Bioinformatic Analysis of Gene Expression in Breast Cancer

Background

Breast cancer is a significant global health challenge, being one of the most commonly diagnosed cancers among women. In Australia, breast cancer represents a significant public health burden. According to the Australian Institute of Health and Welfare (2024), an estimated 21,807 new cases were diagnosed in 2024, establishing it as the second most common cancer in the country. Despite advances in screening, diagnosis, and treatment, breast cancer management remains complex due to considerable variability in patient responses and clinical outcomes.

This variability has driven growing recognition that breast cancer is not a single disease but a collection of molecularly distinct subtypes, each defined by unique biological characteristics, therapeutic sensitivities, and prognoses (Perou et al., 2000; Sorlie et al., 2001). Understanding these subtypes is crucial, as accurate and early molecular classification allows clinicians to tailor treatments to individual patients, avoid ineffective therapies, and ultimately improve survival outcomes while reducing treatment costs.

Building on this understanding, this study investigates single-channel microarray gene expression data from 251 breast cancer patients, integrated with detailed clinical information including histopathological features, hormone receptor status, and survival outcomes. The study aims to: (1) identify distinct molecular subtypes for breast cancer through unsupervised clustering; (2) characterize the biological signatures between these subtypes via differential expression and enrichment analysis; and (3) evaluate the prognostic relevance of identified subtypes through multivariable survival analysis.

Methodology

All analyses were conducted in R. The main R packages used included *cluster* for cluster analysis, *limma* for differential gene expression analysis, *clusterProfiler* for GO enrichment analysis, and *survival* for survival analysis and visualization.

1. Data Pre-processing

All three datasets—gene expression, annotation, and clinical data—were loaded into R from RDS files. The gene expression matrix contained 22,283 probes across 251 patient samples. Clinical variables included patient ID, histological tumour grade, hormone receptor status (ER and PR), age, tumour size, lymph node status, survival time (years), and event indicator. Annotation data linked probe IDs to HGNC gene symbols and Ensembl gene IDs, enabling biological interpretation of gene expression.

Initial exploratory checks revealed no missing values in the expression or annotation datasets, whereas the clinical dataset contained 30 missing entries, including 15 patients with missing survival time or event status. No duplicate entries were detected across any of the datasets.

To minimize technical variability and batch effects, between-array normalization using the scale method was applied to the expression data (Bolstad et al., 2003). Control probes (with "AFFX" prefix) were excluded as they serve technical quality control purposes rather than providing biological meaningful information. Genes with low expression variability (bottom 10% of variance and Median Absolute Deviation across samples) were also filtered out to reduce noise and focus on informative features.

This comprehensive pre-processing workflow ensured that the final expression matrix contained only high-quality, biologically informative data suitable for downstream analyses, including clustering, differential expression, and survival analysis.

2. Clustering Analysis

To identify intrinsic molecular subtypes, unsupervised clustering was performed using pre-processed gene expression data. At this stage, no additional filtering for highly expressed genes was applied, as the dataset had already undergone preprocessing to remove low-expression genes in step 1. Prior to clustering, gene-wise z-score standardization (Eisen et al., 1998) was applied to normalize expression values across samples, ensuring that each gene contributed equally to the distance metrics in the clustering algorithm.

Hierarchical clustering was applied using different linkage methods to identify distinct patient groupings, including complete, average and Ward's linkage, to identify distinct patient groupings. For each method, silhouette values were calculated to determine the optimal number of clusters (Rousseeuw, 1987). Dendrograms were generated to visualize the hierarchical tree structure and branch heights, providing an initial overview of clustering patterns.

To facilitate interpretation, dendrograms combined with heatmaps were generated to visualize cluster assignments and expression patterns. In addition, Principal Component Analysis (PCA) was performed on the top 1,000 most variable genes to examine the variance structure and to validate the clustering results.

3. Gene Expression Analysis

To identify marker genes distinguishing the identified clusters, differential gene expression (DE) analysis was performed using the limma package (Smyth, 2005).

For each gene i, the null and alternative hypotheses were defined as:

$$H_0 : \mu_{i2} = \mu_{i1} \text{vs.} H_1 : \mu_{i2} \neq \mu_{i1},$$

where μ_{i1} and μ_{i2} represent the mean log_2 expression levels of gene i in Cluster 1 (reference) and Cluster 2, respectively.

The classical t-statistic was written as:

$$T_i = \frac{\bar{M}_{i2} - \bar{M}_{i1}}{S_i\sqrt{\frac{1}{n_1} + \frac{1}{n_2}}},$$

where:

$\bar{M}_{i1}$and $\bar{M}_{i2}$are the sample means for gene $i$in each cluster,

$S_i^2$is the pooled within-group variance, and

$n_1$and $n_2$are the numbers of samples in Cluster 1 and Cluster 2, respectively.

To improve statistical power and reduce noise, empirical Bayes moderation was applied. This moderated t-statistic borrows information across all genes:

$$s_i^{*2} = \frac{d_0 s_0^2 + d_i s_i^2}{d_0 + d_i},$$

where $s_i^{*2}$is the moderated variance, $s_i^2$is the gene-specific sample variance, $s_0^2$is the prior (pooled) variance, and $d_0$and $d_i$are the prior and residual degrees of freedom, respectively.

The resulting moderated t-statistic is:

$$t_i = \frac{\hat{\beta}_i}{s_i^* \sqrt{v_i}},$$

where $\hat{\beta}_i$is the estimated logFC between the two clusters and $v_i$is derived from the design matrix.

To account for multiple hypothesis testing, raw p-values were adjusted using the Benjamini–Hochberg false discovery rate (FDR) procedure (Benjamini & Hochberg, 1995). Genes were considered significantly differentially expressed if they satisfied both statistical significance (FDR < 0.05) and biological significance ($| \, log\text{FC} \, |> 1$).

To facilitate interpretation of the results, volcano plots were generated to display the relationship between biological significant and statistical significance, while p-value histograms were used to visualize the distribution of significance levels before and after FDR correction.

4. Gene Ontology (GO) Enrichment Analysis

Gene Ontology (GO) enrichment analysis was performed to investigate the biological processes associated with the differentially expressed genes identified between the two molecular clusters. Marker genes were first extracted based on both statistical significance (FDR < 0.05) and biological significance ($|logFC|$ > 1) from the differential expression analysis. Probe identifiers were then mapped to their corresponding HGNC gene symbols to ensure standardized gene annotation. Enrichment analysis was carried out using the clusterProfiler package in R (Yu et al., 2012). To facilitate interpretation, results were visualized using dot plots to highlight the most significantly enriched biological processes and their corresponding gene ratios.

5. Survival Analysis

To conclude the analysis, survival analysis was conducted to evaluate the association between molecular subtypes and clinical outcomes. Overall survival time was defined as the duration (in years) from diagnosis to death or the end of the study period. Of the initial 251 patients, 15 were excluded due to missing survival time or event status, resulting in a final cohort of 236 patients with complete clinical record.

Kaplan–Meier survival curves were generated to estimate and visualize survival probabilities over time for each molecular cluster (Kaplan & Meier, 1958). This non-parametric method accounts for right-censored observations and provides an intuitive visualization of differences in survival patterns. Differences between molecular clusters were formally tested using the log-rank test (Mantel, 1966), which compares observed and expected event counts under the null hypothesis of no difference in survival distributions.

While Kaplan–Meier survival curves and log-rank tests provide a useful overview of differences in survival distributions between groups, they do not adjust for covariates between different variables. Cox proportional hazards regression was applied to evaluate the independent prognostic value of molecular subtypes (Cox, 1972; Clark et al., 2003). Univariate Cox regression was used to assess the impact of each clinical variable on overall survival, and significant factors were subsequently included in multivariate models to test the independent prognostic value of cluster membership (Clark et al., 2003).

The Univariate model is defined as:

$$h(t \mid X) = h_0(t)\exp(\beta X)$$

where:

$h(t \mid X)$is the hazard at time $t$given covariates X,

$h_0(t)$is the baseline hazard,

$\beta$are regression coefficients.

For ER status and lymph node status, zero events in certain categories led to perfect separation issues. Two adjustment methods were applied and test: (1) complete removal of zero-event categories, and (2) merging uncertain categories with the negative group for conservative analysis.

In addition, two multivariate Cox proportional hazards models were constructed to assess the prognostic value of molecular subtypes while adjusting for clinical variables. The first model included all available clinical covariates regardless of univariate significance, while the second model included only variables that were significant in univariate analysis ($p < 0.05$). Model performance was compared using the likelihood ratio test and concordance index to evaluate the balance between model fit and parsimony. Hazard ratios with 95% confidence intervals were reported to quantify the magnitude and direction of the effect for each covariate

Multivariate Cox proportional hazards model is expressed as:

$$h(t \mid X) = h_0(t)\exp(\beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p)$$

where $\beta_1, \beta_2, \ldots, \beta_p$are regression coefficients corresponding to covariates $X_1, X_2, \ldots, X_p$..

Result Analysis

1. Data Pre-processing

The raw expression data exhibited moderate between-array variation (Figure 1, top panel), with most sample distributions showing similar central tendencies and interquartile ranges, though overall range varied substantially across samples. This pattern suggested that while systematic technical biases were minimal, scale differences existed that could confound downstream analyses. After applying between-array normalization using the scale method (Figure 1, bottom panel), sample distributions displayed more consistent medians and interquartile ranges across all 251 patients. Although some residual range variation remained, this normalization successfully standardized the data while preserving biological signal.

Figure 1. Comparison of raw expression distributions (top), between-array normalized expression distributions (bottom)

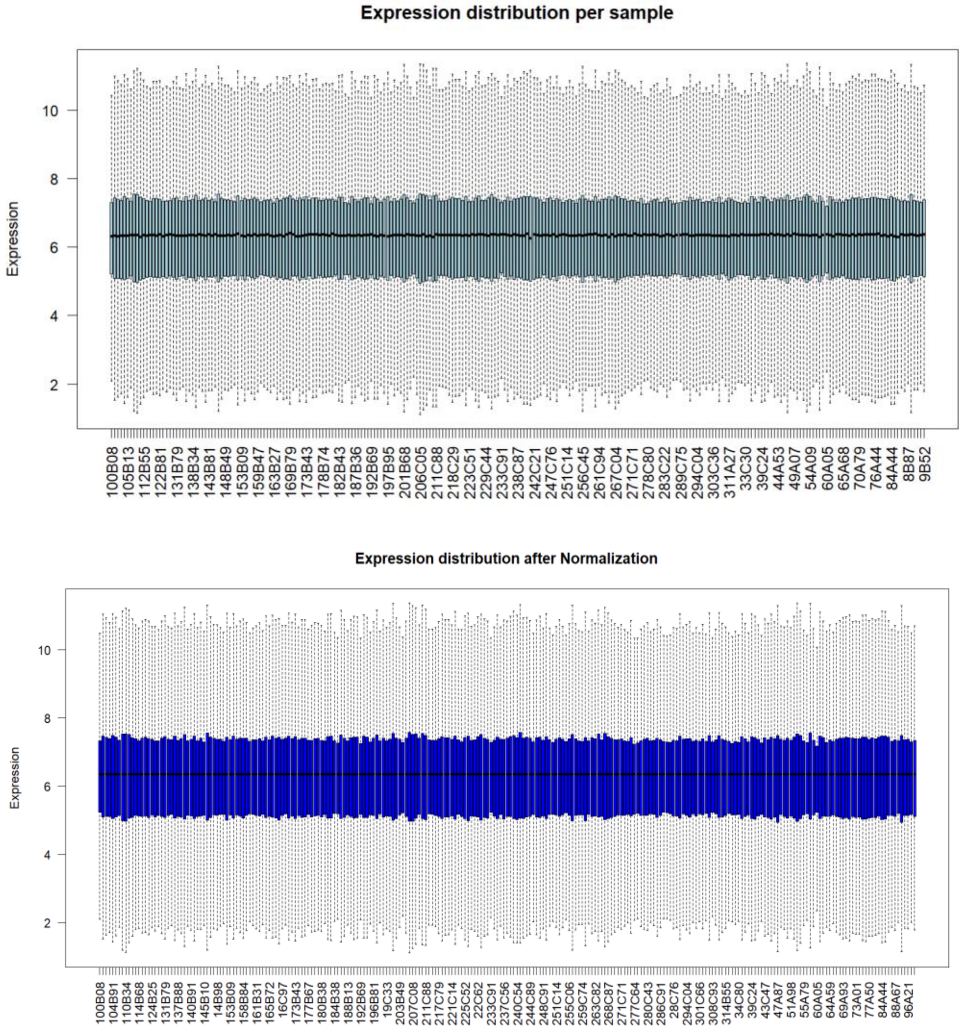
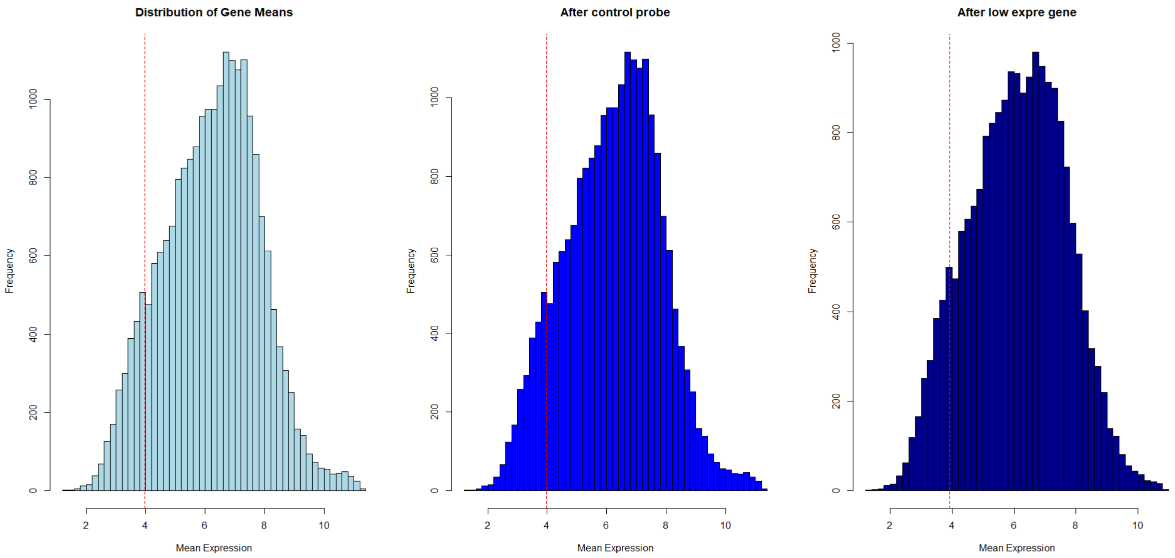


Figure 2 illustrates the progressive quality control applied to the gene expression data. The left panel presents the initial distribution of 22,283 genes, approximately normal around a mean expression of 6–7. The middle panel illustrates the dataset after removal of AFFX-prefixed control probes, retaining 22,215 biologically relevant genes while preserving the distribution shape. The right panel displays the final dataset used for clustering, where genes in the bottom 10% of variance and MAD were excluded, leaving 20,417 high-quality informative genes. This filtering effectively removes non-informative features while maintaining the underlying expression profile.

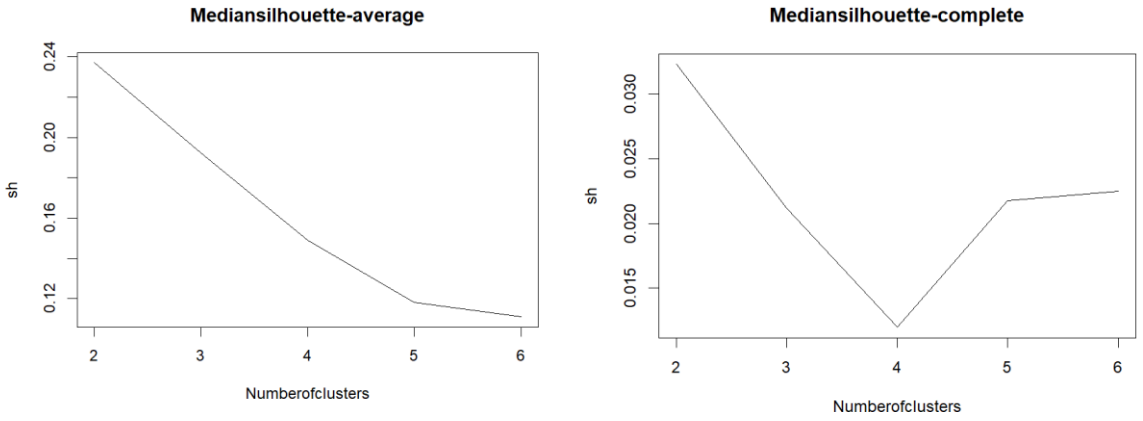
Figure 2. Gene Expression Histogram Comparison (raw expression data, expression after control probs removal, expression after low variance gene removal)



2. Clustering Analysis

Hierarchical clustering was performed using four linkage methods: average, complete, Ward.D, and Ward.D2. Silhouette analysis (Figure 3) showed that the average linkage method with two clusters yield the highest silhouette score of 0.24. However, further examination of the corresponding dendrogram (Figure 4) revealed that average linkage clustering results to one cluster containing one single outlier patient, while the other cluster contained the remaining 250 samples. This indicates that the high silhouette score was primarily driven by an outlier rather than reflecting meaningful statistical or biological separation. The Ward.D linkage method with two clusters achieved the second-highest average silhouette score. Dendrgam show that Ward.D method resulted in a more balanced partition, comprising 134 patients in Cluster 1 and 117 patients in Cluster 2. Therefore, the Ward.D clustering was selected for subsequent analyses.

Figure 3. Silhouette plots for different linkage methods



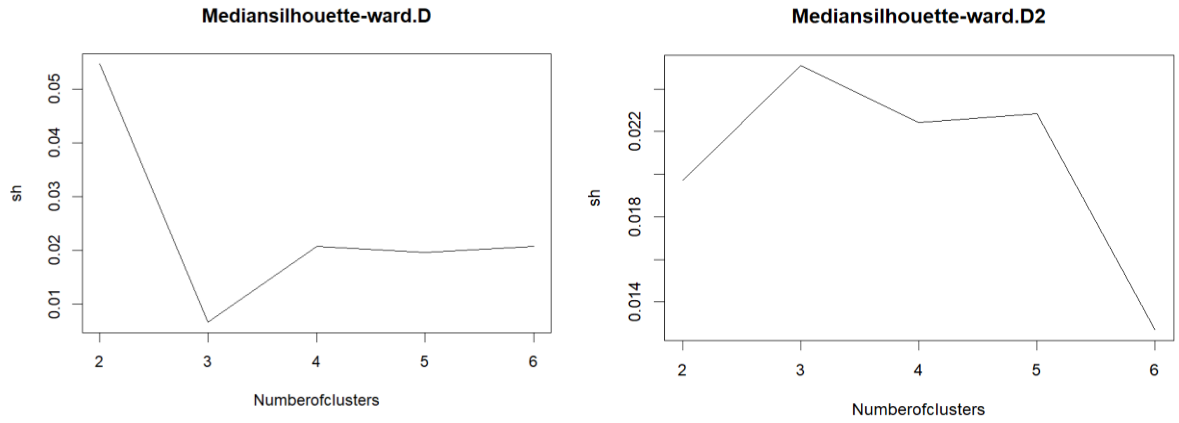
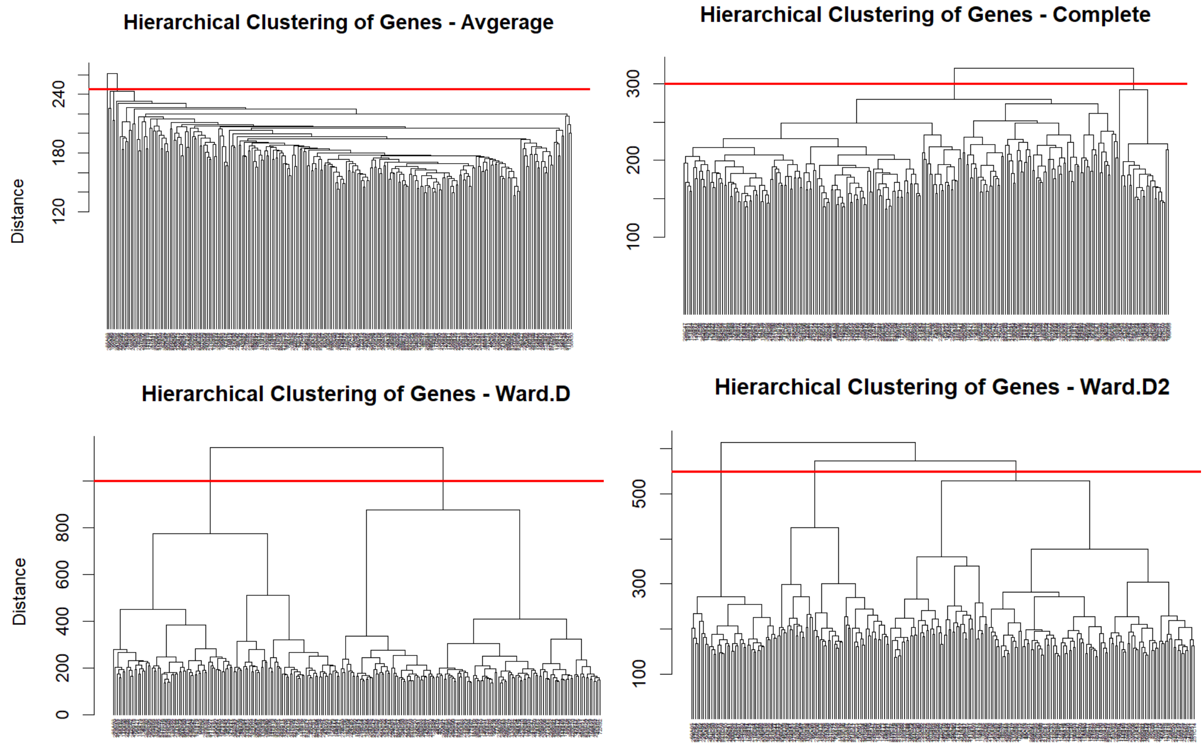


Figure 4. Hierarchical clustering dendrograms using Average, Complete, Ward.D and Ward.D2 methods



Principal Component Analysis (PCA) provided complementary visualization of cluster separation. The PCA plot (Figure 5) revealed relatively clear separation between Cluster 1 (black) and Cluster 2 (red), with the first two principal components captured 10.52% and 5.59% of total variance, 16.11% in total. Although modest, this is typical for high-dimensional gene expression data where variation is distributed across many components. The second plot (MDS using leading log FC) displays a similar clustering pattern, confirming the robustness of the hierarchical clustering.

Figure 5. PCA plot showing two distinct clusters based on PC 1 and PC2

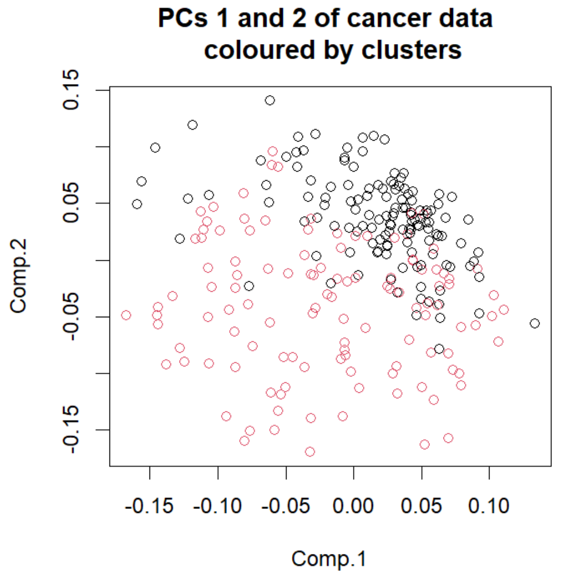
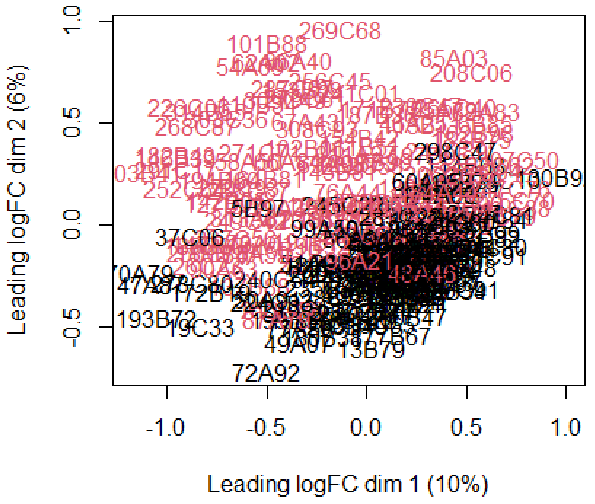
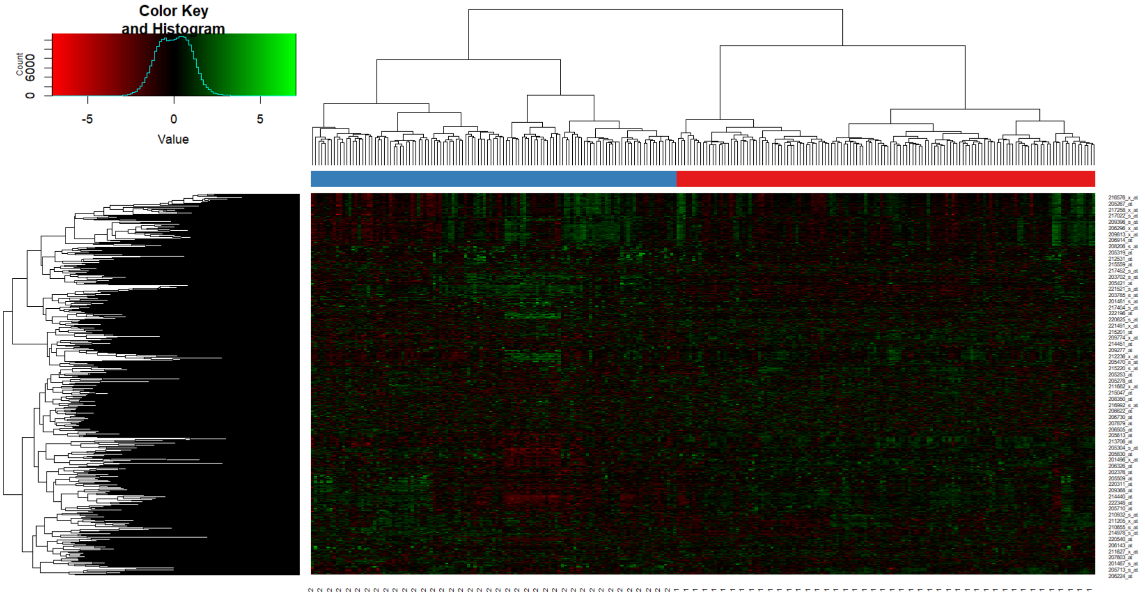


Figure 6. MDS plot (Leading logFC dimensions)



The heatmap of the top 1,000 most variable genes (Figure 7) provides detailed molecular characterization between clusters. Cluster 1 (right side, red dendrogram) exhibits distinctly under-regulated in the gene set listed on the top portion of the heatmap and over-regulated expression for genes in the bottom portion of the heatmap, whereas Cluster 2 (left side, blue dendrogram) shows the opposite pattern: comparatively high expression in the upper gene set and lower expression in the lower gene set. This heatmap demonstrates strong within-cluster homogeneity (samples within each cluster show similar expression patterns) and clear between-cluster heterogeneity (the two clusters show opposite expression patterns for key gene sets), supporting both the robustness and biological relevance of the clustering solution.

Figure 7. Heatmap of top 1,000 variable genes



3. Gene Expression Analysis

Differential expression analysis between the two clusters identified substantial molecular differences. Of 20,417 genes analyzed, 8,451 showed FDR-adjusted p-value < 0.05, and 6,548 showed FDR-adjusted p-value < 0.01, indicating widespread and robust differential expression between subtypes.

Figure 8 shows the *p*-value distributions. The raw *p*-values (left panel) exhibit a sharp peak near zero, reflecting many genes with strong signals before multiple testing correction, alongside a flat distribution for non-significant genes. After FDR adjustment, the peak is reduced but remains distinct, confirming that genuine biological signals persist beyond random variation. The remaining uniform distribution indicates that most genes are not differentially expressed, which is typical in genome-wide studies.

The volcano plot (Figure 9) illustrates the relationship between statistical significance and biological significant. Genes meeting both significance thresholds (FDR < 0.05) and biological relevance (|logFC| > 1) are highlighted in red. With these stringent criteria, 58 genes were identified as biologically significantly differentiated: 26 genes were upregulated by more than two-fold in Cluster 2 compared with Cluster 1, while 32 genes were downregulated in Cluster 2. These genes represent candidate markers that may distinguish the two molecular subtypes.

Figure 8. P-value distributions before (left) and after (right) FDR adjustment

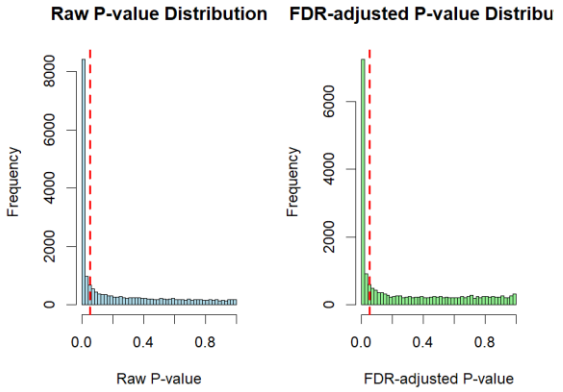
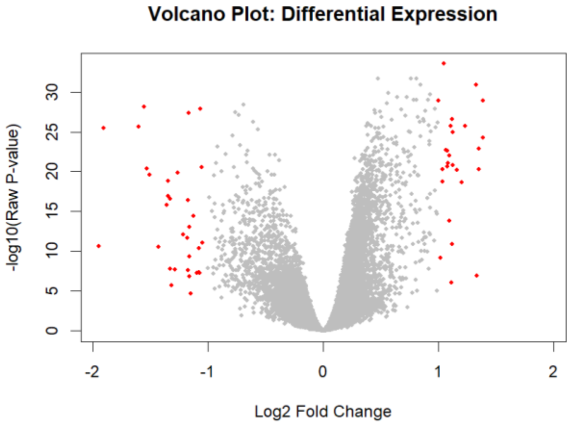


Figure 9. Volcano plot showing differential gene expression



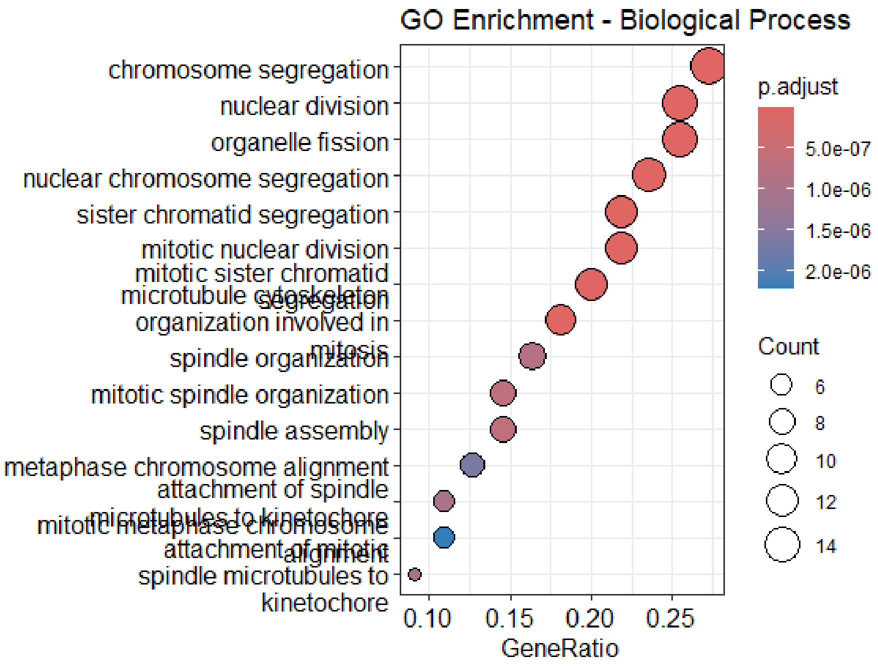
4. GO Enrichment Analyss

Gene Ontology enrichment analysis of the 58 biologically significant marker genes revealed a strong over-representation of genes involved in cell division and mitotic processes, particularly those related to chromosome segregation, nuclear division, and sister chromatid separation. Out of the 58 marker genes analysed, 55 were successfully mapped to entries in the Gene Ontology Biological Process database, and 15 of these were specifically linked to chromosome segregation, showing highly significant p-value. This substantial overlap indicates that cell division activity is a dominant feature in the group of samples studied.

The low adjusted p-values and high fold enrichment values provide strong statistical support for the biological relevance of the associated processes, suggesting that abnormal regulation of cell division and chromosome dynamics may play an important role in tumour development and progression in this cohort.

The dot plot below (Figure 10) summarizes the GO enrichment result. Each dot represents an enriched biological process, with size indicating gene count and color intensity reflecting statistical significance ($-\log_{10}$ *p*-value). The gene ratio highlights chromosome segregation as both the most significant process and the one involving the largest proportion of genes, underscoring its central role in this gene set.

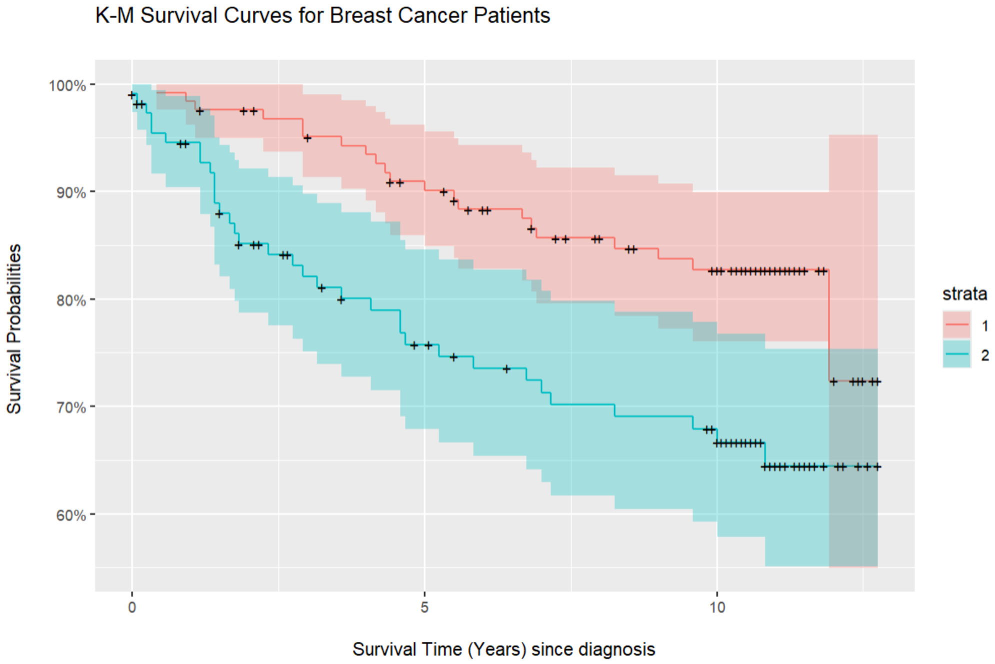
Figure 10. Gene Ontology enrichment analysis



5. Survival Analysis

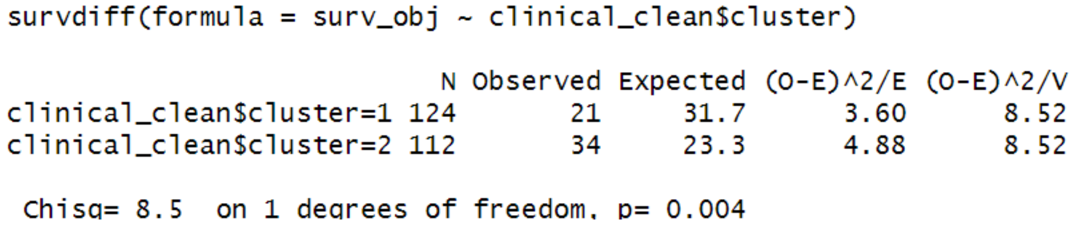
Kaplan-Meier survival curves were generated to provide an intuitive visualization of the survival experience in each cluster group (Figure 12). Cluster 1 (reference group) demonstrated superior survival, with more than 80% survival probability 10 years after diagnosis. In contrast, patients in Cluster 2 showed consistently lower probability of survival at each time point, with survival dropping to less than 70% at 10 years. These findings indicate that Cluster 2 represents a higher-risk group with poorer long-term survival outcomes.

Figure 11. Kaplan-Meier survival curves by cluster membership



Log-rank test result shows that Cluster 2 exhibited more observed events (34) than expected (23.3), whereas Cluster 1 showed fewer observed events (21) than expected (31.7). These numbers indicates that patients in Cluster 2 represent a higher-risk group in terms of survival. The difference in survival distributions was statistically significant (p = 0.004). These findings are consistent with the Kaplan–Meier survival curves presented above.

Table 1. Summary of log-rank test results



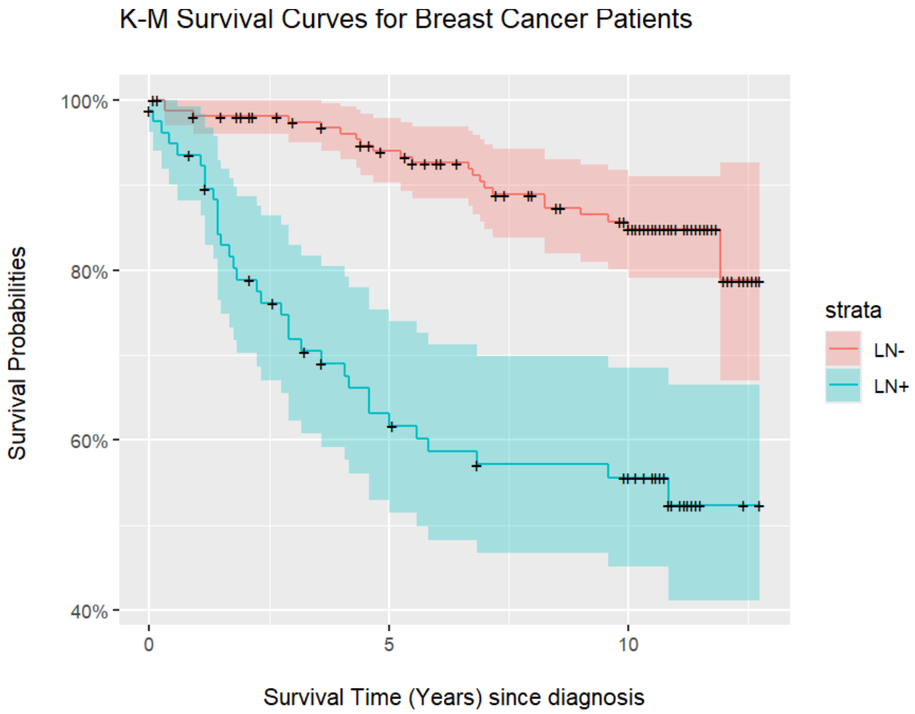
```
survdiff(formula = surv_obj ~ clinical_clean$cluster)

                           N Observed Expected (O-E)^2/E (O-E)^2/V
clinical_clean$cluster=1 124       21     31.7      3.60      8.52
clinical_clean$cluster=2 112       34     23.3      4.88      8.52

 Chisq= 8.5  on 1 degrees of freedom, p= 0.004
```

For clinical context, a Kaplan-Meier survival curves were also generated for lymph node status (Figure 12). Lymph node status showed even more dramatic survival separation in survival compared with molecular subtypes. Node-negative patients (LN−) maintained approximately 85% survival at 10 years, whereas node-positive patients (LN+) experienced a decline to below 60% survival. This highlights lymph node involvement as a strong prognostic factor in this cohort.

Figure 12. Kaplan-Meier survival curves by lymph node status,



As outlined in the methodology, two approaches were evaluated to address zero-event categories in ER and lymph node status. A comparison of univariate Cox regression results (Table 2) indicated that the merging method produced more stable and slightly stronger effect estimates, and therefore adopted for all subsequent multivariate analyses.

Table 2: Univariate Cox Proportional Hazards Regression Result Comparison

	Remove Method		***Merge Method***	
Variable	**HR (Univariate)**	**p-value**	**HR (Univariate)**	**p-value**
Cluster	***2.14***	***0.006***	***2.21***	***0.004***
Tumour Size (per mm)	***1.05***	***7.17e-07***	***1.05***	***6.03e-07***
Lymph Node Status (LN+ vs LN-)	***3.95***	***6.76e-07***	***4.15***	***2.72e-07***
Histologic Grade (G1 vs Reference)	*0.15*	*0.078*	*0.14*	*0.071*
Histologic Grade (G2 vs Reference)	*0.37*	*0.326*	*0.36*	*0.318*
Histologic Grade (G3 vs Reference)	*0.69*	*0.720*	*0.65*	*0.673*
PR Status (PR+ vs PR-)	*0.61*	*0.102*	*0.69*	*0.228*
ER Status (ER+ vs ER-)	*1.11*	*0.818*	*1.28*	*0.569*
Age	*1.00*	*0.985*	*1.00*	*0.788*

Univariate Cox regression identified three variables significantly associated with overall survival (Table 2). Cluster membership showed strong prognostic value (HR=2.21, p=0.004), indicating that patients in Cluster 2 had a 2.2-fold higher risk of death compared with those in Cluster 1. This result aligns with the finding in log-rank test and Kaplan-Meier Survival curve, reinforcing the clinical relevance of the molecular subtypes.

Tumour size was significantly associated with overall survival (HR = 1.05 per mm, p < 0.001), indicating a 5% increase in mortality risk for each additional millimeter in diameter. Positive lymph node status was the strongest univariate predictor (HR = 4.15, p < 0.001), conferring over a fourfold higher mortality risk. Other clinical factors (age, ER status, PR status, histological grade) were not significant, although grade G1 showed a trend toward better outcomes (HR = 0.14, *p* = 0.071).

To evaluate the independent prognostic contribution of molecular subtypes, two multivariate Cox models were compared (Table 3). The full model, which included all available 9 clinical covariates, achieved a slightly higher C-index but did not yield a statistically significant effect for cluster membership. This suggests that the inclusion of non-informative variables may have introduced noise, reducing model efficiency. In contrast, the reduced model incorporated only cluster membership, tumour size, and lymph node status (significant at p < 0.01) achieved comparable discrimination with a stronger likelihood ratio test result, indicating better model fit.

Table 3. Multivariate Cox Regression Models Comparison

Metric	Full Model (9 variables)	Selected Model (3 variables)
Concordance (C-index)	*0.771*	*0.759*
Likelihood Ratio Test	*p = 1e-06*	*p = 1e-08*
Number of Parameters	*9*	*3*
Cluster HR (p-value)	*1.65 (0.124)*	*1.76(0.044)*
Tumour Size HR (p-value)	*1.03(0.008)*	*1.03(0.005)*
LN Status HR (p-value)	*2.85(0.000717)*	*3.14(8.97e-05)*

In the reduced model, cluster membership remained statistically significant (HR = 1.76, *p* = 0.044), confirming its independent prognostic value beyond tumour size and lymph node status. This highlights that molecular subtype contributes unique prognostic information not captured by conventional clinical factors. As expected, hazard ratios for all predictors were lower in the multivariate model than in the univariate analysis (Table 4), reflecting adjustment for confounding and collinearity, typical in multivariate survival modelling.

Table 4. Univariate & Multivariate Cox Regression Result Comparison

Variable	HR (Univ.)	p (Univ.)	HR (Multi)	p (Multi)
Cluster	*2.21*	*0.004*	*1.76*	*0.044*
Tumour Size (mm)	*1.05*	*6.03e-07*	*1.03*	*0.005*
LN Status	*4.15*	*2.72e-07*	*3.14*	*8.97e-05*

Conclusion

This study identified two molecularly distinct breast cancer subtypes through unsupervised clustering of gene expression profiles from 251 patients. Hierarchical clustering using Ward’s linkage method revealed an optimal separation into Cluster 1 (n = 134) and Cluster 2 (n = 117), with relatively clear molecular distinctions confirmed by principal component analysis and heatmap. Differential expression analysis identified 58 genes meeting both statistical and biological significance criteria, and Gene Ontology enrichment highlighted dysregulation of cell division and chromosome dynamics as key molecular features distinguishing the two groups. The identification of mitotic genes as major contributors aligns with studies demonstrating that proliferation markers are among the strongest independent prognosticators, often overshadowing traditional histopathological features (Fen et al., 2006; Desmedt et al., 2007). Analysis result showed that patients in Cluster 2 experienced significantly poorer long-term survival compared with those in Cluster 1. Most critically, cluster membership remained an independent prognostic factor in multivariate Cox regression after adjustment for tumour size and lymph node status (HR=1.76, p=0.044). This independent prognostic value demonstrates that molecular classification captures biological risk not fully explained by classical clinical variables, a finding consistent with multiple studies showing gene expression-based classification provides information beyond conventional histopathological measures (Cardoso et al., 2016; Wu et al., 2017).

Overall, these findings underscore the prognostic utility of gene expression–based molecular classification in breast cancer. By capturing proliferative and mitotic dysregulation, molecular subtyping can improve risk stratification, enable earlier identification of high-risk patients, and guide more tailored therapeutic strategies. Future studies should validate these clusters in independent cohorts and evaluate whether targeting cell cycle pathways can improve outcomes in high-risk groups.

Reference

Australian Institute of Health and Welfare. (2024). *Cancer in Australia 2024*. https://www.aihw.gov.au/reports/cancer/cancer-in-australia-2024/summary

Benjamini, Y., & Hochberg, Y. (1995). Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Methodological), 57*(1), 289–300. https://doi.org/10.1111/j.2517-6161.1995.tb02031.x

Bolstad, B. M., Irizarry, R. A., Åstrand, M., & Speed, T. P. (2003). A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, *19*(2), 185–193. https://doi.org/10.1093/bioinformatics/19.2.185

Bourgon, R., Gentleman, R., & Huber, W. (2010). Independent filtering increases detection power for high-throughput experiments. *Proceedings of the National Academy of Sciences*, *107*(21), 9546–9551. https://doi.org/10.1073/pnas.0914005107

Cardoso, F., van't Veer, L. J., Bogaerts, J., Slaets, L., Viale, C., Russo, L., Negrao, C. E., Leyland-Jones, B., Varughese, M., Bourgeois, H., Marc Aft, L. M., Sgroi, D., Ravdin, P., Peloso, H., M T A A H M, E. M., Stork-Sloots, L., Piccart, M. J., & MINDACT Investigators. (2016). 70-Gene Signature as an Aid to Treatment Decisions in Early-Stage Breast Cancer. *The New England Journal of Medicine, 375(8), 717–729. https://doi.org/10.1056/NEJMoa1602253*

Clark, T. G., Bradburn, M. J., Love, S. B., & Altman, D. G. (2003). Survival analysis part II: Multivariate data analysis—An introduction to concepts and methods. *British Journal of Cancer*, *89*(3), 431–436. https://doi.org/10.1038/sj.bjc.6601119

Cox, D. R. (1972). Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological), 34*(2), 187–202. https://doi.org/10.1111/j.2517-6161.1972.tb00899.x

Desmedt, C., Piette, F., Loi, S., Wang, Y., Lallemand, F., Haibe-Kains, B., Viale, G., Delaloge, S., Turletti, I., Vincent-Salomon, A., Piccart, M., Glas, A. M., Lobbens, E., Sotiriou, C., & Rody, A. (2007). Strong time dependence of the 70-gene signature prognostic power in breast cancer. *Journal of the National Cancer Institute, 99(9), 682–695. https://doi.org/10.1093/jnci/djj214*

Eisen, M. B., Spellman, P. T., Brown, P. O., & Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, *95*(25), 14863–14868. https://doi.org/10.1073/pnas.95.25.14863

Fan, C., Oh, D. S., Wessels, L., Weigelt, B., Michiels, S., Schuck, K., Carey, L. A., van't Veer, L. J., & Perou, C. M. (2006). Concordance of gene expression profiles between formalin-fixed, paraffin-embedded and fresh-frozen breast tumor tissues. *BMC Genomics, 7(1), 21. https://doi.org/10.1186/1471-2164-7-21*

Harrell, F. E., Jr., Lee, K. L., & Mark, D. B. (1996). Multivariable prognostic models: Issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. *Statistics in Medicine*, *15*(4), 361–387. https://doi.org/10.1002/(SICI)1097-0258(19960229)15:4<361::AID-SIM168>3.0.CO;2-4

Heinze, G., & Schemper, M. (2002). A solution to the problem of separation in logistic regression. *Statistics in Medicine*, *21*(16), 2409–2419. https://doi.org/10.1002/sim.1047

Irizarry, R. A., Hobbs, B., Collin, F., Beazer-Barclay, Y. D., Antonellis, K. J., Scherf, U., & Speed, T. P. (2003). Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics*, *4*(2), 249–264. https://doi.org/10.1093/biostatistics/4.2.249

Jolliffe, I. T. (2002). *Principal component analysis* (2nd ed.). Springer. https://doi.org/10.1007/b98835

Kaplan, E. L., & Meier, P. (1958). Nonparametric estimation from incomplete observations. *Journal of the American Statistical Association*, *53*(282), 457–481. https://doi.org/10.1080/01621459.1958.10501452

Kassambara, A., Kosinski, M., & Biecek, P. (2021). *survminer: Drawing survival curves using 'ggplot2'* (R package version 0.4.9). https://CRAN.R-project.org/package=survminer

Maechler, M., Rousseeuw, P., Struyf, A., Hubert, M., & Hornik, K. (2022). *cluster: Cluster analysis basics and extensions* (R package version 2.1.4). https://CRAN.R-project.org/package=cluster

Murtagh, F., & Legendre, P. (2014). Ward's hierarchical agglomerative clustering method: Which algorithms implement Ward's criterion? *Journal of Classification*, *31*(3), 274–295. https://doi.org/10.1007/s00357-014-9161-z

Perou, C. M., Sørlie, T., Eisen, M. B., van de Rijn, M., Jeffrey, S. S., Rees, C. A., Pollack, J. R., Ross, D. T., Johnsen, H., Akslen, L. A., Fluge, Ø., Pergamenschikov, A., Williams, C., Zhu, S. X., Lønning, P. E., Børresen-Dale, A. L., Brown, P. O., & Botstein, D. (2000). Molecular portraits of human breast tumours. *Nature*, *406*(6797), 747–752. https://doi.org/10.1038/35021093

R Core Team. (2023). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. https://www.R-project.org/

Reis-Filho, J. S., & Pusztai, L. (2011). Gene expression profiling in breast cancer: Classification, prognostication, and prediction. *The Lancet*, *378*(9805), 1812–1823. https://doi.org/10.1016/S0140-6736(11)61539-0

Ritchie, M. E., Phipson, B., Wu, D., Hu, Y., Law, C. W., Shi, W., & Smyth, G. K. (2015). limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Research*, *43*(7), e47. https://doi.org/10.1093/nar/gkv007

Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, *20*, 53–65. https://doi.org/10.1016/0377-0427(87)90125-7

Smyth, G. K. (2004). Linear models and empirical Bayes methods for assessing differential expression in microarray experiments. *Statistical Applications in Genetics and Molecular Biology*, *3*(1), Article 3. https://doi.org/10.2202/1544-6115.1027

Soerjomataram, I., Louwman, M. W., Ribot, J. G., Roukema, J. A., & Coebergh, J. W. (2008). An overview of prognostic factors for long-term survivors of breast cancer. *Breast Cancer Research and Treatment*, *107*(3), 309–330. https://doi.org/10.1007/s10549-007-9556-1

Sorlie, T., Perou, C. M., Tibshirani, R., Aas, T., Geisler, S., Johnsen, H., Hastie, T., Eisen, M. B., van de Rijn, M., Jeffrey, S. S., Thorsen, T., Quist, H., Matese, J. C., Brown, P. O., Botstein, D., Lønning, P. E., & Børresen-Dale, A. L. (2001). Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications. *Proceedings of the National Academy of Sciences*, *98*(19), 10869–10874. https://www.pnas.org/doi/10.1073/pnas.191367098

Therneau, T. M. (2023). *survival: Survival analysis* (R package version 3.5-5). https://CRAN.R-project.org/package=survival

Ward, J. H., Jr. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, *58*(301), 236–244. https://doi.org/10.1080/01621459.1963.10500845

Wu, M., Zhao, L., Zhao, J., Zhang, F., Zhang, H., Liu, F., & Chen, J. (2017). Molecular subtypes and risk stratification of breast cancer based on gene expression profiling: A single center study. *Oncology Letters, 13(6), 4057–4064. https://doi.org/10.3892/ol.2017.5954*

Yu, G., Wang, L. G., Han, Y., & He, Q. Y. (2012). clusterProfiler: An R package for comparing biological themes among gene clusters. *OMICS: A Journal of Integrative Biology*, *16*(5), 284–287. https://doi.org/10.1089/omi.2011.0118

Appendix

Appendix A: Univariate Analysis with Zero Event Category removed

```
Cox Result for cluster :
Call:
coxph(formula = formula, data = data)

  n= 224, number of events= 55

          coef exp(coef) se(coef)     z Pr(>|z|)
cluster 0.7592    2.1366   0.2779 2.732  0.00629 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

        exp(coef) exp(-coef) lower .95 upper .95
cluster     2.137      0.468     1.239     3.683

Concordance= 0.605  (se = 0.033 )
Likelihood ratio test= 7.74  on 1 df,   p=0.005
Wald test            = 7.47  on 1 df,   p=0.006
Score (logrank) test = 7.83  on 1 df,   p=0.005


Cox Result for histgrade :
Call:
coxph(formula = formula, data = data)

  n= 224, number of events= 55

                       coef exp(coef) se(coef)      z Pr(>|z|)
factor(histgrade)G1 -1.8867    0.1516   1.0719 -1.760   0.0784 .
factor(histgrade)G2 -1.0009    0.3675   1.0189 -0.982   0.3259
factor(histgrade)G3 -0.3699    0.6908   1.0305 -0.359   0.7196
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

                    exp(coef) exp(-coef) lower .95 upper .95
factor(histgrade)G1    0.1516      6.597   0.01855     1.239
factor(histgrade)G2    0.3675      2.721   0.04989     2.708
factor(histgrade)G3    0.6908      1.448   0.09167     5.206

Concordance= 0.637  (se = 0.032 )
Likelihood ratio test= 13.7  on 3 df,   p=0.003
Wald test            = 12.61  on 3 df,   p=0.006
Score (logrank) test = 14.12  on 3 df,   p=0.003


Cox Result for ERstatus :
Call:
coxph(formula = formula, data = data)

  n= 224, number of events= 55

                      coef exp(coef) se(coef)     z Pr(>|z|)
factor(ERstatus)ER+ 0.0998    1.1050   0.4326 0.231    0.818

                    exp(coef) exp(-coef) lower .95 upper .95
factor(ERstatus)ER+     1.105      0.905    0.4732      2.58

Concordance= 0.502  (se = 0.023 )
Likelihood ratio test= 0.05  on 1 df,   p=0.8
Wald test            = 0.05  on 1 df,   p=0.8
Score (logrank) test = 0.05  on 1 df,   p=0.8


Cox Result for PRstatus :
Call:
```

```
coxph(formula = formula, data = data)

  n= 224, number of events= 55 

                        coef exp(coef) se(coef)      z Pr(>|z|)
factor(PRstatus)PgR+ -0.4955    0.6093   0.3032 -1.634    0.102

                     exp(coef) exp(-coef) lower .95 upper .95
factor(PRstatus)PgR+    0.6093      1.641    0.3363     1.104

Concordance= 0.551  (se = 0.031 )
Likelihood ratio test= 2.46  on 1 df,   p=0.1
Wald test            = 2.67  on 1 df,   p=0.1
Score (logrank) test = 2.73  on 1 df,   p=0.1


Cox Result for age :
Call:
coxph(formula = formula, data = data)

  n= 224, number of events= 55 

         coef exp(coef)  se(coef)     z Pr(>|z|)
age 0.0001966 1.0001966 0.0103653 0.019    0.985

    exp(coef) exp(-coef) lower .95 upper .95
age         1     0.9998    0.9801     1.021

Concordance= 0.512  (se = 0.043 )
Likelihood ratio test= 0  on 1 df,   p=1
Wald test            = 0  on 1 df,   p=1
Score (logrank) test = 0  on 1 df,   p=1


Cox Result for tumor_size_mm :
Call:
coxph(formula = formula, data = data)

  n= 224, number of events= 55 

                  coef exp(coef) se(coef)     z Pr(>|z|)    
tumor_size_mm 0.049511  1.050757 0.009989 4.957 7.17e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

              exp(coef) exp(-coef) lower .95 upper .95
tumor_size_mm     1.051     0.9517      1.03     1.072

Concordance= 0.71  (se = 0.036 )
Likelihood ratio test= 19.59  on 1 df,   p=1e-05
Wald test            = 24.57  on 1 df,   p=7e-07
Score (logrank) test = 25.84  on 1 df,   p=4e-07


Cox Result for LNstatus :
Call:
coxph(formula = formula, data = data)

  n= 224, number of events= 55 

                      coef exp(coef) se(coef)     z Pr(>|z|)    
factor(LNstatus)LN+ 1.3748    3.9543   0.2767 4.968 6.76e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

                    exp(coef) exp(-coef) lower .95 upper .95
factor(LNstatus)LN+     3.954     0.2529     2.299     6.802

Concordance= 0.677  (se = 0.031 )
```

```
Likelihood ratio test= 25.07  on 1 df,   p=6e-07
Wald test            = 24.68  on 1 df,   p=7e-07
Score (logrank) test = 28.7  on 1 df,   p=8e-08
```

Appendix B: Univariate Models Results with Zero Event Category Merged

```
Cox Result for cluster :
Call:
coxph(formula = formula, data = data)

  n= 236, number of events= 55 

          coef exp(coef) se(coef)     z Pr(>|z|)   
cluster 0.7911    2.2057   0.2779 2.846  0.00442 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

        exp(coef) exp(-coef) lower .95 upper .95
cluster     2.206     0.4534     1.279     3.803

Concordance= 0.609  (se = 0.032 )
Likelihood ratio test= 8.41  on 1 df,   p=0.004
Wald test            = 8.1  on 1 df,   p=0.004
Score (logrank) test = 8.53  on 1 df,   p=0.003
```

```
Cox Result for histgrade :
Call:
coxph(formula = formula, data = data)

  n= 236, number of events= 55 

                       coef exp(coef) se(coef)      z Pr(>|z|)  
factor(histgrade)G1 -1.9362    0.1443   1.0717 -1.807   0.0708 .
factor(histgrade)G2 -1.0176    0.3614   1.0188 -0.999   0.3179  
factor(histgrade)G3 -0.4353    0.6470   1.0306 -0.422   0.6727  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

                    exp(coef) exp(-coef) lower .95 upper .95
factor(histgrade)G1    0.1443      6.932   0.01766     1.179
factor(histgrade)G2    0.3614      2.767   0.04907     2.662
factor(histgrade)G3    0.6470      1.545   0.08585     4.877

Concordance= 0.637  (se = 0.032 )
Likelihood ratio test= 13.59  on 3 df,   p=0.004
Wald test            = 12.26  on 3 df,   p=0.007
Score (logrank) test = 13.75  on 3 df,   p=0.003
```

```
Cox Result for ERstatus :
Call:
coxph(formula = formula, data = data)

  n= 236, number of events= 55 

                      coef exp(coef) se(coef)    z Pr(>|z|)
factor(ERstatus)ER+ 0.2465    1.2795   0.4326 0.57    0.569

                    exp(coef) exp(-coef) lower .95 upper .95
factor(ERstatus)ER+      1.28     0.7815     0.548     2.987

Concordance= 0.51  (se = 0.023 )
Likelihood ratio test= 0.35  on 1 df,   p=0.6
Wald test            = 0.32  on 1 df,   p=0.6
Score (logrank) test = 0.33  on 1 df,   p=0.6
```

```
Cox Result for PRstatus :
Call:
coxph(formula = formula, data = data)

  n= 236, number of events= 55 

                        coef exp(coef) se(coef)      z Pr(>|z|)
factor(PRstatus)PgR+ -0.3656    0.6937   0.3031 -1.206    0.228

                     exp(coef) exp(-coef) lower .95 upper .95
factor(PRstatus)PgR+    0.6937      1.441     0.383     1.257

Concordance= 0.541  (se = 0.031 )
Likelihood ratio test= 1.37  on 1 df,   p=0.2
Wald test            = 1.45  on 1 df,   p=0.2
Score (logrank) test = 1.47  on 1 df,   p=0.2


Cox Result for age :
Call:
coxph(formula = formula, data = data)

  n= 236, number of events= 55 

         coef exp(coef)  se(coef)      z Pr(>|z|)
age -0.002733  0.997271  0.010171 -0.269    0.788

    exp(coef) exp(-coef) lower .95 upper .95
age    0.9973      1.003    0.9776     1.017

Concordance= 0.498  (se = 0.043 )
Likelihood ratio test= 0.07  on 1 df,   p=0.8
Wald test            = 0.07  on 1 df,   p=0.8
Score (logrank) test = 0.07  on 1 df,   p=0.8


Cox Result for tumor_size_mm :
Call:
coxph(formula = formula, data = data)

  n= 236, number of events= 55 

                 coef exp(coef) se(coef)    z Pr(>|z|)    
tumor_size_mm 0.05022   1.05150  0.01006 4.99 6.03e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

              exp(coef) exp(-coef) lower .95 upper .95
tumor_size_mm     1.052      0.951     1.031     1.072

Concordance= 0.708  (se = 0.036 )
Likelihood ratio test= 19.8  on 1 df,   p=9e-06
Wald test            = 24.9  on 1 df,   p=6e-07
Score (logrank) test = 26.14  on 1 df,   p=3e-07


Cox Result for LNstatus :
Call:
coxph(formula = formula, data = data)

  n= 236, number of events= 55 

                      coef exp(coef) se(coef)     z Pr(>|z|)    
factor(LNstatus)LN+ 1.4222    4.1462   0.2766 5.142 2.72e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

                    exp(coef) exp(-coef) lower .95 upper .95
```

```
factor(LNstatus)LN+     4.146     0.2412     2.411      7.13

Concordance= 0.683  (se = 0.031 )
Likelihood ratio test= 26.8  on 1 df,   p=2e-07
Wald test            = 26.44  on 1 df,   p=3e-07
Score (logrank) test = 31.08  on 1 df,   p=2e-08
```

Appendix C: Multivariate Model Results (Full Variables Model)

```
Call:
coxph(formula = surv_obj_m2 ~ cluster + histgrade + ERstatus + 
    PRstatus + age + tumor_size_mm + LNstatus, data = clinical_clean)

  n= 236, number of events= 55 

                   coef exp(coef)  se(coef)      z Pr(>|z|)    
cluster        0.501128  1.650582  0.325433  1.540 0.123590    
histgradeG1   -1.793160  0.166433  1.087956 -1.648 0.099313 .  
histgradeG2   -1.382427  0.250969  1.044958 -1.323 0.185852    
histgradeG3   -1.483660  0.226806  1.132400 -1.310 0.190131    
ERstatusER+    0.734152  2.083714  0.549054  1.337 0.181183    
PRstatusPgR+  -0.481981  0.617559  0.448993 -1.073 0.283060    
age            0.002786  1.002789  0.010427  0.267 0.789351    
tumor_size_mm  0.032720  1.033261  0.012292  2.662 0.007772 ** 
LNstatusLN+    1.046290  2.847069  0.309296  3.383 0.000717 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

              exp(coef) exp(-coef) lower .95 upper .95
cluster          1.6506     0.6058   0.87222     3.124
histgradeG1      0.1664     6.0084   0.01973     1.404
histgradeG2      0.2510     3.9846   0.03237     1.946
histgradeG3      0.2268     4.4091   0.02465     2.087
ERstatusER+      2.0837     0.4799   0.71037     6.112
PRstatusPgR+     0.6176     1.6193   0.25615     1.489
age              1.0028     0.9972   0.98250     1.023
tumor_size_mm    1.0333     0.9678   1.00866     1.058
LNstatusLN+      2.8471     0.3512   1.55283     5.220

Concordance= 0.771  (se = 0.031 )
Likelihood ratio test= 44.34  on 9 df,   p=1e-06
Wald test            = 45.43  on 9 df,   p=8e-07
Score (logrank) test = 52.96  on 9 df,   p=3e-08
```

Appendix D: Multivariate Model Results (Selected Variables Model)

```
Call:
coxph(formula = surv_obj_m2 ~ cluster + tumor_size_mm + LNstatus, 
    data = clinical_clean)

  n= 236, number of events= 55 

                 coef exp(coef) se(coef)     z Pr(>|z|)    
cluster       0.56751   1.76386  0.28235 2.010  0.04444 *  
tumor_size_mm 0.03249   1.03303  0.01165 2.790  0.00527 ** 
LNstatusLN+   1.14357   3.13795  0.29196 3.917 8.97e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

              exp(coef) exp(-coef) lower .95 upper .95
cluster           1.764     0.5669     1.014     3.068
tumor_size_mm     1.033     0.9680     1.010     1.057
LNstatusLN+       3.138     0.3187     1.771     5.561

Concordance= 0.759  (se = 0.033 )
```

```
Likelihood ratio test= 39.91  on 3 df,   p=1e-08
Wald test            = 40.93  on 3 df,   p=7e-09
Score (logrank) test = 46.68  on 3 df,   p=4e-10
Score (logrank) test = 44.09  on 3 df,   p=1e-09
```

Appendix E: R Code for All Analysis

```
library(reshape2)
library(limma)
library(cluster)
library(gplots)
library(RColorBrewer)
library(matrixStats)
library(clusterProfiler)
library(org.Hs.eg.db)
library(survival)
#library(ggplot2)

# Read the data files
annotations <- readRDS("STA5MB_2025_BC_annotations.RDS")
clinical <- readRDS("STA5MB_2025_BC_clinical_data.RDS")
expression <- readRDS("STA5MB_2025_BC_expression_data.RDS")

# Basic data exploration
cat("annotations class:", class(annotations), "\n")
cat("clinical class:", class(clinical), "\n")
cat("expr_raw class:", class(expression), "\n\n")

cat("Dataset Dimensions:\n")
cat("Annotations:", dim(annotations), "\n")
cat("Clinical data:", dim(clinical), "\n")
cat("Expression data:", dim(expression), "\n")

# Check each dataset structure
cat("\nData Structure:\n")
str(annotations)
str(clinical)
#str(expression)

summary(annotations)
#is.na(annotations)
summary(clinical)
#summary(expression)
head(expression)

cat("# of missing value in Expression:", sum(rowSums(is.na(expression))),
"\n")
cat("# of missing value in Clinical:", sum(colSums(is.na(clinical))), "\n")
# 30 Null value, Prepare to clean for Survival Analysis
```

```
cat("# of missing value in Annotation:", sum(colSums(is.na(annotations))),
"\n")

# Check clinical data & expression match with "sampleID"
setequal(clinical$sampleID, colnames(expression))

# Re-order clinical data to match with the order in expression
clinical <- clinical[match(colnames(expression), clinical$sampleID),]
# common_ID <- intersect(clinical$sampleID, colnames(expression))

boxplot(expression, outline = FALSE, main = "Expression distribution per
sample",
        ylab = "Expression", xlab = "", las = 2, col = "lightblue")

#hist(as.numeric(expression), breaks = 100, main = "Overall Expression Value
Distribution",
#     xlab = "Expression (log2)", col = "skyblue")


expr_norm <- normalizeBetweenArrays(expression, method = "scale")
boxplot(expr_norm, outline = FALSE, main = "Expression distribution after
Normalization",
        ylab = "Expression", xlab = "", las = 2, col = "blue")

#------------------------------------------
# plot avg gene expression distribution
#------------------------------------------

# Calculate gene expression statistics
gene_means <- rowMeans(expr_norm)
gene_medians <- apply(expr_norm, 1, median)

# Plot the distribution
par(mfrow = c(1,3 ))

# Histogram of mean expression
hist(gene_means, breaks = 50, main = "Distribution of Gene Means",
     xlab = "Mean Expression", col = "lightblue")
abline(v = quantile(gene_means, c(0.10)), col = "red", lty = 2)
# 10% is about 4, which is reasonable cut off line for low expr

cat("Initial dimensions:", dim(expr_norm), "\n")

#-----------------------------------
# Remove control probes
#-----------------------------------
control_probes <- grep("^AFFX", rownames(expr_norm))
if(length(control_probes) > 0) {
  expr_filtered <- expr_norm[-control_probes, ]
```

```r
  cat("Removed", length(control_probes), "control probes\n")
} else {
  expr_filtered <- expr_norm
  cat("No AFFX control probes found\n")
}
gene_filter_means <- rowMeans(expr_filtered)

# Histogram of mean expression after remove control probes
hist(gene_filter_means, breaks = 50, main = "After control probe",
     xlab = "Mean Expression", col = "blue")
abline(v = quantile(gene_filter_means, c(0.10)), col = "red", lty = 2)

cat("Dimensions after control probes removed:", dim(expr_filtered), "\n")

#--------------------------------
# Remove lowe expr gene
#--------------------------------
#  Using variance
#gene_variance <- apply(expr_filtered, 1, var)
gene_variance <- rowVars(expr_filtered)
variance_threshold <- quantile(gene_variance, probs = 0.10)  # Bottom 10%

# Using Median Absolute Deviation
gene_mad <- rowMads(expr_filtered) #apply(expr_filtered, 1, mad)  # MAD =
median absolute deviation
mad_threshold <- quantile(gene_mad, probs = 0.10)  # Bottom 10%
#gene_means <- rowMeans(expr_filtered)
#mean_threshold <- quantile(gene_means, 0.20)  # Remove bottom 20%
#gene_means > mean_threshold
normal_genes <- gene_variance > variance_threshold | gene_mad > mad_threshold
expr_final <- expr_filtered[normal_genes, ]

gene_final_means <- rowMeans(expr_final)
# Histogram of mean expression after remove low expr
hist(gene_final_means, breaks = 50, main = "After low expre gene",
     xlab = "Mean Expression", col = "darkblue")
abline(v = quantile(gene_final_means, c(0.10)), col = "red", lty = 2)

cat("Dimensions after control probes removed:", dim(expr_final), "\n")

# ------------------------------------------
# Clustering
# ------------------------------------------

# Scale expression data
scaled.E <- t(scale(t(expr_final), center = TRUE))
# calculate distance
dist <- dist(t(scaled.E))
```

```r
# **** complete ****
# Perform hierarchical clustering with complete linkage.
hc <- hclust(dist, method = "complete")
K<-2:6
sh<-NULL
for(i in K) {
  sh<-c(sh,median(silhouette(cutree(hc,k=i),dist=dist)[,3],na.rm=T))
}
par(mfrow = c(1,1 ))
#Plotsilhouette
plot(K,sh,type="l",main="Mediansilhouette-complete",xlab="Numberofclusters")

# **** average ****
# Perform hierarchical clustering with average linkage.
hc_avg <- hclust(dist, method = "average")
K<-2:6
sh<-NULL
for(i in K) {
  sh<-c(sh,median(silhouette(cutree(hc_avg,k=i),dist=dist)[,3],na.rm=T))
}
#Plotsilhouette
plot(K,sh,type="l",main="Mediansilhouette-average",xlab="Numberofclusters")

# **** ward.D ****
# Perform hierarchical clustering with ward.D linkage.
hc_w <- hclust(dist, method = "ward.D")
K<-2:6
sh<-NULL
for(i in K) {
  sh<-c(sh,median(silhouette(cutree(hc_w,k=i),dist=dist)[,3],na.rm=T))
}
#Plotsilhouette
plot(K,sh,type="l",main="Mediansilhouette-ward.D",xlab="Numberofclusters")

# **** ward.D2 ****
# Perform hierarchical clustering with ward.D2 linkage.
hc_w2 <- hclust(dist, method = "ward.D2")
K<-2:6
sh<-NULL
for(i in K) {
  sh<-c(sh,median(silhouette(cutree(hc_w2,k=i),dist=dist)[,3],na.rm=T))
}
#Plotsilhouette
plot(K,sh,type="l",main="Mediansilhouette-ward.D2",xlab="Numberofclusters")

#------------------------
# ploting dendrogram
#------------------------
```

```r
plot(hc, main = "Hierarchical Clustering of Genes - Complete",
     xlab = "Genes", ylab = "Distance", cex = 0.3, hang = -1)
# Add cutoff line
abline(h = 300, col = "red", lwd = 2, )

plot(hc_avg, main = "Hierarchical Clustering of Genes - Avgerage",
     xlab = "Genes", ylab = "Distance", cex = 0.3, hang = -1)
# Add cutoff line
abline(h = 245, col = "red", lwd = 2, )

plot(hc_w, main = "Hierarchical Clustering of Genes - Ward.D",
     xlab = "Genes", ylab = "Distance", cex = 0.3, hang = -1)
# Add cutoff line
abline(h = 1000, col = "red", lwd = 2, )

plot(hc_w2, main = "Hierarchical Clustering of Genes - Ward.D2",
     xlab = "Genes", ylab = "Distance", cex = 0.3, hang = -1)
# Add cutoff line
abline(h = 550, col = "red", lwd = 2, )

#select optimal clusters for ward.D
#cl<-cutree(hc_w,k=K[1])
# ward.D has the highest distance
#select optimal clusters for Average
cl<-cutree(hc_w,k=K[1])
# ward.D has the highest distance
table(cl)
# ------------------------------------------
# create Heatmap with high varaince gene
# ------------------------------------------
# Select high variance genes
rv <- rowVars(expr_final)
idx <- order(-rv)[1:1000]
# Specify colour palette
cols <- brewer.pal(length(unique(cl)), "Set1")
# Specify colour palette
#cols <- colors()[seq(8, length(colors()), len = length(unique(cl)))]

# Produce heatmap
heatmap.2(scaled.E[idx, ], labCol = cl, trace = "none", ColSideColors =
cols[cl],
          col = redgreen(100), Colv = as.dendrogram(hc_w))


# ------------------------------------------
# PCA Visualization for 2 Clusters
# ------------------------------------------

idx <- order(-rv)[1:1000]
```

```r
# Specify colour palette
cols <- brewer.pal(length(unique(cl)), "Set1")

# Extract the data for PCA (use the SCALED data for visualization)
par(bg = "white")
pc <- princomp(scaled.E[idx, ])
summary(pc)

# Or extract specific values:
variance_explained <- pc$sdev^2 / sum(pc$sdev^2) * 100
cat("PC1 explains:", round(variance_explained[1], 2), "%\n")
cat("PC2 explains:", round(variance_explained[2], 2), "%\n")
plot(pc$load[, 1:2], col = cl)
title("PCs 1 and 2 of cancer data \n coloured by clusters")

# Create MDS plot, colour coded by cluster
plotMDS(scaled.E[idx, ], col = as.numeric(cl))

# ------------------------------------------
# Gene Expression Analysis
# ------------------------------------------

#Create design matrix based on clusters
design <- model.matrix(~ factor(cl))

# Fit linear model
fit <- lmFit(expr_final, design)
fit <- eBayes(fit)
summary(fit)
#Testing of cluster difference
de_genes_raw <- topTable(fit, coef = 2, number = Inf, adjust.method = "none")
head(de_genes_raw)
de_genes <- topTable(fit, coef = 2, number = Inf, adjust.method = "fdr")
head(de_genes)

bio_sig_genes_raw <- de_genes_raw[abs(de_genes_raw$logFC) > 1 &
de_genes_raw$P.Val < 0.05, ]
nrow(bio_sig_genes_raw)
bio_sig_genes <- de_genes[abs(de_genes$logFC) > 1 & de_genes$adj.P.Val <
0.05, ]
nrow(bio_sig_genes)

# ------------------------------------------
# p-value Histogram (before & after FRD)
# ------------------------------------------
par(mfrow = c(1, 2))

# Raw P-values histogram
```

```r
hist(de_genes_raw$P.Value, breaks = 50,
     main = "Raw P-value Distribution",
     xlab = "Raw P-value", col = "lightblue",
     xlim = c(0, 1))
abline(v = 0.05, col = "red", lwd = 2, lty = 2)


# Adjusted P-values (FDR) histogram
hist(de_genes$adj.P.Val, breaks = 50,
     main = "FDR-adjusted P-value Distribution",
     xlab = "FDR-adjusted P-value", col = "lightgreen",
     xlim = c(0, 1))
abline(v = 0.05, col = "red", lwd = 2, lty = 2)


# ------------------------------------------
# Volcano Plot
# ------------------------------------------
par(mfrow = c(1,1))
# Smaller fold change
cat("Raw p < 0.05",
    sum(de_genes_raw$P.Value < 0.05 ), "\n")
cat("FDR < 0.05",
    sum(de_genes$adj.P.Val < 0.05 ), "\n")
cat("=== DE ANALYSIS ASSESSMENT ===\n")
cat("Total genes analyzed:", nrow(de_genes), "\n")
cat("Significant DE genes (FDR < 0.05):", sum(de_genes$adj.P.Val < 0.05),
"\n")
cat("Significant DE genes (FDR < 0.01):", sum(de_genes$adj.P.Val < 0.01),
"\n")
cat("Biologically significantly Up-regulated in cluster:",
sum(de_genes$adj.P.Val < 0.05 & de_genes$logFC > 1), "\n")
cat("Biologically significantly Down-regulated in cluster:",
sum(de_genes$adj.P.Val < 0.05 & de_genes$logFC < -1), "\n")

# Volcano plot to visualize DE results
volcano_data <- de_genes

plot(volcano_data$logFC, -log10(volcano_data$P.Value),
     xlab = "Log2 Fold Change", ylab = "-log10(Raw P-value)",
     main = "Volcano Plot: Differential Expression",
     pch = 16, cex = 0.6,
     col = ifelse(volcano_data$adj.P.Val < 0.05 & abs(volcano_data$logFC) > 1,
                  "red", "gray"),
     xlim = c(-max(abs(volcano_data$logFC)), max(abs(volcano_data$logFC))))



# ------------------------------------------
# GO Enrichment
# ------------------------------------------
# Create probe_id column
de_genes$probe_id <- rownames(de_genes)

# Merge DE table w annotation
de_annotated <- merge(de_genes, annotations, by.x = "probe_id",
                      by.y = "affy_hg_u133_plus_2", all.x = TRUE)

#cat("Total NA vlaues: ", colSums(is.na(de_annotated)))
# Get significant DE genes from cancer analysis
sig_genes <- de_annotated[de_annotated$adj.P.Val < 0.05 &
abs(de_annotated$logFC) > 1, ]
write.csv(sig_genes, "sig_genes_results.csv", row.names = TRUE)
cat("Total NA vlaues: ", colSums(is.na(sig_genes)))
cat("Total # of Significant Genes: ", nrow(sig_genes))
# Clean symbol data
sig_genes_symbol <- na.omit(unique(sig_genes$hgnc_symbol))
# Covert hgnc_symbol to ENTREZID
sig_entrez <- bitr(sig_genes_symbol, fromType = "SYMBOL", toType = "ENTREZID",
OrgDb = org.Hs.eg.db)

# Perform GO enrichment analysis for Biological Process (BP)
go_bp <- enrichGO(gene         = sig_entrez$ENTREZID,
                  OrgDb        = org.Hs.eg.db,
                  keyType      = "ENTREZID",
                  ont          = "BP",          # Can also be "MF" or "CC"
                  pAdjustMethod= "BH",
                  pvalueCutoff = 0.05,
                  qvalueCutoff = 0.05)

head(go_bp)
write.csv(go_bp, "GOenrichment_results.csv", row.names = TRUE)


# ------------------------------------------
# Visualization of GO enrichment
# ------------------------------------------

# Dotplot for Biological Process
dotplot(go_bp, showCategory=15, title="GO Enrichment - Biological Process")

# GO Network Plot (BP example)
cnetplot(go_bp, showCategory=10, foldChange=sig_genes$logFC)

par(mfrow = c(1, 1))

# ------------------------------------------
# Survival Analysis
# ------------------------------------------
setequal(clinical$sampleID, names(cl))
clinical_clean <- clinical[clinical$sampleID %in% names(cl), ]
clinical_clean$cluster <- cl[match(clinical_clean$sampleID, names(cl))]
cat("Missing Value for Clinical Data:", colSums(is.na(clinical_clean)), "\n")

# Clincial data remove imcompleted record
clinical_clean <- clinical_clean[!is.na(clinical_clean$Surv_time)
& !is.na(clinical_clean$event),]
nrow(clinical_clean)

# ------------------------------------------
# Identify Zero Event Columns and Merge
# ------------------------------------------
unique_values_list <- list()
for (col_name in names(clinical_clean)) {
  unique_values_list[[col_name]] <- unique(clinical_clean[[col_name]])
}
print(unique_values_list)
# ERstatus
er_table <- table(clinical_clean$ERstatus, clinical_clean$event)
cat("\nERstatus event distribution:\n")
print(er_table)

# LNstatus
ln_table <- table(clinical_clean$LNstatus, clinical_clean$event)
cat("\nLNstatus event distribution:\n")
print(ln_table)
#LN-   127   22
#LN?     9    0
#LN+    45   33
cat("ER? represents", round(4/nrow(clinical_clean)*100, 1), "% of total
samples\n")
cat("LN? represents", round(9/nrow(clinical_clean)*100, 1), "% of total
samples\n")
#FALSE TRUE
#ER-    25    6
#ER?     4    0
#ER+   152   49
# ------------------------------------------
# Collapse Problematic Categories (LN?-> LN-, ER? -> ER-)

clinical_clean$LNstatus[clinical_clean$LNstatus == "LN?"] <- "LN-"
clinical_clean$LNstatus <- factor(clinical_clean$LNstatus)
clinical_clean$ERstatus[clinical_clean$ERstatus == "ER?"] <- "ER-"
clinical_clean$ERstatus <- factor(clinical_clean$ERstatus)


# ------------------------------------------
# Log Rank Test & K-M Survival Plot
# -------------------------------------------
surv_obj <- Surv(time = clinical_clean$Surv_time, event =
clinical_clean$event)
km_comp <- survdiff(surv_obj ~ clinical_clean$cluster) # stratify by gender
km_comp

fit_km <- survfit(surv_obj ~ clinical_clean$cluster)
autoplot(fit_km) +
  labs(x = "\n Survival Time (Years) since diagnosis ", y = "Survival
Probabilities \n",
       title = "K-M Survival Curves for Breast Cancer Patients \n")

fit_km_ln <- survfit(surv_obj ~ clinical_clean$LNstatus)
summary(fit_km_ln)

autoplot(fit_km_ln) +
  labs(x = "\n Survival Time (Years) since diagnosis ", y = "Survival
Probabilities \n",
       title = "K-M Survival Curves for Breast Cancer Patients \n")


# -------------------------------------------
# Univariate Cox
# -------------------------------------------
#surv_obj_m2 <- Surv(time = clinical_clean$Surv_time, event =
clinical_clean$event)
surv_obj <- Surv(time = clinical_clean$Surv_time, event =
clinical_clean$event)
# List of variables to test
variables <- c("cluster", "histgrade", "ERstatus", "PRstatus", "age",
"tumor_size_mm", "LNstatus")

# Create a function for univariate analysis
perform_univariate_analysis <- function(data, variables) {
  results <- list()

  for(var in variables) {
    # Create formula
    if(is.numeric(data[[var]])) {
      # For continuous variables
      formula <- as.formula(paste("surv_obj ~", var))
    } else {
      # For categorical variables
      formula <- as.formula(paste("surv_obj ~ factor(", var, ")"))
    }

    # Fit Cox model
    cox_uni <- coxph(formula, data = data)
    cat("\nCox Result for", var, ":\n")
    # Store results
    print(summary(cox_uni))
  }
}

# Perform univariate analysis (merge cat)
uni_results_2 <- perform_univariate_analysis(clinical_clean, variables)

# ------------------------------------------
# Multivariate Model
# ------------------------------------------

# Adjusted for prognostic factors: histological grade, ER, PR, age, tumor
size, lymph nodes
cox_model_c1<- coxph(surv_obj ~ cluster + histgrade + ERstatus + PRstatus +
age + tumor_size_mm + LNstatus,
                    data = clinical_clean)

summary(cox_model_c1)

cox_model_c2<- coxph(surv_obj ~ cluster + tumor_size_mm + LNstatus,
                    data = clinical_clean)

summary(cox_model_c2)
```