Pattern Recognition and Machine Learning!

Meichen Lu (ml574@cam.ac.uk)

May 2, 2018

Contents

1 Overview

Tom M. Mitchell provided a widely quoted, more formal definition: "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E."

1.1 Different classes of learning

1.1.1 Supervised learning

- Classification
- Regression

1.1.2 Unsupervised learning

- Clustering
- Factor analysis, PCA, ICA

1.1.3 Reinforcement learning

1.2 Pipeline of machine learning

1.2.1 Preprocessing

- Dimension reduction

1.2.2 Feature extraction

1.2.3 Training/learning

1.2.4 Validation

- S-fold cross validation: when $S = N$ it is called leave-one-out
- Information criteria

1.3 Red flags

- Overfitting
- Curse of dimensionality

2 Preliminaries

2.1 Probability Theory

Frequentist/Classical Probabilities as the frequencies of random, repeatable events

Bayesian Probabilities as quantification of uncertainty.

2.1.1 Rules of probability

Sum rule

$$p(X) = \sum_{Y} p(X, Y) \qquad / \qquad p(x) = \int p(x, y) dy$$

$p(X)$ is also called marginal probability

Product rule

$$p(X,Y) = p(Y|X)p(X) \qquad / \qquad p(x,y) = p(y|x)p(x)$$

Bayes' theorem Rewritten of product rule

$$p(\boldsymbol{w}|\mathcal{D}) = \frac{p(\mathcal{D}|\boldsymbol{w})p(\boldsymbol{w})}{p(\mathcal{D})}$$

where $\mathcal{D}$ represent data and $\boldsymbol{w}$ represent model.
$p(\mathcal{D}|\boldsymbol{w})$ is called the likelihood function.

$$\text{posterior} \propto \text{likelihood} \times \text{prior}$$

2.1.2 Binomial distribution

Bernoulli distribution

$$\text{Bern}(x|\mu) = \mu^x(1-\mu)^{1-x}$$

$$\mathbb{E}[x] = \mu \qquad \text{var}[x] = \mu(1-\mu)$$

Maximum likelihood from sample is $\mu_{ML} = \frac{1}{N}\sum_{n=1}^{N} x_n$

Binomial distribution

$$\text{Bin}(m|N,\mu) = \binom{N}{m}\mu^m(1-\mu)^{N-m}$$

Maximum likelihood from sample is $\mu_{ML} = \frac{m}{N}$

Beta distribution

$$\text{Beta}(\mu|a,b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}\mu^{a-1}(1-\mu)^{b-1}$$

2.1.3 Polynomial distribution

$\boldsymbol{x}$ is a vector for K categories, e.g. $K = 6$ and $x_3 = 1$, $\boldsymbol{x} = (0,0,1,0,0,0)^T$

$$p(\boldsymbol{x}|\boldsymbol{\mu}) = \prod_{k=1}^{K}\mu_k^{x_k}$$

$$\mathbb{E}[\boldsymbol{x}|\boldsymbol{\mu}] = (\mu_1,...,\mu_M)^T = \boldsymbol{\mu}$$

Maximum likelihood from sample is $\mu_k^{ML} = \frac{m_k}{N}$

Multinomial distribution

$$\text{Mult}(m_1,m_2,...,m_K|N,\boldsymbol{\mu}) = \binom{N}{m_1 m_2 ... m_K}\prod_{k=1}^{K}\mu_k^{m_k}$$

2.1.4 Gaussian distribution

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

For a D-dimensional vector $\boldsymbol{x}$ of continuous variables

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

Maximum likelihood solution for a N *independent and identically distributed* (i.i.d.) numbers from a normal distribution

$$\mu_{ML} = \frac{1}{N}\sum_{n=1}^{N} x_n, \quad \sigma^2_{ML} = \frac{1}{N}\sum_{n=1}^{N}(x_n-\mu_{ML})^2$$

However, the unbaised estimator for the variance is (see why divide by $N-1$)

$$\tilde{\sigma} = \frac{1}{N-1}\sum_{n=1}^{N}(x_n-\mu_{ML})^2$$

2.2 Decision theory

Example: classification: posterior is $p(\mathcal{C}_k|x)$
Basic scenario: To minimise p(mistake)/maximise the p(correct) $\equiv$ maximise posterior probability.
Different cost for different types of mistake: use a loss matrix L_{kj} and minimise

$$\sum_k L_{kj}p(\mathcal{C}_k|x)$$

Rejection option to avoid making decisions on the difficult cases

Inference and decision Three types of approaches with decreasing order of complexity

1. Generative model
   - Solve the inference problem for $p(x|\mathcal{C}_k)$
   - Infer the prior class probabilities $p(\mathcal{C}_k)$
   - We can model joint probability $p(x,\mathcal{C}_k)$ directly
   - It is called generative because we can generate synthetic data points in the input space.
2. Discriminative model: only solve for $p(x|\mathcal{C}_k)$
3. Opaque model without knowing the probabilities

Benefits of knowing the posterior probability

- Minimizing risk (wrt loss matrix)

- Rejection option
- Compensating for class priors: e.g. when different classes are disproportionate, can skew the original distribution.
- Combining models

2.3 Information theory

Entropy of a random variable x

$$H[x] = -\sum_x p(x) \log_2 p(x)$$

or

$$H[x] = -\int p(x) \ln p(x) dx$$

- The noiseless coding theorem (Shannon, 1948) states that the entropy is a lower bound on the number of bits needed to transmit the state of a random variable.
- The continuous expression is called the differential entropy (in 'nats' instead of bits)
- Given the mean of a distribution is μ and the variance is σ^2, the distribution that maximizes the differential entropy is the Gaussian !!! The differential entropy is $H[x] = \frac{1}{2}\{1 + \ln(2\pi\sigma^2)\}$. This entropy can be negative, i.e. for $\sigma^2 < 1/(2\pi e)$. See also *central limit theorem*.

Conditional entropy $H[y|x]$ satiefies $H[x, y] = H[y|x] + H[x]$

Relative entropy and mutual information

$$\begin{aligned} \mathrm{KL}(p||q) &= -\int p(x) \ln q(x) dx - \left(-\int p(x) \ln p(x) dx\right) \\ &= -\int p(x) \ln \left\{\frac{q(x)}{p(x)}\right\} dx \end{aligned} \tag{1}$$

- relative entropy or Kullback-Leibler divergence, or KL divergence
- Not symmetrical $\mathrm{KL}(p||q) \neq \mathrm{KL}(p||q)$

Mutual information

$$\begin{aligned} I[x, y] \equiv & \quad \mathrm{KL}(p(x, y)||p(x)p(y)) \\ = & \quad -\iint \ln\left(\frac{p(x)p(y)}{p(x, y)}\right) dxdy \end{aligned}$$

We can view the mutual information as the reduction in the uncertainty about x by virtue of being told the value of y $I[x, y] = H[x] - H[x|y] = H[y] - H[y|x]$

3 Advanced Probability

3.1 Conjugate prior

Beta function is the conjugate prior of binomial distribution. It describes the distribution of μ based on the hyperpamaters a, b. The posterior distribution given m 'heads' and l 'tails' is

$$p(\mu|m,l,a,b) = \text{Beta}(\mu|a,b)p(\mu|m,l) = \text{Beta}(\mu|a,b)\text{Bin}(m|\mu,l) = \frac{\Gamma(m+a+l+b)}{\Gamma(m+a)\Gamma(l+b)}\mu^{m+a-1}(1-\mu)^{l+b-1}$$

- We can interpret a and b as effective number of observations of $x = 1$ and $x = 0$
- For a finite dataset, μ lies between the prior mean and the MLE of binomial distribution

Dirichlet distribution The conjugate prior for the parameters μ_k of polynomial distribution

$$p(\boldsymbol{\mu}|\boldsymbol{\alpha}) \propto \prod_{k=1}^{K} \mu_k^{\alpha_k - 1}$$

where the normalised form is

$$p(\boldsymbol{\mu}|\boldsymbol{\alpha}) = \frac{\Gamma(\alpha_0)}{\Gamma(\alpha_1)\cdots\Gamma(\alpha_K)} \prod_{k=1}^{K} \mu_k^{\alpha_k - 1}$$

where $\alpha_0 = \sum_{k=1}^{K} \alpha_k$

4 Density estimation

Density estimation walks the line between unsupervised learning, feature engineering, and data modeling.

Density estimation: general framework For a small region $\mathcal{R}$, assume that the probability density is constant $p(x)$. The probability mass associated with the region is

$$P = \int_{\mathcal{R}} p(x)dx$$

If we collected a data set containing N observations drawn from $p(x)$, the probability follows the binomial distribution and the mean of the number of points that lie inside $\mathcal{R}$ is $\mathbb{E}[K] = NP$. The probability density is

$$p(x) \approx \frac{P}{V} = \frac{K}{NV}$$

- If we fix K and determine V from the data, we have the K-nearest-neighbours (§4.2) density estimator.
- If we fix V and determine K from the data, we have the Kernel density estimators (KDE).

4.1 Kernel density estimators

kernel density estimation is a non-parametric way to estimate the probability density function of a random variable. Kernel density estimation is a fundamental data smoothing problem where inferences about the population are made, based on a finite data sample.

Top-hat kernel function We can choose any kernel function subject to the conditions

$$k(\boldsymbol{u}) \geq 0, \tag{2}$$

$$\int k(\boldsymbol{u}) d\boldsymbol{u} = 1 \tag{3}$$

A tophat function describes is a hypercube with sides 1.

$$k(\boldsymbol{u}) = \begin{cases} 1, & |u_i| \leq 1/2, \qquad i = 1, ..., D, \\ 0, & \text{otherwise} \end{cases}$$

The total number of data lying inside the hypercube is

$$K = \sum_{n=1}^{N} k\left(\frac{\boldsymbol{x} - \boldsymbol{x}_n}{h}\right)$$

Thus the estimated density at $\boldsymbol{x}$ is thus

$$p(\boldsymbol{x}) = \frac{K}{NV} = \frac{1}{N}\frac{1}{h^D}\sum_{n=1}^{N} k\left(\frac{\boldsymbol{x} - \boldsymbol{x}_n}{h}\right)$$

Gaussian kernel function

$$p(\boldsymbol{x}) = \frac{1}{N}\sum_{n=1}^{N} \frac{1}{(2\pi h^2)^{1/2}} \exp\left\{-\frac{||\boldsymbol{x} - \boldsymbol{x}_n||^2}{2h^2}\right\}$$

- Our density model is obtained by placing a Gaussian over each data point and then adding up the contributions over the whole data set
- h plays the role of a smoothing parameter.

4.2 K-nearest-neighbour

4.2.1 K-nearest-neighbour density estimation

The parameter h could be dependent on location within the data space the strategy is to fix K and search for K neighbours within a radius R, thereby the volume V can be estimated.

4.2.2 K-nearest-neighbour classifier

Likelihood function:

$$p(\boldsymbol{x}|\mathcal{C}) = \frac{K_k}{N_k V}$$

Unconditional density is (uniform)

$$p(\boldsymbol{x}) = \frac{K}{NV}$$

Class prior

$$p(\mathcal{C}_k) = \frac{N_k}{N}$$

Posterior probability:

$$p(\mathcal{C}_k|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\mathcal{C})p(\mathcal{C}_k)}{p(\boldsymbol{x})} = \frac{K_k}{K}$$

5 Graphical models

Probabilistic graphical models ≡ diagrammatic representations of probability distributions. These offer several useful properties:

1. Visualisation
2. Inspection of the properties, e.g., conditional independence
3. Helps understanding

5.1 Bayesian Networks

Directed acyclic graphs, DAG The directed graphs that we are considering are subject to an important restriction namely that there must be no directed cycles

Factorisation of the joint distribution For a graph with K nodes, the joint distribution is given by

$$p(\boldsymbol{x}) = \prod_{k=1}^{K} p(x_k|pa_k) \tag{4}$$

where $p(x_k|pa_k)$ denotes the set of parents of x_k and $\boldsymbol{x} = x_1, ..., x_K$

5.1.1 Example: Polynomial regression

5.1.2 Dimensionality and solutions

Given parameters $\boldsymbol{\mu} = (\mu_1, ..., \mu_K)^T$ and $\sum_k \mu_k = 1$, the number of parameters increase exponentially with the number of variables M, i.e. $K^2 - 1$

1. Assume that the variables are independent of each other, thus the total number of parameters would be $M(K-1)$

2. Parameter sharing
3. Using parameterized model for the conditional distributions instead of complete tables of conditional probabilities (e.g. using a sigmoid function to represent probability of y)

5.1.3 Conditional independence

Definition: a is conditionally independent of b given c:

$$p(a|b,c) = p(a|c) \text{ notation } a \perp\!\!\!\perp b|c \tag{5}$$

Explaining away Observing one of the many parent nodes (or its descendant) tells us something about the other parent nodes. (IQ, EQ, admission)

D-separation D stans for directed. If all paths are blocked from A to B given X, then A is said to be d-separated from B by C,i.e., $A \perp\!\!\!\perp B|C$ Any such path is said to be blocked if it includes a node such that either **(a)** the arrows on the path meet either head-to-tail or tail-to-tail at the node, and the node is in the set C, or **(b)** the arrows meet head-to-head at the node, and neither the node, nor any of its descendants, is in the set C.

Markov blanket We can think of the Markov blanket of a node x_i as being the minimal set of nodes that isolates x_i from the rest of the graph. The set of nodes consist of the parents, the children and the co-parents. Derivation starts with writing down the conditional probability of $p(x_i|x_{\{j\neq i\}}$

5.1.4 Naive Bayes

Naive Bayes uses a generative model by introducing a multinomial prior of the class label ($p(\boldsymbol{z}|\boldsymbol{\mu})$), together with a conditional distribution for the observed data ($p(\boldsymbol{x}|\boldsymbol{z})$).
Based on the PGM, when z is observed, x_i and x_j are conditional independent. This tells us that in general the marginal density $p(\boldsymbol{x})$ will not factorize with respect to the components of $\boldsymbol{x}$. What does this mean?
The naive Bayes assumption is helpful when

1. The input dimensionality is high
2. There are both discrete and continuous variables since each can be represented separately using appropriate models.

6 Unsupervised learning

6.1 Mixture of Gaussians and Expectation maximization

6.1.1 Mixture of Gaussians

Gaussian mixture model (GMM) as a simple linear superposition of Gaussian components. We now turn to a formulation of Gaussian mixtures in terms of discrete **latent variables**.
Latent variables, as opposed to 'observed variables', are variables that are not directly observed but are rather inferred (through a mathematical model) from other variables that are observed (directly measured). When it is corresponding to physical variable, is also called a **hidden variable** but it can also refer to an abstract concept.

Components

- There are K states, i.e., K Gaussian populations
- Latent variable $\boldsymbol{z}$ is a K-dimensional binary vector with a particular element z_k equal to 1, which means that the k^{th} state is on.
- Observed variable $\boldsymbol{x}$
- GMM likelihood function (a linear superposition of Gaussians)

$$p(\boldsymbol{x}) = \sum_{z} p(\boldsymbol{z}) p(\boldsymbol{x}|\boldsymbol{z}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{6}$$

  - Possibility that the k^{th} state is on: π_k
  - Gaussian parameters of the k^{th} state/population: $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$

6.1.2 GMM for the i.i.d. data set

Given N observations:

- matrix $\mathbf{X} \in \mathbb{R}^{N \times m}$ in which the n^{th} row is given by $x^{(n)^T}$
- matrix $\mathbf{Z} \in \mathbb{R}^{N \times k}$ in which the n^{th} row is given by $z^{(n)^T}$
- **i.i.d.** means that the data points are drawn **independently** from the distribution.

Log-likelihood function for the data set

$$ln(p(\mathbf{X}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}^{(n)}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right\} \tag{7}$$

Derivative of log-likelihood function w.r.t. $\boldsymbol{\mu}_k$

$$-\sum_{n=1}^{N} \frac{\pi_k \mathcal{N}(\boldsymbol{x}^{(n)}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_j \pi_j \mathcal{N}(\boldsymbol{x}^{(n)}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)} \boldsymbol{\Sigma}(\boldsymbol{x}^{(n)} - \boldsymbol{\sigma}_k) = 0 \tag{8}$$

The direct gradient descent algorithm is too complicated, and amazing mathematicians have invented expectation maximization! The strategy will be to instead repeatedly **construct a lower-bound** on p (E-step), and then optimize that lower-bound (M-step).

6.1.3 EM for GMM

In Eq. 8, we see that $\frac{\pi_k \mathcal{N}(\boldsymbol{x}^{(n)}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_j \pi_j \mathcal{N}(\boldsymbol{x}^{(n)}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}$ is the posterior probability, we define it as $\gamma(z_k^{(n)})$.

Posterior probability/Responsibility $\gamma(z_k^{(n)})$ can be viewed as the responsibility that component k takes for explaining the observation$x$.

E step We guess the parameters of the model and compute the posterior probability of the latent variables

M step Using the posterior probability, we **update** the parameters that maximizes the probability of the observed variables.

EM algorithm

1. Initialise the parameter: the means $\boldsymbol{\mu}_k$, covariances $\boldsymbol{\Sigma}_k$ and the mixing coefficients π_k; and evaluate the initial log likelihood
2. **E step** Evaluate the posterior probabilities using the current parameter values

$$\gamma(z_k^{(n)}) = \frac{\pi_k \mathcal{N}(\boldsymbol{x}^{(n)}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}^{(n)}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)} \tag{9}$$

3. **M step** Re-estimate the parameters using the current posterior probabilities

$$\boldsymbol{\mu}_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_k^{(n)}) \boldsymbol{x}^{(n)} \tag{10}$$

$$\boldsymbol{\Sigma}_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_k^{(n)}) (\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k^{new})(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k^{new})^T \tag{11}$$

$$\pi_k^{new} = \frac{N_k}{N} \tag{12}$$

where

$$N_k = \sum_{n=1}^{N} \gamma(z_k^{(n)}) \tag{13}$$

We can interpret N_k as the effective number of points assigned to cluster k.

4. Evaluate the log likelihood and check the convergence. If not, return to step 2

6.2 EM Revisited

Instead of calculating the likelihood of the observed variable $\boldsymbol{x}$, we can treat the vector $[\boldsymbol{x}, \boldsymbol{z}]$ as the *complete* data set. Although we do not know $\boldsymbol{z}$, we can represent it as the posterior distribution, i.e. the*expected value* and it is found in the **E step**. The expectation, denoted $\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{old})$

$$\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{old}) \ln p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}) \tag{14}$$

In the **M step**, the parameters are revised

$$\boldsymbol{\theta}^{new} = \arg\max_{\theta} \mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) \tag{15}$$

6.2.1 Mixture of Gaussians Revisited

Under this framework, the likelihood function for the complete data set $\boldsymbol{X}, \boldsymbol{Z}$ is thus

$$(p(\mathbf{X},\ \mathbf{Z}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \prod_{n=1}^{N} \prod_{k=1}^{K} \pi_k^{z_k^{(n)}} \mathcal{N}(\boldsymbol{x}^{(n)}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)^{z_k^{(n)}} \tag{16}$$

Giving the log-likelihood function of

$$ln(p(\mathbf{X},\ \mathbf{Z}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{n=1}^{N} \sum_{k=1}^{K} z_k^{(n)} \{\ln \pi_k + \ln \mathcal{N}(\boldsymbol{x}^{(n)}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\} \tag{17}$$

Note that this form is more easily optimised than Eq. 7

6.3 K-means clustering

K-means algorithm is a particular limit of EM for Gaussian mixtures.
Consider a Gaussian mixture model in which the covariance matrices of the mixture components are given by $\epsilon\mathbf{I}$, where ϵ is a variance parameter that is shared by all the components.
For $\epsilon \to 0$, it can be shown that responsibilities $\gamma(z_k^{(n)})$ for the data point $\boldsymbol{x}^{(n)}$ all go to zero except for the term j, where the mean $\boldsymbol{\mu}_j$ is closest to the data, which goes to unity. Thus, in this limit, we obtain a **hard assignment** of data points to clusters

6.4 Factor analysis

Factor analysis is a linear-Gaussian latent variable model that is closely related to **probabilistic PCA.** Its definition differs from that of probabilistic PCA only in that the conditional distribution of the observed variable $\boldsymbol{x}$ given the latent variable $\boldsymbol{z}$ is taken to have a diagonal rather than an isotropic covariance.

Motivation: When the feature dimension is much higher than the sample size. (There are interesting latent variables that can explain the data? The features can be split into latent ones and 'explicit' ones?)

6.4.1 Model

Joint distribution $\boldsymbol{x}, \boldsymbol{z}$ where $\boldsymbol{z} \in \mathbb{R}^k$ is a latent random variable

$$\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}) \tag{18}$$

$$\boldsymbol{x}|\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{\mu} + \boldsymbol{W}\boldsymbol{z}, \boldsymbol{\Psi}) \tag{19}$$

Joint probability distribution

$$\begin{bmatrix} \boldsymbol{z} \\ \boldsymbol{x} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{\mu} \end{bmatrix}, \begin{bmatrix} \boldsymbol{I} & \boldsymbol{W}^T \\ \boldsymbol{W} & \boldsymbol{W}\boldsymbol{W}^T + \boldsymbol{\Psi} \end{bmatrix} \right) \tag{20}$$

6.4.2 EM for factor analysis

E step

$$\boldsymbol{\mu}_{\boldsymbol{z}^{(n)}|\boldsymbol{x}^{(n)}} = \boldsymbol{W}^T(\boldsymbol{W}\boldsymbol{W}^T + \boldsymbol{\Psi})^{-1}(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}) \tag{21}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{z}^{(n)}|\boldsymbol{x}^{(n)}} = \boldsymbol{I} - \boldsymbol{W}^T(\boldsymbol{W}\boldsymbol{W}^T + \boldsymbol{\Psi})^{-1}\boldsymbol{W} \tag{22}$$

Thus the posterior distribution

$$p(\boldsymbol{z}^{(n)}|\boldsymbol{x}^{(n)}; \boldsymbol{\mu}, \boldsymbol{W}, \boldsymbol{\Psi}) \sim \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{z}^{(n)}|\boldsymbol{x}^{(n)}}, \boldsymbol{\Sigma}_{\boldsymbol{z}^{(n)}|\boldsymbol{x}^{(n)}}) \tag{23}$$

M step

$$\boldsymbol{W} = \left(\sum_{n=1}^{N} (\boldsymbol{x}^{(n)} - \boldsymbol{\mu}) \boldsymbol{\mu}^T_{\boldsymbol{z}^{(n)}|\boldsymbol{x}^{(n)}} \right) \left(\sum_{n=1}^{N} \boldsymbol{\mu}_{\boldsymbol{z}^{(n)}|\boldsymbol{x}^{(n)}} \boldsymbol{\mu}^T_{\boldsymbol{z}^{(n)}|\boldsymbol{x}^{(n)}} + \boldsymbol{\Sigma}_{\boldsymbol{z}^{(n)}|\boldsymbol{x}^{(n)}} \right) \tag{24}$$

$$\boldsymbol{\mu} = \frac{1}{N} \sum_{n=1}^{N} \boldsymbol{x}^{(n)} \tag{25}$$

$$\boldsymbol{\Psi} = \frac{1}{N} \sum_{n=1}^{N} \boldsymbol{x}^{(n)} \boldsymbol{x}^{(n)^T} - \boldsymbol{x}^{(n)} \boldsymbol{\mu}^T_{\boldsymbol{z}^{(n)}|\boldsymbol{x}^{(n)}} \boldsymbol{W}^T - \boldsymbol{W} \boldsymbol{\mu}_{\boldsymbol{z}^{(n)}|\boldsymbol{x}^{(n)}} \boldsymbol{x}^{(n)^T} + \boldsymbol{W}(\boldsymbol{\mu}_{\boldsymbol{z}^{(n)}|\boldsymbol{x}^{(n)}} \boldsymbol{\mu}^T_{\boldsymbol{z}^{(n)}|\boldsymbol{x}^{(n)}} + \boldsymbol{\Sigma}^T_{\boldsymbol{z}^{(n)}|\boldsymbol{x}^{(n)}}) \boldsymbol{W}^T \tag{26}$$

7 Sequential Data

7.1 Markov models

First-order Markov chain: Each variable is independent of all previous observations except the most recent.

$$p(\boldsymbol{x_1}, ..., \boldsymbol{x_N}) = p(\boldsymbol{x}_1) \prod_{n=1}^{N} p(\boldsymbol{x}|\boldsymbol{x}_{n-1}) \tag{27}$$

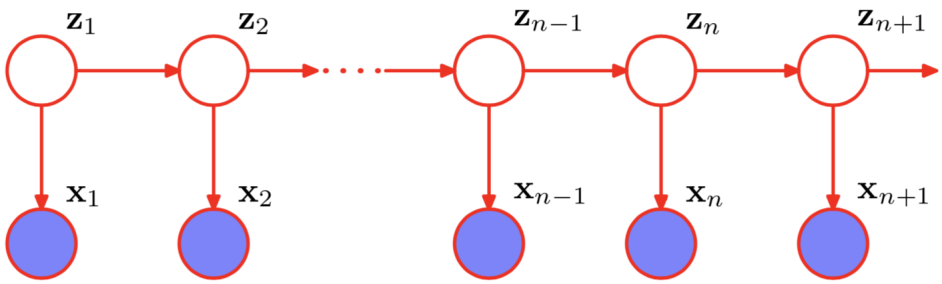


Figure 1: Probabilistic graphical model for HMM. Figure 13.5 from Bishop (2006)

Extensions are: 2^{nd} order Markov chain, ..., M^{th} order Markov chain.

7.2 Hidden Markov models

Suppose we wish to build a model for sequences that is not limited by the Markov assumption to any order and yet that can be specified using a limited number of free parameters. **Use latent variables**.
HMM: for each observation $\boldsymbol{x_n}$, we introduce a corresponding latent variable $\boldsymbol{z}_n$. Given by the PGM in Fig. 1 such that $\boldsymbol{z}_{n+1} \perp\!\!\!\perp \boldsymbol{z}_{n-1}\boldsymbol{z}_n$

Generative model perspective Every time slice of the model can be seen as a mixture distribution, with component densities given by $p(\boldsymbol{x}|\boldsymbol{z})$ and the multinomial variable $\boldsymbol{z}_n$ describes which component of the mixture is responsible for generating the corresponding observation $\boldsymbol{x}_n$

Transition probabilities Conditional probability table represented by matrix $\boldsymbol{A}$ gives $A_{jk} = p(z_{nk} = 1|z_{n-1,j} = 1)$

Emission probabilities $p(\boldsymbol{x}_n|\boldsymbol{z}_n, \phi)$ describes how the observed variable depends on the latent variable (and other parameters ϕ).
We can represent the emission probabilities in the form

$$p(\boldsymbol{x}_n|\boldsymbol{z}_n, \phi) = \prod_{k=1}^{K} p(\boldsymbol{x}_n|\phi_k)^{z_{nk}} \tag{28}$$

left-to-right HMM The state index can only increase

7.2.1 EM for HMM

Given a data set $\mathbf{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_N\}$ we can determine the parameters of an HMM using maximum likelihood but the exponentially growth of number of terms in the summation is difficult to solve.

$$p(\mathbf{X}|\boldsymbol{\theta}) = \sum_{\mathbf{Z}} p(\mathbf{X},\mathbf{Z}|\boldsymbol{\theta}) \tag{29}$$

E-step Find the posterior distribution $p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{old})$ For convenience, we define the following notations

$$\gamma(\boldsymbol{z}_n) = p(\boldsymbol{z}_n|\mathbf{X}, \boldsymbol{\theta}^{old}) \tag{30}$$

$$\eta(\boldsymbol{z}_{n-1}, \boldsymbol{z}_n) = p(\boldsymbol{z}_{n-1}, \boldsymbol{z}_n|\mathbf{X}, \boldsymbol{\theta}^{old}) \tag{31}$$

$\gamma(z_{nk})$ is the conditional probability of $z_{nk} = 1$ and similarly for $\eta(z_{n-1,j}, z_{nk})$. Because the expectation of a binary random variable is just the probability that it takes the value 1, we have

$$\gamma(z_{nk}) = \mathbb{E}[z_{nk}] = \sum_{\boldsymbol{z}} \gamma(\boldsymbol{z}) z_{nk} \tag{32}$$

$$\eta(z_{n-1,j}, z_{nk}) = \mathbb{E}[z_{n-1,j} z_{nk}] = \sum_{\boldsymbol{z}} \gamma(\boldsymbol{z}) z_{n-1,j} z_{nk} \tag{33}$$

M-step Maximize the log-likelihood function with respect to the parameters $\boldsymbol{\theta}$

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) = \sum_{\boldsymbol{Z}} p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{old}) \ln p(\mathbf{X},\mathbf{Z}|\boldsymbol{\theta}) \tag{34}$$

For HMM

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) = \sum_{k=1}^{K} \gamma(z_{1k}) \ln \pi_k + \sum_{n=2}^{N} \sum_{j=1}^{K} \sum_{k=1}^{K} \eta(z_{n-1,j}, z_{nk}) \ln A_{jk} + \sum_{n=1}^{N} \sum_{k=1}^{K} \gamma(z_{nk}) \ln p(\boldsymbol{x}_n|\boldsymbol{\phi}_k) \tag{35}$$

Optimisation algorithm

1. Using Lagrangian multiplier
$$\pi_k = \frac{\gamma(z_{1k})}{\sum_{j=1}^{K} \gamma(z_{1j})} \tag{36}$$
2. Similarly, the transition probabilities
$$A_{jk} = \frac{\sum_{n=2}^{N} \xi(z_{n-1,j}, z_{nk})}{\sum_{l=1}^{K} \sum_{n=2}^{N} \xi(z_{n-1,j}, z_{nl})} \tag{37}$$
3. For the parameter in the emission probabilities, depends on the model but it is the same as N i.i.d. examples in a mixture model.

To initialise the parameters:

1. π_k to random numbers from a uniform distribution that add to 1
2. A_{jk} randomly & add to 1 in each row
3. $\boldsymbol{\phi}$ according to the maximum likelihood of the mixture model assuming that the data are i.i.d

7.2.2 The forward-backward algorithm

The algorithm to evaluate the quantities $\gamma(\boldsymbol{z}_n), \xi(\boldsymbol{z}_{n-1}, \boldsymbol{z}_n)$ efficiently.

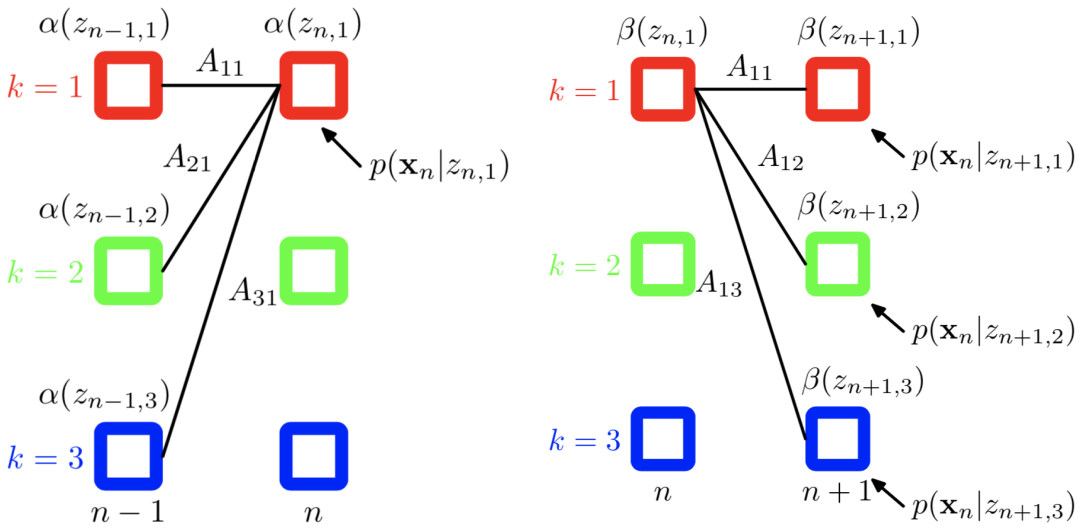


Figure 2: Illustration of the forward recursion for alpha and backward recursion for beta

Derivation using sum and product rules & conditional independence from the PGM. We omit the explicit dependency on $\boldsymbol{\theta}^{old}$.

$$\gamma(\boldsymbol{z}_n) = p(\boldsymbol{z}_n|\mathbf{X}) = \frac{p(\mathbf{X}|\boldsymbol{z}_n)p(\boldsymbol{z}_n)}{p(\mathbf{X})} \tag{38}$$

Rewrite and define $\alpha(\boldsymbol{z}_n)$ and $\beta(\boldsymbol{z}_n)$

$$p(\mathbf{X}|\boldsymbol{z}_n)p(\boldsymbol{z}_n) = p(\boldsymbol{x}_1, ..., \boldsymbol{x}_n, \boldsymbol{z}_n)p(\boldsymbol{x}_{n+1}, ..., \boldsymbol{x}_N|\boldsymbol{z}_n) = \alpha(\boldsymbol{z}_n)\beta(\boldsymbol{z}_n) \tag{39}$$

Deriving recursive relationships

$$\alpha(\boldsymbol{z}_n) = p(\boldsymbol{x}_n|\boldsymbol{z}_n) \sum_{\boldsymbol{z}_{n-1}} \alpha(\boldsymbol{z}_{n-1})p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1}) \tag{40}$$

Initial condition

$$\alpha(\boldsymbol{z}_1) = p(\boldsymbol{x}_1, \boldsymbol{z}_1) = p(\boldsymbol{z}_1)p(\boldsymbol{x}_1|\boldsymbol{z}_1) = \prod_{k=1}^{K} \{\pi_k p(\boldsymbol{x}_1|\boldsymbol{\phi}_k)\}^{z_{1k}} \tag{41}$$

Similarly

$$\beta(\boldsymbol{z}_n) = \sum_{\boldsymbol{z}_{n+1}} \beta(\boldsymbol{z}_{n+1})p(\boldsymbol{x}_{n+1}|\boldsymbol{z}_{n+1})p(\boldsymbol{z}_{n+1}|\boldsymbol{z}_n) \tag{42}$$

Initial condition: $\beta(\boldsymbol{z}_N) = 1$ Fig. 2 shows the two processes

EM with alpha & beta in the M-step the marginal likelihood $p(\mathbf{X})$ cancels out. E.g. for Gaussian mixture model

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^{n} \gamma(z_{nk})\boldsymbol{x}_n}{\sum_{n=1}^{n} \gamma(z_{nk})} = \frac{\sum_{n=1}^{n} \alpha(z_{nk})\beta(z_{nk})\boldsymbol{x}_n}{\sum_{n=1}^{n} \alpha(z_{nk})\beta(z_{nk})} \tag{43}$$

To evaluate the marginal likelihood

$$p(\mathbf{X}) = \sum_{\boldsymbol{z}_n} \alpha(\boldsymbol{z}_n)\beta(\boldsymbol{z}_n) = \sum_{\boldsymbol{z}_n} \alpha(\boldsymbol{z}_n) \tag{44}$$

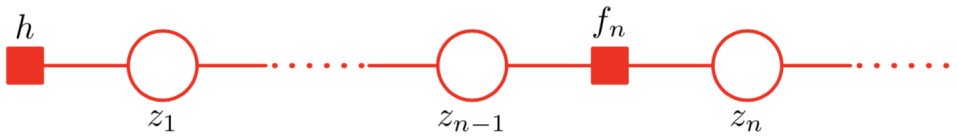


Figure 3: Simplified factor graph for HMM. Figure 13.15 from Bishop (2006)

7.2.3 sum-product algorithm for HMM

Using factor graph (Fig. 3) and sum-product rule, the above relationships can be derived in a much simpler way.

The factors are given by

$$h(\boldsymbol{z}_1) = p(\boldsymbol{z}_1)p(\boldsymbol{x}_1|\boldsymbol{z}_1) \tag{45}$$

$$f_n(\boldsymbol{z}_{n-1}, \boldsymbol{z}_n) = p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1})p(\boldsymbol{x}_n|\boldsymbol{z}_n) \tag{46}$$

The message propagation in HMM is given by

$$\mu_{\boldsymbol{z}_{n-1}\to f_n}(\boldsymbol{z}_{n-1}) = \mu_{f_{n-1}\to \boldsymbol{z}_{n-1}}(\boldsymbol{z}_{n-1}) \tag{47}$$

$$\mu_{f_n\to \boldsymbol{z}_n}(\boldsymbol{z}_n) = \sum_{\boldsymbol{z}_{n-1}} f_n(\boldsymbol{z}_{n-1}, \boldsymbol{z}_n)\mu_{\boldsymbol{z}_{n-1}\to f_n}(\boldsymbol{z}_{n-1}) \tag{48}$$

If we define

$$\alpha(\boldsymbol{z}_n) = \mu_{f_n\to \boldsymbol{z}_n}(\boldsymbol{z}_n)\beta(\boldsymbol{z}_n) = \mu_{f_{n+1}\to \boldsymbol{z}_n}(\boldsymbol{z}_n) \tag{49}$$

we have

$$p(\boldsymbol{z}_n, \mathbf{X}) = \mu_{f_n\to \boldsymbol{z}_n}(\boldsymbol{z}_n)\mu_{f_{n+1}\to \boldsymbol{z}_n}(\boldsymbol{z}_n) = \alpha(\boldsymbol{z}_n)\beta(\boldsymbol{z}_n) \tag{50}$$

After normalisation

$$\gamma(\boldsymbol{z}_n) = \frac{p(\boldsymbol{z}_n, \mathbf{X})}{p(\mathbf{X})} = \frac{\alpha(\boldsymbol{z}_n)\beta(\boldsymbol{z}_n)}{p(\mathbf{X})} \tag{51}$$

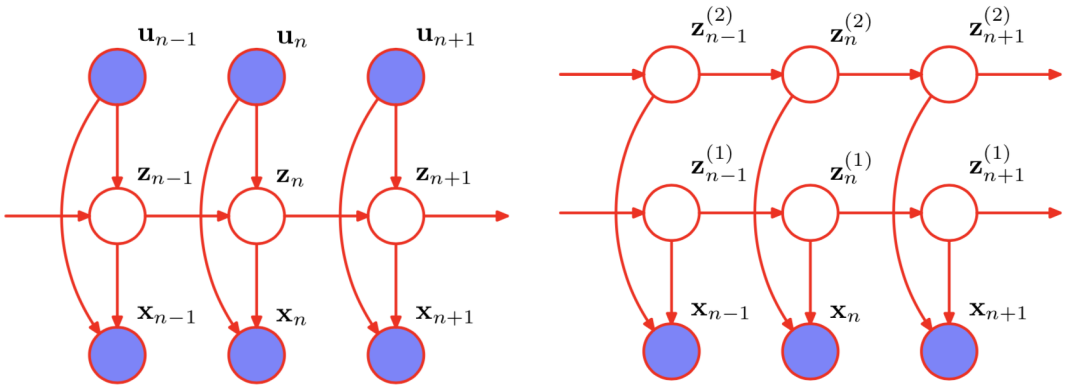
7.2.4 Extensions to HMM

Driving forces "**Graphical models** provide a general technique for motivating, describing, and analysing such structures, and **variational methods** provide a powerful framework for performing inference in those models for which exact solution is intractable."

Discriminative HMM For classification purposes, nstead of the MLE technique, use discriminative techniques. E.g. to optimize the cross-entropy:

$$\sum_{r=1}^{R} \ln p(m_r|\boldsymbol{X}_r) \tag{52}$$

where there are R observed sequences each labelled as class m. Commonly used in speech recognition.



(a) PGM of the input output HMM.

(b) PGM of the factorial HMM.

Figure 4: Figures 13.18 and 13.19 from Bishop (2006)

Modified transition mechanism The probability of a hidden state persisting decays exponentially and is not realistic for state duration for many applications.

- New state transition matrix has $A_k k$ set to zero
- Whether the current state persist or transit depends on another probability distribution $p(T|k)$
- Modification to the EM required

Autoregressive HMM to capture long-range correlations (as in the higher order Markov chain)

Input-output HMM Additional observed variable as input u in Fig. 4a.

Factorial HMM Instead of high-dimensional latent states, use multiple latent chains. (e.g. for 10 bits of information, a single chain HMM needs $K = 2^10 = 1024$ latent states while it can be replaced by 10 binary latent chains.). See Fig. 4b.

8 Learning theory

Courtesy to CS229 by Andrew Ng.

8.1 Bias-variance trade-off

Intuition:

- Bias: Low model complexity → model unable to fit data well (high training and testing error)
  - "Informally, we define the bias of a model to be the expected generalization error even if we were to fit it to a very (say, infinitely) large training set."

- Variance: High model complexity → model unable to generalize well. (low training error and high testing error)
  - The variance could mean that the model fits the variation in the training data.

More quantitative analysis:

The **mean squared error (MSE)** is:

$$\mathbb{E}\left((y - \hat{f}(x))^2\right)$$

If $y = f(x) + \epsilon$ The above equation can be rewritten as

$$\mathbb{E}\left((\epsilon + f(x) - \hat{f}(x))^2\right)$$

Rearrange the terms

$$\text{TestMSE} = \sigma^2 + \left(\mathbb{E}(f(x) - \hat{f}(x))\right)^2 + Var\left(f(x) - \hat{f}(x)\right)$$

The three terms are noise variance, Bias $\hat{f}(x)$ and Variance of the model.

8.2 Error analysis

For a machine learning pipeline

Groud-truth plugin By plugging-in the ground-truth for each component, see how accuracy changes

Ablative analysis By removing a component, see how accuracy changes

Analysing the mistake See whether there is trend.

8.3 Learning theory theorems

8.3.1 Preliminaries

lemma: The union bound The probability of any one of k events happening is at most the sums of the probabilities of the k different events.

lemma: Hoeffding inequality For m i.i.d. random variables drawn from a Bernoulli(ϕ) distribution. Let $\hat{\phi}$ be the mean of the sample. Then

$$P(|\phi - \hat{\phi}| > \gamma) \leq 2\exp(-\gamma^2 m)$$

Some definitions

- **PAC** stands for probably approximately correct which is a **framework** and set of assumptions under which numerous results on learning theory were proved. Of these, the assumption of training and testing on the **same distribution**, and the assumption of the independently drawn training examples, were the most important.
- **training error** (also called the empirical risk or empirical error in learning theory)
- **empirical risk minimization (ERM)** The process of choosing the the hypothesis that minimises the empirical risk (i.e. training error)
- **Hypothesis class** $\mathcal{H}$ used by a learning algorithm is the set of all classifiers considered by it.

8.3.2 Finite hypothesis class

Theorem Let $|\mathcal{H}| = k$, and let any m, δ be fixed. Then with probability at least 1δ, we have that

$$\epsilon(\hat{h}) \leq \left(\min_{h \in \mathcal{H}} \epsilon(h)\right) + 2\sqrt{\frac{1}{2m} \log \frac{2k}{\delta}}$$

- With more complex model ($\mathcal{H} \subseteq \mathcal{H}'$, k is higher), the training error ($\epsilon(h)$) will be lower, the second term is higher (variance)
- We can calculate the size m for a given error $\epsilon(\hat{h})$ for fixed probability.

8.3.3 Infinite hypothesis class

Shatter Given a set (S) of points, we say that $\mathcal{H}$ shatters S if $\mathcal{H}$ can realize any labeling on S.

Vapnik-Chervonenkis dimension Given a hypothesis class $\mathcal{H}$, we then define its Vapnik-Chervonenkis dimension, written VC($\mathcal{H}$), to be the size of the largest set that is shattered by v. (If $\mathcal{H}$ can shatter arbitrarily large sets, then VC($\mathcal{H}$) $= \infty$.)

Theorem Let $\mathcal{H}$ be given, and let $d = \text{VC}(\mathcal{H})$. Then with probability at least 1δ, we have that

$$\epsilon(\hat{h}) \leq \left(\min_{h \in \mathcal{H}} \epsilon(h)\right) + O\left(\sqrt{\frac{d}{m} \log \frac{m}{d} + \frac{1}{m} \log \frac{1}{\delta}}\right)$$