Bioinformatics Portfolio

Emily Gillings

Index:

What did I hope to learn?

This module is an exciting area of science that I have never had the opportunity to explore. I had some experience with coding from when I was younger, and it was always something that I enjoyed. Therefore, Current Topics in Bioinformatics was something that appealed to me, and I was curious to learn something completely new.

Not only this, in parallel with this module I was able to get some work experience at King's College London online in the Bioinformatics sector. Therefore, I was keen to use what I learned from the TAB module to help with my work experience and provide me with a better understanding.

I am always keen to broaden my learning and not limit myself to sectors of science and more that are in my comfort zone. Therefore, I wanted to challenge myself to take a module that I am not familiar with and discover what I can learn from this. The coding modules I have done in the past have focused on learning the coding language and applying this. However, I was keen to discover the connection between biological studies and coding. I wanted to understand the application of this technology in the science sector and what this could represent for future research and discoveries. Not only this, I was also interested in the prevalence that genetics can have on the likelihood of developing a disease, and how programming can be used to predict events such as these.

In total, I chose this module to be able to develop new skills in an area of science I have not before covered. Furthermore, I was hopeful that this module would enable me to apply what I have learnt in theory so far in my degree.

ClevR-vis: Innovative visualisation techniques for clonal evolution

Overall project goal:
Clara's team set about integrating functional data taken from various platforms, such as GWAS, with data from PopHumanVar in order to produce an interactive tool for the analysis of clonal evolution. In other words, she had to consider the current visualisation techniques available for this analysis and work on improving their drawbacks through a new app, known as the clevR-vis tool.

What is clonal evolution?
This essentially outlines the way in which tumours will progress or evolve. A single, normal cell will divide multiple times. During these divisions, many mutations can be acquired, most of which do not impact gene function. However, some mutations can occur that cause the cell to escape regulation and therefore begin uncontrollably dividing.

A clone can be defined as a group of cells with the same mutation. Further cell differentiation can cause these clonal lineages to diverge further and produce what is known as intratumour heterogeneity.

To further understand this concept, I decided to research it further.

Intratumour heterogeneity: the distinct tumour cell populations within the same tumour specimen. In terms of cancer, this represents both genetic and phenotypic diversity in cancer cell populations. This issue is the main cause of treatment failure in patients: different mutations give a higher likelihood of resistance to therapy.

Ramón, S., Sesé, M., Capdevila, C., Aasen, T., De Mattos-Arruda, L., Diaz-Cano, S., Hernández-Losa, J., Castellví J (2020). *Clinical implications of intratumor heterogeneity: challenges and opportunities.* J Mol Med (Berl). 98(2):161-177.

Models of Cloning Evolution:

1. Linear

This is where new mutations provide strong selective advantage that outcompetes previous clones. Punctuated linear evolution is an extremist version of this, where all evolution happens quickly at early stages, where one mutation outcompetes the rest.

2. Branching

This occurs when clones develop in parallel with each other. There are two types. This first is dependent branching, where the clones from a common ancestor. Independent therefore refers to the opposite: they are derived from different cells. Finally, neutral is clonal evolution where the mutations provide no change to the fitness of the cell.

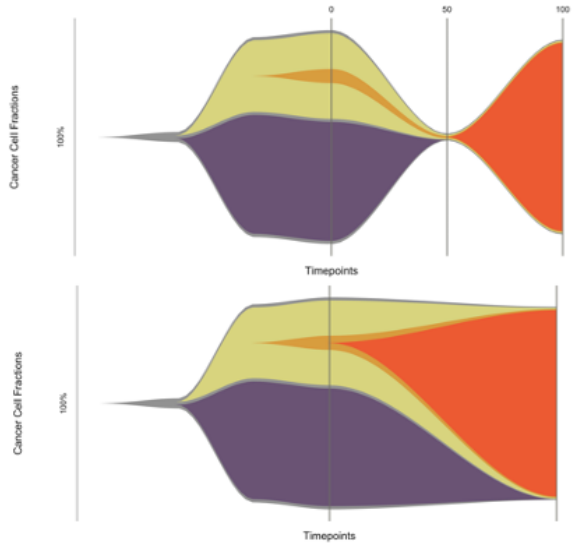
Reconstructing Clonal Evolution:

With this understanding of clonal evolution, Clara then helped provide insight into the data processing of these samples collected. There are three main steps for this reconstruction.

1. Clustering = this involves sequencing the data to produce a mutational profile which indicates the intratumour heterogeneity. This involves annotating the data to assign the percentage that each variant makes up the sample. These become clustered together to indicate variants that represent a single clone.
2. Reconstruction = with this annotated data, a clonal evolution tree can then be produced. This helps indicate the relationship between individual clones in a 'dot and line' diagram.
3. Visualisation = This data can then be visualised further using either Fishplot or Timescape.

Visualisation:
The two current visualisation techniques, Fishplot and Timescape, both have drawbacks which hinder their ability to produce effective clonal evolution maps. As a result, Clara set about taking inspiration from both these techniques and producing a completely new platform with improved features.

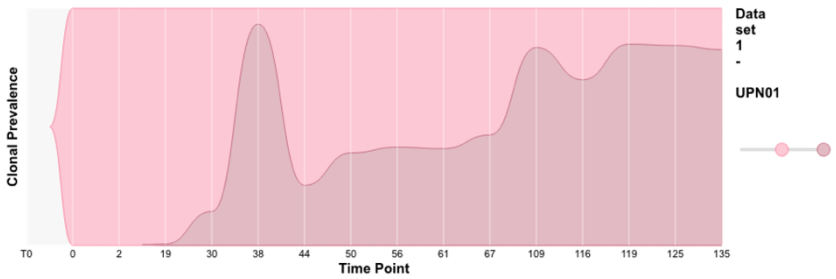
Fishplot visualises different mutations over time points within a tumour, as shown on the right.



In this example, the grey mutation initiated the tumour growth, but was quickly dominated by the purple and yellow mutations. As therapy was applied, the tumour was effectively reduced at time point 50. However, the orange mutation had resistance to the therapy and so the tumour began again.

One of the most prevalent with this tool is that if timepoints 0 and 100 were the only ones outlined, then the introduction of therapy would remain unknown. This was a key part that Clara wanted to address in the new platform.

Timescape is the second visualisation technique, shown below. This technique is not capable of visualising the tumour-sized changes, however, when there are many clones and only few timepoints, it can prove more useful than a fish plot. This is because it can easily show all of the clones at all time points.



Therefore, Clara wanted to take the benefits from both of these techniques and produce an improved platform. On top of the issues already addressed, the new platform aimed to

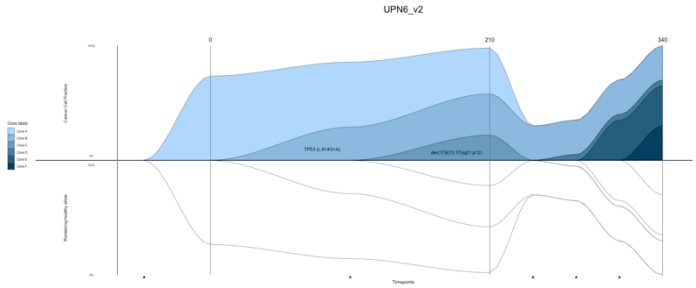
consider variance in different alleles. In other words, whether two variants that affect the same gene are on the same or two different alleles. This can help determine whether there is a remaining healthy copy of an allele.

The clevR-vis pipeline…
This ultimately produced an application capable of combatting all these drawbacks. This essentially holds data from clonal evolution patterns in a database called SeaObjects.

From this, three different types of visualisation plots can be produced. Shark is the tree plot; dolphin is the fish plot; and the final one is plaice, which addresses allelic variation.

Extra timepoints can be added to the SeaObject table. This enables greater visualisation as the impact of various therapies can be detailed.

The most prevalent change compared to the Fishplot or Timescape is the Plaice plot. This essentially gives a dolphin plot with a mirrored version, as shown on the right.



Using this template, a researcher can then manually add colours to the bottom half depending on allelic variance.

For example, say clone A (in light blue) affects gene TP53. Then you discover than clone C also affects the same gene. As a result, you can infer that the variance are in both alleles and so there is no healthy copy available. To visualise this, you paint underneath clone C with clone A's colour.

My conclusions:
Overall, I think that this project is an extremely interesting application of bioinformatics. The production of this easy-to-use and visually appeasing app can be extremely in the future. In particular, it could be used to help clarify which mutations within a tumour are producing resistance to a therapy. As a result, new therapies could be produced to help combat this.

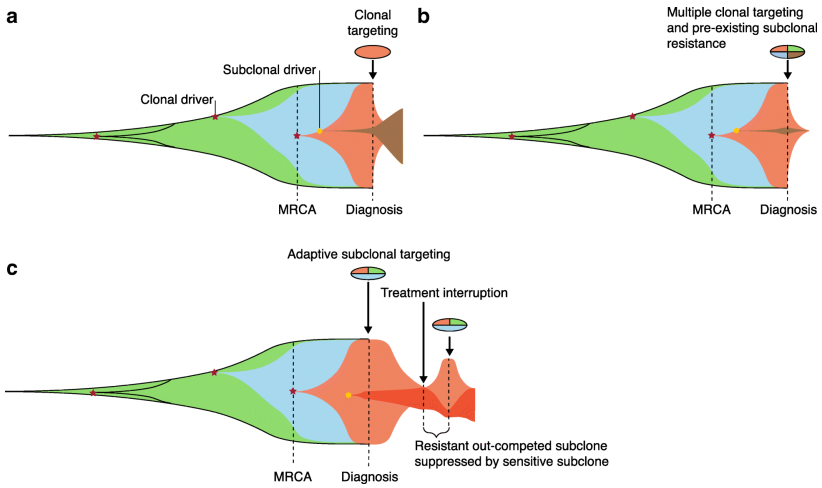
This could be an incredible valuable tool in the cancer research sector that offers new capabilities that had not yet been appreciated.

Further Discussion
To explore this further, I looked into what successes have currently occurred which have produced effective therapies using clonal evolution. Looking through various research papers, it appears that there has been little success in utilising clonal evolution to impact clinical medicine for cancer treatment. Most of the suggestions for its use in therapy remains relatively theoretical and has yet to be successfully used.

The main problem faced is the heterogeneity of a tumour. Therapies produced are only capable of producing short lived responses as new mutations can arise with resistance.

The most effective method to attack these tumours is to target each individual mutation. This is where the new clevR-vis technology could be extremely effective as it can better visualise all of the potential mutations and therefore improve the reliability of deciphering therapy responses.

The figure on the right represents the current target for therapy in terms of gene therapy. These are visualised using Fishplot.



In figure a, the therapy only targets one clonal population. Therefore, this is less likely to eliminate the tumour as resistance from other clones are likely.

In figure b, therapy targets multiple different clones within the tumour sample. This is therefore more likely to help cause extinction of the tumour.

Finally, figure c represents the most effective therapy that has been produced. This is known as adaptive therapy. Here, treatment is discontinued just before sensitive cells, which are not resistant to the therapy, are completely eradicated. This enables them to grow back and therefore repress the resistant clones.

Fittall, M., Van Loo, P (2019). *Translating insights into tumour evolution to clinical practice: promised and challenges.* Genome Med. 11: 20.

In summary:

The above treatments are largely theoretical and have had little success. However, it may be that with the introduction of better visualisation tools, such as clevR-vis, these statistics could change. This technology may provide better insight into more detailed clonal evolution and with its ability to input customised timepoints, there is less room for error. I think this could be an exciting step forward in the cancer research sector.

What did I learn?

I have thoroughly enjoyed this module and feel that I have learnt a range of bioinformatical skills that will benefit me in the future. In particular, this module has been particularly useful alongside my work experience; I have been able to apply the coding skills I learnt from TAB to the Long COVID data I was working with.

The section I enjoyed the most from TAB was definitely working in R studio, as this is particularly relevant to my work experience. This involved utilising techniques to produce graphical visuals of example data. In this way, these graphics can be interpreted and analysed. I learnt how to produce visually appealing scatter graphs, bar graphs and more using the ggplot2 graphical system. This was useful when performing my work experience, as it gave me the knowledge to produce graphs of long COVID data to reach a conclusion.

Furthermore, learning about Genome-Wide Association Studies (GWAS) was another interesting area of the module that I learnt. I have covered GWAS previously at my home university, however, it was useful to go into greater depth about how these are used and applied in the real world. For example, at my home university I was taught about the theory behind Manhattan plots and how they are used. Therefore, it was intriguing to be able to produce my own Manhattan plots in this module to further my understanding.

This concept is reinforced in RNA-seq. This is another approach that I have briefly covered at my home university in theory. This module gave me the opportunity to learn about the coding language necessary to produce expression profiling.

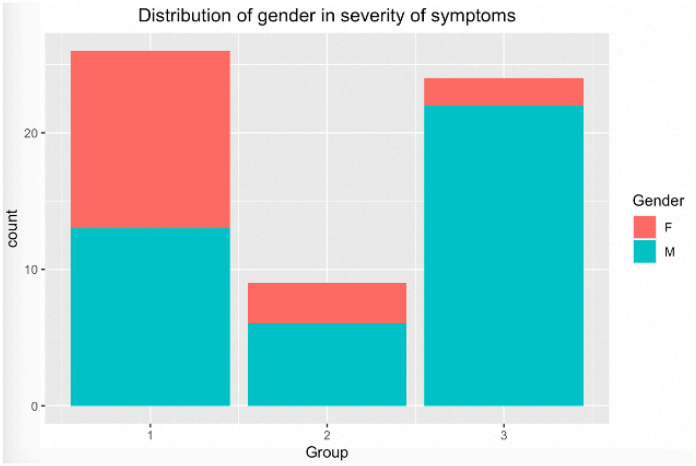
In total, in this module I have learnt how to both manipulate and visualise data. This has involved learning the coding language in both Jupyter Notebook and R studio in order to produce data that can be both interpreted and analysed. This has proven incredibly useful when applied to my work experience. Not only this, I believe that these skills I have acquired will help me in the future for my Biochemical degree.

Calaix de sastre

I had the opportunity to perform work experience alongside my bioinformatics module last semester. This involved using R studio to create a code capable of analysing long covid data. The main aim was to discover whether specific genes contribute to the likelihood of a patient developing long covid. The gene that encodes the growth factor hormone appeared to have the most prevalence, however, a larger data set is necessary to determine the reliability of this.

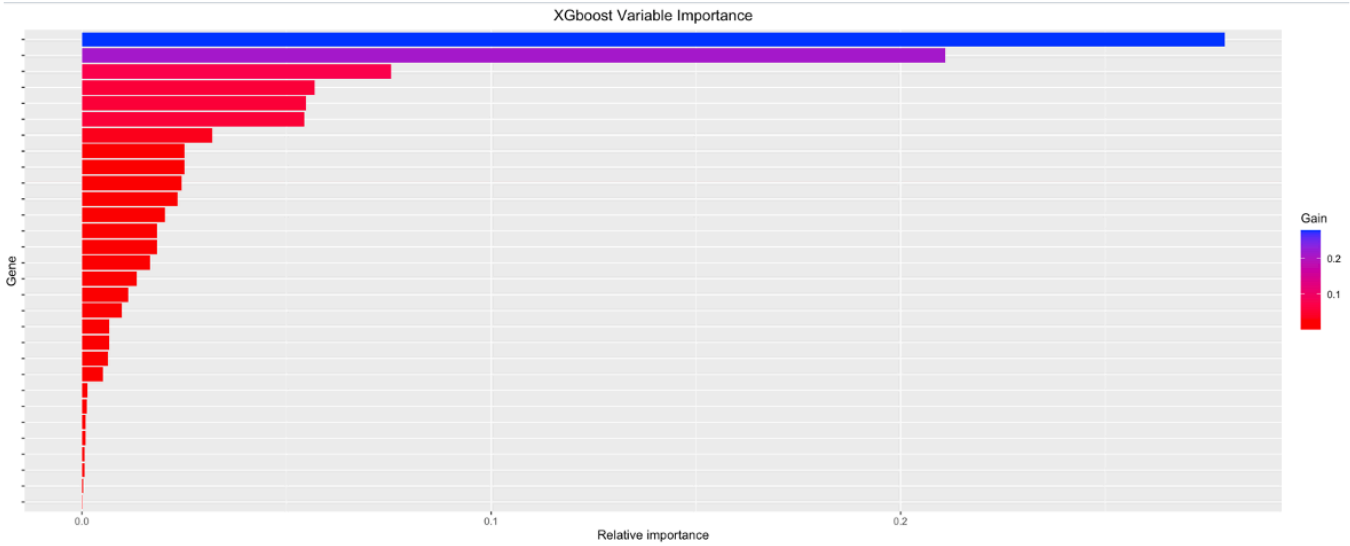
Below I have attached various sections of code that this module has helped me perform successfully. This included understanding how to produce different graph types, utilising colours and headings to give clear, well-presented figures that can be interpreted.

```
p<-ggplot(data = long_covid_data, aes(x = Group, fill = Gender))+
  geom_bar()+
  ggtitle("Distribution of gender in severity of symptoms")+
  theme(plot.title = element_text(hjust = 0.5))
p
```



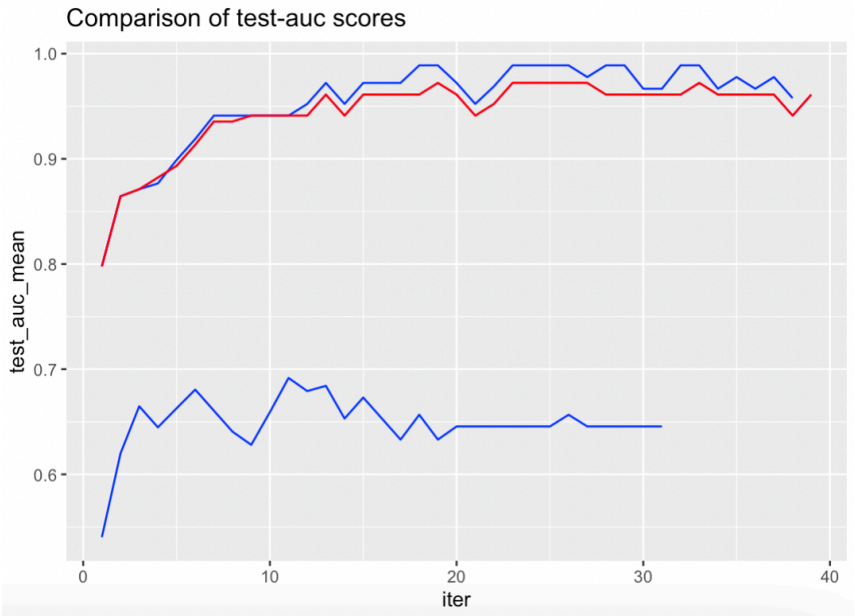
Bar plot to show distribution of gender in severity of symptoms. Predominantly, males show moderate to severe symptoms, more evenly spread for mild. However, consider disproportionate numbers of males and females in sample.

```
# plot predictive features to see which are most important based on gain value.
ggplot(model_1_importance, aes(x=reorder(Feature,Gain), y=Gain, fill=Gain)) +
  geom_bar(stat="identity", position="dodge") +
  coord_flip() +
  ylab("Relative importance")+
  xlab("Gene") +
  ggtitle("XGboost Variable Importance") +
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_fill_gradient(low="red", high="blue") +
  theme(axis.text.y=element_blank())
```



The model is heavily dependent on age (top feature), with a second closely following importance. However, the rest all still contribute.

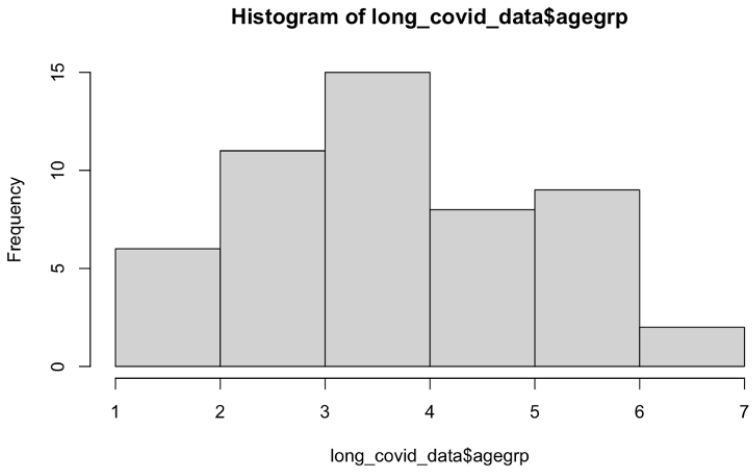
```
# plot
ggplot() +
  geom_line(data = xgb_model_1_cv$evaluation_log, aes(x = iter , y = test_auc_mean), color = "blue") +
  geom_line(data = RFE_results_highest_auc_log, aes(x = iter, y = test_auc_mean), color = "blue") +
  geom_line(data = max_depth_tune_highest_auc_log, aes(x = iter, y = test_auc_mean), color = "blue") +
  geom_line(data = min_child_weight_tune_highest_auc_log, aes(x = iter, y = test_auc_mean), color = "red") +
  ggtitle("Comparison of test-auc scores")
```



Comparing test auc scores for the minimum child weight.

```
table(long_covid_data$Age)
pie(table(long_covid_data$Age))
long_covid_data$agegrp[long_covid_data$Age>=20 & long_covid_data$Age<=29]<-20
long_covid_data$agegrp[long_covid_data$Age>= 30 & long_covid_data$Age<=39]<-30
long_covid_data$agegrp[long_covid_data$Age>=40 & long_covid_data$Age<=49]<-40
long_covid_data$agegrp[long_covid_data$Age>=50 & long_covid_data$Age<=59]<-50
long_covid_data$agegrp[long_covid_data$Age>=60 & long_covid_data$Age<=69]<-60
long_covid_data$agegrp[long_covid_data$Age>=70 & long_covid_data$Age<=79]<-70
long_covid_data$agegrp[long_covid_data$Age>=80 & long_covid_data$Age<=89]<-80
table(long_covid_data$agegrp)

long_covid_data$agegrp<-as.numeric(long_covid_data$agegrp)
hist(long_covid_data$agegrp)
barchart(long_covid_data$agegrp)
```



Histogram to show distribution of age in the data. 1 is the youngest age group, 20-29. Groups are separates into 10 years.