Tunisian Republic
Ministry of Higher Education and Scientific Research
University of Carthage
National Institute of Applied Sciences and Technology

End Of Year Project Report

Predictive modeling and pathway classification in breast cancer: Integrating clinical and transcriptomic data

Presented by

Chaima Ben Mohamed
Malek Mrabet
Syrine Jerbi
Sajanjel Jaballi

Supervised by **Dr. Ferid Abidi**

Co-supervised by **Dr. Slah Ouerhani**

Academic Year: 2023-2024

Acknowledgements

We express our sincerest gratitude to our supervisor Dr. Ferid Abidi and our co-supervisor Dr. Slah Ouerhani for their invaluable guidance, unwavering support, and expert insights throughout our research.

We extend special thanks to our evaluators for their thorough review and constructive feedback, which will undoubtedly enhance the quality and clarity of our work. Lastly, we take great pleasure in acknowledging the collaborative spirit and synergy within our team, which significantly contributed to the success of this project.

Table of Contents

Table of figures

List of tables

General Introduction

The relentless advance of breast cancer, with its complex pathological landscape and profound impact on public health, compelled continuous research and innovation to enhance diagnosis, treatment, and management strategies. Our project explored the convergence of medical science and advanced analytics, particularly focusing on the integration of machine learning techniques to better understand and treat breast cancer. This interdisciplinary approach highlighted the complexity of the disease and the potential of modern technology to revolutionize healthcare.

We started by examining breast cancer comprehensively, detailing its histological subtypes, molecular pathways, and therapeutic strategies, from traditional treatments to cutting-edge gene therapy. We also explored the significance of high-throughput analysis and the pivotal role of machine learning in enhancing model optimization, leveraging large datasets for deeper insights.

The project further delved into the thorough analysis of clinical and molecular data, employing various data analysis techniques to uncover intricate relationships between different attributes. This included exploring mutations and interactions between genes, providing a deep dive into the data manipulation and interpretation required.

We then focused on predictive modeling and pathway classification using machine learning to predict chemotherapy responses and classify cancer pathways. This phase included data pre-processing and the development and optimization of models, emphasizing their implications for personalized medicine and therapeutic strategies enhancement.

Overall, this project deepened our understanding of breast cancer through a combination of medical insights and analytical rigor, pushing the boundaries of how machine learning could be utilized to improve treatment outcomes and advance the field of oncology.

Chapter 2 :

Data exploration and manipulation

Chapter 2 : Data exploration and manipulation

I. Data

1. Data extraction

The dataset utilized in this research project originates from Kaggle, where data pertaining to breast cancer was sourced from the Molecular Taxonomy of Breast Cancer International Consortium (METABRIC) database. This database, a collaboration between Canada and the UK, contains targeted sequencing data from 1,980 primary breast cancer samples, with a shape of (1904, 692).

Figure 15. Data shape with python

2. Data understanding

2.1.Clinical data

The dataset encompasses a comprehensive range of attributes pertinent to breast cancer diagnosis, treatment, and prognosis. These features provide crucial information that supports the development and validation of predictive models, aiding in early detection and personalized treatment plans for breast cancer patients.

- **Cancer Type**

Breast Cancer: This common cancer usually starts in the milk ducts or glands. It is often detected through mammograms and treated with surgery, radiation, or chemotherapy.
Breast Sarcoma: A rare cancer that begins in the connective tissue of the breast, often treated with surgery and sometimes radiation or chemotherapy.

- **Cancer Type Detailed**

Breast Invasive Ductal Carcinoma: The most common type, starting in the milk ducts and spreading to nearby tissues. It is typically treated with a combination of surgery, radiation, and chemotherapy.
Breast Mixed Ductal and Lobular Carcinoma: Combines features of ductal and lobular cancers, starting in the milk ducts and glands. Treatment often requires a tailored approach involving surgery, radiation, and systemic therapies.

Breast Invasive Lobular Carcinoma: Begins in the milk-producing glands and can spread to other parts of the breast and body. Treatment usually includes surgery, radiation, and hormone therapy.
Breast Invasive Mixed Mucinous Carcinoma: Contains regular cancer cells and mucin-producing cells, giving it a unique composition. Treatment often involves surgery and may include radiation and chemotherapy.
Metaplastic Breast Cancer: A rare type with a mix of different cell types, requiring specialized treatment plans, often including surgery, radiation, and chemotherapy.

- **PAM50 and Claudin-Low Subtype**

PAM50 Gene Signature: Categorizes breast cancer tumors into intrinsic subtypes based on the expression levels of 50 genes:
Luminal A: Hormone receptor-positive (ER+ and/or PR+), HER2-negative tumors with low proliferation rates and a relatively good prognosis.
Luminal B: Hormone receptor-positive (ER+ and/or PR+), with higher proliferation rates than Luminal A tumors, and a higher risk of recurrence.
HER2-enriched: HER2-positive tumors with overexpression of the HER2 gene, associated with aggressive behavior and a higher risk of recurrence.
Basal-like: Triple-negative tumors (ER-, PR-, HER2-) with gene expression patterns similar to basal/myoepithelial cells, known for aggressiveness and limited treatment options.
Normal-like: Tumors with gene expression patterns resembling normal breast tissue, relatively rare and heterogeneous, often associated with better outcomes.
Claudin-Low Subtype: Defined by gene expression characteristics, including low expression of cell–cell adhesion genes, high expression of epithelial–mesenchymal transition (EMT) genes, and stem cell-like/less differentiated gene expression patterns. This subtype indicates increased tumor invasiveness and aggressiveness.

- **Type of Breast Surgery**

Mastectomy: This surgery involves the removal of one or both breasts, typically to treat or prevent breast cancer.
Breast Conserving Surgery: Also known as lumpectomy, this procedure removes only the cancerous part of the breast, preserving as much of the breast as possible. It is often followed by radiation therapy to eliminate any remaining cancer cells.

- **Cancer Cellularity Post Chemotherapy**

Refers to the density of cancer cells remaining in a tumor after chemotherapy. Assessing this helps gauge the effectiveness of treatment and plan further therapy if needed.

- **ER Status Measured by IHC**

It indicates whether a tumor is sensitive to hormone therapies, guiding treatment decisions based on estrogen receptor presence.

- **Neoplasm Histologic Grade**

It Evaluates tumor cell abnormality to predict behavior and prognosis. Higher grades suggest more aggressive tumors, influencing treatment decisions and patient management.

- **HER2 Status Measured by SNP6**

It assesses HER2 positivity using advanced molecular techniques, such as next-generation sequencing.

- **Tumor Other Histologic Subtype**

It refers to specific subtypes of tumors that do not fall into common categories, aiding pathologists in accurate classification and appropriate treatment planning.

- **Integrative Cluster**

It is a classification method that combines multiple biological data types to better understand and categorize cancer subtypes, aiding in personalized treatment approaches:

Ductal/NST (No Special Type): The most common type of breast cancer, also known as invasive ductal carcinoma (IDC). It starts in the milk ducts and invades the surrounding breast tissue. Treatment usually involves surgery, radiation, and possibly chemotherapy, hormone therapy, or targeted therapy depending on the tumor characteristics.

Mixed: Tumors exhibit features of more than one type of breast cancer (e.g., both ductal and lobular). The prognosis and treatment depend on the predominant type and characteristics of the mixed tumor.

Lobular: Invasive lobular carcinoma (ILC) starts in the lobules (milk-producing glands) and spreads to nearby tissues. Treatment often involves surgery, hormone therapy, and sometimes chemotherapy. ILC may respond differently to some treatments compared to IDC.

Tubular/Cribriform: Less common and generally less aggressive types of breast cancer. Tubular carcinomas form tube-like structures, while cribriform carcinomas have a pattern of holes or spaces between cancer cells. These types generally have a better prognosis and may require less aggressive treatment.

Mucinous (Colloid): Characterized by the production of mucin (a component of mucus). The cancer cells are surrounded by pools of mucin. Typically, mucinous carcinomas have a favorable prognosis and may be treated with surgery, possibly followed by hormone therapy or radiation.

Medullary: A rare type of breast cancer that often has a better prognosis than other high-grade tumors. Cells appear large and have a clear boundary from the surrounding tissue, often with significant lymphocytic infiltration. Medullary carcinomas are often triple-negative but can respond well to chemotherapy.

Metaplastic: A rare and diverse group of breast cancers where the cancer cells differentiate into multiple cell types, such as squamous cells or spindle cells. Metaplastic breast cancer is often more aggressive and may not respond well to standard treatments. Treatment typically involves surgery, chemotherapy, and radiation.

- **Primary Tumor Laterality**

It specifies whether a tumor is located on the left or right side of the body, crucial for accurate diagnosis and treatment planning.

- **Nottingham Prognostic Index**

It is a tool used to determine the prognosis of breast cancer based on tumor size, lymph node involvement, and histologic grade. This index helps predict patient outcomes and guide treatment decisions.

- **OncoTree Code**

It is a classification system that assigns unique codes to different cancer types and subtypes based on their genetic and molecular characteristics. This coding helps researchers and clinicians accurately identify and study specific cancers, facilitating personalized treatment approaches.

- **Overall Survival Months**

It is the duration from the time of intervention to death.

- **Three Gene Classifier Subtype**

Categorizes breast cancer based on the expression of three key genes, guiding personalized treatment decisions:

ER-/HER2-: Tumors do not have estrogen receptors (ER-) and do not overexpress the HER2 protein (HER2-). These tumors are typically more aggressive and may require chemotherapy.

ER+/HER2- High Prolif: Tumors have estrogen receptors (ER+), do not overexpress the HER2 protein (HER2-), and have a high proliferation rate. These tumors respond to hormonal therapies but may require additional treatments like chemotherapy.

ER+/HER2- Low Prolif: Tumors have estrogen receptors (ER+), do not overexpress the HER2 protein (HER2-), and have a low proliferation rate. These tumors are less aggressive and respond well to hormonal therapies.

HER2+: Tumors overexpress the HER2 protein (HER2+), regardless of estrogen receptor status. HER2+ tumors can be more aggressive but often respond well to HER2-targeted therapies and chemotherapy [25].

2.2.Molecular and mutation data

The genetic attributes within the dataset encompass mRNA level z-scores for 331 genes and mutation data for 175 genes. From CBioPortal, mRNA, detected through DNA probes, represents gene expression, known as the transcriptome. mRNA Z-scores, calculated against a reference population, gauge gene expression relative to the mean, aiding in the identification of up- or down-regulated genes in tumors or normal samples.

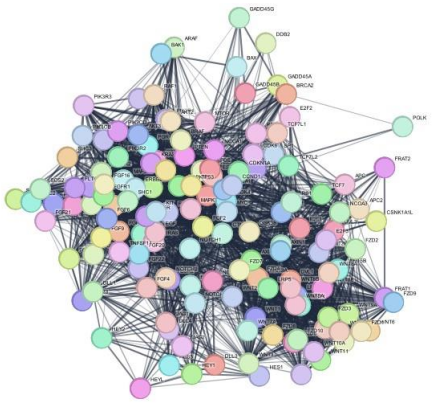


Figure 16. KEGG gene interaction network

Our dataset includes gene expression and mutation data that do not necessarily contribute to cancer-related genes. To address this, we utilized the **Cytoscape network platform**, which integrates data from numerous cancer-related databases and maps gene interactions. We filtered our dataset by removing genes that are not represented in the KEGG database for cancer, ensuring that our analysis focuses on genes with known relevance to cancer.

```
# Check gene names in patient expression data against KEGG interactions
patient_genes = set(data.columns[2:490])
mutations_columns = data.columns[490:663]

# Extract gene names by removing the '_mut' suffix and converting to uppercase
mutations_genes = set(col.split('_mut')[0].upper() for col in mutations_columns if '_mut' in col)

cancer_genes = set(inter[['node1', 'node2']].values.flatten())
# Display gene names present in expression data but absent in KEGG interactions
genes_only_in_patient_data = patient_genes - cancer_genes
print("Gene names present only in patient expression data:")
print(genes_only_in_patient_data)
# Display gene names present in mutation data but absent in KEGG interactions
genes_only_in_patient_mutation = mutations_genes - cancer_genes
print("Gene names present only in patient mutation data:")
print(genes_only_in_patient_mutation)
```

```
✓ 0.0s
Gene names present only in patient expression data:
{'TNK2', 'SETD2', 'BMP5', 'HIST1H2BC', 'AURKA', 'CACNA2D3', 'MAPK4', 'TSC1', 'INHA', 'STAT2', 'AHNAK2', 'SPTBN1', 'THADA', 'CDK8', 'RAD51D', 'HDAC9',
Gene names present only in patient mutation data:
{'SMARCD1', 'PRPS2', 'AFDN', 'CDH1', 'SETD2', 'HIST1H2BC', 'CACNA2D3', 'GLDC', 'UTRN', 'MEN1', 'AHNAK2', 'CTNNA1', 'GATA3', 'SIAH1', 'STAB2', 'CTNNA3
```

Figure 17. Cancer relevant gene filtering

3. Project tools

Table 3 : Tools and Packages

Software/Tool	Purpose	Usage	Software	Important Packages/Libraries
RStudio	Data analysis and visualization	EDA, preprocessing, feature engineering	R	dplyr, tidyr, ggplot2, caret, randomForest
VSCode	Scripting and development	Python scripting, model development	Python	pandas, numpy, matplotlib, seaborn, sklearn, shap
Random Forest	Machine learning algorithm	Prediction and classification models	R, Python	randomForest (R), sklearn.ensemble (Python)
GridSearchCV	Hyperparameter optimization	Optimization of machine learning models	Python	sklearn.model_selection
SHAP	Model interpretability	Understanding feature importances	Python	Shap
k-fold	Model validation	Cross-validation of models	Python	sklearn.model_selection.KFold
STRING	Bioinformatics tool	Analysis of protein networks	Web Interface	None (direct use through web interface)
KEGG	Bioinformatics database	Gene-pathway association studies	R	KEGGREST
Cytoscape	Network visualization tool	Visualization of molecular pathways	Desktop App	None (direct use through application and plugins)
FASMIC	Gene mutation analysis	Analyzing gene mutations for cancer studies	Web Interface	None (direct use through web interface)

II. Exploratory data analysis

Exploratory Data Analysis (EDA) serves as a crucial initial step in data science projects, involving the scrutiny and visualization of data to grasp its fundamental characteristics, uncover patterns, and discern relationships between variables. It encompasses studying and exploring datasets to comprehend their primary traits, reveal patterns, pinpoint outliers, and recognize connections among variables, typically preceding more formal statistical analyses or modeling endeavors. Key components of EDA include the distribution of data points, employing visual representations like histograms and scatter plots, identifying outliers, conducting correlation analysis, handling missing values, computing summary statistics, and validating assumptions. EDA holds considerable significance for several reasons, including facilitating familiarity with the dataset, identifying patterns and relationships, detecting anomalies and outliers, informing feature selection and engineering, optimizing model design, facilitating data cleaning, and enhancing communication of findings [8].

1. Univariate Analysis

In our analysis of the breast cancer dataset, we conducted univariate analysis, focusing on the distribution, central tendency, and dispersion of individual variables like gene expression levels, PAM50 subtypes, and mutation profiles. This method, that examines each variable independently, is crucial for understanding how their distributions might affect further statistical analyses. By thoroughly assessing these elements, we laid a solid foundation for subsequent, more complex investigations into breast cancer prognosis and treatment factors.

In this step, we analyzed the clinical features that had a categorical data nature through pie charts, detailing the distribution of types of breast surgery, cancer type, cancer type detailed categories, cellularity, pam50 + claudin-low subtypes, estrogen receptor status, HER2 status measured by SNP6, inferred menopausal state, primary tumor laterality, oncotype recurrence score, and other histologic subclassifications. (Figure18)

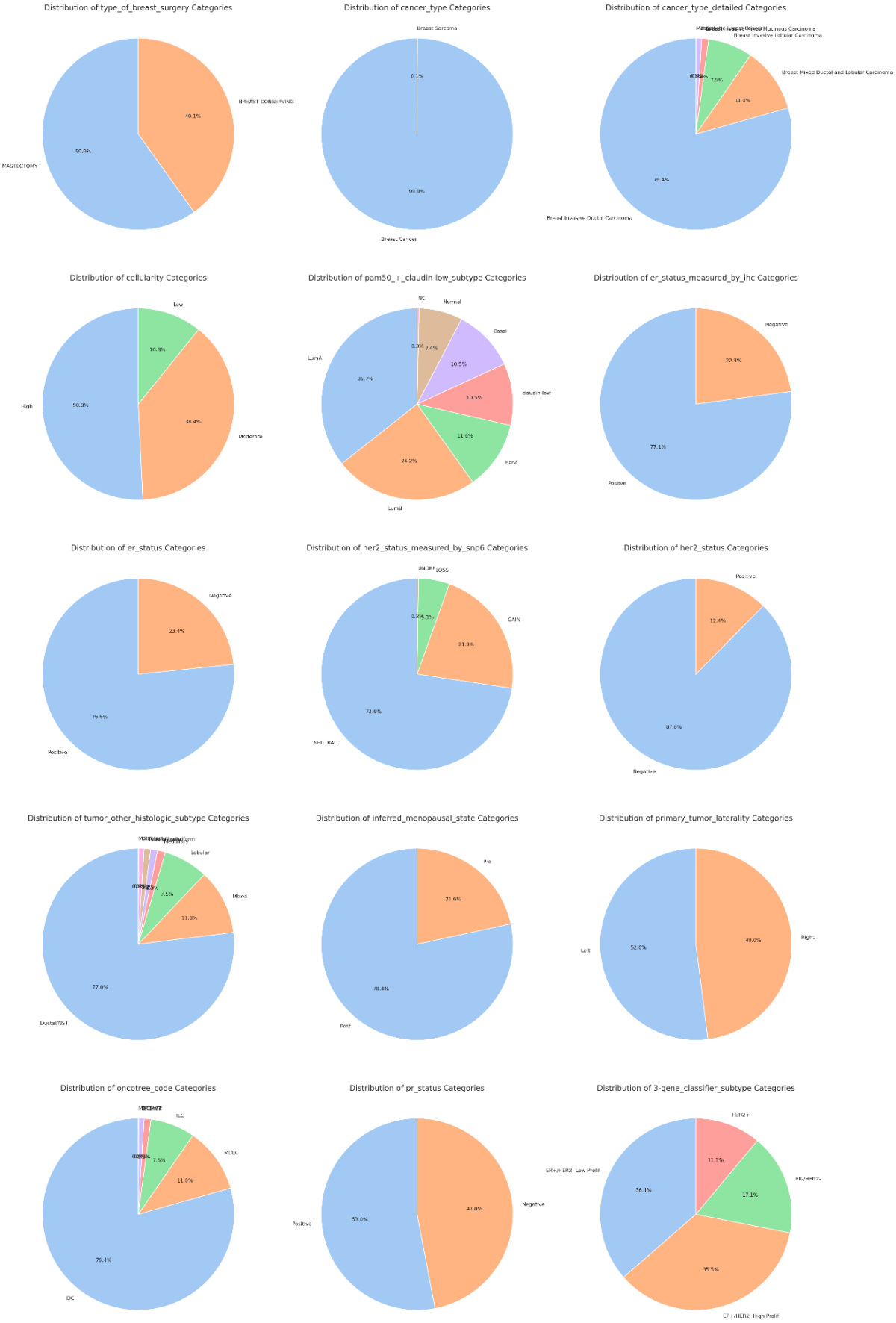


Figure 18. Pie Chart distribution of clinical data across samples

⇨ These visualizations collectively offer a detailed overview of the patient cohort, reflecting the complex interplay of pathological and treatment variables in breast cancer management. Each chart provides a distinct perspective on the diversity of patient characteristics and therapeutic approaches within the dataset.

1.1.Genetic attributes analysis

For the genetic attributes in the dataset, we started by plotting the density of expression levels across different genes, the aim is showcasing the general distribution and range of gene activity observed in the study. (Figure 19)

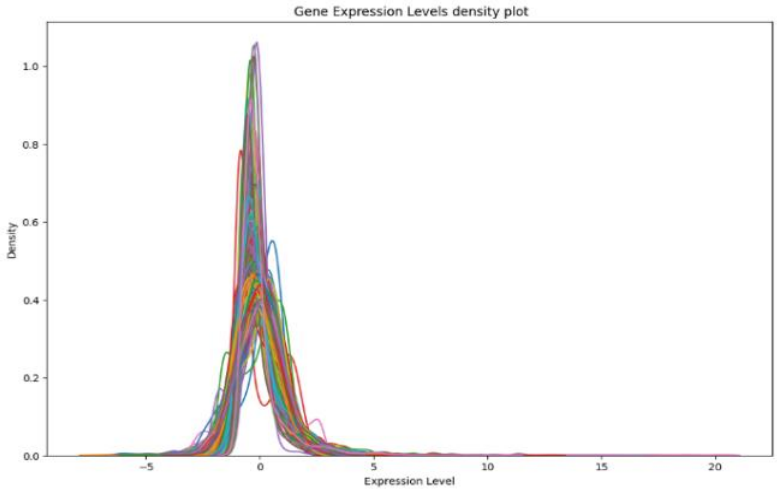


Figure 19. Density Plot of gene expression levels across multiple genes

⇨ This plot presents the kernel density estimates for gene expression levels across multiple genes. It visualizes the distribution's shape and spread, indicating variations in gene expression levels, with most genes showing a central tendency near zero but with different variances.

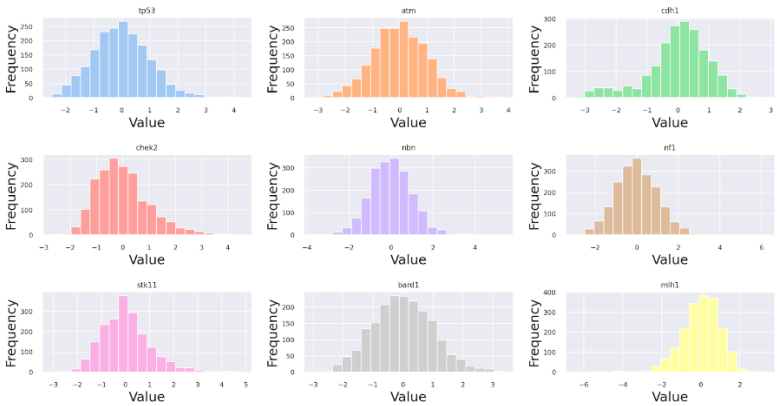


Figure 20. Frequency distribution histograms of Z-score normalized gene expressions

⇨ These histograms display the distribution of gene expression levels for various genes (e.g., TP53, ATM, CDH1) across samples. Each histogram shows a **normal-like distribution** centered around zero, indicative of the standard normalization typically applied to gene expression data for analysis.

Figure 21. Line Plot of gene expression levels across samples for multiple genes

This plot illustrates the expression levels of the studied genes across clinical samples. The variability across samples suggests heterogeneity in tumor genetic profiles, which can inform targeted therapies.

2. Bivariate Analysis

In this step, we employed bivariate analysis to explore and quantify the relationships between pairs of variables. This approach was integral to our exploratory data analysis, enabling us to uncover potential associations that could inform further research and hypothesis testing.

2.1.Clinical attributes analysis

We analyzed the relationships between clinical attributes and patient outcomes, identifying key differences between survivors and non-survivors. This analysis underscores the importance of clinical features in predicting patient prognosis. (Figure 22)

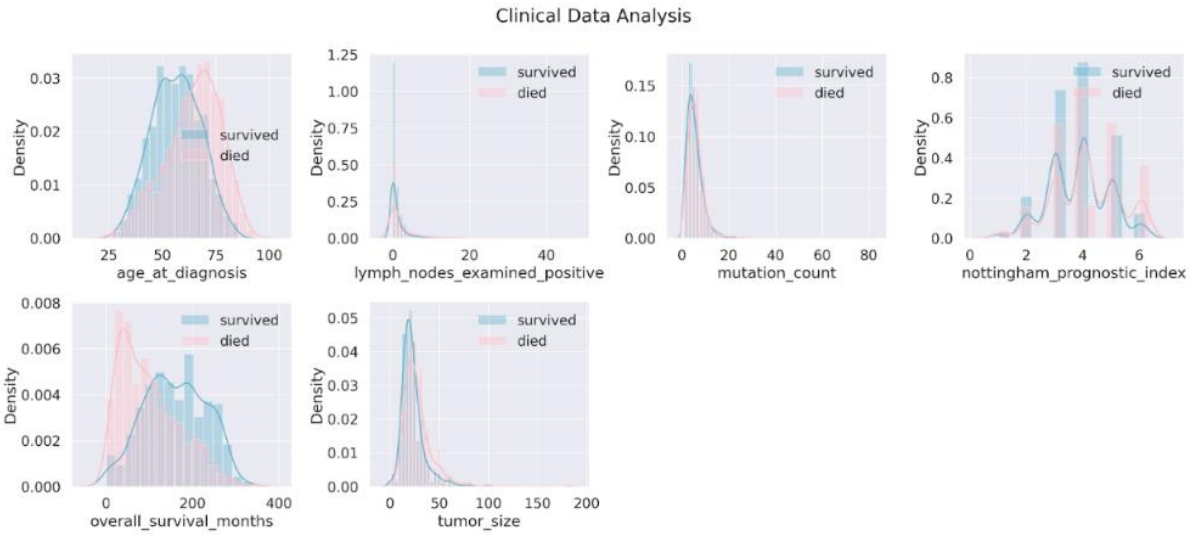


Figure 22. Density Plots of clinical variables by patient outcome for numerical data

- ⇨ Younger age at diagnosis correlates with higher survival rates, reflecting the greater physical resilience of younger patients, which may enhance their response to treatments. Conversely, older age groups show increased mortality, possibly due to diminished biological vitality and comorbid conditions.
- ⇨ Elevated counts of positive lymph nodes and mutations are prevalent among non-survivors, suggesting more advanced and aggressive disease states that challenge effective management and treatment.
- ⇨ The Nottingham prognostic index, which incorporates tumor size, lymph node status, and histological grade, effectively distinguishes between survival outcomes, with lower values indicating less aggressive disease manifestations.
- ⇨ Longer overall survival months observed in survivors align with the typically slower progression of less severe cancer forms, allowing for more effective intervention opportunities.
- ⇨ Smaller tumor sizes are linked to higher survival rates as they are often indicative of earlier stages of cancer, reducing the likelihood of metastasis and improving treatment success rates.

Understanding cancer survival rates hinges significantly on analyzing the age at diagnosis, as it serves as a critical determinant in shaping the prognosis and treatment trajectory for individuals affected by the disease. For that reason, we conducted a brief analysis to display the distribution of patient death by their diagnosis. (Figure 23)

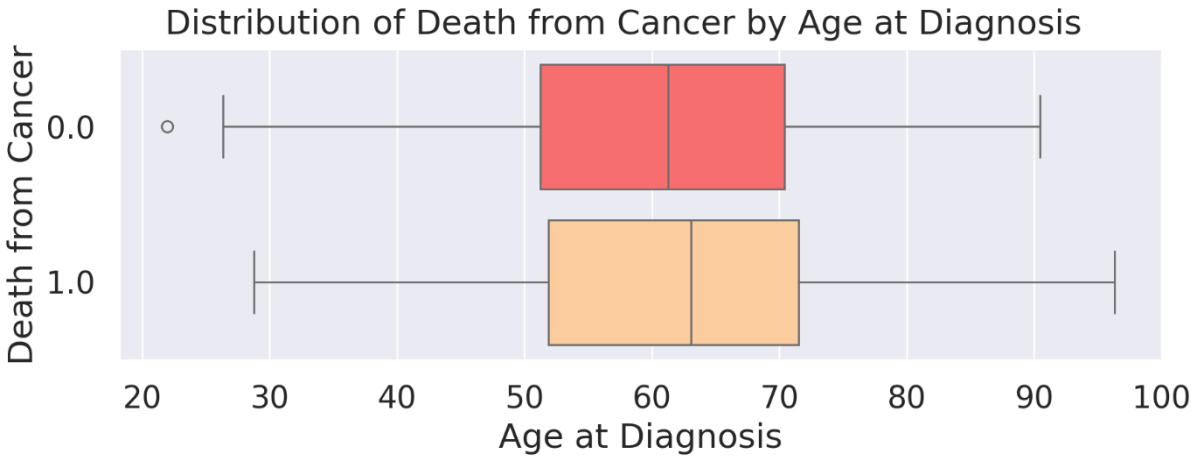


Figure 23. Box Plot of survival time and age at diagnosis by outcome

- The box plot indicates that people who died from cancer were diagnosed at a median age of around 65 years, which is slightly higher than the around 60 years median age of those who survived. This suggests that older age at diagnosis might be linked to higher mortality rates. It underscores the significance of early detection and age-specific interventions to enhance survival rates.

2.2.Genetic attributes analysis

When analyzing genetic attributes, bivariate exploration provides vital insights into the interplay between gene expression and survival outcomes. We generated visualizations that offer valuable glimpses into the complex genetic landscape, revealing potential correlations and patterns influencing the prognostic, starting with a density plot of gene expression by patient survival outcome. (figure 24)

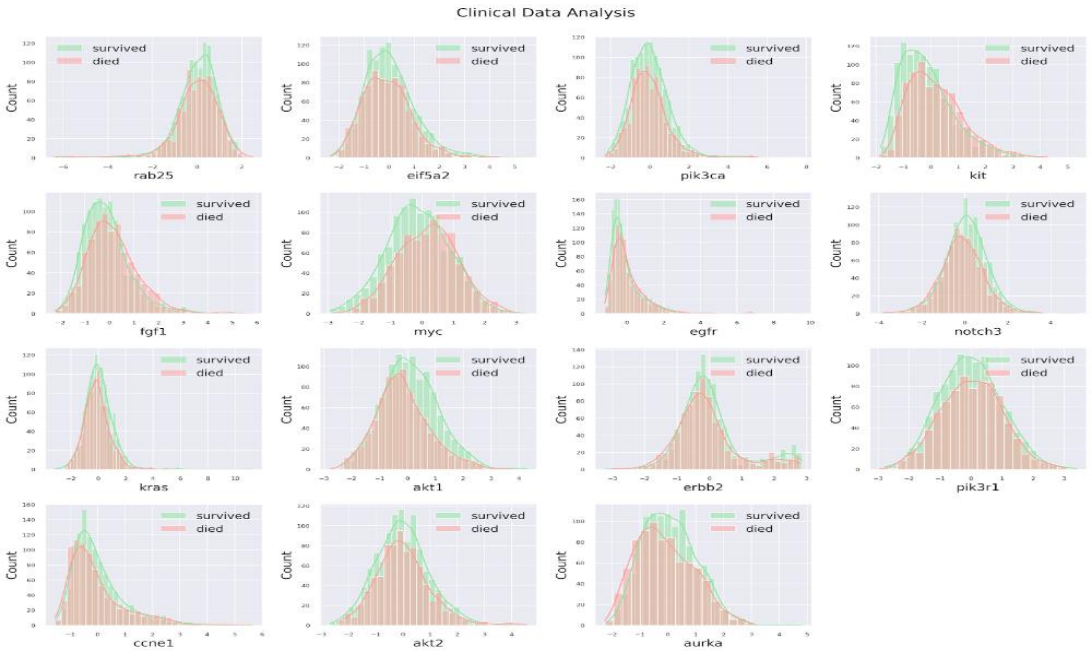


Figure 24. Density Plots of gene expression by survival outcome

- The density plots highlight significant differences in the distributions of clinical markers such as akt1, akt2, egfr, erb2, notch3, and pik3r1 between survivors and non-survivors. These variations suggest the critical roles of the Akt, EGFR, Notch, and PI3K pathways in cancer progression and patient outcomes. The distinct patterns observed in these markers underscore their potential as prognostic biomarkers and therapeutic targets, offering valuable insights for personalized cancer treatment strategies.

3. Multivariate analysis

In breast cancer research, multivariate analysis provides a holistic perspective by simultaneously examining various factors.

We conducted an analysis to understand the relationship between tumor size, tumor stage, and overall survival in breast cancer patients. (figure 25)

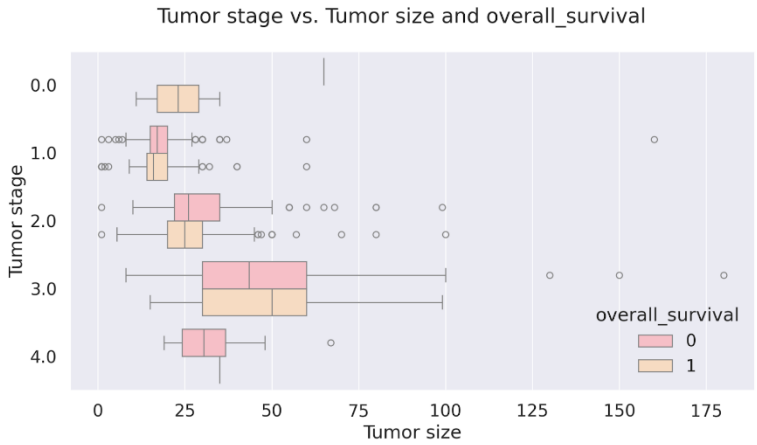


Figure 25. Box Plot of tumor stage versus size and survival

- Tumor Growth: Higher stages generally correlate with larger tumors, indicating progressive tumor growth.
- Survival Variability: Similar tumor sizes across different stages and survival outcomes suggest that size alone doesn't dictate survival, highlighting the complex nature of cancer prognosis.
- Heterogeneity: The range and outliers in tumor sizes at each stage suggest biological variability in how tumors behave and progress.

Next, we analyzed the distribution of age, tumor diameter, and number of positive lymph nodes across different survival outcomes. This histogram-based visualization categorizes patients based on their survival status, distinguishing between those who survived, those who died from cancer, and those who died from other causes. The goal of this analysis is to assess how these clinical factors correlate with patient outcomes, offering insights into their prognostic significance. (Figure 26)

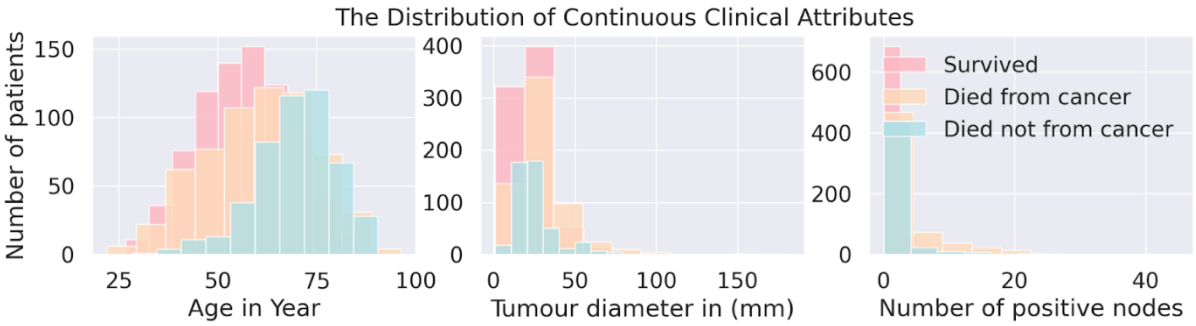


Figure 26. Histograms of continuous clinical attributes by survival and cancer outcome

- ⇨ Age Distribution: Younger patients tend to show higher survival rates, indicating age's influence on prognosis.
- ⇨ Tumor Diameter: Larger tumors correlate with higher cancer mortality, underscoring tumor size's impact on disease severity.
- ⇨ Positive Lymph Nodes: More positive lymph nodes are associated with worse survival and higher cancer mortality, reflecting their role in disease progression.

To visualize the relationships between various clinical and pathological attributes in our breast cancer dataset, we generated a correlation plot that maps out the correlation coefficients between factors such as age at diagnosis, chemotherapy, tumor size, tumor stage, and overall survival, among others. The aim is to identify significant correlations that may influence patient outcomes. (Figure 27)

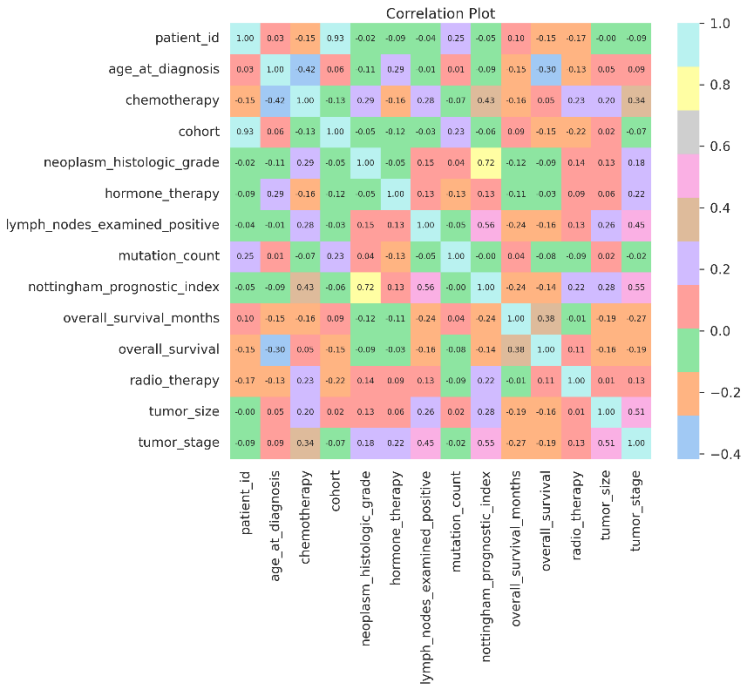


Figure 27. Heatmap of clinical attributes correlation

- ⇨ The correlation plot above reveals key relationships among clinical attributes in cancer prognosis. Notably, lymph_nodes_examined_positive and mutation_count are positively correlated with tumor_stage, indicating more advanced tumors tend to have higher lymph

node involvement and mutations. Chemotherapy and radio_therapy show moderate correlations with longer overall_survival_months, reflecting their effectiveness. The nottingham_prognostic_index effectively predicts patient outcomes, closely linked to survival time and tumor stage.

Looking into the genetic data, we examined the relationship between overall survival for the gene expression of TWIST1, TUBB4A, and tumor stage for the gene expression TAF4B and STAT3 in our breast cancer dataset. The scatter plots we generated show the z-score normalized expression levels of each gene, classified by overall survival status, providing insights into how gene expression correlates with patient outcomes. (Figures 28, 29, 30, 31)

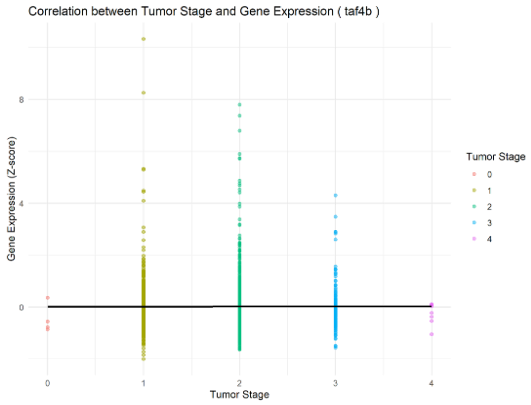


Figure 28. Scatter plot of TAF4B gene expression vs. tumor stage

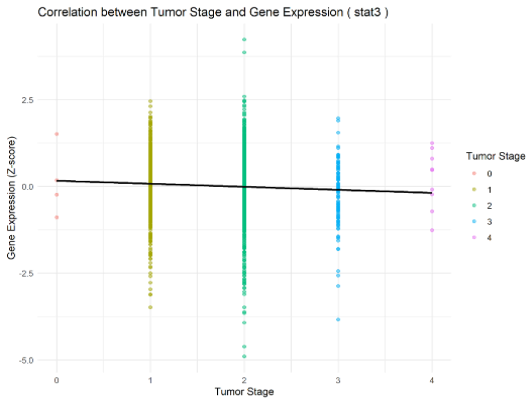


Figure 29. Scatter plot of STAT3 gene expression vs. tumore stage

⇨ The plots illustrate the correlation between the expression of specific genes (e.g., TAF4B, STAT3) and tumor stages. Variability in gene expression across different tumor stages suggests these genes play roles in tumor progression or are affected by tumor microenvironment changes.

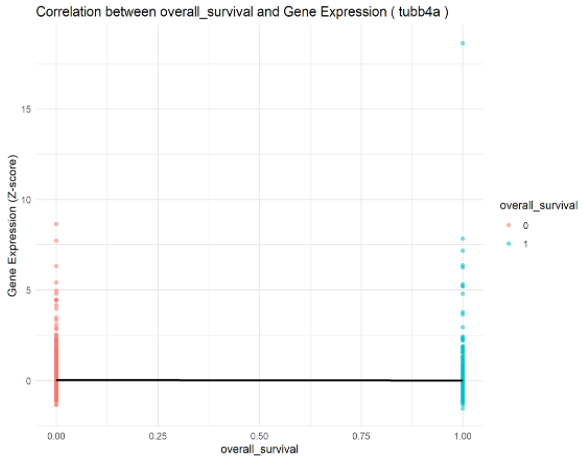


Figure 30. Scatter plot of tubb-4a gene expression vs. overall_survival

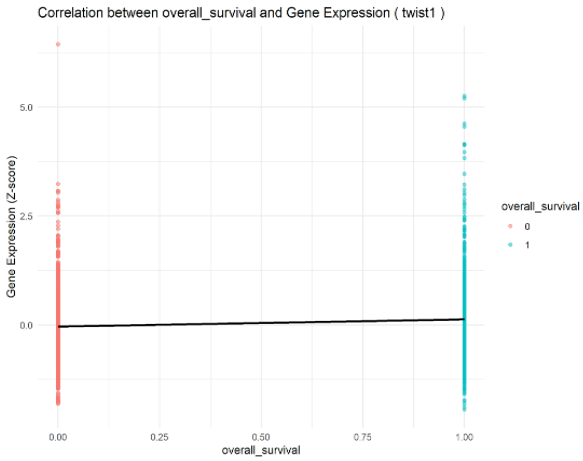


Figure 31. Scatter plot of twist1 gene expression vs.overall_survival

- ⇨ The plots analyze the relationship between overall survival and the expression of genes like TUBB4A and TWIST1. Significant shifts in expression patterns relative to survival outcomes indicate these genes could be prognostic markers, with potential implications for targeting in therapeutic strategies.

Subsequently, we employed principal component analysis (PCA) to create a cluster plot that groups breast cancer patients based on similarities in their clinical and genetic profiles. The plot displays data points in a two-dimensional space, defined by the first two principal components, which explain a significant portion of the variance in the dataset. (Figure 32)

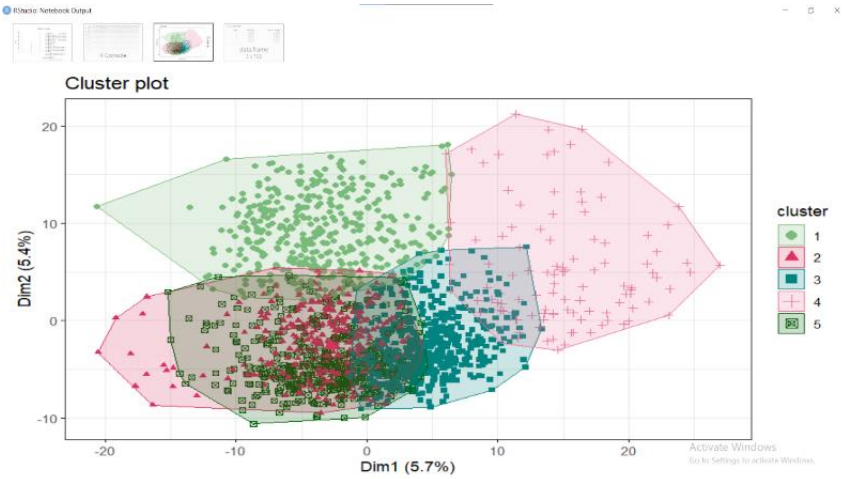


Figure 32. PCA-Based cluster plot for patient data

Cluster Distribution:

Cluster 1: This cluster is tightly grouped, suggesting a high degree of similarity among the samples within this cluster.

Cluster 2: Distributed over a broad area, indicating variability within this group. This might suggest a heterogenous cluster with samples sharing some, but not all, characteristics.

Cluster 3: Similarly broad distribution as Cluster 2, suggesting internal variability.

Cluster 4: Shows some overlap with other clusters, especially Cluster 5, which may indicate shared characteristics or transitional states between these clusters.

Cluster 5: Primarily overlaps with Cluster 4, highlighting potential similarities or a gradient in the patient characteristics between these two clusters.

Interpretations:

Clinical Implications: The distinct clustering of patient data points suggests varying underlying biological or clinical traits among the groups. For instance, clusters with tight grouping (like Cluster 1) may represent a subgroup of patients with similar prognostic factors or treatment responses, which could be crucial for personalized medicine approaches.

Research and Treatment Strategy: Identifying such clusters can help in stratifying patients based on predicted outcomes, leading to more targeted research and potentially more effective treatment strategies tailored to specific patient subgroups.

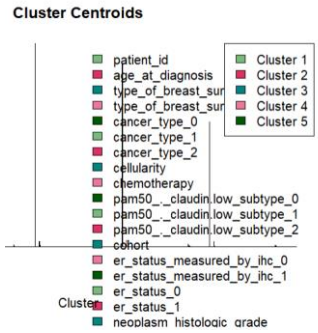


Figure 33. Bar plot of cluster centroids for patient groups

- ⇨ The Centroid Plot shows the centroids of each cluster. it indicates **the variables that are most influential** in determining the cluster centroids, which helps in understanding the characteristics of each cluster.

For further sophisticated clustering, we used K-means clustering combined with two-dimensional principal component analysis (2D PCA) to explore gene expression patterns in our dataset. We instructed this algorithm to visually groups genes based on their expression profiles—overexpressed, underexpressed, and normal expression—represented by different colors in the plot. (Figure 34)

Z-score = 0 ⇒ Gene normally expressed

Z-score > 0 ⇒ Gene overexpressed

Z-score < 0 ⇒ Gene underexpressed

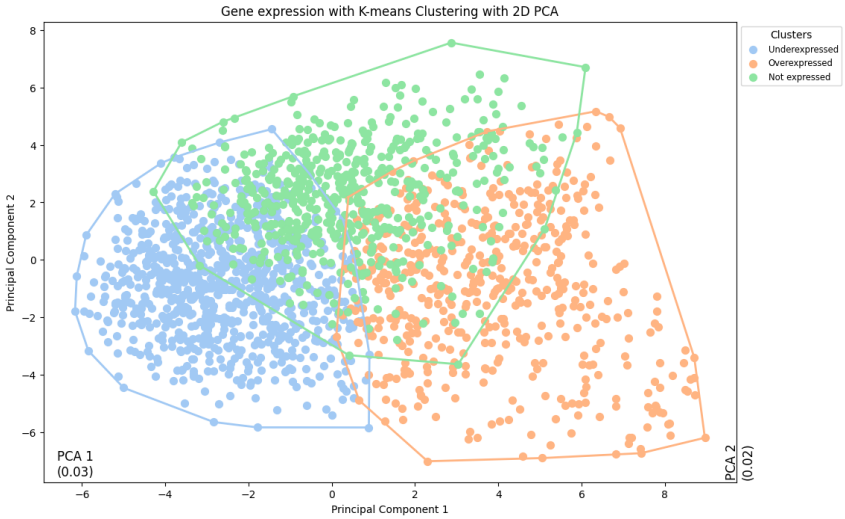


Figure 34. Scatter plot of gene expression with K-means clustering and PCA

In order to explore the correlation between these differentially expressed and clustered genes, we constructed a heatmap to visualize the expression level classification. The purpose of this visualization is to quickly identify patterns of gene expression and potential anomalies across the dataset. (Figure 35)

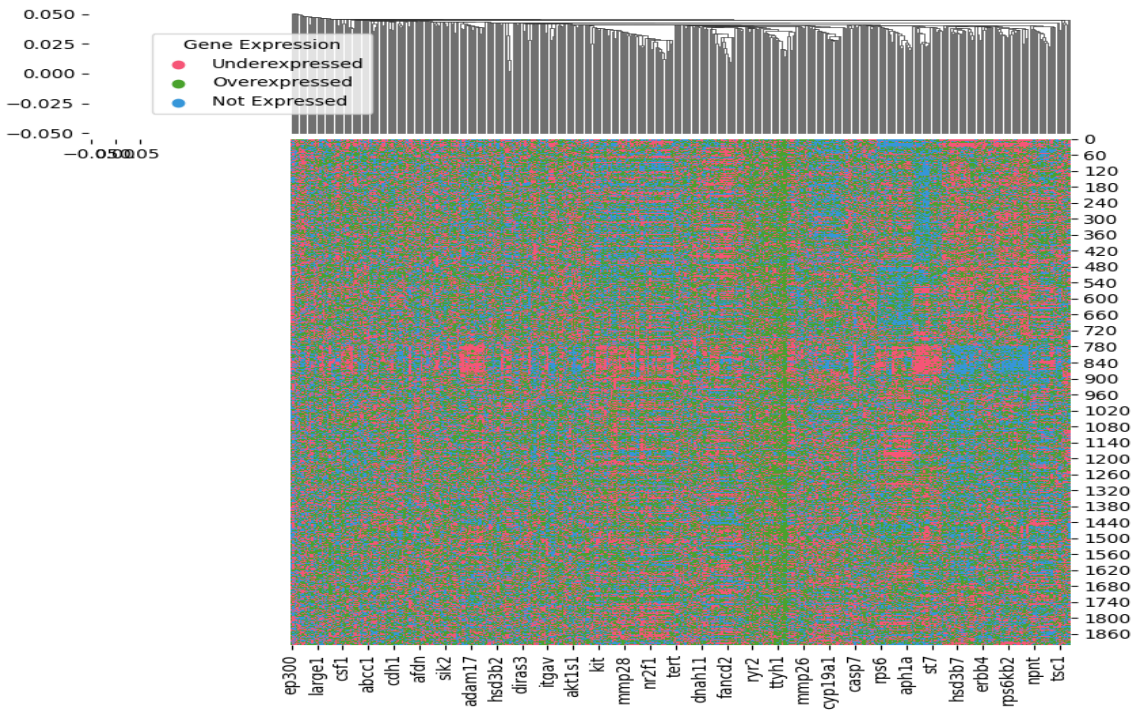


Figure 35. Heatmap of gene expression levels across samples

⇨ Several significant clusters of overexpression are observable across different samples, suggesting that these genes may be key players in disease pathogenesis or potentially useful biomarkers for disease progression and response to treatment. Conversely, the consistently underexpressed genes could be indicative of suppressed pathways that might have pivotal roles in disease mechanisms or therapeutic resistance.

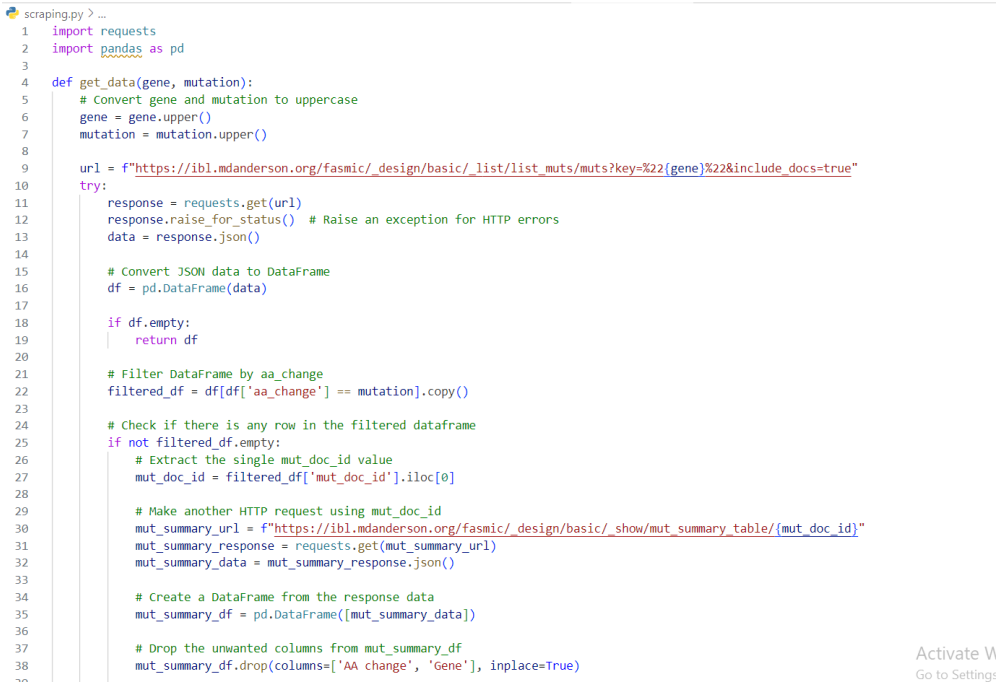
III. Mutation analysis

Gene mutations play a pivotal role in shaping the genetic variation and are crucial elements in understanding both normal biological processes and the onset of various diseases. Our dataset, sourced from the FASMIC (Functional Analysis through Systematic Mining of the Integrated Catalogs) database, provides a comprehensive gene mutation data, facilitating insights into the functional consequences of these mutations.

1. Scraping Mutation Data from FASMIC Database

FASMIC (Functional Analysis through Systematic Mining of Insertion, Deletion, and Copy number variants) is a comprehensive database for functional impact of somatic mutations in cancer.

The scraping process involved navigating through the database, querying for specific mutation-related data, and extracting the results. (Figure 36)



```python
import requests
import pandas as pd

def get_data(gene, mutation):
    # Convert gene and mutation to uppercase
    gene = gene.upper()
    mutation = mutation.upper()

    url = f"https://ibl.mdanderson.org/fasmic/_design/basic/_list/list_muts/muts?key=%22{gene}%22&include_docs=true"
    try:
        response = requests.get(url)
        response.raise_for_status()  # Raise an exception for HTTP errors
        data = response.json()

        # Convert JSON data to DataFrame
        df = pd.DataFrame(data)

        if df.empty:
            return df

        # Filter DataFrame by aa_change
        filtered_df = df[df['aa_change'] == mutation].copy()

        # Check if there is any row in the filtered dataframe
        if not filtered_df.empty:
            # Extract the single mut_doc_id value
            mut_doc_id = filtered_df['mut_doc_id'].iloc[0]

            # Make another HTTP request using mut_doc_id
            mut_summary_url = f"https://ibl.mdanderson.org/fasmic/_design/basic/_show/mut_summary_table/{mut_doc_id}"
            mut_summary_response = requests.get(mut_summary_url)
            mut_summary_data = mut_summary_response.json()

            # Create a DataFrame from the response data
            mut_summary_df = pd.DataFrame([mut_summary_data])

            # Drop the unwanted columns from mut_summary_df
            mut_summary_df.drop(columns=['AA change', 'Gene'], inplace=True)
```

Figure 36. Data scraping

2. Extraction of genes with functional consequences

For each patient in the dataset, we focused on extracting information about gene mutations that had functional consequences.

Functional consequences refer to mutations that alter the structure or function of a gene, potentially leading to changes in protein structure or function, and ultimately impacting cellular processes.

Tableau 4: Highlighted extracted information: feature description

Feature	Description
Gene	The gene where the mutation occurred, identified by its standard gene symbol.
Chromosome Position	The specific location of the mutation on the chromosome, typically represented by chromosome number and base pair position (e.g. chr3:178952085-178952085)
Amino Acid Change	The alteration in the amino acid sequence of the protein encoded by the mutated gene (e.g. H1047R, E17K, Q546K)
Base Change	The specific nucleotide changes in the DNA sequence responsible for the amino acid substitution. (e.g. A>G, C>T).
Variant Classification	The classification of the mutation based on its predicted impact on gene function or disease association. (E.g. missense, nonsense, frameshift).
Variant Type	The type of mutation, indicating the specific nature of the genetic alteration. (e.g. SNV, SNP)
Functional and activation Impact Predictions	Results from algorithms (Consensus call, MCF10A call, MCF10A call, Final call) or computational tools predicting the functional consequences of the mutation (Damaging, Tolerated, possibly damaging, Benign, Disease causing, Neutral)

3. Data Analysis and Visualization

This approach ensures a better representation of the impact of gene mutations on tumor development among the patients, providing a comprehensive overview of the gathered data. (Figure 37)

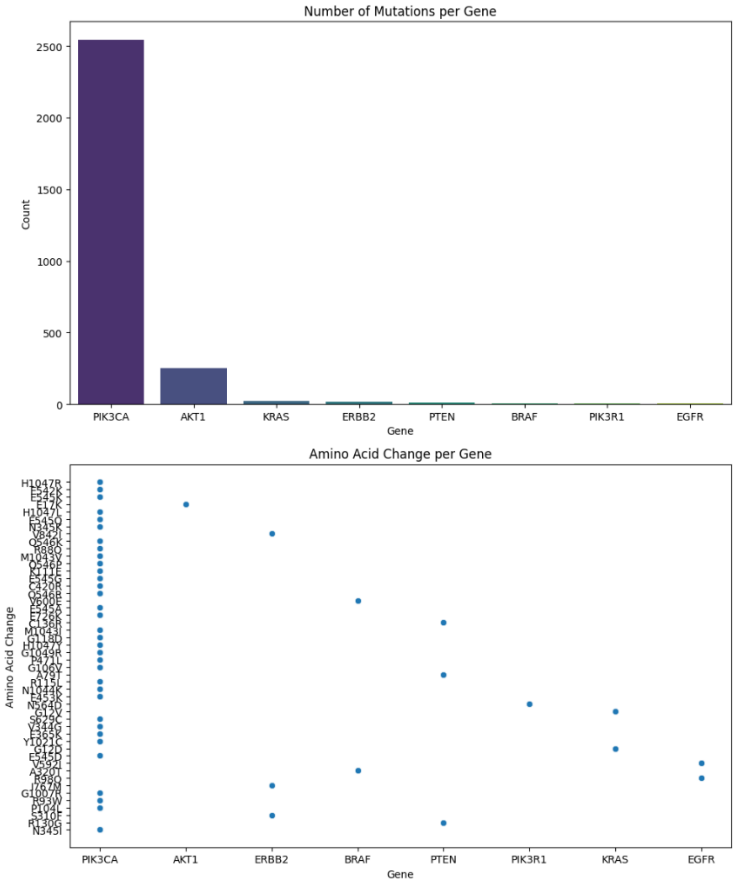
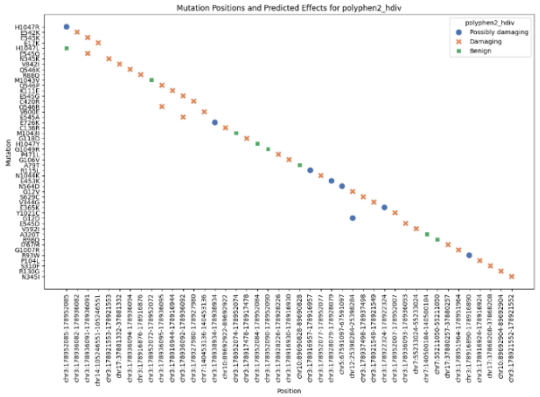
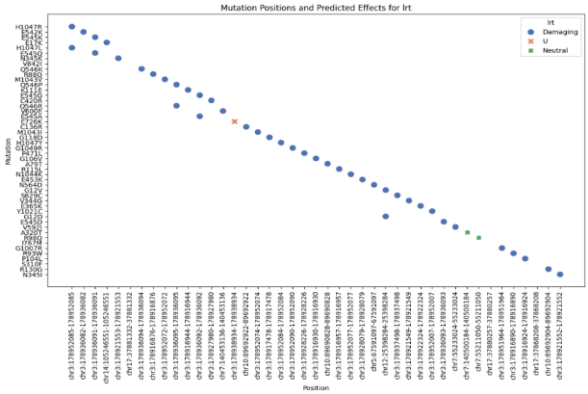


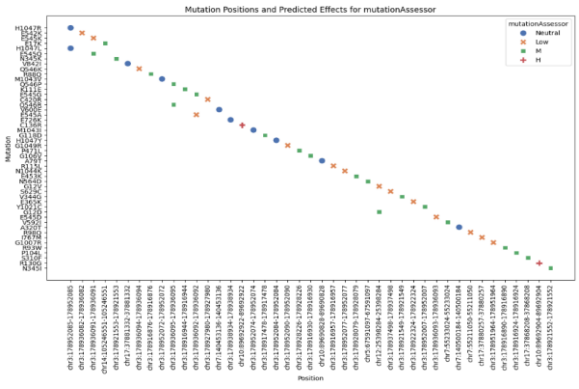
Figure 37. Bar chart of mutations per gene

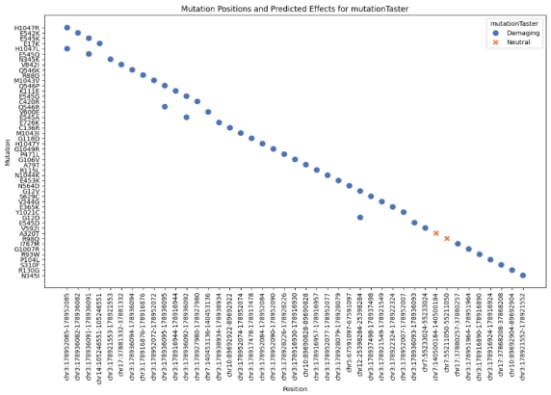
⇒ PIK3CA and EGFR exhibit the highest number of mutations, this underscores their significant role in driving breast cancer growth and progression among the patients in our study.

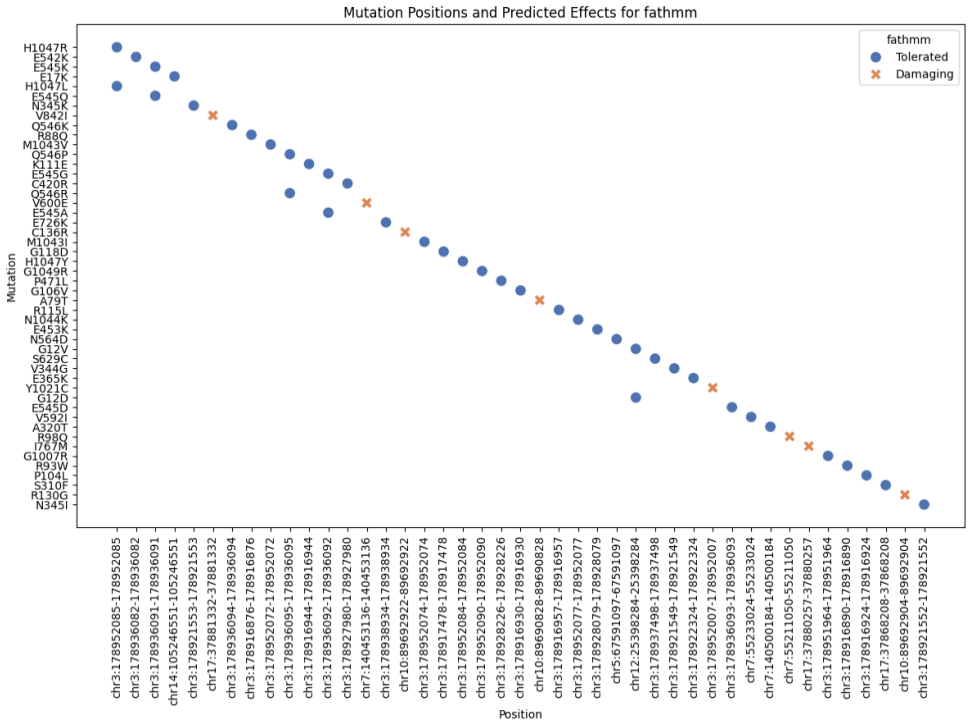
⇒ Mutation hotspots, such as H1047R in PIK3CA, indicate common areas for genetic alterations among the patients, these alterations are pivotal in breast cancer development.


Mutation Positions and Predicted Effects for polyphen2_hdiv
polyphen2_hdiv
Possibly damaging
Damaging
Benign
Mutation
Position


Mutation Positions and Predicted Effects for lrt
lrt
Damaging
U
Neutral
Mutation
Position


Mutation Positions and Predicted Effects for mutationAssessor
mutationAssessor
Neutral
Low
M
H
Mutation
Position


Mutation Positions and Predicted Effects for mutationTaster
mutationTaster
Damaging
Neutral
Mutation
Position


Mutation Positions and Predicted Effects for fathmm
fathmm
Tolerated
Damaging
Mutation
Position

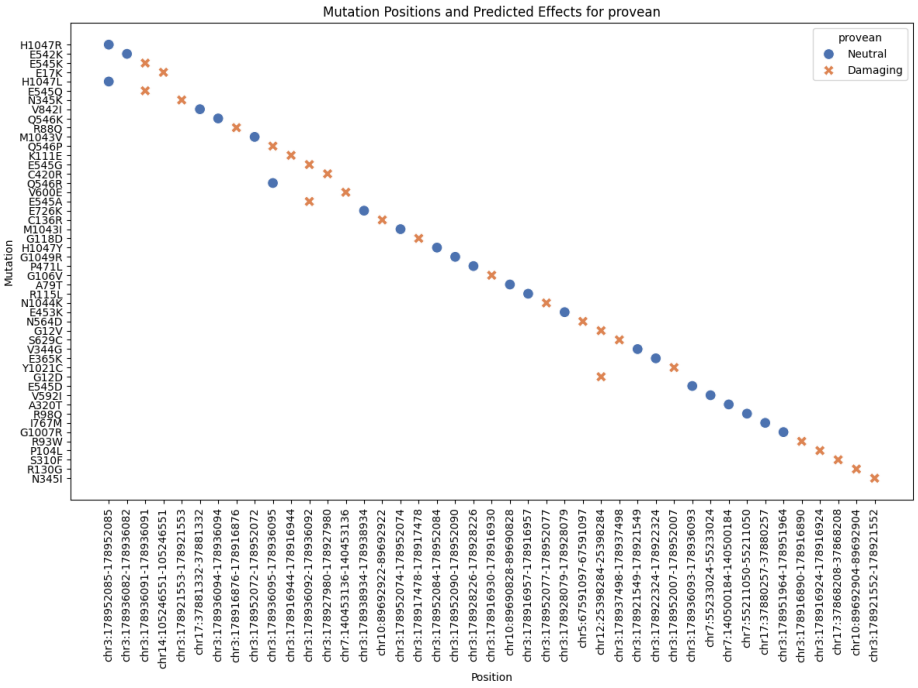


Figure 38. Scatter plots of mutation positions and predicted effects

⇒ Overall, the analysis of the plots suggests a strong convergence among the functional impact Predictions algorithms in predicting a **damaging effect** for most of the genes.

This consensus prediction provides valuable insights into the potential functional consequences of mutations and can guide further investigation into the underlying mechanisms and implications for disease pathology or therapeutic interventions.

Mutation Positions and Predicted Effects for Polyphen2_hdiv:

This plot maps mutation positions along gene sequences and classifies them according to predicted effects (possibly damaging, damaging, benign) based on PolyPhen-2 analysis. It's a vital tool for assessing the potential severity and functional implications of mutations.

Mutation Positions and Predicted Effects for Polyphen2_hvar:

Similar to the previous, this plot also uses PolyPhen-2 analysis but might differ in the version or parameters used for predicting the effects of mutations. It provides an alternative perspective on the potential impact of mutations, complementing the overall mutation analysis.

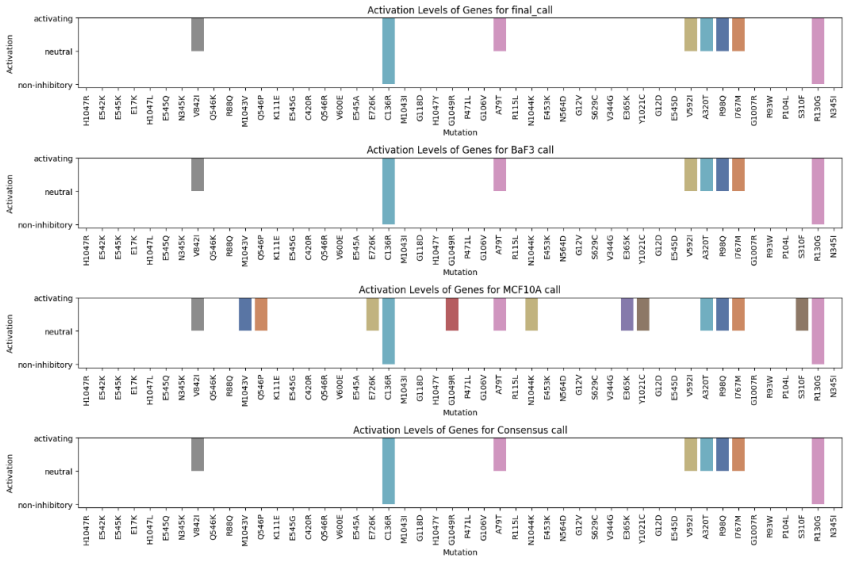


Figure 39. Bar charts of gene activation levels by mutation type

⇒ These plots display the activation status (activating, neutral, non-inhibitory) of various gene mutations across different cell lines (final_call, BaF3, MCF10A, Consensus).

They highlight the diverse functional impacts of mutations indicating whether a mutation is likely activating, neutral, or non-inhibitory in its effect on gene function.

- ⇨ The mutation load across different genes provides insights into their potential as targets for personalized therapy, with implications for both prognostic and therapeutic strategies in oncology.

IV. Pathways data analaysis

After selecting the pertinent genes based on mRNA expression and mutation data from the STRING Network dataset, we visualized and analyzed our findings using Cytoscape software to depict the interactions. Additionally, we utilized Python to visualize the hierarchical clustering of gene correlations. The combined score from STRING's gene interactions output, which ranges from 0 to 1, served as a confidence measure reflecting the reliability of predicted gene interactions. This score aids in prioritizing interactions for further investigation.

1. Network interactions of data

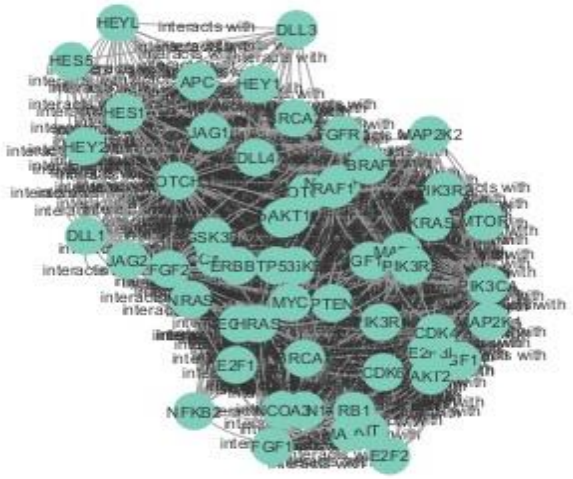


Figure 40. Metabric dataset gene interaction network with Cytoscape

- **Network Overview**

The network visualization shows the interactions between genes related to breast cancer. Each node represents a gene, and each edge represents an interaction between two genes.

- **Node Size and Labels**

Nodes are labeled with gene names, and their size can indicate the number of interactions (degree) or other attributes like expression level or importance in the network.

- **Edges and Interactions**

Edges (lines connecting nodes) represent gene-gene interactions. The density of edges shows the complexity of interactions within the network.

- **Clusters and Modules**

Dense clusters of nodes indicate groups of genes that interact frequently, suggesting they may work together in common pathways or regulatory mechanisms.

- **Key Genes**

Genes with many connections (hubs) are critical in the network, potentially serving as major regulators or central players in breast cancer pathways. For example, BRCA1, TP53, and PIK3CA are well-known genes involved in breast cancer.

2. Heatmap of the Adjacency Matrix

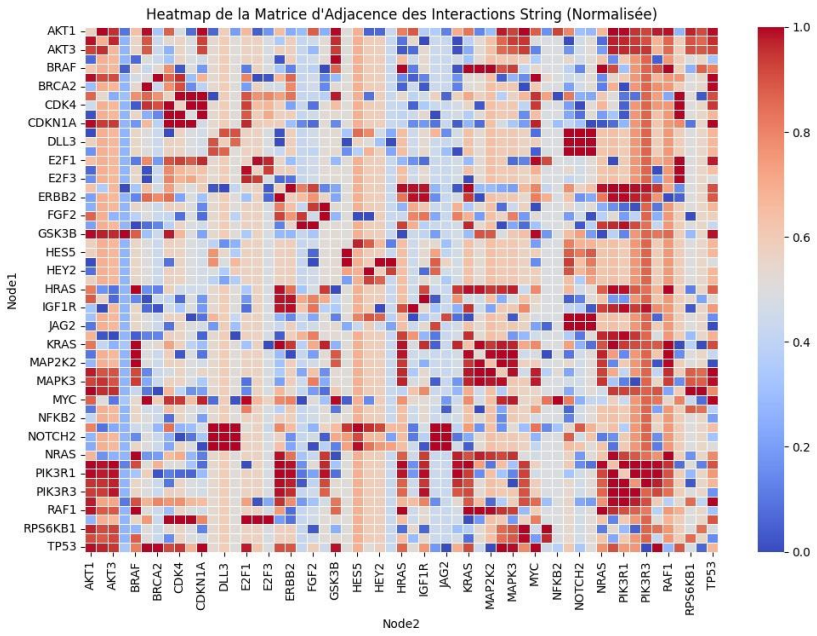


Figure 41. Heatmap of normalized gene interaction matrix

- **Heatmap Overview**

The heatmap displays the normalized adjacency matrix of gene-gene interactions. Each cell in the matrix represents the interaction strength between a pair of genes, with rows and columns corresponding to different genes.

- **Color Scale**

The color scale ranges from blue to red:

Red indicates a high interaction strength or strong positive correlation between genes.

Blue indicates a low interaction strength or negative correlation.

White/Neutral Colors indicate no significant interaction or correlation close to zero.

- **Patterns and Clustering**

The heatmap reveals clusters of genes that interact strongly with each other, shown by blocks of red cells.

Horizontal and vertical lines of red or blue suggest specific genes that interact strongly or weakly with many others.

- **Biological Insights**

Genes that cluster together may be involved in similar biological pathways or processes. For example, clusters involving key breast cancer genes (e.g., BRCA1, BRCA2) can provide insights into the molecular mechanisms of the disease.

3. Visualization of molecular pathways of the data using KEGG database

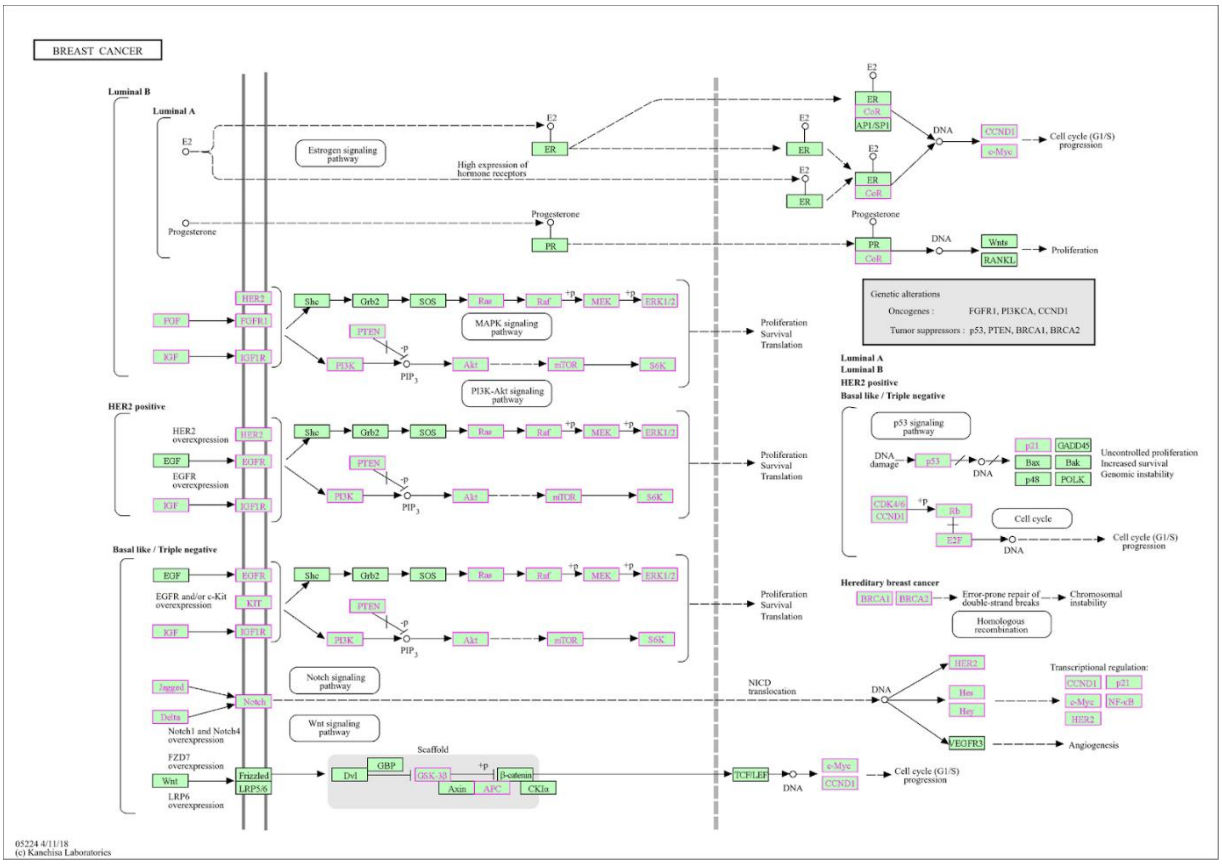


Figure 42. Breast cancer signaling pathways

3.1.Luminal A and B Subtypes (Estrogen Signaling Pathway)

ESR1 (ER): Estrogen receptor alpha (ERα) is critical in Luminal A and B subtypes, facilitating the growth of hormone-responsive breast cancer cells. Its activation by estrogen promotes gene transcription related to cell proliferation and survival. Therapeutic strategies include selective estrogen receptor modulators (SERMs) like tamoxifen and aromatase inhibitors.

CCND1 (Cyclin D1): Cyclin D1 regulates the cell cycle by controlling the transition from the G1 to the S phase. Overexpression of CCND1 is common in ER-positive breast cancer, driving cellular proliferation. Targeting this pathway can involve CDK4/6 inhibitors (e.g., palbociclib).

3.2.HER2 Positive Pathway

HER2 (ERBB2): HER2 is overexpressed in about 20% of breast cancers, leading to aggressive tumor behavior. HER2 activation triggers downstream signaling through the MAPK and PI3K-Akt pathways, promoting cell proliferation and survival. HER2-targeted therapies include trastuzumab (Herceptin) and pertuzumab.

PIK3CA: This gene encodes the p110α catalytic subunit of PI3K, a crucial player in the PI3K-Akt pathway. Mutations in PIK3CA can lead to uncontrolled cell growth. PI3K inhibitors are being explored in clinical trials.

PTEN: PTEN acts as a tumor suppressor by inhibiting the PI3K-Akt pathway. Loss of PTEN function results in enhanced cell survival and proliferation, often observed in HER2-positive cancers.

3.3.Basal-like / Triple Negative Pathway

EGFR (ERBB1): Overexpression or mutation of EGFR is common in triple-negative breast cancer (TNBC), driving tumor growth via the MAPK and PI3K-Akt pathways. EGFR inhibitors, although less effective in breast cancer, are used in other cancer types.

BRCA1/BRCA2: These genes are vital for DNA repair through homologous recombination. Mutations lead to genomic instability and increased cancer risk. PARP inhibitors (e.g., olaparib) are effective in BRCA-mutated cancers.

V. Conclusion

In conclusion, this chapter has provided a comprehensive overview of our data understanding and exploration process. By meticulously analyzing the dataset, we have identified key patterns and insights that are critical for the subsequent stages of our research. The exploratory data analysis (EDA) enabled us to recognize underlying trends, anomalies, and relationships within the data, thereby ensuring a robust foundation for our predictive modeling and hypothesis testing. The insights gained from this phase not only enhance our understanding of the dataset but also guide the direction of our analysis, ensuring that our approach is both informed and methodologically sound. This thorough understanding of the data will be instrumental in driving the accuracy and reliability of our research outcomes.

Chapter 3 :

Predictive modeling and pathways classification

Chapter 3: Predictive modeling and pathways classification

I. Data pre-processing

Data preprocessing is a crucial step in the data preparation process for any data analysis or machine learning task.

It is a critical step in the data science workflow, especially when handling medical datasets such as those related to breast cancer. This phase involves transforming raw data into a clean and structured format, which is essential for building accurate and reliable machine learning models.

Effective preprocessing enhances the quality of the data and ensures that models trained on this data can generalize well to new, unseen instances. (Figure 43)

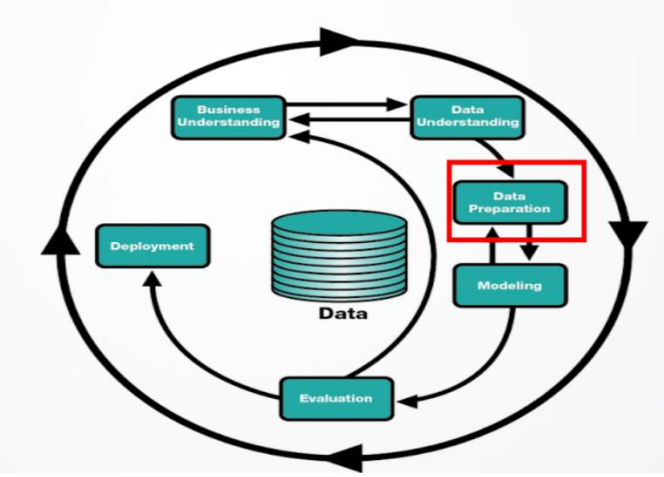


Figure 43. CRISP-DM (Cross-Industry Standard Process for Data Mining) process model

Preprocessing can include dealing with missing values, encoding categorical variables, normalizing numerical features, and addressing class imbalances.

Each of these steps is crucial for maintaining data integrity and preparing the data for effective analysis. For example, handling missing values prevents the introduction of bias, converting categorical variables into a numerical format makes them suitable for machine learning algorithms, and addressing data imbalances ensures that the model's predictions are not skewed.

In this section, we will outline the preprocessing steps applied to our breast cancer dataset, including handling missing values and encoding categorical variables. These steps aim to prepare our data in the best possible way for building robust and accurate predictive models.

1. Data cleaning

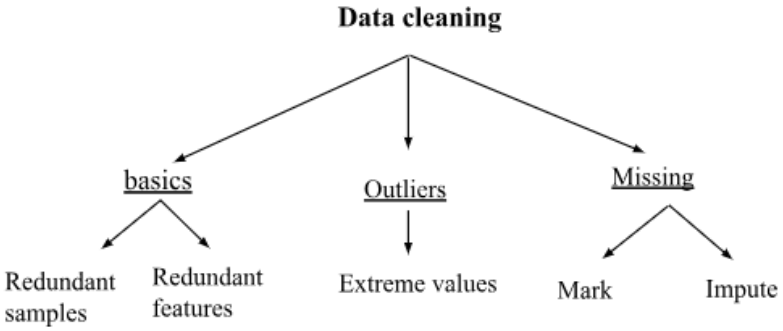


Figure 44. Detailing the data cleaning process

We meticulously addressed data quality issues to ensure robust and reliable analytical outcomes. Our approach involved two primary steps: identifying and handling missing values (NaNs), and removing unnecessary features and samples.

patient_id	age_at_diagnosis	type_of_breast_surgery	cancer_type	cancer_type_detailed	cellularity	chemotherapy	pam50_+_claudin-low_subtype	cohort	er_status_measured_by_ihc	er_status	neoplasm_histologic
0	75.65	MASTECTOMY	Breast Cancer	Breast Invasive Ductal Carcinoma	NaN	0	claudin-low	1.0	Positve	Positive	
2	43.19	BREAST CONSERVING	Breast Cancer	Breast Invasive Ductal Carcinoma	High	0	LumA	1.0	Positve	Positive	
5	48.87	MASTECTOMY	Breast Cancer	Breast Invasive Ductal Carcinoma	High	1	LumB	1.0	Positve	Positive	
6	47.68	MASTECTOMY	Breast Cancer	Breast Mixed Ductal and Lobular Carcinoma	Moderate	1	LumB	1.0	Positve	Positive	
8	76.97	MASTECTOMY	Breast Cancer	Breast Mixed Ductal and Lobular Carcinoma	High	1	LumB	1.0	Positve	Positive	

5 rows × 692 columns

Figure 45. Overview of the dataset on VScode

1.1.Handling missing values

Initially, we conducted a comprehensive examination of our dataset to identify any missing values (NaNs) and potential outliers. NaNs can arise due to various reasons such as data entry errors, sensor malfunctions, or simply the absence of information. Outliers, on the other hand, can result from measurement errors, data recording anomalies, or natural variations in the data. NaNs, if left unaddressed, can significantly skew our analytical results and model performance. (Figure 46)

```
...  type_of_breast_surgery            22
     cancer_type_detailed              15
     cellularity                       54
     er_status_measured_by_ihc         30
     neoplasm_histologic_grade         72
     tumor_other_histologic_subtype    15
     primary_tumor_laterality         106
     mutation_count                    45
     oncotree_code                     15
     3-gene_classifier_subtype        204
     tumor_size                        20
     tumor_stage                      501
     death_from_cancer                  1
     dtype: int64
```

Figure 46. Missing values count before manipulation

To handle the missing values, we opted for the Multiple Imputation by Chained Equations (MICE) technique in R. MICE is a sophisticated method that iteratively imputes missing values by leveraging the relationships among different variables in the dataset.

- **Here's why we chose MICE for our imputation needs**

Multiple Imputation: Unlike single imputation methods that might introduce bias by filling missing values with mean or median, MICE generates multiple plausible imputed datasets. This process accounts for the uncertainty around the missing values.

Chained Equations: MICE uses a series of regression models to predict and impute each missing value. For each variable with missing data, MICE predicts the missing values using other variables as predictors. This iterative process continues until the imputations converge, ensuring that the relationships among variables are maintained.

Flexibility and Robustness: MICE can handle various types of data, including continuous, binary, and categorical variables. This flexibility makes it an ideal choice for complex datasets with different variable types.

Reduction of Bias and Variability: By creating multiple imputed datasets and combining the results, MICE reduces the bias and variability associated with missing data, leading to more accurate and reliable statistical inferences.

- ⇨ In our implementation, we utilized the mice package in R to perform the imputation. The mice function allowed us to specify the number of imputations, the method for each variable, and the number of iterations. The imputed datasets were then analyzed and combined to obtain a single, cohesive dataset ready for subsequent analysis and modeling.

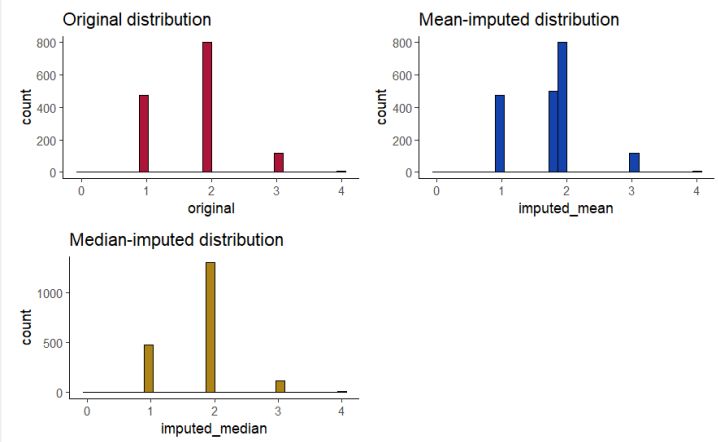


Figure 47. Data distribution comparison after Mice Imputation

- By employing MICE for imputation, we have ensured that our dataset retained its structural integrity by retaining almost the same data distribution after imputation and minimized the biases and inaccuracies introduced by missing data. This rigorous data cleaning process paved the way for more reliable and insightful analytical outcomes, ultimately enhancing the quality and robustness of our models.

1.2.Handling feature and sample redundancy

We ensured the removal of redundant features and samples to streamline the dataset for improved computational efficiency and to avoid overfitting in subsequent modeling. (Figure48)

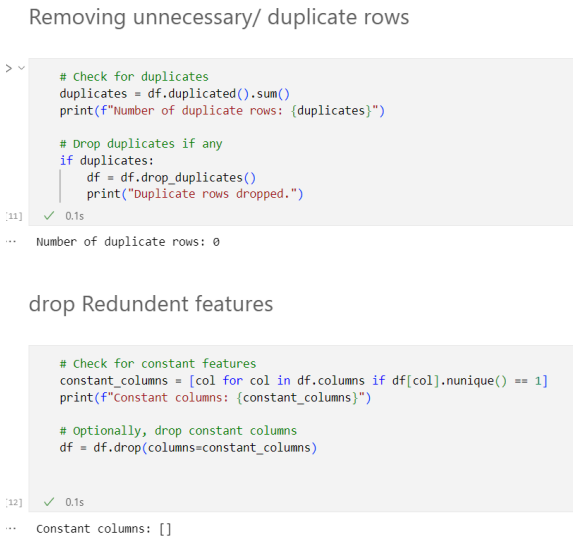


Figure 48. Dropping redundant rows and columns

- **Output**

Our current dataset is complete, with missing values imputed using the MICE technique, redundant features and samples removed, and all variables appropriately processed for further analysis

2. Data transformation

Data transformation is a critical step in the data preprocessing pipeline, particularly for ensuring that different types of data are appropriately formatted for input into machine learning models. This step involves converting data into a suitable format, which enhances the efficiency and effectiveness of the modeling process.

For machine learning models to function optimally, the input data must be numeric. Many algorithms cannot handle categorical data directly and require that such data be transformed into a numeric format. Proper data transformation ensures that the inherent relationships and hierarchies within the data are preserved and effectively utilized by the models.

Given that our dataset comprises both genetic data and clinical data, we adopted a tailored approach to transforming these different data types:

- **Clinical Data**

A lot of the clinical data in our dataset was categorical, necessitating specific encoding techniques to transform it into a numeric format. We divided the categorical data into two types: nominal (without any inherent order or ranking among the categories) and ordinal (meaningful order or ranking among the categories). (Figure 49)

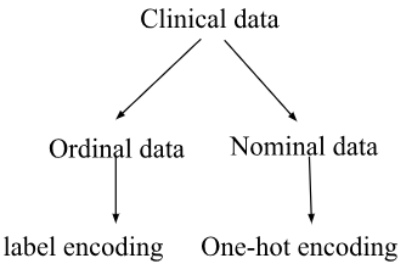


Figure 49. Clinical data encoding

We employed two encoding techniques in our data transformation process:

For nominal data we utilized one-hot encoding, which converts nominal variables into binary columns, ensuring no spurious orderings are introduced.

OneHot encoding for nominal data

```
from sklearn.preprocessing import OneHotEncoder

# Convert columns with mixed types to strings
for col in nominal_cols:
    df[col] = df[col].astype(str)

 #Initialize OneHotEncoder
encoder = OneHotEncoder(handle_unknown='ignore')

# Fit and transform the encoder on categorical columns
encoded_data = encoder.fit_transform(df[nominal_cols]).toarray()

# Create a DataFrame with the encoded variables
encoded_cols = list(encoder.get_feature_names_out(nominal_cols))
df_encoded = pd.DataFrame(encoded_data, columns=encoded_cols)

# Drop the original categorical columns and join with the encoded columns
df = df.drop(nominal_cols, axis=1)
df = pd.concat([df.reset_index(drop=True), df_encoded.reset_index(drop=True)], axis=1)

# Now, your DataFrame df has all nominal  columns encoded and this data should be numeric.
```

Figure 50. One-hot encoding for nominal data

For ordinal data we employed label encoding, assigning a unique integer to each category based on its order.

This preserves the ordinal nature of the data, allowing the model to recognize and utilize the relative ordering among categories.

label encoding for all ordinal data

```
from sklearn.preprocessing import LabelEncoder

# Convert all ordinal columns to type string
df[ordinal_cols] = df[ordinal_cols].astype(str)

#  Initialize LabelEncoder
le = LabelEncoder()

#  Apply LabelEncoder to each ordinal column
for col in ordinal_cols:
    df[col] = le.fit_transform(df[col])
```

Figure 51. Label encoding for ordinal data

- **Genetic Data**

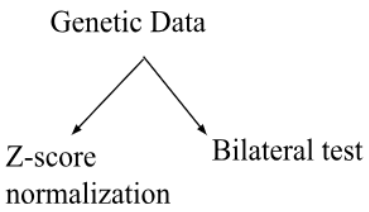


Figure 52. Genetic data transformation

Z-Score Normalization

In our study, the genetic data utilized was already prepared and normalized as z-scores. Z-score normalization, also known as standardization, is a crucial preprocessing step in data analysis, especially when dealing with variables that measure different attributes. This method transforms the data to have a mean (μ) of zero and a standard deviation (σ) of one. The formula used for this transformation is:

$$\mathbf{Z}=\frac{(\mathbf{X}-\boldsymbol{\mu})}{\boldsymbol{\sigma}}$$

X is the original value of a data point. By applying z-score normalization, each genetic feature is scaled to ensure comparability across different scales and units. This is particularly important in genetic studies where expression levels can vary widely between genes.

Bilateral Testing for Significance

Before delving into the advanced analysis of our gene expression data, we noticed that some values take a number very close to 0 such as 0.0024, and considering it's important our Cancer pathway classification model that we classify gene expressions using this Z-score classification:

Z-score = 0 ⇒ Gene normally expressed

Z-score > 0 ⇒ Gene overexpressed

Z-score < 0 ⇒ Gene underexpressed

➔We decided to employ bilateral testing to determine the statistical significance of the observed gene expression levels. This method is critical when assessing whether the normalized expression levels, particularly those close to zero, are significantly different from the expected mean (which is zero in the case of z-score normalized data).

Each gene expression column was subjected to a test where both the maximum and minimum values were evaluated.

For each gene, the maximum and minimum z-score values are identified from the dataset.

In the next step, we conducted a two-tailed test (bilateral test) for each gene to evaluate if these extreme values are statistically significant.

The hypothesis testing framework used is as follows:

Null Hypothesis (H_0): The gene expression level is not significantly different from zero.

Alternative Hypothesis (H_1): The gene expression level is significantly different from zero.

P-Value Calculation: The p-value is calculated for each extreme value. This p-value represents the probability of observing a gene expression level at least as extreme as the one detected under the null hypothesis. A low p-value (typically <0.05) indicates that the observed expression level is statistically significant.

⇒ This bilateral testing allows us to discern the significance of gene expression Z-scores, particularly those that are near the zero point but may still influence the biological interpretation and subsequent analysis.

```python
from scipy.stats import norm

# Suppose data_expression3 est votre DataFrame contenant les données d'expression génique

# Définir un seuil de significativité
seuil_significativite = 0.05

# Fonction pour déterminer la significativité et la direction de chaque valeur de coefficient z
def determine_significance_and_direction(data_expression3, seuil_significativite):
    significatif = pd.DataFrame(index=data_expression3.index, columns=data_expression3.columns)

    for gene in data_expression3.columns:
        for patient in data_expression3.index:
            # Calculer la p-valeur associée à chaque valeur de coefficient z
            p_value = norm.sf(abs(data_expression3.at[patient, gene])) * 2  # Test bilatéral

            # Déterminer si la valeur est significative ou non
            if p_value < seuil_significativite:
                if data_expression3.at[patient, gene] < 0:
                    significatif.at[patient, gene] = "Down"
                else:
                    significatif.at[patient, gene] = "Up"
            else:
                significatif.at[patient, gene] = "Not Significant"

    return significatif

# Appeler la fonction pour déterminer la significativité et la direction
significatif_df = determine_significance_and_direction(data_expression3, seuil_significativite)
```

Figure 53. Python Function for Statistical Significance Testing of Gene Expression

⇨ After cleaning and transforming the data, we now have a refined dataset ready for analysis. Missing values have been imputed using the MICE technique, redundant features and samples have been removed, and all variables have been appropriately transformed. Categorical variables have been encoded, and numerical variables have been normalized. The dataset is now well-structured and prepared for statistical modeling and further analysis.

II. Modeling and classification of cancer patients

1. Chemotherapy response prediction

Using a selection of the data we preprocessed, we have developed a machine learning model designed to predict patient response to chemotherapy for breast cancer. The key outcome measure, or target variable (y), is **cellularity**, which refers to the level of residual tumor cells present after chemotherapy treatment. Cellularity is a crucial biological indicator as it reflects the effectiveness of the chemotherapy in reducing the tumor burden.

Low Cellularity: Indicates a minimal presence of residual tumor cells.

Moderate Cellularity: Indicates a moderate presence of residual tumor cells.

High Cellularity: Indicates a significant presence of residual tumor cells.

1.1.Model development

1.1.1. Model creation

To simplify the predictive modeling, we categorized the cellularity levels into two distinct groups representing the patient's response to chemotherapy.

- **Favorable Response (Good Response)**

  Low Cellularity

  Moderate Cellularity

Patients with low or moderate cellularity are considered to have responded well to the chemotherapy, as these levels indicate a significant reduction in tumor cells ⇒ A better prognosis

- **Unfavorable Response (Poor Response)**

  High Cellularity

Patients with high cellularity are considered to have a poor response to chemotherapy, as this level indicates that a substantial number of tumor cells remain ⇒ a need for alternative therapeutic strategies.

The primary objective of our machine learning model is to predict whether a patient will have a favorable or unfavorable response to chemotherapy based on pre-treatment features.

The model will output :

1 for a favorable response (low/ moderate cellularity)
0 for an unfavorable response (high cellularity)

In this study, we are more inclined to initially employ a Random Forest model for predicting the response to chemotherapy.

Random Forests are selected for their ability to handle both numerical and categorical data, capture complex nonlinear relationships inherent in biological data, and provide a feature importance measure for interpretability.

However, it is important to note that this choice is not final. To ensure a thorough analysis, we explored and compared the performance of multiple models, including Logistic Regression, Decision Trees, Support Vector Machines, and Neural Networks.

By evaluating various models, we aimed to gain a comprehensive understanding of the dataset, potentially revealing insights that may not be apparent with a single model.

1.1.2. Model Evaluation for Predicting Chemotherapy Response

To identify the most accurate predictive model for chemotherapy response based on the feature 'cellularity', we tested several machine learning algorithms.

The accuracy scores for each model are as follows:

```
model_scores
```

Python

```
{'Logistic Regression': 0.6708860759493671,
 'K-Nearest': 0.8734177215189873,
 'Random Forest': 0.8734177215189873,
 'Decision Tree': 0.8227848101265823,
 'SVM': 0.8734177215189873,
 'Naive Bayes': 0.7341772151898734}
```

Figure 54. Model accuracy evaluation across various machine learning algorithms

- **Results**

Table 5: Comparative Analysis of ML Models for Predicting Chemotherapy Response

Model	Accuracy	Performance Summary
Logistic Regression	0.6709	Performed the worst among the tested models, achieving an accuracy of 67.09%, indicating that it may not capture the complexities of the dataset effectively for predicting chemotherapy response based on 'cellularity'.
K-Nearest Neighbors (KNN)	0.8734	Performed very well with an accuracy of 87.34%. This suggests that the model is good at predicting the response to chemotherapy by considering the proximity of data points in the feature space.
Random Forest	0.8734	Similar to KNN, it achieved a high accuracy of 87.34%. This ensemble method leverages multiple decision trees to improve prediction accuracy and handle overfitting effectively.
Decision Tree	0.8228	Accuracy of 82.28%, which is slightly lower than KNN and Random Forest but still reasonably high. Indicates that a single tree can perform well, but it might not be as robust as ensemble methods like Random Forest.
Support Vector Machine (SVM)	0.8734	Accuracy of 87.34%, indicating that it is very effective for this classification problem. Works well with high-dimensional spaces and can be effective in scenarios where the number of dimensions exceeds the number of samples.
Naive Bayes	0.7342	Accuracy of 73.42%, which is moderate compared to the other models. While it is a simple and fast model, it might not capture the nuances in the data as well as the more complex models.

⇒ The final decision was to implement a random forest model.

1.1.3. Implementing Random Forest

```
# Separate features and target variable
x = data.drop("cellularity", axis=1)
y = data["cellularity"]

# Split the data into training and testing sets
xtrain, xtest, ytrain, ytest = train_test_split(x, y, test_size=0.2 ,  shuffle=True, random_state=42)
```

```
# Initialize and fit the Random Forest model
rf = RandomForestClassifier()
rf.fit(xtrain, ytrain)

# Predict on the testing set
y_predicted_testing = rf.predict(xtest)
```

Figure 55. Fitting and testing the model

1.1.4. Random Forest Model evaluation

Model evaluation is a critical step in the machine learning workflow. It involves assessing the performance of a trained model using various metrics to determine how well it generalizes to unseen data. This process ensures that our model is robust, reliable, and suitable for the intended application. (Figure 56)

```
# Evaluate testing results
print("RF")
print("Testing Classification Report:")
print(classification_report(ytest, y_predicted_testing))
print("Testing Accuracy:", accuracy_score(ytest, y_predicted_testing))

# Predict on the training set
y_predicted_training = rf.predict(xtrain)

# Evaluate training results
print("\nTraining Classification Report:")
print(classification_report(ytrain, y_predicted_training))
print("Training Accuracy:", accuracy_score(ytrain, y_predicted_training))
```

Figure 56. Evaluating the testing and training results

1.1.5. Model optimization

Model optimization in machine learning involves refining a model to achieve better performance. This process includes tuning hyperparameters, feature selection, and model selection to enhance the accuracy, efficiency, and generalizability of the model. Here's a brief overview of the key concepts and methods used in model optimization.

1.1.5.1. Feature Selection

- Partial dependence plot

Partial dependence plots (PDPs) illustrate the marginal effect of a feature or a set of features on the predicted outcome of our machine learning model. They provide insights into how the model's predictions change when the feature values vary, while keeping all other features constant.

We began our optimization process by generating partial dependence plots. (Figure 57)

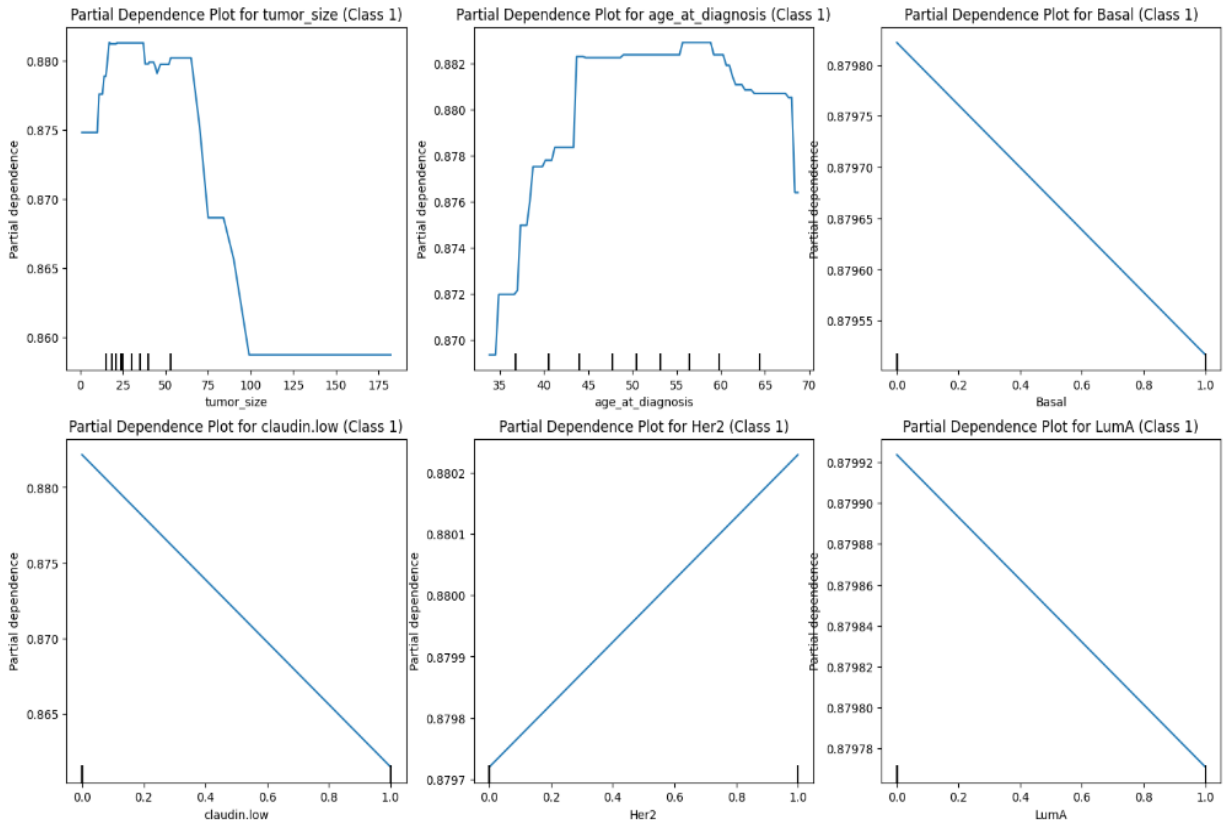


Figure 57. Partial Dependence Plots for Individual Features Influencing Predicted Probability of Class 1

⇒ The partial dependence plots reveal that some features have linear effects while others have non-linear effects on the predicted probability of the target class. Identifying and understanding these relationships helps in feature selection and model interpretation, guiding the optimization process. By removing features with minimal impact or redundant features, we can simplify the model and potentially improve its performance and interpretability.

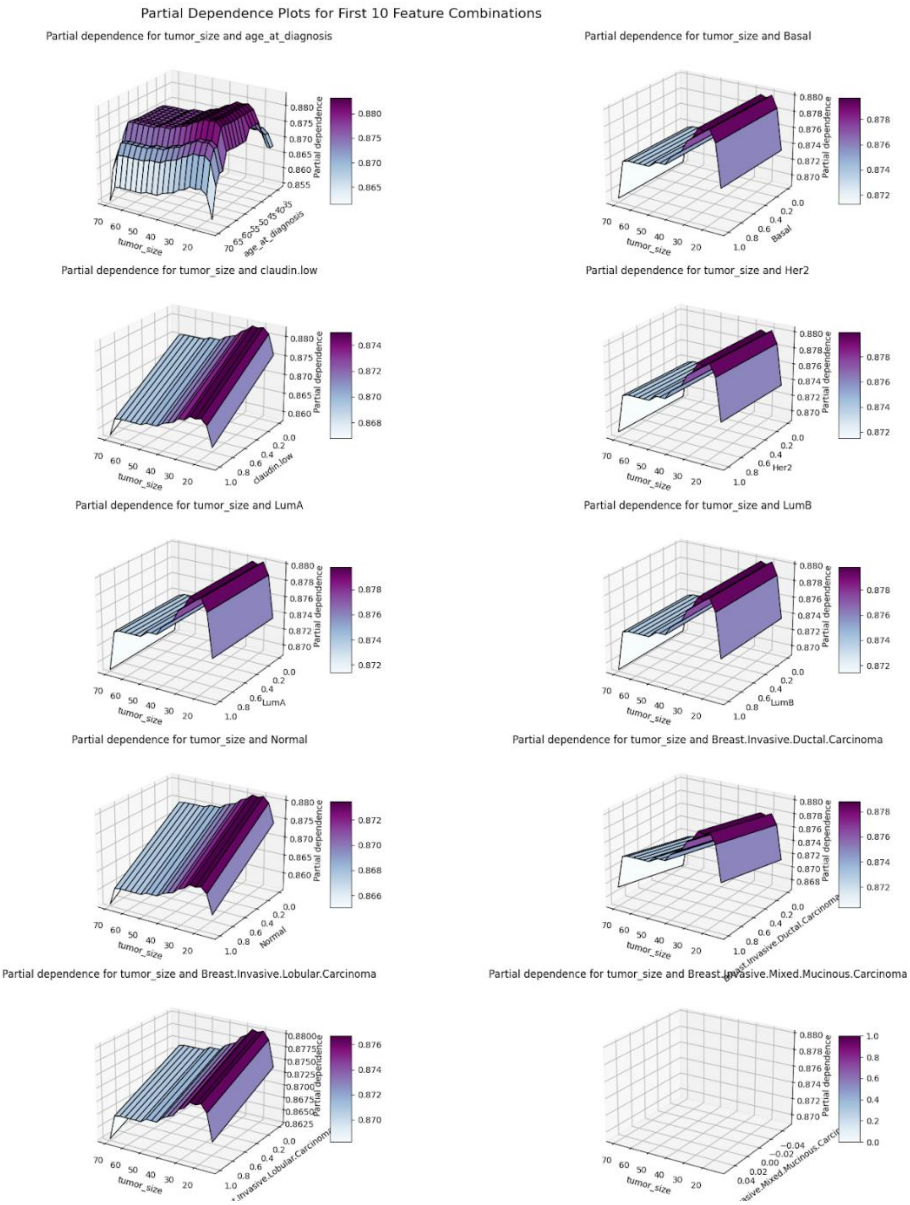


Figure 58. 3D Partial Dependence Plots for Individual Features Influencing Predicted Probability of Class 1

⇒ The 3D partial dependence plots provide a comprehensive view of how tumor size interacts with different breast cancer subtypes to affect the predicted probability of the target class. By analyzing these interactions, we can identify critical features and their combined effects, which helps in feature selection and model interpretation. This detailed understanding guides the optimization process, ensuring that our model captures the complex relationships within the data.

- ❖ SHAP (SHapley Additive exPlanations)

Force plot provides a visualization of feature contributions to the prediction of a particular instance.

Figure 59. SHAP Force Plot for feature contributions to model prediction

Base Value (0.1136): This is the average model output over the training dataset, which serves as a baseline prediction. The SHAP values show how much each feature contributes to deviating from this base value for the specific instance.

Feature Contributions: Each feature's SHAP value is represented as a horizontal bar, where red bars indicate positive contributions (pushing the prediction higher) and blue bars indicate negative contributions (pushing the prediction lower).

➔Specific Feature Contributions we found:

Table 6: Quantitative Impact of Gene Features on Predictive Model Outcomes

Feature	Value	Contribution Type
igf1	2.849	Positive
rb1	1.534	Positive
apc	1.657	Positive
e2f2	-1.540	Positive
mtor	-1.927	Positive
hey1	-1.855	Negative
pik3r3	-0.329	Negative
jag1	-1.493	Negative
cdk6	0.369	Negative
hey1_2	-0.209	Negative
hras	0.983	Negative

⇒ We have identified several features that have limited impact on the performance of our model. Specifically, the features "chemotherapy," "NC cancer type," and "Metaplastic Breast Cancer" were found to have negligible influence on our model's predictive capabilities.

In order to enhance the performance and optimize our model, we have decided to drop these features from our dataset. By doing so, we aim to streamline the model and improve its efficiency without sacrificing predictive accuracy.

1.1.5.2. Parameter tuning and class distribution assessment

In our model optimization process, we aimed to enhance the performance of our Random Forest classifier by employing two key techniques: grid search and stratified k-fold cross-validation.

- Grid Search

Grid search is a systematic method for tuning hyperparameters in machine learning models. It involves specifying a range of values for each hyperparameter and exhaustively evaluating every possible combination to identify the best set. The primary goal is to find the hyperparameters that result in the best model performance based on a chosen evaluation metric.

In our case, we defined a grid of hyperparameters for the Random Forest classifier, including the number of trees, tree depth, minimum samples required for splits and leaves, and whether to use bootstrap sampling. By exploring various combinations of these hyperparameters, we aimed to identify the optimal configuration that maximizes the model's predictive accuracy. (Figure 60)

Figure 60. Hyperparameter tuning with Grid Search

- Stratified K-Fold Cross-Validation

Stratified k-fold cross-validation is an advanced form of cross-validation that ensures each fold of the dataset has a similar class distribution to the overall dataset. This is particularly important for imbalanced datasets, as it prevents any single fold from being unrepresentative of the broader dataset.

In this method, the data is split into k subsets (folds). The model is trained on k-1 folds and validated on the remaining fold. This process is repeated k times, with each fold serving as the validation set once. The results are then averaged to provide a robust estimate of the model's performance. (Figure 61)

```
BOLD = '\033[1m'
END = '\033[0m'
# using a stratfied k fold because we need the distribution of the to classes in all of the folds to be the same.
kfold = StratifiedKFold(n_splits=5, shuffle=True, random_state=42)
```
Python

Figure 61. Implementing K-fold stratifier to adjust classes distribution

- Our Combined Approach

By integrating grid search with stratified k-fold cross-validation, we ensured a comprehensive and reliable optimization process. Grid search evaluated all possible combinations of hyperparameters, while stratified k-fold cross-validation provided an unbiased estimate of model performance by maintaining class distribution consistency across folds. (Figure 62)

```
# Cross-validated recall
cv_recall = cross_val_score(model,
                            x,
                            y,
                            cv=10,
                            scoring='recall')
cv_recall = np.mean(cv_recall)
```
✓ 8.1s

```
print(f"Cross-validated accuracy: %{cv_acc * 100}")
print(f"Cross-validated precision: {cv_precision}")
print(f"Cross-validated recall: {cv_recall}")
```
✓ 0.0s

```
Cross-validated accuracy: %88.3076923076923
Cross-validated precision: 0.8850539811066127
Cross-validated recall: 0.9971428571428571
```

Figure 62. Displaying cross-validated model evaluation metrics

- Baseline Accuracy

Baseline accuracy serves as a reference point to compare the performance of our machine learning model. It is typically calculated based on the most frequent class in the dataset. In our case, we calculated the baseline accuracy using the 'cellularity' feature.

The baseline accuracy for the 'cellularity' feature is 0.882952, indicating that the most frequent class accounts for approximately 88.3% of the instances.

Any model we develop should aim to surpass this baseline accuracy to be considered effective. (Figure 62)

```
print('Baseline accuracy:' )
print(data["cellularity"].value_counts()/data["cellularity"].count())
```
Python

```
Baseline accuracy:
cellularity
1    0.882952
0    0.117048
Name: count, dtype: float64
```

Figure 63. Calculating baseline accuracy to reference model performance

1.2.Model Performance

After training and evaluating multiple machine learning models, the **Random Forest** classifier achieved the best performance. The Random Forest model demonstrated a high accuracy, correctly predicting the

response to chemotherapy with an accuracy of **0.88**. This indicates that the model can reliably distinguish between patients with a favorable response (low to moderate cellularity) and those with an unfavorable response (high cellularity) to chemotherapy.

2. Cancer pathways classification model

The main objective of our model is to cluster patients based on gene expression data, PAM50 subtypes, and mutation profiles. This clustering process categorizes patients based on molecular pathways altered by gene expression patterns (ERK signaling, PI3K signaling, phenomenon linked to genetic changes in BRCA1 or BRCA2), aiming to identify specific molecular alterations that can guide personalized treatment strategies. By incorporating additional data such as PAM50 subtypes and mutation information, we investigate whether these features enhance the predictive accuracy of the model. Ultimately, this approach provides a comprehensive understanding of the molecular basis of cancer in each patient, facilitating more targeted and effective therapeutic interventions.

2.1.Model development

2.1.1. Model creation

We focused on the most pertinent molecular pathways and molecular phenomena in breast cancer, so we break the clustering creation into small parts:

- **Define Relevant Columns for Each Cluster**

- cluster1_cols = ["NRAS", "HRAS", "KRAS", "ARAF", "BRAF", "CRAF", "RAF1", "MAP2K1", "MAP2K2", "MAPK1", "MAPK3"]
- cluster2_cols = ["EGFR", "PIK3CA", "PIK3CB", "PIK3CD", "AKT1", "AKT2", "AKT3", "MTOR", "RPS6KB1", "RPS6KB2", "EGF", "ERBB2", "FGFR1", "FGFR2", "PTEN"]
- cluster3_cols = cluster1_cols
- cluster4_cols = cluster2_cols
- cluster5_cols = ["BRCA1", "BRCA2"]
- cluster6_cols = cluster5_cols

Here, columns representing specific genes are grouped into clusters. Each list contains the names of genes relevant to a specific biological pathway or a set of related pathways.

- **Function to Assign Clusters**

```
# Function to assign clusters
def assign_cluster(row):
```

Figure 64. Clustering Python function

This function will be applied to each row (representing a patient) in the DataFrame to determine the cluster assignment based on gene expression data.

- **Count Up and Down Expressions for Each Cluster**

```
# Count the number of "Up" and "Down" for each group of columns
cluster1_up_count = sum(row[col] == "Up" for col in cluster1_cols if col in row)
cluster1_down_count = sum(row[col] == "Down" for col in cluster1_cols if col in row)

cluster2_up_count = sum(row[col] == "Up" for col in cluster2_cols if col in row)
cluster2_down_count = sum(row[col] == "Down" for col in cluster2_cols if col in row)

cluster3_up_count = cluster1_up_count  # Reuse the counts from cluster1
cluster3_down_count = cluster1_down_count  # Reuse the counts from cluster1

cluster4_up_count = cluster2_up_count  # Reuse the counts from cluster2
cluster4_down_count = cluster2_down_count  # Reuse the counts from cluster2

cluster5_up_count = sum(row[col] == "Up" for col in cluster5_cols if col in row)
cluster6_down_count = sum(row[col] == "Down" for col in cluster6_cols if col in row)
```

Figure 65. Counting gene expression changes across clusters

For each cluster, the function counts how many genes are up-regulated ("Up") or down-regulated ("Down") for the current patient.

- **Determine the Cluster Based on Conditions**

```
# Determine the cluster according to given conditions
if cluster1_up_count > cluster1_down_count:
    if cluster5_up_count > 0:
        return 7
    return 1
elif cluster2_up_count > cluster2_down_count:
    if cluster5_up_count > 0:
        return 7
    return 2
elif cluster3_down_count > cluster3_up_count:
    if cluster6_down_count > 0:
        return 8
    return 3
elif cluster4_down_count > cluster4_up_count:
    if cluster6_down_count > 0:
        return 8
    return 4
elif cluster5_up_count > 0:
    return 5
elif cluster6_down_count > 0:
    return 6
else:
    return 0
```

Figure 66. Decision logic for cluster assignment based on gene expression counts

- **Cluster 1 & Cluster 3**

Genes NRAS, HRAS, KRAS, ARAF, BRAF, CRAF, RAF1, MAP2K1, MAP2K2, MAPK1, MAPK3

Pathway: RAS-RAF-MEK-ERK (MAPK) signaling pathway

Function: This pathway is integral to cell proliferation, differentiation, and survival. It frequently harbors oncogenic mutations and is a common target in cancer therapies.

Implications: Abnormalities within this pathway can result in uncontrolled cell growth and cancer progression, making it critical for cancer research and treatment.

- **Cluster 2 & Cluster 4**

Genes: EGFR, PIK3CA, PIK3CB, PIK3CD, AKT1, AKT2, AKT3, MTOR, RPS6KB1, RPS6KB2, EGF, ERBB2, FGFR1, FGFR2, PTEN

Pathway: PI3K-AKT-mTOR signaling pathway

Function: This pathway plays a significant role in regulating cell growth, proliferation, motility, and survival, as well as metabolism and immune response.

- **Cluster 5 & Cluster 6**

Genes: BRCA1, BRCA2

Pathway: DNA repair pathway

Function: BRCA1 and BRCA2 are essential for maintaining genomic stability through the repair of DNA double-strand breaks via homologous recombination.

➔ we noticed that cluster 8 have few counts which can affect the prediction so we removed it from data.

- **Visualisation of the clusters correlated with genes and clinical data**

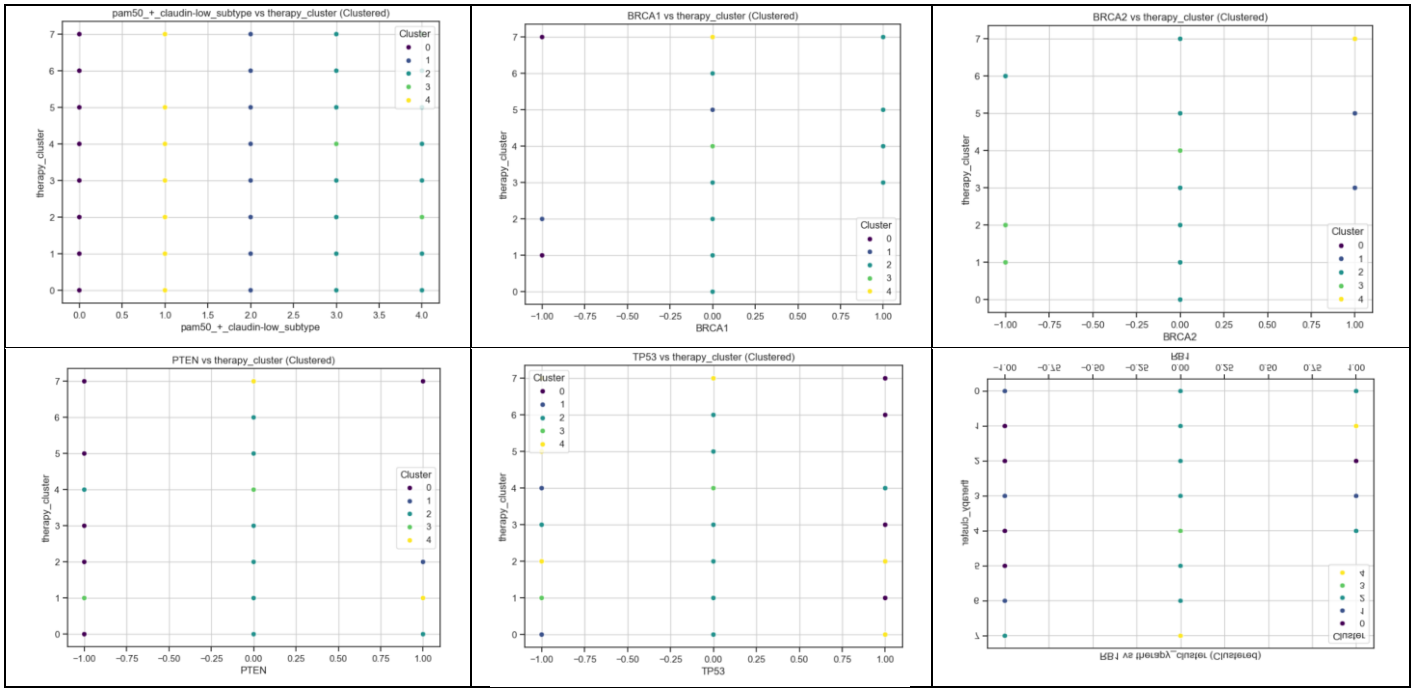


Figure 67. Visualizations of the clusters correlated with genes and clinical

The provided scatter plots show the relationship between specific features and the `therapy_cluster` variable, with the clusters indicated by different colors.

- **pam50_+_claudin-low_subtype vs therapy_cluster (Clustered)**

The x-axis represents the `pam50_+_claudin-low_subtype` feature, and the y-axis represents the `therapy_cluster`.

Each point corresponds to a data sample, and the color indicates the cluster to which that sample belongs.

The plot shows that the `pam50_+_claudin-low_subtype` feature takes discrete values, and there are clear vertical separations for different values of `therapy_cluster`.

The type 1 encoded of pam50 which is her2+ shows that there is no molecular pathway of cluster 6 of the pathways classification which is “Down genes in all pathways”.

- **BRCA1 vs therapy_cluster (Clustered)**

The x-axis represents the `BRCA1` feature, and the y-axis represents the `therapy_cluster`.

The `BRCA1` feature appears to have values ranging from -1 to 1, and the data points are clustered in distinct vertical lines for each `therapy_cluster`.

- **BRCA2 vs therapy_cluster (Clustered)**

The x-axis represents the `BRCA2` feature, and the y-axis represents the `therapy_cluster`.

The `BRCA2` feature also ranges from -1 to 1, with data points forming distinct vertical lines for each `therapy_cluster`.

- **Visualisation of clusters with PCA**

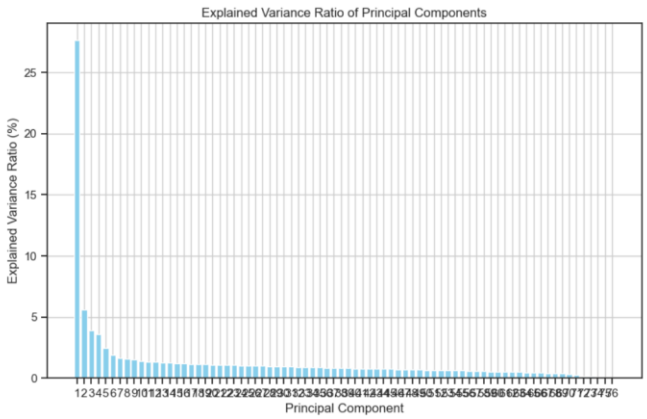


Figure 68. Clusters visualization- PCA

The plot provided shows the explained variance ratio for each principal component (PC) in a PCA analysis.

- **Key Observations**

First Principal Component: The first principal component (PC1) captures the most variance, explaining over 25% of the total variance in the data. This suggests that PC1 is the most informative single dimension in the dataset.

Diminishing Returns: The subsequent principal components (PC2, PC3, etc.) capture progressively less variance. The explained variance drops sharply after the first few components.

Cumulative Variance: After a certain number of components, the explained variance ratio of additional components becomes very small, indicating that they contribute little additional information.

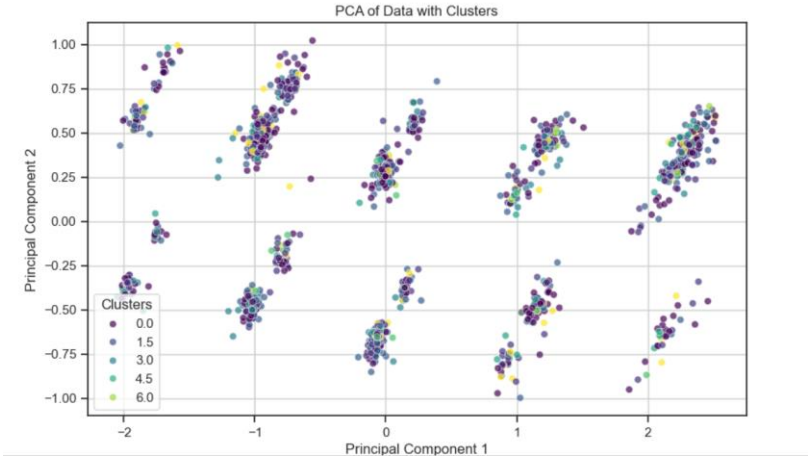


Figure 69. PCA plot of data clusters based on gene expression

The provided plot shows the results of a Principal Component Analysis (PCA) performed on the dataset with the clusters indicated by different colors (the color indicates the cluster to which each K-means cluster belongs) before using the sequential feature selected for our model.

- **Cluster Interpretation**

The clusters overlap significantly, indicating that the separation between clusters is not very distinct. This overlap suggests that while the clustering algorithm has grouped the data into clusters, the principal components do not perfectly separate these clusters. Therefore, we considered employing sequential feature selection, to enhance the clustering process.

2.1.2. Modeling

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

# Split the data into predictive variables (X) and target variable (y)
X = data_cleaned.drop(columns=['therapy_cluster'])  # Use all columns except 'therapy_cluster' as predictive variables
y = data_cleaned['therapy_cluster']

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Create an instance of the Random Forest model
rf_model = RandomForestClassifier(random_state=42)

# Train the model on the training set
rf_model.fit(X_train, y_train)

# Predict the clusters on the test set
y_pred = rf_model.predict(X_test)

# Calculate the model's accuracy
accuracy = accuracy_score(y_test, y_pred)
print(f"Random Forest model accuracy: {accuracy:.2f}")
```

✓ 2.6s

```
Random Forest model accuracy: 0.89
```

Figure 70. Random Forest Classifier implementation for predictive modeling of cluster assignments

⇨ After the pre-processing technics of the data, we executed the Random Forest model which has resulted in an accuracy of 0.89.

2.2.Model Optimisation

2.2.1. Sequential Feature Selector (SFS)

We used the Sequential Feature Selector (SFS) which is a method used to optimize machine learning models by iteratively selecting the most relevant features from the dataset, in order to:

- ❖ Improve the predictive accuracy.
- ❖ Improve the classification
- ❖ Simplify the model.
- ❖ Dimensionality Reduction.
- ❖ Avoid overfitting by eliminating irrelevant or redundant features.

➢ **The application of SFS on our model**

```
Precision of the Random Forest model with feature selection: 0.92
Indices of selected features: [0 1 2 3 8 9 11 13 15 19 21 25 27 28 32 33 35 39 40 41 42 43
44 46 49 50 51 54 55 56 61 65 72 74]
Names of selected features: Index(['pam50_+_claudin-low_subtype', 'BRCA1', 'BRCA2',
'PTEN', 'CDK6', 'MYC',
       'E2F1', 'E2F3', 'DLL3', 'HEY1', 'JAG2', 'HEY1', 'AKT1', 'AKT2', 'BRAF',
       'EGFR', 'ERBB2', 'FGF1', 'FGFR1', 'HRAS', 'IGF1R', 'KRAS', 'MAP2K1',
       'MAPK1', 'MAPK1', 'MAPK3', 'MTOR', 'PIK3CA', 'RAF1', 'RPS6KB1',
       'RPS6KB2', 'NCOA3', 'NRAS', 'AKT3', 'pten_mut', 'pik3r1_mut',
       'akt2_mut', 'nras_mut'],
      dtype='object')
```

Figure 71. Feature selected by Random Forest model

⇨ This method has identified the most significant features for prediction, improving the model's accuracy from 0.89 to 0.92.

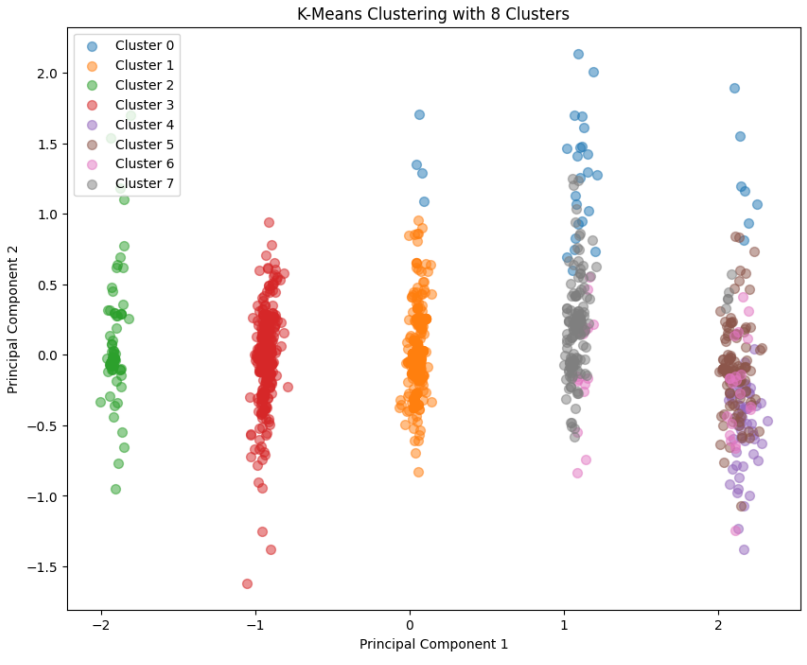


Figure 72. PCA plot of data clusters based on gene expression

- ⇨ After applying sequential feature selector method, the clustering has significantly improved. The clusters are more distinct and coherent, which aligns well with the goals of our classification model for molecular pathways.
- ⇨ This enhancement suggests that the selected features effectively capture the critical variations needed for our accurate classification.

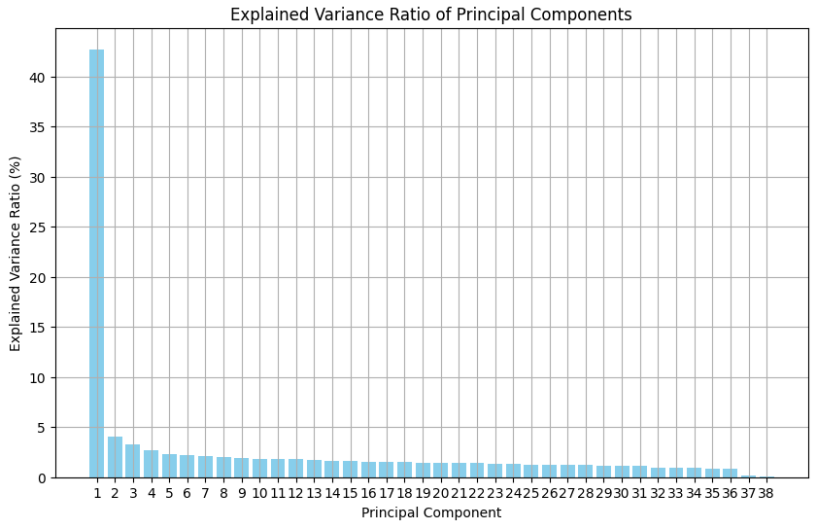


Figure 73. Explained variance ratio of Principal components after optimization with SFS

- ⇨ After applying sequential feature selector method, we noticed that the first principal component now explains over 40% of the variance, significantly higher than before and there are fewer components with very low explained variance, suggesting that the noise and redundancy have been reduced. That explains why the model is more efficient and robust.

2.2.2. GridSearchCV module of optimization

We used the GridSearchCV which is a method used for hyperparameter tuning in machine learning models, It systematically searches through a specified parameter grid, evaluating our model performance for each combination of hyperparameters to identify the optimal set that produces the best results. This exhaustive search approach helps in fine-tuning our model to achieve optimal performance.

- ➢ **The application of GridSearchCV module on our model**

```
Best hyperparameters: {'bootstrap': False, 'max_depth': None, 'min_samples_leaf': 1,
'min_samples_split': 10, 'n_estimators': 300}
```

Figure 74. Optimized Random Forest model performance with feature selection

- ⇨ This method has identified the most contributing hyperptablearamaters to the prediction using the selected features of the SFS method, improving the model's accuracy from 0.89 to 0.92.

III. Conclusion

This chapter showcased the effectiveness of Random Forest and Support Vector Machine models in predicting chemotherapy responses and classifying cancer pathways. Our analyses highlighted key biomarkers such as ER, PR, and HER2 statuses as critical predictors of treatment outcomes, demonstrating the potential of integrating clinical and molecular data to enhance treatment personalization.

The use of bioinformatics tools enabled us to elucidate significant molecular interactions and pathways disrupted in breast cancer, pointing towards targeted therapeutic opportunities. However, the generalizability of these findings is constrained by the heterogeneity of the data. Future work should aim to validate these models on larger, more diverse datasets and consider incorporating additional omics data to broaden the understanding of breast cancer dynamics.

General Conclusion

This project has demonstrated how the integration of clinical data and transcriptomic profiles through advanced predictive modeling can enhance the classification and treatment strategies in breast cancer. Our results underscore the potential of machine learning techniques, particularly ensemble methods like Random Forest, to refine the prediction of chemotherapy responses and pathway classifications, thereby facilitating more personalized treatment approaches.

Our research capitalized on multivariate and machine learning models to analyze a substantial dataset comprising clinical features and gene expression data. The application of Random, Forest and Support Vector Machine (SVM) models not only yielded high accuracy in predicting patient outcomes but also offered insights into the variable importance, highlighting key predictors such as ER status, PR status, and HER2 expression. These findings suggest a valuable framework for predicting chemotherapy efficacy based on molecular and clinical data, offering a pathway towards more targeted and effective treatment regimens.

Further, through detailed exploratory data analyses, our study revealed how specific gene expressions correlate with patient survival rates, providing a deeper understanding of the molecular drivers of breast cancer. Notably, the analysis of interaction networks using tools like KEGG and STRING databases identified crucial pathways that are altered in breast cancer, which could serve as potential targets for therapeutic intervention.

However, the study faced limitations concerning the heterogeneity of the data and the generalizability of the models. Future research should aim to validate these findings across larger and more diverse populations to overcome these limitations.

Researchers can focus on expanding the dataset to include more diverse demographic and genetic profiles, other forms of omics data, such as proteomics and metabolomics, enhancing the generalizability and robustness of the predictive models. This multi-omics approach, combined with advanced machine learning techniques like deep learning and ensemble models, could further improve the accuracy of predictions and offer a more comprehensive understanding of the complex biological interactions

In conclusion, this work has highlighted the effectiveness of combining machine learning with extensive data analysis to provide significant insights into breast cancer pathology and treatment. The models and analytical methods developed here lay the groundwork for future advancements in personalized medicine, emphasizing the need for continued integration of computational and clinical research to maximize the therapeutic outcomes in breast cancer treatment.

List of references

1. Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., & Walter, P. (2002). Molecular Biology of the Cell (4th ed.). [NCBI Bookshelf](https://www.ncbi.nlm.nih.gov/books/NBK21054/).

2. Anderson, M. W., & Schrijver, I. (2010). Next Generation DNA Sequencing and the Future of Genomic Medicine. Genes, 1(1), 38-69. https://doi.org/10.3390/genes1010038.

3. Bast, R. C., Ravdin, P., Hayes, D. F., Bates, S., Fritsche, H., Jessup, J. M., ... & Somerfield, M. R. (2001). 2000 update of recommendations for the use of tumor markers in breast and colorectal cancer: clinical practice guidelines of the American Society of Clinical Oncology. [PubMed](https://pubmed.ncbi.nlm.nih.gov/11230444/).

4. Brown, P. O., & Botstein, D. (1999). Exploring the new world of the genome with DNA microarrays. [Stanford](https://cmgm.stanford.edu/pbrown/pdf/BrownBotstein1999.pdf).

5. Cosentino, G., Plantamura, I., Tagliabue, E., Iorio, M., & Cataldo, A. (2021). Breast Cancer Drug Resistance: Overcoming the Challenge by Capitalizing on MicroRNA and Tumor Microenvironment Interplay. Cancers. [Cancers](https://www.mdpi.com/2072-6694/13/3/527).

6. Garrett, T. P. J., McKern, N. M., Lou, M. (2003). The crystal structure of a truncated ErbB2 ectodomain reveals an active conformation, poised to interact with other ErbB receptors. Mol Cell.

7. Holohan, C., Van Schaeybroeck, S., Longley, D. B., Johnston, P. G. (2013). Cancer drug resistance: An evolving paradigm. Nat. Rev. Cancer.

8. Joshi, A., Bhargava, R., & Aggarwal, P. (2020). Exploratory Data Analysis: A Comprehensive Review. Journal of Big Data. [ResearchGate](https://www.researchgate.net/publication/344251578_Exploratory_Data_Analysis_A_Comprehensive_Review).

9. Kuchenbaecker, K. B., Hopper, J. L., Barnes, D. R. (2017). Risks of breast, ovarian, and contralateral breast cancer for BRCA1 and BRCA2 mutation carriers. JAMA.

10. Kumar, V., Green, S., Stack, G., Berry, M., Jin, J. R., Chambon, P. (1987). Functional domains of the human estrogen receptor. Cell.

11. Liu, J., Chen, C., Xie, Z., Zhan, Y., & Zhu, K. (2020). Identification of Biomarkers Associated with Cancer Using Integrated Bioinformatic Analysis. IntechOpen. [IntechOpen](https://www.intechopen.com/chapters/73634).

12. Łukasiewicz, S., Czeczelewski, M., Forma, A., Baj, J., Sitarz, R., Stanisławek, A. (2021). Breast Cancer-Epidemiology, Risk Factors, Classification, Prognostic Markers, and Current Treatment Strategies-An Updated Review. Cancers.

13. Miller, T. W., Rexer, B. N., Garrett, J. T., Arteaga, C. L. (2011). Mutations in the phosphatidylinositol 3-kinase pathway: role in tumor progression and therapeutic implications in breast cancer. Breast Cancer Res.

14. Moccia, C., & Haase, K. (2021). Engineering Breast Cancer On-chip—Moving Toward Subtype Specific Models. Frontiers in Bioengineering and Biotechnology, 9, 694218. [Frontiers](https://doi.org/10.3389/fbioe.2021.694218).

15. Morére, J. F., Grellier, B., Durand-Zaleski, I., et al. (2011). Cancer du sein. Fondation ARC pour la recherche sur le cancer.

16. Mukohara, T. (2015). PI3K mutations in breast cancer: prognostic and therapeutic implications. Breast Cancer (Dove Med Press).

17. Nwabo Kamdje, A. H., Seke Etet, P. F., Vecchio, L., Muller, J. M., Krampera, M., Lukong, K. E. (2014). Signaling pathways in breast cancer: therapeutic targeting of the microenvironment. Cell Signal.

18. Perou, C. M., Sørlie, T., Eisen, M. B., van de Rijn, M., Jeffrey, S. S., Rees, C. A., ... & Botstein, D. (2000). Molecular portraits of human breast tumours.
[PubMed](https://pubmed.ncbi.nlm.nih.gov/10963602/).

19. Renoir, J. M., Marsaud, V., Lazennec, G. (2013). Estrogen receptor signaling as a target for novel breast cancer therapeutics. Biochem Pharmacol.

20. Sever, R., Brugge, J. S. (2015). Signal transduction in cancer. Cold Spring Harb Perspect Med.

21. Stark, R., Grzelak, M., & Hadfield, J. (2019). RNA sequencing: the teenage years. Nature Reviews Genetics. [Nature](https://www.nature.com/articles/s41576-019-0150-2).

22. Stephens, P. J., Tarpey, P. S., Davies, H., Van Loo, P., Greenman, C., Wedge, D. C., ... & Futreal, P. A. (2012). The landscape of cancer genes and mutational processes in breast cancer. Nature.
[PubMed](https://pubmed.ncbi.nlm.nih.gov/23000897/).

23. Yang, K., Wang, X., Zhang, H. (2016). The evolving roles of canonical WNT signaling in stem cells and tumorigenesis: implications in targeted cancer therapies. Lab Invest.

24. Ali Jafari (2024). https://www.tandfonline.com/doi/full/10.1080/21681163.2023.2299093

25. Professor Carlos Caldas from Cambridge Research Institute and Professor Sam Aparicio from the British Columbia Cancer Centre in Canada (2016).
https://www.kaggle.com/datasets/raghadalharbi/breast-cancer-gene-expression-profiles-metabric

26. Sonika Bhatnagar (2021). https://onlinelibrary.wiley.com/doi/10.1002/minf.202100115

27. Vladimir Nasteski (2017).
https://www.researchgate.net/publication/328146111_An_overview_of_the_supervised_machine_learning_methods

28. S. Ndichu, Tao Ban, Tao Ban. https://www.researchgate.net/publication/357257616_Detecting_Web-Based_Attacks_with_SHAP_and_Tree_Ensemble_Machine_Learning_Methods