

## Appendix C. Matrix Derivatives

### Chapter 3 Linear Models for Regression

#### Definition. *Introduction*

1. **Problem** Given training set  $\{x_n\}$ , where  $n = 1, \dots, N$  and corresponding target values  $\{t_n\}$ , the goal is to predict value of  $t$  for a new value of  $\mathbf{x}$ .
2. **Approach** Construct a function  $y(\mathbf{x})$  whose value is the prediction for some  $\mathbf{x}$ . More generally, we aim to model the predictive distribution  $p(t|\mathbf{x})$  as it represent uncertainty of  $t$  given  $\mathbf{x}$ . We want to find a  $y(\mathbf{x})$  in such a way so as to minimize the expected loss of a chosen loss function (i.e. for squared loss,  $y(\mathbf{x}) = \mathbb{E}_t \{t|\mathbf{x}\}$ , the mean of predictive distribution)

#### 3.1 Linear Basis Function Models

##### Definition. *Linear Basis Function*

1. **Model** A linear model with respect to parameters  $w_0, \dots, w_D$

$$y(\mathbf{x}, \mathbf{w}) = \sum_{j=0}^{M-1} w_j \phi_j(\mathbf{x}) = \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}) \quad \phi_0(\mathbf{x}) = 1$$

where  $\phi_j(\mathbf{x})$  are basis functions and  $M$  is the total number of parameters in the model, including bias.  $\mathbf{w} = (w_0, \dots, w_{M-1})^T$  and  $\boldsymbol{\phi} = (\phi_0, \dots, \phi_{M-1})^T$ . Here is some basis functions

$$\phi_j(x) = x^j \text{ (polynomial)} \quad \phi_j(x) = \exp -\frac{(x - \mu_j)^2}{2s^2} \text{ (gaussian)}$$

$$\phi_j(x) = \sigma\left(\frac{x - \mu_j}{s}\right) \quad \sigma(x) = \frac{1}{1 + e^{-x}} \text{ (logistic)}$$

**Definition. Maximum Likelihood and Least Squares** Here we show that fitting a linear basis function by minimizing a sum-of-square error function can be motivated as the maximum likelihood solution under an assumed Gaussian noise model. Let target variable determined by  $y(\mathbf{w}, \mathbf{x})$  with Gaussian noise

$$t = y(\mathbf{x}, \mathbf{w}) + \epsilon$$

We can model the predictive distribution with

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

For squared loss, the optimal prediction for new value of  $\mathbf{x}$  is given by conditional mean of target variable,

$$\mathbb{E} \{t|\mathbf{x}\} = \int t p(t|\mathbf{x}) dt = y(\mathbf{x}, \mathbf{w}) \quad \text{i.e.} \quad \mathbb{E} \{p(t|\mathbf{x}, \mathbf{w}, \beta)\} = y(\mathbf{x}, \mathbf{w})$$

Let  $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$  with target  $\mathbf{t} = \{t_1, \dots, t_N\}$ , we get likelihood function

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

Now we try to minimize likelihood

$$\ln p(\mathbf{t}|\mathbf{w}, \beta) = \sum_n \ln \mathcal{N}(t_n | \mathbf{w}^T \phi(\mathbf{x}_n), \beta^{-1}) = \frac{N}{2} \ln \beta - \frac{N}{2} \ln 2\pi - \beta E_D(\mathbf{w})$$

where sum of squared error function is defined by

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

Now we find maximum likelihood estimator for  $\mathbf{w}$  and  $\beta$ ,

$$\frac{\partial}{\partial \mathbf{w}} = \sum_{n=1}^N (t_n - \mathbf{w}^T \phi(\mathbf{x}_n)) \phi(\mathbf{x}_n)^T \quad \rightarrow \quad \sum_{n=1}^N t_n \phi(\mathbf{x}_n)^T = \mathbf{w}^T \left( \sum_{n=1}^N \phi(\mathbf{x}_n) \phi(\mathbf{x}_n)^T \right)$$

$$\mathbf{w}_{mle} = (\Phi^T \Phi)^{-1} \Phi^T \mathbf{t}$$

where  $\Phi$  is called the design matrix

$$\Phi = \begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \cdots & \phi_{M-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & \cdots & \phi_{M-1}(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & \cdots & \phi_{M-1}(\mathbf{x}_N) \end{pmatrix}$$

### 3.13 Sequential Learning

**Definition. Sequential/On-line Learning** Instead of processing the entire training set in one go, sequential algorithms consider each data points one at a time. **Stochastic gradient descent** is an online algorithm. Given error function as a sum over errors of all training samples, i.e.  $E = \sum_n E_n$ ,  $w$  is updated according to

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta \nabla E_n$$

where  $\tau$  is iteration number and  $\eta$  is a learning rate parameter. For sum-of-squares error function, this gives

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta (t_n - \mathbf{w}^{(\tau)T} \phi_n) \phi_n \quad \phi_n = \phi(\mathbf{x}_n)$$

**Definition. Regularized Least Squares** Adding regularization term to an error function to control over-fitting,

$$E_D(\mathbf{w}) + \lambda E_W(\mathbf{w})$$

In case of sum-of-squares error with sum-of-squares regularizer

$$\frac{1}{2} \sum_{n=1}^N (t_n - \mathbf{w}^T \phi(\mathbf{x}_n))^2 + \frac{\lambda}{2} \mathbf{w}^T \mathbf{w}$$

The particular choice is called **weight decay**, an example of parameter shrinkage method because it shrinks parameter to 0. In this case, maximum likelihood estimator has closed form

$$\mathbf{w} = (\lambda \mathbf{I} + \Phi^T \Phi)^{-1} \Phi^T \mathbf{t}$$

A more general regularizer has form

$$\frac{1}{2} \sum_{n=1}^N (t_n - \mathbf{w}^T \phi(\mathbf{x}_n))^2 + \frac{\lambda}{2} \sum_{j=1}^M |w_j|^q$$

## 3.2 The Bias-Variance Decomposition

### 1.1.5 Background on Decision Theory

**Definition. Bias-Variance Decomposition** Decision theory for regression consists of choosing a particular estimate  $y(\mathbf{x})$  of value of  $t$  for each input  $\mathbf{x}$  such that the expected loss is minimized. For squared loss function, the optimal prediction  $y(\mathbf{x})$  is given by

$$h(\mathbf{x}) = \mathbb{E} \{t|\mathbf{x}\} = \int t p(t|\mathbf{x}) dt$$

We can rewrite the expected loss as

$$\mathbb{E} \{L\} = \int (y(\mathbf{x}) - h(\mathbf{x})) d\mathbf{x} + \int (h(\mathbf{x}) - t)^2 p(\mathbf{x}) d\mathbf{x}$$

where the first term is minimized if  $y(\mathbf{x}) = \mathbb{E} \{t|\mathbf{x}\}$  and the second term, independent of  $y(\mathbf{x})$  is the variance of the distribution of  $t$  averaged over  $\mathbf{x}$ , representing the intrinsic variability of the data. We can decompose expected loss in terms of bias, variance, and noise

$$\mathbb{E} \{L\} = \int (\mathbb{E}_{\mathcal{D}} \{y(\mathbf{x}; \mathcal{D})\} - h(\mathbf{x}))^2 p(\mathbf{x}) d\mathbf{x} + \int \mathbb{E}_{\mathcal{D}} \{y(\mathbf{x}; \mathcal{D}) - \mathbb{E}_{\mathcal{D}} \{y(\mathbf{x}; \mathcal{D})\}\}^2 p(\mathbf{x}) d\mathbf{x} + \int (h(\mathbf{x}) - t)^2 p(\mathbf{x}, t) d\mathbf{x} dt$$

given a particular dataset  $\mathcal{D}$ . The first term is **squared bias**, represents extend to which average prediction over all data sets differs from the desired regression function. The second term, called **variance**, measures the extend to which solutions for individual datasets vary around their average (average of  $y(\mathbf{x}; \mathcal{D})$ s), hence measures the extent to which  $y(\mathbf{x}; \mathcal{D})$  is sensitive to a particular choice of dataset.

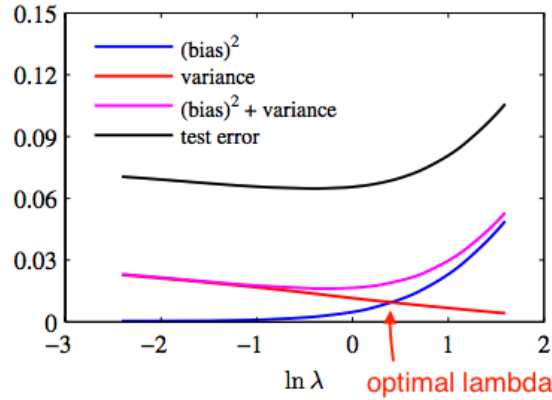
As an example, assuming we sample  $L$  datasets and fit a model by minimizing regularized squared error function to give  $L$  prediction function  $y^{(l)}(x)$ . The average prediction is estimated by

$$\bar{y}(x) = \frac{1}{L} \sum_{l=1}^L y^{(l)}(x)$$

and the integrated squared bias and integrated variance is approximated by finite sums

$$\begin{aligned} (\text{bias})^2 &= \frac{1}{N} \sum_{n=1}^N (\bar{y}(x_n) - h(x_n))^2 \\ \text{variance} &= \frac{1}{N} \sum_{n=1}^N \frac{1}{L} \sum_{l=1}^L (y^{(l)}(x_n) - \bar{y}(x_n))^2 \end{aligned}$$

Note values of these terms depend on choice of regularization parameter, changes to  $\lambda$  adjusts for the bias-variance tradeoff.



**Definition. Convex Function** A function  $f$  is convex if and only if for all  $\theta_1, \theta_2$ , and for all  $\alpha \in [0, 1]$ , we have

$$f(\alpha\theta_1 + (1 - \alpha)\theta_2) \leq \alpha f(\theta_1) + (1 - \alpha)f(\theta_2)$$

**Definition. Newton's Method** is a method for finding successively better approximations to the roots of a real-valued function. The algorithm starts with a function  $f$ , the derivative  $f'$  and the initial guess  $x_0$  for the root of  $f$ , a better approximation is given by updating  $x_n$  until convergence

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

$(x_n, 0)$  is the intersection of  $x$ -axis and the tangent of the graph of  $f$  at  $(x_n, f(x_n))$ . Note the equation of tangent line is given by

$$y = f'(x_n)(x - x_n) + f(x_n) \quad \rightarrow \quad 0 = f'(x_n)(x_{n+1} - x_n) + f(x_n)$$

We can use Newton's Method to find a minimum or maximum of a function  $f$  by applying Newton's method to the derivative

$$x_{n+1} = x_n - \frac{f'(x_n)}{f''(x_n)}$$

In the multivariate case, we have

$$y = f(\mathbf{x} + \Delta \mathbf{x}) \approx f(\mathbf{x}) + \nabla f(\mathbf{x})^T \Delta \mathbf{x} + \frac{1}{2} \Delta \mathbf{x}^T \mathbf{H}(\mathbf{x}) \Delta \mathbf{x}$$

Take the gradient

$$\nabla_{\Delta \mathbf{x}} f(\mathbf{x} + \Delta \mathbf{x}) \approx \nabla f(\mathbf{x}) + \mathbf{H} \Delta \mathbf{x}$$

Setting gradient to zero provides the Newton step

$$\Delta \mathbf{x} = -H^{-1}y(\mathbf{x}) \quad \rightarrow \quad \mathbf{x}_{n+1} = \mathbf{x}_n + \Delta \mathbf{x}$$

Computing and storing hessian matrix takes  $\Theta(n^2)$  memory, infeasible for high dimensional functions.

1. **Quasi-Newton Method** Replaces the exact Hessian with an approximation.

**Definition. Stochastic Gradient Descent** is a stochastic approximation of the gradient descent optimization and iterative method for minimizing an objective function that is written as a sum of differentiable functions. Sum-minimization problems arise in least squares and in maximum likelihood estimation

$$Q(w) = \frac{1}{n} \sum_{i=1}^N Q_i(w)$$

where we want to estimate  $w$  such that  $Q(w)$  is minimized. A batch gradient descent updates weight with

$$w = w - \eta \nabla Q(w) = w - \frac{\eta}{N} \sum_{i=1}^N \nabla Q_i(w)$$

The idea is that the summand functions have a simple form that enables inexpensive evaluation of sum and gradient operations for objective function with a good form. In stochastic gradient descent, the gradient  $Q(w)$  is approximated by a gradient at a single example

$$w = w - \eta \nabla Q_i(w)$$

which performs update for each training example, in multiple epochs, such that the algorithm converges. To choose an appropriate step size by picking a sequence of  $\eta_t$  such that

$$\sum_t \eta_t = \infty \quad \sum_t \eta_t^2 < \infty$$

satisfied by  $\eta_T \propto 1/t$

### 3.3 Bayesian Linear Regression

definition of  $p(\mathbf{t}|\mathbf{w})$  as product of independent gaussians with  $\mathbf{w}^T \phi(\mathbf{x}_n)$  as mean  
Formula for marginal and Conditional Gaussian in determining Gaussian posterior

**Definition. Parameter Distribution** Bayesian treatment of linear regression, Given a zero-mean isotropic ( $\Sigma = \sigma \mathbf{I}$ ) Gaussian prior and likelihood

$$p(\mathbf{w}|\alpha) = \mathcal{N}(\mathbf{w}|\mathbf{0}, \alpha^{-1}\mathbf{I}) \quad p(\mathbf{t}|\mathbf{w}) = \prod_{n=1}^N \mathcal{N}(t_n|\mathbf{w}^T \phi(\mathbf{x}_n), \beta^{-1})$$

then we have posterior

$$p(\mathbf{w}|\mathbf{t}) = \mathcal{N}(\mathbf{w}|\mathbf{m}_N, \mathbf{S}_N) \quad \mathbf{m}_N = \beta \mathbf{S}_N \Phi^T \mathbf{t} \quad \mathbf{S}_N^{-1} = \alpha \mathbf{I} + \beta \Phi^T \Phi$$

Maximization of posterior distribution of log likelihood with respect to  $\mathbf{w}$  is equivalent to minimization of sum-of-squares error function with the addition of a quadratic regularization term with  $\lambda = \alpha/\beta$

$$\ln p(\mathbf{w}|\mathbf{t}) = -\frac{\beta}{2} \sum_{n=1}^N (t_n - \mathbf{w}^T \Phi(\mathbf{x}_n))^2 + \frac{\alpha}{2} \mathbf{w}^T \mathbf{w}$$

#### 3.3.2 Predictive Distribution

**Definition. Posterior Predictive Distribution** [wiki](#) Posterior predictive distribution in gaussian linear model is given by marginalizing distribution of prediction  $t$  over parameters  $\mathbf{w}$  over posterior distribution of estimated parameters given training dataset  $\mathbf{w}, \mathbf{t}$ .

$$p(t|\mathbf{t}, \alpha, \beta) = \int p(t|\mathbf{w}, \beta) p(\mathbf{w}|\mathbf{t}, \alpha, \beta) d\mathbf{w}$$

where integrands refer to [3.8](#) and [3.49](#). Note the posterior predictive distribution is also a gaussian distribution.

### 3.4 Bayesian Model Comparison

#### 3.6 Limitations of Fixed Basis Functions