Music Machine Learning

VII – Bayesian inference

Master ATIAM - Informatique

Philippe Esling (esling@ircam.fr)

Maître de conférences – UPMC

Equipe représentations musicales (IRCAM, Paris)

Bayes' rule

Definition: $\mathcal{P}(a|b) = \dfrac{\mathcal{P}(a,b)}{\mathcal{P}(b)}$

$$\mathcal{P}(a|b)\mathcal{P}(b) = \mathcal{P}(a,b) = \mathcal{P}(b,a) = \mathcal{P}(b|a)\mathcal{P}(a)$$

Bayes' rule $\mathcal{P}(a|b) = \dfrac{\mathcal{P}(b|a)\mathcal{P}(a)}{\mathcal{P}(b)}$

a = class

b = evidence

Let's say we have a classification problem in which

Easy

Hard — $\mathcal{P}(c|e) = \dfrac{\mathcal{P}(e|c)\mathcal{P}(c)}{\mathcal{P}(e)}$ **Bayes' rule**

Bayesian classification

Now starting from Bayes' rule $\mathcal{P}(c|e) = \frac{\mathcal{P}(e|c)\mathcal{P}(c)}{\mathcal{P}(e)}$

- I have several classes, I have to decide given some evidence
- If I have the probability of the evidence given each class
- And the a priori probability of each class
- Then I'm done … **as the denominator is the same for all classes**
- But usually there is more than one piece of evidence

$$\mathcal{P}(c_i|e) = \frac{\mathcal{P}(e_1, ...e_n|c_i)\mathcal{P}(c_i)}{d}$$

- So what if these pieces of evidence are independent (given the class)

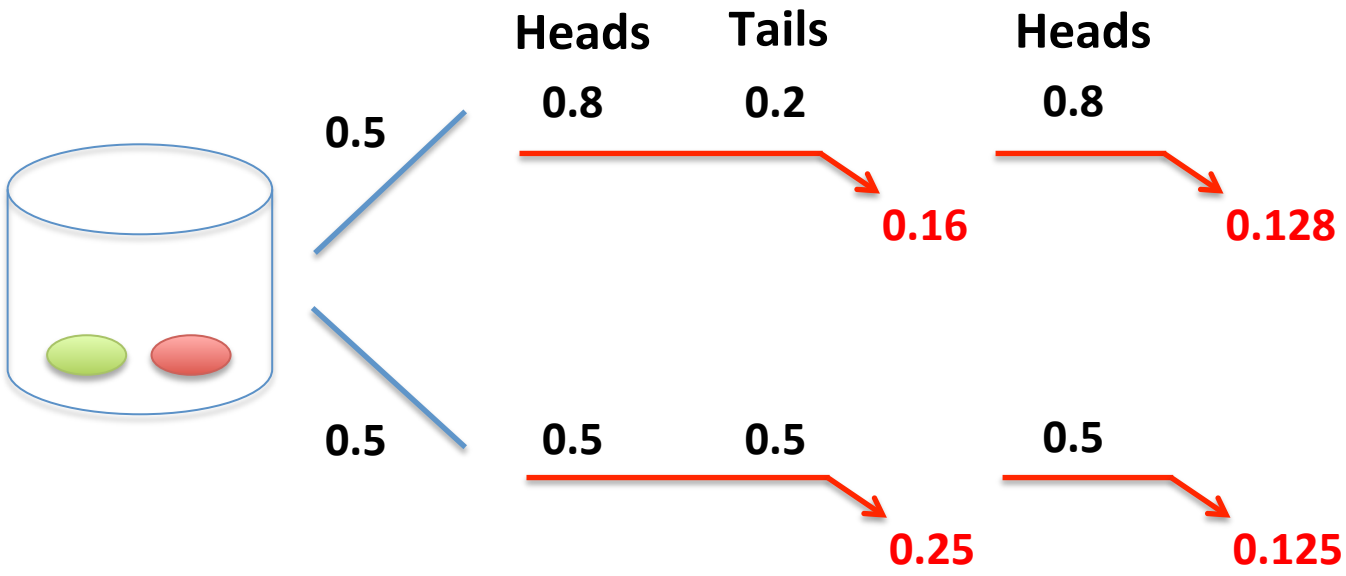
$$\mathcal{P}(c_i|e) = \frac{\mathcal{P}(e_1|c_i)\mathcal{P}(e_2|c_i)...\mathcal{P}(e_n|c_i)\mathcal{P}(c_i)}{d}$$

- So we just need to go through every class and select the biggest one

Bayesian classification

- Let's say I have two coins, one is fair and one is rigged

$$\mathcal{P}(Heads) = 0.5 \quad \mathcal{P}(Heads) = 0.8$$
$$\mathcal{P}(Tails) = 0.5 \quad \mathcal{P}(Tails) = 0.2$$



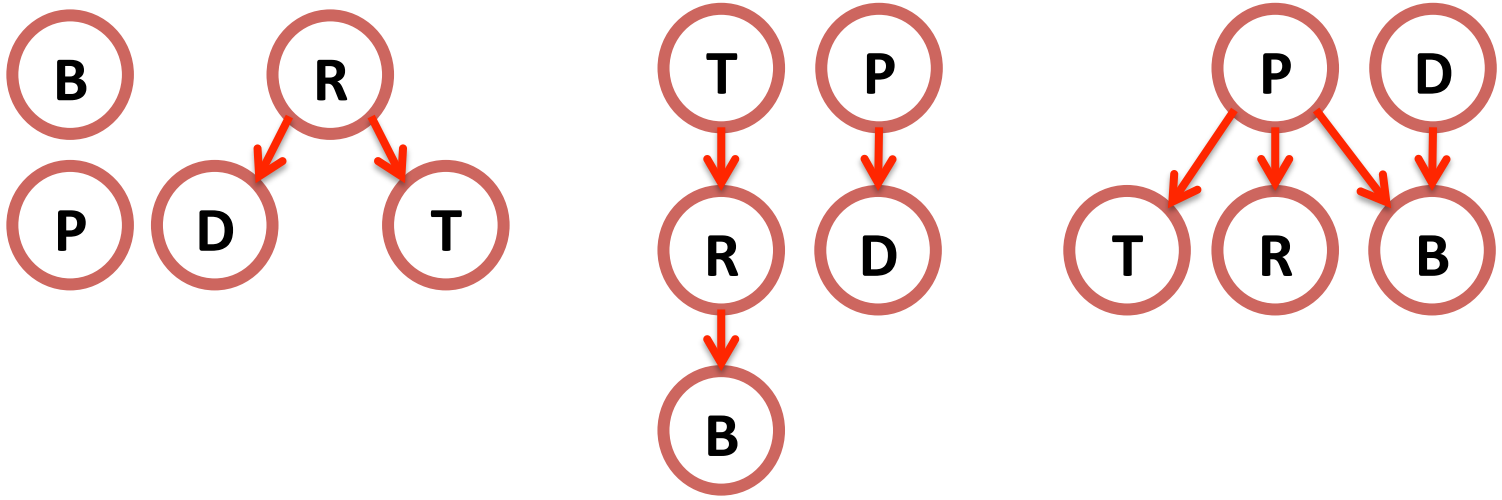
$$\mathcal{P}(c_i|e) = \frac{\mathcal{P}(e_1|c_i)\mathcal{P}(e_2|c_i)...\mathcal{P}(e_n|c_i)\mathcal{P}(c_i)}{d}$$

Structure discovery

- Remember that we wanted to select between two models
- We can use the same Bayesian hack to compute

$$\mathcal{P}(model_i|data) = \frac{\mathcal{P}(data|model_i)\mathcal{P}(model_i)}{\mathcal{P}(data)}$$

- So with Bayesian classification, we can go one step more …
- And it became **structure discovery**
- **We can generalize by performing random connexions**



Bayesian inference

Make inferences about unknown quantities using available information.

Inference -- make probability statements

Unknowns -- parameters, functions of parameters, states or latent variables, "future" outcomes, classes labels

Information – data-based / non data-based

theories of behavior; "subjective views" there is an underlying structure

parameters are finite or in some range

Bayesian inference

Likelihood function, *probability of the data given the model*

Prior probability, the probability of the model on previous knowledge

$$\mathcal{P}(model|data) = \frac{\mathcal{P}(data|model)\mathcal{P}(model)}{\mathcal{P}(data)}$$

Posterior probability, *the probability of the model (class) given the data*

Probability of the data, *normalizing constant ensuring the probabilities sum to 1*

Posterior ∝ "Likelihood" × Prior

- Likelihood contains all information relevant for inference.
- Same likelihood function = same inferences about unknowns.

Bayesian Inference **= Find the explanation with highest posterior probability**

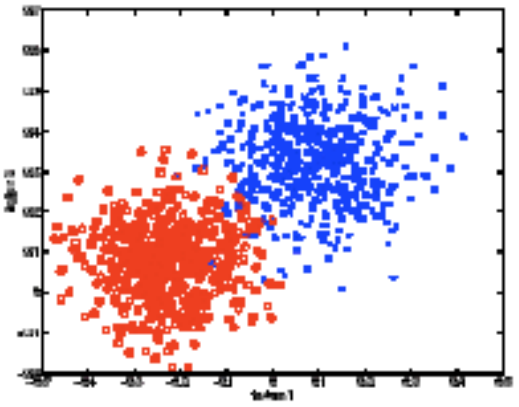
Bayesian decision theory

- We can apply this to **classification**
  - Try to find the right class in a set

$$g(x) = i, \quad i \in \{1, \ldots, M\}$$

  - We can introduce a [0,1] **loss function**

$$L[g(x), y] = \begin{cases} 1, & g(x) \neq y \\ 0, & g(x) = y \end{cases}$$



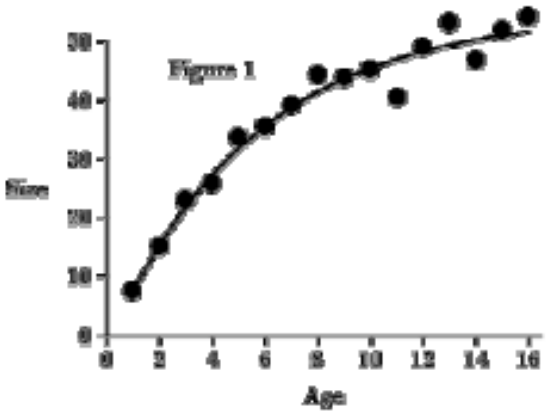
- This also applies to the **regression problem**
  - Try to predict a continuous function

$$g(x) \in \Re$$

  - Same problem if suitable loss function

$$L[g(x), y] = \|y - g(x)\|^2$$

Error function



Bayesian inference

- Goal is to represent the **belief** of learning agents
- Bayesian inference can be interpreted as

Update the prior beliefs with a new information x

Bayes' Rule:

$$Pr(\omega_i) \cdot \frac{p(x|\omega_i)}{\sum_j p(x|\omega_j) \cdot Pr(\omega_j)} = Pr(\omega_i|x)$$

Likelihood

Prior probability

'Evidence' = p(x)

Posterior probability

- Priors are usually unknown
  - Can be removed by **assuming equal belief** in all models at the start
- **Posterior is prior influenced by the likelihood according to new evidence** (information)

Bayesian inference

- How to choose the best class given the data?
- Choose the **Maximum A Posteriori (MAP)** class

$$\hat{\omega} = \underset{\omega_i}{\operatorname{argmax}}\ Pr(\omega_i|x)$$

- Intuitive: Choose the most probable class given the observation(s)
- We don't know $Pr(\omega_i|x)$
- But we know $Pr(x|\omega_i)$
- So we apply Bayes' rule

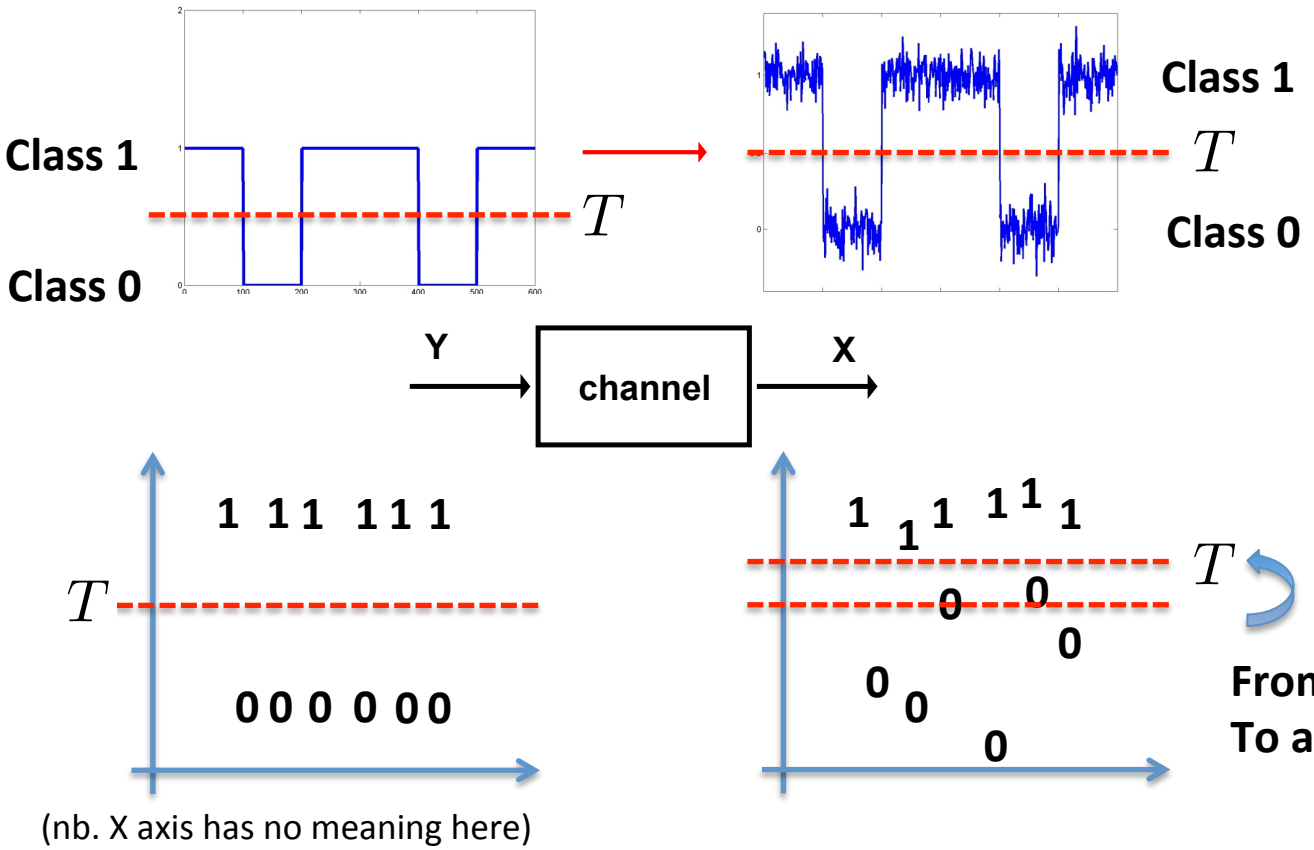
the search for $\hat{\omega} = \underset{\omega_i}{\operatorname{argmax}}\ Pr(\omega_i|x)$

becomes $\underset{\omega_i}{\operatorname{argmax}}\ \dfrac{p(x|\omega_i)\cdot Pr(\omega_i)}{\sum_j p(x|\omega_j)\cdot Pr(\omega_j)}$

but denominator = $p(x)$ **is the same over all** ω_i

hence $\hat{\omega} = \underset{\omega_i}{\operatorname{argmax}}\ p(x|\omega_i)\cdot Pr(\omega_i)$

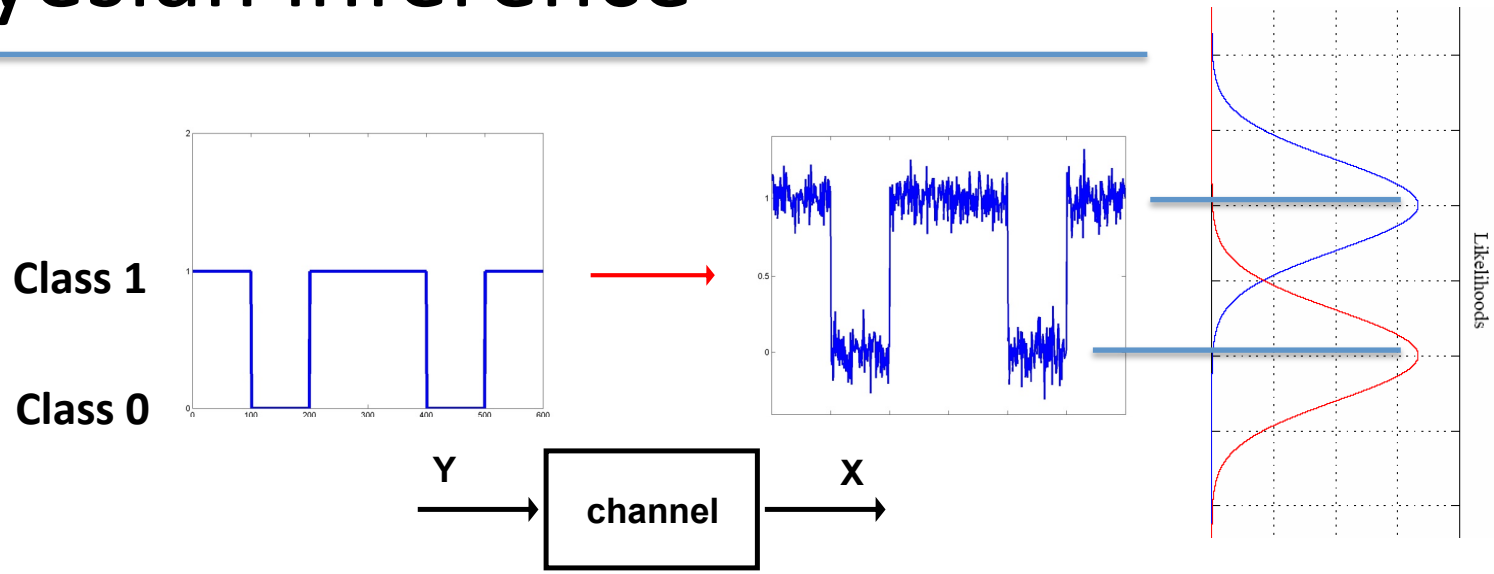
Bayesian inference



(nb. X axis has no meaning here)

Intuitively $$Y = \begin{cases} 0, & \text{if } x > T \\ 1, & \text{if } x > T \end{cases}$$

Bayesian inference



- **What do we need** to lay a Bayesian inference ?
  - (A priori) **class probabilities**

    Without other knowledge **an equal belief**

    $$P_Y(0) = P_Y(1) = 1/2$$
  - **Class-conditionnal probabilities**

    Here the class depends on **the noise added by the channel**

    By the central limit theorem we can **assume that noise is Gaussian**

$$X \sim N(\mu, \sigma)$$

Bayesian inference

- **Gaussian probability density** function (pdf) $P_X(x) = G(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$
- Since we assumed that the noise is Gaussian and **additive**

$$X = Y + \varepsilon, \qquad \varepsilon \sim N(0, \sigma^2)$$

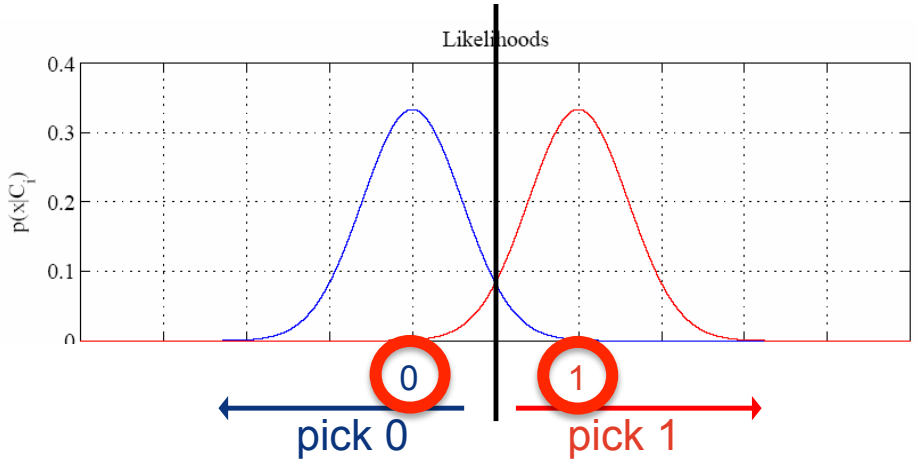
- So X corresponds to the input (Y) plus Gaussian noise

Probability of input given classes

Class 0 $P_{X|Y}(x \mid 0) = G(x, 0, \sigma)$

Class 1 $P_{X|Y}(x \mid 1) = G(x, 1, \sigma)$

$$P_Y(0) = P_Y(1) = 1/2$$



Bayesian inference

- **Gaussian probability density** function (pdf) $P_X(x) = G(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$
- Since we assumed that the noise is Gaussian and **additive**

$$X = Y + \varepsilon, \qquad \varepsilon \sim N(0, \sigma^2)$$

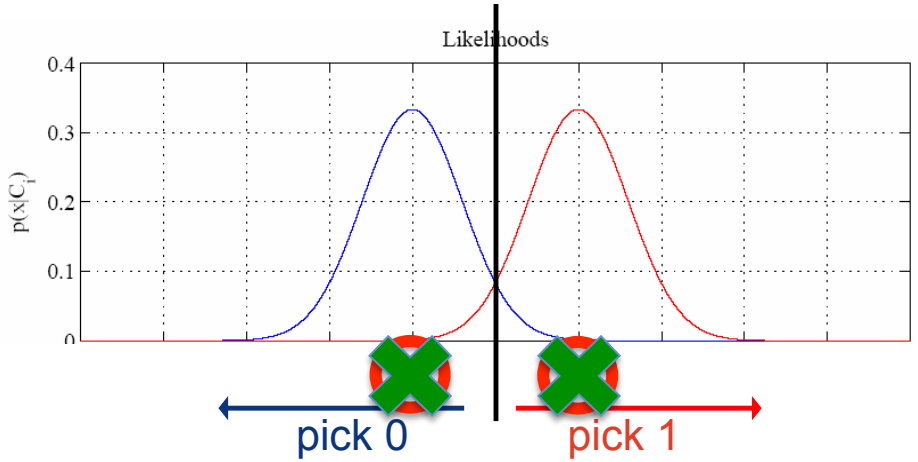
- So X corresponds to the input (Y) plus Gaussian noise

Probability of input given classes

Class 0 $P_{X|Y}(x \mid 0) = G(x, \sigma)$

Class 1 $P_{X|Y}(x \mid 1) = G(x, \sigma)$

$$P_Y(0) = P_Y(1) = 1/2$$



Real-world

$$P_{X|Y}(x \mid 0) = G(x, \mu_0, \sigma)$$
$$P_{X|Y}(x \mid 1) = G(x, \mu_1, \sigma)$$

Bayesian inference

- What happens for the general case $P_{X|Y}(x\,|\,0) = G(x, \mu_0, \sigma)$ $P_{X|Y}(x\,|\,1) = G(x, \mu_1, \sigma)$
- To compute the Bayesian Decision Rule (BDR), we can use **log probabilities**

$$i^*(x) = \arg\max_i \left[\log P_{X|Y}(x|i) + \log P_Y(i)\right]$$

- And note that the priors are equal for everybody so

$$i^*(x) = \arg\max_i \log P_{X|Y}(x|i)$$

- If we develop this equation

$$\begin{aligned} i^*(x) &= \arg\max_i \log P_{X|Y}(x\,|\,i) \\ &= \arg\max_i \log\left\{\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu_i)^2}{2\sigma^2}}\right\} \\ &= \arg\max_i \left\{-\frac{1}{2}\log(2\pi\sigma^2) - \frac{(x-\mu_i)^2}{2\sigma^2}\right\} \\ &= \arg\min_i \frac{(x-\mu_i)^2}{2\sigma^2} \end{aligned}$$

Bayesian decision theory

- If we consider that both distributions have the same variance

$$i^* = \arg\min_i \frac{(x-\mu_i)^2}{2\sigma^2}$$
$$= \arg\min_i (x^2 - 2x\mu_i + \mu_i^2)$$
$$= \arg\min_i (-2x\mu_i + \mu_i^2)$$

- So the optimal decision will be

$$-2x\mu_0 + \mu_0^2 < -2x\mu_1 + \mu_1^2$$
$$2x(\mu_1 - \mu_0) < \mu_1^2 - \mu_0^2$$

$$\boxed{x < \frac{\mu_1 + \mu_0}{2}}$$

- All this work to find this … but we did find back the intuition
- And we had to **make lots of assumptions**
  - **Uniform class probabilities, additive noise, gaussianity**

Bayesian decision rule

- Let's pump up the BDR for **multivariate Gaussian**

$$i^*(x) = \arg\max_i \left[\log P_{X|Y}(x \mid i) + \log P_Y(i)\right]$$

covariance

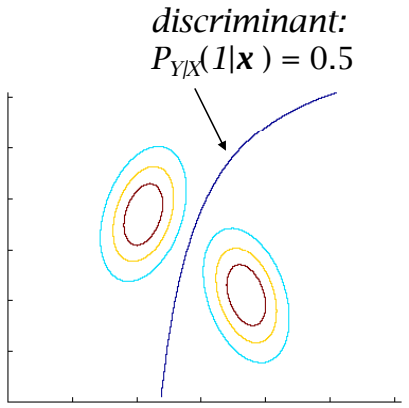
$$P_{X|Y}(x \mid i) = \frac{1}{\sqrt{(2\pi)^d \mid \Sigma_i \mid}} \exp\left\{-\frac{1}{2}(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i)\right\}$$

- Thanks to the *log* the BDR becomes

$$i^*(x) = \arg\max_i \left[-\frac{1}{2}(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i) - \frac{1}{2}\log(2\pi)^d |\Sigma_i| + \log P_Y(i)\right]$$

$$d_i(x,y) = (x-y)^T \Sigma_i^{-1}(x-y) \qquad \alpha_i = \log(2\pi)^d |\Sigma_i| - 2\log P_Y(i)$$

So the **final BDR** is $i^*(x) = \arg\min\left[d_i(x,\mu_i) + \alpha_i\right]$



- The optimal rule is to **assign x to the closest class**
- Mesured with the Mahalanobis distance (d)
- To which a **constant** is added to account for class prior

Summary of BDR

- The Bayesian Decision Rule is the **optimal one**
- The models reflect a causal interpretation of the problem
- Natural decomposition of the problem into
  - « what we knew » **(prior)**
  - « what the data tells us » **(observation)**
- No need for heuristics to combine these two informations
- However BDR optimal **only if models are correct**

Maximum Likelihood

- So we have the optimal (and geometric) solution

$$i^*(x) \quad = \quad \arg\max_i [\underbrace{\mu_i^T \Sigma^{-1} x}_{w_i^T} - \underbrace{\frac{1}{2}\mu_i^T \Sigma^{-1} \mu_i + 2 \log P_Y(i)}_{w_{i0}}]$$

- But we still don't know the parameters $\mu, \Sigma, P_Y(i)$
- We have to **estimate** these values from a **training set**

(ex. use the average value as estimate for the mean)

We rely on the **Maximum Likelihood (ML)** principle

1. Choose a **parametric model** for probabilities (function of parameters)
2. Assemble a **training dataset**
3. Find the **parameters that maximize the probabilities**

Maximum Likelihood

1. **Choose a parametric model** for all probabilities
   - We usually denote parameters by Θ and the class-conditional distributions by

$$P_{X|Y}(x \mid i; \Theta)$$

   - Θ is not a random variable but a parameter (probabilities are function of it)

2. Assemble a **collection of datasets** $\mathcal{D}^{(i)} = \{x_1^{(i)}, \ldots, x_n^{(i)}\}$
   - Set of examples **independently drawn** from class i (cf. sampling schemes)

3. Select the **parameters that maximize the probability of data** (for that i)

$$\Theta_i = \arg\max_{\Theta} P_{X|Y}\left(D^{(i)} \mid i; \Theta\right)$$

$$= \arg\max_{\Theta} \log P_{X|Y}\left(D^{(i)} \mid i; \Theta\right)$$

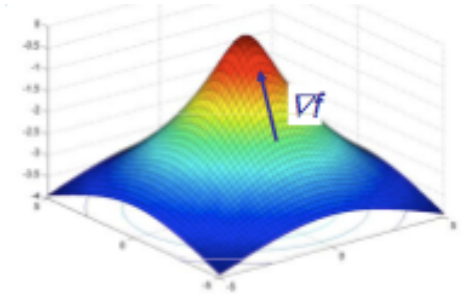
How do we solve this ?

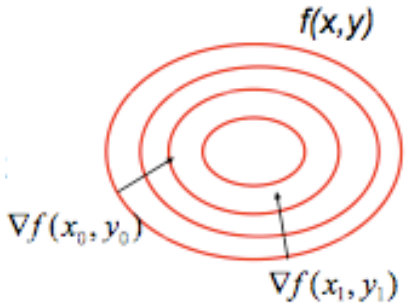
Maximum Likelihood

- The gradient:
  - in higher dimensions, the generalization of the derivative is the gradient. The gradient of a function f(x) at z is:

$$\nabla f(z) = \left(\frac{\partial f}{\partial x_0}(z), \ldots, \frac{\partial f}{\partial x_{n-1}}(z) \right)^T$$

  - It has a nice geometric interpretation:
    - It points in the direction of *maximum growth* of the function
    - *Perpendicular* to the contour where the function is constant





Maximum Likelihood

- The Hessian:
  - extension of the 2nd-order derivative is the Hessian Matrix:

$$\nabla^2 f(x) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_0^2} & \frac{\partial^2 f}{\partial x_0\, \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_0\, \partial x_{n-1}} \\ \frac{\partial^2 f}{\partial x_1\, \partial x_0} & \frac{\partial^2 f}{\partial x_1^2} & \cdots & \frac{\partial^2 f}{\partial x_1\, \partial x_{n-1}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_{n-1}\, \partial x_0} & \frac{\partial^2 f}{\partial x_{n-1}\, \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_{n-1}^2} \end{bmatrix}$$

  - In an ML setup we have a *maximum* when Hessian is negative definite or

$$x^T \nabla^2 f(x) x \leq 0$$

Maximum Likelihood

- In summary:
  1. Choose a parametric model for probabilities $P_X(x;\Theta)$
  2. Assemble $D = \{X_1, \ldots, X_n\}$ of independently drawn examples
  3. Select parameters that maximize the probability of the data
     - or Given a data-set we need to solve

$$\begin{aligned}\Theta^* &= \arg\max_{\Theta} P_X(D;\Theta) \\ &= \arg\max_{\Theta} \log P_X(D;\Theta)\end{aligned}$$

     - The solutions are the parameters such that

$$\begin{aligned}\nabla_\Theta P_X(D;\Theta) &= 0 \\ \theta^t \nabla^2_\Theta P_X(D;\theta)\theta &\leq 0,\ \forall\theta \in \mathbb{R}^n\end{aligned}$$

Frequentist vs. Bayesian

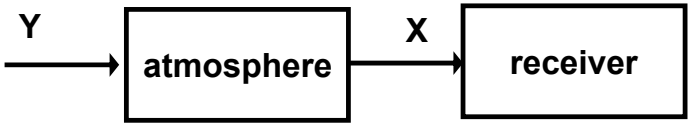
- But wait … why not **using Bayes to estimate the parameters** ?!
- In fact we can ! There is just a **difference in interpretation**
- **Frequentist** (ML) view
  - Probabilities are relative frequencies
  - Makes sense with lot of observations
  - But in most cases we do not have lots of observations
  - And the probabilities are mostly not objective
- **Bayesian** view
  - Probabilities are *subjective* (not equal to relative count)
  - Probabilities are **degrees of belief** on the outcome

Frequentist vs. Bayesian

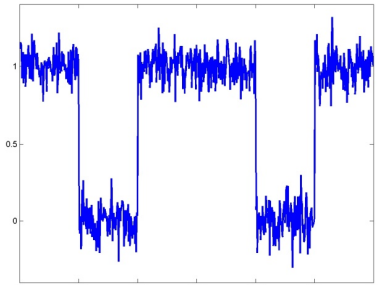
- **Optimal estimate**
  - Under ML there is one « best » estimate (optimization-wise)
  - Under Bayes there is no « best » estimate
  - In the probabilistic framework, « best » has no real sense
- **Predictions**
  - We do not really care about the parameters themselves
  - Only in the fact that they build models …
  - Models can be used to make predictions
  - Unlike ML, Bayes uses **all information in training to predict**

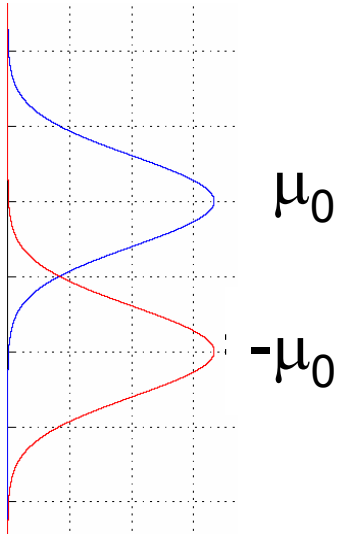
Example

- communications problem



- two states:
  - Y=0 transmit signal s = $-\mu_0$
  - Y=1 transmit signal s = μ_0





- noise model

$$X = Y + \varepsilon, \qquad \varepsilon \sim N(0, \sigma^2)$$

Example

- **Calibration mode**
  - Ask the system to **transmit a single class** and measure X
  - Compute the **ML estimate of the mean of X**

$$\mu = \frac{1}{n}\sum_i X_i$$

- Result: the **estimate is different than** μ_0
- We need to combine **two forms of information**

Our **prior** was $\mu \sim N(\mu_0, \sigma^2)$

Our **data-driven estimate** is $X \sim N(\hat{\mu}, \sigma^2)$

We could stop there but think about **erratic jitter** on one calibration

Bayesian solution

- Gaussian **likelihood** (observations)

$$P_{T|\mu}(D \mid \mu) = G(D, \mu, \sigma^2) \qquad \sigma^2 \text{is known}$$

The **calibration data** we receive — The **mean we estimated** from it

- Gaussian **prior** (what we knew)

$$P_{\mu}(\mu) = G(\mu, \mu_0, \sigma_0^2)$$

The **mean we used** up to now

μ_0, σ_0^2 are known **hyper-parameters**

- We **need to compute** the **posterior distribution** for μ

$$P_{\mu|T}(\mu \mid D) = \frac{P_{T|\mu}(D \mid \mu) P_{\mu}(\mu)}{P_T(D)}$$

The « real » (optimal) **mean we will use** afterwards

Conjugate prior

- For those of you who **sweat over the mathematics, note**
  - the prior $P_\mu(\mu) = G\left(\mu, \mu_0, \sigma_0^2\right)$ is Gaussian
  - the posterior $P_{\mu|T}(\mu \mid D) = G\left(x, \mu_n, \sigma_n^2\right)$ is Gaussian
- Whenever the **posterior is in the same family as the prior**
  - $P_\mu(\mu)$ is a conjugate prior for the likelihood $P_{X|\mu}(x \mid \mu)$
  - posterior $P_{\mu|T}(\mu \mid D)$ is the reproducing density
- A number of likelihoods have **conjugate priors**

Likelihood	Conjugate prior
Bernoulli	Beta
Poisson	Gamma
Exponential	Gamma
Normal (known σ^2)	Gamma

Conjugate prior

- For those of you who **sweat over the mathematics, note**
  - the prior $P_\mu(\mu) = G\left(\mu, \mu_0, \sigma_0^2\right)$ is Gaussian
  - the posterior $P_{\mu|T}(\mu \mid D) = G\left(x, \mu_n, \sigma_n^2\right)$ is Gaussian
- Whenever the **posterior is in the same family as the prior**
  - $P_\mu(\mu)$ is a conjugate prior for the likelihood $P_{X|\mu}(x \mid \mu)$
  - posterior $P_{\mu|T}(\mu \mid D)$ is the reproducing density
- A number of likelihoods have **conjugate priors**

Likelihood	Conjugate prior
Bernoulli	Beta
Poisson	Gamma
Exponential	Gamma
Normal (known σ^2)	Gamma

- And **development of posterior is straightforward**