

*** Applied Machine Learning Fundamentals ***

Probability Density Estimation (PDE)

Daniel Wehner

SAP SE

September 1, 2019



Agenda September 1, 2019

① Introduction

What about continuous Data?
Methods for PDE

② Parametric Models

General Idea
Parameter Learning and Assumptions
Maximum Likelihood Estimation (MLE)

③ Non-parametric Models

④ Mixture Models

⑤ Wrap-Up

Summary
Lecture Overview
Self-Test Questions
Recommended Literature and further Reading

Section:
Introduction



Probability Density Estimation (PDE)

- We have learned about Bayes' optimal classifiers which classify data based on the probability distribution $p(\mathbf{x}|\mathcal{C}_k) \cdot p(\mathcal{C}_k)$
- Naïve Bayes is an instance of PDE for **discrete data**
- **How to get these probabilities for the continuous case?**
 - The prior $p(\mathcal{C}_k)$ is still easy to compute
 - The estimation of class conditional probabilities $p(\mathbf{x}|\mathcal{C}_k)$ is more complicated
 - Assume labeled data; estimate the density separately for each class \mathcal{C}_k
- NB: For ease of notation: $p(\mathbf{x}) \equiv p(\mathbf{x}|\mathcal{C}_k)$

Training Data Example

Overview of the Methods for PDE

① Parametric models (Maximum likelihood estimation)

- Assume a fixed parametric form (e. g. a Gaussian distribution)
- Estimate the parameters such that the model fits the data best

② Non-parametric models

- Often we do not know the functional form of the density
- Estimate probability directly from the data without an explicit model

③ Mixture models

- Combination of ① and ②
- EM algorithm

Section:
Parametric Models



General Approach

- Given some (continuous) training data $\mathbf{X} = \{x^{(i)}\}_{i=1}^n$ (where all $x^{(i)}$ belong to the same class):
- Estimate $p(x)$ using a fixed parametric form:

Example: Gaussian Distribution

- One common case is the **Gaussian distribution**:

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

- Notation for parametric models:

- $p(x|\theta)$
- In the case of a Gaussian: $\theta = \{\mu, \sigma^2\}$

$\mu \hat{=}$ mean
 $\sigma^2 \hat{=}$ variance

Learning the Parameters

- Learning = Estimation of the parameters θ given the data \mathbf{X}
- **Likelihood** of the parameters θ :
 - Is defined as the probability that \mathbf{X} was generated by a probability density function (pdf) with parameters θ

$$\mathcal{L}(\theta) = p(\mathbf{X}|\theta) \quad (2)$$

- We want to **maximize** the likelihood

⇒ **Maximum likelihood estimation (MLE)**

A fundamental Assumption

- How to compute $\mathcal{L}(\boldsymbol{\theta})$?
- The data is assumed to be **i.i.d.** (independent and identically distributed):
 - Two random variables x_1 and x_2 are independent if

$$P(x_1 \leq \alpha, x_2 \leq \beta) = P(x_1 \leq \alpha) \cdot P(x_2 \leq \beta) \quad \forall \alpha, \beta \in \mathbb{R} \quad (3)$$

- Two random variables x_1 and x_2 are identically distributed if

$$P(x_1 \leq \alpha) = P(x_2 \leq \alpha) \quad \forall \alpha \in \mathbb{R} \quad (4)$$

Computation of the Likelihood

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}) &= p(\mathbf{X}|\boldsymbol{\theta}) \\ &= p(x^{(1)}, x^{(2)}, \dots, x^{(n)}|\boldsymbol{\theta})\end{aligned}$$

data is independent:

$$= p(x^{(1)}|\boldsymbol{\theta}) \cdot p(x^{(2)}|\boldsymbol{\theta}) \cdot \dots \cdot p(x^{(n)}|\boldsymbol{\theta})$$

data is identically distributed:

$$= \prod_{i=1}^n p(x^{(i)}|\boldsymbol{\theta})$$

What is the problem here?

(5)

Computation of the Likelihood (Ctd.)

- **Problem:** Large n might cause arithmetic underflows! (why?)
- Transform the likelihood using the logarithm \Rightarrow **log-likelihood**

$$\mathcal{LL}(\boldsymbol{\theta}) = \log \mathcal{L}(\boldsymbol{\theta})$$

Why is this an
allowed transformation?

$$= \log \prod_{i=1}^n p(x^{(i)} | \boldsymbol{\theta})$$

$$\log \Pi = \Sigma \log$$

$$= \sum_{i=1}^n \log p(x^{(i)} | \boldsymbol{\theta}) \quad (6)$$

Maximum Likelihood of a Gaussian

- $\theta = \{\mu, \sigma^2\}$

$$\begin{aligned}\mathcal{LL}(\{\mu, \sigma^2\}) &= \sum_{i=1}^n \log \mathcal{N}(x^{(i)} | \mu, \sigma^2) \\ &= \sum_{i=1}^n \log \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left\{ -\frac{(x^{(i)} - \mu)^2}{2\sigma^2} \right\}\end{aligned}$$

- Find μ_{ml} and σ_{ml}^2 which maximize the log-likelihood:

$$\mu_{ml}, \sigma_{ml}^2 = \arg \max_{\mu, \sigma^2} \mathcal{LL}(\theta)$$

Maximum Likelihood of a Gaussian (Ctd.)

- Compute the partial derivatives with respect to the parameters θ
- Derivative w. r. t. μ :

$$\nabla_{\mu} \mathcal{L}(\theta) = \nabla_{\mu} \sum_{i=1}^n \log \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left\{ -\frac{(x^{(i)} - \mu)^2}{2\sigma^2} \right\} = \sum_{i=1}^n \frac{x^{(i)} - \mu}{\sigma^2}$$

- Set derivative to zero:

$$\sum_{i=1}^n x^{(i)} - \mu \stackrel{!}{=} 0 \Leftrightarrow n \cdot \mu = \sum_{i=1}^n x^{(i)} \Leftrightarrow \mu = \frac{1}{n} \sum_{i=1}^n x^{(i)}$$

Maximization of the Likelihood

We can classify!

Looks familiar?

- Maximum likelihood parameters:

$$\mu_{ml} = \frac{1}{n} \sum_{i=1}^n x^{(i)}$$

$$\sigma_{ml}^2 = \frac{1}{n} \sum_{i=1}^n (x^{(i)} - \mu_{ml})^2$$

- Now we can use Bayes' rule to predict class labels
 - We have the priors...
 - ...and the class conditionals
- Also, the **decision boundary** can be computed

Multivariate Case

- The solution above is for 1-D data, what if we have more dimensions?
- Multivariate Gaussian distribution:**

$$\mathcal{N}_D(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^D |\boldsymbol{\Sigma}|}} \exp \left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right\} \quad (7)$$

- Luckily, the derivations don't change

$$\boldsymbol{\mu}_{ml} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}^{(i)} \quad \boldsymbol{\Sigma}_{ml} = \frac{1}{n} \sum_{i=1}^n (\mathbf{x}^{(i)} - \boldsymbol{\mu}_{ml})(\mathbf{x}^{(i)} - \boldsymbol{\mu}_{ml})^\top \quad (8)$$

MLE for the Example Data Set

Section:
Non-parametric Models



Disadvantages of parametric Models

- Until now we used a fixed parametric form (e.g. a Gaussian) which is governed by a small amount of parameters
- **This assumption may be wrong:**
 - Another distribution (exponential, gamma, ...) may fit better
 - A suitable 'text-book distribution' may not exist

We don't want to make any assumptions about the underlying distribution!

Non-parametric Approaches

- ① Histograms (Binning)
- ② Kernel density estimation (KDE)
- ③ Nearest neighbors (kNN)

Histograms

- Histograms partition the data $\mathbf{X} = \{\mathbf{x}^{(i)}\}_{i=1}^n$ into distinct **bins** of volume v_j ...
- ...and subsequently count the number of instances k_j falling into the j -th bin
- Approximate the probability $p(\mathbf{x})$ by:

$$p(\mathbf{x}) \approx \frac{k_j}{n \cdot v_j} \quad \text{for } \mathbf{x} \text{ in bin } j \quad (9)$$

- The sum of all probabilities equals 1: $\sum_{j=1}^m \frac{k_j}{n \cdot v_j} = 1$
- v_j is a **hyper-parameter** (usually, all bins have equal size)

Histograms (Ctd.)

Drawbacks of Histograms

- Histograms are mostly unsuited for many applications
- **Drawbacks:**
 - ① **Discontinuities** due to bin edges
 - ② Number of bins **explodes** with growing number of dimensions D

The latter issue is known as the curse of dimensionality

An alternative Approach

- Don't use a fixed number of pre-determined bins
- Instead, employ a **sliding window** approach by centering a region \mathcal{R} (bin) around the data point of interest \mathbf{x}

$$p(\mathbf{x}) \approx \frac{k}{n \cdot v} \quad (10)$$

- This gives rise to two different techniques:
 - ① **Kernel density estimation** (Fix v and determine k)
 - ② **k-nearest neighbors** (Fix k and determine v)

Kernel Density Estimation: Parzen Window

- \mathcal{R} is a D -dimensional **hyper-cube** of edge length h centered on \mathbf{x}
- Determine if a data point falls in region \mathcal{R} :

$$H(\mathbf{u}) = \begin{cases} 1 & \text{if } |u_d| \leq h/2, d = 1, 2, \dots, D \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

- The total number of data points falling into region \mathcal{R} is given by:

$$k(\mathbf{x}) = \sum_{i=1}^n H(\mathbf{x} - \mathbf{x}^{(i)}) \quad (12)$$

Kernel Density Estimation: Parzen Window (Ctd.)

- The volume v is simple to compute:

$$v = \int H(\mathbf{u}) d\mathbf{u} = h^D \quad (13)$$

- Putting it all together we get:

$$p(\mathbf{x}) \approx \frac{k(\mathbf{x})}{n \cdot v} = \frac{1}{n \cdot h^D} \sum_{i=1}^n H(\mathbf{x} - \mathbf{x}^{(i)}) \quad (14)$$

Section:
Mixture Models



Section:
Wrap-Up



Summary

Lecture Overview

Unit I: Machine Learning Introduction

Self-Test Questions

Recommended Literature and further Reading

Thank you very much for the attention!

Topic: *** Applied Machine Learning Fundamentals *** Probability Density Estimation (PDE)

Date: September 1, 2019

Contact:

Daniel Wehner (D062271)

SAP SE

daniel.wehner@sap.com

Do you have any questions?