Predicting Cancer Through Differential and Unsupervised Gene Expression Analysis

Nathalie Bonin, Franco Krepel, William Mukuvi, Julien Wakim

Abstract

Early diagnosis of cancer is vital in the treatment and overall survival rate for patients no matter the tumor. A wide variety of procedures are used to detect and diagnose such disease, ranging from radiology to biochemical tests. Yet, molecular profiling, in particular the use of blood-based liquid biopsies, has drawn interest due to their convenient non-invasive ways. However, implementation of these biopsies for cancer detention has been hindered due to their inability to pinpoint the exact primary tumor. Here we determine the capabilities of using tumor-educated blood platelets for early detections of cancers by mRNA sequencing of 285 samples. With 230 cancer and 55 healthy samples, we show the ability to predict the presence of cancer based on the gene expression. After performing differential and enrichment analyses alongside multiple clustering methods, clear differences were identified between the two groups. In particular, a significant difference occurred between the clustering methods as the proportions were not the same in the comparisons between the healthy and cancer samples. Thus, our results demonstrate that observing gene expressions of tumor-educated blood platelets serves as a plausible methodology in identifying the presence of cancer. This ability to identify tumors with just one drop of blood marks just the beginning of the advancements that will occur due to liquid biopsies, improving the time at which diagnoses are made and ultimately decreasing overall cancer fatalities.

Introduction

As scientific breakthroughs occur at a rapid pace, one disease remains at the forefront of research: cancer. Involving the uncontrollable growth of tumorous cells, cancer inhibits normal cells from functioning properly. This suppression can ultimately prove to be fatal, as without these cells fulfilling their functions, parts of the body begin to fail. Cancer's true magnitude can be witnessed in the death toll it has, as annually an estimated 10 million deaths are observed globally (WHO, 2022).

A necessary combat to the fatalities associated with cancer can be found through early diagnoses of patients. The earlier the cancer is diagnosed, the more promptly treatment can be conducted, thus leading to a greater chance of survival. This sentiment is most clearly observed in prostate cancer and breast cancer. These cancers, compared to the other 29 common cancers, have the highest survival rates (Hawkes, 2019). This is due to the high percentage of prostate and breast cancers being detected at an early stage, furthering the significance of an early diagnosis.

In order to diagnose cancer, physicians utilize various methodologies. These include radiology, biochemical tests, and pathological analysis of tumor tissue. The latter has witnessed heightened interest, as large-scale genomics projects have better detailed the molecular specificities of thousands of tumors (Hoadley et al., 2014). In particular, the Cancer Genome Atlas greatly impacted the field, as the groundwork for sequencing and analyzing cancers was laid. Due to these advancements, molecular profiling and systematic analyses of tumor tissue began to be used as a cancer classifying method (Kandoth et al., 2013). Despite being the golden standard for biopsies, acquiring tissue has its limitations, as clinical complications, inconvenience, and heightened costs for the patient create barriers for such a procedure (Diaz Jr & Bardelli, 2014).

Recently, the use of blood-based liquid biopsies has garnered interest, as such molecular diagnostics has been said to likely fundamentally alter the way patients with cancer are treated (Haber & Velculescu, 2014). Using liquid biopsy provides a quick and minimally invasive procedure; however, the development of this method to detect cancer early continues to be a major challenge as efficiency lacks in pinpointing the exact nature of the primary tumor (Bettegowda et al., 2014). Nevertheless, reports have emerged claiming that examining tumor-educated platelets may form the basis of an effective prognostic tool (Calverley et al., 2010). These blood platelets are the cells that circulate within our blood and are known for their role in wound healing; however, they also play a role in metastasis. These platelets have a rich repertoire of RNAs, and the examinations of their expression patterns have been used for various reasons, most specifically to identify biomarkers of diseases (Schubert et al., 2014). In addition, these tumor-educated platelets ingest circulating mRNAs (Best et al., 2015). All these characteristics make these platelets have great potential in being applicable to cancer diagnostics.

Therefore, to assess whether such analysis holds promise, we compared the platelet mRNA profiles of 230 cancer patients to 55 healthy donors in order to see if we could predict the presence of cancer based only on the gene expression. The 230 cancer patients were composed of 60 with non-small cell lung carcinoma, 42 with colorectal cancer, 40 with glioblastoma, 35 with pancreatic cancer, 14 with hepatobiliary cancer, and 39 with breast cancer. Through a series of differential analysis, enrichment analysis, and clustering algorithms, this paper presents the procedures to differentiate between varying mRNA samples, specifically those with and without cancer. This approach allows us to potentially detect cancer with only the gene expression. Thus, enabling the diagnosis and treatment of cancers at a faster rate, as only a drop of blood will be needed

Methods

The script used to generate the results has been published to GitHub: https://github.com/Isabella136/cgs4144-project. The rest of this section will highlight some of the methods that were implemented in the script.

Obtaining Data

The GEO data provided for this analysis was formatted as a matrix of samples by genes. The genes were described with their ENSEMBL names and therefore had to first be converted into Hugo Gene Names using AnnotationHub's org packages for *Homo sapiens* (Carlson, 2019).

This way the matrix was finalized with appropriate gene names mapped to samples. Because numerous genes lacked expression data with some of the samples, the count values were incremented by one before log-scaling the data, and a density plot was created showing the distribution of these log-scaled values throughout the data.

Dimensionality Reduction

To use Dimensionality Reduction techniques, the DESeq2 R package was used to generate an expression matrix from the counts file (Love et al., 2014). A DESeqDataSet object (which will be referred to as DDS) was created to store the read counts and other statistically-relevant values such as p-values and log-fold change. Additionally, the dataset requires a design formula that includes information on each sample's treatment group (healthy or cancerous).

Once the DDS object was created, DeSeq2's member functions were implemented to aid in the building of the PCA (Love et al., 2014) and t-SNE plots (John et al., 2020). To visualize the data for downstream analyses, further transformation of the count data was required. This new transformation included variance stabilization, and it was set to not be blind to sample information specified by the design formula.

For the PCA plot, two arguments were passed into the plotVCA function: 1) the transformed data object and 2) the treatment groups to which we want to visualize the overall effect of experimental covariates and batch effects. A very similar process was conducted for the t-SNE plot using DeSeq2's member function *tsne*, where simply the read counts were passed in along with the treatment group labels.

Differential Analysis

For Differential Analysis, we use DeSeq2's wrapper function DeSeq inputted with the DDS object, which handles the differential analysis when run (Love et al., 2014). Using *lcfshrink,* noise in the data was reduced, and estimates were added to our resultant table by getting shrunken log fold change estimates. As the data is not filtered, tidyverse was used to clean it up: the list of differentially expressed genes were added as a column to a dataframe, and another column was added to specify the significance threshold result. The dataframe was then sorted by this threshold value.

To create the volcano plot, the Enhanced Volcano package (Blighe et al., 2022) was used in combination with the result of the differentially expressed genes: the differentially expressed genes dataframe was passed to the EnhancedVolcano function, with the x-intercept set as the $\log_2$ fold change, the y-intercept set as the adjusted p-values, and the p-value cutoff loosened to 0.01.

Clustering

We used four different clustering algorithms: K-means clustering, ConsensusClusterPlus (Wilkerson & Hayes, 2010), PAM clustering (Maechler et al., 2022), and Gaussian clustering (Scrucca et al., 2016).

- K-means clustering was used to illustrate a combination of the five two-dimensional K-means clustering analysis. A k-value was required. The

K-value of 5 was chosen after multiple analyses. 5 was the smallest most appropriate value as it distinguished outliers in the cluster. Anything higher than 5 created an overlap.

- ConsensusClusterPlus had trouble identifying large clusters. Using a max K-value of 3, only one cluster of a sample is displayed. Any K-value greater than 3 made it more difficult to find clusters.
- PAM clustering yielded more stable results. A K-value was also required. A K-value of 7 was chosen as it was the highest value that avoided overlap. Using this value yielded two outliers.
- Gaussian clustering chooses a k-value on its own from a range of 1-9. Out of analyses between six different models and K-values, the value that derives the least uncertainty is outputted.

Four alluvial graphs, one for each clustering algorithm, were created (Brunson, 2020). Each was used to show different cluster groups using a different amount of genes. We used different shades/colors of strands to represent distinct clusters. Gene counts on the horizontal axis were represented as a Log of 10.

HeatMaps

Whether the heatmap was used to show cluster groups from the 5000 most variable genes or visualize the overall significantly differentially expressed genes, the setup was very similar. Because the data is large, it is necessary to filter it or reduce noise before making the actual heatmap. Afterward, each cell that will be mapped to the heatmap is normalized and log-scaled after being incremented by one.

In both cases, the tables or dataframes of variable genes or differentially expressed genes versus samples are converted to a matrix which is the required type for HeatMap function input (Gu, 2022; Gu et al., 2016). For the different clustering methods, the treatment group column is added to the list of cluster group output; this list is then used as input to the *HeatmapAnnotation* object. For the heatmap of just the significantly expressed genes, a similar process is done except only a treatment group list must be added.

Enrichment Analysis

For the different enrichment analysis methods, each method was written similarly but with the respective libraries' functions and required input. For topGo (Alexa & Rahnenfuhrer, 2022), sample data was created by selecting the genes with p-values under 0.01 from all the genes, selecting the biological process ontology and then running that sample data with the *runtest* function, first by using a Fisher test for the statistic, and then again by using a Fisher elimination test as the statistic. Both *runtest* results were appended to a table.

The clusterProfiler method similarly uses its own clustering function *gseGO* (Wu et al., 2021; Yu et al., 2012). Both the input and the ontology are the same as topGo; however null values in the gene list have to be manually omitted, and the *require(DOSE)* parameter must be set to true. The rest of the input parameters are allowed to stay standard. The *Dotplot* function can then be used to input the *gseGO* data, with *showCategory* being 10 and splitting it by :.sign" (Yu et al., 2022).

GProfiler2 follows the same trend (Kolberg et al., 2020), with just the list of significant genes inputted in their *gost* method to create gProfiler data object and the chosen ontology also inputted again. The *gostplot* and *publish_gosttable* functions are then used with the resultant object created to highlight certain results of the enrichment analysis method.

Statistical Analysis

To ensure significance between the cancer and healthy sample, a chi-squared test was conducted. Significance was also tested between each pairwise clustering method. In order to conduct these chi-squared tests, new dataframes needed to be created. For comparing our two sample groups, one column has the sample type and the other has the number of clusters associated with that sample. With this dataset, a chi-squared analysis was able to be completed. For comparing the clustering algorithms, sample names were changed to the type of cluster in one column and the number of clusters associated with that sample in the other column. Once again with this data, the chi-squared test was completed after using the table() function on the data.

Results

To answer our original question, the overall results show confidence in the ability to predict the presence of cancer based on a patient's gene expressions. We were able to demonstrate this through a variety of techniques and indicators.

The first large indicator of our conclusion was when the gene expression data frame object was put under differential expression and statistical analysis, a process that allows us to identify when a gene has an observed *significant* difference in read counts (a difference that can not be accounted for due to natural variance). It does this by comparing a group's member variations with variations between groups as a whole. This process was vital as it yielded clear results that illustrated the presence of such differentially expressed genes in the cancer groups.

The next large indicator was running different enrichment analysis techniques from the previously produced list of differentially expressed genes. This method aims to locate over-represented classes of genes in the large dataset. Four different pathway enrichment analysis methods were explored using different ontologies. All methods were able to clearly identify a very small fraction of the large list of expressed genes that had clear overrepresentation in the data, related to cancer phenotypes. Using the top 10 most significant pathways, different methods found an overrepresentation of transcription and translation, gene expression, or protein conformation and localization (expected due to the nature of comparison of cancer and healthy groups); others identified terms related with cells response to outer stimuli such as drugs, defense responses to viruses, and largely mesenchyme development, which is known to have a link to tumor growth. Other methods confirmed the presence of significant genes related to protein binding and abnormal growths, all of which make sense in the context of cancer and healthy group samples, serving as confident results in the mapping of gene expression data with cancerous groups.

Finally, by using 5000 of the most variable genes of the GEO dataset, 4 different clustering methods were used to try and identify similar gene expression data between the samples. The result of these clustering methods was then compared to our already known "Cancer" and "Healthy" groups to measure how accurate the methods were in "clustering" the data together (note: we clustered by samples, not genes): demonstrating possible strengths of the algorithm in the future prediction of the existence of certain treatment groups (in our case cancer and healthy) within similar gene expression data. How close the clustering algorithms were to matching the true data was then compared by running a chi-squared test for significance. When running all 4 clustering methods in this statistical analysis, all of the resultant chi-squared values were much larger than the critical values for those tests at the specified degrees of freedom and adjusted p-values. This meant that we can reject the null hypothesis that the number of clusters for healthy samples and cancerous samples are the same, and demonstrate that the algorithms were able to accurately distinguish "cluster" groups of similar samples together according to their expression data, that closely align with the reality of the gene expressions and sample treatment groups.

When analyzing the results, a recurring issue occurred due to the fact that there were six different tumors within the cancer sample group. Thus, this was indeed a weakness of this project, as in all analyses and clusters, while different from the healthy samples, there was not a definitive grouping or similarity within the cancer samples. Thus, specific conclusions on the type of genes that cause cancer are difficult to make, as there were a wide variety of genes that varied due to the nature of the sample. In short, different tumors have differentially expressed genes, thus pinpointing any gene would be inefficient, as there were a wide variety involved. Thus, in the future, a possible approach that can be taken with this project would be the implementation of more testing groups in our analysis. Instead of just having two groups of just healthy and cancer, we can instead have healthy and each cancer be its own group. By separating the tumor types, more distinct relationships may be witnessed. If this data were to be used again, there would be seven groups (one healthy and one for each of the six cancers). From there, we could also enhance our question to see if it would be possible to identify a specific cancer based on the gene expression. This would serve to be even more advantageous, as identifying a specific cancer can have more practical outcomes, as specific treatments or precautionary measures can be taken.

In our research we attempted to solve the same question as did those that collected the data we used: whether cancer can be diagnosed early in patients through their gene expression data. Regardless of the current efficacy of such research, its possible bioethical implications if effective must be explored. The main point is what kind of obligations would this bring to the researcher in terms of what information they should share with the patient, if the patient provided their data and it turns out they are on a clear track towards cancer, should the researchers *have* to give this data to the individual who shared their samples, or stay silent? Is that the *correct* thing

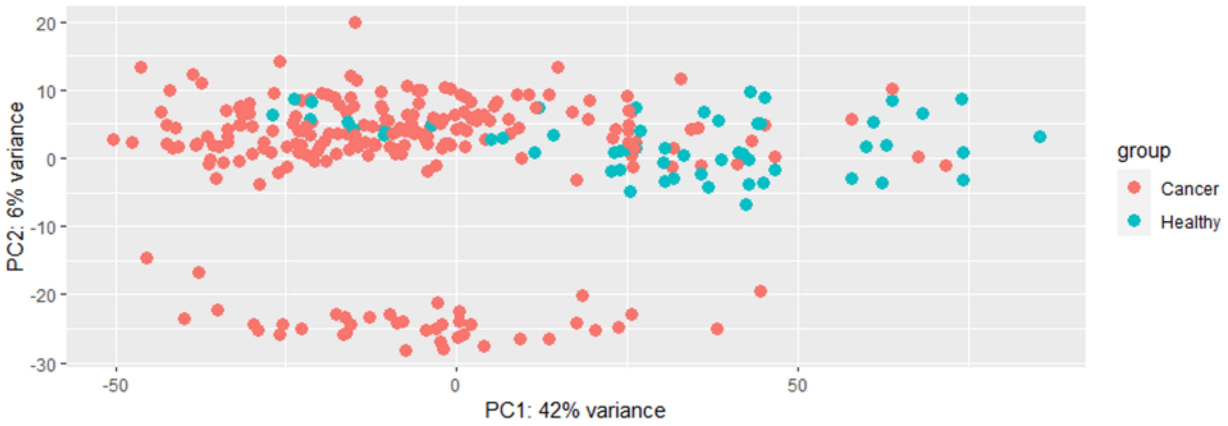


Figure 1. PCA plot of the 285 samples. Was created using the *plotPCA* function (Love et al., 2014). Each sample is represented by a point with 230 being "Cancer" and 55 being "Healthy." The axes represent the variation with PC1 and PC2 scores.

to do, as even though they might not have asked for it, at the end of the day it would help them as they could potentially seek treatment earlier? But then, some people do not want to know such information due to the anxiety and emotional stress it can cause, and how it can impact how they live the rest of their lives. This comes especially if they believe they might not be able to afford treatment for the disease. Also, if insurance companies acquire such information, how would the patients' rates be affected? These questions are vital when thinking of collecting data from individuals of higher risk-groups or especially children. Additionally, like many other forms of biomedical research, the privacy and ownership of the genetic information must always remain a key concern and handled very delicately.

Differential Expression and Enrichment Analysis

PCA Plot. Figure 1 contains the PCA plot conducted on the 285 samples. The 230 which are from patients with one of six types of cancer all belong in the "Cancer" group and are identified in red. The remaining 55 are from healthy individuals and are identified in blue.

Both groups show a high degree of variation among the x-axis, although for the most part a general trend can be observed as the cancer group is more concentrated to the left while the healthy group is more concentrated to the right. However, on the y-axis we only witness variation among the cancer group; indeed, a few of them are separated from the rest of the data due to being given a much lower PC2 score. Since PCA plots are based on similarity, the separation between the healthy and cancer samples support the assertion that genetic makeup can help differentiate between the two. Also since the cancer group is composed of six different types of tumors, an explanation is given as to why a portion of the cancer samples are separated vertically, as this group may represent a specific type of tumor that varies the genetic expression more than the others.

t-SNE Plot. Figure 2 contains the t-SNE plot. Similarly to the PCA plot, the cancer samples are in red and the healthy samples are in blue. A perplexity value of 10 was agreed upon, as there

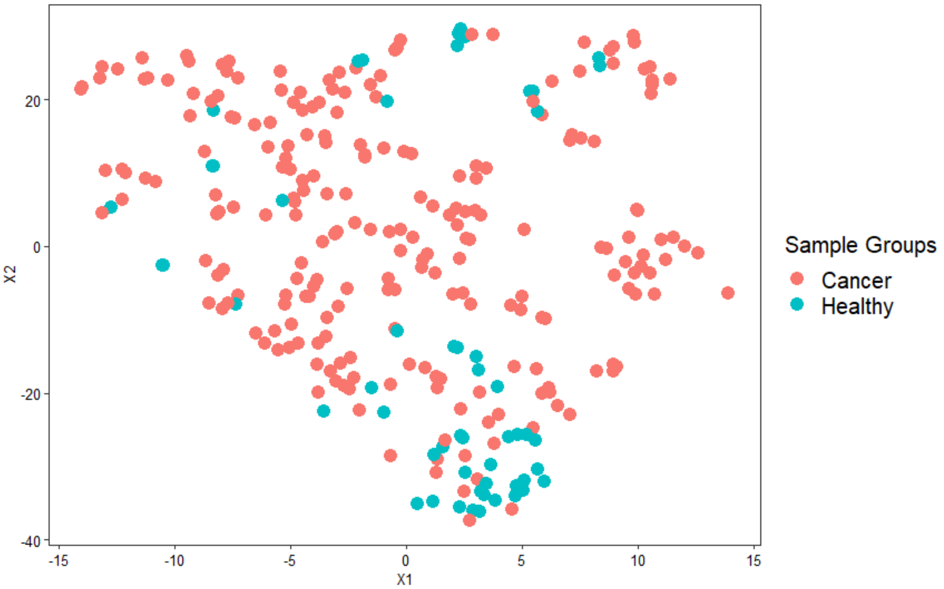
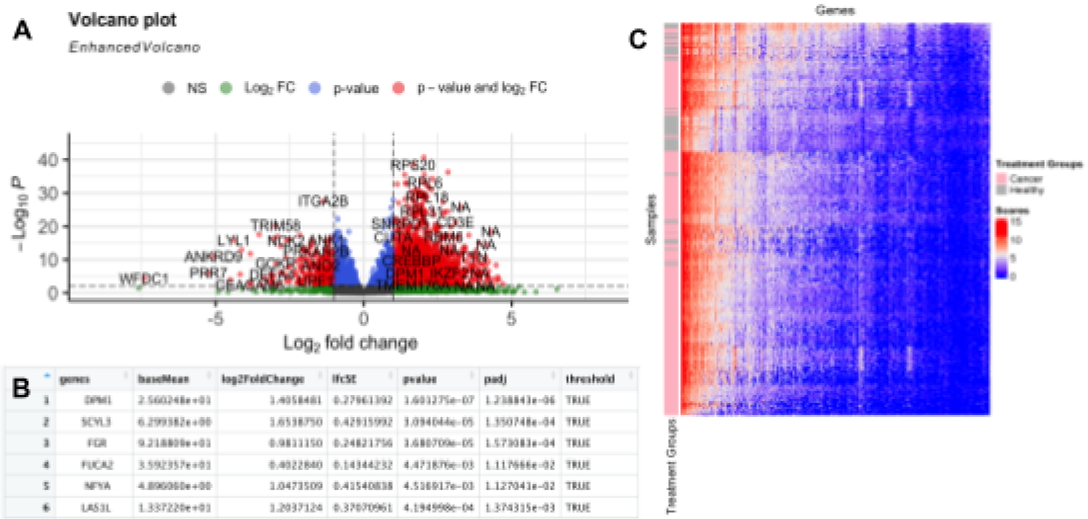


Figure 2. t-SNE plot of the 285 samples. Was created using the *tsne* function (John et al., 2020) with a perplexity value of 10. Each sample is represented by a point with 230 being "Cancer" and 55 being "Healthy."

always seemed to be an obvious cluster of healthy samples in every run (evident in the bottom of this plot).

This plot represents points that are close in multiple dimensions (their various gene expressions) in a 2D form so that similarities between samples can be more easily viewable. Therefore, comparing this plot to the PCA, a similar trend of the majority of the healthy samples being grouped near each other away from the cancer samples is witnessed. While there is a general cluster for the health samples, there is not one for the cancer samples, and the reasoning behind this is because within the cancer samples are six different types. This suggests that while



	genes	baseMean	log2FoldChange	lfcSE	pvalue	padj	threshold
1	DPM1	2.560248e+01	1.4058481	0.27961392	1.601275e-07	1.238843e-06	TRUE
2	SCYL3	6.299382e+00	1.6538750	0.42915992	3.094044e-05	1.350748e-04	TRUE
3	FGR	9.218809e+01	0.9811150	0.24821756	1.680709e-05	1.573083e-04	TRUE
4	FUCA2	3.592357e+01	0.4022840	0.14344232	4.471876e-03	1.117666e-02	TRUE
5	NFYA	4.896060e+00	1.0473509	0.41540838	4.516917e-03	1.127041e-02	TRUE
6	LAS1L	1.337220e+01	1.2037124	0.37070961	4.194998e-04	1.374315e-03	TRUE

Figure 3. Results of differential analysis. Figure 3A was created using the *EnhancedVolcano* function (Blighe et al., 2022) with the dataframe that was created by the *DESeq* function (Love et al., 2014). A subsection of this dataframe is observed in Figure 3B, as the first 6 of 57736 rows are displayed. Figure 3C was made using *Heatmap* function with differentially expressed genes, which were normalized and logscaled for noise reduction, and assigned a treatment group annotation (Gu, 2022; Gu et al., 2016).

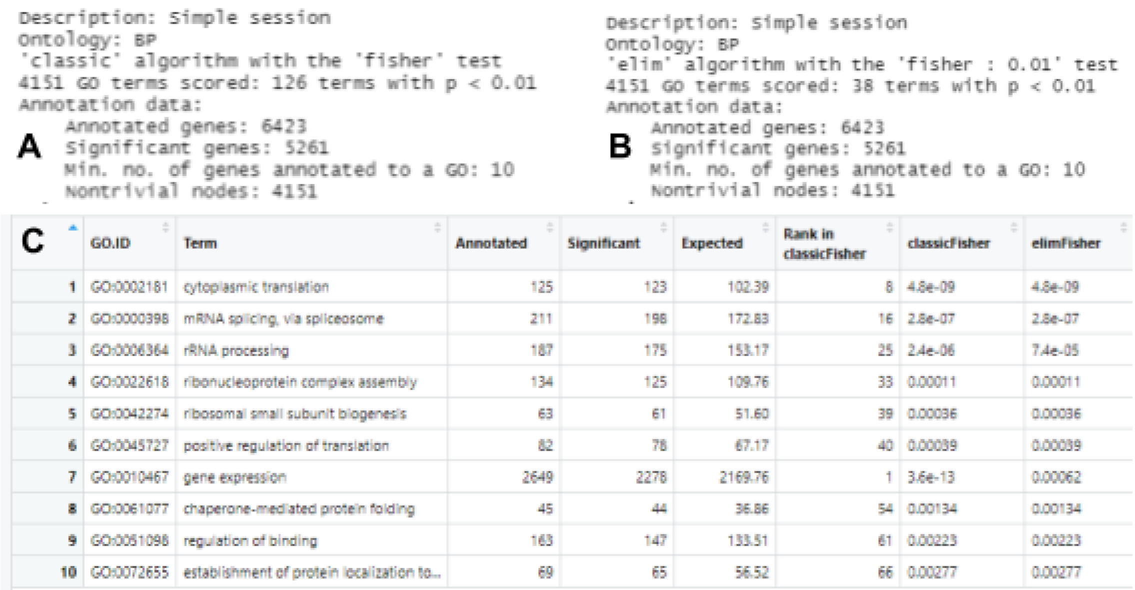
different from healthy samples, each cancer also evolves from varying gene expressions. If we were to have 6 groups for the cancer samples, more defined clusters would be evident.

Differential Analysis. Figure 3A contains the volcano plot that was created after performing differential analysis on the samples. Figure 3B represents the first 6 rows of the table created after running differential analysis. Figure 3C contains the heatmap created using the extracted significantly expressed genes.

Volcano plots show statistical significance (measured in p value) versus the magnitude of change (measured in fold change). From Figure 3A, genes with large changes in magnitude that are also statistically significant are observed. These statistically significant genes could thus be playing the biggest role in the prevalence of cancer. Specifically, in 3A, these significant genes are colored red, as the genes in the top right and top left represent the genes that are expressed significantly. Within Figure 3A, the dots in green and blue are not necessarily the most statistically different genes.

The table represented in Figure 3B represents the first 6 of 57736 rows of the expressed genes. The reasoning behind doing this is that it allows us to see if a given gene has an observed significant difference in read counts (whether the difference is greater than what would be expected with natural variance). For these first 6 genes, we can see that they are indeed significantly different, indicated by the “True” in the last column.

Figure 3C represents a heatmap created from the list of significantly differentially expressed genes, aimed to visualize the gene expression matrix. With normalized and then log scaled counts, the heatmap shows that there is a clear higher score associated with the cancerous cells with certain genes, hinting towards the possibility of finding genes specific to the treatment group, especially with how vertically broad the warmer regions are.

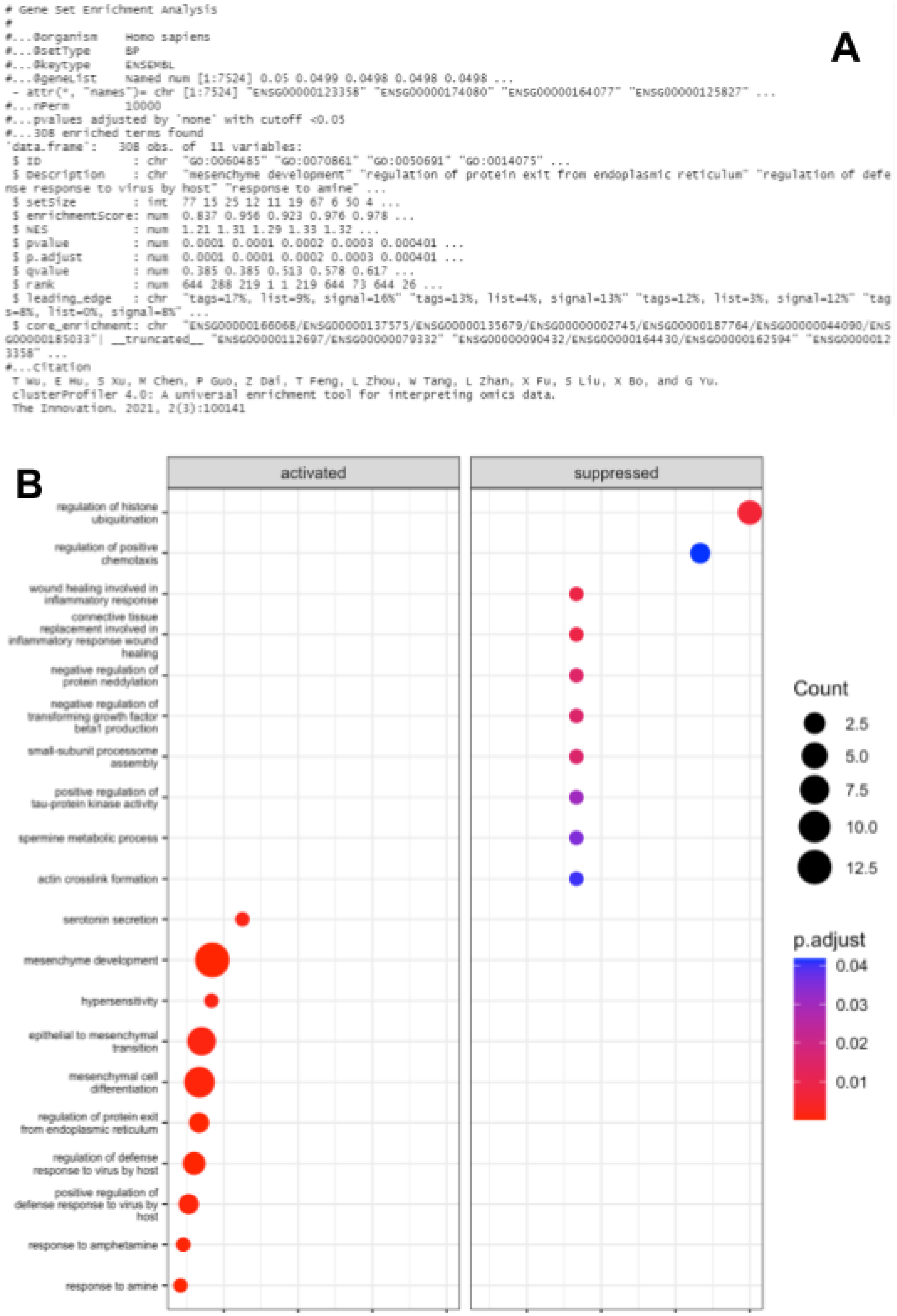


```
Description: Simple session
Ontology: BP
'classic' algorithm with the 'fisher' test
4151 GO terms scored: 126 terms with p < 0.01
Annotation data:
    Annotated genes: 6423
    Significant genes: 5261
    Min. no. of genes annotated to a GO: 10
    Nontrivial nodes: 4151
```

```
Description: Simple session
Ontology: BP
'elim' algorithm with the 'fisher : 0.01' test
4151 GO terms scored: 38 terms with p < 0.01
Annotation data:
    Annotated genes: 6423
    Significant genes: 5261
    Min. no. of genes annotated to a GO: 10
    Nontrivial nodes: 4151
```

	GO.ID	Term	Annotated	Significant	Expected	Rank in classicFisher	classicFisher	elimFisher
1	GO:0002181	cytoplasmic translation	125	123	102.39	8	4.6e-09	4.6e-09
2	GO:0000398	mRNA splicing, via spliceosome	211	198	172.83	16	2.8e-07	2.8e-07
3	GO:0006364	rRNA processing	187	175	153.17	25	2.4e-06	7.4e-05
4	GO:0022618	ribonucleoprotein complex assembly	134	125	109.76	33	0.00011	0.00011
5	GO:0042274	ribosomal small subunit biogenesis	63	61	51.60	39	0.00036	0.00036
6	GO:0045727	positive regulation of translation	82	78	67.17	40	0.00039	0.00039
7	GO:0010467	gene expression	2649	2278	2169.76	1	3.6e-13	0.00062
8	GO:0061077	chaperone-mediated protein folding	45	44	36.86	54	0.00134	0.00134
9	GO:0051098	regulation of binding	163	147	133.51	61	0.00223	0.00223
10	GO:0072655	establishment of protein localization to...	69	65	56.52	66	0.00277	0.00277

Figure 4. Results of topGO enrichment analysis. Figure 4A and 4B were executed using the *runTest* function (Alexa & Rahnenfuhrer, 2022) with the topGO data and corresponding algorithm (classic and elimination). Figure 4C was created using the *GenTable* function.



```
# Gene Set Enrichment Analysis
#
#...@organism    Homo sapiens
#...@setType     BP
#...@keytype     ENSEMBL
#...@geneList    Named num [1:7524] 0.05 0.0499 0.0498 0.0498 0.0498 ...
 - attr(*, "names")= chr [1:7524] "ENSG00000123358" "ENSG00000174080" "ENSG00000164077" "ENSG00000125827" ...
#...nPerm        10000
#...pvalues adjusted by 'none' with cutoff <0.05
#...308 enriched terms found
'data.frame':   308 obs. of  11 variables:
 $ ID             : chr  "GO:0060485" "GO:0070861" "GO:0050691" "GO:0014075" ...
 $ Description    : chr  "mesenchyme development" "regulation of protein exit from endoplasmic reticulum" "regulation of defe
nse response to virus by host" "response to amine" ...
 $ setSize        : int  77 15 25 12 11 19 67 6 50 4 ...
 $ enrichmentScore: num  0.837 0.956 0.923 0.976 0.978 ...
 $ NES            : num  1.21 1.31 1.29 1.33 1.32 ...
 $ pvalue         : num  0.0001 0.0001 0.0002 0.0003 0.000401 ...
 $ p.adjust       : num  0.0001 0.0001 0.0002 0.0003 0.000401 ...
 $ qvalue         : num  0.385 0.385 0.513 0.578 0.617 ...
 $ rank           : num  644 288 219 1 1 219 644 73 644 26 ...
 $ leading_edge   : chr  "tags=17%, list=9%, signal=16%" "tags=13%, list=4%, signal=13%" "tags=12%, list=3%, signal=12%" "tag
s=8%, list=0%, signal=8%" ...
 $ core_enrichment: chr  "ENSG00000166068/ENSG00000137575/ENSG00000135679/ENSG00000002745/ENSG00000187764/ENSG00000044090/ENS
G00000185033"| __truncated__ "ENSG00000112697/ENSG00000079332" "ENSG00000090432/ENSG00000164430/ENSG00000162594" "ENSG0000012
3358" ...
#...Citation
 T Wu, E Hu, S Xu, M Chen, P Guo, Z Dai, T Feng, L Zhou, W Tang, L Zhan, X Fu, S Liu, X Bo, and G Yu.
 clusterProfiler 4.0: A universal enrichment tool for interpreting omics data.
 The Innovation. 2021, 2(3):100141
```

B
activated
suppressed
regulation of histone ubiquitination
regulation of positive chemotaxis
wound healing involved in inflammatory response
connective tissue replacement involved in inflammatory response wound healing
negative regulation of protein neddylation
negative regulation of transforming growth factor beta1 production
small-subunit processome assembly
positive regulation of tau-protein kinase activity
spermine metabolic process
actin crosslink formation
serotonin secretion
mesenchyme development
hypersensitivity
epithelial to mesenchymal transition
mesenchymal cell differentiation
regulation of protein exit from endoplasmic reticulum
regulation of defense response to virus by host
positive regulation of defense response to virus by host
response to amphetamine
response to amine
Count
2.5
5.0
7.5
10.0
12.5
p.adjust
0.04
0.03
0.02
0.01

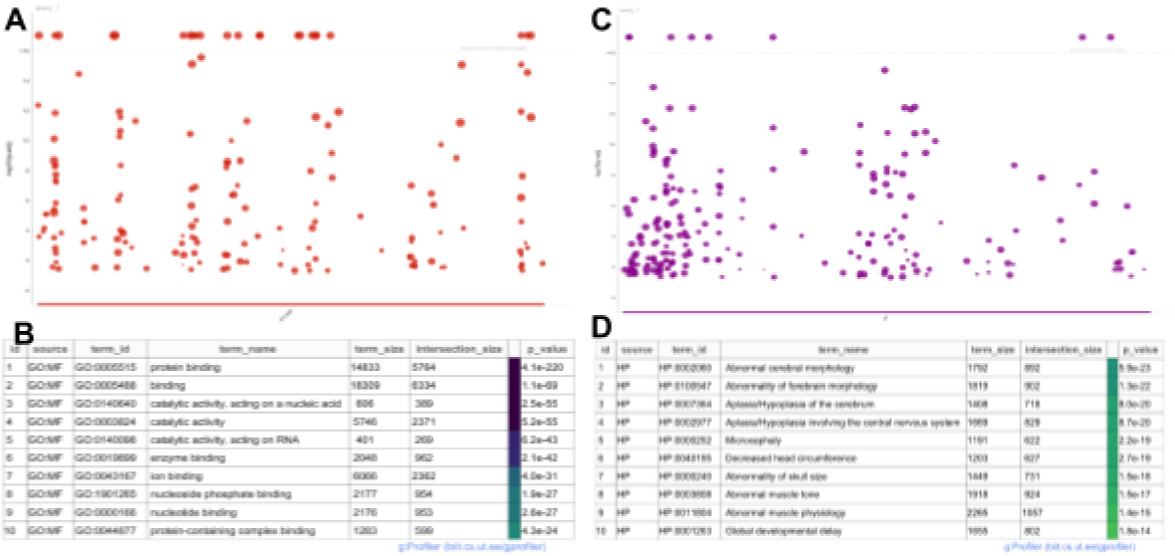
Figure 5. Results of clustProfiler enrichment analysis. Figure 5A was executed using the *gseGO* function (Wu et al., 2021; Yu et al., 2012) with the significantly expressed genes sorted in decreasing order. Figure 5B was created using the *dotplot* function and the output of *gseGO*.

Enrichment Analysis. Figure 4A and Figure 4B represent the classic fisher test and the elimination fisher test respectively when using the topGO method with the biological process ontology. Figure 4C lists the top 10 significant pathways according to the elimination algorithm (the rank based on the classic algorithm is also listed in the table).

The topGo method and its tests found 5261 significant genes. That being said, the elimination algorithm was more rigorous as it has approximately ⅔ less terms with a p-value less than 0.01. Figure 4C and the top 10 significant pathways help in determining the type of genes that were shown to be significantly different. Using the “Term” column it can be concluded that these pathways are related to transcription and translation, gene expression, or protein conformation and localization. The two rightmost columns in the table correspond to the adjusted p values for each GO term as determined by both the classic algorithm and the elimination algorithm, respectively.

Figure 5A represents the enrichment analysis through the clustProfiler method with the biological process. Figure 5B represents this analysis in a dotplot, revealing the significantly activated and suppressed genes. The plot shows a lower GeneRatio accompanied with a lower constant adjusted p-value and higher counts for the activated data. Whereas for the suppressed data you have constant lower GeneRatios accompanied with higher adjusted p-values, except for regulation of cellular response to heat which differed with a high count, adjusted p-value and GeneRatio. With GeneRatio being the fraction of differentially expressed genes found in the gene set, the fact that these regulator genes are the most varying seems to be consistent, as cancer often impacts these regulators.

Figure 6 represents the enrichment analysis through the gProfiler2 method with 6A and 6B depicting the gene ontology molecular function and 6C and 6D showing the human phenotype ontology. Within Figure 6A and 6C, the higher the element, the more significantly



id	source	term_id	term_name	term_size	intersection_size		p_value
1	GO:MF	GO:0005515	protein binding	14833	5784		4.1e-220
2	GO:MF	GO:0005488	binding	18309	6334		1.1e-69
3	GO:MF	GO:0140640	catalytic activity, acting on a nucleic acid	606	389		2.5e-55
4	GO:MF	GO:0003824	catalytic activity	5746	2371		5.2e-55
5	GO:MF	GO:0140098	catalytic activity, acting on RNA	401	269		6.2e-43
6	GO:MF	GO:0019899	enzyme binding	2046	962		2.1e-42
7	GO:MF	GO:0043167	ion binding	6066	2362		4.0e-31
8	GO:MF	GO:1901265	nucleoside phosphate binding	2177	954		1.9e-27
9	GO:MF	GO:0000166	nucleotide binding	2176	953		2.6e-27
10	GO:MF	GO:0044877	protein-containing complex binding	1263	599		4.3e-24

gProfiler (biit.cs.ut.ee/gprofiler)

id	source	term_id	term_name	term_size	intersection_size		p_value
1	HP	HP:0002060	Abnormal cerebral morphology	1792	892		5.9e-23
2	HP	HP:0100547	Abnormality of forebrain morphology	1819	902		1.3e-22
3	HP	HP:0007364	Aplasia/Hypoplasia of the cerebrum	1408	718		8.0e-20
4	HP	HP:0002977	Aplasia/Hypoplasia involving the central nervous system	1669	829		8.7e-20
5	HP	HP:0000252	Microcephaly	1161	622		2.2e-19
6	HP	HP:0040195	Decreased head circumference	1203	627		2.7e-19
7	HP	HP:0000240	Abnormality of skull size	1449	731		1.5e-18
8	HP	HP:0003808	Abnormal muscle tone	1918	924		1.5e-17
9	HP	HP:0011804	Abnormal muscle physiology	2265	1057		1.4e-15
10	HP	HP:0001263	Global developmental delay	1655	802		1.8e-14

gProfiler (biit.cs.ut.ee/gprofiler)

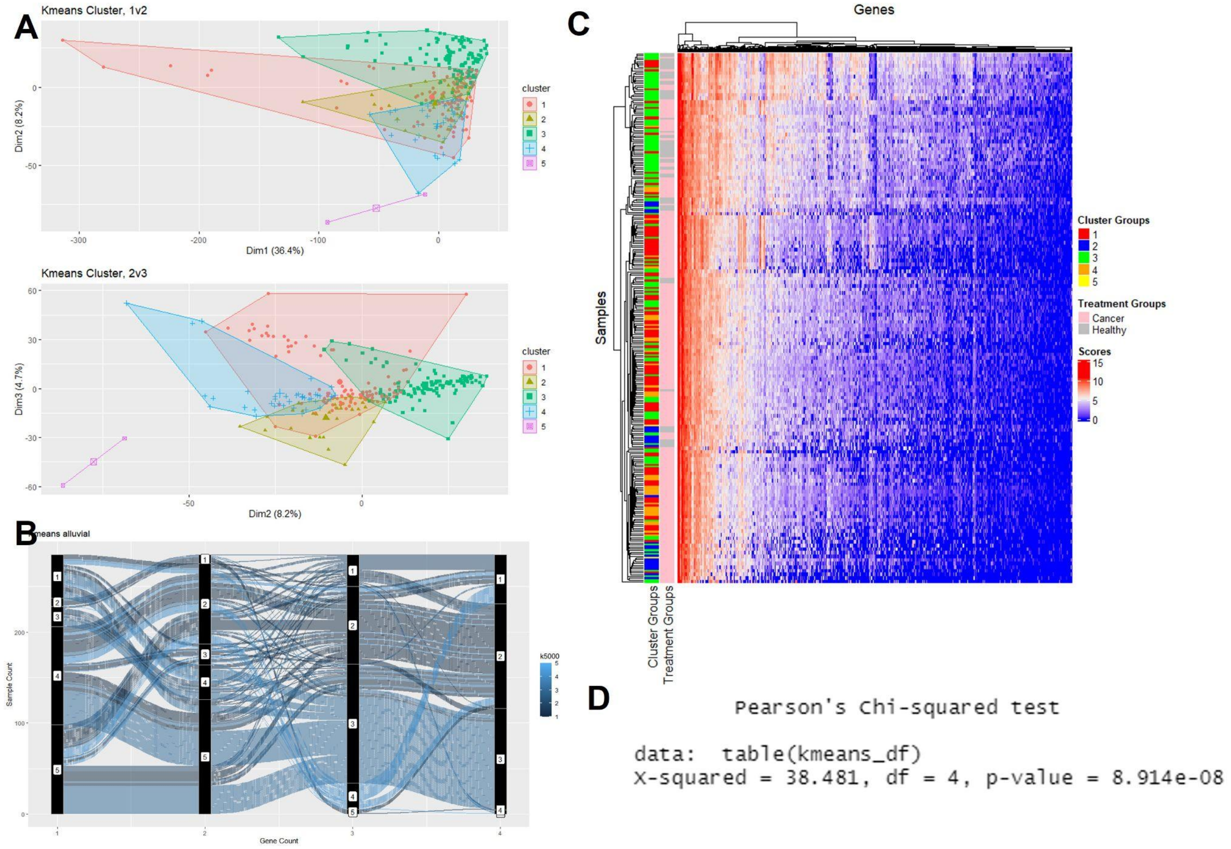
Figure 6. Results of gProfiler2 enrichment analysis. Figure 6A was created using the *gostplot* function and Figure 6B was made using the *publish_gosttable* function with the molecular function gene ontology that was conducted by using *gost* (Kolberg et al., 2020). Figure 6C and 6D used the same corresponding functions, but with an altered source, using the human phenotype ontology.

different that gene is represented. Figure 6B and 6D are an extension to the plot, giving more detailed information about the gene, as well as the size and intersection size (the number of genes in the input query that are annotated to the corresponding term precision). Comparing the two plots, greater variance was evident in the molecular function rather than phenotypic abnormalities. This is consistent with the knowledge of cancer, as this contrast suggests that the differentially expressed genes revolve around molecular mechanisms, which many tumors alter.

Unsupervised Analysis

K-Means Clustering. Figure 7A contains two cluster plots. The first demonstrates the five clusters identified by the K-means clustering analysis represented in two dimensions. Because the K-means analysis requires us to input a k value, multiple analyses were necessary to see which was the best. Five was chosen because it was the smallest k value that was able to single out the two outliers (shown in green). Their nature as outliers becomes much more obvious in the next graph which maps the cluster against dimensions 2 and 3.

Figure 7B is an alluvial graph that shows all of the different cluster groups found by the k-means clustering analysis when using a different amount of genes. One notable observation



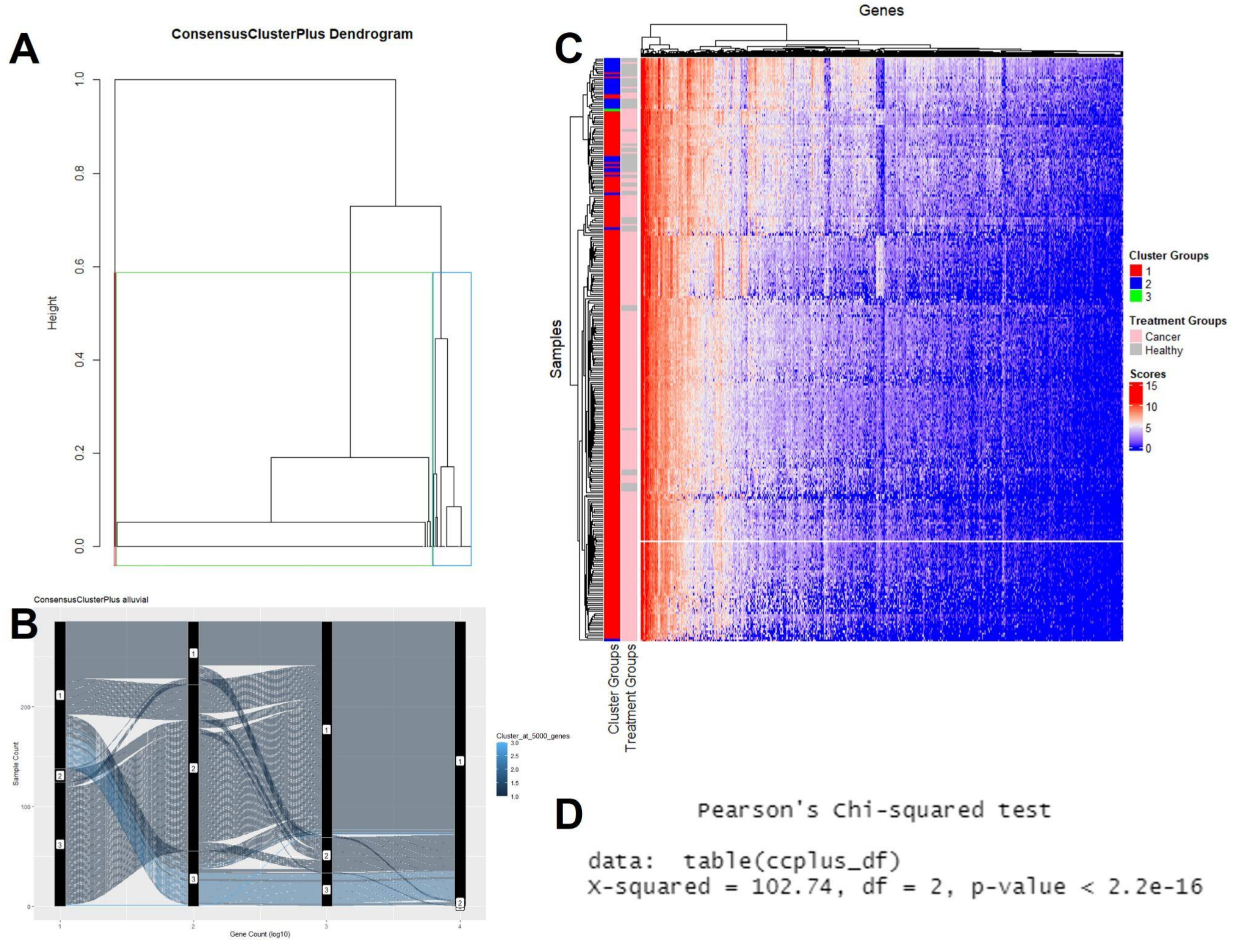
```
Pearson's Chi-squared test

data:  table(kmeans_df)
X-squared = 38.481, df = 4, p-value = 8.914e-08
```

Figure 7. Results of K-Means clustering analysis. 7A was plotted using the *fviz_cluster* function (Kassambara & Mundt, 2020). 7B was made with the ggalluvial package, which is an extension of ggplot (Brunson, 2020). The horizontal axis represents the gene count as a log of ten, and the color of the different strands represent the cluster they are in when using 5000 genes. 7C uses the HeatMap package (Gu, 2022; Gu et al., 2016) and separates each cluster and treatment group into different colors. Finally, 7D is a chi-squared test run with R's default *chisq* function.

illustrated by this graph is that there are a few unstable samples that move from one cluster to the next. For the most part, however, most samples tried to stay together in one cluster, especially when going from 1000 to 10000 genes.

Additionally, both Figures 7C and 7D illustrate the merit of the K-means clustering analysis. The two ladders at the left side of 7C demonstrate that healthy samples are usually found in cluster three, which would therefore indicate a moderate correlation between clustering and treatment group. Additionally, the chi-square test performed in 7D resulted in a high chi-squared value and in a p-value much smaller than 0.05, thus rejecting the null hypothesis that there is no correlation between clustering and treatment groups.

ConsensusClusterPlus. This clustering algorithm had trouble finding large clusters in the GEO data. As shown in Figure 8A, when running with a max k value of 3, it already found one cluster of size one (at the very left of the dendrogram). Increasing the k value made this phenomenon worse, and the *ConsensusClusterPlus* function wouldn't allow a k value of 2.

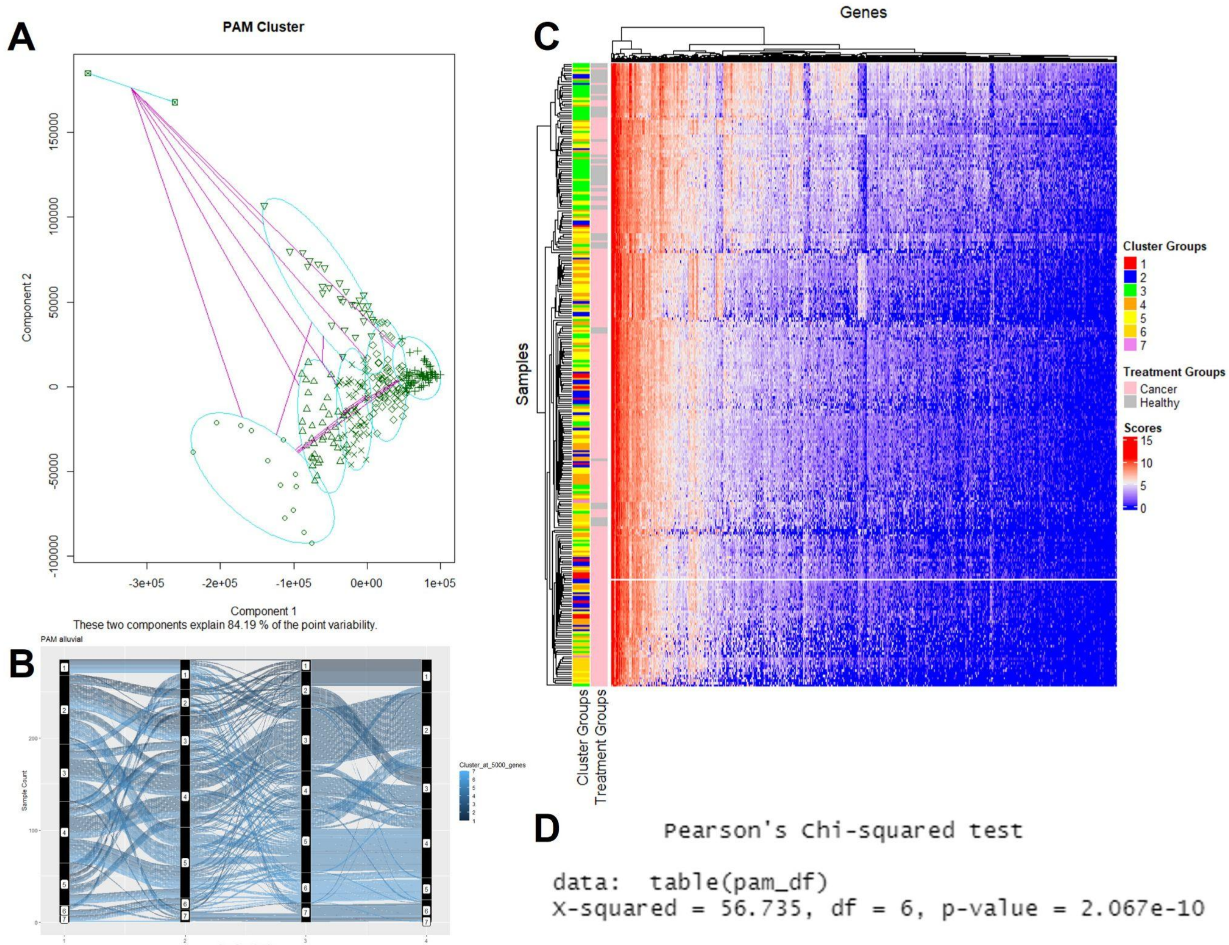


```
        Pearson's Chi-squared test

data:  table(ccplus_df)
X-squared = 102.74, df = 2, p-value < 2.2e-16
```

Figure 8. Results of ConsensusClusterPlus clustering analysis (Wilkerson & Hayes, 2010). 8A was plotted using the *as.dendrogram* and *hclust* functions which naturally comes in R's stats package. 8B was made with the ggalluvial package, which is an extension of ggplot (Brunson, 2020). The horizontal axis represents the gene count as a log of ten, and the color of the different strands represent the cluster they are in when using 5000 genes. 8C uses the HeatMap package (Gu, 2022; Gu et al., 2016) and separates each cluster and treatment group into different colors. Finally, 8D is a chi-squared test run with R's default *chisq* function.

The reason as to why ConsensusClusterPlus didn't fare well with the GEO data may be related to the high amount of genes used in the analysis. This is especially obvious in Figure 8B, where the alluvial graph demonstrates that lower gene counts resulted in larger clusters. The 100-gene analysis appears to be most optimal by having an adequate distribution of samples. However, a gene count of 10000 led to one large cluster, a second very small cluster, and one cluster of size one. Additionally, when looking at the one-sample cluster from the 5000-gene analysis, one can observe that it actually joins the second very small cluster in the 10000-gene analysis instead of staying by itself. This can also be an indication of the instability of ConsensusClusterPlus clusters when using higher gene counts.

That being said, both Figures 8C and 8D seem to indicate that the ConsensusClusterPlus method has some merit in differentiating between healthy and cancer samples. A majority of healthy samples are part of cluster two. Additionally, because the chi-squared value is high and



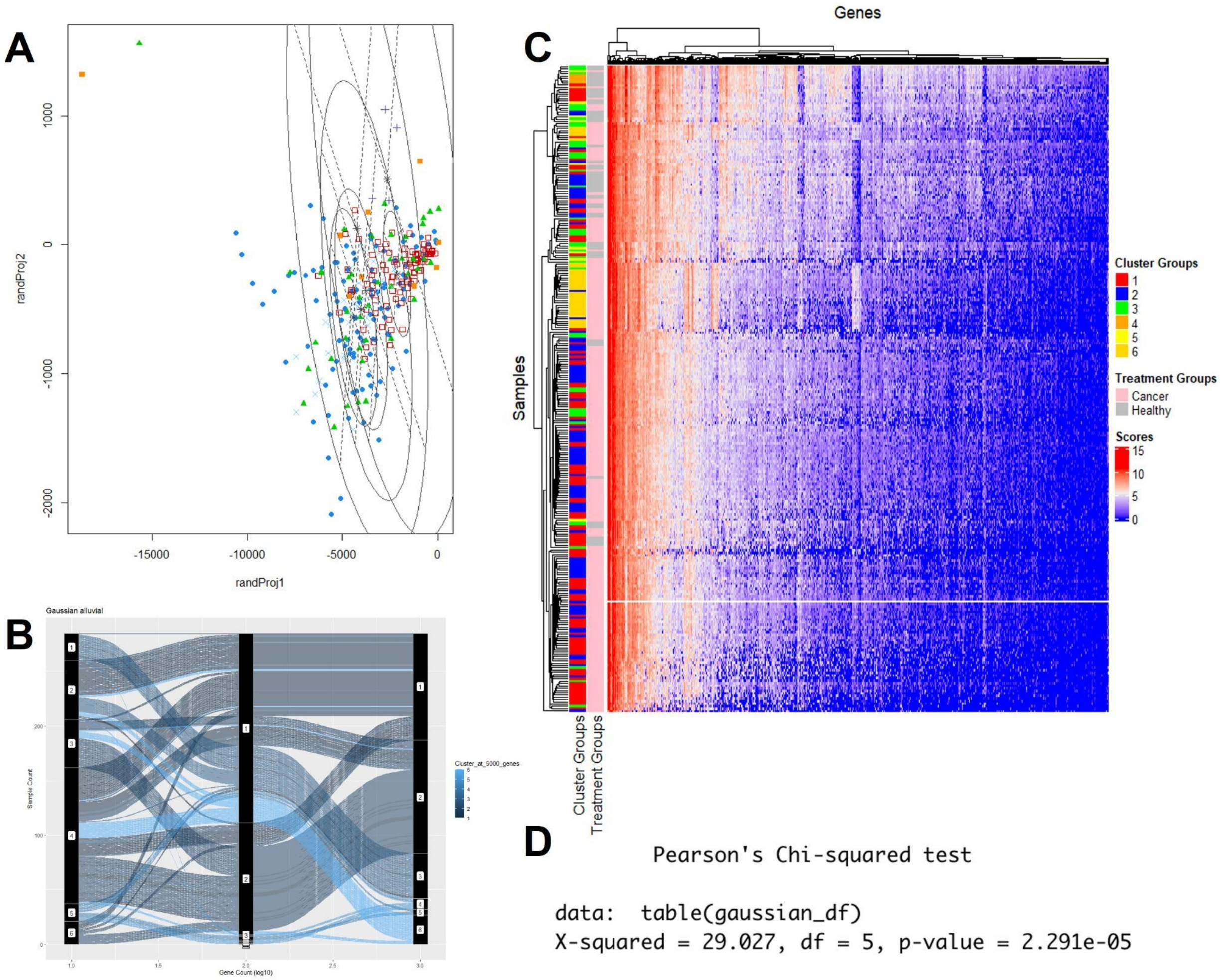
```
        Pearson's Chi-squared test

data:  table(pam_df)
X-squared = 56.735, df = 6, p-value = 2.067e-10
```

Figure 9. Results of PAM clustering analysis (Maechler et al., 2022). 9A was plotted using the *plot* function which naturally comes in R's graphics package. The expression data was mapped against the two components that were determined to have the highest correlation with point variability. 9B was made with the ggalluvial package, which is an extension of ggplot (Brunson, 2020). The horizontal axis represents the gene count as a log of ten, and the color of the different strands represent the cluster they are in when using 5000 genes. 9C uses the HeatMap package (Gu, 2022; Gu et al., 2016) and separates each cluster and treatment group into different colors. Finally, 9D is a chi-squared test run with R's default *chisq* function.

the p-value is much smaller than 0.05, the null hypothesis that there is no correlation between clustering and treatment groups can be rejected.

PAM. This clustering analysis resulted in stable and well-established clusters, as seen in Figure 9A. A k-value had to be provided, and because 7 is the highest value that avoids overlap, it was chosen for the rest of the analysis. Additionally, just like with the K-Means clustering method, two outliers in the samples have been identified by this method.

The alluvial graph in Figure 9B shows all of the different cluster groups found by the PAM clustering analysis when using a different amount of genes. One thing to note is that although clusters seem to be mostly stable when going from 1000 to 10000 genes, the transition from 100 to 1000 genes completely jumbles up the samples from one cluster to another. This could signify that contrary to the ConsensusClusterPlus analysis, the PAM clustering analysis is better suited for higher gene counts.



```
    Pearson's Chi-squared test

data:  table(gaussian_df)
X-squared = 29.027, df = 5, p-value = 2.291e-05
```

Figure 10. Results of Gaussian clustering analysis (Scrucca et al., 2016). Because this clustering method can only be graphed gene vs gene, 10A was plotted using the *randProj* function which belongs in the same package as the function used to run the analysis. 10B was made with the ggalluvial package, which is an extension of ggplot (Brunson, 2020). The horizontal axis represents the gene count as a log of ten, and the color of the different strands represent the cluster they are in when using 5000 genes. 10C uses the HeatMap package (Gu, 2022; Gu et al., 2016) and separates each cluster and treatment group into different colors. Finally, 10D is a chi-squared test run with R's default *chisq* function.

Figure 9C demonstrates a moderate correlation between cluster groups and treatment. A majority of healthy samples in cluster three. However, a substantial number of healthy samples were also sorted into the much smaller cluster six, perhaps indicating that a k-value of 7 may be too high for predicting the treatment group of a sample. Finally, Figure 9D reveals a high chi-squared value and a p-value much smaller than 0.05, thus rejecting the null hypothesis that there is no correlation between clustering and treatment groups.

Gaussian. This analysis is the only one that determines the k-value on its own. This method runs an analysis for 6 different models for different k-values (by default, from 1 to 9). The model/k-value combination that has the least amount of uncertainty is given as output. Because this cluster can only be graphed gene vs gene, we chose to represent the clusters by using a random projection, as shown in Figure 10A.

Due to being highly expensive in terms of time and memory, the Gaussian method couldn't run a 10000-gene analysis. Therefore, we only used 10, 100, and 1000 genes, as shown in Figure 10B. Additionally, while the 10-gene and the 1000-gene clustering analysis also identified six clusters, the 100-gene clustering analysis found nine clusters. Finally, while there are a few samples that move clusters (especially when going from 10 to 100 genes), most samples tend to remain grouped in the same cluster. Unfortunately, due to not being able to compute the 10000-gene analysis, it is impossible to determine whether Gaussian is well-suited for higher gene counts from an analytical point of view.

Meanwhile, Figures 10C and 10D show conflicting results in terms of the correlation between the cluster groups and the treatment groups. Figure 4C indicates that healthy samples can belong to any cluster except six, thus implying a weak correlation. However, because the p-value expressed in Figure 10D is lower than 0.05 and the chi-squared value is high, the null hypothesis that there is no correlation between clustering and treatment groups still gets rejected.

Conclusion

Most important outcome of our work was that cancerous cells were significantly different from healthy cells to a great extent. According to the PCA plot, with a 42% variance, healthy cells differed greatly from cancerous cells. Results from a statistical analysis regarding the first six rows of expressed genes that showed a significant difference in read counts was confirmed. The volcano plot showed gene difference with great changes in both statistical significance and magnitude. With just this handful of a wide range of data, we can say with confidence that we have gained significant results when attempting to answer the question of whether cancer can be predicted based on gene expression data. This is crucial, as these samples are a result of liquid biopsies that analyze blood platelets, a relatively new technique that serves to be minimally invasive. With such a heightened promise, these liquid biopsies can serve to better diagnose cancer at greater rate while also not being too much of an inconvenience to the patient. However, large portions of our cancerous samples of gene expression served as outliers, having little relation or proximity to healthy gene expression in our plots. With insufficient evidence to conclude, we can imply these outliers may be insignificant at the moment to cause change. We could also imply the types of cancers they came from may play a role in relation to the healthy samples. In relation to the issue addressed, a question that could still be asked and further studied in the future is “Can a specific type of tumor be identified based on just gene expression data?”

References

Alexa, A., & Rahnenfuhrer, J. (2022). *topGO: Enrichment Analysis for Gene Ontology* (2.50.0). Bioconductor version: Release (3.16). https://doi.org/10.18129/B9.bioc.topGO

Best, M. G., Sol, N., Kooi, I., Tannous, J., Westerman, B. A., Rustenburg, F., Schellen, P., Verschueren, H., Post, E., Koster, J., Ylstra, B., Ameziane, N., Dorsman, J., Smit, E. F., Verheul, H. M., Noske, D. P., Reijneveld, J. C., Nilsson, R. J. A., Tannous, B. A., … Wurdinger, T. (2015). RNA-Seq of Tumor-Educated Platelets Enables Blood-Based Pan-Cancer, Multiclass, and Molecular Pathway Cancer Diagnostics. *Cancer Cell*, *28*(5), 666–676. https://doi.org/10.1016/j.ccell.2015.09.018

Bettegowda, C., Sausen, M., Leary, R. J., Kinde, I., Wang, Y., Agrawal, N., Bartlett, B. R., Wang, H., Luber, B., Alani, R. M., Antonarakis, E. S., Azad, N. S., Bardelli, A., Brem, H., Cameron, J. L., Lee, C. C., Fecher, L. A., Gallia, G. L., Gibbs, P., … Diaz, L. A. (2014). Detection of Circulating Tumor DNA in Early- and Late-Stage Human Malignancies. *Science Translational Medicine*, *6*(224), 224ra24. https://doi.org/10.1126/scitranslmed.3007094

Blighe, K., Rana, S., Turkes, E., Ostendorf, B., Grioni, A., & Lewis, M. (2022). *EnhancedVolcano: Publication-ready volcano plots with enhanced colouring and labeling* (1.16.0). Bioconductor version: Release (3.16). https://doi.org/10.18129/B9.bioc.EnhancedVolcano

Brunson, J. C. (2020). ggalluvial: Layered Grammar for Alluvial Plots. *Journal of Open Source Software*, *5*(49), 2017. https://doi.org/10.21105/joss.02017

Calverley, D. C., Phang, T. L., Choudhury, Q. G., Gao, B., Oton, A. B., Weyant, M. J., & Geraci, M. W. (2010). Significant Downregulation of Platelet Gene Expression in Metastatic Lung Cancer. *Clinical and Translational Science*, *3*(5), 227–232. https://doi.org/10.1111/j.1752-8062.2010.00226.x

Carlson, M. (2019). *org.Hs.eg.db: Genome wide annotation for Human* (3.8.2). Bioconductor version: Release (3.16). https://doi.org/doi:10.18129/B9.bioc.org.Hs.eg.db

Diaz Jr, L. A., & Bardelli, A. (2014). Liquid Biopsies: Genotyping Circulating Tumor DNA. *Journal of Clinical Oncology : Official Journal of the American Society of Clinical Oncology*, *32*(6), 579. https://doi.org/10.1200/JCO.2012.45.2011

Gu, Z. (2022). Complex heatmap visualization. *IMeta*, *1*(3), e43. https://doi.org/10.1002/imt2.43

Gu, Z., Eils, R., & Schlesner, M. (2016). Complex heatmaps reveal patterns and correlations in multidimensional genomic data. *Bioinformatics*, *32*(18), 2847–2849. https://doi.org/10.1093/bioinformatics/btw313

Haber, D. A., & Velculescu, V. E. (2014). Blood-Based Analyses of Cancer: Circulating Tumor Cells and Circulating Tumor DNA. *Cancer Discovery*, *4*(6), 650–661. https://doi.org/10.1158/2159-8290.CD-13-1014

Hawkes, N. (2019). Cancer survival data emphasise importance of early diagnosis. *BMJ : British Medical Journal (Online)*, *364*. https://doi.org/10.1136/bmj.l408

Hoadley, K. A., Yau, C., Wolf, D. M., Cherniack, A. D., Tamborero, D., Ng, S., Leiserson, M. D.

M., Niu, B., McLellan, M. D., Uzunangelov, V., Zhang, J., Kandoth, C., Akbani, R., Shen, H., Omberg, L., Chu, A., Margolin, A. A., van't Veer, L. J., Lopez-Bigas, N., … Stuart, J. M. (2014). Multi-platform analysis of 12 cancer types reveals molecular classification within and across tissues-of-origin. *Cell*, *158*(4), 929–944. https://doi.org/10.1016/j.cell.2014.06.049

John, C. R., Watson, D., Russ, D., Goldmann, K., Ehrenstein, M., Pitzalis, C., Lewis, M., & Barnes, M. (2020). M3C: Monte Carlo reference-based consensus clustering. *Scientific Reports*, *10*(1), Article 1. https://doi.org/10.1038/s41598-020-58766-1

Kandoth, C., McLellan, M. D., Vandin, F., Ye, K., Niu, B., Lu, C., Xie, M., Zhang, Q., McMichael, J. F., Wyczalkowski, M. A., Leiserson, M. D. M., Miller, C. A., Welch, J. S., Walter, M. J., Wendl, M. C., Ley, T. J., Wilson, R. K., Raphael, B. J., & Ding, L. (2013). Mutational landscape and significance across 12 major cancer types. *Nature*, *502*(7471), 333–339. https://doi.org/10.1038/nature12634

Kassambara, A., & Mundt, F. (2020). *factoextra: Extract and Visualize the Results of Multivariate Data Analyses* (1.0.7). https://CRAN.R-project.org/package=factoextra

Kolberg, L., Raudvere, U., Kuzmin, I., Vilo, J., & Peterson, H. (2020). gprofiler2—An R package for gene list functional enrichment analysis and namespace conversion toolset g:Profiler. *F1000Research*, *9*, ELIXIR-709. https://doi.org/10.12688/f1000research.24956.2

Love, M. I., Huber, W., & Anders, S. (2014). Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology*, *15*(12), 550. https://doi.org/10.1186/s13059-014-0550-8

Maechler, M., original), P. R. (Fortran, original), A. S. (S, original), M. H. (S, Hornik [trl, K., maintenance(1999-2000)), ctb] (port to R., Studer, M., Roudier, P., Gonzalez, J., Kozlowski, K., pam()), E. S. (fastpam options for, & Murphy (volume.ellipsoid({d >= 3})), K. (2022). *cluster: "Finding Groups in Data": Cluster Analysis Extended Rousseeuw et al.* (2.1.4). https://CRAN.R-project.org/package=cluster

Schubert, S., Weyrich, A. S., & Rowley, J. W. (2014). A tour through the transcriptional landscape of platelets. *Blood*, *124*(4), 493–502. https://doi.org/10.1182/blood-2014-04-512756

Scrucca, L., Fop, M., Murphy, T., Brendan, & Raftery, A., E. (2016). mclust 5: Clustering, Classification and Density Estimation Using Gaussian Finite Mixture Models. *The R Journal*, *8*(1), 289. https://doi.org/10.32614/RJ-2016-021

WHO. (2022, February 3). *Cancer*. https://www.who.int/news-room/fact-sheets/detail/cancer

Wilkerson, M. D., & Hayes, D. N. (2010). ConsensusClusterPlus: A class discovery tool with confidence assessments and item tracking. *Bioinformatics*, *26*(12), 1572–1573. https://doi.org/10.1093/bioinformatics/btq170

Wu, T., Hu, E., Xu, S., Chen, M., Guo, P., Dai, Z., Feng, T., Zhou, L., Tang, W., Zhan, L., Fu, X., Liu, S., Bo, X., & Yu, G. (2021). clusterProfiler 4.0: A universal enrichment tool for interpreting omics data. *The Innovation*, *2*(3), 100141.

https://doi.org/10.1016/j.xinn.2021.100141

Yu, G., Hu, E., & Gao, C.-H. (2022). *enrichplot: Visualization of Functional Enrichment Result* (1.18.0). Bioconductor version: Release (3.16). https://doi.org/10.18129/B9.bioc.enrichplot

Yu, G., Wang, L.-G., Han, Y., & He, Q.-Y. (2012, May 3). *clusterProfiler: An R Package for Comparing Biological Themes Among Gene Clusters | OMICS: A Journal of Integrative Biology*. https://doi.org/10.1089/omi.2011.0118