Breast Cancer Gene Expression Interactive Visualization Dashboard

Nathan Devaux[1] and Louis-David Piron[1]

[1] *Data Science Master Program (DATS2M), UCLouvain*

LDATA2010 - Information Visualization | Group P | Academic Year 2025-2026

Abstract

This dashboard makes exploratory analysis of high-dimensional breast cancer genomics accessible to users with diverse expertise levels. Implemented on the METABRIC dataset (1,904 patients, 693 variables), it integrates dimensionality reduction (PCA, UMAP with rank-based quality metrics), clustering (K-means, hierarchical, DBSCAN), and interactive linked views. The dual learner-expert interface balances pedagogical guidance for less experienced users and analytical flexibility for more experienced ones.

1. Introduction

Although standard clinical breast cancer classification techniques based on receptor status or tumor histology provide valuable guidance for diagnosis and prognosis, they do not fully account for the molecular heterogeneity that shapes disease behavior. Genomic data offer a way to characterize that diversity, but their high dimensionality can be a barrier, especially for clinicians, researchers, or students who lack experience with high dimensional data analysis.

To address this gap, we developed an interactive Shiny dashboard that democratizes the exploration of breast cancer genomic profiles by combining rigorous dimensionality reduction and clustering techniques with an adaptive user experience. The proposed dashboard features two modes: a *Learning Mode*, that provides non-experts a guided workflow and theoretical explanations, and an *Expert mode*, that grants a flexible platform compiling resources for a fast and complete analysis.

The report is organized as follows : Section 2 gives an overview of the data used for the dashboard as well as the preprocessing steps; Section 3 details the dashboard workflow and design choices; Section 4 provides theoretical background for interpreting the dashboard's dimensionality reduction and clustering modules. Finally, Section 5 reflects on the dashboard limitations, and how the sofware could be improved.

2. METABRIC Dataset

The dashboard uses breast cancer genomic data from the Molecular Taxonomy of Breast Cancer Consortium (METABRIC) study [Christina Curtis, 2012]. METABRIC comprises genomic and transcriptomic profiles of approximately 1,900 breast tumors collected between 1977 and 2005 across five medical centers in the United Kingdom and Canada.

The original METABRIC study [Christina Curtis, 2012] identified ten molecular subtypes termed *integrative clusters* (IntClusts), characterized by distinct genomic drivers. These IntClusts exhibit associations with clinical outcomes and traditional pathological features [Mukherjee et al., 2018]. Our dashboard enables exploration of this classification alongside more classical molecular classifications (*e.g.*, PAM50+Claudin-low).

2.1. Preprocessing

The METABRIC dataset is already extensively preprocessed. However, some additional preprocessing steps were performed to enhance downstream analyses. First, mutation columns (`*_mut`) and the variable `cancer_type` were excluded from the analysis, as the former consisted of sparse categorical variables and the latter contained only a single instance of ”Breast Sarcoma”, with the remainder classified as ”Breast Cancer”. Then, rather than using imputation, missing values were preserved and mapped to the `UNK` category to account for *informative missingness.*

Gene expression variables represent 489 (94%) of the 519 features in the preprocessed dataset. No additional preprocessing was required for these variables, as they were already standardized (z-scores[1]) and did not present any missing values. For a visual confirmation, the `Quality Check` pane of the dashboard dislpays a heatmap of the data missingess.

To facilitate analysis, visualization, and improve user experience, variables were organized into biologically and clinically meaningful categories, including: demographic characteristics, medical interventions, clinical outcomes, tumor features, molecular classifications, and gene expression profiles. A detailed explanation of all features belonging to these categories (apart from gene expression) is provided in the `About METABRIC` pane of the dashboard.

3. Workflow & UI Design

The dashboard is structured to guide users through a natural left-to-right progression (Figure 1), though users can navigate freely between tables :

1. `Data Understanding & QC` ⟶ 2. `EDA` ⟶ 3. `DR` ⟶ 4. `Clustering`

Figure 1: Dashboard’s suggested workflow

Design choices are mainly inspired by the following principles: Cleveland-McGill’s perceptual effectiveness ranking for the charts [Cleveland and McGill, 1984], the CRAP framework (Contrast, Repetition, Alignment, Proximity) for the interface layout [Williams, 2015], and Midway’s data visualization principles [Midway, 2020]. The following subsections detail steps 2 to 4, highlighting specific design choices.

3.1. Learning and Expert Modes

Two modes are offered to the user: a Learning Mode (🎓), which provides *Learner Guide* tabs with theoretical explanations and recommended parameters; an Expert Mode (🔬), that removes instructional content and unlocks more parameter settings. Users can toggle from one mode to another via the navigation bar at any time.

3.2. Exploratory Data Analysis (EDA)

After having understood the purpous of the dashboard, the overall data structure, and made sure that the data met quality requirements (`Overview` pane), one of the most important part when we are first identifying a dataset is the EDA.

The first two views of the EDA pane consist of a univariate analysis (to get a feeling of distributions) and a bivariate one (to better grasp associations). Each view displays two plots side-by-side, automat-

[1] Without z-score standardization, highly expressed genes would dominate the during PCA and clustering, regardless of their biological relevance.

ically adapting charts based on variable data types [2]. Following Cleveland and McGill's perceptual hierarchy Cleveland and McGill [1984], which ranks position along common scales as most accurate, we prioritize (among others) histograms, bar charts, and scatter plots over less effective encodings like pie charts.

The third view is a `Gene Expression` tab featuring a Volcano Plot, which enables users to take a snapshot of the differential gene expression between two groups. It quickly reveals which genes show the strongest differences, helping users identify candidate biomarkers, whithout resorting to more advanced DR and Clustering techniques. Interactive thresholds allow filtering by effect size and statistical confidence (Figure 2).

From a visualization design perspective, this approach demonstrates several key principles. Following Midway's third and fourth principles [3] [Midway, 2020],all 489 genes remain visible as individual points rather than being reduced to summaries. And color encoding provides semantic meaning: red and green distinguish up- versus down-regulated genes. Furthermore, the interactive thresholds implement CRAP's contrast principle [Williams, 2015] by visually partitioning the expression space into distinct regions, guiding users toward genes exhibiting both large magnitude changes and strong statistical evidence.

3.3. Dimension Reduction (DR)

As the overall dashboard, the DR module is structured in a sequential way. First, the user is invited to run an analysis using either the default parameters or their own. Based on these inputs, an entire workflow is executed in background. Progress is communicated through a series of pop-up, with messages adapted to the user's parameter choices.

The final output with default parameters is given as an example in 3 : tab 3a, provides an estimate of the underlying dimensionality of the data ; tab 3b, visualises the UMAP and PCA projections ; tab 3c, assesses how faithfully each DR technique preserves the neighbourhood structure based on several rank-based metrics. The complete workflow processes the full cohort in under 2 minutes.[4] All in all, it would provide valuable insights to, say, a researcher that would like to understand receptor status heterogeneity from a genomic standpoint.

Design-wise, the workflow is articulated around 2 main principles : (i) Traceability and (ii) Comparison. We considered (i) by setting internal seeds for stochastic parts of the workflow and by saving each user run into a `past run` pane. This pane enables both (i) and (ii), as the user can trace past parameters and compare previous runs in terms of the AUC (of any rank-based quality metric). (ii) is also supported through other parts of the module : 2 different intrinsic dimensionality estimation methods ; 2 different DR techniques ; 4 different quality metrics.

3.4. Clustering

The clustering section also has a sidebar similar to the DR section to change the parameters of the clustering methods. By maintaining the same structure, we ensure that the user does not need to relearn a new layout, and it facilitates a transition between analysis stages, we adhere to the Repetition principle [Williams, 2015].

We also designed the three clustering parts with the same comprehensive layout. In each method, the dashboard is composed of a scatter plot, representing the clusters in the reduced space ; quality

[2] Colors follow the OKABE color-blind palette

[3] Respectively : "Use an Effective Geometry and Show Data" and "Colors Always Mean Something"

[4] First run includes intrinsic dimensionality estimation (140s); subsequent ones take 90s. Users can further adjust computational speed by reducing the sample size.

metrics panel summarizing performance ; and visualization of the evolution of the clustering metrics.

Figures 4a, 4b and 4c represent an exemple if the user keeps the default parameters. The different algorithms are fast to compute as we used the Joblib library in our implementation.

In the "Univariate" panel present in the first three parts, the user can check the resulting clusters. They are able to see how the cluster relate to original clinical variables. As proved in [Tufte, 2006], "enforcing comparison" is fundamental principle in visualisation.

Finally, the comparison tab serves as a conclusion to the analysis. In fact, a history panel allows the user to check the previous iterations. It also provides a comparative view of the different clustering iteration and their corres visualizations.

4. Theory Behind the dashboard

4.1. Dimensionality Reduction

In a nutshell, dimensionality reduction seeks to discover a mapping $f : \mathbb{R}^p \to \mathbb{R}^d$ that preserves the intrinsic geometry of an initial high-dimensional space $\mathbb{R}^p$.

One motivation behind such a technique, is that real-world data rarely exploits the full ambient space $\mathbb{R}^p$. Instead, we assume the *manifold hypothesis*: high-dimensional observations are sampled form a lower-dimensional manifold $\mathcal{M}$, embedded in $\mathbb{R}^p$, with intrinsic dimensionality $d \ll p$.

The scikit-dimension package provides a collection of mathematical techniques for estimating the intrinsic dimensionality. Among the available methods, we retained the exponential correlation estimator [Peter Grassberger, 1983], which remains one of the most widely used benchmarks, as well as the Fisher separability analysis [Albergante et al., 2019], which has proven particularly robust for high-dimensional biological data.

For the implementation of the DR methods, we used `scikit-learn` Pedregosa et al. [2011] for PCA and the `umap-learn` package McInnes et al. [2018] for UMAP.

4.1.1. Principal Component Analysis

PCA is a linear dimensionality reduction method as it assumes that $\mathcal{M}$ is a linear subspace. That $d-$dimensional linear subspace is simply composed of d so called *principal components* that are linear combinations of the p initial features.

That combination is chosen to maximize the variance of the projected data. Let $\mathbf{x}_i \in \mathbb{R}^p$ represent patient i's gene expression profile. The projection onto unit vector $\mathbf{v}$ yields coordinates $z_i = \mathbf{v}^\top \mathbf{x}_i$, with variance: $\sigma_v^2 = \mathbf{v}^\top \mathbf{\Sigma} \mathbf{v}$, where $\mathbf{\Sigma}$ is the empirical covariance matrix.

Maximizing σ_v^2 subject to $\|\mathbf{v}\| = 1$ via Lagrangian optimization yields the eigenvalue problem: $\mathbf{\Sigma}\mathbf{v} = \lambda\mathbf{v}$. The k-th principal component $\mathbf{v}_k$ is the k-th eigenvector of $\mathbf{\Sigma}$, ordered by decreasing eigenvalue $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0$. Crucially, each eigenvalue equals the variance explained by its component: $\sigma_{v_k}^2 = \lambda_k$. Hence, the proportion of variance captured by the first k components is: $\tau_k = \sum_{\alpha=1}^{k} \lambda_\alpha / \sum_{\alpha=1}^{p} \lambda_\alpha$. This can be used to assess the quality of DR. The closer it is to 100%, the better we explain the variance of the original data.

Why PCA for gene expression? PCA provides a simple (linear, interpretable and parameter-free) baseline for exploring the gene expression space. Since all genes are already standardized to

z-scores, each feature contributes on the same scale. The leading principal components then allow us to reveal dominant sources of variation, such as differences between patient subgroups (e.g., ER+ vs ER–), that explain the most variation in gene expression.

4.1.2. Uniform Manifold Approximation and Projection

UMAP is a non-linear dimensionality reduction method operating in two phases McInnes et al. [2020]: (i) for each observation, construct a local fuzzy simplicial set based on its k nearest neighbors, then combine these via fuzzy set union to form a global topological representation; (ii) initialize a low-dimensional embedding via spectral methods, then iteratively optimize it by minimizing cross-entropy between high- and low-dimensional graph structures. Key hyperparameters include `n_neighbors` (balancing local vs. global structure preservation), `min_dist` (minimum separation in embedding space), and `metric` (distance function, only tunable in Expert Mode).

Why UMAP for gene expression? UMAP provides a powerful non-linear alternative to PCA. Unlike t-SNE, UMAP scales efficiently to large cohorts. This combination of expressiveness, and speed makes it ideal for genomic data.

4.1.3. Quality Assessment via Co-Ranking Framework

Following Lee and Verleysen [2009], we use a co-ranking framework to asses the quality of our DR. The co-ranking matrix **Q** systematically compares neighborhood rankings between spaces. Each element (i, j) represents the number of times that the $i-th$ neighbor in the high-dimensional space receives a different ranking j in the low-dimensional space [Remacle, 2025].

Diagonal entries represent perfectly preserved neighborhoods. Off-diagonal entries quantify two types of rank errors: **intrusions** (distant points incorrectly brought together) ; and **extrusions** (nearby points incorrectly separated).

Our dashboard displays 4 metrics [5] that can be directly derived from **Q** :

1. **Trustworthiness ($T(K)$) and Continuity ($C(K)$)** : $T(K)$ quantifies wether the embedding preserves local relationships found in the original data. On the other hand, $C(K)$ quantifies how accurately the embedding reflects the original structure of the data.
2. **Behavior $B_{NX}(K)$:** quantifies the tendency of the embedding to favor either extrusive or intrusive behaviors. A positive (negative) $B_{NX}(K)$ indicates that we have more (less) mild intrusions than extrusions.
3. **Normalized Quality $R_{NX}(K)$:** measures the proportion of correctly ranked neighbors up to size K, producing a curve that reveals whether the projection emphasizes local structure (high values at low K) or global relationships (plateau across all K). It is often preferred over the unscaled Quality $Q_{NX}(K)$, as $R_{NX}(K)$ yield a value of 0 for a random embedding, unlike $Q_{NX}(K)$.

An optimal embedding would yield to $R_{NX}(k)$ $T(k)$, $C(k)$ of 1 and $B_{NX}(k)$ of 0, $\forall k \in \{1, \dots, K\}$. We refer the user to Remacle [2025] for the detailed formulas as well as a thorough assessment of those metrics.

To provide a scalar summary, we also compute the **Area Under the Curve (AUC)** by integrating one of the above metrics (except $B_{NX}(K)$) over K. Higher AUC values indicate better overall ~~neighborhood preservation across the full~~ sample.[6]

[5] Our dashboard rely on some function of Zhang et al. [2021] as well as built-in functions to compute those metrics.

[6] User in learning mode only have the AUC of the R_{NX}. The AUC is computed using trapezoidal integration.

4.2. Clustering

Three clustering techniques were implemented to compare results and give different perspectives on the data structure. This variety of choices offers the user the opportunity to test and visualize how different clustering methods group patients together based on their genomic profile. The results of the methods. Indeed, given the high dimensionality, these clustering methods lead to distinct groups, that bear different meanings. Consequently, the user has the power to select the most coherent visualization for the data.

All three methods are possibly falling in the "Curse of Dimensionality" [Beyer et al., 1999]. This phenomenon happens when distance metrics behave unintuitively in high-dimensional spaces. For this reason, we employ DR techniques before implementing the clustering methods.

We give to the user the same technique as in the previous section: PCA and UMAP. He can change the parameters of these two techniques directly in the clustering section or import the parameters used in the last DR iteration.

K-Means and DBSCAN were implemented with the `scikit-learn` Pedregosa et al. [2011] library and the Hierarchical Clustering was implemnted with SciPy, the library was used for computing the linkage between the clusters and build the dendrogram.

4.2.1. K-means

K-means is a fundamental clustering algorithm. It's an algorithm divided into three stages.

1. **Initialization** : k random points are placed within the data space, called centroids.
2. **Assignment** : each observation is assigned to the closest centroid based on Euclidean distance.
3. **Update** : all the centroid are recalculated to the mean position of the new group.

The **Assigment** and **Update** steps iterate until convergence. The convergence occurs when centroids no longer move.

Since the initialization of the algorithm is random, thus its outcomes, we compute the algorithm 10 times and select the best iteration (minimizing the within-cluster sum of squares).

To evaluate the clarity and separation of the clusters, we use the Calinsk-Harabasz (CH) [Caliński and Harabasz, 1974] score and also the Gap Statistic [Tibshirani et al., 2001].

The **Calinski-Harabasz** index represents the ratio of the between sum of square and the within sum of square dispersion between the clusters. The objective of the metric is to have a small distance between the point in the clusters, and to have a clear distance between the clusters.

The **Gap Statistic** index compares the within-cluster variation for k with their expected values under the uniform distribution of the data. The goal is to maximize cluster compactness in comparaison of the uniform distribution.

4.2.2. Hierarchical Clustering

In the Hierarchical Clustering, we use the same quality metrics as in K-means part.

The algorithm follows an agglomerative (bottom-up) approach, divided into three stages. Initially, it starts at the finest partition where each individual forms a single cluster. Then, the algorithm iteratively aggregates or merges classes one by one up to the coearsest partition.

These aggregations rely on **Linkage methods** to define distances between clusters. We allow the user to choose between the **Centroid Linkage** and **Complete Linkage (Max-Linkage)**. The **Centroid Linkage** measures the distance between the geometric centers of two clusters and the **Complete Linkage** measures the maximum distance between any pair of points in two different clusters. These distances are well used in bioinformatics as in [Eisen et al., 1998], and we selected them to give robust options without overwheling the user with too much distances.

A specific feature of Hierarchical Clustering is the **Dendogram**. This tree diagram displays the hierarchy of clusters produced by the corresponding linkage method. Since it was impossible to show all 1900 individuals (patients) in the dendogram, it was set to be reactive to the number of dimensions in the preprocessing phase. Although hierarchical clustering is not a semi-supervised technique we require initializing a number of clusters to get a neater layout.

4.2.3. DBScan

The Density-Based Spatial Clustering of Applications with Noise is a method introduced by [Ester et al., 1996] to capture clusters of arbitraly complex shapes.

It works with the concept of noise and density. Observations are not pushed into a cluster as they could be considered as noise. The method groups points that are closely packed together based on a distance threshold ϵ. It marks points located in low density regions as a noise. This technique is able to identify sub-populations of patients while putting apart ambiguous genomic porfiles that do not fit into any specific cluster.

Howerver, selectig a coherent ϵ and min samples is challenging. In fact, when the user chooses an ϵ that is too small, valid data could be treated as noise, and when the user choose a too large ϵ, distinct groups merge into one big cluster.

The K-distance graph helps the user to choose a coherent ϵ. The plot gives the distance to nearest neigbors for every points, and it is sorted from smallest to largest. The point of maximum curvature indicates an optimal ϵ. Before the maximum curvature, the points are represented within clusters. The rest of the point, on the right of the maximum curvature, corresponds to outliers (categorised as noise).

The DBSCAN quality metric is the DBCV (Density-Based Clustering Validation) metric [Moulavi et al., 2014]. The metric varies between -1 to 1, a positive value indicates a good separation whereas a negative value means that the separation is not effective.

5. Future Improvements

Reflecting on our work, several promising directions emerge for extending the dashboard's capabilities.

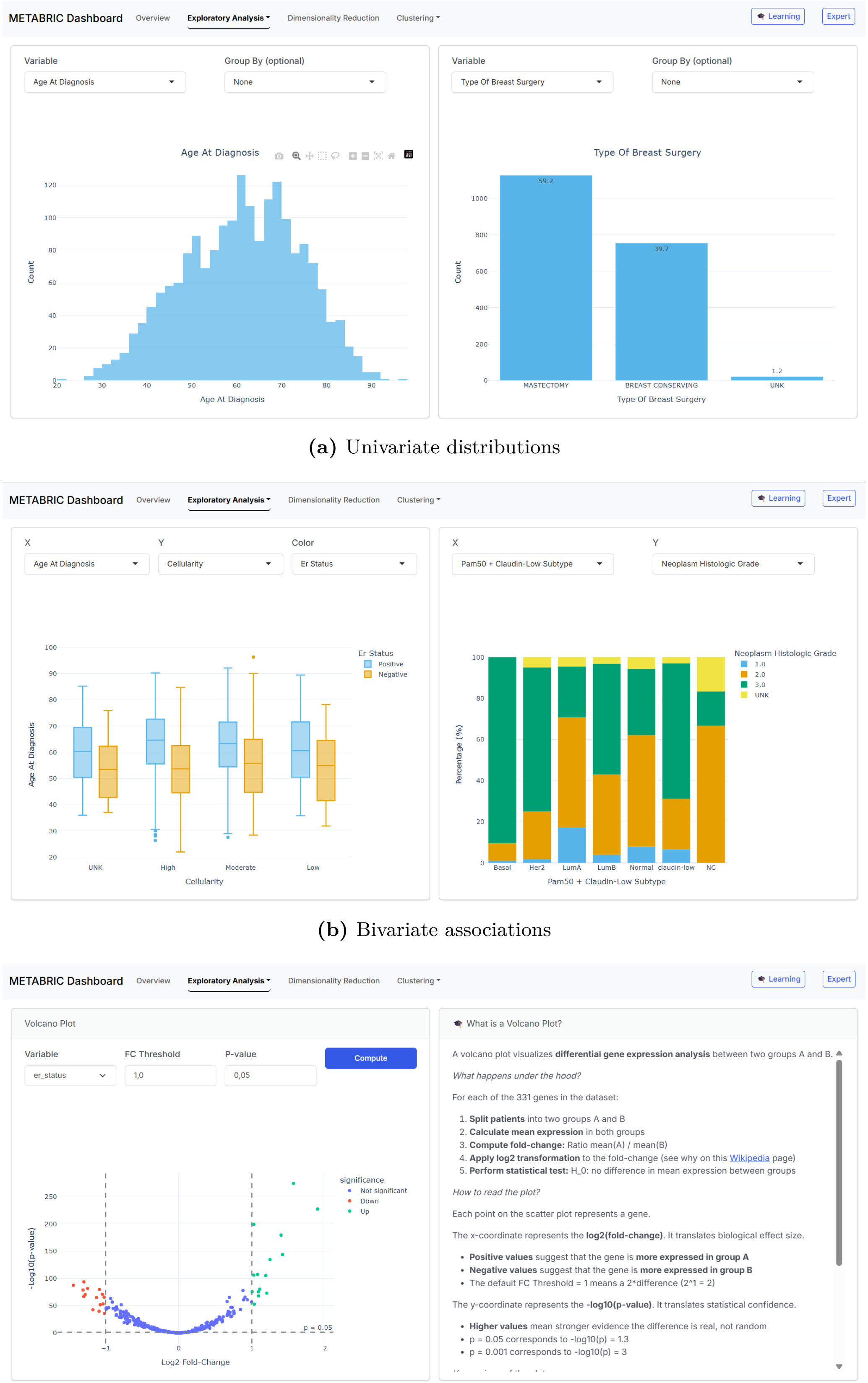
First, while we implemented parallelization for clustering algorithms, systematically extending this approach to the rest of the dashboard would enhance performance for larger cohorts.

Second, we deliberately focused on gene expression profiles as they constitute the richest signal in METABRIC, but integrating mutation data could reveal additional associations.

Third, incorporating survival analysis or more standard machine learning algorithms would be insightfull given the rich outcome variables the dataset disposes. Domain-specific methods from the breast cancer literature could also be integrated in the workflow.

Finally, though cross-filtering between plots was not deemed essential for our use-case, it could be worth implementing to support interactive exploration in more complex research scenarios.

6. Appendix



Variable: Age At Diagnosis
Group By (optional): None

Variable: Type Of Breast Surgery
Group By (optional): None

(a) Univariate distributions

X: Age At Diagnosis
Y: Cellularity
Color: Er Status

X: Pam50 + Claudin-Low Subtype
Y: Neoplasm Histologic Grade

(b) Bivariate associations

Volcano Plot

Variable: er_status
FC Threshold: 1,0
P-value: 0,05
Compute

What is a Volcano Plot?

A volcano plot visualizes **differential gene expression analysis** between two groups A and B.

What happens under the hood?

For each of the 331 genes in the dataset:

1. **Split patients** into two groups A and B
2. **Calculate mean expression** in both groups
3. **Compute fold-change:** Ratio mean(A) / mean(B)
4. **Apply log2 transformation** to the fold-change (see why on this Wikipedia page)
5. **Perform statistical test:** H_0: no difference in mean expression between groups

How to read the plot?

Each point on the scatter plot represents a gene.

The x-coordinate represents the **log2(fold-change)**. It translates biological effect size.

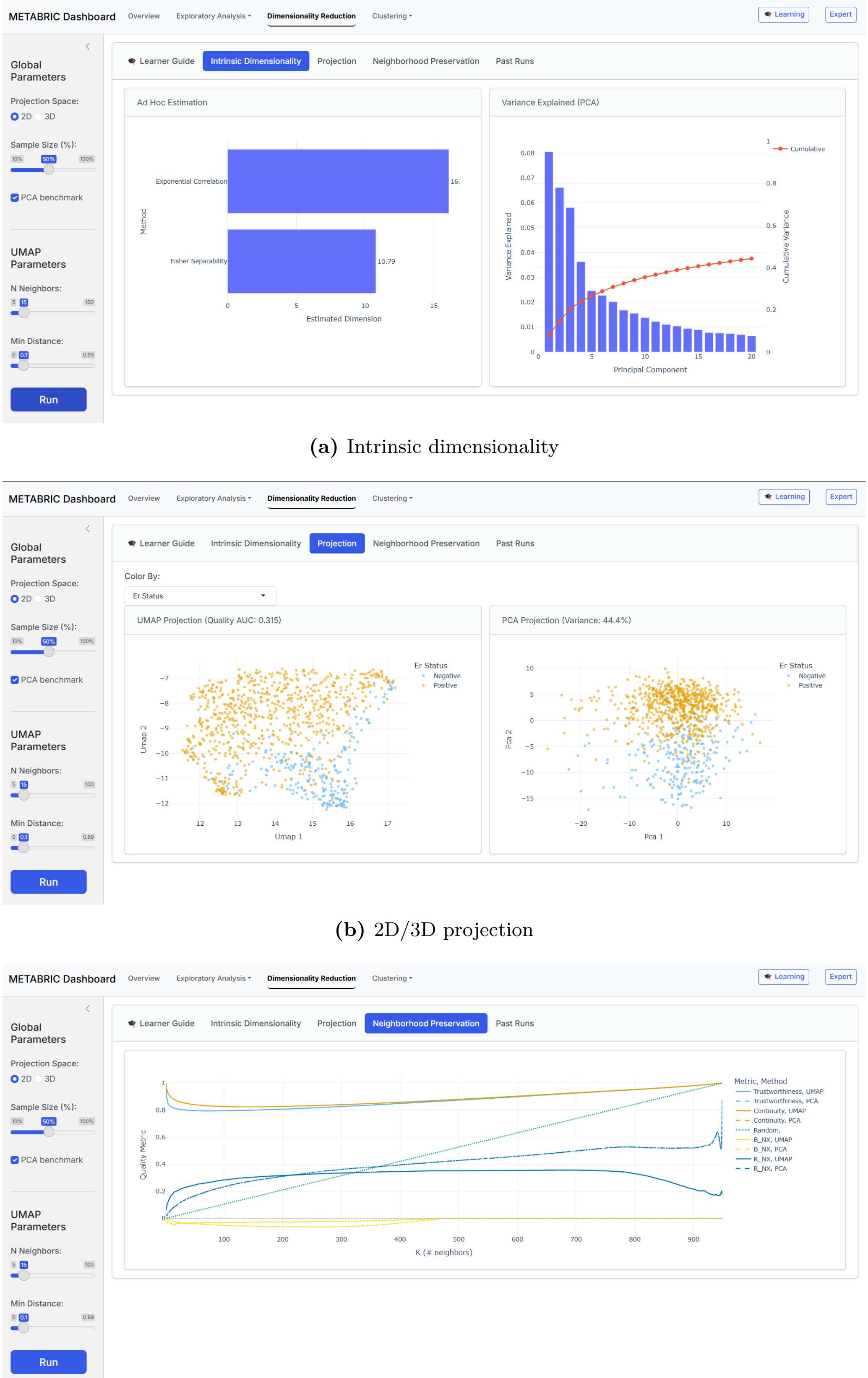
- **Positive values** suggest that the gene is **more expressed in group A**
- **Negative values** suggest that the gene is **more expressed in group B**
- The default FC Threshold = 1 means a 2*difference (2^1 = 2)

The y-coordinate represents the **-log10(p-value)**. It translates statistical confidence.

- **Higher values** mean stronger evidence the difference is real, not random
- p = 0.05 corresponds to -log10(p) = 1.3
- p = 0.001 corresponds to -log10(p) = 3

(c) Volcano plot

Figure 2: Exploratory Data Analysis pane (learning mode on).



Projection Space: ● 2D ○ 3D

Sample Size (%): 10% 50% 100%

☑ PCA benchmark

N Neighbors: 5 15 100

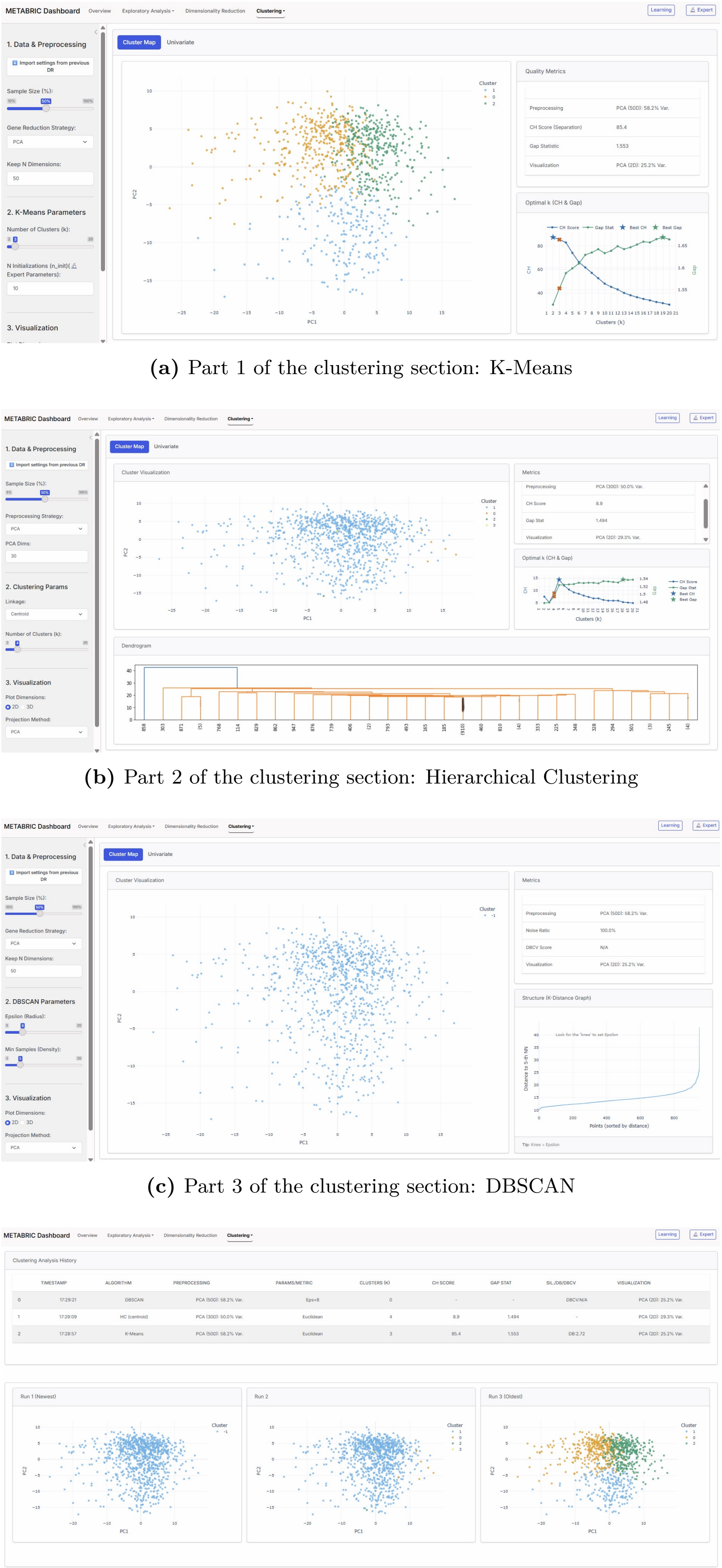
Min Distance: 0 0.1 0.99

(a) Intrinsic dimensionality

(b) 2D/3D projection

(c) Quality metrics

Figure 3: Dimensionality Reduction workflow (learning mode on).



1. Data & Preprocessing

Import settings from previous DR

Sample Size (%): 50%

Gene Reduction Strategy: PCA

Keep N Dimensions: 50

2. K-Means Parameters

Number of Clusters (k): 3

N Initializations (n_init)(Expert Parameters): 10

3. Visualization

Quality Metrics

Preprocessing	PCA (50D): 58.2% Var.
CH Score (Separation)	85.4
Gap Statistic	1.553
Visualization	PCA (2D): 25.2% Var.

(a) Part 1 of the clustering section: K-Means

1. Data & Preprocessing

Import settings from previous DR

Sample Size (%): 50%

Preprocessing Strategy: PCA

PCA Dims: 30

2. Clustering Params

Linkage: Centroid

Number of Clusters (k): 4

3. Visualization

Plot Dimensions: ● 2D ○ 3D

Projection Method: PCA

Metrics

Preprocessing	PCA (30D): 50.0% Var.
CH Score	8.9
Gap Stat	1.494
Visualization	PCA (2D): 29.3% Var.

(b) Part 2 of the clustering section: Hierarchical Clustering

1. Data & Preprocessing

Import settings from previous DR

Sample Size (%): 50%

Gene Reduction Strategy: PCA

Keep N Dimensions: 50

2. DBSCAN Parameters

Epsilon (Radius): 8

Min Samples (Density): 5

3. Visualization

Plot Dimensions: ● 2D ○ 3D

Projection Method: PCA

Metrics

Preprocessing	PCA (50D): 58.2% Var.
Noise Ratio	100.0%
DBCV Score	N/A
Visualization	PCA (2D): 25.2% Var.

(c) Part 3 of the clustering section: DBSCAN

Clustering Analysis History

	TIMESTAMP	ALGORITHM	PREPROCESSING	PARAMS/METRIC	CLUSTERS (K)	CH SCORE	GAP STAT	SIL./DB/DBCV	VISUALIZATION
0	17:29:21	DBSCAN	PCA (50D): 58.2% Var.	Eps=8	0	-	-	DBCV:N/A	PCA (2D): 25.2% Var.
1	17:29:09	HC (centroid)	PCA (30D): 50.0% Var.	Euclidean	4	8.9	1.494	-	PCA (2D): 29.3% Var.
2	17:28:57	K-Means	PCA (50D): 58.2% Var.	Euclidean	3	85.4	1.553	DB:2.72	PCA (2D): 25.2% Var.

(d) Part 4 of the clustering section: Cluster Comparison

Figure 4: Clustering Section

References

Luca Albergante, Jonathan Bac, and Andrei Zinovyev. Estimating the effective dimension of large biological datasets using fisher separability analysis, January 2019.

Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft. When is "nearest neighbor" meaningful? *Database Theory—ICDT'99*, pages 217–235, 1999.

Tadeusz Caliński and Jerzy Harabasz. A dendrite method for cluster analysis. *Communications in Statistics*, 3(1):1–27, 1974.

Suet-Feung Chin Gulisa Turashvili Oscar M. Rueda Mark J. Dunning Doug Speed Christina Curtis, Sohrab P. Shah. The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups. *Nature*, 486(7403):346–352, 2012. doi: 10.1038/nature10983. URL https://www.nature.com/articles/nature10983.

William S Cleveland and Robert McGill. Graphical perception: Theory, experimentation, and application to the development of graphical methods. *Journal of the American Statistical Association*, 79(387): 531–554, 1984.

Michael B Eisen, Paul T Spellman, Patrick O Brown, and David Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95(25): 14863–14868, 1998.

Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD'96)*, 96(34):226–231, 1996.

John A. Lee and Michel Verleysen. Quality assessment of dimensionality reduction: Rank-based criteria. *Neurocomputing*, 72(7-9):1431–1443, March 2009. doi: 10.1016/j.neucom.2008.12.017.

Leland McInnes, John Healy, Nathaniel Saul, and Lukas Grossberger. Umap: Uniform manifold approximation and projection. *The Journal of Open Source Software*, 3(29):861, 2018.

Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. (arXiv:1802.03426), September 2020. doi: 10.48550/arXiv.1802.03426.

Stephen R. Midway. Principles of effective data visualization. *Patterns*, 1(9):100141, December 2020. doi: 10.1016/j.patter.2020.100141.

Davoud Moulavi, Pablo A. Jaskowiak, Ricardo J. G. B. Campello, Arthur Zimek, and Jörg Sander. Density-based clustering validation. *Proceedings of the 2014 SIAM International Conference on Data Mining (SDM)*, pages 839–847, 2014.

A. Mukherjee, R. Russell, Suet-Feung Chin, B. Liu, O. M. Rueda, H. R. Ali, G. Turashvili, B. Mahler-Araujo, I. O. Ellis, S. Aparicio, C. Caldas, and E. Provenzano. Associations between genomic stratification of breast cancer and centrally reviewed tumour pathology in the metabric cohort. *npj Breast Cancer*, 4(1):5, March 2018. doi: 10.1038/s41523-018-0056-8.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12: 2825–2830, 2011.

Itamar Procaccia Peter Grassberger. Measuring the strangeness of strange attractors, 1983. Received 16 November 1982, Revised 26 May 1983.

Pierre Remacle. Directional quality assessment for nonlinear dimensionality reduction in data visualisation. Master's thesis, Université catholique de Louvain, 2025.

Robert Tibshirani, Guenther Walther, and Trevor Hastie. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):411–423, 2001.

Edward R Tufte. *Beautiful Evidence*. Graphics Press Cheshire, CT, 2006.

Robin Williams. *The Non-Designer's Design Book*. Peachpit Press, Berkeley, California, 4th edition, 2015. Design and typographic principles for the visual novice.

Yinsheng Zhang, Qian Shang, and Guoming Zhang. pydrmetrics – a python toolkit for dimensionality reduction quality assessment. *Heliyon*, 7(2):e06199, 2021. doi: 10.1016/j.heliyon.2021.e06199.