Decision Theory

Recall:

- Frequentist approach: maximizing likelihood.
- Bayesian approach: maximizing posterior.

In the next 2 classes we will learn about **regression** and **classification**, 2 main problems of machine learning.

Example

Say we want to make a medical diagnosis. The input is an X-ray image x, and the classes are c_2 if the person has cancer and c_2 if not.

A natural solution by the Bayesian approach for the problem is to find the class label that is most probable, given the input image: $p(c_k|x) = \frac{p(x|x_k)p(c_k)}{p(x)}$

We will then try to find $maxarg_{k\in\{1,2\}}p(c_k|x)$

But – will that be the best solution?

Misclassification Rate

Decision region R_k: all points in R_k are assigned to class C_k

Decision boundary (surface): the boundary between the decision regions.

Then the mistake is:

$$p(mistake) = p(x \in R_1, C_2) + p(x \in R_2, C_1) = \int_{R_1} p(x, C_2)dx + \int_{R_2} p(x, C_1)dx =$$

And we would want to minimize the mistake rate. Here – only 2 cases we can make a mistake. This can be written using the posterior:

$$\int_{R_1} p(C_2|x)p(x)dx + \int_{R_2} p(C_1|x)p(x)dx$$

⇒ We need to maximize the posterior. Maximizing the posterior is the same as minimizing the number of mistakes.

Expected Loss

In reality making some mistakes can be much more harmful than others, for instance diagnosing cancer when it's not there is less severe than missing the cancer in the diagnosis.

Loss matrix: L_{kj} is the loss associated with assigning data in region R_j to C_k (coloumn R_j, row C_k):

	Cancer	Normal
Cancer	0	1000
Normal	1	0

$$E[L] = \sum_k \sum_j \int_{R_j} L_{kj}p(x, C_k)dx \propto \sum_k L_{kj}p(C_k|x)$$

And we want to determine a decision boundary such that the expected loss is minimized, which is simply minimizing the weighted probability of making mistakes.

Inference and Decision

There are 3 different approaches:

Generative learning:

- Inference stage: first we estimate class-conditional densities $p(x|C_k)$ (likelihood) and prior $p(C_k)$.
  for instance: $p(C_k)$ is some statistics on cancer across genders / ages etc.
- Use Bayes $p(C_k|x) = \frac{p(x|C_k)p(C_k)}{p(x)}$ to find the posterior probabilities.

Discriminative approach:

Directly try to model the posterior class probabilities $p(C_k|x)$.

Discriminant function:

Direct analytical approach and find a mapping $f(x)$ that maps inputs to labels with no probabilities.

Generative vs. Discriminative

GEN:

- Most demanding
- But, can directly and explicitly impose a prior, and in many cases we have knowledge of prior probabilities that we would want to incorporate in the decision process.
- Can determine marginal density $p(x) = \sum_k p(x|C_k)p(C_k)$ (for outlier detection).

DISC:

- No need to model joint dist. – efficient for classification.

DET:

- No ways to make guarantees of accuracy, as no access to posterior probs.

Loss function for Regression

$$E[L] = \int\int \{y(x) - t\}^2 p(x,t)dxdt$$

We want to make the squared distance between the predicted value and target such the expectation is minimized.

We have a set of pairs $\{(x,t)_i\}_{i=1}^n$, and $y(x)$

We are looking for: $y^* = argmin_{y(x)} E[L]$

To find y^* we derive and compare to 0:

$\frac{\partial E[L]}{\partial y(x)} = 2\int (y(x) - t)p(x,t)dt = 0 \Leftrightarrow$

$\int y(x)p(x,t)dt - \int tp(x,t)dt = 0 \Leftrightarrow$

$y(x)p(x) - \int tp(x,t)dt = 0 \Leftrightarrow$

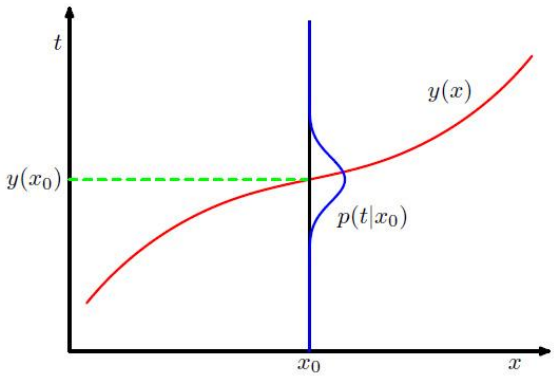
$$y(x) = \frac{\int tp(x,t)dt}{p(x)} = \frac{\int tp(t|x)p(x)dt}{p(x)} = \int tp(t|x)dt = \boxed{E_t[t|x]}$$

So y^* is the curve that is the expectation of t (the target points) given the data points x.

Note: there could be multiple targets t_i for the same x.

The function we got $E_t[t|x]$ is the regression function.

The regression function $y(x)$, which minimizes the expected squared loss, is given by the mean of the conditional distribution $p(t|x)$.



Different perspective:

If we expand the square first before deriving, we are left with:

$$E[L] = \int \{y(x) - E[t|x]\}^2 p(x)dx + \int \{E[t|x] - t\}^2 p(x)dx$$

And those terms that were left are the variance – minimizing the variances.

Linear Regression

Curve Fitting

Given N observations $x \equiv (x_1, \dots, x_N)^T$ with their corresponding values $t \equiv (t_1, \dots, t_N)^T$, predict $\hat{t}$ for any $\hat{x}$ – that is, estimate the generative model.

Fitting a Mth order polynomial: $y(x, w) = w_0 + w_1 x + \cdots + w_M x^M = \sum_{j=0}^{M} w_j x^j$

This is a nonlinear function of x but is a linear function of the coefficients w that we need to estimate (hence it is called linear regression).

$$w^* = argmin_w E(w) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n, w) - t_n\}^2$$

And then take the derivate, equate to zero and so on.

But how do we choose N?

For polynomials of degree $N - 1$ we have a perfect match of N points, however, is that the best model?

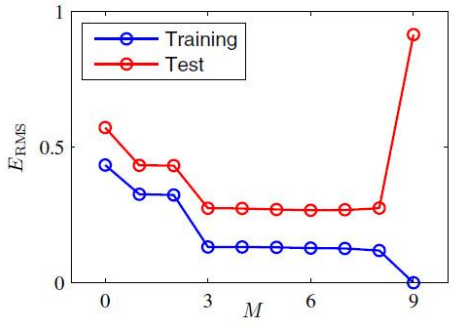
That's over-fitting! The perfect fit will be perfect for the N points, but there's no guarantee to how it will fit other points.

Cross-validation:

For instance, cut the data in half. Use one half for regression, and the other for testing, and vice-versa. That can also be done in leave-one-out N experiments.

We may want to minimize N that still achieves good results $E_{RMS} = \sqrt{\frac{2E(w^*)}{N}}$

Graphs of the root-mean-square error, defined by (1.3), evaluated on the training set and on an independent test set for various values of M.



Increase the amount of data:

If we have lots of data, we can have a better fit, but we have no control over this parameter most of the time.

Curve-fitting revisited

We have: $t = y(x, w) + \epsilon$

Where ϵ is discrepancy, and assume it has a Gaussian distribution, which would mean the likelihood is a Gaussian.

Likelihood:

$$p(t|x, w, \beta) = \mathcal{N}(t|y(x, w), \beta^{-1})$$

$\epsilon \sim \mathcal{N}(0, \beta^{-1})$ where $\beta^{-1} = \sigma^2 \Rightarrow \boxed{t \sim \mathcal{N}(y(x, w), \beta^{-1})}$ since $\epsilon = t - y(x, w) \sim \mathcal{N}(0, \beta^{-1})$

Therefore:

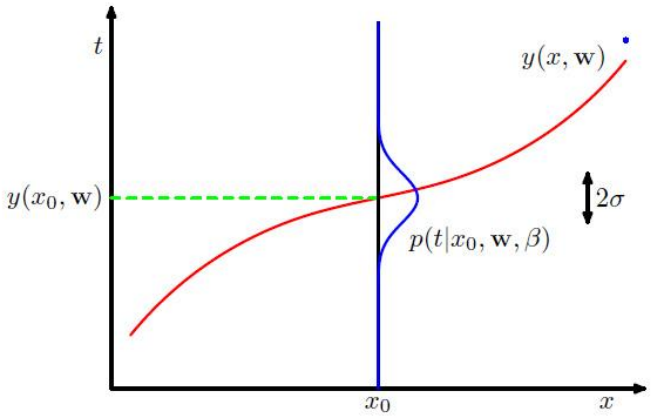
$$p(t|x, w, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n|y(x_n, w), \beta^{-1})$$

Take the log:

$$\ln p(t|x, w, \beta) = -\frac{\beta}{2}\sum_{n=1}^{N}\{y(x_n, w) - t_n\}^2 + \frac{N}{2}\ln\beta - \frac{N}{2}\ln(2\pi)$$

And $\boxed{w_{ML}}$ is the one that minimized the error function $E(w)$. Similarly: $\frac{1}{\beta_{ML}} = \frac{1}{N}\sum_{n=1}^{N}\{y(x_n, w) - t_n\}^2$

Figure 1.16 Schematic illustration of a Gaussian conditional distribution for t given x given by (1.60), in which the mean is given by the polynomial function $y(x, \mathbf{w})$, and the precision is given by the parameter β, which is related to the variance by $\beta^{-1} = \sigma^2$.



MAP curve fitting:

The likelihood is:

$$p(t|x, w, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n|y(x_n, w), \beta^{-1})$$

Assume a prior:

$$p(w|\alpha) = \mathcal{N}(w|0, \alpha^{-1}I) = \left(\frac{\alpha}{2\pi}\right)^{\frac{M+1}{2}} \exp\left[-\frac{\alpha}{2}w^T w\right]$$

Compute the posterior:

$$p(w|x, t, \alpha, \beta) \propto p(t|x, w, \beta)p(w|\alpha)$$

Maximize:

$$argmax_w p(w|x,t,\alpha,\beta) = argmin_w \frac{\beta}{2}\sum_{n=1}^{N}\{y(x_n,w)-t_n\}^2 + \frac{\alpha}{2}w^T w$$

This is the error function $E(w)$ with regularization (smoothing term) on w.

Linear Models for Regression

Linear basis function models

Let's generalize the simple function $y(x,w) = \sum_{j=0}^{D} w_j x^j$ in the polynomial case, to:

$$y(x,w) = \sum_{j=0}^{D} w_j \phi_j(x)$$

In the poly case $\phi_j(x) \coloneqq x^j$

Denote x_j as the basis, $[x_1, \dots, x_D]^T$ the bases. We can write the above as follows:

$$y(x,w) = [w_0 \dots w_D] \cdot \begin{bmatrix} \phi_0(x) \\ \vdots \\ \phi_D(x) \end{bmatrix}$$

And again, it is linear in the coefficients w we want to fit.

Examples

Gaussian: $\phi_j(x) = \exp\left[-\frac{(x-\mu_j)^2}{2s^2}\right]$

Sigmoidal basis function: $\phi_j(x) = \sigma\left(\frac{x-\mu_j}{s}\right)$ where σ is the logistic sigmoidal function $\sigma(a) = \frac{1}{1+\exp(-a)}$

Maximum Likelihood

Generative model: $t = y(x,w) + \epsilon$

Where ϵ is a zero mean Gaussian with precision β.

We do exactly the same – taking the logarithm and maximizing it, trying to minimize the squared difference. Eventually we get:

$$E_D(w) = \frac{1}{2}\sum_{n=1}^{N}\{t_n - w^T\phi(x_n)\}^2$$

Where

$$w_{ML} = argmin_w \frac{1}{2}\sum_{n=1}^{N}\{t_n - w^T\phi(x_n)\}^2$$

Where $w^T\phi(x_n) = [w_0 \dots w_D] \cdot \begin{bmatrix} \phi_0(x_n) \\ \vdots \\ \phi_D(x_n) \end{bmatrix}$

Solving it:

Take the derivative with respect to w and compare to 0:

$$\nabla \ln p(t|w,\beta) = \sum_{n=1}^{N} \{t_n - w^T \phi(x_n)\} \phi(x_n)^T = 0$$

Solution (this is discussed on page 142 in the [BISHOP 2006]):

$$\Leftrightarrow \sum_{n=1}^{N} [t_n \phi(x_n)^T - w^T \phi(x_n) \phi(x_n)^T] = 0 \Leftrightarrow \cdots ???$$

And eventually we get:

$w_{ML} = (\Phi^T \Phi)^{-1} \Phi^T t = \Phi^? t$ (? Is the cross-like symbol)

Where Φ is the design matrix:

$$\Phi = \begin{bmatrix} \phi_0(x_1) & \phi_1(x_1) & \dots & \phi_{M-1}(x_1) \\ \phi_0(x_2) & \phi_1(x_2) & \dots & \phi_{M-1}(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(x_N) & \phi_1(x_N) & \dots & \phi_{M-1}(x_N) \end{bmatrix}$$

Multiple Outputs

When we have K target variables $t = (t_1, \dots, t_K)^T$ for each input $x = (x_1, \dots, x_D)^T$ we use the same set of basis functions to model all of the components of the target vector: $y(x, w) = W^T \phi(x)$

… (Skipped this phase because it is similar to before)

Bayesian linear regression

Likelihood:

$$p(t|w,\beta) = \prod_{n=1}^{N} \mathcal{N}(t_n | w^T \phi(x_n), \beta^{-1})$$

Prior:

$$p(w) = \mathcal{N}(w|m_0, S_0)$$

Posterior:

…

Bayes' theorem for Gaussian variables

$$p(x) = \mathcal{N}(x|\mu, \Lambda^1), p(y|x) = \mathcal{N}(y|\Lambda x + b, L^{-1})$$

Consider the joint distribution $p(z)$ where z is the stacked vector $\binom{x}{y}$.

…

Once we find $cov[z], E[z]$ we can find $p(y)$ the marginal dist. And conditional dist. – the posterior: $p(x|y)$

Then we just plug it in: we have $p(w), p(t|w)$ so we can get $p(w|t)$ – the posterior we wanted.

Where:

$$p(w|t) = \mathcal{N}(w|m_N, S_N)$$

Isotropic prior

$p(w|\alpha) = \mathcal{N}(w|0, \alpha^{-1} I)$ – in the slides.

Predictive Distribution

Directly make predictions of t for new values of x:

$p(t|\boldsymbol{t},\alpha,\beta) = \int p(t|w,\beta)p(w|\boldsymbol{t},\alpha,\beta)dw$ (Integral of the likelihood times the posterior)

Equivalent Kernel

Posterior:

$p(w|t) = \mathcal{N}(w|m_N, S_N)$

Where:

$m_N = S_N(S_0^{-1}m_0 + \beta\Phi^T\boldsymbol{t})$

$S_N^{-1} = S_0^{-1} + \beta\Phi^T\Phi$