LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Graduate School of
Quantitative Biosciences Munich

An Introduction to Python

Dr. Markus Hohle

Gene Center of the LMU
Feodor-Lynen-Strasse 25
81377 Munich

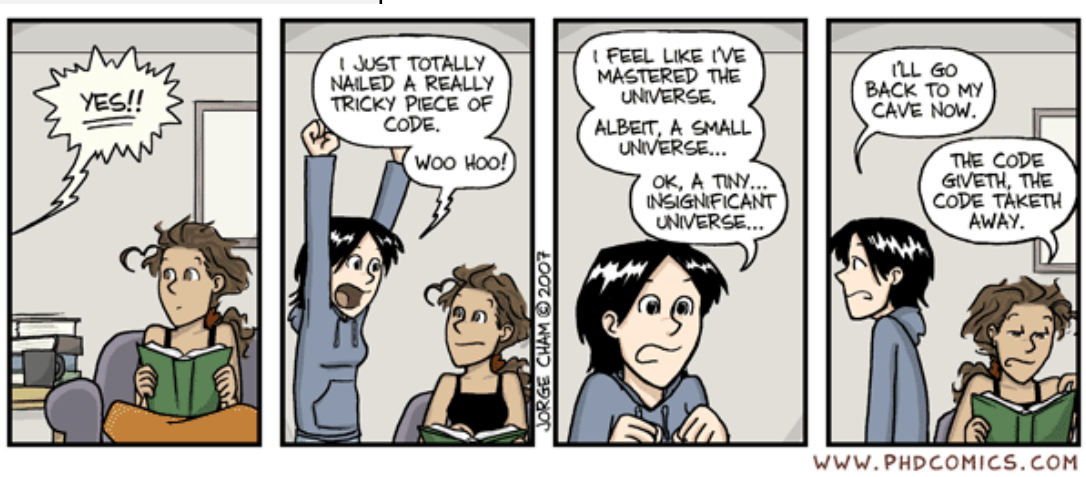
outline:

this part:

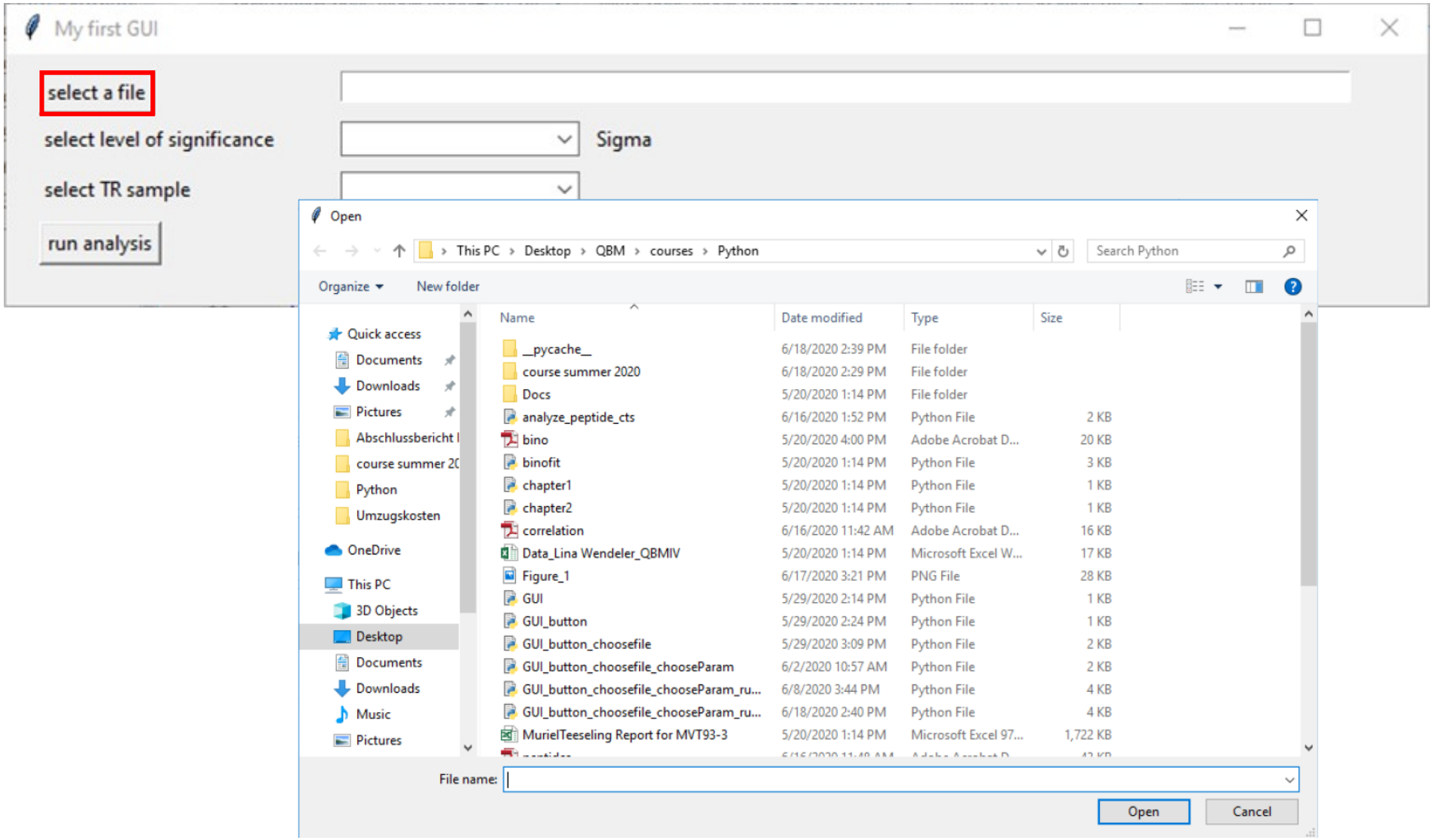
- *intro*
- *getting started: anaconda & spider*
- *data formats, reading .xlsx, .csv and .txt files*
- *writing and running scripts*
- *plots*
- *putting it all together*
- *about the pythonian logic*

2nd part:

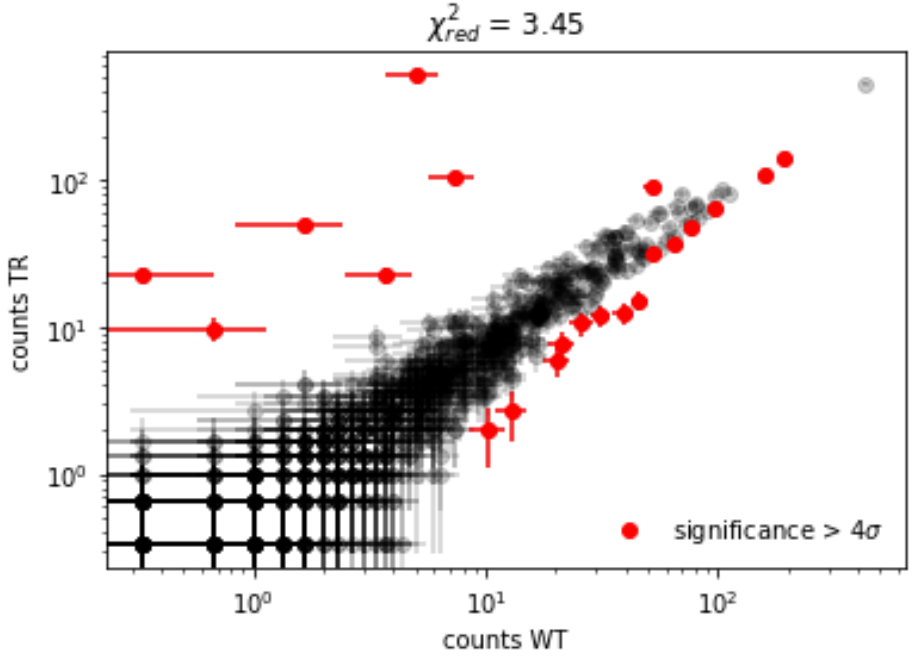
- building a GUI



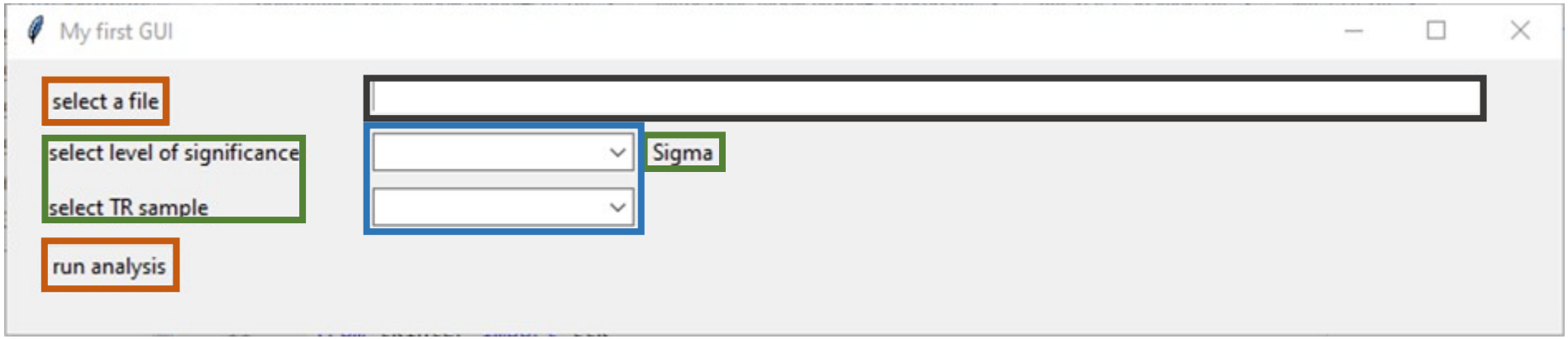
Graphical **U**ser **I**nterface



Graphical **U**ser **I**nterface



Graphical User Interface



- **buttons** have a function, like actual buttons
- **labels** don't have any function – just for the user
- **combobox** a pull down menu for selecting options
- **entry** works as a display

Graphical **U**ser **I**nterface

https://www.tutorialspoint.com/python/python_gui_programming.htm

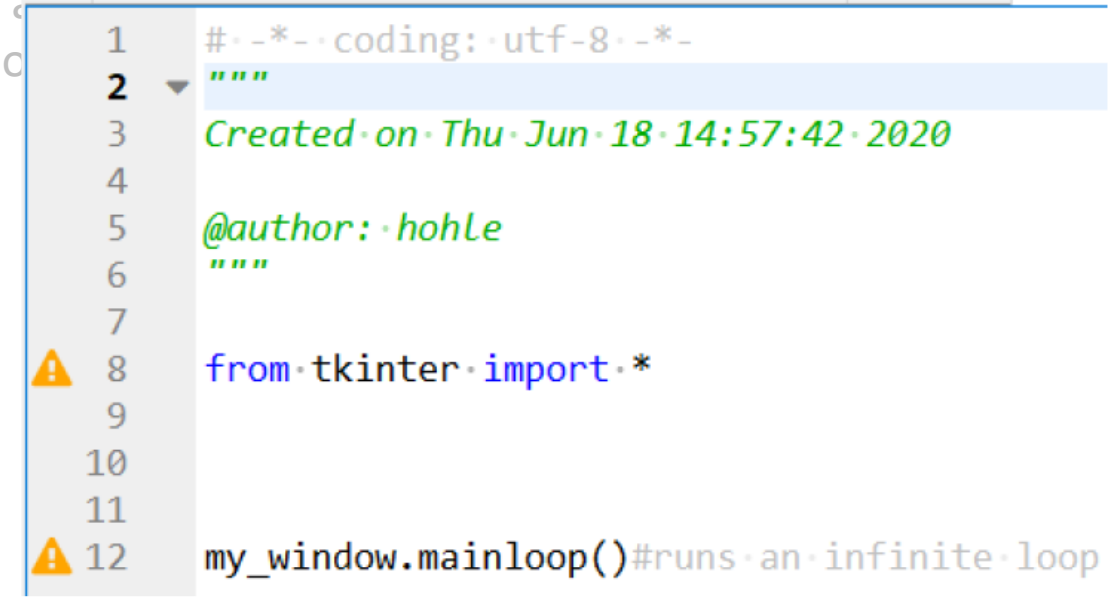
1) open a new script (as usual)

2) each GUI starts and ends with the following lines

```
from tkinter import * #imports all tools "*"
                      #from the tkinter library
                      #(required for building a GUI)

my_window.mainloop()  #runs a
                      #→ clo
```

2) save the GUI as e.g. My_GUI.py



Graphical User Interface

```
from tkinter import *


my_window = Tk() #creates a window
my_window.title('My first GUI')


my_window.mainloop() #runs an infinite loop
```

run the code

1) **create the geometry and buttons/windows**
2) assigning a function to the buttons/windows

```
from tkinter import *

my_window = Tk() #creates a window
my_window.title('My first GUI')
my_window.geometry('850x150')

file_choose_button = Button(my_window, text = 'select a file')
file_choose_button.place(x = 20, y = 10)

my_window.mainloop() #runs an infinite loop
```

note: the reference to the particular window is optional, but is required if using more than one window

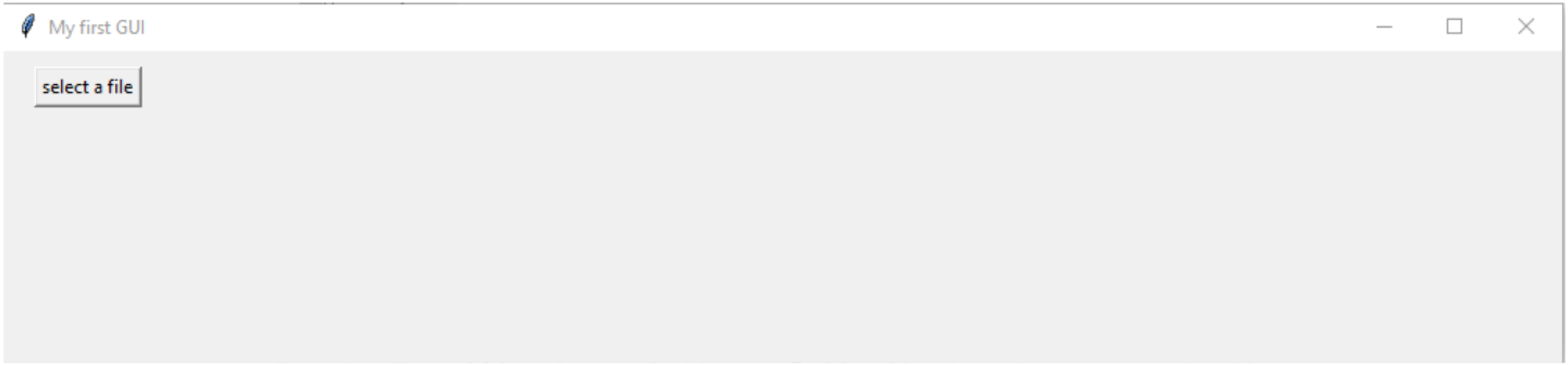
1) **create the geometry and buttons/windows**
2) assigning a function to the buttons/windows

```
from tkinter import *

my_window = Tk() #creates a window
my_window.title('My first GUI')
my_window.geometry('850x150') #size in pixel

file_choose_button = Button(my_window,text = 'select a file')
file_choose_button.place(x = 20, y = 10)

my_window.mainloop() #runs an infinite loop
```



1) **create the geometry and buttons/windows**
2) assigning a function to the buttons/windows

```
from tkinter import *

my_window = Tk() #creates a window
my_window.title('My first GUI')
my_window.geometry('850x150') #size in pixel

file_choose_button = Button(my_window,text = 'select a file')
file_choose_button.place(x = 20, y = 10)

filedir = Entry(my_window, text = " ", width = 100)#leave text empty
                                                   #as a place holder
filedir.place(x = 200, y = 10)


my_window.mainloop() #runs an infinite loop
```

1) **create the geometry and buttons/windows**
2) assigning a function to the buttons/windows

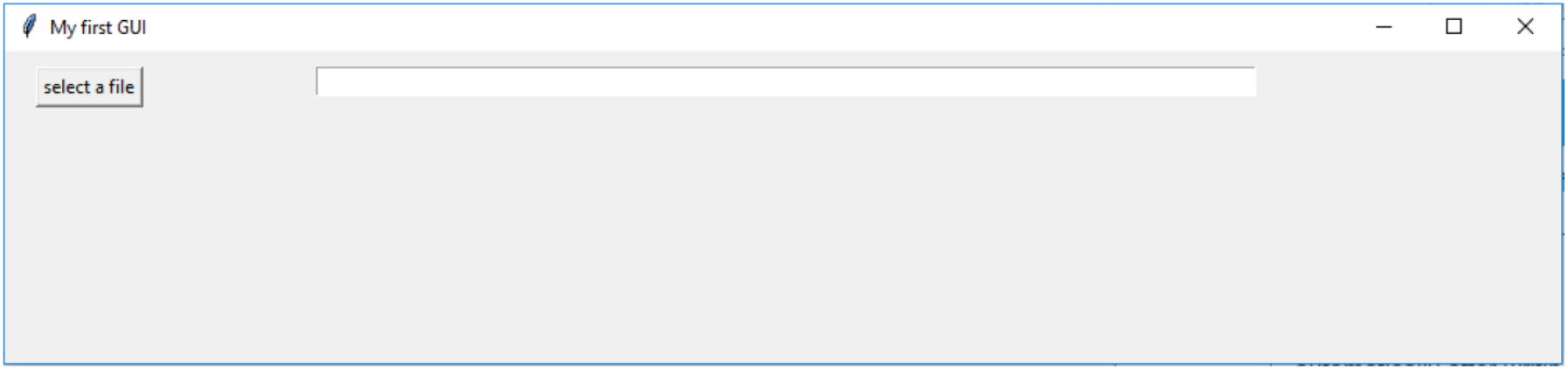
```
from tkinter import *

my_window = Tk() #creates a window
my_window.title('My first GUI')
my_window.geometry('850x150') #size in pixel

file_choose_button = Button(my_window,text = 'select a file')
file_choose_button.place(x = 20, y = 10)

filedir = Entry(my_window, text = " ", width = 100)#leave text empty
                                                   #as a place holder

filedir.place(x = 200, y = 10)
```



1) **create the geometry and buttons/windows**
2) assigning a function to the buttons/windows

now, let us include a **combobox** for selecting level of **significance** and the TR **sample**

```
L1 = Label(my_window, text = 'select level of significance')
L1.place(x = 20, y = 40)

L2 = Label(my_window, text = 'Sigma')
L2.place(x = 350, y = 40)

CB1 = ttk.Combobox(my_window, values = [1, 2, 3, 4, 5, 6])
CB1.place(x = 200, y = 40)
```

1) **create the geometry and buttons/windows**
2) assigning a function to the buttons/windows

now, let us include a **combobox** for selecting level of **significance** and the TR **sample**

We also need to call the **`ttk library`** in the **header**

```
from tkinter import ttk
```

```
CB1 = ttk.Combobox(my_window, values = [1, 2, 3, 4, 5, 6])
CB1.place(x = 200, y = 40)
```

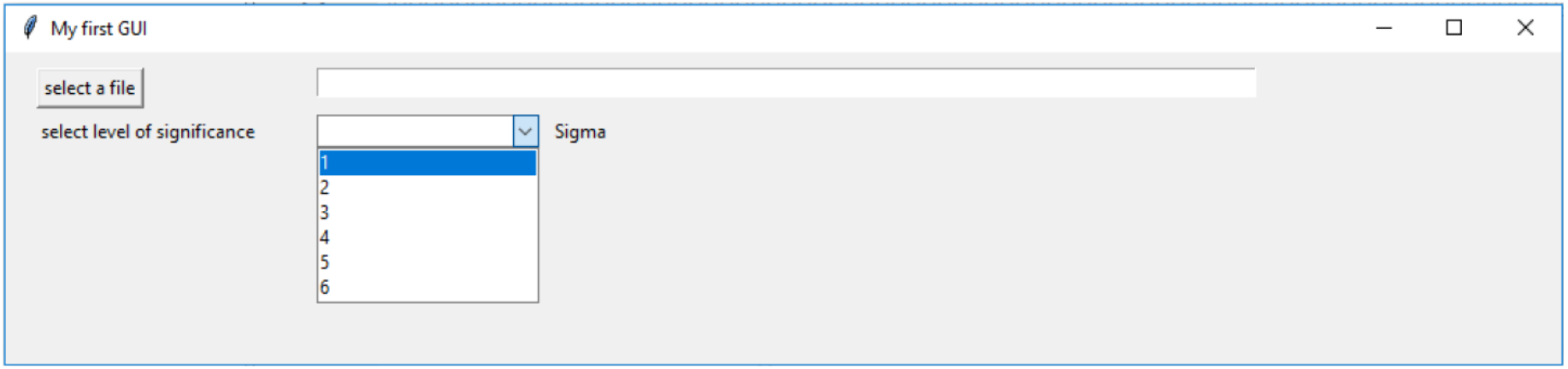
1) **create the geometry and buttons/windows**
2) assigning a function to the buttons/windows

now, let us include a **combobox** for selecting level of **significance** and the TR **sample**

```
L1 = Label(my_window, text = 'select level of significance')
L1.place(x = 20, y = 40)

L2 = Label(my_window, text = 'Sigma')
L2.place(x = 350, y = 40)

CB1 = ttk.Combobox(my_window, values = [1, 2, 3, 4, 5, 6])
CB1.place(x = 200, y = 40)
```



1) **create the geometry and buttons/windows**
2) assigning a function to the buttons/windows

now, let us include a **combobox** for selecting level of **significance** and the TR **sample**

```
L1 = Label(my_window, text = 'select level of significance')
L1.place(x = 20, y = 40)

L2 = Label(my_window, text = 'Sigma')
L2.place(x = 350, y = 40)

CB1 = ttk.Combobox(my_window, values = [1, 2, 3, 4, 5, 6])
CB1.place(x = 200, y = 40)

L3 = Label(my_window, text = 'select TR sample')
L3.place(x = 20, y = 70)

CB2 = ttk.Combobox(my_window, values = [1, 2, 3])
CB2.place(x = 200, y = 70)
```

1) **create the geometry and buttons/windows**
2) assigning a function to the buttons/windows

+ the **run button** for running the actual analysis

```
run_button = Button(my_window, text = 'run analysis')
run_button.place(x = 20, y = 100)
```

summary

```
from tkinter import *
from tkinter import ttk
```
calling the library

```
my_window = Tk()
my_window.title('My first GUI')
my_window.geometry('850x150')
```
opening the GUI window itself

```
file_choose_button = Button(my_window,text = 'select a file')
file_choose_button.place(x = 20, y = 10)
```
buttons

```
filedir = Entry(my_window, text=" ", width = 100)
filedir.place(x = 200, y = 10)
```
entry

```
L1 = Label(my_window, text = 'select level of significance')
L1.place(x = 20, y = 40)

L2 = Label(my_window, text = 'Sigma')
L2.place(x = 350, y = 40)
```
labels

```
CB1 = ttk.Combobox(my_window, values = [1, 2, 3, 4, 5, 6])
CB1.place(x = 200, y = 40)
```
combobox

```
L3 = Label(my_window,text = 'select TR sample')
L3.place(x = 20, y = 70)
```
labels

```
CB2 = ttk.Combobox(my_window, values = [1, 2, 3])
CB2.place(x = 200, y = 70)
```
combobox

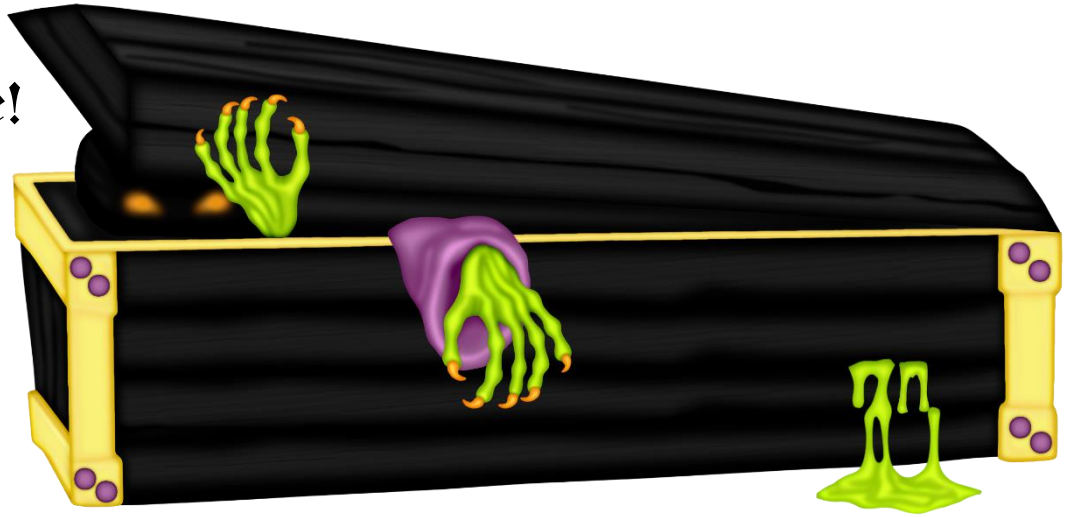
```
my_window.mainloop()
```
infinite loop


Time to take a break

1) create the geometry and buttons/windows
2) **assigning a function to the buttons/windows**

so far, we just created the make up, but the GUI doesn't do anything yet

Let's awake the GUI to life!



1) create the geometry and buttons/windows
2) **assigning a function to the buttons/windows**


```
file_choose_button = Button(my_window,text = 'select a file')
file_choose_button.place(x = 20, y = 10)
```

<u>we want to:</u>
- choose
- and display a file here
- and also save the path for filename for the analysis part

writing a function to ***display*** the filename

```
def open_file():
    filename = askopenfilename()
    filedir.delete(0, "end")
    filedir.insert(0, filename)
```

opens the file menu

keeps entry empty, until file selected

displays the file name

1) create the geometry and buttons/windows
2) **assigning a function to the buttons/windows**

Now, we also need to ***call*** the function open_file

```
file_choose_button = Button(my_window,text = 'select a file',\
                                        command = lambda: open_file())

file_choose_button.place(x = 20, y = 10)
```

also include in the header:

```
from tkinter.filedialog import askopenfilename
```

```
def open_file():
    filename = askopenfilename()
    filedir.delete(0, "end")
    filedir.insert(0, filename)
```

opens the file menu

keeps entry empty, until file selected

displays the file name

1) create the geometry and buttons/windows
2) **assigning a function to the buttons/windows**

Now, we also need to ***call*** the function open_file

```
file_choose_button = Button(my_window,text = 'select a file',\
                                        command = lambda: open_file())

file_choose_button.place(x = 20, y = 10)
```

also include in the header:

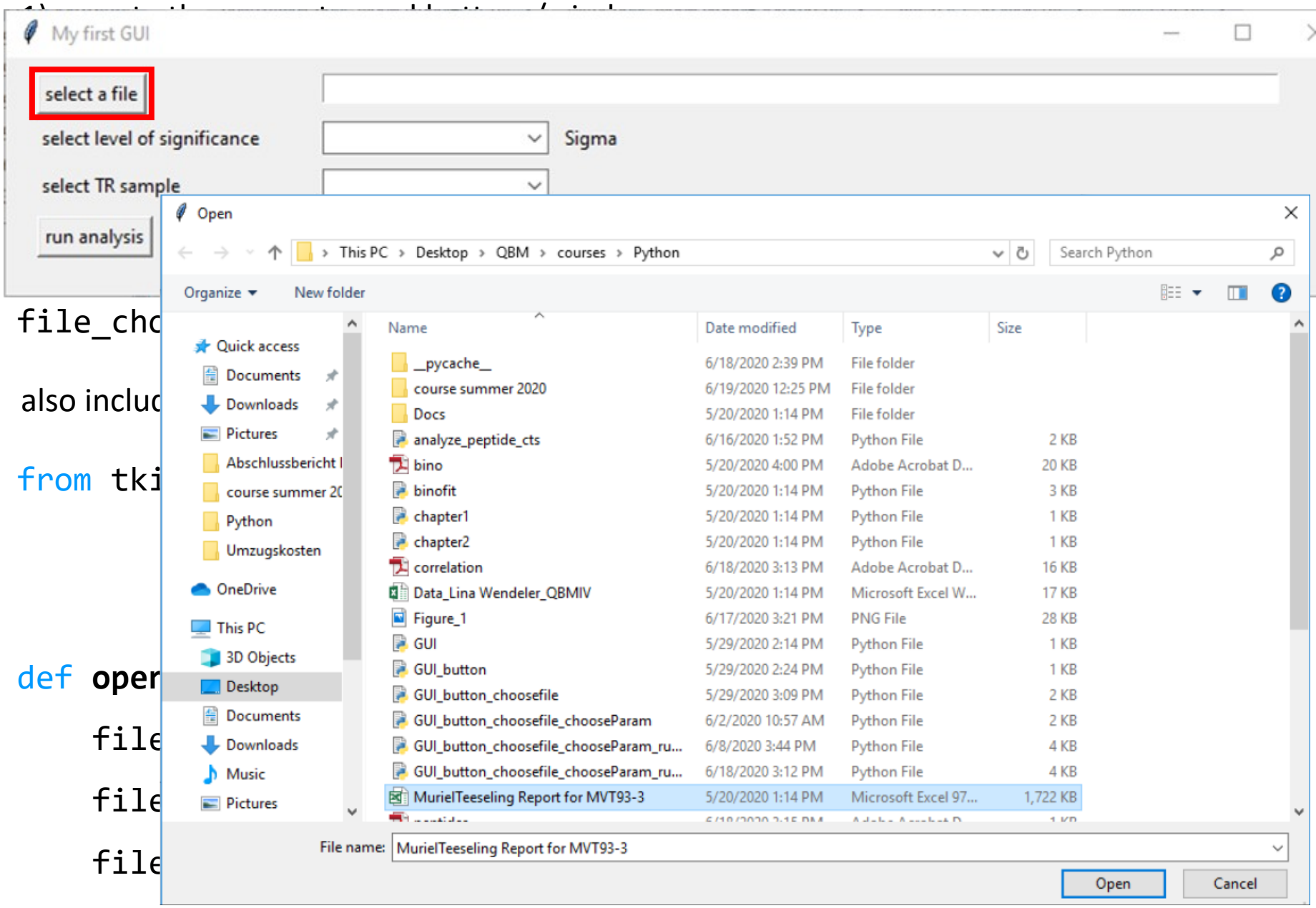
```
from tkinter.filedialog import askopenfilename
```

```
def open_file():
    filename = askopenfilename()
    filedir.delete(0, "end")
    filedir.insert(0, filename)
```

opens the file menu

keeps entry empty, until file selected

displays the file name




```
file_cho

also includ

from tki

def oper
    file
    file
    file
```

1) create the geometry and buttons/windows
2) **assigning a function to the buttons/windows**

Now, we also need to ***call*** the function open_file

```
file_choose_button = Button(my_window,text = 'select a file',\
                                        command = lambda: open_file())

file_choose_button.place(x = 20, y = 10)
```

we want to:
- choose
- and display a file here
- and also **save the path** for filename for the analysis part

```
def get_filename():
    filename = filedir.get()
    return(filename)
```

The command *get* returns the value from *Entry*

1) create the geometry and buttons/windows
2) **assigning a function to the buttons/windows**

same for the other buttons:

```
def get_sigma():
    sigma  = CB1.get()
    return(sigma)
```

The command *get* returns the value from the combobox

```
def get_sample():
    sample = CB2.get()
    return(sample)
```

1) create the geometry and buttons/windows
2) **assigning a function to the buttons/windows**

There should be the following functions now:

```
def open_file():
    filename = askopenfilename()
    filedir.delete(0, "end")
    filedir.insert(0, filename)
```

opens the file menu, gets called in `file_choose_button`

```
def get_filename():
    filename = filedir.get()
    return(filename)
```

reads filename, not called yet

```
def get_sigma():
    sigma  = CB1.get()
    return(sigma)
```

reads sigma, not called yet

```
def get_sample():
    sample = CB2.get()
    return(sample)
```

reads sample, not called yet

1) create the geometry and buttons/windows
2) **assigning a function to the buttons/windows**

finally: we want to use the **run button** to run the analysis
→ it also calls the other subroutines now!

```
def run():

    filename = get_filename()
    sigma    = get_sigma()
    sample   = get_sample()

    sigma  = int(sigma)
    sample = int(sample)

    my_data_analysis(filename,sample,sigma)

run_button = Button(my_window, text = 'run analysis',\
                                command = lambda: run())

run_button.place(x = 20, y = 100)
```

1) create the geometry and buttons/windows
2) **assigning a function to the buttons/windows**

we are **almost done**

include in the header: `from MyModule import my_data_analysis`

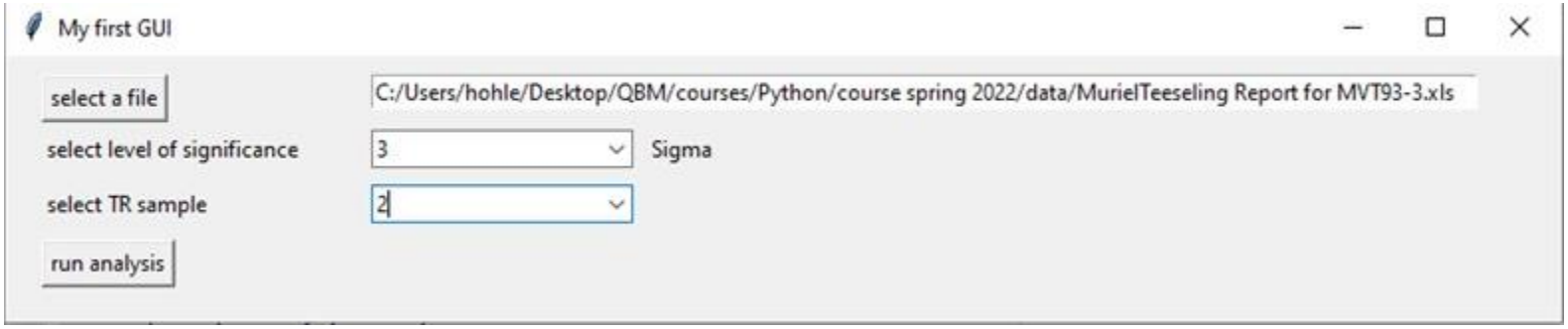
sometimes the plot does not show up:

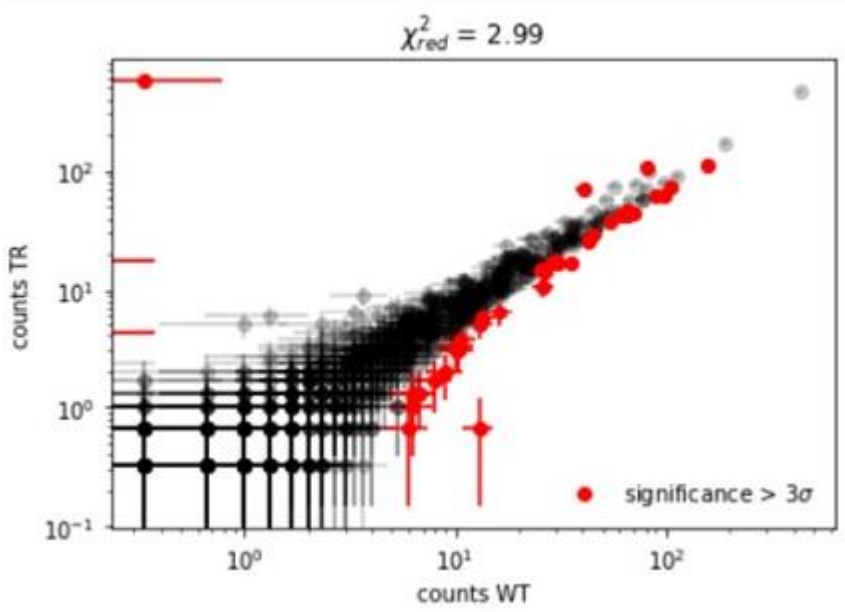
→ go to `my_data_analysis.py`
→ add `plt.show(block=False)`

above the line `return(LamWT,LamTR,Chi2_red)`

1) create the geometry and buttons/windows
2) **assigning a function to the buttons/windows**

save and run the GUI
- it's gonna take a while to finish





Your first Python GUI

Congratulations