Carnegie Mellon University
Heinz College

94-775 Unstructured Data Analytics

Lecture 4: Co-occurrence analysis (cont'd), visualizing high-dimensional data with PCA

Slides by George H. Chen

Administrivia

- Office hours have been posted on Canvas
- Your TA Johnna's OH are Mondays 3pm-4pm HBH 1111
- My OH are Tuesdays 12:30pm-1:30pm HBH 2216

Recap

To find specific person/company pairs that are interesting:

- Can rank using co-occurrence probability but doing so can sometimes be misleading
- Can alternatively rank using PMI
- Other score functions exist, such as Jaccard index:

$$\frac{P(A,B)}{P(A)+P(B)-P(A,B)}$$

Log probabilities show up often in machine learning methods — we'll see them later on in the course!

Important: Defining Probability Models

Previously, we used the following definitions:

$$P(A, B) = P(\text{ randomly chosen doc mentions both } A \text{ \& } B\text{ })$$
$$P(A) = P(\text{ randomly chosen doc mentions } A\text{ })$$

These could be defined differently, and what the "best" definition is depends on the problem you are addressing… For example:

$$P(A, B) = P(\text{ randomly chosen } \textbf{sentence} \text{ mentions both } A \text{ \& } B\text{ })$$
$$P(A) = P(\text{ randomly chosen } \textbf{sentence} \text{ mentions } A\text{ })$$

Even more complicated:

$$P(A, B) = P(\text{ randomly chosen } \textbf{sentence} \text{ mentions both } A \text{ \& } B \text{ within } k \text{ words from each other })$$

Major Running Theme of the Course: "Design Choices"

"Design choices":

- Should we convert all words to lower-case?
- Should we lemmatize?
- Should we remove stop words?
- Should we use a unigram model or an *n*-gram model for $n > 1$?
- How should we define our probability model?
- …

Not sure about what to use? Try multiple options and see whether conclusions drawn from the analysis changes!

- It's good for you to figure out which design choices lead to significant changes and which do not
- Keep in mind: often when working with a company/organization, there will typically not be a person to tell you what the "correct" way is to choose all the design choices — you have to decide on these!

Co-occurrence Analysis Applications

- If you're an online store/retailer:
  anticipate *when* certain products are likely to be purchased/ rented/consumed more
    - Products & dates
- If you have a bunch of physical stores:
  anticipate *where* certain products are likely to be purchased/ rented/consumed more
    - Products & locations
- If you're the police department:
  create "heat map" of where different criminal activity occurs
    - Types of crime & locations

Co-occurrence Analysis Applications

- If you're an online store/retailer:
  anticipate *when* certain products are likely to be purchased/
  rente
  - P
- If yo
  antic
  rente
  - P
- If yo
  crea
  - Types of crime & locations

Examples of data to take advantage of:

- data collected by your organization
- social networks
- news websites
- blogs

Web scraping frameworks can be helpful:

- Scrapy
- Selenium (great with JavaScript-heavy pages)

Co-occurrences at the Character Level

Given a character, what is the distribution of the character that is next?

$$P(A, B) = P(\text{ randomly chosen sequence of 2 consecutive characters is } A \text{ followed by } B\text{ })$$

$$P(A) = P(\text{ randomly chosen sequence of 2 consecutive characters starts with } A\text{ })$$

$$P(B|A) = \frac{P(A, B)}{P(A)}$$

$$= \frac{\text{\# seq. of 2 consecutive characters equal to } A \text{ followed by } B}{\text{\# seq. of 2 consecutive characters starting with } A}$$

Co-occurrences at the Character Level

A is now a sequence of L characters

Given L characters, what is the distribution of the character that is next?

$$P(A, B) = P(\text{ randomly chosen sequence of } L+1 \text{ consecutive characters is } A \text{ followed by } B\text{ })$$

$$P(A) = P(\text{ randomly chosen sequence of } L+1 \text{ consecutive characters starts with } A\text{ })$$

$$P(B|A) = \frac{P(A,B)}{P(A)}$$

$$= \frac{\#\text{ seq. of } L+1 \text{ consecutive characters equal to } A \text{ followed by } B}{\#\text{ seq. of } L+1 \text{ consecutive characters starting with } A}$$

Sampling from this distribution to generate text was already suggested by Claude Shannon in 1948 and is an extremely basic pre-cursor to GPT

“A Mathematical Theory of Communication”

(Shannon, 1948)

3. The Series of Approximations to English

To give a visual idea of how this series of processes approaches a language, typical sequences in the approximations to English have been constructed and are given below. In all cases we have assumed a 27-symbol “alphabet,” the 26 letters and a space.

1. Zero-order approximation (symbols independent and equi-probable).

   XFOML RXKHRJFFJUJ ZLPWCFWKCYJ
   FFJEYVKCQSGXYD QPAAMKBZAACIBZLHJQD

2. First-order approximation (symbols independent but with frequencies of English text).

   OCRO HLI RGWR NMIELWIS EU LL NBNESEBYA TH EEI
   ALHENHTTPA OOBTTVA NAH BRL

3. Second-order approximation (digram structure as in English).

   ON IE ANTSOUTINYS ARE T INCTORE ST BE S DEAMY
   ACHIN D ILONASIVE TUCOOWE AT TEASONARE FUSO
   TIZIN ANDY TOBE SEACE CTISBE

4. Third-order approximation (trigram structure as in English).

   IN NO IST LAT WHEY CRATICT FROURE BIRS GROCID
   PONDENOME OF DEMONSTURES OF THE REPTAGIN IS
   REGOACTIONA OF CRE

Co-occurrences at the Character Level

Demo

Course Outline

Part I: Exploratory data analysis

Identify structure present in "unstructured" data

- Frequency and co-occurrence analysis *Basic probability & statistics*
- Visualizing high-dimensional data/dimensionality reduction
- Clustering
- Topic modeling

Part II: Predictive data analysis

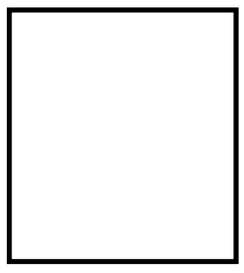
Make predictions using known structure in data

- Basic concepts and how to assess quality of prediction models
- Neural nets and deep learning for analyzing images and text

Visualizing High-Dimensional Data

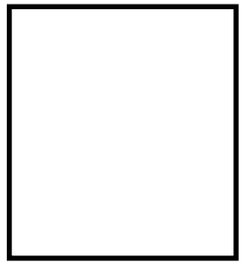
(Flashback) Multiple Documents

Choose a common vocabulary to use across all documents



Text doc #1

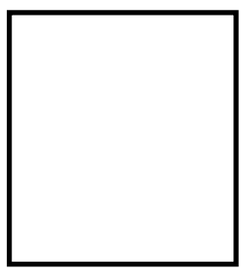
[Feature vector #1]



Text doc #2

[Feature vector #2]

How do we visualize all *n* text docs at once if *n* is large?



Text doc #*n*

[Feature vector #*n*]

This idea of representing data as feature vectors is very general — not just for text!

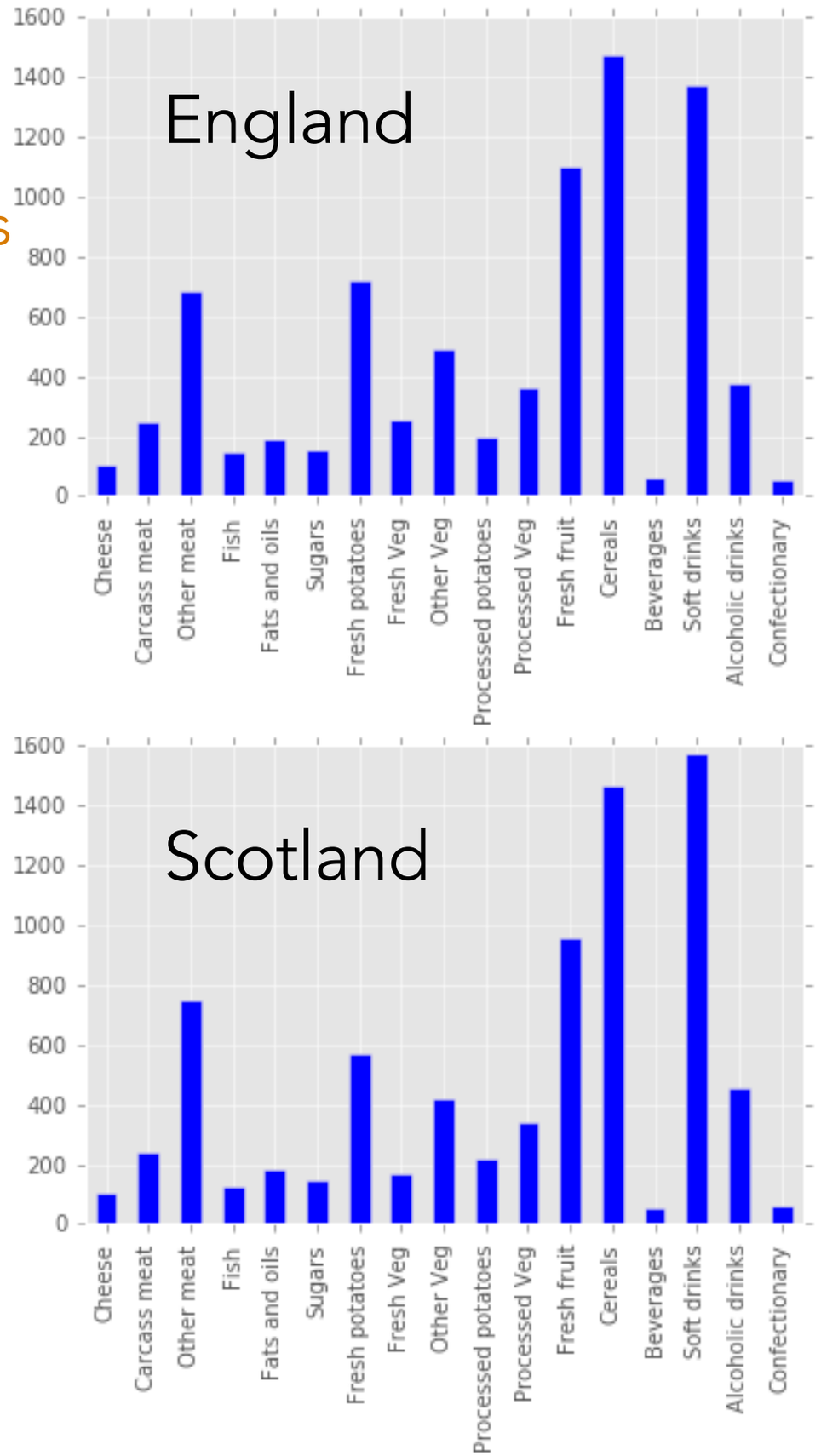
Here’s another concrete example

This is a *structured* dataset but it’ll be helpful in understanding concepts

Imagine we had hundreds of these

Impractical: manually examine hundreds of histograms

How do we make a visualization that quickly tells us how these differ?



Source: http://setosa.io/ev/principal-component-analysis/

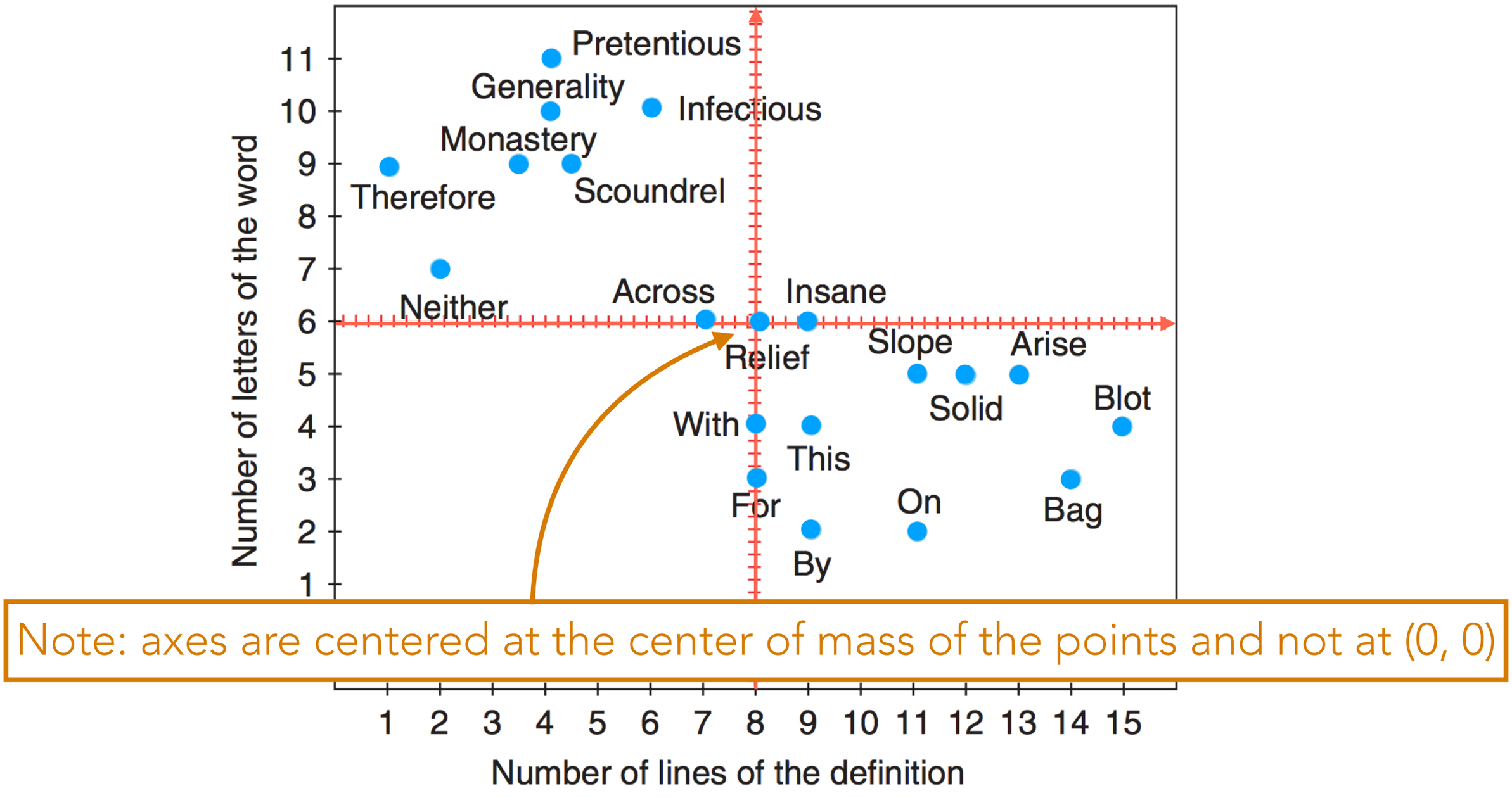
The issue is that as humans we can only really visualize up to 3 dimensions easily

Goal: Somehow reduce data dimensionality to 1, 2, or 3

We will begin with the most famous dimensionality reduction method: principal component analysis (PCA)

Principal Component Analysis (PCA)

How to project 2D data down to 1D?

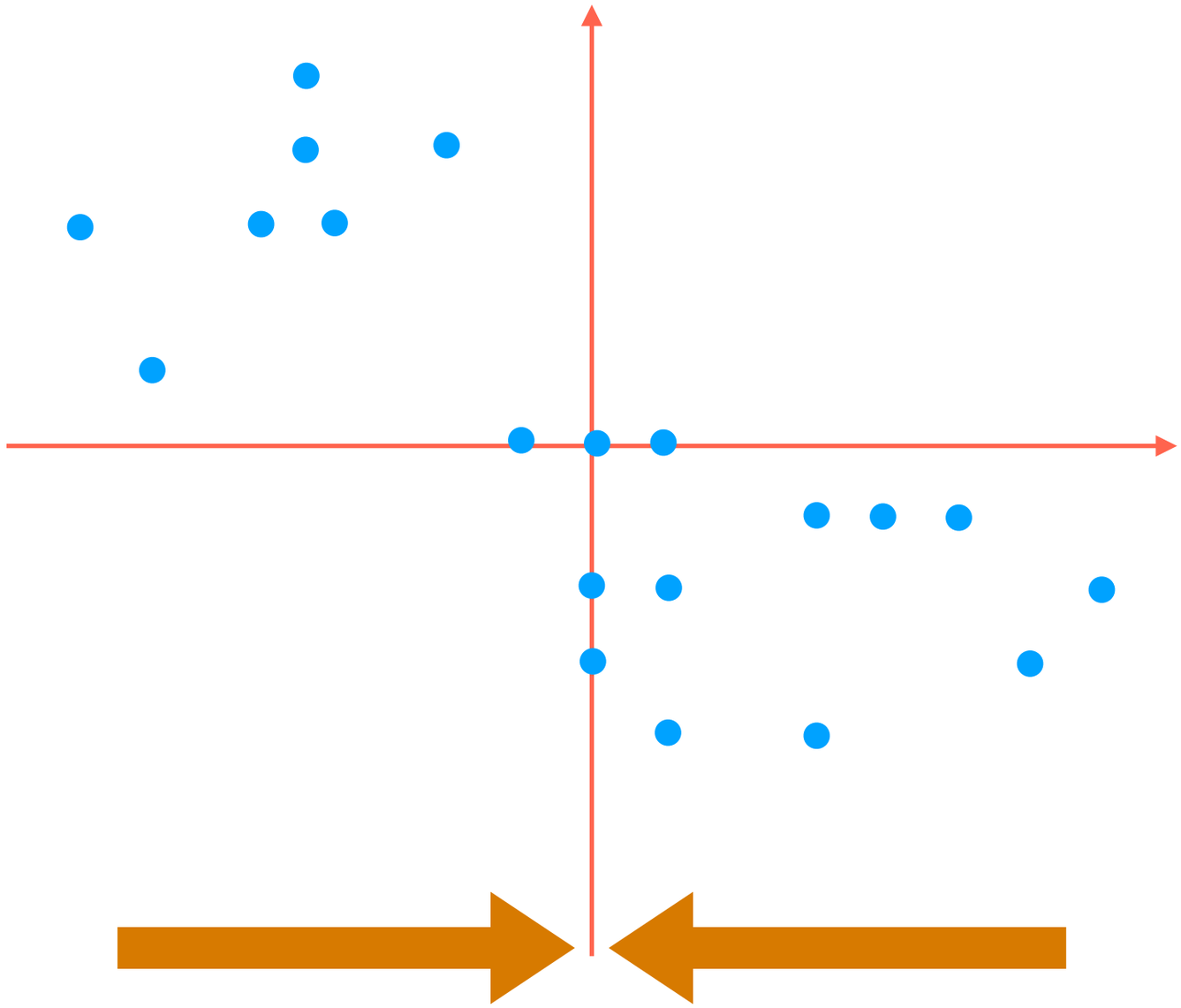


Note: axes are centered at the center of mass of the points and not at (0, 0)

Hervé Abdi and Lynne J. Williams. Principal component analysis. Wiley Interdisciplinary Reviews: Computational Statistics. 2010.

Principal Component Analysis (PCA)

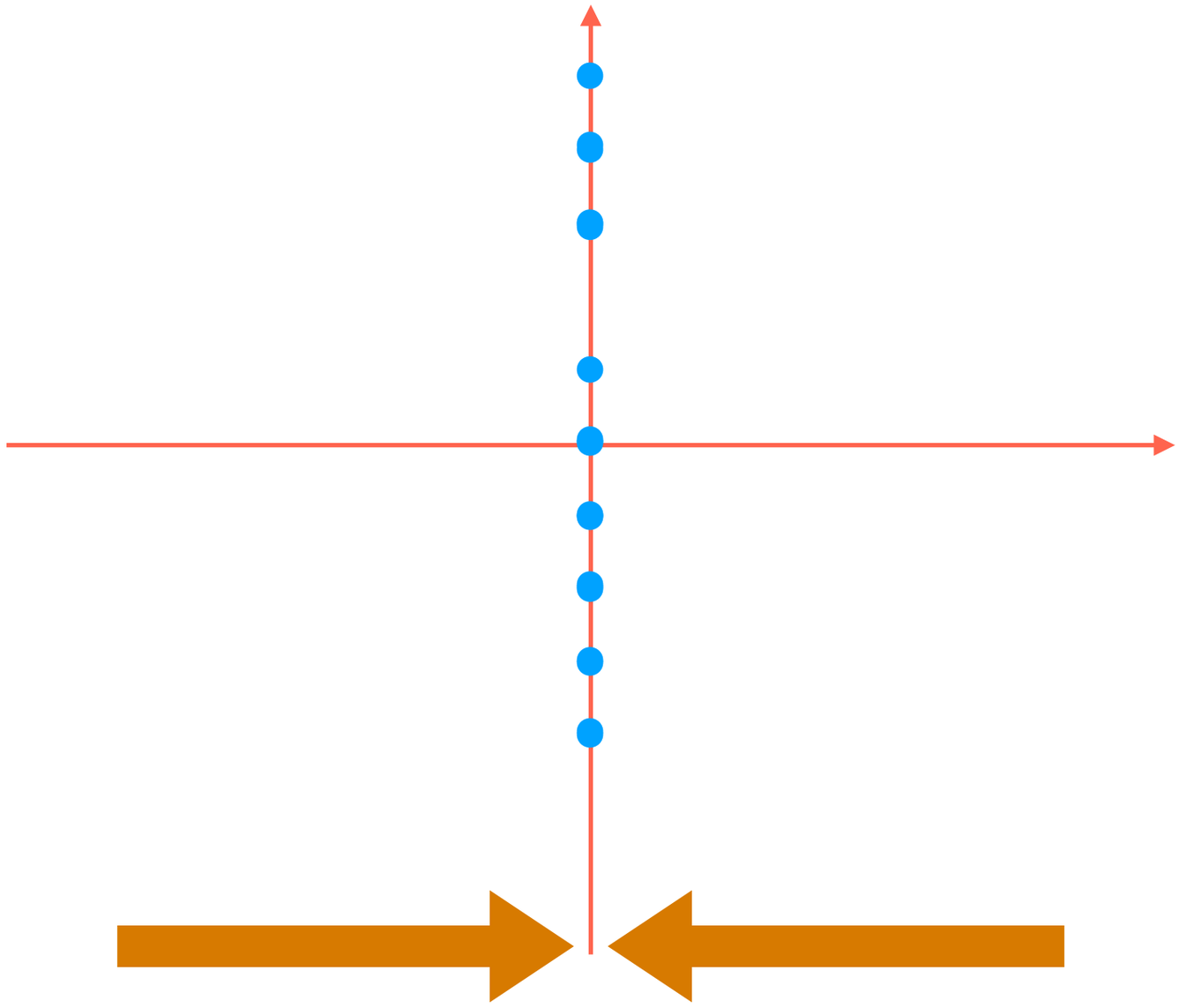
How to project 2D data down to 1D?



Simplest thing to try: flatten to one of the red axes

Principal Component Analysis (PCA)

How to project 2D data down to 1D?

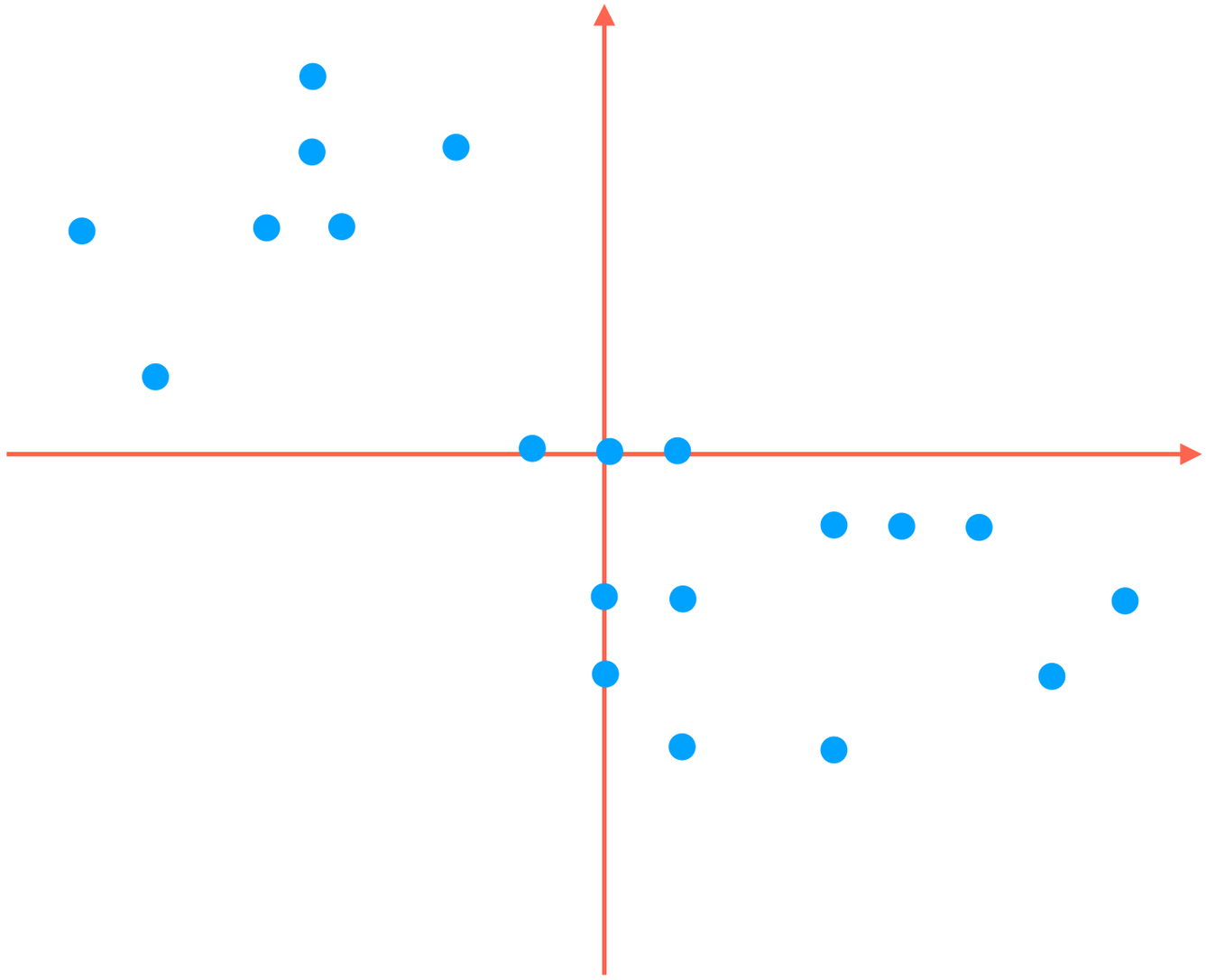


Simplest thing to try: flatten to one of the red axes

(We could of course flatten to the other red axis)

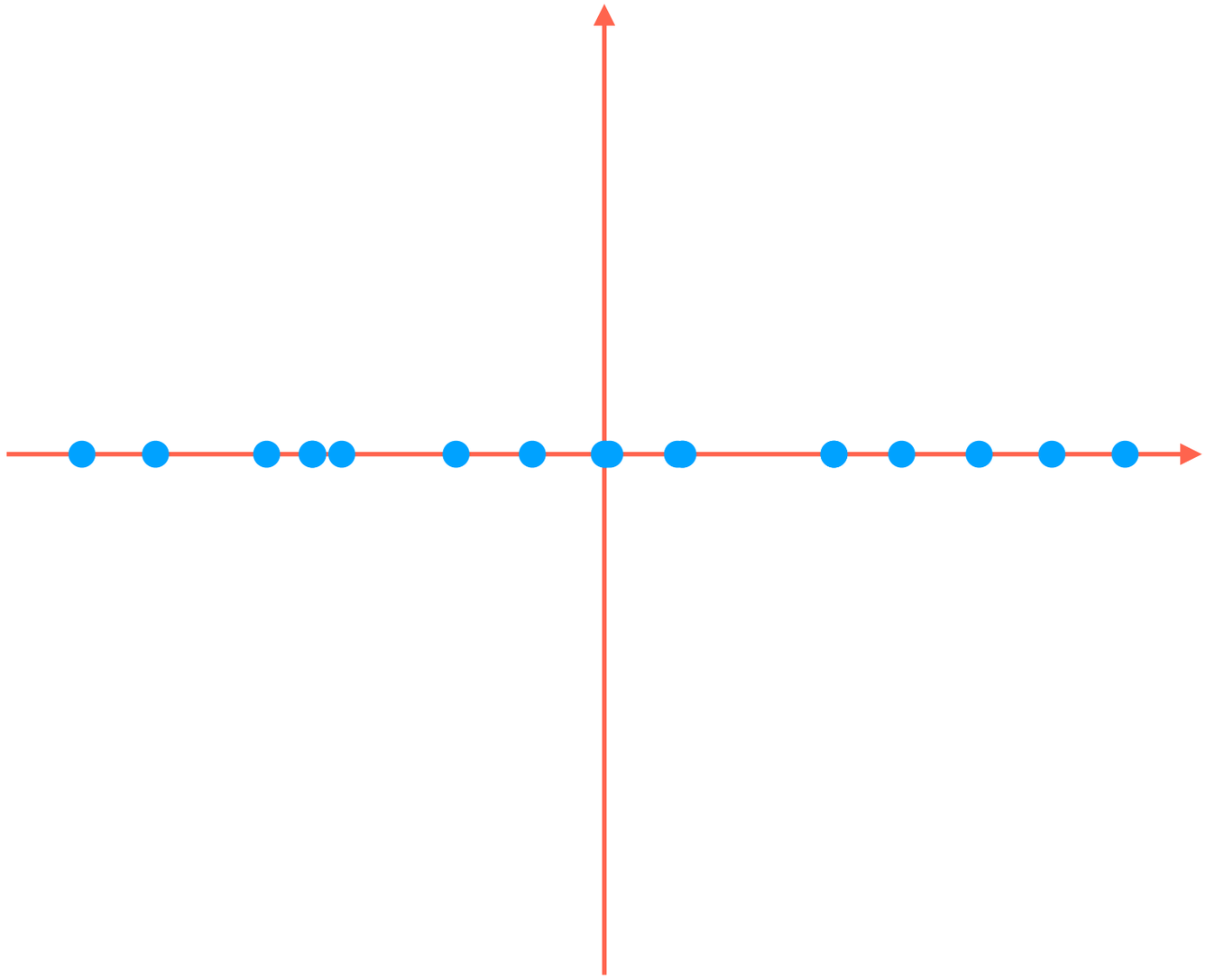
Principal Component Analysis (PCA)

How to project 2D data down to 1D?



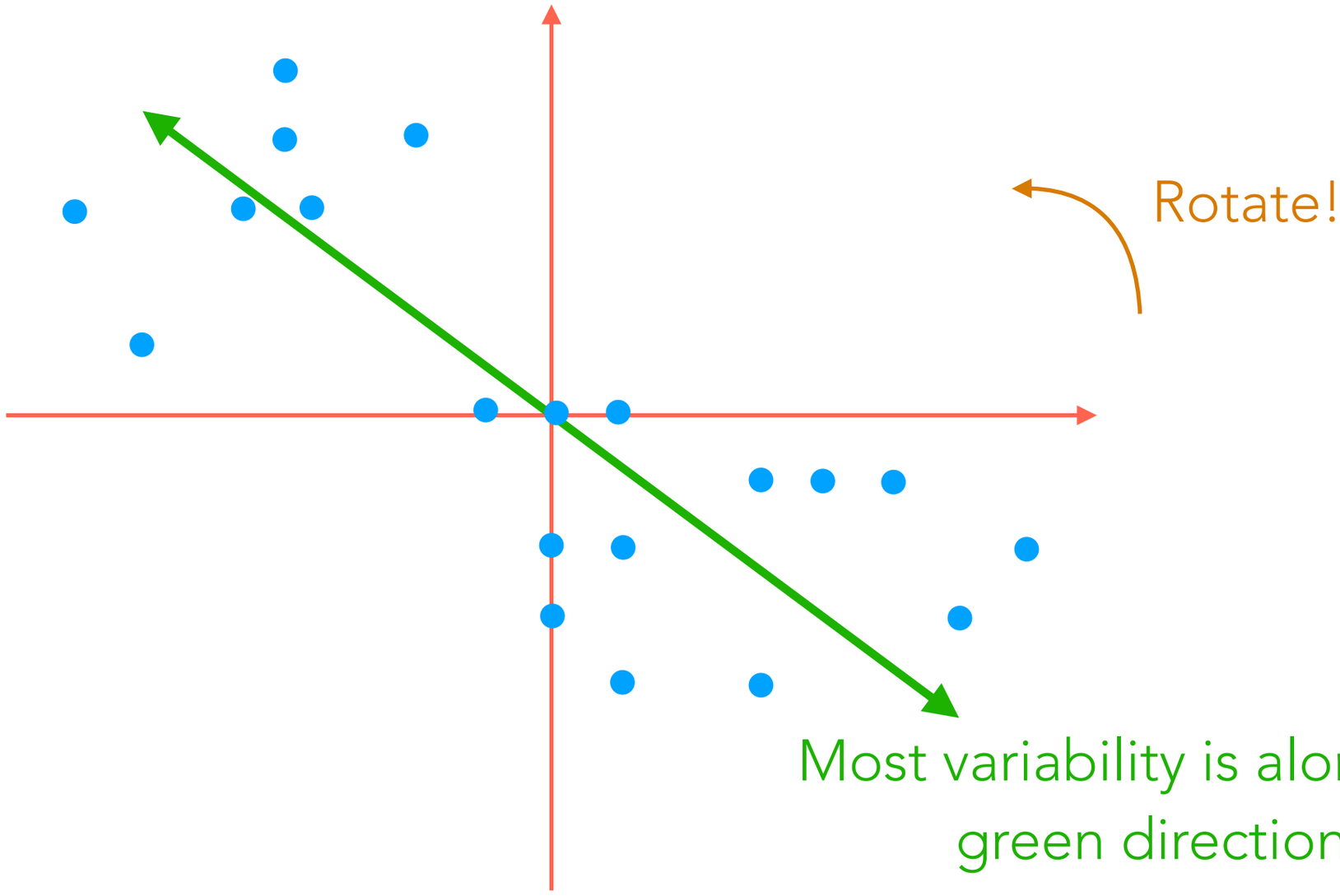
Principal Component Analysis (PCA)

How to project 2D data down to 1D?



Principal Component Analysis (PCA)

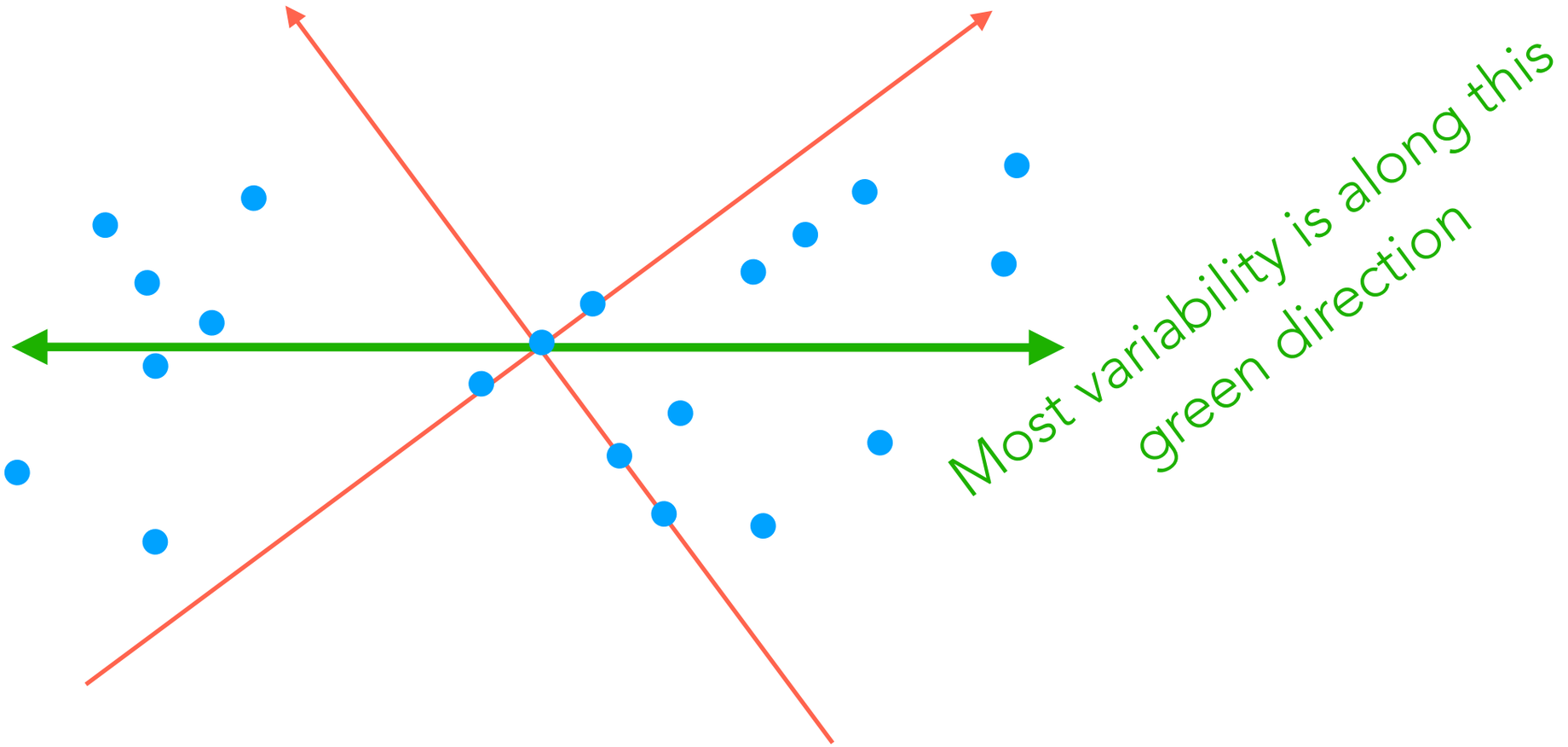
How to project 2D data down to 1D?



But notice that most of the variability in the data is *not* aligned with the red axes!

Principal Component Analysis (PCA)

How to project 2D data down to 1D?



Principal Component Analysis (PCA)

How to project 2D data down to 1D?

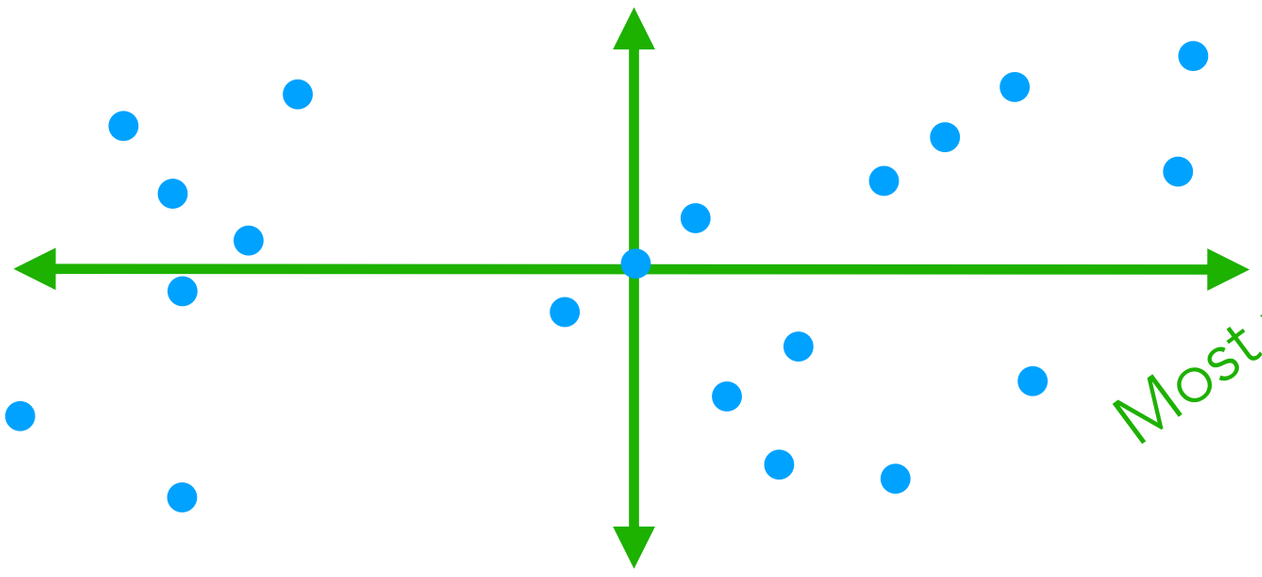
The idea of PCA actually works for 2D ➔ 2D as well
(and just involves rotating, and not “flattening” the data)

Principal Component Analysis (PCA)

~~How to project 2D data down to 1D?~~

How to rotate 2D data so 1st axis has most variance

before "flattening"



The idea of PCA actually works for 2D ➔ 2D as well
(and just involves rotating, and not "flattening" the data)

2nd green axis chosen to be 90° ("orthogonal") from first green axis

3D Dataset Example

http://setosa.io/ev/principal-component-analysis/

PCA in Higher Dimensions

- Finds top *k* orthogonal directions that explain the most variance in the data
  - 1st component: explains most variance along 1 direction
  - 2nd component: explains most of remaining variance along next direction that is orthogonal to 1st direction
  - 3rd component: explains most of remaining variance along next direction that is orthogonal to both the 1st and 2nd directions
  - …
- "Flatten" data by retaining only the top *k* dimensions
  (if *k* < original dimension, then we are doing dimensionality reduction)

Principal Component Analysis (PCA)

Demo