04/5.

MLA

K- Nearest NEIGBOURS

- Classification that is non parametric / distribution free method of assigning group membership
- Makes no assumption about spread of data within each class
- Consequence of this is that class assignment is fixed, with no measurement of uncertainty concerning any particular assignment
- Classification techniques that do not make distributional assumptions at least allow quantification of the uncertainty in group membership

- K nearest neighbour simply look at the k-closest points of known origin to the point of unknown origin
- The point is then classified as belonging to the group which contains the most of these k point's
- Results are not invariant to the scaling of the original variables, nor to the method in which distance is calculated, as these can change the nearest neighbour for any particular point.

Advantage: No assumptions, so assumptions can't be wrong
Disadvantage: Cannot expect probabilistic uncertainty about our classification ie class assignment is fixed eg point A is in class A as opposed to 65% point is in class A.

We look at the K nearest point of known label to our new point and classify it as belonging to the class which is most prevalent.

Choosing K.
- The classification varies with K.
- One approach for chosing K is to split labelled data into 3 parts:
  → Training - parts whose labels are used to classify unlabelled points 50%
  → Test - parts we know the label for but consider unlabelled in order to find best K to classify 25%
  → Validation - Remaining labelled are considered unlabelled in order to estimate classification error R 25%
  best K in test step

- Plot as function of k the proportion incorrectly classified

Why Validate?
- The correct classification rate for test data typically overestimated the percentage correct classifications in validation data. Because value of k is chosen specifically for the test set, and may not be representative for another unlabelled sample
- Validation data is not used at any stage of the model fitting, and hence offers a more reasonable estimate of the correct classification rate

Cross-Validation
- In the problem (leave-one-out) cross validation would be used as follows:
  → For each value of k remove each data point and determine if that data point would be correctly classified knowing the labels of all other data points e.g. 100 points, making 100 classifications of 1 point based on labels of other 99
  → Example. 3 data points (x_1, x_2, x_3) See if we correctly classify x_1 given labels for x_2 x_3, x_2 for x_1, x_3 x_3 for x_1, x_2.
  Each value of k will either get non correct, one correct, two or all correct
  → Select value of k that has best classification rate

K MEANS
- Aim is to divide the data into k distinct groups so that observations within a group are similar, whilst observations between groups are different
- K-means clustering is an iterative, rather than hierarchical, clustering algorithm
- Means that at each stage of algorithm data point will be assigned to a fixed number of clusters (contrasts with hierarchical clustering where the number of clusters ranges from the number of data points to a single cluster)
- Can select k from expert knowledge
- Or use previous results from preliminary data exploration
- Simple and computationally efficient, can be sensitive to selection of starting points
- Running K-means several times with different start points can help check robustness of results

MLA

24/15 K-MEANS

Pseudo Code

1. Choose the number of clusters K and designate cluster centers
2. Assign each data point to the cluster whose center is closest.
3. For cluster i, calculate its centroid $C_i^T = (C(i)_1, C(i)_2 \ldots C(i)_m)$ where m is number of variables in observation → these are found by averaging variable scores for data points within the cluster.
4. Calculate the sum of squared distance to each object to its cluster centroid:

$$SS = \sum_{i=1}^{n} \sum_{j=1}^{m} (x_{ij} - C(i)_j)^2$$

Assume a total of n observations. Minimize SS value

5. Re-assign each observation to the cluster whose centroid is closest
6. Repeat (3) → (5) until convergence

- Randomly assign centers of each group
- Initial partition can be constructed in several ways:
  - A random selection of K observations
  - Specify selection based on prior knowledge
  - By using results from an exploratory hierarchical clustering algorithm

- label point according to which center is closest
- update values for the three centers (provisional)
- repeat

- K means has converged when no points are moved between groups on an iteration
- Once this happens, centers will no longer change, points etc
- No convergence criteria might not be suitable in some cases e.g. if n is very large, and alternatives are possible, e.g. within cluster sum of squares does not change over 3 iterations etc

Choosing value for k
- Guideline
- Generally shall run K mean for different number of value
- Why not chose k that minimize SS? the more cluster there are fitted be smaller the D
- General rule is to plot K against SS and look for a "kink" in the curve If there is no kink, then there is a trade off between additional complexity by increasing k and better fit by reducing the SS

Local Minima
- K mean algorithm can give different answer when initial at different starting values
- Mean algo will not always find min value of the Total Within Sum of Squares
- Chose lowest TWSS

K-Means Clustering - Model Based
- Model based clustering make use of statistical method and is this parameter in nature

- Several possible extension
- we used euclidean distance, and new center as mean value of vector assigned to that group
- Choice of dissimilarity and choice of center are related

Extension
- Can use different measure of dissimilarity
- Could assign x_i to group k so that $d(x_i, \mu_k)$ is minimal but use different measure of dissimilarity that euclidean.
- Can help algorithm from forming circular clusters
- The new center for the group could be computed by minimizing

$$\sum_{i=1}^{n} d(x_i, \mu_k) I_{ik}$$

Where $I_{ik} = 1$ if observation i is assigned to cluster k and 0 otherwise
- The partioning around medoid algorithm does this

MLA 5

K-MEANS CLUSTERING

Cluster Medoids:
- A medoid is a representative object of a cluster so that its average dissimilarity to all the data points in the cluster is minimal
- Unlike the mean or centroid used in k-means, a medoid has to be an actual data point within the cluster
- like mean & median
- Medoid used in applications where a mean or centroid are conceptually difficult to conceptually understand, or it may not even be defined e.g. 3D-bezeiers

Partering around Medoid (PAM) Pseudo Code
1. Select a dissimilarity metric to be used
2. Initialize by selecting k of the n data points to be medoids
3. Cluster each data point to belong to the same group as the medoid it is closest to under the dissimilarity metric selected
4. For each medoid x^*:
   - For each non-medoid point x_i swap x^* with x_i and compute the total dissimilarity cost of the configuration
5. Select the configuration with lowest total dissimilarity cost.
6. Repeat 3 to 5 until convergence i.e. no change in medoid.

Mixture Model
- Suppose we have data $x_i^T (x_{i1}, x_{i2} \dots x_{im})$ which is known to come from one of k populations
- Within each population (cluster) the data follow a density $F(x_i|\theta_I)$ for $I = 1 \dots k$ where θ_I are the parameter governing population I
- Suppose the probability that a data point is from population j is π_j
- Then the data can be modelled using a mixture model:

$$P(x_i) = \sum_{j=1}^{k} P(x_i \in j) P(x_i | x_i \in j)$$

$$p(x_i) = \sum_{j=1}^{k} \pi_j F(x_i|\theta_I)$$

- Mixture model can be used to form a model based clustering technique

Mixture Models: ME1

- The π_j are called or called the mixing proportions, and $f(\cdot|\theta_j)$ is known as the j^{th} component density
- Models offer good model flexibility by allowing both μ and the model parameters within each population to vary
- Common form of MM is the mixture of normals, where each component density is a normal distribution density for multivariate normal density

In the univariate case: $p(x_i) = \sum_{j=1}^{k} \pi_j \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(x_i-\mu_j)^2}{2\sigma_j^2}\right]$

- Assumes that the data in each population is normal with mean μ_j and variance σ_j^2
- Similar idea to LDA, QDA

Cluster Shapes

- K means bias for circular shapes
- LDA fits ellipses of the same shape and direction (equal variance matrix assumption)
- QDA allows for different cov matrices between groups i.e. different shapes and directions
- Model based clustering allow different shapes and directions as well as varying k.

Multivariate Mixture Models: M7

- Normal MM can be easily extended to a multivariate mixture model
- In this case it is assumed that the data within group j follows a multivariate normal distribution with mean μ_j and cov matrix Σ_j
- We could constrain cov matrix in different ways to allow model flexibility

Decomposing Cov Matrix

- Can decompose using eigenvalue decomposition: $\Sigma = \lambda DAD'$
  - λ = a constant
  - D = orthogonal matrix of eigenvectors
  - A = Diagonal matrix where entries proportional to eigenvalues

MLA 2

K-means

Decomposing Cov Mat

- Seen that the contours of the density for a MVN form ellipses, the shape can be controlled by the cov mat Σ.

A - control shape of ellipse
D - control orientation of ellipse
λ - control size of the ellipse

- If A=D=1 ellipse becomes circle
- If D=1 and A is unrestricted, ellipse becomes aligned with axes
- When we move to mixture of normals, the flexibility increases. This is the basis for model based clustering

- Main aim is to find clusters in data where K is unknown (unsupervised)
- Idea is to fit a mixture of ~~normals~~ normals to the data set for a range of possible values for K and for a range of different possibilities for the Σ_k's
- Use a MLE approach via Expectation Maximisation (EM), this is generally what happens in K-means and PAM algorithms
- For a given cluster number K and a likelihood model $L(\theta|X,z)$ for the probability of parameters θ given data set X and cluster alignment z the pseudo EM algorithm should between the following:
  - → E-step: find expected value of θ as a function of z using given likelihood model
  - → M-step: change cluster alignment z to max the expected likelihood from E-step

How to chose optimum model?

- Use Bayesian Information Criterion
  BIC = -2 x maximised likelihood of data + (log N x # of parameters)
- Take model with smallest BIC
- First term rewards good fit, second term penalises complexity (parsimonious)
- Model based clustering will return optimal # of groups in data and indicate optimal constraint decomposition of the Σ_k's (can incorporate it into likelihood model). Can return estimates of group membership of each point and an estimate of the uncertainty in the group assignment