Biostatistical Informatics Assignment 1

Steven Mitchell / 40175882

1. Summarise the composition of the data

Out of the 70 patients, 60% were male (N=42) and over two-thirds did not respond to treatment (N=47). The majority of patients (70%) had oesophageal adenocarcinoma, while the remaining patients were diagnosed with gastric adenocarcinoma (N=16) or squamous cell carcinoma (N=11). Patient tumours were categorised mainly as poorly or moderately differentiated (50% and 44% respectively), and had between 1 to 21 positive nodes examined. The median tumour dimensions were 5cm (length), 4cm (width), and 32,000 mm^3 (volume). Protein A levels were reported as low in three-fifths of patients (n = 45). The median survival for the group was 578 days, with a 95% confidence interval of 413-1211 and 56 events (deaths) recorded.

TABLE 1

Data composition summary

Category	Level	N	n
Gender	*Male*	42	32
	Female	34	24
Age (years)	*Median: 64 (Range: 34 - 86)*	70	56
Response	*Non-Responder*	47	42
	Responder	26	11
	NA: 3		
Histology	*Gastric adenocarcinoma*	16	14
	Oesophageal adenocarcinoma	49	35
	Squamous cell carcinoma	11	7
Differentiation	*Poor*	35	29
	Moderate	31	24
	Well	10	3
Number of Positive Nodes	*1*	12	6
	2	6	6
	3	5	4
	4	3	3
	5	6	5
	6	3	3
	7	2	2
	8	4	4
	9	3	3
	10 or greater	8	7
	NA: 24		
Length (cm)	*Median: 5.0 (Range: 1.5 - 16.0)*	70	56
	NA: 9		
Width (cm)	*Median: 4.0 (Range: 0.1 - 15.0)*	70	56
	NA: 9		
Volume (mm^3)	*Median: 32000 (Range: 45 - 1687500)*	70	56
	NA: 9		
Protein A	*High*	34	23
	Low	42	33
Survival (Days)	*Median: 578 (413 LCL - 1211 UCL)*	70	56
	NA: 6		

N = sub-total
n = number of events

2. (a) One of the collaborators is concerned that Protein A levels may be linked to the age of the patient and/or their sex. You are asked to look at the data to see if there is any information to support this.

TABLE 2

Protein A expression vs Gender

	High	Low
Female	16	18
Male	18	24

Methods

To determine if there was an association between gender (male/female) and levels of protein A (high/low), a Pearson's Chi-squared test with Yates' continuity correction and Fisher's Exact Test for Count Data was applied. To determine if there was an association between age and levels of protein A (high/low), a linear regression analysis was utilised. P-value <0.05 was considered significant.

Results

Table 2 shows the contingency table for comparing the distribution of protein A levels against gender. The value of χ^2 is 0.018039, indicating a very small deviation from what would be expected if the two variables were independent. With a p-value of 0.8932, (>0.05), it is suggested there is no significant association between protein A and sex. Fisher's Exact Test for Count Data provided an output p-value of 0.8176, again greater than the significance level 0.05, indicating that there is no significant association between protein A and sex. The sample estimate for the odds ratio is 1.182527, which also suggests that there is no significant association between protein A and sex.

The regression equation is given as *Age = 62.0294 + 0.8992 * Protein A* (*High* protein A = 1, and *Low* protein A = 0). The intercept of 62.0294 represents the estimated mean age of individuals with *Low* Protein A levels, and the slope of 0.8992 indicates the expected change in age associated with an increase in Protein A level from *Low* to *High*. The estimated coefficient for *Low* Protein A is not statistically significant with a p-value >0.05 (p=0.747), which suggests that there is no significant difference in age between individuals with *High* and *Low* levels of Protein A. The R-squared value of 0.00142 signifies that only a small proportion of the variability in age can be explained by the predictor variable, Protein A. The F-statistic of 0.1052 with a p-value of 0.7466 (>0.05) suggests that the regression model is not statistically significant.

(b) Using a univariate non-parametric method, can you provide information to support the collaborators' hypothesis that Protein A predicts survival?

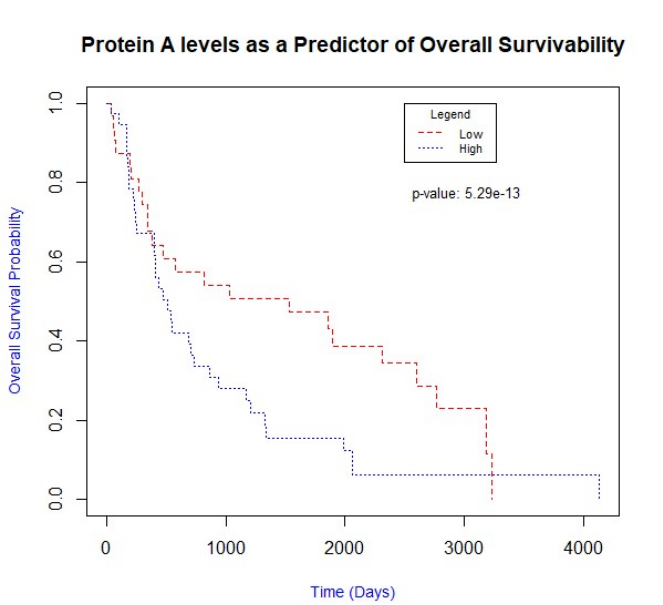

Fig. 1

Kaplan-Meier curves showing the overall survival probability for individuals with high and low levels of Protein A.

Methods

To determine if there was an association between Protein A level and survival, a Kaplan-Meier estimation and the log-rank test were applied. P-value <0.05 was considered significant.

Results

The log-rank test resulted in a significant difference in survival between the two groups ($\chi^2 = 3.9$, p= 0.05), with individuals with low levels of Protein A showing a higher overall survival probability. *Figure 1* shows the Kaplan-Meier plots for the two groups, with Protein A levels (low and high) represented by the red and blue lines, respectively.

3. Another collaborator favours the profiling of genes rather than focusing on measuring one protein by IHC (assignOct2.txt). Which, if any, of the genes have expression levels that are associated with the levels of Protein A?

TABLE 3

Mann-Whitney test results for the association between the expression levels of 14 genes and Protein A.

Gene	P value	Sig (p<0.05)	Sig (p<0.0036)*
GeneA1	0.6532	No	No
GeneA2	0.6244	No	No
GeneA3	0.2789	No	No
GeneA4	0.3782	No	No
GeneA5	0.5415	No	No
GeneA6	0.8492	No	No
GeneA7	0.5764	No	No
GeneA8	0.5188	No	No
GeneA9	0.7124	No	No
GeneA10	0.1466	No	No
GeneA11	0.9681	No	No
GeneA12	0.4575	No	No
GeneA13	0.1199	No	No
GeneA14	0.9266	No	No

*Bonferroni correction

Methods

To determine if there was an association between Protein A levels (High/Low) and expression levels of genes A1-A14, a Mann-Whitney Unpaired test was applied. A p-value less than 0.0036 was considered significant (corrected for multiple comparisons).

Results

Figure 2 shows the box plots of gene expression levels (genes A1-A14) separated into two groups: High Protein A and Low Protein A. It was determined none of the genes demonstrated a significant difference in median levels between protein A level groups ($p > 0.0036$) as shown in Table 3.

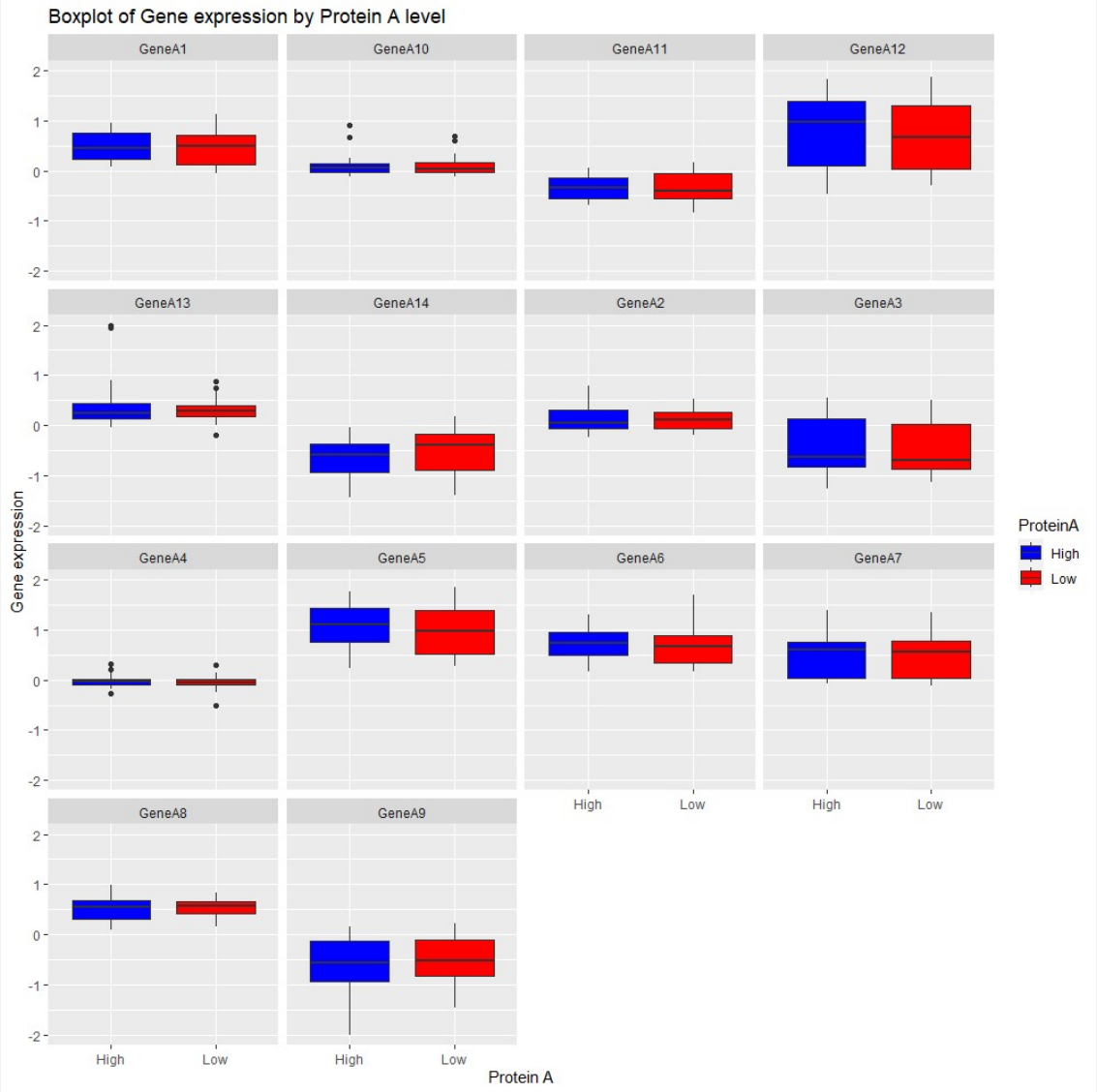

Fig. 2

Boxplots showing the expression levels of genes A1-14 associated with high and low levels of Protein A. Each box represents the median and interquartile range of expression values for a specific gene in either the high or low Protein A condition. The red and blue colours indicate high and low Protein A levels, respectively.

4. (a) The same collaborator wishes to save money and reduce the number of genes profiled. Highlight the genes which have statistically-related expression profiles.

Methods

To identify highly correlated genes a correlation matrix was created, and threshold correlation coefficients of >0.8 was set. To visualise correlation, a heatmap and dendrogram were created. Hierarchical clustering was used, specifying the ward.D2 method.

Results

Correlation coefficients where determined for all genes, with the gene pairs showing the highest statistically related expression profiles shown in *Table 4.* Specifically, GeneA11 and GeneA3 had a correlation coefficient of 0.85, GeneA6 and GeneA5 0.92, GeneA12 and GeneA5 0.87, and GeneA8 and GeneA7 0.85. These findings suggest that these genes may have similar functions and may be co-regulated. Therefore, it could be suggested that only one column of genes in *Table 4* ('Gene a' or 'Gene b') could be used without losing important information or compromising the results of downstream analyses. If further reduction was required to save cost, *figures 3 & 4 p*rovide further indications of correlations that may exist in the data set and could be explored further.

TABLE 4

GeneA pairs with correlation coefficients greater than 0.8

Gene a	Gene b	Correlation Coefficient
A6	A5	0.92
A12	A5	0.87
A8	A7	0.85
A11	A3	0.85

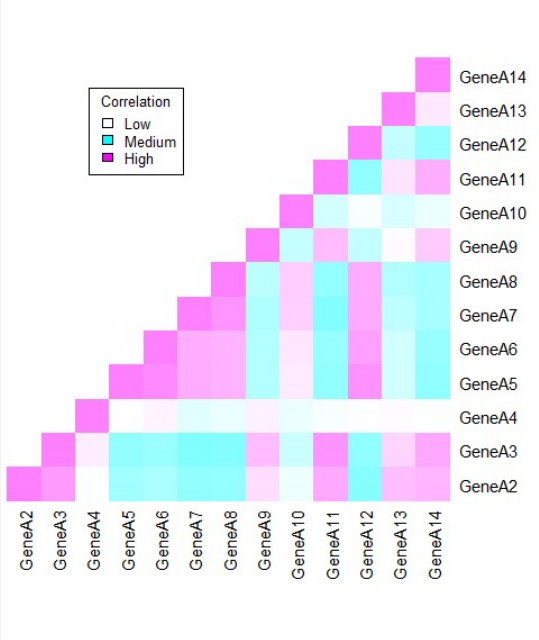

Fig. 3

Heatmap of the correlation matrix of gene expression data. The colour scale represents the strength of the correlation, with cyan indicating a lower correlation, and magenta indicating a higher correlation.

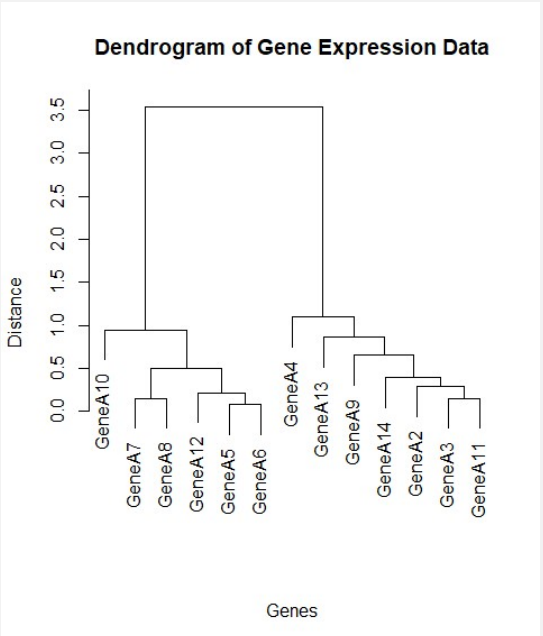

Fig. 4

Dendrogram of gene expression data showing the hierarchical clustering of genes based on their expression levels. Shorter branch lengths indicate higher similarities between clusters.

(b) The collaborators are interested in the relationships between genes A2, A9 and A14. Given the observed strength of the relationships between these three genes (based on expression levels), what is the minimum number of samples in the next phase of the study required to achieve 80% power?

Methods

The R function pwr.t.test was used to calculate the effect size using the means and standard deviations of the three gene expression groups. The minimum sample size for 80% power at a 5% significance level was then calculated.

Results

The analysis showed the minimum sample size required for 80% power was 224. Therefore, if the sample size is less than 224, the study may not have sufficient power to detect an effect of the size calculated using the means and standard deviations of the three gene expression profiles. However, if the sample size is 224 or greater, the study will have an 80% chance of detecting the effect size calculated at a 5% significance level ($p = <0.05$).

5. You are interested in the behaviour of the genes profiled, as a group. You carry out clustering (Euclidean/Ward.d2) (you are not required to use bootstrapping).

(a) Using visual inspection, select the most defined cluster structure.

Methods

A scaled data frame was created to construct a dendrogram using hierarchical clustering to group genes that have similar expression patterns across different samples. The dissimilarity matrix was calculated using Euclidean distance, and Ward's method (D2) was used to determine the clusters. Visual inspection was used to identify a defined cluster structure. These findings were supported by the use of two statistical methods, the gap statistic method and the silhouette method, to further estimate the optimal composition of cluster structure. After determining the optimal number of clusters, the cutree() function is used to allocate each observation to a cluster. A boarder was drawn around each cluster of the dendrogram, and a supporting scatter plot was created to further visualise the cluster structure.

Results

From examining the dendrogram shown in *figure 5*, a two-part cluster structure was initially identified. Analysis using the gap statistic method also suggested a two-cluster structure was optimal, further supported by the gap statistic output of k=2, as shown in *figure 6*. The scatterplot in *figure 7* also indicates that the two clusters are well separated from each other, suggesting that the genes in each cluster have similar expression profiles.

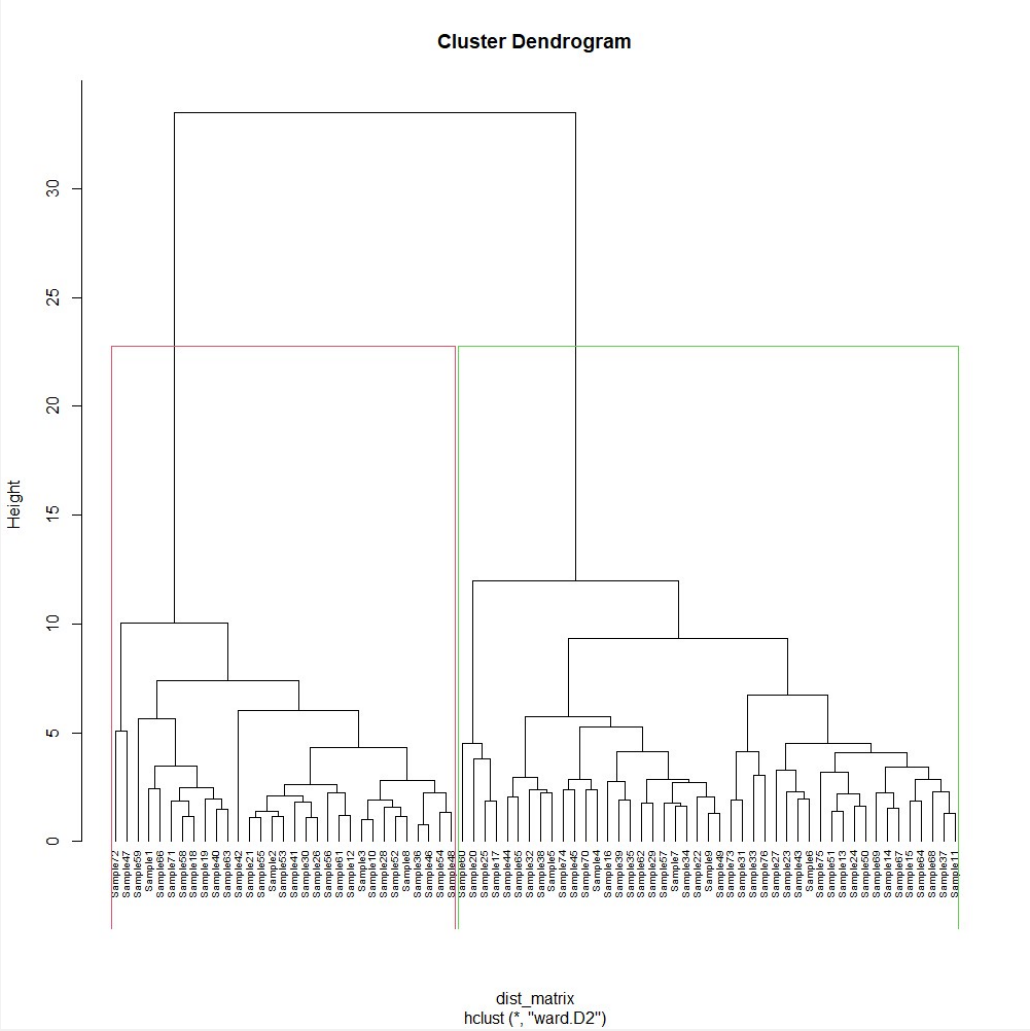

Fig. 5

Dendrogram of gene expression data showing the hierarchical clustering of genes based on their expression levels. Shorter branch lengths indicate higher similarities between clusters.

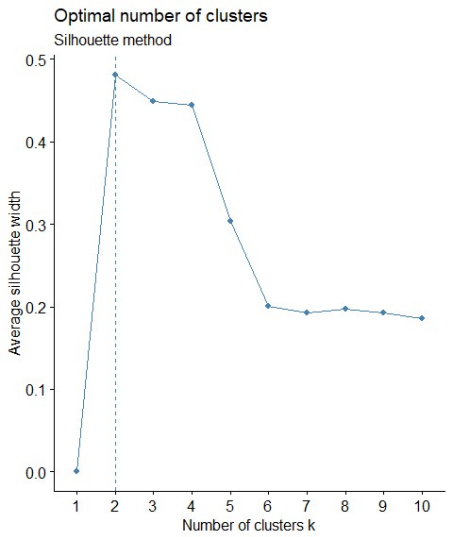

Fig. 6

Estimation of the optimal number of clusters using the silhouette method.

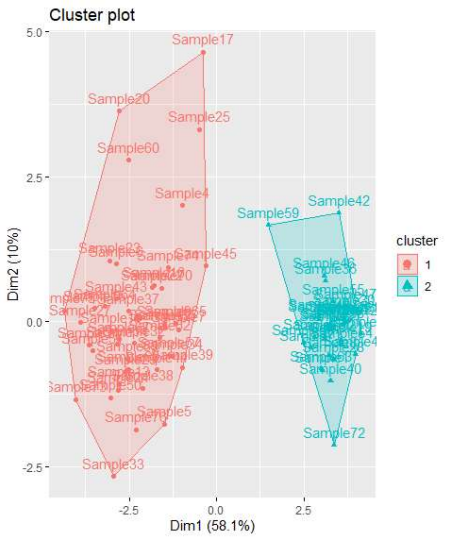

Fig. 7

Cluster scatter plot showing the distribution of data points in the two clusters.

(b) How do the cluster labels relate to Differentiation and Histology?

Methods

A contingency table was created for each variable against the cluster labels. For both differentiation and histology, a chi-squared test was performed along with a Fisher's exact test to provide information about how the cluster labels relate to differentiation and histology. By analysing the p-values, the presence of a statistically significant association between the cluster labels and the two variables can be determined.

Results

As shown in *table 5*, for the differentiation variable, a p-value of 0.34 was observed with the chi-squared statistical test. The accompanying Fisher's exact test producing a p-value of 0.3626. Therefore, no statistically significant relationship between the cluster labels and differentiation variable was observed. In contrast, for the histology variable, Pearson's Chi-squared test produced a p-value of 0.0477, and the Fisher's exact test produced a p-value of 0.04633. Therefore, with both p-values less than 0.05, it is indicated there is a statistically significant relationship between the cluster labels and histology variable.

TABLE 5

Comparison of Chi-squared and Fisher's exact tests for Differentiation and Histology variables.

Variable	Chi-squared	Fisher's exact
Differentiation	0.34	0.3626
Histology	0.0477	0.04633

(c) Survival (using a univariate, non-parametric method)?

Methods

To determine the relationship between cluster labels and survival, a Kaplan-Meier survival analysis was used. Kaplan-Meier survival curves were plotted for each cluster using the "ggsurvplot" function, followed by further analysis using the log-rank test to compare the survival curves between the clusters. The output from the log-rank test provided a Chi-squared statistic, degrees of freedom, and a p-value. Additionally, a Cox proportional hazards model was performed to assess the relationship between the cluster labels and survival, with outputs of p-value, concordance statistic, and Wald, likelihood, and score statistics. The outputs obtained from the statistical tests were used to determine if there was a significant association between cluster labels and survival.

Results

Kaplan-Meier analysis was performed to determine the relationship between cluster labels and survival. *Figure 8* shows the survival curves for each cluster, which showed no statistically significant relationship between survival and cluster label (p=0.55). As shown in *table 6*, the log-rank test indicated that there was no significant difference in survival between the two clusters (p=0.6). Furthermore, the Cox proportional hazards model also showed no significant association between cluster labels and survival (p=0.6, concordance=0.6).

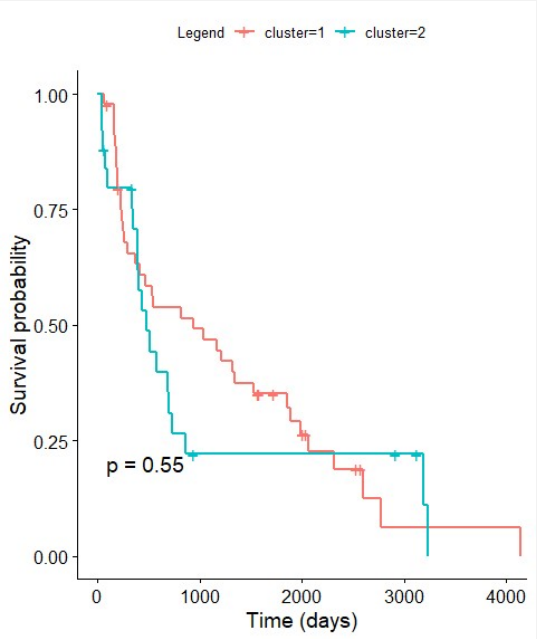

Fig. 8

Kaplan-Meier survival curves for each cluster. The curves represent two clusters identified in the dataset, with no statistically significant difference observed in survival between the two clusters (p=0.55).

TABLE 6

Table 1: Results of Statistical Tests. The table summarizes the output and p-values of the Kaplan-Meier, Log-Rank, and Cox regression tests.

Statistical Test	Output	P value
Kaplan-Meier	*p*	0.55
Log-Rank	*p*	0.6
Cox	*Concordance*	0.6
	Wald test	0.6
	Score	0.6

```
# Appendix

getwd()

#QUESION 1

  # Read in data to data frame. Header row confirm, first col names confirm
    assign1 <- read.table("assignOct1.txt", header=TRUE, row.names=1)

  # Display first rows of df - check read in
    head(assign1)

  # display df dimensions
    dim(assign1)

  # categorical variables

    table(assign1$Gender)      # Gender
    table(assign1$Response)    # Response
    table(assign1$Histology)   # Histology
    table(assign1$Differentiation)  # Differentiation
    table(assign1$ProteinA)  # Protein A
    table(assign1$PositiveNodes)    # PositiveNodes
    table(assign1$Event)       # Events

    table(assign1$Gender,assign1$Event)      # Gender (+ events '1' vs non-events '0')
    table(assign1$Response,assign1$Event)    # Response (+ events)
    table(assign1$Histology,assign1$Event)   # Histology (+ events)
    table(assign1$Differentiation,assign1$Event)  # Differentiation (+ events)
    table(assign1$ProteinA,assign1$Event)          # Protein A (+ events)
    table(assign1$PositiveNodes,assign1$Event)     # PositiveNodes   (+ events)
          summary(assign1$PositiveNodes) #PosNodes NAs

        # Vector to store number of positive nodes
          num_nodes <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, "10 or greater", "NA")

        # Quant of positive nodes
          node_counts <- table(ifelse(is.na(assign1$PositiveNodes), "NA",
ifelse(assign1$PositiveNodes >= 10, "10 or greater", assign1$PositiveNodes)))

        # Create table + print
          node_table <- data.frame(Number_of_nodes = num_nodes, Count = node_counts)

          print(node_table)

  # continuous variables
    summary(assign1$Length)      #Length Median, Range, NAs
    summary(assign1$Width)       #Width
    summary(assign1$Vol)         #Volume
    summary(assign1$Age)         #Age

  # Median survival
    library(survival)
    fit <- survfit(Surv(assign1$Survival, assign1$Event) ~ 1)
    print(fit)


# QUESTION 2

  # part a : Protein A levels linked to age or sex

    # protein a v sex
      proteinA.vs.Sex<- table(assign1$Gender,assign1$ProteinA)
      proteinA.vs.Sex
      # table
        chisq.test(proteinA.vs.Sex)
          ## output = (p-value) 0.8932 therefore non-significant
        fisher.test(proteinA.vs.Sex)
          ## output = (p-value) 0.8176 therefore non-significant
```

```
      # protein a v age
         lm.fit <- lm(assign1$Age ~ assign1$ProteinA, data = assign1)
      # display results
         summary(lm.fit)
          ## output = Multiple R-squared:  0.00142 (0.14% of age variation explained by
Protein A level), p-value: 0.7466 (no significant relationship level & age)


  # part b : univariate non-parametric method

    # load package
      library(survival)

    # create survfit object & Kaplan-Meier curves for Protein A levels
      KM.Test <- survfit(Surv(assign1$Survival,
assign1$Event)~assign1$ProteinA,data=assign1)

    # perform a log-rank test
      survdiff(Surv(assign1$Survival, assign1$Event)~assign1$ProteinA,data=assign1)
      ## log-rank output: Chisq= 3.9  on 1 degrees of freedom, p= 0.05

    # create plot of the Kaplan-Meier
      plot(KM.Test, main="Protein A levels as a Predictor of Overall Survivability",
col.main="black",
           xlab="Time (Days)", ylab="Overall Survival Probability",
           col.lab="blue", cex.lab=0.9,col=c("red","blue"), lty = 2:3)
      legend(2500, 1.0, title="Legend",c("Low","High"),
             lty = 2:3,col=c("red","blue"),cex=0.7)
      # add legend to plot
        legend(2300, .82, c("p-value: 5.29e-13"), cex=0.8,box.col="white")

# QUESTION 3

    #Stat test
      # Read in data
        assign2 <- read.table("assignOct2.txt", header=TRUE, row.names=1)
        #check
          head(assign2)
        # display df dimensions
          dim(assign2)
      # sort by Sample ID
        sort1<-assign1[sort(row.names(assign1)),]
        sort2<-assign2[sort(row.names(assign2)),]
      # Wilcoxon rank sum test for each gene in assign2
        p_values <- numeric(length = ncol(assign2))
        for (i in seq_along(p_values)) {
          p_values[i] <- wilcox.test(sort2[,i] ~ sort1[,4])$p.value
        }
      # create a data frame to store results
        results <- data.frame(Gene = colnames(assign2), P_value = p_values)
      # sort the results by p-value
        results_sorted <- results[order(results$P_value),]
      # view for excel
        View(results_sorted)

    #boxplots
      #standard individual
        # create a vector of colors for the two groups
          colors <- c("blue", "red")
        # create the boxplot with customized colors
          boxplot(assign2$GeneA1 ~ assign1$ProteinA, data=assignOct2,
                  xlab="Protein A", ylab="Gene A1 expression",
                  main="Boxplot of GeneA1 expression by Protein A level",
                  col=colors)

      #combined

        # load package
```

```
          library(ggplot2)
        # create a data frame with the relevant columns
          df <- data.frame(
            GeneA = rep(paste0("GeneA", 1:14), each = length(assign1$ProteinA)),
            ProteinA = rep(assign1$ProteinA, times = 14),
            Expression = c(assign2$GeneA1, assign2$GeneA2, assign2$GeneA3,
                           assign2$GeneA4, assign2$GeneA5, assign2$GeneA6,
                           assign2$GeneA7, assign2$GeneA8, assign2$GeneA9,
                           assign2$GeneA10, assign2$GeneA11, assign2$GeneA12,
                           assign2$GeneA13, assign2$GeneA14)
          )
        # create the boxplot using ggplot2
          ggplot(df, aes(x = ProteinA, y = Expression, fill = ProteinA)) +
            geom_boxplot(size = 0.5) +
            scale_fill_manual(values = c("blue", "red")) +
            labs(x = "Protein A", y = "Gene expression",
                 title = "Boxplot of Gene expression by Protein A level") +
            facet_wrap(~ GeneA, ncol = 4)


# QUESTION 4

      #part a

        #cor matrix data
          # extract gene expression columns
            genes<- assign2[, 2:14]
          # create correlation matrix
            cor_matrix <- cor(genes)
          # identify genes with high correlation coefficients
            high_corr_genes <- which(cor_matrix > 0.8 & cor_matrix < 1, arr.ind=TRUE)
          # remove duplicate gene pairs
            high_corr_genes <- high_corr_genes[!duplicated(t(apply(high_corr_genes, 1,
sort))),]
          # print the pairs of highly correlated genes and their correlation
coefficients
            cat("Pairs of highly correlated genes (correlation coefficient > 0.8):\n")
            for (i in 1:nrow(high_corr_genes)) {
              gene1 <- colnames(genes)[high_corr_genes[i,1]]
              gene2 <- colnames(genes)[high_corr_genes[i,2]]
              corr <- cor_matrix[high_corr_genes[i,1], high_corr_genes[i,2]]
              cat(gene1, "and", gene2, "with correlation coefficient", round(corr, 2),
"\n")
               }

        #heatmap visual
          # extract gene expression columns
            genes<- assign2[, 2:14]
          # create correlation matrix
            cor_matrix <- cor(genes)
          # remove duplicates
            cor_matrix[lower.tri(cor_matrix)] <- NA
          # create heatmap
            heatmap(cor_matrix, Rowv=NA, Colv=NA, col = cm.colors(256), scale="none")
            # legend
              legend("topleft",
                     legend = c("Low","Medium", "High "),
                     fill = c("white","cyan", "magenta"),
                     title = "Correlation",
                     cex = 0.8)

          #dendrogram visual
            # extract gene expression columns
              genes <- assign2[, 2:14]
            # create correlation matrix
              cor_matrix <- cor(genes)
            # create distance matrix
              dist_matrix <- 1 - cor_matrix
            # hierarchical clustering
```

```
              hc <- hclust(as.dist(dist_matrix), method="ward.D2")
            # plot dendrogram
              plot(hc, main="Dendrogram of Gene Expression Data", xlab="Genes", sub="",
ylab="Distance")


        #part b
            # load package
              library(pwr)
            # calculate effect size (d) using the means and standard deviations of
three groups
            n <- pwr.t.test(n = NULL,
                            d = abs(diff(c(mean(assign2$GeneA1[assign2$GeneA1<0]),
                                           mean(assign2$GeneA2[assign2$GeneA2<0]),

mean(assign2$GeneA9[assign2$GeneA9<0])))/sd(c(assign2$GeneA1[assign2$GeneA1<0],

assign2$GeneA2[assign2$GeneA2<0],

assign2$GeneA9[assign2$GeneA9<0]))))),
            # min sample size for 80% power at 5% sig lvl
              sig.level = 0.05,
              power = 0.8,
              type = "one.sample")
            # print to the nearest integer
              cat("Minimum sample size required for 80% power:", ceiling(n))



# QUESTION 5

      # part a

          # read in
             df.5a <- assign2
            # remove missing values
              df.5a <- na.omit(assign2)
            # standardise
              df.5a <- scale(df.5a)
            # check
              head(df.5a)

          # dendrogram
            # compute dissimilarity matrix using Euclidean distance
              dist_matrix <- dist(df.5a, method = "euclidean")
            # Hierarchical clustering using Ward d2
              cluster_model <- hclust(dist_matrix, method = "ward.D2" )
            # plot dendrogram
              plot(cluster_model, cex = 0.6, hang = -1)

          # estimate  optimal number of clusters
            # gap statistic method
              library("factoextra")
              set.seed(123)
              fviz_nbclust(df.5a, kmeans, nstart = 25,  method = "gap_stat", nboot =
500)+
                labs(subtitle = "Gap statistic method")
                ## output 2
            # silhouette method
              fviz_nbclust(df.5a, FUN = hcut, method = "silhouette")+
                labs(subtitle = "Silhouette method")
                ## output 2

           # allocate clusters
              # cut tree into 2 clusters
              cluster_labels <- cutree(cluster_model, k = 2)
              # members of each cluster
                table(cluster_labels)
              # plot dendrogram
```

```
          plot(cluster_model, cex = 0.6, hang = -1)
        # draw cluster border
          rect.hclust(cluster_model, k = 2, border = 2:5)

      # cluster scatter plot
         fviz_cluster(list(data = df.5a, cluster_labels = sub_grp))



# part b

      # contingency table for cluster labels and differentiation
        cont_table_dif <- table(cluster_labels, assign1$Differentiation)
        # perform chi-squared test
          chi_dif <- chisq.test(cont_table_dif)
          # results
             chi_dif
        # fishers
          fisher_dif <- fisher.test(cont_table_dif)
           # results
              fisher_dif


      # contingency table for cluster labels and histology
        cont_table_his <- table(cluster_labels, assign1$Histology)
        # perform chi-squared test
          chi_his <- chisq.test(cont_table_his)
          # view the results
            chi_his
        # fishers
          fisher_his <- fisher.test(cont_table_his)
          # results
            fisher_his

# part c

              # Kaplan-Meier survival analysis
                library(survival)

              # create survival object
                surv_obj <- Surv(time = assign1$Survival, event = assign1$Event)

              # create dataframe with  cluster labels & survival object
                cluster_surv <- data.frame(cluster = cluster_labels, surv_obj)

              # plot Kaplan-Meier survival curves for each cluster
                ggsurvplot(survfit(surv_obj ~ cluster, data = cluster_surv),
                           pval = TRUE,
                           legend.title = "Legend",
                           xlab = "Time (days)",
                           ylab = "Survival probability")

              # log-rank test to compare the survival curves between the clusters
                survdiff(surv_obj ~ cluster, data = cluster_surv)
                  ## output: Chisq= 0.4  on 1 degrees of freedom, p= 0.6


              # cox
                # perform Cox proportional hazards model
                  cox_model <- coxph(surv_obj ~ cluster_surv$cluster)
                # print summary of Cox proportional hazards model
                  summary(cox_model)
                  ## output: p = 0.552, concordanc= 0.534, wald + liklihood +
score all = 0.6
```