Lecture Outline

Reading: Chapter 1

- Curve Fitting Reformulated
- Maximum Likelihood Solution for Model Coefficients
- Maximum A Posterior Solution for Model Coefficients

Curve Fitting Reviewed

Here we return to the curve fitting example and view it from a probabilistic perspective. We will connect the solution via ML to the squared error function and solution via MAP to the regularized squared error function.

The goal in the curve fitting problem is to be able to make predictions for the target variable t given some new value of the input variable x on the basis of a set of training data comprising N input values $\mathtt{x} = (x_1, \ldots, x_N)$ and their corresponding target values $\mathtt{t} = (t_1, \ldots, t_N)$.

Figure 1: Block diagrams for training and test stages

In our original problem forumation, we used the Least-Squares (LS) cost function

$$E(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\left[y(x_n, \mathbf{w}) - t_n\right]^2. \tag{1}$$

and later the LS cost function with regularization

$$\tilde{E}(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\left[y(x, \mathbf{w}) - t_n\right]^2 + \frac{\lambda}{2}||\mathbf{w}||^2 \tag{2}$$

which allowed us to conveniently solve for $\mathbf{w}^*$. This was then used to make predictions, $y(x, \mathbf{w}^*)$.

1.2.5 Curve Fitting Reformulated

We now would like to reformulate the problem in a way that allows us to not only solve for $\mathbf{w}^*$, but to also express the uncertainty over the value of the predicted target variable using a probability distribution. This reformulation leads to a much more powerful way of model-building and prediction.

Figure 2: Block diagram for predictive distribution

For this purpose, we shall assume that, given the value of x, the corresponding value of t has a Gaussian distribution with a mean equal to the value of $y(x, \mathbf{w})$ of the polynomial and variance equal to β^{-1}. Thus

$$p(t|x, \mathbf{w}, \beta) = \mathcal{N}(t|\underbrace{y(x, \mathbf{w})}_{\mu}, \underbrace{\beta^{-1}}_{\sigma^2}) \tag{3}$$

where the precision β is the inverse variance of the distribution. See figure below. How we actually predict t from $p(t|x, \mathbf{w}, \beta)$ is the subject of decision theory.

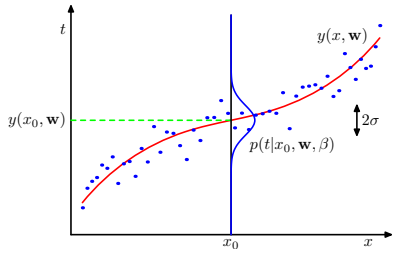

Figure 3: (Figure 1-16)

Maximum Likelihood Solution

We now use the training data $\{\mathsf{x}, \mathsf{t}\}$ to determine the values of the unknown parameters $\mathbf{w}$ and β by ML. Assuming the data are i.i.d., the likelihood function is

$$p(\mathsf{t}|\mathsf{x}, \mathbf{w}, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n|y(x_n, \mathbf{w}), \beta^{-1}). \tag{4}$$

The log-likelihood function is given by

$$\ln p(\mathsf{t}|\mathsf{x}, \mathbf{w}, \beta) = \frac{-\beta}{2} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}) - t_n)\}^2 + \frac{N}{2} \ln \beta - \frac{N}{2} \ln(2\pi). \tag{5}$$

Step 1: solve for $\mathbf{w}_{\mathrm{ML}}$. Consider first the determination of the ML solution for the polynomial coefficients $\mathbf{w}_{\mathrm{ML}}$. We maximize (5) with respect to $\mathbf{w}$. Maximizing (5) is equivalent to minimizing

$$\frac{1}{2} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}) - t_n)\}^2 \tag{6}$$

since the last two terms in (5) do not depend on $\mathbf{w}$, scaling by $\beta > 0$ does alter the location of the maximum, and maximizing the log is equivalent to minimizing the negative log. We see that solving for the polynomial

coefficients $\mathbf{w}$ via ML is equivalent to minimizing the sum-of-squares error function (like we did earlier but heuristically!).

Step 2: solve for β. We maximize (5) with respect to β. Taking the derivative of (5) w.r.t. β and setting equal to zero yields

$$0 = -\frac{1}{2}\sum_{n=1}^{N}\{y(x_n, \mathbf{w}_{\mathrm{ML}}) - t_n)\}^2 + \frac{N}{2\beta} \tag{7}$$

which gives the solution

$$\frac{1}{\beta_{\mathrm{ML}}} = \frac{1}{N}\sum_{n=1}^{N}\{y(x_n, \mathbf{w}_{\mathrm{ML}}) - t_n)\}^2. \tag{8}$$

Having now determined the parameters $\mathbf{w}_{\mathrm{ML}}$ and β_{ML}, we can now make predictions of t for new values of x and quantify our uncertainty in predicting t. Because we now have a probabilistic model, these are expressed in terms of the *predictive distribution* that gives the probability distribution over t, rather than simply a point estimate. The probability distribution is then

$$p(t|x, \mathbf{w}_{\mathrm{ML}}, \beta_{\mathrm{ML}}) = \mathcal{N}(t|y(x, \mathbf{w}_{\mathrm{ML}}), \beta_{\mathrm{ML}}^{-1}). \tag{9}$$

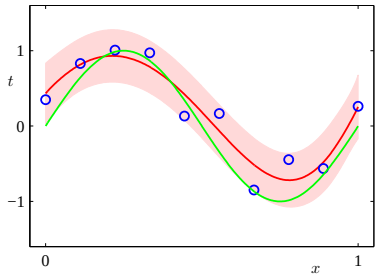

Figure 4: (Figure 1-17)

Note that one way to use the predictive distribution, $p(t|x, \mathbf{w}_{\mathrm{ML}}, \beta_{\mathrm{ML}})$ to make a prediction is to compute the conditional mean, $\mathbb{E}\left[t|x, \mathbf{w}_{\mathrm{ML}}, \beta_{\mathrm{ML}}\right]$ which for this example is $y(x, \mathbf{w}_{\mathrm{ML}})$ [see (3)].

Maximum Posterior Solution (Bayesian Curve Fitting)

Now let us take a step towards a more Bayesian approach and introduce a prior distribution over the polynomial coefficients (ML solution did not use a prior on $\mathbf{w}$). This prior allows us to "tighten" the value range for $\mathbf{w}$. For simplicity, we assume the prior has a Gaussian distribution of the form

$$\begin{aligned} p(\mathbf{w}|\alpha) &= \mathcal{N}(\mathbf{w}|\mathbf{0}, \alpha^{-1}\mathbf{I}) \\ &= \frac{1}{(2\pi)^{(M+1)/2}|\alpha^{-1}\mathbf{I}|^{1/2)}} \exp\left\{-\frac{1}{2}(\mathbf{w}-\mathbf{0})^T(\alpha^{-1}\mathbf{I})^{-1}(\mathbf{w}-\mathbf{0})\right\} \\ &= \left(\frac{\alpha}{2\pi}\right)^{(M+1)/2} \exp\left\{-\frac{\alpha}{2}\mathbf{w}^T\mathbf{w}\right\} \end{aligned} \tag{10}$$

where α (hyperparameter) is the precision of this distribution and $\mathbf{w}$ is an $(M+1)\times 1$ random vector. Using Bayes' theorem the posterior distribution is given by

$$p(\mathbf{w}|\mathsf{x}, \mathsf{t}, \alpha, \beta) \propto p(\mathsf{t}|\mathsf{x}, \mathbf{w}, \beta)p(\mathbf{w}|\alpha). \tag{11}$$

We can now determine $\mathbf{w}$ by finding the most probable value of $\mathbf{w}$ given the data, in other words by maximizing the *posterior* distribution. This technique is called *maximum a posterior* (MAP). We can show that the maximum of the posterior is given by the minimum of

$$\frac{\beta}{2}\sum_{n=1}^{N}\{y(x_n,\mathbf{w})-t_n)\}^2+\frac{\alpha}{2}\mathbf{w}^T\mathbf{w}. \tag{12}$$

We see that solving for the polynomial coefficients $\mathbf{w}$ via MAP is equivalent to minimizing the regularized sum-of-squares error function with $\lambda=\alpha/\beta$. As we will see later the values of $\alpha,\ \beta$ can be inferred from the data.

The result of the MAP solution is the predictive distribution

$$p(t|x,\mathsf{x},\mathsf{t}) \quad = \quad \mathcal{N}(t|m(x),s^2(x)) \tag{13}$$

where the mean and variance are given by

$$\begin{aligned} m(x) &= \beta\boldsymbol{\phi}(x)^{\mathrm{T}}\mathbf{S}\sum_{n=1}^{N}\boldsymbol{\phi}(x_n)t_n \\ s^2(x) &= \underbrace{\beta^{-1}}_{\text{Term1}}+\underbrace{\boldsymbol{\phi}(x)^{\mathrm{T}}\mathbf{S}\boldsymbol{\phi}(x)}_{\text{Term2}} \end{aligned} \tag{14}$$

with

$$\mathbf{S}^{-1} \quad = \quad \alpha\mathbf{I}+\beta\sum_{n=1}^{N}\boldsymbol{\phi}(x_n)\boldsymbol{\phi}(x_n)^{\mathrm{T}}. \tag{15}$$

Term 1 uncertainty in predicted t due to noise on targets in training data [see (9)].

Term 2 uncertainty in parameters $\mathbf{w}$ and is a consequence of Bayesian treatment.

Note that one way to use the predictive distribution, $p(t|x,\mathsf{x},\mathsf{t})$ to make a prediction is to compute $\mathbb{E}\left[t|x,\mathsf{x},\mathsf{t}\right]$ which for this example is $m(x)$.

1.3 Model Selection

If data is plentiful, then one approach is simply to use some of the available data to train a range of models, or a given model with a range of values for its complexity parameters, and then to compare them on independent data, sometimes called a *validation set* and select the one having the best predictive performance. If the model is iterated many times using a limited size data set, then some over-fitting to the validation set can occur and so it may be necessary to keep aside a third *test set* on which the performance of the selected model is finally evaluated.

In many applications, however, the supply of data for training and testing will be limited, an din order to build good models, we wish to use as much of the available data as possible for training. However, if the evaluation set is small, it will give a relatively noisy estimate of predictive performance. One solution to this dilemma is to use *cross-validation*.

This allows a proportion $(S-1)/S$ of the available data to be used for training while making use of all of the data to assess performance. When data is particularly scarce, it may be appropriate to consider the case $S=N$, where N is the total number of data points, which gives the *leave-one-out* technique. One major drawback of cross-validation is that the number of training runs that must be performed is increased by a factor of S.

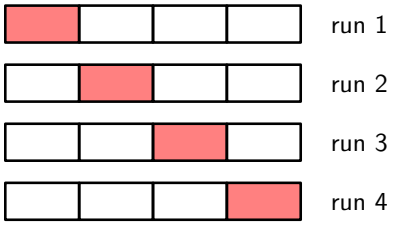

Figure 5: (Figure 1-18)

Example: If we choose $S = 10$ then we would use 90% of the data to train on and 10% to test with. In this case we could cycle over all 10 subsets of the data partitions for *10-fold cross-validation*.

Example: If we have $N = 10$ data points and chose S as in the above example, we would train on 9 out of the 10 data points and test on the one data point we held out. We would repeat this by cycling over all possible hold out data points.

1.4 The Curse of Dimensionality

When building a classifier, we have to partition the input space into regular cells. When we are given a test point and we wish to predict its class, we first decide which cell it belongs to, and we then find all of the training points that fall into the same cell. The identity of the test point is predicted as being in the same class having the largest number of training points in the same cell as the test point.

The problem with this approach is when the input has a high dimension and consequently the input space has a high dimensionality. If we divide the region into regular cells, then the number of cells grows exponentially with the dimensionality of the space. The problem with an exponentially large number of cells is that we would need an exponentially large quantity of training data in order to ensure that the cells are not empty. This severe difficulty that can arise in spaces of many dimensions is sometimes called the *curse of dimensionality*.

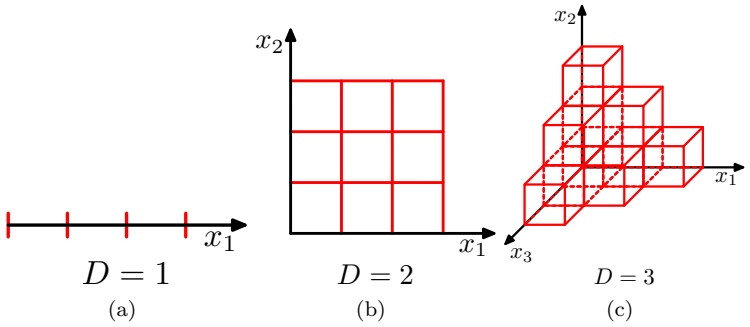

Figure 6: (Figure 1-21)

1.5 Inference and Decision

Students should read this section on their own.