Vector Autoregressions (VARs)

Matthew Cocci

1. Intuition

Let's build up to the intuition of VARs by starting of its relatives, the zero-mean AR(1):

$$x_t = \varphi x_{t-1} + \varepsilon_t$$

The AR(1) forecasts *this* period's value of some variable, x, given the *last* period. But we don't need to stop with just last period's lags. We can include arbitrarily many:

$$x_t = a_1 x_{t-1} + a_2 x_{t-2} + \cdots + a_p x_{t-p} + \varepsilon_t$$

This gives us the AR(p) model. But again, we don't need to stop there. Suppose that we include the lagged values of *other* variables that might help predict x, like say y:

$$x_t = a_1 x_{t-1} + \cdots + a_p x_{t-p} + b_1 y_{t-1} + \cdots + b_p y_{t-p} + \varepsilon_t \tag{1}$$

And presumably, if x helps predict y, why not use y to forecast x? Perhaps they're jointly determined, as we might expect in economic equilibrium or any situation with endogeneity. So let's write

$$y_t = c_1 x_{t-1} + \cdots + c_p x_{t-p} + d_1 y_{t-1} + \cdots + d_p y_{t-p} + \eta_t \tag{2}$$

As you can guess, we can simplify this a lot. In fact, we might toss Equations 1 and 2 into matrices and vectors to simplify notation. This will be particularly useful when we have lots and lots of variables, parameters, and lags. So we might write

$$\begin{pmatrix} x_t \\ y_t \end{pmatrix} = \begin{pmatrix} a_1 & b_1 & \cdots & a_p & b_p \\ c_1 & d_1 & \cdots & c_p & d_p \end{pmatrix} \begin{pmatrix} x_{t-1} \\ y_{t-1} \\ \vdots \\ x_{t-p} \\ y_{t-p} \end{pmatrix} + \begin{pmatrix} \varepsilon_t \\ \eta_t \end{pmatrix} \tag{3}$$

And viola. We have a bona fide *vector autoregression*, or VAR.

We arrived at it through successive generalizations of very natural intuitive ideas. So if you know time series, and if you know linear algebra, you're home. You know this stuff already. And had you done this 40 years ago and pushed the technique and consequences to their natural implications, you might have even won a Nobel Prize.

2. Definition and Notation

Now, we'll generalize to non-zero mean process, and we'll fix our notation, which was a bit sloppy above. So let y_t denote an $n \times 1$ vector of observables we want to predict.

Now define the VAR(p) model as follows:

$$y_t = \phi_0 + \phi_1 y_{t-1} + \cdots + \phi_p y_{t-p} + u_t \qquad u_t \sim \mathrm{N}_n(0, \Sigma) \tag{4}$$

where ϕ_0 is $n \times 1$ and where Σ and ϕ_i $(i = 1, \ldots, p)$ are $n \times n$. So clearly, y_t can be quite a complicated linear function of its previous lags and its components.

Next, let's write Equation 4 in more compact, matrix notation. Specifically, concatenate the ϕ_i column vectors horizontally, and stack the lags of y into one big column vector:

$$y_t = \underbrace{\begin{pmatrix}\phi_1 & \cdots & \phi_p & \phi_0\end{pmatrix}}_{\Phi'} \underbrace{\begin{pmatrix} y_{t-1} \\ \vdots \\ y_{t-p} \\ 1 \end{pmatrix}}_{x_t} + u_t \qquad u_t \sim \mathrm{N}_n(0, \Sigma)$$

$$\Leftrightarrow \qquad y_t = \Phi' x_t + u_t \tag{5}$$

where Φ is $(np + 1) \times n$ and x_t is $(np + 1) \times 1$.

3. OLS Estimator

Now that we have the model, let's find the OLS for the parameters in Φ. To do so, we'll take one element (or component) of the vector y_t at a time, so one row of y_t and Φ at a time. So let's minimize the sum of squared errors for column/component i:

$$\min_{\Phi'_{\cdot i}} \sum_{i=1}^{T} \left(y_t^{(i)} - \Phi'_{\cdot i} x_t \right)^2 \tag{6}$$

Now this estimator looks just like standard multivariate OLS regression, resulting in the "the normal equation":

$$0 = \frac{d}{d\Phi'_{\cdot i}} \left\{ \sum_{i=1}^{T} \left(y_t^{(i)} - \Phi'_{\cdot i} x_t \right)^2 \right\} = -2 \sum_{i=1}^{T} x_t' \left(y_t^{(i)} - \Phi'_{\cdot i} x_t \right)$$

$$\Rightarrow \quad \hat{\Phi}_{\cdot i} = \frac{\sum_{t=1}^{T} x_t' y_t^{(i)}}{\sum_{t=1}^{T} x_t' x_t} = (X'X)^{-1} X' Y^{(i)}$$

where X is $T \times (np + 1)$ and $Y^{(i)}$ is $T \times 1$ where

$$X = \begin{pmatrix} x_1' \\ \vdots \\ x_T' \end{pmatrix} \qquad Y^{(i)} = \begin{pmatrix} y_1^{(i)} \\ \vdots \\ y_T^{(i)} \end{pmatrix} \tag{7}$$

So now we have the estimator for each *column* of Φ, which allows us to write

$$\hat{\Phi} = \begin{pmatrix} \hat{\Phi}_{\cdot 1} & \cdots & \hat{\Phi}_{\cdot n} \end{pmatrix} = (X'X)^{-1}X'Y \tag{8}$$

where X is as above and Y is $T \times n$, which is a matix with y_t' stacked on top of each other for $t = 1, \ldots, T$. So it's the $Y^{(i)}$ defined above placed next to each other in a row.

This notation we've developed also gives us a convenient way to define the *sum of squared OLS residuals* matrix:

$$\hat{S} = (Y - X\hat{\Phi})'(Y - X\hat{\Phi}) \tag{9}$$

Eventually, when we start doing the Bayesian computations, we'll use this in the multivariate analog of a result we recall from univariate mathematical statistics, both of which are written below:

$$\begin{aligned} \sum_{i=1}^{m}(y_i - \mu)^2 &= \sum_{i=1}^{m}(y_i - \hat{\mu})^2 + m(\hat{\mu} - \mu)^2 \\ (Y - X\Phi)'(Y - X\Phi) &= \hat{S} + (\Phi - \hat{\Phi})'X'X(\Phi - \hat{\Phi}) \end{aligned} \tag{10}$$

where anything with " $\hat{}$ " denotes the sample estimate/analog.

4. Likelihood Function

Now that we have an easy representation of the our VAR(p) model, we move to the likelihood function. By our definition, y_t, conditional on x_t (the lags $y_{t-1}, \ldots, y_{t-p}$), happens to be normally distributed:

$$\begin{aligned} & p(y_t \mid x_t, \Phi, \Sigma) \sim \mathrm{N}_n(\Phi x_t,\ \Sigma) \\ \Rightarrow \quad & p(y_t \mid x_t, \Phi, \Sigma) \propto |\Sigma|^{-1/2} \exp\left\{ -\frac{1}{2}\left(y_t - \Phi' x_t\right)' \Sigma^{-1} (y_t - \Phi' x_t) \right\} \end{aligned} \tag{11}$$

Now we use the trick from the trace section in the appendix to rewrite the previous line

$$\Rightarrow \quad p(y_t \mid x_t, \Phi, \Sigma) \propto |\Sigma|^{-1/2} \exp\left\{ -\frac{1}{2}\mathrm{tr}\left[\Sigma^{-1}(y_t - \Phi' x_t)\left(y_t - \Phi' x_t\right)'\right] \right\} \tag{12}$$

5. Joint Density Function

Now, we construct the *joint density function* for the entire series, $Y_{1:T}$, where

$$Y_{t_0:t_1} = (y_{t_0}, \ldots, y_{t_1})$$

Therefore, conditional on a presample $y_{-p+1}, \ldots, y_0$, the jdf is written

$$\begin{aligned} p(Y_{1:T} \mid Y_{-p+1:0}, \Phi, \Sigma) &= \prod_{t=1}^{T} p(y_t \mid Y_{-p+1:t-1}, \Phi, \Sigma) = \prod_{t=1}^{T} p(y_t \mid Y_{t-p:t-1}, \Phi, \Sigma) \qquad (13) \\ &\propto \prod_{t=1}^{T} |\Sigma|^{-\frac{1}{2}} \exp\left\{ -\frac{1}{2} \text{tr}\left[\Sigma^{-1} (y_t - \Phi' x_t)(y_t - \Phi' x_t)' \right] \right\} \end{aligned}$$

where we could make the jump in Equation 13 because the density of y_t depends only on the previous p lags, not the entire history up until t. And given that the trace of a sum of two matrices is the sum of the traces, we can simplify the jdf

$$\begin{aligned} p(Y_{1:T} \mid Y_{-p+1:0}, \Phi, \Sigma) &\propto |\Sigma|^{-\frac{T}{2}} \exp\left\{ -\frac{1}{2} \text{tr}\left[\Sigma^{-1} \sum_{t=1}^{T} (y_t - \Phi' x_t)(y_t - \Phi' x_t)' \right] \right\} \\ &\propto |\Sigma|^{-\frac{T}{2}} \exp\left\{ -\frac{1}{2} \text{tr}\left[\Sigma^{-1} (Y - X\Phi)'(Y - X\Phi) \right] \right\} \qquad (14) \end{aligned}$$

where X and Y are defined as above in Equation 7 and the subsequent section discussion.

Alternative Representation Now, we use a few tricks to rewrite the jdf representation in Equation 14. (This will aid in posterior inference.) Namely, we'll want to take care of the nastiness in the trace operator, so let's simplify that straight away (letting $\hat{\beta} = \text{vec}(\hat{\Phi})$, a vector of coefficients):

$$\begin{aligned} \text{By Eqn 10:} \quad \text{tr}\left[\Sigma^{-1} (Y - X\Phi)'(Y - X\Phi) \right] &= \text{tr}\left[\Sigma^{-1} \left\{ \hat{S} + (\Phi - \hat{\Phi})' X' X (\Phi - \hat{\Phi}) \right\} \right] \\ &= \text{tr}\left[\Sigma^{-1} \hat{S} \right] + \text{tr}\left[\Sigma^{-1} (\Phi - \hat{\Phi})' X' X (\Phi - \hat{\Phi}) \right] \\ \text{By Eqn 17} \quad &= \text{tr}\left[\Sigma^{-1} \hat{S} \right] + \text{tr}\left[(\Phi - \hat{\Phi})' X' X (\Phi - \hat{\Phi}) \Sigma^{-1} \right] \\ \text{By Eqn 19} \quad &= \text{tr}\left[\Sigma^{-1} \hat{S} \right] + (\beta - \hat{\beta})' \left[\Sigma^{-1} \otimes (X'X) \right] (\beta - \hat{\beta}) \\ \text{By Kronecker Prop. 3} \quad &= \text{tr}\left[\Sigma^{-1} \hat{S} \right] + (\beta - \hat{\beta})' \left[\Sigma \otimes (X'X)^{-1} \right]^{-1} (\beta - \hat{\beta}) \end{aligned}$$

Now substitute this back into the jdf to get

$$\begin{aligned} p(Y_{1:T} \mid Y_{-p+1:0}, \Phi, \Sigma) &\propto |\Sigma|^{-\frac{T}{2}} \exp\left\{ -\frac{1}{2} \text{tr}\left[\Sigma^{-1} \hat{S} \right] \right\} \\ &\quad \times \exp\left\{ -\frac{1}{2} (\beta - \hat{\beta})' \left[\Sigma \otimes (X'X)^{-1} \right]^{-1} (\beta - \hat{\beta}) \right\} \qquad (15) \end{aligned}$$

6. Bayesian Analysis

So we saw above how to arrive at a new representation for the jdf in Equation 15, and in this section we'll see why that was useful.

6.1. Flat Prior, Known Variance

This is perhaps the easiest case, where we want to find the distributions for Φ assuming a flat prior for Φ and that Σ is given. This gives us a posterior proportional to the likelihood, which we pull straight from equation 15:

$$\begin{aligned} p(\Phi \mid Y, \Sigma) &\propto p(Y \mid \Phi, \Sigma) p(\Phi \mid \Sigma) \propto p(Y \mid \Phi, \Sigma) \\ &\propto |\Sigma|^{-\frac{T}{2}} \exp\left\{-\frac{1}{2}\text{tr}\left[\Sigma^{-1}\hat{S}\right]\right\} \exp\left\{-\frac{1}{2}(\beta-\hat{\beta})'\left[\Sigma \otimes (X'X)^{-1}\right]^{-1}(\beta-\hat{\beta})\right\} \end{aligned}$$

6.2. Minnesota Prior

The Minnesota Prior is a procedure to implement a prior using dummy observations (X^*, Y^*), which are built from a small presample. The fact that we use dummy observations to implement the prior implies that the parameters have a prior distribution conjugate to the likelihood function of the data.

The dummy observiations we will use to implement the Minnesota prior center the distribution at a random walk. This implies that the coefficient on the first lag will cluster near unity, while the other coefficients are centered around zero—an approach favoring persistence in shocks to the data series. The relative "strength," "tightness," or "influence" of this prior can be dialed up or down as well via the approach. Finally, we will conclude the Minnesota prior implementation by considering different restrictions we might want to place on how coefficients relate to each other.[1]

Now that we have the background, let's detail the procedure to form the dummy observations (Y^*, X^*):

1. First, estimate the mean vector, $\bar{y}$, and the standard deviation vector, $\hat{s}$, based on a short presample of the data from time $-\tau$ to 0 (given that we start at time 1.

$$\bar{y} = \text{mean}\left(Y_{-\tau:0}\right) \qquad \hat{s} = \text{std}\left(Y_{-\tau:0}\right)$$

2. Next, we'll choose hyperparameters λ_1 and λ_2, which control the tightness of our parameters. We'll see below how they factor in.

[1] The details here need not be followed exactly. In particular, if the data series are detrended and seem stationary, a prior centered around a random walk would be a poor choice. In that case, this approach should be modififed to center all parameters at 0. Similarly, the general guidelines described below can be easily modified to suit any assumed distribution for the parameters.

3. Set up dummies for the *first lag* on the n components of the dependent variable vector, centering the prior around unity (a random walk):

$$
\begin{aligned}
Y^*_{1:n} &= X^*_{1:n}\Phi + U \\
\begin{pmatrix} \lambda_1\hat{s}_1 & 0 & \cdots & 0 \\ 0 & \lambda_1\hat{s}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_1\hat{s}_n \end{pmatrix} &= \begin{pmatrix} \lambda_1\hat{s}_1 & 0 & \cdots & 0 & \underbrace{0 & \cdots & 0}_{n(p-1)+1 \text{ columns of zeros}} \\ 0 & \lambda_1\hat{s}_2 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_1\hat{s}_n & 0 & \cdots & 0 \end{pmatrix} \Phi + \begin{pmatrix} u_{11} & \cdots & u_{1n} \\ u_{11} & \cdots & u_{1n} \\ \vdots & \ddots & \vdots \\ u_{n1} & \cdots & u_{nn} \end{pmatrix}
\end{aligned}
$$

   Now let's take a moment to understand what this does. We just specified n observations above, each observation corresponding to a different row of the relation we just wrote. These n "dummy observations" will simply be tacked onto our sample—appended to the bottom of our sample vector and matrices, X and Y. This allows us to estimate Φ via simple OLS (as above).

   But we know that adding these observations implement a prior. Now just what kind of prior is that exactly? Well, to find out, we upack the matrix relationship we wrote above, considering the generic ith row (and ith dummy observation) for generality. Letting ϕ_{ij} be the ij entry of Φ, we find that the ith row imposes

$$
\begin{aligned}
\lambda_1\hat{s}_i &= (\lambda_1\hat{s}_i)\,\phi_{ii} + u_{ii} \qquad i = 1, \ldots, n \\
0 &= (\lambda_1\hat{s}_i)\,\phi_{ji} + u_{ji} \qquad \{j \neq i \mid j = 1, \ldots, n\}
\end{aligned}
$$

   Rearranging

$$
\begin{aligned}
\phi_{ii} = 1 - \frac{u_{ii}}{\lambda_1\hat{s}_i} \quad &\Rightarrow \quad \phi_{ii} \sim \mathrm{N}\left(1,\ \frac{\Sigma_{ii}}{\lambda_1^2\hat{s}_i^2}\right) \qquad i = 1, \ldots, n \\
\phi_{ji} = -\frac{u_{ji}}{\lambda_1\hat{s}_i} \quad &\Rightarrow \quad \phi_{ji} \sim \mathrm{N}\left(0,\ \frac{\Sigma_{ji}}{\lambda_1^2\hat{s}_i^2}\right) \qquad \{j \neq i \mid j = 1, \ldots, n\}
\end{aligned}
$$

   Thus we can see how the parameter λ_1 (which must be chosen and cannot be inferred) affects the prior "tightness" via the prior variance.

4. Now we set up dummies for the *remaining lags*, all of which we will center at zero. So for the ℓth lag (where $\ell = 2, \ldots, p$), we will set up the following:

$$
\begin{aligned}
Y^*_{a:b} = X^*_{a:b}\Phi + U \qquad \text{where } &\begin{cases} a = n(\ell-1)+1 \\ b = n\ell \end{cases} \\
(\mathbf{0})_{n\times n} = \begin{pmatrix} \underbrace{0 & \cdots & 0}_{n(\ell-1) \text{ cols of zeros}} & \lambda_1\hat{s}_1\ell^{\lambda_2} & 0 & \cdots & 0 & \underbrace{0 & \cdots & 0}_{n(p-\ell)+1 \text{ cols of zeros}} \\ 0 & \cdots & 0 & 0 & \lambda_1\hat{s}_2\ell^{\lambda_2} & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & 0 & \cdots & \lambda_1\hat{s}_n\ell^{\lambda_2} & 0 & \cdots & 0 \end{pmatrix} \Phi + U
\end{aligned}
$$

Similar to what we saw above, we find that this setup (for all $\ell = 2, \ldots, p$) implies the following equations and, as a result, the following prior:

$$0 = \left(\lambda_1 \hat{s}_i \ell^{\lambda_2}\right) \phi_{ji} + u_{ji} \qquad \begin{cases} \ell = 2, \ldots, p \\ j = n(\ell - 1) + 1, \ldots, \ell n \\ i = 1, \ldots, n \end{cases}$$

$$\Rightarrow \quad \phi_{ji} \sim \mathrm{N}\left(0,\ \frac{\Sigma_{ji}}{\lambda_1^2 \hat{s}_i^2 \ell^{2\lambda_2}}\right)$$

It's clear that, with λ_1 fixed from Step 3, increasing λ_2 will increasingly tighten the prior on coefficients about 0. Moreover, this effect will grow with the number of lags, placing a sensible "high bar" to yield nonzero coefficients for deeper lags once we get to the posterior.

5. Next, we set up the prior on the entries of the covariance matrix Σ. We do so by adding λ_3 entries of the following dummies:

$$\begin{pmatrix} \hat{s}_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \hat{s}_n \end{pmatrix} = \left(\mathbf{0}\right)_{n \times (np+1)} \Phi + U$$

By repeating these λ_3 dummies more or less times (by changing λ_3), we change the tightness of the prior on Σ.

6. Finally, we

A. Trace

Definition: If A is an $n \times n$ matrix, then

$$\text{tr}[A] = \sum_{i=1}^{n} a_{ii} \tag{16}$$

which is the sum of diagonal elements.

Trace Fact: If X is $m \times n$ and Y is $n \times m$, then

$$\text{tr}[XY] = \text{tr}[YX] \tag{17}$$

This isn't dificult to prove, just tedious.

Useful Trick: Suppose that a is an $n \times 1$ vector and B is a symmetric positive definite $n \times n$ matrix. Then

$$a'Ba \text{ is a scalar}$$

Then, since the trace of a scalar is just eqal to that scalar, we can rewrite

$$\begin{aligned} a'Ba &= tr\left[a'Ba\right] \\ &= tr\left[a'(Ba)\right] \end{aligned}$$

Now if we use Equation 17, taking $X = a'$ and $Y = Ba$, we can carry on from the simplification above to write

$$\begin{aligned} a'Ba &= tr\left[a'(Ba)\right] = tr\left[(Ba)a'\right] \\ &= tr\left[Baa'\right] \end{aligned}$$

B. Kronecker Product and Vec Operator

B.1. Definitions

Suppose we have two matrices, A which is $m \times n$ and B which is $p \times q$. Then the *Kronecker Product* of A and B is

$$A \otimes B = \begin{pmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{pmatrix}$$

which implies that the new matrix is $(mp) \times (nq)$.

Next, the *vec operator* takes any matrix A that is $m \times n$ and stacks to columns on top of each other (left to right) to form a column vector of length mn. Supposing that a_i are column vectors to simplify notation:

$$\text{if } A = \begin{pmatrix} a_1 & \cdots & a_n \end{pmatrix} \qquad a_i \in \mathbb{R}^{n \times 1}$$

$$\text{then vec } A = \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix}$$

B.2. Properties

Property 1 Let A be $m \times n$, B be $p \times q$, C be $n \times r$, and D be $q \times s$. Then

$$(A \otimes B)(C \otimes D) = AC \otimes BD \tag{18}$$

Property 2 $(A \otimes B)' = (A' \otimes B')$.

Property 3 $(A \otimes B)^{-1} = (A^{-1} \otimes B^{-1})$.

Property 4 $\text{tr}[A'BCD'] = \text{vec}(A)'(D \otimes B)\text{vec}(C)$.

How about some proofs?

B.3. Proof of Property 4

Of all the properties, this one strikes me the most of "Now what the hell?" There is zero intuition for this that I can see, so let's try to derive it.

$$\text{tr}\left[A'BCD'\right] = \text{vec}(A)'(D \otimes B)\text{vec}(C) \tag{19}$$

Proof. We know that the matrix that results from $A'BCD'$ must be square to apply the trace operator to it. So let's call that the beast inside the trace operator E and suppose that it is $q \times q$. Now what does that imply about the size of our other matrices?

Well to allow the matrix multiplication to be carried out (i.e. we can only multiply matrix X and matrix Y if the columns of X equal the rows of Y), and to have E be $q \times q$, this forces

$$A \text{ is } m \times q \qquad B \text{ is } m \times n \qquad C \text{ is } n \times p \qquad D \text{ is } q \times p$$

where m, n, p, and q are all left unspecified.

Okay then, let's get a move on

$$\text{tr}[A'BCD'] = \text{tr}[E] = \sum_{i=1}^{q} e_{ii}$$

Now suppose define $X = A'B$ and $Y = CD'$, implying that X is $q \times n$ and Y is $n \times q$. We can rewrite the diagonal components of E in terms of the rows of X, the $x_{i\cdot}$, and the colums of Y, $y_{\cdot i}$.

$$\text{tr}[E] = \sum_{i=1}^{q} e_{ii} = \sum_{i=1}^{q} x_{i\cdot} y_{\cdot i}$$

Now let's not forget where the rows and columns of X and Y come from. The ith row of X is the product of the ith row of A' (or the ith column of A) with each subsequent column of B. Similarly, each column of Y is the dot product of each successive row of C with the ith column of D' (or the ith row of D). Mathematically,

$$\begin{aligned}
\text{tr}[E] &= \sum_{i=1}^{q} e_{ii} = \sum_{i=1}^{q} x_{i\cdot} y_{\cdot i} \\
&= \sum_{i=1}^{q} a'_{\cdot i} \begin{pmatrix} b_{\cdot 1} & \cdots & b_{\cdot n} \end{pmatrix} \begin{pmatrix} c_{1\cdot} \\ \vdots \\ c_{n\cdot} \end{pmatrix} d'_{i\cdot} \qquad (20) \\
&= \sum_{i=1}^{q} a'_{\cdot i} BC d'_{i\cdot} \qquad (21)
\end{aligned}$$

where $a'_{\cdot i}$ is the ith column of A, transposed, and $d'_{i\cdot}$ is the ith row of D, transposed.

Alright, where did that get us? Well, let's look at that last expression. Let's suppress the three components to the write of $a'_{\cdot i}$—those nasty vectors and matrices with b's, c's, and d's in them. We'll do so by just grouping those three terms into a "back-box" term called "z_i," which we'll postpone computing for now. This let's us write

$$\text{tr}[E] = \sum_{i=1}^{q} a'_{\cdot i} z_i \tag{22}$$

where z_i is of size $m \times 1$.

But hey, that's a regular old dot product, so let's rewrite it as such with simple matrix notation:

$$\begin{aligned} \text{tr}[E] &= \sum_{i=1}^{q} a'_{\cdot i} z_i \\ &= \begin{pmatrix} a'_{\cdot 1} & \cdots & a'_{\cdot q} \end{pmatrix} \begin{pmatrix} z_1 \\ \vdots \\ z_q \end{pmatrix} = \text{vec}(A)' \, Z \end{aligned} \tag{23}$$

Now hopefully you were able to make the jump to Equation 23 yourself once you saw what we were doing. And now we at least have *something* from Equation 19. Now let's try to pin down those z_i terms that helped us get here.

Now, let's pin down that Z matrix and show that, in fact,

$$Z = (D \otimes B)\text{vec}(C) \tag{24}$$

We do so by first using Equations 21, 22, and 23 to infer the following:

$$Z = \begin{pmatrix} z_1 \\ \vdots \\ z_q \end{pmatrix} = \begin{pmatrix} BCd'_{1\cdot} \\ \vdots \\ BCd'_{q\cdot} \end{pmatrix} \tag{25}$$

Next, let's consider the jth row of the last vector, then rewrite C, expanding it into a different but equivalent representation—a row of column vectors all lined up:

$$\begin{aligned} BCd'_{j\cdot} &= B \begin{pmatrix} c_{\cdot 1} & \cdots & c_{\cdot p} \end{pmatrix} d'_{j\cdot} \\ \text{expanding } d'_{j\cdot} \quad &= B \begin{pmatrix} c_{\cdot 1} & \cdots & c_{\cdot p} \end{pmatrix} \begin{pmatrix} d_{j1} \\ \vdots \\ d_{jp} \end{pmatrix} \end{aligned}$$

Now distribute B and dot product with the $d'_{j\cdot}$:

$$\begin{aligned} BCd'_{j\cdot} &= \left(Bc_{\cdot 1}d_{j1} + \cdots + Bc_{\cdot p}d_{jp}\right) \\ \text{Since the } d_{jm} \text{ are scalars} \quad &= \left(d_{j1}Bc_{\cdot 1} + \cdots + d_{jp}Bc_{\cdot p}\right) \end{aligned}$$

Amost there, I promise.

Now stack these $BCd'_{j\cdot}$ terms to get Z as in Equation 25:

$$Z = \begin{pmatrix} z_1 \\ \vdots \\ z_q \end{pmatrix} = \begin{pmatrix} BCd'_{1\cdot} \\ \vdots \\ BCd'_{q\cdot} \end{pmatrix} = \begin{pmatrix} d_{11}Bc_{\cdot 1} + \cdots + d_{1p}Bc_{\cdot p} \\ \vdots \\ d_{q1}Bc_{\cdot 1} + \cdots + d_{qp}Bc_{\cdot p} \end{pmatrix} \tag{26}$$

Finally, take out the expanded C term from each row and rewrite the sums in each row as a result of matrix multiplication

$$Z = \begin{pmatrix} d_{11}Bc_{\cdot 1} + \cdots + d_{1p}Bc_{\cdot p} \\ \vdots \\ d_{q1}Bc_{\cdot 1} + \cdots + d_{qp}Bc_{\cdot p} \end{pmatrix} = \begin{pmatrix} d_{11}B & \cdots & d_{1p}B \\ \vdots & \ddots & \vdots \\ d_{q1}B & \cdots & d_{qp}B \end{pmatrix} \begin{pmatrix} c_{\cdot 1} \\ \vdots \\ c_{\cdot p} \end{pmatrix} \tag{27}$$

Or in other words, if you look at the last line closely

$$Z = (D \otimes B)\text{vec}(C)$$

QED (drops mic).

□