Learning as Inference

DD2421

Giampiero Salvi

HT2017

Outline

Probabilistic Classification and Regression

- In both cases estimate posterior

$$P(y \mid x) = \frac{P(x \mid y)P(y)}{P(x)}$$

- Classification: y is discrete
- Regression: y is continuous

Until now we assumed we knew:

- $P(y) \leftarrow$ *Prior*
- $P(x \mid y) \leftarrow$ *Likelihood*
- $P(x) \leftarrow$ *Evidence*

How can we obtain this information from observations (data)?

Learning as Inference

Given:

- the training data $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_N, y_N)\}$
- a new observation $\mathbf{x}$

Estimate the posterior probability of the answer y:

$$P(y|\mathbf{x}, \mathcal{D})$$

Parametric vs Non-parametric Inference

Parametric:

- First make the model parameters explicit: $P(y|x) = P(y|x, \theta)$
- estimate the optimal parameter $\hat{\theta}$ using the data
- compute the posterior $P(y|x, \hat{\theta})$

Learning corresponds to finding $\hat{\theta}$

Non-Parametric:

- Use a parametric model as before: $P(y|x) = P(y|x, \theta)$
- but estimate the posterior of the parameter given the data: $P(\theta|\mathcal{D})$
- Compute the posterior $P(y|x, \mathcal{D})$ by marginalizing out the parameter θ

Three Approaches

Parametric:

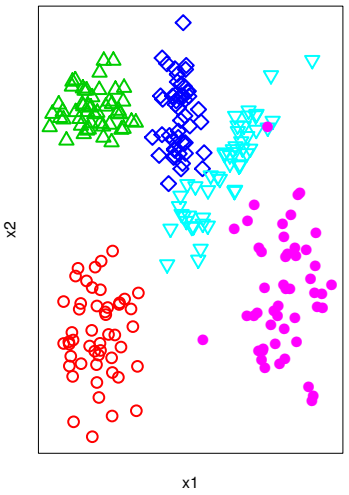
- Maximum Likelihood (ML)
- Maximum A Posteriori (MAP)

Non-parametric:

- Bayesian methods

Example: Classification

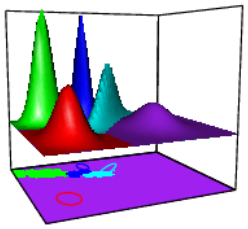
Classification



features: $\mathbf{x} \in \mathbb{R}^d$

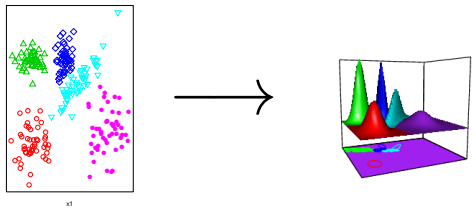
class: $y \in \{y_1, \ldots, y_K\}$

$$k_{\text{MAP}} = \arg\max_k P(y_k|\mathbf{x})$$
$$= \arg\max_k P(y_k)P(\mathbf{x}|y_k)$$



Example: Classification

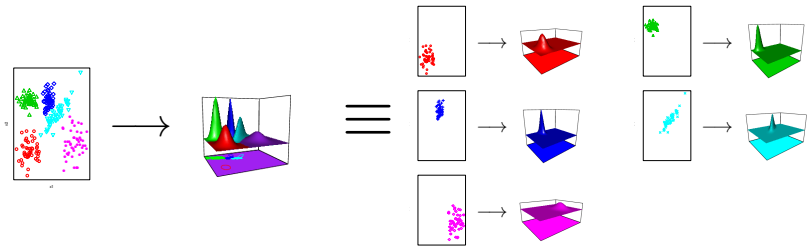
Assumption # 1: Class Independence



Assumptions:

- samples from class i do not influence estimate for class j, $i \neq j$
- Generative vs discriminative models

Class Independence Assumption



- each distribution is a likelihood in the form $P(\mathbf{x}|y_i, \mathcal{D})$
- where $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ with fixed y_i for class i
- in the following we drop the class index and talk about $P(\mathbf{x}|y, \mathcal{D})$

Assumption #2: i.i.d.

Samples from each class are independent and identically distributed:

$$\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$$

The likelihood of the whole data set can be factorized:

$$P(\mathcal{D}|y) = P(\mathbf{x}_1, \ldots, \mathbf{x}_N|y) = \prod_{i=1}^{N} P(\mathbf{x}_i|y)$$

And the log-likelihood becomes:

$$\log P(\mathcal{D}|y) = \sum_{i=1}^{N} \log P(\mathbf{x}_i|y)$$

Maximum Likelihood Estimate

- define parametric form for the likelihood distributions:

$$P(\mathbf{x}|y) \equiv P(\mathbf{x}|y, \theta)$$

- find optimal value for the parameter θ_{ML} by maximizing the likelihood of the data:

$$\theta_{\text{ML}} = \arg\max_{\theta} P(\mathcal{D}|y, \theta)$$

- approximate the likelihood given the data with this distribution:

$$P(\mathbf{x}|y, \mathcal{D}) \approx P(\mathbf{x}|y, \theta_{\text{ML}})$$

MLE Example: Discrete Variables

Will I play tennis dependent on the weather?

$$\begin{aligned} x &\in \{\text{sunny}, \text{overcast}, \text{rainy}\} \\ y &\in \{\text{yes}, \text{no}\} \end{aligned}$$

$$\begin{aligned} x &\sim \text{Cat}(\lambda_1, \ldots, \lambda_k) \\ y &\sim \text{Bernoulli}(\alpha) \\ x|y &\sim \text{Cat}(\lambda'_1, \ldots, \lambda'_k) \\ y|x &\sim \text{Bernoulli}(\alpha') \end{aligned}$$

Training data

i example	x_i outlook	y_i play	i example	x_i outlook	y_i play
1	sunny	no	8	sunny	no
2	sunny	no	9	sunny	yes
3	overcast	yes	10	rainy	yes
4	rainy	yes	11	sunny	yes
5	rainy	yes	12	overcast	yes
6	rainy	no	13	overcast	yes
7	overcast	yes	14	rainy	no

MLE: Bernoulli

$$p(y) = \begin{cases} \alpha & \text{if } y = \text{yes} \\ 1-\alpha & \text{if } y = \text{no} \end{cases}$$

1. compute (log) likelihood of the data $P(\mathcal{D}|\alpha)$
2. find α_{ML} that optimizes $P(\mathcal{D}|\alpha)$

i example	x_i outlook	y_i play	i example	x_i outlook	y_i play
1	sunny	no	8	sunny	no
2	sunny	no	9	sunny	yes
3	overcast	yes	10	rainy	yes
4	rainy	yes	11	sunny	yes
5	rainy	yes	12	overcast	yes
6	rainy	no	13	overcast	yes
7	overcast	yes	14	rainy	no

MLE: Bernoulli

$$p(y) = \begin{cases} \alpha & \text{if } y = \text{yes} \\ 1-\alpha & \text{if } y = \text{no} \end{cases}$$

Likelihood of the data
(n=number of yes in $\mathcal{D}$, N=number of examples):

$$\begin{aligned} P(\mathcal{D}|\alpha) &= \prod_i P(y_i|\alpha) = \prod_{i \text{ s.t. } y=\text{yes}} \alpha \prod_{i \text{ s.t. } y=\text{no}} (1-\alpha) \\ &= \alpha^n (1-\alpha)^{N-n} \\ \log P(\mathcal{D}|\alpha) &= n \log \alpha + (N-n) \log(1-\alpha) \\ \frac{d}{d\alpha} \log P(\mathcal{D}|\alpha) &= \frac{n - N\alpha}{\alpha(1-\alpha)} = 0 \iff \alpha_{\text{ML}} = \frac{n}{N} \end{aligned}$$

MLE Example: Discrete Variables

Will I play tennis dependent on the weather?

$$\begin{aligned} x &\in \{\text{sunny}, \text{overcast}, \text{rainy}\} \\ y &\in \{\text{yes}, \text{no}\} \end{aligned}$$

$$\begin{aligned} y &\sim \text{Bernoulli}(\alpha) \\ \alpha_{\text{ML}} &= \frac{9}{14} \end{aligned}$$

Training data

i example	x_i outlook	y_i play	i example	x_i outlook	y_i play
1	sunny	no	8	sunny	no
2	sunny	no	9	sunny	yes
3	overcast	yes	10	rainy	yes
4	rainy	yes	11	sunny	yes
5	rainy	yes	12	overcast	yes
6	rainy	no	13	overcast	yes
7	overcast	yes	14	rainy	no

MLE: Categorical

Similar derivation:

$$\lambda_{k,\text{ML}} = \frac{n_k}{N}$$

where n_k is the number of examples of the kth

$$\begin{aligned} x &\sim \text{Cat}(\lambda_1, \ldots, \lambda_k) \\ \lambda_{\text{ML}} &= \{\frac{5}{14}, \frac{4}{14}, \frac{5}{14}\} \\ x|y &\sim \text{Cat}(\lambda_1, \ldots, \lambda_k) \\ \lambda'_{\text{ML}}(\text{yes}) &= \{\frac{2}{9}, \frac{4}{9}, \frac{3}{9}\} \\ \lambda'_{\text{ML}}(\text{no}) &= \{\frac{3}{5}, 0, \frac{2}{5}\} \end{aligned}$$

Training data

i example	x_i outlook	y_i play	i example	x_i outlook	y_i play
1	sunny	no	8	sunny	no
2	sunny	no	9	sunny	yes
3	overcast	yes	10	rainy	yes
4	rainy	yes	11	sunny	yes
5	rainy	yes	12	overcast	yes
6	rainy	no	13	overcast	yes
7	overcast	yes	14	rainy	no

But..., will I play tennis?

Let's say it is rainy:

$$P(y = \text{yes}|\text{outlook=rainy}) = \frac{P(\text{outlook=rainy}|y = \text{yes})P(y = \text{yes})}{P(\text{outlook=rainy})} = \frac{\frac{3}{9}\frac{9}{14}}{\frac{5}{14}} = \frac{3}{5}$$

$$P(y = \text{no}|\text{outlook=rainy}) = \frac{P(\text{outlook=rainy}|y = \text{no})P(y = \text{no})}{P(\text{outlook=rainy})} = \frac{\frac{2}{5}\frac{5}{14}}{\frac{5}{14}} = \frac{2}{5}$$

Then

$$y_{\text{MAP}} = \arg\max_y P(y|\text{outlook=rainy}) = \text{yes}$$

$$y_{\text{ML}} = \arg\max_y P(\text{outlook=rainy}|y) = \text{no}$$

Source of confusion

We did Maximum a Posteriori (MAP) and Maximum Likelihood (ML) classification

$$y_{\text{MAP}} = \arg\max_y P(y|x, \theta_{\text{ML}})$$

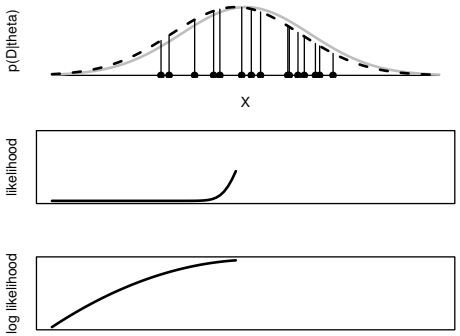
$$y_{\text{ML}} = \arg\max_y P(x|y, \theta_{\text{ML}})$$

with parameters θ estimated by Maximum Likelihood (ML):

$$\theta_{\text{ML}} = \arg\max_\theta P(D|y, \theta) = \arg\max_\theta \prod_i P(x_i|y_i, \theta)$$

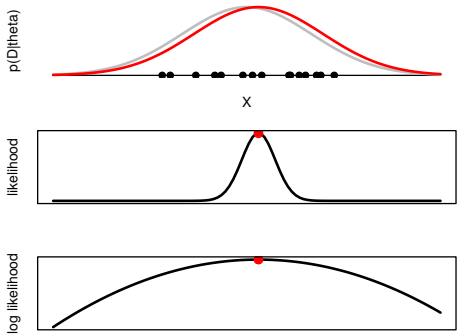
MLE Illustration: continuous variables

Find parameter vector θ_{ML} that maximizes $P(\mathcal{D}|\theta)$ with $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$



MLE Illustration: continuous variables

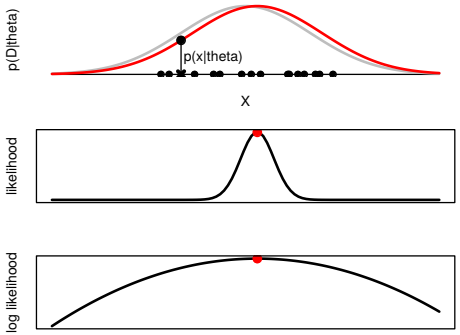
Find parameter vector θ_{ML} that maximizes $P(\mathcal{D}|\theta)$ with $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$



1. estimate the optimal parameters of the model

MLE Illustration: continuous variables

Find parameter vector θ_{ML} that maximizes $P(\mathcal{D}|\theta)$ with $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$



1. estimate the optimal parameters of the model
2. evaluate the predictive distribution on new data points

ML estimation of Gaussian parameters

$$\mu_{\mathsf{ML}} = \frac{1}{N}\sum_{i=1}^{N} x_i$$

$$\sigma^2_{\mathsf{ML}} = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu_{\mathsf{ML}})^2$$

- same result by minimizing the sum of square errors!
- but we make assumptions explicit

ML estimation of Gaussian mean

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

Log-likelihood of data (i.i.d. samples):

$$\log P(\mathcal{D}|\theta) = \sum_{i=1}^{N}\log\mathcal{N}(x_i|\mu,\sigma^2) = -N\log\left(\sqrt{2\pi\sigma^2}\right) - \sum_{i=1}^{N}\frac{(x_i-\mu)^2}{2\sigma^2}$$

$$0 = \frac{d\log P(\mathcal{D}|\theta)}{d\mu} = \sum_{i=1}^{N}\frac{(x_i-\mu)}{\sigma^2} = \frac{\sum_{i=1}^{N} x_i - N\mu}{\sigma^2} \iff$$

$$\mu_{\mathsf{ML}} = \frac{1}{N}\sum_{i=1}^{N} x_i$$

Problem: Curse of Dimensionality

i	$\mathbf{x}_i$				y_i
example	outlook	temperature	humidity	windy	play
1	sunny	hot	high	false	no
2	sunny	hot	high	true	no
3	overcast	hot	high	false	yes
4	rainy	mild	high	false	yes
5	rainy	cool	normal	false	yes
6	rainy	cool	normal	true	no
7	overcast	cool	normal	true	yes
8	sunny	mild	high	false	no
9	sunny	cool	normal	false	yes
10	rainy	mild	normal	false	yes
11	sunny	mild	normal	true	yes
12	overcast	mild	high	true	yes
13	overcast	hot	normal	false	yes
14	rainy	mild	high	true	no

difficult to model $P(\text{outlook}, \text{temperature}, \text{humidity}, \text{windy}|\text{play})$

Problem: Curse of Dimensionality

- Size of feature space exponential in number of features.
- More features $\Longrightarrow$ potential for better description of the objects but...
- More features $\Longrightarrow$ more difficult to model $P(\mathbf{x} \mid y)$.

Extreme Solution: Naïve Bayes classifier

- All features (dimensions) regarded as conditionally independent.
- Model D one-dimensional distributions instead of one D-dimensional distribution.

$P(\text{outlook}|\text{play}), P(\text{temperature}|\text{play}), P(\text{humidity}|\text{play}), P(\text{windy}|\text{play})$

Naïve Bayes Classifier

- $\mathbf{x}$ is a vector $(x_1, \ldots, x_D)$ of attribute or feature values.
- Let $\mathcal{Y} = \{1, 2, \ldots, Y\}$ be the set of possible classes.
- The MAP estimate of y is

$$y_{\text{MAP}} = \arg\max_{y \in \mathcal{Y}} P(y \mid x_1, \ldots, x_D) = \arg\max_{y \in \mathcal{Y}} \frac{P(x_1, \ldots, x_D \mid y)\, P(y)}{P(x_1, \ldots, x_D)}$$
$$= \arg\max_{y \in \mathcal{Y}} P(x_1, \ldots, x_D \mid y)\, P(y)$$

- Naïve Bayes assumption: $P(x_1, \ldots, x_D \mid y) = \prod_{d=1}^{D} P(x_d \mid y)$
- Naïve Bayes classifier:

$$y_{\text{MAP}} = \arg\max_{y \in \mathcal{Y}} P(y) \prod_{d=1}^{D} P(x_d \mid y)$$

Naïve Bayes Classifier

- One of the most common learning methods.

When to use:

- Moderate or large training set available.
- Features x_i of a data instance $\mathbf{x}$ are conditionally independent given classification (or at least reasonably independent, still works with a little dependence).

Successful applications:

- Medical diagnoses (symptoms independent)
- Classification of text documents (words independent)
- Acoustic modelling in Automatic Speech Recognition

Example: Play Tennis?

Question: Will I go and play tennis given the forecast?

My measurements:

- **outlook** $\in$ {sunny, overcast, rainy},
- **temperature** $\in$ {hot, mild, cool},
- **humidity** $\in$ {high, normal},
- **windy** $\in$ {false, true}.

Possible decisions: $y \in$ {yes, no}

Example: Play Tennis?

What I did in the past:

i	$\mathbf{x}_i$				y_i
example	outlook	temperature	humidity	windy	play
1	sunny	hot	high	false	no
2	sunny	hot	high	true	no
3	overcast	hot	high	false	yes
4	rainy	mild	high	false	yes
5	rainy	cool	normal	false	yes
6	rainy	cool	normal	true	no
7	overcast	cool	normal	true	yes
8	sunny	mild	high	false	no
9	sunny	cool	normal	false	yes
10	rainy	mild	normal	false	yes
11	sunny	mild	normal	true	yes
12	overcast	mild	high	true	yes
13	overcast	hot	normal	false	yes
14	rainy	mild	high	true	no

Example: Play Tennis?

Counts of when I played tennis (did not play)

Outlook			Temperature			Humidity		Windy	
sunny	overcast	rain	hot	mild	cool	high	normal	false	true
2 (3)	4 (0)	3 (2)	2 (2)	4 (2)	3 (1)	3 (4)	6 (1)	6 (2)	3 (3)

Prior of whether I played tennis or not

Counts:

Play	
yes	no
9	5

Prior Probabilities:

Play	
yes	no
$\frac{9}{14}$	$\frac{5}{14}$

Likelihood of attribute when tennis played $P(x_i \mid \text{y=yes})(P(x_i \mid \text{y=no}))$

Outlook			Temperature			Humidity		Windy	
sunny	overcast	rain	hot	mild	cool	high	normal	false	true
$\frac{2}{9}$ ($\frac{3}{5}$)	$\frac{4}{9}$ ($\frac{0}{5}$)	$\frac{3}{9}$ ($\frac{2}{5}$)	$\frac{2}{9}$ ($\frac{2}{5}$)	$\frac{4}{9}$ ($\frac{2}{5}$)	$\frac{3}{9}$ ($\frac{1}{5}$)	$\frac{3}{9}$ ($\frac{4}{5}$)	$\frac{6}{9}$ ($\frac{1}{5}$)	$\frac{6}{9}$ ($\frac{2}{5}$)	$\frac{3}{9}$ ($\frac{3}{5}$)

Example: Play Tennis?

Inference: Use the learnt model to classify a new instance.

New instance:

$$\mathbf{x} = (\text{sunny, cool, high, true})$$

Apply Naïve Bayes Classifier:

$$y_{\text{MAP}} = \arg \max_{y \in \{\text{yes, no}\}} P(y) \prod_{i=1}^{4} P(x_i \mid y)$$

$$P(\text{yes})\, P(\text{sunny} \mid \text{yes})\, P(\text{cool} \mid \text{yes})\, P(\text{high} \mid \text{yes})\, P(\text{true} \mid \text{yes}) = \frac{9}{14} \times \frac{2}{9} \times \frac{3}{9} \times \frac{3}{9} \times \frac{3}{9} = .005$$

$$P(\text{no})\, P(\text{sunny} \mid \text{no})\, P(\text{cool} \mid \text{no})\, P(\text{high} \mid \text{no})\, P(\text{true} \mid \text{no}) = \frac{5}{14} \times \frac{3}{5} \times \frac{1}{5} \times \frac{4}{5} \times \frac{3}{5} = .021$$

$\Longrightarrow y_{\text{MAP}} = \text{no}$

Naïve Bayes: Independence Violation

- Conditional independence assumption:

$$P(x_1, x_2, \ldots, x_D \mid y) = \prod_{d=1}^{D} P(x_d \mid y)$$

  often violated - but it works surprisingly well anyway!
- **Note:** Do not need the posterior probabilities $P(y \mid \mathbf{x})$ to be correct. Only need y_{MAP} to be correct.
- Since dependencies ignored, naïve Bayes posteriors often unrealistically close to 0 or 1.
  Different attributes say the same thing to a higher degree than we expect as they are correlated in reality.

Naïve Bayes: Estimating Probabilities

- **Problem:** What if none of the training instances with target value y have attribute x_i? Then

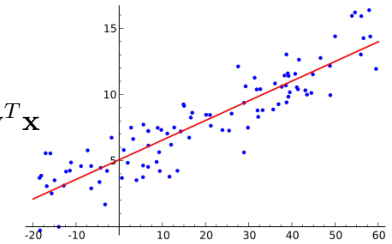
$$P(x_i \mid y) = 0 \quad \Longrightarrow \quad P(y)\prod_{i=1}^{D} P(x_i \mid y) = 0$$

- **Simple solution:** add pseudocounts to all counts so that no count is zero
- This is a form of regularization or smoothing

Classical Linear Regression

Model (deterministic):

$$\begin{aligned}\hat{y} &= w_0 + w_1 x_1 + \cdots + w_d x_d \\ &= \begin{bmatrix} w_0 & w_1 & \dots & w_d \end{bmatrix} \begin{bmatrix} 1 \\ x_1 \\ \vdots \\ x_d \end{bmatrix} = \mathbf{w}^T\mathbf{x}\end{aligned}$$



Minimize sum of square errors

$$\mathbf{w}_{\text{opt}} = \arg\min_{\mathbf{w}} \sum_{i=1}^{N} (y_i - \hat{y})^2 = \arg\min_{\mathbf{w}} \sum_{i=1}^{N} (y_i - \mathbf{w}^T\mathbf{x}_i)^2$$

Probabilistic Linear Regression

Model (deterministic):

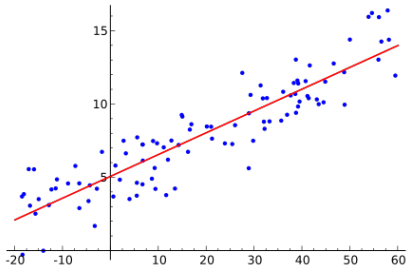
$$y = \mathbf{w}^T\mathbf{x} + \epsilon$$

But now:

$$\epsilon \sim \mathcal{N}(0, \sigma^2)$$

Therefore:

$$\begin{aligned} y &\sim \mathcal{N}(\mu_Y(\mathbf{x}), \sigma_Y^2(\mathbf{x})) \\ &= \mathcal{N}(\mathbf{w}^T\mathbf{x}, \sigma^2)\end{aligned}$$



Learning: find $\mathbf{w}$ that maximizes $P(Y|X, \mathbf{w}, \sigma^2)$

Maximize the posterior directly $\Longrightarrow$ discriminative method

Probabilistic Linear Regression: MLE

$$\begin{aligned}\log P(Y|X, \mathbf{w}, \sigma^2) &= \log \prod_i P(y_i|\mathbf{x}_i, \mathbf{w}, \sigma^2) \\ &= \sum_i \log P(y_i|\mathbf{x}_i, \mathbf{w}, \sigma^2) \\ &= \sum_i \log \left[\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i - \mathbf{w}^T\mathbf{x}_i)^2}{2\sigma^2}} \right] \\ &= \sum_i \left[-\frac{1}{2}\log(2\pi\sigma^2) - \frac{(y_i - \mathbf{w}^T\mathbf{x}_i)^2}{2\sigma^2} \right]\end{aligned}$$

$$\arg\max_{\mathbf{w}} \left[P(Y|X, \mathbf{w}, \sigma^2) \right] = \arg\min_{\mathbf{w}} \sum_i (y_i - \mathbf{w}^T\mathbf{x}_i)^2$$

Maximizing $P(Y|X, \mathbf{w}, \sigma^2)$ equivalent to minimizing sum of squares!

Logistic Regression

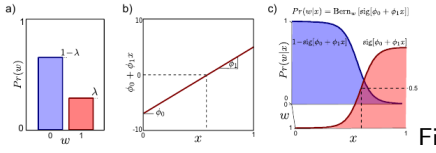


Figure from Prince

- binary classification problem: $y \in \{0, 1\}$
- treat as regression problem: $y \rightarrow \lambda$ (Bernoulli parameter)

$$
\begin{aligned}
y &\sim \text{Bernoulli}(\lambda) = \lambda^y (1-\lambda)^{(1-y)} \\
\lambda &= \lambda(\mathbf{x}) = \text{sig}(\mathbf{w}^T \mathbf{x}) = \frac{1}{1+e^{-\mathbf{w}^T \mathbf{x}}}
\end{aligned}
$$

Learning: maximize $P(y|\lambda(\mathbf{x}))$
Discriminative method

Logistic Regression: Properties

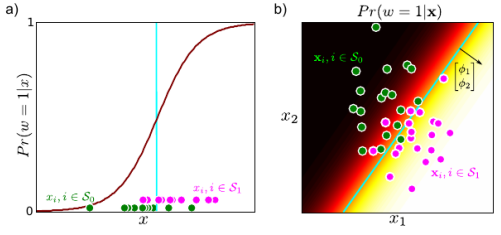


Figure from Prince

- binary classification
- discriminative learning
- only capable of linear discrimination

Logistic Regression: MLE

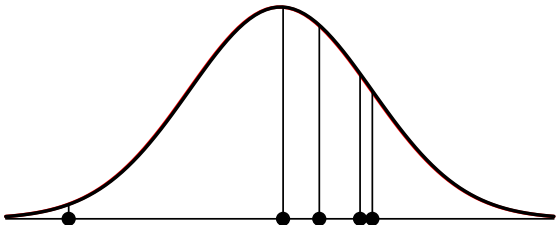
$$
\begin{aligned}
P(y|X, \mathbf{w}) &= \prod_{i=1}^{N} \lambda^{y_i} (1-\lambda)^{(1-y_i)} \\
\log P(y|X, \mathbf{w}) &= \sum_{i=1}^{N} [y_i \log \lambda + (1-y_i) \log(1-\lambda)] \\
&= \sum_{i=1}^{N} y_i \log \frac{1}{1+e^{-\mathbf{w}^T x_i}} + \sum_{i=1}^{N} (1-y_i) \log \frac{e^{-\mathbf{w}^T x_i}}{1+e^{-\mathbf{w}^T x_i}}
\end{aligned}
$$

Note: $\frac{d}{d\mathbf{w}} \log P(y|X, \mathbf{w}) = 0$ has no *closed form* solution

Use approximate method instead (Newton, gradient descent)

Problem: few data points

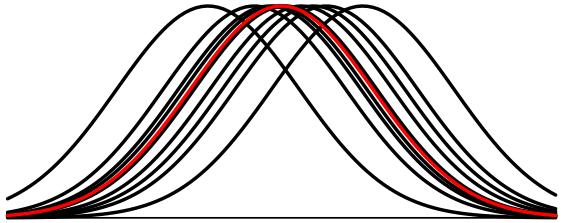
10 repetitions with 5 points each



X

Problem: few data points

10 repetitions with 5 points each



X

Maximum a Posteriori Estimation

$$
\begin{aligned}
(\mu, \sigma^2)_{\text{MAP}} &= \arg\max_{\mu,\sigma^2} P(\mu, \sigma^2|\mathcal{D}) \\
&= \arg\max_{\mu,\sigma^2} \frac{P(\mathcal{D}|\mu, \sigma^2)P(\mu, \sigma^2)}{P(\mathcal{D})} \\
&= \arg\max_{\mu,\sigma^2} P(\mathcal{D}|\mu, \sigma^2)P(\mu, \sigma^2) \\
&= \arg\max_{\mu,\sigma^2} \left[\prod_{i=1}^{N} P(x_i|\mu, \sigma^2)P(\mu, \sigma^2)\right] \\
&= \arg\max_{\mu,\sigma^2} \sum_{i=1}^{N} \log\left[P(x_i|\mu, \sigma^2)P(\mu, \sigma^2)\right]
\end{aligned}
$$

where the prior $P(\mu, \sigma^2)$ needs a specific mathematical form for closed solution

Conjugate Prior

Definition:

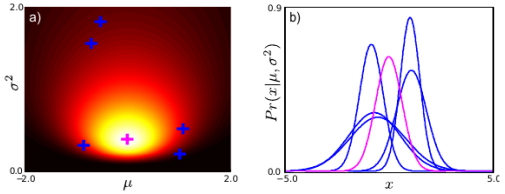
if posterior and prior in the same family of functions

Examples:

Likelihood	Conjugate prior
Bernoulli	Beta
Binomial	Beta
Categorical	Dirichlet
Normal	Normal
Normal	Normal-inverse Gamma

Example: Normal-inverse Gamma (Normal Conjugate)

$$
P(\mu, \sigma^2|\alpha, \beta, \gamma, \delta) = \frac{\sqrt{\gamma}}{\sigma\sqrt{2\pi}} \frac{\beta^\alpha}{\Gamma(\alpha)} \left(\frac{1}{\sigma^2}\right)^{\alpha+1} \exp\left[-\frac{2\beta + \gamma(\delta - \mu)^2}{2\sigma^2}\right]
$$

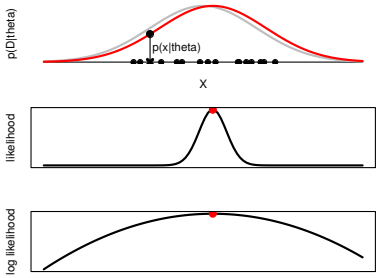


$$
\begin{aligned}
\mu_{\text{MAP}} &= \frac{N}{N+\gamma}\,\mu_{\text{ML}} + \frac{\gamma}{N+\gamma}\,\delta \\
\sigma^2_{\text{MAP}} &= \frac{N}{N+3+2\alpha}\sigma_{\text{ML}}{}^2 + \frac{2\beta + \gamma(\delta + \mu_{\text{MAP}})^2}{N+3+2\alpha}
\end{aligned}
$$

where $\alpha, \beta, \gamma, \delta$ are parameters of the prior distribution

ML, MAP and Point Estimates

- Both ML and MAP produce point estimates of θ
- Assumption: there is a true value for θ
- advantage: once $\hat{\theta}$ is found, everything is known



Bayesian estimation

- Consider θ as a random variable
- characterize θ with the posterior distribution $P(\theta|\mathcal{D})$ given the data

$$\begin{array}{lll} \text{ML:} & \mathcal{D} & \rightarrow \theta_{\text{ML}} \\ \text{MAP:} & \mathcal{D}, P(\theta) & \rightarrow \theta_{\text{MAP}} \\ \text{Bayes:} & \mathcal{D}, P(\theta) & \rightarrow P(\theta|\mathcal{D}) \end{array}$$

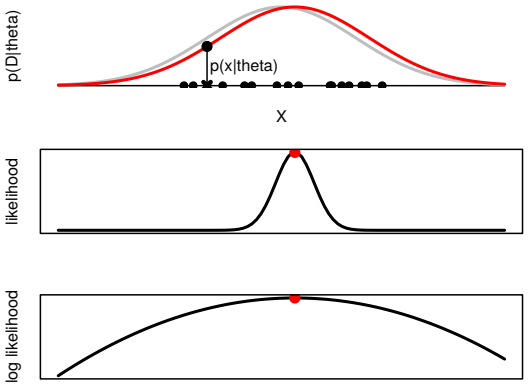
- for new data points, instead of $P(\mathbf{x}_{\text{new}}|\theta_{\text{ML}})$ or $P(\mathbf{x}_{\text{new}}|\theta_{\text{MAP}})$, compute:

$$P(\mathbf{x}_{\text{new}}|\mathcal{D}) = \int_{\theta\in\Theta} P(\mathbf{x}_{\text{new}}|\theta)P(\theta|\mathcal{D})d\theta$$

Bayesian estimation (cont.)

- we can compute $P(\mathbf{x}|\mathcal{D})$ instead of $P(\mathbf{x}|\hat{\theta})$
- integrate the joint density $P(\mathbf{x}, \theta|\mathcal{D}) = P(\mathbf{x}|\theta)P(\theta|\mathcal{D})$

$P(\mathbf{x}|\hat{\theta})$

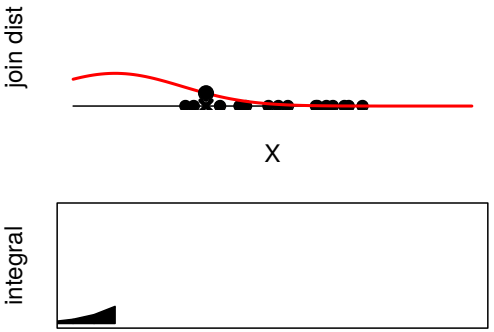


Bayesian estimation

- we can compute $P(\mathbf{x}|\mathcal{D})$ instead of $P(\mathbf{x}|\hat{\theta})$
- integrate the joint density $P(\mathbf{x}, \theta|\mathcal{D}) = P(\mathbf{x}|\theta)P(\theta|\mathcal{D})$

$P(\mathbf{x}|\mathcal{D}) =$

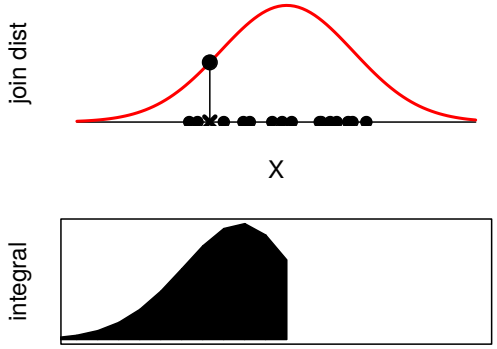
$\int P(\mathbf{x}|\theta)P(\theta|\mathcal{D})d\theta$



Bayesian estimation

- we can compute $P(\mathbf{x}|\mathcal{D})$ instead of $P(\mathbf{x}|\hat{\theta})$
- integrate the joint density $P(\mathbf{x}, \theta|\mathcal{D}) = P(\mathbf{x}|\theta)P(\theta|\mathcal{D})$

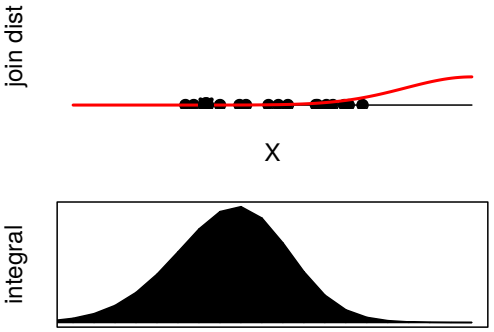
$$P(\mathbf{x}|\mathcal{D}) = \int P(\mathbf{x}|\theta)P(\theta|\mathcal{D})d\theta$$



Bayesian estimation

- we can compute $P(\mathbf{x}|\mathcal{D})$ instead of $P(\mathbf{x}|\hat{\theta})$
- integrate the joint density $P(\mathbf{x}, \theta|\mathcal{D}) = P(\mathbf{x}|\theta)P(\theta|\mathcal{D})$

$$P(\mathbf{x}|\mathcal{D}) = \int P(\mathbf{x}|\theta)P(\theta|\mathcal{D})d\theta$$



Bayesian estimation (cont.)

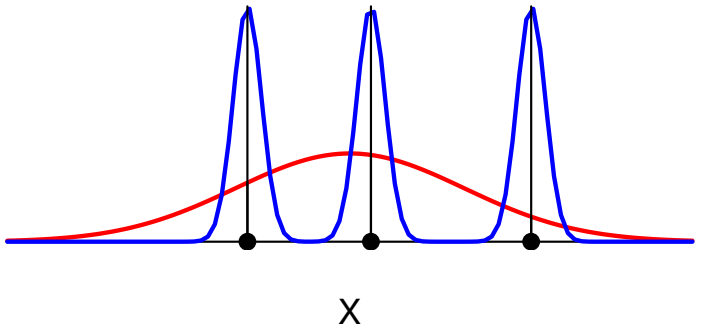
Pros:

- better use of the data
- makes a priori assumptions explicit
- can be implemented recursively (if conjugate prior)
  - use posterior $P(\theta|\mathcal{D})$ as new prior
- reduce overfitting

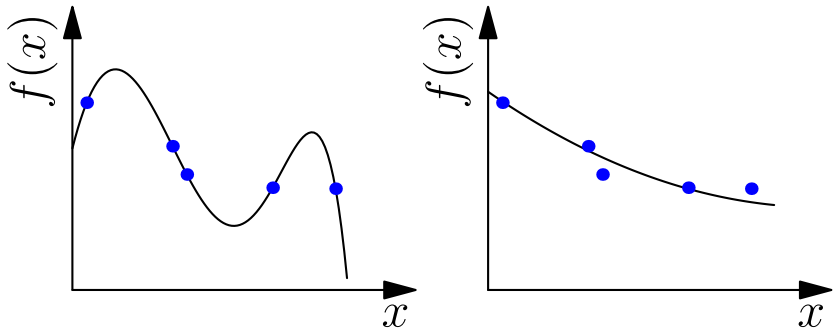
Cons:

- definition of noninformative priors can be tricky
- often requires numerical integration

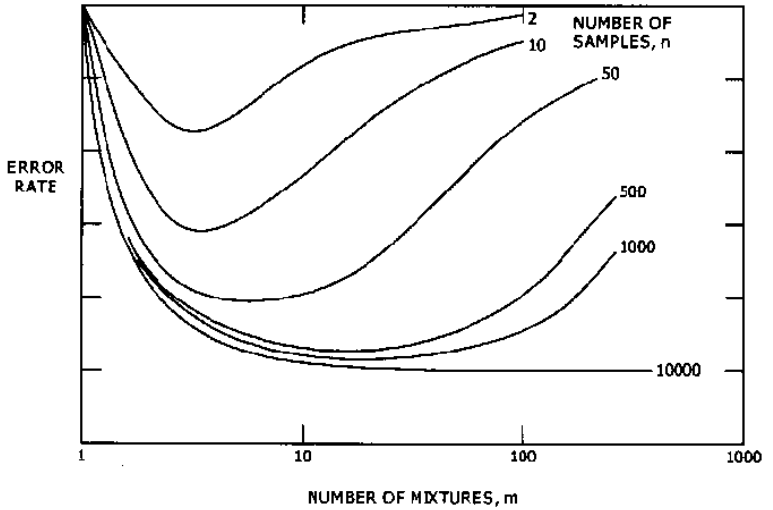
Model Selection and Overfitting



Overfitting



Overfitting: Phoneme Discrimination



Occam's Razor

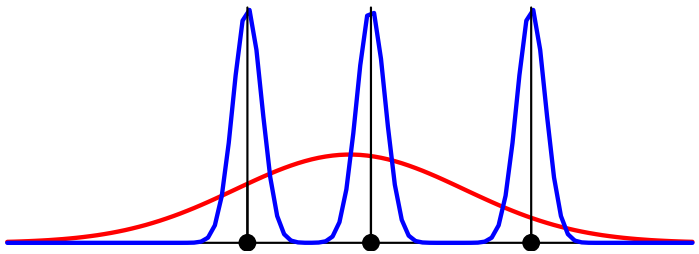
Choose the simplest explanation for the observed data

Important factors:

- number of model parameters
- number of data points
- model fit to the data

Overfitting and Maximum Likelihood

we can make the likelihood arbitrary large by increasing the number of parameters



X

Occam's Razor and Bayesian Learning

Remember that:

$$P(\mathbf{x}_{\text{new}}|\mathcal{D}) = \int_{\theta \in \Theta} P(\mathbf{x}_{\text{new}}|\theta) P(\theta|\mathcal{D}) d\theta$$

Intuition:

More complex models fit the data very well (large $P(\mathcal{D}|\theta)$) but only for small regions of the parameter space Θ.

Summary

If you are interested in learning more take a look at:

C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer Verlag 2006.