

# Tutorial on “Cancer Genes Analysis Across Multiple Cancer Types via Topological Classification of scRNA-seq Data”

## 1. Open the Repository

Go to the following GitHub link:

[https://github.com/Sudarshan-Gogoi/Topological\\_Classification](https://github.com/Sudarshan-Gogoi/Topological_Classification)

## 2. Download the Required Data Files and Code

Download the following files from the repository:

### ● Example Data Files (choose one depending on your dataset):

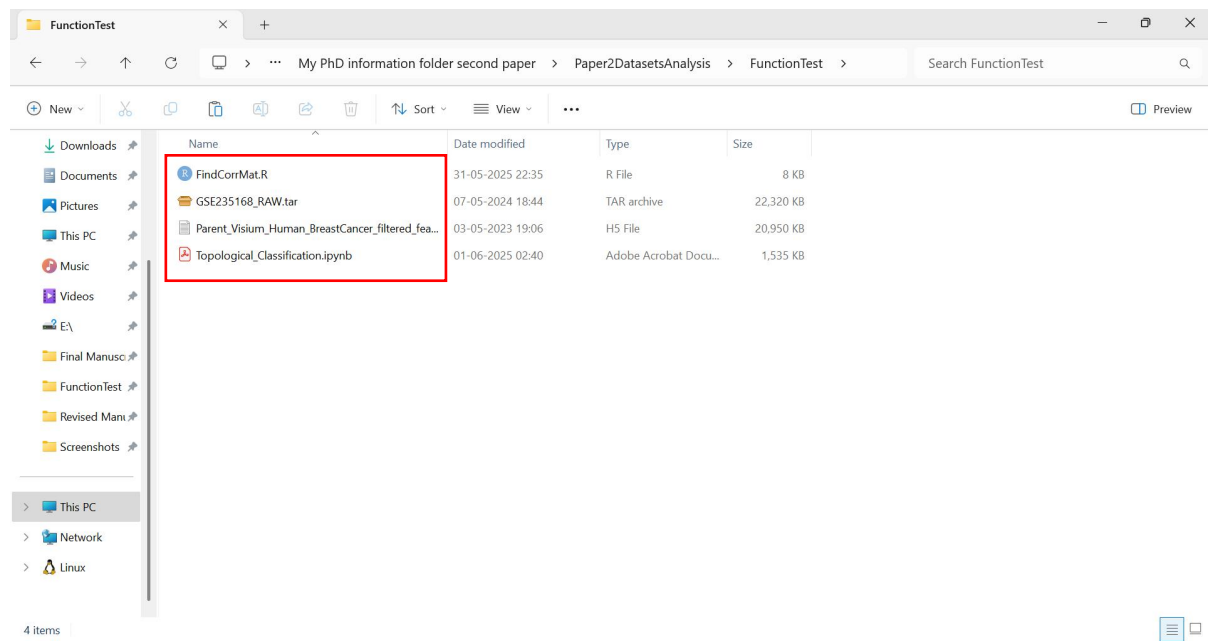
✧ GSE235168\_RAW.tar  
*or*

✧ Parent\_Visium\_Human\_BreastCancer\_filtered\_feature\_bc\_matrix.h5

### ● Code Files:

✧ FindCorrMat.R

✧ Topological\_Classification.ipynb



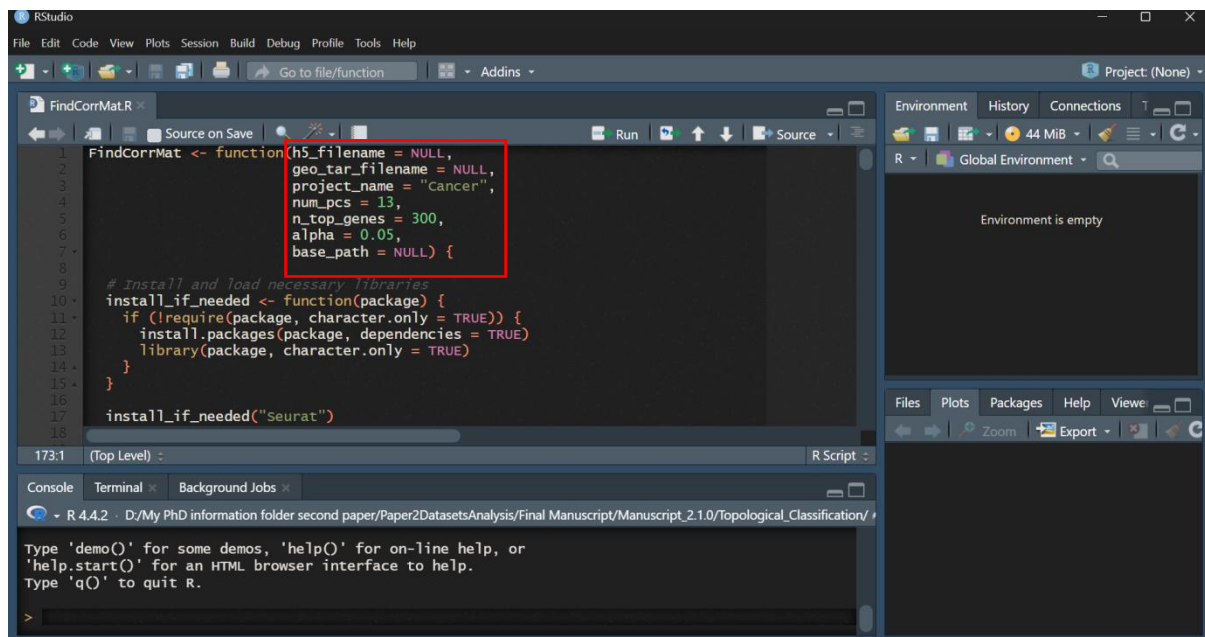
## 3. Open the FindCorrMat.R Script in RStudio

Launch RStudio (version: RStudio-2023.12.1-402) and open the file FindCorrMat.R.

## 4. Set the Input Parameters for the Function:

The function accepts the following input parameters. You can use the default values provided for the example dataset, or adjust them based on your specific dataset as described below:

- `num_pcs`: Number of principal components to use.  
*Default: 13*  
To determine the optimal value for your dataset, run the function and examine the **Elbow Plot**. The “elbow point” in the plot typically indicates the best value to use for `num_pcs`.
- `n_top_genes`: Number of top genes to select per principal component.  
*Default: 300*
- `alpha`: Significance threshold for correlation analysis.  
*Default: 0.05*



## 5. Set the Required Input Paths:

Before running the code, provide the following input parameters:

- `geo_tar_filename`: Specify the name of the dataset file you are using. Choose one of the following:
  - ✧ `GSE235168_RAW.tar`
  - or
  - ✧ `Parent_Visium_Human_BreastCancer_filtered_feature_bc_matrix.h5`
- `base_path`: Define the directory path where all your data files and code are located.  
*Tip: Use the same `base_path` throughout all scripts and notebooks to simplify the workflow and avoid file path errors.*

```

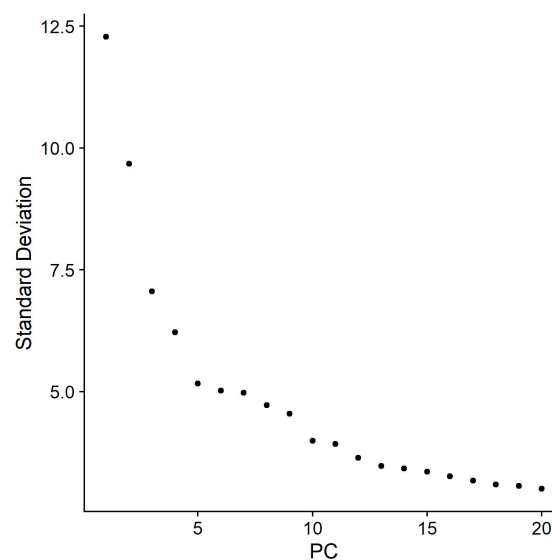
161
162 # Save the correlation matrix as a csv file
163 correlation_matrix_path <- file.path(output_path, "CancerCorrMat.csv")
164 fwrite(x = filtered_correlation, row.names = TRUE, file = correlation_matrix_path)
165
166 # Print messages about saved files
167 message("Violin plot saved at: ", violin_plot_path)
168 message("Scatter plot saved at: ", scatter_plot_path)
169 message("Variable features plot saved at: ", variable_features_plot_path)
170 message("Elbow plot saved at: ", elbow_plot_path)
171 message("Cancer correlation matrix saved at: ", correlation_matrix_path)
172 }
173
174 # Example usage:
175 # FindCorrMat(gse_filename = "parent_vistul_human_breastcancer_filtered_feature_bc_matrix.h5",
176 FindCorrMat(gse_filename = "GSE235168_RAW.tar", base_path = "D:/My PhD information folder")
177

```

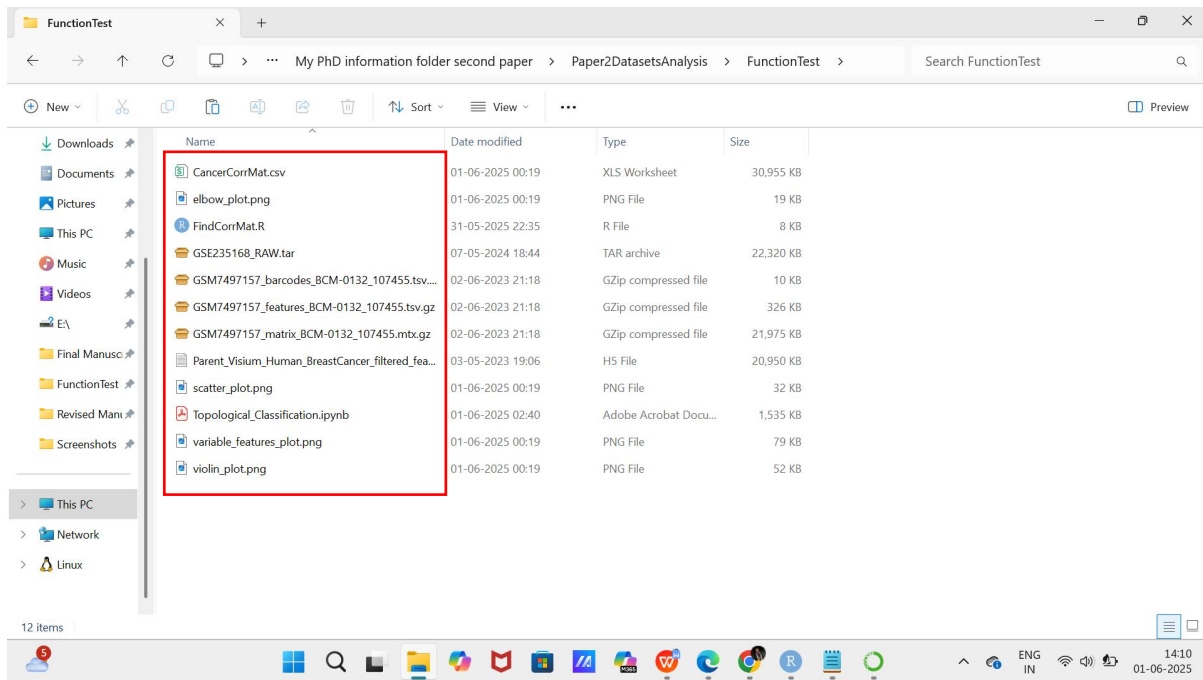
## 6. Output Files (Saved in base\_path)

After running the code, the following output files will be generated and saved in the specified base\_path directory:

- violin\_plot.png – Quality control plots showing metrics such as gene counts, feature counts, etc.
- scatter\_plot.png – Scatter plot of selected features for visual assessment.
- variable\_features\_plot.png – Plot displaying the most variable genes across the dataset.
- elbow\_plot.png – PCA elbow plot used to determine the optimal number of principal components (num\_pcs).
- CancerCorrMat.csv – Final correlation matrix of top genes used for downstream topological classification.

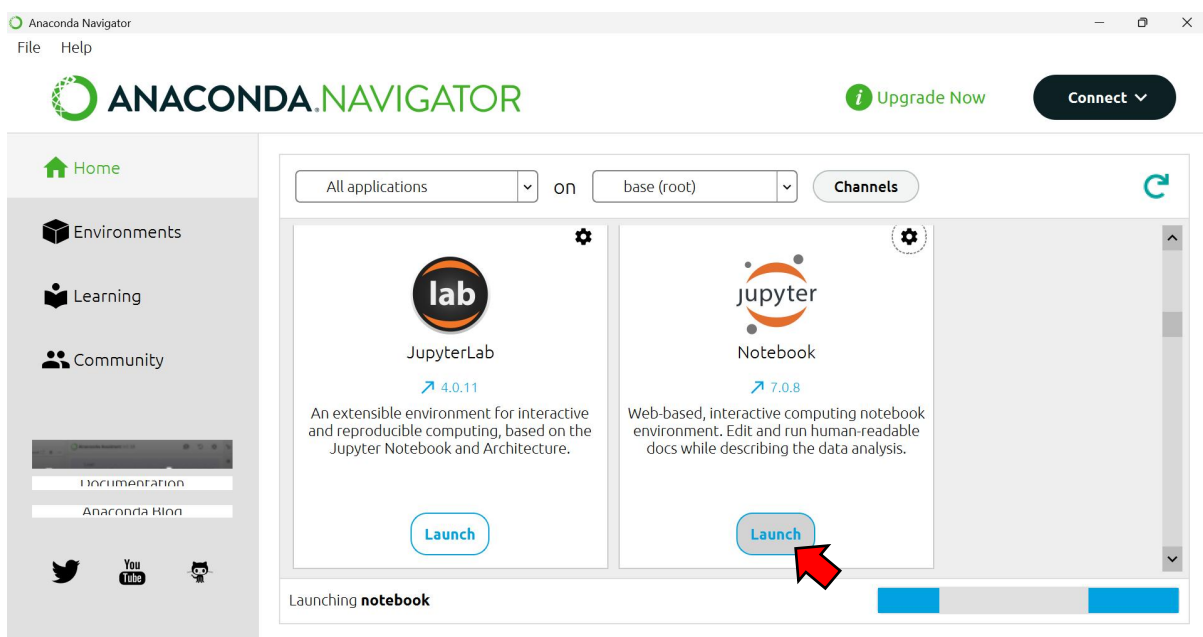


**Elbow Plot**



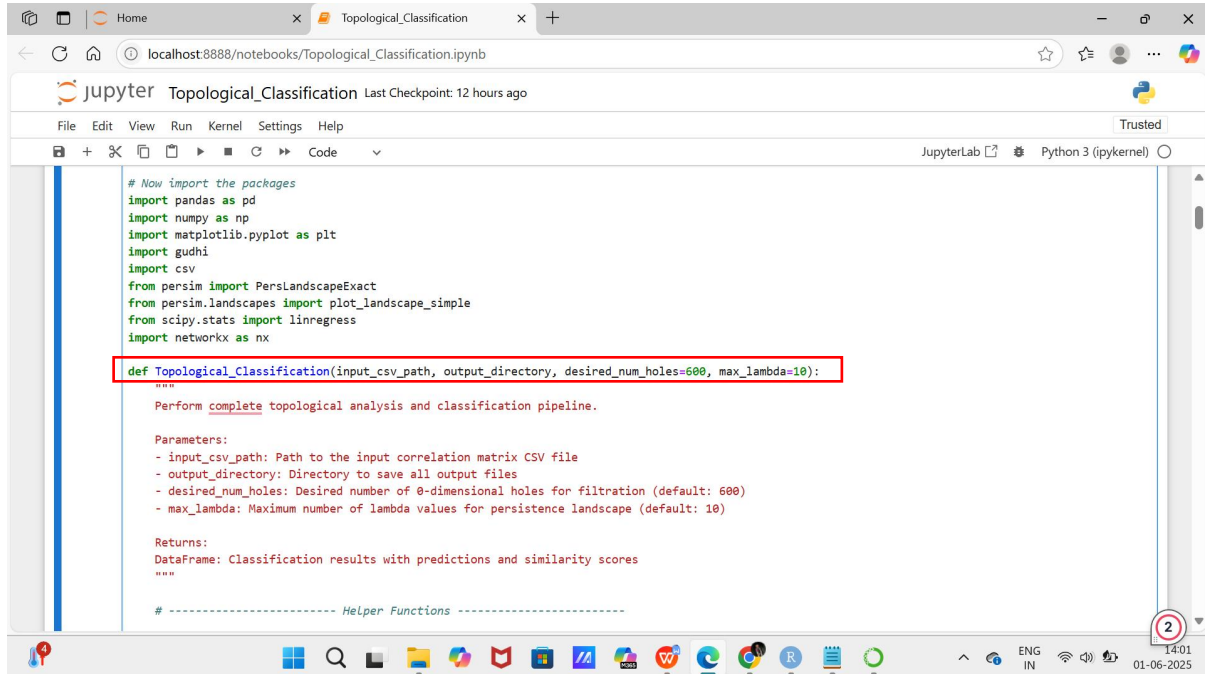
## 7. Launch Jupyter Notebook via Anaconda Navigator

- **Open Anaconda Navigator**  
Start Anaconda Navigator from your system's applications or start menu.
  - **Launch Jupyter Notebook (Version 7.0.8)**
- ✧ In the Anaconda Navigator interface, locate **Jupyter Notebook**.
- ✧ Click the **Launch** button beneath it.
- ✧ This will open Jupyter Notebook in your default web browser.



## 8. Load the Analysis Notebook

- Once Jupyter Notebook is open in your browser, navigate to the directory where you saved the project files (i.e., the `base_path`).
- Click on `Topological_Classification.ipynb` to open the notebook.



```
# Now import the packages
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import gudhi
import csv
from persim import PersLandscapeExact
from persim.landscapes import plot_landscape_simple
from scipy.stats import linregress
import networkx as nx

def Topological_Classification(input_csv_path, output_directory, desired_num_holes=600, max_lambda=10):
    """
    Perform complete topological analysis and classification pipeline.

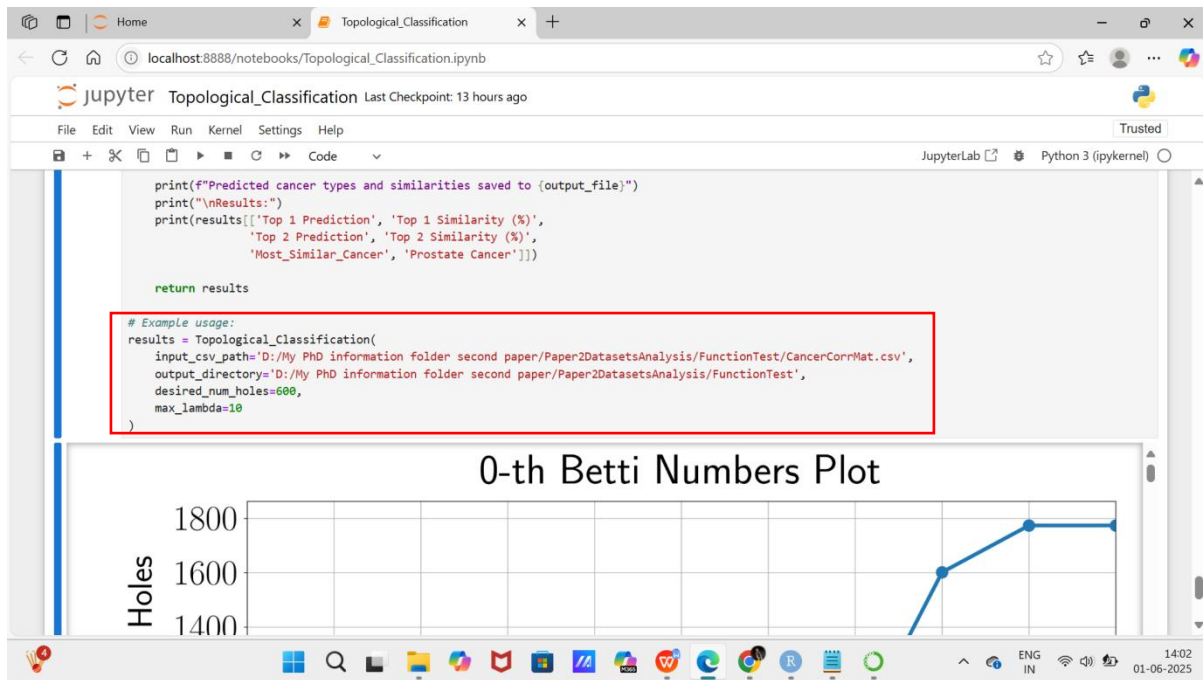
    Parameters:
    - input_csv_path: Path to the input correlation matrix CSV file
    - output_directory: Directory to save all output files
    - desired_num_holes: Desired number of 0-dimensional holes for filtration (default: 600)
    - max_lambda: Maximum number of lambda values for persistence landscape (default: 10)

    Returns:
    DataFrame: Classification results with predictions and similarity scores
    """
    # ----- Helper Functions -----
```

## 9. Set the Input Parameters in the Notebook

Before running the cells in `Topological_Classification.ipynb`, set the following input parameters:

- `input_csv_path`:  
Path to the input correlation matrix CSV file generated by the `FindCorrMat` function.
- `output_directory`:  
Directory where all output files will be saved. This is required.
- `desired_num_holes` (*optional*):  
Desired number of 0-dimensional topological features (holes) to retain.  
*Default: 600*
- `max_lambda` (*optional*):  
Maximum lambda value used for computing the persistence landscape.  
*Default: 10*

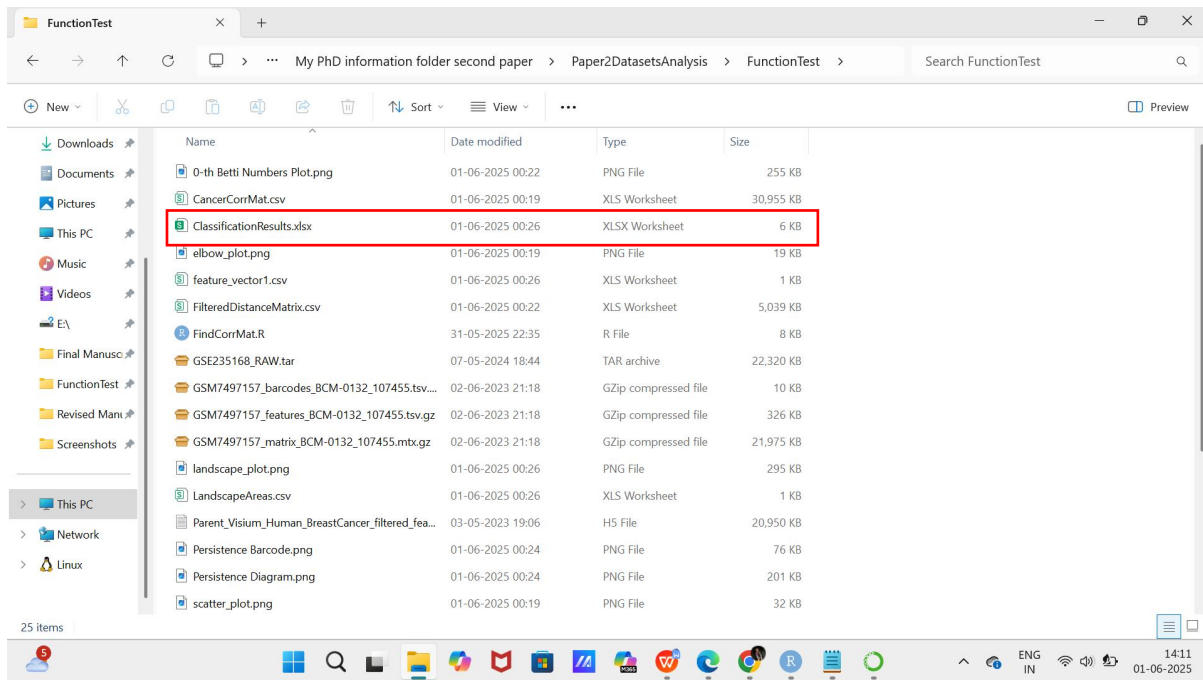


## 10. Output Files (Saved in `output_directory`)

After running the notebook, the following output files will be generated and saved in the specified `output_directory`:

- `0-th Betti Numbers Plot.png` – Visualizes the number of 0-dimensional topological features (connected components or “holes”).
- `FilteredDistanceMatrix.csv` – Filtered version of the correlation matrix used for topological analysis.
- `SignificantGenes.csv` – List of genes identified as significant based on topological filtering.
- `Persistence Diagram.png` – Persistence diagram showing birth and death of topological features.
- `Persistence Barcode.png` – Barcode plot representing the lifespan of topological features.
- `simplicial_complex_plot.png` – Network visualization of the simplicial complex structure.
- `simplicial_complex_data.csv` – Contains network statistics derived from the simplicial complex.
- `feature_vector1.csv` – Topological features extracted for classification.
- `landscape_plot.png` – Plot of the persistence landscape for the selected features.
- `LandscapeAreas.csv` – Numerical values of landscape areas used for trend and classification analysis.
- `TrendlinePlot.png` – Visual trendline of landscape areas.
- `TrendlineDetails.csv` – Statistical details of the trendline analysis.
- `ClassificationResults.xlsx` – Final classification results based on extracted topological features.





## 11. Interpretation of Significant Genes

The **significant genes** identified in `SignificantGenes.csv` are considered **primary significant genes**. These are initially selected based on the topological analysis of the correlation matrix.

To further refine these results, a **comparative filtering step** is performed:

- Run the same analysis pipeline on other cancer datasets.
- Compare the significant genes across datasets.
- Use **classification performance metrics** to identify genes that are consistently significant and contribute meaningfully to accurate classification.

