Gram-Schmidt Orthogonalization*

Steven J. Cox

This work is produced by The Connexions Project and licensed under the
Creative Commons Attribution License †

Suppose that M is an m-dimensional subspace with basis

$$\{x_1, \ldots, x_m\}$$

We transform this into an orthonormal basis

$$\{q_1, \ldots, q_m\}$$

for M via

1. Set $y_1 = x_1$ and $q_1 = \frac{y_1}{\|y_1\|}$
2. $y_2 = x_2$ minus the projection of x_2 onto the line spanned by q_1. That is

$$y_2 = x_2 - q_1\left({q_1}^T q_1\right)^{-1} {q_1}^T x_2 = x_2 - q_1 {q_1}^T x_2$$

   Set $q_2 = \frac{y_2}{\|y_2\|}$ and $Q_2 = [q_1, q_2]$.
3. $y_3 = x_3$ minus the projection of x_3 onto the plane spanned by q_1 and q_2. That is

$$y_3 = x_3 - Q_2\left({Q_2}^T Q_2\right)^{-1} {Q_2}^T x_3 = x_3 - q_1 {q_1}^T x_3$$

   Set $q_3 = \frac{y_3}{\|y_3\|}$ and $Q_3 = \{q_1, q_2, q_3\}$. Continue in this fashion through step (m).

- (m) $y_m = x_m$ minus its projection onto the subspace spanned by the columns of Q_{m-1}. That is

$$y_m = x_m - Q_{m-1}\left({Q_{m-1}}^T Q_{m-1}\right)^{-1} {Q_{m-1}}^T x_m x_m - \sum_{j=1}^{m-1} q_j {q_j}^T x_m$$

Set $q_m = \frac{y_m}{\|y_m\|}$ To take a simple example, let us orthogonalize the following basis for $\mathbb{R}^3$,

$$x_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} x_2 = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} x_3 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

1. $q_1 = y_1 = x_1$.

*Version 2.4: Jul 28, 2005 7:42 pm GMT-5
†http://creativecommons.org/licenses/by/1.0

2. $y_2 = x_2 - q_1 {q_1}^T x_2 = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix}^T$ and so, $q_2 = y_2$.
3. $y_3 = x_3 - q_1 {q_1}^T x_3 = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^T$ and so, $q_3 = y_3$.

We have arrived at

$$q_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} q_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} q_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

. Once the idea is grasped the actual calculations are best left to a machine. Matlab accomplishes this via the orth command. Its implementation is a bit more sophisticated than a blind run through our steps (1) through (m). As a result, there is no guarantee that it will return the same basis. For example

```
≫X=[1 1 1;0 1 1 ;0 0 1];

≫Q=orth(X)

Q=

 0.7370  -0.5910  0.3280

 0.5910   0.3280 -0.7370

 0.3280   0.7370  0.5910
```

This ambiguity does not bother us, for one orthogonal basis is as good as another. Let us put this into practice, via (10.8).