Streams Processing

Principal Component Analysis: Linear Algebra

Dimensionality reduction

- Linear
  - Principal Component Analysis
    - Matrix sketching
  - Compressed sensing
- Non-linear
  - Kernel PCA
  - Isometric mapping

PCA

Linear Algebra

Change of basis

Statistics

Probability density estimation

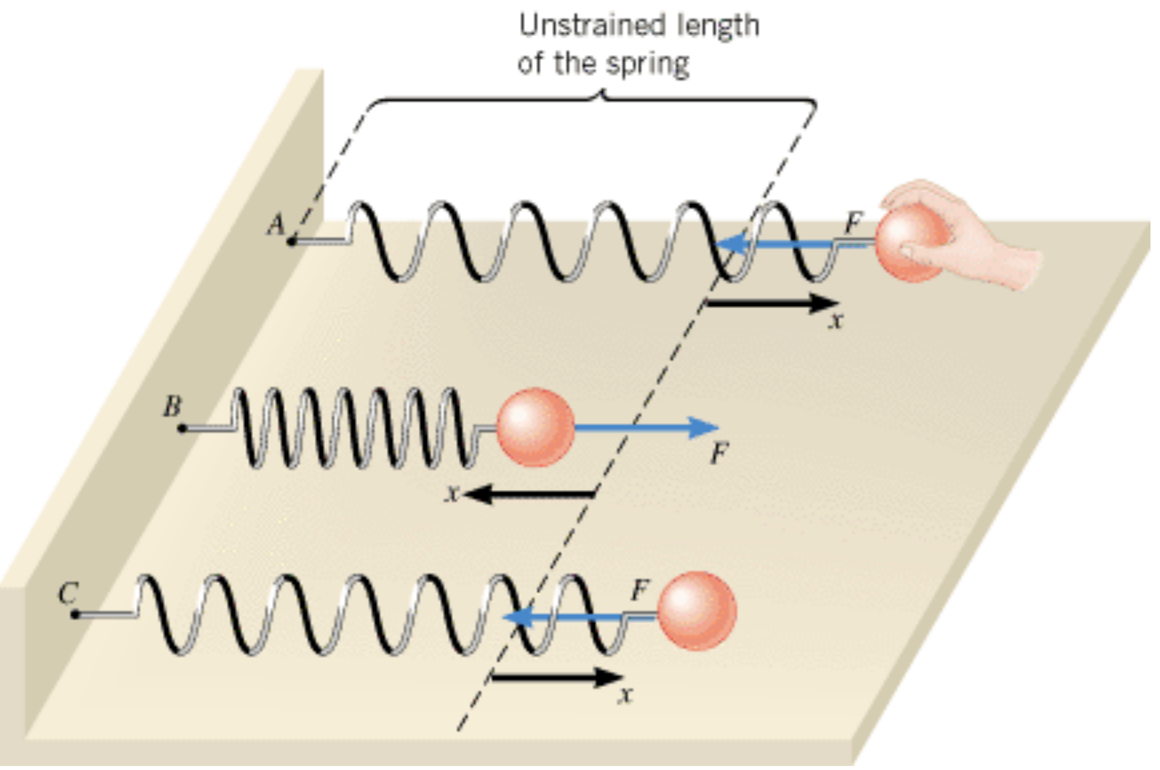
Optimization

Minimization of the reconstruction error considering a low-rank model

Example

Ideal spring mass system

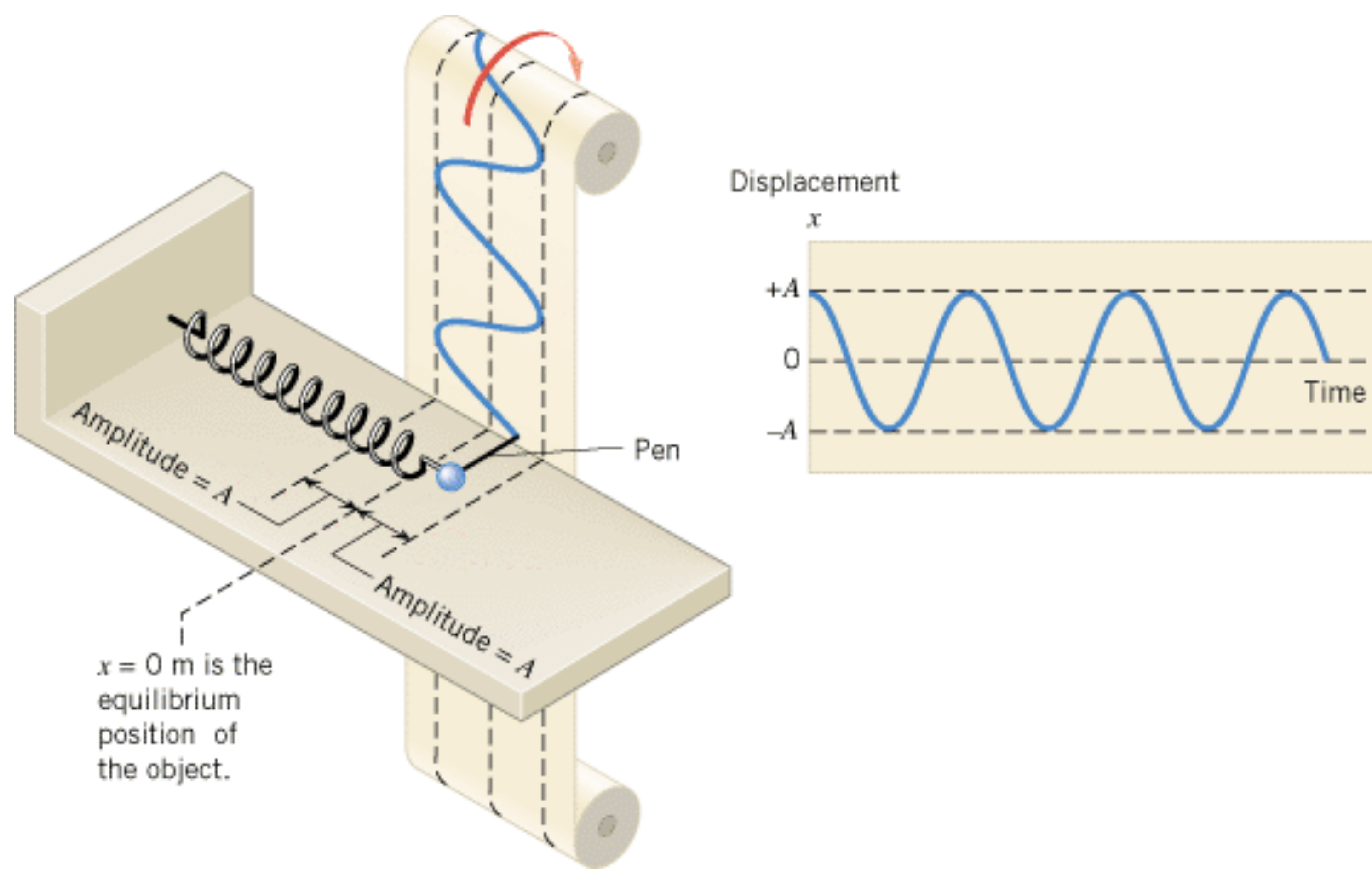
$$F = -kx$$



http://demo.webassign.net/ebooks/cj6demo/pc/c10/read/main/c10x10_1.htm

Example

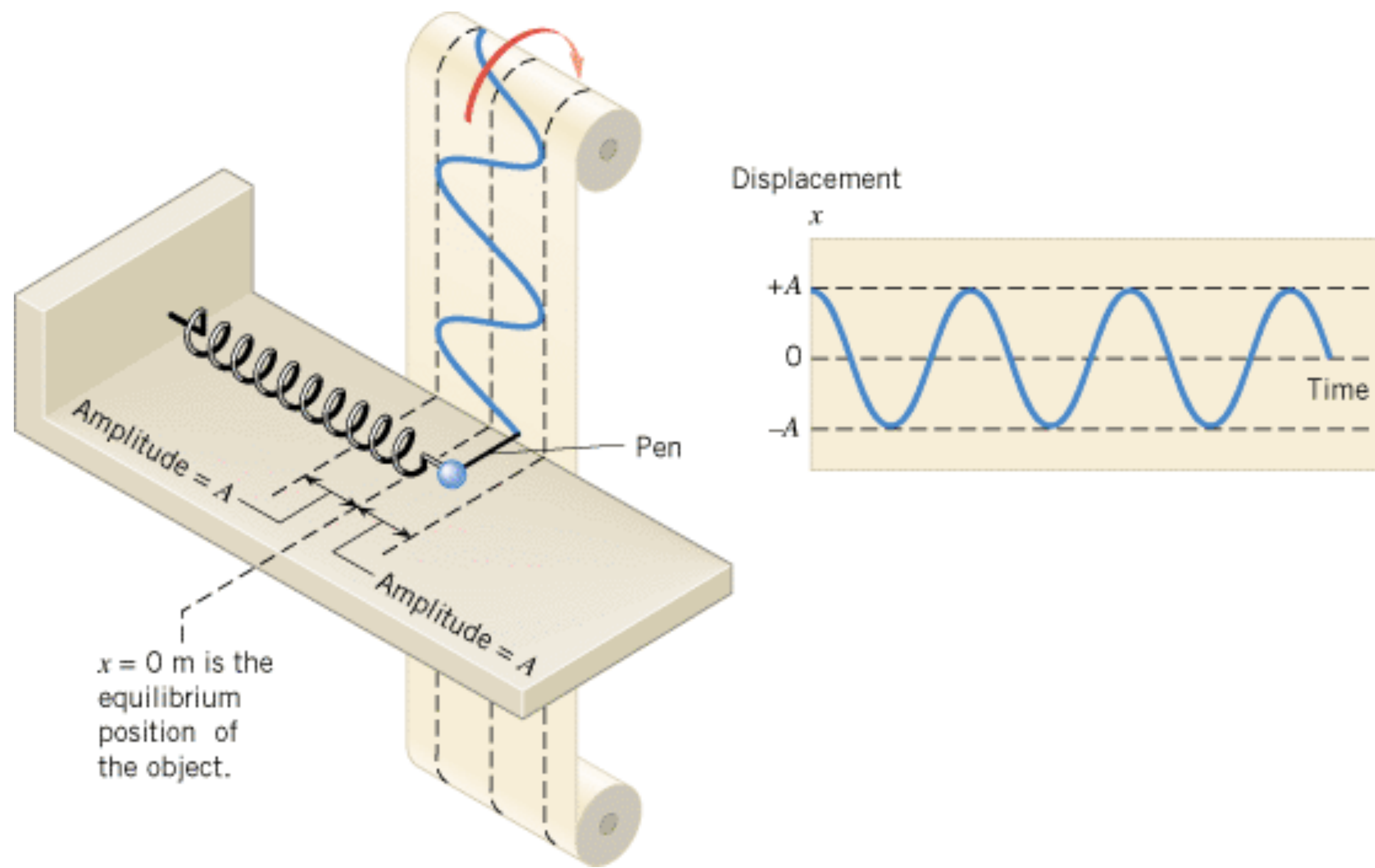
Ideal spring mass system



http://demo.webassign.net/ebooks/cj6demo/pc/c10/read/main/c10x10_1.htm

Example

The motion equation is an explicit function of time, in one variable, x

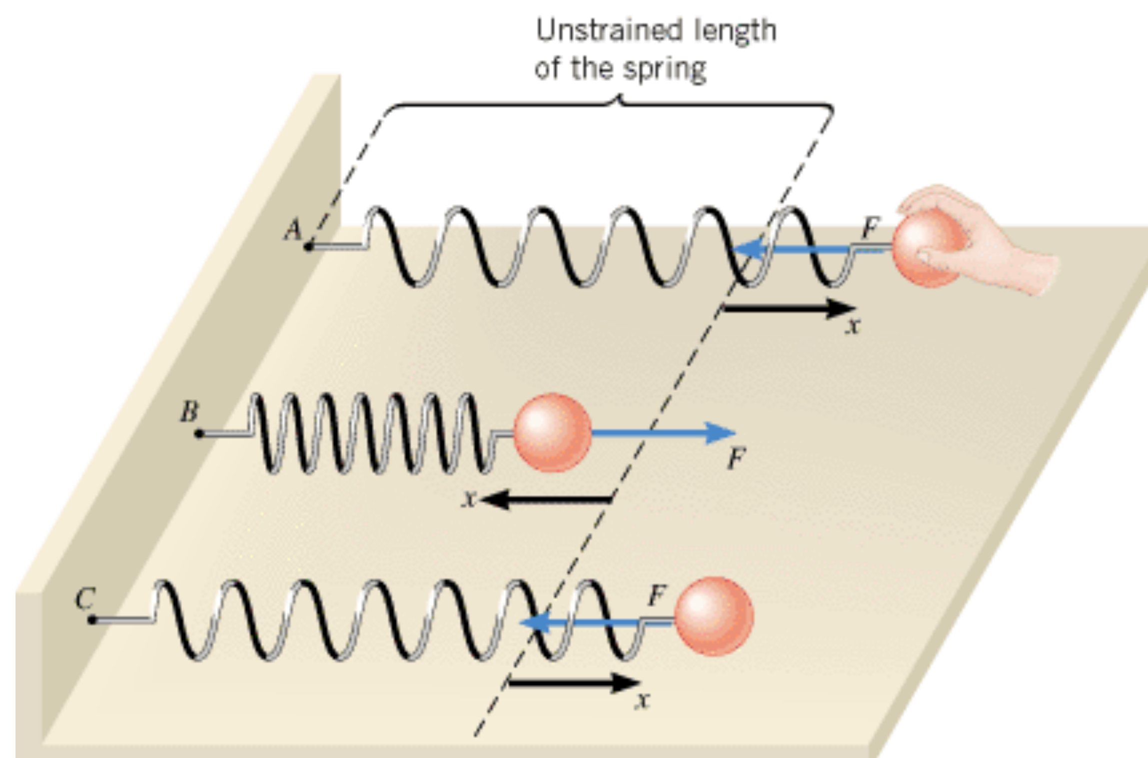


http://demo.webassign.net/ebooks/cj6demo/pc/c10/read/main/c10x10_1.htm

Example

Assume we are ignorant experimenters

We know nothing about the motion laws of an ideal spring and mass system



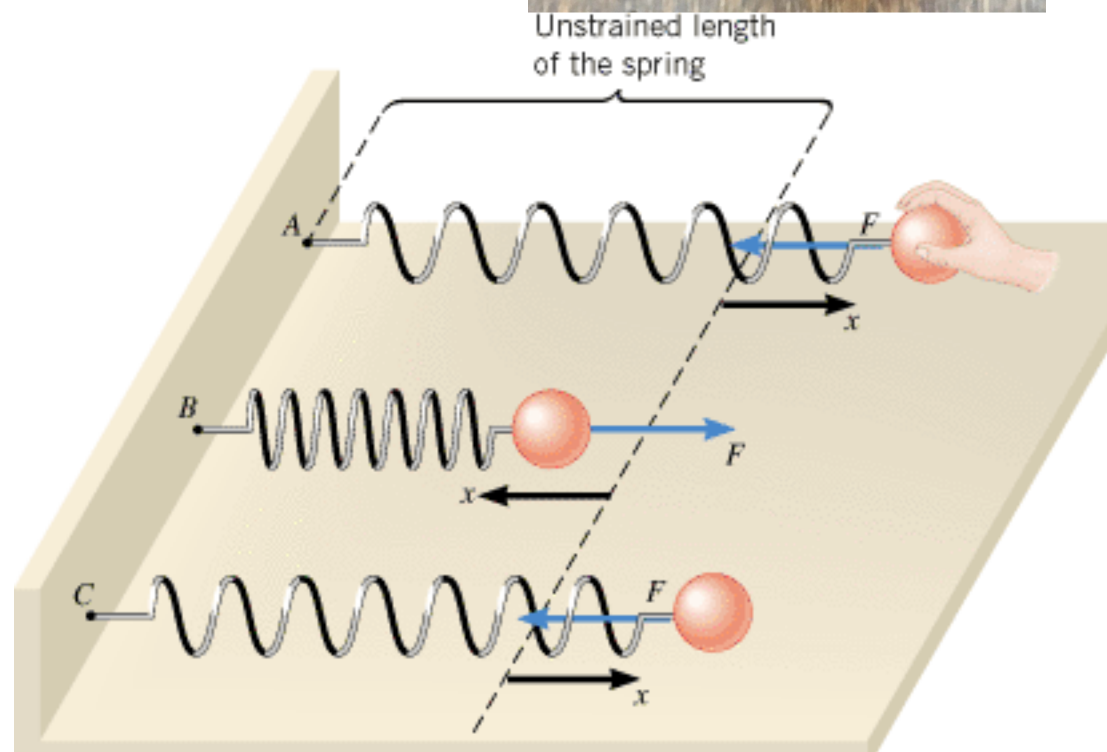
But we want to study the motion of the ball, learning from data

Example

We are ignorant experimenters: we don't know the x,y,z axis

So, we install randomly 3 cameras to measure the ball's position

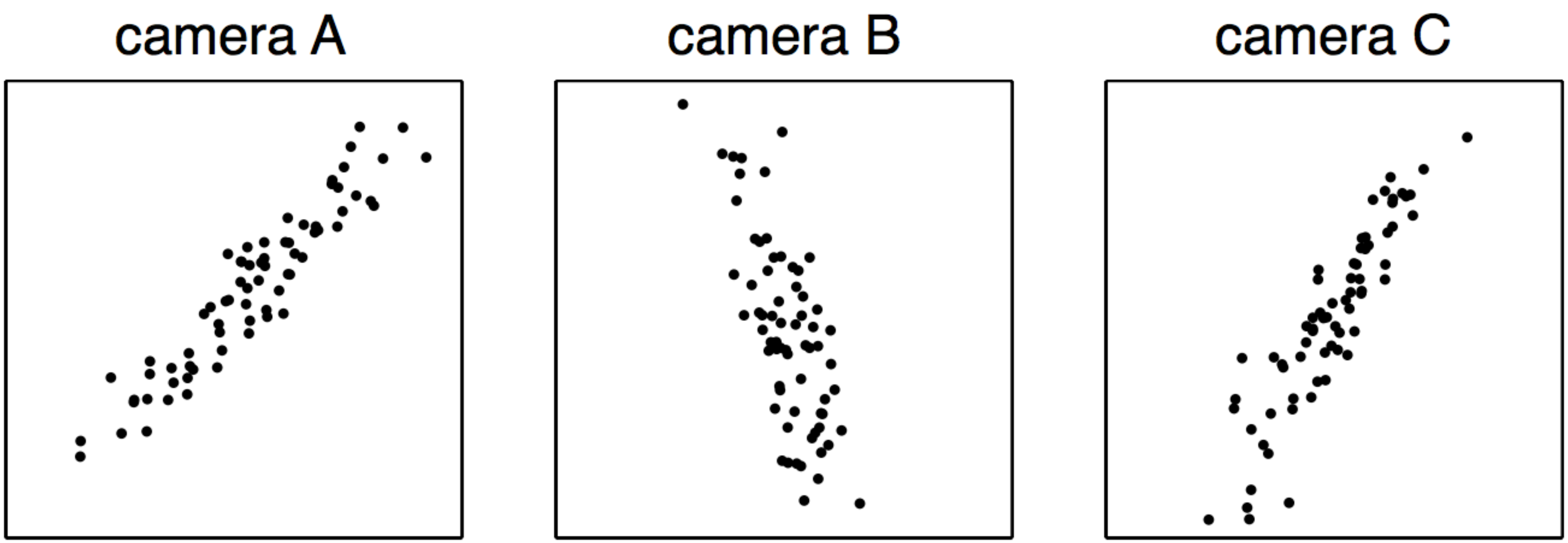
At 120Hz, each camera records an image from which we extract a 2D position of the ball (a projection)



https://goo.gl/EMQpzy

Example

We record with the cameras for several minutes…



From the example to processing Big Data

In general, we don't know which data are redundant or which features best reflect the underlying process we want to uncover

Also, real world contaminates our dataset with noise

This example illustrates what people working with data face everyday

Our dataset

$$\mathbf{x}(t) = \begin{bmatrix} x_A(t) \\ y_A(t) \\ x_B(t) \\ y_B(t) \\ x_C(t) \\ y_C(t) \end{bmatrix}$$

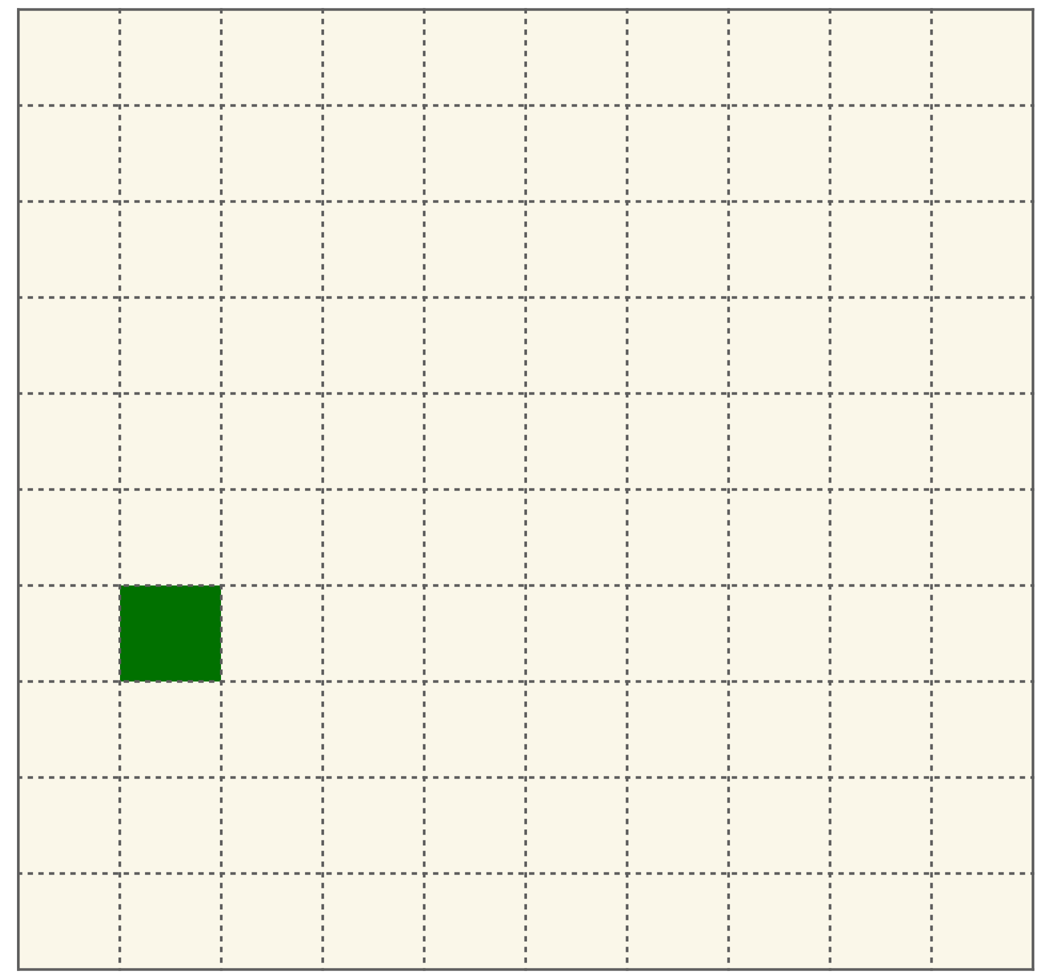
We record 10mn

$$10 \times 60 \times 120 = 72.000$$

$$X = \begin{bmatrix} \mathbf{x}(1) & \mathbf{x}(2) & \cdots & \mathbf{x}(n) \end{bmatrix}$$

Choice of basis and data collection

Point (2,4) = 4 units up and 2 units to the left



The basis we use to represent our data is a consequence of the method we used to collect our data

Choice of basis and data collection

Standard basis for a Euclidean space of dim *m*: the set of unit vectors pointing in the direction of the Cartesian coordinate system

$$\mathbf{B} = \begin{bmatrix} \mathbf{b_1} \\ \mathbf{b_2} \\ \vdots \\ \mathbf{b_m} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} = \mathbf{I}$$

Another **possible basis**, for Euclidean 2D:

$$\left\{ \left(\frac{\sqrt{3}}{2}, \frac{1}{2} \right), \left(\frac{1}{2}, \frac{-\sqrt{3}}{2} \right) \right\}$$

A linear subspace basis is a **linear combination** of other basis: $PX = Y$

Key assumption: linearity

Is there another **linear combination** of the current basis that better expresses our dataset?

PCA powerful assumption: Linearity, restricting the set of potential bases…

LA refresher: change of basis

$$PX = Y$$

P transforms *X* into *Y*

P is a rotation and a stretch (if not orthonormal)

The rows of *P* are the set of new basis vectors for the columns of *X:*

$$\mathbf{y}_i = \begin{bmatrix} \mathbf{p}_1 \mathbf{x}_i \\ \vdots \\ \mathbf{p}_m \mathbf{x}_i \end{bmatrix}$$ the dot products of the rows of *P* and the ith column of *X*

Questions

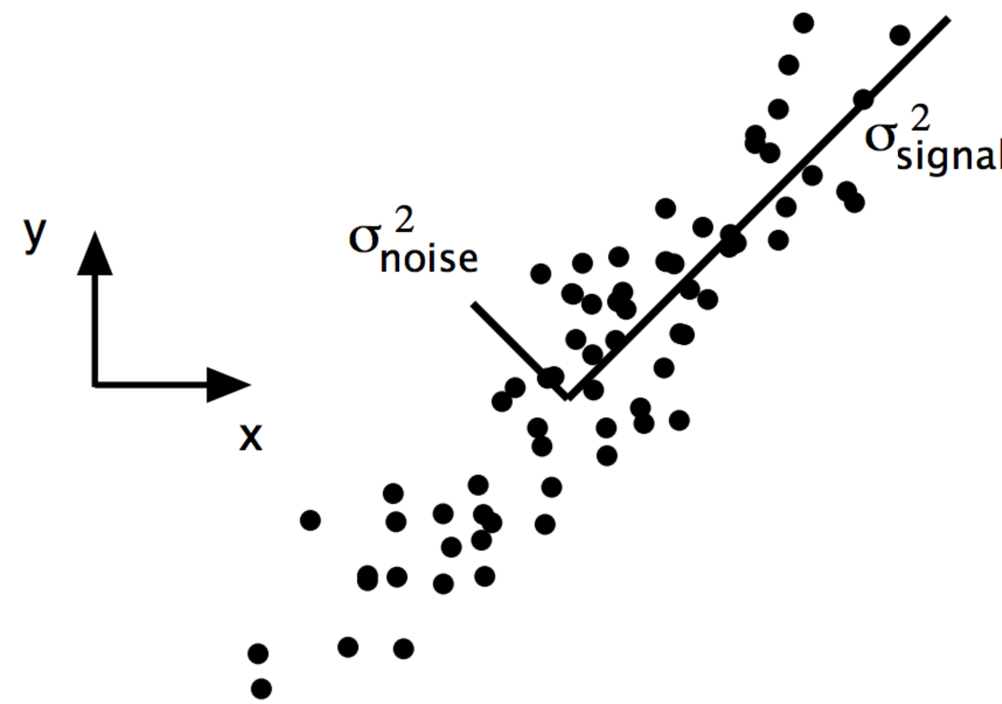
What is the best way to re-express X?

What is a good basis P?

But first, what features would we like to see in our Y?

SNR

$$SNR = \frac{\sigma^2_{signal}}{\sigma^2_{noise}}$$



High SNR: precise measurement

Low SNR: noisy data

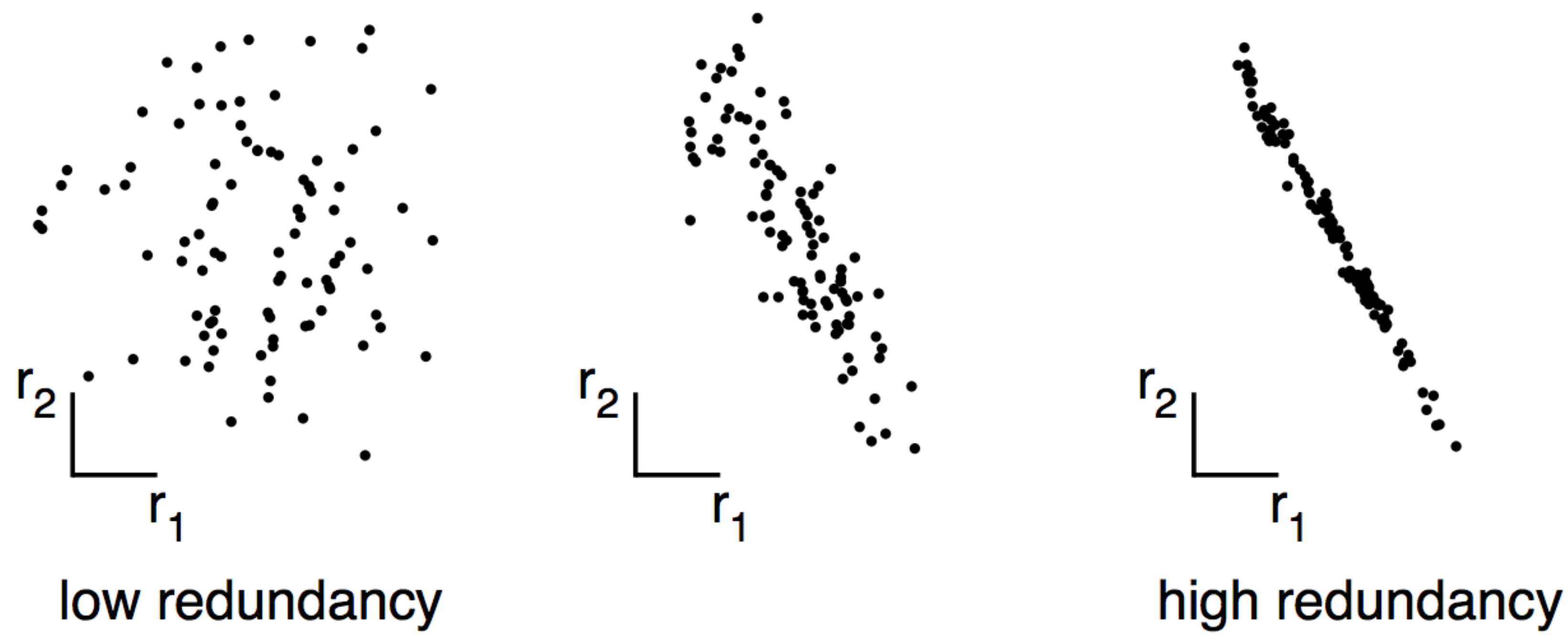
We assume interesting dynamics occur on the direction of larger variance (high SNR)

As the motion of the ball is in a straight line, we assume the perpendicular direction to the line of best fit corresponds to noise

Each camera records 2 variables. Are they really necessary?

Redundancy

In the spring example three sensors capture the **same information**



Dimensionality reduction: If we can compute one variable from the other we should keep the more concise representation.

(x_A, x_B) if A and B are close

$(x_A, \tilde{x}_A)$ using different units

Covariance matrix

X **with zero mean:**

$$C_X = \frac{1}{n} X X^T$$

Important properties:

Square symmetric matrix m x m

Diagonal contains the variances for each feature. Large variances are interesting structure

Off-diagonal contains covariance between different features. Large variances are redundancy

Covariance matrix

Goals of PCA:

Minimize redundancy, measured by the magnitude of the covariance;

Maximize the signal, measured by the variance.

Diagonalize the covariance matrix

Our ideal covariance matrix is diagonal. Our goals translate into finding a change of basis that returns a representation of the data with diagonal covariance

$$PX = Y$$

$$\begin{aligned} C_Y &= \frac{1}{n} Y Y^T \\ &= P C_X P^T \end{aligned}$$

The rows of the orthonormal P are the principal components

Solving PCA with EVD

EVD: Any symmetric matrix can be diagonalized $A = EDE^T$

D is diagonal and E's columns are the eigenvectors of A

We select the matrix P to be a matrix where each row is an eigenvector of C_X

$$P = E^T$$

With this selection and using the fact that the inverse of an orthogonal matrix is its transpose

$$C_Y = D$$

Solving PCA with SVD

$$X = U\Sigma V^T$$

Any arbitrary matrix can be written as the product of orthogonal, diagonal, orthogonal

V spans the column space of *X*, so the columns of V are the principal components of *X*

In julia, do

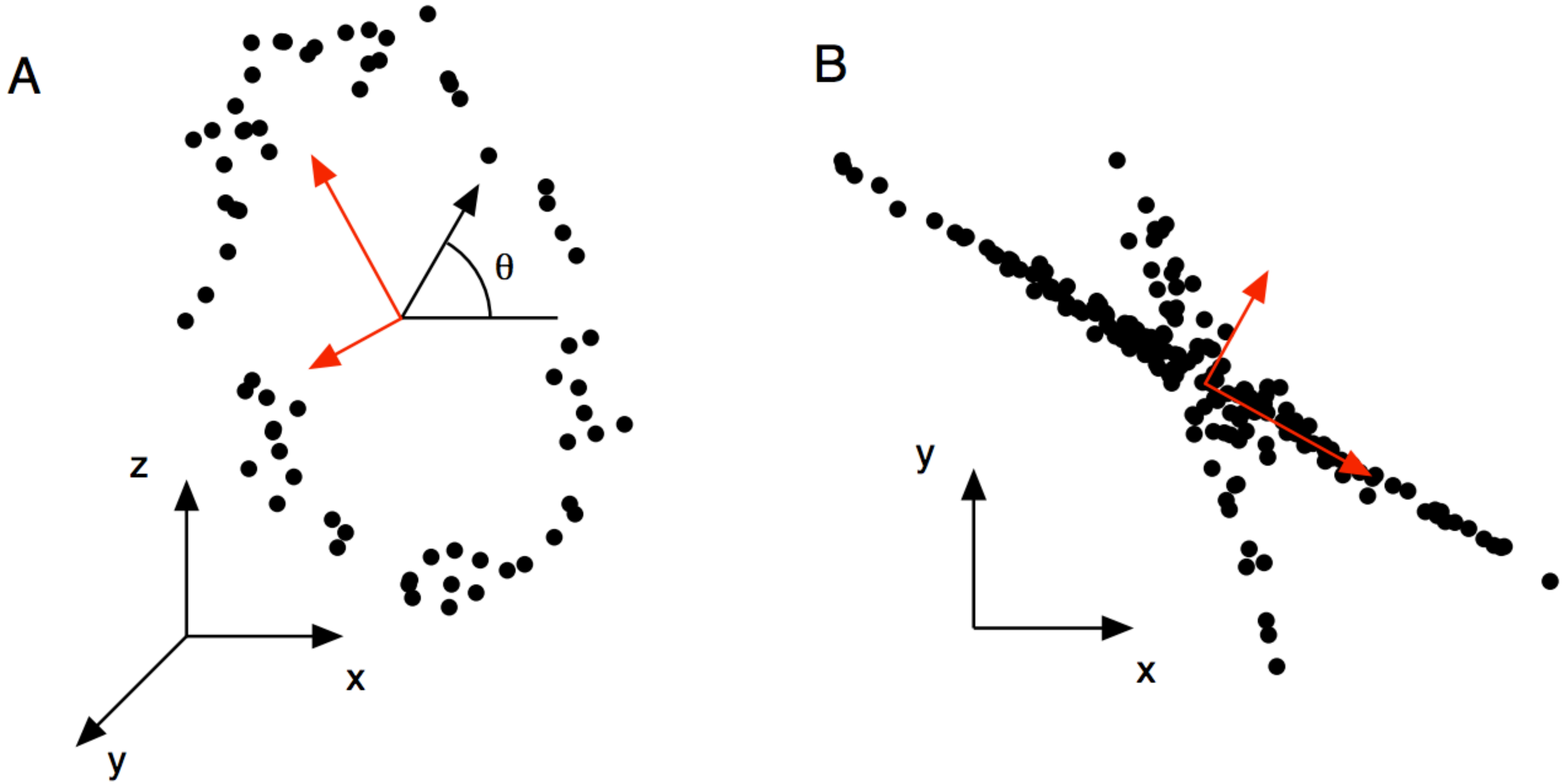
```
A = [1. 0. 0. 0. 2.; 0. 0. 3. 0. 0.; 0. 0. 0. 0. 0.; 0. 2. 0. 0. 0.]

U, S, V = svd(A)
```

Quick Summary of PCA

1. Organize data as an $m \times n$ matrix, where m is the number of measurement types and n is the number of samples.
2. Subtract off the mean for each measurement type.
3. Calculate the SVD or the eigenvectors of the covariance.

Limits



Assumptions

1. Linearity
2. Large variances have important structure
3. The principal components are orthogonal