.Example of clustering method: single-link clustering

*Jean-Charles LAMIREL 2017*

Considering a set of cooking receipts (i.e. data) containing (i.e. indexed by features) described as:

D1: Potato – Leek

D2: Carrot – Tomato

D3: Potato – Carrot – Tomato

D4: Leek – Potato

D5: Potato – Tomato

D6: Tomato – Grapefruit

D7: Orange – Lemon – Grapefruit

D8: Orange – Lemon

D9: Orange – Grapefruit

D10: Lemon – Grapefruit

**1. Step of computation of association strength (correlation) between features**

Computation of the raw co-occurrence between features (number of associations between the different features in the data) results in:

|  |  |
| --- | --- |
| Detected associations | Frequency  (raw co-occurrence) |
| Potato - Leek | 2 |
| Carrot - Tomato | 2 |
| Potato - Tomato | 2 |
| Potato - Carrot | 1 |
| Tomato - Grapefruit | 1 |
| Grapefruit - Lemon | 2 |
| Grapefruit - Orange | 2 |
| Orange - Lemon | 2 |

**Problem with raw co-occurrence**:

It is not directly representative of the strength of association between features because it depends directly on the raw frequency of features. Two very common features may have a strong number of co-occurrences without yet be systematically associated in the data (low effective association). Conversely, two quite infrequent features may have a low number of co-occurrences whilst being systematically involved in the data (strong effective association). In these two cases the strength of association between features is poorly illustrated by raw co-occurrence. To overcome this problem, we use so the coefficient of equivalence described herein.

**Computation of the strength of association between features using equivalence coefficient**:

Calculation of the strength of association between features using the coefficient of equivalence is expressed as follows:



where represents the raw co-occurrence between *i* and *j* features*,* and represents the raw frequency of feature *i*, the raw frequency of feature *j*.

The new computation of association strength between features follows:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Detected association | Raw co-occurrence (reminder) | Frequency  1st feature | Frequency  2nd feature | Equivalence  coefficient |
| Potato - Leek | 2 | 4 | 2 | 1/2 |
| Carrot - Tomato | 2 | 2 | 4 | 1/2 |
| Potato - Tomato | 2 | 4 | 4 | 1/4 |
| Potato - Carrot | 1 | 4 | 2 | 1/8 |
| Tomato - Grapefruit | 1 | 4 | 4 | 1/16 |
| Grapefruit - Lemon | 2 | 4 | 3 | 1/3 |
| Grapefruit - Orange | 2 | 4 | 3 | 1/3 |
| Orange - Lemon | 2 | 3 | 3 | 4/9 |

Remark: According to the coefficient of equivalence, we note that association Potato - Tomato is less relevant than the number of raw co-occurrences because keywords are very frequent individually (frequency 4 for each of them).

**Ranking of associations by decreasing value of equivalence coefficient (association file):**

|  |  |  |  |
| --- | --- | --- | --- |
| Reference | Ranking | Detected associations | Equivalence  coefficient |
| A1 | 1 | Potato - Leek | 1/2 |
| A2 | 1 | Carrot - Tomato | 1/2 |
| A3 | 3 | Orange - Lemon | 4/9 |
| A4 | 4 | Grapefruit - Lemon | 1/3 |
| A5 | 4 | Grapefruit - Orange | 1/3 |
| A6 | 6 | Potato - Tomato | 1/4 |
| A7 | 7 | Potato - Carrot | 1/8 |
| A8 | 8 | Tomato - Grapefruit | 1/16 |

**2. Clustering step (single link clustering)**

Initially each keyword forms a full cluster.

The scan of the association file (see above) is then operated by decreasing value of the equivalence coefficient. For each encountered association between features the clusters to which the features belong of the association are merged, if the following two conditions are true:

* The coefficient of equivalence representing the association between feature is strong enough,
* The size of the resulting cluster (after fusion) does not exceed a certain limit.

Where the merging does not happen, the association is maintained as an external association marking the relationship between the respective affiliation clusters.

Remark: The association **A8**:Tomato-Grapefruit verifies none of the conditions to merger set forth above. The coefficient of equivalence of the association is low, what makes that association can be considered a marginal association. In addition, taking account of this association for the merging of clusters would create a global cluster containing all of the features of the dataset, which would lose any meaning to the result of the clustering (see the Powerpoint animation dynamically describing the clustering process).

**3. Clustering quality evaluation**

**How to evaluate internal cluster coherency?**

In our case, the coherence of a cluster can be comparable to its transitive closure rate, measuring the saturation of the associations between the elements of the cluster. This rate is expressed in the following manner for a given cluster c:



In the illustrative example:

*CCC1*= 2/3 et *CCC2*= 1.

Remark: For a class consisting of n features, the number of potential associations systematically takes the value:



On average for all clusters (i.e. for the global model C), we get:

**How to figure out inter-cluster differences?**

In our case, we use the number of external links between clusters for highlight the difference. The more external links there are in a given model, the higher is the difference.

**How to choose the suitable number of clusters?**

If we consider single link clustering as a hierarchical process (see the Powerpoint animation dynamically describing the clustering process), we can draw a table with the value of the 2 indicators (*ECC* and *ICC*) at each step of the hierarchical process and use the trends to decide at which point to stop the hierarchical process.

The table of the value of the two indicators by level follows:

|  |  |  |
| --- | --- | --- |
| *Level* | *ICC* | *ECC* |
| 0 | 1 | 0.11 |
| 1 | 1 | 0.12 |
| 2 | 1 | 0.14 |
| 3 | 1 | 0.17 |
| 4 (step 5) | 1 | 0.25 |
| 5 (step 6) | 0.83 | 0.5 |
| 6 (step 7) | 0.38 | 1 |

The related trends are represented hereafter:

Elbow

Break-even point

Two strategies can be used to figure out what could be the best level for stopping the hierarchical process:

* The elbow method consists in finding important changes in the quality curves and stopping at the level to which these important changes are occurring.
* The break-even point method consists in finding the compromise or intersection point between the two quality curved (coherency curve and cluster difference curves) that evolve in a converse way and stopping at (or just before) level to which this breakeven point occurs.

Here the best level in both cases is the level 5.

**3. General remarks**

The single-link clustering method is very sensitive to the polysemy as the agglomeration of features into clusters does obey simple chaining rules. Polysemy (i.e. a same feature used in two associations according to two different meanings in each of them) can produce adverse effect of transitive cluster agglomerations.

The single-link method must be considered an incomplete classification method, even partially incoherent. Indeed, association strength between some features cannot be calculated if there is no observable co-occurrence between them. The absence of observable co-occurrence between two feature can lead to the assumption their strength of association is null. However, it is noted that this absence may be simply related to a relationship of negative correlation between these features, each of the feature alternately appearing in data. Negatively correlated features may have a null strength of association, this situation returns then in total contradiction with the previous hypothesis.

**4. Association rules extraction**

Association rules can be used to detect dependencies and correlation between features. This approach can be considered as complementary to the clustering process. It is usually associated to symbolic learning but it can also be exploited in the clustering context.

Rules can be extracted using the co-occurrence matrix (see the Powerpoint animation highlighting this process clustering process).

Two kind of rules can be characterized in such a way.

1) Implication rules illustrate the dependency of a feature relatively to another:

If then there is the implication rule

On the considered data, feature *i* occurs whenever feature *j* occurs, but feature *j* can occur sometimes without feature *i.*

If then there is the implication rule

On the considered data, feature *j* occurs whenever feature *i* occurs, but feature *i* can occur sometimes without feature *i.*

2) Equivalence rules highlight the correlation between 2 features:

If then there is the equivalence rule

On the considered data, feature *i* and *j* always occurs together.

Note that association rules are not logical rules even if they behave in a similar way. Hence, such kind of rules depend on the available data. Such rules can thus be verified on a given subset of data, but they cannot be still verified if the whole set of data is considered.

Plausibility of rules (implication and equivalence) mainly depends on their support and their confidence:

The support of the rule is computed as where : where is the total number of data considered.

The support of a rule is computed by exploiting the feature with the smallest frequency in a rule. It could be considered as an estimation of the probability of this feature (*P(i)*).

The confidence of the rule is computed as . It could be thus reduced after simplification to: . It could be considered as an estimation of the conditional probability : |*i*).

The confidence is 1 is the rule is true for all the considered data.

The support and the confidence of the rules highlighted in your example follows:

|  |  |  |
| --- | --- | --- |
| Rule | Support | Confidence |
| Potato Leek | 0.2 | 1 |
| Carrot Tomato | 0.2 | 1 |
| Orange Lemon | 0.3 | 1 |

**5. Other clustering methods**

Single-link clustering is a specific and simple method of clustering directly based on features. Usual clustering methods rather consider documents as vectors. The dimension of the exploited vector space is usually equal to the number of features. One can thus speak about feature space.

For transforming former document description into vectors a single process can be applied:

1. Each document become a vector whose dimension is the one of the feature space,
2. (all documents are vectors of equal dimension: the number of features),
3. Each feature is associated to a specific component of the vector,
4. If a feature is present in a document, its associated component in the feature vector get the unity value (1), other it takes the 0 value.

In our example, there are 7 different features and the size of the associated feature space is thus 7.

Each feature will be associated to a specific component of the vector:

F1: Potato Component 1

F2 : Leek Component 2

F3: Tomato Component 3

F4 : Carrot Component 4

F5 : Orange Component 5

F6 : Lemon Component 6

F7 : Grapefruit Component 7

In such a way the document D1 can be represented as the vector V1:

D1: Potato – Leek

V1: 1 2 3 4 5 6 7

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 |

The whole dataset can be therefore represented as a **Documents X Features** matrix:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | F1 | F2 | F3 | F4 | F5 | F6 | F7 |
| D1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| D2 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| D3 | 1 | 0 | 1 | 1 | 0 | 0 | 0 |
| D4 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| D5 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| D6 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| D7 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| D8 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| D9 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| D10 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

Such kind of clustering is usually exploiting Euclidean distance or Cosine distance to compare documents vectors during the clustering process. Note that Coefficient of Equivalence is a Cosine similarity in the document space which is the Dual space to the feature space. This document space representation is obtained by transposing the **Documents X Features** matrix that become thus a **Feature X Documents** matrix.

This model is compliant with frequency of features inside documents. In the case frequencies are considered, the value of the vector component associated to a feature in a document vector takes the value of the frequency of the feature inside the document. Feature weights can also be used as well instead of frequencies.