Crime Statistics and ML

UNIVERSITY *of* WASHINGTON

A “standard” ML perspective

Y from X

Can we predict ~~crime~~?

Can we prevent crime? → can we change Y?

And if we can do either, what are the right measures of effectiveness?

loss function

A (slightly) more nuanced set of questions

What if our predictions are only effective for some types of crime?

For some types of neighborhoods?

loss function misspecified

What features are *acceptable* to use in predicting crime?

Demographic
Race
Gender
Class
...

How are these features/labels gathered?
What if they are gathered in an uneven manner?

Not iid

And what will be done with these predictions?

Would most of our concerns be mitigated by:

Removing demographic information from a dataset?

What is the concern here?

In the US, policing, arresting, charging, and convicting for certain crimes has been applied to different populations at *very* unequal rates.

E.g., illicit drug use is charged at much higher rates for minority persons

Moreover, crimes charged at higher rates for certain demographics have been deigned more dangerous than similar ones charged in other demographics

E.g. crack cocaine having higher legal penalties than powder cocaine

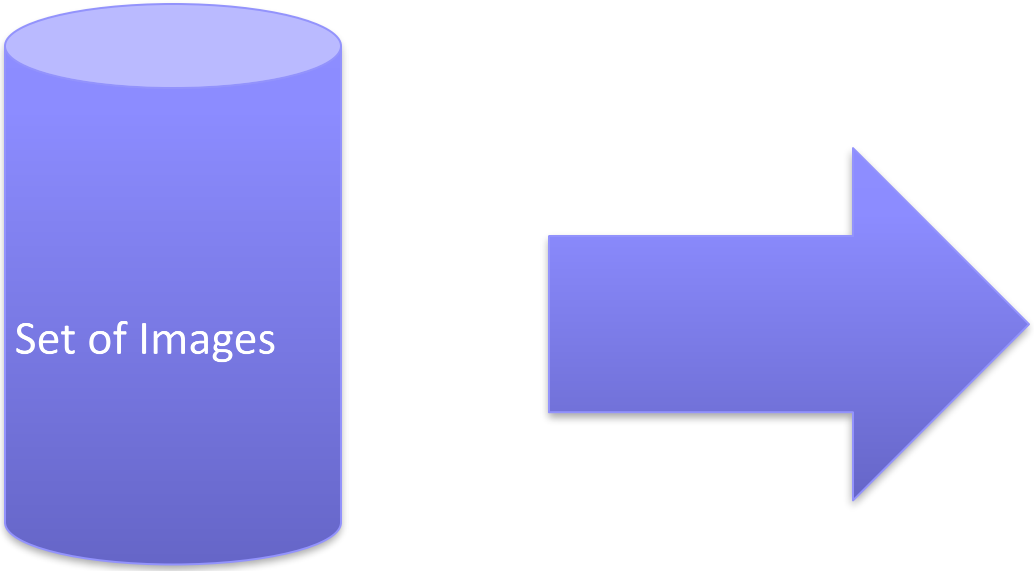
The racial and class makeup of neighborhoods today are the result of decades and centuries of design by lawmakers.

Clustering
K-means

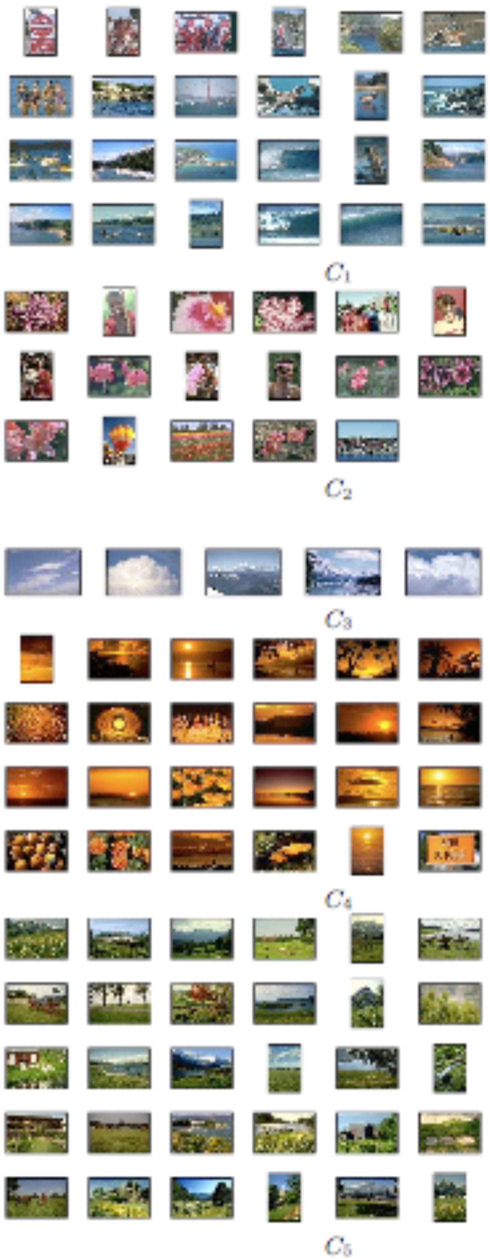
UNIVERSITY of WASHINGTON

©Kevin Jamieson 2018

Clustering images

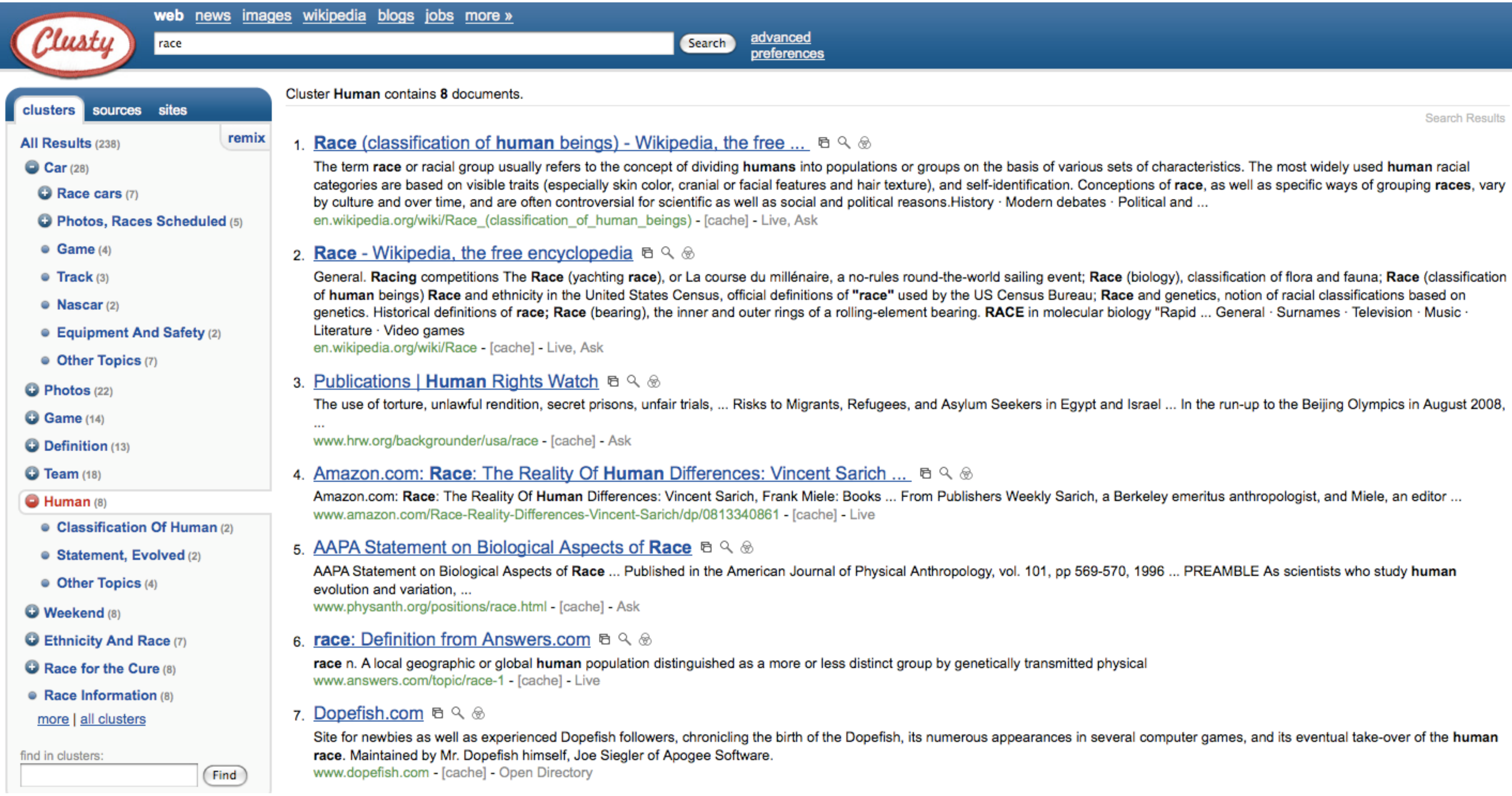


Set of Images



[Goldberger et al.]

Clustering web search results

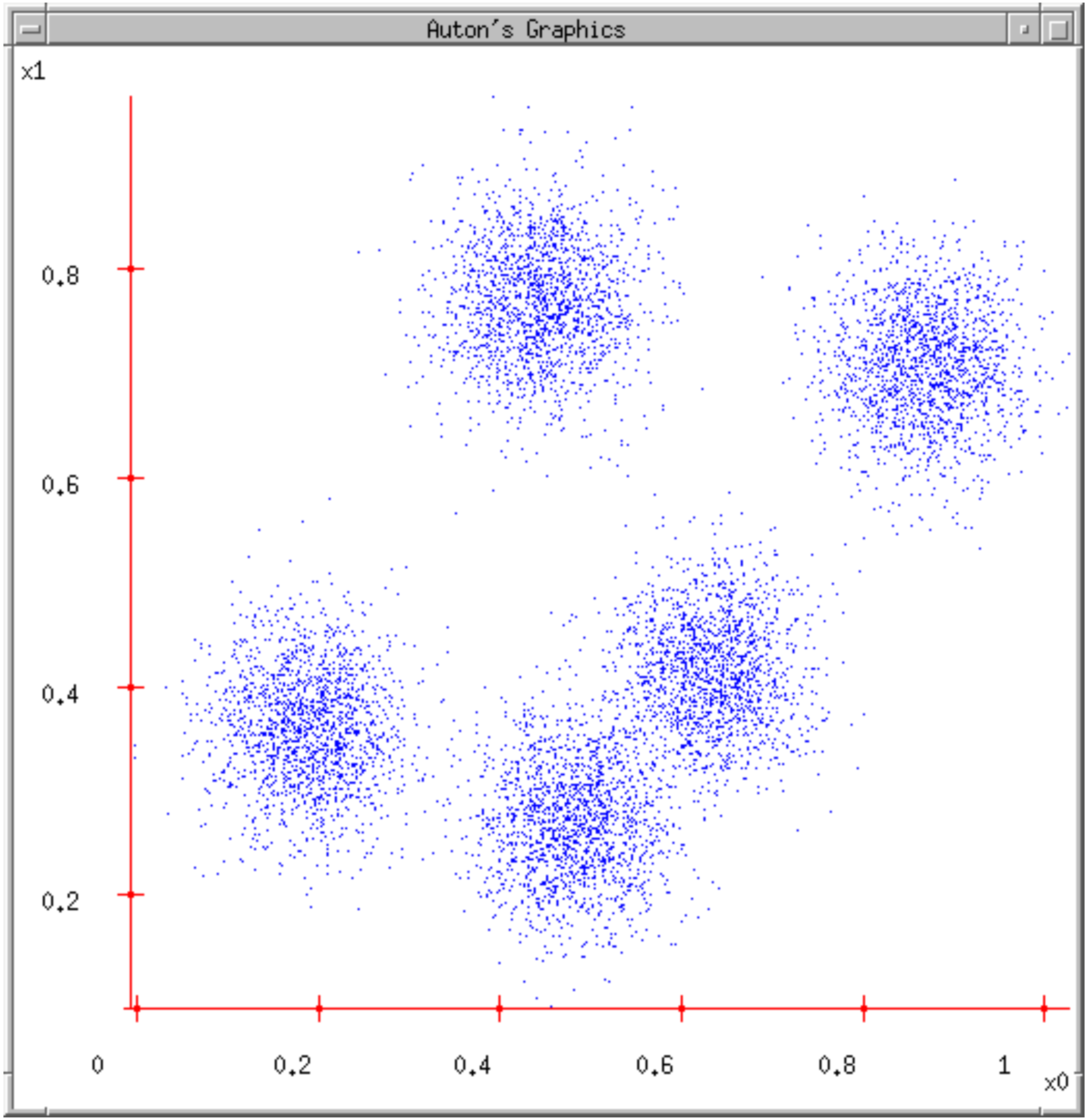


- All Results (238)
  - Car (28)
    - Race cars (7)
    - Photos, Races Scheduled (5)
    - Game (4)
    - Track (3)
    - Nascar (2)
    - Equipment And Safety (2)
    - Other Topics (7)
  - Photos (22)
  - Game (14)
  - Definition (13)
  - Team (18)
  - Human (8)
    - Classification Of Human (2)
    - Statement, Evolved (2)
    - Other Topics (4)
  - Weekend (8)
  - Ethnicity And Race (7)
  - Race for the Cure (8)
  - Race Information (8)
  - more | all clusters

Cluster **Human** contains **8** documents.

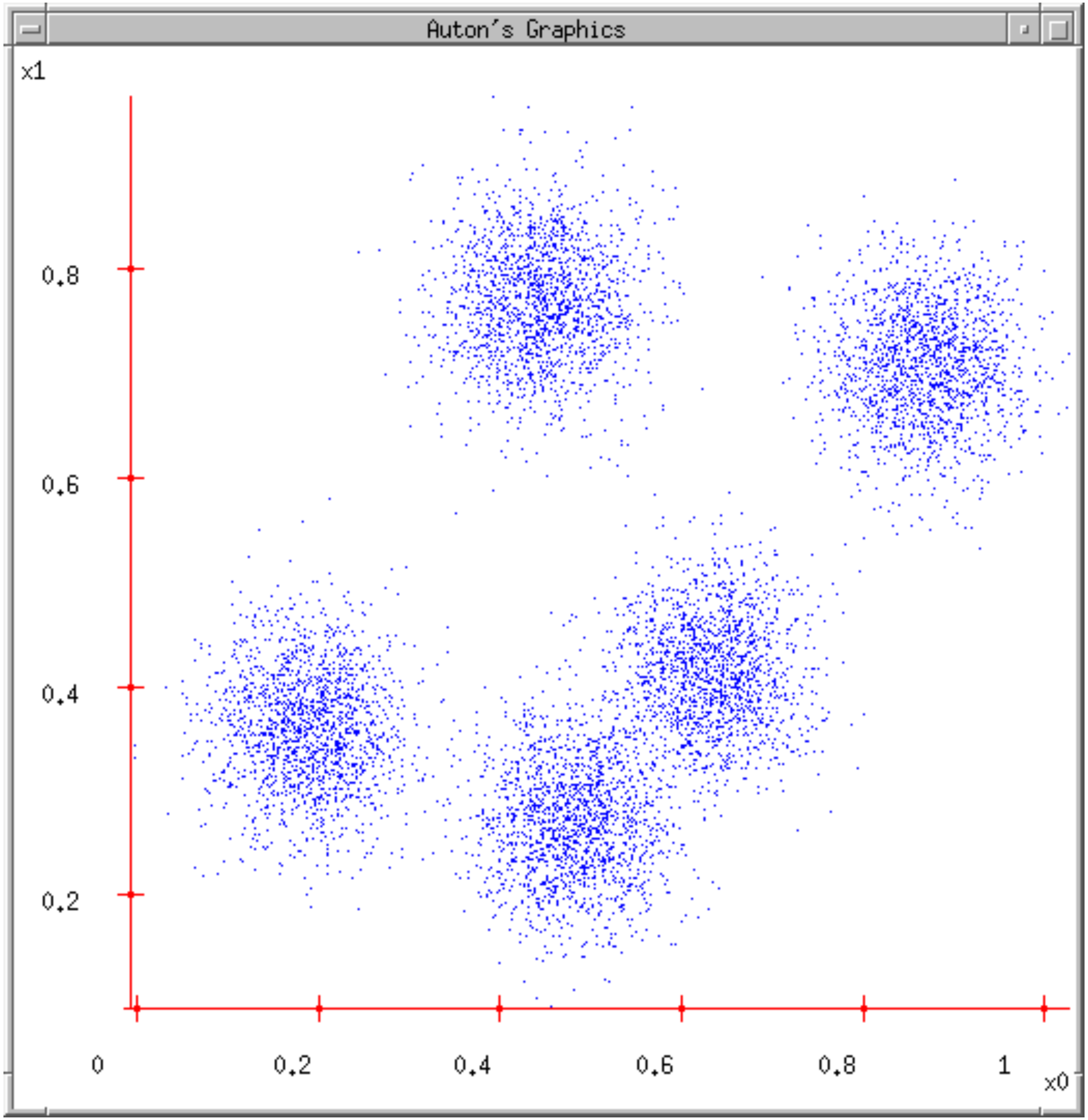
1. **Race** (classification of **human** beings) - Wikipedia, the free ...
   The term **race** or racial group usually refers to the concept of dividing **humans** into populations or groups on the basis of various sets of characteristics. The most widely used **human** racial categories are based on visible traits (especially skin color, cranial or facial features and hair texture), and self-identification. Conceptions of **race**, as well as specific ways of grouping **races**, vary by culture and over time, and are often controversial for scientific as well as social and political reasons.History · Modern debates · Political and ...
   en.wikipedia.org/wiki/Race_(classification_of_human_beings) - [cache] - Live, Ask
2. **Race** - Wikipedia, the free encyclopedia
   General. **Racing** competitions The **Race** (yachting **race**), or La course du millénaire, a no-rules round-the-world sailing event; **Race** (biology), classification of flora and fauna; **Race** (classification of **human** beings) **Race** and ethnicity in the United States Census, official definitions of **"race"** used by the US Census Bureau; **Race** and genetics, notion of racial classifications based on genetics. Historical definitions of **race; Race** (bearing), the inner and outer rings of a rolling-element bearing. **RACE** in molecular biology "Rapid ... General · Surnames · Television · Music · Literature · Video games
   en.wikipedia.org/wiki/Race - [cache] - Live, Ask
3. Publications | **Human** Rights Watch
   The use of torture, unlawful rendition, secret prisons, unfair trials, ... Risks to Migrants, Refugees, and Asylum Seekers in Egypt and Israel ... In the run-up to the Beijing Olympics in August 2008, ...
   www.hrw.org/backgrounder/usa/race - [cache] - Ask
4. Amazon.com: **Race**: The Reality Of **Human** Differences: Vincent Sarich ...
   Amazon.com: **Race**: The Reality Of **Human** Differences: Vincent Sarich, Frank Miele: Books ... From Publishers Weekly Sarich, a Berkeley emeritus anthropologist, and Miele, an editor ...
   www.amazon.com/Race-Reality-Differences-Vincent-Sarich/dp/0813340861 - [cache] - Live
5. AAPA Statement on Biological Aspects of **Race**
   AAPA Statement on Biological Aspects of **Race** ... Published in the American Journal of Physical Anthropology, vol. 101, pp 569-570, 1996 ... PREAMBLE As scientists who study **human** evolution and variation, ...
   www.physanth.org/positions/race.html - [cache] - Ask
6. **race**: Definition from Answers.com
   race n. A local geographic or global **human** population distinguished as a more or less distinct group by genetically transmitted physical
   www.answers.com/topic/race-1 - [cache] - Live
7. Dopefish.com
   Site for newbies as well as experienced Dopefish followers, chronicling the birth of the Dopefish, its numerous appearances in several computer games, and its eventual take-over of the **human race**. Maintained by Mr. Dopefish himself, Joe Siegler of Apogee Software.
   www.dopefish.com - [cache] - Open Directory

Some Data



Clustering

1. Ask user how many clusters they'd like. *(e.g. k=5)*
2. Pick clusters to minimize some objective fn.



Clustering

1. Fix a # of clusters *(e.g. k=5)*
2. Choose/Assign each point x_j to $C(j) \in \{1,\dots, k\}$
    1. Sometimes, pick centers $\boldsymbol{\mu}_1 , \dots \boldsymbol{\mu}_k$

To minimize

assignment

$$\boxed{F(\mu, C)}$$

K-means refers to optimizing this objective:

$$F(\mu, C) = \sum_{j=1}^{m} ||\underline{\mu_{C(j)} - x_j||}^2$$

K-means refers to optimizing this objective:

$$F(\mu, C) = \sum_{j=1}^{m} ||\mu_{C(j)} - x_j||^2$$

How to minimize this quantity?

K-means refers to optimizing this objective:

$$F(\mu, C) = \sum_{j=1}^{m} ||\mu_{C(j)} - x_j||^2$$

How to minimize this quantity?

NP-Hard to minimize exactly. :(

K-means refers to optimizing this objective:

$$F(\mu, C) = \sum_{j=1}^{m} ||\mu_{C(j)} - x_j||^2$$

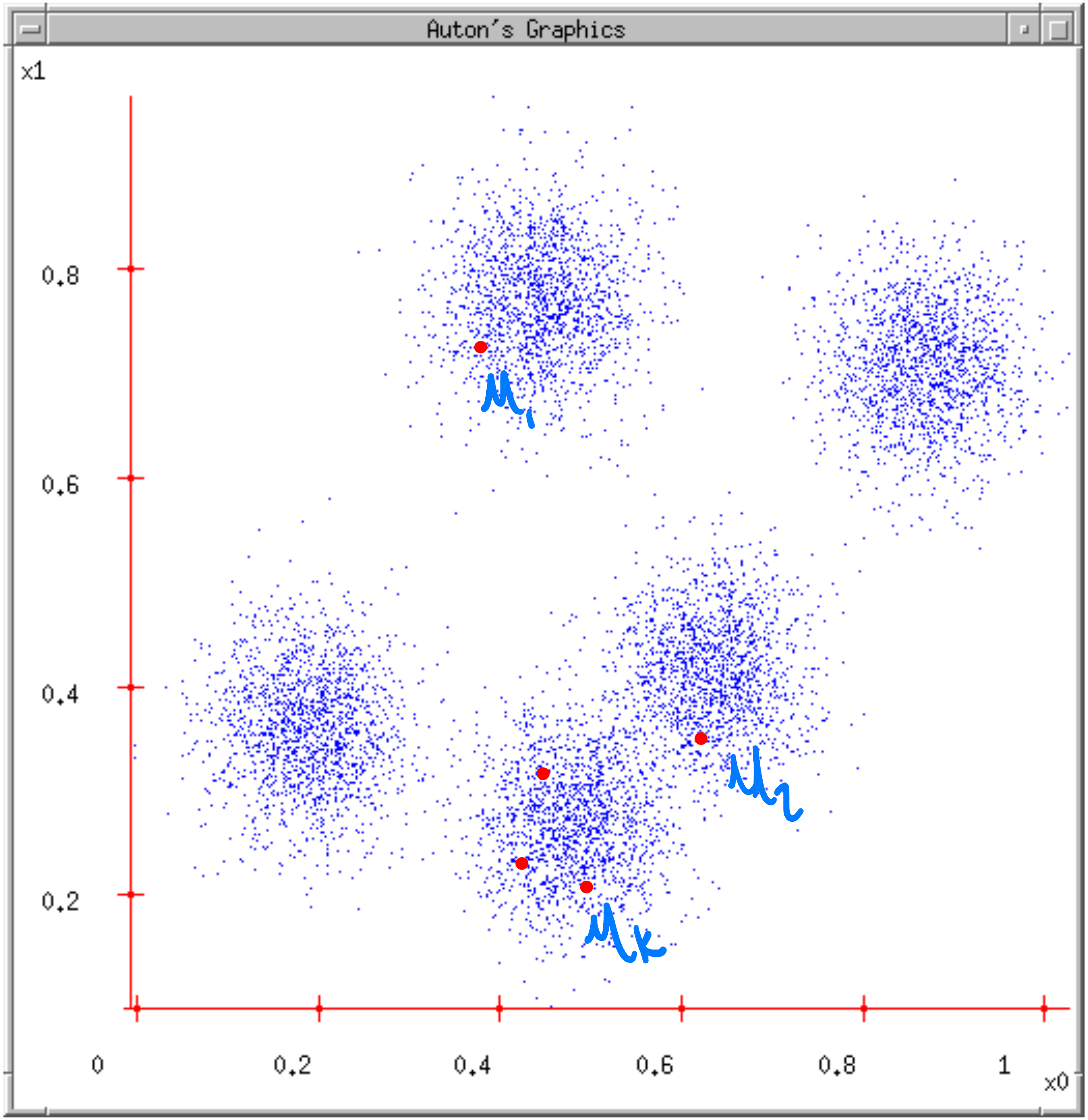
How to minimize this quantity?

NP-Hard to minimize exactly. :(

But, several natural algorithms work well in practice!

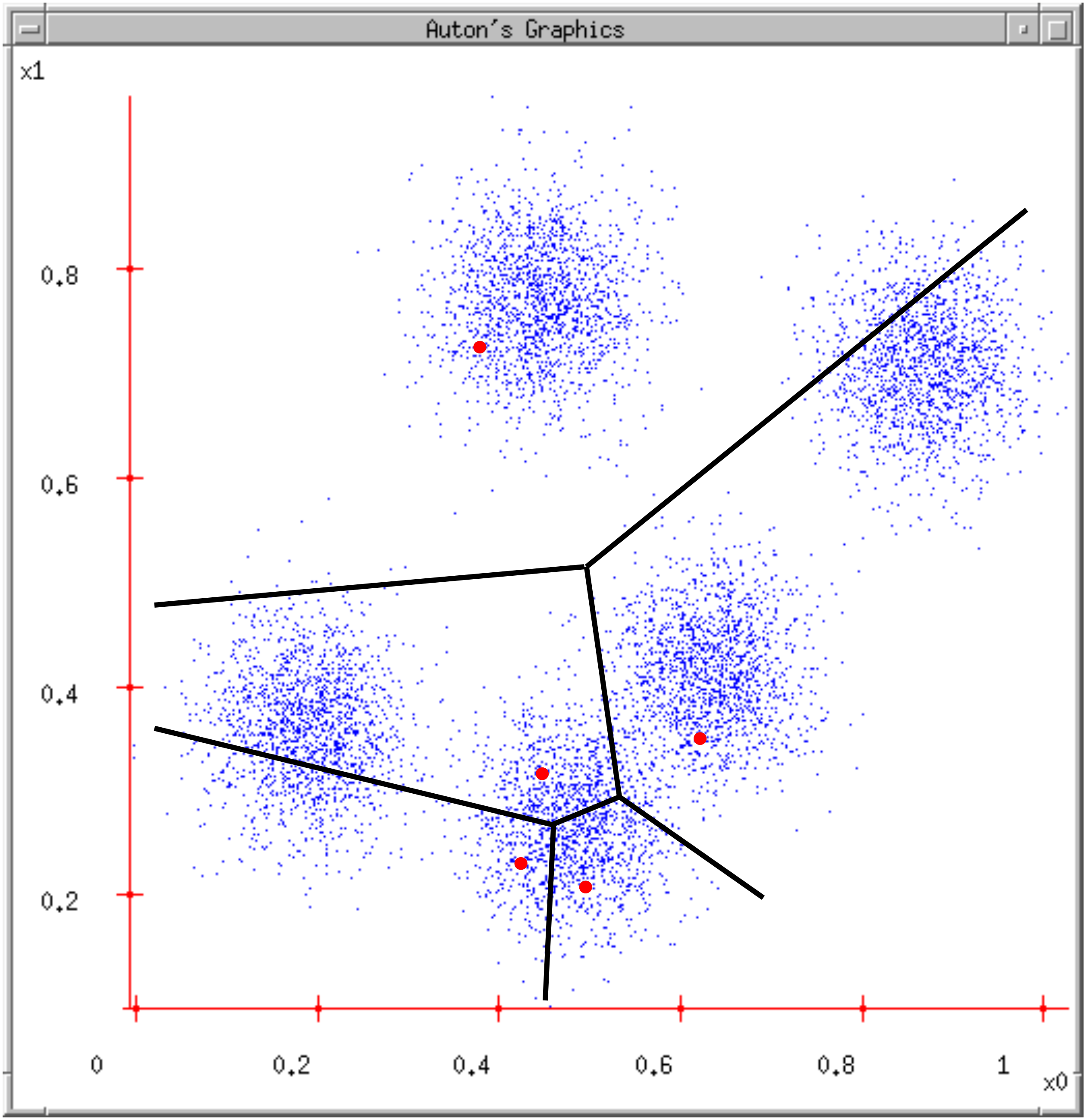
Lloyd's algorithm

1. Ask user how many clusters they'd like. *(e.g. k=5)*
2. Randomly guess k cluster Center locations



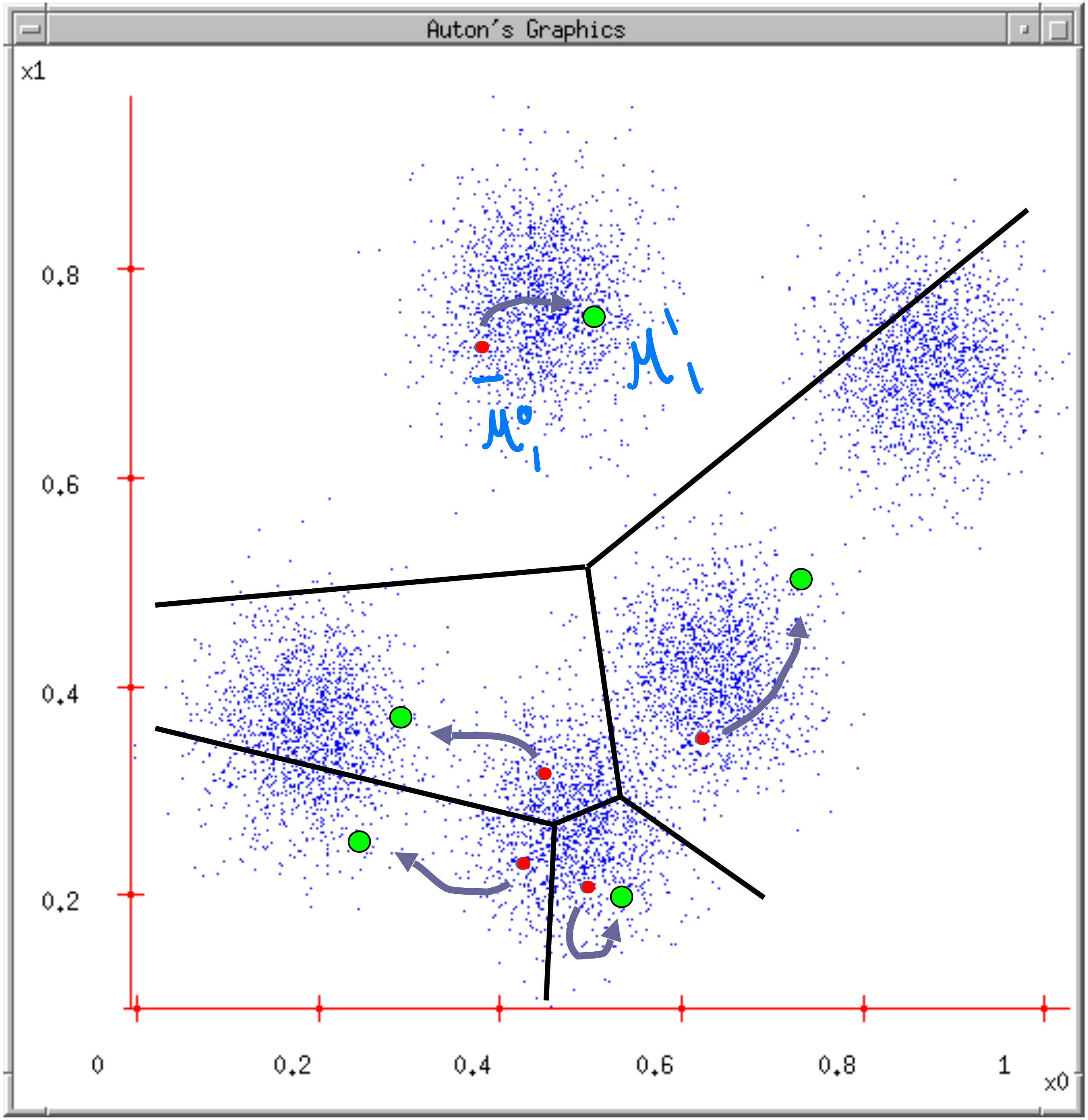
Lloyd's algorithm

1. Ask user how many clusters they'd like. *(e.g. k=5)*
2. Randomly guess k cluster Center locations
3. Each datapoint finds out which Center it's closest to. (Each center "owns" a set of datapoints)



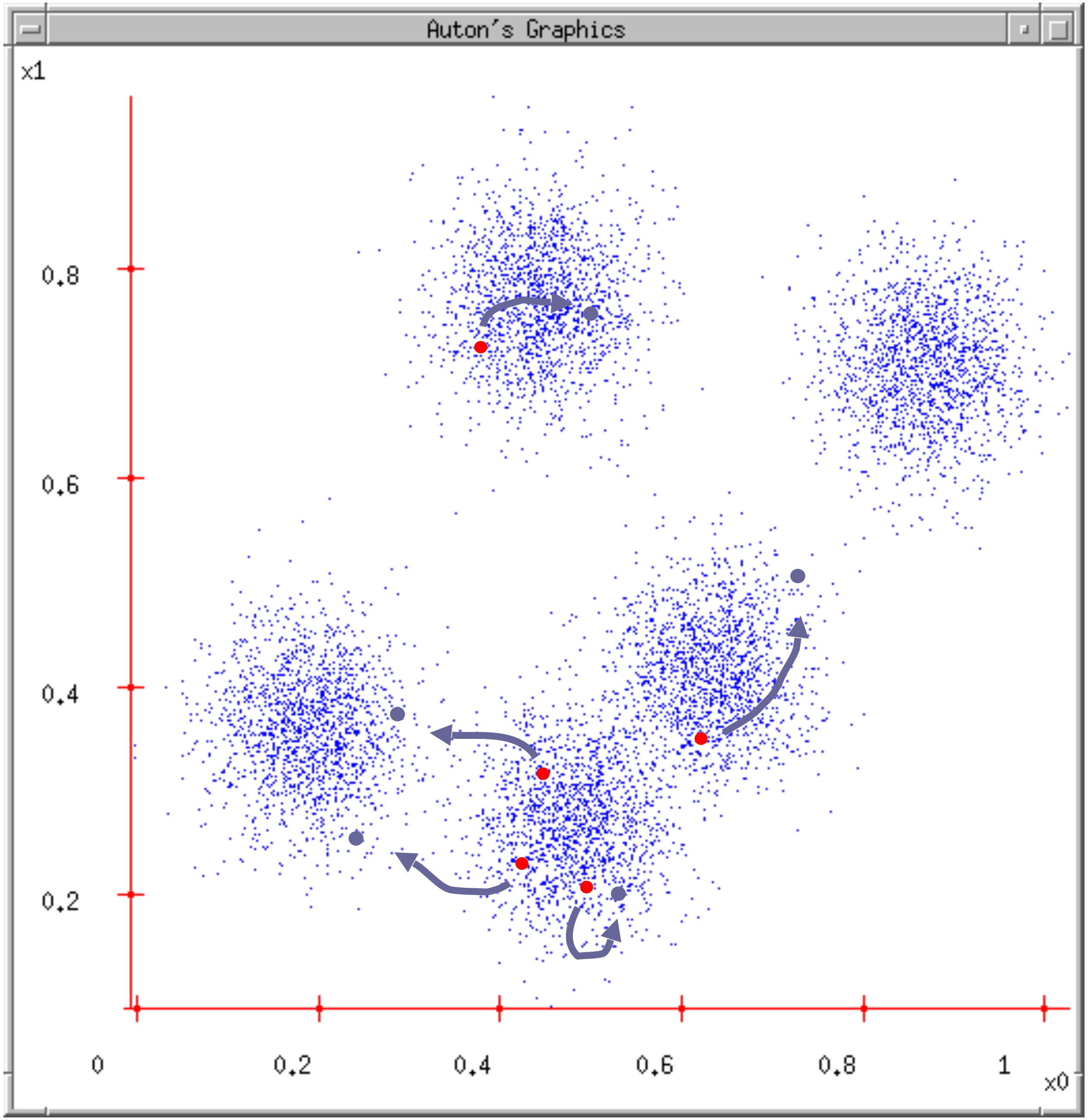
Lloyd’s algorithm

1. Ask user how many clusters they’d like. *(e.g. k=5)*
2. Randomly guess k cluster Center locations
3. Each datapoint finds out which Center it’s closest to.
4. Each Center finds the centroid of the points it owns



Lloyd's algorithm

1. Ask user how many clusters they'd like. *(e.g. k=5)*
2. Randomly guess k cluster Center locations
3. Each datapoint finds out which Center it's closest to.
4. Each Center finds the centroid of the points it owns…
5. …and jumps there
6. …Repeat until terminated!



Lloyd's algorithm

$$F(\mu, C) = \sum_{j=1}^{m} ||\mu_{C(j)} - x_j||^2$$

1. Ask user how many clusters they'd like. *(e.g. k=5)*
2. Randomly guess k cluster Center locations
3. Each datapoint finds out which Center it's closest to.
4. Each Center finds the centroid of the points it owns…
5. …and jumps there
6. …Repeat until terminated!

Lloyd's algorithm

$$F(\mu, C) = \sum_{j=1}^{m} ||\mu_{C(j)} - x_j||^2$$

1. Ask user how many clusters they'd like. *(e.g. k=5)*
2. Randomly guess k cluster Center locations
3. Each datapoint finds out which Center it's closest to.
4. Each Center finds the centroid of the points it owns…
5. …and jumps there
6. …Repeat until terminated!

$$C^{(t)}\underline{(j)} \leftarrow \arg\min_{\underline{i}} ||\mu_i - \boxed{x_j}||^2$$

$$\underline{\mu_i^{(t+1)}} \leftarrow \arg\min_{\mu} \underline{\sum_{j:C(j)=i} ||\mu - x_j||^2}$$

Does Lloyd's algorithm converge??? Part 1

$$\min_{\mu} \min_{C} F(\mu, C) = \min_{\mu} \min_{C} \sum_{i=1}^{k} \sum_{j:C(j)=i} ||\mu_i - x_j||^2$$

First, fix **μ**

minimize w.r.t C

Does Lloyd’s algorithm converge??? Part 2

$$\min_{\mu}\min_{C} F(\mu, C) = \min_{\mu}\min_{C} \sum_{i=1}^{k} \sum_{j:C(j)=i} ||\mu_i - x_j||^2$$

First, fix **μ**

minimize w.r.t C

Then, fix C

minimize w.r.t **μ**

$F(\mu, C)$ decreases each step $\Rightarrow$ the algorithm doesn’t cycle

Only $\underline{\binom{n}{k}} \approx \underline{n^k}$ configuarations $\Rightarrow$ converges in finite # iterations

A cool application of k-means clustering: compression

Vector Quantization, Fisher Vectors

Vector Quantization (for compression)

1. Represent image as grid of patches
2. Run k-means on the patches to build code book
3. Represent each patch as a code word.

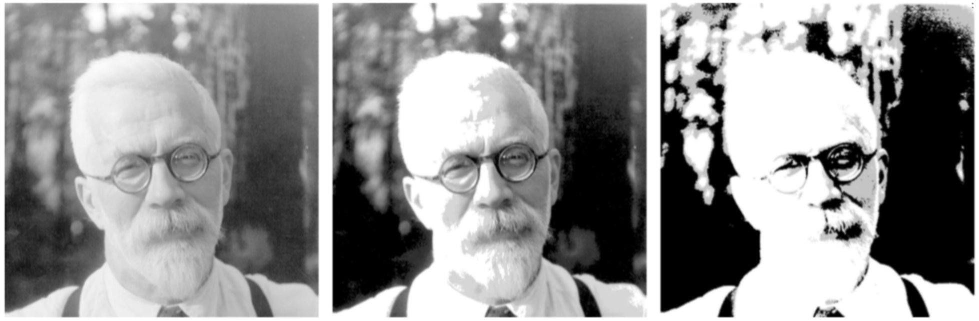


FIGURE 14.9. *Sir Ronald A. Fisher* (1890 – 1962) *was one of the founders of modern day statistics, to whom we owe maximum-likelihood, sufficiency, and many other fundamental concepts. The image on the left is a* 1024×1024 *grayscale image at* 8 *bits per pixel. The center image is the result of* 2 × 2 *block VQ, using* 200 *code vectors, with a compression rate of* 1.9 *bits/pixel. The right image uses only four code vectors, with a compression rate of* 0.50 *bits/pixel*

Vector Quantization, Fisher Vectors

Vector Quantization (for compression)

1. Represent image as grid of patches
2. Run k-means on the patches to build code book

   (k = # of codewords, center is code!)
3. Represent each patch as a code word.

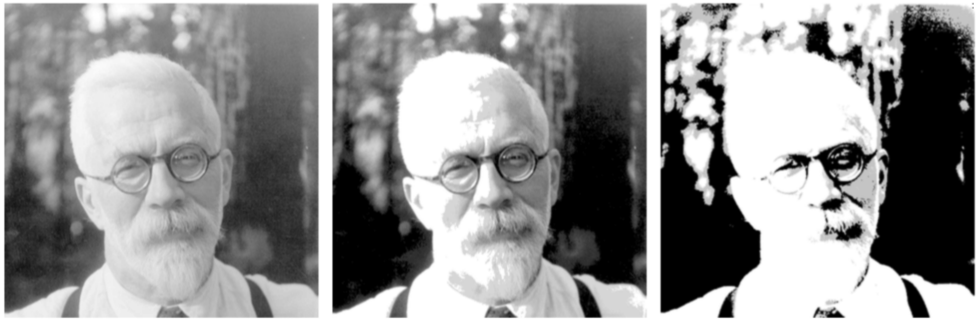
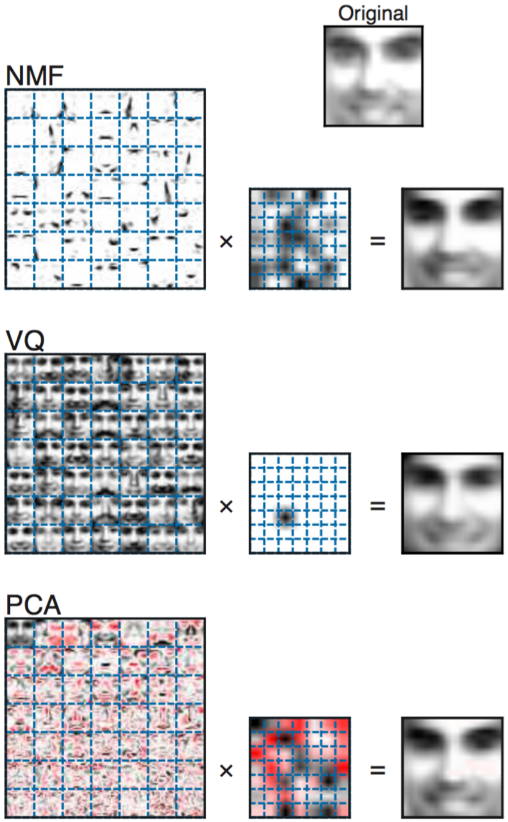


FIGURE 14.9. *Sir Ronald A. Fisher* (1890 – 1962) *was one of the founders of modern day statistics, to whom we owe maximum-likelihood, sufficiency, and many other fundamental concepts. The image on the left is a* 1024×1024 *grayscale image at* 8 *bits per pixel. The center image is the result of* 2×2 *block VQ, using* 200 *code vectors, with a compression rate of* 1.9 *bits/pixel. The right image uses only four code vectors, with a compression rate of* 0.50 *bits/pixel*

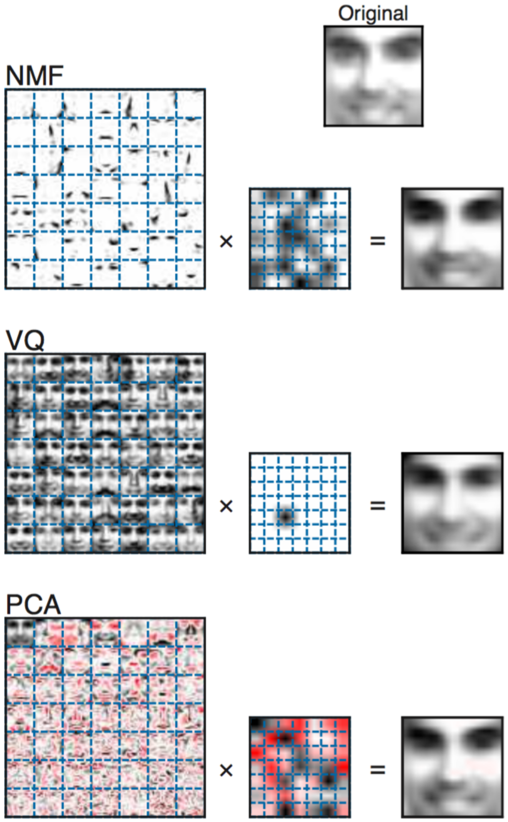


Vector Quantization, Fisher Vectors

Vector Quantization (for compression)

1. Represent image as grid of patches
2. Run k-means on the patches to build code book

   (k = # of codewords, center is code!)
3. Represent each patch as a code word.



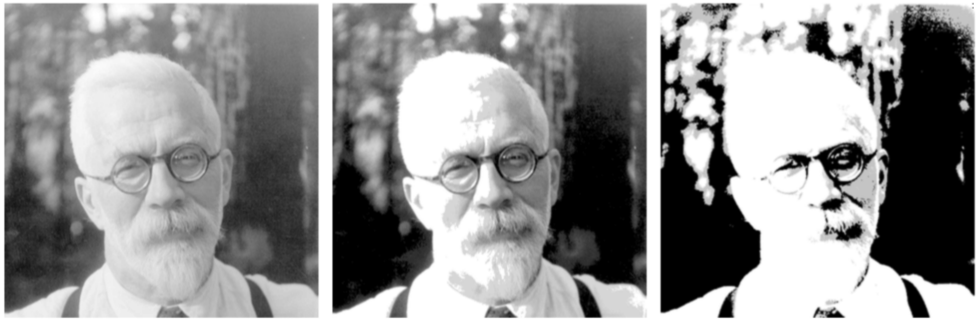


FIGURE 14.9. *Sir Ronald A. Fisher (1890 – 1962) was one of the founders of modern day statistics, to whom we owe maximum-likelihood, sufficiency, and many other fundamental concepts. The image on the left is a* 1024×1024 *grayscale image at* 8 *bits per pixel. The center image is the result of* 2 × 2 *block VQ, using* 200 *code vectors, with a compression rate of* 1.9 *bits/pixel. The right image uses only four code vectors, with a compression rate of* 0.50 *bits/pixel*

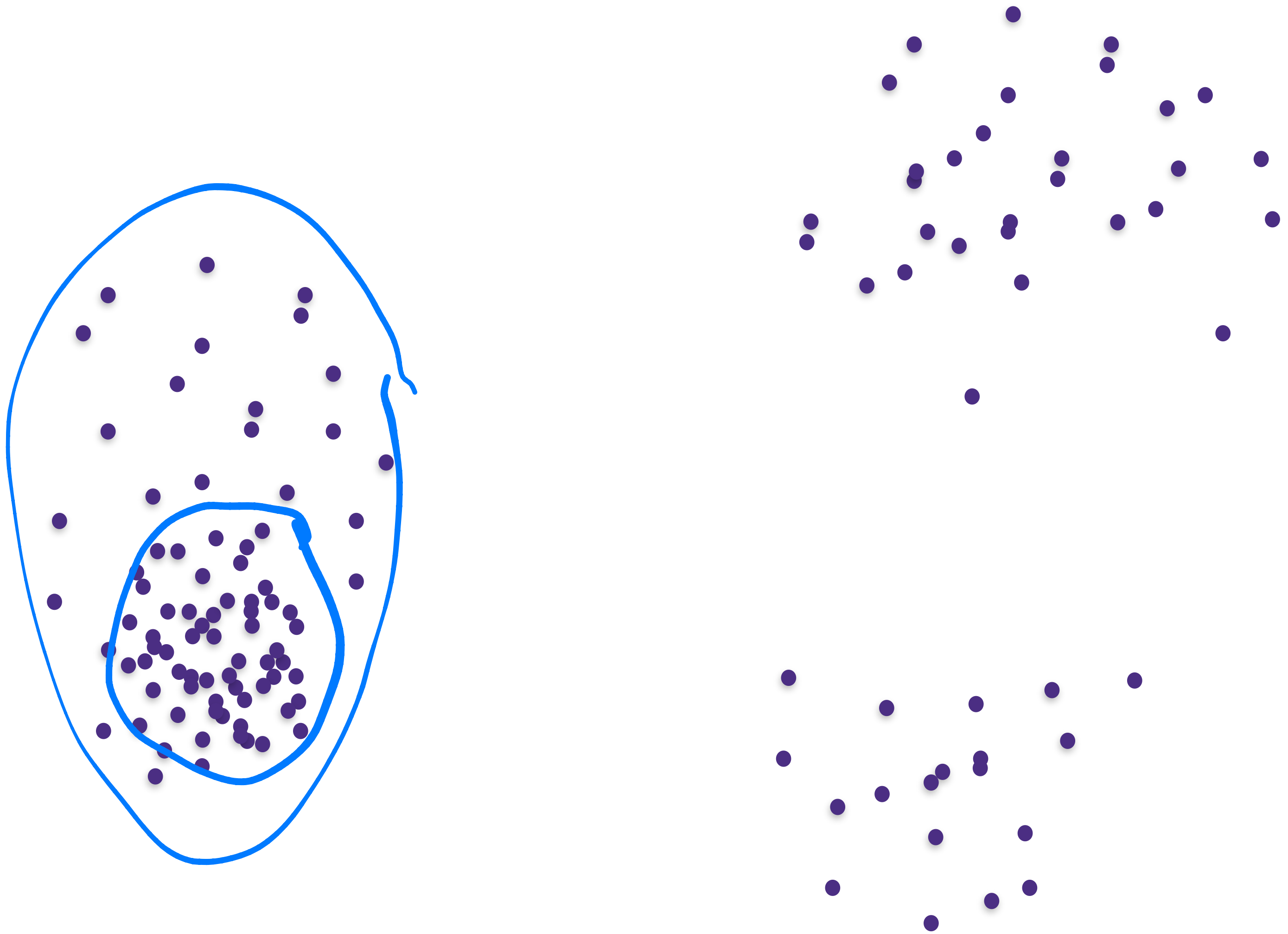
Coates, Ng, *Learning Feature Representations with K-means*, 2012

When to use K-means, or something else?

What sort of groupings are desired?

- Nonoverlapping, similar diameter clusters: k-means may work well
- Otherwise, might want another objective function (spectral, k-median, k-mode, …)

One bad case for k-means



K-means summary

A clustering objective

- minimize average L2 distance to centers of clusters

Lloyd’s algorithm: a greedy heuristic for minimizing it

- Will converge in finite time, may not find global minimum

Good for finding similar width, nonoverlapping clusters

Sensitive to initial center selection, and random may not be the best a priori

- See k-means++, *The Advantages of Careful Seeding,* Arthur and Vassilvitskii

Principal Component Analysis

UNIVERSITY of WASHINGTON

Linear projections

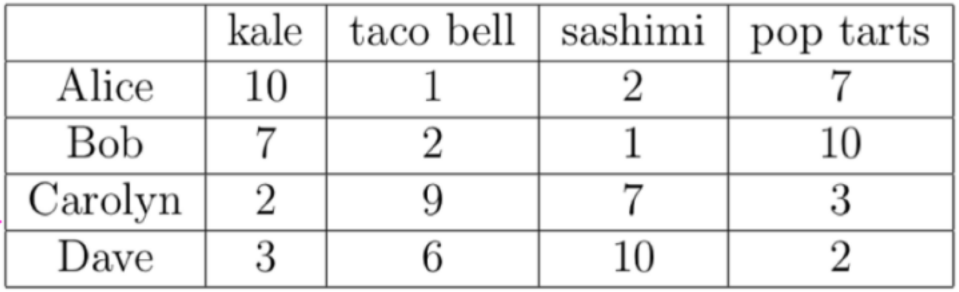
Given $\overbrace{x_1, \ldots, x_n}^{X \in \mathbb{R}^{n \times d}} \in \mathbb{R}^d$, for $\underline{q \ll d}$ find a compressed representation with $\lambda_1, \ldots, \lambda_n \in \mathbb{R}^q$ such that $\underline{x_i \approx \mu + \mathbf{V}_q \lambda_i}$ and $\mathbf{V}_q^T \mathbf{V}_q = I$

$$\min_{\mu, \mathbf{V}_q, \{\lambda_i\}_i} \sum_{i=1}^n \|x_i - \mu - \mathbf{V}_q \lambda_i\|_2^2 \longrightarrow \mu = \text{avg vec}(X)$$

PCA

Data dependent dimensionality reduction

- Useful for
  - Visualization
  - Interpretation
  - Compression
  - Understanding “intrinsic dimension”



	kale	taco bell	sashimi	pop tarts
Alice	10	1	2	7
Bob	7	2	1	10
Carolyn	2	9	7	3
Dave	3	6	10	2

Figure credit:
Karlin
Roughgarden + Va
Benedetto
Novembre et al
Alex Williams
Sandipan Dey
Victor Lavenko

PCA

Claim:

Each row can be expressed approximately as

$$\boxed{x_i \approx \bar{x} + \underline{a_{i1}\vec{v_1}} + \underline{a_{i2}\vec{v_2}}}$$

$$\left[\begin{array}{l} \vec{v_1} = [3 \quad -3 \quad -3 \quad 3] \\ \vec{v_2} = [1 \quad -1 \quad 1 \quad -1] \end{array}\right.$$

Feature1 Feature n

		kale	taco bell	sashimi	pop tarts
x_1	Alice	10	1	2	7
	Bob	7	2	1	10
	Carolyn	2	9	7	3
x_n	Dave	3	6	10	2

$$\bar{x} = [\underset{\uparrow}{5.5} \quad \underset{\uparrow}{4.5} \quad 5 \quad 5.5]$$

Figure credit:

Karlin

Roughgarden + Va

Benedetto

Novembre et al

Alex Williams

Sandipan Dey

Victor Lavenko

PCA in one dimension

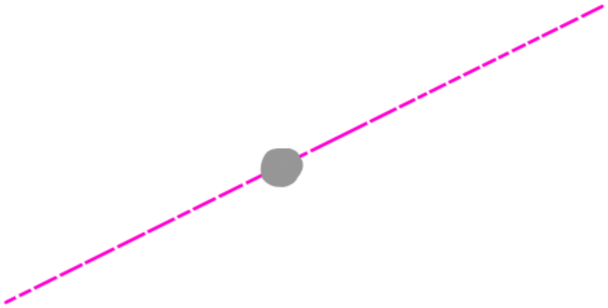
Goal: find a k < d-dimensional representation of X

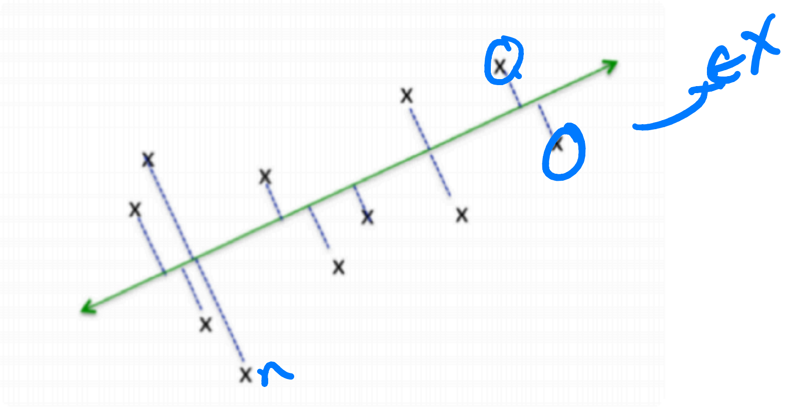
For k = 1:

Choose $\vec{v} \in \mathbb{R}^d, ||v|| = 1$

to minimize

$$\frac{1}{n}\sum_{i=1}^{n} dist(x_i, \text{line defined by } \vec{v})$$





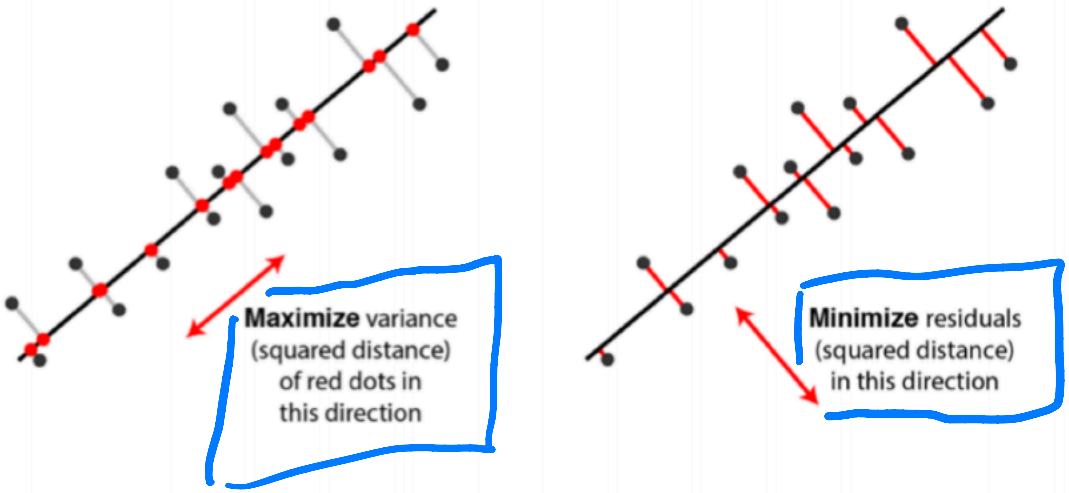
PCA in one dimension, 2 equivalent views

Goal: find a k < d-dimensional representation of X

For k = 1:

Choose $\vec{v} \in \mathbb{R}^d, ||v|| = 1$
to minimize

$$\frac{1}{n}\sum_{i=1}^{n} dist(x_i, \text{line defined by } \vec{v})$$



Two equivalent views of principal component analysis.

PCA: a high-fidelity linear projection

Given $x_1, \ldots, x_n \in \mathbb{R}^d$, for $q \ll d$ find a compressed representation with $\lambda_1, \ldots, \lambda_n \in \mathbb{R}^q$ such that $x_i \approx \mu + \mathbf{V}_q \lambda_i$ and $\mathbf{V}_q^T \mathbf{V}_q = I$

$$\min_{\mu, \mathbf{V}_q, \{\lambda_i\}_i} \sum_{i=1}^{n} \|x_i - \mu - \mathbf{V}_q \lambda_i\|_2^2$$

mimizing avg dist^2

Fix $\mathbf{V}_q$ and solve for μ, λ_i:

$$\boxed{\begin{aligned} \mu &= \bar{x} = \tfrac{1}{n} \textstyle\sum_{i=1}^n x_i \quad \in \mathbb{R}^n \\ \lambda_i &= \mathbf{V}_q^T (x_i - \bar{x}) \quad \in \mathbb{R}^d \end{aligned}}$$

Which gives us:

$$\min_{\mathbf{V}_q} \sum_{i=1}^{N} ||(x_i - \bar{x}) - \mathbf{V}_q \mathbf{V}_q^T (x_i - \bar{x})||^2.$$

$\mathbf{V}_q \mathbf{V}_q^T$ is a *projection matrix* that minimizes error in basis of size q

PCA: a high-fidelity linear projection

$$\sum_{i=1}^{N} ||(x_i - \bar{x}) - \mathbf{V}_q \mathbf{V}_q^T (x_i - \bar{x})||_2^2$$

$$\boxed{\Sigma := \sum_{i=1}^{N} (x_i - \bar{x})(x_i - \bar{x})^T}$$

$$\mathbf{V}_q^T \mathbf{V}_q = I_q$$

$$\min_{\mathbf{V}_q} \sum_{i=1}^{N} ||(x_i - \bar{x}) - \mathbf{V}_q \mathbf{V}_q^T (x_i - \bar{x})||_2^2 = \boxed{\min_{\mathbf{V}_q} Tr(\Sigma) - Tr(\mathbf{V}_q^T \Sigma \mathbf{V}_q)}$$

→ $\min -Tr[V_q^T \Sigma V_q]$

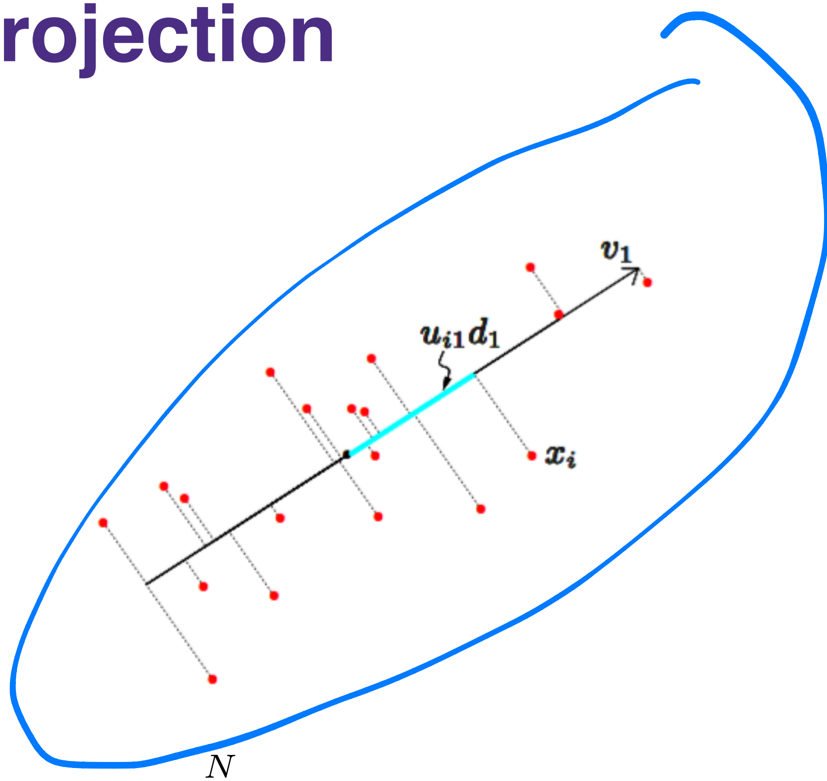
$\max Tr[V_q^T \Sigma V_q]$

Eigenvalue decomposition

$$\boxed{\mathbf{V}_q \text{ are the first } q \text{ eigenvectors of } \Sigma}$$

Minimize reconstruction error and capture the most variance in your data.

PCA: a high-fidelity linear projection



Given $x_i \in \mathbb{R}^d$ and some $q < d$ consider

$$\min_{\mathbf{V}_q} \sum_{i=1}^{N} ||(x_i - \bar{x}) - \mathbf{V}_q \mathbf{V}_q^T (x_i - \bar{x})||^2.$$

where $\mathbf{V}_q = [v_1, v_2, \ldots, v_q]$ is orthonormal:

$$\mathbf{V}_q^T \mathbf{V}_q = I_q$$

$\mathbf{V}_q$ are the first q eigenvectors of Σ

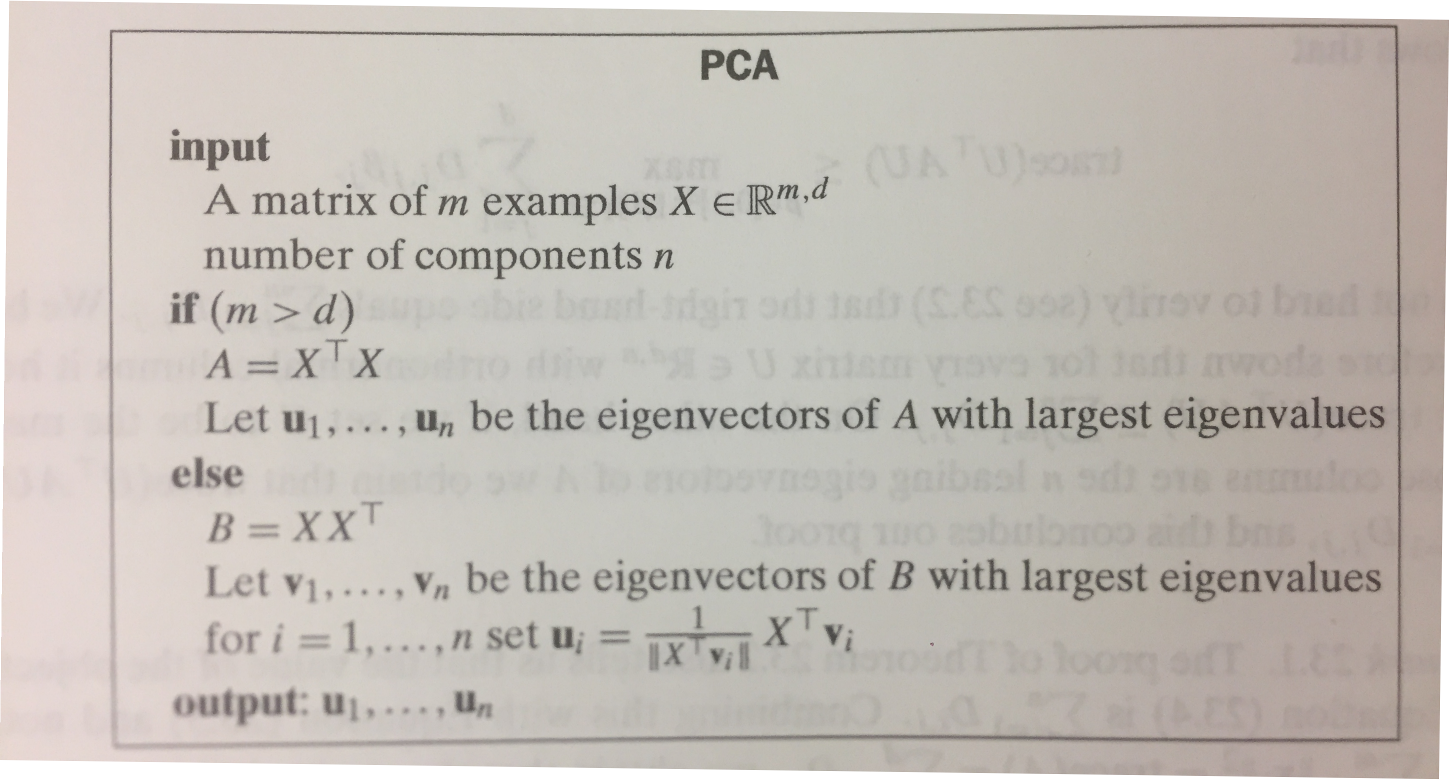
$$\Sigma := \sum_{i=1}^{N} (x_i - \bar{x})(x_i - \bar{x})^T$$

$\mathbf{V}_q$ are the first q *principal components*

Principal Component Analysis (PCA) projects $(\mathbf{X} - \mathbf{1}\bar{x}^T)$ down onto $\mathbf{V}_q$

$$(\mathbf{X} - \mathbf{1}\bar{x}^T)\mathbf{V}_q = \mathbf{U}_q \text{diag}(d_1, \ldots, d_q) \qquad \mathbf{U}_q^T \mathbf{U}_q = I_q$$

PCA Algorithm



```
PCA

input
    A matrix of $m$ examples $X \in \mathbb{R}^{m,d}$
    number of components $n$
if $(m > d)$
    $A = X^\top X$
    Let $\mathbf{u}_1, \ldots, \mathbf{u}_n$ be the eigenvectors of $A$ with largest eigenvalues
else
    $B = XX^\top$
    Let $\mathbf{v}_1, \ldots, \mathbf{v}_n$ be the eigenvectors of $B$ with largest eigenvalues
    for $i = 1, \ldots, n$ set $\mathbf{u}_i = \frac{1}{\|X^\top \mathbf{v}_i\|} X^\top \mathbf{v}_i$
output: $\mathbf{u}_1, \ldots, \mathbf{u}_n$
```

How do we compute the principal components?

1. Power iteration
2. **Solving for a singular value decomposition (SVD)**

Singular Value Decomposition (SVD)

Theorem (SVD): Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ with rank $r \leq \min\{m, n\}$. Then $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T$ where $\mathbf{S} \in \mathbb{R}^{r \times r}$ is diagonal with positive entries, $\mathbf{U}^T\mathbf{U} = I$, $\mathbf{V}^T\mathbf{V} = I$.

$$\mathbf{A}^T\mathbf{A}v_i =$$

$$\mathbf{A}\mathbf{A}^T u_i =$$

Singular Value Decomposition (SVD)

Theorem (SVD): Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ with rank $r \leq \min\{m, n\}$. Then $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T$ where $\mathbf{S} \in \mathbb{R}^{r \times r}$ is diagonal with positive entries, $\mathbf{U}^T\mathbf{U} = I$, $\mathbf{V}^T\mathbf{V} = I$.

$$\mathbf{A}^T\mathbf{A}v_i = \mathbf{S}_{i,i}^2 v_i$$

$$\mathbf{A}\mathbf{A}^T u_i = \mathbf{S}_{i,i}^2 u_i$$

$\mathbf{V}$ are the first r eigenvectors of $\mathbf{A}^T\mathbf{A}$ with eigenvalues $\mathrm{diag}(\mathbf{S})$
$\mathbf{U}$ are the first r eigenvectors of $\mathbf{A}\mathbf{A}^T$ with eigenvalues $\mathrm{diag}(\mathbf{S})$

Computational complexity of SVD

Theorem (SVD): Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ with rank $r \leq \min\{m, n\}$. Then $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T$ where $\mathbf{S} \in \mathbb{R}^{r \times r}$ is diagonal with positive entries, $\mathbf{U}^T\mathbf{U} = I$, $\mathbf{V}^T\mathbf{V} = I$.

at most r singular values

irrelevant | n - m | last columns of U

Computing the remaining economy-sized SVD takes time O(n m r)

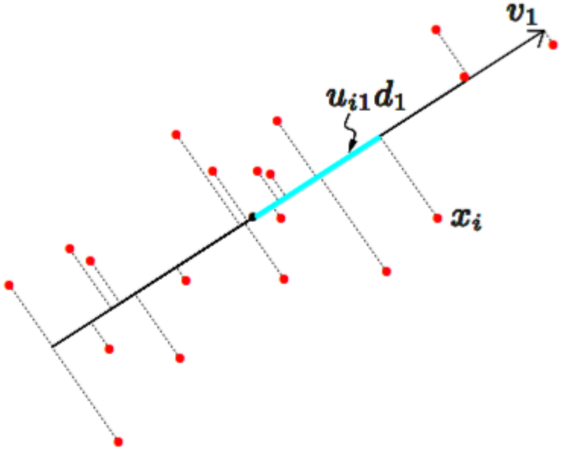
Linear projections

Given $x_i \in \mathbb{R}^d$ and some $q < d$ consider

$$\min_{\mathbf{V}_q} \sum_{i=1}^{N} ||(x_i - \bar{x}) - \mathbf{V}_q \mathbf{V}_q^T (x_i - \bar{x})||^2.$$

where $\mathbf{V}_q = [v_1, v_2, \dots, v_q]$ is orthonormal:

$$\mathbf{V}_q^T \mathbf{V}_q = I_q$$



$\mathbf{V}_q$ are the first q eigenvectors of Σ

$\mathbf{V}_q$ are the first q *principal components*

$$\Sigma := \sum_{i=1}^{N} (x_i - \bar{x})(x_i - \bar{x})^T$$

Principal Component Analysis (PCA) projects $(\mathbf{X} - \mathbf{1}\bar{x}^T)$ down onto $\mathbf{V}_q$

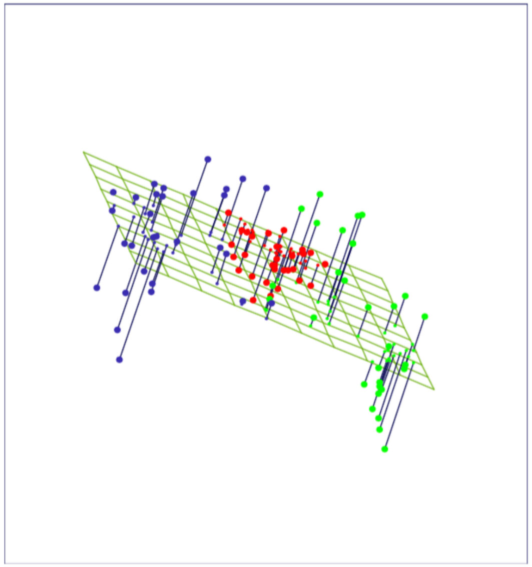
$$(\mathbf{X} - \mathbf{1}\bar{x}^T)\mathbf{V}_q = \mathbf{U}_q \text{diag}(d_1, \dots, d_q) \qquad \mathbf{U}_q^T \mathbf{U}_q = I_q$$

Singular Value Decomposition defined as

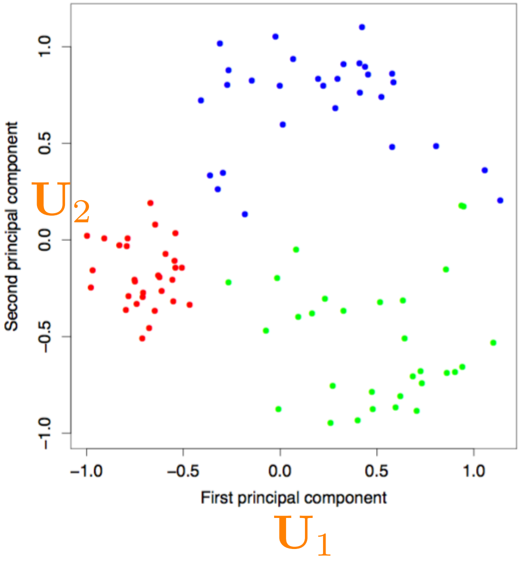
$$\mathbf{X} - \mathbf{1}\bar{x}^T = \mathbf{U}\mathbf{S}\mathbf{V}^T$$

Dimensionality reduction

$\mathbf{V}_q$ are the first q eigenvectors of Σ and *SVD* $\mathbf{X} - \mathbf{1}\bar{x}^T = \mathbf{U}\mathbf{S}\mathbf{V}^T$



$\mathbf{X} - \mathbf{1}\bar{x}^T$

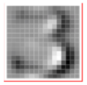
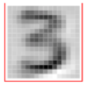


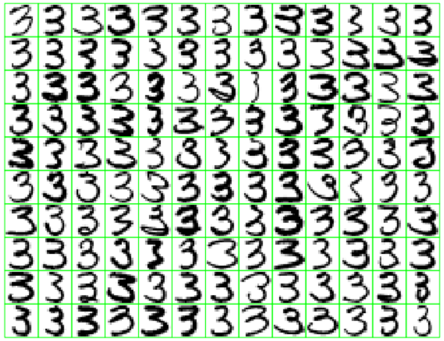
Dimensionality reduction

$\mathbf{V}_q$ are the first q eigenvectors of Σ and *SVD* $\mathbf{X} - \mathbf{1}\bar{x}^T = \mathbf{USV}^T$

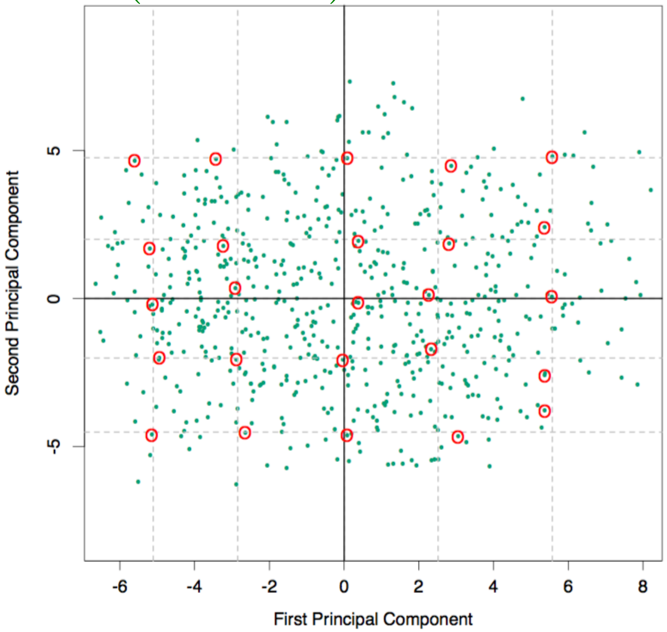
Handwritten 3's, 16x16 pixel image so that $x_i \in \mathbb{R}^{256}$

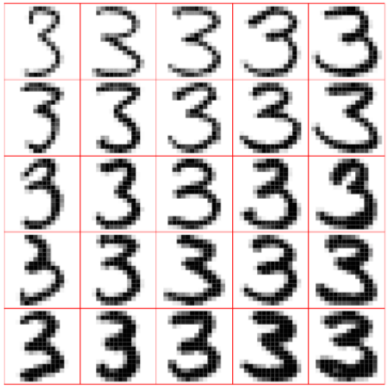
$$\hat{f}(\lambda) = \bar{x} + \lambda_1 v_1 + \lambda_2 v_2$$

$=$ $+ \lambda_1 \cdot$  $+ \lambda_2 \cdot$ .



$(\mathbf{X} - \mathbf{1}\bar{x}^T)\mathbf{V}_2 = \mathbf{U_2 S_2} \in \mathbb{R}^{n \times 2}$





$\text{diag}(\mathbf{S})$

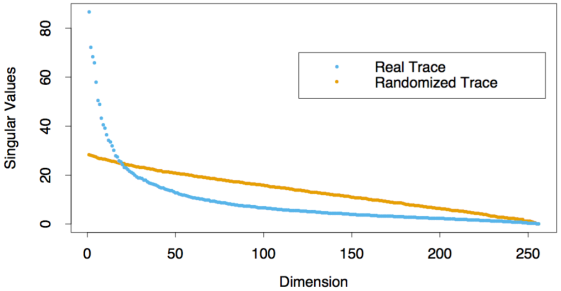
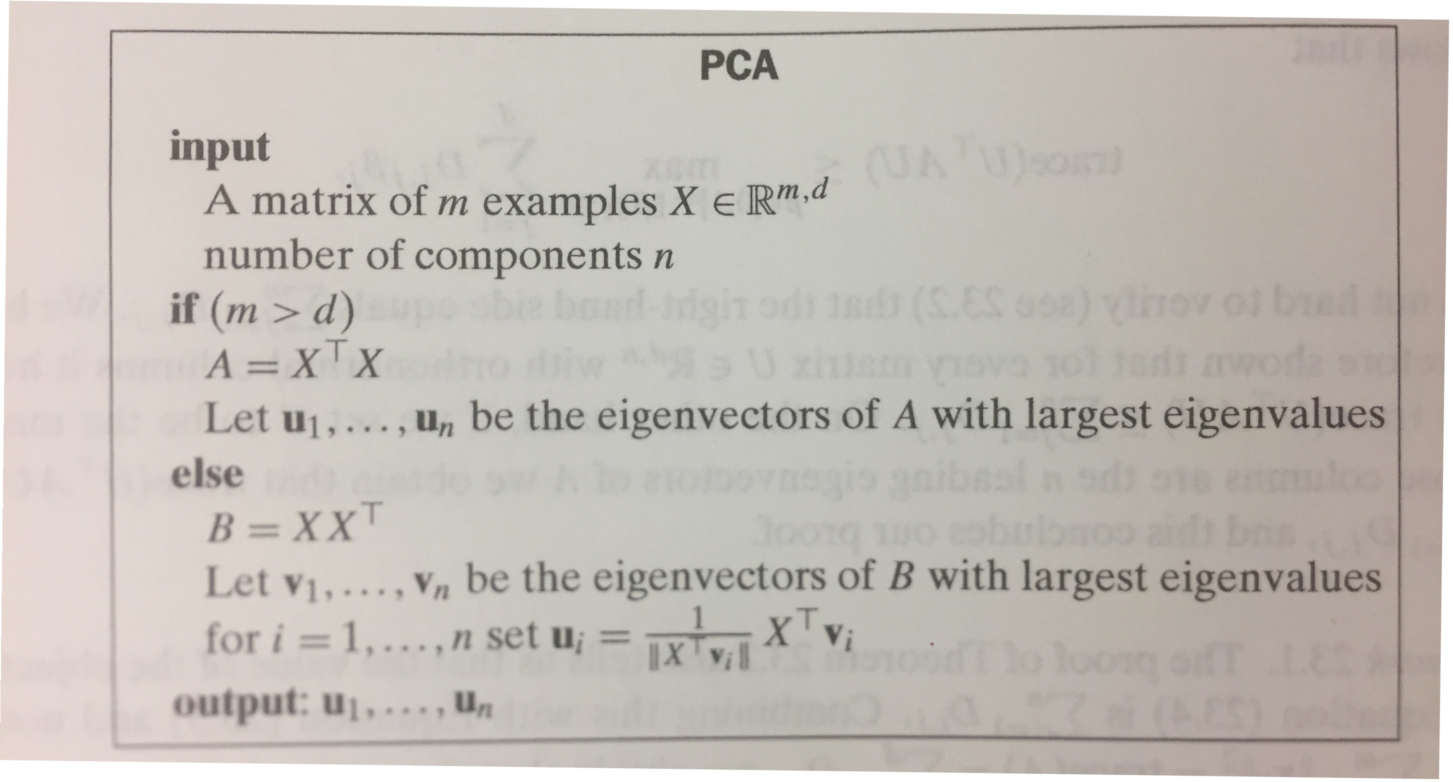


FIGURE 14.24. *The 256 singular values for the digitized threes, compared to those for a randomized version of the data (each column of* **X** *was scrambled).*

PCA Algorithm

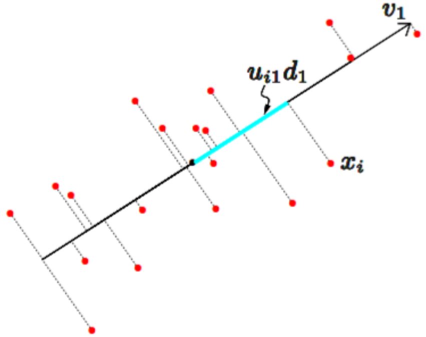


```
PCA

input
  A matrix of $m$ examples $X \in \mathbb{R}^{m,d}$
  number of components $n$
if $(m > d)$
  $A = X^\top X$
  Let $\mathbf{u}_1, \ldots, \mathbf{u}_n$ be the eigenvectors of $A$ with largest eigenvalues
else
  $B = X X^\top$
  Let $\mathbf{v}_1, \ldots, \mathbf{v}_n$ be the eigenvectors of $B$ with largest eigenvalues
  for $i = 1, \ldots, n$ set $\mathbf{u}_i = \frac{1}{\|X^\top \mathbf{v}_i\|} X^\top \mathbf{v}_i$
output: $\mathbf{u}_1, \ldots, \mathbf{u}_n$
```

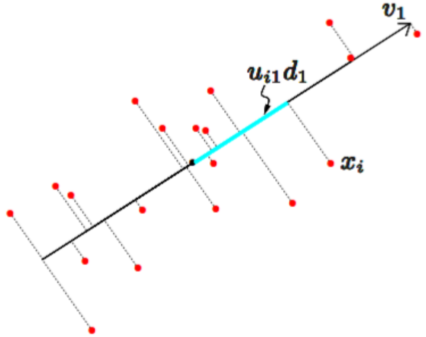
Power method - one at a time

$$\Sigma := \sum_{i=1}^{N} (x_i - \bar{x})(x_i - \bar{x})^T \qquad v_* = \arg\max_v v^T \Sigma v$$



Power method - one at a time

$$\Sigma := \sum_{i=1}^{N} (x_i - \bar{x})(x_i - \bar{x})^T \qquad v_* = \arg\max_v v^T \Sigma v$$



$$z_0 \sim \mathcal{N}(0, I)$$

Iterate: $z_{t+1} = \dfrac{\Sigma z_t}{\|\Sigma z_t\|_2}$

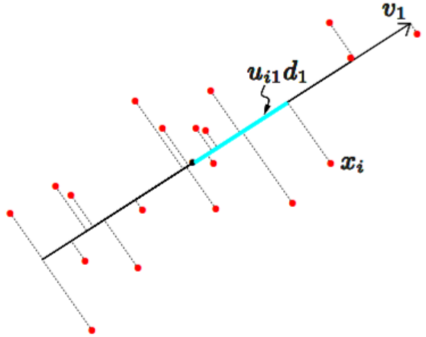
To analyze write: $\Sigma = \mathbf{V}\mathbf{D}\mathbf{V}^T \qquad z_t =: \mathbf{V}\alpha_t$

$$\alpha_{t+1} = \mathbf{V}^T z_{t+1} = \frac{\mathbf{V}^T \Sigma z_t}{\|\Sigma z_t\|} = \frac{\mathbf{D}\alpha_t}{\|\mathbf{D}\alpha_t\|} = \frac{\mathbf{D}^2\alpha_{t-1}}{\|\mathbf{D}^2\alpha_{t-1}\|} = \frac{\mathbf{D}^t\alpha_0}{\|\mathbf{D}^t\alpha_0\|}$$

$$\mathbf{D}^t = (\mathbf{D}_{1,1})^t(\mathbf{D}/\mathbf{D}_{1,1})^t \to (\mathbf{D}_{1,1})^t \mathbf{e}_1\mathbf{e}_1^T \text{ since } \mathbf{D}_{i,i}/\mathbf{D}_{1,1} < 1$$

Power method - one at a time

$$\Sigma := \sum_{i=1}^{N} (x_i - \bar{x})(x_i - \bar{x})^T \qquad v_* = \arg\max_v v^T \Sigma v$$



$$z_0 \sim \mathcal{N}(0, I)$$

Iterate: $z_{t+1} = \dfrac{\Sigma z_t}{\|\Sigma z_t\|_2}$

To analyze write: $\Sigma = \mathbf{VDV}^T \qquad z_t =: \mathbf{V}\alpha_t$