Linear regression

Sep 2022

Murphy chap7

Linear regression - MSE

Typically we have a set of training data $(x_1, y_1) \ldots (x_N, y_N)$ from which to estimate the parameters β. Each $x_i = (x_{i1}, x_{i2}, \ldots, x_{ip})^T$ is a vector of feature measurements for the ith case. The most popular estimation method is *least squares*, in which we pick the coefficients $\beta = (\beta_0, \beta_1, \ldots, \beta_p)^T$ to minimize the residual sum of squares

$$\begin{aligned} \mathrm{RSS}(\beta) &= \sum_{i=1}^{N} (y_i - f(x_i))^2 \\ &= \sum_{i=1}^{N} \Big(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \Big)^2. \end{aligned} \tag{3.2}$$

How do we minimize (3.2)? Denote by $\mathbf{X}$ the $N \times (p+1)$ matrix with each row an input vector (with a 1 in the first position), and similarly let $\mathbf{y}$ be the N-vector of outputs in the training set. Then we can write the residual sum-of-squares as

$$\mathrm{RSS}(\beta) = (\mathbf{y} - \mathbf{X}\beta)^T (\mathbf{y} - \mathbf{X}\beta). \tag{3.3}$$

Linear regression - MSE

$$\mathrm{RSS}(\beta) = (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta). \tag{3.3}$$

This is a quadratic function in the $p + 1$ parameters. Differentiating with respect to β we obtain

$$\begin{aligned} \frac{\partial \mathrm{RSS}}{\partial \beta} &= -2\mathbf{X}^T(\mathbf{y} - \mathbf{X}\beta) \\ \frac{\partial^2 \mathrm{RSS}}{\partial \beta \partial \beta^T} &= 2\mathbf{X}^T\mathbf{X}. \end{aligned} \tag{3.4}$$

Assuming (for the moment) that $\mathbf{X}$ has full column rank, and hence $\mathbf{X}^T\mathbf{X}$ is positive definite, we set the first derivative to zero

$$\mathbf{X}^T(\mathbf{y} - \mathbf{X}\beta) = 0 \tag{3.5}$$

to obtain the unique solution

$$\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}. \tag{3.6}$$

Linear regression - MSE

We denote the column vectors of $\mathbf{X}$ by $\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_p$, with $\mathbf{x}_0 \equiv 1$.

These vectors span a subspace of $\mathbb{R}^N$, also referred to as the column space of $\mathbf{X}$.

We minimize $\text{RSS}(\beta) = \|\mathbf{y} - \mathbf{X}\beta\|^2$ by choosing $\hat{\beta}$ so that the residual vector $\mathbf{y} - \hat{\mathbf{y}}$ is orthogonal to this subspace.

$$\mathbf{X}^T(\mathbf{y} - \mathbf{X}\beta) = 0$$

Linear regression - Geometry

We seek a vector $\hat{\mathbf{y}} \in \mathbb{R}^N$ that lies in this linear subspace and is as close as possible to $\mathbf{y}$, i.e., we want to find

$$\underset{\hat{\mathbf{y}} \in \text{span}(\{\tilde{\mathbf{x}}_1, \ldots, \tilde{\mathbf{x}}_D\})}{\text{argmin}} \|\mathbf{y} - \hat{\mathbf{y}}\|_2. \tag{7.18}$$

Since $\hat{\mathbf{y}} \in \text{span}(\mathbf{X})$, there exists some weight vector $\mathbf{w}$ such that

$$\hat{\mathbf{y}} = w_1 \tilde{\mathbf{x}}_1 + \cdots + w_D \tilde{\mathbf{x}}_D = \mathbf{X}\mathbf{w} \tag{7.19}$$

To minimize the norm of the residual, $\mathbf{y} - \hat{\mathbf{y}}$, we want the residual vector to be orthogonal to every column of $\mathbf{X}$, so $\tilde{\mathbf{x}}_j^T(\mathbf{y} - \hat{\mathbf{y}}) = 0$ for $j = 1 : D$. Hence

$$\tilde{\mathbf{x}}_j^T(\mathbf{y} - \hat{\mathbf{y}}) = 0 \Rightarrow \mathbf{X}^T(\mathbf{y} - \mathbf{X}\mathbf{w}) = \mathbf{0} \Rightarrow \mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \tag{7.20}$$

Hence our projected value of $\mathbf{y}$ is given by

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\mathbf{w}} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \tag{7.21}$$

This corresponds to an **orthogonal projection** of $\mathbf{y}$ onto the column space of $\mathbf{X}$. The projection matrix $\mathbf{P} \triangleq \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ is called the **hat matrix**, since it "puts the hat on $\mathbf{y}$".

Linear regression - Probability

Linear regression

One of the most widely used models for regression is known as **linear regression**. This asserts that the response is a linear function of the inputs. This can be written as follows:

$$y(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + \epsilon = \sum_{j=1}^{D} w_j x_j + \epsilon \tag{1.4}$$

where $\mathbf{w}^T\mathbf{x}$ represents the inner or **scalar product** between the input vector $\mathbf{x}$ and the model's **weight vector** $\mathbf{w}$[7], and ϵ is the **residual error** between our linear predictions and the true response.

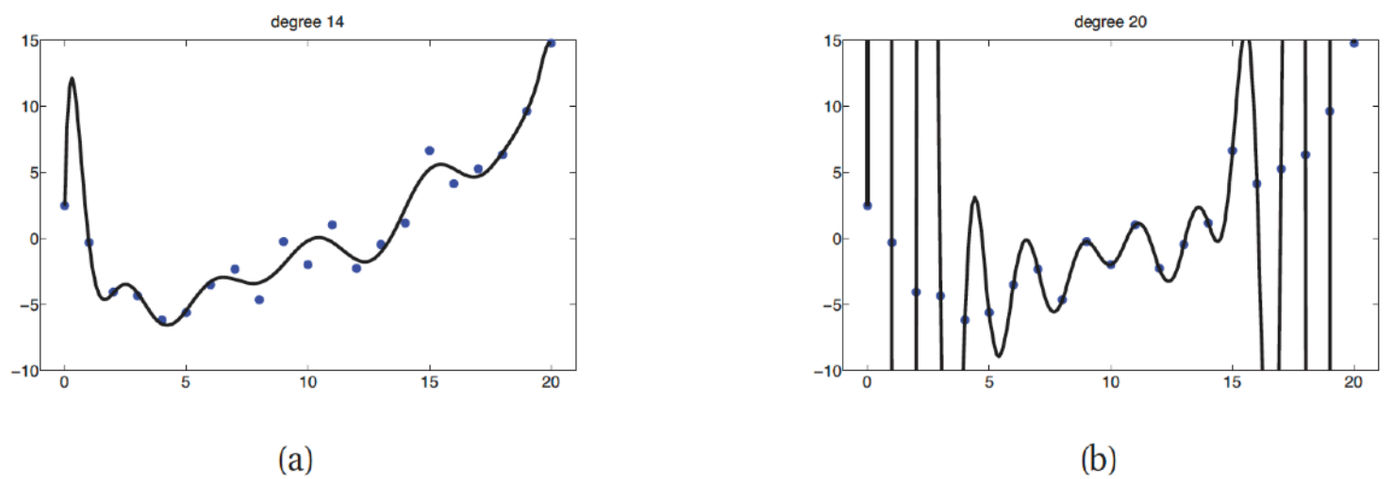


Figure 1.18 Polynomial of degrees 14 and 20 fit by least squares to 21 data points. Figure generated by `linregPolyVsDegree`.

Linear regression - Probability

To make the connection between linear regression and Gaussians more explicit, we can rewrite the model in the following form:

$$p(y|\mathbf{x}, \boldsymbol{\theta}) = \mathcal{N}(y|\mu(\mathbf{x}), \sigma^2(\mathbf{x})) \tag{1.5}$$

This makes it clear that the model is a conditional probability density. In the simplest case, we assume μ is a linear function of $\mathbf{x}$, so $\mu = \mathbf{w}^T\mathbf{x}$, and that the noise is fixed, $\sigma^2(x) = \sigma^2$. In this case, $\boldsymbol{\theta} = (\mathbf{w}, \sigma^2)$ are the parameters of the model.

For example, suppose the input is 1 dimensional. We can represent the expected response as follows:

$$\mu(\mathbf{x}) = w_0 + w_1 x = \mathbf{w}^T\mathbf{x} \tag{1.6}$$

where w_0 is the intercept or **bias** term, w_1 is the slope, and where we have defined the vector $\mathbf{x} = (1, x)$. (Prepending a constant 1 term to an input vector is a common notational trick which

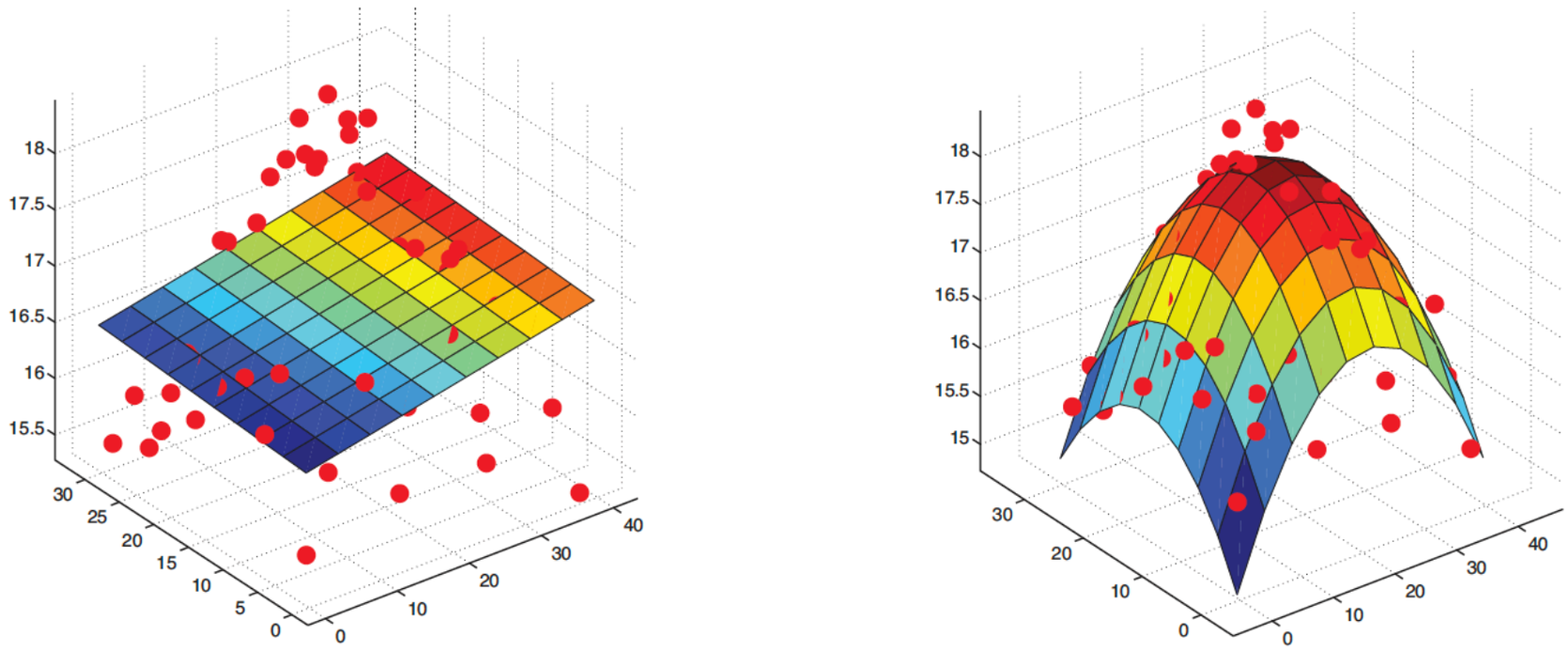
Linear regression - Probability

Linear regression can be made to model non-linear relationships by replacing $\mathbf{x}$ with some non-linear function of the inputs, $\phi(\mathbf{x})$. That is, we use

$$p(y|\mathbf{x}, \boldsymbol{\theta}) = \mathcal{N}(y|\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}), \sigma^2) \tag{1.7}$$

This is known as **basis function expansion**. For example, Figure 1.18 illustrates the case where $\phi(\mathbf{x}) = [1, x, x^2, \ldots, x^d]$, for $d = 14$ and $d = 20$; this is known as **polynomial regression**. We will consider other kinds of basis functions later in the book. In fact, many popular machine learning methods — such as support vector machines, neural networks, classification and regression trees, etc. — can be seen as just different ways of estimating basis functions from data, as we discuss in Chapters 14 and 16.

Linear regression - Probability



(a) (b)

Figure 7.1 Linear regression applied to 2d data. Vertical axis is temperature, horizontal axes are location within a room. Data was collected by some remote sensing motes at Intel's lab in Berkeley, CA (data courtesy of Romain Thibaux). (a) The fitted plane has the form $\hat{f}(\mathbf{x}) = w_0 + w_1 x_1 + w_2 x_2$. (b) Temperature data is fitted with a quadratic of the form $\hat{f}(\mathbf{x}) = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_1^2 + w_4 x_2^2$. Produced by `surfaceFitDemo`.

Linear regression - MLE

Maximum likelihood estimation (least squares)

A common way to esitmate the parameters of a statistical model is to compute the MLE, which is defined as

$$\hat{\boldsymbol{\theta}} \triangleq \arg\max_{\boldsymbol{\theta}} \log p(\mathcal{D}|\boldsymbol{\theta}) \tag{7.4}$$

It is common to assume the training examples are independent and identically distributed, commonly abbreviated to **iid**. This means we can write the log-likelihood as follows:

$$\ell(\boldsymbol{\theta}) \triangleq \log p(\mathcal{D}|\boldsymbol{\theta}) = \sum_{i=1}^{N} \log p(y_i|\mathbf{x}_i, \boldsymbol{\theta}) \tag{7.5}$$

Instead of maximizing the log-likelihood, we can equivalently minimize the **negative log likelihood** or **NLL**:

$$\text{NLL}(\boldsymbol{\theta}) \triangleq -\sum_{i=1}^{N} \log p(y_i|\mathbf{x}_i, \boldsymbol{\theta}) \tag{7.6}$$

The NLL formulation is sometimes more convenient, since many optimization software packages are designed to find the minima of functions, rather than maxima.

Linear regression - MLE

Now let us apply the method of MLE to the linear regression setting. Inserting the definition of the Gaussian into the above, we find that the log likelihood is given by

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^{N} \log \left[\left(\frac{1}{2\pi\sigma^2} \right)^{\frac{1}{2}} \exp \left(-\frac{1}{2\sigma^2} (y_i - \mathbf{w}^T \mathbf{x}_i)^2 \right) \right] \tag{7.7}$$

$$= \frac{-1}{2\sigma^2} RSS(\mathbf{w}) - \frac{N}{2} \log(2\pi\sigma^2) \tag{7.8}$$

RSS stands for **residual sum of squares** and is defined by

$$\text{RSS}(\mathbf{w}) \triangleq \sum_{i=1}^{N} (y_i - \mathbf{w}^T \mathbf{x}_i)^2 \tag{7.9}$$

The RSS is also called the **sum of squared errors**, or SSE, and SSE/N is called the **mean squared error** or **MSE**. It can also be written as the square of the ℓ_2 **norm** of the vector of

Assumption: fixed sigma, independence responses

Minimize sum of squared error <==> Maximize log likelihood

Gauss-Markov theorem

$$\hat{\beta} = \left(X^T X\right)^{-1} X^T y.$$

Theorem 1 (**Gaussian-Markov**). *Linear regression solution has the least variance among all unbiased linear estimators of β, i.e. if $\beta_1 = \alpha_1^T y$ is an unbiased estimator of β, then $Var(\beta_1) \geqslant Var(\beta)$ in the sense of matrix.*

Proof. Recall that the linear regression model is equivalent to

$$y = X\beta + \epsilon,\ \epsilon \sim \mathcal{N}\left(0, \sigma^2\right).$$

We may write $\alpha_1^T = \left(X^T X\right)^{-1} X^T + D$. D is a $(p+1) \times N$ matrix.

$$\begin{aligned} E(\beta_1) = E\left(\alpha_1^T y\right) &= E\left(\left(\left(X^T X\right)^{-1} X^T + D\right)(X\beta + \epsilon)\right) \\ &= \left(\left(X^T X\right)^{-1} X^T + D\right) X\beta = (1 + DX)\beta. \end{aligned}$$

Since β_1 is an unbiased estimator of β, it follows that

$$DX = 0.$$

Gauss-Markov theorem

Then

$$\begin{aligned} Var(\beta_1) &= Var\left(\alpha_1^T y\right) = \alpha_1^T Var(y)\alpha_1 \\ &= \sigma^2\Big(\left(X^TX\right)^{-1}X^T + D\Big)\Big(\left(X^TX\right)^{-1}X^T + D\Big)^T \\ &= \sigma^2\Big(\left(X^TX\right)^{-1}X^T + D\Big)\Big(X\left(XX^T\right)^{-1} + D^T\Big) \\ &= \sigma^2\Big(\left(XX^T\right)^{-1} + \left(X^TX\right)^{-1}X^TD^T + DX\left(XX^T\right)^{-1} + DD^T\Big) \\ &= \sigma^2\left(XX^T\right)^{-1} + \sigma^2 DD^T. \end{aligned}$$

$$\begin{aligned} Var\left(\hat{\beta}\right) = Var\left(\alpha_1^T y\right) &= \left(X^TX\right)^{-1}X^T Var(y)\Big(\left(X^TX\right)^{-1}X^T\Big)^T \\ &= \sigma^2\left(X^TX\right)^{-1}X^TX\left(XX^T\right)^{-1} = \left(XX^T\right)^{-1}. \end{aligned}$$

So

$Var(\beta_1) - Var\left(\hat{\beta}\right)$ is a nonnegative definite matrix.

Linear regression - Bayes

Assumption: fixed sigma, see 7.6.3 for general case

In linear regression, the likelihood is given by

$$p(\mathbf{y}|\mathbf{X}, \mathbf{w}, \mu, \sigma^2) = \mathcal{N}(\mathbf{y}|\mu + \mathbf{X}\mathbf{w}, \sigma^2\mathbf{I}_N) \tag{7.52}$$

$$\propto \exp\left(-\frac{1}{2\sigma^2}(\mathbf{y} - \mu\mathbf{1}_N - \mathbf{X}\mathbf{w})^T(\mathbf{y} - \mu\mathbf{1}_N - \mathbf{X}\mathbf{w})\right) \tag{7.53}$$

form $p(\mu) \propto 1$, and then integrate it out to get **Exercise**

$$p(\mathbf{y}|\mathbf{X}, \mathbf{w}, \sigma^2) \propto \exp\left(-\frac{1}{2\sigma^2}||\mathbf{y} - \overline{y}\mathbf{1}_N - \mathbf{X}\mathbf{w}||_2^2\right) \tag{7.54}$$

where $\overline{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$ is the empirical mean of the output. For notational simplicity, we shall assume the output has been centered, and write $\mathbf{y}$ for $\mathbf{y} - \overline{y}\mathbf{1}_N$.

Linear regression - Bayes

Exercise

$$p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \mathcal{N}(y|w_0 + \boldsymbol{w}^\mathsf{T}\boldsymbol{x}, \sigma^2)$$

$$\hat{\boldsymbol{w}} = (\mathbf{X}_c^\mathsf{T}\mathbf{X}_c)^{-1}\mathbf{X}_c^\mathsf{T}\boldsymbol{y}_c = \left[\sum_{i=1}^{N_\mathcal{D}}(\boldsymbol{x}_n - \overline{\boldsymbol{x}})(\boldsymbol{x}_n - \overline{\boldsymbol{x}})^\mathsf{T}\right]^{-1}\left[\sum_{i=1}^{N_\mathcal{D}}(y_n - \overline{y})(\boldsymbol{x}_n - \overline{\boldsymbol{x}})\right]$$

$$\hat{w}_0 = \frac{1}{N}\sum_n y_n - \frac{1}{N}\sum_n \boldsymbol{x}_n^\mathsf{T}\hat{\boldsymbol{w}} = \overline{y} - \overline{\boldsymbol{x}}^\mathsf{T}\hat{\boldsymbol{w}}$$

$$\boldsymbol{x}_n^c = \boldsymbol{x}_n - \overline{\boldsymbol{x}}$$

$$\boldsymbol{y}_c = \boldsymbol{y} - \overline{\boldsymbol{y}}$$

Linear regression - Bayes

The conjugate prior to the above Gaussian likelihood is also a Gaussian, which we will denote by $p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{w}_0, \mathbf{V}_0)$. Using Bayes rule for Gaussians, Equation 4.125, the posterior is given by

$$p(\mathbf{w}|\mathbf{X}, \mathbf{y}, \sigma^2) \propto \mathcal{N}(\mathbf{w}|\mathbf{w}_0, \mathbf{V}_0)\mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{w}, \sigma^2\mathbf{I}_N) = \mathcal{N}(\mathbf{w}|\mathbf{w}_N, \mathbf{V}_N) \tag{7.55}$$

$$\mathbf{w}_N = \mathbf{V}_N\mathbf{V}_0^{-1}\mathbf{w}_0 + \frac{1}{\sigma^2}\mathbf{V}_N\mathbf{X}^T\mathbf{y} \tag{7.56}$$

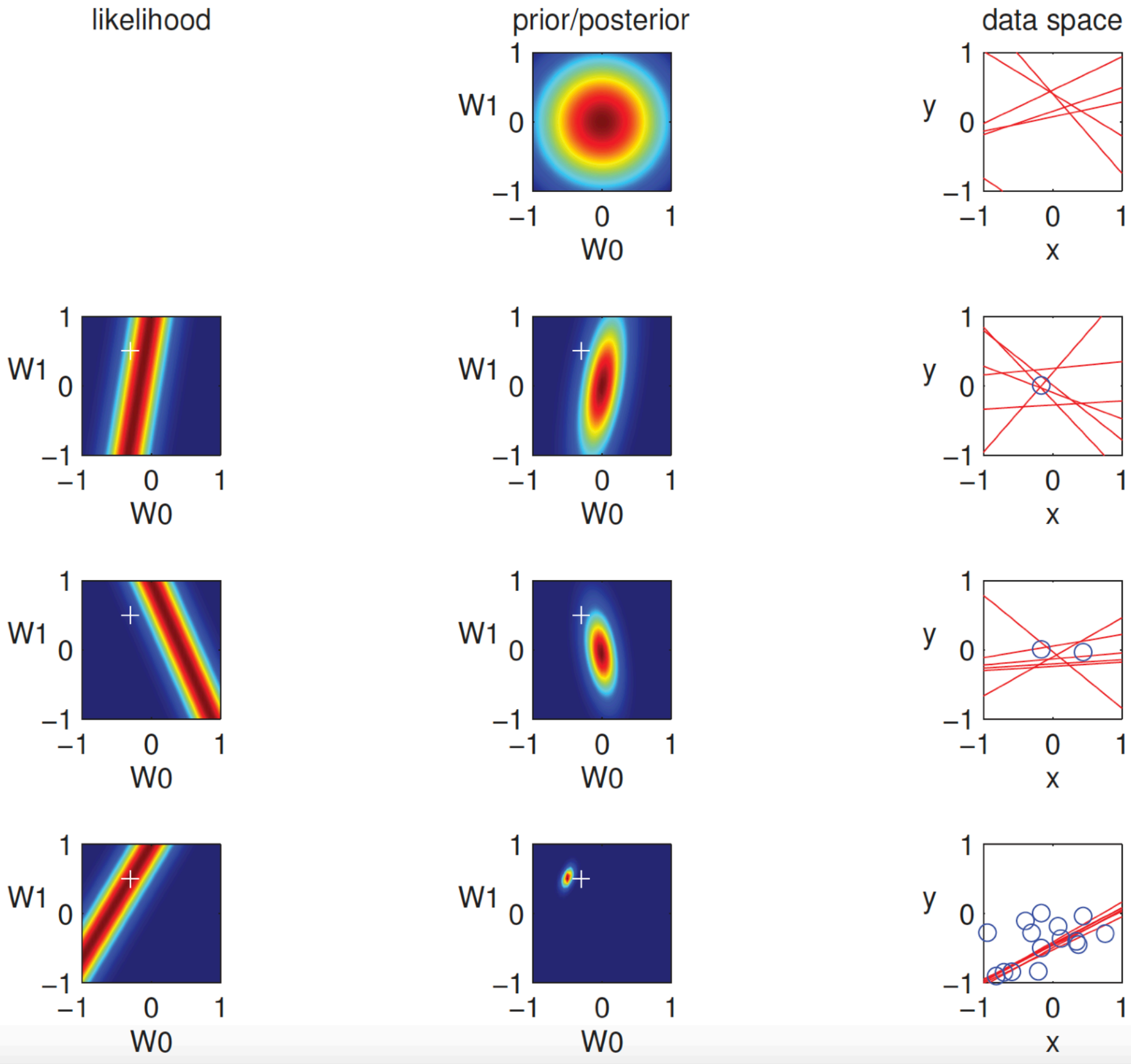
$$\mathbf{V}_N^{-1} = \mathbf{V}_0^{-1} + \frac{1}{\sigma^2}\mathbf{X}^T\mathbf{X} \tag{7.57}$$

$$\mathbf{V}_N = \sigma^2(\sigma^2\mathbf{V}_0^{-1} + \mathbf{X}^T\mathbf{X})^{-1} \tag{7.58}$$

Theorem 4.4.1 (Bayes rule for linear Gaussian systems)**.** *Given a linear Gaussian system, as in Equation 4.124, the posterior* $p(\mathbf{x}|\mathbf{y})$ *is given by the following:*

$$\begin{aligned} p(\mathbf{x}|\mathbf{y}) &= \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_{x|y}, \boldsymbol{\Sigma}_{x|y}) \\ \boldsymbol{\Sigma}_{x|y}^{-1} &= \boldsymbol{\Sigma}_x^{-1} + \mathbf{A}^T\boldsymbol{\Sigma}_y^{-1}\mathbf{A} \\ \boldsymbol{\mu}_{x|y} &= \boldsymbol{\Sigma}_{x|y}[\mathbf{A}^T\boldsymbol{\Sigma}_y^{-1}(\mathbf{y} - \mathbf{b}) + \boldsymbol{\Sigma}_x^{-1}\boldsymbol{\mu}_x] \end{aligned} \tag{4.125}$$

Linear regression - Bayes



Ridge regression

Ridge regression shrinks the regression coefficients by imposing a penalty on their size. The ridge coefficients minimize a penalized residual sum of squares,

$$\hat{\beta}^{\text{ridge}} = \underset{\beta}{\text{argmin}} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}. \tag{3.41}$$

Here $\lambda \geq 0$ is a complexity parameter that controls the amount of shrink-

An equivalent way to write the ridge problem is

$$\hat{\beta}^{\text{ridge}} = \underset{\beta}{\text{argmin}} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2,$$

$$\text{subject to } \sum_{j=1}^{p} \beta_j^2 \leq t,$$

Ridge regression

replace x_{ij} with $x_{ij} - \bar{x}_j$

Writing the criterion in (3.41) in matrix form,

$$\mathrm{RSS}(\lambda) = (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta) + \lambda\beta^T\beta, \tag{3.43}$$

the ridge regression solutions are easily seen to be

$$\hat{\beta}^{\mathrm{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}, \tag{3.44}$$

where $\mathbf{I}$ is the $p \times p$ identity matrix