ME 536

Weeks 2 ...
Some math 2 *remember*

Adopted Material

Several of the slides in this presentation are adopted from **Dr. Sekmen**'s workshop at TSU (Tennessee State University, Computer Science Department) where the videos and powerpoint slides are accessible at:

http://www.tnstate.edu/computer_science/datascience/mini_courses.aspx

Adopted slides will have a logo at that slide.

Collections - sets of things:

A subset of Real numbers

Readings from a sensor array

Elements in a set - people around a table

Grades you get in a semester, or all of them in your transcript

Set of movies you have seen

Association of Meaning to Things based on:

Relationship of elements in a set:

Are there meaningful subsets ?

Change in the values:

Is the sensor trying to tell something ?

Past trend to guess:

Get ready for what is next...

But after all, these are all numbers... in a sense...

Values - in this context ***scalars***

Will mostly come from the set of ***Real Numbers***

Other possible alternatives are:

- Integers
- Complex numbers
- ...

Collection of values form:

- Vectors,
- Matrices,
- In general, Tensors of different degrees (orders)

Vectors

- Has a magnitude and a direction (meaningful mostly in *2-3D*) - *ME intuition*
- Represents collection of values / numbers that represent:
    - Direction and magnitude :)
    - Location
    - Area
    - Volume
    - Hyper-planes, -volumes

Vectors

A vector:

- Number of elements $\rightarrow$ Dimension of the vector
- Dimension of the vector $\rightarrow$ Dimension of the ambient **space** it lives in
- Might have finite or infinite elements

Convention on Notation

In our lecture notes and assignments:

scalars will be represented with lowercase italic letters: s

vectors will be represented with bold lowercase letters: $\mathbf{v}$

MATRICES will be represented with bold captial letters: $\mathbf{M}$

$\mathbf{r}_i^{\mathbf{M}}$ indicates the i^{th} row of matrix $\mathbf{M}$

$\mathbf{c}_j^{\mathbf{M}}$ indicates the j^{th} column of matrix $\mathbf{M}$

In the presence of a single matrix of interest,
i^{th} row and j^{th} column is represented as $\mathbf{r}_i$ and $\mathbf{c}_j$ respectivey.

Given that we assume data is in the columns,
j^{th} column of matrix $\mathbf{M}$ as $\mathbf{m}_j$
Note that matrix is upper-case and its column being a vector
it is lower-case, yet both bold.

m_{ij} or $m_{i,j}$ or $\mathbf{M}(i,j)$ indicates
the element of $\mathbf{M}$ at location (i,j)

Note that, the letter 'm' is due to it being an element of $\mathbf{M}$
Similarly for some matrix $\mathbf{Q}$, element at $(3,4)$ is $q_{3,4}$

$\tilde{\mathbf{M}}$ is an approximation of $\mathbf{M}$, i.e. $\tilde{\mathbf{M}} \approx \mathbf{M}$

$\tilde{\mathbf{M}}_k$ is a *rank-k* approximation of $\mathbf{M}$

Finally, unit vectors will be given with a hat on top.
Ex: If vector $\mathbf{b}$ with some random length is normalized, we will get $\hat{\mathbf{b}}$

Vector Space - yes they live in spaces

- A vector space **is a set of objects** that may be added together or multiplied by numbers (called scalars)
  - Scalars are typically real numbers
    - But can be complex numbers, rational numbers, or generally any field
  - Vector addition and scalar multiplication must satisfy certain requirements (called axioms)

Example: $\mathbb{R}^2$

1. You can add $x \in \mathbb{R}^2$ and $y \in \mathbb{R}^2$ and $x + y \in \mathbb{R}^2$
2. You can multiple $x \in \mathbb{R}^2$ by $\alpha \in \mathbb{R}$ and $\alpha x \in \mathbb{R}^2$

Vector Space

A set $\mathcal{V}$ is called a vector space over field $\mathcal{F}$ (often $\mathbb{R}$ or $\mathbb{C}$) when vector addition and scalar multiplication satisfies:

Vector Addition

1. $v + w \in \mathcal{V}$ for all $v, w \in \mathcal{V}$
2. $(v + w) + u = v + (w + u) \in \mathcal{V}$ for all $v, w, u \in \mathcal{V}$ Associativity
3. $v + w = w + v$ for all $v, w \in \mathcal{V}$ Commutativity
4. There is an element $0 \in \mathcal{V}$ such that $v + 0 = v$ for all $v \in \mathcal{V}$ Identity Element
5. For every $v \in \mathcal{V}$, there exist $-v \in \mathcal{V}$ such that $v + (-v) = 0$ Inverse Element

Scalar Multiplication

1. $\alpha v \in \mathcal{V}$ for all $\alpha \in \mathcal{F}$ and $v \in \mathcal{V}$
2. $(\alpha\beta)v = \alpha(\beta v)$ for all $\alpha, \beta \in \mathcal{F}$ and $v \in \mathcal{V}$ Compatibility
3. $\alpha(v + w) = \alpha v + \alpha w$ for all $\alpha \in F$ and $v, w \in V$
4. $(\alpha + \beta)v = \alpha v + \beta v$ for all $\alpha, \beta \in \mathcal{F}$ and $v \in V$

Distributivity

5. $1v = v$ for all $v \in V$ Identity Element

Vector Space - yes they are a set of objects

A set $\mathcal{V}$ is called a vector space over field $\mathcal{F}$ (often $\mathbb{R}$ or $\mathbb{C}$) when vector addition and scalar multiplication satisfies:

Vector Addition

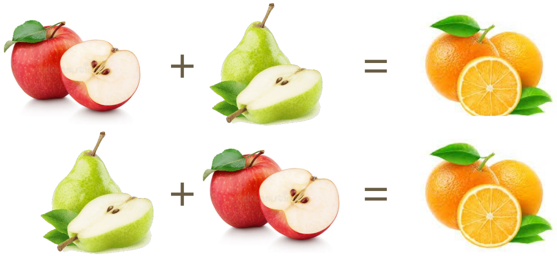
1. $v + w \in \mathcal{V}$ for all $v, w \in \mathcal{V}$
2. $(v + w) + u = v + (w + u) \in \mathcal{V}$ for all $v, w, u \in \mathcal{V}$ Associativity
3. $v + w = w + v$ for all $v, w \in \mathcal{V}$ Commutativity
4. There is an element $0 \in \mathcal{V}$ such that $v + 0 = v$ for all $v \in \mathcal{V}$ Identity Element
5. For every $v \in \mathcal{V}$, there exist $-v \in \mathcal{V}$ such that $v + (-v) = 0$ Inverse Element

Scalar Multiplication

1. $\alpha v \in \mathcal{V}$ for all $\alpha \in \mathcal{F}$ and $v \in \mathcal{V}$
2. $(\alpha\beta)v = \alpha(\beta v)$ for all $\alpha, \beta \in \mathcal{F}$ and $v \in \mathcal{V}$ Compatibility
3. $\alpha(v + w) = \alpha v + \alpha w$ for all $\alpha \in F$ and $v, w \in V$ Distributivity
4. $(\alpha + \beta)v = \alpha v + \beta v$ for all $\alpha, \beta \in \mathcal{F}$ and $v \in V$
5. $1v = v$ for all $v \in V$ Identity Element

With: **a set of objects** {  }

- 2 operators: addition and multiplication
- Some properties for each operation



...

In general - Even if we are interested in objects

They are translated into numbers

- Therefore a vector is not necessarily an arrow :) might have a corresponding meaning
- We might need to assess **how close** 2 vectors are

$$+ \begin{bmatrix} 2.5 \\ 1.25 \\ 1.5 \end{bmatrix}$$

$$\begin{bmatrix} 2.5 \\ 1.25 \\ 1.5 \end{bmatrix} > ? < \begin{bmatrix} 1.5 \\ 2.5 \\ 2.25 \end{bmatrix}$$

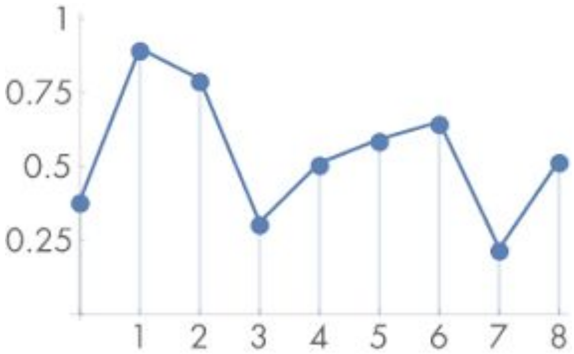
$$+ \begin{bmatrix} 1.5 \\ 2.5 \\ 2.25 \end{bmatrix}$$

Vector Space

- A vector space may have additional structures such as a norm or inner product
  - This is typical for infinite dimensional function spaces whose vectors are functions
- Many practical problems require ability to decide whether a sequence of vectors converges to a given vector
  - In order to allow proximity and continuity considerations, most vector spaces are endowed with a suitable topology
    - A topology is a structure that allows to define "being close to each other"
    - Such topological vector spaces have richer theory
      - Banach space topology is given by a norm
      - Hilbert space topology is given by an inner product

Vectors to represents functions

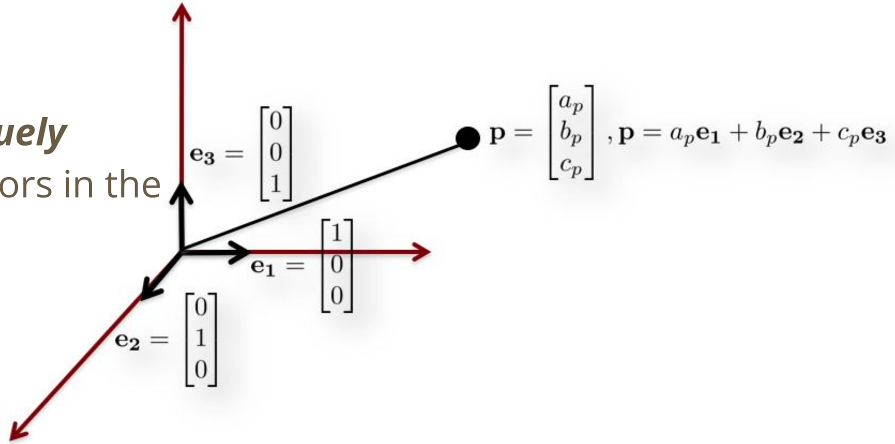
Just recall ME310



But let the number of elements $\rightarrow \infty$

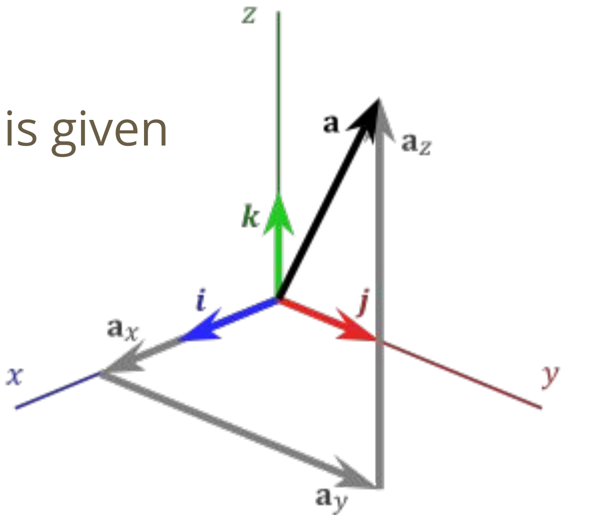
On what basis?

- For every vector space **bases** (i.e. more than 1 basis) can be written
- A basis is formed by just sufficient number of vectors to **span** that **space**
- Hence:
  - All vectors in this vector space can be ***uniquely*** defined as a linear combination of the vectors in the selected basis
  - Vector $\rightarrow$ ***Matrix*** $\mathbf{E}$ times a vector $\mathbf{p}$: $\mathbf{Ep}$
  - ***Matrix*** contains $\mathbf{e_i}$ in its columns



Standard basis

- Standard basis is implied if no basis is given
- Invisibly there all the time
- Simple, but not unique



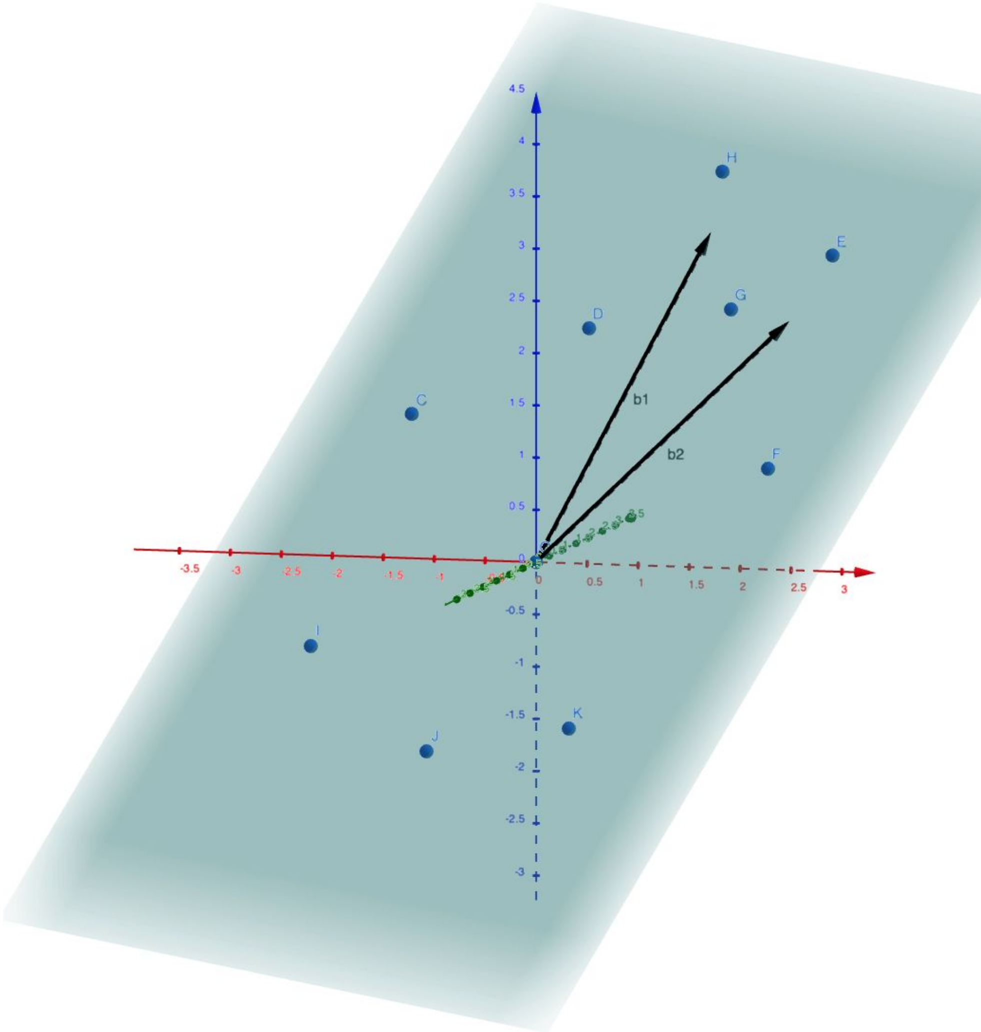
$$\mathbf{e}_1 = (1,0,0), \quad \mathbf{e}_2 = (0,1,0), \quad \mathbf{e}_3 = (0,0,1).$$

After a certain number of dimensions letters will run out!!!

Some 2 - 3D exercises

Standard vs other bases

Why do we need them?



Exercise

Write the following with respect to standard basis e_i:

$$\begin{bmatrix} 2 \\ 5 \end{bmatrix} =$$

Exercise

Write the following with respect to basis: $\mathbf{b}_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \mathbf{b}_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$$\begin{bmatrix} 2 \\ 5 \end{bmatrix} = x\mathbf{b}_1 + y\mathbf{b}_2 = \begin{bmatrix} \mathbf{b}_1 & \mathbf{b}_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 2 \\ 5 \end{bmatrix}$$

$x = ? \quad y = ?$

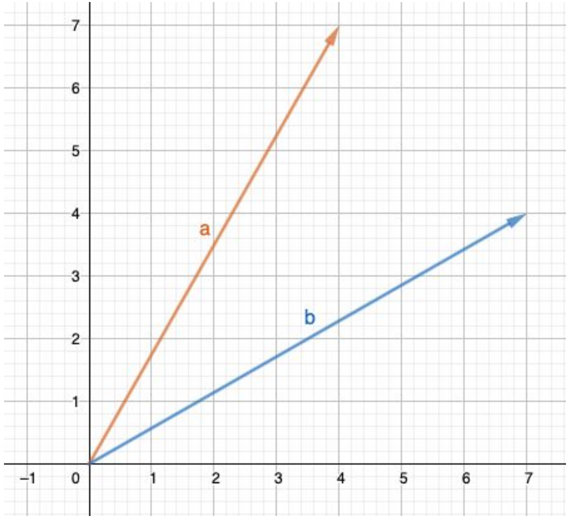
$$\begin{bmatrix} \mathbf{b}_1 & \mathbf{b}_2 \end{bmatrix} \begin{bmatrix} x_1 & x_2 \\ y_1 & y_2 \end{bmatrix} = \begin{bmatrix} 3 & 0 \\ 5 & 1 \end{bmatrix}$$

$x_i = ? \quad y_i = ?$

Recall: Dot Product

We will later refer to Dot Product as *Inner Product*:

$$< \mathbf{a}, \mathbf{b} >= ||\mathbf{a}|| \, ||\mathbf{b}|| cos(\theta)$$

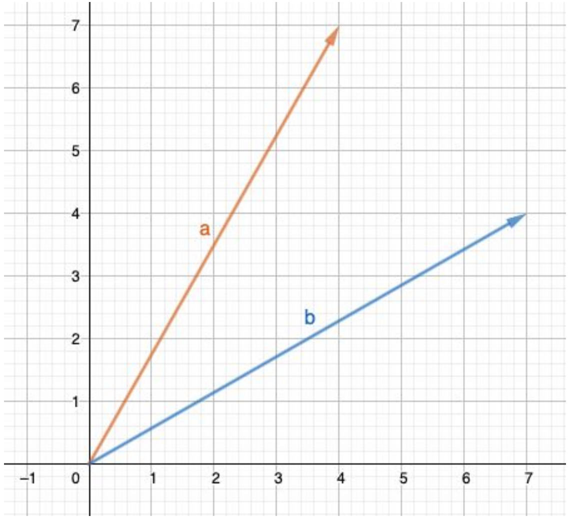


Recall: Dot Product

We will later refer to Dot Product as *Inner Product*:

$$< \mathbf{a}, \mathbf{b} >= \|\mathbf{a}\| \|\mathbf{b}\| cos(\theta)$$

$$< \mathbf{a}, \mathbf{b} >=$$



$$\mathbf{a} = \begin{bmatrix} 4 \\ 7 \end{bmatrix}$$

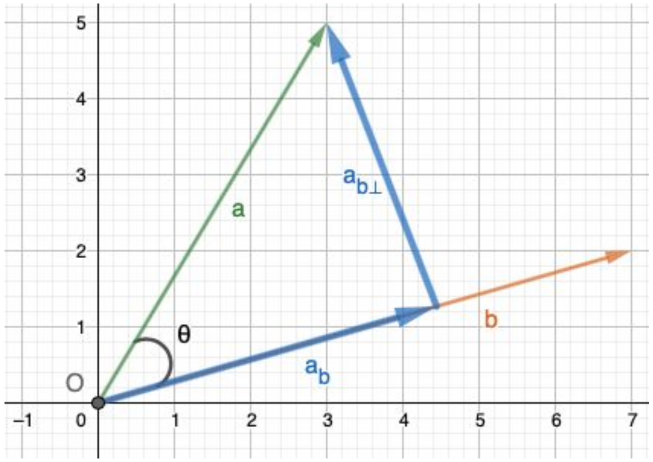
$$\mathbf{b} = \begin{bmatrix} 7 \\ 4 \end{bmatrix}$$

Exercise:

On colab, help me write a function that generates N random data points on a plane spanned by v1, v2 in 3D.

If spanning vectors are not given, they should be random. The function should also allow addition of noise to the data

Recall: Orthogonal decomposition & Projection



Let $\mathbf{a}, \mathbf{b} \in \mathbb{R}^2$ and $\mathbf{a}_b$ is the component of $\mathbf{a}$ along $\mathbf{b}$, and $\mathbf{a}_{b\perp}$ is the component of $\mathbf{a}$ that is perpendicular to $\mathbf{b}$.

Question is, given $\mathbf{a}$ and $\mathbf{b}$ how can you find $\mathbf{a}_b$ and $\mathbf{a}_{b\perp}$?

Note that:

$$\mathbf{a} = \mathbf{a}_b + \mathbf{a}_{b\perp}$$

Therefore, if we find $\mathbf{a}_b$, then,

$$\mathbf{a}_{b\perp} = \mathbf{a} - \mathbf{a}_b$$

Finding $\mathbf{a}_b$ then simply is a unit projection vector of $\mathbf{a}$ along $\mathbf{b}$ and multiplying it with the $||\mathbf{a}_b||$.
Note that:

$$||\mathbf{a}_b|| = ||\mathbf{a}||cos(\theta)$$

Recall by definition:

$$< \mathbf{a}, \mathbf{b} >= ||\mathbf{a}||\,||\mathbf{b}||cos(\theta)$$

Hence,

$$||\mathbf{a}_b|| = \frac{<\mathbf{a},\mathbf{b}>}{||\mathbf{b}||}$$

If this magnitude is multiplied by a unit vector along $\mathbf{b}$ we are done with $\mathbf{a}_b$ as:

$$\mathbf{a}_b = \frac{<\mathbf{a},\mathbf{b}>}{||\mathbf{b}||}\frac{\mathbf{b}}{||\mathbf{b}||} = \frac{<\mathbf{a},\mathbf{b}>}{||\mathbf{b}||^2}\mathbf{b} = \frac{<\mathbf{a},\mathbf{b}>}{<\mathbf{b},\mathbf{b}>}\mathbf{b}$$

Finally:

$$\mathbf{a}_{b\perp} = \mathbf{a} - \mathbf{a}_b = \mathbf{a} - \frac{<\mathbf{a},\mathbf{b}>}{||\mathbf{b}||^2}\mathbf{b}$$

Also note that:

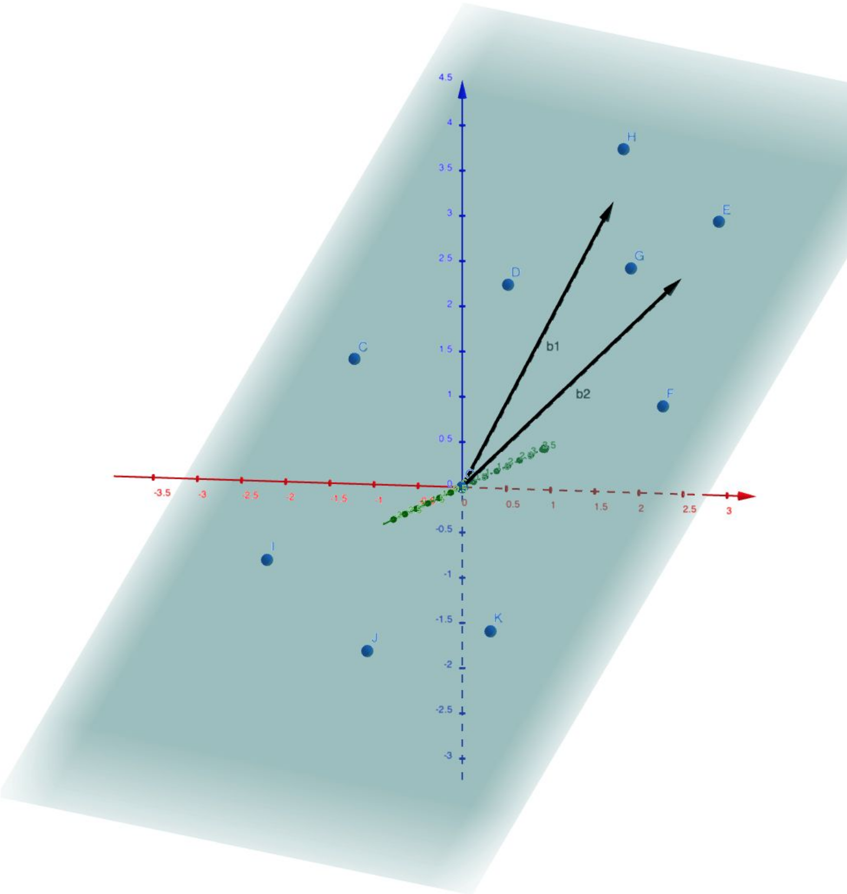
$$\mathbf{a}_{b\perp} = \mathbf{a} - \frac{<\mathbf{a},\mathbf{b}>}{<\mathbf{b},\mathbf{b}>}\mathbf{b}$$

If $\mathbf{b}$ was already unit, things would have been easier:

$$\mathbf{a}_b =< \mathbf{a}, \mathbf{b} > \mathbf{b}$$
$$\mathbf{a}_{b\perp} = \mathbf{a}- < \mathbf{a}, \mathbf{b} > \mathbf{b}$$

How to find an orthonormal basis given data

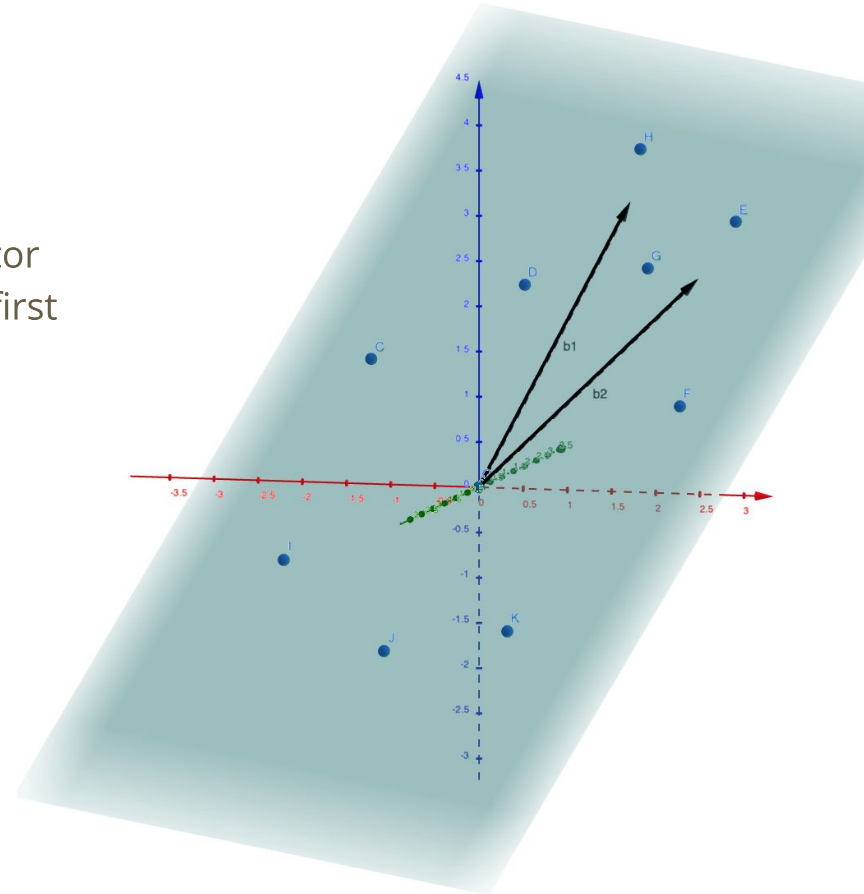
What do you think?



How to find an orthonormal basis given data

Gram-Schmidt says:

- Take one of the data point to be the first basis vector
- Take another, find the perpendicular comp to the first
- Take another, find the perpendicular comp to the previous two
- Take another, find the perpendicular comp to the previous three
- ...



How to find an orthonormal basis given data:

Gram-Schmidt way - w/ geogebra example

Let the data matrix $\mathbf{M}_{nxm} = [\mathbf{c}_1\mathbf{c}_2 \cdots \mathbf{c}_m]$ where $\mathbf{c}_i$, i.e. columns of $\mathbf{M}$, are our data points, and $\mathbf{c}_i \in \mathbb{R}^n$. Let's assume that data lives in a d dimensional subspace in $\mathbb{R}$ where $d < n$ and for the sake of argument assume that we have enough or more than enough data, i.e. $d \leq m$.

To find an **orthogonal** basis for the data:

1. select one of the data points as the frist basis vector, without loss of generality, we can choose the first data point

$$\mathbf{v}_1 = \mathbf{c}_1$$

2. find a perpendicular vector to $\mathbf{v}_1$ using $\mathbf{c}_2$:

$$\mathbf{v}_2 = \mathbf{c}_2 - \frac{\mathbf{c}_2 \cdot \mathbf{v}_1}{\mathbf{v}_1 \cdot \mathbf{v}_1}\mathbf{v}_1$$

3. find a perpendicular vector to *Span*$\{\mathbf{v}_1, \mathbf{v}_2\}$ using $\mathbf{c}_3$:

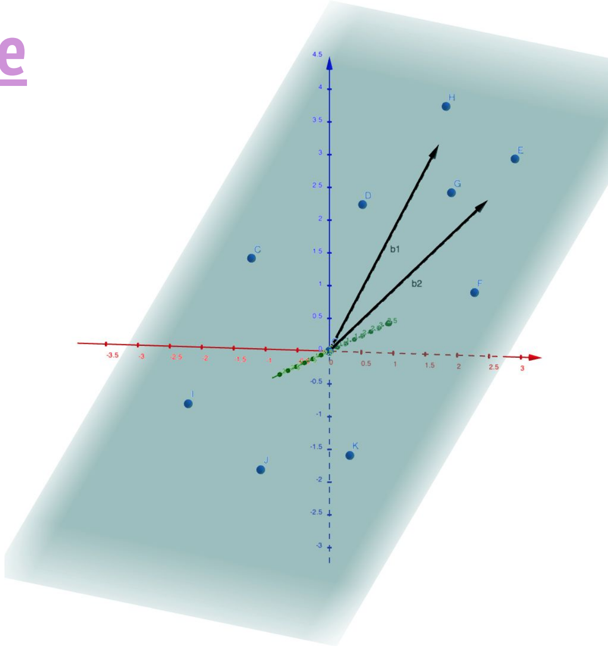
$$\mathbf{v}_3 = \mathbf{c}_3 \underbrace{- \frac{\mathbf{c}_3 \cdot \mathbf{v}_1}{\mathbf{v}_1 \cdot \mathbf{v}_1}\mathbf{v}_1 - \frac{\mathbf{c}_3 \cdot \mathbf{v}_2}{\mathbf{v}_2 \cdot \mathbf{v}_2}\mathbf{v}_2}_{- \perp\{\mathbf{v}_1, \mathbf{v}_2\}}$$

...

continue until you find all basis vectors i.e. $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_d\}$

If an **orthonormal** basis is required, normalize $\mathbf{v}_i$:

$$\mathbf{v}_i = \frac{1}{\sqrt{\mathbf{v}_i \cdot \mathbf{v}_i}}\mathbf{v}_i$$



RECALL:

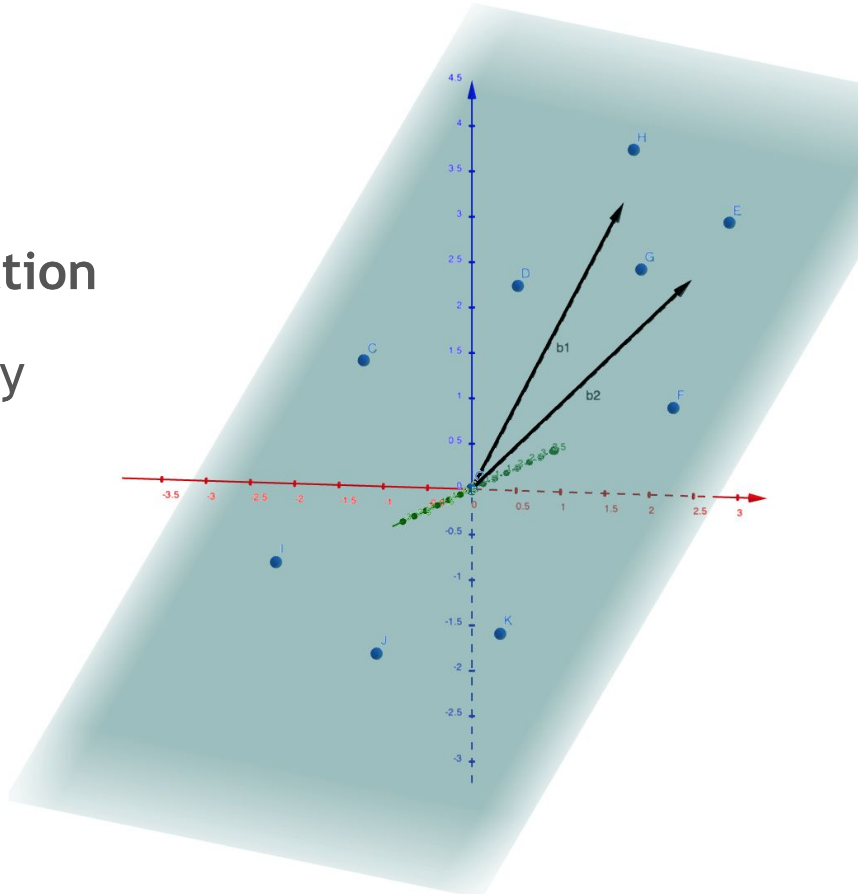
$$\mathbf{a}_{b\perp} = \mathbf{a} - \mathbf{a}_b = \mathbf{a} - \frac{\langle \mathbf{a},\mathbf{b} \rangle}{\|\mathbf{b}\|^2}\mathbf{b}$$

How to find an orthonormal basis given data: Alternative to Gram-Schmidt

Check out:

Householder Orthagonalization

for better numerical stability



How to find an orthonormal basis given data: Python way

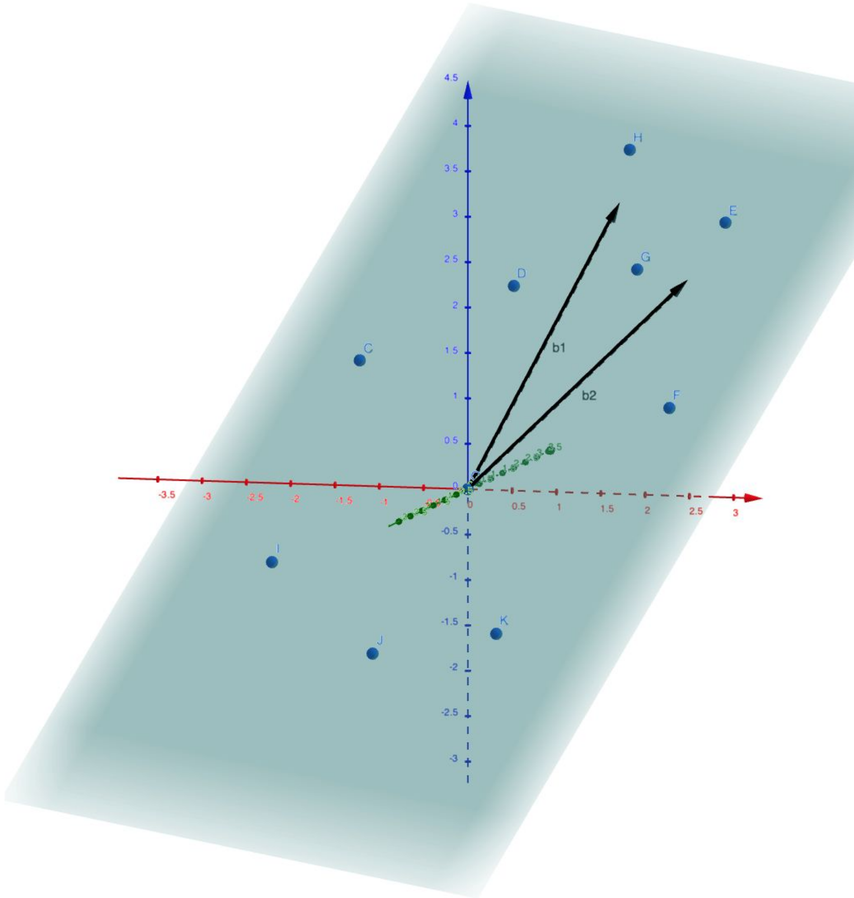
Check out:

```
scipy.linalg.orth()
```

and

```
numpy.linalg.qr()
```

To get an insight about each



Recall: Orthogonal Matrices (a.k.a. Orthogonal)

`scipy.orth` just returns an orthonormal basis:

```
Data Matrix
|      9.00      0.00      4.00      8.00      8.00 |
|      3.00      2.00      8.00      8.00      2.00 |
|      8.00      0.00      2.00      6.00      0.00 |
```

B = orth(**D**)

Why do we want and like orthogonal matrices?

```
B
|    -0.72     0.45    -0.54 |
|    -0.53    -0.85     0.00 |
|    -0.45     0.28     0.84 |
```

```
B @ B.T
|      1.00     -0.00      0.00 |
|     -0.00      1.00      0.00 |
|      0.00      0.00      1.00 |

B.T @ B
|      1.00     -0.00      0.00 |
|     -0.00      1.00     -0.00 |
|      0.00     -0.00      1.00 |
```

Example: `scipy.orth(D)` just returns the basis: B

Hence we have **D**, and **B** = orth(**D**)

Data Matrix

9.00	0.00	4.00	8.00	8.00
3.00	2.00	8.00	8.00	2.00
8.00	0.00	2.00	6.00	0.00

B

-0.72	0.45	-0.54
-0.53	-0.85	0.00
-0.45	0.28	0.84

B @ B.T

1.00	-0.00	0.00
-0.00	1.00	0.00
0.00	0.00	1.00

B.T @ B

1.00	-0.00	0.00
-0.00	1.00	-0.00
0.00	-0.00	1.00

But what are the coordinates **C** of **D** with respect to **B**?

$$\mathbf{D} = \mathbf{BC}$$

$$\mathbf{C} = \mathbf{B}^{-1}\mathbf{D}$$

Is **B** always invertible?

Example: Find an orthogonal basis using QR

```
Data Matrix
|      9.00      0.00      4.00      8.00      8.00 |
|      3.00      2.00      8.00      8.00      2.00 |
|      8.00      0.00      2.00      6.00      0.00 |
```

What will be the shape and structure of these matrices where **D = QR**?

```
Q
|     -0.73      0.18     -0.66 |
|     -0.24     -0.97     -0.00 |
|     -0.64      0.16      0.75 |
```

```
Q @ Q.T
|      1.00      0.00     -0.00 |
|      0.00      1.00      0.00 |
|     -0.00      0.00      1.00 |
Q.T @ Q
|      1.00     -0.00     -0.00 |
|     -0.00      1.00      0.00 |
|     -0.00      0.00      1.00 |
```

```
R
|    -12.41     -0.48     -6.12    -11.60     -6.29 |
|      0.00     -1.94     -6.72     -5.35     -0.50 |
|      0.00      0.00     -1.16     -0.83     -5.31 |
```

Example: change of basis using $\mathtt{D} = \mathtt{QR}$

```
Data Matrix
|      9.00      0.00      4.00      8.00      8.00 |
|      3.00      2.00      8.00      8.00      2.00 |
|      8.00      0.00      2.00      6.00      0.00 |
```

=

```
Q
|    -0.73      0.18     -0.66 |
|    -0.24     -0.97     -0.00 |
|    -0.64      0.16      0.75 |
```

```
R
|   -12.41     -0.48     -6.12    -11.60     -6.29 |
|     0.00     -1.94     -6.72     -5.35     -0.50 |
|     0.00      0.00     -1.16     -0.83     -5.31 |
```

Coordinates with respect to:

```
Data Matrix
|      9.00      0.00      4.00      8.00      8.00 |
|      3.00      2.00      8.00      8.00      2.00 |
|      8.00      0.00      2.00      6.00      0.00 |
```

Standard basis $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$

```
R
|   -12.41     -0.48     -6.12    -11.60     -6.29 |
|     0.00     -1.94     -6.72     -5.35     -0.50 |
|     0.00      0.00     -1.16     -0.83     -5.31 |
```

Columns of $\mathbf{Q}$ i.e. $\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3$

Exercise: Is it always good to go by the book?

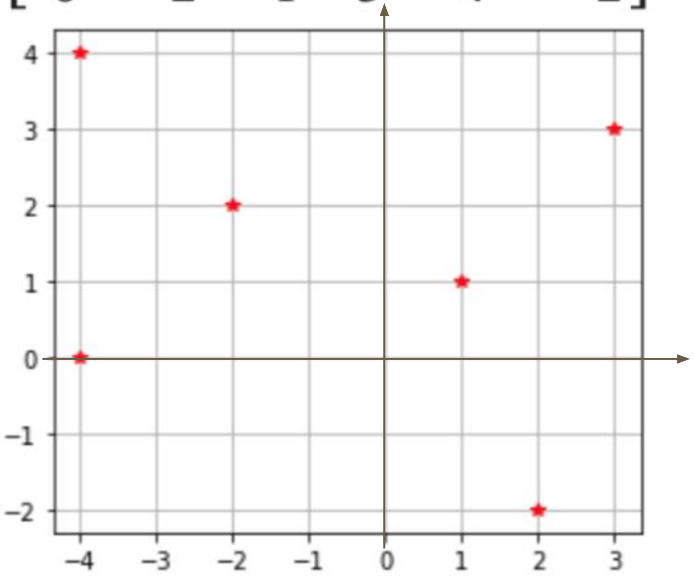
Find an orthogonal basis for the data points on the columns of

$$\begin{bmatrix} -4 & -2 & 1 & 3 & -4 & -2 \\ 0 & 2 & 1 & 3 & 4 & -2 \end{bmatrix}$$

Exercise:

Should we rush into the Gram-Schmidt right away or first analyze data?

$$\begin{bmatrix} -4 & -2 & 1 & 3 & -4 & -2 \\ 0 & 2 & 1 & 3 & 4 & -2 \end{bmatrix}$$



Note that choice of the first column as the first basis is OK but not optimal

Write the coordinates of given points w.r.t. the following bases and observe the difference: *colab might help*

a. column 1 and column 2
b. column 2 and column 3

Exercise: Find a *basis*

$$\begin{bmatrix} 4 & 2 & -1 & -8 & 12 & 1 \\ 2 & 6 & -3 & -4 & 6 & 3 \\ 5 & 4 & -2 & -10 & 15 & 2 \end{bmatrix}$$

Is orthogonal always the best?

Find an orthogonal basis and write down the coordinates wrt to the this basis

Exercise: Find a *basis*

$$\begin{bmatrix} 4 & 2 & -1 & -8 & 12 & 1 \\ 2 & 6 & -3 & -4 & 6 & 3 \\ 5 & 4 & -2 & -10 & 15 & 2 \end{bmatrix}$$

Is orthogonal always the best?

Find an orthogonal basis and write down the coordinates wrt to the this basis

First attempt: $\mathbf{M} = \mathbf{M}\mathrm{b} * \mathbf{M}\mathrm{new} \rightarrow \mathbf{M}\mathrm{new} = \mathbf{M}\mathrm{b}^{-1} * \mathbf{M}$

```
M = np.array([[4,2,-1,-8,12,1], [2,6,-3,-4,6,3],[5,4,-2,-10,15,2]])
Mb = orth(M)
Mnew = inv(Mb) @ M
```


```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-11-f7786f902e17> in <cell line: 6>()
      4 M = np.array([[4,2,-1,-8,12,1], [2,6,-3,-4,6,3],[5,4,-2,-10,15,2]])
      5 Mb = orth(M)
----> 6 Mnew = inv(Mb) @ M
      7 Mnew = inv(Mb.T@Mb)@Mb.T@ M
      8 #print(f'new basis:\n{np.around(Mb,2)}')

/usr/local/lib/python3.10/dist-packages/scipy/linalg/_basic.py in inv(a, overwrite_a, check_finite)
    940     a1 = _asarray_validated(a, check_finite=check_finite)
    941     if len(a1.shape) != 2 or a1.shape[0] != a1.shape[1]:
--> 942         raise ValueError('expected square matrix')
    943     overwrite_a = overwrite_a or _datacopied(a1, a)
    944     getrf, getri, getri_lwork = get_lapack_funcs(('getrf', 'getri',

ValueError: expected square matrix
```

Exercise: Find a *basis*

$$\begin{bmatrix} 4 & 2 & -1 & -8 & 12 & 1 \\ 2 & 6 & -3 & -4 & 6 & 3 \\ 5 & 4 & -2 & -10 & 15 & 2 \end{bmatrix}$$

Is orthogonal always the best?

Find an orthogonal basis and write down the coordinates wrt to the this basis

```
M = np.array([[4,2,-1,-8,12,1], [2,6,-3,-4,6,3],[5,4,-2,-10,15,2]])
Mb = orth(M)
Mnew = inv(Mb.T@Mb)@Mb.T@ M
```

We know $(M_\circ)^{-1}$ did not work

```
New cooridates of data matrix M
|     4.00      2.00     -1.00     -8.00     12.00      1.00 |
|     2.00      6.00     -3.00     -4.00      6.00      3.00 |
|     5.00      4.00     -2.00    -10.00     15.00      2.00 |
with respect to basis
|    -0.57      0.38 |
|    -0.36     -0.91 |
|    -0.74      0.14 |
are
|    -6.70     -6.23      3.12     13.39    -20.09     -3.12 |
|     0.41     -4.15      2.07     -0.83      1.24     -2.07 |
```

What is **interesting** here?

Pay attention to the **dimensions** of the new coordinate matrix!!!

Exercise: Find a *basis*

$$\begin{bmatrix} 4 & 2 & -1 & -8 & 12 & 1 \\ 2 & 6 & -3 & -4 & 6 & 3 \\ 5 & 4 & -2 & -10 & 15 & 2 \end{bmatrix}$$

Is orthogonal always the best?

Find a better basis but how?

Exercise: Find a *basis*

May be it is just in front of your eyes?

$$\begin{bmatrix} 4 & 2 & -1 & -8 & 12 & 1 \\ 2 & 6 & -3 & -4 & 6 & 3 \\ 5 & 4 & -2 & -10 & 15 & 2 \end{bmatrix}$$

Find a better basis [but how]?

```
new coordinates of
[[  4   2  -1  -8  12   1]
 [  2   6  -3  -4   6   3]
 [  5   4  -2 -10  15   2]]
wrt
 [[4 2]
 [2 6]
 [5 4]]
are:
 [[ 1.  -0.   0.  -2.   3.  -0. ]
 [-0.   1.  -0.5  0.  -0.   0.5]]
```

Exercise: Find a *basis*

$$\begin{bmatrix} 4 & 2 & -1 & -8 & 12 & 1 \\ 2 & 6 & -3 & -4 & 6 & 3 \\ 5 & 4 & -2 & -10 & 15 & 2 \end{bmatrix}$$

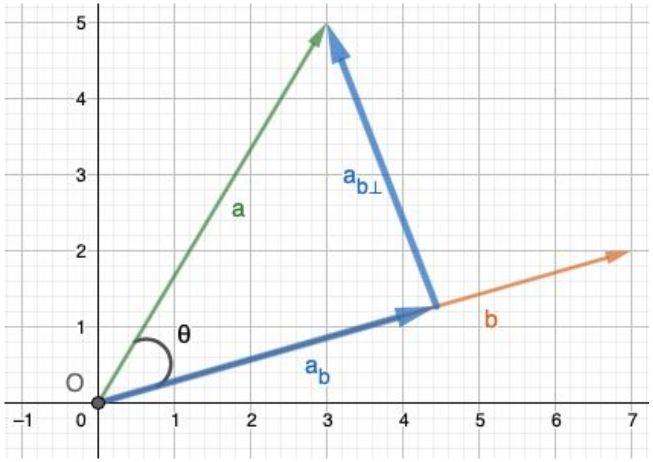
Is orthogonal always the best?

Find a better basis but how?

```
[[ -6.7    -6.23    3.12   13.39 -20.09   -3.12]
 [  0.41  -4.15    2.07  -0.83    1.24  -2.07]]
vs
[[ 1.   -0.    0.   -2.    3.   -0. ]
 [-0.    1.   -0.5  0.   -0.    0.5]]
```

STILL HOW if this was a higher dimensional much bigger matrix?

Recall: Orthogonal decomposition & Projection



Let $\mathbf{a}, \mathbf{b} \in \mathbb{R}^2$ and $\mathbf{a}_b$ is the component of $\mathbf{a}$ along $\mathbf{b}$, and $\mathbf{a}_{b\perp}$ is the component of $\mathbf{a}$ that is perpendicular to $\mathbf{b}$.

Question is, given $\mathbf{a}$ and $\mathbf{b}$ how can you find $\mathbf{a}_b$ and $\mathbf{a}_{b\perp}$?

Note that:

> $\mathbf{a} = \mathbf{a}_b + \mathbf{a}_{b\perp}$

Therefore, if we find $\mathbf{a}_b$, then,

> $\mathbf{a}_{b\perp} = \mathbf{a} - \mathbf{a}_b$

Finding $\mathbf{a}_b$ then simply is a unit projection vector of $\mathbf{a}$ along $\mathbf{b}$ and multiplying it with the $\|\mathbf{a}_b\|$.
Note that:

> $\|\mathbf{a}_b\| = \|\mathbf{a}\| cos(\theta)$

Recall by definition:

> $< \mathbf{a}, \mathbf{b} >= \|\mathbf{a}\| \|\mathbf{b}\| cos(\theta)$

Hence,

> $\|\mathbf{a}_b\| = \frac{<\mathbf{a},\mathbf{b}>}{\|\mathbf{b}\|}$

If this magnitude is multiplied by a unit vector along $\mathbf{b}$ we are done with $\mathbf{a}_b$ as:

> $\mathbf{a}_b = \frac{<\mathbf{a},\mathbf{b}>}{\|\mathbf{b}\|} \frac{\mathbf{b}}{\|\mathbf{b}\|} = \frac{<\mathbf{a},\mathbf{b}>}{\|\mathbf{b}\|^2} \mathbf{b} = \frac{<\mathbf{a},\mathbf{b}>}{<\mathbf{b},\mathbf{b}>} \mathbf{b}$

Finally:

> $\mathbf{a}_{b\perp} = \mathbf{a} - \mathbf{a}_b = \mathbf{a} - \frac{<\mathbf{a},\mathbf{b}>}{\|\mathbf{b}\|^2} \mathbf{b}$
> Also note that:
> $\mathbf{a}_{b\perp} = \mathbf{a} - \frac{<\mathbf{a},\mathbf{b}>}{<\mathbf{b},\mathbf{b}>} \mathbf{b}$

If $\mathbf{b}$ was already unit, things would have been easier:

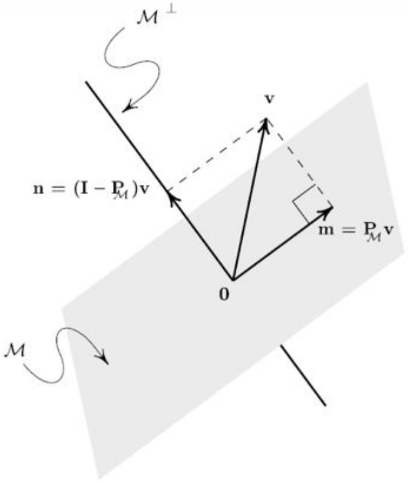
> $\mathbf{a}_b =< \mathbf{a}, \mathbf{b} > \mathbf{b}$
> $\mathbf{a}_{b\perp} = \mathbf{a} - < \mathbf{a}, \mathbf{b} > \mathbf{b}$

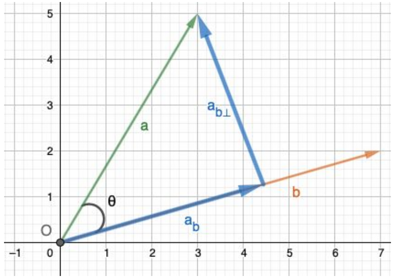
Orthogonal Projection

Orthogonal Projection

For $\mathbf{v} \in \mathcal{V}$, let $\mathbf{v} = \mathbf{m} + \mathbf{n}$, where $\mathbf{m} \in \mathcal{M}$ and $\mathbf{n} \in \mathcal{M}^{\perp}$.

- $\mathbf{m}$ is called the *orthogonal projection* of $\mathbf{v}$ onto $\mathcal{M}$.
- The projector $\mathbf{P}_{\mathcal{M}}$ onto $\mathcal{M}$ along $\mathcal{M}^{\perp}$ is called the *orthogonal projector* onto $\mathcal{M}$.
- $\mathbf{P}_{\mathcal{M}}$ is the unique linear operator such that $\mathbf{P}_{\mathcal{M}}\mathbf{v} = \mathbf{m}$





What if you project $\mathbf{m}$ again with $\mathbf{P}_M$?

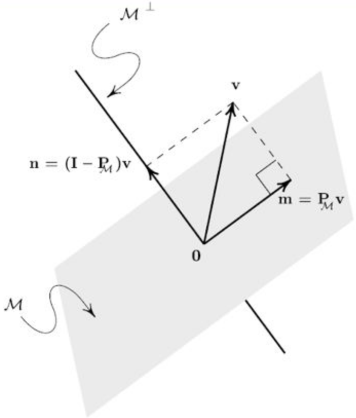
Again and again and ...

Orthogonal Projection

Orthogonal Projection

For $\mathbf{v} \in \mathcal{V}$, let $\mathbf{v} = \mathbf{m} + \mathbf{n}$, where $\mathbf{m} \in \mathcal{M}$ and $\mathbf{n} \in \mathcal{M}^{\perp}$.

- $\mathbf{m}$ is called the *orthogonal projection* of $\mathbf{v}$ onto $\mathcal{M}$.
- The projector $\mathbf{P}_{\mathcal{M}}$ onto $\mathcal{M}$ along $\mathcal{M}^{\perp}$ is called the *orthogonal projector* onto $\mathcal{M}$.
- $\mathbf{P}_{\mathcal{M}}$ is the unique linear operator such that $\mathbf{P}_{\mathcal{M}}\mathbf{v} = \mathbf{m}$



Finding the projection matrix $\mathbf{P}_M$

Recall from previous slides (with a bit of name swapping) that a vector $\mathbf{v}$ can be projected on another vector $\mathbf{a_i}$ as:

$$\mathbf{v}_{a_i} = \frac{<\mathbf{v},\mathbf{a_i}>}{<\mathbf{a_i},\mathbf{a_i}>}\mathbf{a_i}$$

If the columns of matrix $\mathbf{A}$ are orthonormal and span the subspace $\mathbf{M}$, then the projection matrix $\mathbf{P}_M$ can be written as follows:

$$\mathbf{P}_M = \mathbf{A}\mathbf{A}^T$$

Without columns of $\mathbf{A}$ satisfying orthonomality condition, the projection matrix $\mathbf{P}_M$ can be written *in general* as follows:

$$\mathbf{P}_M = \mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T$$

Proof (of at least the special case) is left to you as an exercise or is it?

Note that:

If $\mathbf{v}$ is was already in span of $\mathbf{A}$, no information is lost, or in otherwords, $\mathbf{v}_{\mathbf{M}\perp} = \mathbf{0}$