Unsupervised Learning: Clustering and Dimensionality Reduction

DUKE
FUQUA
SCHOOL OF BUSINESS

Quick point on module 4/homework 4

- Why did we use "random states" in much of our code?

Examples from module 4:

Let's just fit an unconstrained tree to derive a classification model.

```
tree = DecisionTreeClassifier(random_state=42).fit(X_train, Y_train)
```

```
random_forest = RandomForestClassifier(random_state=42, max_depth=5, n_estimators=5).fit(X_train, Y_train)
```

Ensemble models and complexity: a paradox?

- *Ensembles appear to increase complexity … so, their ability to generalize better seems to violate the preference for simplicity summarized by Occam's Razor.*

  - John Elder, "The Generalization Paradox of Ensembles"

- How do we explain this?

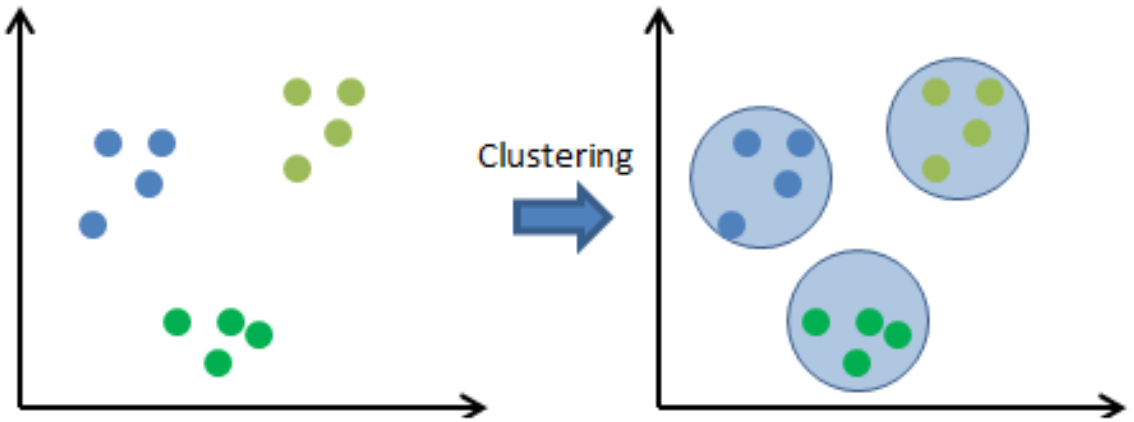
(Class 1) "Supervised" vs. "Unsupervised" problems

- Supervised problems involve a specific *target* (or response) variable, and the goal is to understand the relationship of the target to other feature variables
  - Example: "Can we find groups of bank customers that have a higher probability of subscribing to a term deposit?"
  - Most supervised learning problems involve *predictive modeling* techniques, such as classification and regression.

Today!

- Unsupervised problems do not have a target variable, and the goal is to understand the structure of the data.
  - Example: "Do there exist natural groups of bank customers? If so, what features distinguish these groups?"
  - Clustering is a canonical unsupervised learning problem.

In cluster analysis, we are trying to discover if our data falls into natural groups that are "similar"



- We will study three widely used forms of cluster analysis:
  - K-means
  - Hierarchical
  - Density-based
- Issues:
  - How many clusters?
  - How do we measure similarity?

Example: clustering whiskeys

	name	category	rating	alcohol	age	price	description
0	Johnnie Walker Blue Label, 40%	Blended Scotch Whisky	97.0	40.0	NaN	225.0	Magnificently powerful and intense. Caramels, ...
1	Black Bowmore, 1964 vintage, 42 year old, 40.5%	Single Malt Scotch	97.0	40.5	42.0	4500.0	What impresses me most is how this whisky evol...
2	Bowmore 46 year old (distilled 1964), 42.9%	Single Malt Scotch	97.0	42.9	46.0	13500.0	There have been some legendary Bowmores from t...
3	Compass Box The General, 53.4%	Blended Malt Scotch Whisky	96.0	53.4	NaN	325.0	With a name inspired by a 1926 Buster Keaton m...
4	Chivas Regal Ultis, 40%	Blended Malt Scotch Whisky	96.0	40.0	NaN	160.0	Captivating, enticing, and wonderfully charmin...
...	...	...	...	...	...	...	...
2242	Duncan Taylor (distilled at Cameronbridge), Ca...	Grain Scotch Whisky	72.0	54.4	28.0	125.0	Its best attributes are vanilla, toasted cocon...
2243	Distillery Select 'Craiglodge' (distilled at L...	Single Malt Scotch	71.0	45.0	8.0	60.0	Aged in a sherry cask, which adds sweet notes ...
2244	Edradour Barolo Finish, 11 year old, 57.1%	Single Malt Scotch	70.0	57.1	11.0	80.0	Earthy, fleshy notes with brooding grape notes...
2245	Highland Park, Cask #7380, 1981 vintage, 25 ye...	Single Malt Scotch	70.0	55.0	25.0	225.0	The sherry is very dominant and cloying, which...
2246	Distillery Select 'Inchmoan' (distilled at Loc...	Single Malt Scotch	63.0	45.0	13.0	60.0	Fiery peat kiln smoke, tar, and ripe barley on...

- Data set of 2,247 whiskeys, scraped from whiskeyadvocate.com and available on kaggle
- 7 features per whiskey
  - 2 text features (name and description)
  - 1 categorical (type of whiskey, e.g., Single Malt Scotch, etc.)
  - 4 numerical features: rating, alcohol %, age in years, price in $

- Is there a natural way to group these whiskeys?

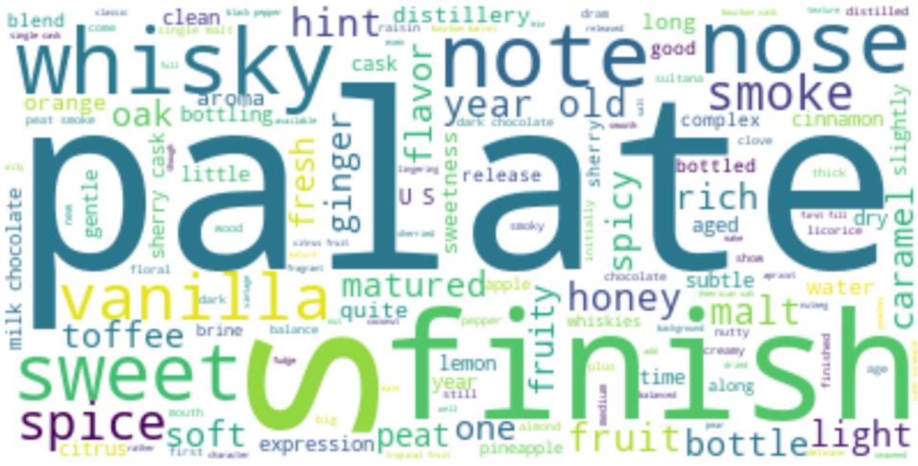
Summary information about whiskeys

- Statistics for numerical features:

```
df[['rating', 'alcohol', 'age', 'price']].describe()
```

	rating	alcohol	age	price
count	2223.000000	2206.000000	1197.000000	2223.000000
mean	86.696356	47.925335	21.004177	655.586895
std	4.049291	5.876252	10.067456	4737.398537
min	63.000000	40.000000	3.000000	12.000000
25%	84.000000	43.000000	13.000000	70.000000
50%	87.000000	46.000000	18.000000	110.000000
75%	90.000000	52.200000	26.000000	200.000000
max	97.000000	67.400000	70.000000	157000.000000

- Word cloud:



- What does it mean for two whiskeys to be similar or “close?”

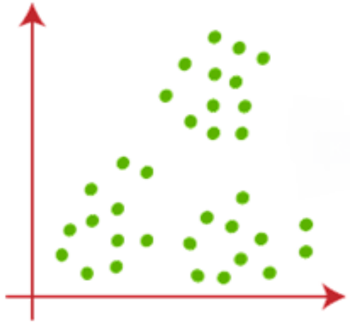
What does it mean for two data points to be "close?"

- In clustering, we view the feature values as coordinates describing a "location" of a data point
- Typical distance measures:
  - Euclidean: "as the crow flies"
  - Manhattan: distance on a grid
- Issues?

- Many others:
  - Cosine: how "aligned" two points are
  - Jaccard: how much "overlap" there is
  - …

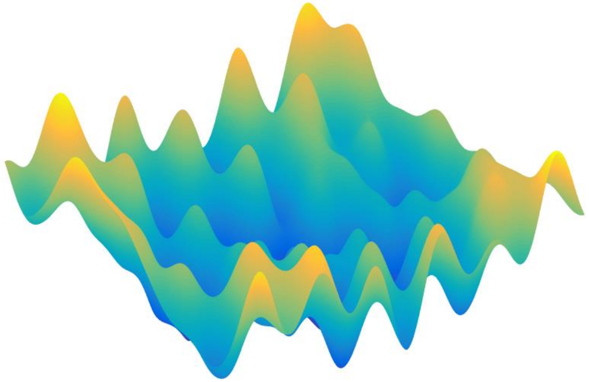
Clustering is an optimization problem

- Fix a k and a distance measure



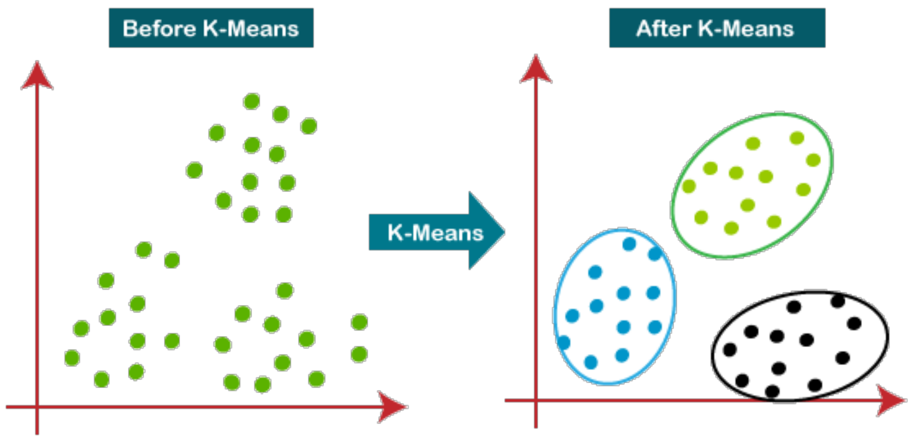
- How can we formulate clustering as an optimization problem?
  - Decision variables? • locations of k centers
  - Objectives? • min. sum of total assigned distances
  - Constraints?

- Clustering is an example of a "nonconvex" optimization problem that is difficult to solve exactly:



How k-means clustering works

- Input: k (= number of clusters)
- Output: k “centroids” which each have dimension = number of features.
- These centroids define the clusters: each data point “belongs” to cluster associated with closest centroid.



- k-means is an iterative algorithm. Starting with initial guess for all centroids:
  1. Assign each data point to its closest current centroid for form k clusters.
  2. Update centroids by taking the mean of each cluster.
  3. Repeat.

Measuring goodness of k-means clustering

- Silhouette coefficient: for every data point, calculate and then average:

higher better

$$\frac{\text{Avg Distance to Those in Closest Cluster} - \text{Avg Distance to Those in My Cluster}}{\text{Maximum of Two Distances in Numerator}}$$

$\in [-1, +1]$

- Calinski-Harabasz Index (or “Variance Ratio”):

higher better

$$\frac{\text{Sum of Variances Between Clusters}}{\text{Sum of Variances Within Clusters}} \times \text{Factor that depends on k}$$

$\frac{n-k}{n-1}$

- Within-Cluster Sum of Squares (WCSS or “inertia”): the sum of all (squared) distances of points to their closest cluster centroids

lower better

D-B → lower better

Back to our whiskey data set

- Data (with one-hot encoding and a bit of cleaning):

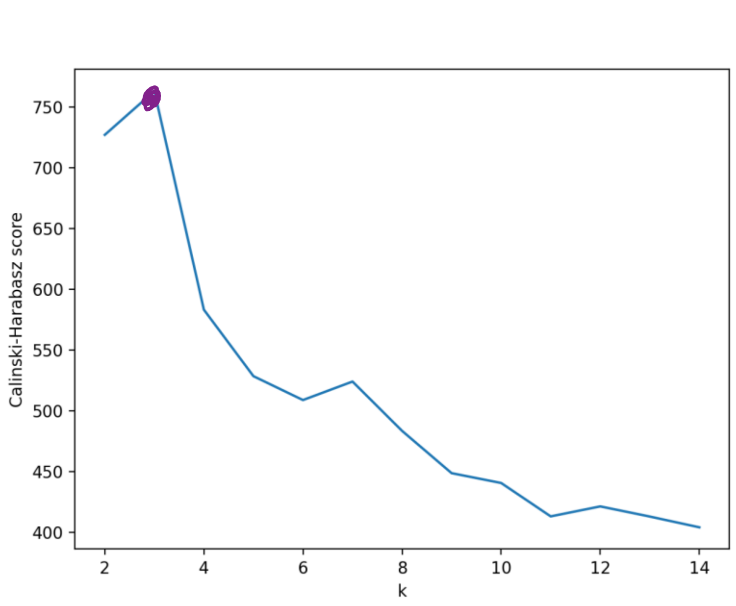
	name	rating	alcohol	age price	category_Blended Malt Scotch Whisky	category_Blended Scotch Whisky	category_Grain Scotch Whisky	category_Single Grain Whisky	category_Single Malt Scotch
0	Johnnie Walker Blue Label, 40%	97.0	40.0	225.0	0	1	0	0	0
1	Black Bowmore, 1964 vintage, 42 year old, 40.5%	97.0	40.5	4500.0	0	0	0	0	1
2	Bowmore 46 year old (distilled 1964), 42.9%	97.0	42.9	13500.0	0	0	0	0	1
3	Compass Box The General, 53.4%	96.0	53.4	325.0	1	0	0	0	0
4	Chivas Regal Ultis, 40%	96.0	40.0	160.0	1	0	0	0	0
...	...	...	...	...	...	...	...	...	...
2242	Duncan Taylor (distilled at Cameronbridge), Ca...	72.0	54.4	125.0	0	0	1	0	0
2243	Distillery Select 'Craiglodge' (distilled at L...	71.0	45.0	60.0	0	0	0	0	1
2244	Edradour Barolo Finish, 11 year old, 57.1%	70.0	57.1	80.0	0	0	0	0	1
2245	Highland Park, Cask #7380, 1981 vintage, 25 ye...	70.0	55.0	225.0	0	0	0	0	1
2246	Distillery Select 'Inchmoan' (distilled at Loc...	63.0	45.0	60.0	0	0	0	0	1

2223 rows × 9 columns

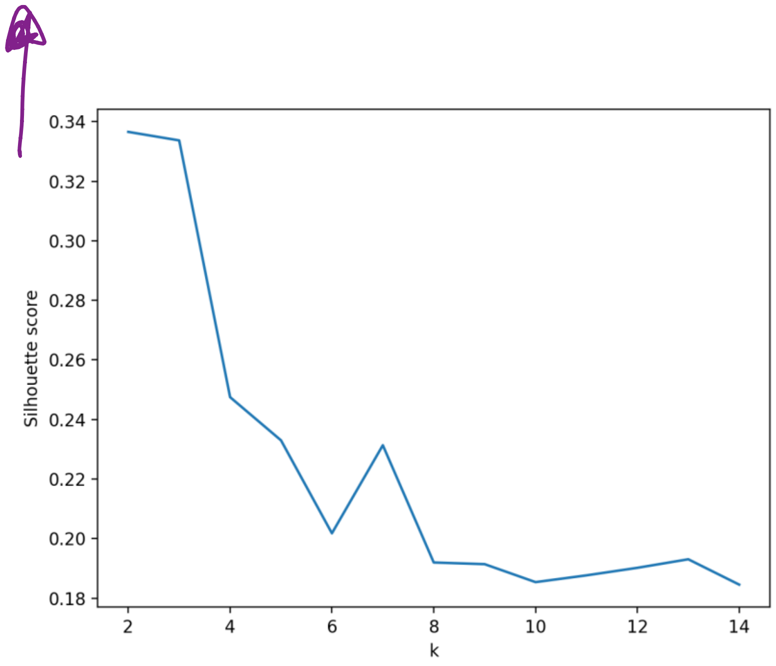
We will cluster along these numerical features.

Optimal number of clusters under various metrics

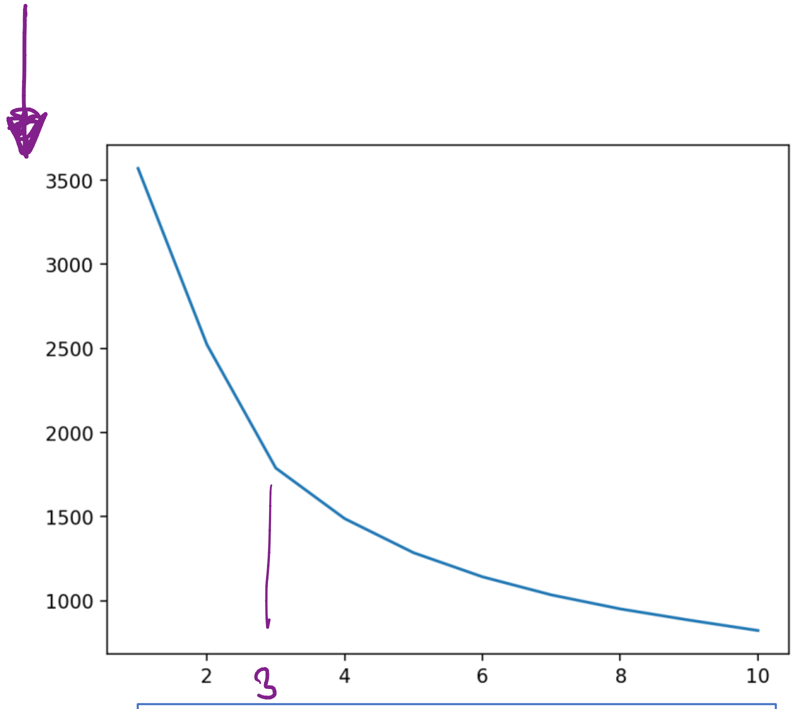
- The "optimal" number depends on how we measure goodness of clustering



Calinkski-Harabasz
Optimal = 3



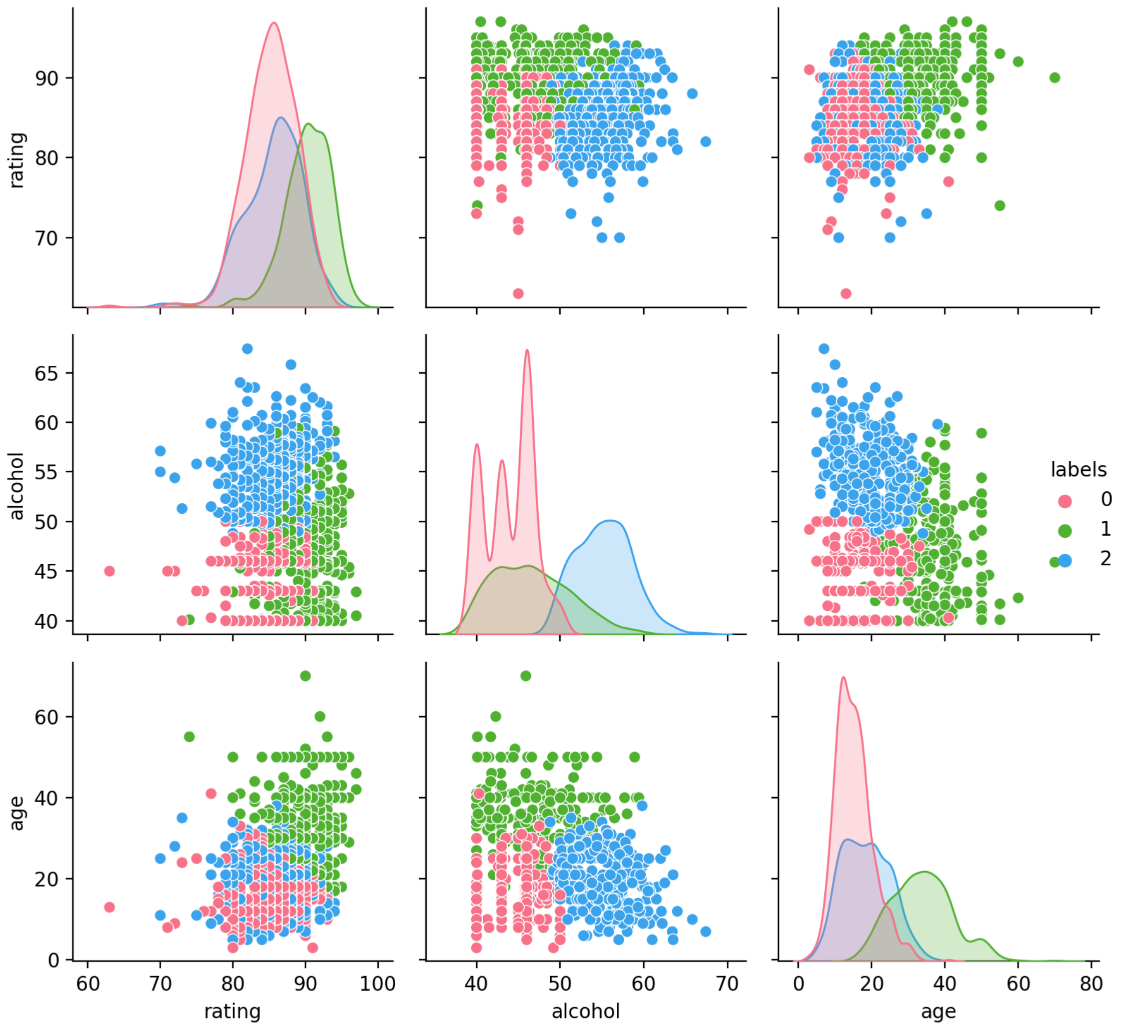
Silhouette
Optimal = 2



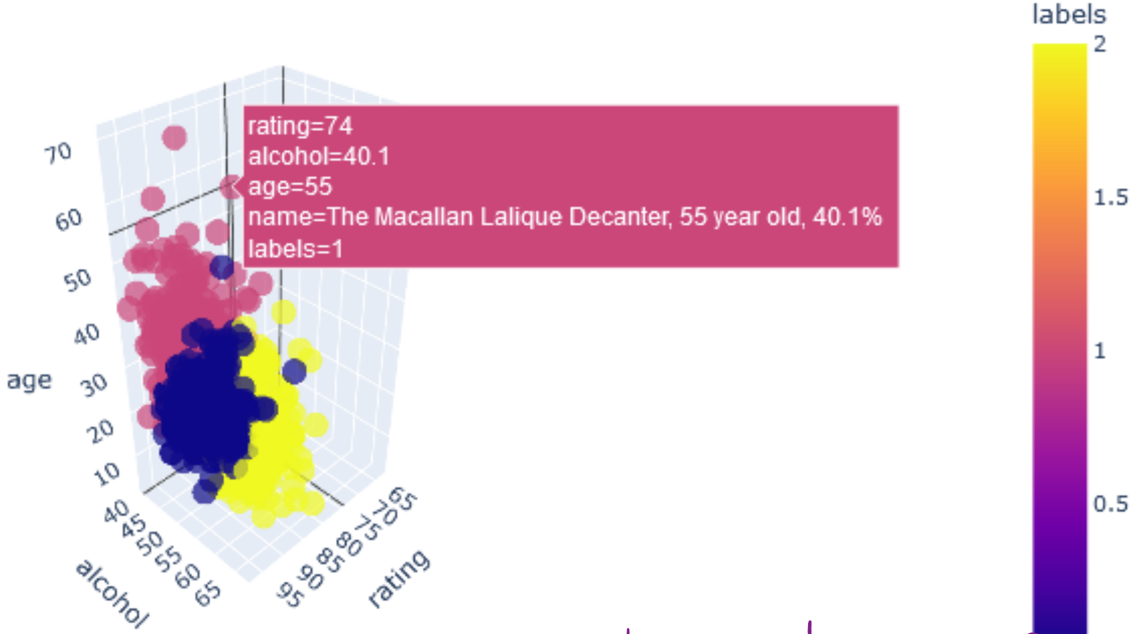
Within cluster sum of squares
Optimal = ? 3

“Optimal” whiskey clusters with k=3

Using Seaborn’s pairplot:



Using plotly.express.scatter_3d



	Cluster	rating	alc.	age
(red)	0	L	L	L
(green)	1	H	L	H
(blue)	2	L	H	L

Further interpreting the whiskey results

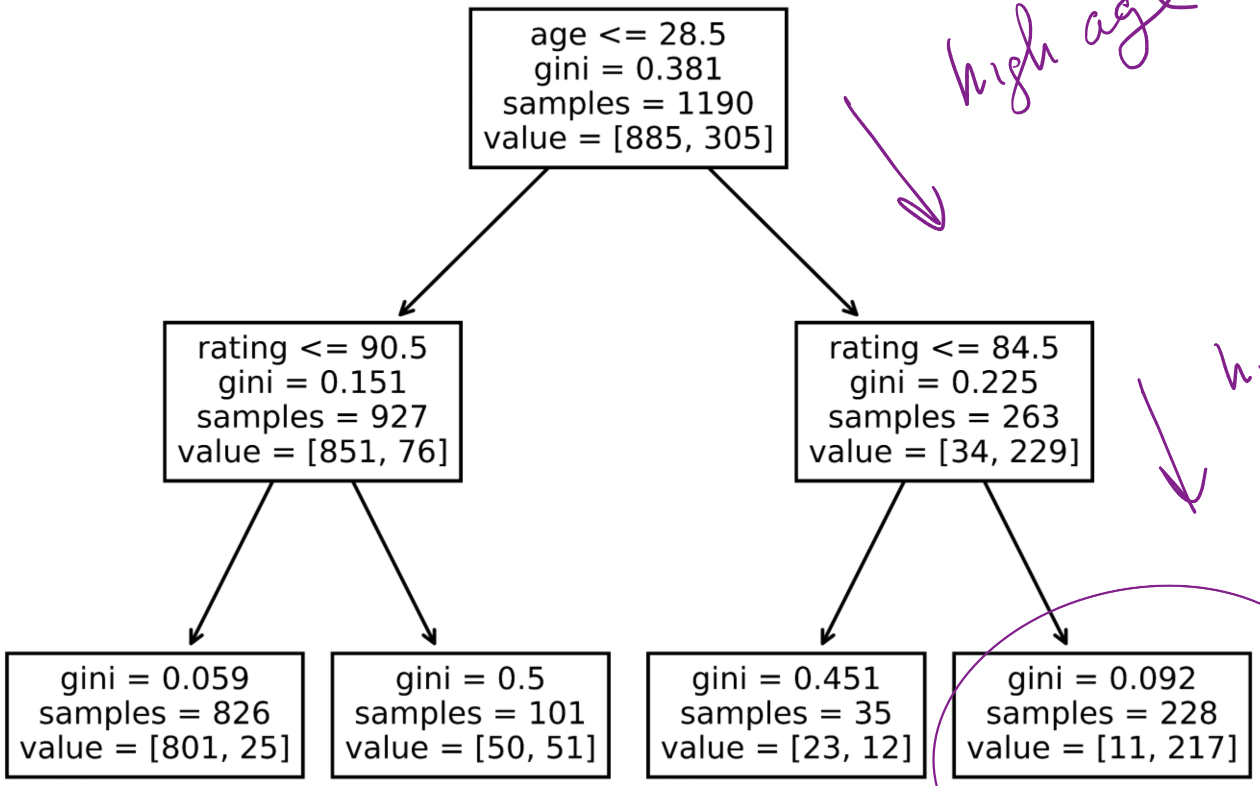
- Clustering results with k=3 and one-hot encoding:

X Y

	rating	alcohol	age	labels_0	labels_1	labels_2
1	97.0	40.5	42.0	0	1	0
2	97.0	42.9	46.0	0	1	0
7	96.0	44.8	40.0	0	1	0
8	96.0	52.8	50.0	0	1	0
11	96.0	45.4	29.0	0	1	0
...	...	...	...	...	...	...
2242	72.0	54.4	28.0	0	0	1
2243	71.0	45.0	8.0	1	0	0
2244	70.0	57.1	11.0	0	0	1
2245	70.0	55.0	25.0	0	0	1
2246	63.0	45.0	13.0	1	0	0

- Say we want to understand what traits define one of the clusters. How could we use *supervised learning* techniques to do this?

Whiskey cluster 1: classification tree of depth 2

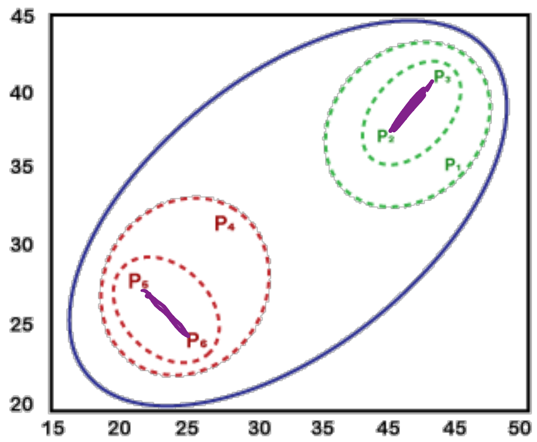


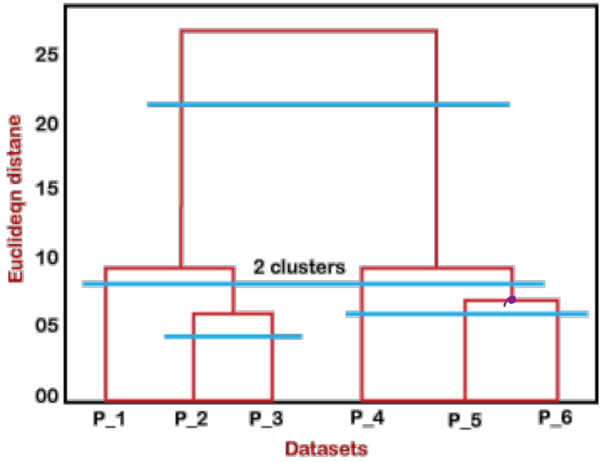
high age

high rating

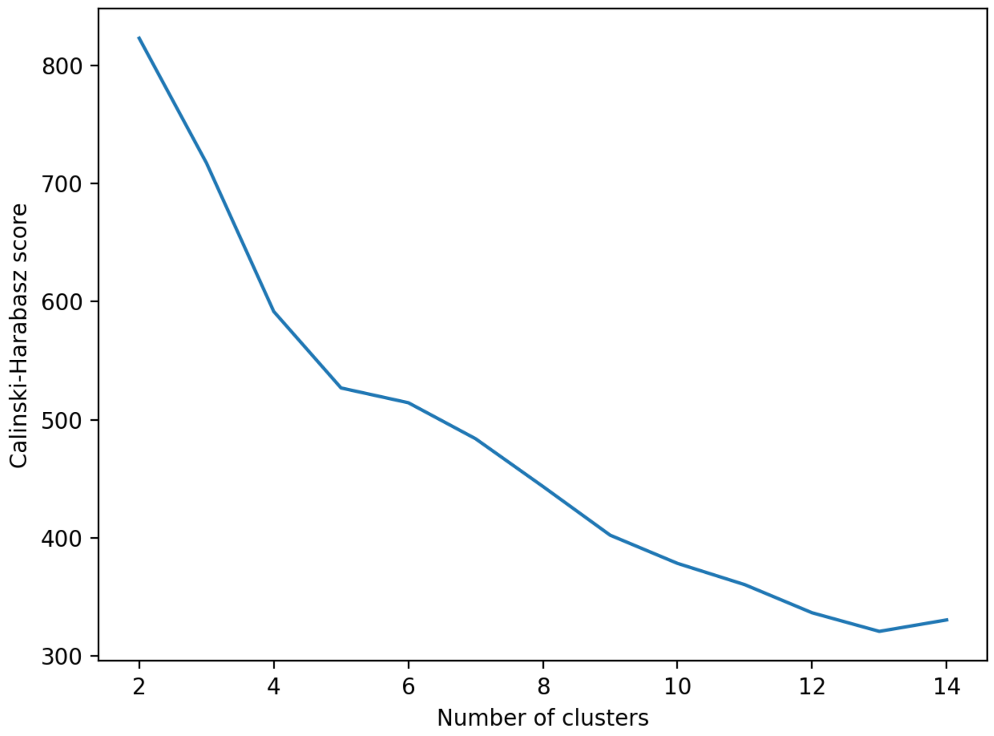
Hierarchical clustering is a form of “agglomerative” clustering

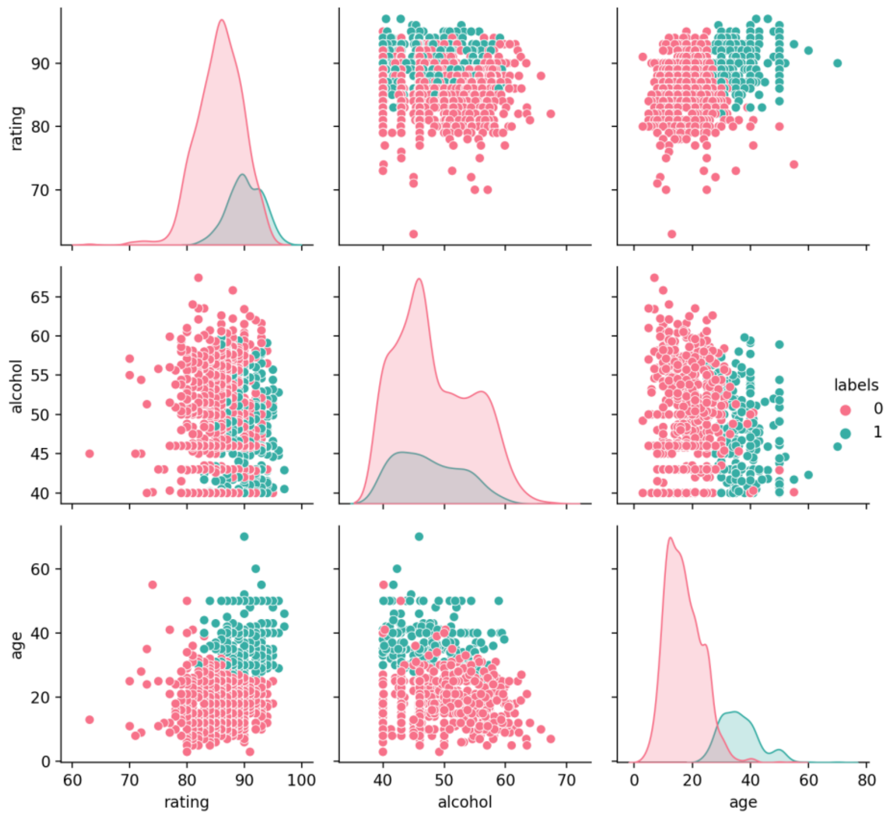
- We progressively group data points together; key ingredients:
  - Measure of distance between data points (e.g., Euclidean)
  - Measure of distance between clusters (“linkage function”):
    - Shortest distance between points within each cluster
    - Farthest distance between points within each cluster
    - Average distance between points within each cluster
    - Etc…



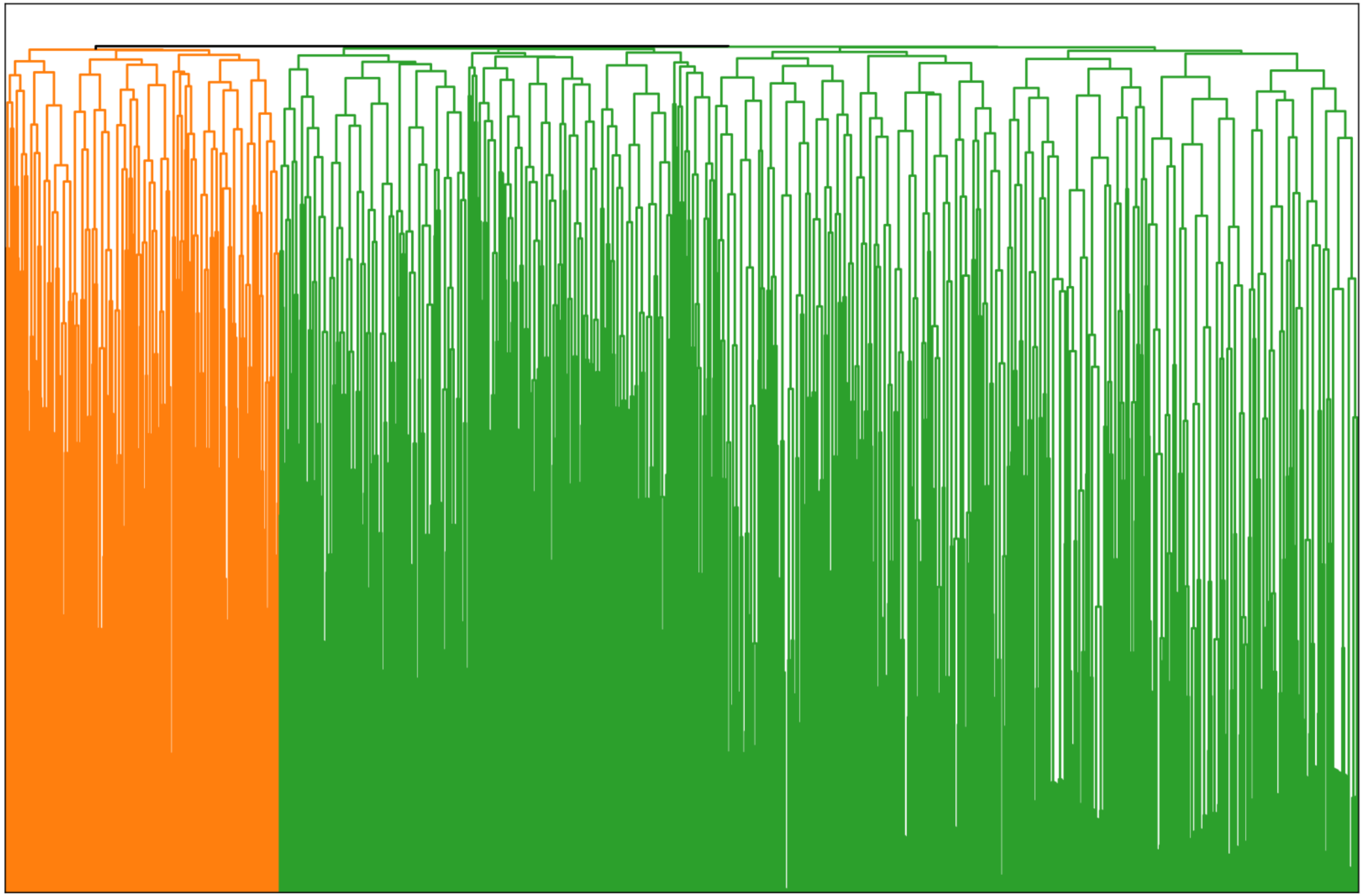


Hierarchical clustering on whiskey dataset

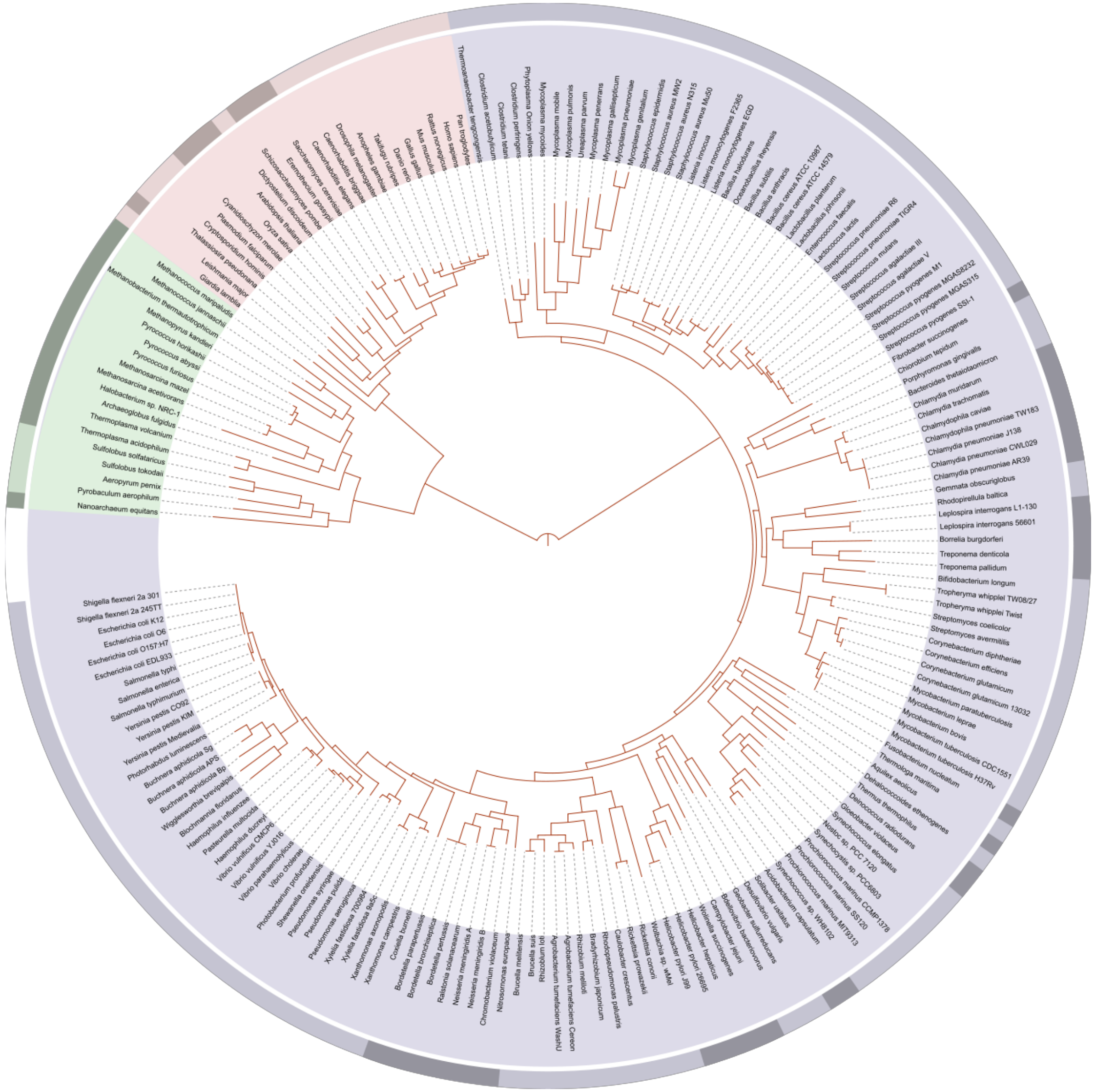




The whiskey dendogram



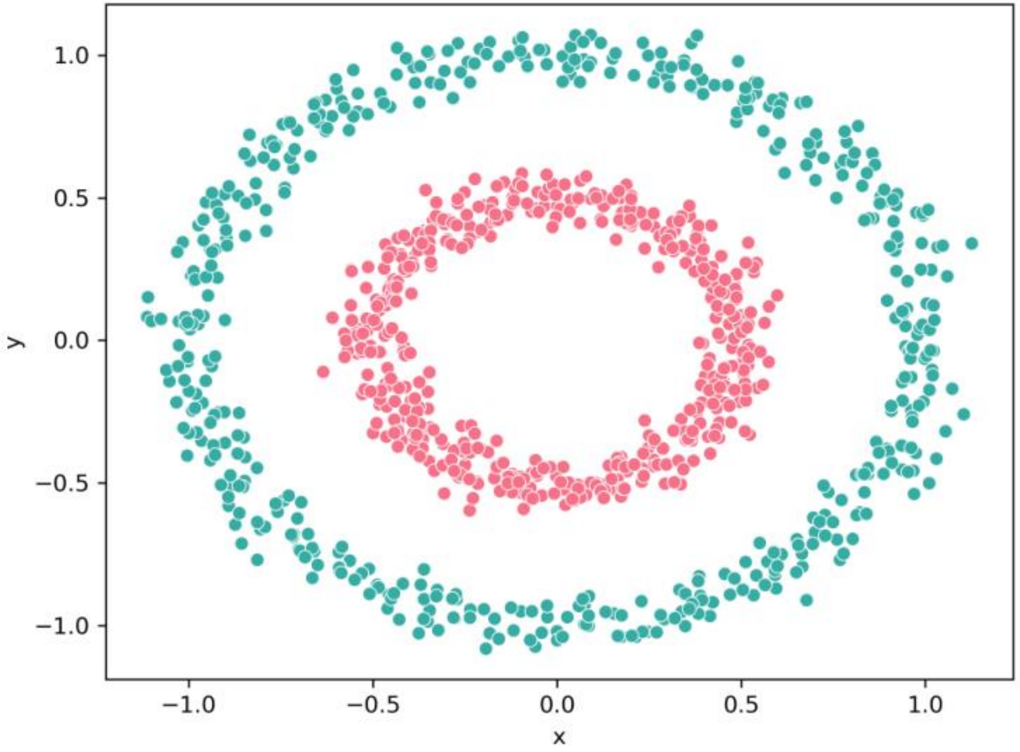
Aside: the "phylogenetic tree" (dendogram of species)



Source: Wikipedia.org

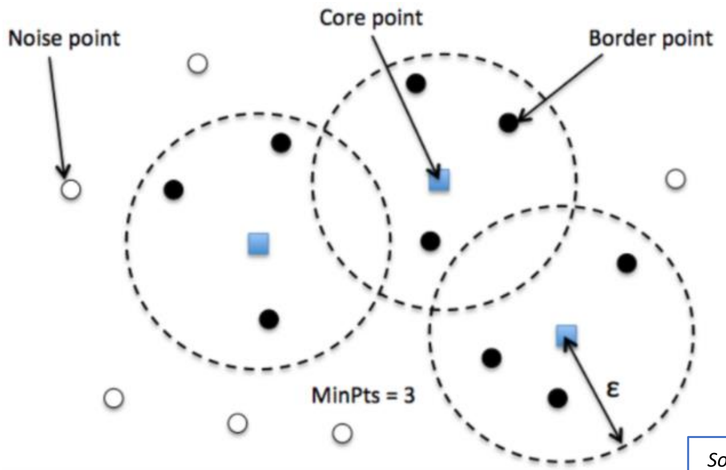
Density-based clustering (DBSCAN) is useful for data that is not inherently spherical

- Example from module 5 (noisy circles):



DBSCAN, under the hood (briefly)

- Two tuning parameters: a radius (epsilon) and a number of points (min pts)
- The algorithm categorizes every data point in one of three categories:
  1. A core point: if min pts are within epsilon of the point.
  2. A border point: not a core point, but there is a path of points from the point to a core point, with each step of the path being at most epsilon.
  3. An outlier (or noise) point: all other points.

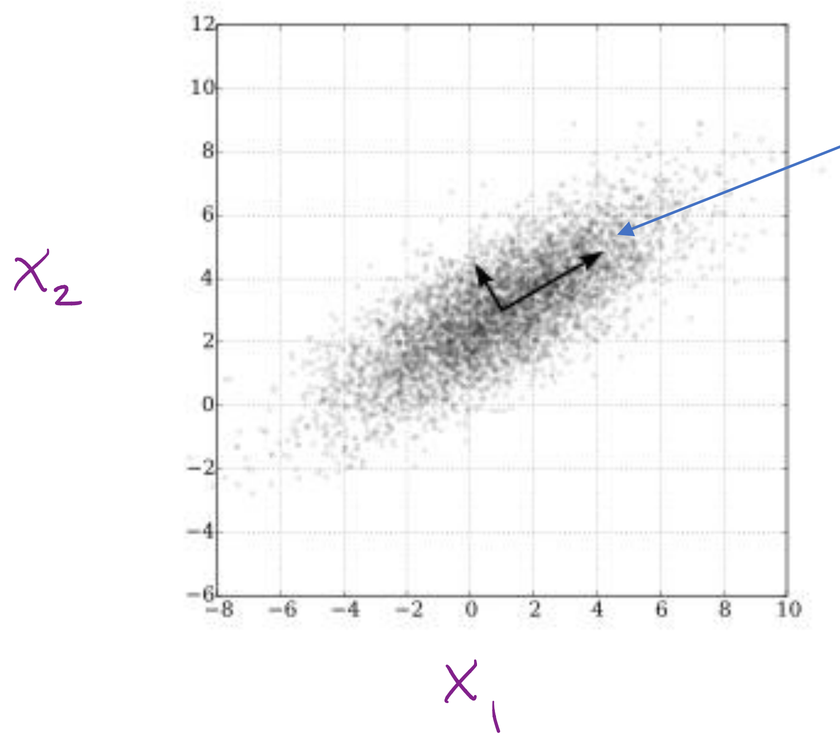


Source: https://www.kdnuggets.com/2020/04/dbscan-clustering-algorithm-machine-learning.html

Principal components analysis (PCA)

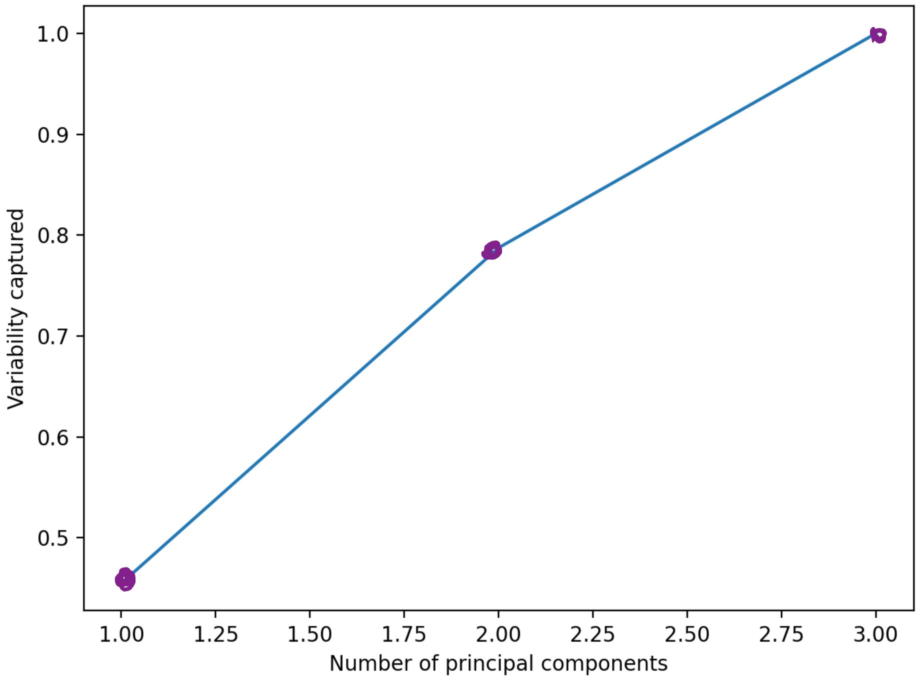
- In PCA, the goal is to see if a lower dimensional representation of the data explains much of the variation

Illustrative example in two dimensions:



These are the "principal components" – they are weighted combinations of the original features. The principal components form an "orthonormal basis" of the data.

PCA on the whiskey dataset




```
pc1 = list(zip(whisky_data_no_nuls.columns, np.round(best_pca_model.components_[0], 3)))
pc1
```

```
[('rating', -0.702), ('alcohol', 0.095), ('age', -0.706)]
```

```
pc2 = list(zip(whisky_data_no_nuls.columns, np.round(best_pca_model.components_[1], 3)))
pc2
```

```
[('rating', 0.11), ('alcohol', 0.994), ('age', 0.025)]
```

```
pc3 = list(zip(whisky_data_no_nuls.columns, np.round(best_pca_model.components_[2], 3)))
pc3
```

```
[('rating', -0.704), ('alcohol', 0.06), ('age', 0.708)]
```

PCR

Classic application of clustering: customer segmentation

- Illustrative example: segmenting customers purchasing wine promotions

32

3~~1~~ wine promotions (or offers):

	Campaign	Varietal	Minimum Qty (ltr)	Discount (%)	Origin	Past Peak	Offer #
0	January	Malbec	72	56	France	False	1
1	January	Pinot Noir	72	17	France	False	2
2	February	Espumante	144	32	Oregon	True	3
3	February	Champagne	72	48	France	True	4
4	February	Cabernet Sauvignon	144	44	New Zealand	True	5
5	March	Prosecco	144	86	Chile	False	6
6	March	Prosecco	6	40	Australia	True	7
7	March	Espumante	6	45	South Africa	False	8
8	April	Chardonnay	144	57	Chile	False	9
9	April	Prosecco	72	52	California	False	10
10	May	Champagne	72	85	France	False	11
11	May	Prosecco	72	83	Australia	False	12
12	May	Merlot	6	43	Chile	False	13
13	June	Merlot	72	64	Chile	False	14
14	June	Cabernet Sauvignon	144	19	Italy	False	15
15	June	Merlot	72	88	California	False	16
16	July	Pinot Noir	12	47	Germany	False	17
17	July	Espumante	6	50	Oregon	False	18
18	July	Champagne	12	66	Germany	False	19
19	August	Cabernet Sauvignon	72	82	Italy	False	20
20	August	Champagne	12	50	California	False	21
21	August	Champagne	72	63	France	False	22
22	September	Chardonnay	144	39	South Africa	False	23
23	September	Pinot Noir	6	34	Italy	False	24
24	October	Cabernet Sauvignon	72	59	Oregon	True	25
25	October	Pinot Noir	144	83	Australia	False	26
26	October	Champagne	72	88	New Zealand	False	27
27	November	Cabernet Sauvignon	12	56	France	True	28
28	November	Pinot Grigio	6	87	France	False	29
29	December	Malbec	6	54	France	False	30
30	December	Champagne	72	89	France	False	31
31	December	Cabernet Sauvignon	72	45	Germany	True	32

324 transactions (which of 100 customers purchased which promotions):

	Customer Last Name	Offer #
0	Smith	2
1	Smith	24
2	Johnson	17
3	Johnson	24
4	Johnson	26
...	...	...
319	Fisher	11
320	Fisher	22
321	Fisher	28
322	Fisher	30
323	Fisher	31

324 rows × 2 columns

- How would we see if the customers naturally fall into separate groups?

Wine customer segmentation purchase matrix

- We can cluster customers according to how similar their offer purchase behavior is:

Offer #	1	2	3	4	5	6	7	8	9	10	...	23	24	25	26	27	28	29	30	31	32
Customer Last Name																					
Adams	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0
Allen	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	...	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0
Anderson	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	1.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
Bailey	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0
Baker	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	1.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
Williams	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0
Wilson	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0
Wood	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0
Wright	0.0	0.0	0.0	1.0	0.0	1.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0
Young	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	1.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0

100 rows × 32 columns

These are the data points we can cluster.

Python code and data: https://github.com/bnsheehy/Customer_Clustering Adapted from chapter 6 of *Data Smart* by John W. Foreman.

Some optimal clusters with k=8

- Cluster 1 (head of dataframe and sorted):

	Offer #	Campaign	Varietal	Minimum Qty (ltr)	Discount (%)	Origin	Past Peak	0	1	2	3	4	5	6	7
23	24	September	Pinot Noir	6	34	Italy	False	0.0	12.0	0.0	0.0	0.0	0.0	0.0	0.0
25	26	October	Pinot Noir	144	83	Australia	False	0.0	8.0	0.0	0.0	0.0	0.0	3.0	4.0
16	17	July	Pinot Noir	12	47	Germany	False	0.0	7.0	0.0	0.0	0.0	0.0	0.0	0.0
1	2	January	Pinot Noir	72	17	France	False	0.0	4.0	0.0	0.0	0.0	3.0	0.0	3.0
30	31	December	Champagne	72	89	France	False	0.0	0.0	0.0	14.0	1.0	1.0	1.0	0.0
29	30	December	Malbec	6	54	France	False	8.0	0.0	10.0	1.0	0.0	3.0	0.0	0.0
28	29	November	Pinot Grigio	6	87	France	False	8.0	0.0	5.0	1.0	0.0	0.0	0.0	3.0
27	28	November	Cabernet Sauvignon	12	56	France	True	0.0	0.0	0.0	1.0	0.0	4.0	0.0	1.0
26	27	October	Champagne	72	88	New Zealand	False	0.0	0.0	0.0	1.0	3.0	1.0	0.0	4.0
24	25	October	Cabernet Sauvignon	72	59	Oregon	True	0.0	0.0	1.0	0.0	0.0	4.0	1.0	0.0

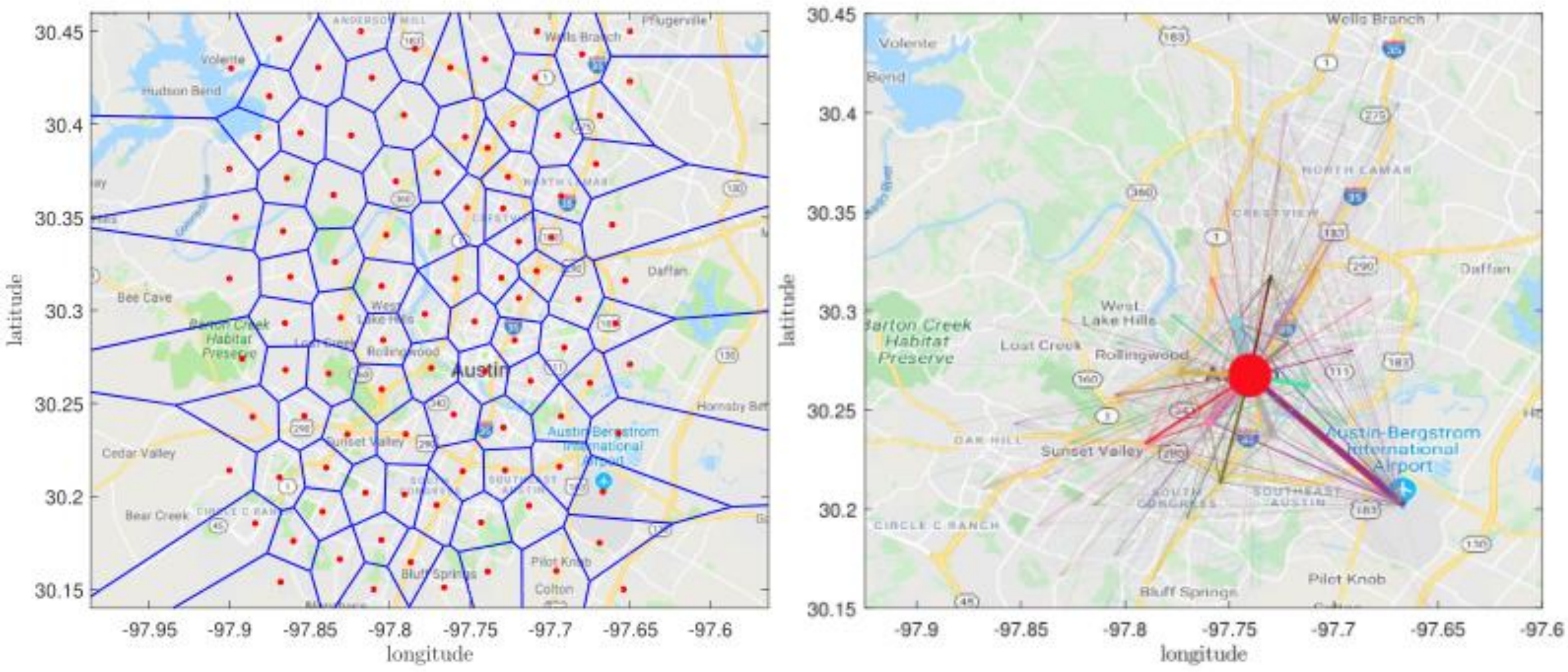
Some optimal clusters with k=8 (continued)

- Cluster 2 (head of dataframe and sorted):

	Offer #	Campaign	Varietal	Minimum Qty (ltr)	Discount (%)	Origin	Past Peak	0	1	2	3	4	5	6	7
29	30	December	Malbec	6	54	France	False	8.0	0.0	10.0	1.0	0.0	3.0	0.0	0.0
17	18	July	Espumante	6	50	Oregon	False	4.0	0.0	9.0	1.0	0.0	0.0	0.0	0.0
7	8	March	Espumante	6	45	South Africa	False	4.0	0.0	6.0	1.0	2.0	1.0	0.0	6.0
28	29	November	Pinot Grigio	6	87	France	False	8.0	0.0	5.0	1.0	0.0	0.0	0.0	3.0
12	13	May	Merlot	6	43	Chile	False	4.0	0.0	2.0	0.0	0.0	0.0	0.0	0.0
10	11	May	Champagne	72	85	France	False	0.0	0.0	1.0	2.0	0.0	7.0	2.0	1.0
24	25	October	Cabernet Sauvignon	72	59	Oregon	True	0.0	0.0	1.0	0.0	0.0	4.0	1.0	0.0
5	6	March	Prosecco	144	86	Chile	False	0.0	0.0	1.0	3.0	5.0	0.0	3.0	0.0
8	9	April	Chardonnay	144	57	Chile	False	0.0	0.0	1.0	2.0	0.0	0.0	3.0	4.0
9	10	April	Prosecco	72	52	California	False	0.0	0.0	1.0	3.0	0.0	0.0	1.0	2.0

Example of clustering in my research

- Project involved studying dynamic pricing techniques in ridesharing; tested models on data from RideAustin (Fuqua Media Relations piece)



Summary

- In unsupervised learning, we are trying to better understand the structure of our data, but have no specific predictive goal
  - Clustering is the canonical example, and there are many clustering methods
    - K-means
    - Hierarchical
    - Density-based
  - In PCA, we try to see if a lower dimensional representation of the data suffices

- These techniques are powerful, but require supervision! In clustering for example, we still need to tune various parameters.

Looking ahead to next time (Class 6)

- Homework 5 due at 11:59pm on Monday
  - Main goals: practice your understanding of clustering and dimensionality reduction
  - TA support available over the weekend!

- Class 6:
  - Data analytics in “the real world”
  - Course review and wrap-up
  - Homework 6 posted!

A joke for the weekend…