T-61.5040

Learning Models and Methods, (5 cr) L

Spring 2007

Lecturer:	**Petteri Pajunen**
Course Assistant:	**Ville Viitaniemi**

LECTURE 4: 7.2.2007

BAYESIAN INFERENCE

- Dutch Book and Cox's Axioms support using probability as a measure of uncertainty
- Dutch Book Theorem implies that a measure of uncertainty must have certain properties, such as
  - product rule: $p(AB|C) = p(A|BC)p(B|C)$ (AB means both events A and B happen)
  - sum rule: $p(A|B) + p(\overline{A}|B) = 1$ ($\overline{A}$ means A does not happen)
  - equivalent events: $p(A) = p(B)$ if A and B are equivalent events
- Cox's axioms yield the same properties

Bayes' Theorem

- consider uncertain events A and B
- product rule gives

$$p(BA) = p(B|A)p(A)$$
$$p(AB) = p(A|B)p(B)$$

- AB and BA are *equivalent events*
- their probabilities must be the same

- equating $p(BA)$ and $p(AB)$ gives

$$p(B|A)p(A) = p(A|B)p(B)$$

- dividing by $p(B)$ gives the *Bayes' Theorem*

$$p(A|B) = \frac{p(B|A)p(A)}{p(B)}$$

- BT allows us to *reverse* conditional probabilities
- in context of the learning problem, BT can be used when
  - prior uncertainty gives the prior $p(\theta)$
  - modeling assumptions give $p(D|\theta)$
- BT results in the *posterior distribution*

$$p(\theta|D) = \frac{p(D|\theta)p(\theta)}{p(D)}$$

- constant denominator $p(D)$ is often omitted:

$$p(\theta|D) \propto p(D|\theta)p(\theta)$$

- when necessary, it can be computed as

$$p(D) = \int p(D|\theta)p(\theta)d\theta$$

- unnormalized posterior can often be used directly

- how to get $p(D|\theta)$ from uncertainty?
- Dutch Book and Cox do not directly give conditional probabilities
- instead, one can start with the *full probability model*

$$p(\theta, D)$$

- obtained by using all relevant quantities in the distribution
- Dutch Book and Cox say that this distribution exists
- the full probability model defines all other probabilites involving θ and D

- integrate over data (sum rule):

$$p(\theta) = \int p(\theta, D) dD$$

- likelihood by the product rule:

$$p(D|\theta) = p(\theta, D) / p(\theta)$$

- both are obtained from the full probability model using rules of computing with probabilities
- in practice, $p(\theta)$ and $p(D|\theta)$ are usually specified directly

- $p(D|\theta)$ and $p(\theta)$ are specified using the information available about the problem
- using the Bayes' Theorem in unnormalized form gives the posterior as

$$p(\theta|D) \propto p(D|\theta)p(\theta)$$

- posterior quantifies the remaining uncertainty about θ taking the information in D into account
- this result is unique, if we believe that our probability model is correct

- note that $p(\theta, D)$ does not have to be "correct" in any general sense
- correctness here means "correctly quantifies our subjective information"
- then the posterior correctly quantifies our information after having seen data D
- practical reasons may justify other learning methods, but in principle learning proceeds as described above

- *point estimation*: many learning methods would give a single value for θ
- posterior does not define a unique point estimate
- a loss function is normally required (not part of the probability model)
- e.g. minimum MSE, maximum likelihood
- exception: the posterior is concentrated on a single value

- example: cross a bridge with a 10-ton truck
- somehow you believe that the bridge holds 11 tons with $p = 0.8$ and 9 tons with $p = 0.2$
- a reasonable point estimate might suggest 11 tons (would you use it and cross the bridge?)
- consequences imply here a nonsymmetric loss

- overfitting: for example, $\hat{\theta} = \operatorname{argmax} p(\theta|D)$
- consider a real-valued θ
- then the density $p(\hat{\theta}|D)$ is not a probability
- but an integral over some neighbourhood of $\hat{\theta}$ is:

$$p(\hat{\theta} - \epsilon \leq \theta \leq \hat{\theta} + \epsilon | D) = \int_{\hat{\theta}-\epsilon}^{\hat{\theta}+\epsilon} p(\theta|D) d\theta$$

- it is possible that $p(\hat{\theta}|D)$ is large, but the integral above is small (even for somewhat large ϵ)
- visually, this happens when the posterior has a "narrow peak" around $\hat{\theta}$
- this means that it is not very probable that θ is close to $\hat{\theta}$ (exercise problem)

- recall the linefitting problem:

$$\begin{aligned} y &= \mu x + \beta + n, \;\; n \sim N(0, \sigma^2) \\ &= \begin{bmatrix} x & 1 \end{bmatrix} \theta + n, \;\; \theta = \begin{bmatrix} \mu & \beta \end{bmatrix}' \end{aligned}$$

- $N(y|a,b)$ means a Normal density for y with mean a and variance b

- denote

$$X = \begin{bmatrix} x_1 & x_2 & \dots & x_n \\ 1 & 1 & \dots & 1 \end{bmatrix}'$$
$$Y = [y_1, \dots, y_n]'$$

- likelihood is $p(Y|\theta, X) = N(Y|X\theta, \sigma^2 I)$
- choose first a prior $p(\theta|X) \propto c$ (constant prior)

- use "completing the square"
- consider any expression

$$C \exp(-\frac{1}{2}(\theta' A \theta + \theta' b + b' \theta + c))$$

- this is an unnormalized Normal distribution $N(\theta|m, R)$: write the exponent as

$$-\frac{1}{2}(\theta - m)' R^{-1} (\theta - m)$$

- we see that $R = A^{-1}$ and $m = -Rb$

- compute the posterior:

$$\begin{aligned} p(\theta|Y, X) &\propto p(Y|\theta, X) p(\theta|X) \\ &= N(Y|X\theta, \sigma^2 I) p(\theta|X) \end{aligned}$$

- use "completing the square" on the exponent (ignore additive constants):

$$\begin{aligned} \log p(\theta|Y, X) &\propto -\frac{1}{2}\sigma^{-2}(Y - X\theta)'(Y - X\theta) \\ &= -\frac{1}{2}\sigma^{-2}(Y'Y + \theta' X' X \theta - \theta' X' Y - Y' X \theta) \\ &= -\frac{1}{2}(\theta - m_\theta)' R_\theta^{-1} (\theta - m_\theta) \\ &= -\frac{1}{2}\sigma^{-2}(\theta - (X'X)^{-1} X' Y)' X' X (\theta - (X'X)^{-1} X' Y) \end{aligned}$$

- therefore the posterior is

$$p(\theta|Y,X) = N(\theta|m_\theta, R_\theta)$$
$$m_\theta = (X'X)^{-1}X'Y$$
$$R_\theta = \sigma^2(X'X)^{-1}$$

- the posterior mean is familiar from standard linear regression as it is obtained using the *pseudoinverse* $(X'X)^{-1}X'$
- what if the prior is not constant?
- lets choose a Normal prior $p(\theta|X) = N(\theta|a,B)$

- using the same technique as above (completing the square) we obtain

$$p(\theta|Y,X) = N(\theta|m_\theta, R_\theta)$$
$$m_\theta = (\sigma^{-2}X'X + B^{-1})^{-1}(\sigma^{-2}X'Y + B^{-1}a)$$
$$R_\theta = (\sigma^{-2}(X'X) + B^{-1})^{-1}$$

- this is easy to compute for any a, B
- check what happens when $B = \sigma_p^2 I$ and σ_p^2 gets very large

- why did we get the pseudoinverse result at first?
- when the prior is *constant*, the posterior is always proportional to the *likelihood*:

$$p(\theta|D) \propto p(D|\theta)p(\theta) \propto p(D|\theta)$$

- for a Normal posterior, $\max p(\theta|D) = \mathrm{E}(\theta|D) = m_\theta$
- *maximum likelihood* would then also give m_θ
- BI gives this result and the posterior variance directly

Marginalization

- Bayesian Inference seems almost too trivial to be useful and interesting
- but practical applications can be (and usually are) quite complicated
- some interesting properties can be demonstrated without going to problem-specific details
- *marginalization*: removing uninteresting but relevant quantities

- example: which taxi company $C \in \{1,2\}$ to take to airport
- goal: minimize driving time t
- two routes $r \in \{1,2\}$
- assume we know distribution of routes for each company $p(r|C)$ and distribution of driving times $p(t|r)$ on each route

- we need to find $p(t|C)$
- why bother with route r if we just care about the driving time?

$$\begin{aligned} p(t|C) &= p(t, r=1|C) + p(t, r=2|C) \\ &= p(t|r=1)p(r=1|C) + p(t|r=2)p(r=2|C) \end{aligned}$$

- route must be included, since it links C and t
- but we don't care about the route, so $p(t|C)$ is what we want to compute

- removing uninteresting but relevant quantities is called *marginalization*
- suppose $\theta = (x, y, z)$ and y is uninteresting
- then the goal is to compute $p(x, z|D)$
- if we get it as $p(x, z|D) \propto p(D|x, z)p(x, z)$, then y is irrelevant
- heuristic, wrong solution is to compute $p(x, y, z|D)$ and estimate y, then use $p(x, \hat{y}, z|D)$ as the distribution over x, z

- correct solution from rules of probability:

$$p(x, z|D) = \int p(x, y, z|D) dy$$

- called marginalization, because we are computing a marginal distribution of the full posterior
- intuition: all possible values of y are considered, and their effect is weighted by the full posterior

- example: noise variance in regression
- often regression noise is modeled as $N(0, \sigma^2)$
- variance σ^2 assumed to be an unknown constant
- leads to the familiar least-squares solution
- Bayes requires considering *all values of* σ^2
- we'll see later that this gives a different result

Model Averaging

- another way to think about marginalization, especially when predicting observable values
- consider regression, where output y is predicted as a function of an input x
- assume we are interested in predicting $\tilde{y}$ at an input $\tilde{x}$
- non-Bayesian solution: find an estimate $\hat{\theta}$ for the regression parameters, then predict $\tilde{y} = f(\tilde{x}|\hat{\theta})$
- this can lead to overfitting as seen before

- let's predict $\tilde{y}$ using probabilities
- $\tilde{y}$ is unknown so we must have a distribution over it
- we know $\tilde{x}$ and D, so the distribution must be conditional to these values
- *we don't know* θ, so the distribution is *not* conditional to θ

- we want to compute the *predictive distribution*

$$p(\tilde{y}|\tilde{x}, D)$$

- in general, a predictive distribution is any distribution over an observable quantity (e.g. $p(D)$)
- assume we obtained $p(\theta|D)$
- θ must be included in the analysis, because it is clearly relevant

- use marginalization backwards:

$$p(\tilde{y}|\tilde{x}, D) = \int p(\tilde{y}, \theta|\tilde{x}, D) d\theta$$

- the integrand is the *full posterior of* $\tilde{y}, \theta$
- use the product rule to split the integrand:

$$p(\tilde{y}|\tilde{x}, D) = \int p(\tilde{y}|\tilde{x}, \theta, D) p(\theta|\tilde{x}, D) d\theta$$

- remove unnecessary quantities (requires model assumptions):

$$p(\tilde{y}|\tilde{x}, D) = \int p(\tilde{y}|\tilde{x}, \theta) p(\theta|D) d\theta$$

- $p(\tilde{y}|\tilde{x}, \theta, D) = p(\tilde{y}|\tilde{x}, \theta)$ since θ determines predictions
- $p(\theta|\tilde{x}, D) = p(\theta|D)$ since pairs (x, y) determine θ
- $p(\tilde{y}|\tilde{x}, \theta)$ is also a predictive distribution
- it cannot be used directly since θ is unknown

- integral *averages the predictive distributions using the posterior over* θ
- predictions made by more probable θ carry more weight and vice versa
- follows from marginalization and the product rule (no heuristics used)
- result is the only possible distribution of $\tilde{y}$ quantifying the uncertainty in the prediction

- possible misunderstanding of model averaging
- *averaging is over probabilities, not predicted values!*
- assume you write “0” badly, so it looks a bit like “9”
- handwritten digit recognition model might predict “0” for θ_1 and “9” for θ_2 ($p(\theta_i|D) = 0.5,\ i = 1, 2$)
- model averaging does not yield 4.5
- instead, model averaging puts some probability on both predictions

Overfitting

- caused by too many values of θ fitting to data
- for any such θ, $p(\theta|D)$ may be high
- if the predictions for such θ's are different, this is overfitting
- this does not always happen: the predictions may also be similar

- model averaging automatically reveals overfitting
- assume $p(\theta_1|D) = p(\theta_2|D) = p(\theta_3|D) = 1/3$
- if θ_i predicts $\tilde{y} = i$ with probability one, then

$$p(\tilde{y} = i|\tilde{x}, D) = 1/3, \quad i \in \{1, 2, 3\}$$

- but if each θ_i predicts $\tilde{y} = 1$, then the predictive distribution is $p(\tilde{y} = 1|\tilde{x}, D) = 1$

- different predictions by probable θ's mean a *high predictive variance* and thus overfitting
- similar predictions lead to *low predictive variance*
- thus, the predictive distribution reveals overfitting and model-selection heuristics are not needed
- predictive variance can be different at different $\tilde{x}$: the whole model does not necessarily overfit

- example: linear regression using fixed, nonlinear basis functions
- can be solved almost as easily as the linear example
- predictive distributions can also be computed easily
- demo: `demo_breg.R`