EECS 189 Introduction to Machine Learning
Fall 2020 **Note 7**

1 MLE and MAP for Regression (Part II)

The power of probabilistic thinking is that it allows us a way to model situations that arise and adapt our approaches in a reasonably principled way. This is particularly true when it comes to incorporating information about the situation that comes from the physical context of the data gathering process. In this note, we will explore what happens as we vary our assumptions about the noise in our data and the priors for our parameters, as well as the "importance" of certain training points.

So far we have used MLE and MAP to justify the optimization formulation of OLS and ridge regression, respectively. The MLE formulation assumes that the observation Y_i is a noisy version of the true underlying output:

$$Y_i = f(\mathbf{x}_i) + Z_i$$

where the noise for each datapoint is crucially i.i.d. The MAP formulation assumes that the model parameter W_j is according to an i.i.d. Gaussian prior

$$W_j \overset{iid}{\sim} \mathcal{N}(\mu_j, \sigma_h^2)$$

.

So far, we have restricted ourselves to the case when the noise/parameters are i.i.d:

$$\mathbf{Z} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}), \quad \mathbf{W} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{W}}, \sigma_h^2 \mathbf{I})$$

However, what about the case when N_i's/W_j's are non-identical or dependent on one another? We would like to explore the case when the observation noise and underlying parameters are jointly Gaussian with arbitrary individual covariance matrices, but are independent of each other.

$$\mathbf{Z} \sim \mathcal{N}(\mathbf{0}, \Sigma_{\mathbf{Z}}), \quad \mathbf{W} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{W}}, \boldsymbol{\Sigma}_{\mathbf{W}})$$

It turns out that via a change of coordinates, we can reduce these non-i.i.d. problems back to the i.i.d. case and solve them using the original techniques we used to solve OLS and Ridge Regression! Changing coordinates is a powerful tool in thinking about machine learning.

1.1 Weighted Least Squares

The basic idea of **weighted least squares** is the following: we place more emphasis on the loss contributed from certain data points over others - that is, we care more about fitting some data points over others. It turns out that this weighted perspective is very useful as a building block when we go beyond traditional least-squares problems.

1.1.1 Optimization View

From an optimization perspective, the problem can be expressed as

$$\hat{\mathbf{w}}_{\text{WLS}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \left(\sum_{i=1}^{n} \omega_i (y_i - \mathbf{x}_i^\top \mathbf{w})^2 \right)$$

This objective is the same as OLS, except that each term in the sum is weighted by a positive coefficient ω_i. As always, we can vectorize this problem:

$$\hat{\mathbf{w}}_{\text{WLS}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} (\mathbf{y} - \mathbf{X}\mathbf{w})^\top \mathbf{\Omega} (\mathbf{y} - \mathbf{X}\mathbf{w})$$

Where the i'th row $\mathbf{X}$ is $\mathbf{x}_i^\top$, and $\mathbf{\Omega} \in \mathbb{R}^{n\times n}$ is a diagonal matrix with $\mathbf{\Omega}_{ii} = \omega_i$.
We rewrite the WLS objective to an OLS objective:

$$\begin{aligned}
\hat{\mathbf{w}}_{\text{WLS}} &= \arg\min_{\mathbf{w}\in\mathbb{R}^d} (\mathbf{y} - \mathbf{X}\mathbf{w})^\top \mathbf{\Omega} (\mathbf{y} - \mathbf{X}\mathbf{w}) \\
&= \arg\min_{\mathbf{w}\in\mathbb{R}^d} (\mathbf{y} - \mathbf{X}\mathbf{w})^\top \mathbf{\Omega}^{1/2}\mathbf{\Omega}^{1/2} (\mathbf{y} - \mathbf{X}\mathbf{w}) \\
&= \arg\min_{\mathbf{w}\in\mathbb{R}^d} (\mathbf{\Omega}^{1/2}\mathbf{y} - \mathbf{\Omega}^{1/2}\mathbf{X}\mathbf{w})^\top (\mathbf{\Omega}^{1/2}\mathbf{y} - \mathbf{\Omega}^{1/2}\mathbf{X}\mathbf{w}) \\
&= \arg\min_{\mathbf{w}\in\mathbb{R}^d} \|\mathbf{\Omega}^{1/2}\mathbf{y} - \mathbf{\Omega}^{1/2}\mathbf{X}\mathbf{w}\|^2
\end{aligned}$$

This formulation is identical to OLS except that we have scaled the data matrix and the observation vector by $\mathbf{\Omega}^{1/2}$, and we conclude that

$$\hat{\mathbf{w}}_{\text{WLS}} = \left((\mathbf{\Omega}^{1/2}\mathbf{X})^\top (\mathbf{\Omega}^{1/2}\mathbf{X})\right)^{-1} \left(\mathbf{\Omega}^{1/2}\mathbf{X}\right)^\top \mathbf{\Omega}^{1/2}\mathbf{y} = (\mathbf{X}^\top \mathbf{\Omega}\mathbf{X})^{-1}\mathbf{X}^\top \mathbf{\Omega}\mathbf{y}$$

1.1.2 Probabilistic View

As in MLE, we assume that our observations $\mathbf{y}$ are noisy, but now suppose that some of the y_i's are more noisy than others. How can we take this into account in our learning algorithm so we can get a better estimate of the weights? Our probabilistic model looks like

$$Y_i = \mathbf{x}_i^\top \mathbf{w} + Z_i$$

where the Z_i's are still independent Gaussians random variables, but not necessarily identical: $Z_i \sim \mathcal{N}(0, \sigma_i^2)$. Jointly, we have that $\mathbf{Z} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{Z}}, \mathbf{\Sigma}_{\mathbf{Z}})$, where

$$\mathbf{\Sigma}_{\mathbf{Z}} = \begin{bmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & \sigma_n^2 \end{bmatrix}$$

We can morph the problem into an MLE one by scaling the data to make sure all the Z_i's are identically distributed, by dividing by σ_i:

$$\frac{Y_i}{\sigma_i} = \frac{\mathbf{x}_i^\top}{\sigma_i}\mathbf{w} + \frac{Z_i}{\sigma_i}$$

Note that the scaled noise entries are now i.i.d:

$$\frac{Z_i}{\sigma_i} \overset{\text{iid}}{\sim} \mathcal{N}(0, 1)$$

Jointly, we can express this change of coordinates as

$$\boldsymbol{\Sigma}_{\mathbf{Z}}^{-\frac{1}{2}}\mathbf{y} \sim \mathcal{N}(\boldsymbol{\Sigma}_{\mathbf{Z}}^{-\frac{1}{2}}\mathbf{X}\mathbf{w}, \boldsymbol{\Sigma}_{\mathbf{Z}}^{-\frac{1}{2}}\boldsymbol{\Sigma}_{\mathbf{Z}}\boldsymbol{\Sigma}_{\mathbf{Z}}^{-\frac{\top}{2}}) = \mathcal{N}(\boldsymbol{\Sigma}_{\mathbf{Z}}^{-\frac{1}{2}}\mathbf{X}\mathbf{w}, \mathbf{I})$$

This change of variable is sometimes called the **reparameterization trick**. Now that the noise is i.i.d. using the change of coordinates, we rewrite our original problem as a scaled MLE problem:

$$\hat{\mathbf{w}}_{\text{WLS}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \left(\sum_{i=1}^{n} \frac{\left(\frac{y_i}{\sigma_i} - \frac{\mathbf{x}_i^\top}{\sigma_i}\mathbf{w}\right)^2}{2}\right) + n\log\sqrt{2\pi}$$

$$= \arg\min_{\mathbf{w}\in\mathbb{R}^d} \sum_{i=1}^{n} \frac{1}{\sigma_i^2}(y_i - \mathbf{x}_i^\top\mathbf{w})^2$$

The MLE estimate of this scaled problem is equivalent to the WLS estimate of the original problem:

$$\hat{\mathbf{w}}_{\text{WLS}} = (\mathbf{X}^\top\boldsymbol{\Sigma}_{\mathbf{Z}}^{-\frac{1}{2}}\boldsymbol{\Sigma}_{\mathbf{Z}}^{-\frac{1}{2}}\mathbf{X})^{-1}\mathbf{X}^\top\boldsymbol{\Sigma}_{\mathbf{Z}}^{-\frac{1}{2}}\boldsymbol{\Sigma}_{\mathbf{Z}}^{-\frac{1}{2}}\mathbf{y} = (\mathbf{X}^\top\boldsymbol{\Sigma}_{\mathbf{Z}}^{-1}\mathbf{X})^{-1}\mathbf{X}^\top\boldsymbol{\Sigma}_{\mathbf{Z}}^{-1}\mathbf{y}$$

As long as no σ is 0, $\boldsymbol{\Sigma}_{\mathbf{Z}}$ is invertible. Note that ω_i from the optimization perspective is directly related to σ_i^2 from the probabilistic perspective: $\omega_i = \frac{1}{\sigma_i^2}$. Or at the level of matrices, $\boldsymbol{\Omega} = \boldsymbol{\Sigma}_{\mathbf{Z}}^{-1}$. As the variance σ_i^2 of the noise corresponding to data point i decreases, the weight ω_i increases: we are more concerned about fitting data point i because it is likely to match the true underlying de-noised point. Inversely, as the variance σ_i^2 increases, the weight ω_i decreases: we are less concerned about fitting data point i because it is noisy and should not be trusted.

1.2 Generalized Least Squares

Now let's consider the case when the noise random variables are dependent on one another. We have

$$\mathbf{Y} = \mathbf{X}\mathbf{w} + \mathbf{Z}$$

where $\mathbf{Z}$ is now a jointly Gaussian random vector. That is,

$$\mathbf{Z} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\mathbf{Z}}), \quad \mathbf{Y} \sim \mathcal{N}(\mathbf{X}\mathbf{w}, \boldsymbol{\Sigma}_{\mathbf{Z}})$$

This problem is known as **generalized least squares**. Our goal is to maximize the probability of our data over the set of possible $\mathbf{w}$'s:

$$\hat{\mathbf{w}}_{\text{GLS}} = \arg\max_{\mathbf{w}\in\mathbb{R}^d} \frac{1}{\sqrt{\det(\boldsymbol{\Sigma}_{\mathbf{Z}})}}\frac{1}{(\sqrt{2\pi})^n} e^{-\frac{1}{2}(\mathbf{y}-\mathbf{X}\mathbf{w})^\top\boldsymbol{\Sigma}_{\mathbf{Z}}^{-1}(\mathbf{y}-\mathbf{X}\mathbf{w})}$$

$$= \arg\min_{\mathbf{w}\in\mathbb{R}^d} (\mathbf{y}-\mathbf{X}\mathbf{w})^\top \mathbf{\Sigma}_\mathbf{Z}^{-1}(\mathbf{y}-\mathbf{X}\mathbf{w})$$

The optimization problem is therefore given by

$$\hat{\mathbf{w}}_{\text{GLS}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} (\mathbf{y}-\mathbf{X}\mathbf{w})^\top \mathbf{\Sigma}_\mathbf{Z}^{-1}(\mathbf{y}-\mathbf{X}\mathbf{w})$$

Since $\mathbf{\Sigma}_\mathbf{Z}$ is symmetric, we can decompose it into its eigen structure using the spectral theorem:

$$\mathbf{\Sigma}_\mathbf{Z} = \mathbf{Q}\begin{bmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & \sigma_n^2 \end{bmatrix}\mathbf{Q}^\top$$

where $\mathbf{Q}$ is orthonormal. As before with weighted least squares, our goal is to find an appropriate linear transformation so that we can reduce the problem into the i.i.d. case.

Consider

$$\mathbf{\Sigma}_\mathbf{Z}^{-\frac{1}{2}} = \mathbf{Q}\begin{bmatrix} \frac{1}{\sigma_1} & 0 & \cdots & 0 \\ 0 & \frac{1}{\sigma_2} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & \frac{1}{\sigma_n} \end{bmatrix}\mathbf{Q}^\top$$

We can scale the data to morph the problem into an MLE problem with i.i.d. noise variables, by premultiplying the data matrix $\mathbf{X}$ and the observation vector $\mathbf{y}$ by $\mathbf{\Sigma}_\mathbf{Z}^{-\frac{1}{2}}$. Jointly, we can express this change of coordinates as

$$\mathbf{\Sigma}_\mathbf{Z}^{-\frac{1}{2}}\mathbf{y} \sim \mathcal{N}(\mathbf{\Sigma}_\mathbf{Z}^{-\frac{1}{2}}\mathbf{X}\mathbf{w}, \mathbf{\Sigma}_\mathbf{Z}^{-\frac{1}{2}}\mathbf{\Sigma}_\mathbf{Z}\mathbf{\Sigma}_\mathbf{Z}^{-\frac{\top}{2}}) = \mathcal{N}(\mathbf{\Sigma}_\mathbf{Z}^{-\frac{1}{2}}\mathbf{X}\mathbf{w}, \mathbf{I}).$$

Consequently, in a very similar fashion to the independent noise problem, the MLE of the scaled dependent noise problem is

$$\hat{\mathbf{w}}_{\text{GLS}} = (\mathbf{X}^\top\mathbf{\Sigma}_\mathbf{Z}^{-1}\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{\Sigma}_\mathbf{Z}^{-1}\mathbf{y}.$$

1.3 "Ridge Regression" with Dependent Parameters

In the ordinary least squares (OLS) statistical model, we assume that the output $\mathbf{Y}$ is a linear function of the input, plus some Gaussian noise. We take this one step further in MAP estimation, where we assume that the weights are a random variable. The new statistical model is

$$\mathbf{Y} = \mathbf{X}\mathbf{W} + \mathbf{Z}$$

where $\mathbf{Y}$ and $\mathbf{Z}$ are n-dimensional random vectors, $\mathbf{W}$ is a d-dimensional random vector, and $\mathbf{X}$ is a fixed $n \times d$ matrix. Note that random vectors are not notationally distinguished from matrices here, so keep in mind what each symbol represents.

We have seen that ridge regression can be derived by assuming a prior distribution on $\mathbf{W}$ in which W_i are i.i.d. (univariate) Gaussian, or equivalently,

$$\mathbf{W} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$$

But more generally, we can allow $\mathbf{W}$ to be any multivariate Gaussian:

$$\mathbf{W} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{W}}, \boldsymbol{\Sigma}_{\mathbf{W}})$$

Recall that we can rewrite a multivariate Gaussian variable as an affine transformation of a standard Gaussian variable:

$$\mathbf{W} = \boldsymbol{\Sigma}_{\mathbf{W}}^{1/2}\mathbf{V} + \boldsymbol{\mu}_{\mathbf{W}}, \quad \mathbf{V} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$$

Plugging this parameterization into our previous statistical model gives

$$\mathbf{Y} = \mathbf{X}(\boldsymbol{\Sigma}_{\mathbf{W}}^{1/2}\mathbf{V} + \boldsymbol{\mu}_{\mathbf{W}}) + \mathbf{Z}$$

But this can be re-written

$$\mathbf{Y} - \mathbf{X}\boldsymbol{\mu}_{\mathbf{W}} = \mathbf{X}\boldsymbol{\Sigma}_{\mathbf{W}}^{1/2}\mathbf{V} + \mathbf{Z}$$

Assignment Project Exam Help

which we see has the form of the statistical problem that underlies traditional Ridge Regression with $\lambda = 1$, and therefore

$$\hat{\mathbf{v}} = (\boldsymbol{\Sigma}_{\mathbf{W}}^{\top/2}\mathbf{X}^{\top}\mathbf{X}\boldsymbol{\Sigma}_{\mathbf{W}}^{1/2} + \mathbf{I})^{-1}\boldsymbol{\Sigma}_{\mathbf{W}}^{\top/2}\mathbf{X}^{\top}(\mathbf{y} - \mathbf{X}\boldsymbol{\mu}_{\mathbf{W}})$$

https://powcoder.com

However $\mathbf{V}$ is not what we care about – we need to convert back to the actual weights $\mathbf{W}$ in order to make predictions. Since $\mathbf{W}$ is completely determined by $\mathbf{V}$ (assuming fixed mean and covariance),

Add WeChat powcoder

$$\begin{aligned}
\hat{\mathbf{w}} &= \boldsymbol{\Sigma}_{\mathbf{W}}^{1/2}\hat{\mathbf{v}} + \boldsymbol{\mu}_{\mathbf{W}} \\
&= \boldsymbol{\mu}_{\mathbf{W}} + \boldsymbol{\Sigma}_{\mathbf{W}}^{1/2}(\boldsymbol{\Sigma}_{\mathbf{W}}^{\top/2}\mathbf{X}^{\top}\mathbf{X}\boldsymbol{\Sigma}_{\mathbf{W}}^{1/2} + \mathbf{I})^{-1}\boldsymbol{\Sigma}_{\mathbf{W}}^{\top/2}\mathbf{X}^{\top}(\mathbf{y} - \mathbf{X}\boldsymbol{\mu}_{\mathbf{W}}) \\
&= \boldsymbol{\mu}_{\mathbf{W}} + (\mathbf{X}^{\top}\mathbf{X} + \underbrace{\boldsymbol{\Sigma}_{\mathbf{W}}^{-\top/2}\boldsymbol{\Sigma}_{\mathbf{W}}^{-1/2}}_{\boldsymbol{\Sigma}_{\mathbf{W}}^{-1}})^{-1}\mathbf{X}^{\top}(\mathbf{y} - \mathbf{X}\boldsymbol{\mu}_{\mathbf{W}}) \\
&= \boldsymbol{\mu}_{\mathbf{W}} + (\mathbf{X}^{\top}\mathbf{X} + \boldsymbol{\Sigma}_{\mathbf{W}}^{-1})^{-1}\mathbf{X}^{\top}(\mathbf{y} - \mathbf{X}\boldsymbol{\mu}_{\mathbf{W}})
\end{aligned}$$

Note that there are two terms: the prior mean $\boldsymbol{\mu}_{\mathbf{W}}$, plus another term that depends on both the data and the prior. The positive-definite precision matrix of $\mathbf{W}$'s prior ($\boldsymbol{\Sigma}_{\mathbf{W}}^{-1}$) controls how the data fit error affects our estimate. This is called Tikhonov regularization in the literature and generalizes ridge regularization.

To gain intuition, let us consider the simplified case where

$$\boldsymbol{\Sigma}_{\mathbf{W}} = \begin{bmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_d^2 \end{bmatrix}$$

When the prior variance σ_j^2 for dimension j is large, the prior is telling us that W_j may take on a wide range of values. Thus we do not want to penalize that dimension as much, preferring to let the data fit sort it out. And indeed the corresponding entry in $\mathbf{\Sigma}_\mathbf{W}^{-1}$ will be small, as desired.

Conversely if σ_j^2 is small, there is little variance in the value of W_j, so $W_j \approx \mu_j$. As such we penalize the magnitude of the data-fit contribution to $\hat{W}_j$ more heavily.

If all the σ_j^2 are the same, then we have traditional ridge regularization.

1.3.1 Alternative derivation: directly conditioning jointly Gaussian random variables

In an explicitly probabilistic perspective, MAP with colored noise (and known $\mathbf{X}$) can be expressed as:

$$\mathbf{U}, \mathbf{V} \overset{\text{iid}}{\sim} \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{1}$$

$$\begin{bmatrix} \mathbf{Y} \\ \mathbf{W} \end{bmatrix} = \begin{bmatrix} \mathbf{R_Z} & \mathbf{XR_W} \\ \mathbf{0} & \mathbf{R_W} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{V} \end{bmatrix} \tag{2}$$

where $\mathbf{R_Z}$ and $\mathbf{R_W}$ are relationships with $\mathbf{W}$ and $\mathbf{Z}$, respectively. Note that the $\mathbf{R_W}$ appears twice because our model assumes $\mathbf{Y} = \mathbf{XW} + \text{noise}$, so if $\mathbf{W} = \mathbf{R_W V}$, then we must have $\mathbf{Y} = \mathbf{XR_W V} + \text{noise}$.

We want to find the posterior $\mathbf{W} \mid \mathbf{Y} = \mathbf{y}$. The formulation above makes it relatively easy to find the posterior of $\mathbf{Y}$ conditioned on $\mathbf{W}$ (see below), but not vice-versa. So let's pretend instead that

$$\mathbf{U}', \mathbf{V}' \overset{\text{iid}}{\sim} \mathcal{N}(\mathbf{0}, \mathbf{I})$$

$$\begin{bmatrix} \mathbf{W} \\ \mathbf{Y} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{0} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{U}' \\ \mathbf{V}' \end{bmatrix}$$

Now $\mathbf{W} \mid \mathbf{Y} = \mathbf{y}$ is straightforward. Since $\mathbf{V}' = \mathbf{D}^{-1}\mathbf{Y}$, the conditional mean and variance of $\mathbf{W} \mid \mathbf{Y} = \mathbf{y}$ can be computed as follows:

$$\begin{aligned}
\mathbb{E}[\mathbf{W} \mid \mathbf{Y} = \mathbf{y}] &= \mathbb{E}[\mathbf{AU}' + \mathbf{BV}' \mid \mathbf{Y} = \mathbf{y}] \\
&= \mathbb{E}[\mathbf{AU}' \mid \mathbf{Y} = \mathbf{y}] + \mathbb{E}[\mathbf{BD}^{-1}\mathbf{Y} \mid \mathbf{Y} = \mathbf{y}] \\
&= \mathbf{A} \underbrace{\mathbb{E}[\mathbf{U}']}_{\mathbf{0}} + \mathbb{E}[\mathbf{BD}^{-1}\mathbf{Y} \mid \mathbf{Y} = \mathbf{y}] \\
&= \mathbf{BD}^{-1}\mathbf{y} \\
\mathrm{Var}(\mathbf{W} \mid \mathbf{Y} = \mathbf{y}) &= \mathbb{E}[(\mathbf{W} - \mathbb{E}[\mathbf{W}])(\mathbf{W} - \mathbb{E}[\mathbf{W}])^\top \mid \mathbf{Y} = \mathbf{y}] \\
&= \mathbb{E}[(\mathbf{AU}' + \mathbf{BD}^{-1}\mathbf{Y} - \mathbf{BD}^{-1}\mathbf{Y})(\mathbf{AU}' + \mathbf{BD}^{-1}\mathbf{Y} - \mathbf{BD}^{-1}\mathbf{Y})^\top \mid \mathbf{Y} = \mathbf{y}] \\
&= \mathbb{E}[(\mathbf{AU}')(\mathbf{AU}')^\top \mid \mathbf{Y} = \mathbf{y}] \\
&= \mathbb{E}[\mathbf{AU}'(\mathbf{U}')^\top\mathbf{A}^\top] \\
&= \mathbf{A} \underbrace{\mathbb{E}[\mathbf{U}'(\mathbf{U}')^\top]}_{=\mathrm{Var}(\mathbf{U}')=\mathbf{I}} \mathbf{A}^\top \\
&= \mathbf{AA}^\top
\end{aligned}$$

In both cases above where we drop the conditioning on $\mathbf{Y}$, we are using the fact $\mathbf{U}'$ is independent of $\mathbf{V}'$ (and thus independent of $\mathbf{Y} = \mathbf{D}\mathbf{V}'$). Therefore

$$\mathbf{W} \mid \mathbf{Y} = \mathbf{y} \sim \mathcal{N}(\mathbf{B}\mathbf{D}^{-1}\mathbf{y}, \mathbf{A}\mathbf{A}^\top)$$

Recall that a Gaussian distribution is completely specified by its mean and covariance matrix. We see that the covariance matrix of the joint distribution is

$$\begin{aligned}
\mathbb{E}\left[\begin{bmatrix}\mathbf{W}\\ \mathbf{Y}\end{bmatrix}\begin{bmatrix}\mathbf{W}^\top & \mathbf{Y}^\top\end{bmatrix}\right] &= \begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{0} & \mathbf{D}\end{bmatrix}\begin{bmatrix}\mathbf{A}^\top & \mathbf{0}\\ \mathbf{B}^\top & \mathbf{D}^\top\end{bmatrix}\\
&= \begin{bmatrix}\mathbf{A}\mathbf{A}^\top + \mathbf{B}\mathbf{B}^\top & \mathbf{B}\mathbf{D}^\top\\ \mathbf{D}\mathbf{B}^\top & \mathbf{D}\mathbf{D}^\top\end{bmatrix}\\
&= \begin{bmatrix}\Sigma_\mathbf{W} & \Sigma_{\mathbf{W},\mathbf{Y}}\\ \Sigma_{\mathbf{Y},\mathbf{W}} & \Sigma_\mathbf{Y}\end{bmatrix}
\end{aligned}$$

Matching the corresponding terms, we can express the conditional mean and variance of $\mathbf{W} \mid \mathbf{Y} = \mathbf{y}$ in terms of these (cross-)covariance matrices:

$$\mathbf{B}\mathbf{D}^{-1}\mathbf{Y} = \mathbf{B}\underbrace{\mathbf{D}^\top\mathbf{D}^{-\top}}_{\mathbf{I}}\mathbf{D}^{-1}\mathbf{Y} = (\mathbf{B}\mathbf{D}^\top)(\mathbf{D}\mathbf{D}^\top)^{-1}\mathbf{Y} = \Sigma_{\mathbf{W},\mathbf{Y}}\Sigma_\mathbf{Y}^{-1}\mathbf{Y}$$

$$\begin{aligned}
\mathbf{A}\mathbf{A}^\top &= \mathbf{A}\mathbf{A}^\top + \mathbf{B}\mathbf{B}^\top - \mathbf{B}\mathbf{B}^\top\\
&= \mathbf{A}\mathbf{A}^\top + \mathbf{B}\mathbf{B}^\top - \mathbf{B}\underbrace{\mathbf{D}^\top\mathbf{D}^{-\top}}_{\mathbf{I}}\underbrace{\mathbf{D}^{-1}\mathbf{D}}_{\mathbf{I}}\mathbf{B}^\top\\
&= \mathbf{A}\mathbf{A}^\top + \mathbf{B}\mathbf{B}^\top - (\mathbf{B}\mathbf{D}^\top)(\mathbf{D}\mathbf{D}^\top)^{-1}\mathbf{D}\mathbf{B}^\top\\
&= \Sigma_\mathbf{W} - \Sigma_{\mathbf{W},\mathbf{Y}}\Sigma_\mathbf{Y}^{-1}\Sigma_{\mathbf{Y},\mathbf{W}}
\end{aligned}$$

We can then apply the same reasoning to the original setup:

$$\begin{aligned}
\mathbb{E}\left[\begin{bmatrix}\mathbf{Y}\\ \mathbf{W}\end{bmatrix}\begin{bmatrix}\mathbf{Y}^\top & \mathbf{W}^\top\end{bmatrix}\right] &= \begin{bmatrix}\mathbf{R}_\mathbf{Z}\mathbf{R}_\mathbf{Z}^\top + \mathbf{X}\mathbf{R}_\mathbf{W}\mathbf{R}_\mathbf{W}^\top\mathbf{X}^\top & \mathbf{X}\mathbf{R}_\mathbf{W}\mathbf{R}_\mathbf{W}^\top\\ \mathbf{R}_\mathbf{W}\mathbf{R}_\mathbf{W}^\top\mathbf{X}^\top & \mathbf{R}_\mathbf{W}\mathbf{R}_\mathbf{W}^\top\end{bmatrix}\\
&= \begin{bmatrix}\Sigma_\mathbf{Y} & \Sigma_{\mathbf{Y},\mathbf{W}}\\ \Sigma_{\mathbf{W},\mathbf{Y}} & \Sigma_\mathbf{W}\end{bmatrix}
\end{aligned}$$

Therefore after defining $\Sigma_\mathbf{Z} = \mathbf{R}_\mathbf{Z}\mathbf{R}_\mathbf{Z}^\top$, we can read off

$$\begin{aligned}
\Sigma_\mathbf{W} &= \mathbf{R}_\mathbf{W}\mathbf{R}_\mathbf{W}^\top\\
\Sigma_\mathbf{Y} &= \Sigma_\mathbf{Z} + \mathbf{X}\Sigma_\mathbf{W}\mathbf{X}^\top\\
\Sigma_{\mathbf{Y},\mathbf{W}} &= \mathbf{X}\Sigma_\mathbf{W}\\
\Sigma_{\mathbf{W},\mathbf{Y}} &= \Sigma_\mathbf{W}\mathbf{X}^\top
\end{aligned}$$

Plugging this into our estimator yields

$$\begin{aligned}
\hat{\mathbf{w}} &= \mathbb{E}[\mathbf{W} \mid \mathbf{Y} = \mathbf{y}]\\
&= \Sigma_{\mathbf{W},\mathbf{Y}}\Sigma_Y^{-1}\mathbf{y}
\end{aligned}$$

$$= \mathbf{\Sigma_W X^\top (\Sigma_Z + X \Sigma_W X^\top)^{-1} y}$$

One may be concerned because this expression does not take the form we expect – the inverted matrix is hitting $\mathbf{y}$ directly, unlike in other solutions we've seen. Although this form will turn out to be quite informative when we introduce the idea of the kernel trick in machine, learning, it is still disconcertingly different from what we are used to.

However, by using a lot of algebra together with the Woodbury matrix identity[1], it turns out that we can rewrite this expression as

$$\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{\Sigma_Z^{-1} X} + \mathbf{\Sigma_W^{-1}})^{-1} \mathbf{X}^\top \mathbf{\Sigma_Z^{-1} y}$$

which looks more familiar. In fact, you can recognize this as the general solution when we have both a generic Gaussian prior on the parameters and colored noise in the observations.

1.4 Summary of Linear Gaussian Statistical Models

We have seen a number of related linear models, with varying assumptions about the randomness in the observations and the weights. We summarize these below:

$\mathbf{W}$ \ $\mathbf{Z}$	$\mathcal{N}(\mathbf{0}, \mathbf{I})$	$\mathcal{N}(\mathbf{0}, \mathbf{\Sigma_Z})$
No prior	$\hat{\mathbf{w}}_{\text{OLS}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$	$\hat{\mathbf{w}}_{\text{GLS}} = (\mathbf{X}^\top \mathbf{\Sigma_Z^{-1} X})^{-1} \mathbf{X}^\top \mathbf{\Sigma_Z^{-1} y}$
$\mathcal{N}(\mathbf{0}, \lambda^{-1}\mathbf{I})$	$\hat{\mathbf{w}}_{\text{RIDGE}} = (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^\top \mathbf{y}$	$(\mathbf{X}^\top \mathbf{\Sigma_Z^{-1} X} + \lambda \mathbf{I})^{-1} \mathbf{X}^\top \mathbf{\Sigma_Z^{-1} y}$
$\mathcal{N}(\boldsymbol{\mu}_\mathbf{W}, \mathbf{\Sigma_W})$	$\boldsymbol{\mu}_\mathbf{W} + (\mathbf{X}^\top \mathbf{X} + \mathbf{\Sigma_W^{-1}})^{-1} \mathbf{X}^\top (\mathbf{y} - \mathbf{X} \boldsymbol{\mu}_\mathbf{W})$	$\boldsymbol{\mu}_\mathbf{W} + (\mathbf{X}^\top \mathbf{\Sigma_Z^{-1} X} + \mathbf{\Sigma_W^{-1}})^{-1} \mathbf{X}^\top \mathbf{\Sigma_Z^{-1}} (\mathbf{y} - \mathbf{X} \boldsymbol{\mu}_\mathbf{W})$

[1] $(\mathbf{A} + \mathbf{UCV})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{U}(\mathbf{C}^{-1} + \mathbf{VA}^{-1}\mathbf{U})^{-1}\mathbf{VA}^{-1}$