Carnegie Mellon University
Heinz College

94-775 Unstructured Data Analytics

Lecture 6: Wrap up manifold learning, intro to clustering

Slides by George H. Chen

Administrivia

- 😱 Quiz 1 is this Friday during your recitation slot (11am-11:40am HBH 2008)
  - Material coverage: up to and including last Friday's recitation
- HW1 is due tonight
  - Friendly reminder: please look at the PDF that you're submitting before you submit it **(check that everything displays correctly)**
- There's an optional Quiz 1 review session tomorrow (Wed Mar 26) 5pm-6pm run by your TA Johnna
  - It's over Zoom (check Canvas -> Zoom for the link)
- Hopefully you've starting thinking about your final projects!

Reminder of Quiz 1 (and Quiz 2) Format

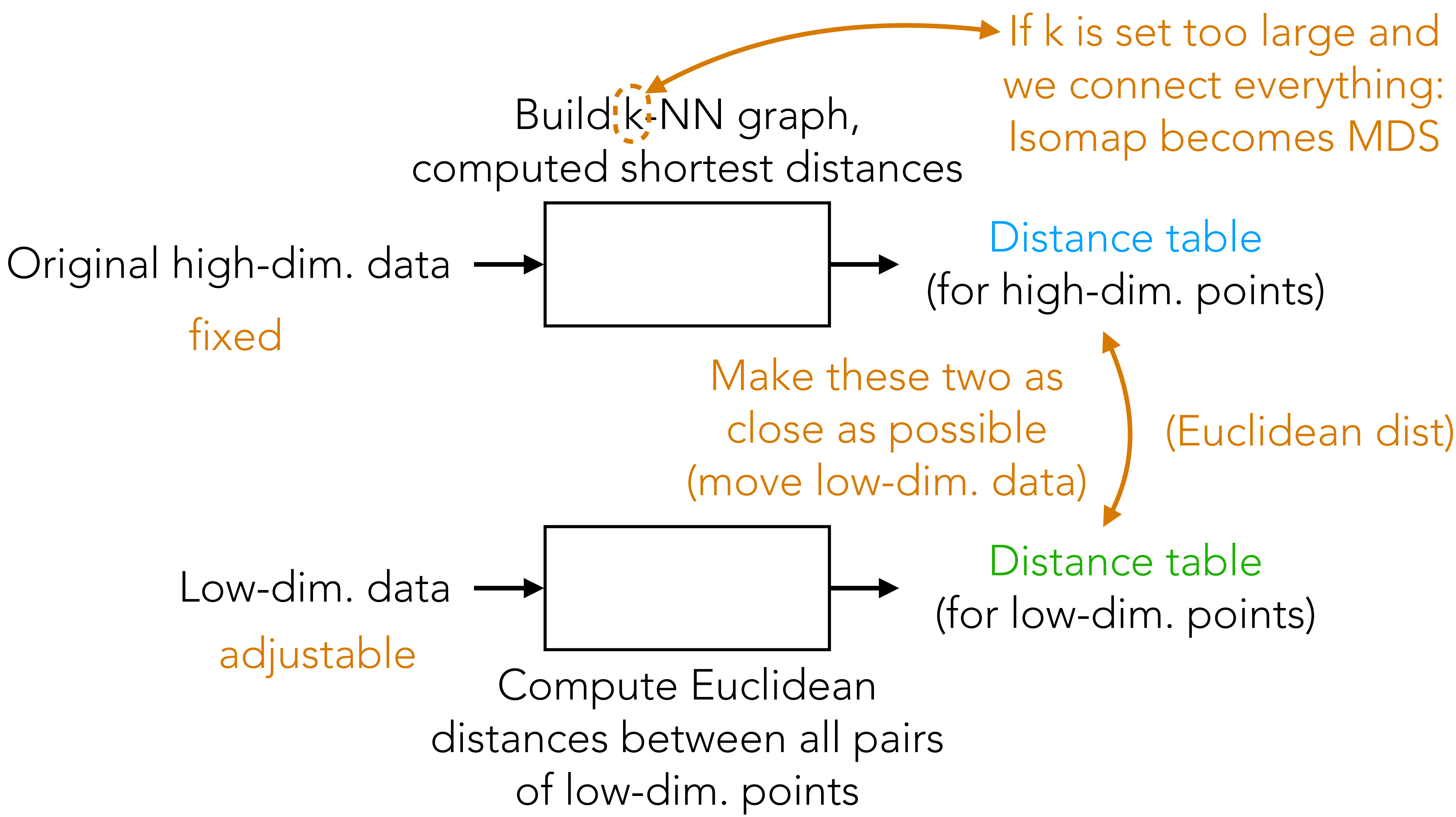
Format:

- **In-person, on paper**
- Each quiz is **40 minutes**
- No electronics may be used during the exam
  (e.g., do <u>not</u> use a laptop, tablet, phone, calculator)
- Open notes (must be on paper and <u>not electronic</u>)

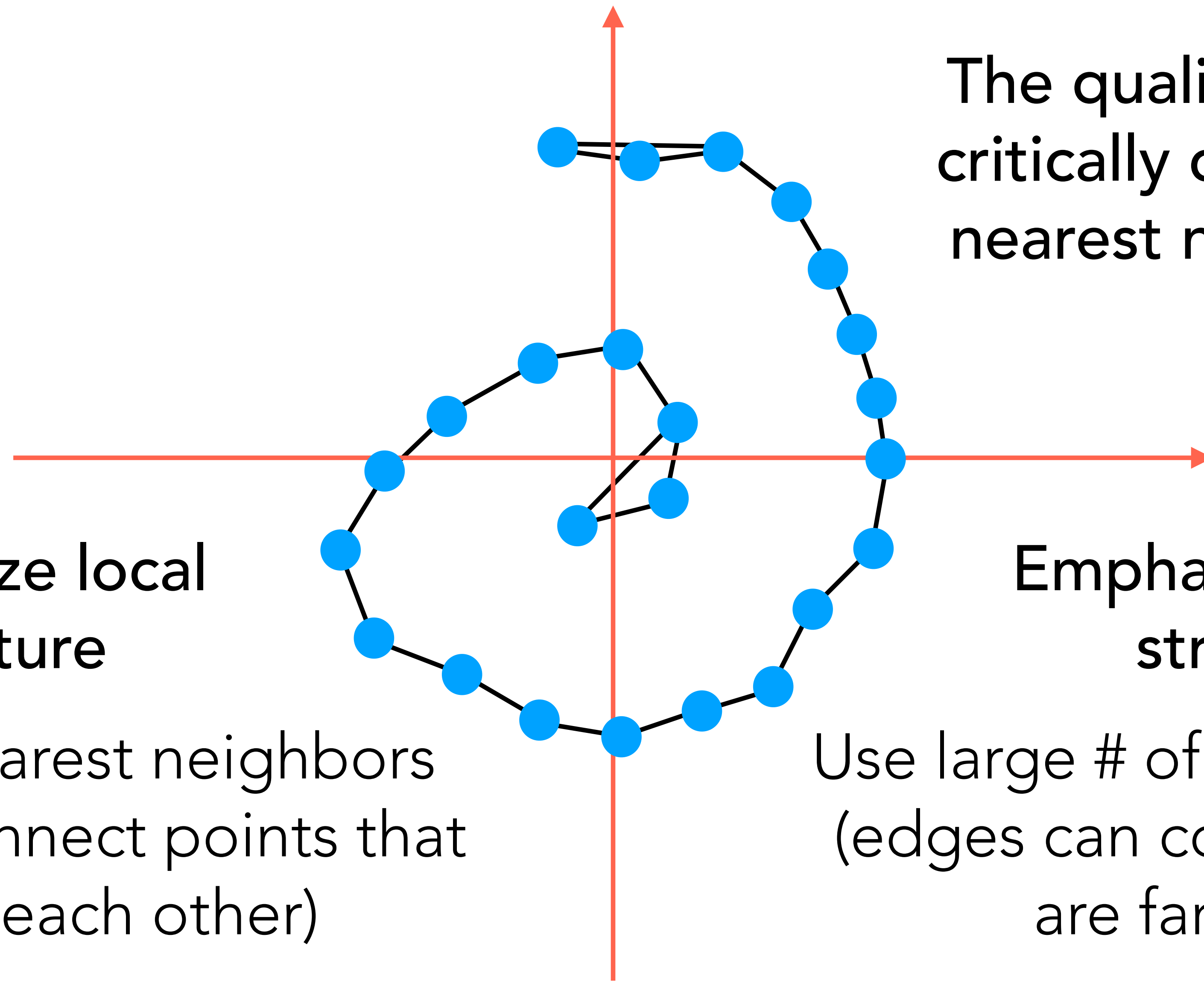
There is no limit on how many sheets of notes you bring

This is a brand new format of quizzes for 94-775!
We're trying to have each of these be lower stakes and short!

(Flashback) Isomap



(Flashback) Some Observations on Isomap



The quality of the result critically depends on the nearest neighbor graph

Emphasize local structure

Use small # of nearest neighbors (edges tend to connect points that are closer to each other)

There might not be enough edges

Emphasize global structure

Use large # of nearest neighbors (edges can connect points that are farther apart)

Might connect points that shouldn't be connected

In general: try different parameters for nearest neighbor graph construction when using Isomap + visualize

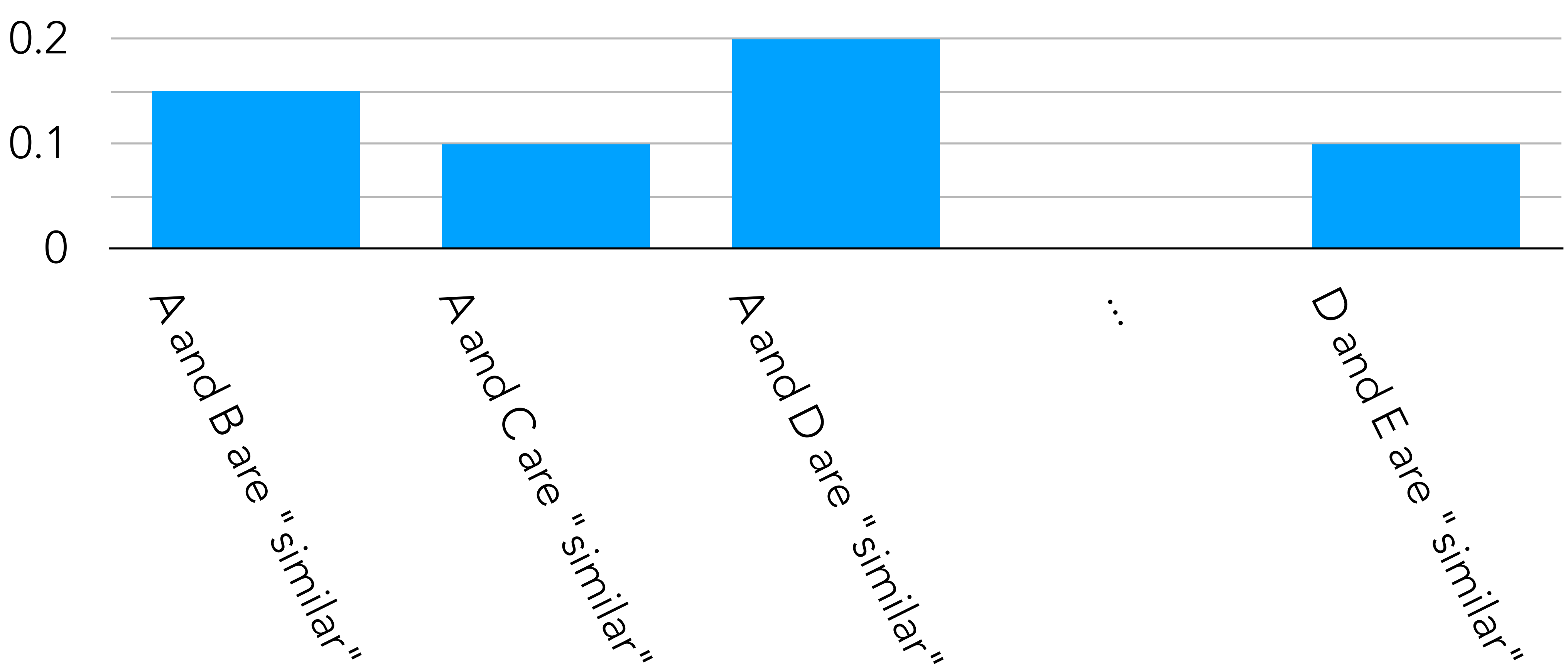
t-SNE
(t-distributed stochastic neighbor embedding)

High-level t-SNE Idea

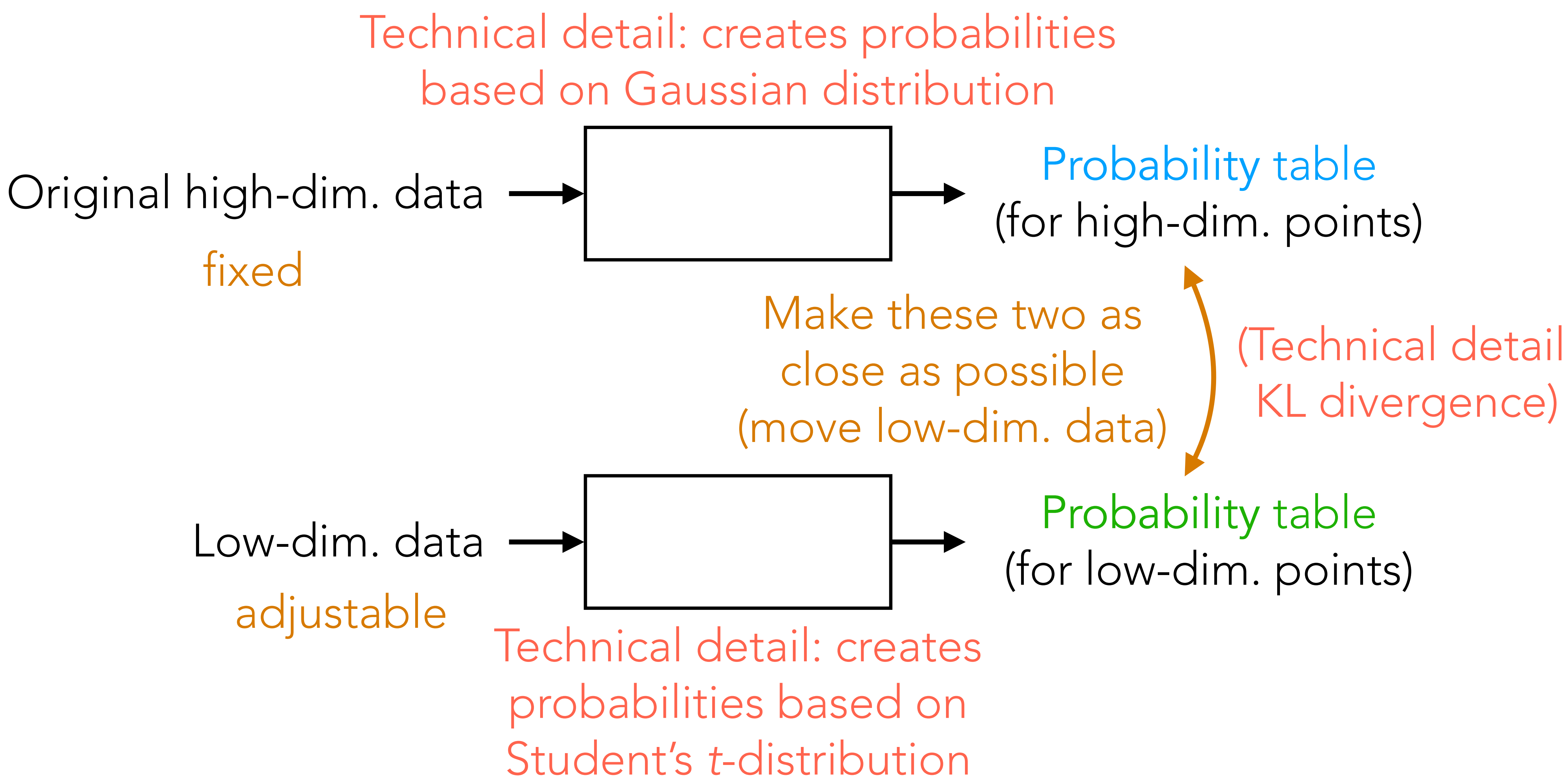
- Don't use deterministic definition of which points are neighbors

	A	B	C	D	E
A	0	5	8	13	16
B	5	0	5	10	13
C	8	5	0	5	8
D	13	10	5	0	5
E	16	13	8	5	0

- Use probabilistic notation instead



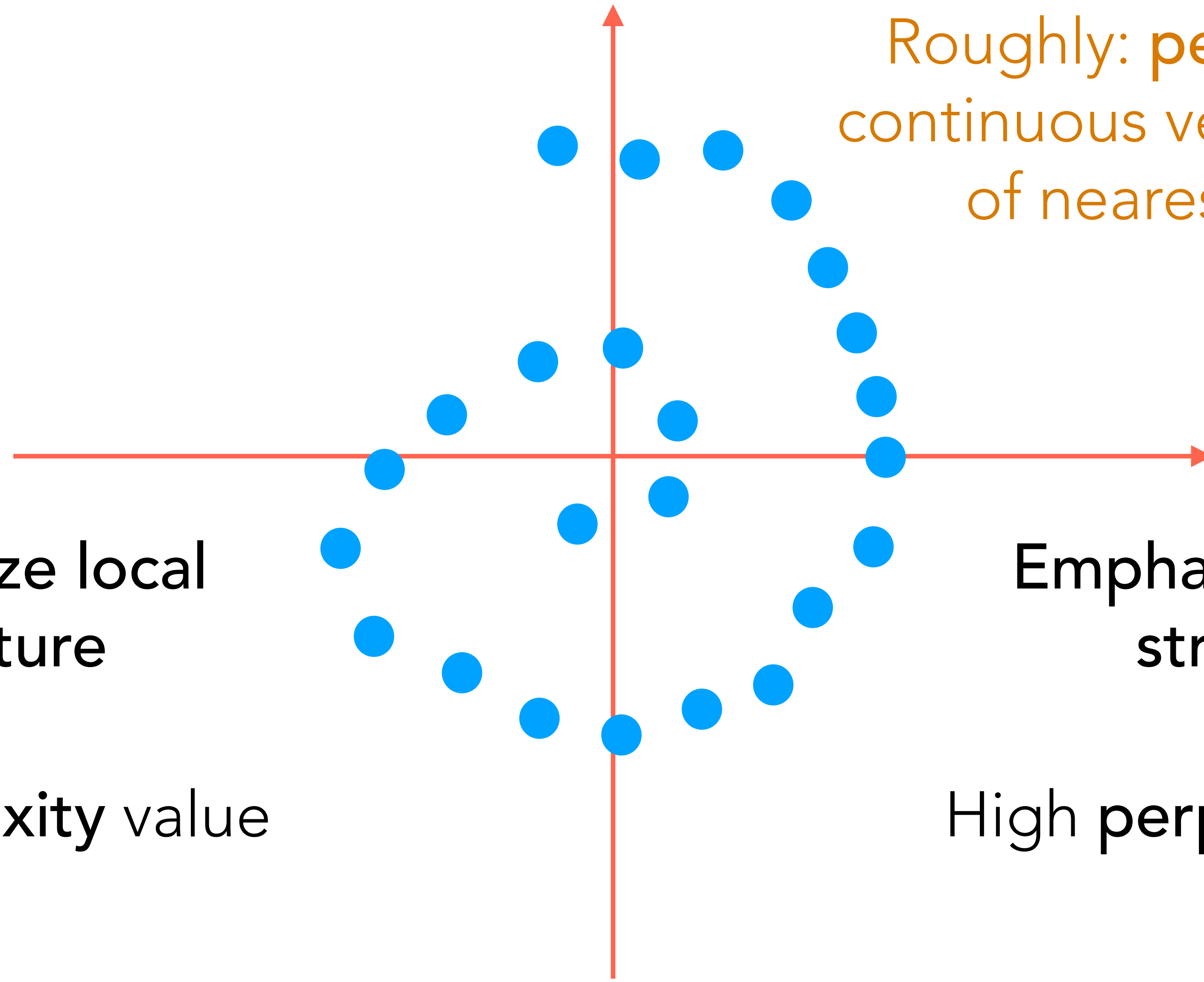
t-SNE



Technical details are in separate slides (posted on webpage)

t-SNE Parameters...

Roughly: **perplexity** is like a continuous version of "number of nearest neighbors"



Emphasize local structure

Low **perplexity** value

Emphasize global structure

High **perplexity** value

Also: play with # iterations, learning rate

how many times to try to improve guess of low-dim. representation

each time we try to improve low-dim. representation, how much we can change it

In practice, often people initialize with PCA

Manifold Learning with t-SNE

Demo

Interpreting t-SNE Plots

Required reading:
"How to Use t-SNE Effectively" (Wattenberg et al 2016)
https://distill.pub/2016/misread-tsne/

Let’s look at images

(Flashback) Multiple Documents

Choose a common vocabulary to use across all documents

☀ ☁ ☂ ☃

Text doc #1 → [Feature vector #1]

Text doc #2 → [Feature vector #2]

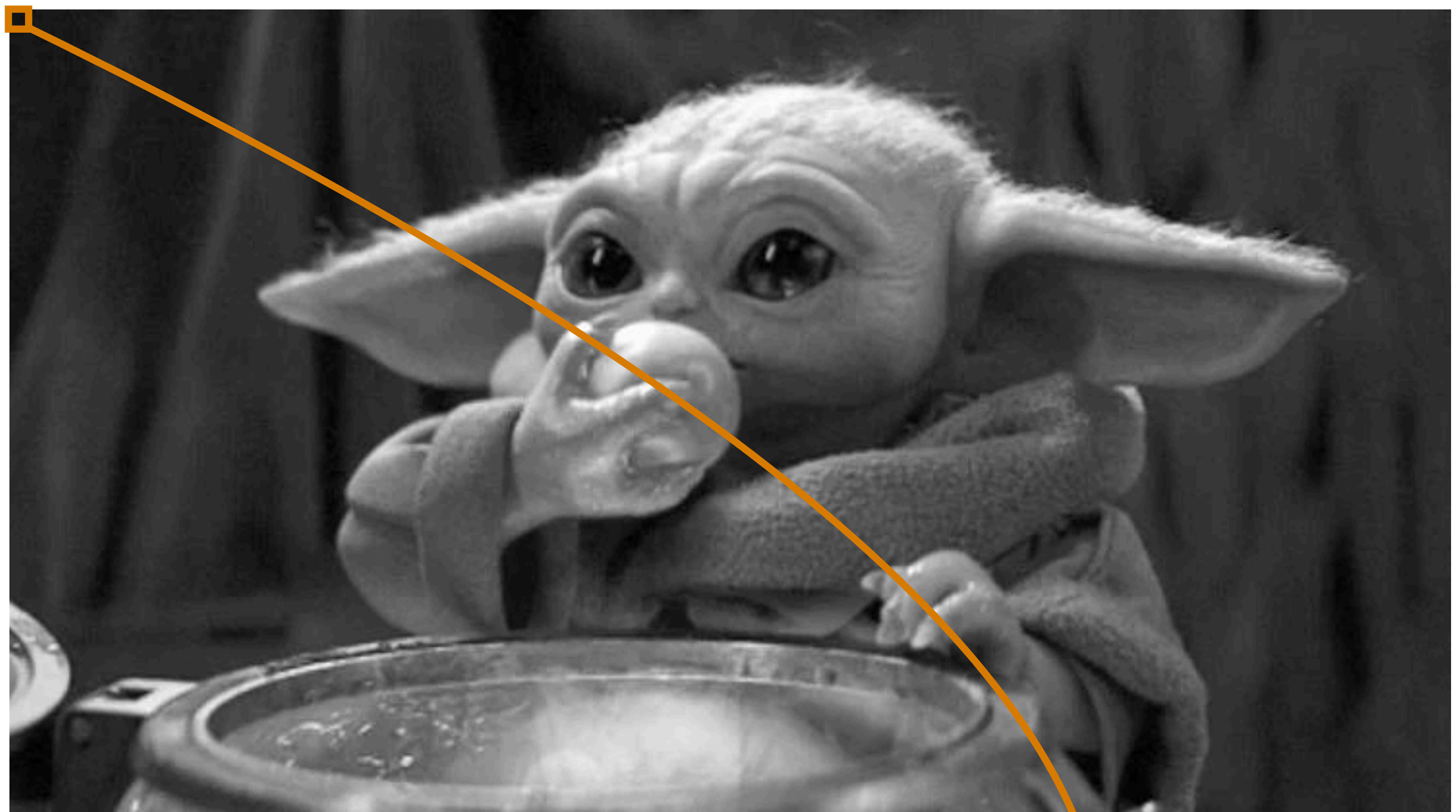
⋮

Text doc #*n* → [Feature vector #*n*]

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

Example: Representing an Image

Go row by row and look at pixel values



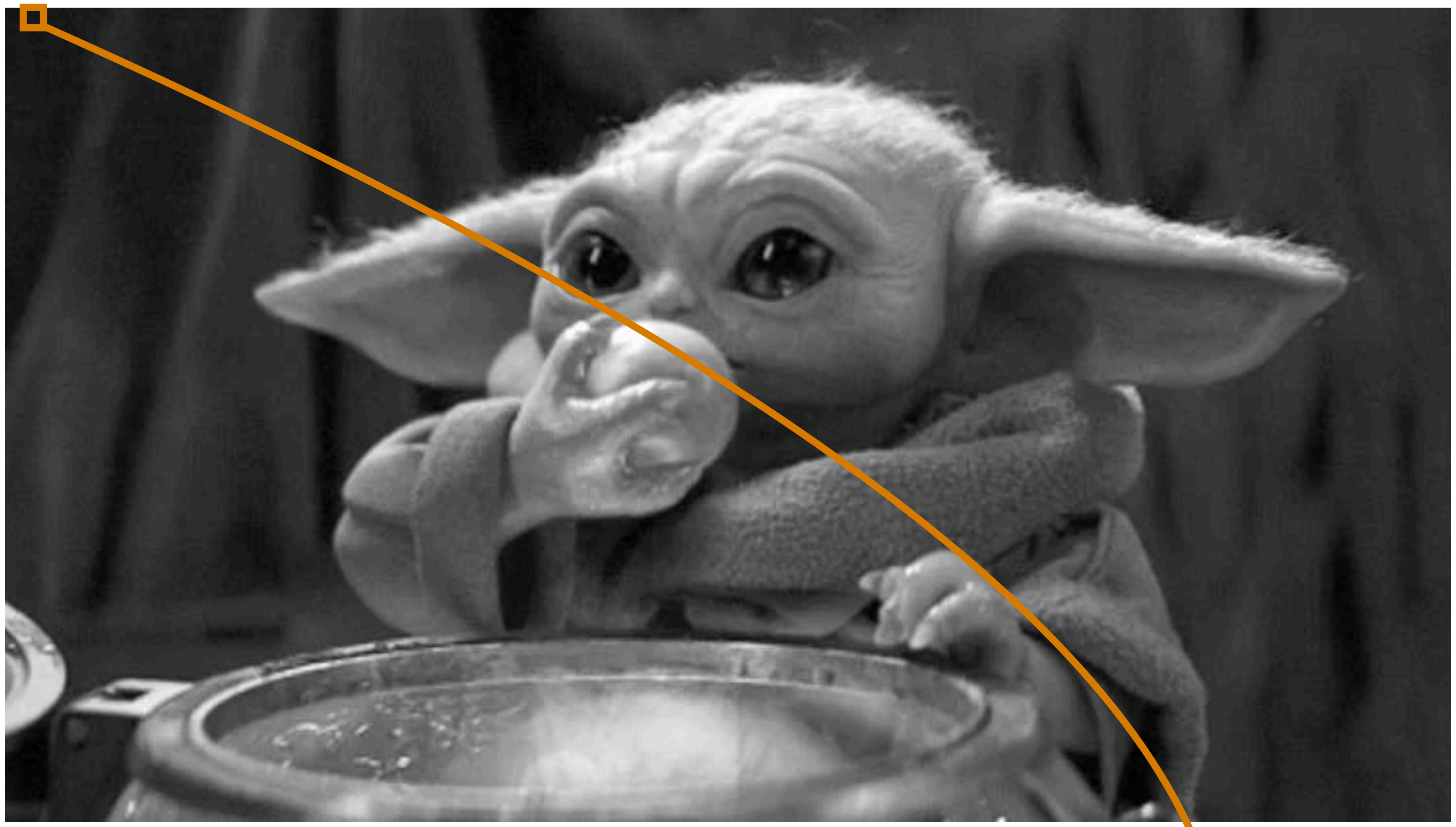
1: black
0: white

[1]

Image source: The Mandalorian

Example: Representing an Image

Go row by row and look at pixel values



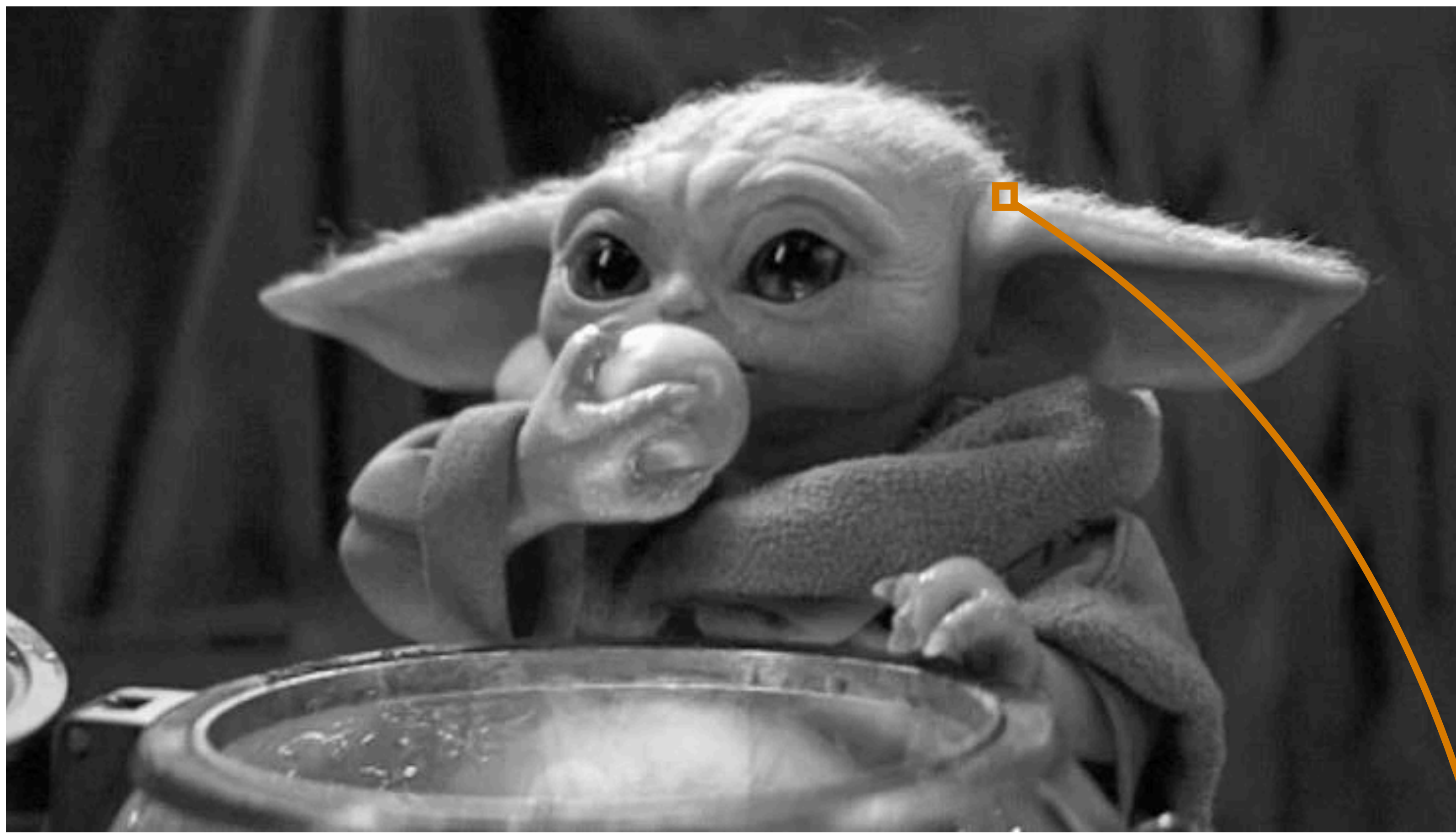
1: black
0: white

[1   0.9   ]

Image source: The Mandalorian

Example: Representing an Image

Go row by row and look at pixel values



1: black
0: white

$[\ 1 \quad 0.9 \quad \cdots \quad 0.1 \qquad\qquad]$

Image source: The Mandalorian

Example: Representing an Image

Go row by row and look at pixel values

1: black
0: white

$[\ 1 \quad 0.9 \quad \cdots \quad 0.1 \quad \cdots \quad 0.9\ ]$

\# dimensions = image height × image width

Image source: The Mandalorian

Very high dimensional!

Terminology Remark

$$[\ 1 \quad 0.9 \quad \cdots \quad 0.1 \quad \cdots \quad 0.9\ ]$$

⚠️ We use "dimension" to means two different things:

- number of axes we can index into for a table/array (e.g., 2D means there are rows & columns)

  # dimensions = 1

- total number of entries in the table/array

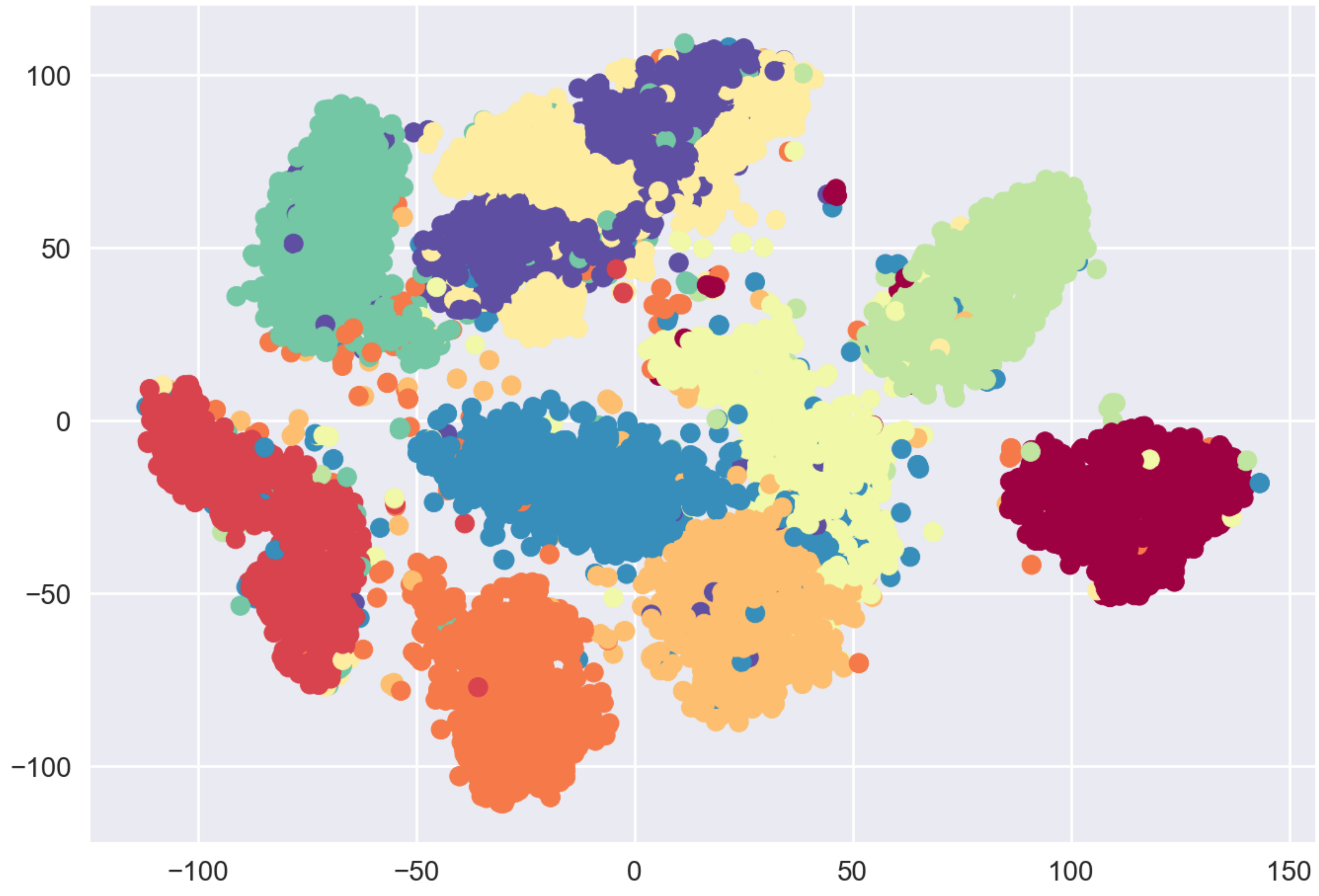
  # dimensions = image height × image width

Dimensionality Reduction for Images

Demo

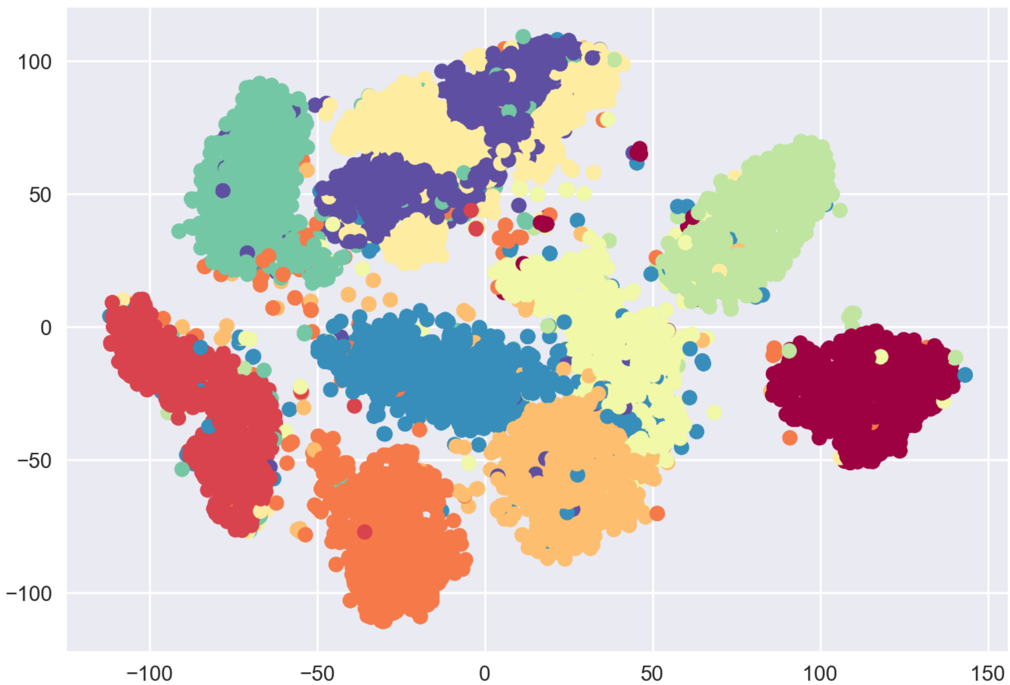
Dimensionality Reduction for Visualization

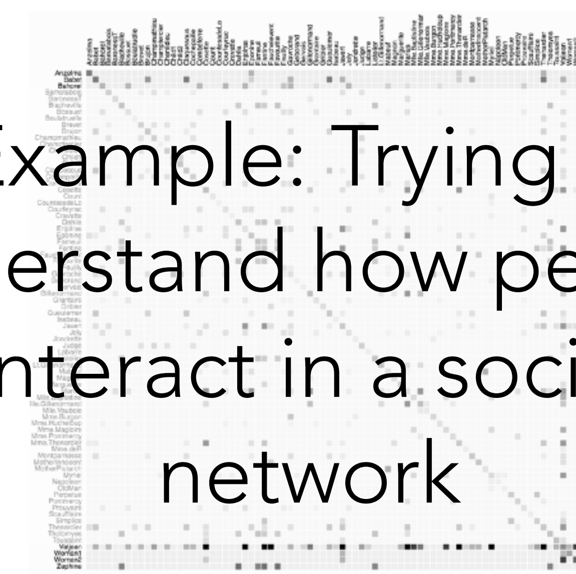
- There are *many* methods (I've posted a link on the course webpage to a scikit-learn example using ~10 methods)
- PCA is very well-understood; the new axes can be interpreted
- Nonlinear dimensionality reduction (manifold learning): new axes may not really be all that interpretable
- PCA is good to try first (look at plot & explained variance ratios)
  - If PCA works poorly, then t-SNE could be a good 2nd thing to try
- If you have good reason to believe that only certain features matter, of course you could restrict your analysis to those!
- t-SNE can be annoying to use but is still very popular
  - Promising recently developed alternative: PaCMAP (Wang et al 2021) accounts for local and global structure simultaneously and also uses "mid-near" neighbors of points — link on course webpage



2D t-SNE plot of handwritten digit images shows clumps that correspond to real digits — this is an example of **clustering structure** showing up in real data

Remark on "Label Information"





Example: Trying to understand how people interact in a social network

Important:
Handwritten digit demo is a toy example where we know which images correspond to digits 0, 1, …, 9

Many real UDA problems:
The data are messy and it's not obvious what the "correct" labels/answers look like — what "correct" means can be ambiguous!

Later on in the course (when we cover predictive analytics), we look at how to take advantage of knowing the "correct" answers

Top right image source: https://bost.ocks.org/mike/miserables/

Clustering Structure Often Occurs!

Lots of real examples, such as:

- Crime might happen more often in specific hot spots
- People applying for micro loans have a few specific uses in mind (education, electricity, healthcare, etc)
- Users in a recommendation system can share similar taste in products

Clustering methods aim to group together data points that are "similar" into "clusters", while having different clusters be "dissimilar"

But what does "similar" or "dissimilar" mean?

Clustering methods will either directly assume a specific meaning of "similarity", or some allow you to specify a similarity/distance function

Note: distance is inversely related to similarity
(two points being more similar $\Longleftrightarrow$ they are closer in distance)