Model Learning and Clustering

CPS170
Ron Parr

material from: Lise Getoor, Andrew Moore, Tom Dietterich, Sebastian Thrun, Rich Maclin

Unsupervised Learning

• Supervised learning: Data <x1, x2, ... xn, y>
• Unsupervised Learning: Data <x1, x2, ... xn>

• So, what’s the big deal?
• Isn’t y just another feature?
• No explicit performance objective
  – Bad news: Problem not necessarily well defined without further assumptions
  – Good news: Results can be useful for more than predicting y
Model Learning

- Produce a global summary of the data
- Not an exact copy
- Consider space of models M and dataset D
- One approach: Maximize $P(M|D)$
- How to do this? Bayes Rule:

$$P(M|D) = \frac{P(D|M)P(M)}{P(D)}$$

Example: Modeling Coin Flips

- Suppose we have observed: $D=HTTHT$
- Which is a better model?
  - $P(H=0.4)$
  - $P(H=0.5)$

$$P(M|D) = \frac{P(D|M)P(M)}{P(D)}$$

$P(D|P(H = 0.5)) = 0.5^5 = 0.312$

$P(D|P(H = 0.4)) = 0.4^2 * 0.6^3 = 0.3456$

What about $P(D)$ and $P(M)$???
Model Learning With Bayes Rule

\[ P(M \mid D) = \frac{P(D \mid M)P(M)}{P(D)} \]

- We call \( P(D \mid M) \) the *likelihood*
- We can ignore \( P(D) \)... Why?
- What about \( P(M) \)?
  - Call this a our *prior probability* on models
  - If \( P(M) \) is uniform (all models equally likely) then
    maximizing \( P(D \mid M) \) is equivalent to maximizing \( P(M \mid D) \)
    (Call this the *maximum likelihood* approach.)

Using Priors

- Suppose we have good reason to expect that the coin is fair
- Should we really conclude \( P(H)=0.4 \)?
- Suppose we think \( P(P(H=0.5)) = 2 \times P(P(H=0.4)) \)
- This means \( P(D \mid P(H=0.4)) \) must be 2X larger than \( P(D \mid P(H=0.5)) \) to compensate if \( P(H=0.4) \) is to maximize the *posterior probability*

\[ P(M \mid D) = \frac{P(D \mid M)P(M)}{P(D)} \]
Specifying Priors

- In our coin example, we considered just two models $P(H=0.4)$ and $P(H=0.5)$
- In general, we might want to specify a distribution over all possible coin probabilities

- This introduces complications:
  - $P(M)$ is now a distribution over a continuous parameter
  - Need to use calculus to find maximizer of $P(D|M)P(M)$
Conjugate Priors

- A likelihood and prior are said to be **conjugate** if their product has the same parametric form as the prior
- (This is outside the scope of the class, but we provide one nice example.)
- The beta distribution is conjugate to the binomial distribution
  - Can think of the beta distribution as specifying a number of "imagined" heads and tails
  - Maximum of the posterior adds together observed heads and tails with imagined heads and tails
- Examples:
  - Prior of 100 heads and 100 tails is a strong prior towards fairness
  - Prior of 1 head and 1 tail is a weak prior towards fairness

Clustering as Modeling

- Clustering assigns points in a space to clusters
- Example: By examining x-rays of cancer tumors, one might identify different subtypes of cancer based upon growth patterns
- Each cluster has its own probabilistic model describing how items of that cluster’s type behave
Examples of Clustering Applications

- **Marketing**: Help marketers discover distinct groups in their customer bases, and then use this knowledge to develop targeted marketing programs
- **Land use**: Identification of areas of similar land use in an earth observation database
- **Insurance**: Identifying groups of motor insurance policy holders with similar claim cost
- **City-planning**: Identifying groups of houses according to their house type, value, and geographical location
- **Earth-quake studies**: Observed earth quake epicenters should be clustered along continent faults

Example of Subtleties in Clustering

- **Household Dataset**: location, income, number of children, rent/own, crime rate, number of cars

- **Appropriate clustering may depend on use**:
  - Goal to minimize delivery time ⇒ cluster by location
  - Others?
  - Clustering work often suffers from mismatch between the clustering objective function and the performance criterion
Clustering Desiderata

- Decomposition or partition of data into groups so that
  - Points in one group are **similar** to each other
  - Are as **different** as possible from the points in other groups
- Measure of **distance** is fundamental
- Explicit representation:
  - \( D(x(i),x(j)) \) for each \( x \)
  - Only feasible for small domains
- Implicit representation by measurement:
  - Distance computed from features
  - Implement this as a function

Families of Clustering Algorithms

- **Partition-based methods**
  - e.g., K-means
- **Hierarchical clustering**
  - e.g., hierarchical agglomerative clustering
- **Probabilistic model-based clustering**
  - e.g., mixture models
- **Graph-based Methods**
  - e.g., spectral methods
K-means

- Start with randomly chosen cluster centers
- Assign points to closest cluster
- Recompute cluster centers
- Reassign points
- Repeat until no changes

K-means example
K-means example
K-means example

K-means example #2
K-means example #2
Demo

http://home.dei.polimi.it/matteucc/Clustering/tutorial_html/AppletKM.html

Complexity

• Does algorithm terminate?  
  yes

• Does algorithm converge to optimal clustering?  
  Can only guarantee local optimum

• Time complexity one iteration?  
  nk
Understanding k-Means

- Implicitly models data as coming from a Gaussian distribution centered at cluster centers
- \( \log P(\text{data}) \sim \text{sum of squared distances} \)

\[
P(x_i \in c_j) \propto e^{-\|x_i - c_j\|^2}
\]

\[
P(\text{data}) = \prod_i P(x_i \in c_{\text{clustering}(i)})
\]

\[
\log(P(\text{data})) = \alpha \sum_i (x_i - c_{\text{clustering}(i)})^2
\]

Understanding k-Means II

- Each step of k-Means increases \( P(\text{data}) \)
  - Reassigning points moves points to clusters for which their coordinates have higher probability
  - Recomputing means moves cluster centers to increase the average probability of points in the cluster
- Fixed number of assignments and monotonic score implies convergence
Understanding k-Means III

\[ P(M \mid D) = \frac{P(D \mid M) P(M)}{P(D)} \]

- Can view k-means as max likelihood method with a twist
  - Unlike the coin toss example, there is a hidden variable with each datum – the cluster membership
  - k-means iteratively improves its guesses about these hidden pieces of information
- k-means can be interpreted as an instance of a general approach to dealing with hidden variables called Expectation Maximization (EM)

But How Do We Pick k?

- Sometimes there will be an obvious choice given background knowledge or the intended use of the clustering output
- What if we just iterated over k?
  - Picking \( k=n \) will always maximize \( P(D \mid M) \)
  - We could introduce a prior over models using \( P(M) \) in Bayes rule
- Compare prior over models with regularization:
  - Regularization in regression penalized overly complex solutions
  - We can assign models with a high number of clusters low probability to achieve a similar effect
  - (In general, use of priors subsumes regularization.)
Is Clustering Well Defined?

- Clustering is not clearly axiomatized
- Can we define an “optimal” clustering w/o specifying an a priori preference (prior) on the cluster sizes or making additional assumptions?
- Kleinberg: Clustering is impossible under some plausible assumptions (IOW, union of unstated assumptions made by clustering algorithms is logically inconsistent)
- Recent efforts make progress putting clustering on more solid ground

Model Learning Conclusion

- Often seek to find the most likely model given the data
- Can be viewed as maximizing the posterior P(M|D) using Bayes rule
- Model learning can be applied to:
  - Coin flips
  - Clustering
  - Learning parameters of Bayes nets or HMMs
  - etc.
- Some care must go into formulation of modeling assumptions to avoid degenerate solutions, e.g., assigning every point to its own cluster
- Priors can help avoid degenerate solutions