Density estimation

Milos Hauskrecht
milos@cs.pitt.edu
5329 Sennott Square

Announcements

Next lecture:
• Matlab tutorial

Rules for attending the class:
• Registered for credit
• Registered for audit (only if there are available seats)

Rules for audit:
• Homework assignments
Review

Design cycle

Data

Feature selection

Model selection

Learning

Evaluation

Require prior knowledge
## Data

**Data may need a lot of:**

- Cleaning
- Preprocessing (conversions)

### Cleaning:

- Get rid of errors, noise,
- Removal of redundancies

### Preprocessing:

- Renaming
- Rescaling (normalization)
- Discretizations
- Abstraction
- Aggregation
- New attributes

---

## Data biases

- **Watch out for data biases:**
  - Try to understand the data source
  - It is very easy to derive “unexpected” results when data used for analysis and learning are biased (pre-selected)

- **Results (conclusions) derived for pre-selected data do not hold in general !!!**
Data biases

Example 1: Risks in pregnancy study
- Sponsored by DARPA at military hospitals
- Study of a large sample of pregnant woman who visited military hospitals
- **Conclusion:** the factor with the largest impact on reducing risks during pregnancy (statistically significant) is a pregnant woman being single
- Single woman → the smallest risk
- What is wrong?

---

Data

Example 2: Stock market trading (example by Andrew Lo)
- Data on stock performances of companies traded on stock market over past 25 year
- **Investment goal:** pick a stock to hold long term
- **Proposed strategy:** invest in a company stock with an IPO corresponding to a Carmichael number
- **Evaluation result:** excellent return over 25 years
- Where the magic comes from?
Feature selection

- **The size (dimensionality) of a sample** can be enormous
  \[ x_i = (x_i^1, x_i^2, \ldots, x_i^d) \quad d \quad - \text{very large} \]
- **Example: document classification**
  - 10,000 different words
  - Inputs: counts of occurrences of different words
  - Too many parameters to learn (not enough samples to justify the estimates the parameters of the model)
- **Dimensionality reduction: replace inputs with features**
  - **Extract relevant inputs** (e.g. mutual information measure)
  - **PCA** – principal component analysis
  - **Group (cluster) similar words** (uses a similarity measure)
  - Replace with the group label
Model selection

- **What is the right model to learn?**
  - E.g what polynomial to use
  - A prior knowledge helps a lot, but still a lot of guessing
  - **Initial data analysis and visualization**
    - We can make a good guess about the form of the distribution, shape of the function

- **Overfitting problem**
  - Take into account the **bias and variance** of error estimates
  - Simpler (more biased) model – parameters can be estimated more reliably (smaller variance of estimates)
  - Complex model with many parameters – parameter estimates are less reliable (large variance of the estimate)
**Solutions for overfitting**

**How to make the learner avoid the overfit?**

- **Assure sufficient number of samples** in the training set
  - May not be possible (small number of examples)
- **Hold some data out of the training set = validation set**
  - Train (fit) on the training set (w/o data held out);
  - Check for the generalization error on the validation set, choose the model based on the validation set error
    (random resampling validation techniques)
- **Regularization (Occam’s Razor)**
  - Penalize for the model complexity (number of parameters)
  - Explicit preference towards simple models

---

**Design cycle**

- **Data**
- **Feature selection**
- **Model selection**
- **Learning**
- **Evaluation**

*Require prior knowledge*
Learning

- **Learning = optimization problem.** Various criteria:
  - **Mean square error**
    \[
    w^* = \text{arg min}_w \text{Error}(w) \quad \text{Error}(w) = \frac{1}{N} \sum_{i=1}^{N} (y_i - f(x_i, w))^2
    \]
  - **Maximum likelihood (ML) criterion**
    \[
    \Theta^* = \text{arg max}_\Theta P(D | \Theta) \quad \text{Error}(\Theta) = -\log P(D | \Theta)
    \]
  - **Maximum posterior probability (MAP)**
    \[
    \Theta^* = \text{arg max}_\Theta P(\Theta | D) \quad P(\Theta | D) = \frac{P(D | \Theta)P(\Theta)}{P(D)}
    \]

Learning

**Learning = optimization problem**

- Optimization problems can be hard to solve. Right choice of a model and an error function makes a difference.

- **Parameter optimizations**
  - Gradient descent, Conjugate gradient (1st order method)
  - Newton-Rhapson (2nd order method)
  - Levenberg-Marquard

  Some can be carried on-line on a sample by sample basis

- **Combinatorial optimizations (over discrete spaces):**
  - Hill-climbing
  - Simulated-annealing
  - Genetic algorithms
Evaluation.

- **Simple holdout method.**
  - Divide the data to the training and test data.
- **Other more complex methods**
  - Based on random re-sampling validation schemes:
    - cross-validation, random sub-sampling.
  - What if we want to compare the predictive performance on a classification or a regression problem for two different learning methods?
- **Solution:** compare the error results on the test data set
  - The method with better (smaller) testing error gives a better generalization error.
  - But we need statistics to show significance
Density estimation

Outline:

- **Density estimation:**
  - Maximum likelihood (ML)
  - Bayesian parameter estimates
  - MAP
- Bernoulli distribution.
- Binomial distribution
- Multinomial distribution
- Normal distribution
Density estimation

**Data:** \( D = \{D_1, D_2, \ldots, D_n\} \)

\( D_i = x_i \)  a vector of attribute values

**Attributes:**

- modeled by random variables \( \mathbf{X} = \{X_1, X_2, \ldots, X_d\} \) with:
  - Continuous values
  - Discrete values

  E.g. *blood pressure* with numerical values
  or *chest pain* with discrete values
    
    [no-pain, mild, moderate, strong]

**Underlying true probability distribution:**

\[ p(\mathbf{X}) \]

---

**Density estimation**

**Data:** \( D = \{D_1, D_2, \ldots, D_n\} \)

\( D_i = x_i \)  a vector of attribute values

**Objective:** try to estimate the underlying ‘true’ probability distribution over variables \( \mathbf{X} \), \( p(\mathbf{X}) \), using examples in \( D \)

![Diagram](image.png)

**Standard (iid) assumptions:** Samples

- are independent of each other
- come from the same (identical) distribution (fixed \( p(\mathbf{X}) \))

---

CS 2750 Machine Learning
Density estimation

Types of density estimation:

**Parametric**
- the distribution is modeled using a set of parameters \( \Theta \)
  
  \[ p(\mathbf{X} | \Theta) \]
- **Example**: mean and covariances of a multivariate normal
- **Estimation**: find parameters \( \Theta \) describing data \( D \)

**Non-parametric**
- The model of the distribution utilizes all examples in \( D \)
- As if all examples were parameters of the distribution
- **Examples**: Nearest-neighbor

**Semi-parametric**

Learning via parameter estimation

In this lecture we consider **parametric density estimation**

**Basic settings**:
- A set of random variables \( \mathbf{X} = \{X_1, X_2, \ldots, X_d\} \)
- **A model of the distribution** over variables in \( \mathbf{X} \)
  with parameters \( \Theta \) : \( \hat{p}(\mathbf{X} | \Theta) \)

- **Data** \( D = \{D_1, D_2, \ldots, D_n\} \)

**Objective**: find parameters \( \Theta \) such that \( p(\mathbf{X} | \Theta) \) describes data \( D \) the best
Parameter estimation.

• **Maximum likelihood (ML)**
  
  maximize $ p(D \mid \Theta, \xi) $  
  
  yields: one set of parameters $ \Theta_{ML} $  
  
  the target distribution is approximated as:  
  
  $$ \hat{p}(X) = p(X \mid \Theta_{ML}) $$

• **Bayesian parameter estimation**
  
  uses the posterior distribution over possible parameters  
  
  $$ p(\Theta \mid D, \xi) = \frac{p(D \mid \Theta, \xi) p(\Theta \mid \xi)}{p(D \mid \xi)} $$
  
  Yields: all possible settings of $ \Theta $ (and their “weights”)  
  
  The target distribution is approximated as:  
  
  $$ \hat{p}(X) = p(X \mid D) = \int p(X \mid \Theta) p(\Theta \mid D, \xi) d\Theta $$

Other possible criteria:

• **Maximum a posteriori probability (MAP)**
  
  maximize $ p(\Theta \mid D, \xi) $ (mode of the posterior)  
  
  Yields: one set of parameters $ \Theta_{MAP} $  
  
  Approximation:  
  
  $$ \hat{p}(X) = p(X \mid \Theta_{MAP}) $$

• **Expected value of the parameter**
  
  $ \hat{\Theta} = E(\Theta) $ (mean of the posterior)  
  
  Expectation taken with regard to posterior $ p(\Theta \mid D, \xi) $  
  
  Yields: one set of parameters  
  
  Approximation:  
  
  $$ \hat{p}(X) = p(X \mid \hat{\Theta}) $$
Parameter estimation. Coin example.

**Coin example:** we have a coin that can be biased

**Outcomes:** two possible values -- head or tail

**Data:** \( D \) a sequence of outcomes \( x_i \) such that

- **head** \( x_i = 1 \)
- **tail** \( x_i = 0 \)

**Model:** probability of a head \( \theta \)
probability of a tail \( (1 - \theta) \)

**Objective:**
We would like to estimate the probability of a head \( \hat{\theta} \)
from data

Parameter estimation. Example.

- Assume the unknown and possibly biased coin
- Probability of the head is \( \theta \)
- **Data:**
  
  \[
  \begin{array}{cccccccccccccccc}
  H & H & T & T & H & H & T & H & T & T & T & H & T & H & T & H & H & H & H & H & H & T \\
  \end{array}
  \]
  - **Heads:** 15
  - **Tails:** 10

What would be your estimate of the probability of a head?

\( \hat{\theta} = ? \)
Parameter estimation. Example

• **Assume** the unknown and possibly biased coin
• Probability of the head is \( \theta \)

**Data:**

\[
\begin{align*}
\text{H H T T H H T H T T H T H T H H H T H H H T T H}
\end{align*}
\]

– **Heads:** 15
– **Tails:** 10

What would be your choice of the probability of a head?

**Solution:** use frequencies of occurrences to do the estimate

\[
\hat{\theta} = \frac{15}{25} = 0.6
\]

This is the **maximum likelihood estimate** of the parameter \( \theta \)

Probability of an outcome

**Data:** \( D \) a sequence of outcomes \( x_i \) such that

• **head** \( x_i = 1 \)
• **tail** \( x_i = 0 \)

**Model:** probability of a head \( \theta \) probability of a tail \( (1 - \theta) \)

**Assume:** we know the probability \( \theta \)

**Probability of an outcome of a coin flip** \( x_i \)

\[
P(x_i \mid \theta) = \theta^{x_i} (1 - \theta)^{1-x_i} \quad \text{Bernoulli distribution}
\]

– Combines the probability of a head and a tail
– So that \( x_i \) is going to pick its correct probability
– Gives \( \theta \) for \( x_i = 1 \)
– Gives \( (1 - \theta) \) for \( x_i = 0 \)
Probability of a sequence of outcomes.

Data: \( D \) a sequence of outcomes \( x_i \) such that

- head \( x_i = 1 \)
- tail \( x_i = 0 \)

Model: probability of a head \( \theta \)
probability of a tail \( 1 - \theta \)

Assume: a sequence of independent coin flips

\( D = H \ H \ T \ H \ T \ H \) (encoded as \( D = 110101 \))

What is the probability of observing the data sequence \( D \):

\[
P(D | \theta) = ?
\]
Probability of a sequence of outcomes.

Data: \( D \) a sequence of outcomes \( x_i \) such that
- head \( x_i = 1 \)
- tail \( x_i = 0 \)

Model: probability of a head \( \theta \)
probability of a tail \( 1 - \theta \)

Assume: a sequence of coin flips \( D = H H T H T H \)
encoded as \( D = 110101 \)

What is the probability of observing a data sequence \( D \):\n
\[
P(D \mid \theta) = \theta \theta (1 - \theta) \theta (1 - \theta) \theta
\]

likelihood of the data

Can be rewritten using the Bernoulli distribution:
The goodness of fit to the data.

Learning: we do not know the value of the parameter $\theta$

Our learning goal:
- Find the parameter $\theta$ that fits the data $D$ the best?

One solution to the “best”: Maximize the likelihood

$$P(D \mid \theta) = \prod_{i=1}^{n} \theta^{x_{i}} (1 - \theta)^{(1-x_{i})}$$

Intuition:
- more likely are the data given the model, the better is the fit

Note: Instead of an error function that measures how bad the data fit the model we have a measure that tells us how well the data fit:

$$\text{Error} (D, \theta) = -P(D \mid \theta)$$

Example: Bernoulli distribution.

Coin example: we have a coin that can be biased

Outcomes: two possible values -- head or tail

Data: $D$ a sequence of outcomes $x_{i}$ such that
- head $x_{i} = 1$
- tail $x_{i} = 0$

Model: probability of a head $\theta$
- probability of a tail $(1 - \theta)$

Objective:
- We would like to estimate the probability of a head $\hat{\theta}$

Probability of an outcome $x_{i}$

$$P(x_{i} \mid \theta) = \theta^{x_{i}} (1 - \theta)^{(1-x_{i})} \quad \text{Bernoulli distribution}$$
Maximum likelihood (ML) estimate.

Likelihood of data:
\[ P(D \mid \theta, \xi) = \prod_{i=1}^{n} \theta^{x_i} (1 - \theta)^{(1-x_i)} \]

Maximum likelihood estimate
\[ \theta_{ML} = \arg \max_{\theta} P(D \mid \theta, \xi) \]

Optimize log-likelihood (the same as maximizing likelihood)
\[ l(D, \theta) = \log P(D \mid \theta, \xi) = \log \prod_{i=1}^{n} \theta^{x_i} (1 - \theta)^{(1-x_i)} = \sum_{i=1}^{n} x_i \log \theta + (1-x_i) \log(1-\theta) = \log \theta \sum_{i=1}^{n} x_i + \log(1-\theta) \sum_{i=1}^{n} (1-x_i) \]

\[ N_1 \text{ - number of heads seen} \quad N_2 \text{ - number of tails seen} \]

Maximum likelihood (ML) estimate.

Optimize log-likelihood
\[ l(D, \theta) = N_1 \log \theta + N_2 \log(1-\theta) \]

Set derivative to zero
\[ \frac{\partial l(D, \theta)}{\partial \theta} = \frac{N_1}{\theta} - \frac{N_2}{(1-\theta)} = 0 \]

Solving
\[ \theta = \frac{N_1}{N_1 + N_2} \]

ML Solution:
\[ \theta_{ML} = \frac{N_1}{N} = \frac{N_1}{N_1 + N_2} \]
Maximum likelihood estimate. Example

• **Assume** the unknown and possibly biased coin
• Probability of the head is $\theta$

• **Data:**
  
  H H T T H H T H T T T H T H T H T H H H T H H H H T
  
  – **Heads:** 15
  – **Tails:** 10

What is the ML estimate of the probability of a head and a tail?

\[
\text{Head: } \theta_{ML} = \frac{N_1}{N} = \frac{N_1}{N_1 + N_2} = \frac{15}{25} = 0.6
\]

\[
\text{Tail: } (1 - \theta_{ML}) = \frac{N_2}{N} = \frac{N_2}{N_1 + N_2} = \frac{10}{25} = 0.4
\]