Processing BigData (PB Class)

- Schedule
- How it works
- Project
- Materials

Goal of the project

- Deal with “reality” (big dirty data)
  - High dimensionall (big but not really BIG… much less REALLY BIG !)
  - Missing Data
  - Heterogeneous
  - Mostly unsupervised
  - Outliers … really ? What is an outlier ?
- The need for a model and to “do something with the data”!
  - Linear Models
  - Convexity
- “BIG DATA IS LOW RANK “
  - Motto of the project (and most of the course)

Theretical Class

PB Class - P= project+practical

Data matrix: Handling numerical and categorical attributes

•Exploratory data analysis and visualization

•Linear dimensionality reduction with PCA

•Distributed PCA for big data

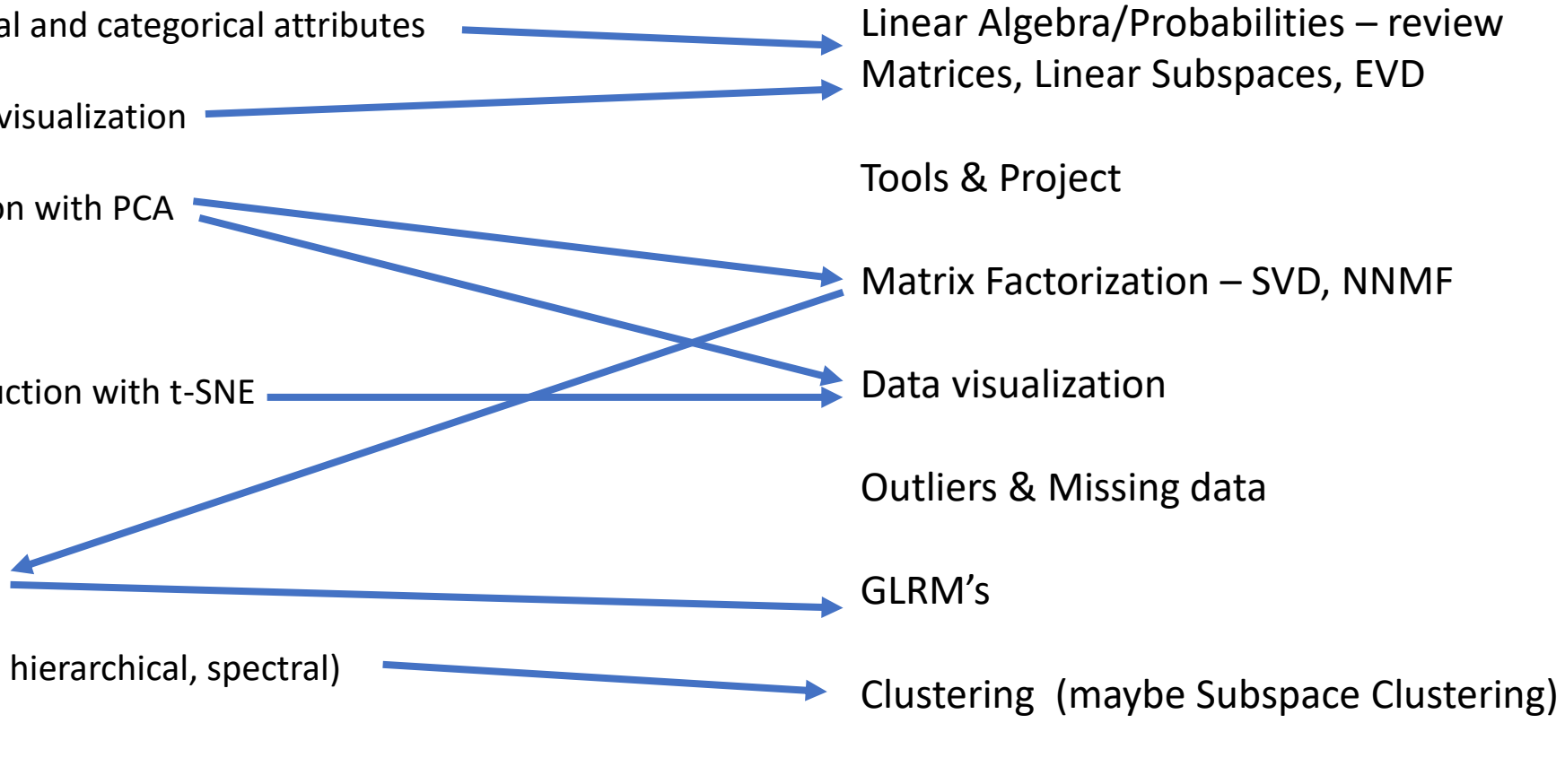
•Nonlinear dimensionality reduction with t-SNE

•Probabilistic PCA

•Generalized low-rank models

•Clustering (assignment-based, hierarchical, spectral)

•Large-scale optimization



Linear Algebra/Probabilities – review

Matrices, Linear Subspaces, EVD

Tools & Project

Matrix Factorization – SVD, NNMF

Data visualization

Outliers & Missing data

GLRM's

Clustering (maybe Subspace Clustering)

Theretical Class

PB Class - P= project+practical

Data matrix: Handling numerical and categorical a

•Exploratory data analysis and visualization

•Linear dimensionality reduction with PCA

•Distributed PCA for big data

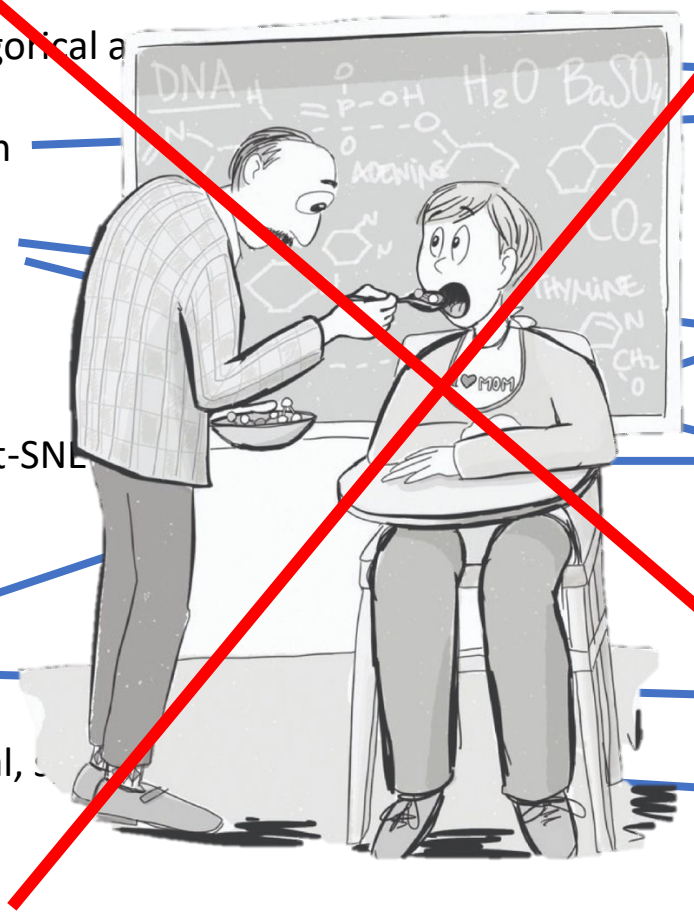
•Nonlinear dimensionality reduction with t-SNE

•Probabilistic PCA

•Generalized low-rank models

•Clustering (assignment-based, hierarchical, s

•Large-scale optimization



Linear Algebra/Probabilities – review

Matrices, Linear Subspaces, EVD

Tools & Project

Matrix Factorization – SVD, NNMF

Data visualization

Outliers & Missing data

GLRM's

Clustering (maybe Subspace Clustering)

„ No worries. You will get your degree. Just swallow_"

It's a new field, wider than my knowledge and we are all "discovering"

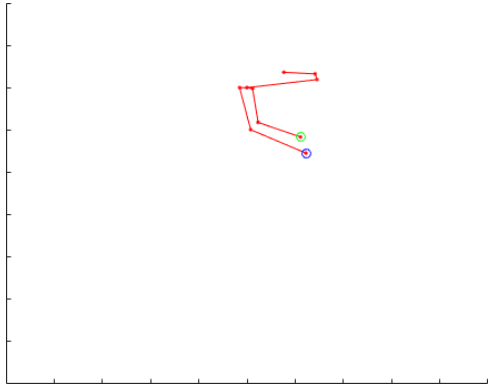
Project – Similar to last year’s

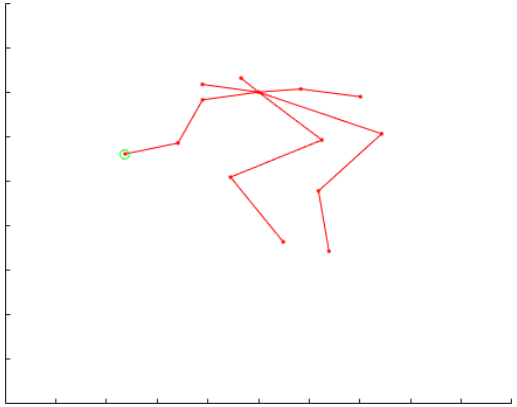
Introduction

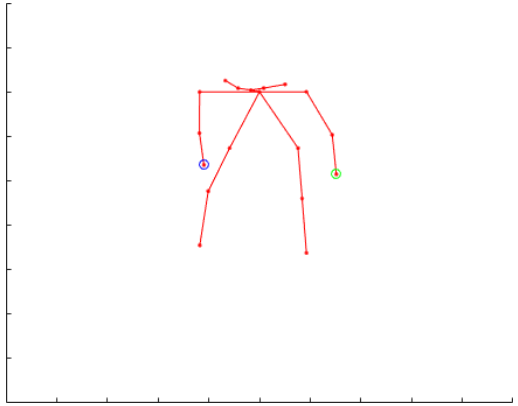
MPII Human Pose dataset is a state of the art benchmark for evaluation of articulated human pose estimation. The dataset includes around **25K images** containing over **40K people** with annotated body joints. The images were systematically collected using an established taxonomy of every day human activities. Overall the dataset covers **410 human activities** and each image is provided with an activity label. Each image was extracted from a YouTube video and provided with preceding and following un-annotated frames. In addition, for the test set we obtained richer annotations including body part occlusions and 3D torso and head orientations.

Following the best practices for the performance evaluation benchmarks in the literature we withhold the test annotations to prevent overfitting and tuning on the test set. We are working on an automatic evaluation server and performance analysis tools based on rich test set annotations.

Big issue: missing data and outliers !





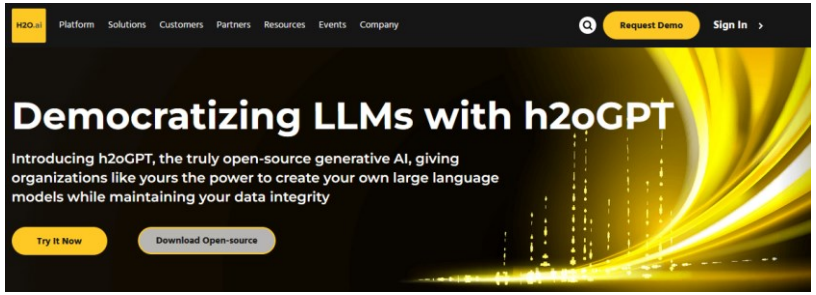


Two types of solutions

- Factorization with missing data

$$\arg\min_{X,Y} \|Z - XY^T\|_F^2,$$
$$s.t. rank(X) = rank(Y) = k$$

- GLRM's – involves a solver (H20)

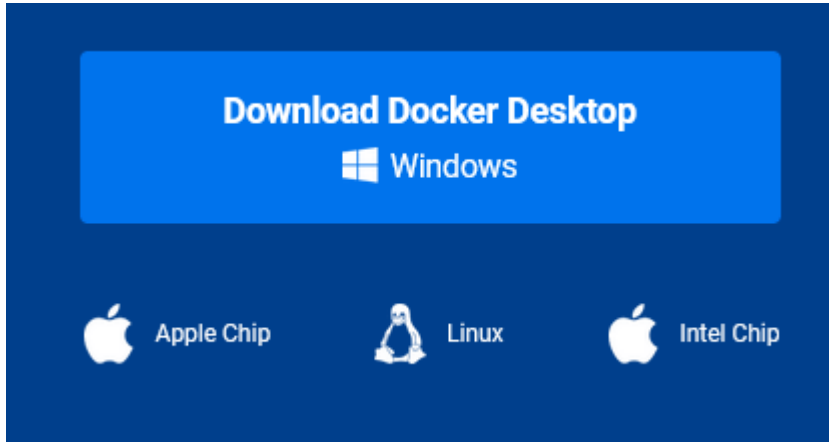


https://h2o.ai/

https://h2o-release.s3.amazonaws.com/h2o/rel-zz_kurka/4/docs-website/h2o-docs/downloading.html#install-in-python

https://h2o-release.s3.amazonaws.com/h2o/rel-zz_kurka/4/index.html

It’s voluntary but … we recommend you install a “container management” system



Quick demo next class

Unfortuneatly...time for

☺

Data Matrix

(for some reason we give you shots of linear algebra ... it is a matrix !)

$$\mathbf{D} = \begin{pmatrix} & X_1 & X_2 & \cdots & X_d \\ \mathbf{x}_1 & x_{11} & x_{12} & \cdots & x_{1d} \\ \mathbf{x}_2 & x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{x}_n & x_{n1} & x_{n2} & \cdots & x_{nd} \end{pmatrix}$$

	Sepal length X_1	Sepal width X_2	Petal length X_3	Petal width X_4	Class X_5
$\mathbf{x}_1$	5.9	3.0	4.2	1.5	Iris-versicolor
$\mathbf{x}_2$	6.9	3.1	4.9	1.5	Iris-versicolor
$\mathbf{x}_3$	6.6	2.9	4.6	1.3	Iris-versicolor
$\mathbf{x}_4$	4.6	3.2	1.4	0.2	Iris-setosa
$\mathbf{x}_5$	6.0	2.2	4.0	1.0	Iris-versicolor
$\mathbf{x}_6$	4.7	3.2	1.3	0.2	Iris-setosa
$\mathbf{x}_7$	6.5	3.0	5.8	2.2	Iris-virginica
$\mathbf{x}_8$	5.8	2.7	5.1	1.9	Iris-virginica
⋮	⋮	⋮	⋮	⋮	⋮
$\mathbf{x}_{149}$	7.7	3.8	6.7	2.2	Iris-virginica
$\mathbf{x}_{150}$	5.1	3.4	1.5	0.2	Iris-setosa

Topics – next 3 classes (among other things)

- Data matrix, attributes, vector space, basis, subspace.
- Distance and angles . Centering data.
- Linear indepence, rank, range and null space.
- Orthogonal projections and orthogonal projectors..
- Probability, Mean, Sample Mean, Dispersion, Covariance, Sample Covariance.
- Multivariate Normal Distribution. Correlation, EVD and how to decorrelate a gaussian distrib…etc
- Chapters 1,2,6 Zeki.

Some data to play with

- Until I get Fenix

- http://drive.sipg.tecnico.ulisboa.pt/s/pbd

Vectors and vector spaces

Let $\mathbf{a}, \mathbf{b} \in \mathbb{R}^m$ be two m-dimensional vectors given as

$$\mathbf{a} = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}$$

Dot Product

The *dot product* between $\mathbf{a}$ and $\mathbf{b}$ is defined as the scalar value

$$\mathbf{a}^T\mathbf{b} = \begin{pmatrix} a_1 & a_2 & \cdots & a_m \end{pmatrix} \times \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix} = a_1b_1 + a_2b_2 + \cdots + a_mb_m = \sum_{i=1}^{m} a_i b_i$$

Length

The *Euclidean norm* or *length* of a vector $\mathbf{a} \in \mathbb{R}^m$ is defined as

$$\|\mathbf{a}\| = \sqrt{\mathbf{a}^T\mathbf{a}} = \sqrt{a_1^2 + a_2^2 + \cdots + a_m^2} = \sqrt{\sum_{i=1}^{m} a_i^2}$$

The *unit vector* in the direction of $\mathbf{a}$ is given as

$$\mathbf{u} = \frac{\mathbf{a}}{\|\mathbf{a}\|} = \left(\frac{1}{\|\mathbf{a}\|}\right)\mathbf{a}$$

L_p*-norm*, defined as

$$\|\mathbf{a}\|_p = \left(|a_1|^p + |a_2|^p + \cdots + |a_m|^p\right)^{\frac{1}{p}} = \left(\sum_{i=1}^{m} |a_i|^p\right)^{\frac{1}{p}}$$

Distance

From the Euclidean norm we can define the *Euclidean distance* between $\mathbf{a}$ and $\mathbf{b}$, as follows

$$\|\mathbf{a}-\mathbf{b}\| = \sqrt{(\mathbf{a}-\mathbf{b})^T(\mathbf{a}-\mathbf{b})} = \sqrt{\sum_{i=1}^{m}(a_i-b_i)^2} \tag{1.4}$$

Angle

The cosine of the smallest angle between vectors $\mathbf{a}$ and $\mathbf{b}$, also called the *cosine similarity*, is given as

$$\cos\theta = \frac{\mathbf{a}^T\mathbf{b}}{\|\mathbf{a}\|\|\mathbf{b}\|} = \left(\frac{\mathbf{a}}{\|\mathbf{a}\|}\right)^T\left(\frac{\mathbf{b}}{\|\mathbf{b}\|}\right) \tag{1.6}$$

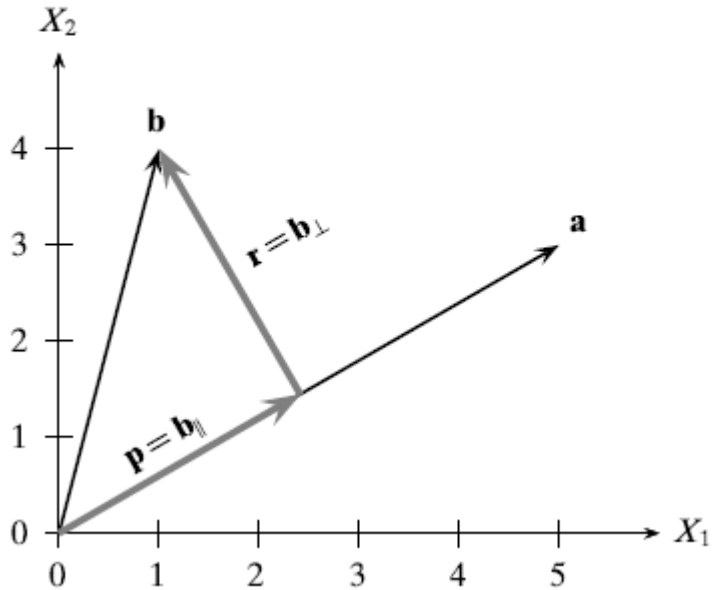
Thus, the cosine of the angle between $\mathbf{a}$ and $\mathbf{b}$ is given as the dot product of the unit vectors $\frac{\mathbf{a}}{\|\mathbf{a}\|}$ and $\frac{\mathbf{b}}{\|\mathbf{b}\|}$.

The *Cauchy–Schwartz* inequality states that for any vectors $\mathbf{a}$ and $\mathbf{b}$ in $\mathbb{R}^m$

$$|\mathbf{a}^T\mathbf{b}| \leq \|\mathbf{a}\| \cdot \|\mathbf{b}\|$$

$$\mathbf{b} = \mathbf{b}_{\parallel} + \mathbf{b}_{\perp} = \mathbf{p} + \mathbf{r}$$

$$\mathbf{p}^T\mathbf{r} = (c\mathbf{a})^T(\mathbf{b}-c\mathbf{a}) = c\mathbf{a}^T\mathbf{b} - c^2\mathbf{a}^T\mathbf{a} = 0$$



which implies that

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

Therefore, the projection of $\mathbf{b}$ on $\mathbf{a}$ is given as

$$\mathbf{p} = c\mathbf{a} = \left(\frac{\mathbf{a}^T\mathbf{b}}{\mathbf{a}^T\mathbf{a}}\right)\mathbf{a} \tag{1.11}$$

The scalar offset c along $\mathbf{a}$ is also called the *scalar projection* of $\mathbf{b}$ on $\mathbf{a}$, denoted as

$$\text{proj}_{\mathbf{a}}(\mathbf{b}) = \left(\frac{\mathbf{b}^T\mathbf{a}}{\mathbf{a}^T\mathbf{a}}\right) \tag{1.12}$$

Therefore, the projection of $\mathbf{b}$ on $\mathbf{a}$ can also be written as

$$\mathbf{p} = \text{proj}_{\mathbf{a}}(\mathbf{b}) \cdot \mathbf{a}$$

I like columns better ☺

2.1.3 Linear Combinations

We can add and scale vectors in the same equation.

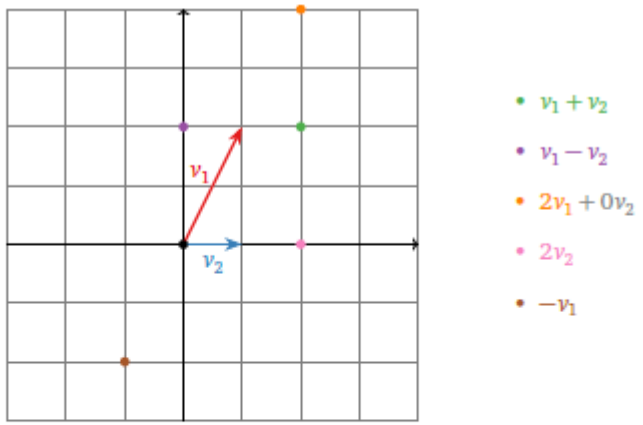
Definition. Let $c_1, c_2, \ldots, c_k$ be scalars, and let $v_1, v_2, \ldots, v_k$ be vectors in $\mathbf{R}^n$. The vector in $\mathbf{R}^n$

$$c_1 v_1 + c_2 v_2 + \cdots + c_k v_k$$

is called a **linear combination** of the vectors $v_1, v_2, \ldots, v_k$, with **weights** or **coefficients** $c_1, c_2, \ldots, c_k$.

Geometrically, a linear combination is obtained by stretching / shrinking the vectors $v_1, v_2, \ldots, v_k$ according to the coefficients, then adding them together using the parallelogram law.

Example. Let $v_1 = \binom{1}{2}$ and $v_2 = \binom{1}{0}$. Here are some linear combinations of v_1 and v_2, drawn as points.



The locations of these points are found using the parallelogram law for vector addition. Any vector on the plane is a linear combination of v_1 and v_2, with suitable coefficients.

Essential Definition. Let $v_1, v_2, \ldots, v_k$ be vectors in $\mathbf{R}^n$. The **span** of $v_1, v_2, \ldots, v_k$ is the collection of all linear combinations of $v_1, v_2, \ldots, v_k$, and is denoted $\text{Span}\{v_1, v_2, \ldots, v_k\}$. In symbols:

$$\text{Span}\{v_1, v_2, \ldots, v_k\} = \left\{x_1 v_1 + x_2 v_2 + \cdots + x_k v_k \mid x_1, x_2, \ldots, x_k \text{ in } \mathbf{R}\right\}$$

We also say that $\text{Span}\{v_1, v_2, \ldots, v_k\}$ is the subset **spanned by** or **generated by** the vectors $v_1, v_2, \ldots, v_k$.

In “Data” notation !

Row and Column Space

There are several interesting vector spaces associated with the data matrix $\mathbf{D}$, two of which are the column space and row space of $\mathbf{D}$. The *column space* of $\mathbf{D}$, denoted $col(\mathbf{D})$, is the set of all linear combinations of the d attributes $X_j \in \mathbb{R}^n$, that is,

$$col(\mathbf{D}) = span(X_1, X_2, \ldots, X_d)$$

By definition $col(\mathbf{D})$ is a subspace of $\mathbb{R}^n$. The *row space* of $\mathbf{D}$, denoted $row(\mathbf{D})$, is the set of all linear combinations of the n points $\mathbf{x}_i \in \mathbb{R}^d$, that is,

$$row(\mathbf{D}) = span(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$$

By definition $row(\mathbf{D})$ is a subspace of $\mathbb{R}^d$. Note also that the row space of $\mathbf{D}$ is the column space of $\mathbf{D}^T$:

$$row(\mathbf{D}) = col(\mathbf{D}^T)$$

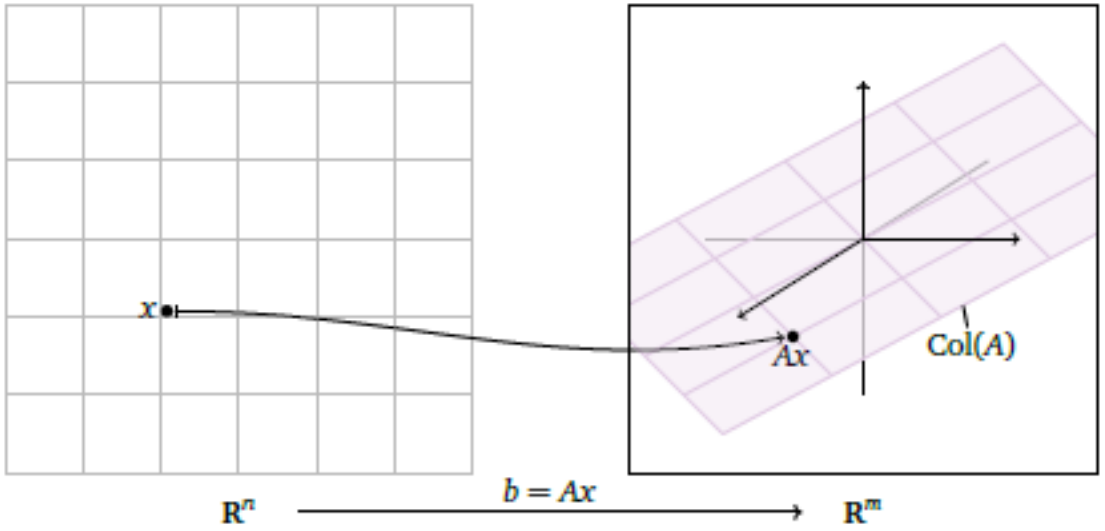
Linear Independance

By the way … What is a matrix ?

By the way … What is a matrix ?

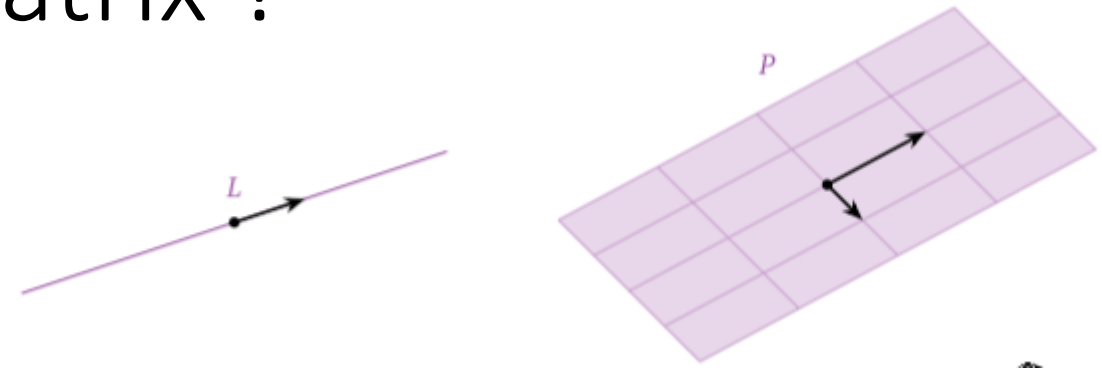
- The independent variable (the input) is x, which is a vector in $\mathbf{R}^n$.
- The dependent variable (the output) is b, which is a vector in $\mathbf{R}^m$.

The set of all possible output vectors are the vectors b such that $Ax = b$ has some solution; this is the same as the column space of A by this note in Section 2.3.



By the way … What is a matrix ?

- Explore its structure

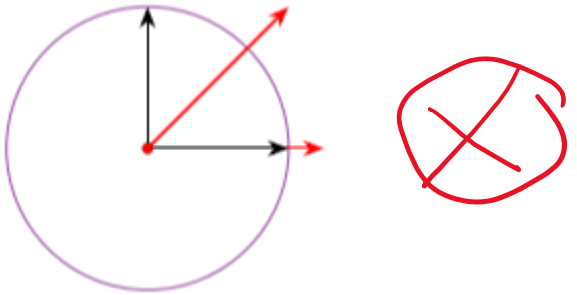


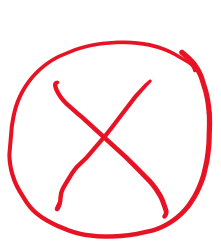
Definition. A **subspace** of $\mathbf{R}^n$ is a subset V of $\mathbf{R}^n$ satisfying:

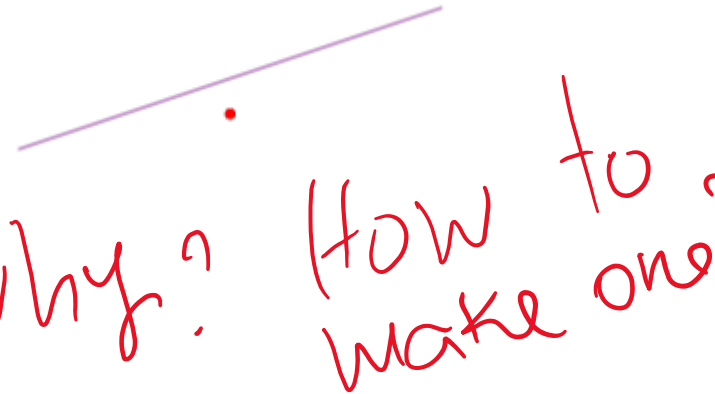
1. ***Non-emptiness:*** The zero vector is in V.
2. ***Closure under addition:*** If u and v are in V, then $u+v$ is also in V.
3. ***Closure under scalar multiplication:*** If v is in V and c is in $\mathbf{R}$, then cv is also in V.

As a consequence of these properties, we see:

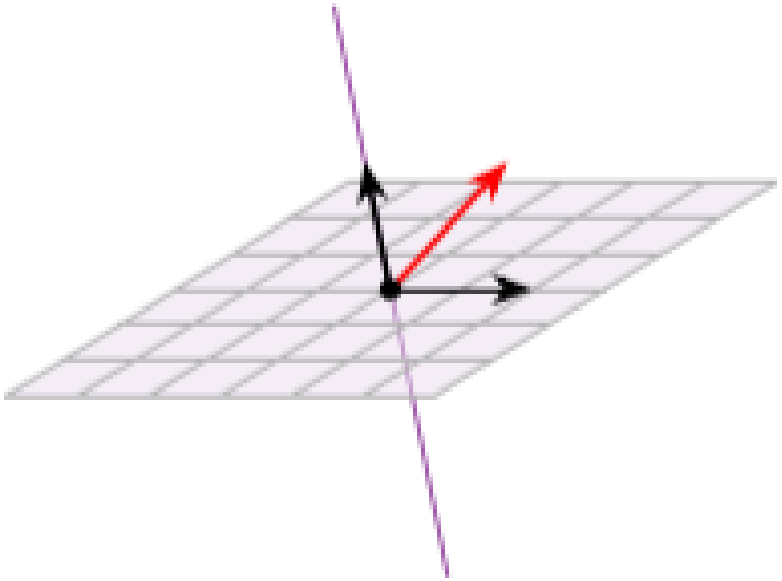
- If v is a vector in V, then all scalar multiples of v are in V by the third property. In other words the line through any nonzero vector in V is also contained in V.
- If u, v are vectors in V and c, d are scalars, then cu, dv are also in V by the third property, so $cu+dv$ is in V by the second property. Therefore, all of $\text{Span}\{u, v\}$ is contained in V
- Similarly, if $v_1, v_2, \ldots, v_n$ are all in V, then $\text{Span}\{v_1, v_2, \ldots, v_n\}$ is contained in V. In other words, *a subspace contains the span of any vectors in it.*

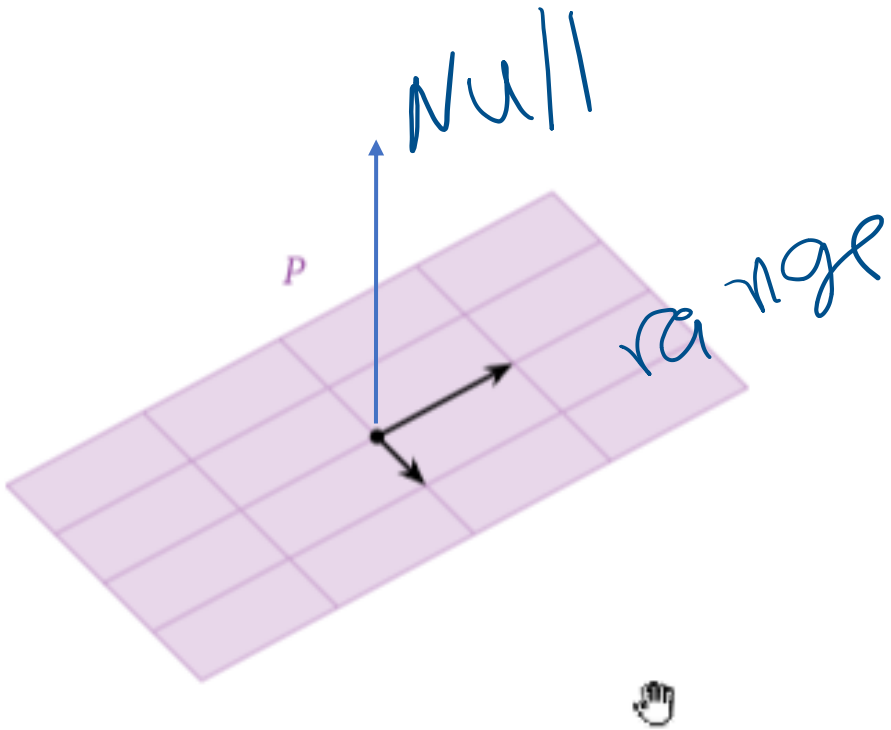






Subspace clustering ???





Row and Column Space

There are several interesting vector spaces associated with the data matrix $\mathbf{D}$, two of which are the column space and row space of $\mathbf{D}$. The *column space* of $\mathbf{D}$, denoted $col(\mathbf{D})$, is the set of all linear combinations of the d attributes $X_j \in \mathbb{R}^n$, that is,

$$col(\mathbf{D}) = span(X_1, X_2, \ldots, X_d)$$

By definition $col(\mathbf{D})$ is a subspace of $\mathbb{R}^n$. The *row space* of $\mathbf{D}$, denoted $row(\mathbf{D})$, is the set of all linear combinations of the n points $\mathbf{x}_i \in \mathbb{R}^d$, that is,

$$row(\mathbf{D}) = span(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$$

By definition $row(\mathbf{D})$ is a subspace of $\mathbb{R}^d$. Note also that the row space of $\mathbf{D}$ is the column space of $\mathbf{D}^T$:

$$row(\mathbf{D}) = col(\mathbf{D}^T)$$

Linear Independence

$D = [B][C]$

$Dx = 0$

Subspaces

Essential Definition. Let V be a subspace of $\mathbf{R}^n$. A **basis** of V is a set of vectors $\{v_1, v_2, \ldots, v_m\}$ in V such that:

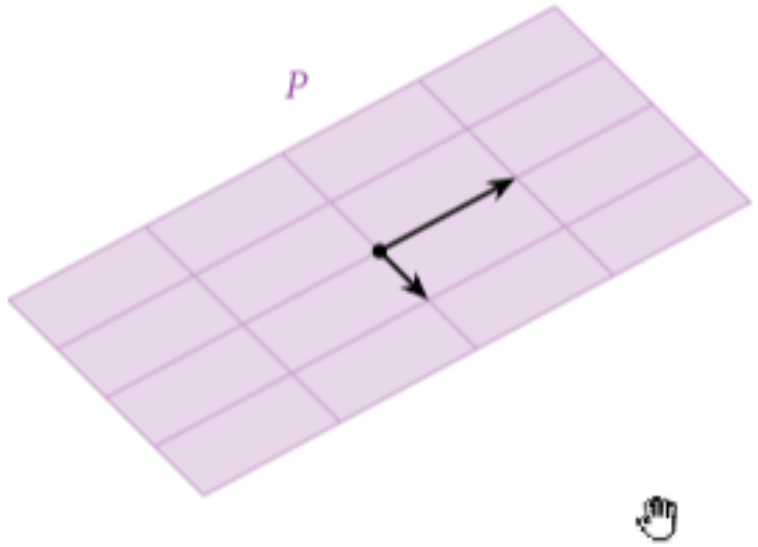
1. $V = \text{Span}\{v_1, v_2, \ldots, v_m\}$, and
2. the set $\{v_1, v_2, \ldots, v_m\}$ is linearly independent.

Recall that a set of vectors is *linearly independent* if and only if, when you remove any vector from the set, the span shrinks (Theorem 2.5.10). In other words, if $\{v_1, v_2, \ldots, v_m\}$ is a basis of a subspace V, then no proper subset of $\{v_1, v_2, \ldots, v_m\}$ will span V: it is a *minimal* spanning set. Any subspace admits a basis by this theorem in Section 2.6.

A nonzero subspace has *infinitely many* different bases, but they all contain the same number of vectors.

Subspaces:

- Rank
- Base
- Range
- Null
- Orthogonal projectors



- How to get a base ? With noisy data ?