CS 189 Introduction to Machine Learning
Fall 2017 Note 7

1 MAP with Colored Noise

Recall the ordinary least squares (OLS) model. We have a dataset $\mathscr{D} = \{(\vec{a}_i, y_i)\}_{i=1}^n$ and assume that each y_i is a linear function of $\vec{a}_i$, plus some independent Gaussian noise, which we have rescaled to have variance 1:

$$z_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1) \tag{1}$$

$$y_i = \vec{a}_i^\top \vec{w} + z_i \tag{2}$$

Initially we used the geometric interpretation of OLS to solve for $\vec{w}$. The previous two lectures showed how we can find $\vec{w}$ with **estimators** instead:

1. Maximum likelihood estimation (**MLE**):

$$\vec{w}^* = \arg\max_{\vec{w}} \log P(\mathscr{D} \mid \vec{w})$$

2. Maximum a posteriori estimation (**MAP**):

$$\vec{w}^* = \arg\max_{\vec{w}} \log P(\vec{w} \mid \mathscr{D}) = \arg\max_{\vec{w}} \log P(\mathscr{D} \mid \vec{w}) + \log P(\vec{w})$$

When deriving ridge regression via MAP estimation, our prior assumed that w_i were i.i.d. (univariate) Gaussian, but more generally, we can allow $\vec{w}$ to be any multivariate Gaussian:

$$\vec{w} \sim \mathcal{N}(\vec{\mu}_w, \Sigma_w)$$

Recall (see Discussion 4) that we can rewrite a multivariate Gaussian variable as an affine transformation of a standard Gaussian variable:

$$\vec{w} = \Sigma_w^{1/2} \underbrace{\vec{v}}_{\text{noise}} + \underbrace{\vec{\mu}_w}_{\text{mean}} \qquad \vec{v} \sim \mathcal{N}(0, I)$$

This change of variable is sometimes called the *reparameterization trick*.

Plugging this reparameterization into our approximation $\vec{Y} \approx A\vec{w}$ gives

$$\begin{aligned}
\vec{Y} &\approx A\Sigma_w^{1/2}\vec{v} + A\vec{\mu}_w \\
A\Sigma_w^{1/2}\vec{v} &\approx \vec{Y} - A\vec{\mu}_w \\
\hat{\vec{v}} &= (\Sigma_w^{T/2} A^\top A \Sigma_w^{1/2} + I)^{-1} \Sigma_w^{T/2} A^\top (\vec{y} - A\vec{\mu}_w)
\end{aligned}$$

Since the variance from data and prior have both been normalized, the noise-to-signal ratio (λ) is equal to 1.

However $\vec{v}$ is not what we care about – we need to convert back to the actual weights $\vec{w}$ in order to make predictions. Using our identity again,

$$\begin{aligned}\hat{\vec{w}} &= \vec{\mu}_w + \Sigma_w^{1/2}(\Sigma_w^{T/2}A^\top A\Sigma_w^{1/2}+I)^{-1}\Sigma_w^{T/2}A^\top(\vec{y}-A\vec{\mu}_w)\\ &= \vec{\mu}_w + (A^\top A + \underbrace{\Sigma_w^{-T/2}\Sigma_w^{-1/2}}_{\Sigma_w^{-1}})^{-1}A^\top(\vec{y}-A\vec{\mu}_w)\end{aligned}$$

Note that there are two terms: the prior mean $\vec{\mu}_w$, plus another term that depends on both the data and the prior. The precision matrix of $\vec{w}$'s prior (Σ_w^{-1}) controls how the data fit error affects our estimate.

To gain intuition, let us consider the simplified case where

$$\Sigma_w = \begin{bmatrix} \sigma_{w,1}^2 & 0 & \cdots & 0 \\ 0 & \sigma_{w,2}^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_{w,n}^2 \end{bmatrix}$$

When the prior variance $\sigma_{w,j}^2$ 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 Σ_w^{-1} will be small, as desired.

Conversely if $\sigma_{w,j}^2$ 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.

1.1 Alternative derivation

MAP with colored noise can be expressed as:

$$\vec{u},\vec{v} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\vec{0},I) \tag{3}$$

$$\begin{bmatrix}\vec{Y}\\ \vec{w}\end{bmatrix} = \begin{bmatrix}R_z & AR_w\\ 0 & R_w\end{bmatrix}\begin{bmatrix}\vec{u}\\ \vec{v}\end{bmatrix} \tag{4}$$

where R_z and R_w are relationships with w and z, respectively. Note that the R_w appears twice because our model assumes $\vec{Y} = A\vec{w} + \text{noise}$, so if $\vec{w} = R_w\vec{v}$, then we must have $\vec{Y} = AR_w\vec{v} + \text{noise}$.

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

$$\begin{aligned}&u',v' \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\vec{0},I)\\ &\begin{bmatrix}\vec{w}\\ \vec{Y}\end{bmatrix} = \begin{bmatrix}A & B\\ 0 & D\end{bmatrix}\begin{bmatrix}u'\\ v'\end{bmatrix}\end{aligned}$$

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

$$
\begin{aligned}
\mathbb{E}[\vec{w} \mid \vec{Y}] &= \mathbb{E}[Au' + Bv' \mid \vec{Y}] \\
&= \mathbb{E}[Au' \mid \vec{Y}] + \mathbb{E}[BD^{-1}\vec{Y} \mid \vec{Y}] \\
&= A\underbrace{\mathbb{E}[u']}_{\vec{0}} + \mathbb{E}[BD^{-1}\vec{Y} \mid \vec{Y}] \\
&= BD^{-1}\vec{Y} \\
\operatorname{var}(\vec{w} \mid \vec{Y}) &= \mathbb{E}[(\vec{w} - \mathbb{E}[\vec{w}])(\vec{w} - \mathbb{E}[\vec{w}])^\top \mid \vec{Y}] \\
&= \mathbb{E}[(Au' + BD^{-1}\vec{Y} - BD^{-1}\vec{Y})(Au' + BD^{-1}\vec{Y} - BD^{-1}\vec{Y})^\top \mid \vec{Y}] \\
&= \mathbb{E}[(Au')(Au')^\top \mid \vec{Y}] \\
&= \mathbb{E}[Au'(u')^\top A^\top] \\
&= A\underbrace{\mathbb{E}[u'(u')^\top]}_{=\operatorname{var}(u')=I} A^\top \\
&= AA^\top
\end{aligned}
$$

In both cases above where we drop the conditioning on $\vec{Y}$, we are using the fact u' is independent of v' (and thus independent of $\vec{Y} = Dv'$). Therefore

$$
\vec{w} \mid \vec{Y} \sim \mathcal{N}(BD^{-1}\vec{Y}, AA^\top)
$$

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

$$
\begin{aligned}
\mathbb{E}\left[\begin{bmatrix}\vec{w} \\ \vec{Y}\end{bmatrix} \begin{bmatrix}\vec{w}^\top & \vec{Y}^\top\end{bmatrix}\right] &= \begin{bmatrix}A & B \\ 0 & D\end{bmatrix}\begin{bmatrix}A^\top & 0 \\ B^\top & D^\top\end{bmatrix} \\
&= \begin{bmatrix}AA^\top + BB^\top & BD^\top \\ DB^\top & DD^\top\end{bmatrix} \\
&= \begin{bmatrix}\Sigma_w & \Sigma_{w,Y} \\ \Sigma_{Y,w} & \Sigma_Y\end{bmatrix}
\end{aligned}
$$

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

$$
\begin{aligned}
BD^{-1}\vec{Y} &= B\underbrace{D^\top D^{-T}}_{I} D^{-1}\vec{Y} = (BD^\top)(DD^\top)^{-1}\vec{Y} = \Sigma_{w,Y}\Sigma_Y^{-1}\vec{Y} \\
AA^\top &= AA^\top + BB^\top - BB^\top \\
&= AA^\top + BB^\top - B\underbrace{D^\top D^{-T}}_{I}\underbrace{D^{-1}D}_{I}B^\top \\
&= AA^\top + BB^\top - (BD^\top)(DD^\top)^{-1}DB^\top \\
&= \Sigma_w - \Sigma_{w,Y}\Sigma_Y^{-1}\Sigma_{Y,w}
\end{aligned}
$$

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

$$\mathbb{E}\left[\begin{bmatrix}\vec{Y}\\ \vec{w}\end{bmatrix}\begin{bmatrix}\vec{Y}^\top & \vec{w}^\top\end{bmatrix}\right] = \begin{bmatrix}R_zR_z^\top + AR_wR_w^\top A^\top & AR_wR_w^\top \\ R_wR_w^\top A^\top & R_wR_w^\top\end{bmatrix}$$
$$= \begin{bmatrix}\Sigma_Y & \Sigma_{Y,w} \\ \Sigma_{w,Y} & \Sigma_w\end{bmatrix}$$

Therefore after defining $\Sigma_z = R_zR_z^\top$, we can read off

$$\begin{aligned}
\Sigma_w &= R_wR_w^\top \\
\Sigma_Y &= \Sigma_z + A\Sigma_w A^\top \\
\Sigma_{Y,w} &= A\Sigma_w \\
\Sigma_{w,Y} &= \Sigma_w A^\top
\end{aligned}$$

Plugging this into our estimator yields

$$\begin{aligned}
\hat{\vec{w}} &= \mathbb{E}[\vec{w} \mid \vec{Y} = \vec{y}] \\
&= \Sigma_{w,Y}\Sigma_Y^{-1}\vec{y} \\
&= \Sigma_w A^\top(\Sigma_z + A\Sigma_w A^\top)^{-1}\vec{y}
\end{aligned}$$

One may be concerned because this expression does not take the form we expect – the inverted matrix is hitting $\vec{y}$ directly, unlike in other solutions we've seen. But using the Woodbury matrix identity[1], it turns out that we can rewrite this expression as

$$\hat{\vec{w}} = (A^\top\Sigma_z^{-1}A + \Sigma_w^{-1})^{-1}A^\top\Sigma_z^{-1}\vec{y}$$

which looks more familiar.

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