Bayesian Learning

Complicated derivation of known things.

- Maximal aposteriory probability hypothesis (MAP) (nejpravděpodobnější hypotéza)
- Maximum likelihood hypothesis (ML) (maximálně věrohodná hypotéza)
- Bayesian optimal prediction (Bayes Rate)
- **EM algorithm**
- **Naive Bayes model (classifier)**

Candy Example (Russel, Norvig: Artif. Intell. a MA)

- Our favorite candy comes in two flavors: cherry and lime, both in the same wrapper.
- They are in a bag in one of following rations of cherry candies and prior probability of bags:

hypothesis (bag type)	h_1	h_2	h_3	h_4	h_5
cherry	100%	75%	50%	25%	0%
prior probability h_i	10%	20%	40%	20%	10%

- The first candy is cherry.

MAP Which of h_i is the most probable given first candy is cherry?

yes estimate What is the probability next candy from the same bag is cherry?

Maximum Aposteriory Probability Hypothesis (MAP)

- We assume large bags of candies, the result of one missing candy in the bag is negligable.
- Recall Bayes formula:

$$P(h_i|B=c) = \frac{P(B=c|h_i)\cdot P(h_i)}{\sum_{j=1,\ldots,5} P(B=c|h_j)\cdot P(h_j)} = \frac{P(B=c|h_i)\cdot P(h_i)}{P(B=c)}$$

- We look for the MAP hypothesis **maximálně aposteriorně pravděpodobná**

$$argmax_i P(h_i|B=c) = argmax_i P(B=c|h_i)\cdot P(h_i).$$

- Aposteriory probabilities of hypotheses are in the following table.

Candy Example: Aposteriory Probability of Hypotheses

index	prior	cherry ratio	cherry AND h_i	aposteriory prob. h_i
i	$P(h_i)$	$P(B=c\|h_i)$	$P(B=c\|h_i) \cdot P(h_i)$	$P(h_i\|B=c)$
1	0.1	1	0.1	0.2
2	0.2	0.75	0.15	0.3
3	0.4	0.5	0.2	**0.4**
4	0.2	0.25	0.05	0.1
5	0.1	0	0	0

- Which hypothesis is most probable?

$$h_{MAP} = argmax_i P(data|h_i) \cdot P(h_i)$$

- What is the prediction of a new candy according the most probable hypothesis h_{MAP}?

Bayesian Learning, Bayesian Optimal Prediction

- **Bayesian optimal prediction** is weigthed average of predictions of all hypotheses:

$$\begin{aligned} P(N=c|data) &= \sum_{j=1,\ldots,5} P(N=c|h_j, data) \cdot P(h_j|data) \\ &= \sum_{j=1,\ldots,5} P(N=c|h_j) \cdot P(h_j|data) \end{aligned}$$

- If our model is correct, no prediction has smaller expected error then Bayesian optimal prediction.
- We always assume i.i.d. data, independently identically distributed.
- We assume the hypothesis fully describes the data behavior. Observations are mutually conditionally independent given the hypothesis. This allows the last equation above.

Candy Example: Bayesian Optimal Prediction

i	$P(h_i \mid B = c)$	$P(N = c \mid h_i)$	$P(N = c \mid h_i) \cdot P(h_i \mid B = c)$
1	0.2	1	0.2
2	0.3	0.75	0.225
3	0.4	0.5	0.2
4	0.1	0.25	0.02
5	0	0	0
$\sum$	1		0.645

Maximum Likelihood Estimate (ML)

- Usually, we do not know prior probabilities of hypotheses.
- Setting <u>all prior probabilities equal</u> leads to **Maximum Likelihood Estimate, maximálně věrohodný odhad**

$$h_{ML} = argmax_i P(data|h_i)$$

- Probability of <u>data</u> given hypothesis = likelihood of <u>hypothesis</u> given data.
- Find the ML estimate:

index	prior	cherry ration	cherry AND h_i	Aposteriory prob. h_i
i	$P(h_i)$	$P(B=c\|h_i)$	$P(B=c\|h_i)\cdot P(h_i)$	$P(h_i\|B=c)$
1	0.1	1	0.1	0.2
2	0.2	0.75	0.15	0.3
3	0.4	0.5	0.2	0.4
4	0.2	0.25	0.05	0.1
5	0.1	0	0	0

- In this example, do you prefer ML estimate or MAP estimate?
- (Only few data, overfitting, penalization is usefull. AIC, BIC)

MAP and Penalized Methods

- MAP hypothesis maximizes:

$$h_{MAP} = argmax_i P(data|h_i) \cdot P(h_i)$$

- therefore minimizes:

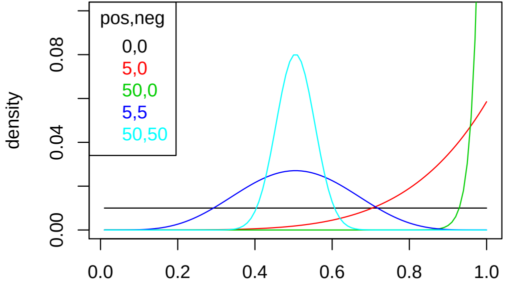
$$\begin{aligned}
h_{MAP} &= argmax_h P(data|h)P(h) \\
&= argmin_h[-log_2 P(data|h) - log_2 P(h)] \\
&= argmin_h[-loglik + \text{complexity penalty}] \\
&= argmin_h[RSS + \text{complexity penalty}] \text{ Gaussian models} \\
&= argmax_h[loglik - \text{complexity penalty}] \text{ Categorical models}
\end{aligned}$$

Remark: Bayesian Parameter Learning

- We represent probability distribution on parameters.
- For binary features, Beta function is used, a is the number of positive examples, b the number of negative examples.

$$beta[a,b](\theta) = \alpha\theta^{a-1}(1-\theta)^{b-1}$$

- (For categorical features, Dirichlet priors and multinomial distribution is used. (Dirichlet-multinomial distribution).
- For Gaussian, μ has Gaussian prior, $\frac{1}{\sigma}$ has gamma prior (to stay in exponential family).)
- Beta Function:



Maximum Likelihood: Continuous Parameter θ

- New producer on the market. We do not know the ratios of candies, any h_θ, kde $\theta \in \langle 0; 1 \rangle$ is possible, any prior probabilities h_θ are possible.
- We look for maximum likelihood estimate.
- For a given hypothesis h_θ, the probability of a cherry candy is θ, of a lime candy $1 - \theta$.
- Probability of a sequence of c cherry and l lime candies is:

$$P(data|h_\theta) = \theta^c \cdot (1 - \theta)^l.$$

ML Estimate of Parameter θ

- Probability of a sequence of c cherry and l lime candies is:

$$P(data|h_\theta) = \theta^c \cdot (1-\theta)^l$$

- Usual trick is to take logarithm:

$$LL(h_\theta; data) = c \cdot \log_2 \theta + l \cdot \log_2(1-\theta)$$

- To find the maximum of LL (log likelihood of the hypothesis) with respect to θ we set the derivative equal to 0:

$$\begin{aligned} \frac{\partial LL(h_\theta; data)}{\partial \theta} &= \frac{c}{\theta} - \frac{l}{1-\theta} \\ \frac{c}{\theta} &= \frac{l}{1-\theta} \\ \theta &= \frac{c}{c+l}. \end{aligned}$$

ML Estimate of Multiple Parameters

- Producer introduced two colors of wrappers - red r and green g.
- Both flavors are wrapped in both wrappers, but with different probability of the red/green wrapper.
- We need three parameters to model this situation:

$P(B=c)$	$P(W=r\|B=c)$	$P(W=r\|B=l)$
θ_0	θ_1	θ_2

- Following table denotes observed frequences:

wrapper\ flavor	cherry	lime
red	r_c	r_l
green	g_c	g_l

ML Estimate of Multiple Parameters

Parameters are:

$P(B=c)$	$P(W=r\mid B=c)$	$P(W=r\mid B=l)$
θ_0	θ_1	θ_2

Probability of data given the hypothesis $h_{\theta_0,\theta_1,\theta_2}$ is:

$$\begin{aligned}
P(data|h_{\theta_0,\theta_1,\theta_2}) &= \theta_1^{r_c}\cdot(1-\theta_1)^{g_c}\cdot\theta_0^{r_c+g_c}\cdot\theta_2^{r_l}\cdot(1-\theta_2)^{g_l}\cdot(1-\theta_0)^{r_l+g_l}\\
LL(h_{\theta_0,\theta_1,\theta_2};data) &= r_c\log_2\theta_1+g_c\log_2(1-\theta_1)+(r_c+g_c)\log_2\theta_0\\
&\quad +r_l\log_2\theta_2+g_l\log_2(1-\theta_2)+(r_l+g_l)\log_2(1-\theta_0)
\end{aligned}$$

We look for maximum:

$$\begin{aligned}
\frac{\partial LL(h_{\theta_0,\theta_1,\theta_2};data)}{\partial\theta_0} &= \frac{r_c+g_c}{\theta_0}-\frac{r_l+g_l}{1-\theta_0}\\
\theta_0 &= \frac{(r_c+g_c)}{r_c+g_c+r_l+g_l}\\
\frac{\partial LL(h_{\theta_0,\theta_1,\theta_2};data)}{\partial\theta_2} &= \frac{r_l}{\theta_2}-\frac{g_l}{1-\theta_2}\\
\theta_2 &= \frac{r_l}{r_l+g_l}.
\end{aligned}$$

Discrete Variables

- Maximum Likelihood estimate is the ratio of fequences.
- **Naive Bayes Model, Bayes Classifier** assumes independent features given the class variable.
  - Calculate prior probability of classes $P(c_i)$
  - For each feature f, calculate for each class the probability of this feature $P(f|c_i)$
  - For a new observation of features f predict the most probable class $argmax_{c_i} P(f|c_i) \cdot P(c_i)$.

Bayes factor

  - We can start with a comparison ratio of two classes $\frac{P(c_i)}{P(c_j)}$
  - after each observation x_p multiply it by the **bayes factor** $\frac{P(x_p|c_i)}{P(x_p|c_j)}$
  - that is:

$$\frac{P(c_i|x_1,\ldots,x_p)}{P(c_j|x_1,\ldots,x_p)} = \frac{P(c_i)}{P(c_j)} \cdot \frac{P(x_1|c_i)}{P(x_1|c_j)} \cdot \ldots \cdot \frac{P(x_p|c_i)}{P(x_p|c_j)}.$$

- Bayesian Networks learn more complex (in)dependencies between features.

Bayesian Information Criterion BIC

- Suppose a set of candidate models $\mathcal{M}_m, m = 1, \dots M$ and corresponding parameters θ_m
- training data $\mathbf{Z} = \{x_i, y_i\}_{i=1}^N$

$$
\begin{aligned}
P(\mathcal{M}_m|\mathbf{Z}) &\propto P(\mathcal{M}_m) \cdot P(\mathbf{Z}|\mathcal{M}_m) \\
&\propto P(\mathcal{M}_m) \cdot \int P(\mathbf{Z}|\theta_m, \mathcal{M}_m) P(\theta_m|\mathcal{M}_m) d\theta_m
\end{aligned}
$$

- Typically we assume that the prior over models is uniform.
- For $P(\mathbf{Z}|\mathcal{M}_m)$ a Laplace approximation is used

$$
\log P(\mathbf{Z}|\mathcal{M}_m) = \log P(\mathbf{Z}|\hat{\theta}_m, \mathcal{M}_m) - \frac{d_m}{2} \log N + O(1)
$$

  - $\hat{\theta}_m$ maximum likelihood estimate
  - d_m the number of free parameters in model $\mathcal{M}_m$.
- If we define our loss function to be $-2 log P(\mathbf{Z}|\hat{\theta}_m, \mathcal{M}_m)$
- we get the BIC criterion

$$
BIC = -2 \cdot loglik + (log N) \cdot d.
$$

BIC Model Comparison

- We can estimate the posterior probability of each model $\mathcal{M}_m$ as

$$\hat{P}(\mathcal{M}_m) = \frac{e^{-\frac{1}{2}\cdot BIC_m}}{\sum_{\ell=1}^{M} -\frac{1}{2} \cdot BIC_\ell}.$$

Bayesian Methods

- We specify a sampling model $P(\mathbf{Z}|\theta)$
- and a prior distribution for parameters $P(\theta)$
- then we compute

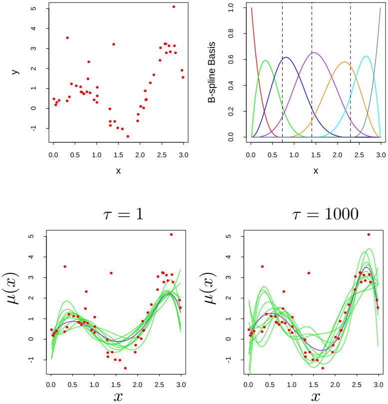
$$P(\theta|\mathbf{Z}) = \frac{P(\mathbf{Z}|\theta) \cdot P(\theta)}{\int P(\mathbf{Z}|\theta) \cdot P(\theta) d\theta},$$

  - we may draw samples
  - or summarize by the mean or mode.
  - it provides the predictive distribution:

$$P(z^{new}|\mathbf{Z}) = \int P(z^{new}|\theta) \cdot P(\theta|\mathbf{Z}) d\theta.$$

Bayesian smoothing example

- Training data $\mathbf{Z} = \{z_i, \ldots, z_N\}$, $z_i = (x_i, y_i)$, $i = 1, \ldots, N$.
- We look for a cubic spline with three knots in quartiles of the X values. It corresponds to B-spline basis $h_j(x)$, $j = 1, \ldots, 7$.
- We estimate the conditional mean $\mathbb{E}(Y|X = x)$: $\mu(x) = \sum_{j=1}^{7} \beta_j h_j(x)$
- Let $\mathbf{H}$ be the N matrix $h_j(x_i)$.
- RSS β estimate is $\hat{\beta} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{y}$.

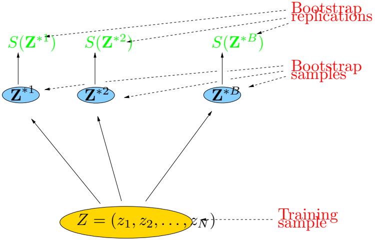


We assume to know σ^2, fixed x_i, we specifying prior on $\beta \sim N(0, \tau\Sigma)$.

$$\mathbb{E}(\beta|\mathbf{Z}) = (\mathbf{H}^T\mathbf{H} + \frac{\sigma^2}{\tau}\Sigma^{-1})^{-1}\mathbf{H}^T\mathbf{y}$$

$$\mathbb{E}(\mu(x)a|\mathbf{Z}) = h(t)^T(\mathbf{H}^T\mathbf{H} + \frac{\sigma^2}{\tau}\Sigma^{-1})^{-1}\mathbf{H}^T\mathbf{y}.$$

Bootstrap



- We select N samples with replacement
- the probability of not being selected is roughly 0.368
- $\widehat{Err}^{(1)} = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{|C^{-i}|}\sum_{b\in C^{-i}} L(y_i, \hat{f}^{*b}(x_i))$.
- more in Model assessment and selection slides.

Missing data (T.D. Nielsen)

Die tossed N times. Result reported via noisy telephone line. When transmission not clearly audible, record missing value:

$$4, 2, ?, 6, 5, 4, ?, 3, 4, 1, \ldots$$

"2" and "3" sound similar, therefore:

$$P(Y_i = ?|X_i = k) = P(M_i = 1|X_i = k) = \begin{cases} 1/4 & \text{k} = 2{,}3 \\ 1/8 & \text{k} = 1{,}4{,}5{,}6 \end{cases}$$

Distribution of the Y is (for fair die):

?	$\frac{1}{3}\frac{1}{4} + \frac{2}{3}\frac{1}{8} = \frac{1}{6}$
2,3	$\frac{1}{6}\frac{3}{4} = \frac{1}{8}$
1,4,5,6	$\frac{1}{6}\frac{7}{8} = \frac{7}{48}$

If we simply ignore the missing data items, we obtain as the maximum likelihood estimate for the parameters of the die:

$$\theta^* = \left(\frac{7}{48}, \frac{1}{8}, \frac{1}{8}, \frac{7}{48}, \frac{7}{48}, \frac{7}{48}\right) * \frac{6}{5} = (0.175, 0.15, 0.15, 0.175, 0.175, 0.175)$$

Incomplete data

How do we handle cases with missing values:

- Faulty sensor readings.
- Values have been intentionally removed.
- Some variables may be unobservable.

How is the data missing?
We need to take into account how the data is missing:

- **Missing completely at random** The probability that a value is missing is independent of both the observed and unobserved values (a monitoring system that is not completely stable and where some sensor values are not stored properly).
- **Missing at random** The probability that a value is missing depends only on the observed values (a database containing the results of two tests, where the second test has only performed (as a "backup test") when the result of the first test was negative).
- **Non-ignorable** Neither MAR nor MCAR (an exit poll, where an extreme right-wing party is running for parlament).

EM - Algorithm

- EM algorithm is used for learning a model with unobserved variables (for example, cluster membership).
- We assume (hope) they are missing at random.
- It is an iterative algorithm with two steps:
  - **Expectation**, fills in the unobserved data based on current M model, and
  - **Maximize**, finds maximum (log)likelihood model given the data filled in E step.

Example: T.D. Nielsen

Learning by EM - Algorithm

- Clustering (observed may be of categorical and/or continuous)
- Hidden Markov Models
- Latent Dirichlet Allocation
- Hierarchical Mixtures of Experts
- and others.

ML Estimate of Gaussian Distribution Parameters

- Assume x to have Gaussian distribution with unknown parameters μ a σ.
- Our hypotheses are $h_{\mu,\sigma} = \frac{1}{\sqrt{2\pi\sigma}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$.
- We have observed $x_1, \dots, x_n$.
- Log likelihood is:

$$\begin{aligned} LL &= \sum_{j=1}^{N} log \frac{1}{\sqrt{2\pi\sigma}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \\ &= N \cdot (log \frac{1}{\sqrt{2\pi\sigma}}) - \sum_{j=1}^{N} \frac{(x_j-\mu)^2}{2\sigma^2} \end{aligned}$$

- Find the maximum.

Linear Gaussian Distribution

- Assume random variable (feature) X.
- Assume goal variable Y with linear gaussian distribution where $\mu = b \cdot x + b_0$ and fixed variance σ^2 $p(Y|X = x) = N(b \cdot x + b_0; \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(y-((b \cdot x + b_0))^2}{2\sigma^2}}$.
- Find maximum likelihood estimate of b, b_0 given a set of observations $data = \{\langle x_1, y_1 \rangle, \ldots, \langle x_N, y_N \rangle\}$.
- (Look for maximum of the logarithm of it; change the max to min with the opostite sign. Do you know this formula?)

$$argmax_{b,b_0}(log_e(\Pi_{i=1}^{N}(e^{-(y_i-(b \cdot x_i + b_0))^2}))) = argmin_{b,b_0}(?)$$

Reasons for Modelling Unobserved Variables

- We know the model structure, observations are missing.
- Unobserved variable makes many features conditionally independent (that is, simplifies the model).
- Often, mixtures of Gaussians are used. It is also our example: clustering.
- Also used to learn Hidden Markov Models.

EM Algorithm

- We have a model from the previous step (at the beginning, we may choose random cluster centers and/or uniformly distributed values or values based on sample mean and variance.
- Use weighted data, each row i with unobserved variables filled by j is the weight γ_{ij}.
- **Expectation** step: For each data row:
  - Calculate the conditional probability of possible values of unobserved variables given the model.
- **Maximize** step: for some models we know:
  - gaussians - mean and standard deviation are maximum likelihood estimates of μ, σ,
  - discrete - the ratios of observed counts.

Mixture of Two Gaussians, one input feature x

- Model parameters: $\pi, \mu_1, \sigma_1^2, \mu_2, \sigma_2^2$, initialize μ randomly, $\pi = 0.5$, $\sigma^2 =$ sample variance, π prior of the second cluster
- Expectation – step: fill the data, estimate weights, $\gamma_i = P(C_i = 2|x_i)$:

$$\gamma_i = \frac{\pi \phi_{\theta_2}(x_i)}{(1-\pi)\phi_{\theta_1}(x_i) + \pi \phi_{\theta_2}(x_i)}$$

- Maximize – step: estimate new model,

$$\mu_1 = \frac{\sum_{i=1}^{N}(1-\gamma_i)x_i}{\sum_{i=1}^{N}(1-\gamma_i)}$$

$$\sigma_2^2 = \frac{\sum_{i=1}^{N}\gamma_i(x_i-\mu_2)^2}{\sum_{i=1}^{N}\gamma_i}$$

$$\pi = \frac{\sum_{i=1}^{N}\gamma_i}{N}$$

- iterate E–M until convergence.

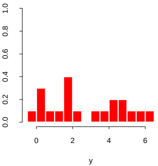
Mixture of K Gaussians

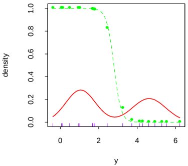
- Model parameters $\pi_1, \dots, \pi_k, \mu_1, \dots, \mu_k, \Sigma_1, \dots, \Sigma_k$ such that $\sum_{k=1}^{K} \pi_k = 1$.
- Expectation: weights of unobserved 'fill–ins' k of variable C:

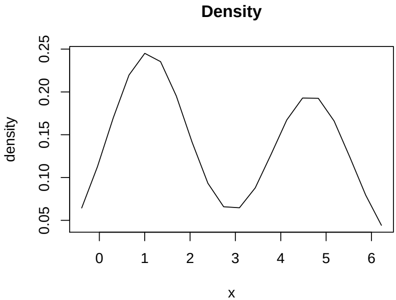
$$
\begin{aligned}
p_{ik} &= P(C = k|x_i) = \alpha \cdot P(x_i|C_i = k) \cdot P(C_i = k) \\
&= \frac{\pi_k \phi_{\theta_k}(x_i)}{\sum_{l=1}^{K} \pi_l \phi_{\theta_l}(x_i)} \\
p_k &= \sum_{i=1}^{N} p_{ik}
\end{aligned}
$$

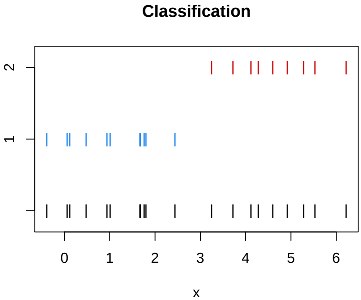
- Maximize: mean, variance and cluster 'prior' for each cluster k:

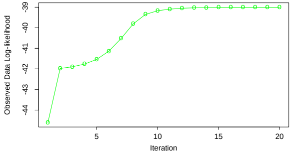
$$
\begin{aligned}
\mu_k &\leftarrow \sum_i \frac{p_{ik}}{p_k} x_i \\
\Sigma_k &\leftarrow \sum_i \frac{p_{ik}}{p_k} (x_i - \mu_k)(x_i - \mu_k)^T \\
\pi_k &\leftarrow \frac{p_k}{\sum_{l=1}^{K} p_l}
\end{aligned}
$$

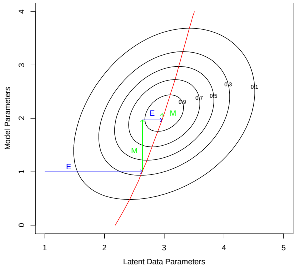





Density
density
0.05
0.10
0.15
0.20
0.25
0
1
2
3
4
5
6
x

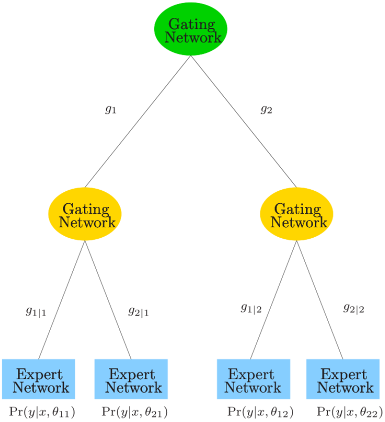

Classification
1
2
0
1
2
3
4
5
6
x


Observed Data Log-likelihood
-44
-43
-42
-41
-40
-39
5
10
15
20
Iteration

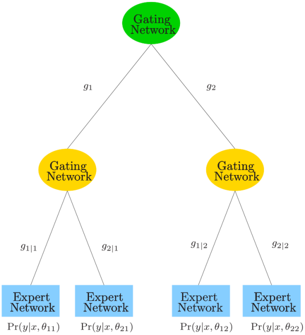

Model Parameters
Latent Data Parameters
0
1
2
3
4
1
2
3
4
5
E
M
E
M
0.9
0.7
0.5
0.3
0.1

Hierarchical Mixture of Experts

- a hierarchical extension of naive Bayes (latent class model)
- a decision tree with 'soft splits'
- splits are probabilistic functions of a linear combination of inputs (not a single input as in CART)
- terminal nodes called 'experts'
- non–terminal nodes are called gating network
- may be extended to multilevel.



Hierarchical Mixture of Experts



- data (x_i, y_i), $i = 1, \ldots, N$, y_i continuous or categorical, first $x_i \equiv 1$ for intercepts.
- $g_i(x, \gamma_j) = \frac{e^{\gamma_j^T x}}{\sum_{k=1}^K e^{\gamma_k^T x}}$, $j = 1, \ldots, K$ children of the root,
- $g_{\ell|j}(x, \gamma_{j\ell}) = \frac{e^{\gamma_{j\ell}^T x}}{\sum_{k=1}^K e^{\gamma_{jk}^T x}}$, $\ell = 1, \ldots, K$ children of the root,
- Terminals (Experts)

ession: Gaussian linear reg. model, $\theta_{j\ell} = (\beta_{j\ell}, \sigma j\ell^2)$, $Y = \beta_{j\ell}^T + \epsilon$

cation: The linear logistic reg. model: $Pr(Y = 1|x, \theta_{j\ell}) = \frac{1}{1+e^{-\theta_{j\ell}^T x}}$

- EM algorithm
- Δ_i, $\Delta_{\ell|j}$ 0–1 latent variables – branching

E step: expectations for Δ's

M step: estimate parameters HME by a version of multiple logistic regression.