample exactly on a data point $x = x_j$. 
k-Nearest Neighbor: Examples

- not smooth enough
- about OK
- too smooth

$K$ acts as a smoother.

Image source: C.M. Bishop, 2006
Summary: Kernel and k-NN Density Estimation

- **Properties**
  - Very general. In the limit \( N \to \infty \), every probability density can be represented.
  - No computation involved in the training phase
    ⇒ Simply storage of the training set

- **Problems**
  - Requires storing and computing with the entire dataset.
    ⇒ Computational cost linear in the number of data points.
    ⇒ This can be improved, at the expense of some computation during training, by constructing efficient tree-based search structures.
  - Kernel size / \( K \) in K-NN?
    - Too large: too much smoothing
    - Too small: too much noise
K-Nearest Neighbor Classification

- Bayesian Classification

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

- Here we have

\[
\begin{align*}
    p(x) &\approx \frac{K}{NV} \\
p(x \mid C_j) &\approx \frac{K_j}{N_jV} \\
p(C_j) &\approx \frac{N_j}{N}
\end{align*}
\]

\[
p(C_j \mid x) \approx \frac{K_j}{N_jV} \frac{N_j}{N} \frac{NV}{K} = \frac{K_j}{K}
\]

k-Nearest Neighbor classification
K-Nearest Neighbors for Classification

Image source: C.M. Bishop, 2006
K-Nearest Neighbors for Classification

- Results on an example data set

\[
K = 1 \\
K = 3 \\
K = 31
\]

- \(K\) acts as a smoothing parameter.

- Theoretical guarantee
  - For \(N \to \infty\), the error rate of the 1-NN classifier is never more than twice the optimal error (obtained from the true conditional class distributions).

B. Leibe

Image source: C.M. Bishop, 2006
Bias-Variance Tradeoff

- Probability density estimation
  - Histograms: bin size?
    - $\Delta$ too large: too smooth
    - $\Delta$ too small: not smooth enough
  - Kernel methods: kernel size?
    - $h$ too large: too smooth
    - $h$ too small: not smooth enough
  - K-Nearest Neighbor: $K$?
    - $K$ too large: too smooth
    - $K$ too small: not smooth enough

- This is a general problem of many probability density estimation methods
  - Including parametric methods and mixture models

Slide credit: Bernt Schiele
Discussion

- The methods discussed so far are all simple and easy to apply. They are used in many practical applications.
- However...
  - Histograms scale poorly with increasing dimensionality.
    - Only suitable for relatively low-dimensional data.
  - Both k-NN and kernel density estimation require the entire data set to be stored.
    - Too expensive if the data set is large.
  - Simple parametric models are very restricted in what forms of distributions they can represent.
    - Only suitable if the data has the same general form.
- We need density models that are efficient and flexible!
  - Next lecture...
References and Further Reading

• More information in Bishop’s book
  - Gaussian distribution and ML: Ch. 1.2.4 and 2.3.1-2.3.4.
  - Bayesian Learning: Ch. 1.2.3 and 2.3.6.
  - Nonparametric methods: Ch. 2.5.

• Additional information can be found in Duda & Hart
  - ML estimation: Ch. 3.2
  - Bayesian Learning: Ch. 3.3-3.5
  - Nonparametric methods: Ch. 4.1-4.5

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

R.O. Duda, P.E. Hart, D.G. Stork
Pattern Classification
2nd Ed., Wiley-Interscience, 2000