Entre los dias 19 y 20 de Mayo tuvimos una nueva reunión para seguir trabajando en lo poco que quedaba para acabar el TFG.

En la primera parte de la reunión y tras explicar el correo comente lo que había hecho antes de la misma, que eran 2 cosas principalmente, comenzar a hacer la función del reset bien, cosa que daba un problema de recurrencia al llamar de nuevo al editor. Y lo segundo seguir ajustando el pop up del área pequeña para que quedara en un tamaño idóneo.

La primera parte la solucionamos mirando los distintos métodos que tiene el kivy, entre los que esta parar su ejecución, por lo que lo que había que hacer (y hemos hecho) entonces es parar la ejecución y después volver a arrancarla teniendo en cuenta que hay que inicializar todo, es decir, volver a borrar recortes para que no haya fallo de que ya estan los recortes creados, poner las variables a su estado inicial, cosa que ya había realizado antes de la reunion, etc …

Con eso, añadido a restaurar los textos de coeficiente y estimación (Los eliminas para que cuando se vuelva a ejecutar el kivy vuelvan a aparecer, pero no tengan por debajo el texto anterior) y restaurando tambien los botones a sus valores iniciales (solo se tiene que mostrar el de cargar, y el de salir) como siempre ocultándolos hasta que no llegue el momento de que se muestren la función en si ya estaría hecha. Sin embargo hay un problema al cargar la imagen, que te aparece el filechooser que estas utilizando y el anterior que no se mueve.

Lo que hay que hacer es eliminar ese primer filechooser para evitar este problema pero tras varias pruebas esto sigue fallando (Hemos probado eso con del, con dejar registrado un content por defecto inicial para siempre que se ejecute la aplicación, con hacer una prueba aparte en otro programa, … ) y **hay que continuar buscando soluciones. Para ello subiré el programa a github (incluido el de prueba aparte “prueba Filechooser”) y Domingo lo revisara.**

Despues el tema del pop up lo hemos desarrollado, ya que lo que buscamos es que se cierren cuando ya tienes el dato correcto para que siga trabajando el programa. Habia un inconveniente y es que en la prueba lo que cerrabas era un pop up y la otra pantalla era el programa en si. A la hora de implementarlo en el programa real los 2 pop ups son distintos y van dentro dicho programa, por lo que la cosa cambia teniendo que crear 2 pop ups, teniendo que navegar entre ellos y finalmente cerrarlos ambos a la vez si la aplicación funciona correctamente.

Los popups se cierran con .dismiss(), y tras probarlo en el programa auxiliar (“Prueba Pop up”), lo hemos introducido en nuestro programa donde **quedaría mejor importarlo desde otro programa en si, para que quede mas limpio el código.** Entonces en el programa principal lo que hacemos es en el punto donde se pedia el ap, crear un popup (llamado showap) de la forma en la que se creo en el programa auxiliar. Este popup tiene una variable en el kivy llamada ap y que igualaremos al ap que escribiremos en dicho popup. El otro popup se llama Popupwindow como en el programa auxiliar.

Para cerrar las dos ventanas de popup a la vez lo que hemos decidido es crear una función (cerrarpopup) en la que tiene que ejecutar las dos ordenes de dismiss, y lo que hace el botón si se le pulsa es entrar en esta función, ya que si le ponemos que haga dismiss de uno de los popups, el otro no se cerraría como es lógico. Ademas vamos a aprovechar esta función para añadir las instrucciones que iban en el load después de tener el ap (Crear los recortes, …) . Esto tiene mucho sentido porque asi evitamos otro fallo que tenemos ahora en el programa que es que a pesar de estar esperando el pop up y que entre un valor, el resto de la funcion se esta ejecutando con un ap por defecto que hay por el programa para que no de error. Hemos probado con bucles while, hemos probado inicializando el área pequeña a -100 y ninguna de las soluciones es la que buscamos por lo que esperamos que con este cambio si que podamos solucionar el fallo.

Cuando esto funcione habra que hacer como hicimos en el programa auxiliar las pruebas de si le metes al programa un carácter / símbolo o algo que no sea un entero. De igual forma hay que comprobar que el ap no sea mayor que el ag. En caso de que no se cumplan estas condiciones no tiene que cerrar los 2 popups, sino solo el segundo que te avisa de que hay un error en el dato, y el pop up principal te tiene que permitir probar de nuevo a indicar un valor para el ap. **De nuevo todo esto lo echara un vistazo Domingo cuando le suba el programa principal.**

Cambiando de tema miramos un poco por encima las simulaciones y analice un poco las conclusiones que se sacan y que se pueden añadir a la memoria. Por ejemplo una de las deducciones que se sacan es que al estar comparándose dos matrices que provienen del mismo sistema de datos, es posible que se parezcan mucho y por tanto que al llegar al 50% de imágenes recorridas ya este en el rango la diferencia de varianza y se acabe esa simulación en concreto. Esto es obviamente menos frecuente cuanto mas igualdad (de varianzas) necesitas para salirte (No es lo mismo que necesites que se salgan al parecerse en un 70%, es decir, 30% de diferencia, que si se tienen que parecer en un 99% (1% de diferencia).

Otra conclusión es que contra mas datos hay en la imagen, menos se parecen las graficas probablemente porque hay mas muestras.

Todo esto lo ire anotando en la memoria en la que me centrare estas semanas, para acabar el capitulo del diseño, el de las simulaciones y el de las pruebas si es posible.

Domingo como he comentado se centrara en solucionar los fallos explicados de los programas para lo cual le pasar tanto el programa principal (Prueba 4), como los auxiliares.

La proxima reunión será en 2 semanas y la idea es concretarla la semana que viene. Dejo las tareas en modo esquemas para que quede claro en lo que hay que trabajar.

Tareas:

* Reinicio (Problema con el filechooser que aparece sobrepuesto) (Domingo)
* Pop up Area pequeña que no se ejecuta correctamente (Domingo)
  + Moverlo a otro fichero e importarlo
* Memoria
  + Simulaciones
  + Diseño
  + Pruebas