STAT 215A Fall 2020 Week 6

Theo Saarinen

Slides in part thanks to past 215A GSIs (Omer Ronen, James Duncan, Tiffany Tang, Zoe Vernon, Rebecca Barter, and others.)

Announcements

- Lab 2 released, due in two weeks **10/09 at 11:59pm**
- Will send out peer reviews today, these are due **10/09 at 11:59pm**
  - Each student will get the blinded version of 2 labs, then will need to fill out Google form with feedback: https://forms.gle/2iVwgvXV7tkrT4Nb8

Announcements

Reminders when submitting the homework + lab

- Submit the HW 2 by pushing *homework2.pdf* **inside** your lab2 folder.
- Please **do not** push the raw data with your submission, this slows down the grading process.
- Please submit your lab within a folder called lab2 within the github repo, if you are uploading through the desktop and don't know how to create a folder, see: https://stackoverflow.com/questions/12258399/how-do-i-create-a-folder-in-a-github-repository

Outline for today

- Choosing K for NMF
- Spectral clustering
- DBSCAN (time allowing)
- Lab 2 check-in

Nonnegative Matrix Factorization (NMF)

- Given a non-negative matrix $\mathbf{X}$, NMF solves

$$\underset{\mathbf{W}\geq 0, \mathbf{H}\geq 0}{\operatorname{argmin}} ||\mathbf{X} - \mathbf{W}\,\mathbf{H}||_F^2 = \sum_{i,j} (\mathbf{X}_{ij} - \mathbf{W}_i^\top \mathbf{H}_j)^2$$

- You can modify this to work for $\mathbf{X}$ with missing data (in R: `NNLM:nnmf()`[1]):

$$\underset{\mathbf{W}\geq 0, \mathbf{H}\geq 0}{\operatorname{argmin}} \sum_{\substack{(i,j) \\ \text{not missing}}} (\mathbf{X}_{ij} - \mathbf{W}_i^\top \mathbf{H}_j)^2$$

1. NNLM was archived on CRAN, so to install you need to use `devtools::install_github("linxihui/NNLM")`

Nonnegative Matrix Factorization (NMF)

Missing Data NMF: $\underset{\mathbf{W}\geq 0, \mathbf{H}\geq 0}{\text{argmin}} \sum_{\substack{(i,j) \\ \text{not missing}}} (\mathbf{X}_{ij} - \mathbf{W}_i^\top \mathbf{H}_j)^2$

Idea for choosing K:

- Randomly leave out entries from the data matrix $\mathbf{X}$
- For each potential choice of K:
  1. Apply NMF to the data with missing values: $\mathbf{W}_M$ and $\mathbf{H}_M$
  2. Impute the missing values of $\mathbf{X}$ using corresponding entries of $\mathbf{W}_M\mathbf{H}_M$
  3. Compute the imputation error (MSE of difference between imputed and observed values)
  4. Repeat many times and compute the mean and SE for this K
- Choose K by taking the minimum or using the 1-SE rule (Breiman, 1984)

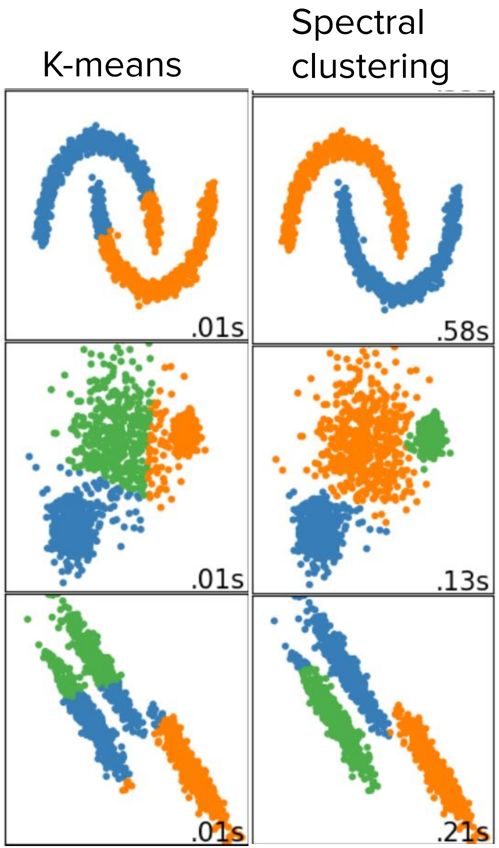
Spectral clustering: a “good” method

Advantages:

- Simple to implement
- Stable to underlying data generation mechanism

Disadvantages

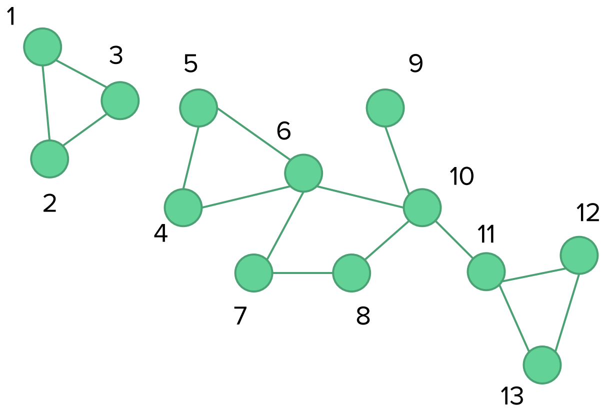
- Need to represent the data as a graph
- Still need to choose K
- Not advised for problems with large numbers of clusters



More details: http://people.csail.mit.edu/dsontag/courses/ml14/notes/Luxburg07_tutorial_spectral_clustering.pdf
Nice summary: https://eric-bunch.github.io/blog/spectral-clustering

Setup

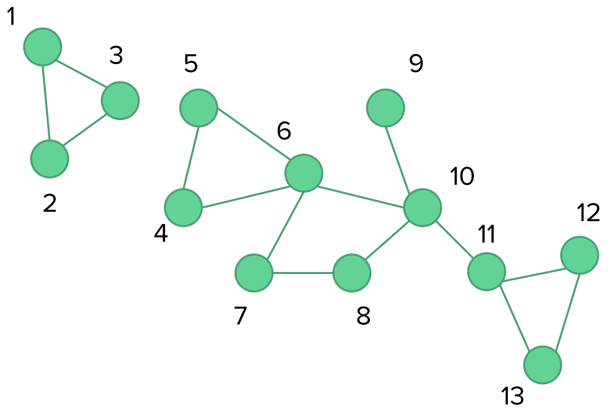
- Graph: $G = (E,V)$
- Weighted adjacency matrix: $W = (w_{ij})_{i,j=1,\dots,n}$
  - 0 on the diagonal
  - $w_{ij} = 0$ if there is no edge between v_i and v_j
- Diagonal degree matrix $D : D_{ii} = \sum_j W_{ij}$
- (Unnormalized) graph Laplacian: $L = D - W$



What can the graph Laplacian tell us?

- Let's try to find eigenvectors $Lx = \lambda x$

$$L = \begin{pmatrix} \sum_j w_{1j} & -w_{12} & -w_{13} & 0 & \cdots \\ -w_{12} & \sum_j w_{2j} & -w_{23} & 0 & \cdots \\ -w_{13} & -w_{23} & \sum_j w_{3j} & 0 & \cdots \\ 0 & 0 & 0 & \ddots & \\ \vdots & \vdots & \vdots & & \end{pmatrix}$$

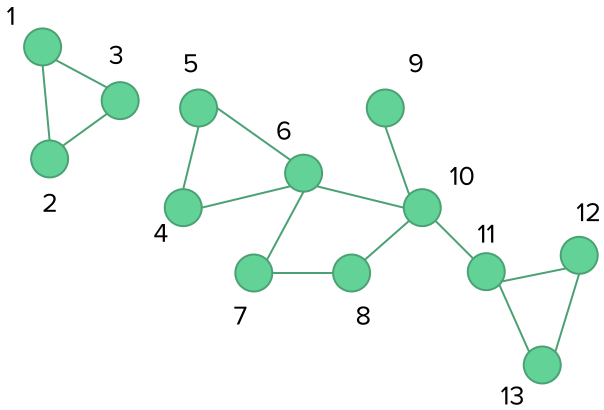


What can the graph Laplacian tell us?

- Let’s try to find eigenvectors $Lx = \lambda x$

$\mathbb{1}$, the vector with all entries equal to 1

$$L = \begin{pmatrix} \sum_j w_{1j} & -w_{12} & -w_{13} & 0 & \cdots \\ -w_{12} & \sum_j w_{2j} & -w_{23} & 0 & \cdots \\ -w_{13} & -w_{23} & \sum_j w_{3j} & 0 & \cdots \\ 0 & 0 & 0 & \ddots & \\ \vdots & \vdots & \vdots & & \end{pmatrix}$$



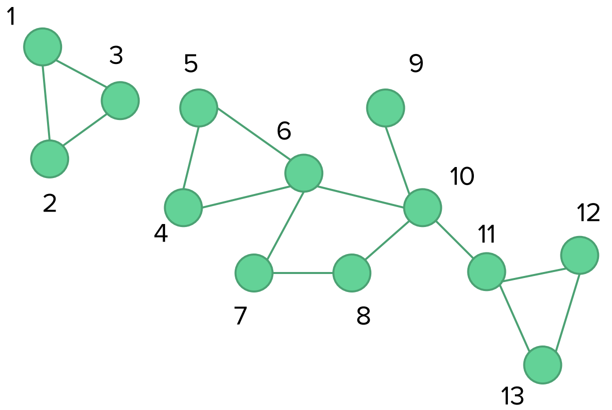
What can the graph Laplacian tell us?

- Let's try to find eigenvectors $Lx = \lambda x$

$\mathbb{1}$, the vector with all entries equal to 1

$$L = \begin{pmatrix} \sum_j w_{1j} & -w_{12} & -w_{13} & 0 & \cdots \\ -w_{12} & \sum_j w_{2j} & -w_{23} & 0 & \cdots \\ -w_{13} & -w_{23} & \sum_j w_{3j} & 0 & \cdots \\ 0 & 0 & 0 & \ddots & \\ \vdots & \vdots & \vdots & & \end{pmatrix}$$

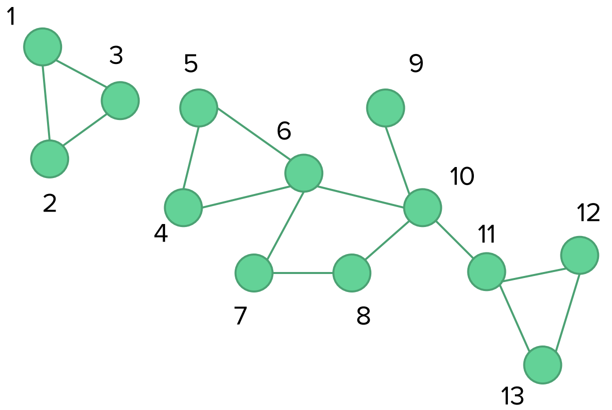
$$\begin{pmatrix} 1 & 1 & 1 & 0 \cdots 0 \end{pmatrix}^\top$$



What can the graph Laplacian tell us?

- Let's try to find eigenvectors $Lx = \lambda x$

$\mathbb{1}$, the vector with all entries equal to 1



$$L = \begin{pmatrix} \sum_j w_{1j} & -w_{12} & -w_{13} & 0 & \cdots \\ -w_{12} & \sum_j w_{2j} & -w_{23} & 0 & \cdots \\ -w_{13} & -w_{23} & \sum_j w_{3j} & 0 & \cdots \\ 0 & 0 & 0 & \ddots & \\ \vdots & \vdots & \vdots & & \end{pmatrix}$$

$(1 \quad 1 \quad 1 \quad 0 \cdots 0)^\top$ and also $(0 \quad 0 \quad 0 \quad 1 \cdots 1)^\top$

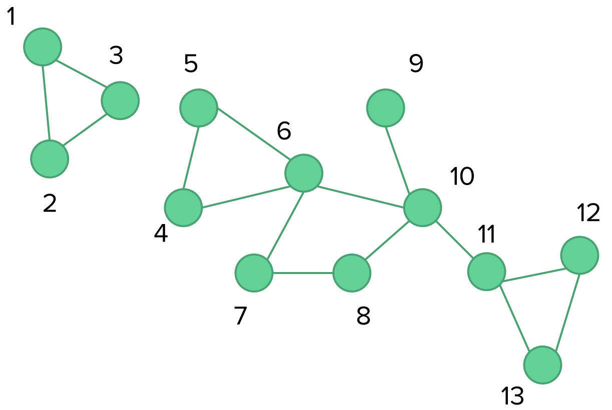
What can the graph Laplacian tell us?

- Let's try to find eigenvectors $Lx = \lambda x$

$\mathbb{1}$, the vector with all entries equal to 1

$$L = \begin{pmatrix} \sum_j w_{1j} & -w_{12} & -w_{13} & 0 & \cdots \\ -w_{12} & \sum_j w_{2j} & -w_{23} & 0 & \cdots \\ -w_{13} & -w_{23} & \sum_j w_{3j} & 0 & \cdots \\ 0 & 0 & 0 & \ddots & \\ \vdots & \vdots & \vdots & & \end{pmatrix}$$

$$(1 \quad 1 \quad 1 \quad 0 \cdots 0)^\top \text{ and also } (0 \quad 0 \quad 0 \quad 1 \cdots 1)^\top$$



$\lambda_1 = 0$ in these cases, so the multiplicity of the eigenvalue 0 tells us about the number of **connected components** in the graph.

What about the other eigenvectors of L?

- Fact[1]: $$\lambda_2 = \min_{x:||x||=1} x^\top M x$$

Second smallest eigenvalue

M symmetric

Constraint basically ensures this is perpendicular to the eigenvector corresponding to the smallest eigenvalue

1 See https://math.stackexchange.com/questions/1403920/second-smallest-eigenvalue-as-min-x-fracxtaxxtx for a proof

What about the other eigenvectors of L ?

- Fact[1]: $\lambda_2 = \min_{x:||x||=1} x^\top M x$

Second smallest eigenvalue

M symmetric

- Consider $x^\top L x$

1 See https://math.stackexchange.com/questions/1403920/second-smallest-eigenvalue-as-min-x-fracxtaxxtx for a proof

What about the other eigenvectors of L ?

- Fact[1]: $$\lambda_2 = \min_{x:||x||=1} x^\top M x$$

Second smallest eigenvalue

M symmetric

- Consider $$x^\top L x = \frac{1}{2} \sum_{i,j} w_i (x_i - x_j)^2$$

1 See https://math.stackexchange.com/questions/1403920/second-smallest-eigenvalue-as-min-x-fracxtaxxtx for a proof

What about the other eigenvectors of L ?

- Fact[1]: $$\lambda_2 = \min_{x:||x||=1} x^\top M x$$

Second smallest eigenvalue

M symmetric

- Consider $$x^\top L x = \frac{1}{2}\sum_{i,j} w_i (x_i - x_j)^2$$

- So, $$\lambda_2 = \min_{x:||x||=1} \sum_{i,j} w_{ij}(x_i - x_j)^2$$

1 See https://math.stackexchange.com/questions/1403920/second-smallest-eigenvalue-as-min-x-fracxtaxxtx for a proof

What about the other eigenvectors of L ?

$$\lambda_2 = \min_{x:\|x\|=1} \sum_{i,j} w_{ij}(x_i - x_j)^2$$

- Call the minimizer x^*

What about the other eigenvectors of L ?

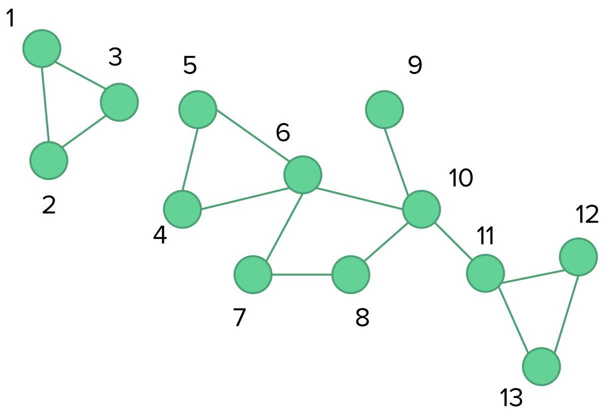
$$\lambda_2 = \min_{x:||x||=1} \sum_{i,j} w_{ij}(x_i - x_j)^2$$

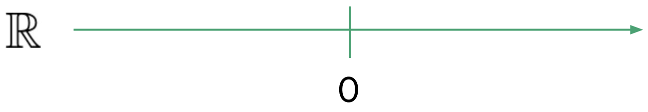
- Call the minimizer x^*
- Another Fact: The eigenvectors with distinct eigenvalues of a real symmetric matrix are orthogonal. $\sum_{i=1}^{n} x_i^* = 0$

What about the other eigenvectors of L?

$$\lambda_2 = \min_{x:||x||=1} \sum_{i,j} w_{ij}(x_i - x_j)^2$$

- Call the minimizer x^*
- Another Fact: The eigenvectors with distinct eigenvalues of a real symmetric matrix are orthogonal. $\sum_{i=1}^{n} x_i^* = 0$

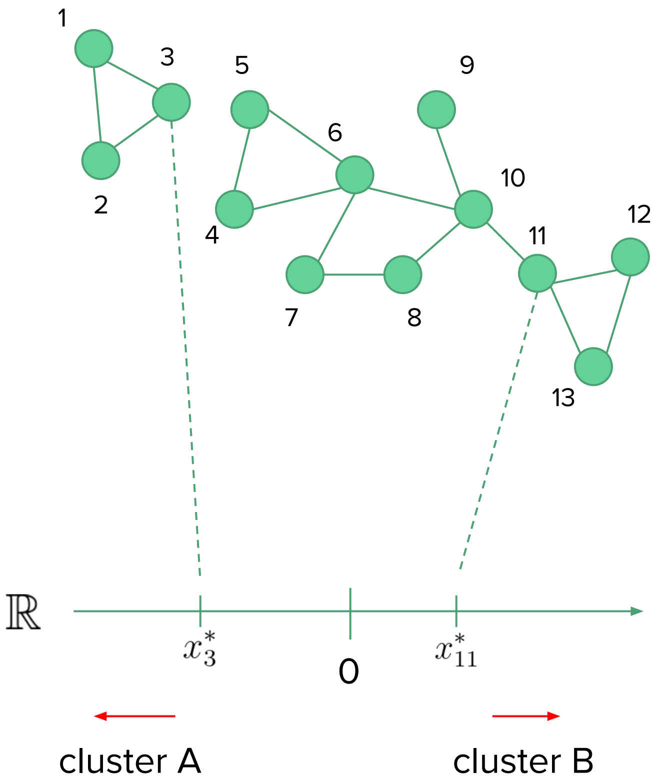




What about the other eigenvectors of L ?

$$\lambda_2 = \min_{x:\|x\|=1} \sum_{i,j} w_{ij}(x_i - x_j)^2$$

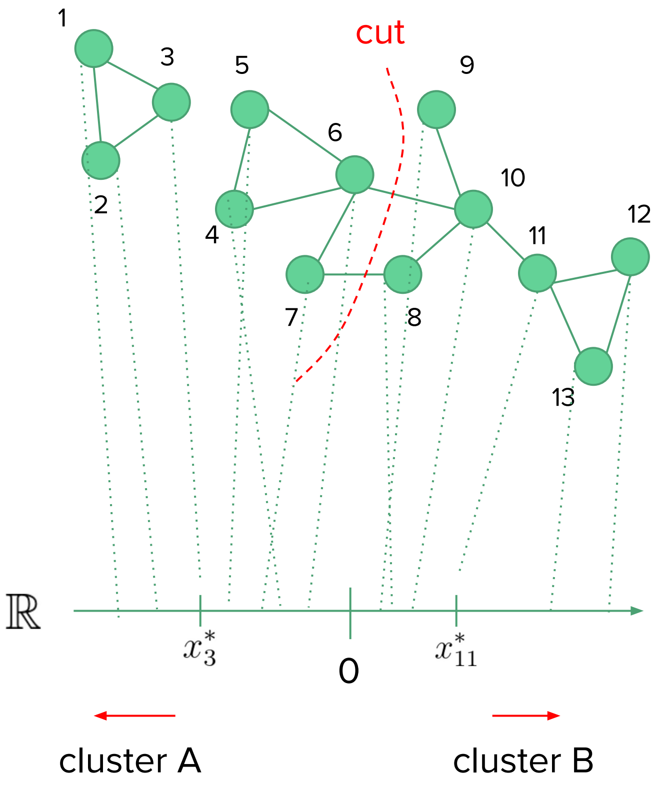
- Call the minimizer x^*
- Another Fact: The eigenvectors with distinct eigenvalues of a real symmetric matrix are orthogonal. $\sum_{i=1}^{n} x_i^* = 0$
- Look at where the values of x^* fall on the real line.



What about the other eigenvectors of L ?

$$\lambda_2 = \min_{x:\|x\|=1} \sum_{i,j} w_{ij}(x_i - x_j)^2$$

- Call the minimizer x^*
- Another Fact: The eigenvectors with distinct eigenvalues of a real symmetric matrix are orthogonal. $\sum_{i=1}^{n} x_i^* = 0$
- Look at where the values of x^* fall on the real line.
- Cut at the point that separates the observations with negative values from those with positive values in x^*



Ratio Cut vs Normalized Cut

Given a similarity graph with adjacency matrix W, the simplest and most direct way to construct a partition of the graph is to solve the mincut problem. To define it, please recall the notation $W(A, B) := \sum_{i\in A, j\in B} w_{ij}$ and $\overline{A}$ for the complement of A. For a given number k of subsets, the mincut approach simply consists in choosing a partition $A_1, \ldots, A_k$ which minimizes

$$\mathrm{cut}(A_1, \ldots, A_k) := \frac{1}{2}\sum_{i=1}^{k} W(A_i, \overline{A}_i).$$

RatioCut and **NormalizedCut** objectives differ in the factor we use to measure the size of a set of vertices

$$\mathrm{RatioCut}(A_1, \ldots, A_k) := \frac{1}{2}\sum_{i=1}^{k} \frac{W(A_i, \overline{A}_i)}{|A_i|} = \sum_{i=1}^{k} \frac{\mathrm{cut}(A_i, \overline{A}_i)}{|A_i|}$$

$$\mathrm{Ncut}(A_1, \ldots, A_k) := \frac{1}{2}\sum_{i=1}^{k} \frac{W(A_i, \overline{A}_i)}{\mathrm{vol}(A_i)} = \sum_{i=1}^{k} \frac{\mathrm{cut}(A_i, \overline{A}_i)}{\mathrm{vol}(A_i)}.$$

Normalization

- In the simple example above, we used the **unnormalized** graph Laplacian, which solves an approximation of the **RatioCut** objective.
- We could instead use the **normalized** graph Laplacian:
    - Remember D is the diagonal degree matrix: $D : D_{ii} = \sum_j W_{ij}$

$$L_{sym} = I - D^{-1/2} A D^{-1/2}$$

    - Leads to an approximate solution of the **NormalizedCut** objective
    - Theoretically analyzed (Sarkar and Bickel, 2015)

Representing your data as a graph

- ε **- neighborhood graph**: connect all points whose pairwise distances are smaller than ε
- k**-nearest neighbors graph**: connect v_i and v_j if they v_j is one of v_i's nearest neighbors
  - Variant: only connect if they are mutually nearest neighbors
- **Fully connected with similarity function**: compute the pairwise similarities or distances between observations
  - Example: Gaussian similarity $\exp\{-\|x_i - x_j\|^2/2\sigma^2\}$

Spectral clustering algorithm

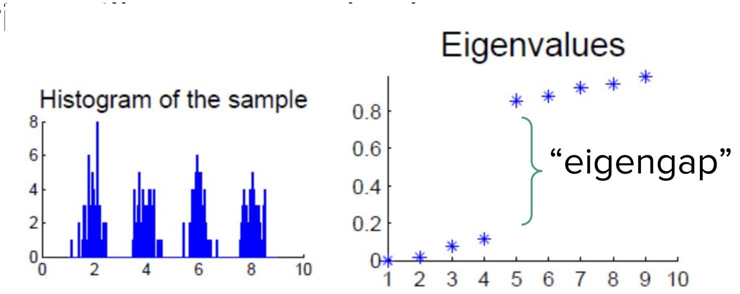
For a given choice of number of clusters K:

1. Construct a similarity graph and its weighted adjacency matrix W
2. Compute the normalized Laplacian L_{sym}
3. Compute the first K eigenvectors of L_{sym} and collect as the columns of a matrix U
4. Normalized the rows of U to have norm 1
5. Cluster the rows of U using K-means or your preferred “traditional” clustering algorithm

In R: `kernlab::specc()`

Choice of K

- Open area of research
- Method-specific tools:
  - K-means, hierarchical clustering, spectral clusteri
  - NMF: cross-validation / missing data imputation
  - Spectral clustering: Eigengap heuristic



- More general idea/tool: stability
  - Data perturbations (e.g. bootstrap, subsampling)
  - Algorithmic perturbations (e.g., random initialization)
- Still, it’s a tough problem…

But what if we didn't have to choose K!?

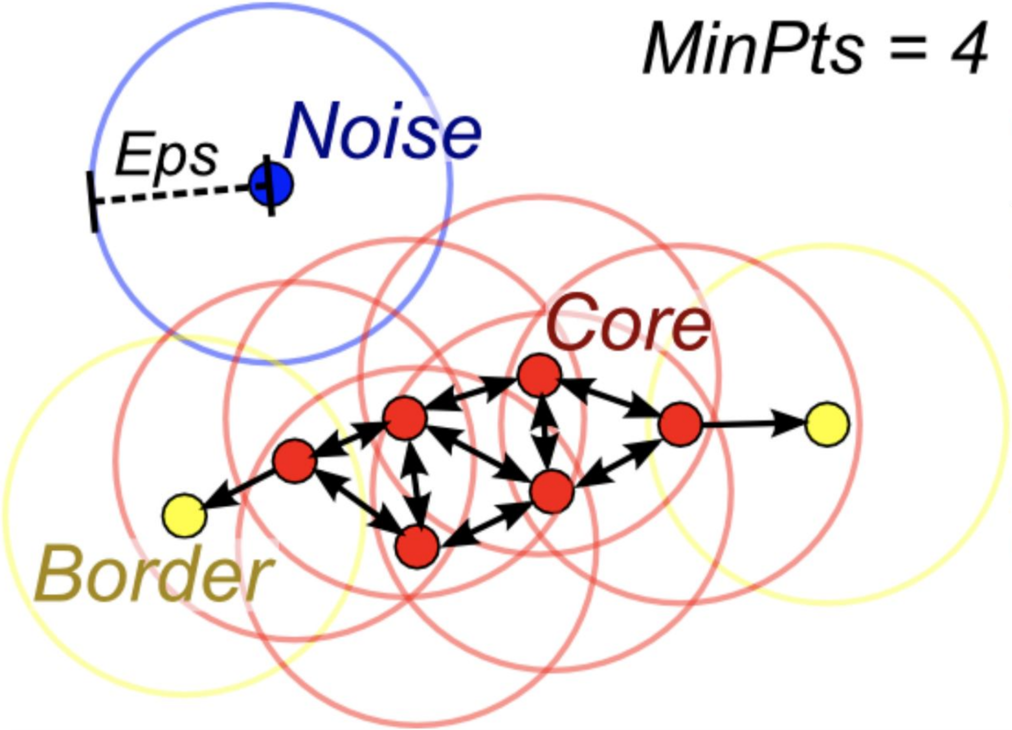
DBSCAN (Ester, Martin, et al. 1996)

- Density-based clustering
- **Idea**: group together points that are closely packed together (points with many nearby neighbors) while marking points that lie in low-density regions as outliers
- Choose (two?) parameters:
  - ε: how close points should be to each other to be in the same cluster (so we need a distance metric)
  - minPts = minimum number of points require to form a dense region

Source: https://medium.com/@elutins/dbscan-what-is-it-when-to-use-it-how-to-use-it-8bd506293818

The slides on DBSCAN thanks to Tiffany Tang

DBSCAN



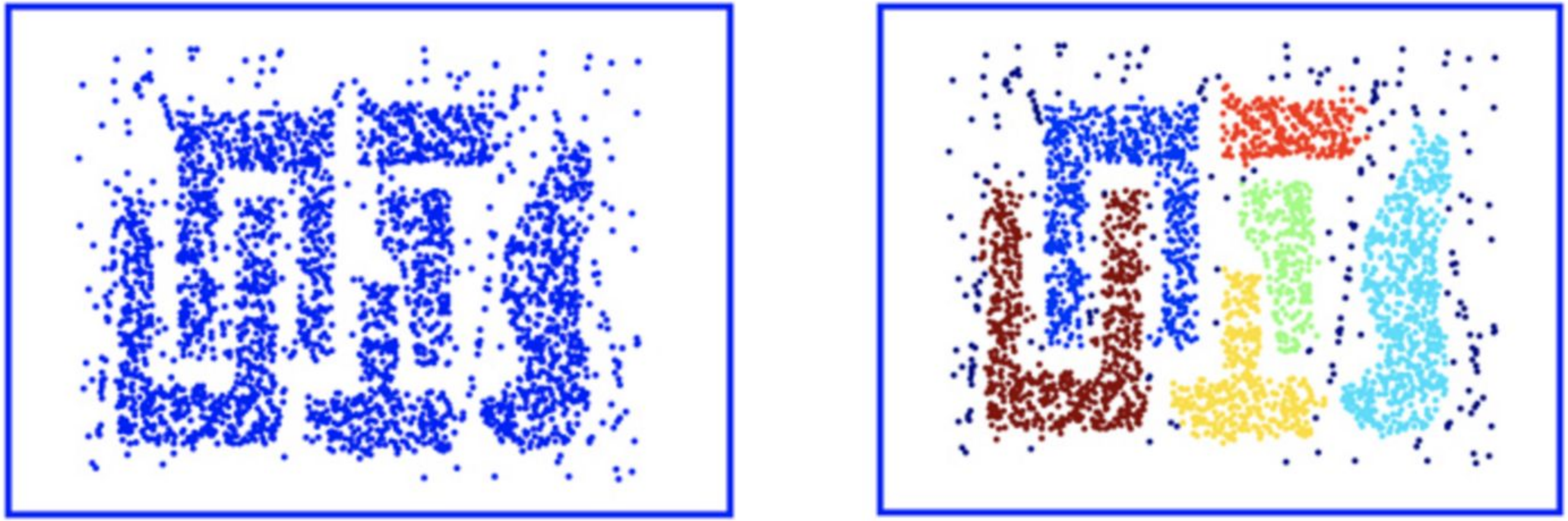
Red: Core Points

Yellow: Border points. Still part of the cluster because it's within epsilon of a core point, but not does not meet the min_points criteria

Blue: Noise point. Not assigned to a cluster

Source: https://medium.com/@elutins/dbscan-what-is-it-when-to-use-it-how-to-use-it-8bd506293818

DBSCAN

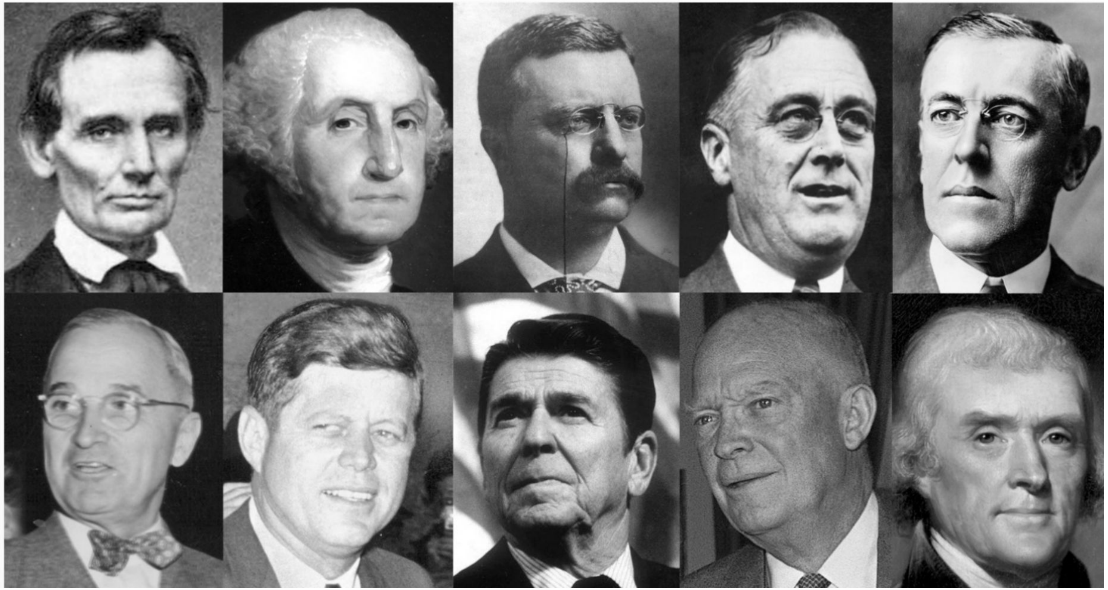


In R: `dbscan::dbscan()`

Source: https://medium.com/@elutins/dbscan-what-is-it-when-to-use-it-how-to-use-it-8bd506293818

Presidential speech dataset

See quick example in `clustering_demo.R` in the week6 folder



DBSCAN

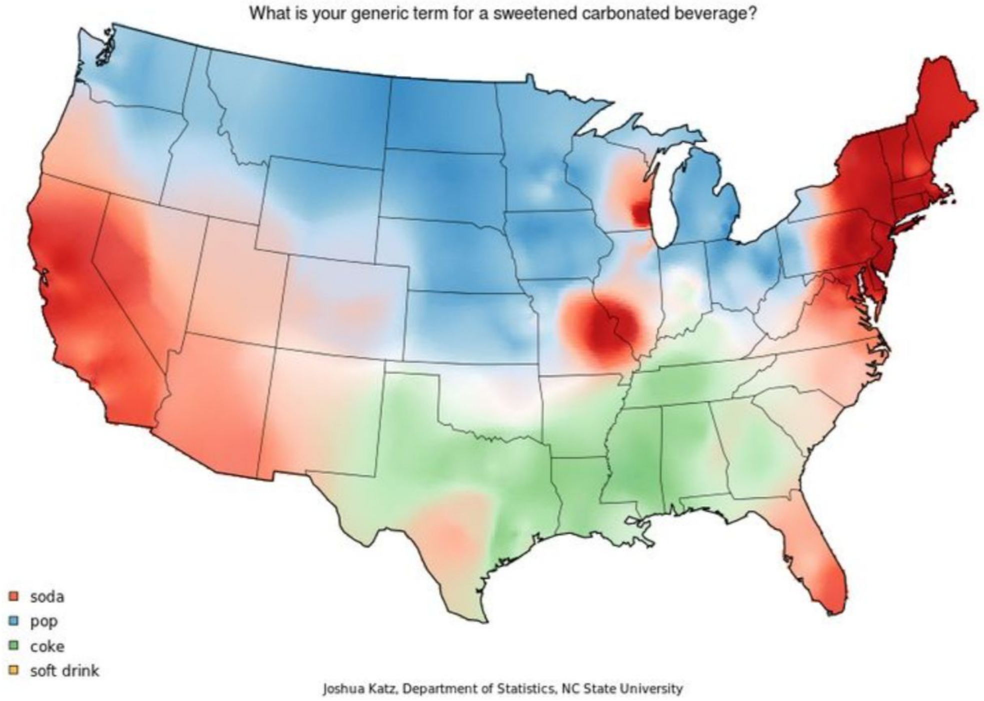
Advantages:

- Don’t have to choose K (but depends on choice of ε and minPts)
- Great for spatial data
- Great at separating clusters of similar densities that are well separated
- Robust to outliers
- Flexible to arbitrarily-shaped clusters

Disadvantages

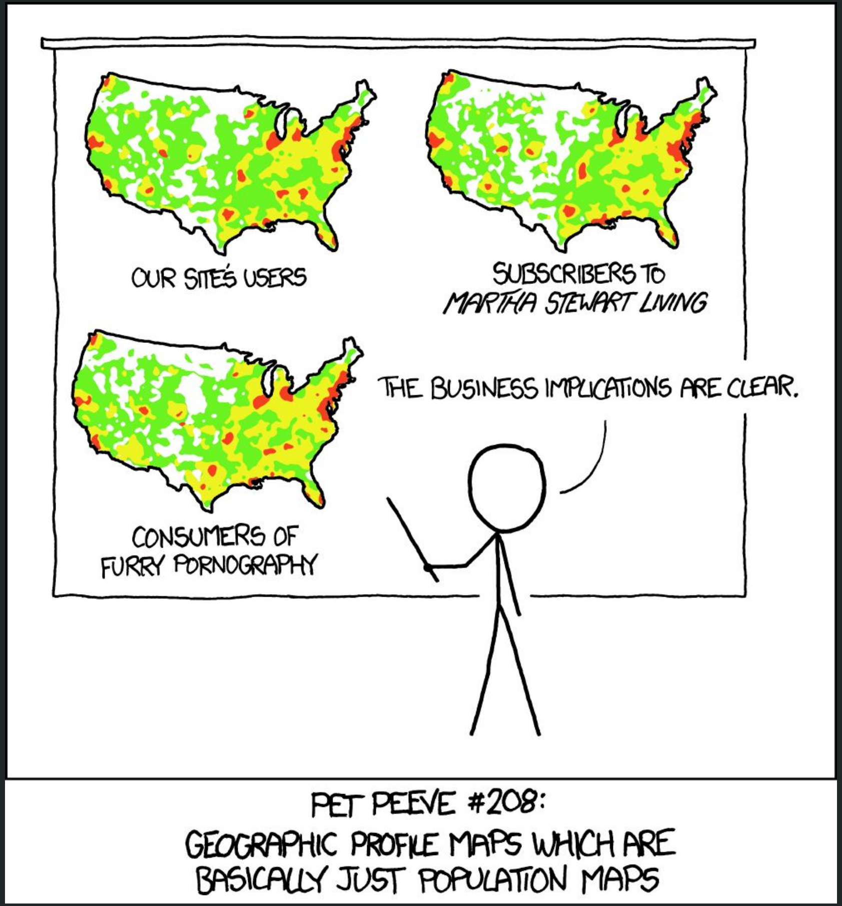
- If the data and scale are not well understood, choosing a meaningful distance threshold ε and minPts can be difficult
- Struggles when clusters are of varying densities since (ε, minPts) cannot be chosen appropriately for all clusters
- Curse of dimensionality when distance metric is Euclidean distance
- Algorithm depends on ordering of points; border points that are reachable from more than one cluster can be part of either cluster, depending on the order the data are processed

Questions on HW2 or Lab 2?



https://www.businessinsider.com/22-maps-that-show-the-deepest-linguistic-conflicts-in-america-2013-6#ok-this-one-is-crazy-everyone-pronounces-pecan-pie-differently-10

Is your map more than a map of population density?



Lab 2 reminders

- Use figure captions for cross-referencing: `fig.cap=”My awesome caption”`
- Use png and adjust DPI, e.g.: `dev=”png”, dpi = 300`
- Folder structure for submission:
  - `stat-215-a/`
    - `lab2/`
      - `lab2.Rmd & .pdf`
      - `lab2_blind.Rmd & .pdf`
      - `R/`
      - `other/`

Optional, for .bib files or other things necessary to reproduce your lab (but don’t over do it!).

- Be careful when using section headers ‘#’

In-class labs

- **Week 1**: tidyverse basics
- **Week 2**: ggplot + Rmd tips and tricks
- **Week 3**: more ggplot + additional plotting tools (pair plots, heatmaps, etc.)
- **Week 4**: PCA
- **Week 5**: K-means, hierarchical clustering NMF