HUMAN LEARNING

• Memorization
  • Accumulate individual facts
  • Problems with this?
    • Not enough time to observe everything
    • Not enough memory to observe everything
HUMAN LEARNING

• Generalization
  • Deduct new facts from old ones
  • Some do this better than others
  • There is no single technique
  • Machine learning does this
WHAT IS A TREE

TRUE IMPOSTOR
CLASSIFICATION — TREE

IMPOSTOR
CLASSIFICATION — COLOR

![Trees](image1)

![Broccoli](image2)

![Imposter Tree](image3)
MACHINE LEARNING COMPONENTS

• Data representation, features
  • Color, height, width, number of leaves, GPS coordinate

• Metric for assessing the goodness of the model
  • Given a new object with properties above, how good is the model at guessing

• Optimization method for learning the model
  • Linear regression, etc
MACHINE LEARNING

• Build programs that learn
  • Automating automation

• Automatically learn to recognize complex patterns and make intelligent decisions based on data
  • Let the data do the work
  • Generate programs that create useful outputs from data
  • Inductive inference
INDUCTIVE INFERENCE

• Observe examples
  • Incomplete information about a statistical phenomenon
  • Draw general rules from specific instances

EXAMPLES
- Green circular top
- Brown stem

GENERALIZED MODEL

- Is this new object a tree?
- Can I separate my examples into groups?

PREDICT
WHAT CAN WE LEARN?

• Two broad classes of machine learning

• Is this new object a tree?
  • Supervised learning

• Can I uncover regularities/ anomalies in my examples?
  • Unsupervised learning
SUPERVISED LEARNING

• Many examples, each example has a label
• Infer a rule
• Predict what label a previously unseen object should get

Leafless tree  Evergreen tree  Autumn tree  Palm tree
Oak tree
SUPERVISED LEARNING

• Many examples, each example has a label
• Infer a rule
• Predict what label a previously unseen object should get
SUPERVISED LEARNING

• Many examples, each example has a label
• Infer a rule
• Predict what label a previously unseen object should get
SUPERVISED LEARNING EXAMPLE

• Predict whether a person speaks English

• Represent a person with
  • Eye color
  • Gender
  • Citizenship
  • Whether they speak English (Label)
SUPERVISED LEARNING EXAMPLE

• Given:

• P1 < blue, male, US > : speaks English
• P2 < brown, male, US > : speaks English
• P3 < blue, female, France > : does not speak English
SUPERVISED LEARNING EXAMPLE

• What rule(s) can we learn based on these features and these 3 people?

• For all \( x, y < x, y, \text{US} > : \text{speaks English} \)  
  
• For all \( x, z < x, \text{male}, z > : \text{speaks English} \)
PROBLEM WITH THE EXAMPLE

• How many examples?
  • 3

• How many features?
  • 3

• Number of examples is too small relative to the number of features
  • Easy to make false generalizations
HOW TO IMPROVE THE EXAMPLE

• Add more examples!
  • Take 100 million samples from around the globe

• More likely to learn the rule
  • For all $x, y < x, y, \text{ US }> :$ speaks English

• Does this mean that if you are from the US you speak English?
  • No
  • Use a method that allows for training error and learn something that is probably true
UNSUPERVISED LEARNING

• Many examples with features but no labels
• The rules aim to look for hidden structures
• Group examples into clusters
UNSUPERVISED LEARNING
UNSUPERVISED LEARNING
CLUSTERING IN EVERYDAY LIFE

• Marketing
  • Groups of customers with similar behavior (based on past buying records)

• Biology
  • Classify plants and animals given physical features

• Music
  • Group people who like the same songs based on listening history
UNSUPERVISED LEARNING

• Clustering
  • Important application of unsupervised learning
  • Find structure in a collection of unlabeled data

Cluster based on height
UNSUPERVISED LEARNING

• Clustering
  • Important application of unsupervised learning
  • Find structure in a collection of unlabeled data

![Diagram showing clusters based on weight and height](image-url)
UNSUPERVISED LEARNING

- **Clustering**
  - Important application of unsupervised learning
  - Find structure in a collection of unlabeled data

Cluster based on height and weight
WHAT MAKES A CLUSTERING GOOD

• Add another feature, eye color
• Good clustering depends on the application
BASKETBALL PLAYER CLUSTERS

• Want to cluster to choose who would be a good basketball player
SUMO WRESTLER CLUSTERS

• Want to cluster to choose who would be a good sumo wrestler
MUTANT CLUSTERS

• Want to cluster to choose who is a mutant
AN OPTIMIZATION PROBLEM

• Decided why you are clustering
• Next step is to look at it like an optimization problem
• Minimize/maximize an objective function
OUR OBJECTIVE

• Low intra-cluster dissimilarity
  • Points in same cluster are similar

• High inter-cluster dissimilarity
  • Points in different clusters should be different
MORE RIGOROUSLY

• Assess a set of clusters

\[ \text{variance}(c) = (\text{mean}(c) - x)^2 \]

\[ \text{badness}(C) = \sum_{x \in C} \sum_{c \in C} \text{variance}(c) \]

Mean = \( \frac{1+1+2+3+3}{5} = 2 \)
Var = \( (2-1)^2 + (2-1)^2 + (2-2)^2 + (2-3)^2 + (2-3)^2 = 4 \)

Mean = 8
Var = 0

Badness = 4
MORE RIGORously

• Assess a set of clusters

$$\text{variance}(c) = (\text{mean}(c) - x)^2$$

$$\text{badness}(C) = \sum_{c \in C} \text{variance}(c)$$

- Mean = 3
  - Mean = 8

- Var = 0
  - Var = 0

- $$\text{Mean} = \frac{1+1+2}{3} = 1.33$$
  - $$\text{Var} = (1.33-1)^2 + (1.33-1)^2 + (1.33-2)^2 = 0.67$$

- $$\text{Badness} = 0.67$$
BADNESS

• What is the simplest case? What clusters can we pick to always get a badness value of 0?
  • Each point is its own cluster

• Add additional constraints
  • Set a maximum number of clusters
  • Set a maximum distance between clusters
COMPUTING OPTIMAL SOLUTION

• Computationally intensive
  • Search all possible combinations of clusters!

• Resort to greedy algorithms
  • Hierarchical clustering
  • K-means clustering
HIERARCHICAL CLUSTERING

• Given a set of N items to be clustered, and an N*N distance (or similarity) matrix

• Start by assigning each item to a cluster, so that if you have N items, you now have N clusters, each containing just one item.

• Find the closest (most similar) pair of clusters and merge them into a single cluster, so that now you have one cluster less.

• Continue the process until all items are clustered into a single cluster of size N.
CLUSTERING MONTHLY TEMPS
HOW TO READ A DENDROGRAM
HOW TO READ A DENDROGRAM
HOW TO READ A DENDROGRAM
HOW TO READ A DENDROGRAM
HOW TO READ A DENDROGRAM
HOW TO READ A DENDROGRAM
HOW TO READ A DENDROGRAM
HOW TO READ A DENDROGRAM
HOW TO READ A DENDROGRAM
HOW TO READ A DENDROGRAM

[Diagram showing a dendrogram with months on the x-axis: Jan, Feb, Dec, Nov, Mar, Apr, Oct, May, Sep, Jun, Jul, Aug. The dendrogram has branches connecting the months, indicating the relationships between them.]
HOW TO READ A DENDROGRAM

12 clusters
HOW TO READ A DENDROGRAM

6 clusters
HOW TO READ A DENDROGRAM

3 clusters
DENDROGRAMS

• Show progression of merges
• Can easily tell elements in each cluster set
• Cluster sets will depend on your distance criteria
  • Single linkage
  • Complete linkage
  • Average linkage
CLUSTER DISTANCES

- Single linkage method
  - aka minimum linkage
  - Distance between 2 clusters is the smallest distance from a member in one cluster to a member in the other cluster
CLUSTER DISTANCES

• Single linkage method
  • aka minimum linkage
  • Distance between 2 clusters is the smallest distance from a member in one cluster to a member in the other cluster

Dist = 1
CLUSTER DISTANCES

• Complete linkage method
  • aka maximum linkage
  • Distance between 2 clusters is the largest distance from a member in one cluster to a member in the other cluster
CLUSTER DISTANCES

• Complete linkage method
  • aka maximum linkage
  • Distance between 2 clusters is the largest distance from a member in one cluster to a member in the other cluster

Dist = 5
CLUSTER DISTANCES

• Average linkage method
  • Distance between 2 clusters is the average distance from any member in one cluster to members in the other cluster
CLUSTER DISTANCES

• Average linkage method
  • Distance between 2 clusters is the average distance from any member in one cluster to members in the other cluster

\[
\text{Dist} = \frac{((4-1)+(5-1)+(6-1))}{3} + \frac{((4-2)+(5-2)+(6-2))}{3} + \frac{((4-3)+(5-3)+(6-3))}{3} \]

\[
\text{Dist} = 3
\]
K-MEANS CLUSTERING

• Start with a set of point and k, number of clusters
• Approximately minimize the objective function

\[ \sum_{c=1}^{k} \sum_{x \in c} \left\| x - \mu_c \right\|^2 \]

- clusters
- points
- centroid
- Distance from a point to the centroid
K-MEANS CLUSTERING

• Decide on a value for k – the total number of clusters one wants at the end
• Choose k points as the initial centroids (could be at random)
• Assign each point to the nearest centroid
• Choose a new centroid for each of the k clusters
• Assign each point to the nearest centroid
• Repeat prev steps until the change is “small”
HIERARCHICAL vs K-MEANS

• Hierarchical is deterministic
• K-means is non-deterministic

• Hierarchical looks at different numbers of clusters, 1 to N
• K-means looks at many ways of creating k clusters

• Hierarchical is slow
• K-means is fast
NEXT TIME

• Run times of these two clustering methods
• Weaknesses of these two clustering methods
• What happens when we have more than one feature
• How to compare features
• Code