UNIVERSIDAD DE MÁLAGA

Aprendizaje Computacional

Lab 1:
Clustering

AUTHORS:

janeczko tomasz

lukáš hofman

December 11, 2024

1 Introduction to problem

1.1 Yeast

Yeast cells have the ability to undergo sporulation, transitioning into a specialized cell type known as a spore. This biological process is pivotal for understanding gene regulation mechanisms. By examining gene activation and deactivation patterns during sporulation, insights into gene functions can be obtained.
In this study, K-means clustering is used to examine gene expression patterns during yeast sporulation. The findings are then compared to previous researches from A1 and A2 documents to evaluate the accuracy of the clustering and its biological relevance.

1.2 Possible solutions

- **K-means Clustering**: This is a partition-based algorithm, which is probably the most popular one. K-means aims at minimizing within-cluster variance by iterative assignments of data points to centroids. It is efficient for large datasets, but also requires pre-specification of the number of clusters.
- **Hierarchical Clustering**: It produces nested clusters either by successively combining (agglomerative) or dividing the data (divisive).
- **Self-Organizing Maps (SOMs)**: That is a neural network-based unsupervised learning approach representing multi-dimensional data within a two-dimensional grid.

1.3 Dataset

The raw dataset consists of 474 genes with expression values recorded over time intervals $t_0, t_{0.5}, t_2, t_5, t_7, t_9, t_{11.5}$. This dataset found in the provided file `sporulation-filtered.txt` captures dynamic gene activity during sporulation.

Genes	t0	t0,5	t2	t5	t7	t9	t11,5
YAL025C	1,163781	-1,758143	-0,31495	0,283874	-0,6779	0,470183	0,833156
YAL036C	0,955552	-1,48956	-0,499057	-0,099143	-0,674453	1,414381	0,39228
YAL040C	1,454598	-0,735149	-0,073602	-0,618896	-0,789697	1,391507	-0,62876
YDL037c	1,698578	-1,268638	-0,375511	0,996817	-0,319158	-0,344837	-0,387251
YDR184C	1,302831	-1,338633	-0,555163	-0,101402	-0,674167	1,317301	0,049233
YDR299W	1,431937	-1,819908	0,056708	0,628218	-0,461532	0,001474	0,163103
YDR380W	0,472715	-1,713187	-0,916225	-0,131575	0,433669	0,731209	1,123394
YDR398W	1,235163	-1,796627	-0,062241	0,41659	-0,767331	0,539733	0,434713
YER006w	1,927985	-1,138106	-0,583176	0,056551	-0,731212	0,225849	0,242109
YER064c	1,93726	-1,094734	-0,423042	-0,165499	-0,823443	0,316565	0,252894

Fig. 1. Raw dataset

2 Description of the methods

2.1 Dataset preprocessing

The following preprocessing steps were performed to normalize our dataset:

- The 'Genes' column was removed
- Values were converted from string format to float
- Quantile normalization was applied to ensure uniform distribution of expression values
- Additional dataset $X_{\text{without\_time\_step}}$ was created by removing specific time steps (t_i for $i \in \{0, 0.5, 2, 5, 7, 9, 11.5\}$) for sensitivity analysis

	t0	t0,5	t2	t5	t7	t9	t11,5
469	0.568757	-0.399072	0.761932	-0.981090	-1.266280	-0.407033	0.715941
470	0.600150	-0.617564	-0.001640	-0.986150	-0.604192	-0.220052	0.766354
471	0.912232	0.445928	0.362816	-1.230235	-0.686758	-0.542334	0.068079
472	0.820508	0.517894	-0.823741	-0.785677	-0.823741	-0.517034	0.505396
473	0.604755	0.732262	-0.049824	-1.458405	-0.993510	-0.604192	0.772927

Fig. 2. Filtered dataset

2.2 Basic idea of K-mean algorithm

The K-means algorithm has two main steps:
1. **Initialization**: The algorithm places each item into one of k groups
2. **Iteration**: The algorithm measures the distance between each item and each group, then moves the item to the closest group
Of course we need to adapt this algorithm to our case, where 'groups' correspond to 'clusters', and 'items' correspond to 'genes'.

2.3 Silhouette Index Calculation

The Silhouette Index is a metric to evaluate clustering quality. It ranges from -1 to 1. Values close to 1 indicate well-clustered data points, values close to 0 suggest points on cluster boundaries, and values close to -1 indicate potential misclustering. It was calculated using the formula:

$$S(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}$$

where $a(i)$ is the average distance between a point and others within the same cluster, and $b(i)$ is the average distance between a point and points in the nearest cluster.

3 Relevant results

3.1 K-means Clustering

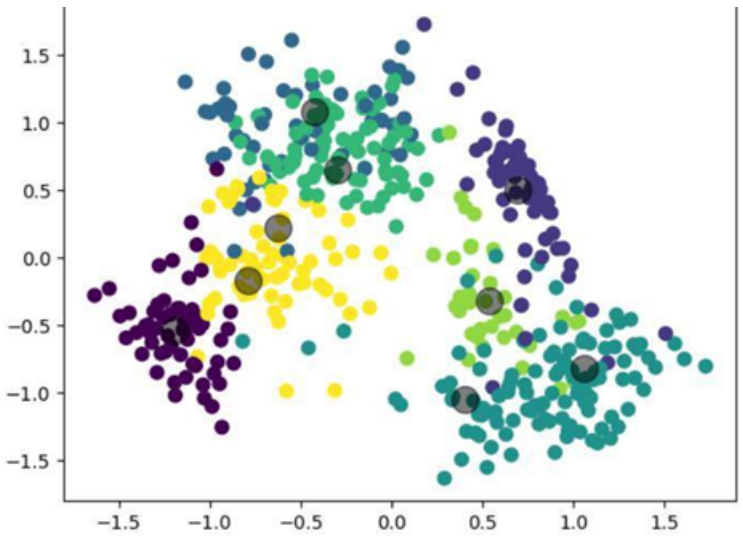

Fig. 3. K-Means clusters

The K-Means clustering algorithm has identified 6 distinct groups within the data. The clusters are well-separated, but still some overlap exists. The centroids are well-placed. Further exploration with different parameter settings and initialization methods could potentially improve the clustering results.

3.2 Silhouette Index Calculation

```
Silhoutte score for kmeans using the dataset without time step 1:  0.29813392976351577
Silhoutte score for kmeans using the dataset without time step 2:  0.3039963424358252
Silhoutte score for kmeans using the dataset without time step 3:  0.31165917723978304
Silhoutte score for kmeans using the dataset without time step 4:  0.30831446643986377
Silhoutte score for kmeans using the dataset without time step 5:  0.28466261832707046
Silhoutte score for kmeans using the dataset without time step 6:  0.2688603904229043
Silhoutte score for kmeans using the dataset without time step 7:  0.2814715284938461
Silhoutte score for kmeans using the whole dataset:  0.3052002656405528
Silhoutte score for kmeans with 2 clusters:  0.41119467900095547
Silhoutte score for kmeans with 3 clusters:  0.38667901669748783
Silhoutte score for kmeans with 4 clusters:  0.3810762842794927
Silhoutte score for kmeans with 5 clusters:  0.386493449058845
Silhoutte score for kmeans with 6 clusters:  0.31073493173064404
Silhoutte score for kmeans with 7 clusters:  0.3052048580414312
Silhoutte score for kmeans with 8 clusters:  0.29964948021510374
Silhoutte score for kmeans with 9 clusters:  0.24759439741064307
```

Fig. 4. Silhoutte scores

Removing individual time steps generally led to a slight decrease in the silhouette score, indicating that while certain time steps might contain redundant information, they also contribute to the overall clustering structure.
The highest silhouette score was achieved with 2 clusters, thus suggesting that a simpler clustering solution might be more appropriate for this dataset. As the number of clusters increased, the silhouette score generally decreased, which indicates a potential overfitting of the data.
While removing specific time steps or increasing the number of clusters did not significantly improve the clustering results, the optimal configuration seems to be a simple model with fewer clusters.

4 Comparison with published results

4.1 Average proportion of non-overlap measure

$$V_1(K) = \frac{1}{Ml} \sum_{g=1}^{M} \sum_{i=1}^{l} \left(1 - \frac{n(C^{g,i} \cap C^{g,0})}{n(C^{g,0})} \right)$$

As written in *Comparisons and validation of statistical clustering techniques for microarray gene expression data*: 'This measure computes the (average) proportion of genes that are not put in the same cluster by the clustering method under consideration on the basis of the full data and the data obtained by deleting the expression levels at one time point at a time.'
RECEIVED RESULT: 0.2198740163236565
The calculated value closely matches published results, affirming the algorithm's consistency in identifying stable clusters when reducing the dataset.

4.2 Average distance between means measure

$$V_2(K) = \frac{1}{Ml} \sum_{g=1}^{M} \sum_{i=1}^{l} d\left(\overline{x}_{Cg,i}, \overline{x}_{Cg,0}\right)$$

As written in *Comparisons and validation of statistical clustering techniques for microarray gene expression data*: 'This measure computes the (average) distance between the mean expression ratios (log transformed) of all genes that are put in the same cluster by the clustering method under consideration on the basis of the full data and the data obtained by deleting the expression levels at one time point at a time.'
RECEIVED RESULT: 2.372057346679048
The result we got is similar to what other researchers have found. This shows that the K-means algorithm is good at keeping the average distance between gene expression profiles the same across different experimental conditions.

4.3 Average distance measure

$$V_3(K) = \frac{1}{Ml} \sum_{g=1}^{M} \sum_{i=1}^{l} \frac{1}{n(Cg,0)n(Cg,i)} \times \sum_{g \in Cg,0, g' \in Cg,i} d(x_g, x_{g'}),$$

As written in *Comparisons and validation of statistical clustering techniques for microarray gene expression data*: 'This measure computes the (average) proportion of genes that are not put in the same cluster by the clustering method under consideration on the basis of the full data and the data obtained by deleting the expression levels at one time point at a time.'
RECEIVED RESULT: 0.04429019394535473
The significantly lower value (around 12 times smaller) suggests either differences in datasets or preprocessing methods. We need to look more closely at why this difference exists.

5 Analysis of the clustering performance of k-Means

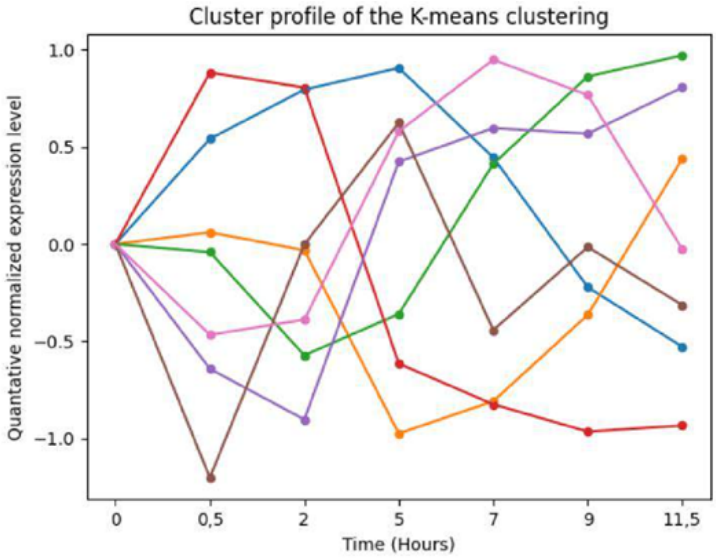

Fig. 5. Our results after normalization

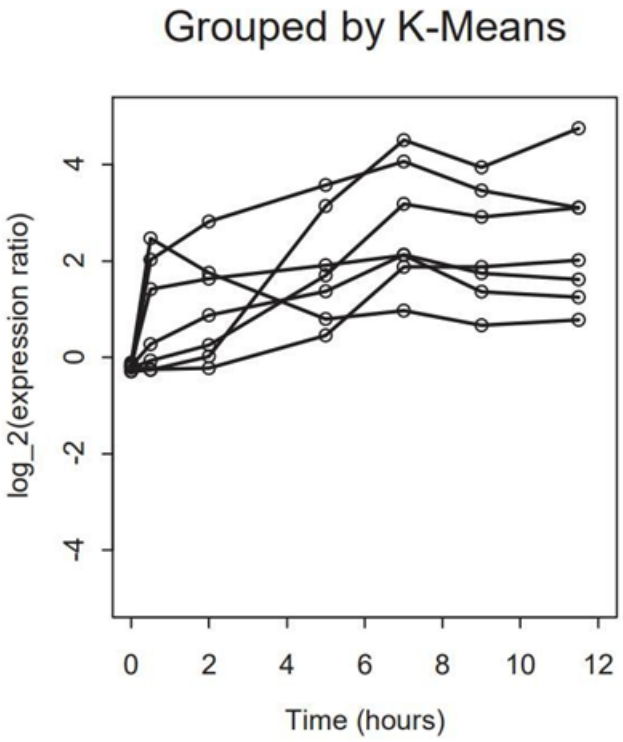

Fig. 6. Published results

Comparison between the two graphs shows a strong alignment in general clustering patterns. However Figure 5 shows more clusters with diverse profiles, while Figure 6 shows fewer clusters with similar trends. In conclusion, some clusters differ in size and composition, likely due to preprocessing steps or parameter variations in the K-means implementation.

6 Conclusions

6.1 Similarity in clusters

Genes in the same cluster often show similar patterns of expression over time, which suggests they might be regulated together or play roles in related biological processes.

6.2 Differences among clusters

The different patterns seen in various clusters show that gene expression is diverse within the dataset. This indicates that the clustering method has successfully grouped genes with different functions.

6.3 Algorithm Robustness

Despite limitations, the K-means algorithm demonstrated robustness across reduced datasets, maintaining biologically meaningful clusters.

6.4 Future Improvements

Incorporating hybrid methods, such as hierarchical clustering with K-means, could address overlapping clusters and improve accuracy.