Implementing the Symmetric Prior

Kevin S. Van Horn

July 13, 2015

In a previous note [1] I proposed a prior on the regression coefficients for a nominal input variable that leads to a symmetric prior on the effects:

$$\begin{aligned}
\beta &\sim \text{Normal}\left(\mathbf{0}, W\Sigma W'\right) \\
W &= X^{-1} \\
\Sigma_{jk} &= \begin{cases} \sigma^2 & \text{if } j = k \\ \rho\sigma^2 & \text{if } j \neq k \end{cases} \\
\rho &= -\frac{1}{K-1}
\end{aligned}$$

where β is the vector of regression coefficients, K is the number of levels, X and Σ are square matrices with $K-1$ rows/columns, row k of X is the encoding for level $k \neq K$, and level K is encoded as $-\sum_{k=1}^{K-1} X_k$. This gives an induced prior on the K effects $\alpha_k = X_k\beta$ that is symmetric in the effects, has a mean of 0 and variance of σ^2 for each α_k, and is degenerate, in that the sum of the effects is 0.

It may be desirable to avoid actually constructing the matrix Σ, either for uniformity in the implementation of a Bayesian regression model (e.g., independent normal priors for each regression coefficient over the entire set of predictor variables) or because K is large. In this note I discuss two ways of achieving this goal.

1 Small to moderate number of levels

When K is small or moderate, we may achieve independent normal distributions for each of the regression coefficients β_k by careful choice of encoding. If X is chosen such that $W\Sigma W' = \sigma^2 I$, then the prior

$$\beta_k \sim \text{Normal}\left(0, \sigma\right) \quad \text{for } 1 \leq k \leq K-1$$

gives us the desired symmetric prior on the effects.

To find such an encoding X we begin with an eigendecomposition of Σ: eigenvalues $\lambda_1, \ldots, \lambda_{K-1}$ and corresponding orthonormal eigenvectors $u_1, \ldots, u_{K-1}$ of Σ. Then

$$\Sigma = U \Lambda U^{-1}$$

where U is the matrix obtained by stacking the eigenvectors side by side, and Λ is the diagonal matrix whose k-th diagonal element is λ_k:

$$\begin{aligned} U &= (u_1, \ldots, u_{K-1}) \\ \Lambda &= \text{diagonal}\,(\lambda_1, \ldots, \lambda_{K-1}) \,. \end{aligned}$$

We choose

$$\begin{aligned} X &= UD \\ D &= \text{diagonal}\,(d_1, \ldots, d_{K-1}) \\ d_k &= \frac{\lambda_k^{1/2}}{\sigma}. \end{aligned}$$

The eigenvector matrix U has the property that $U' = U^{-1}$. Then

$$W = D^{-1}U^{-1} = D^{-1}U',$$

giving

$$\begin{aligned} W \Sigma W' &= \left(D^{-1}U^{-1}\right)\left(U \Lambda U^{-1}\right)\left(U D^{-1}\right) \\ &= D^{-1} \Lambda D^{-1} \\ &= \text{diagonal}\left(\lambda_1/d_1^2, \ldots, \lambda_{K-1}/d_{K-1}^2\right) \\ &= \text{diagonal}\left(\sigma^2, \ldots, \sigma^2\right). \end{aligned}$$

As shown in Appendix A, the orthonormal eigenvectors and corresponding eigenvalues of of Σ are

$$\begin{aligned} u_1 &= (K-1)^{-1/2}\,(1, \ldots, 1)' \\ \lambda_1 &= \frac{\sigma^2}{K-1} \\ d_1 &= (K-1)^{-1/2} \end{aligned}$$

and, for $2 \le k \le K-1$,

$$\begin{aligned}
u_{ki} &= (k(k-1))^{-1/2} \cdot \begin{cases} -1 & \text{if } i < k \\ k-1 & \text{if } i = k \\ 0 & \text{if } i > k \end{cases} \\
\lambda_k &= \frac{K\sigma^2}{K-1} \\
d_k &= \left(\frac{K}{K-1}\right)^{1/2}.
\end{aligned}$$

Using $X = UD$, which multiplies column k of U by d_k, we have

$$X_{ik} = d_k u_{ki};$$

in particular,

$$\begin{aligned}
X_{i1} &= \frac{1}{K-1} \\
X_{ik} &= 0 \text{ if } 1 < k < i \\
X_{ii} &= \left(\frac{K\,(k-1)}{k\,(K-1)}\right)^{1/2} \quad \text{if } i \neq 1 \\
X_{ik} &= -\left(\frac{K}{k(k-1)\,(K-1)}\right)^{1/2} \quad \text{if } k > i.
\end{aligned}$$

This gives the required encodings for the first $K-1$ levels. The encoding for level K is the negative sum of the encodings for levels 1 to $K-1$; as shown in Appendix B, this is

$$X_K = (-1, 0, \ldots, 0)\,.$$

As a check, the $K \times K$ covariance matrix for the full effects vector

$$\tilde{\alpha} = \begin{pmatrix} \alpha \\ \alpha_K \end{pmatrix}$$

is $\sigma^2 \tilde{X}\tilde{X}'$, where $\tilde{X}$ is obtained by appending X_K to X as an additional row. The Mathematica program of Appendix C verifies that

$$\tilde{X}\tilde{X}' = \begin{pmatrix} 1 & \rho & \cdots & \rho & \rho \\ \rho & 1 & \cdots & \rho & \rho \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \rho & \rho & \cdots & 1 & \rho \\ \rho & \rho & \cdots & \rho & 1 \end{pmatrix} \quad \text{where } \rho = -\frac{1}{K-1}$$

for all K from 3 to 100, as expected.

2 Large number of levels

If K is large, as occurs in a multilevel model with a prior on σ, the above approach may be inefficient. Rather than doing a dot product of a level encoding with a vector of regression coefficients, it may be preferable to instead

1. directly create a vector of effects α_k, $1 \leq k \leq K-1$, with prior

$$\alpha \sim \text{Normal}\left(\mathbf{0}, \Sigma\right);$$

2. compute $\alpha_K = -\sum_{i=1}^{K-1} \alpha_k$; then

3. use the level k to index into the extended vector (α, α_K).

If we are estimating the model using Hamilton Monte Carlo or similar methods, such as the NUTS sampler used in Stan [2], then there is the question of efficiently computing the log of the prior density for α. We would like to avoid actually constructing the large matrix Σ. First note the following:

1. The inverse of Σ is

$$\Lambda = \frac{K-1}{\sigma^2 J}\left(I + J\right)$$

   where $J = \mathbf{11'}$ is the $(K-1) \times (K-1)$ matrix whose elements are all 1. Furthermore,

$$\alpha' \Lambda \alpha = \tilde{\sigma}^{-2} \sum_{k=1}^{K} \alpha_k^2$$

   where

$$\tilde{\sigma}^2 = \frac{K}{K-1}\sigma^2$$

   (See Appendix D.)

2. Since $\Sigma = \sigma^2 S$, where S is a matrix that is a function only of K and not of σ, we have

$$\det \Sigma = \sigma^{2(K-1)} \det S$$

   where $\det S$ has no dependence on σ.

Then, writing $\equiv$ for "equal except for an additive term that has no dependence on α or σ," we have

$$\begin{aligned}
\log\left(\text{Normal}\left(\alpha \mid \mathbf{0}, \Sigma\right)\right) &\equiv -\frac{1}{2}\log\det\Sigma - \frac{1}{2}\alpha'\Lambda\alpha \\
&\equiv -\left(K-1\right)\log\sigma - \frac{1}{2}\tilde{\sigma}^{-2}\sum_{k=1}^{K}\alpha_k^2 \\
&\equiv \sum_{k=1}^{K-1}\log\left(\text{Normal}\left(\alpha_k \mid 0, \tilde{\sigma}\right)\right) - \frac{1}{2}\tilde{\sigma}^{-2}\alpha_K^2.
\end{aligned}$$

In a Stan model the implementation would look something like this:

```
transformed parameters {
  ...
  vector[K] alpha1;
  ...
  for (k in 1:(K-1))
    alpha1[k] <- alpha[k];
  alpha1[K] <- -sum(alpha);
}
model {
  ...
  increment_log_prob(
    -(K-1)*log(sigma)
    -(K-1)/(2*K*sigma^2)*dot_self(alpha1));
}
```

References

[1] Van Horn, Kevin S. (2015). "A Symmetric Prior for the Regression Coefficients of a Nominal Input Variable." http://bayesium.com/wp-content/uploads/2015/07/symmetric-effects-priors.pdf.

[2] Stan Development Team (2015). "Stan: A C++ Library for Probability and Sampling, Version 2.6.0." http://mc-stan.org.

A Eigendecomposition of Σ

The first eigenvector e_1 is a vector of all 1's:

$$\begin{aligned} e_{1,i} &= 1 \\ \lambda_1 &= \frac{\sigma^2}{K-1}. \end{aligned}$$

This eigenvector has norm $\|e_1\| = K - 1$, and so the corresponding normalized eigenvector is

$$u_1 = \frac{e_1}{(K-1)^{1/2}}.$$

To verify that e_1 is an eigenvector with eigenvalue λ_1, let $v = \Sigma\, e_1$. Then

$$\begin{aligned} v_i &= \sum_{j=1}^{K-1} \Sigma_{ij} e_{1,j} \\ &= (K-2)\rho\sigma^2 + \sigma^2 \\ &= -\frac{K-2}{K-1}\sigma^2 + \frac{K-1}{K-1}\sigma^2 \\ &= \frac{1}{K-1}\sigma^2 \\ &= \frac{\sigma^2}{K-1} e_{1,i}. \end{aligned}$$

There are also $K - 2$ orthogonal (but unnormalized) eigenvectors e_k, $2 \leq k \leq K - 1$, defined by

$$e_{ki} = \begin{cases} -1 & \text{if } i < k \\ k-1 & \text{if } i = k \\ 0 & \text{if } i > k \end{cases}$$

all with the same eigenvalue:

$$\lambda_k = \frac{K\sigma^2}{K-1}.$$

These eigenvectors have norm

$$\|e_k\| = (k-1)^2 + k - 1 = k(k-1)$$

and so the corresponding normalized eigenvectors are

$$u_k = \frac{e_k}{\left(k(k-1)\right)^{1/2}}.$$

To verify that these are indeed eigenvectors with the claimed eigenvalues, let $v = \Sigma\, e_k$. Then

$$v_i = -\sum_{j=1}^{k-1} \Sigma_{ij} + (k-1)\,\Sigma_{ik}$$

for all i. We will use the fact that

$$\frac{K}{K-1} = 1 - \rho.$$

If $i < k$ then

$$\begin{aligned} v_i &= -\left((k-2)\,\rho\sigma^2 + \sigma^2\right) + (k-1)\,\rho\sigma^2 \\ &= \rho\sigma^2 - \sigma^2 \\ &= (1-\rho)\,\sigma^2 e_{ki}. \end{aligned}$$

If $i = k$ then

$$\begin{aligned} v_i &= -(k-1)\,\rho\sigma^2 + (k-1)\,\sigma^2 \\ &= (1-\rho)\,(k-1)\,\sigma^2 \\ &= (1-\rho)\,\sigma^2 e_{ki}. \end{aligned}$$

If $i > k$ then

$$\begin{aligned} v_i &= -(k-1)\rho\sigma^2 + (k-1)\,\rho\sigma^2 \\ &= 0 \\ &= (1-\rho)\,\sigma^2 e_{ki}. \end{aligned}$$

Now we must verify that the eigenvectors are orthogonal to each other. Σ is a symmetric, real matrix, and so eigenvectors corresponding to different eigenvalues are guaranteed to be orthogonal. In particular, e_1 and e_k are orthogonal for $k \geq 2$. The remaining $K-2$ eigenvectors all have the same eigenvalue, so we must show that they are orthogonal.

For $j < k$, $2 \leq j, k \leq K-1$, we find

$$e_{ki} e_{ji} = \begin{cases} 1 & \text{if } i < j \\ -(j-1) & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

and hence

$$e_k' e_j = (j-1) - (j-1) = 0.$$

B Encoding of level K

We use $X_K = -\sum_{i=1}^{K-1} X_i$ and work out the specific value of the row vector X_K. Its first element is given by

$$X_{K1} = -\sum_{i=1}^{K-1} \frac{1}{K-1} = -1.$$

Element k, for $k \geq 2$, is given by

$$\begin{aligned}
X_{Kk} &= -\sum_{i=1}^{K-1} X_{ik} \\
&= -(k-1)\left(\frac{K}{k(k-1)\,(K-1)}\right)^{1/2} + \left(\frac{Kk-K}{Kk-k}\right)^{1/2} \\
&= -\left(\frac{K(k-1)}{k\,(K-1)}\right)^{1/2} + \left(\frac{Kk-K}{Kk-k}\right)^{1/2} \\
&= 0
\end{aligned}$$

and so the encoding for level K is given by

$$X_K = (-1, 0, \ldots, 0)\,.$$

C Verifying the encoding

This is the Mathematica program used to verify that $\tilde{X}\tilde{X}'$ is a matrix with 1's on the diagonal and $-1/(K-1)$ at every non-diagonal position, for every K from 3 to 100:

```
Enc[K_, i_, k_] :=
  Which[i == K, If[k == 1, -1, 0],
        k == 1, 1/(K - 1),
        k < i, 0,
        k == i, Sqrt[K (k - 1)/(k (K - 1))],
        k > i, -Sqrt[K/(k (k - 1) (K - 1))]]
Encoding[K_] := Table[Enc[K, i, k], {i, K}, {k, K - 1}]
EffectsCovMatrix[K_] := Encoding[K].Transpose[Encoding[K]]
TargetCovMatrix[K_] :=
  Table[If[i == k, 1, -1/(K - 1)], {i, K}, {k, K}]
CheckEncoding[K_] := (EffectsCovMatrix[K] == TargetCovMatrix[K])
Check3to100 = Table[CheckEncoding[K], {K, 3, 100}]
And @@ Check3to100
```

D Inverse of Σ

We claim that $\Sigma^{-1} = \Lambda$, where

$$\Lambda = \frac{K-1}{\sigma^2 K}(I+J)$$

and $J = \mathbf{1}\mathbf{1}'$ is the $(K-1) \times (K-1)$ matrix whose elements are all 1. Using

$$\begin{aligned} \Sigma &= (1-\rho)\,\sigma^2 I + \rho\sigma^2 J \\ &= \frac{\sigma^2}{K-1}(KI - J) \end{aligned}$$

we verify this claim by showing that $\Lambda\Sigma = I$:

$$\begin{aligned} \Lambda\Sigma &= \frac{K-1}{\sigma^2 K}(I+J)\frac{\sigma^2}{K-1}(KI-J) \\ &= K^{-1}\left(KI - J + KJ - J^2\right) \\ &= K^{-1}(KI - J + KJ - (K-1)J) \\ &= I. \end{aligned}$$

Furthermore, letting

$$c = \frac{K-1}{\sigma^2 K}$$

we have

$$\begin{aligned} \alpha'\Lambda\alpha &= c\alpha'(I+J)\,\alpha \\ &= c\left(\alpha'\alpha + \alpha'\mathbf{1}\mathbf{1}'\alpha\right) \\ &= c\left(\alpha'\alpha + \left(\alpha'\mathbf{1}\right)^2\right) \\ &= c\left(\alpha'\alpha + \alpha_K^2\right) \\ &= c\left(\sum_{k=1}^{K}\alpha_k^2\right). \end{aligned}$$