Introduction

Colorectal cancer has become the most prevalent cancer over the past couple decades, now accounting for around a tenth of cancer mortality in the United States (Kuipers et al). This cancer occurs when colon or rectum cells grow uncontrollably, and often takes effect in the mucus-creating cells of these internal organs. The increase in colorectal cancer in the West is largely due to an overall increase in aging individuals in the population and poor lifestyle habits (Hoffman et al).

As a result of this prevalence, it is vital that we have a proper understanding of this cancer and the factors affecting it. One such factor are genes. Genes are important as they contain the instructions that are used to synthesize bodily proteins which are needed for our cells. It is mutations to these genes that so often result in uncontrollable cell growth and division which could lead to cancers such as colorectal cancer. Data from the Human Genome Project has found that humans have over 20,000 genes, many of which could have an effect on the survival rate of a patient.

For this project, I was interested in studying the gene MYC Proto-Oncogene, BHLH Transcription Factor, or MYC for short. The MYC transcription factor often leads to up-regulation of other genes that are vital in the cell cycle process, apoptosis as well as other cellular processes (Surat et al). This effectively increases the overall number of cells which can eventually lead to the development of cancer. My project poses the question of what populations, between young and old, tend to over-express the gene MYC? In addition to this, how does over-expression of gene MYC affect survival?

Methods

Firstly, the appropriate packages for the project were dowloaded. The TCGA colon cancer clinical and RNAseq data were accessed with the R package TCGAbiolinks and DESeq2. This was installed using BiocManager in order to get the data from the Summarized Experiment. Next to access the packages, the library() command was used allowing everything to be loaded into the R session.

The first thing I had to do was ensure that the MYC gene I was interested in was actually included in the genes provided in the Summarized Experiments. Upon this verification, I then wanted to categorize the data in colData into a new column that divided the patients by age. This was done using an ifelse statement, with 50 as the threshold for the age. I then wanted to check for NA values in this new column called Age Category. After verifying the number of NAs, I removed them and make a new variable to store the Age Category data without these NAs. For the gene MYC, I needed to find the row that corresponded to it in the appropriate column. Then I created a mask so that I would have easy access to the gene.

In terms of the plots used, I chose to use a barplot to visually display the amount of old vs young patients in the data Summarized Experiments. I also chose to make a boxplot to represent the Age Category and the gene counts. Finally, I wanted to use a Kaplan-Meier plot to look at survival based on age.

Results

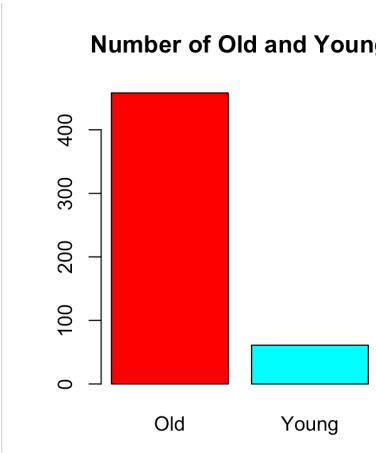

Figure 1: Number of Old and Young Patients

The first graph is a bar graph plot that represents how many young and old patients there are. This was important information in this project as the patients were split by age in order to see how the data in Summarized Experiments was divided.

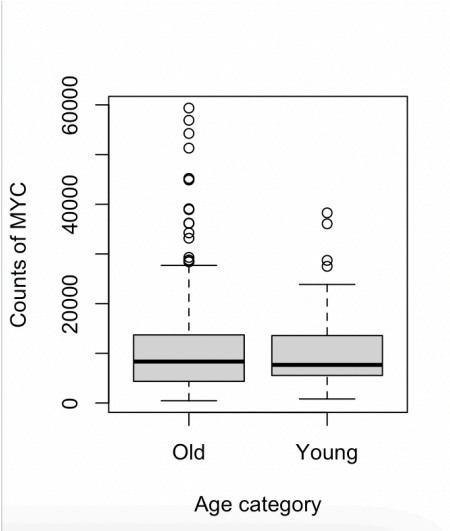

Figure 2: Counts of MYC vs Age Category

Discussion

The results of this project were quite interesting. Firstly, and quite unsurprisingly, the number of "old" patients far exceeded the number of "young" patients in the data set sum_exp. This corroborates the idea that although various cancer types can develop at any age, the likeliness of developing cancer generally increases with age (National Institute of Health). According to medical research, this is due to changes in the body's biochemical processes that occur with age. It is these processes that regulate and control genes so this results in the altering of vital methyl groups that affect the DNA as well as sites for protein synthesis.

In addition to this, the box plot shows the amount of counts of the MYC gene. This count data from RNA-seq represents the number of reads that overlap this gene. From this plot, we see that the old category once again has more counts of the MYC gene. This is also interesting because research shows that MYC is a gene that commonly over expressed. Taking a look at how gender affects survival rate, the plot shows that men have a lower survival time than women. The survival plot also shows that there is a correlation between the over-expression of the gene and reduce survival times. This is further explained by findings in Hofmann et al which explain how the over-expression of this important gene make cells more sensitive and prone to cell apoptosis, especially of the death ligands necrosis factor as well as other tumor necrosis related factors. In short, MYC over-expression leads to a higher rate of cancer tumor formation which decreases survival rate (Gabay et al).

References

Gabay, Meital et al. “MYC activation is a hallmark of cancer initiation and maintenance.” *Cold Spring Harbor perspectives in medicine* vol. 4,6 a014241. 2 Jun. 2014, doi:10.1101/cshperspect.a014241

Hoffman, B., Liebermann, D. Apoptotic signaling by c-MYC. *Oncogene* **27,** 6462–6472 (2008). https://doi.org/10.1038/onc.2008.312

Hofmann, Jeffrey W et al. “Reduced expression of MYC increases longevity and enhances healthspan.” *Cell* vol. 160,3 (2015): 477-88. doi:10.1016/j.cell.2014.12.016

Kuipers, Ernst J et al. “Colorectal cancer.” *Nature reviews. Disease primers* vol. 1 15065. 5 Nov. 2015, doi:10.1038/nrdp.2015.65

“NIH Study Offers Insight into Why Cancer Incidence Increases with Age.” *National Instituteof Health*, U.S. Department of Health and Human Services, 8 Sept. 2015, https://www.nih.gov/news-events/news-releases/nih-study-offers-insight-into-why-cancer-incidence-increases-age.

P, Surat. "Myc Mutations and Cancer". *News-Medical*. 04 March 2022. <https://www.news-medical.net/health/Myc-Mutations-and-Cancer.aspx>.

Review Questions

General Concepts:

1. What is TCGA and why is it important?

   - TCGA stands for The Cancer Genome Atlas and this is a cancer genome program that includes data from all sorts of patients and cancer types. TCGA is so important because it has helps with improvements in the ability to diagnose, treat, and prevent cancers that are prevalent in society today.

2. What are some strengths and weaknesses of TCGA?

   - There are many strengths of TCGA along with certain weaknesses. A benefit is it has large datasets that we are able to access at a little to no cost, the data is broad and generally accurate. Despite this, a drawback is that since there is so much information it can be almost too much or intimidating and can cause some confusion. They may also require a very good system to properly download all the material.

3. How does the central dogma of biology (DNA → RNA → protein) relate to the data we are exploring?

   - TCGA employs many methods to categorize DNA, RNA and proteomic data and allows us to draw conclusions based on the analyses we conduct in R as we learned in class.

Coding Skills:

1. What commands are used to save a file to your GitHub repository?

   When you are in the terminal you can **cd** into the correct local repository on your computer and then you can use the **git status** command to make sure that the correct updates will be made once the files have been saved. Then you can use **git add** and then **git commit -m**, which is a short message that gives detail on the file being saved to the repository. Finally, **git push** is used to save onto GitHub.

2. What command must be run in order to use a package in R?

   - When you want to use a package in run, it must first be installed. This is done by using the install.packages("") command and then to actually use the package you use the library() command to load it into the R session.

3. What is boolean indexing? What are some applications of it?

   - Boolean indexing helps select data in a data frame, especially if you don't know where exactly you don't know where exactly the data is found in a specific row or column. So you can use a mask and apply boolean vectors to a column or row to get your data. An application we used in our class assignments was to delete NA values in data. Another common example we used was when we subset for "young" and "old" in the data frames.

4. Draw out a dataframe of your choice. Show an example of the following and explain what each line of code does.

1. an ifelse() statement

- This example is based on the clinic data frame that we use in class.

- clinic$age_category = ifelse(clinic$age_at_index < 50, “young", “old”)

- Firstly, everything to the left of the equal sign is the name of the new variable being made. To the right of the equal sign we have the ifelse statement. The dollar sign means that in the clinic data, we are specifically looking at the age_at_index data. So if the value here is less than 50, then this will be assigned the word “young”. If the value here is greater than or equal to 50, then at this variable age category, the word assigned will be “old”.

2. boolean indexing

- clinic_data_young = clinic[clinic$age_category == "young",]

- Continuing on with the same data frame used above, this example uses the age_category variable that was just made using the ifelse statement. This is just one example of boolean indexing when making the variable called clinic_data_young. Here boolean indexing is used to subset the young patients into their own data frames. Then we are inside the clinic data and looking specifically at the new column made called age_category, which is represented by the $. Then we want only the “young” data so we use the == operation.