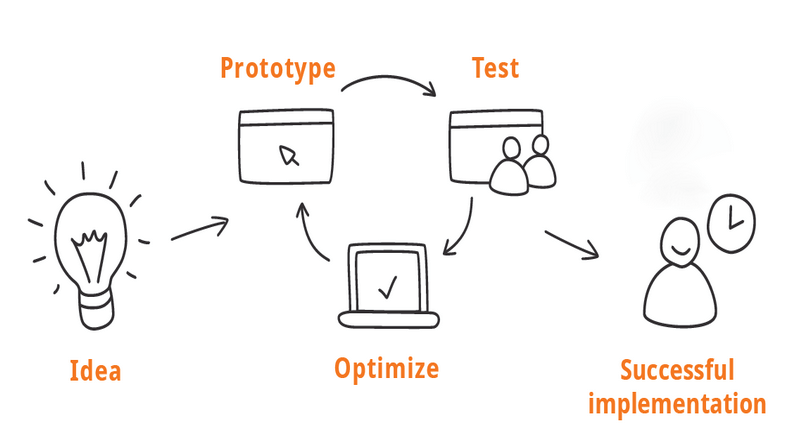
**Programming for Prototyping**



shorturl.at/korBF

**Robert Gatt**

**SM4391**

**9th April 2023**

**SMc20374**

**Programming for prototyping**

**Assignment 2 of 2**

**Section A**

**A1. Acceptance Document**

**User stories**

1. As a researcher, I want to be able to upload a covariance matrix text file in order to analyse my data.
2. As a researcher, I want to be able to upload a text file that includes all the fiducial parameters in order for the application to calculate the fisher matrix.
3. As a researcher, I want to be able to choose two parameters in order for the application to draw the error ellipses.
4. As a researcher, I would like to plot the error ellipses on a graph in the application.
5. As a researcher, I would like to see the error ellipses on the application.
6. As a researcher, I would like to be able to save a statistical report to a text file.

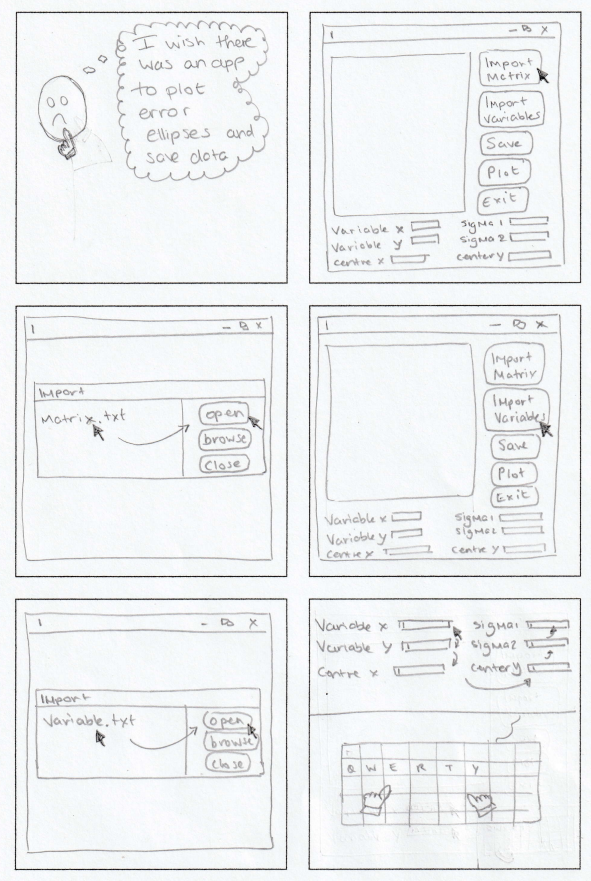
**Constraints and limitations**

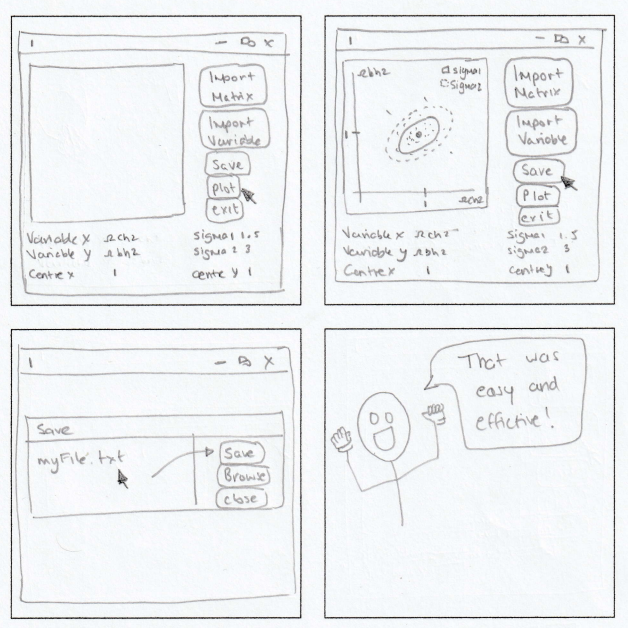
1. This application needs to perform complex mathematical calculation on a large set of data; thus, the constraint is the limited processing resources, which might vary from user to user.
2. The application needs to be designed for cross-platform compatibility in order to support a large number of users. Cross-platform compatibility could therefore be a restriction.
3. The application's user interface (UI) must be simple, intuitive, and provide all the information, which may be a constraint given that different types of study may require different amounts of data.
4. Security is an issue; the files will be saved as a text file, which might be leaked or overwritten.
5. Depending on the level of functionality and features required by the research team, time restrictions in producing the final MVP (minimum viable product) could be a problem.
6. Because there are only a limited number of resources available, developing an application requires a budget, which may affect the number of features and functionality that the finished product will have.

**Prototypes that will be delivered**

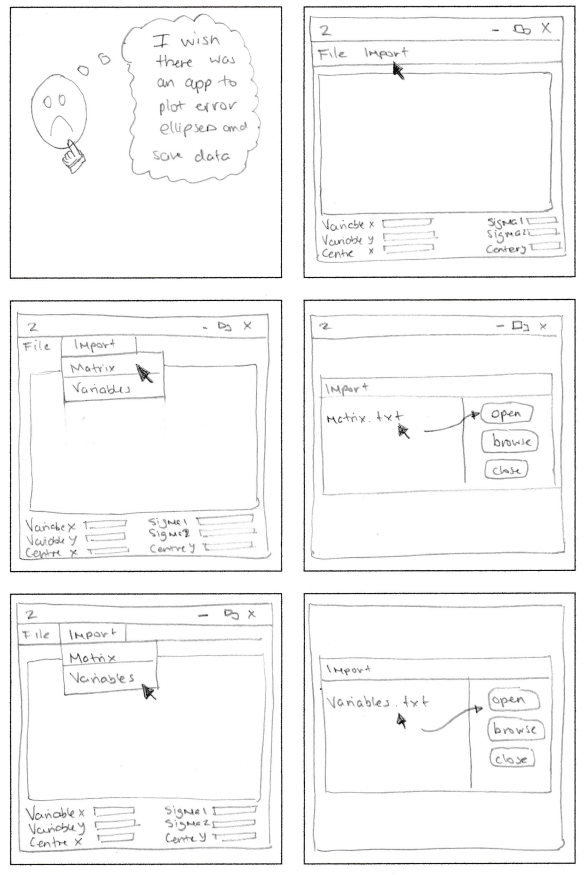
1. Hand drawn storyboard portraying the environment, sequence and satisfaction of utilizing the application.
2. Couple of low-fidelity prototypes created using office supplies in order to choose the best option to prototype.
3. A PowerPoint mock-up of one of the chosen low-fidelity prototypes that will have the logical flow of how the user can use the application.
4. A high-fidelity mock-up of the GUI.
5. A high-fidelity graphical plot using a covariance matrix, with the following five parameters for the X and Y coordinates:
   1. Ωbh 2 vs Ωch 2,
   2. Ωch 2 vs w,
   3. ln(A) vs ns,
   4. τ vs w, and,
   5. user defined plots

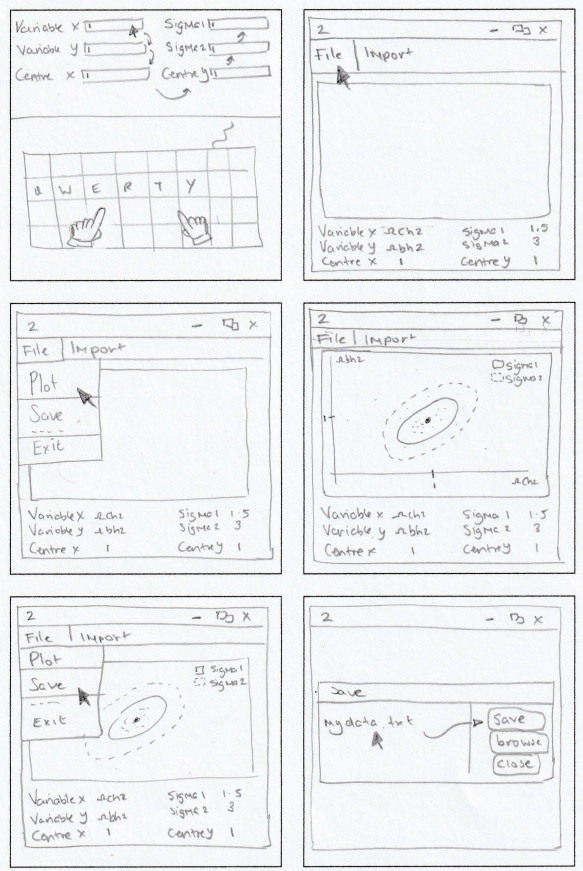
**A2.1 Storyboard – Prototype 1**

****

****

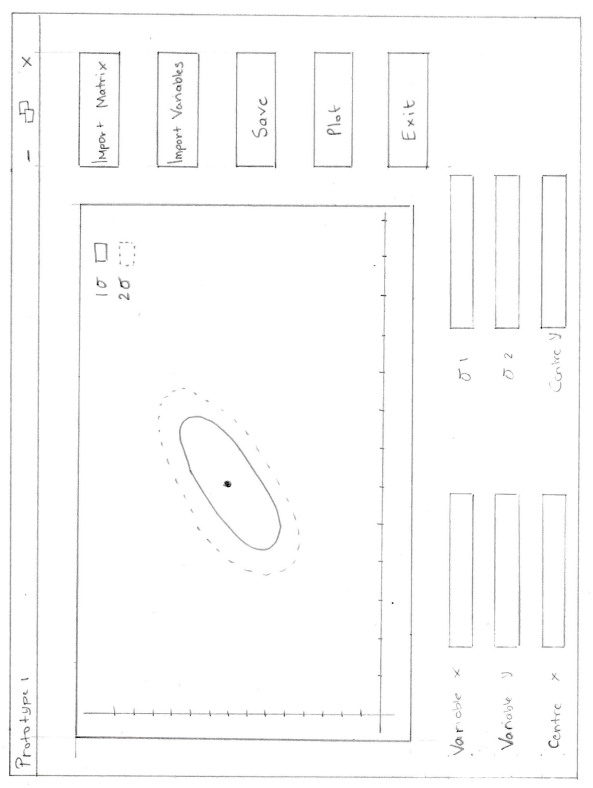
**A2.2 Storyboard – Prototype 2**

****

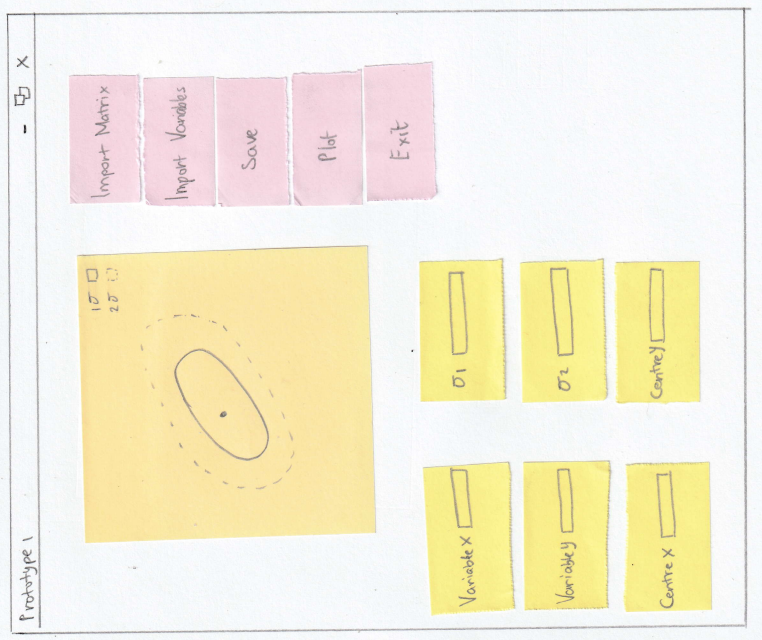
****

****

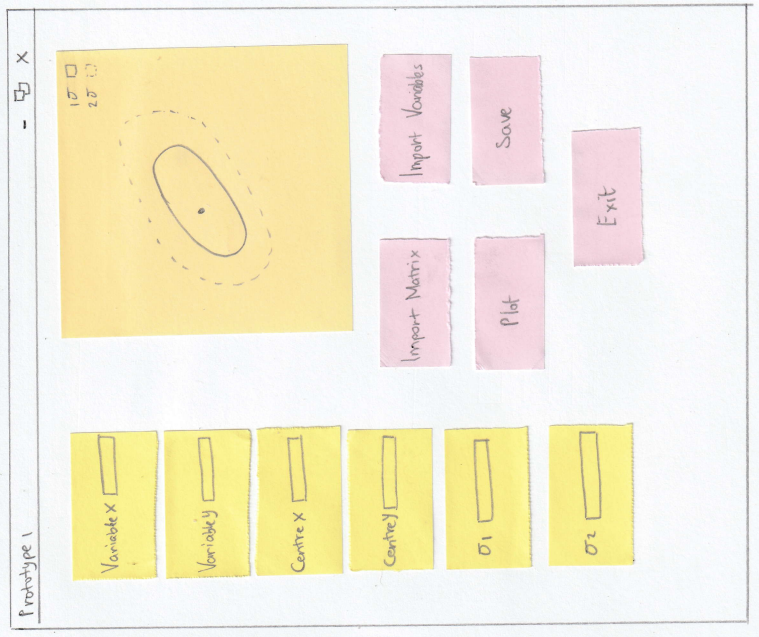
**A3.1.1 Office-stationary based prototype – hand-drawn sketch for prototype 1**

****

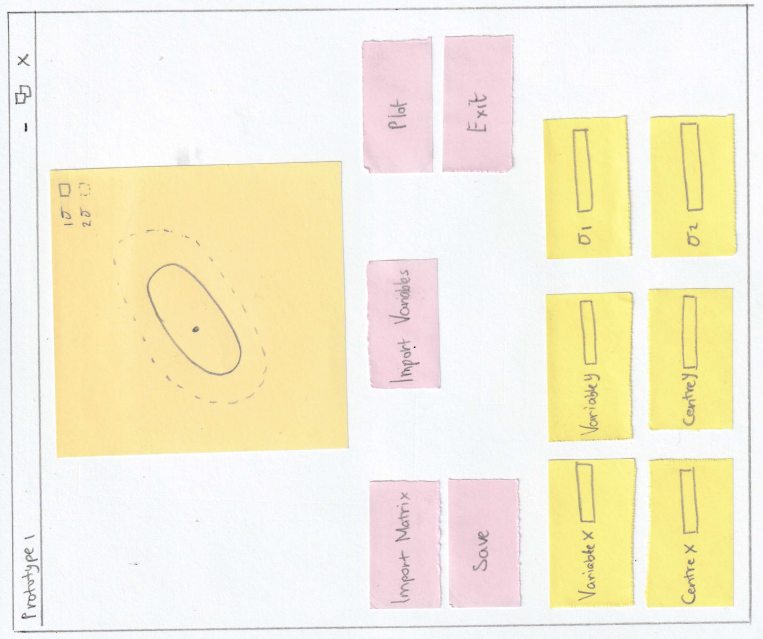
**A3.1.2 Office-stationary based prototype – variation one for prototype one**

****

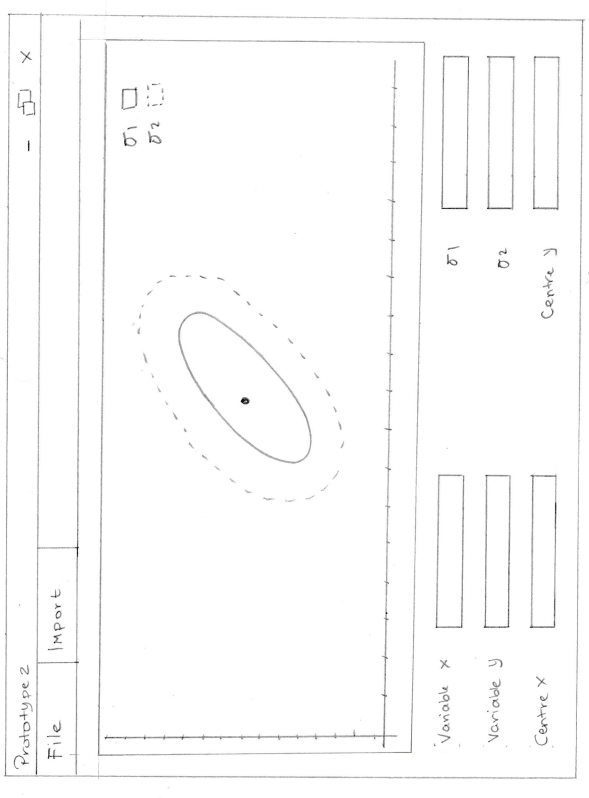
**A3.1.3 Office-stationary based prototype – variation two for prototype one**

****

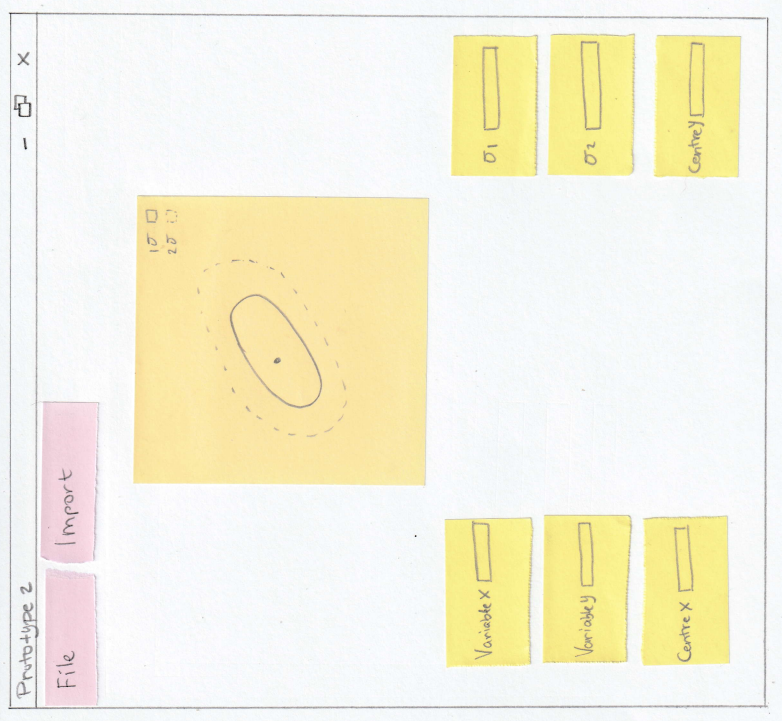
**A3.1.4 Office-stationary based prototype – variation three for prototype one**

****

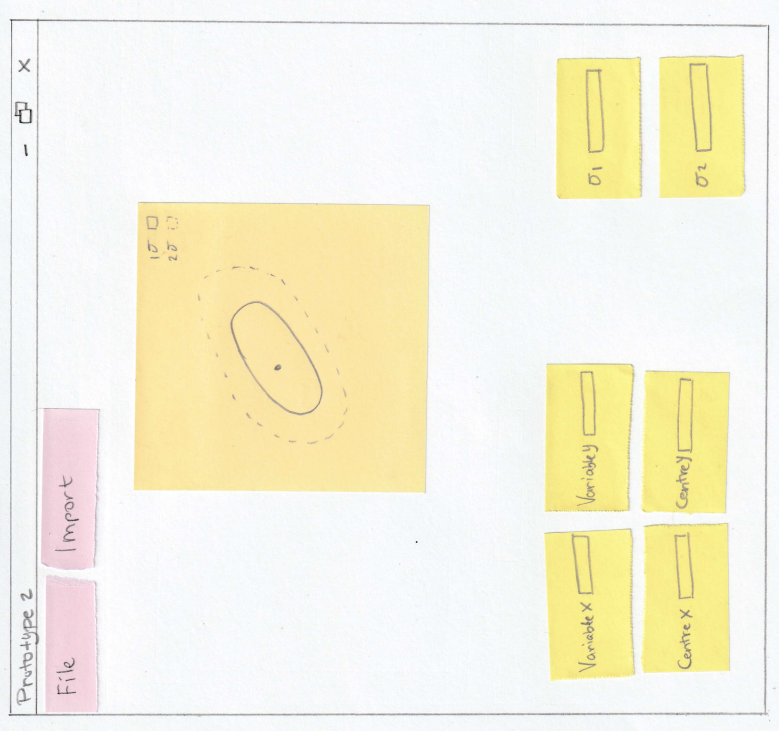
**A3.2.1 Office-stationary based prototype – hand-drawn sketch for prototype 2**

****

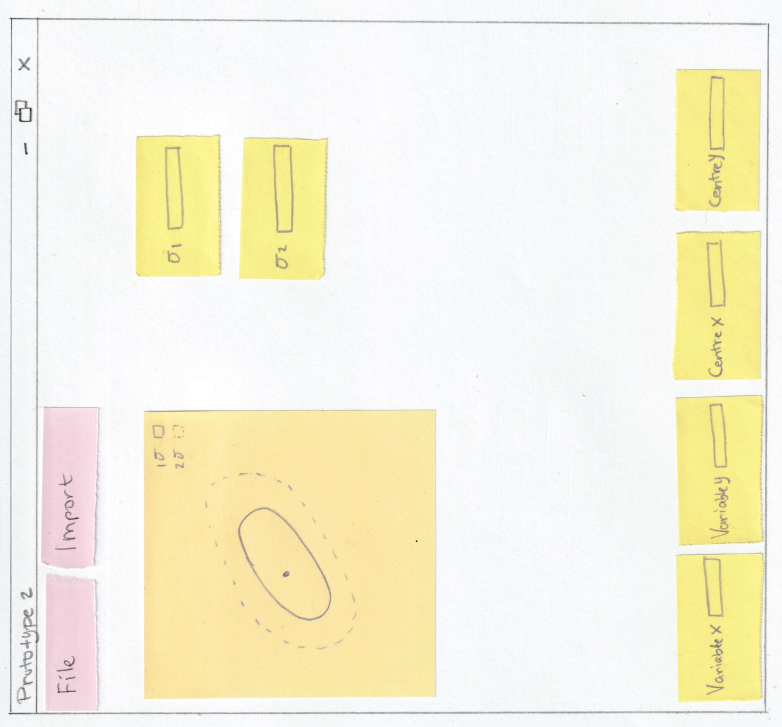
**A3.2.2 Office-stationary based prototype – variation one for prototype two**

****

**A3.2.3 Office-stationary based prototype – variation two for prototype two**

****

**A3.2.4 Office-stationary based prototype – variation three for prototype two**

****

**A4. Power point mock-up**

*This is enclosed in the same zip folder as a separate file.*

**Section B**

**B1. Plan**

The plan for creating this prototype is to first develop the main UI to demonstrate to users how the application would look. Afterward, the main application will be implemented to display and plot five ellipses using a covariance matrix and five parameters for the X and Y values, four of which are predefined, and one is defined by the user.

* Ωbh 2 vs Ωch 2,
* Ωch 2 vs w,
* ln(A) vs ns,
* τ vs w, and,
* user defined plots

This prototype will make use of three libraries:

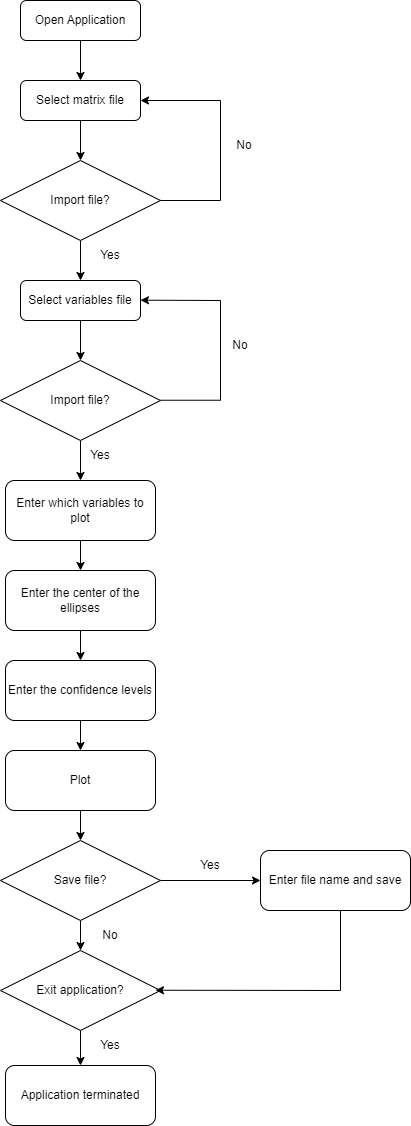
* tkinter
* matplotlib
* numpy

Tkinter is a built-in GUI package that provides a standard Python interface (Docs, 2023). Its main advantage is that it is easy to use and allows for the creation of fast prototypes. At this stage, the application does not contain many buttons or user inputs, making it easy to modify the UI based on user feedback. However, if the application were to grow and researchers require more functions and features, tkinter may need to be replaced by more sophisticated GUI packages such as PyJamas and WxPython.

*“Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Matplotlib makes easy things easy and hard things possible.”* (matplotlib, 2023)*.* Hence, the use of this library is ideal for this prototype, as it offers a wide variety of features, including plot and scatter, which are necessary for creating the required visualizations.

The last package to be used for this prototype is NumPy. According to its official website, NumPy provides “*powerful n-dimensional arrays*” (NumPy, 2023) and “*numerical computing tools*” (NumPy, 2023), making it ideal for the type of prototype required. Specifically, NumPy will be used to import the covariance matrix in *.covmat* file format and to compute the Fisher matrix using built-in functions for matrix inversion.

**B2. Flow chart**



**B3. Interface – Documentation**

The work in section A served as the foundation for the following GUI, which is an illustration rather than a finished product.

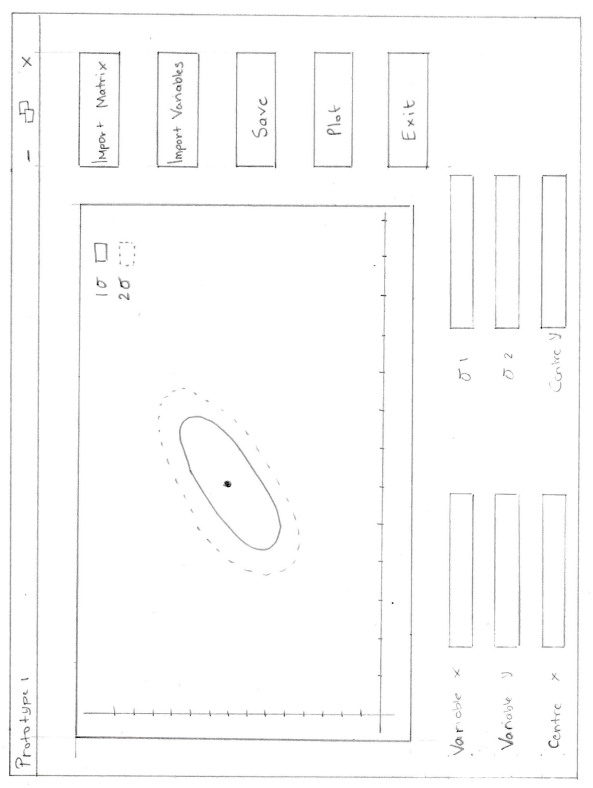
****

Figure 1 - Prototype

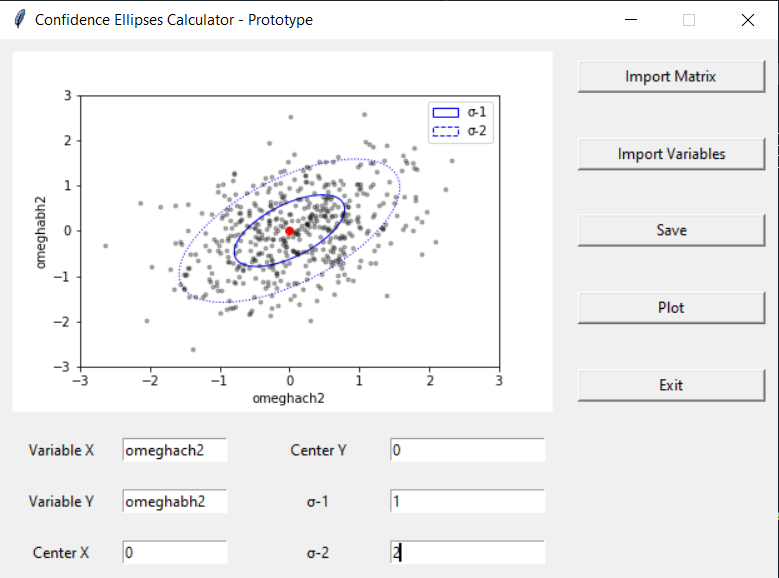


Figure 2 - GUI implementation using tkinter

This program starts by importing the necessary packages:

* tkinter
* matplotlib
* numpy

**B4. Implementation - Documentation**

*This Python programming file is contained within the same zipped folder as this document.*

# References

Docs, P. (2023, 04 11). *Python Docs*. Retrieved from tkinter — Python interface to Tcl/Tk: https://docs.python.org/3/library/tkinter.html

matplotlib. (2023, 04 11). *matplotlib*. Retrieved from Matplotlib: Visualization with Python: https://matplotlib.org/

NumPy. (2023, 04 11). *NumPy*. Retrieved from NumPy: https://numpy.org/