Introduction to Machine Learning (67577)
Lecture 14

Shai Shalev-Shwartz

School of CS and Engineering,
The Hebrew University of Jerusalem

Generative Models

Generative Models

- The Generative Approach: try to learn the distribution $\mathcal{D}$ over the data

Generative Models

- The Generative Approach: try to learn the distribution $\mathcal{D}$ over the data
- If we know $\mathcal{D}$ we can predict using the Bayes optimal classifier

Generative Models

- The Generative Approach: try to learn the distribution $\mathcal{D}$ over the data
- If we know $\mathcal{D}$ we can predict using the Bayes optimal classifier
- Usually, it is much harder to learn $\mathcal{D}$ than to simply learn a predictor

Generative Models

- The Generative Approach: try to learn the distribution $\mathcal{D}$ over the data
- If we know $\mathcal{D}$ we can predict using the Bayes optimal classifier
- Usually, it is much harder to learn $\mathcal{D}$ than to simply learn a predictor
- However, in some situations, it is reasonable to adopt the generative learning approach:

.

Generative Models

- The Generative Approach: try to learn the distribution $\mathcal{D}$ over the data
- If we know $\mathcal{D}$ we can predict using the Bayes optimal classifier
- Usually, it is much harder to learn $\mathcal{D}$ than to simply learn a predictor
- However, in some situations, it is reasonable to adopt the generative learning approach:
  - Computational reasons

.

Generative Models

- The Generative Approach: try to learn the distribution $\mathcal{D}$ over the data
- If we know $\mathcal{D}$ we can predict using the Bayes optimal classifier
- Usually, it is much harder to learn $\mathcal{D}$ than to simply learn a predictor
- However, in some situations, it is reasonable to adopt the generative learning approach:
  - Computational reasons
  - We sometimes don't have a specific task at hand

.

Generative Models

- The Generative Approach: try to learn the distribution $\mathcal{D}$ over the data
- If we know $\mathcal{D}$ we can predict using the Bayes optimal classifier
- Usually, it is much harder to learn $\mathcal{D}$ than to simply learn a predictor
- However, in some situations, it is reasonable to adopt the generative learning approach:
  - Computational reasons
  - We sometimes don't have a specific task at hand
  - Interpretability of the data

.

Outline

Maximum Likelihood Estimator

- We assume as prior knowledge that the underlying distribution is parameterized by some θ

Maximum Likelihood Estimator

- We assume as prior knowledge that the underlying distribution is parameterized by some θ
- Learning the distribution corresponds to finding θ

Maximum Likelihood Estimator

- We assume as prior knowledge that the underlying distribution is parameterized by some θ
- Learning the distribution corresponds to finding θ
- Example: let $\mathcal{X} = \{0, 1\}$ then the set of distributions over $\mathcal{X}$ are parameterized by a single number $\theta \in [0, 1]$ corresponding to $\mathbb{P}_{x \sim \mathcal{D}_\theta}[x = 1] = \mathcal{D}_\theta(\{1\}) = \theta$

Maximum Likelihood Estimator

- We assume as prior knowledge that the underlying distribution is parameterized by some θ
- Learning the distribution corresponds to finding θ
- Example: let $\mathcal{X} = \{0,1\}$ then the set of distributions over $\mathcal{X}$ are parameterized by a single number $\theta \in [0,1]$ corresponding to $\mathbb{P}_{x\sim\mathcal{D}_\theta}[x=1] = \mathcal{D}_\theta(\{1\}) = \theta$
- The goal is to learn θ from a sequence of i.i.d. examples $S = (x_1,\ldots,x_m) \sim \mathcal{D}_\theta^m$

Maximum Likelihood Estimator

- Likelihood: The likelihood of S, assuming the distribution is $\mathcal{D}_\theta$, is defined to be

$$\mathcal{D}_\theta^m(\{S\}) = \prod_{i=1}^m \mathcal{D}_\theta(\{x_i\}) = \prod_{i=1}^m \underset{X \sim \mathcal{D}_\theta}{\mathbb{P}}[X = x_i]$$

Maximum Likelihood Estimator

- Likelihood: The likelihood of S, assuming the distribution is $\mathcal{D}_\theta$, is defined to be

$$\mathcal{D}_\theta^m(\{S\}) = \prod_{i=1}^m \mathcal{D}_\theta(\{x_i\}) = \prod_{i=1}^m \underset{X \sim \mathcal{D}_\theta}{\mathbb{P}}[X = x_i]$$

- Log-Likelihood: it is convenient to denote

$$L(S;\theta) = \log\left(\mathcal{D}_\theta^m(\{S\})\right) = \sum_{i=1}^m \log\left(\underset{X \sim \mathcal{D}_\theta}{\mathbb{P}}[X = x_i]\right)$$

Maximum Likelihood Estimator

- Likelihood: The likelihood of S, assuming the distribution is $\mathcal{D}_\theta$, is defined to be

$$\mathcal{D}_\theta^m(\{S\}) = \prod_{i=1}^m \mathcal{D}_\theta(\{x_i\}) = \prod_{i=1}^m \underset{X\sim\mathcal{D}_\theta}{\mathbb{P}}[X = x_i]$$

- Log-Likelihood: it is convenient to denote

$$L(S;\theta) = \log\left(\mathcal{D}_\theta^m(\{S\})\right) = \sum_{i=1}^m \log\left(\underset{X\sim\mathcal{D}_\theta}{\mathbb{P}}[X = x_i]\right)$$

- Maximum Likelihood Estimator (MLE): estimate θ based on S according to

$$\hat{\theta}(S) = \underset{\theta}{\operatorname{argmax}}\, L(S;\theta) \ .$$

Maximum Likelihood Estimator for Bernoulli variables

- Suppose $\mathcal{X} = \{0, 1\}$, $\mathcal{D}_\theta$ is the distribution s.t. $P_{x \sim \mathcal{D}_\theta}[x = 1] = \theta$

Maximum Likelihood Estimator for Bernoulli variables

- Suppose $\mathcal{X} = \{0,1\}$, $\mathcal{D}_\theta$ is the distribution s.t. $P_{x\sim\mathcal{D}_\theta}[x=1] = \theta$
- The log-likelihood function:

$$L(S;\theta) \;=\; \log(\theta)\cdot|\{i : x_i = 1\}| + \log(1-\theta)\cdot|\{i : x_i = 0\}|$$

Maximum Likelihood Estimator for Bernoulli variables

- Suppose $\mathcal{X} = \{0, 1\}$, $\mathcal{D}_\theta$ is the distribution s.t. $P_{x \sim \mathcal{D}_\theta}[x = 1] = \theta$
- The log-likelihood function:

$$L(S; \theta) \;=\; \log(\theta) \cdot |\{i : x_i = 1\}| + \log(1 - \theta) \cdot |\{i : x_i = 0\}|$$

- Maximizing w.r.t. θ gives the ML estimator. Taking derivative w.r.t. θ and comparing to zero gives:

$$\frac{|\{i : x_i = 1\}|}{\hat{\theta}} - \frac{|\{i : x_i = 0\}|}{1 - \hat{\theta}} = 0 \quad \Rightarrow \quad \hat{\theta} \;=\; \frac{|\{i : x_i = 1\}|}{m}$$

Maximum Likelihood Estimator for Bernoulli variables

- Suppose $\mathcal{X} = \{0, 1\}$, $\mathcal{D}_\theta$ is the distribution s.t. $P_{x\sim\mathcal{D}_\theta}[x = 1] = \theta$
- The log-likelihood function:

$$L(S; \theta) \;=\; \log(\theta) \cdot |\{i : x_i = 1\}| + \log(1-\theta) \cdot |\{i : x_i = 0\}|$$

- Maximizing w.r.t. θ gives the ML estimator. Taking derivative w.r.t. θ and comparing to zero gives:

$$\frac{|\{i : x_i = 1\}|}{\hat{\theta}} - \frac{|\{i : x_i = 0\}|}{1-\hat{\theta}} = 0 \quad \Rightarrow \quad \hat{\theta} \;=\; \frac{|\{i : x_i = 1\}|}{m}$$

- That is, $\hat{\theta}$ is the average number of ones in S

Maximum Likelihood for Continuous Variables

- Example: $\mathcal{X} = [0,1]$ and $\mathcal{D}_\theta$ is the uniform distribution. Then, $\mathcal{D}_\theta(\{x\}) = 0$ for all x so $L(S;\theta) = -\infty$ ...

Maximum Likelihood for Continuous Variables

- Example: $\mathcal{X} = [0,1]$ and $\mathcal{D}_\theta$ is the uniform distribution. Then, $\mathcal{D}_\theta(\{x\}) = 0$ for all x so $L(S;\theta) = -\infty$ ...
- To overcome the problem, we define L using the density distribution:

$$L(S;\theta) = \sum_{i=1}^{m} \log\left(\mathcal{P}_{x\sim\mathcal{D}_\theta}[x = x_i]\right)$$

Maximum Likelihood for Continuous Variables

- Example: $\mathcal{X} = [0,1]$ and $\mathcal{D}_\theta$ is the uniform distribution. Then, $\mathcal{D}_\theta(\{x\}) = 0$ for all x so $L(S;\theta) = -\infty$ ...
- To overcome the problem, we define L using the density distribution:

$$L(S;\theta) = \sum_{i=1}^{m} \log\left(\mathcal{P}_{x\sim\mathcal{D}_\theta}[x = x_i]\right)$$

- E.g., for Gaussian distribution, with $\theta = (\mu, \sigma)$,

$$\mathcal{P}_{x\sim\mathcal{D}_\theta}(x_i) \;=\; \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x_i-\mu)^2}{2\sigma^2}\right)$$

and

$$L(S;\theta) = -\frac{1}{2\sigma^2}\sum_{i=1}^{m}(x_i-\mu)^2 - m\log(\sigma\sqrt{2\,\pi}) \ .$$

Maximum Likelihood for Continuous Variables

- Example: $\mathcal{X} = [0,1]$ and $\mathcal{D}_\theta$ is the uniform distribution. Then, $\mathcal{D}_\theta(\{x\}) = 0$ for all x so $L(S;\theta) = -\infty$ ...
- To overcome the problem, we define L using the density distribution:

$$L(S;\theta) = \sum_{i=1}^{m} \log\left(\mathcal{P}_{x\sim\mathcal{D}_\theta}[x = x_i]\right)$$

- E.g., for Gaussian distribution, with $\theta = (\mu, \sigma)$,

$$\mathcal{P}_{x\sim\mathcal{D}_\theta}(x_i) \;=\; \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x_i-\mu)^2}{2\sigma^2}\right)$$

and

$$L(S;\theta) = -\frac{1}{2\sigma^2}\sum_{i=1}^{m}(x_i-\mu)^2 - m\log(\sigma\sqrt{2\,\pi}) \;.$$

- MLE becomes: $\hat{\mu} \;=\; \frac{1}{m}\sum_{i=1}^m x_i$ and $\hat{\sigma} \;=\; \sqrt{\frac{1}{m}\sum_{i=1}^m (x_i-\hat{\mu})^2}$

Outline

Naive Bayes

- A classical demonstration of how generative assumptions and parameter estimations simplify the learning process

Naive Bayes

- A classical demonstration of how generative assumptions and parameter estimations simplify the learning process
- Goal: learn function $h : \{0,1\}^d \rightarrow \{0,1\}$

Naive Bayes

- A classical demonstration of how generative assumptions and parameter estimations simplify the learning process
- Goal: learn function $h : \{0,1\}^d \to \{0,1\}$
- Bayes optimal classifier:

$$h_{\text{Bayes}}(\mathbf{x}) \;=\; \operatorname*{argmax}_{y \in \{0,1\}} \mathcal{P}[Y = y | X = \mathbf{x}] \; .$$

Naive Bayes

- A classical demonstration of how generative assumptions and parameter estimations simplify the learning process
- Goal: learn function $h : \{0,1\}^d \to \{0,1\}$
- Bayes optimal classifier:

$$h_{\text{Bayes}}(\mathbf{x}) = \operatorname*{argmax}_{y \in \{0,1\}} \mathcal{P}[Y = y | X = \mathbf{x}] .$$

- Need 2^d parameters for describing $\mathcal{P}[Y = y | X = \mathbf{x}]$ for every $\mathbf{x} \in \{0,1\}^d$

Naive Bayes

- A classical demonstration of how generative assumptions and parameter estimations simplify the learning process
- Goal: learn function $h : \{0,1\}^d \to \{0,1\}$
- Bayes optimal classifier:

$$h_{\text{Bayes}}(\mathbf{x}) \;=\; \operatorname*{argmax}_{y \in \{0,1\}} \mathcal{P}[Y = y | X = \mathbf{x}] \ .$$

- Need 2^d parameters for describing $\mathcal{P}[Y = y | X = \mathbf{x}]$ for every $\mathbf{x} \in \{0,1\}^d$
- Naive generative assumption: features are independent given the label:

$$\mathcal{P}[X = \mathbf{x} | Y = y] \;=\; \prod_{i=1}^{d} \mathcal{P}[X_i = x_i | Y = y]$$

Naive Bayes

- With this (rather naive) assumption and using Bayes rule, the Bayes optimal classifier can be further simplified:

$$
\begin{aligned}
h_{\text{Bayes}}(\mathbf{x}) &= \underset{y\in\{0,1\}}{\operatorname{argmax}}\, \mathcal{P}[Y=y|X=\mathbf{x}] \\
&= \underset{y\in\{0,1\}}{\operatorname{argmax}}\, \mathcal{P}[Y=y]\mathcal{P}[X=\mathbf{x}|Y=y] \\
&= \underset{y\in\{0,1\}}{\operatorname{argmax}}\, \mathcal{P}[Y=y]\prod_{i=1}^{d}\mathcal{P}[X_i=x_i|Y=y] \ .
\end{aligned}
$$

Naive Bayes

- With this (rather naive) assumption and using Bayes rule, the Bayes optimal classifier can be further simplified:

$$
\begin{aligned}
h_{\text{Bayes}}(\mathbf{x}) &= \operatorname*{argmax}_{y\in\{0,1\}} \mathcal{P}[Y=y|X=\mathbf{x}] \\
&= \operatorname*{argmax}_{y\in\{0,1\}} \mathcal{P}[Y=y]\mathcal{P}[X=\mathbf{x}|Y=y] \\
&= \operatorname*{argmax}_{y\in\{0,1\}} \mathcal{P}[Y=y]\prod_{i=1}^{d}\mathcal{P}[X_i=x_i|Y=y] \ .
\end{aligned}
$$

- Now, number of parameters to estimate is $2d+1$

Naive Bayes

- With this (rather naive) assumption and using Bayes rule, the Bayes optimal classifier can be further simplified:

$$\begin{aligned} h_{\text{Bayes}}(\mathbf{x}) &= \operatorname*{argmax}_{y\in\{0,1\}} \mathcal{P}[Y=y|X=\mathbf{x}] \\ &= \operatorname*{argmax}_{y\in\{0,1\}} \mathcal{P}[Y=y]\mathcal{P}[X=\mathbf{x}|Y=y] \\ &= \operatorname*{argmax}_{y\in\{0,1\}} \mathcal{P}[Y=y]\prod_{i=1}^{d}\mathcal{P}[X_i=x_i|Y=y] \ . \end{aligned}$$

- Now, number of parameters to estimate is $2d+1$
- Reduces both runtime and sample complexity

Outline

Linear Discriminant Analysis

- Another demonstration of how generative assumptions simplify the learning process

Linear Discriminant Analysis

- Another demonstration of how generative assumptions simplify the learning process
- Goal: learn $h : \mathbb{R}^d \to \{0, 1\}$

Linear Discriminant Analysis

- Another demonstration of how generative assumptions simplify the learning process
- Goal: learn $h : \mathbb{R}^d \to \{0, 1\}$
- The generative assumption: y is generated based on $\mathcal{P}[Y = 1] = \mathcal{P}[Y = 0] = 1/2$ and given y, $\mathbf{x} \sim \mathbb{N}(\boldsymbol{\mu}_y, \Sigma)$:

$$\mathcal{P}[X = \mathbf{x}|Y = y] = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_y)^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}_y)\right)$$

Linear Discriminant Analysis

- Another demonstration of how generative assumptions simplify the learning process
- Goal: learn $h : \mathbb{R}^d \to \{0, 1\}$
- The generative assumption: y is generated based on $\mathcal{P}[Y = 1] = \mathcal{P}[Y = 0] = 1/2$ and given y, $\mathbf{x} \sim \mathbb{N}(\boldsymbol{\mu}_y, \Sigma)$:

$$\mathcal{P}[X = \mathbf{x}|Y = y] \;=\; \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_y)^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}_y)\right)$$

- Bayes rule:

$$h_{\text{Bayes}}(\mathbf{x}) \;=\; \underset{y \in \{0,1\}}{\operatorname{argmax}} \, \mathcal{P}[Y = y] \mathcal{P}[X = \mathbf{x}|Y = y]$$

Linear Discriminant Analysis

- Another demonstration of how generative assumptions simplify the learning process
- Goal: learn $h : \mathbb{R}^d \to \{0,1\}$
- The generative assumption: y is generated based on $\mathcal{P}[Y=1] = \mathcal{P}[Y=0] = 1/2$ and given y, $\mathbf{x} \sim \mathbb{N}(\boldsymbol{\mu}_y, \Sigma)$:

$$\mathcal{P}[X = \mathbf{x} | Y = y] \;=\; \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp\left(-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_y)^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}_y) \right)$$

- Bayes rule:

$$h_{\text{Bayes}}(\mathbf{x}) \;=\; \underset{y \in \{0,1\}}{\operatorname{argmax}} \, \mathcal{P}[Y = y] \mathcal{P}[X = \mathbf{x} | Y = y]$$

- This means we will predict $h_{\text{Bayes}}(\mathbf{x}) = 1$ iff

$$\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_0)^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}_0) - \frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_1)^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}_1) > 0$$

Linear Discriminant Analysis

- Equivalent to $\langle \mathbf{w}, \mathbf{x} \rangle + b > 0$ where

$$\mathbf{w} = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)^T \Sigma^{-1} \quad \text{and} \quad b = \frac{1}{2} \left(\boldsymbol{\mu}_0^T \Sigma^{-1} \boldsymbol{\mu}_0 - \boldsymbol{\mu}_1^T \Sigma^{-1} \boldsymbol{\mu}_1 \right)$$

Linear Discriminant Analysis

- Equivalent to $\langle \mathbf{w}, \mathbf{x} \rangle + b > 0$ where

$$\mathbf{w} \;=\; (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)^T \Sigma^{-1} \quad \text{and} \quad b = \frac{1}{2}\left(\boldsymbol{\mu}_0^T \Sigma^{-1} \boldsymbol{\mu}_0 - \boldsymbol{\mu}_1^T \Sigma^{-1} \boldsymbol{\mu}_1\right)$$

- That is, Bayes optimal is a halfspace in this case

Linear Discriminant Analysis

- Equivalent to $\langle \mathbf{w}, \mathbf{x} \rangle + b > 0$ where

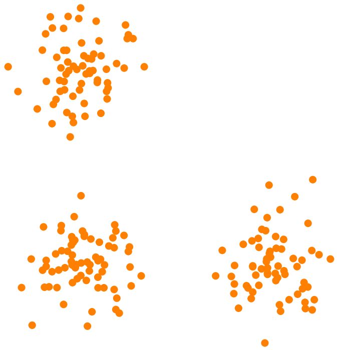
$$\mathbf{w} = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)^T \Sigma^{-1} \quad \text{and} \quad b = \frac{1}{2}\left(\boldsymbol{\mu}_0^T \Sigma^{-1} \boldsymbol{\mu}_0 - \boldsymbol{\mu}_1^T \Sigma^{-1} \boldsymbol{\mu}_1\right)$$

- That is, Bayes optimal is a halfspace in this case
- But, instead of learning the halfspace directly, we'll learn $\boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \Sigma$ using maximum likelihood.

Outline

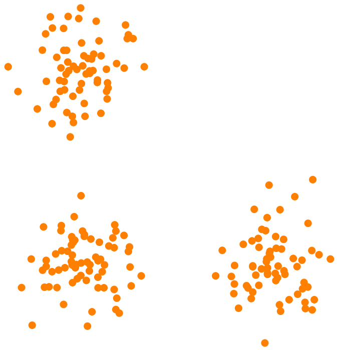
Latent Variables

- Sometimes, it is convenient to express the distribution over $\mathcal{X}$ using latent random variables



Latent Variables

- Sometimes, it is convenient to express the distribution over $\mathcal{X}$ using latent random variables
- Mixture of Gaussians: Each $\mathbf{x} \in \mathbb{R}^d$ is generated by first selecting a random y from $[k]$, then choose $\mathbf{x}$ according to $N(\boldsymbol{\mu}_y, \Sigma_y)$



Mixture of Gaussians

- Each $\mathbf{x} \in \mathbb{R}^d$ is generated by first selecting a random y from $[k]$, then choose $\mathbf{x}$ according to $N(\boldsymbol{\mu}_y, \Sigma_y)$
- The density can be written as:

$$
\begin{aligned}
\mathcal{P}[X = \mathbf{x}] &= \sum_{y=1}^{k} \mathcal{P}[Y = y]\mathcal{P}[X = \mathbf{x}|Y = y] \\
&= \sum_{y=1}^{k} c_y \, \frac{1}{(2\pi)^{d/2}|\Sigma_y|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_y)^T \Sigma_y^{-1} (\mathbf{x} - \boldsymbol{\mu}_y)\right)
\end{aligned}
$$

- Note: Y is a hidden variable that we do not observe in the data. It is just used to simplify the parametric description of the distribution

Latent Variables

- More generally,

$$\log\left(\mathcal{P}_{\boldsymbol{\theta}}[X = \mathbf{x}]\right) = \log\left(\sum_{y=1}^{k} \mathcal{P}_{\boldsymbol{\theta}}[X = \mathbf{x}, Y = y]\right) .$$

Latent Variables

- More generally,

$$\log\left(\mathcal{P}_{\boldsymbol{\theta}}[X=\mathbf{x}]\right) = \log\left(\sum_{y=1}^{k} \mathcal{P}_{\boldsymbol{\theta}}[X=\mathbf{x}, Y=y]\right) .$$

- Maximum Likelihood:

$$\underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{i=1}^{m} \log\left(\sum_{y=1}^{k} \mathcal{P}_{\boldsymbol{\theta}}[X=\mathbf{x}_i, Y=y]\right) .$$

Latent Variables

- More generally,

$$\log\left(\mathcal{P}_{\boldsymbol{\theta}}[X=\mathbf{x}]\right) = \log\left(\sum_{y=1}^{k} \mathcal{P}_{\boldsymbol{\theta}}[X=\mathbf{x}, Y=y]\right) .$$

- Maximum Likelihood:

$$\underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{i=1}^{m} \log\left(\sum_{y=1}^{k} \mathcal{P}_{\boldsymbol{\theta}}[X=\mathbf{x}_i, Y=y]\right) .$$

- In many cases, the summation inside the log makes the above optimization problem computationally hard

Latent Variables

- More generally,

$$\log\left(\mathcal{P}_{\boldsymbol{\theta}}[X = \mathbf{x}]\right) = \log\left(\sum_{y=1}^{k} \mathcal{P}_{\boldsymbol{\theta}}[X = \mathbf{x}, Y = y]\right) .$$

- Maximum Likelihood:

$$\underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{i=1}^{m} \log\left(\sum_{y=1}^{k} \mathcal{P}_{\boldsymbol{\theta}}[X = \mathbf{x}_i, Y = y]\right) .$$

- In many cases, the summation inside the log makes the above optimization problem computationally hard
- A popular heuristic: *Expectation-Maximization* (EM), due to Dempster, Laird and Rubin

Expectation-Maximization (EM)

- Designed for cases in which, had we known the values of the latent variables Y, then the maximum likelihood optimization problem would have been tractable.

Expectation-Maximization (EM)

- Designed for cases in which, had we known the values of the latent variables Y, then the maximum likelihood optimization problem would have been tractable.
- Precisely, define the following function over $m \times k$ matrices and the set of parameters $\boldsymbol{\theta}$:

$$F(Q, \boldsymbol{\theta}) = \sum_{i=1}^{m} \sum_{y=1}^{k} Q_{i,y} \log \left(\mathcal{P}_{\boldsymbol{\theta}}[X = \mathbf{x}_i, Y = y] \right)$$

Expectation-Maximization (EM)

- Designed for cases in which, had we known the values of the latent variables Y, then the maximum likelihood optimization problem would have been tractable.
- Precisely, define the following function over $m \times k$ matrices and the set of parameters $\boldsymbol{\theta}$:

$$F(Q, \boldsymbol{\theta}) = \sum_{i=1}^{m} \sum_{y=1}^{k} Q_{i,y} \log \left(\mathcal{P}_{\boldsymbol{\theta}}[X = \mathbf{x}_i, Y = y] \right)$$

- Interpret $F(Q, \boldsymbol{\theta})$ as the expected log-likelihood of $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)$

Expectation-Maximization (EM)

- Designed for cases in which, had we known the values of the latent variables Y, then the maximum likelihood optimization problem would have been tractable.
- Precisely, define the following function over $m \times k$ matrices and the set of parameters $\boldsymbol{\theta}$:

$$F(Q, \boldsymbol{\theta}) = \sum_{i=1}^{m} \sum_{y=1}^{k} Q_{i,y} \log \left(\mathcal{P}_{\boldsymbol{\theta}}[X = \mathbf{x}_i, Y = y] \right)$$

- Interpret $F(Q, \boldsymbol{\theta})$ as the expected log-likelihood of $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)$
- Assumption: For any matrix $Q \in [0,1]^{m,k}$, such that each row of Q sums to 1, the optimization problem $\operatorname{argmax}_{\boldsymbol{\theta}} F(Q, \boldsymbol{\theta})$ is tractable.

Expectation-Maximization (EM)

- "chicken and egg" problem: Had we known Q, easy to find $\boldsymbol{\theta}$. Had we known $\boldsymbol{\theta}$, we can set $Q_{i,y} = \mathcal{P}_{\boldsymbol{\theta}}[Y = y | X = \mathbf{x}_i]$

Expectation-Maximization (EM)

- "chicken and egg" problem: Had we known Q, easy to find $\boldsymbol{\theta}$. Had we known $\boldsymbol{\theta}$, we can set $Q_{i,y} = \mathcal{P}_{\boldsymbol{\theta}}[Y = y | X = \mathbf{x}_i]$
- **E**xpectation step: set

$$Q_{i,y}^{(t+1)} = \mathcal{P}_{\boldsymbol{\theta}^{(t)}}[Y = y | X = \mathbf{x}_i] \ .$$

Expectation-Maximization (EM)

- "chicken and egg" problem: Had we known Q, easy to find $\boldsymbol{\theta}$. Had we known $\boldsymbol{\theta}$, we can set $Q_{i,y} = \mathcal{P}_{\boldsymbol{\theta}}[Y = y | X = \mathbf{x}_i]$
- **E**xpectation step: set

$$Q_{i,y}^{(t+1)} = \mathcal{P}_{\boldsymbol{\theta}^{(t)}}[Y = y | X = \mathbf{x}_i] \ .$$

- **M**aximization step: set

$$\boldsymbol{\theta}^{(t+1)} = \operatorname*{argmax}_{\boldsymbol{\theta}} F(Q^{(t+1)}, \boldsymbol{\theta}) \ .$$

EM as an alternate maximization algorithm

- EM can be viewed as alternate maximization on the objective

$$G(Q, \boldsymbol{\theta}) = F(Q, \boldsymbol{\theta}) - \sum_{i=1}^{m} \sum_{y=1}^{k} Q_{i,y} \log(Q_{i,y}) \ .$$

EM as an alternate maximization algorithm

- EM can be viewed as alternate maximization on the objective

$$G(Q, \boldsymbol{\theta}) = F(Q, \boldsymbol{\theta}) - \sum_{i=1}^{m} \sum_{y=1}^{k} Q_{i,y} \log(Q_{i,y}) \ .$$

- Lemma: The EM procedure can be rewritten as:

$$\begin{aligned} Q^{(t+1)} &= \operatorname*{argmax}_{Q \in [0,1]^{m,k} : \forall i, \sum_y Q_{i,j} = 1} G(Q, \boldsymbol{\theta}^{(t)}) \\ \boldsymbol{\theta}^{(t+1)} &= \operatorname*{argmax}_{\boldsymbol{\theta}} G(Q^{(t+1)}, \boldsymbol{\theta}) \ . \end{aligned}$$

Furthermore, $G(Q^{(t+1)}, \boldsymbol{\theta}^{(t)}) = L(S; \boldsymbol{\theta}^{(t)})$.

EM as an alternate maximization algorithm

- EM can be viewed as alternate maximization on the objective

$$G(Q,\boldsymbol{\theta}) = F(Q,\boldsymbol{\theta}) - \sum_{i=1}^{m}\sum_{y=1}^{k} Q_{i,y}\log(Q_{i,y}) ~.$$

- Lemma: The EM procedure can be rewritten as:

$$\begin{aligned} Q^{(t+1)} &= \operatorname*{argmax}_{Q\in[0,1]^{m,k}:\forall i, \sum_y Q_{i,j}=1} G(Q,\boldsymbol{\theta}^{(t)}) \\ \boldsymbol{\theta}^{(t+1)} &= \operatorname*{argmax}_{\boldsymbol{\theta}} G(Q^{(t+1)},\boldsymbol{\theta}) ~. \end{aligned}$$

Furthermore, $G(Q^{(t+1)},\boldsymbol{\theta}^{(t)}) = L(S;\boldsymbol{\theta}^{(t)})$.

- Corollary: $L(S;\boldsymbol{\theta}^{t+1}) \ge L(S;\boldsymbol{\theta}^{(t)})$

EM for Mixture of Gaussians (soft k-means)

- Mixture of k Gaussians in which $\boldsymbol{\theta} = (\mathbf{c}, \{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k\})$

EM for Mixture of Gaussians (soft k-means)

- Mixture of k Gaussians in which $\boldsymbol{\theta} = (\mathbf{c}, \{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k\})$
- **E**xpectation step:

$$Q_{i,y}^{(t)} \propto \; c_y^{(t)} \; \exp\left(-\frac{1}{2}\|\mathbf{x}_i - \boldsymbol{\mu}_y^{(t)}\|^2\right)$$

EM for Mixture of Gaussians (soft k-means)

- Mixture of k Gaussians in which $\boldsymbol{\theta} = (\mathbf{c}, \{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k\})$
- **E**xpectation step:

$$Q_{i,y}^{(t)} \propto \; c_y^{(t)} \; \exp\left(-\frac{1}{2}\|\mathbf{x}_i - \boldsymbol{\mu}_y^{(t)}\|^2\right)$$

- **M**aximization step:

$$\boldsymbol{\mu}_y^{(t+1)} \;\propto\; \sum_{i=1}^{m} Q_{i,y}^{(t)}\,\mathbf{x}_i \quad \text{and} \quad c_y^{(t+1)} \;\propto\; \sum_{i=1}^{m} Q_{i,y}^{(t)}$$

Outline

Bayesian Reasoning

- So far, we treated θ as an unknown parameter

Bayesian Reasoning

- So far, we treated θ as an unknown parameter
- Bayesians treat uncertainty as randomness

Bayesian Reasoning

- So far, we treated θ as an unknown parameter
- Bayesians treat uncertainty as randomness
- Formally, think on θ as a random variable with prior probability $P[\theta]$

Bayesian Reasoning

- So far, we treated θ as an unknown parameter
- Bayesians treat uncertainty as randomness
- Formally, think on θ as a random variable with prior probability $P[\theta]$
- The probability of X given S is

$$\mathcal{P}[X = x|S] = \sum_{\theta} \mathcal{P}[X = x|\theta, S]\,\mathcal{P}[\theta|S] = \sum_{\theta} \mathcal{P}[X = x|\theta]\,\mathcal{P}[\theta|S]$$

Bayesian Reasoning

- So far, we treated θ as an unknown parameter
- Bayesians treat uncertainty as randomness
- Formally, think on θ as a random variable with prior probability $P[\theta]$
- The probability of X given S is

$$\mathcal{P}[X = x|S] = \sum_{\theta} \mathcal{P}[X = x|\theta, S]\, \mathcal{P}[\theta|S] = \sum_{\theta} \mathcal{P}[X = x|\theta]\, \mathcal{P}[\theta|S]$$

- Using Bayes rule we obtain a posterior distribution

$$\mathcal{P}[\theta|S] = \frac{\mathcal{P}[S|\theta]\, \mathcal{P}[\theta]}{\mathcal{P}[S]}$$

Bayesian Reasoning

- So far, we treated θ as an unknown parameter
- Bayesians treat uncertainty as randomness
- Formally, think on θ as a random variable with prior probability $P[\theta]$
- The probability of X given S is

$$\mathcal{P}[X = x|S] = \sum_{\theta} \mathcal{P}[X = x|\theta, S] \, \mathcal{P}[\theta|S] = \sum_{\theta} \mathcal{P}[X = x|\theta] \, \mathcal{P}[\theta|S]$$

- Using Bayes rule we obtain a posterior distribution

$$\mathcal{P}[\theta|S] = \frac{\mathcal{P}[S|\theta] \, \mathcal{P}[\theta]}{\mathcal{P}[S]}$$

- Therefore,

$$\mathcal{P}[X = x|S] = \frac{1}{\mathcal{P}[S]} \sum_{\theta} \mathcal{P}[X = x|\theta] \prod_{i=1}^{m} \mathcal{P}[X = x_i|\theta] \, \mathcal{P}[\theta] \; .$$

Bayesian Reasoning

- Example: suppose $\mathcal{X} = \{0, 1\}$ and the prior on θ is uniform over $[0, 1]$

Bayesian Reasoning

- Example: suppose $\mathcal{X} = \{0,1\}$ and the prior on θ is uniform over $[0,1]$
- Then:

$$\mathcal{P}[X = x|S] \propto \int \theta^{x+\sum_i x_i} (1-\theta)^{1-x+\sum_i (1-x_i)} \, d\theta \ .$$

Bayesian Reasoning

- Example: suppose $\mathcal{X} = \{0,1\}$ and the prior on θ is uniform over $[0,1]$
- Then:

$$\mathcal{P}[X = x|S] \propto \int \theta^{x+\sum_i x_i}(1-\theta)^{1-x+\sum_i(1-x_i)}\, d\theta \ .$$

- Solving (using integration by parts) we obtain

$$\mathcal{P}[X = 1|S] = \frac{(\sum_i x_i) + 1}{m+2} \ .$$

Bayesian Reasoning

- Example: suppose $\mathcal{X} = \{0, 1\}$ and the prior on θ is uniform over $[0, 1]$
- Then:

$$\mathcal{P}[X = x|S] \propto \int \theta^{x+\sum_i x_i}(1-\theta)^{1-x+\sum_i (1-x_i)}\, d\theta \ .$$

- Solving (using integration by parts) we obtain

$$\mathcal{P}[X = 1|S] = \frac{(\sum_i x_i) + 1}{m + 2} \ .$$

- Recall that Maximum Likelihood in this case is $\mathcal{P}[X = 1|\hat{\theta}] = \frac{\sum_i x_i}{m}$

Bayesian Reasoning

- Example: suppose $\mathcal{X} = \{0, 1\}$ and the prior on θ is uniform over $[0, 1]$
- Then:

$$\mathcal{P}[X = x|S] \propto \int \theta^{x+\sum_i x_i}(1-\theta)^{1-x+\sum_i(1-x_i)}\, d\theta \ .$$

- Solving (using integration by parts) we obtain

$$\mathcal{P}[X = 1|S] = \frac{(\sum_i x_i) + 1}{m+2} \ .$$

- Recall that Maximum Likelihood in this case is $\mathcal{P}[X = 1|\hat{\theta}] = \frac{\sum_i x_i}{m}$
- Therefore, uniform prior is similar to maximum likelihood, except it adds "pseudoexamples" to the training set

Maximum A-Posteriori

- In many situations, it is difficult to find a closed form solution to the integral in the definition of $\mathcal{P}[X = x|S]$
- A popular approximation is to find a single θ which maximizes $\mathcal{P}[\theta|S]$
- This value is called the Maximum A-Posteriori estimator
- Once this value is found, we can calculate the probability that $X = x$ given the maximum a-posteriori estimator and independently on S.

Summary

- Generative approach: model the distribution over the data
- Parametric density estimation: estimate the parameters characterizing the distribution
- Rules: Maximum Likelihood, Bayesian estimation, maximum a posteriori.
- Algorithms: Naive Bayes, LDA, EM