Clustering
What is Unsupervised Learning?

- Unlike in supervised learning, in unsupervised learning, there are no labels
- We simply search for patterns in the data

Examples

- Clustering
- Density Estimation
- Dimensionality Reduction
What is Clustering?

- A form of unsupervised learning used to separate a large group of observations into smaller subgroups of similar observations
- Examples
  - Topic modeling: clustering documents by subject (politics, sports, etc.)
  - Identifying hot spots of police or gang violence in urban areas
  - Image segmentation, gene expression, etc.
- Relevance to EDA
  - Clustering is used to identify and visualize patterns in data
  - Helps identify outliers and/or with formulating hypotheses
$k$-means Clustering

- $k$-means clustering is the most common type of clustering
- The $k$ in $k$-means refers to the number of clusters
- Pros:
  - It is easy to understand and to implement, so it can produce quick and dirty results
- Cons:
  - You must specify $k$ in advance; if $k$ is too large, it will find clusters where there are none; if $k$ is too small, it will miss “real” clusters
Remember This?

$k$-means clustering requires that you specify how many clusters you want to group the dance attendees into.

Let’s pick three clusters, so $k = 3$. 

Image and example from John W. Foreman’s *Data Smart* book
A Middle School Dance

- $k$-means clustering starts with three initial points (cluster centers), one per cluster, spread out across the dance floor.
- Dancers are assigned to the cluster that’s nearest to them.
- The algorithm then slides the cluster centers and their corresponding clusters around until it finds a good fit.

Image and example from John W. Foreman’s *Data Smart* book
A Middle School Dance

The algorithm is initialized with three centers (three black circles).

Each data points is then assigned to the nearest center.

In effect, this operation divides the space into three clusters, which are depicted here as variously shaded regions.
Cluster centers are then recomputed.

Observe how they move towards the data points.

And the process continues (until some stopping criterion is met).
A Middle School Dance

Final form!

Note the locations of the cluster centers and the divisions between clusters.
What do the clusters mean?

- It is never a good idea to take an algorithm’s word for it. We must always apply human insight to interpret an algorithm’s output.
- In the case of clustering, we ask what the clusters might signify?
- For a middle school dance, they could be cliques. The kids might be too timid to dance with kids outside their comfort zone!
- $k$-means allows us to cluster data, but we cannot accept a clustering if we cannot attribute meaning to the clusters. We must be able to understand the why behind the assignment.
How does it work?

Step 1: Choose a desired number of clusters, $k$

Step 2: Randomly assign each data point to an initial cluster

Step 3: Compute cluster centroids (this is the $k$-centroids algorithm right?)

Step 4: Re-assign each point to the closest cluster centroid

Step 5: Re-compute cluster centroids

Step 6: Repeat steps 4 and 5 until a stopping criterion is met
We’re still missing something key!

- Goal: to group “similar” objects into meaningful subgroups, called clusters
- Points on the Cartesian plane are similar if the distance between them is small
- But more generally, how do we define the similarity between two observations? Answer: we use a metric.

A note on nomenclature: Sometimes we speak of the similarity of two observations, and other times we speak of their difference/distance. A term that captures both similarity and distance is proximity.
What is a metric?

- A way to gauge similarity (or dissimilarity) among things
- A distance metric satisfies these three properties:
  - Symmetry: A is the same distance from B as B is from A
  - Positivity: Always non-negative (What would a negative distance mean?)
  - Triangle Inequality: $A + B > C$, $B + C > A$, and $A + C > C$

Note: A measure of similarity does not have to be a distance metric to be useful!
Euclidean Distance

- Distance “as the crow flies” is called Euclidean distance.
- In two dimensions: $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$
- In $n$ dimensions:
  $\sqrt{\sum_{i=1}^{n}(p_i - q_i)^2}$

$(1, 3)$

$D = \sqrt{(1 - 4)^2 + (3 - 2)^2}$
$= \sqrt{9 + 1} = \sqrt{10}$
$= 3.16$
Manhattan Distance

Euclidean distance probably isn’t the best measure of distance in Manhattan, because the streets form a grid. A better idea:

\[ |x_1 - y_2| + |x_2 - y_2| \quad \text{or} \quad \sum_{i=1}^{n} |p_i - q_i| \]

\[ D = |1 - 4| + |3 - 2| = 4 \]
Cosine Similarity

- Metrics need not capture spatial/physical distance
- Cosine similarity is the cosine of the angle between two vectors
- It measures the extent to which two vectors point in the same direction
  - 1 if the vectors point in exactly the same direction
  - 0 if the vectors form a right angle
  - -1 if the vectors point in completely opposite directions

If $\theta$ is the angle between two points, then cosine similarity is just $\cos(\theta)$
Cosine Similarity

Pointwise multiplication: e.g.,
\[(a_x)(b_x) + (a_y)(b_y) = (2)(1) + (2)(-3) = 2 - 6 = -4\]

\[
\frac{a \cdot b}{||a||_2 + ||b||_2}
\]

Euclidean distance of points \(a\) and \(b\) from origin

For \(a = (2, 2)\) and \(b = (1, -3)\):

\[
\frac{2 \cdot 1 + 2 \cdot -3}{\sqrt{2^2 + 2^2} + \sqrt{1^2 + (-3)^2}}
\]

\[
\frac{-4}{6} = -0.67
\]
Pearson’s Correlation Coefficient

● Remember correlation?
  ○ Positive if when $X$ increases, $Y$ also increases
  ○ Negative if when $X$ increases, $Y$ also decreases (or vice versa)

● Let’s apply the same idea to two $n$-dimensional observations

\[
r = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^{n} (y_i - \bar{y})^2}}
\]
Hamming Distance

The metrics we’ve discussed so far make sense for data with numerical features. There are other metrics for categorical data!

Given two ordered vectors of the same length (i.e., representing the same features), the Hamming distance is the number of feature values that differ.

Assume we have compiled a group of people’s preferences for their favorite (food, movie, book, subject, season):

- Person A: (banana, The Stranger, Forrest Gump, CS, Winter)
- Person B: (apple, Harry Potter, Star Trek, CS, Winter)
- Person A and B differ in the food, movie, and book categories, so their Hamming distance is 3, and their normalized Hamming distance is $\frac{3}{5}$. 
Jaccard Similarity

Amazon might represent each of its users by a vector of the products they buy. But then, to measure similarity between users, almost all entries are 0, so all users look very similar. Jaccard similarity measures similarity among non-zeros.

E.g., \( x = (1,0,0,0,0,0,0,0,0,0) \) and \( y = (1,0,0,0,0,1,0,0,0,1) \)

\[
M_{11} = 1 \quad M_{10} = 0 \quad M_{01} = 2 \quad M_{00} = 7
\]

\[
J = \frac{M_{11}}{M_{01} + M_{10} + M_{11}} = \frac{1}{3}
\]

Simple Matching Coefficient = \( \frac{M_{00} + M_{11}}{M_{00} + M_{01} + M_{10} + M_{11}} = \frac{1}{10} \)
What makes a good clustering?

- The *intra-cluster* similarity is high
- The *inter-cluster* similarity is low

All of the aforementioned metrics measure intra-cluster similarity (i.e., *similarity within clusters*). To evaluate a clustering, we also need metrics to measure inter-cluster similarity: i.e., *similarity between one cluster and another*. 
Linkage Metrics

Given two clusters $A$ and $B$:

- **Complete**
  - Find the maximum distance between all pairs $a \in A$ and $b \in B$

- **Single**
  - Find the minimum distance between all pairs $a \in A$ and $b \in B$

- **Centroid**
  - Find the distance between the centroid of $A$ and the centroid of $B$

- **Mean/Average**
  - Find the average of the distance between all pairs $a \in A$ and $b \in B$
What makes a good clustering?

- The intra-cluster similarity is high
- The inter-cluster similarity is low

The quality of a clustering depends on the choice of similarity metric

- Lots of different choices of metrics!
- Think about what makes the most sense for your data
- **Important**: you can’t compare distances without first normalizing your data
  - E.g., sleep (on average 8) dominates GPA (on average 3), so a clustering by both would cluster only by sleep if the data were not first normalized
Hierarchical Clustering
Two Approaches

Agglomerative
  - Bottom-up approach
    - Each observation starts in its own cluster
    - Clusters are merged as one moves up the hierarchy
    - Until all observations are in the same cluster

Divisive
  - Top-down approach
    - All observations start in the same cluster
    - Clusters are divided as one moves down the hierarchy
    - Until each observation comprises its own cluster
Agglomerative Algorithm

- Initialize the algorithm with each data point in its own cluster
- Calculate the distances between all clusters
- Combine the two closest clusters into one
- Repeat until all data points are in the same cluster
Agglomerative Example
Agglomerative vs. Divisive
Clustered Iris data set
(the labels give the true flower species)