Lecture 3: Linear Algebra: Minimization and equilibrium, S. Ch 1 & 2

The convergence of a Markov chain density function to a steady state is easy to show for diagonalizable transition matrices **W**. What about non-diagonalizable **W**? This question is of more general interest for dynamical systems and iterative solution of equations. So, consider the powers $\mathbf{A}^n$, $n = 1, 2, \ldots$. of a m x m real matrix **A**.
Definition: The spectral radius is the maximal modulus of any eigenvalue,

$$\rho(\mathbf{A}) = \max|\lambda_i|$$

Theorem. If $\rho(\mathbf{A}) < 1$, $\lim_{n \to \infty} \mathbf{A}^n = \mathbf{0}$

We will outline the proof, leaving some details out. First, the *Schur* theorem guarantees that any square matrix can be triangularized by a unitary matrix **Q**:

$$\mathbf{A} = \mathbf{QUQ}^H, \mathbf{QQ}^H = \mathbf{I}$$

This is a similarity transformation: **U** and **A** have the same eigenvalues. **U** is upper triangular, and **Q** can be chosen to put the eigenvalues of **A** in any order on the diagonal of **U**. The proof of this relies on the fact that any matrix has an eigenvalue and an eigenvector, but does not tell how to calculate it.
So, let there be q different eigenvalues, and arrange them in blocks down the diagonal,

$$\mathbf{U} = \begin{pmatrix} \mathbf{U}_1 & x & x & x \\ 0 & \mathbf{U}_2 & x & x \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & \mathbf{U}_q \end{pmatrix}, \mathbf{U}_k = \lambda_k(\mathbf{I} + \mathbf{N}_k), k = 1, \ldots, q,$$

$$\mathbf{N}_k = \begin{pmatrix} 0 & x & x & \ldots & x \\ 0 & 0 & x & \ldots & x \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & 0 & x \\ 0 & 0 & \ldots & 0 & 0 \end{pmatrix}, n_k \times n_k$$

$\mathbf{N}_k$ is upper triangular with zeros on the diagonal, and so *nilpotent*, $\mathbf{N}_k^{n_k} = \mathbf{0}$.
One can also find a similarity transformation **S** (but not unitary) such that

$$\mathbf{SUS}^{-1} = \begin{pmatrix} \mathbf{U}_1 & 0 & \ldots & 0 \\ 0 & \mathbf{U}_2 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & \mathbf{U}_q \end{pmatrix} = \mathbf{F}, \mathbf{F}^n = \begin{pmatrix} \mathbf{U}_1^n & 0 & \ldots & 0 \\ 0 & \mathbf{U}_2^n & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & \mathbf{U}_q^n \end{pmatrix}$$

so we can now focus attention on the powers of the diagonal blocks $\mathbf{U}_k$
The binomial expansion says, for an m x m matrix **U**,

$$\mathbf{U}^n = \lambda^n(\mathbf{I} + \mathbf{N})^n = \lambda^n \sum_{k=0}^{n} \binom{n}{k} \mathbf{N}^k = \lambda^n \sum_{k=0}^{m-1} \binom{n}{k} \mathbf{N}^k$$

because the unit matrix commutes with any matrix, and all powers > m-1 of **N** vanish.
The final step uses a norm estimate,

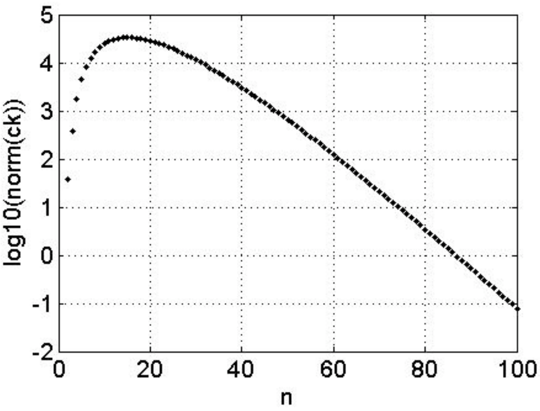
$$\left\|\mathbf{U}^n\right\| \le |\lambda|^n \sum_{k=0}^{m-1} \binom{n}{k} \left\|\mathbf{N}\right\|^k \le |\lambda|^n \max(m, \left\|\mathbf{N}\right\|^{m-1}) n^{m-1} = C|\lambda|^n n^{m-1}$$

which tends to 0 as *n* grows when $|\lambda| < 1$. This finishes the proof.

However, the matrix grows polynomially initially. Here is an example:

$$\mathbf{U} = \begin{pmatrix} a & m & m & m \\ 0 & a & m & m \\ 0 & 0 & a & m \\ 0 & 0 & 0 & a \end{pmatrix}, a = 0.8, m = 10$$

The plot shows $\|\mathbf{U}^n\mathbf{c}\|_2$ vs. *n*, $c = (1,1,1,1)^T$.



Least squares approximation: Normal equations, QR, and SVD.

Ex. Given data points (x_i,f_i), $i = 1,2,\ldots,m$, find a polynomial $p(x) = a_0 + a_1x + a_2x^2$ which approximates the data, $p(x_i) = f_i$. This is a linear system **Va** = **f**,

$$\begin{pmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ 1 & x_3 & x_3^2 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix}$$

for $m = 3$: When the x_i are distinct, there is a unique interpolation polynomial for any data (x_i,f_i). It may not be obvious that the columns of **V** are linearly independent, but we may compute the polynomial by another ansatz:

$$p(x) = c_0 + c_1(x - x_1) + c_2(x - x_1)(x - x_2):$$
$$f_1 = p(x_1) = c_0 + 0 + 0$$
$$f_2 = p(x_2) = c_0 + c_1(x_2 - x_1) + 0$$
$$f_3 = p(x_3) = c_0 + c_1(x_3 - x_1) + c_2(x_3 - x_1)(x_3 - x_2)$$

$$\begin{pmatrix} 1 & 0 & 0 \\ 1 & x_2 - x_1 & 0 \\ 1 & x_3 - x_1 & (x_3 - x_1)(x_3 - x_2) \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ f \end{pmatrix}$$

The coefficient matrix is lower triangular, and the system can be solved for **c** as long as the diagonal elements are non-zero. But the *c*-form and the *a*-form both generate all quadratic polynomials, so this shows that the system for the *a*-form is always non-singular. Indeed, the *Vandermonde* determinant may be calculated:

$$\det\begin{pmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ 1 & x_3 & x_3^2 \end{pmatrix} = (x_3 - x_1)(x_3 - x_2)(x_2 - x_1)$$

Next, we consider $m > 3$. We choose to find coefficients to minimize the sum of squares of discrepancies,

$$\min_a \sum_{j=1}^{m} (f_j - p(x_j))^2 = \min_a (\mathbf{r},\mathbf{r}), \mathbf{r} = (r_1, r_2, ..., r_m)^T, r_j = f_j - p(x_j)$$

The development exploits the scalar product (inner-product) (.,.),

Ex.

For the vector space $\mathbf{R}^n$ of real n-vectors, the standard inner product is

$$(\mathbf{x},\mathbf{y}) = \sum x_i y_i = \mathbf{x}^T\mathbf{y}$$

and we can define the Euclidean vector norm $\|\mathbf{x}\|_2^2 = (\mathbf{x},\mathbf{x})$ with the Cauchy-Schwarz inequality $(\mathbf{x},\mathbf{y}) \le \|\mathbf{x}\|_2 \|\mathbf{y}\|_2$. So we can define angles between vectors,

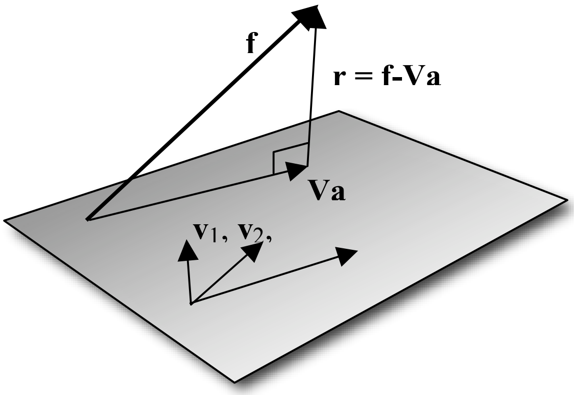
$$\frac{(\mathbf{x},\mathbf{y})}{\|\mathbf{x}\|_2 \|\mathbf{y}\|_2} = \cos\theta \text{ , etc.}$$

x and **y** are orthogonal if (**x**,**y**) = 0.

The optimum coefficients **a** - giving residuals **r*** - is characterized by:

Let $\mathbf{r} = \mathbf{f} - \sum a_j \mathbf{v}_j$ (written **r** = **f** – **Va** above)

Then $(\mathbf{r}^*,\mathbf{r}^*) \le (\mathbf{r},\mathbf{r})$ if and only if *(***r****,***v***)* = 0 for all **v** in the column space of **V, i.e.,** *the optimal residual vector is normal to all vectors in* **V** *– the normal equations***.**



Here is the picture:
The point **Va*** in the subspace spanned by the $\mathbf{v}_i$ has minimal distance to **f**. Perturbing the **a** to **a***+ $s\cdot$**c** changes **Va** by $s\cdot\mathbf{v} = s\cdot\mathbf{Vc}$ (s is a scalar multiple) and **r** by - $s\cdot\mathbf{v}$.

So for all s,

$$(\mathbf{r}^*,\mathbf{r}^*) \le (\mathbf{r}^* - s\mathbf{v}, \mathbf{r}^* - s\mathbf{v}) = (\mathbf{r}^*,\mathbf{r}^*) + s^2(\mathbf{v},\mathbf{v}) - 2s(\mathbf{r}^*,\mathbf{v}) =$$

$$= (\mathbf{r}^*,\mathbf{r}^*) + (\mathbf{v},\mathbf{v})(s - \frac{(\mathbf{r}^*,\mathbf{v})}{(\mathbf{v},\mathbf{v})})^2 - \frac{(\mathbf{r}^*,\mathbf{v})^2}{(\mathbf{v},\mathbf{v})}.$$

Choosing $s = (\mathbf{r}^*,\mathbf{v})^2/(\mathbf{v},\mathbf{v})$ we see that only (**r***,**v**) = 0 for all **v** in the subspace can satisfy the inequality.

We obtain the normal equations by taking **v** = $\mathbf{v}_1$, $\mathbf{v}_2$,…,$\mathbf{v}_m$, the columns of **V**:

$$\mathbf{V}^T\mathbf{V}\mathbf{a} = \mathbf{V}^T\mathbf{f}$$

The system can be solved by **LU** or $\mathbf{LDL}^T$ factorization. We will look at another idea: To obtain an orthogonal basis for **V** by e.g. the Gram-Schmidt orthogonalization. *Suppose for the moment that the* $\mathbf{v}_i$ *are orthogonal*, let us call them $\mathbf{q}_i$ Then

$$\mathbf{Q}^T\mathbf{Q}\mathbf{a} = \mathbf{Q}^T\mathbf{f}, \text{ and } \mathbf{Q}^T\mathbf{Q} = diag(\mathbf{q}_k{}^T\mathbf{q}_{k},), \text{ so}$$

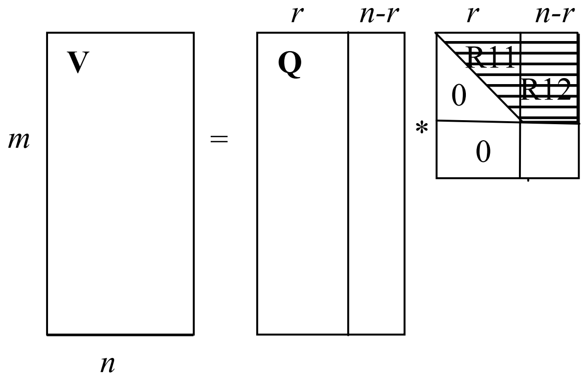
$$a_k = \mathbf{q}_k{}^T\mathbf{f} \,/\, \mathbf{q}_k{}^T\mathbf{q}_k,\ k = 1,2,\ldots, m$$

Here is a variant of the Gram-Schmidt algorithm:

```
$q_1 := v_1 / \|v_1\|$
for $k = 2,3,...$
    $q_k := v_k - \sum_{j=1}^{k-1} r_{jk} q_j ; r_{jk} = (v_k, q_j)$
    $q_k := q_k / \|q_k\|$
end
```

The values of r_{jk} makes $\mathbf{q}_k$ orthogonal to all earlier $\mathbf{q}j$, and they are normalized to unit length. If the $\mathbf{v}j$ are linearly dependent, $\mathbf{q}_k$ may become zero before one has used all the $\mathbf{v}j$. The algorithm is therefore combined with column reordering to choose the largest remaining **v** at every step.

Then, the process finishes with an upper triangular **R**-matrix whose last n-r rows are zeros, where r is the column rank of **V**. The corresponding columns of **Q** can be chosen arbitrarily, orthogonal to the r first.



From VP = QR follows $\mathbf{QRP}^T\mathbf{a} = \mathbf{f}$
so with $\mathbf{RP}^T\mathbf{a} = \mathbf{y}$ we have

$$y_k = 0,\ k = r{+}1, r{+}2,\ldots,n$$

and the solution to the normal equations

$$y_i = (\mathbf{q}_i,\mathbf{f}),\ i = 1,2,\ldots,r$$

which gives the minimal distance. If $r < n$, we get

$$\mathbf{a}_1 = \mathbf{R}11^{-1}(\mathbf{y}_1 - \mathbf{R}12\ \mathbf{a}_2)$$

where $\mathbf{y}_1 = (y_1,y_2,\ldots,y_r)^T$, etc.

A unique **a**-solution can be defined as the one with minimal number of non-zeros, i.e., $\mathbf{a}_2 = 0$. This is what `Matlab`'s backslash gives:

```
a = V\f;
```

Choosing the solution of minimal l2-norm defines the pseudo-inverse, $\mathbf{V}^+$,

```
a = pinv(V)*f;
```

This is computed by the singular value decomposition, developed into a practical tool by the Gene Golub (-2007) and Cleve Moler.

Any real *m x n* matrix **A** admits the factorization

$$\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T,$$

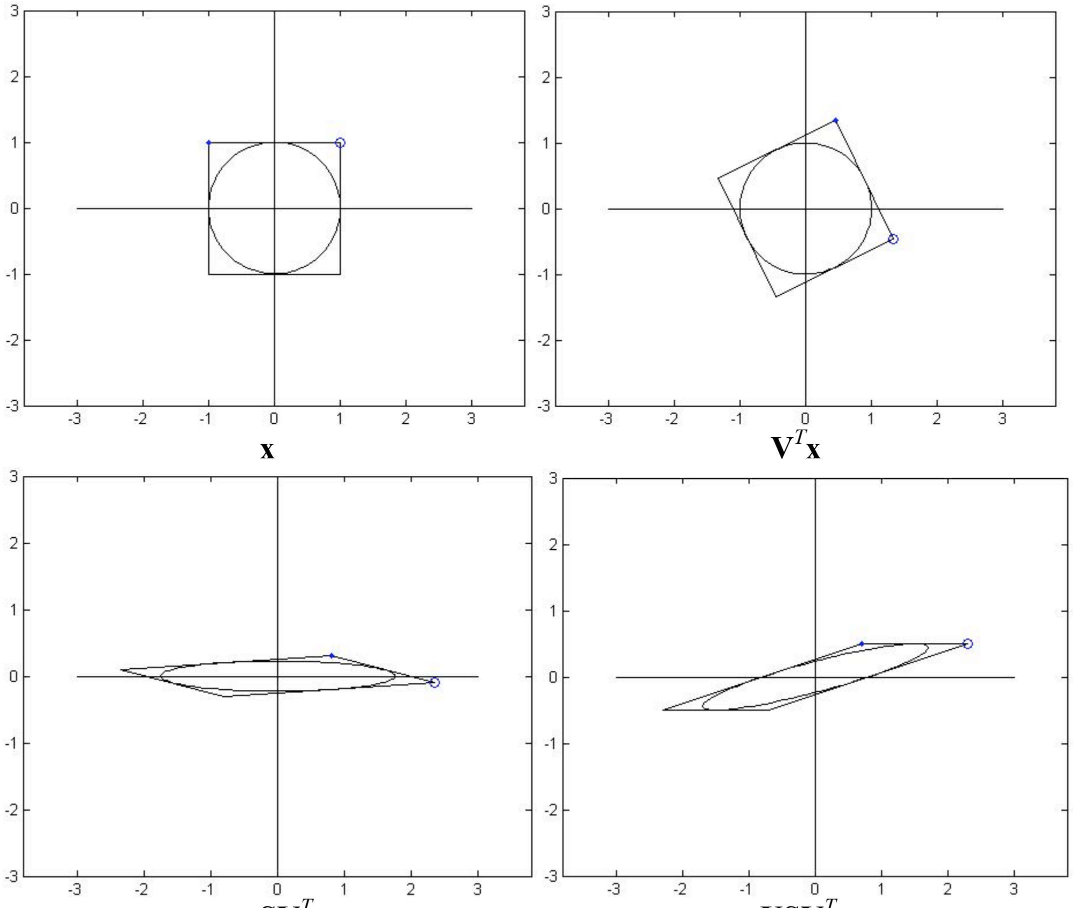
where **U** is *mxm*, **V** is *nxn*, both orthogonal. The first *r* columns of **U** is an orthogonal basis for the column space of **A**, and the *r* first columns of **V** are an orthogonal basis for the row space. **S** is *mxn*, non-zeros only on the diagonal $s_{ii} = \sigma_i$, sorted $\sigma_1 > \sigma_2 > \ldots > \sigma_r$, the singular values of **A**.
The *m-r* last columns of **U** can be chosen at will, if orthogonal to the *r* first columns, d:o for **V**.
The pseudo-inverse $\mathbf{S}^+$ is obtained by inverting the non-zeros of **S**, so

$$\mathbf{A}^+ = \mathbf{V}\mathbf{S}^+\mathbf{U}^T$$

The SVD can in principle be computed from eigenvalues and -vectors of $\mathbf{A}^T\mathbf{A}$ but the Golub-Reinsch algorithm uses a bi-diagonalization procedure which avoids the formation of the matrix product.

Ex. What is the SVD of an *mxn* rank-1 matrix $\mathbf{A} = \mathbf{u}\mathbf{v}^T$
The only non-zero singular value is ||**u**|| ||**v**||, the first column of **U** is **u**/||**u**||, d:o **V**. The rest of **U** (and **V**) is "arbitrary" and can be computed by orthogonalizing a set of linearly independent vectors (such as the set of unit vectors) against **u**, etc.

The Singular Value Decomposition describes a linear mapping as a rotation (possibly with a reflection)), followed by a stretching of the coordinate axes, and another rotation . Here is a mapping R^2 - > R^2



x $\quad$ $\mathbf{V}^T\mathbf{x}$

$\mathbf{S}\mathbf{V}^T\mathbf{x}$ $\quad$ $\mathbf{U}\mathbf{S}\mathbf{V}^T\mathbf{x}$

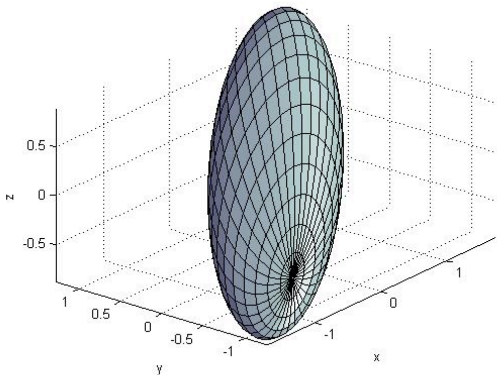
$$\mathbf{A} = \begin{pmatrix} 0.8 & 1.5 \\ 0 & 0.5 \end{pmatrix} = \mathbf{USV}^T, \mathbf{S} = \begin{pmatrix} 1.7573 & 0 \\ 0 & 0.2276 \end{pmatrix},$$

$$\mathbf{V} = \begin{pmatrix} 0.4401 & -0.8979 \\ 0.8979 & 0.4401 \end{pmatrix}, \phi = 64^o, \mathbf{U} = \begin{pmatrix} 0.9668 & -0.2555 \\ 0.2555 & 0.9668 \end{pmatrix}, \phi = 15^o$$

The singular values are the half-axes of the ellipsoidal image of the unit ball:

$$\mathbf{A} = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0.8 & 1 \\ 0.2 & 0.7 & 0.5 \end{pmatrix},$$

$$\sigma = 2.2311, 0.6397, 0.1822$$



Mechanical models: Balls on springs.

We consider *Hookean* springs, for which the restoring force is proportional to the extension/compression of the spring:

$$F = -K(l - l_0)$$

where l is the extended length and l_0 is called the natural (force-free) length.

A torsion spring produces a torque proportional to the rotation angle,

$$M = -K\,\phi$$

The work done *against the spring force* when the spring is extended from length l_0 to $l_0 + e$ is

$$W = \int_0^e Kl \cdot dl = 1/2Kl^2$$

so, there is energy stored in the process, $W = 1/2Ke^2$. It follows, that

$$F = -\frac{dW}{dl}$$

Ex. Homogeneous gravity field directed in the negative z-direction, on a mass point m

$$\mathbf{F} = -gm\mathbf{e}_z$$

The work an external force has to do to move from A to B is

$$W = \int_A^B gm\mathbf{e}_z \cdot d\mathbf{r} = gm\int_A^B dz = mg(z(\mathrm{B}) - z(\mathrm{A}))$$

The work is independent of the path between A and B. It follows, that the force is the negative gradient of the *potential energy* function,

$$\mathbf{F} = -\frac{\partial W}{\partial z}\mathbf{e}_z$$