1 Regularization with priors: quick refresher

1.1 MAP inference

We have previously discussed the idea of adding a prior (or equivalently, a penalty) to regularize weights in a GLM or other regression model.

That is, we seek the maximum a posteriori (MAP) estimate:

$$\hat{\mathbf{w}}_{\text{map}} = \arg \max_{\mathbf{w}} \log p(\mathbf{w}|Y, X, \theta) = \arg \max_{\mathbf{w}} \left[ \log p(Y|X, \theta) + \log p(\mathbf{w}|\theta) \right].$$

For the linear-Gaussian model we can compute the MAP estimate in closed form, but for the other models we’ve considered (Bernoulli and Poisson GLM), we must find the maximum via numerical optimization.

Maximizing the log-posterior is equivalent to maximizing the rightmost expression (the log-likelihood plus log-prior) because the denominator in Bayes’ rule is a constant does not depend on $\mathbf{w}$.

1.2 Ridge and Smoothing Priors

The priors we have considered so far are the ridge prior, $\mathbf{w} \sim \mathcal{N}(0, \frac{1}{\lambda} I)$, which has log-prior (or penalty) equal to

$$\text{penalty} = \log p(\mathbf{w}|\lambda) = -\frac{1}{2} \lambda \mathbf{w}^\top \mathbf{w},$$

and the graph Laplacian smoothing prior, which has log-prior (or penalty) equal to

$$\text{penalty} = \log p(\mathbf{w}|\lambda) = -\frac{1}{2} \lambda \mathbf{w}^\top L \mathbf{w},$$

where $L$ is the graph Laplacian.

Here we have written the prior so that $\lambda$ is equal to the inverse variance of the prior. Thus $\lambda = 0$ corresponds to a prior with infinitely broad variance (in which case we have no regularization, so the MAP estimate is equal to the maximum likelihood estimate). At the other extreme, $\lambda = \infty$ corresponds to an infinitely strong penalty, or a prior that is infinitely concentrated at zero (that is, a Dirac delta function).

Note we previously use $\theta$ to parametrize the prior, where $\theta$ denoted the prior variance but we switch here to $\lambda = 1/\theta$ simply because this notation is more common!
1.3 The key question: how to set $\lambda$?

The question we have (so far) neglected is: how do we choose the strength of the regularizer, or in Bayesian terms, the width of the prior? We will consider two possible approaches: (1) Cross-validation; (2) maximum marginal likelihood / evidence optimization.

2 Cross-validation

The simplest and arguably most popular approach for determining the amount of regularization is cross-validation.

The idea is to divide the data $(X,Y)$ into a training set $(X_{\text{train}}, Y_{\text{train}})$ and a test set $(X_{\text{test}}, Y_{\text{test}})$.

To perform cross-validation, we consider a grid of hyperparameter values $\{\lambda_1, \lambda_2, \ldots, \lambda_m\}$. For each value $\lambda_j$, we will compute the MAP estimate using the training data:

$$\hat{w}_{\text{map}} = \arg \max_{\bar{w}} \log p(\bar{w} | Y_{\text{train}}, X_{\text{train}}, \lambda_j),$$  

(4)

then compute the test log-likelihood

$$L_{j}^{\text{test}} = \log p(Y_{\text{test}} | X_{\text{test}}, \hat{w}_{\text{map}}).$$  

(5)

The test log-likelihood $L_{j}^{\text{test}}$ as a function of $\lambda$ will typically have an inverted-U shape with a maximum at some intermediate value of $\lambda$. By contrast, the training log-likelihood, equal to

$$L_{j}^{\text{train}} = \log p(Y_{\text{train}} | X_{\text{train}}, \hat{w}_{\text{map}}),$$  

(6)

is always a decreasing function of $\lambda$. Values to the left of the maximum of the test log-likelihood correspond to overfitting, where the model parameters have too many degrees of freedom (i.e., the prior is too wide / the penalty is too weak), causing the fitted parameters to fit features of the training set that are not useful for predicting the test set.

Values to the right of the maximum correspond to underfitting, where the weights are too close to zero or too smooth because prior is too narrow / penalty is too strong, so the model does not have enough flexibility to capture enough of the structure of the training data.

Other topics discussed:

- **n-fold cross-validation** - instead of dividing the data into a single training and test set, n-fold cross-validation involves dividing the data into equal subsets called “folds”. We then consider $n$ different train-test splits, where each fold is held out as test set while the remaining folds are used for training. Average the test-loglikelihood across folds in order to determine the “best” optimum for selecting $\lambda$.

- **test vs. validation set** - in practice it is often necessary to divide data into three different sets: a training set (used to fit the parameters for each value of $\lambda$), a validation set (used
for selecting the optimal hyperparameters $\lambda$, and test set (which is kept aside for reporting “true” test performance). This approach scrupulously avoids overfitting because the test set is not used for fitting hyperparameters.

- *test log-likelihood vs. test error* - in cases with Gaussian noise it is common to report test mean-squared-error (MSE) instead of test log-likelihood. Thus, we look for a minimum of the test (or validation) error, which is equivalent to a maximum of the test (or validation) log-likelihood.

### 3 Maximum marginal likelihood & Empirical Bayes

An alternate approach for setting hyperparameters, which has the advantage that it does not require splitting the data into training and test sets, is to maximize the marginal likelihood:

$$\hat{\lambda} = \arg \max P(Y|X, \lambda)$$

where recall that the *marginal likelihood* or *evidence*, which is the denominator in Bayes’ rule, is given by:

$$P(Y|X, \lambda) = \int P(Y|X, \vec{w})P(\vec{w}|\lambda)d\vec{w}.$$  

(8)

The marginal likelihood typically has an upside-down-U shape as a function of $\lambda$, which in many cases has maximum that is close to the maximum of the test-loglikelihood curve obtained in cross-validation.

This approach to setting hyperparameters forms the first step of an inference procedure known as *Empirical Bayes*. Empirical Bayes is a two-step inference method consisting of:

1. Set hyperparameters by maximum marginal likelihood

   $$\hat{\lambda} = \arg \max P(Y|X, \lambda)$$

2. Perform MAP inference for weights given the hyperparameter estimate

   $$\hat{w} = \arg \max_{\vec{w}} P(\vec{w}|Y, X, \hat{\lambda})$$

### 4 Laplace Approximation

In cases where the prior is Gaussian but the observation model is non-Gaussian (e.g., Bernoulli or Poisson GLM), the posterior has no tractable analytic form. In such cases, we can obtain a Gaussian approximation to the posterior known as the *Laplace Approximation*.

The basic idea is to do a second-order Taylor series expansion of the log-posterior centered on the MAP estimate. Let $\vec{w} = \vec{w}_{map} + \vec{v}$ for some vector perturbation $\vec{v}$, and let $f(\vec{w}) = \log p(\vec{w}|X, Y, \theta)$.
Then we can write the 2nd-order Taylor series expansion of the log-posterior as:

\[ f(\hat{w}_{map} + \vec{v}) \approx f(\hat{w}_{map}) + (\nabla f(\hat{w}_{map}))^\top \vec{v} + \frac{1}{2} \vec{v}^\top (\nabla \nabla f(\hat{w}_{map})) \vec{v} \quad (9) \]

where the vector \( \vec{g} = \nabla f(\cdot) \) is the gradient of the log-posterior and the matrix \( H = \nabla \nabla f(\cdot) \) is known as the Hessian, whose \( i, j \) entry corresponds to

\[ H_{ij} = \frac{\partial^2}{\partial w_i \partial w_j} f(\vec{w}). \quad (10) \]

Since the MAP estimate is a maximum of \( f \), the gradient \( \vec{g} = 0 \), and we are left with the constant and Hessian term. Substituting \( \vec{w} = \hat{w}_{map} - \vec{v} \), we obtain:

\[ \log p(\vec{w}|X,Y,\theta) \approx \frac{1}{2}(\vec{w} - \hat{w}_{map})^\top H(\vec{w} - \hat{w}_{map}) + \text{const}, \quad (11) \]

and exponentiating gives the approximation:

\[ p(\vec{w}|X,Y,\theta) \propto \mathcal{N}(\hat{w}_{map}; \Lambda) = \frac{1}{|2\pi \Lambda|^\frac{1}{2}} e^{-\frac{1}{2}(\vec{w} - \hat{w}_{map})^\top \Lambda^{-1}(\vec{w} - \hat{w}_{map})}, \quad (12) \]

where \( \Lambda = -\frac{1}{2}H^{-1} \). This is the form of a multivariate normal distribution with covariance equal to \( \Lambda \). Since we know how to compute the normalizing constant for a Gaussian distribution, we can write the full Laplace approximation as:

\[ p(\vec{w}|X,Y,\theta) \approx \mathcal{N}(\hat{w}_{map}; \Lambda) = \frac{1}{|2\pi \Lambda|^\frac{1}{2}} e^{-\frac{1}{2}(\vec{w} - \hat{w}_{map})^\top \Lambda^{-1}(\vec{w} - \hat{w}_{map})}. \quad (13) \]

### 4.1 Laplace approximation for Poisson-GLM

To show that this is a (relatively) easy thing to do, let’s derive the Laplace approximation for the simplest kind of Poisson-GLM. If we use the canonical nonlinearity \( f(\cdot) = \exp(\cdot) \), which is equivalent to using the canonical “log” link function, the log-likelihood can be written:

\[ \log P(Y|X,\vec{w}) = Y^\top \log f(X\vec{w}) - I^\top f(X\vec{w}) \]

\[ = Y^\top X\vec{w} - I^\top e^{X\vec{w}}. \quad (15) \]

We can then compute the Hessian of the log-likelihood as follows:

\[ \nabla \nabla \log P(Y|X,\vec{w}) = \nabla \nabla \left( Y^\top X\vec{w} - I^\top e^{X\vec{w}} \right) \]

\[ = \nabla \left( X^\top Y - X^\top e^{X\vec{w}} \right) \quad (16) \]

\[ = -X^\top \Gamma X, \quad (17) \]

where \( \Gamma = \text{diag}(e^{X\vec{w}}) \) is a diagonal matrix with the vector \( e^{X\vec{w}} \) along the main diagonal.

If the prior is Gaussian, \( \vec{w} \sim \mathcal{N}(0, C) \), then the Hessian of the log-posterior is:

\[ \nabla \nabla p(\vec{w}|X,Y,\theta) = H = -X^\top \Gamma X - C^{-1}, \quad (19) \]
since the Hessian of the log-prior is simply \(-C^{-1}\).

Thus the posterior covariance matrix for the Laplace approximation is given by:

\[ \Lambda = -H^{-1} = (X^\top \Gamma X + C^{-1})^{-1}. \]  

(20)

(It’s a little bit more work to derive the Hessian for a Poisson-GLM with other choices of nonlinearity, since we have to handle the contribution from the spiking term \(Y^\top \log f(X\bar{w})\), but not too much!)

### 4.2 Approximate Marginal Likelihood

Lastly, we can use the Laplace approximation to obtain a handy approximation to the marginal likelihood by reversing the terms in Bayes rule:

\[
P(Y|X, \theta) \approx \frac{\text{Likelihood} \times \text{Prior}}{(\text{approximate posterior})} = \frac{P(Y|X, \bar{w})N(\bar{w}|0, C)}{N(\bar{w}|\hat{w}_{map}, \Lambda)}, \tag{21}
\]

and evaluating the right-hand-side at \(\bar{w} = \hat{w}_{map}\) gives the formula:

\[
P(Y|X, \theta) \approx \frac{P(Y|X, \hat{w}_{map})N(\hat{w}_{map}|0, C)}{N(\hat{w}_{map}|\hat{w}_{map}, \Lambda)}. \tag{22}
\]

We can use this approximation to evaluate marginal likelihood at a grid of hyperparameter values \(\theta\) and thereby perform empirical Bayesian inference for \(\bar{w}\).