Linear Algebra Methods for Data Mining

Saara Hyvönen, Saara.Hyvonen@cs.helsinki.fi

Spring 2007

2. Basic Linear Algebra continued

Happened so far:

- matrix-vector multiplication, matrix-matrix multiplication
- vector norms, matrix norms
- distances between vectors
- eigenvalues, eigenvectors
- linear independence
- basis
- orthogonality

Eigenvalues and eigenvectors

- Let $\mathbf{A}$ be a $n \times n$ matrix. The vector $\mathbf{v} \neq \mathbf{0}$ that satisfies

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$$

for some scalar λ is called the **eigenvector** of $\mathbf{A}$ and λ is the **eigenvalue** corresponding to the eigenvector $\mathbf{v}$.

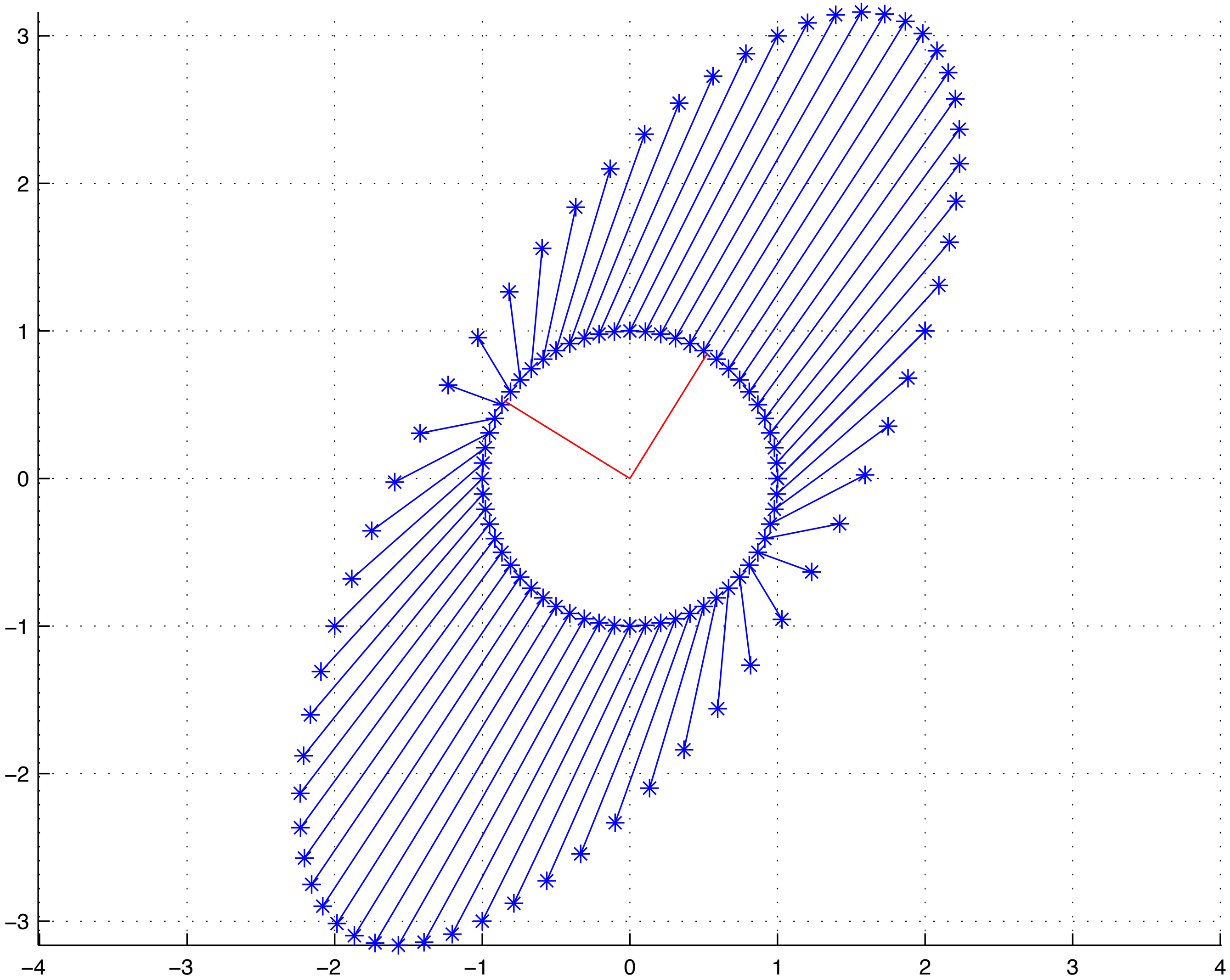
Example

$$\mathbf{A}\mathbf{v} = \begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix} \mathbf{v} = \lambda \mathbf{v}.$$

$$\det(\mathbf{A} - \lambda \mathbf{I}) = \begin{vmatrix} 2-\lambda & 1 \\ 1 & 3-\lambda \end{vmatrix} = (2-\lambda)(3-\lambda) - 1 = 0 \quad \Rightarrow$$

$$\lambda_1 = 3.62 \qquad \lambda_2 = 1.38$$

$$\mathbf{v}_1 = \begin{pmatrix} 0.52 \\ 0.85 \end{pmatrix} \qquad \mathbf{v}_2 = \begin{pmatrix} 0.85 \\ -0.52 \end{pmatrix}$$



3
2
1
0
−1
−2
−3
−4
−3
−2
−1
0
1
2
3
4

Orthogonality

- Two vectors $\mathbf{x}$ and $\mathbf{y}$ are **orthogonal**, if $\mathbf{x}^T\mathbf{y} = 0$.

- Let $\mathbf{q}_j$, $j = 1, \dots, n$ be orthogonal, i.e. $\mathbf{q}_i^T\mathbf{q}_j = 0$, $i \neq j$. Then they are linearly independent.

- Let the set of orthogonal vectors $\mathbf{q}_j$, $j = 1, \dots, m$ in $\mathbb{R}^m$ be normalized, $\|\mathbf{q}\| = 1$. Then they are **orthonormal**, and constitute an **orthonormal basis** in $\mathbb{R}^m$.

- A matrix $\mathbb{R}^{m \times m} \ni \mathbf{Q} = [\mathbf{q}_1 \, \mathbf{q}_2 \, \dots \, \mathbf{q}_m]$ with orthonormal columns is called an **orthogonal matrix**.

Example

In the previous example we determined the eigenvectors of the matrix

$$\begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix} \quad \text{to be} \quad \mathbf{v}_1 = \begin{pmatrix} 0.52 \\ 0.85 \end{pmatrix}, \quad \mathbf{v}_2 = \begin{pmatrix} 0.85 \\ -0.52 \end{pmatrix}.$$

The vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ are orthogonal:

$$\mathbf{v}_1^T \mathbf{v}_2 = 0.52 \cdot 0.85 + 0.85 \cdot (-0.52) = 0.$$

This is no coincidence!

Eigenvalues and eigenvectors of a symmetric matrix

The eigenvectors of a symmetric matrix are mutually orthogonal and its eigenvalues are real.

$$(\mathbf{A} \in \mathbb{R}^{n \times n} \text{ is a symmetric matrix, if } \mathbf{A} = \mathbf{A}^T.)$$

Eigendecomposition

A symmetric matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ can be written in the form

$$\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T,$$

where the columns of $\mathbf{U}$ are the eigenvectors of $\mathbf{A}$ and $\mathbf{\Lambda}$ is a diagonal matrix, the diagonal elements of which are the corresponding eigenvalues of $\mathbf{A}$. Note, that $\mathbf{U}$ is orthogonal. This is called the eigendecomposition or symmetric Schur decomposition of $\mathbf{A}$.

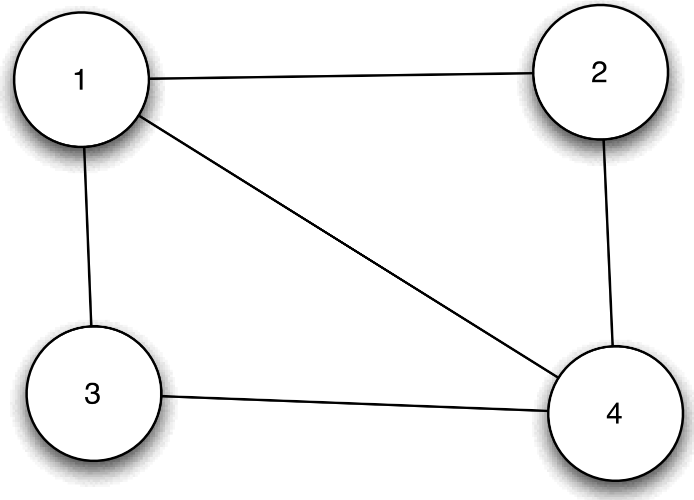
Check

```
U=[.52 0.85; 0.85 -0.52]'

U =
    0.5200    0.8500
    0.8500   -0.5200

Lambda=[3.62 0;0 1.38]

Lambda =
    3.6200         0
         0    1.3800
```

```
U*Lambda*U'

ans =

    1.9759    0.9901
    0.9901    2.9886
```

Example of symmetric matrices: graphs

The adjacency matrix of an undirected graph is a symmetric matrix:



$$\mathbf{A} = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

Example: virus propagation

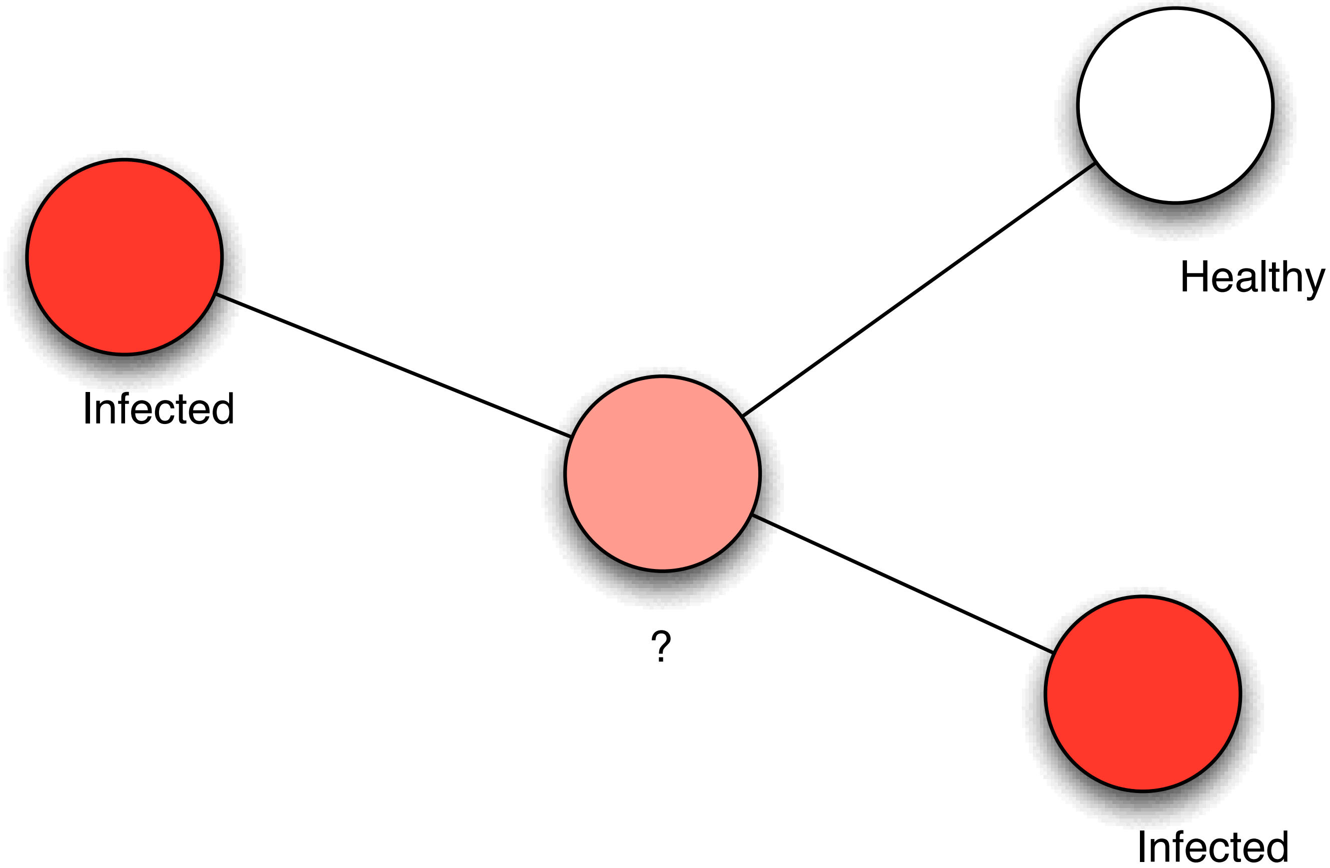
- Question 1: How does a virus spread across an arbitrary network?

- Question 2: Will it create an epidemic?

- Question 3: What can we do about it?

Model:

- the Susceptible-Infected-Susceptible (SIS) model
- cured nodes immediately become susceptible
- virus birth rate β: probability that an infected neighbor attacks
- virus death rate δ: probability that an infected node heals
- virus "strength" $s = \beta/\delta$.


Healthy
Infected
?
Infected

Epidemic threshold τ

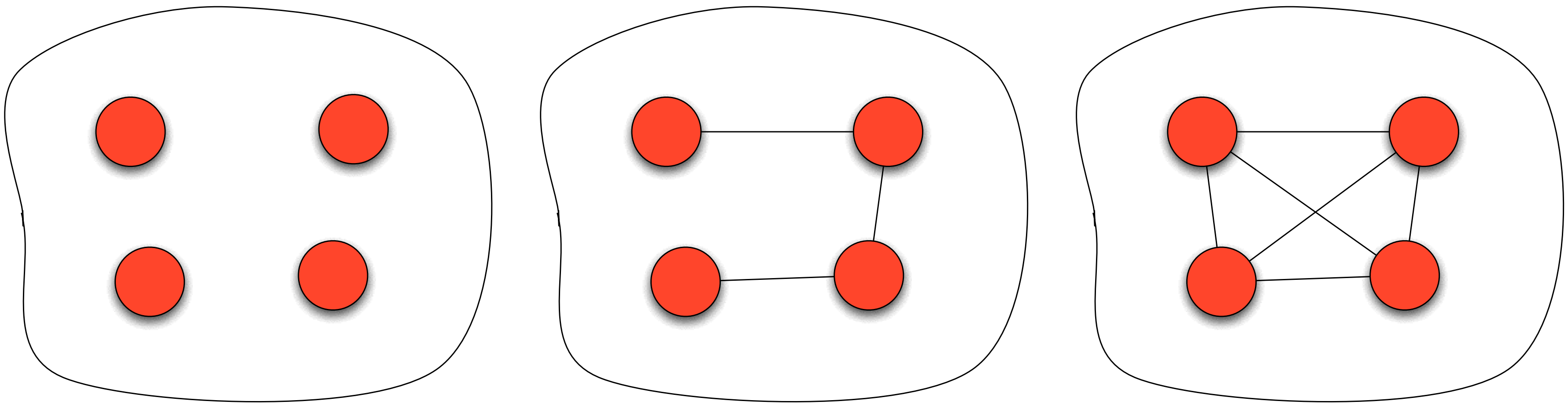
- The epidemic threshold of a graph is the largest value of τ for which it holds that if the virus strength

$$s = \beta/\delta < \tau,$$

  an epidemic can not happen.

- Problem: given a graph G, compute the epidemic threshold τ.

- The epidemic threshold depends only on the graph! But which properties of the graph?



Answer (to question 2)

Theorem. (Wang et al, 2003) Let us have a graph with the adjacency matrix $\mathbf{A}$. We have no epidemic, if

$$\beta/\delta < \tau = 1/\lambda_{\mathbf{A}},$$

where

- $\lambda_{\mathbf{A}}$ the largest eigenvalue of $\mathbf{A}$
- β is the prob. that an infected neighbor attacks,
- δ is the prob. that an infected node heals.

What can we do with this information?

Q: Who is the best person/computer to immunize against the virus?

A: The one the removal of which will make the largest difference in $\lambda_{\mathbf{A}}$.

Note: Eigenvalues are strongly related to graph topology!

Other questions that can be answered in a similar way:

- who is the best customer to advertise to?
- who originated a raging rumor?
- in general: how important is a node in a network?

What if the matrix is not symmetric?

- If $\mathbf{A}$ is symmetric, we can write $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$, where $\mathbf{U}$ and $\mathbf{\Lambda}$ contain the eigenvectors and corresponding eigenvalues of $\mathbf{A}$.

- What if $\mathbf{A} \in \mathbb{R}^{m \times n}$? Definitely not symmetric!

- Then we can use the singular value decomposition (SVD):

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

  where $\mathbf{U}$ and $\mathbf{V}$ are orthogonal, and $\mathbf{\Sigma}$ is diagonal.

Example:

customer \ day	Wed	Thu	Fri	Sat	Sun
ABC Inc.	1	1	1	0	0
CDE Co.	2	2	2	0	0
FGH Ltd.	1	1	1	0	0
NOP Inc.	5	5	5	0	0
Smith	0	0	0	2	2
Brown	0	0	0	3	3
Johnson	0	0	0	1	1

$$
\begin{pmatrix}
1 & 1 & 1 & 0 & 0 \\
2 & 2 & 2 & 0 & 0 \\
1 & 1 & 1 & 0 & 0 \\
5 & 5 & 5 & 0 & 0 \\
0 & 0 & 0 & 2 & 2 \\
0 & 0 & 0 & 3 & 3 \\
0 & 0 & 0 & 1 & 1
\end{pmatrix}
=
\begin{pmatrix}
0.18 & 0 \\
0.36 & 0 \\
0.18 & 0 \\
0.90 & 0 \\
0 & 0.53 \\
0 & 0.80 \\
0 & 0.27
\end{pmatrix}
\times
\begin{pmatrix}
9.64 & 0 \\
0 & 5.29
\end{pmatrix}
\times
$$

$$
\begin{pmatrix}
0.58 & 0.58 & 0.58 & 0 & 0 \\
0 & 0 & 0 & 0.71 & 0.71
\end{pmatrix}
$$

How to compute matrix decompositions?

- We have had a brief glimpse at eigenvalue decomposition and singular value decomposition.
- Before taking a closer look: how can we compute these?
- Answer 1: use LAPACK, Matlab, Mathematica, ... they are implemented everywhere!
- Answer 2: we need more linear algebra tools!
- Back to basics...

Givens (plane) rotations

$$\mathbf{G} = \begin{pmatrix} c & s \\ -s & c \end{pmatrix}, \quad c^2 + s^2 = 1.$$

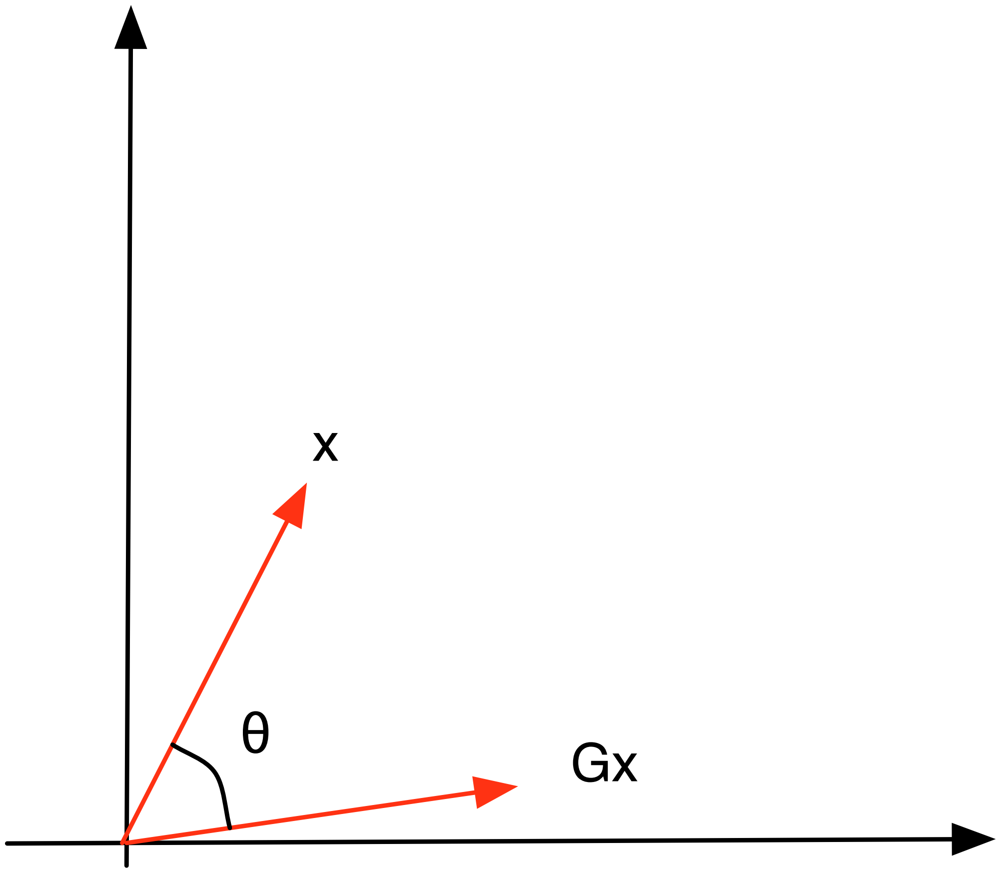
```
theta=pi/8;c=cos(theta);s=sin(theta);
G=[c s;-s c]

    0.9239    0.3827
   -0.3827    0.9239

G'*G

    1.0000    0.0000
    0.0000    1.0000
```

Givens (plane) rotations

$$\mathbf{G} = \begin{pmatrix} c & s \\ -s & c \end{pmatrix}, \quad c^2 + s^2 = 1.$$

- We can choose c and s so that $c = \cos(\theta)$, $s = \sin(\theta)$ for some θ.

- Then multiplication of a vector $\mathbf{x}$ by $\mathbf{G}$ means that we rotate the vector in th $\mathbb{R}^2$ by an angle θ:


x
θ
Gx

Embed a 2-D rotation in a larger unit matrix:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & c & 0 & s \\ 0 & 0 & 1 & 0 \\ 0 & -s & 0 & c \end{pmatrix}$$

```
G =

    1.0000         0         0         0
         0    0.9239         0    0.3827
         0         0    1.0000         0
         0   -0.3827         0    0.9239

G'*G

ans =

    1.0000         0         0         0
         0    1.0000         0    0.0000
         0         0    1.0000         0
         0    0.0000         0    1.0000
```

- Givens rotations can be used to zero elements in vectors and matrices.

- Rotation matrix $\mathbf{G} = \begin{pmatrix} c & s \\ -s & c \end{pmatrix}$, $c^2 + s^2 = 1$.

- Choose c (and s) so that $\mathbf{G}\mathbf{v} = \begin{pmatrix} \alpha \\ 0 \end{pmatrix}$:

- Choose c (and s) so that $\mathbf{Gv} = \begin{pmatrix} \alpha \\ 0 \end{pmatrix}$:

$$c = \frac{v_1}{\sqrt{v_1^2 + v_2^2}}, \quad s = \frac{v_2}{\sqrt{v_1^2 + v_2^2}}$$

- (Check that this does what it is supposed to do!)

We can choose c and s in

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & c & 0 & s \\ 0 & 0 & 1 & 0 \\ 0 & -s & 0 & c \end{pmatrix}$$

so that we can zero element 4 in a vector by **a rotation in plane (2,4)**.

```
x=[1;2;3;4];
sq=sqrt(x(2)^2+x(4)^2);
c=x(2)/sq;s=x(4)/sq;
G=[1 0 0 0;0 c 0 s;0 0 1 0;0 -s 0 c];

y=G*x

y =

    1.0000
    4.4721
    3.0000
         0
```

Reminder

- The inverse of an orthogonal matrix $\mathbf{Q}$ is $\mathbf{Q}^{-1} = \mathbf{Q}^T$.

- The Euclidean length of a vector is invariant under an orthogonal transformation $\mathbf{Q}$:

$$\|\mathbf{Q}\mathbf{x}\|^2 = (\mathbf{Q}\mathbf{x})^T\mathbf{Q}\mathbf{x} = \mathbf{x}^T\mathbf{x} = \|\mathbf{x}\|^2.$$

- The product of two orthogonal matrices $\mathbf{Q}$ and $\mathbf{P}$ is orthogonal:

$$\mathbf{X}^T\mathbf{X} = (\mathbf{P}\mathbf{Q})^T\mathbf{P}\mathbf{Q} = \mathbf{Q}^T\mathbf{P}^T\mathbf{P}\mathbf{Q} = \mathbf{Q}^T\mathbf{Q} = \mathbf{I}.$$

Transforming a vector $\mathbf{v}$ to $\alpha \mathbf{e}_1$

Summarize

- $\alpha \mathbf{e}_1 = \mathbf{G}_3(\mathbf{G}_2(\mathbf{G}_1 \mathbf{v})) = (\mathbf{G}_3 \mathbf{G}_2 \mathbf{G}_1)\mathbf{v}$.

- Denote
$$\mathbf{P} = \mathbf{G}_3 \mathbf{G}_2 \mathbf{G}_1.$$
$\mathbf{P}$ is orthogonal, and $\mathbf{P}\mathbf{v} = \alpha \mathbf{e}_1$.

- Since $\mathbf{P}$ is orthogonal, euclidean length is preserved, and
$$\alpha = \|\mathbf{v}\| = \sqrt{\sum_{i=1}^{n} v_i^2}.$$

Number of floating point operations

- What is the number of flops needed to transform the first column of a $m \times n$ matrix to $\alpha \mathbf{e}_1$, a multiple of the first unit vector?

- the computation of

$$\begin{pmatrix} c & s \\ -s & c \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} cx + sy \\ -sx + cy \end{pmatrix}$$

  requires 4 multiplications and 2 additions, i.e. 6 flops.

- Applying such a transformation to a $m \times n$ matrix requires $6n$ flops.

- In order to zero all elements but one in the first column of the matrix we must apply $(m-1)$ rotations.

- overall flop count: $6(m-1)n \approx 6mn$.
- the so called Householder transformation, with which one can do the same thing, is slightly cheaper: $4mn$.

What was that all about?

- We can use very simple, orthogonal transformations (e.g. Givens rotations) to zero elements in a matrix.

- In principle, this is what is done when matrix decompositions are calculated.

Matrix decompositions

- We wish to **decompose** the matrix $\mathbf{A}$ by writing it as a product of two or more matrices:

$$\mathbf{A}_{m\times n} = \mathbf{B}_{m\times k}\mathbf{C}_{k\times n}, \quad \mathbf{A}_{m\times n} = \mathbf{B}_{m\times k}\mathbf{C}_{k\times r}\mathbf{D}_{r\times n}$$

- This is done in such a way that the right side of the equation yields some useful information or insight to the nature of the data matrix $\mathbf{A}$.

- Or is in other ways useful for solving the problem at hand.

Matrix decompositions

- There are numerous examples of useful matrix decompositions:
- Eigendecomposition: $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$, $\mathbf{A}$ symmetric, $\mathbf{U}$ and $\mathbf{\Lambda}$ eigenvectors and eigenvalues.
- Singular value decomposition: $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, $\mathbf{U}$, $\mathbf{V}$ orthogonal, $\mathbf{\Sigma}$ diagonal.
- Matrix **factorization** is the same thing as matrix decomposition (e.g. NMF = nonnegative matix factorization, $\mathbf{V} = \mathbf{W}\mathbf{H}$, all elements nonnegative.)

How to compute these?

- Roughly (attn: in reality there is more to it than this!),

(1) Manipulate $\mathbf{A}$ by multiplying it by intelligently chosen, fairly simple (orthogonal) matrices from both sides:

$$\mathbf{V}_k \ldots \mathbf{V}_2 \mathbf{V}_1 \mathbf{A} \mathbf{W}_1 \ldots \mathbf{W}_s = \mathbf{B}, \quad \text{until } \mathbf{B} \text{ is "nice"}.$$

(2) Denote $\mathbf{V} = \mathbf{V}_k \ldots \mathbf{V}_2 \mathbf{V}_1$, $\mathbf{W} = \mathbf{W}_1 \ldots \mathbf{W}_s$. Now $\mathbf{A} = \mathbf{V}\mathbf{B}\mathbf{W}^T$.

- But how to choose $\mathbf{V}_1, ..., \mathbf{V}_k, \mathbf{W}_1, ..., \mathbf{W}_s$?

- Needed: linear algebra tools for transforming matrices in an orderly fashion: Givens!

- By applying several Givens rotations in succession, we can transform

$$\mathbf{x} \longrightarrow \begin{pmatrix} \alpha \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

QR transformation

- We transform $\mathbf{A} \rightarrow \mathbf{Q}^T\mathbf{A} = \mathbf{R}$, where $\mathbf{Q}$ is orthogonal and $\mathbf{R}$ is upper triangular.
- Example:

Note!

- Any matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, $m \geq n$, can be transformed to upper triangular form by an orthogonal matrix.

- If the columns of $\mathbf{A}$ are linearly independent, then $\mathbf{R}$ is non-singular.

QR decomposition

$$\mathbf{A} = \mathbf{Q} \begin{pmatrix} \mathbf{R} \\ \mathbf{0} \end{pmatrix}$$

References

[1] Lars Eldén: Matrix Methods in Data Mining and Pattern Recognition, SIAM 2007.

[2] G. H. Golub and C. F. Van Loan. Matrix Computations. 3rd ed. Johns Hopkins Press, Baltimore, MD., 1996.

[3] Y. Wang, D. Chakrabarti, C. Wang and C. Faloutsos, Epidemic Spreading in Real Networks: an Eigenvalue Viewpoint, SRDS 2003, Florence, Italy.