TECHNISCHE
UNIVERSITÄT
DARMSTADT

Statistical Machine Learning

Lecture 3: Probability Density Estimation

Simone Schaub-Meyer & Marcus Rohrbach
Department of Computer Science
TU Darmstadt

Summer Term 2025

Today’s Objectives

- Understanding on how to estimate $p(x)$
- Covering topics
  - Density Estimation
  - Maximum Likelihood Estimation
  - Non-Parametric Models
  - Mixture Models

Outline

1. Probability Density Estimation

2. Parametric Density Models
   Maximum Likelihood Method

3. Non-Parametric Models
   Histograms
   Kernel Density Estimation
   K-nearest Neighbors

4. Mixture models

5. Wrap-Up

Outline

1. Probability Density Estimation

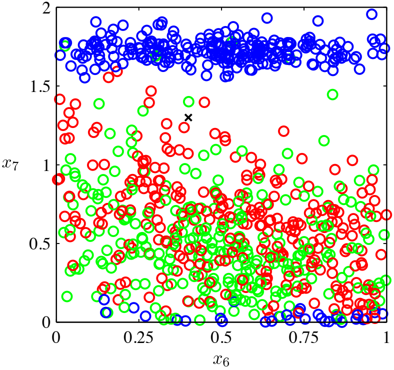
2. Parametric Density Models
- Maximum Likelihood Method

3. Non-Parametric Models
- Histograms
- Kernel Density Estimation
- K-nearest Neighbors

4. Mixture models

5. Wrap-Up

Training Data



- How do we get the probability distributions from this so that we can classify with them?

Probability Density Estimation

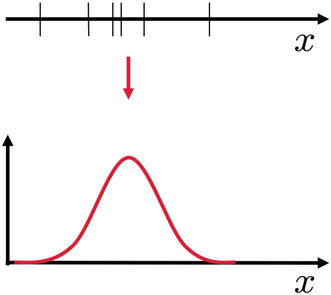
- So far we have seen:
  - **Optimal Bayes Classification**, based on probability distributions $p(x|C_k)p(C_k)$

- The prior $p(C_k)$ is easy to deal with. We can "just count" the number of occurrences of each class in the training data

- We need to estimate/learn the **class-conditional probability density** $p(x|C_k)$
  - **Supervised training**: we know the input data points and their true labels (classes)
  - Estimate the density separately for each class C_k

Probability Density Estimation

- **Remember:** The relationship between the outcomes of a random variable x and its probability $Pr(X = x)$ is referred to as the probability density, or simply the "density."



- Training data

$$x_1, x_2, x_3, \ldots$$

- Estimation $p(x)$

Types of Probability Density Estimation models

- **Parametric probability density estimation** involves selecting a common distribution and estimating the distribution parameters from data samples.
- **Non-parametric probability density estimation** involves fitting a model to the arbitrary distribution of the data, e.g., kernel density estimation – every known data point in the dataset is used as a parameter.
- **Mixture density models** are flexible models that combine parametric and non-parametric estimations.

Outline

1. Probability Density Estimation

2. Parametric Density Models
   - Maximum Likelihood Method

3. Non-Parametric Models
   - Histograms
   - Kernel Density Estimation
   - K-nearest Neighbors

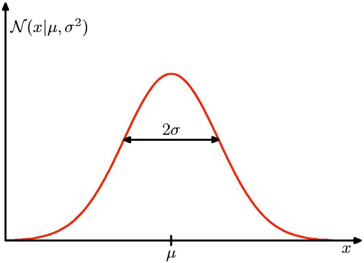
4. Mixture models

5. Wrap-Up

Parametric Density Models

- Simple case: **Gaussian Distribution**

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



- Is governed by two parameters: mean and variance. If we know these parameters, we can fully describe $p(x)$

Parametric Density Models

- Notation for **parametric density models**

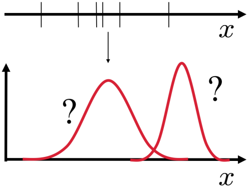
$$x \sim p(x|\theta)$$

- For the Gaussian distribution

$$\begin{aligned}\theta &= (\mu, \sigma)\\ x &\sim p(x|\mu, \sigma)\end{aligned}$$

Parametric Density Models

- **Learning** means to estimate the parameters θ given the training data $\mathcal{D} = \{x_1, x_2, \ldots\}$



- **Likelihood** of θ is defined as the probability that the data $\mathcal{D}$ was generated from the probability density function with parameters θ

$$L(\theta) = p(\mathcal{D}|\theta)$$

Maximum Likelihood Method

- Consider a set of points $\mathcal{D} = \{x_1, \dots, x_N\}$, we are interested in the likelihood of all data $p(\mathcal{D}|\theta)$.
- **Assumption**: the **data is i.i.d.**(independent and identically distributed)
    - The random variables x_1 and x_2 are independent if

$$P\left(x_1 \leq \alpha, x_2 \leq \beta\right) = P\left(x_1 \leq \alpha\right) P\left(x_2 \leq \beta\right) \quad \forall \alpha, \beta \in \mathbb{R}$$

    - The random variables x_1 and x_2 are identically distributed if

$$P\left(x_1 \leq \alpha\right) = P\left(x_2 \leq \alpha\right) \quad \forall \alpha \in \mathbb{R}$$

Maximum Likelihood Method

- Likelihood

$$L(\theta) = p(\mathcal{D}|\theta) = p(x_1, \ldots, x_N|\theta)$$

(using the i.i.d. assumption)

$$= p(x_1|\theta) \cdot \ldots \cdot p(x_n|\theta)$$

$$= \prod_{n=1}^{N} p(x_n|\theta)$$

- **Maximum Likelihood Estimation**: $\hat{\theta}_{\mathrm{ML}} = \arg\max_\theta p(\mathcal{D}|\theta)$ seeks for the parameter $\hat{\theta}_{\mathrm{ML}}$, which best explains the data $\mathcal{D}$
- $\hat{\theta}_{\mathrm{ML}}$ is a **random variable** – it is an estimate based on the available dataset. Consequently, we are interested in its **mean value** (the **most probable value**) and its **variance**.

Maximum log-Likelihood Method

- It is more convenient and numerical stable to maximize the log-likelihood w.r.t. θ

$$LL(\theta) = \log L(\theta) = \log p(\mathcal{D}|\theta) = \log \prod_{n=1}^{N} p(x_n|\theta) = \sum_{n=1}^{N} \log p(x_n|\theta)$$

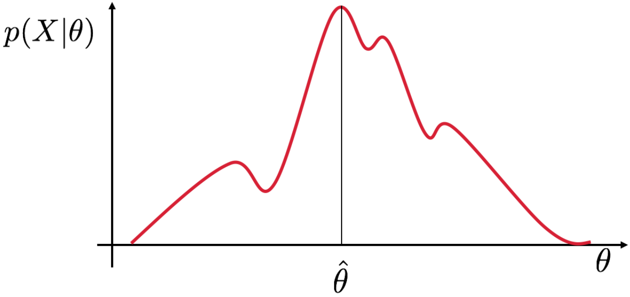
- Because the logarithm is monotonically increasing, it holds

$$\hat{\theta}_{\mathrm{ML}} = \arg\max_{\theta} \sum_{n=1}^{N} \log p(x_n|\theta) = \arg\max_{\theta} LL(\theta)$$

- Maximizing a sum of terms is always easier than maximizing a product; cf., the difficulty of expressing the derivative of a long product of terms.

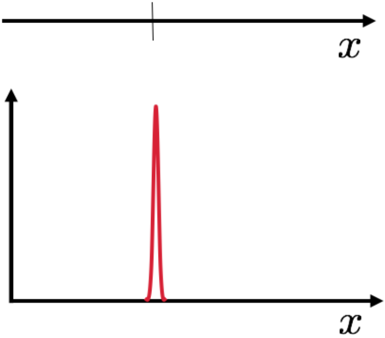
Likelihood Estimation

$$L(\theta) = p(\mathcal{D}|\theta) = \prod_{n=1}^{N} p(x_n|\theta)$$



Likelihood Estimation: Degenerate case

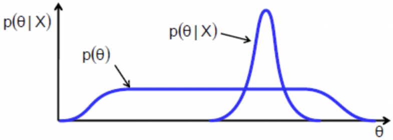
- If we try estimate a single data point $N = 1$, $\mathcal{D} = \{x_1\}$, the resulting Gaussian “looks like”



- More formally speaking, the probability density of the Gaussian becomes a Dirac δ.

Bayesian Estimation

- The MLE provides point estimates about the model parameters.
- The Bayesian treatment represents our uncertainty about the parameters θ using a prior over θ and updating our posterior estimation via Bayes rule!



- Bayesian estimation/learning of parametric distributions assumes that the **parameters are random variables** too.
- This allows us to use **prior knowledge** about the parameters

Bayesian Estimation

- Formalize this as a **conditional probability** $p\left(x|\mathcal{D}\right)$:

$$p\left(x|\mathcal{D}\right) = \int p\left(x, \theta|\mathcal{D}\right) \mathrm{d}\theta$$
$$p\left(x, \theta|\mathcal{D}\right) = p\left(x|\theta, \mathcal{D}\right) p\left(\theta|\mathcal{D}\right)$$

- The parametric density $p(x \mid \theta)$ can be fully determined with the parameters θ, i.e., θ is a **sufficient statistic**. Hence, we have $p\left(x|\theta, \mathcal{D}\right) = p\left(x|\theta\right)$

$$p\left(x|\mathcal{D}\right) = \int p\left(x|\theta\right) p\left(\theta|\mathcal{D}\right) \mathrm{d}\theta$$

Bayesian Estimation

- **How to evaluate:**

$$p\left(x|\mathcal{D}\right) = \int p\left(x|\theta\right) p\left(\theta|\mathcal{D}\right) \mathrm{d}\theta$$

- **Parameter Likelihood**

$$p\left(\theta|\mathcal{D}\right) = \frac{p\left(\mathcal{D}|\theta\right) p\left(\theta\right)}{p\left(\mathcal{D}\right)} = L\left(\theta\right) \frac{p\left(\theta\right)}{p\left(\mathcal{D}\right)}$$

- **Evidence**

$$p\left(\mathcal{D}\right) = \int p\left(\mathcal{D}|\theta\right) p\left(\theta\right) \mathrm{d}\theta = \int L\left(\theta\right) p\left(\theta\right) \mathrm{d}\theta$$

Bayesian Estimation

$$p\left(x|\mathcal{D}\right) = \int p\left(x|\theta\right) p\left(\theta|\mathcal{D}\right) \mathrm{d}\theta = \frac{1}{p\left(\mathcal{D}\right)} \int p\left(x|\theta\right) L\left(\theta\right) p\left(\theta\right) \mathrm{d}\theta$$

- The probability $p\left(\theta|\mathcal{D}\right)$ makes it explicit how the parameter estimation depends on the training data.
- If $p\left(\theta|\mathcal{D}\right)$ is small in most places, but large for a single $\hat{\theta}$ then we can approximate
  $$p\left(x|\mathcal{D}\right) \approx p(x|\hat{\theta})$$
  - Sometimes referred to as the **Bayes point**.
- The more uncertain we are about estimating $\hat{\theta}$, the more the density is averaged across multiple θ

Bayesian Estimation

- **Problem**: In general, it is intractable to integrate out the parameters θ (or only possible to do so numerically).
- Example with closed form solution:
  - For Gaussian data distribution, the variance is known and fixed
  - We estimate the distribution of the mean

$$p\left(\mu|\mathcal{D}\right) = \frac{p\left(\mathcal{D}|\mu\right) p\left(\mu\right)}{p\left(\mathcal{D}\right)}$$

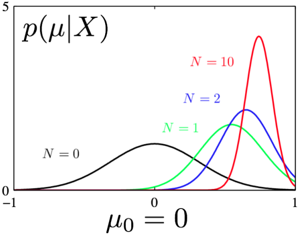
  - With prior

$$p\left(\mu\right) = \mathcal{N}\left(\mu_0, \sigma_0^2\right)$$

Bayesian Estimation

- **Sample mean**

$$\bar{x} = \frac{1}{N}\sum_{n=1}^{N} x_n$$



- **Bayesian estimation**

$$p\left(\mu|\mathcal{D}\right) \sim \mathcal{N}\left(\mu_N, \sigma_N^2\right)$$

$$\mu_N = \frac{N\sigma_0^2\bar{x} + \sigma^2\mu_0}{N\sigma_0^2 + \sigma^2}, \quad \frac{1}{\sigma_N^2} = \frac{N}{\sigma^2} + \frac{1}{\sigma_0^2}$$

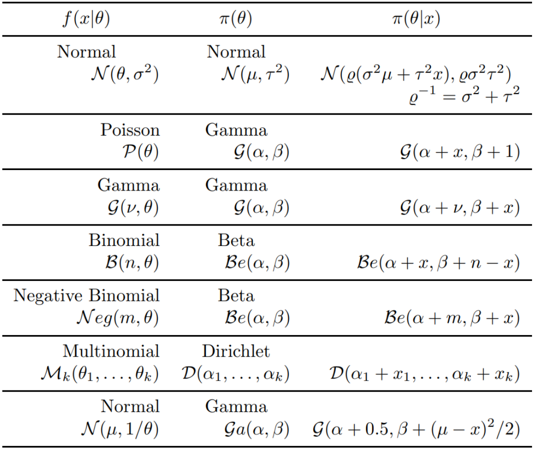
- Check what happens when *N* grows to infinity...

Conjugate Priors

- **Conjugate Priors** are prior distributions for the parameters that do not “change” the type of the parametric model
- This means that both the prior and the posterior lie in the same distribution family
- For example, as we saw that a Gaussian prior on the mean is conjugate to the Gaussian model. This works here because...
  - The product of two Gaussians is a Gaussian.
  - The marginal of a Gaussian is a Gaussian.
- Generally, it is not as easy!

Conjugate Priors

Table 3.3.1. *Natural conjugate priors for some common exponential families*



$f(x\|\theta)$	$\pi(\theta)$	$\pi(\theta\|x)$
Normal $\mathcal{N}(\theta,\sigma^2)$	Normal $\mathcal{N}(\mu,\tau^2)$	$\mathcal{N}(\varrho(\sigma^2\mu+\tau^2x),\varrho\sigma^2\tau^2)$ $\varrho^{-1}=\sigma^2+\tau^2$
Poisson $\mathcal{P}(\theta)$	Gamma $\mathcal{G}(\alpha,\beta)$	$\mathcal{G}(\alpha+x,\beta+1)$
Gamma $\mathcal{G}(\nu,\theta)$	Gamma $\mathcal{G}(\alpha,\beta)$	$\mathcal{G}(\alpha+\nu,\beta+x)$
Binomial $\mathcal{B}(n,\theta)$	Beta $\mathcal{B}e(\alpha,\beta)$	$\mathcal{B}e(\alpha+x,\beta+n-x)$
Negative Binomial $\mathcal{N}eg(m,\theta)$	Beta $\mathcal{B}e(\alpha,\beta)$	$\mathcal{B}e(\alpha+m,\beta+x)$
Multinomial $\mathcal{M}_k(\theta_1,\ldots,\theta_k)$	Dirichlet $\mathcal{D}(\alpha_1,\ldots,\alpha_k)$	$\mathcal{D}(\alpha_1+x_1,\ldots,\alpha_k+x_k)$
Normal $\mathcal{N}(\mu,1/\theta)$	Gamma $\mathcal{G}a(\alpha,\beta)$	$\mathcal{G}(\alpha+0.5,\beta+(\mu-x)^2/2)$

Robert, C. P. (2007). The Bayesian choice: from decision-theoretic foundations to computational implementation (Vol. 2). New York: Springer.

Outline

1. Probability Density Estimation

2. Parametric Density Models
   - Maximum Likelihood Method

3. **Non-Parametric Models**
   - Histograms
   - Kernel Density Estimation
   - K-nearest Neighbors

4. Mixture models

5. Wrap-Up

Non-Parametric Models

Why use **non-parametric representations**?

Non-Parametric Models

- Often, we do not know what functional form the class-conditional density takes (or we do not know what function family we need)
- Probability density is estimated directly from the data (i.e. without an explicit parametric model):
  - **Histograms**
  - **Kernel density estimation** (Parzen windows)
  - **K-nearest neighbors**
- Every data point is a parameter, so non-parametric models have an uncertain and possibly infinite number of parameters

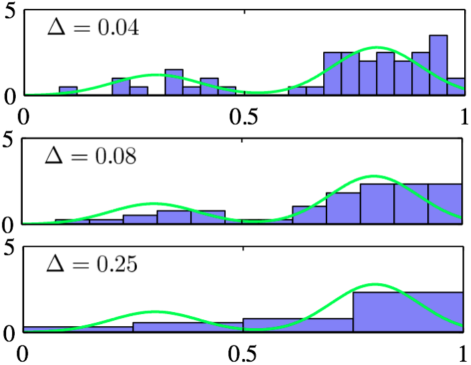
Histograms

- Discretize the feature space into bins:

Not smooth enough

About right

Too smooth



Histograms: Properties

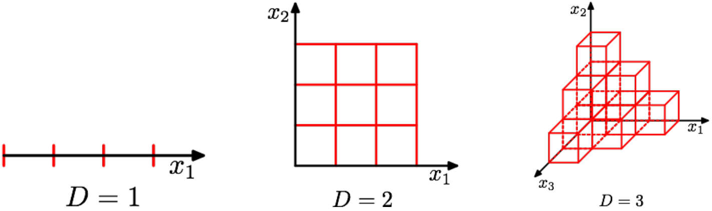
- In the infinitesimal bin-width limit, any probability density can be approximated arbitrarily well.
- Assuming some locality metric, e.g., Euclidean distance

Histograms: Problems

- High-dimensional feature spaces: Exponential increase in the number of bins. Hence it requires exponentially much data. This is commonly known as the **Curse of dimensionality**.
- The estimated histogram density has discontinuities due to the bin edges
- How to choose the size of the bins?
    - The bin width controls the **smoothness** – the value of the smoothing parameter should not be too large or too small to obtain good results

Curse of Dimensionality

- For histograms



- We will see that this is a general issue that we have to keep in mind

Kernel Density Estimators

- Data point $\mathbf{x}$ is sampled from probability density $p(\mathbf{x})$.
- Probability that $\mathbf{x}$ falls in region R

$$P(\mathbf{x} \in R) = \int_R p(\mathbf{x})\, \mathrm{d}\mathbf{x}$$

- If R is sufficiently small, with volume V, then $p(\mathbf{x})$ is almost constant

$$P(\mathbf{x} \in R) = \int_R p(\mathbf{x})\, \mathrm{d}\mathbf{x} \approx p(\mathbf{x})\, V$$

- If R is sufficiently large

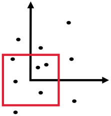
$$P(\mathbf{x} \in R) = \frac{K}{N} \implies p(\mathbf{x}) \approx \frac{K}{NV}$$

  where N is the number of total points and K is the number of points falling in the region R

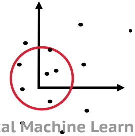
Kernel Density Estimators

$$p(\mathbf{x}) \approx \frac{K}{NV}$$

- **Kernel density estimation** (KDE) – Fix V and determine K
  - Example: determine the number of data points K in a fixed hypercube



- **K-nearest neighbors** (kNN) – Fix K and determine V
  - Example: increase the size of a sphere until K data points fall into the sphere



Parzen Window

- Approximating the density

$$p(\mathbf{x}) \approx \frac{K}{NV} = \frac{1}{Nh^d} \sum_{n=1}^{N} k(\mathbf{x} - \mathbf{x}_n)$$

$$K = \sum_{n=1}^{N} k(\mathbf{x} - \mathbf{x}_n), \quad V = \int k(\mathbf{u})\,\mathrm{d}\mathbf{u} = h^d$$

$$k(\mathbf{u}) = \begin{cases} 1 & |u_j| \leq \frac{h}{2},\ j = 1, \ldots, d \\ 0 & \text{otherwise} \end{cases}$$

- Hypercubes in d dimensions with edge length h

Gaussian Kernel

- Approximating the density

$$p(\mathbf{x}) \approx \frac{K}{NV} = \frac{1}{N\left(\sqrt{2\pi h^2}\right)^d} \sum_{n=1}^{N} \exp\left\{ -\frac{\left\|\mathbf{x} - \mathbf{x}_n\right\|^2}{2h^2} \right\}$$

$$K = \sum_{n=1}^{N} k(\mathbf{x} - \mathbf{x}_n), \quad V = \int k(\mathbf{u})\,\mathrm{d}\mathbf{u} = 1$$

$$k(\mathbf{u}) = \frac{1}{\left(\sqrt{2\pi h^2}\right)^d} \exp\left\{ -\frac{\left\|\mathbf{u}\right\|^2}{2h^2} \right\}$$

General Formulation – Arbitrary Kernel

$$p(\mathbf{x}) \approx \frac{K}{NV} = \frac{1}{Nh^d} \sum_{n=1}^{N} k\left(\frac{\|\mathbf{x} - \mathbf{x}_n\|}{h}\right)$$

$$K = \sum_{n=1}^{N} k\left(\frac{\|\mathbf{x} - \mathbf{x}_n\|}{h}\right), \quad V = h^d$$

$$k(\mathbf{u}) \geq 0, \quad \int k(\mathbf{u})\, \mathrm{d}\mathbf{u} = 1$$

Common Kernels ($d = 1$)

- **Gaussian Kernel**

$$k(u) = \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{1}{2}u^2\right\}$$

- **Parzen window**

$$k(u) = \begin{cases} 1 & |u| \leq 1/2 \\ 0 & \text{otherwise} \end{cases}$$

  - Not very smooth results

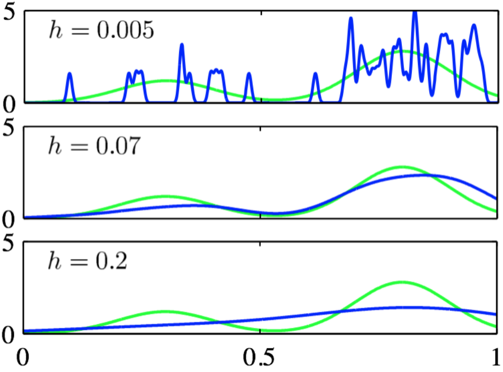
- Problem with kernel methods: We have to select the kernel bandwidth h appropriately, as it controls smoothness

Gaussian KDE Example

Not smooth enough

About right

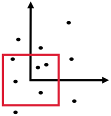
Too smooth



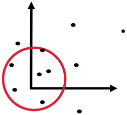
Back to our definition

$$p(\mathbf{x}) \approx \frac{K}{NV}$$

- **Kernel density estimation** (KDE) – Fix V and determine K
  - Example: determine the number of data points K in a fixed hypercube



- **K-nearest neighbors** (kNN) – Fix K and determine V
  - Example: increase the size of a sphere until K data points fall into the sphere

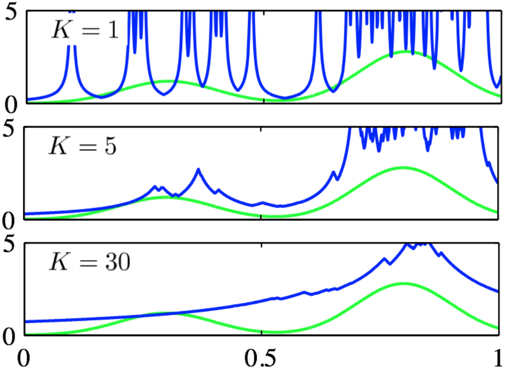


K-Nearest Neighbors (kNN)

Not smooth enough

About right

Too smooth



- Note: Blue rescaled for visualization
- The model of the KNN is not a true density model because the integral over all spaces diverges

K-Nearest Neighbors (kNN)

- Bayesian classification

$$P(C_j|x) = \frac{P(x|C_j)\,P(C_j)}{P(x)}$$

- k-Nearest Neighbors classification
    - Assume we have a dataset of N points, where N_j is the number of data points in class C_j and $\sum_j N_j = N$. To classify a point x we draw a sphere centered in x that contains K points (from any classes). Assume the sphere has volume V and contains K_j points of class C_j

$$P(x) \approx \frac{K}{NV}, \quad P(x|C_j) \approx \frac{K_j}{N_j V}, \quad P(C_j) \approx \frac{N_j}{N}$$

$$P(C_j|x) \approx \frac{K_j}{N_j V}\frac{N_j}{N}\frac{NV}{K} = \frac{K_j}{K}$$

K-Nearest Neighbors (kNN): Advantage

- **The main advantage** - a very simple approximation of the (optimal) Bayes classifier!
  - If we wish to minimize the misclassification rate, we assign a point x to the class with the largest posterior probability (largest K_j/K).

Bias-Variance Problem

- Non-parametric probability density estimation:
  - **Histograms**: Size of the bins?
    - Bin too large: too smooth
    - Bin too small: not smooth enough
  - **Kernel density estimation**: Kernel bandwidth?
    - h too large: too smooth
    - h too small: not smooth enough
  - **K-nearest neighbor**: Number of neighbors?
    - K too large: too smooth
    - K too small: not smooth enough
- A general problem of many density estimation approaches, including parametric and mixture models.

Outline

1. **Probability Density Estimation**

2. **Parametric Density Models**
   Maximum Likelihood Method

3. **Non-Parametric Models**
   Histograms
   Kernel Density Estimation
   K-nearest Neighbors

4. **Mixture models**

5. **Wrap-Up**

4. Mixture models

Parametric models

- Gaussian, (simple) Neural Networks, ...
- Good analytic properties
- Simple
- Small memory requirements
- Fast

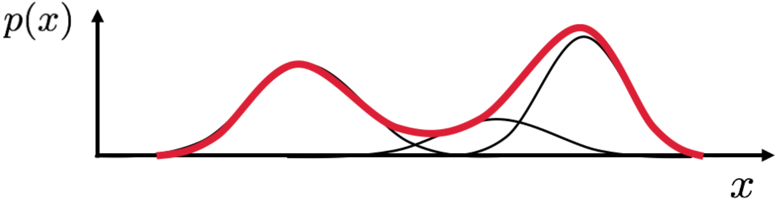
Nonparametric models

- Kernel Density Estimation, k-Nearest Neighbors, ...
- General
- Large memory requirements
- Slow

Mixture models are a mix of parametric and nonparametric models.

Mixture of Gaussians

- Sum of individual Gaussian distributions



- In the limit (i.e. with many mixture components) this can approximate every (smooth) density

$$p\left(x\right) = \sum_{j=1}^{M} p\left(x|z_j\right) p\left(z_j\right)$$

Mixture of Gaussians

- Let z be random variable representing the discrete set of mixtures $\{1, .., M\}$, with $z_j = 1$ for the j-th component and 0 elsewhere
- The marginal distribution of x

$$p(x) = \sum_{j=1}^{M} p(x|z_j)\, p(z_j)$$

- where the conditional distribution of x given a component z_j is

$$p(x|z_j) = \mathcal{N}(x|\mu_j, \sigma_j) = \frac{1}{\sqrt{2\pi\sigma_j^2}} \exp\left\{ -\frac{(x-\mu_j)^2}{2\sigma_j^2} \right\}$$

Mixture of Gaussians

- The prior over z is specified w.r.t. the mixing coefficients π_j that are also probabilities!

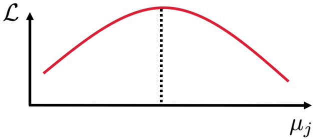
$$p\left(z_j\right) = \pi_j \quad \text{with} \quad 0 \leq \pi_j \leq 1, \sum_{j=1}^{M} \pi_j = 1$$

- **Remarks**
  - The mixture density integrates to 1: $\int p\left(x\right) \mathrm{d}x = 1$
  - The mixture parameters are: $\theta = \{\mu_1, \sigma_1, \pi_1, \ldots, \mu_M, \sigma_M, \pi_M\}$

Mixture of Gaussians – MLE

- Maximum (log-)Likelihood Estimation (for the means μ_j)
    - Dataset with N i.i.d. points $\{x_1, \ldots, x_N\}$

$$\mathcal{L} = \log L(\theta) = \sum_{n=1}^{N} \log p(x_n|\theta)$$



$$\frac{\partial \mathcal{L}}{\partial \mu_j} = 0$$

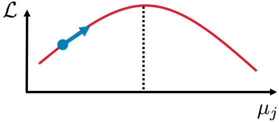
$$\hat{\mu}_j = \frac{\sum_{n=1}^{N} p(z_j|x_n)\, x_n}{\sum_{n=1}^{N} p(z_j|x_n)}$$

- **What is the problem with this approach?**
- Circular dependency – No analytical solution!

Mixture of Gaussians – MLE Gradient Ascent

- Maximum (log-)Likelihood Estimation
  - Dataset with N i.i.d. points $\{x_1, \dots, x_N\}$

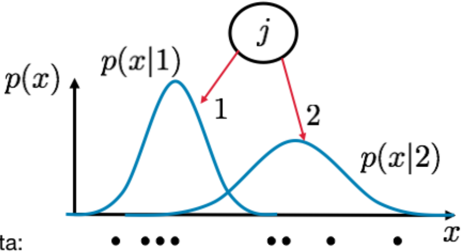
$$\mathcal{L} = \log L(\theta) = \sum_{n=1}^{N} \log p(x_n|\theta)$$



$$\frac{\partial \mathcal{L}}{\partial \mu_j} = 0$$

- Gradient ascent
  - Complex gradient (nonlinear, circular dependencies)
  - Optimization of one Gaussian component depends on all other components

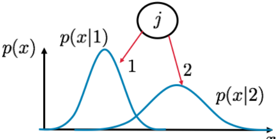
Mixture of Gaussians – Different strategy



Observed data:	●	●●●	●●	●	●
Unobserved	1	111	22	2	2
$p(j=1 \mid x)$:	1	111	00	0	0
$p(j=2 \mid x)$:	0	000	11	1	1

Unobserved $:=$ **hidden** or **latent** variables $(z_j|x)$

Mixture of Gaussians – Different strategy

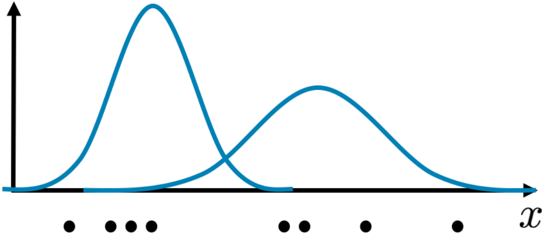


Observed data:	•	•••	••	•	•
Unobserved	1	111	22	2	2
$p(j=1\|x)$:	1	111	00	0	0
$p(j=2\|x)$:	0	000	11	1	1

- Suppose we knew the **observed** and **unobserved dataset** (also called the *complete* dataset).
- Then we can compute the maximum likelihood solution of components 1 and 2

$$\hat{\mu}_1 = \frac{\sum_{n=1}^{N} p(1|x_n)\, x_n}{\sum_{n=1}^{N} p(1|x_n)}, \qquad \hat{\mu}_2 = \frac{\sum_{n=1}^{N} p(2|x_n)\, x_n}{\sum_{n=1}^{N} p(2|x_n)}$$

Mixture of Gaussians – Different strategy



- Suppose we knew the **distributions**
- We could infer the unobserved data using Bayes Decision Rule. Namely we decide 1 if

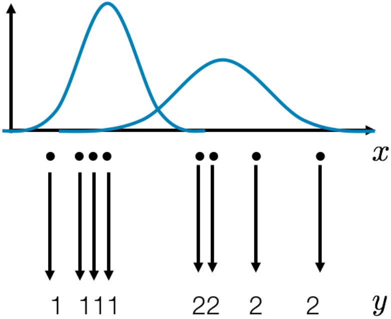
$$p\left(j=1|x\right)>p\left(j=2|x\right)$$

Mixture of Gaussians – Chicken and Egg problem

- We have big problem at hand... we neither know the distribution nor the unobserved data!
- To break this loop, we need some estimation of the unobserved data z_j.
- Temporary solution: Clustering.
- A more general Expectation-Maximization will be introduced at later lectures

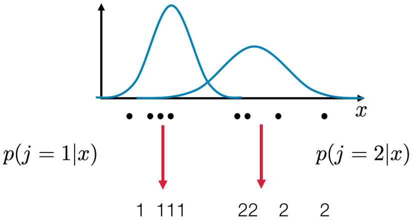
Estimation using Clustering

- Clustering with *hard* assignments
- Somehow assign mixture labels to each data point
- Estimate the mixture component only from its data



Mixture of Gaussians

- Suppose we guessed the distribution, but did not know the unobserved data



- Compute the probability for each mixture component:

$$p(j=1|x) = \frac{p(x|1)\,p(1)}{p(x)} = \frac{p(x|1)\,\pi_1}{\sum_{j=1}^{M} p(x|j)\,\pi_j}$$

$$p(j=2|x) = \frac{p(x|2)\,p(2)}{p(x)} = \frac{p(x|2)\,\pi_2}{\sum_{j=1}^{M} p(x|j)\,\pi_j}$$

Outline

1. Probability Density Estimation

2. Parametric Density Models
   Maximum Likelihood Method

3. Non-Parametric Models
   Histograms
   Kernel Density Estimation
   K-nearest Neighbors

4. Mixture models

5. Wrap-Up

5. Wrap-Up

Now you know:

- The parametric, non-parametric, and mixture models.
- More about the likelihood function and how to derive the maximum likelihood estimators for the Gaussian distribution
- What Bayesian estimation is
- Different non-parametric models (histogram, kernel density estimation and k-nearest neighbors)

Self-Test Questions

- What are parametric methods, and how to obtain their parameters?
- How many parameters have non-parametric methods?
- What are mixture models?
- Should gradient methods be used for training mixture models?
- What is the biggest problem of mixture models?

Reading Assignments

To get a deeper understanding of today's topics:

- Bishop 2006, Chapter 2.3, 2.5
- EM for Mixture Models: Bishop 2006, Chapter 9