

# CLUSTERING

*Naumaan Nayyar*

---

## CLUSTERING

---

# LEARNING OBJECTIVES

- ▶ Supervised vs unsupervised algorithms
- ▶ Understand and apply k-means clustering
- ▶ Density-based clustering: DBSCAN
- ▶ Silhouette Metric

---

**OPENING**

---

# UNSUPERVISED LEARNING

---

## UNSUPERVISED LEARNING

---

- ▶ So far all the algorithms we have used are *supervised*: each observation (row of data) came with one or more *labels*, either *categorical variables* (classes) or *measurements* (regression)
- ▶ **Unsupervised learning** has a different goal: **feature discovery**
- ▶ **Clustering** is a common and fundamental example of unsupervised learning
- ▶ **Clustering** algorithms try to find meaningful groups within data

---

**CLUSTERING**

---

# CLUSTERING

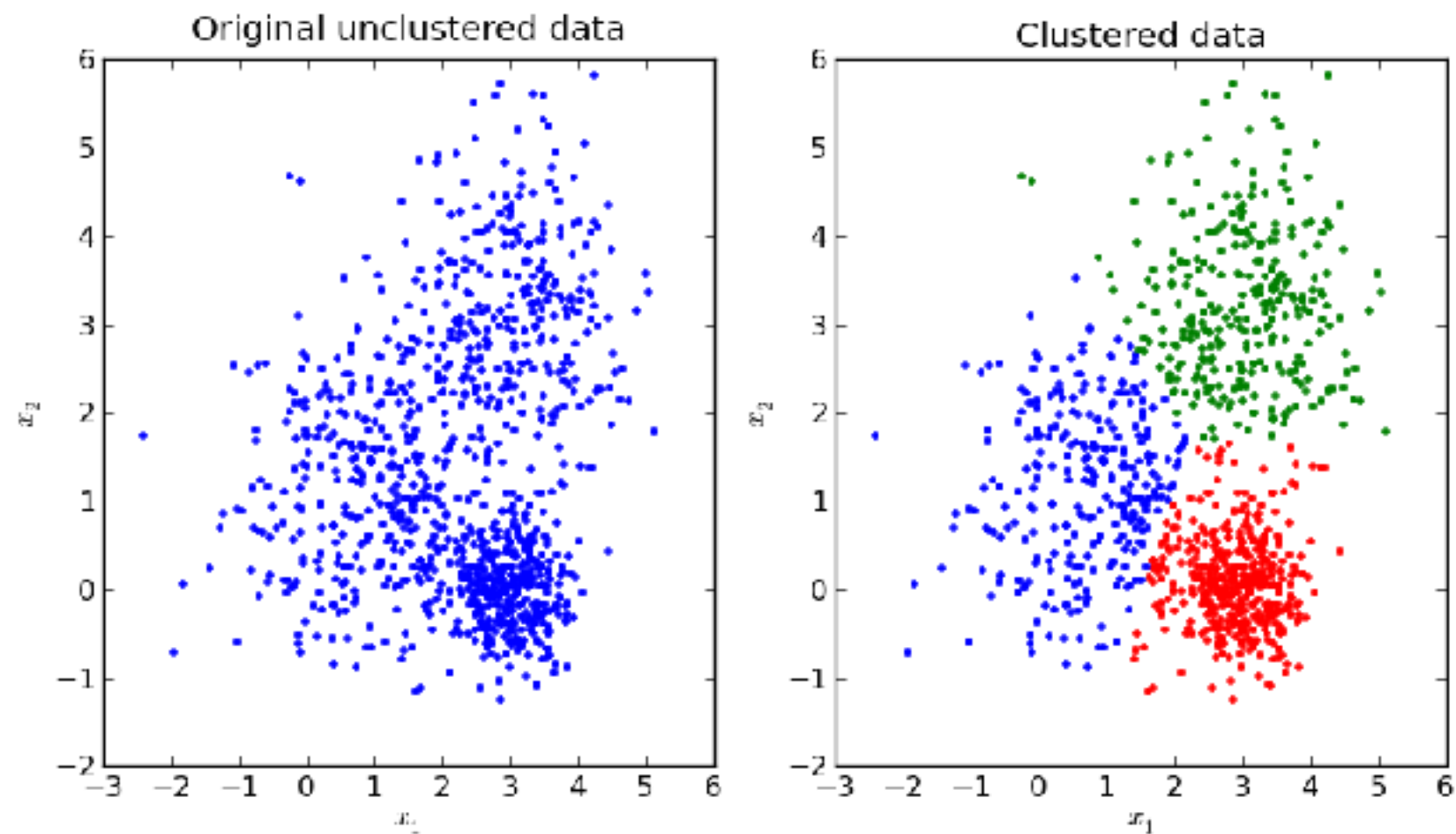
---

## **CLUSTERING: Types of clustering algorithms**

---

- ▶ Clustering algorithms are of several types based on model assumptions:
  - ▶ Centroid-based
  - ▶ Density-based
  - ▶ Hierarchical (connectivity- or linkage-based)
  - ▶ Distribution-based

# CLUSTERING: Centroids



Source: <http://stackoverflow.com/questions/24645068/k-means-clustering-major-understanding-issue>

---

# ACTIVITY: KNOWLEDGE CHECK

---



## EXERCISE

### ANSWER THE FOLLOWING QUESTIONS

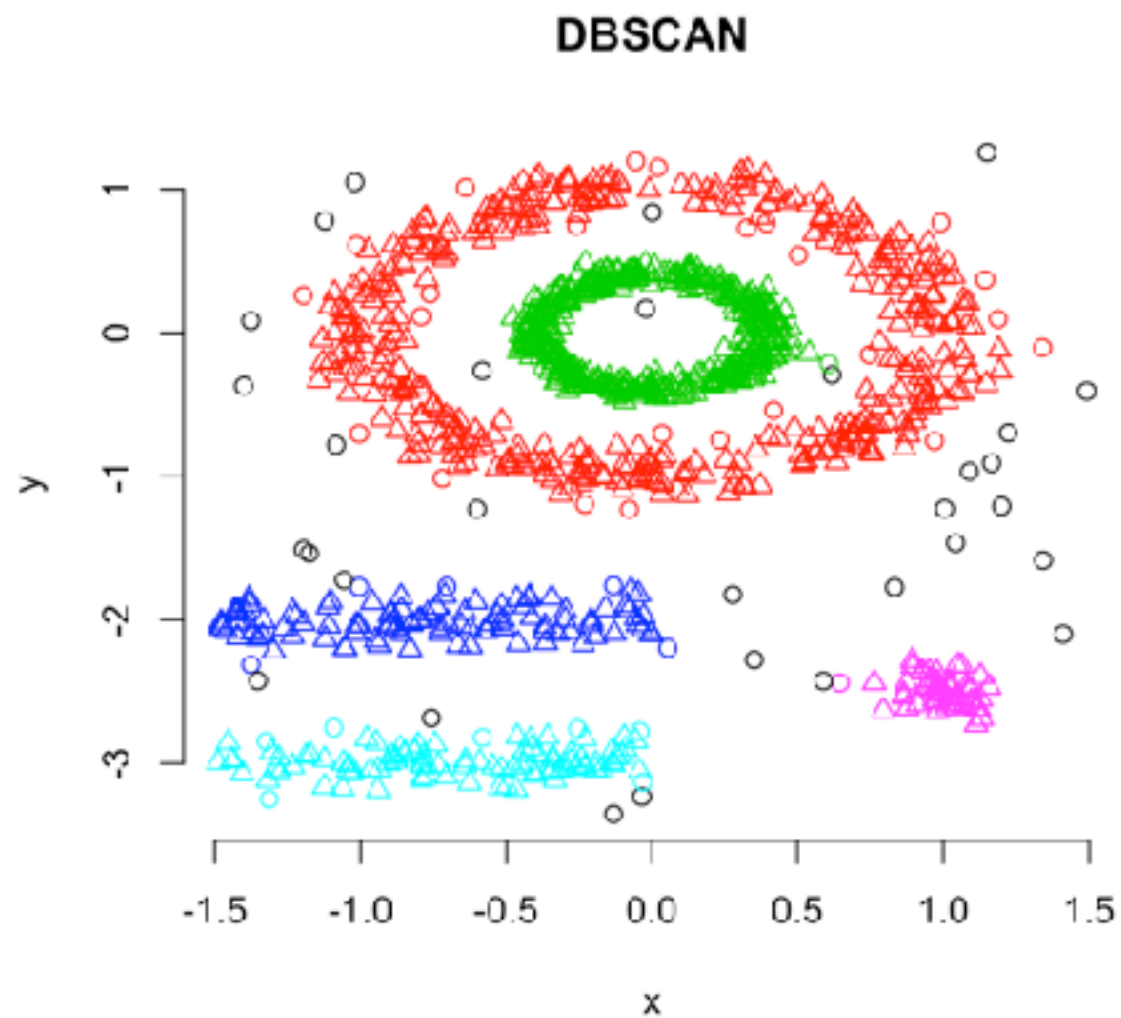
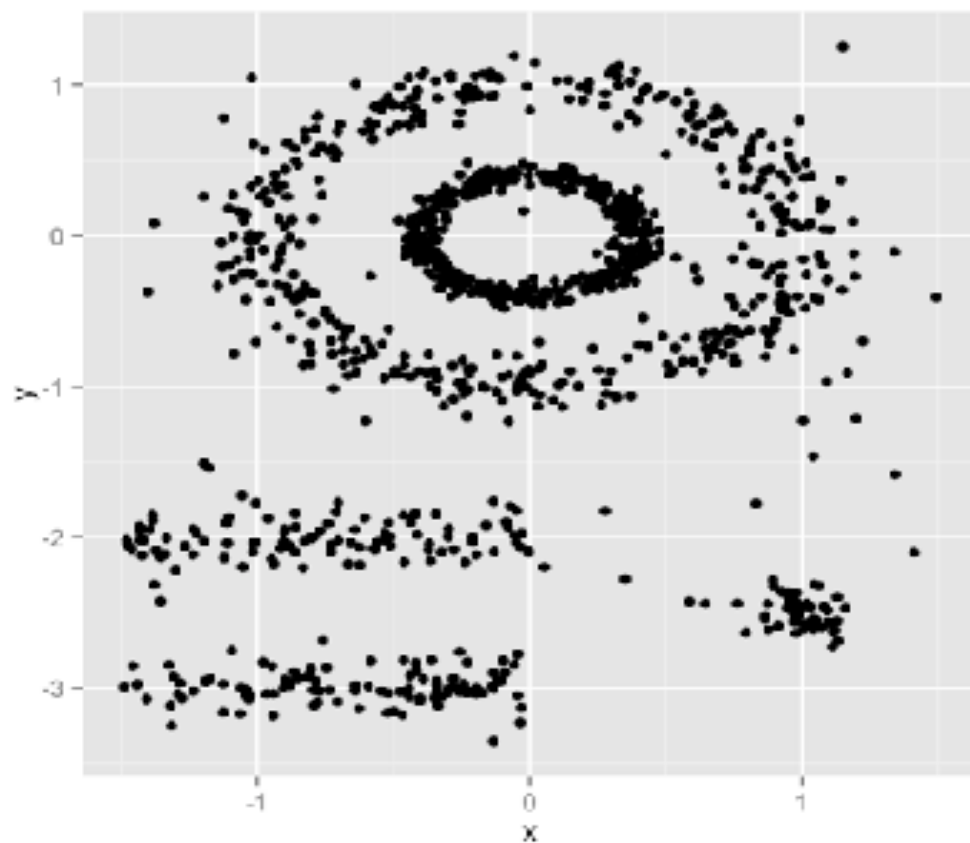
1. Why might data often appear in centered clusters?

### DELIVERABLE

Answers to the above questions



# CLUSTERING: Density-Based



---

# ACTIVITY: KNOWLEDGE CHECK

---



## EXERCISE

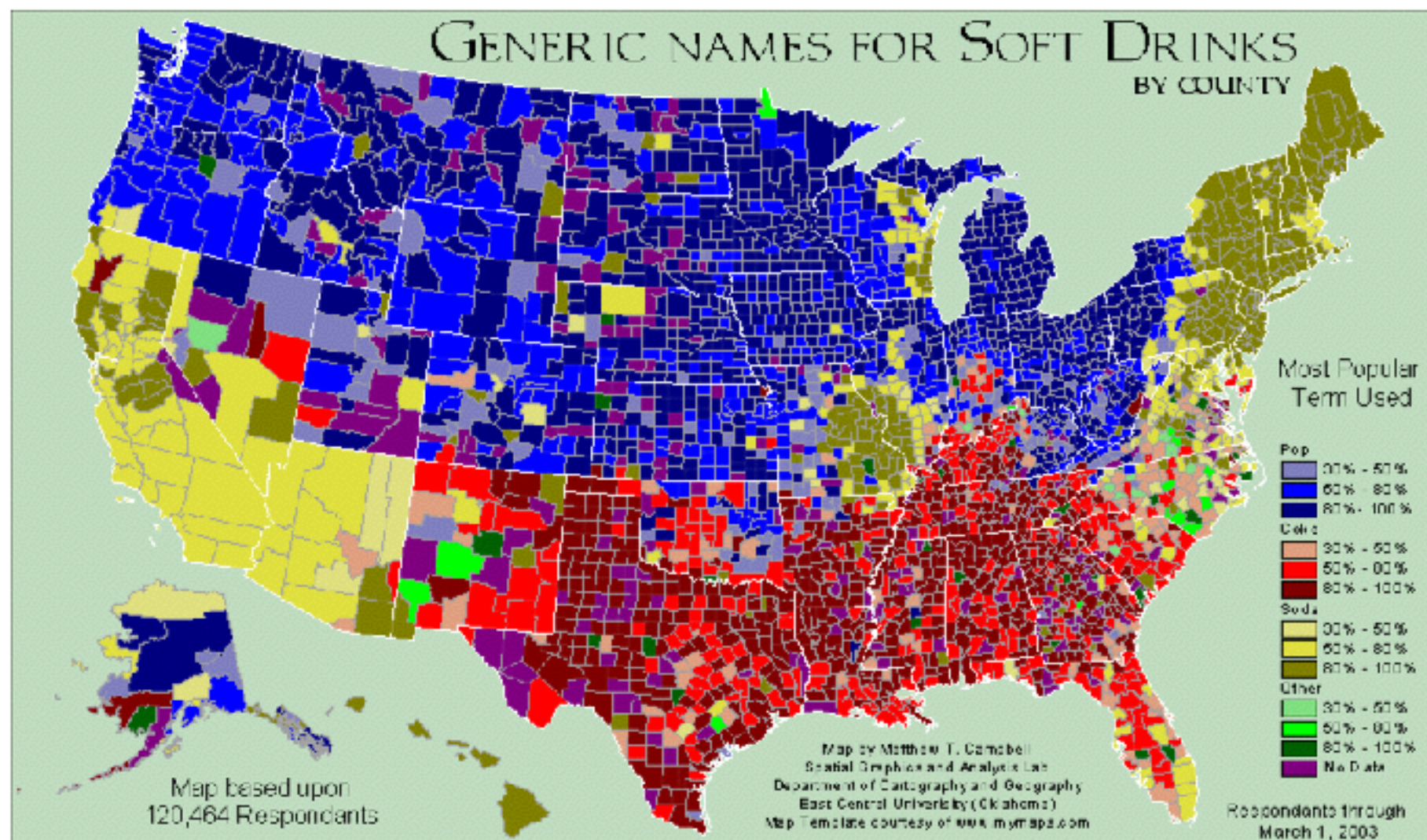
### ANSWER THE FOLLOWING QUESTIONS

1. Why might data often appear in density-based clusters?

### DELIVERABLE

Answers to the above questions

## ACTIVITY: KNOWLEDGE CHECK



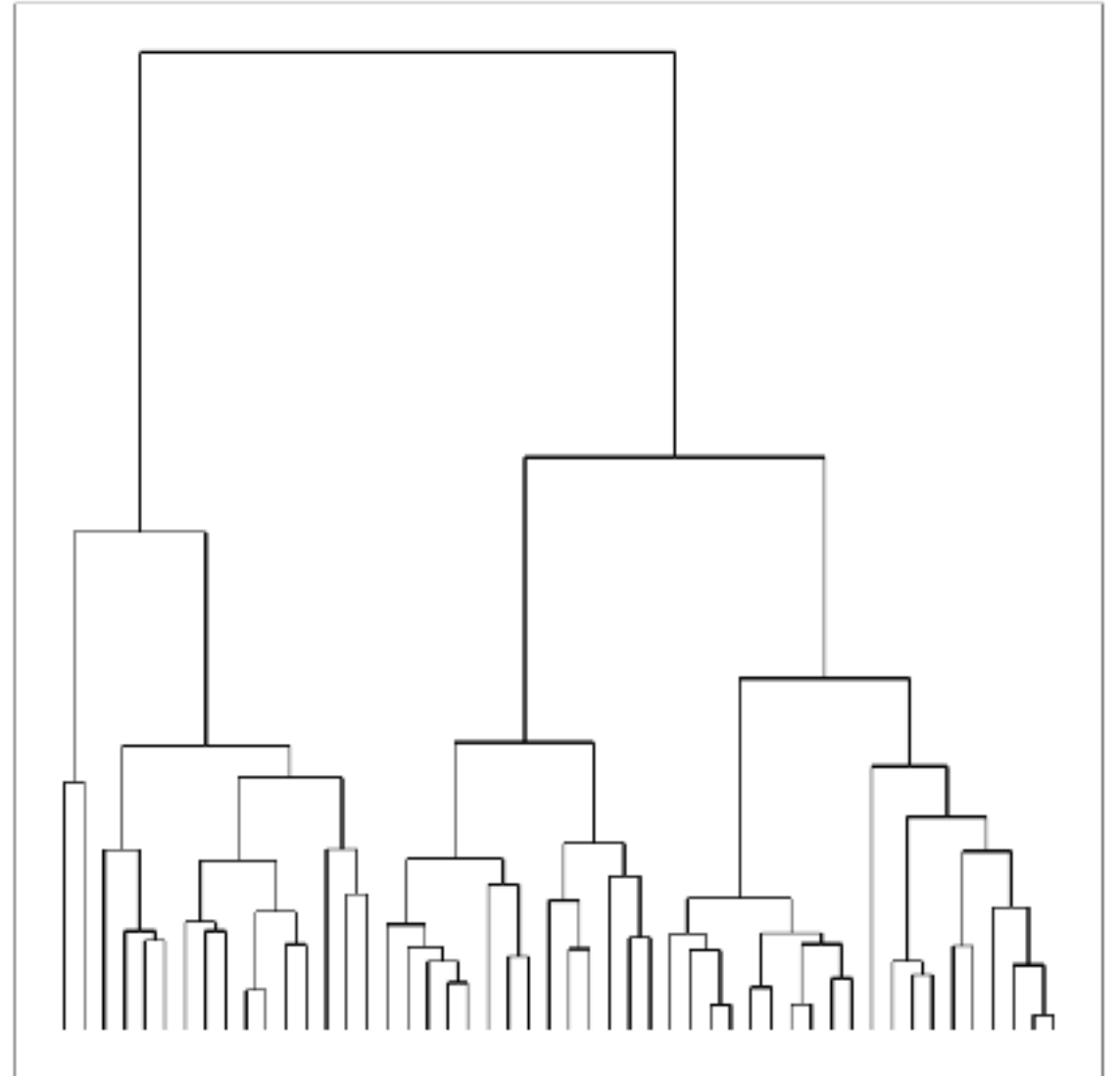
See also: <http://www4.ncsu.edu/~jakatz2/files/dialectposter.png>

---

## CLUSTERING: Hierarchical

---

- ▶ Build hierarchies that form clusters
- ▶ Based on classification trees (next lesson)



---

# ACTIVITY: KNOWLEDGE CHECK

---



## EXERCISE

### ANSWER THE FOLLOWING QUESTIONS

1. How is unsupervised learning different from classification?

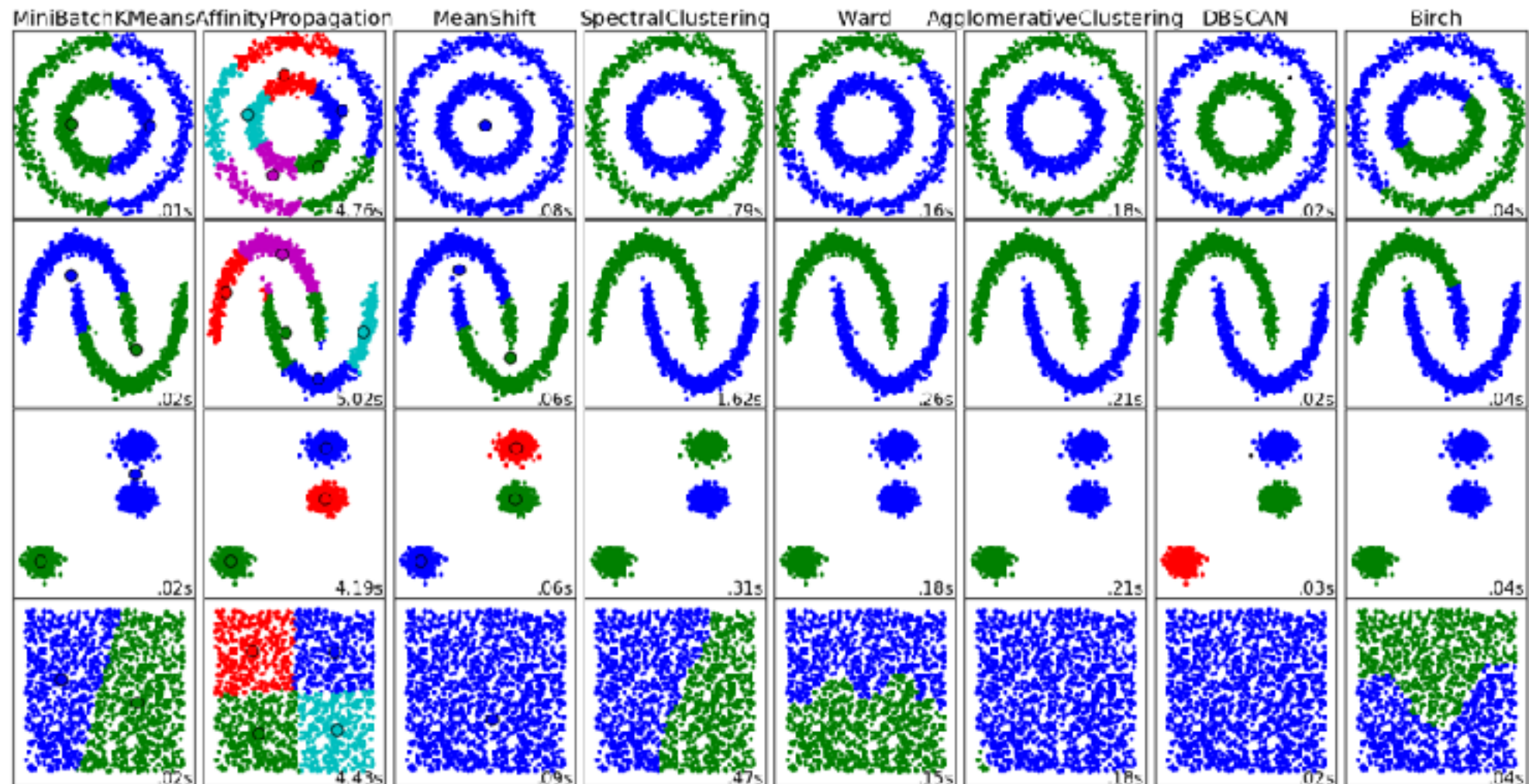
### DELIVERABLE

Answers to the above questions



# CLUSTERING

- There are [many clustering algorithms](#)



---

## ACTIVITY: KNOWLEDGE CHECK

---



### EXERCISE

#### ANSWER THE FOLLOWING QUESTIONS

1. Can you think of a real-world clustering application?

#### DELIVERABLE

Answers to the above questions

---

# ACTIVITY: KNOWLEDGE CHECK

---

## ANSWERS



EXERCISE

1. Recommendation Systems e.g. Netflix genres
2. Medical Imaging: differentiate tissues
3. Identifying market segments
4. Discover communities in social networks
5. Lots of applications for genomic sequences (homologous sequences, genotypes)
6. Earthquake epicenters
7. Fraud detection



## CLUSTERING

---

# K-MEANS: CENTRIOD CLUSTERING

---

## K-MEANS CLUSTERING

---

- ▶ [k-Means](#) clustering is a popular centroid-based clustering algorithm
- ▶ Basic idea: find  $k$  clusters in the data centrally located around various mean points
- ▶ [Awesome Demo](#)

## K-MEANS CLUSTERING

---

- ▶ k-Means seeks to minimize the sum of squares about the means
- ▶ Precisely, find  $k$  subsets  $S_1, \dots, S_k$  of the data with means  $\mu_1, \dots, \mu_k$  that minimizes:

$$\arg \min_{\mathbf{S}} \sum_{i=1}^k \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \boldsymbol{\mu}_i\|^2$$

---

## K-MEANS CLUSTERING

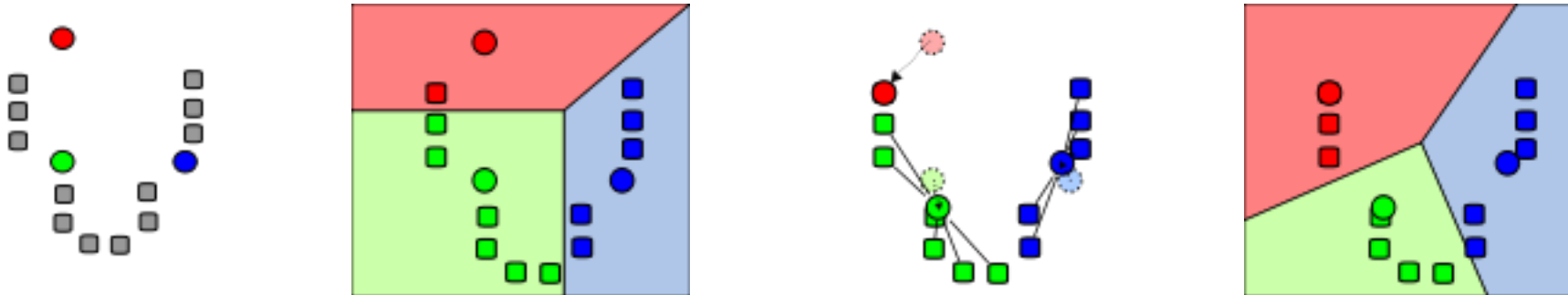
---

- ▶ This is a computationally difficult problem to solve so we rely on heuristics
- ▶ The “standard” heuristic is called “Lloyd’s Algorithm”:
  - ▶ Start with  $k$  initial mean values
  - ▶ Data points are then split up into a [Voronoi diagram](#)
    - ▶ Each point is assigned to the “closest” mean
  - ▶ Calculate new means based on centroids of points in the cluster
  - ▶ Repeat until clusters do not change

# K-MEANS CLUSTERING

---

- ▶ Start with initial k mean values
- ▶ Data points are then split up into a [Voronoi diagram](#)
- ▶ Calculate new means based on centroids



---

# K-MEANS CLUSTERING

---

- ▶ `from sklearn.cluster import KMeans`
- ▶ `est = KMeans(n_clusters=3)`
- ▶ `est.fit(X)`
- ▶ `labels = est.labels_`

Let's try it out!

---

# ACTIVITY: KNOWLEDGE CHECK

---



## EXERCISE

### ANSWER THE FOLLOWING QUESTIONS

1. How do we assign meaning to the clusters we find?
2. Do clusters always have meaning?

### DELIVERABLE

Answers to the above questions

---

# K-MEANS CLUSTERING

---

- ▶ Assumptions are important! k-Means assumes:
  - ▶ k is the correct number of clusters
  - ▶ the data is isotropically distributed (circular/spherical distribution)
  - ▶ the variance is the same for each variable
  - ▶ clusters are roughly the same size

Nice counterexamples / cases where assumptions are not met:

- <http://varianceexplained.org/r/kmeans-free-lunch/>
- [Scikit-Learn Examples](#)



---

# K-MEANS CLUSTERING

---

- ▶ Netflix prize: Predict how users will rate a movie
  - ▶ How might you do this with clustering?
  - ▶ Cluster similar users together and take the average rating for a given movie by users in the cluster (which have rated the movie)
  - ▶ Use the average as the prediction for users that have not yet rated the movie
- ▶ In other words, fit a model to users in a cluster for each cluster and make predictions per cluster
- ▶ [k-Means for the Netflix Prize](#)

## **CLUSTERING**

---

# **DBSCAN: DENSITY BASED CLUSTERING**

---

# DBSCAN CLUSTERING

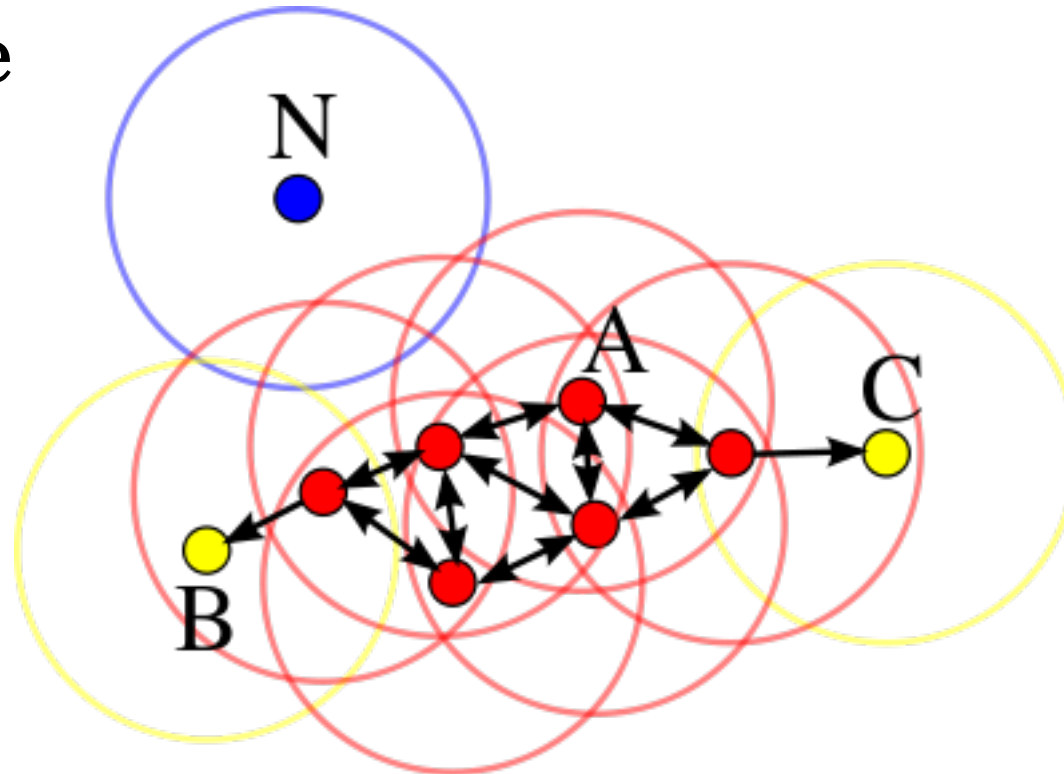
---

- ▶ [DBSCAN](#): Density-based spatial clustering of applications with noise (1996)
- ▶ Main idea: Group together closely-packed points by identifying
  - ▶ Core points
  - ▶ Reachable points
  - ▶ Outliers (not reachable)
- ▶ Two parameters:
  - ▶ min\_samples
  - ▶ eps

# DBSCAN CLUSTERING

---

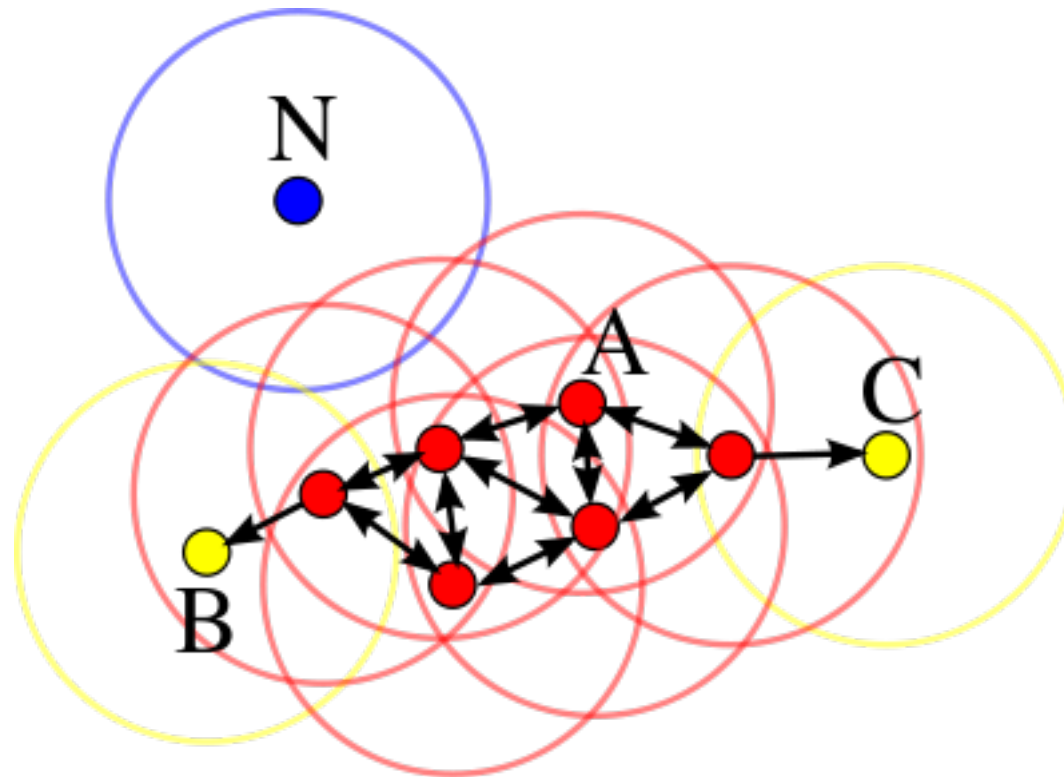
- ▶ Core points: at least **min\_samples** points within **eps** of the core point
  - ▶ Such points are *directly reachable* from the core point
- ▶ Reachable: point  $q$  is reachable from  $p$  if there is a path of core points from  $p$  to  $q$
- ▶ Outlier: not reachable



## DBSCAN CLUSTERING

---

- ▶ A cluster is a collection of connected core and reachable points



---

## CLUSTERING: Density-Based

---

- ▶ Another example: [Page 6](#)
- ▶ [Awesome Demo](#)

---

## DBSCAN CLUSTERING

---

- ▶ `from sklearn.cluster import DBSCAN`
- ▶ `est = DBSCAN(eps=0.5, min_samples=10)`
- ▶ `est.fit(X)`
- ▶ `labels = est.labels_`

Let's try it out!

---

# ACTIVITY: KNOWLEDGE CHECK

---



## EXERCISE

### ANSWER THE FOLLOWING QUESTIONS

1. How does DBSCAN differ from k-means?

### DELIVERABLE

Answers to the above questions



---

# DBSCAN CLUSTERING

---

- ▶ DBSCAN advantages:
  - ▶ Can find arbitrarily-shaped clusters
  - ▶ Don't have to specify number of clusters
  - ▶ Robust to outliers
- ▶ DBSCAN disadvantages:
  - ▶ Doesn't work well when clusters are of varying densities
    - ▶ hard to chose parameters that work for all clusters
  - ▶ Can be hard to chose correct parameters regardless

---

**CLUSTERING**

---

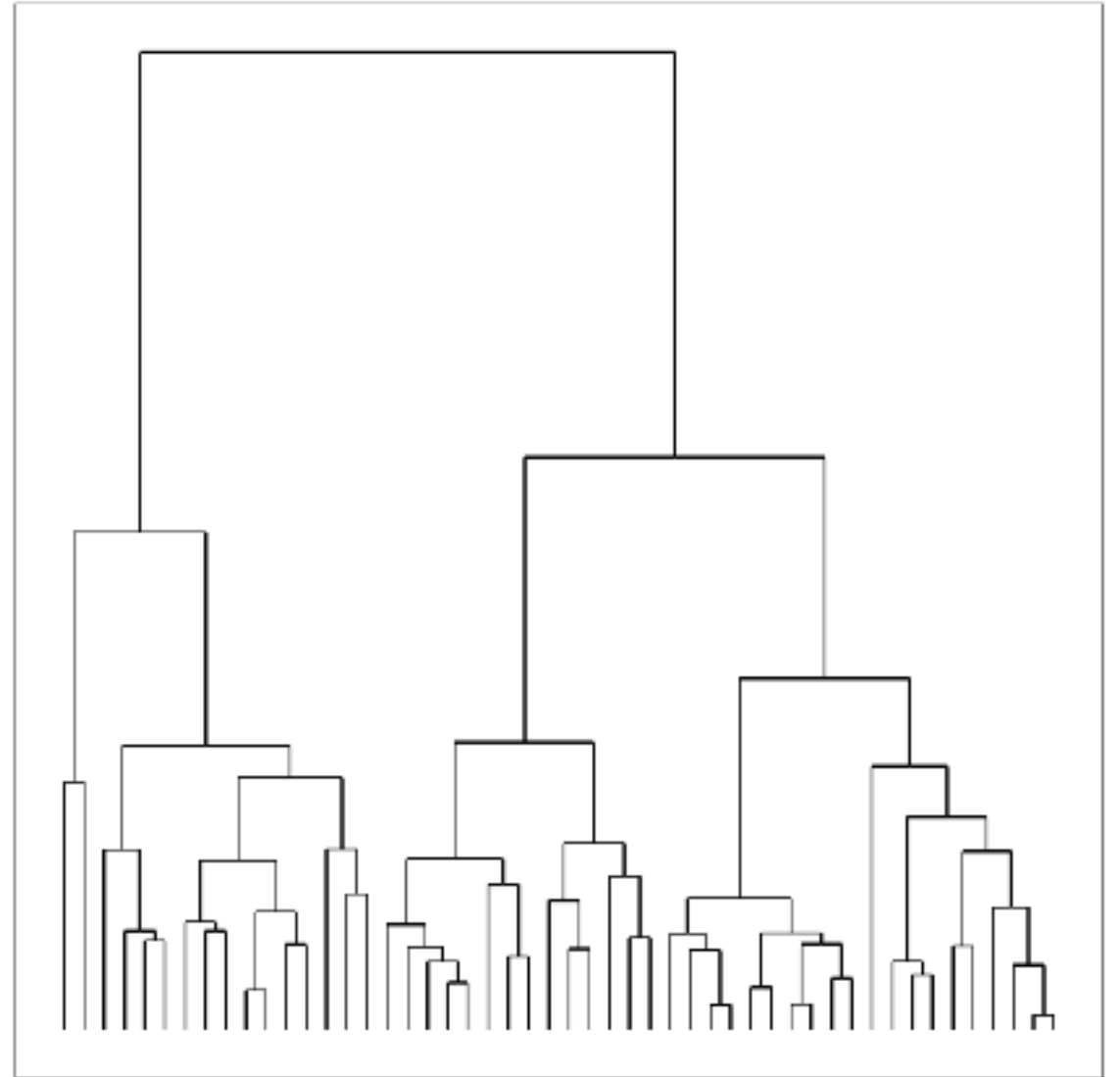
# **HIERARCHICAL CLUSTERING**

---

## CLUSTERING: Hierarchical

---

- ▶ Build hierarchies that form clusters
- ▶ Based on classification trees (next lesson)



---

# HIERARCHICAL CLUSTERING

---

We'll discuss the details once we cover decision trees. For now we can black box the model and fit with sklearn

```
▶ from sklearn.cluster import AgglomerativeClustering
▶ est = AgglomerativeClustering(n_clusters=4)
▶ est.fit(X)
▶ labels = est.labels_
```

Let's try it out!

---

**CLUSTERING**

---

# CLUSTERING METRICS

---

# CLUSTERING METRICS

---

- As usual we need a metric to evaluate model fit
- For clustering we use a metric called the [Silhouette Coefficient](#)
  - **a** is the mean distance between a sample and all other points in the cluster
  - **b** is the mean distance between a sample and all other points in the *nearest* cluster

- The Silhouette Coefficient is:

$$\frac{b - a}{\max(a, b)}$$

- Ranges between 1 and -1
- Average over all points to judge the cluster algorithm

---

## CLUSTERING METRICS

---

- ▶ `from sklearn import metrics`
- ▶ `from sklearn.cluster import KMeans`
- ▶ `kmeans_model = KMeans(n_clusters=3, random_state=1).fit(X)`
- ▶ `labels = kmeans_model.labels_`
- ▶ `metrics.silhouette_score(X, labels, metric='euclidean')`

---

# CLUSTERING METRICS

---

- ▶ There are a number of [other metrics](#) based on:
  - ▶ Mutual Information
  - ▶ Homogeneity
  - ▶ Adjusted Rand Index (when you know the labels on the training data)



---

**PUTTING IT TOGETHER**

---

# **CLUSTERING, CLASSIFICATION, AND REGRESSION**

---

# ACTIVITY: KNOWLEDGE CHECK

---



## EXERCISE

### ANSWER THE FOLLOWING QUESTIONS

1. How might we combine clustering and regression or classification?

### DELIVERABLE

Answers to the above questions

---

## CLUSTERING, CLASSIFICATION, AND REGRESSION

---

- ▶ We can use clustering to discover new features and then use those features for either classification or regression
- ▶ For classification, we could use e.g. k-NN to classify new points into the discovered clusters
- ▶ For regression, we could use a dummy variable for the clusters as a variable in our regression

---

# ACTIVITY: CLUSTERING + REGRESSION

---

## EXERCISE



EXERCISE

1. Follow along as we do an example of clustering with regression.

## DELIVERABLE

A completed notebook

---

**CONCLUSION**

---

# TOPIC REVIEW

---

## REVIEW AND NEXT STEPS

---

- ▶ Clustering is used to discover features, e.g. segment users or assign labels (such as species)
- ▶ Clustering may be the goal (user marketing) or a step in a data science pipeline

**COURSE**

---

**BEFORE NEXT CLASS**

---

## **BEFORE NEXT CLASS**

---

# **UPCOMING**

- ▶ Final Project part 2
  - ▶ Due Date: June 15th



---

## LESSON

---

Q & A

## LESSON

---

# EXIT TICKET

**DON'T FORGET TO FILL OUT YOUR EXIT TICKET**

<https://goo.gl/forms/ZF7B45rp0zkchaRJ2>