Announcements

- Exam dates
  - According to rwth online, the exam dates are:
    - 1st try: Sat 02.03.2019, 10:30 – 12:00h
    - 2nd try: Thu 21.03.2019, 13:30 – 15:30h
  - Exam registration will start in early December...

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)
    - Take the derivative and set it to zero.
      $$\frac{\partial}{\partial \theta} E(\theta) = -\sum_{n=1}^{N} \frac{\partial p(x_n|\theta)}{p(x_n|\theta)} \cdot \ln p(x_n|\theta) = 0$$

Recap: Kernel Density Estimation

- Approximation formula:
  $$p(x) \approx \frac{K}{NV}$$
  - fixed $V$ determine $K$
  - fixed $K$ determine $V$

- Kernel methods
  - Place a kernel window $k$ at location $x$ and count how many data points fall inside it.
- K-Nearest Neighbor
  - Increase the volume $V$ until the $K$ nearest data points are found.
Topics of This Lecture

- Mixture distributions
  - Mixture of Gaussians (MoG)
  - Maximum Likelihood estimation attempt
- K-Means Clustering
  - Algorithm
  - Applications
- EM Algorithm
  - Credit assignment problem
  - MoG estimation
  - EM Algorithm
  - Interpretation of K-Means
  - Technical advice
- Applications

Mixture Distributions

- A single parametric distribution is often not sufficient
  - E.g. for multimodal data

Mixture of Gaussians (MoG)

- Sum of $M$ individual Normal distributions
  \[ f(x) = \sum_{j=1}^{M} p(x|\theta_j) p(j) \]

  - In the limit, every smooth distribution can be approximated this way (if $M$ is large enough)

Mixture of Gaussians

\[ p(x|\theta) = \sum_{j=1}^{M} p(x|\theta_j) p(j) \]

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

\[ p(j) = \pi_j \text{ with } 0 < \pi_j < 1 \text{ and } \sum_{j=1}^{M} \pi_j = 1 \]

Notes

- The mixture density integrates to 1:
  \[ \int p(x) dx = 1 \]

- The mixture parameters are
  \[ \theta = (\pi_1, \mu_1, \sigma_1, \ldots, \pi_M, \mu_M, \sigma_M) \]
Mixture of Multivariate Gaussians

- Multivariate Gaussians
  \[ p(x|\theta) = \sum_{j=1}^{M} p(x|\theta_j)p(j) \]
  \[ p(x|\theta_j) = \frac{1}{(2\pi)^{D/2}|\Sigma_j|^{1/2}} \exp\left\{-\frac{1}{2}(x - \mu_j)^T \Sigma_j^{-1}(x - \mu_j)\right\} \]
  - Mixture weights / mixture coefficients:
    \[ p(j) = \pi_j \text{ with } 0 < \pi_j < 1 \text{ and } \sum_{j=1}^{M} \pi_j = 1 \]
  - Parameters:
    \[ \theta = (\pi_1, \mu_1, \Sigma_1, \ldots, \pi_M, \mu_M, \Sigma_M) \]

Mixture of Gaussians – 1st Estimation Attempt

- Maximum Likelihood
  - Minimize \( E = -\ln L(\theta) = -\sum_{n=1}^{N} \ln p(x_n|\theta) \)
  - Let’s first look at \( \mu_j \):
    \[ \frac{\partial E}{\partial \mu_j} = 0 \]
    - We can already see that this will be difficult, since
      \[ \ln p(X|\pi, \mu, \Sigma) = \sum_{n=1}^{N} \left\{ \sum_{j=1}^{M} \ln \pi_j \mathcal{N}(x_n|\mu_j, \Sigma_j) \right\} \]
      This will cause problems!

Mixture of Gaussians – 1st Estimation Attempt

- But...
  \[ \mu_j = \frac{\sum_{n=1}^{N} \gamma_j(x_n) x_n}{\sum_{n=1}^{N} \gamma_j(x_n)} = \frac{\pi_j \mathcal{N}(x_n|\mu, \Sigma)}{\sum_{k=1}^{K} \pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)} \]
  - I.e. there is no direct analytical solution!
    \[ \frac{\partial E}{\partial \mu_j} = f(x_1, \mu_j, \Sigma_1, \ldots, \mu_M, \Sigma_M) \]
    - Complex gradient function (non-linear mutual dependencies)
    - Optimization of one Gaussian depends on all other Gaussians!
    - It is possible to apply iterative numerical optimization here, but in the following, we will see a simpler method.

Mixture of Multivariate Gaussians

- “Generative model”

\[ p(x|\theta) = \sum_{j=1}^{M} \pi_j p(x|\theta_j) \]

Mixture of Gaussians – Other Strategy

- Other strategy:
  - Observed data:
  - Unobserved data:
    - Unobserved = “hidden variable”: \( j \)
    \[ h(j = 1|x_n) = \begin{cases} 1 & 111 \\ 0 & 000 \end{cases} \]
    \[ h(j = 2|x_n) = \begin{cases} 0 & 111 \\ 111 & 000 \end{cases} \]
Mixture of Gaussians – Other Strategy

- Assuming we knew the values of the hidden variable...

\[ f(x) \]

ML for Gaussian #1

assumed known

\[ h(j = 1|x_n) = \begin{pmatrix} 1 \\ 111 \end{pmatrix} \]

\[ h(j = 2|x_n) = \begin{pmatrix} 0 \\ 000 \end{pmatrix} \]

\[ \mu_1 = \frac{\sum_{n=1}^{N} h(j = 1|x_n)x_n}{\sum_{n=1}^{N} h(j = 1|x_n)} \]

\[ \mu_2 = \frac{\sum_{n=1}^{N} h(j = 2|x_n)x_n}{\sum_{n=1}^{N} h(j = 2|x_n)} \]

ML for Gaussian #2

Bayes decision rule: Decide \( j = 1 \) if

\[ p(j = 1|x_n) > p(j = 2|x_n) \]

Assuming we knew the mixture components...

\[ f(x) \]

\[ p(j = 1|x) \]

\[ p(j = 2|x) \]

\[ j \]

B. Leibe

Slide credit: Bernt Schiele

Clustering with Hard Assignments

- Let’s first look at clustering with “hard assignments”

\[ f(x) \]

\[ x \]

\[ 1 \quad 111 \quad 22 \quad 2 \quad 2 \quad j \]

B. Leibe

Slide credit: Bernt Schiele

K-Means Clustering

- Iterative procedure
  1. Initialization: pick \( K \) arbitrary centroids (cluster means)
  2. Assign each sample to the closest centroid.
  3. Adjust the centroids to be the means of the samples assigned to them.
  4. Go to step 2 (until no change)

- Algorithm is guaranteed to converge after finite #iterations.
  - Local optimum
  - Final result depends on initialization.

B. Leibe

Slide credit: Bernt Schiele

Topics of This Lecture

- Mixture distributions
  - Mixture of Gaussians (MoG)
  - Maximum Likelihood estimation attempt

- K-Means Clustering
  - Algorithm
  - Applications

- EM Algorithm
  - Credit assignment problem
  - MoG estimation
  - EM Algorithm
  - Interpretation of K-Means
  - Technical advice

- Applications

B. Leibe

Slide credit: Bernt Schiele

K-Means – Example with \( K=2 \)
**K-Means Clustering**

- K-Means optimizes the following objective function:
  \[ J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} ||x_n - \mu_k||^2 \]
- where \( r_{nk} = \begin{cases} 1 & \text{if } k = \text{arg min}_j ||x_n - \mu_j||^2 \\ 0 & \text{otherwise} \end{cases} \)
- I.e., \( r_{nk} \) is an indicator variable that checks whether \( \mu_k \) is the nearest cluster center to point \( x_n \).
- In practice, this procedure usually converges quickly to a local optimum.

**Example Application: Image Compression**

- **Original image**
- **K = 2**
- **K = 3**
- **K = 10**

**Summary K-Means**

- **Pros**
  - Simple, fast to compute
  - Converges to local minimum of within-cluster squared error
- **Problem cases**
  - Setting \( k \)?
  - Sensitive to initial centers
  - Sensitive to outliers
  - Detects spherical clusters only
- **Extensions**
  - Speed-ups possible through efficient search structures
  - General distance measures: k-medoids

**EM Clustering**

- Clustering with “soft assignments”
  - Expectation step of the EM algorithm
  \[ f(x) \]
  \[ p(j|x) \]
  \[ p(1|x), 0.99, 0.8, 0.2, 0.01, j \]
  \[ p(2|x), 0.01, 0.2, 0.8, 0.99 \]
EM Clustering

- Clustering with “soft assignments”
  - Maximization step of the EM algorithm
    \[ f(x) = \frac{\sum_{n=1}^{N} p(j|x_n)x_n}{\sum_{n=1}^{N} p(j|x_n)} \]
  - \( \mu_j = \sum_{n=1}^{N} p(j|x_n)x_n \)

- An Example
  - Will converge to a local optimum of \( f(x) \) if \( x_n \) is exactly centered on data point \( n \).
  - We can however evaluate the posterior probability that an observed \( x_n \) was generated from the first mixture component.

Technical Advice

- Convergence is relatively slow.
- Need to introduce regularization
  - Enforce minimum width for the Gaussians
  - E.g., instead of \( \Sigma^{-1} \), use \( (\Sigma + \sigma^2 I)^{-1} \)

Credit Assignment Problem

- “Credit Assignment Problem”
  - If we are just given \( x \), we don’t know which mixture component this example came from
    \[ p(x|\theta) = \frac{1}{\sum_{j=1}^{K} p(x|\theta_j)} \]
  - We can however evaluate the posterior probability that an observed \( x \) was generated from the first mixture component.
    \[ p(j=1|x, \theta) = \frac{p(j=1, x|\theta)}{p(x|\theta)} \]
    \[ p(j=1|x) = \frac{p(x|\theta_j)p(j=1)}{\sum_{j=1}^{K} p(x|\theta_j)p(j)} = \gamma_j(x) \]
    \( \text{“responsibility” of component } j \text{ for } x. \)

EM Algorithm

- Expectation-Maximization (EM) Algorithm
  - E-Step: softly assign samples to mixture components
    \[ \gamma_j(x_n) = \frac{p(N(x_n|\mu_j, \Sigma_j))}{\sum_{j=1}^{K} p(N(x_n|\mu_j, \Sigma_j))} \quad \forall j = 1, \ldots, K; \ n = 1, \ldots, N \]
  - M-Step: re-estimate the parameters (separately for each mixture component) based on the soft assignments
    \[ \hat{N}_j = \sum_{n=1}^{N} \gamma_j(x_n) = \text{soft number of samples labeled } j \]
    \[ \hat{\mu}_j = \frac{1}{\hat{N}_j} \sum_{n=1}^{N} \gamma_j(x_n)x_n \]
    \[ \hat{\Sigma}_j = \frac{1}{\hat{N}_j} \sum_{n=1}^{N} \gamma_j(x_n)(x_n - \hat{\mu}_j)(x_n - \hat{\mu}_j) \]

EM – Technical Advice

- When implementing EM, we need to take care to avoid singularities in the estimation!
  - Mixture components may collapse on single data points.
  - E.g. consider the case \( \Sigma_k = \sigma^2 I \) (this also holds in general)
  - Assume component \( j \) is exactly centered on data point \( x_n \). This data point will then contribute a term in the likelihood function
    \[ N(x_n|x_n, \sigma^2 I) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{|x_n - \mu_j|^2}{2\sigma^2}} \]
    - For \( \sigma \to 0 \), this term goes to infinity!
  - Need to introduce regularization
    - Enforce minimum width for the Gaussians
    - E.g., instead of \( \Sigma^{-1} \), use \( (\Sigma + \sigma^2 I)^{-1} \)

EM – Technical Advice (2)

- EM is very sensitive to the initialization
  - Will converge to a local optimum of \( E \).
  - Convergence is relatively slow.
  - Initialize with k-Means to get better results!
    - k-Means is itself initialized randomly, will also only find a local optimum.
    - But convergence is much faster.
  - Typical procedure
    - Run k-Means \( M \) times (e.g. \( M = 10–100 \)).
    - Pick the best result (lowest error \( J_j \)).
    - Use this result to initialize EM
      - Set \( \mu_j \) to the corresponding cluster mean from k-Means.
      - Initialize \( \Sigma_j \) to the sample covariance of the associated data points.
K-Means Clustering Revisited

• Interpreting the procedure
  1. Initialization: pick $K$ arbitrary centroids (cluster means)
  2. Assign each sample to the closest centroid. (E-Step)
  3. Adjust the centroids to be the means of the samples assigned to them. (M-Step)
  4. Go to step 2 (until no change)

Summary: Gaussian Mixture Models

• Properties
  - Very general, can represent any (continuous) distribution.
  - Once trained, very fast to evaluate.
  - Can be updated online.

• Problems / Caveats
  - Some numerical issues in the implementation
    - Need to apply regularization in order to avoid singularities.
  - EM for MoG is computationally expensive
    - Especially for high-dimensional problems!
    - More computational overhead and slower convergence than k-Means
  - Results very sensitive to initialization
    - Run k-Means for some iterations as initialization!
  - Need to select the number of mixture components $K$.
    - Model selection problem (see later lecture)

Applications

• Mixture models are used in many practical applications.
  - Wherever distributions with complex or unknown shapes need to be represented...

  - Popular application in Computer Vision
    - Model distributions of pixel colors.
    - Each pixel is one data point in, e.g., RGB space.
    - Learn a MoG to represent the class-conditional densities.
    - Use the learned models to classify other pixels.

Application: Background Model for Tracking

• Train background MoG for each pixel
  - Model “common” appearance variation for each background pixel.
  - Initialization with an empty scene.
  - Update the mixtures over time
    - Adapt to lighting changes, etc.

  - Used in many vision-based tracking applications
    - Anything that cannot be explained by the background model is labeled as foreground (=object).
    - Easy segmentation if camera is fixed.

Topics of This Lecture

• Mixture distributions
  - Recap: Mixture of Gaussians (MoG)
  - Maximum Likelihood estimation attempt

• K-Means Clustering
  - Algorithm
  - Applications

• EM Algorithm
  - Credit assignment problem
  - MoG estimation
  - EM Algorithm
  - Interpretation of K-Means
  - Technical advice

• Applications
Application: Image Segmentation

- User assisted image segmentation
  - User marks two regions for foreground and background.
  - Learn a MoG model for the color values in each region.
  - Use those models to classify all other pixels.
  - Simple segmentation procedure
    (building block for more complex applications)

References and Further Reading

- More information about EM and MoG estimation is available in Chapter 2.3.9 and the entire Chapter 9 of Bishop’s book (recommendable to read).

  - More information
    - Original EM paper:
    - EM tutorial:
      - J.A. Bilmes, “A Gentle Tutorial of the EM Algorithm and Its Application to
        Parameter Estimation for Gaussian Mixture and Hidden Markov Models”
      - TR-97-021, ICSI, U.C. Berkeley, CA,USA