School of Computing and Information Systems
The University of Melbourne
COMP90049 Introduction to Machine Learning (Semester 1, 2023)
Week 6

1. Given the following dataset, we wished to perform feature selection on this dataset, where the class is PLAY:

ID	*Outl*	*Temp*	*Humi*	*Wind*	PLAY
A	s	h	h	F	N
B	s	h	h	T	N
C	o	h	h	F	Y
D	r	m	h	F	Y
E	r	c	n	F	Y
F	r	c	n	T	N
G	o	c	n	T	?
H	s	m	h	F	?

(i). Which of `Humi` and `Wind` has the greatest *Pointwise Mutual Information* for the class Y? What about N?

To determine Pointwise Mutual Information (PMI), we compare the joint probability to the product of the prior probabilities as follows:

$$PMI(A,C) = log_2 \frac{P(A \cap C)}{P(A)P(C)}$$

Note that this formulation only really makes sense for binary attributes and binary classes (which we have here.)

$$PMI(Humi = h, Play = Y) = log_2 \frac{P(Humi = h \ \cap Play = Y)}{P(Humi = h)P(Play = Y)}$$

$$= log_2 \frac{\frac{2}{6}}{\frac{4}{6}\frac{3}{6}} = log_2(1) = 0$$

$$PMI(Wind = T, Play = Y) = log_2 \frac{P(Wind = T \cap Play = Y)}{P(Wind = T)P(Play = Y)}$$

$$= log_2 \frac{\frac{0}{6}}{\frac{2}{6}\frac{3}{6}} = log_2(0) = -\infty$$

So, we find that Wind=T is (perfectly) negatively correlated with PLAY=Y; whereas Humi=h is (perfectly) uncorrelated.

You should compare these results with the negative class PLAY=N, where Wind=T is positively correlated, but Humi=h is still uncorrelated.

(ii). Which of the attributes has the greatest *Mutual Information* for the class, as a whole?

A general form of the Mutual Information (MI) is as follows:

$$MI(X,C) = \sum_{x \in X} \sum_{c \in \{Y,N\}} P(x,c)PMI(x,c)$$

Effectively, we're going to consider the PMI of every possible attribute value–class combination, weighted by the proportion of instances that actually had that combination.

To handle cases like PMI(Wind) above, we are going to equate 0 log 0 ≡ 0 (which is true in the limit anyway).

For Outl, this is going to look like:

$$MI(Outl) = P(s,Y)PMI(s,Y) + P(o,Y)PMI(o,Y) + P(r,Y)PMI(r,Y) +$$

$$P(s,N)PMI(s,N) + P(o,N)PMI(o,N) + P(r,N)PMI(r,N)$$

$$= \frac{0}{6}log_2\frac{\frac{0}{6}}{\frac{2}{6}\frac{3}{6}} + \frac{1}{6}log_2\frac{\frac{1}{6}}{\frac{1}{6}\frac{3}{6}} + \frac{2}{6}log_2\frac{\frac{2}{6}}{\frac{3}{6}\frac{3}{6}} +$$

$$\frac{2}{6}log_2\frac{\frac{2}{6}}{\frac{2}{6}\frac{3}{6}} + \frac{0}{6}log_2\frac{\frac{0}{6}}{\frac{1}{6}\frac{3}{6}} + \frac{1}{6}log_2\frac{\frac{1}{6}}{\frac{3}{6}\frac{3}{6}}$$

$$\approx 0\ log_2 0 + (0.1667)(1) + (0.3333)(0.4150) +$$

$$(0.3333)(1) + 0\ log_2 0 + (0.1667)(-0.5850)$$

$$\approx 0.541$$

It's worth noting that while some individual terms can be negative, the sum must be at least zero.

For Temp, this is going to look like:

$$MI(Temp) = P(h,Y)PMI(h,Y) + P(m,Y)PMI(m,Y) + P(c,Y)PMI(c,Y) +$$

$$P(h,N)PMI(h,N) + P(m,N)PMI(m,N) + P(c,N)PMI(c,N)$$

$$= \frac{1}{6}log_2\frac{\frac{1}{6}}{\frac{3}{6}\frac{3}{6}} + \frac{1}{6}log_2\frac{\frac{1}{6}}{\frac{1}{6}\frac{3}{6}} + \frac{1}{6}log_2\frac{\frac{1}{6}}{\frac{2}{6}\frac{3}{6}} +$$

$$\frac{2}{6}log_2\frac{\frac{2}{6}}{\frac{2}{6}\frac{3}{6}} + \frac{0}{6}log_2\frac{\frac{0}{6}}{\frac{1}{6}\frac{3}{6}} + \frac{1}{6}log_2\frac{\frac{1}{6}}{\frac{2}{6}\frac{3}{6}}$$

$$\approx (0.1667)(-0.5850) + (0.1667)(1) + (0.1667)(0) +$$

$$(0.3333)(0.4150) + 0\ log_2 0 + (0.1667)(-0.5850)$$

$$\approx 0.110$$

We will leave the workings as an exercise, but the Mutual Information for Humi is 0, and for Wind is 0.459.

Consequently, Outl appears to be the best attribute (perhaps as we might expect), and Wind also seems quite good; whereas Temp is not very good, and Humi is completely unhelpful.

2. Consider the following dataset:

id	*apple*	*ibm*	*lemon*	*sun*	**label**
A	4	0	1	1	fruit
B	5	0	5	2	fruit
C	2	5	0	0	comp
D	1	2	1	7	comp
E	2	0	3	1	?
F	1	0	1	0	?

(i). Treat the problem as an unsupervised machine learning problem (excluding the *label* attributes) and calculate the clusters according to ***k*-means** with $k = 2$, using the

Manhattan distance, and instances A and F as starting seeds (initialized cluster centers).

This is an unsupervised problem, so we ignore (or don't have access to) the label attribute. (We're going to ignore id as well because it obviously isn't a meaningful point of comparison.)

We begin by setting the initial centroids for our two clusters, let's say cluster 1 has centroid C1 = (4, 0, 1, 1) and cluster 2 C2 = (1, 0, 1, 0).

We now calculate the (Manhattan) distance for each instance ("training" and "test" are equivalent in this context) to the centroids of each cluster:

d(A, C_1) = | 4 – 4 | + | 0 – 0 | + | 1 – 1 | + | 1 – 1 | = 0
d(A, C_2) = | 4 – 1 | + | 0 – 0 | + | 1 – 1 | + | 1 – 0 | = 4
d(B, C_1) = | 5 – 4 | + | 0 – 0 | + | 5 – 1 | + | 2 – 1 | = 6
d(B, C_2) = | 5 – 1 | + | 0 – 0 | + | 5 – 1 | + | 2 – 0 | = 10
d(C, C_1) = | 2 – 4 | + | 5 – 0 | + | 0 – 1 | + | 0 – 1 | = 9
d(C, C_2) = | 2 – 1 | + | 5 – 0 | + | 0 – 1 | + | 0 – 0 | = 7
d(D, C_1) = | 1 – 4 | + | 2 – 0 | + | 1 – 1 | + | 7 – 1 | = 11
d(D, C_2) = | 1 – 1 | + | 2 – 0 | + | 1 – 1 | + | 7 – 0 | = 9
d(E, C_1) = | 2 – 4 | + | 0 – 0 | + | 3 – 1 | + | 1 – 1 | = 4
d(E, C_2) = | 2 – 1 | + | 0 – 0 | + | 3 – 1 | + | 1 – 0 | = 4
d(F, C_1) = | 1 – 4 | + | 0 – 0 | + | 1 – 1 | + | 0 – 1 | = 4
d(F, C_2) = | 1 – 1 | + | 0 – 0 | + | 1 – 1 | + | 0 – 0 | = 0

We now assign each instance to the cluster with the smallest (Manhattan) distance to the cluster's centroid: for A, this is C1 because 0 < 4, for B, this is C1 because 6 < 10, and so on. We will see that C is closer to cluster 2, D to cluster 2, F to cluster 2, and for E we have a tie.

Let's say we randomly break the tie for instance E by assigning it to cluster 2. (We'll see what would have happened if we'd assigned E to cluster 1 below.) So, cluster 1 is {A,B} and cluster 2 is {C,D,E,F}. We re-calculate the centroids:

$$C1 = \left(\frac{4+5}{2}, \frac{0+0}{2}, \frac{1+5}{2}, \frac{1+2}{2}\right) = (4.5, 0, 3, 1.5)$$

$$C2 = \left(\frac{2+1+2+1}{4}, \frac{5+2+0+0}{4}, \frac{0+1+3+1}{4}, \frac{0+7+1+0}{4}\right) = (1.5, 1.75, 1.25, 2)$$

Now, let's re-calculate the distances according to these new centroids:

d(A, C_1) = | 4 – 4.5 | + | 0 – 0 | + | 1 – 3 | + | 1 – 1.5 | = 3
d(A, C_2) = | 4 – 1.5 | + | 0 – 1.75 | + | 1 – 1.25 | + | 1 – 2 | = 5.5
d(B, C_1) = | 5 – 4.5 | + | 0 – 0 | + | 5 – 3 | + | 2 – 1.5 | = 3
d(B, C_2) = | 5 – 1.5 | + | 0 – 1.75 | + | 5 – 1.25 | + | 2 – 2 | = 9
d(C, C_1) = | 2 – 4.5 | + | 5 – 0 | + | 0 – 3 | + | 0 – 1.5 | = 12
d(C, C_2) = | 2 – 1.5 | + | 5 – 1.75 | + | 0 – 1.25 | + | 0 – 2 | = 7
d(D, C_1) = | 1 – 4.5 | + | 2 – 0 | + | 1 – 3 | + | 7 – 1.5 | = 13
d(D, C_2) = | 1 – 1.5 | + | 2 – 1.75 | + | 1 – 1.25 | + | 7 – 2 | = 6
d(E, C_1) = | 2 – 4.5 | + | 0 – 0 | + | 3 – 3 | + | 1 – 1.5 | = 3
d(E, C_2) = | 2 – 1.5 | + | 0 – 1.75 | + | 3 – 1.25 | + | 1 – 2 | = 5
d(F, C_1) = | 1 – 4.5 | + | 0 – 0 | + | 1 – 3 | + | 0 – 1.5 | = 7
d(F, C_2) = | 1 – 1.5 | + | 0 – 1.75 | + | 1 – 1.25 | + | 0 – 2 | = 4.5

What are the assignments of instances to clusters now? Cluster 1 {A,B,E} and cluster 2 {C,D,F}. (Note that we're at the same place now that we would have been if we'd randomly broke the tie for instance E to cluster 1 earlier.)

We calculate the new centroids based on these instances:

$$C1 = \left(\frac{4+5+2}{3}, \frac{0+0+0}{3}, \frac{1+5+3}{3}, \frac{1+2+1}{3}\right) \approx (3.67,0,3,1.33)$$

$$C2 = \left(\frac{2+1+1}{3}, \frac{5+2+0}{3}, \frac{0+1+1}{3}, \frac{0+7+0}{3}\right) \approx (1.33,2.33,0.67,2.33)$$

We re-calculate the distances according to these new centroids:

$d(A, C_1) \approx |4-3.67| + |0-0| + |1-3| + |1-1.33| \approx 2.67$

$d(A, C_2) \approx |4-1.33| + |0-2.33| + |1-0.67| + |1-2.33| \approx 6.67$

$d(B, C_1) \approx |5-3.67| + |0-0| + |5-3| + |2-1.33| \approx 4$

$d(B, C_2) \approx |5-1.33| + |0-2.33| + |5-0.67| + |2-2.33| \approx 10.67$

$d(C, C_1) \approx |2-3.67| + |5-0| + |0-3| + |0-1.33| \approx 11$

$d(C, C_2) \approx |2-1.33| + |5-2.33| + |0-0.67| + |0-2.33| \approx 6.33$

$d(D, C_1) \approx |1-3.67| + |2-0| + |1-3| + |7-1.33| \approx 12.33$

$d(D, C_2) \approx |1-1.33| + |2-2.33| + |1-0.67| + |7-2.33| \approx 5.67$

$d(E, C_1) \approx |2-3.67| + |0-0| + |3-3| + |1-1.33| \approx 2$

$d(E, C_2) \approx |2-1.33| + |0-2.33| + |3-0.67| + |1-2.33| \approx 6.67$

$d(F, C_1) \approx |1-3.67| + |0-0| + |1-3| + |0-1.33| \approx 6$

$d(F, C_2) \approx |1-1.33| + |0-2.33| + |1-0.67| + |0-2.33| \approx 5.33$

The new assignments of instances to clusters are cluster 1 {A,B,E} and cluster 2 {C,D,F}. This is the same as the last iteration, so we stop (and this is the final assignment of instances to clusters).

(ii). Perform agglomerative clustering of the above dataset (excluding the *id* and *label* attributes), using the Euclidean distance and calculating the group average as the cluster centroid.

In the lectures you have been introduced to single link, complete link and group average methods to compute the distance of two clusters. Here we are using the cluster centroid. In this method we simply compute the average of the instances in a cluster as the new cluster centroid and compute the distance of two clusters based on the Euclidean distance of the cluster centroids.

We begin by finding the pairwise similarities — or distances, in this case, between each instance. I'm going to skip the Euclidian distance calculations (you can work through them as an exercise) and go straight to the proximity matrix:

	A	**B**	**C**	**D**	**E**	**F**
A	-	$\sqrt{18}$	$\sqrt{31}$	$\sqrt{49}$	$\sqrt{8}$	$\sqrt{10}$
B	$\sqrt{18}$	-	$\sqrt{63}$	$\sqrt{61}$	$\sqrt{14}$	$\sqrt{36}$
C	$\sqrt{31}$	$\sqrt{63}$	-	$\sqrt{60}$	$\sqrt{35}$	$\sqrt{27}$
D	$\sqrt{49}$	$\sqrt{61}$	$\sqrt{60}$	-	$\sqrt{45}$	$\sqrt{53}$
E	$\sqrt{8}$	$\sqrt{14}$	$\sqrt{35}$	$\sqrt{45}$	-	$\sqrt{6}$
F	$\sqrt{10}$	$\sqrt{36}$	$\sqrt{27}$	$\sqrt{53}$	$\sqrt{6}$	-

We can immediately observe (without simplifying the square roots) that the most similar instances (with the smallest distance) are E and F.

We will then form a new cluster EF, for which we calculate the centroid: (1.5, 0,2,0.5), and then we must calculate the distances to this new cluster[1].

	A	B	C	D	EF
A	-	$\sqrt{18}$	$\sqrt{31}$	$\sqrt{49}$	$\sqrt{\mathbf{7.5}}$
B	$\sqrt{18}$	-	$\sqrt{63}$	$\sqrt{61}$	$\sqrt{\mathbf{23.5}}$
C	$\sqrt{31}$	$\sqrt{63}$	-	$\sqrt{60}$	$\sqrt{\mathbf{29.5}}$
D	$\sqrt{49}$	$\sqrt{61}$	$\sqrt{60}$	-	$\sqrt{\mathbf{47.5}}$
EF	$\sqrt{\mathbf{7.5}}$	$\sqrt{\mathbf{23.5}}$	$\sqrt{\mathbf{29.5}}$	$\sqrt{\mathbf{47.5}}$	-

The closest distance now is A with the new cluster EF; the resulting cluster AEF has the centroid $(\frac{7}{3}, 0, \frac{5}{3}, \frac{2}{3})$.

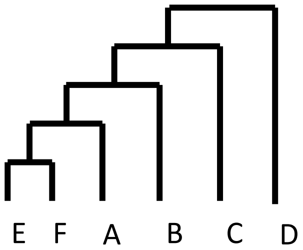
	AEF	B	C	D
AEF	-	$\sqrt{\mathbf{20}}$	$\sqrt{\mathbf{28.3}}$	$\sqrt{\mathbf{46.3}}$
B	$\sqrt{\mathbf{20}}$	-	$\sqrt{63}$	$\sqrt{61}$
C	$\sqrt{\mathbf{28.3}}$	$\sqrt{63}$	-	$\sqrt{60}$
D	$\sqrt{\mathbf{46.3}}$	$\sqrt{61}$	$\sqrt{60}$	-

Now B gets clustered with AEF; ABEF has the centroid (3, 0, 2.5, 1).

	ABEF	C	D
ABEF	-	$\sqrt{\mathbf{33.25}}$	$\sqrt{\mathbf{46.25}}$
C	$\sqrt{\mathbf{33.25}}$	-	$\sqrt{60}$
D	$\sqrt{\mathbf{46.25}}$	$\sqrt{60}$	-

All that is left now is to assign C to ABEF; there is no need to calculate the centroid anymore, as there are only two clusters (ABCEF and D) remaining.

Hence, we have here the agglomerate clustering E-F, A, B, C, D.



3. Revise the concept of *unsupervised* and *supervised* **evaluation** for clustering evaluation.

   (i). Explain the two main concepts that we use to measures the goodness of a clustering structure without respect to external information.

   The two main concepts that we check in unsupervised clustering evaluation are the clusters *cohesiveness* and *separability*. We want the members of each cluster to be as integrated and close to each other as possible and meanwhile we want the clusters to be as separate and independent as possible from other clusters. Using the SSE metric, we can use W-SSE (Within Cluster Sum of Squared Errors) to measure intra-cluster cohesion and B-SSE (Between Cluster Sum of Squared Errors) to measure inter-cluster separation.

   (ii). Explain the two main concepts that we use to measure the how well do cluster labels match externally supplied class labels.

   The two main factors in supervised clustering evaluation metrics are *homogeneity* and *completeness*. Homogeneity measures if all the elements of each cluster have the same true

[1] There are other ways of performing this step, for example, **single link**: using the shortest distance out of the ones calculated above to the points in this cluster, so that the distance from A to EF is min $(\sqrt{8}, \sqrt{10}) = \sqrt{8}$

label. We can use measure the homogeneity of a cluster using Entropy and Purity. Completeness metric measure if all the members of a class are assigned to the same cluster.

In other words, Homogeneity only considers if all the members of a cluster have the same label. But if there are too many clusters with the same label the homogeneity will not be able to detect it. For example, in a dataset with ten zeros and ten ones, a proper clustering would be forming two clusters each with 10 members. But if our system makes 20 clusters (one per instance), the homogeneity index (purity or entropy) will still show a perfect result, although the clustering is not very useful. So, we need another metric that indicate if all the members of a given label are assigned to the same cluster. This metric is completeness.

4. [OPTIONAL] Using the dataset introduced in Question 2, consider the clusters, C1: {A, B, E} and C2: {C, D, F} and compare them with clusters C1': {A, B, E, F} and C2': {C, D}

(i). Using the cohesion and separation of the clusters.

To calculate the cohesion and separation for clusters {A, B, E} and {C, D, F}, we need to first determine the centroids for each cluster. We can do this by calculating the mean values for each attribute for each cluster:

Cluster {A, B, E}: Centroid: [3.67, 0, 3, 1.33]

Cluster {C, D, F}: Centroid: [1.33, 2.33, 0.67, 2.33]

Using these centroids, we can then calculate the cohesion and separation for the two clusters:

Cohesion:

Cohesion measures how closely related the data points within a cluster are to each other. We can calculate the cohesion of a cluster by inversing the sum distances between each data point in the cluster and the centroid. Using squared Euclidian to calculate the distance we will have.

For cluster C1 = {A, B, E}:

$$\sum_{x \in C1} Distance(m_i, x) = ((3.67-4)^2 + (0-0)^2 + (3-1)^2 + (1.33-1)^2 \\ + (3.67-5)^2 + (0-0)^2 + (3-5)^2 + (1.33-2)^2 \\ + (3.67-2)^2 + (0-0)^2 + (3-3)^2 + (1.33-1)^2) \\ = 13.33$$

$$C1_{Cohesion} == \frac{1}{13.33} = 0.075$$

For cluster C2 = {C, D, F}:

$$\sum_{x \in C2} Distance(m_i, x) \\ = ((1.33-2)^2 + (2.33-5)^2 + (0.67-0)^2 + (2.33-0)^2 \\ + (1.33-1)^2 + (2.33-2)^2 + (0.67-1)^2 + (2.33-7)^2 \\ + (1.33-1)^2 + (2.33-0)^2 + (0.67-1)^2 + (2.33-0)^2 \\ = 46.64$$

$$C2_{Cohesion} = \frac{1}{46.64} = 0.021$$

The final Cohesion of this clustering is $C1_{Cohesion} + C2_{Cohesion} = 0.075 + 0.021 = 0.096$

Separation:

Separation measures how well separated the clusters are from each other. We can calculate the separation between two clusters by taking the squared Euclidean distance between their centroids.

$$Separation = (1.33 - 3.67)^2 + (2.33 - 0)^2 + (0.67 - 3)^2 + (2.33 - 1.33)^2 = 17.33$$

Note that higher cohesion values indicate that the data points within a cluster are more closely related to each other, and higher separation values indicate that the clusters are more well separated from each other.

Similarly, we can calculate the Cohesion and Separation for clusters {A, B, E, F} and {C, D}. Using the same calculation as above the final results will be as follows.

Cohesion = 0.043 + 0.033 = 0.076

Separation = 24.75

We can infer that C1: {A, B, E} and C2: {C, D, F} have higher cohesion in compared with C1': {A, B, E, F} and C2': {C, D}. It means that C1 and C2 are denser than C1' and C2'.

On the other hand, separation between the clusters C1: {A, B, E} and C2: {C, D, F} is smaller compared to the separation between C1': {A, B, E, F} and C2': {C, D}. The higher separation between C1' and C2' suggests that these two clusters are more distinct and dissimilar to each other.

(ii). Using the purity of the clusters.

To calculate cluster purity and entropy, we first need to calculate the class distribution for each cluster.

For cluster C1:{A, B, E}, the class distribution is:

fruit: 2 (A, B)
unknown: 1 (E)

For cluster C2:{C, D, F}, the class distribution is:

comp: 2 (C, D)
unknown: 1 (F)

To calculate purity, we take the maximum class frequency in each cluster and sum them up, and then divide by the total number of instances in the clusters.

For cluster {A, B, E}:

max class frequency: 2 (fruit)
purity: $\frac{2}{3} = 0.67$

For cluster {C, D, F}:

max class frequency: 2 (comp)
purity: $\frac{2}{3} = 0.67$

The sum of purity for clusters C1 and C2 is 0.67 + 0.67 = 1.34

Similarly, we can calculate the Cohesion and Separation for clusters C1': {A, B, E, F} and C2': {C, D}. Using the same calculation as above the final results will be:

For cluster {A, B, E, F}:

purity: $\frac{2}{4} = 0.5$

For cluster {C, D}:

purity: $\frac{2}{2} = 1$

The sum of purity for clusters C1' and C2' is 0.5 + 0.1 = 1.5.

This suggests that the clustering result for C1' and C2' is better in terms of purity than the clustering result for C1 and C2.

However, it's important to note that purity alone may not provide a complete evaluation of clustering performance, as it doesn't consider if all the members of a given label are assigned to the same cluster. We also need a metric to calculate the *completeness* of our clusters. But since the lectures did not cover an implicit metric for measuring the completeness of the clusters this exercise is not covering that.