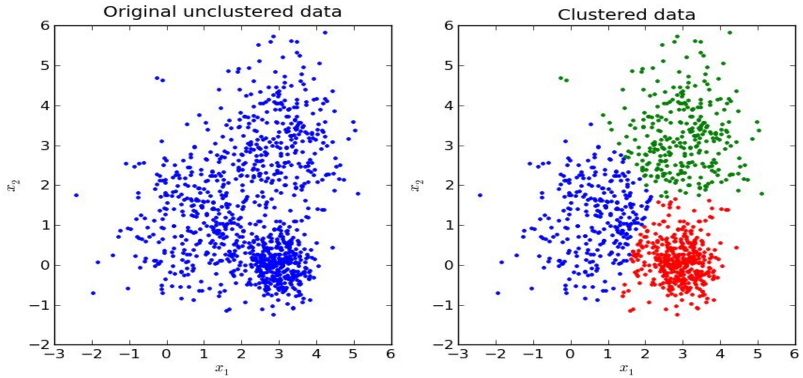
A Clustering Approach to detect Dependencies between Test Cases

Cecilia Battinelli, Magnus Orrsveden, Han Swee Yang

Table of Contents

Introduction



Assignment Description

Ground Truth

TC	DependsOn
TC0000	TC0001
TC0000	TC0002
TC0001	TC0002
TC0018	TC0019
TC0018	TC0020
⋮	⋮

Data as 64-dimensional vectors

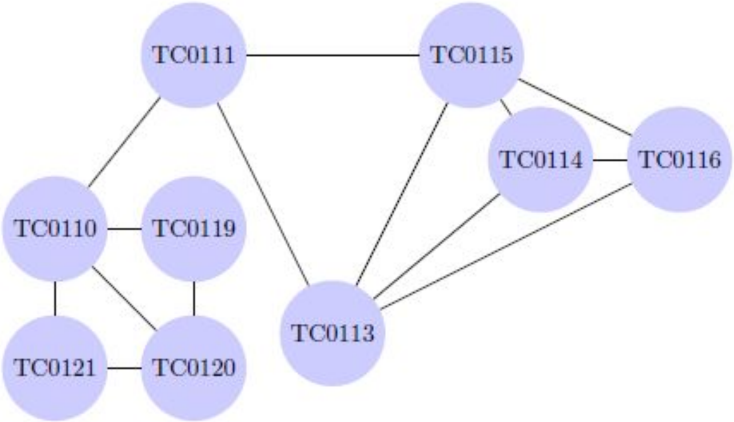
TC	d_0	d_1	$\cdots$	d_{63}
TC0000	1.3540313243865967	-0.3458509147167206	$\cdots$	-0.526816725730896
TC0001	1.1611902713775635	-0.2490767240524292	$\cdots$	-0.6400560736656189
TC0002	1.0486541986465454	-0.11896729469299316	$\cdots$	-0.48657703399658203
⋮	⋮	⋮		⋮

Aim : Find the best algorithm to cluster the high dimensional feature vectors

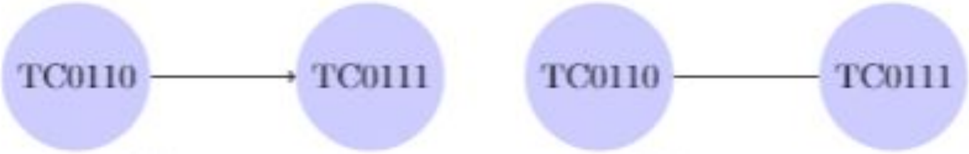
How: Evaluate performance of different clustering methods through evaluation metrics

Assumptions

- Not-overlapping clustering



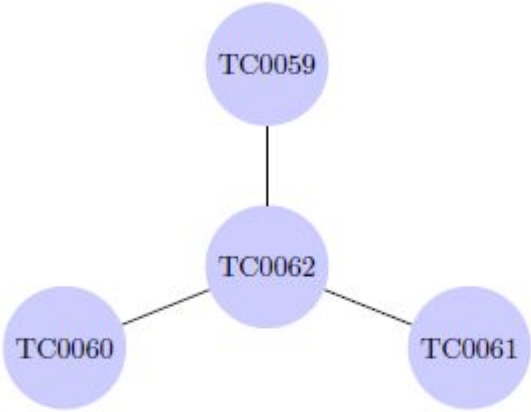
- Bidirectional dependencies

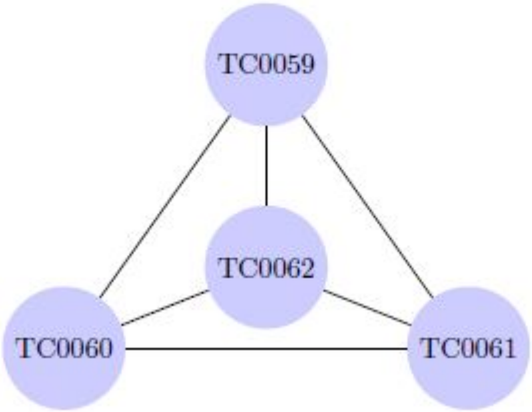


((a)) Directional ((b)) Bidirectional

Assumptions

- Transitive Dependencies within GT Clusters





- Each TC must be assigned to one and only one cluster - specific for K-Means

Method

1. Import High Dimensional Data
2. Format GT to get clusters accordingly to Assumptions
3. Compute Indep List
4. If wanted, perform PCA to maintain 80% variability
5. Run Clustering Algorithm
6. Compute Result Matrix
7. Evaluate results

Performance Analysis

Algorithm 5 Detection of TP and FN pairwise instances

```
Data: Results matrix and formatted Ground Truth GT.csv
Result: TP and FN
initialization to 0 for TP and FN
for every TC do
    for every line in GT.csv do
        if TC is in line then
            for every element in line do
                if element != TC and element > TC then
                    if they have same label in Results different from -1 then
                        TP += 1
                    else
                        FN += 1
                    end
                end
            end
        end
    end
end
```

Performance Analysis (Cont'd)

Algorithm 6 Detection of FP pairwise instances

Data: Results matrix & formatted Ground Truth `GT.csv` & `Indep` list
Result: FP

```
initialisation to 0 for FP
for every pair of TCs i,j do
    if j > i then
        if i,j have same label in Result & this label is not -1 then
            for each line in GT.csv do
                if i is in the line but j not then
                    FP += 1
                end
            end
        end
    end
end
for every pair of TCs i,j do
    if j > i then
        if i,j have same label in Result & this label is not -1 then
            if i & j are both in Indep then
                FP += 1
            end
        end
    end
end
```

Performance Analysis (Cont'd)

Pairwise labeling

- Not ordered combinations w/o repetition
- ">" condition to avoid double counting
- TP + FN fixed sum

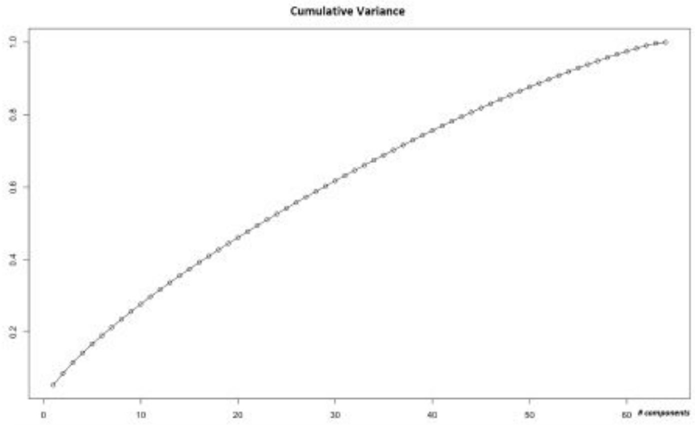
$$TP + FN = \sum_{i=1}^{NR} \binom{n_i}{2}$$

- TN = N - (TP + FN + FP)

$$N = \binom{1748}{2} = 1526878$$

-1 labeling → valid only for HDBSCAN

PCA Procedure



- Linear increasing rate
- 80% variability → 44 components.

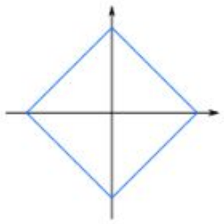
Parameter Selection

K-Means

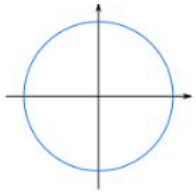
- K = {400,600,800,1000,1200}

HDBSCAN

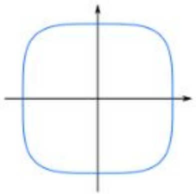
- **Distance:** Canberra, Manhattan, Euclidean & Minkowski (p=4)
- **Alpha:** distance tuning parameter → a = {0.75,1.0,1.2}
- **Min Cluster Size:** MCS = {2,3}
- **Cluster Selection Method:** granularity of clusters → {eom,leaf}



Manhattan (p=1)



Euclidean (p=2)



Minkowski (p=4)

Results – Tables

K-Means

- Without PCA

K	Recall	Precision	F-measure
400	0.093	0.036	0.052
600	0.156	0.313	0.216
800	0.176	0.009	0.016
1000	0.165	0.007	0.013
1200	0.189	0.008	0.015

- With PCA

K	Recall	Precision	F-measure
400	0.086	0.187	0.118
600	0.155	0.310	0.207
800	0.171	0.008	0.016
1000	0.189	0.008	0.015
1200	0.170	0.474	0.250

HDBSCAN

- Without PCA

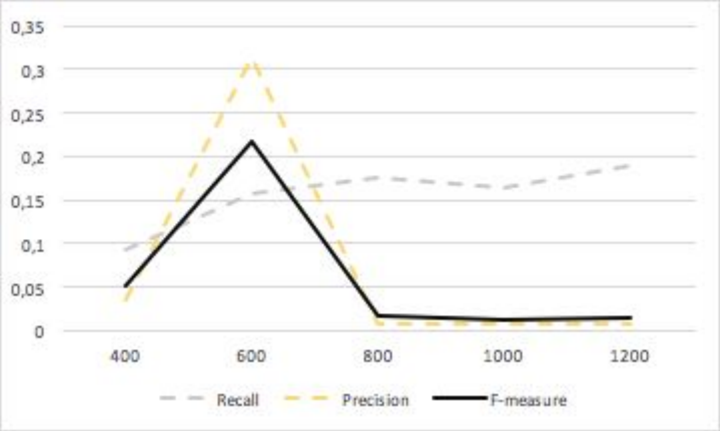
Distance	Alpha	Min Cluster Size	Cluster Selection Method	Recall	Precision	F-measure
Canberra	0,75 & 1,0	2	eom	0,261	0,269	0,265
Canberra	0,75 & 1,0	2	leaf	0,223	0,301	0,256
Canberra	0,75	3	leaf	0,280	0,252	0,265
Euclidean	0,75 & 1,0	2	leaf	0,233	0,449	0,307
Euclidean	0,75	3	eom	0,305	0,315	0,310
Euclidean	1,2	3	eom	0,957	0,002	0,005
Euclidean	0,75	3	leaf	0,290	0,391	0,333
Manhattan	0,75 & 1,0	2	eom	0,243	0,323	0,277
Manhattan	1,2	2	eom	0,217	0,444	0,292
Manhattan	1,2	2	leaf	0,201	0,627	0,304
Manhattan	0,75	3	eom	0,296	0,325	0,309
Manhattan	1,0	3	eom	0,256	0,394	0,310
Manhattan	0,75	3	leaf	0,270	0,405	0,324
Minkowski	0,75 & 1,0	2	leaf	0,233	0,449	0,307
Minkowski	0,75	3	eom	0,305	0,315	0,310
Minkowski	1,2	3	eom	0,957	0,002	0,005
Minkowski	0,75	3	leaf	0,290	0,391	0,333

- With PCA

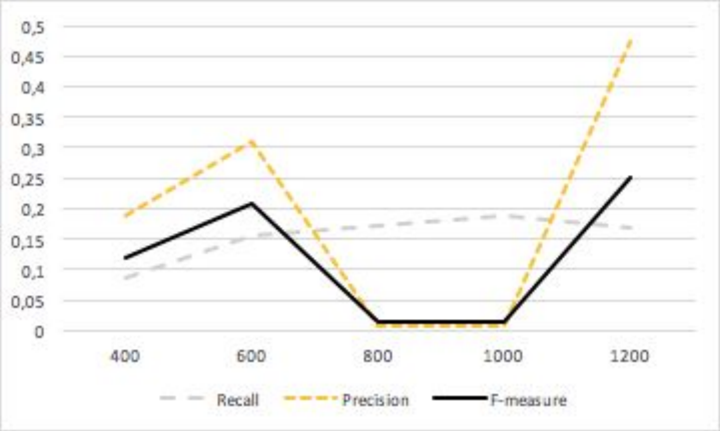
Distance	Alpha	Min Cluster Size	Cluster Selection Method	Recall	Precision	F-measure
Canberra	0.75 & 1.0 & 1.2	2	eom	0.254	0.290	0.271
Canberra	0.75 & 1.0 & 1.2	2	leaf	0.227	0.278	0.250
Euclidean	0.75 & 1.0 & 1.2	2	eom	0.244	0.377	0.296
Euclidean	0.75 & 1.0 & 1.2	3	eom	0.997	0.002	0.003
Manhattan	0.75 & 1.0 & 1.2	2	eom	0.238	0.431	0.306
Manhattan	0.75 & 1.0 & 1.2	3	eom	0.999	0.002	0.003
Manhattan	0.75 & 1.0 & 1.2	2	leaf	0.184	0.391	0.250
Minkowski	0.75 & 1.0 & 1.2	2	eom	0.244	0.377	0.296
Minkowski	0.75 & 1.0 & 1.2	3	eom	0.997	0.002	0.003

Results - K-means

K-means without PCA

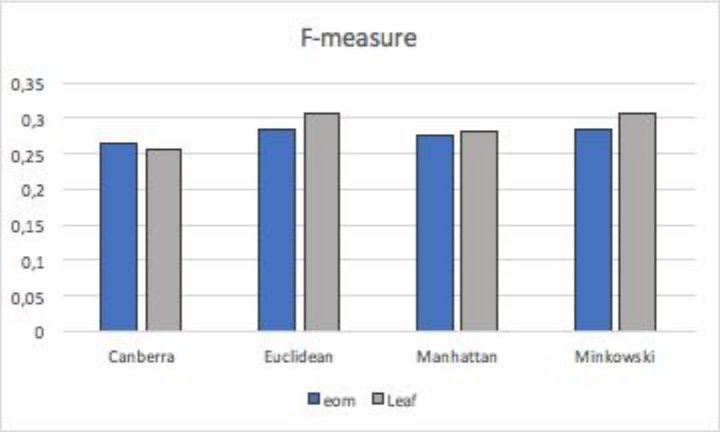


K-means with PCA

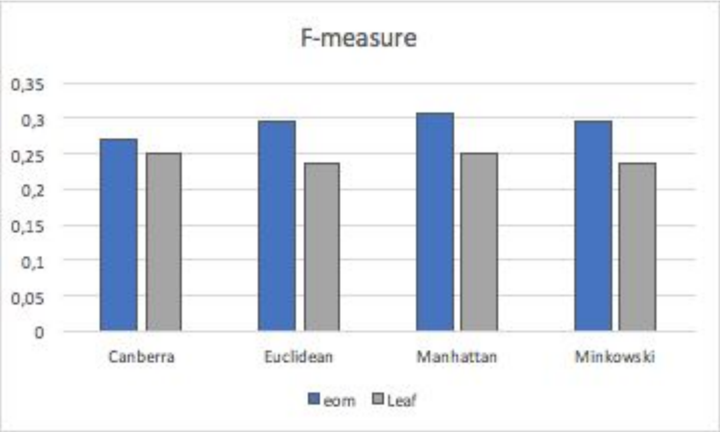


Results – HDBSCAN

HDBSCAN (alpha = 1, MinClusterSize = 2)
Without PCA



HDBSCAN (alpha =1 MinClusterSize = 2)
With PCA



Analysis - K-Means

- Generally weaker results

- Advantages
  - Relatively fast iteration

- Disadvantages
  - To be run over more iterations
    - Offsets speed advantage over HDBSCAN
  - Value of Clusters must be known beforehand
  - Only ideal for capturing ‘globular’/spherical clusters
  - Poorer performance in capturing clusters of varying variance
  - Does not separate outliers/noise/independent data

Analysis – HDBSCAN

- Generally More Reliable
  - More consistent, better results
  - Separate outliers/noise/independent data
  - More parameters can be changed
  - Does not need multiple instances
    - Global Optimum found
    - Reproducible by external parties
- Considerations
  - Still does not capture large clusters in its entirety

Analysis – is PCA good?

- **Different parameters → different behaviour**
- K-Means:
  - K=600 worsening using PCA
  - K=1200 improvement using PCA

- HDBSCAN:
  - Canberra [2,eom] → improvement using PCA
  - Canberra [2,leaf] → worsening using PCA
  - Manhattan[2,eom] → best results in PCA
  - More degenerate cases

- Best K-Means F-Measure if using PCA
- Best HDBSCAN F-Measure if *not* using PCA

It depends on:

- Parameters
- Requirements

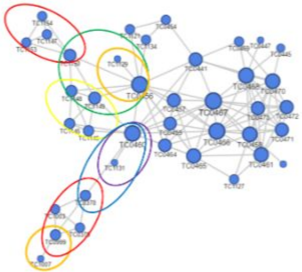
Discussion & Improvements

- Limitations of Current Algorithms
  - Non-overlapping Clusters
    - Highly Restrictive in Clustering
  - Direct/Directional Dependencies not Shown
    - Further Processing Required

- Limitations of Performance Metric
  - Precision & Recall equal weightage
    - May be different in reality
  - Effect of True Negatives is ignored
    - Only TP, FP, and FN taken into consideration
  - Size of Positive/Negative allocations in clustering not considered
    - Size of dependent/independent results may skew performance metrics

Discussion & Improvements (Cont'd)

- Use of other algorithms
  - Fuzzy C-Means
    - "Upgrade" from K-Means
    - Degree of membership into clusters
    - More flexible than K-Means/HDBSCAN



  - Subspace Clustering Methods
    - Forms clusters in data's subspaces
    - 'Noise' from irrelevant dimensions removed/ignored
    - SUBCLU, CLIQUE, DOC, MAFIA

- Use of other metrics/indexes
  - F_β Scores - Different weightage of Precision/Recall

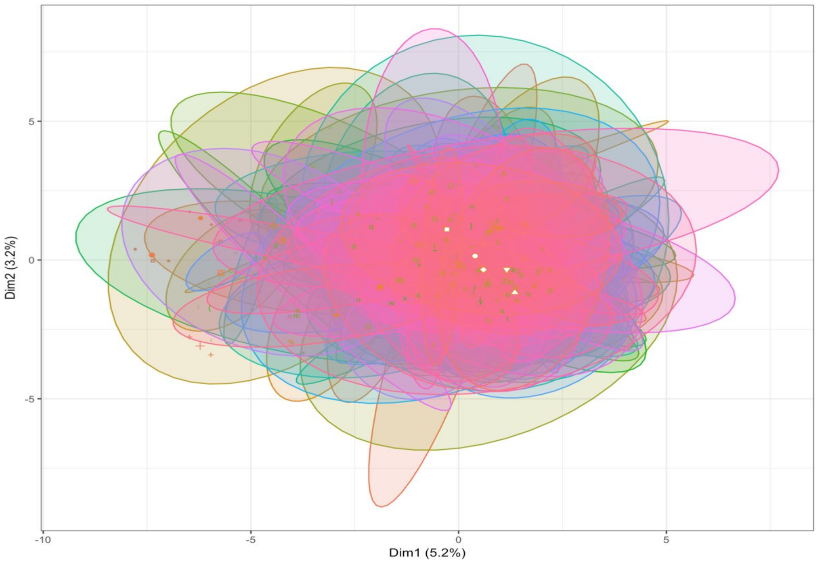
Distance	PCA	Alpha	MCS/K	CSM	F_1 Score	$F_{0,5}$ Score	F_2 Score
K-Means	Yes	N.A.	1200	N.A.	0,250	0,349	0,195
Minkowski	No	0.75	3	leaf	0.333	0,366	0,306
Manhattan	No	1,2	2	leaf	0,304	0,440	0,233
Manhattan	Yes	1,0	2	eom	0,306	0,371	0,261
Minkowski	No	0,75	3	eom	0,310	0,313	0,306
Canberra	Yes	1,0	2	eom	0,271	0,282	0,260

  - Matthew's Correlation Coefficient (MCC) - Prioritises True Positives and True Negatives equally

Distance	PCA	Alpha	MCS/K	CSM	TP	FP	FN	TN	F_1 Score	MCC Score
K-Means	Yes	N.A.	1200	N.A.	264	293	1.289	1.525.560	0,250	0,283
Minkowski	No	0.75	3	leaf	451	703	1.102	1.524.622	0.333	0,484
Manhattan	No	1,2	2	leaf	312	186	1.241	1.525.139	0,304	0,335
Manhattan	Yes	1,0	2	eom	369	488	1.184	1.524.837	0,306	0,396
Minkowski	No	0,75	3	eom	473	1.027	1.080	1.524.298	0,310	0,507
Canberra	Yes	1,0	2	eom	394	963	1.159	1.524.362	0,271	0,422

Q&A and some very intuitive plots

K-Means



HDBSCAN