Projections & Orthogonalisation

Mainly two resources have been used to setup this notebook:

Sources:

1. `Linear Algebra : Theory, Intuition, Code` author: Mike X Cohen, publisher: sincXpress
2. `No bullshit guide to linear algebra` author: Ivan Savov
3. `Matrix Methods for Computational Modeling and Data Analytics` author: Mark Embree, Virginia Tech

Projection (Part 1) / Projection on a vector

A vector $\mathbf{b}$ shall be projected onto another vector $\mathbf{a}$.

$\mathbf{p} = \beta \cdot \mathbf{a} = proj_{(a)}(b)$ denotes the projection vector. Then vector $\mathbf{b}$ can be decomposed into the sum of two vectors $\mathbf{p}$ and $\mathbf{r}$.

$$\mathbf{b} = \mathbf{p} + \mathbf{r}$$

In computing the projection vector $\mathbf{p}$ the scalar β shall be chosen such as to <u>minimise</u> the norm of the residual vector $\mathbf{r}$

For the norm $||\mathbf{r}||$ we get:

$$||\mathbf{r}|| = ||\mathbf{b} - \beta \cdot \mathbf{a}||$$

Minimising $||\mathbf{r}||$ by proper choice of β is equivalent to minimising the quadratic norm $||\mathbf{r}||^2$:

$$||\mathbf{r}||^2 = \mathbf{r}^T \cdot \mathbf{r} = \left(\mathbf{b}^T - \beta \cdot \mathbf{a}^T\right) \cdot (\mathbf{b} - \beta \cdot \mathbf{a})$$
$$||\mathbf{r}||^2 = \mathbf{b}^T \cdot \mathbf{b} - 2 \cdot \beta \cdot \mathbf{b}^T \cdot \mathbf{a} + \beta^2 \cdot \mathbf{a}^T \cdot \mathbf{a}$$

Differentiating $||\mathbf{r}||^2$ with respect to β yields:

$$\frac{d||\mathbf{r}||^2}{d\beta} = -2\mathbf{b}^T \cdot \mathbf{a} + 2 \cdot \beta \cdot \mathbf{a}^T \cdot \mathbf{a}$$

The optimum β which minimises $||\mathbf{r}||$ is therefor:

$$\beta = \frac{\mathbf{b}^T \cdot \mathbf{a}}{\mathbf{a}^T \cdot \mathbf{a}}$$

Thus we can express vector $\mathbf{b}$ as:

$$\mathbf{b} = \mathbf{p} + \mathbf{r} = \beta \cdot \mathbf{a} + \mathbf{r} = \underbrace{\frac{\mathbf{b}^T \cdot \mathbf{a}}{\mathbf{a}^T \cdot \mathbf{a}} \cdot \mathbf{a}}_{\mathbf{p}} + \mathbf{r}$$

The projection vector $\mathbf{p}$ is in the direction of vector $\mathbf{a}$.

$$\mathbf{p} = \frac{\mathbf{b}^T \cdot \mathbf{a}}{\mathbf{a}^T \cdot \mathbf{a}} \cdot \mathbf{a} = \frac{\mathbf{b}^T \cdot \mathbf{a}}{||\mathbf{a}||^2} \cdot \mathbf{a} = \mathbf{b}^T \cdot \frac{\mathbf{a}}{||\mathbf{a}||} \cdot \frac{\mathbf{a}}{||\mathbf{a}||}$$

In this equation the vector $\frac{\mathbf{a}}{||\mathbf{a}||}$ denotes the unit vector in the direction of $\mathbf{a}$ for which we introduce the notation:

$$\mathbf{a}_u = \frac{\mathbf{a}}{||\mathbf{a}||}$$

$$\mathbf{p} = \left(\mathbf{b}^T \cdot \mathbf{a}_u\right) \cdot \mathbf{a}_u$$

For the residual vector $\mathbf{r}$ we get:

$$\mathbf{r} = \mathbf{b} - \mathbf{p} = \mathbf{b} - \left(\mathbf{b}^T \cdot \mathbf{a}_u\right) \cdot \mathbf{a}_u$$

orthogonality of $\mathbf{r}$ and $\mathbf{p}$

It shall be shown that $\mathbf{r}$ is orthogonal to <u>any</u> vector $\alpha \cdot \mathbf{a}_u$. We must show that $\alpha \cdot \mathbf{r}^T \cdot \mathbf{a}_u = 0$:

$$\alpha \cdot \mathbf{r}^T \cdot \mathbf{a}_u = \alpha \cdot \mathbf{b}^T \cdot \mathbf{a}_u - \alpha \cdot \left(\mathbf{b}^T \cdot \mathbf{a}_u\right) \cdot \mathbf{a}_u^T \cdot \mathbf{a}_u$$

$$= \alpha \cdot \mathbf{b}^T \cdot \mathbf{a}_u - \alpha \cdot \left(\mathbf{b}^T \cdot \mathbf{a}_u\right) \cdot ||\mathbf{a}_u||$$

$$\alpha \cdot \mathbf{r}^T \cdot \mathbf{a}_u = \alpha \cdot \mathbf{b}^T \cdot \mathbf{a}_u - \alpha \cdot \mathbf{b}^T \cdot \mathbf{a}_u = 0$$

projectors

The projection $\mathbf{p}$ of vector $\mathbf{b}$ onto vector $\mathbf{a}$

$$\mathbf{p} = \frac{\mathbf{b}^T \cdot \mathbf{a}}{\mathbf{a}^T \cdot \mathbf{a}} \cdot \mathbf{a}$$

can be re-arranged. Since the expression $\frac{\mathbf{b}^T \cdot \mathbf{a}}{\mathbf{a}^T \cdot \mathbf{a}}$ is a scalar we may write:

$$\mathbf{p} = \mathbf{a} \cdot \frac{\mathbf{b}^T \cdot \mathbf{a}}{\mathbf{a}^T \cdot \mathbf{a}}$$

$$= \mathbf{a} \cdot \frac{\mathbf{a}^T \cdot \mathbf{b}}{\mathbf{a}^T \cdot \mathbf{a}}$$

$$= \left(\frac{\mathbf{a} \cdot \mathbf{a}^T}{\mathbf{a}^T \cdot \mathbf{a}} \right) \cdot \mathbf{b}$$

The expression

$\left(\frac{\mathbf{a} \cdot \mathbf{a}^T}{\mathbf{a}^T \cdot \mathbf{a}} \right)$ denotes a square symmetric matrix which only depends on the elements of vector $\mathbf{a}$. Multiplying this matrix from then right by a vector $\mathbf{b}$ yields the *best/orthogonal* projection onto vector $\mathbf{a}$.

The matrix is named the projector onto vector $\mathbf{a}$ and a specific symbol $\mathbf{P}_\mathbf{a}$ is introduced:

$$\mathbf{P}_\mathbf{a} = \frac{\mathbf{a} \cdot \mathbf{a}^T}{\mathbf{a}^T \cdot \mathbf{a}}$$

Some properties of projectors are summarised here:

$\mathbf{P}_\mathbf{a}$ is symmetric. This propery follows from the fact that the matrix is obtained from the outer product of two identical vectors.

$\mathbf{P}_\mathbf{a} \cdot \mathbf{P}_\mathbf{a} = \mathbf{P}_\mathbf{a}$. To see this

$$\mathbf{P}_\mathbf{a} \cdot \mathbf{P}_\mathbf{a} = \frac{\mathbf{a} \cdot \mathbf{a}^T}{\mathbf{a}^T \cdot \mathbf{a}} \cdot \frac{\mathbf{a} \cdot \mathbf{a}^T}{\mathbf{a}^T \cdot \mathbf{a}} = \frac{\mathbf{a} \cdot \left(\mathbf{a}^T \cdot \mathbf{a} \right) \cdot \mathbf{a}^T}{\mathbf{a}^T \cdot \mathbf{a} \cdot \mathbf{a}^T \cdot \mathbf{a}} = \frac{\mathbf{a} \cdot \mathbf{a}^T}{\mathbf{a}^T \cdot \mathbf{a}} = \mathbf{P}_\mathbf{a}$$

Another useful identity is:

$$\left(\mathbf{I} - \mathbf{P}_\mathbf{a} \right) \cdot \mathbf{P}_\mathbf{a} = \mathbf{0}$$

The derivation of this identity uses the property $\mathbf{P}_\mathbf{a} \cdot \mathbf{P}_\mathbf{a} = \mathbf{P}_\mathbf{a}$:

$$\left(\mathbf{I} - \mathbf{P}_\mathbf{a} \right) \cdot \mathbf{P}_\mathbf{a} = \mathbf{P}_\mathbf{a} - \mathbf{P}_\mathbf{a} \cdot \mathbf{P}_\mathbf{a} = \mathbf{P}_\mathbf{a} - \mathbf{P}_\mathbf{a} = \mathbf{0}$$

If vector $\mathbf{b}$ is already orthogonal to vector then

$$\mathbf{P}_\mathbf{a} \cdot \mathbf{b} = \mathbf{0}$$

The expression

$$\left(\mathbf{I} - \mathbf{P}_{\mathbf{a}}\right) \cdot \mathbf{b} = \mathbf{r}$$
$$\rightarrow$$
$$\mathbf{r}^T \cdot \mathbf{a} = 0$$

is just the residual vector which is orthogonal to $\mathbf{a}$.

Summary

The projection $\mathbf{p}$ of vector $\mathbf{b}$ onto some vector $\mathbf{a}$ is computed from this equation:

$$\mathbf{p} = \frac{\mathbf{a}^T \cdot \mathbf{b}}{\mathbf{a}^T \cdot \mathbf{a}} \cdot \mathbf{a}$$

The projector onto $\mathbf{a}$ is defined as:

$$\mathbf{P}_{\mathbf{a}} = \frac{\mathbf{a} \cdot \mathbf{a}^T}{\mathbf{a}^T \cdot \mathbf{a}}$$

$$\mathbf{p} = \mathbf{P}_{\mathbf{a}} \cdot \mathbf{a}$$

The residual vector $\mathbf{r}$ is orthogonal to vectors $\mathbf{a},\ \mathbf{p}$.

$$\mathbf{b} = \mathbf{p} + \mathbf{r}$$
$$\rightarrow$$
$$\mathbf{r} = \mathbf{b} - \mathbf{p} = \mathbf{b} - \frac{\mathbf{a}^T \cdot \mathbf{b}}{\mathbf{a}^T \cdot \mathbf{a}} \cdot \mathbf{a}$$

Cauchy-Schwarz Inequality

The `Cauchy-Schwarz` inequality states:

$$|\mathbf{v}^T \cdot \mathbf{w}| \leq ||\mathbf{v}|| \cdot ||\mathbf{w}||$$

Proof

We look at the quadratic norm

$$||\mathbf{w} + \alpha \cdot \mathbf{v}||^2 = \mathbf{w}^T \cdot \mathbf{w} + 2 \cdot \alpha \cdot \mathbf{w}^T \cdot \mathbf{v} + \alpha^2 \cdot \mathbf{v}^T \cdot \mathbf{v}$$
$$0 \leq ||\mathbf{w}||^2 + 2 \cdot \alpha \cdot \mathbf{w}^T \cdot \mathbf{v} + \alpha^2 \cdot ||\mathbf{v}||^2$$

The right side of the inequality describes a parabola $f(\alpha)$ which must be strictly non-negative. Thus solutions α to $f(\alpha) = 0$ must be complex.

$$f(\alpha) = ||\mathbf{w}||^2 + 2\cdot\alpha\cdot\mathbf{w}^T\cdot\mathbf{v} + \alpha^2\cdot||\mathbf{v}||^2 f(\alpha) = ||\mathbf{v}||^2\cdot\left(\alpha^2 + 2\cdot\alpha\cdot\frac{\mathbf{w}^T\cdot\mathbf{v}}{||\mathbf{v}||^2} + \frac{||\mathbf{w}||^2}{||\mathbf{v}||^2}\right)$$

Ignoring (for a moment) the case $\mathbf{v} = \mathbf{0}$ we are looking for those values α for which $f(\alpha) \geq 0$:

$$0 \leq \alpha^2 + 2\cdot\alpha\cdot\frac{\mathbf{w}^T\cdot\mathbf{v}}{||\mathbf{v}||^2} + \frac{||\mathbf{w}||^2}{||\mathbf{v}||^2}$$

$$\left(\alpha + \frac{\mathbf{w}^T\cdot\mathbf{v}}{||\mathbf{v}||^2}\right)^2 + \frac{||\mathbf{w}||^2}{||\mathbf{v}||^2} - \left(\frac{\mathbf{w}^T\cdot\mathbf{v}}{||\mathbf{v}||^2}\right)^2 \geq 0$$

$$\left(\alpha + \frac{\mathbf{w}^T\cdot\mathbf{v}}{||\mathbf{v}||^2}\right)^2 \geq \left(\frac{\mathbf{w}^T\cdot\mathbf{v}}{||\mathbf{v}||^2}\right)^2 - \frac{||\mathbf{w}||^2}{||\mathbf{v}||^2}$$

$$\alpha + \frac{\mathbf{w}^T\cdot\mathbf{v}}{||\mathbf{v}||^2} \geq \sqrt{\left(\frac{\mathbf{w}^T\cdot\mathbf{v}}{||\mathbf{v}||^2}\right)^2 - \frac{||\mathbf{w}||^2}{||\mathbf{v}||^2}}$$

For complex zeros α we need:

$$\left(\frac{\mathbf{w}^T\cdot\mathbf{v}}{||\mathbf{v}||^2}\right)^2 - \frac{||\mathbf{w}||^2}{||\mathbf{v}||^2} \leq 0$$

$$\left(\frac{\mathbf{w}^T\cdot\mathbf{v}}{||\mathbf{v}||^2}\right)^2 \leq \frac{||\mathbf{w}||^2}{||\mathbf{v}||^2}$$

$$\frac{\left(\mathbf{w}^T\cdot\mathbf{v}\right)^2}{||\mathbf{v}||^4} \leq \frac{||\mathbf{w}||^2}{||\mathbf{v}^2||}$$

$$|\mathbf{w}^T\cdot\mathbf{v}|^2 \leq ||\mathbf{w}||^2\cdot||\mathbf{v}||^2$$

$$\rightarrow$$

$$|\mathbf{w}^T\cdot\mathbf{v}| \leq ||\mathbf{w}||\cdot||\mathbf{v}||$$

The last equation completes the proof. We have ignored the case $\mathbf{v} = \mathbf{0}$. But in this case we obviously have:

$$\mathbf{w}^T\cdot\mathbf{v} = 0 = ||\mathbf{w}||\cdot||\mathbf{v}||$$

Triangle Inequality

The `Triangel Inequality` states:

$$||\mathbf{v} + \mathbf{w}|| \leq ||\mathbf{v}|| + ||\mathbf{w}||$$

Proof

Instead of dealing directly with the norm we look at the squared norm which is easier to evaluate.

$$||\mathbf{v}+\mathbf{w}||^2 = ||\mathbf{v}||^2 + 2\cdot\mathbf{w}^T\cdot\mathbf{v} + ||\mathbf{w}||^2$$

From the `Cauchy-Schwarz` inequality we know:

$$\mathbf{w}^T\cdot\mathbf{v} \le |\mathbf{w}^T\cdot\mathbf{v}| \le ||\mathbf{w}||\cdot||\mathbf{v}||$$

$$||\mathbf{v}+\mathbf{w}||^2 \le ||\mathbf{v}||^2 + 2\cdot||\mathbf{w}||\cdot||\mathbf{v}|| + ||\mathbf{w}||^2 = (||\mathbf{v}|| + ||\mathbf{w}||)^2$$
$$\rightarrow$$
$$||\mathbf{v}+\mathbf{w}|| \le ||\mathbf{v}|| + ||\mathbf{w}||$$

The last equation completes the proof.

Projections with more than one vector

Matrix $\mathbf{A}$ is of type $m \times n$ and vector $\mathbf{x}$ has n elements. Thus the product $\mathbf{A}\cdot\mathbf{x}$ is defined.

It follows from the columns perspective that the matrix vector product $\mathbf{A}\cdot\mathbf{x}$ is a `m`-element column vector which is a linear combination of column vectors of matrix $\mathbf{A}$ with weighting / scaling factors being elements of vector $\mathbf{x}$.

An arbitrarily chosen `m`-element columns vector $\mathbf{b}$ shall be *approximated* by $\mathbf{A}\cdot\mathbf{x}$. Defining a residual vector $\mathbf{r}$ by

$$\mathbf{r} = \mathbf{b} - \mathbf{A}\cdot\mathbf{x}$$

Ideally we would have $\mathbf{r} = \mathbf{0}$. But that is only possible if $\mathbf{b}$ is in the subspace spanned by the columns of matrix $\mathbf{A}$. Apart from this special case we have $\mathbf{r} \neq \mathbf{0}$ regardless of the choice of vector $\mathbf{x}$.

A more *relaxed* requirement is to demand that vector $\mathbf{r}$ shall be orthogonal to each column of matrix $\mathbf{A}$. Thus we require:

$$\mathbf{A}^T\cdot(\mathbf{b} - \underbrace{\mathbf{A}}_{\mathbf{r}}\cdot\mathbf{x}) = \mathbf{0}$$

The equation above is transformed in a couple of step in something that is easier to interpret.

$$
\begin{gathered}
\mathbf{A}^T \cdot \mathbf{b} - \mathbf{A}^T \cdot \mathbf{A} \cdot \mathbf{x} = \mathbf{0} \\
\mathbf{A}^T \cdot \mathbf{A} \cdot \mathbf{x} = \mathbf{A}^T \cdot \mathbf{b} \\
\left(\mathbf{A}^T \cdot \mathbf{A}\right)^{-1} \cdot \mathbf{A}^T \cdot \mathbf{A} \cdot \mathbf{x} = \left(\mathbf{A}^T \cdot \mathbf{A}\right)^{-1} \cdot \mathbf{A}^T \cdot \mathbf{b} \\
\rightarrow \\
\mathbf{x} = \underbrace{\left(\mathbf{A}^T \cdot \mathbf{A}\right)^{-1} \cdot \mathbf{A}^T}_{\textit{left inverse}} \cdot \mathbf{b}
\end{gathered}
$$

The left inverse can be computed if these conditions are fulfilled:

1. $\mathbf{A}$ is square and is full rank. Then $\mathbf{A}^T \cdot \mathbf{A}$ is also full rank and has an inverse
2. $\mathbf{A}$ is a `tall` matrix with full column rank

How to solve it

1. directly apply the formula of the left matrix inverse
2. more elegantly use a routine from `numpy`. `numpy.linalg.lstsq`

The code blocks below demonstrate both methods.

In [1]:
```
import numpy as np

# a random 4 x 2 matrix
Amat = np.random.randn(4, 2)
# a random 4 element column vector
bvec = np.random.randn(4)
```

In [2]:
```
# computing the left inverse (see formula)

Ileft = np.linalg.inv(Amat.T @ Amat) @ Amat.T
xvec_m1 = Ileft @ bvec
print(f"direct method -> xvec_m1 : {xvec_m1}")
```

```
direct method -> xvec_m1 : [-0.39320706  0.60134133]
```

In [3]:
```
# computing from least-squares

xvec_m2, residuals_1, rank, singular_values = np.linalg.lstsq(Amat, bvec, rcond=Non
print(f"least square method -> xvec_m2 : {xvec_m2}\n")
print(f"residuals : {residuals_1}")
```

```
least square method -> xvec_m2 : [-0.39320706  0.60134133]

residuals : [1.47929639]
```

In [4]:
```
# another way to compute the residuals
# -> quite similar values ...

r = bvec - Amat @ xvec_m2
```

```
residuals_2 = np.linalg.norm(r)**2
print(f"residuals : {residuals_2}")
```

```
residuals : 1.4792963946375453
```

Orthogonal matrices

Properties of orthogonal matrices:

1. column vectors are orthogonal; `i'th` column is orthogonal to `j'th` column for $i \neq j$.
2. all columns have length 1

The orthogonality / orthonormality of column vectors is summarized in this matrix product:

$$\mathbf{Q}^T \cdot \mathbf{Q} = \mathbf{I}$$

If $\mathbf{Q}$ is square it has a left- and right-sided inverse.

$$\mathbf{Q}^{-1} = \mathbf{Q}^T$$

For a tall matrix with orthonormal column vectors only the left-sided inverse exists.

No inverse is defined if the matrix is wide.

Projections (Part 2)

orthogonal projection

For orthogonal vectors $\mathbf{q}_1,\ \mathbf{q}_2, \ldots, \mathbf{q}_n$ the projection vector $\mathbf{p}$ of vector $\mathbf{b}$ is defined as:

$$\mathbf{p} = \frac{\mathbf{b}^T\mathbf{q}_1}{\mathbf{q}_1^T\mathbf{q}_1} \cdot \mathbf{q}_1 + \frac{\mathbf{b}^T\mathbf{q}_2}{\mathbf{q}_2^T\mathbf{q}_2} \cdot \mathbf{q}_2 + \cdots + \frac{\mathbf{b}^T\mathbf{q}_n}{\mathbf{q}_n^T\mathbf{q}_n} \cdot \mathbf{q}_n$$

The residual vector $\mathbf{r} = \mathbf{b} - \mathbf{p}$ is then orthogonal to each vector $\mathbf{q}_1,\ \mathbf{q}_2, \ldots, \mathbf{q}_n$.

proof

It must be shown that $\mathbf{q}_j^T \cdot \mathbf{r} = 0$.

$$\mathbf{q}_j^T \cdot \mathbf{r} = \mathbf{q}_j^T \cdot \mathbf{b} - \mathbf{q}_j^T \cdot \mathbf{p}$$

$$\mathbf{q}_j^T \cdot \mathbf{r} = \mathbf{q}_j^T \cdot \mathbf{b} - \frac{\mathbf{b}^T\mathbf{q}_1}{\mathbf{q}_1^T\mathbf{q}_1} \cdot \mathbf{q}_j^T \cdot \mathbf{q}_1 + \mathbf{q}_j^T \cdot \frac{\mathbf{b}^T\mathbf{q}_2}{\mathbf{q}_2^T\mathbf{q}_2} \cdot \mathbf{q}_j^T \cdot \mathbf{q}_2 + \cdots + \mathbf{q}_j^T \cdot \frac{\mathbf{b}^T\mathbf{q}_n}{\mathbf{q}_n^T\mathbf{q}_n} \cdot \mathbf{q}_j^T \cdot \mathbf{q}_n$$

$$\mathbf{q}_j^T \cdot \mathbf{r} = \mathbf{q}_j^T \cdot \mathbf{b} - \frac{\mathbf{b}^T\mathbf{q}_j}{\mathbf{q}_j^T\mathbf{q}_j} \cdot \mathbf{q}_j^T \cdot \mathbf{q}_j = \mathbf{q}_j^T \cdot \mathbf{b} - \mathbf{b}^T\mathbf{q}_j = 0$$

Computing an orthogonal basis

also known as `Gram-Schmidt` procedure.

Two very readable accounts I found here:

1. `QR Decomposition with Gram-Schmidt` , Igor Yanovsky (Math 151B TA)
2. `Lecture 4: Applications of Orthogonality: QR Decomposition`, author: Padraic Bartlett, UCSB 2014

A matrix $\mathbf{A}$ has `n` column vectors $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n$. These column vectors are linearly independent and span a vector space. The column vectors are in general not orthogonal / orthonomal.

Task

1. Derive a set of orthogonal vectors $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n$ which span the same vector space.
2. The set of orthogonal vectors shall be constructed from the column vectors of matrix $\mathbf{A}$.
3. normalise the set of orthogonal vectors $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n$ to obtain a set of <u>orthonormal</u> vectors $\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_n$.

The orthogonal vectors are constructed from a series of `n` steps. Each steps generates a (the next) orthonormal vector which is used in subsequent steps.

<u>step#1</u>

Take the first column vector $\mathbf{a}_1$ as orthogonal vector $\mathbf{u}_1$.

$$\mathbf{u}_1 = \mathbf{a}_1$$

<u>step#2</u>

using $\mathbf{u}_1$ and the projection theorem it is known that the residual $\mathbf{u}_2$ is orthogonal to $\mathbf{u}_1$.

$$\mathbf{u}_2 = \mathbf{a}_2 - \frac{\mathbf{a}_2 \cdot \mathbf{u}_1}{\mathbf{u}_1^T \cdot \mathbf{u}_1} \cdot \mathbf{u}_1$$

$\mathbf{u}_2$ is orthogonal to $\mathbf{u}_1$

<u>step#3</u>

$$\mathbf{u}_3 = \mathbf{a}_3 - \frac{\mathbf{a}_3 \cdot \mathbf{u}_1}{\mathbf{u}_1^T \cdot \mathbf{u}_1} \cdot \mathbf{u}_1 - \frac{\mathbf{a}_3 \cdot \mathbf{u}_2}{\mathbf{u}_2^T \cdot \mathbf{u}_2} \cdot \mathbf{u}_2$$

$\mathbf{u}_3$ is orthogonal to $\mathbf{u}_1$ and $\mathbf{u}_2$

step#(k+1) `(k > 1)`

$$\mathbf{u}_{(k+1)} = \mathbf{a}_{(k+1)} - \sum_{i=1}^{k} \frac{\mathbf{a}_{(k+1)} \cdot \mathbf{u}_i}{\mathbf{u}_i^T \cdot \mathbf{u}_i} \cdot \mathbf{u}_i$$

Repeating these step up to $k+1=n$ the complete set of orthogonal vectors $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n$ has been found. By normalising these vectors using $\mathbf{q}_i = \frac{\mathbf{u}_i}{||\mathbf{u}_i||} = \frac{\mathbf{u}_i}{\sqrt{\mathbf{u}_i^T \cdot \mathbf{u}_i}}$ the set of orthonormal vectors $\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_n$ is generated.

QR Decomposition

From the orthogonalisation procedure we express the column vectors $\mathbf{a}_k$:

step#1

$$\mathbf{a}_1 = \mathbf{u}_1$$

step#(k+1) `(k > 1)`

$$\mathbf{a}_{(k+1)} = \mathbf{u}_{(k+1)} + \sum_{i=1}^{k} \frac{\mathbf{a}_{(k+1)} \cdot \mathbf{u}_i}{\mathbf{u}_i^T \cdot \mathbf{u}_i} \cdot \mathbf{u}_i$$

The column vectors $\mathbf{a}_i$ are therefore expressed as weigthed additions of the orthogonal vectors $\mathbf{a}_{(k+1)}$.

$$\mathbf{A} = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{u}_1 & \mathbf{u}_2 & \mathbf{u}_3 & \cdots & \mathbf{u}_n \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \cdot \begin{bmatrix} 1 & \frac{\mathbf{a}_2 \cdot \mathbf{u}_1}{\mathbf{u}_1^T \cdot \mathbf{u}_1} & \frac{\mathbf{a}_3 \cdot \mathbf{u}_1}{\mathbf{u}_1^T \cdot \mathbf{u}_1} & \cdots & \frac{\mathbf{a}_n \cdot \mathbf{u}_1}{\mathbf{u}_1^T \cdot \mathbf{u}_1} \\ 0 & 1 & \frac{\mathbf{a}_3 \cdot \mathbf{u}_2}{\mathbf{u}_2^T \cdot \mathbf{u}_2} & \cdots & \frac{\mathbf{a}_n \cdot \mathbf{u}_2}{\mathbf{u}_2^T \cdot \mathbf{u}_2} \\ 0 & 0 & 1 & \cdots & \frac{\mathbf{a}_n \cdot \mathbf{u}_3}{\mathbf{u}_3^T \cdot \mathbf{u}_3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

We observe that matrix $\mathbf{A}$ is the product of two matrices. The left matrix has mutual orthogonal column vectors $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n$ while the right matrix is a upper triangular matrix of *weighting* factors.

The final step involves multiplication of the column vectors $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n$ to obtain the set of orthonormal vectors $\mathbf{q}_1 = \frac{1}{||\mathbf{u}_1||} \cdot \mathbf{u}_1, \mathbf{q}_2 = \frac{1}{||\mathbf{u}_2||} \cdot \mathbf{u}_2, \ldots, \mathbf{q}_n = \frac{1}{||\mathbf{u}_n||} \cdot \mathbf{u}_n$

To compensate for this scaling the columns of the upper triangular matrix must be scaled as well. The first row vector is scaled by $||\mathbf{u}_1||$. The second row is scaled by $||\mathbf{u}_2||$ and so on. After application of these scaling operations matrix $\mathbf{A}$ is expressed like this:

$$\mathbf{A} = \mathbf{Q} \cdot \mathbf{R} = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{q}_1 & \mathbf{q}_2 & \mathbf{q}_3 & \cdots & \mathbf{q}_n \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \cdot \left[\begin{matrix} ||\mathbf{u}_1|| & ||\mathbf{u}_1|| \cdot \frac{\mathbf{a}_2 \cdot \mathbf{u}_1}{\mathbf{u}_1^T \cdot \mathbf{u}_1} & ||\mathbf{u}_1|| \cdot \frac{\mathbf{a}_3 \cdot \mathbf{u}_1}{\mathbf{u}_1^T \cdot \mathbf{u}_1} & \cdots & ||\mathbf{u}_1 \\ 0 & ||\mathbf{u}_2|| & ||\mathbf{u}_2|| \cdot \frac{\mathbf{a}_3 \cdot \mathbf{u}_2}{\mathbf{u}_2^T \cdot \mathbf{u}_2} & \cdots & ||\mathbf{u}_2 \\ 0 & 0 & ||\mathbf{u}_3|| & \cdots & ||\mathbf{u}_3 \\ \vdots & \vdots & \vdots & \ddots & \\ 0 & 0 & 0 & 0 & | \end{matrix} \right.$$

The last matrix product is known as `QR-Decomposition`.

A slightly different form is obtained by transforming the upper triangular matrix.

$$\mathbf{A} = \mathbf{Q} \cdot \mathbf{R} = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{q}_1 & \mathbf{q}_2 & \mathbf{q}_3 & \cdots & \mathbf{q}_n \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \cdot \begin{bmatrix} ||\mathbf{u}_1|| & \mathbf{a}_2 \cdot \mathbf{q}_1 & \mathbf{a}_3 \cdot \mathbf{q}_1 & \cdots & \mathbf{a}_n \cdot \mathbf{q}_1 \\ 0 & ||\mathbf{u}_2|| & \mathbf{a}_3 \cdot \mathbf{q}_2 & \cdots & \mathbf{a}_n \cdot \mathbf{q}_2 \\ 0 & 0 & ||\mathbf{u}_3|| & \cdots & \mathbf{a}_n \cdot \mathbf{q}_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & ||\mathbf{u}_n|| \end{bmatrix}$$

A numerical example of QR decomposition

In [5]:
```
Amat = np.array([[2, 1, 3, 3], [2, 1, -1, 1], [2, -1, 3, -3], [2, -1, -1, -1]])

Qmat, Rmat = np.linalg.qr(Amat, mode='complete')

# compute Amat from Qmat, Rmat as a sanity check (should be identical apart from ro
Amat_c = Qmat @ Rmat

print(f"Amat    :\n{Amat}\n")
print(f"Qmat    :\n{Qmat}\n")
```

```
print(f"Rmat   :\n{Rmat}\n")
print(f"Amat_c :\n{Amat_c}")
```

```
Amat   :
[[ 2  1  3  3]
 [ 2  1 -1  1]
 [ 2 -1  3 -3]
 [ 2 -1 -1 -1]]

Qmat   :
[[-0.5 -0.5 -0.5 -0.5]
 [-0.5 -0.5  0.5  0.5]
 [-0.5  0.5 -0.5  0.5]
 [-0.5  0.5  0.5 -0.5]]

Rmat   :
[[-4.  0. -2.  0.]
 [ 0. -2.  0. -4.]
 [ 0.  0. -4.  0.]
 [ 0.  0.  0. -2.]]

Amat_c :
[[ 2.  1.  3.  3.]
 [ 2.  1. -1.  1.]
 [ 2. -1.  3. -3.]
 [ 2. -1. -1. -1.]]
```

Application of QR decomposition

The matrix equation

$$\mathbf{A} \cdot \mathbf{x} = \mathbf{b}$$

shall be solved using the `QR-decompostion`. (assuming that $\mathbf{A}$ has an inverse)

Re-writing the matrix equation

$$\mathbf{Q} \cdot \mathbf{R} \cdot \mathbf{x} = \mathbf{b}$$

and left multiplying both sides by $\mathbf{Q}^T$ yields:

$$\mathbf{Q}^T \cdot \mathbf{Q} \cdot \mathbf{R} \cdot \mathbf{x} = \mathbf{Q}^T \cdot \mathbf{b}$$
$$\mathbf{R} \cdot \mathbf{x} = \mathbf{Q}^T \cdot \mathbf{b} = \mathbf{c}$$

The fact that $\mathbf{R}$ is an upper-triangular matrix makes computation of the elements of vector $\mathbf{x}$ fairly easy.

In [ ]:

```

```