

Memoria de prácticas **Introducción al análisis de datos con ROOT**

Tutores: S.E.A. Orrigo y A. Tolosa Delgado

Contexto

La memoria de esta práctica consistirá en hacer un breve análisis de datos. La estructura de un TTree es una representación de cómo funcionó el montaje experimental. Por ejemplo, datos de un mismo detector (tiempo, energía, etc) se representan como hojas y se agrupan en una misma rama del TTree.

Este TTree se correspondería a un experimento que estudia cómo se desintegran algunos isótopos (“experimentos de decay”). Para generar éstos isótopos se puede recurrir a una instalación “in-flight”. En este tipo de instalación, núcleos estables pesados (^{64}Zn , ^{238}U , etc) son acelerados y fisionados “en-vuelo”. Los fragmentos de fisión (isótopos inestables) son identificados por un separador de masas. El separador de masas mide la masa y el número atómico de estos isótopos inestables.

Tras recorrer el separador de masas, los isótopos se paran *-implantan-* en el interior de un detector (de silicio, un cristal centelleador, etc). Éste detector asigna un tiempo al momento de la implantación. Estos isótopos inestables se desintegrarán beta para volver al valle de la estabilidad. El mismo detector donde se pararon se usa también para detectar una desintegración beta.

Aunque la desintegración nuclear es un proceso aleatorio (no podemos saber cuándo se desintegrará un núcleo), es un proceso estadístico que sigue la distribución de Poisson. Es decir, si acumulamos en un histograma el tiempo que tardó cada núcleo en desintegrarse* reproduciremos una curva de desintegración, con un perfil exponencial de “pendiente” la constante de desintegración “ λ ” de ese núcleo.

$$A_{\beta}(t) = A_{\beta}(0) e^{-\lambda t}$$

*El tiempo que tarda un núcleo en desintegrarse nos lo da la diferencia temporal entre la detección de la beta y la implantación de ese núcleo.

Material:

Fichero de ROOT. En este mismo fichero se guardarán diferentes objetos de ROOT.

Instrucciones:

1. Abrir ROOT. En bash, “root -l”
2. Usando la clase “TFile”, abrir el fichero en modo “update” (este modo permite añadir objetos de ROOT a este fichero)

```
TFile * ifile = TFile::Open( “aFile.root” , “update” );
```
3. Comprobar si hay dentro un TTree llamado decayTree. Asignar este objeto a una variable tipo TTree* usando la función “GetObject”,

```
Ttree * itree = 0;  
ifile->GetObject( “decayTree” , itree );
```

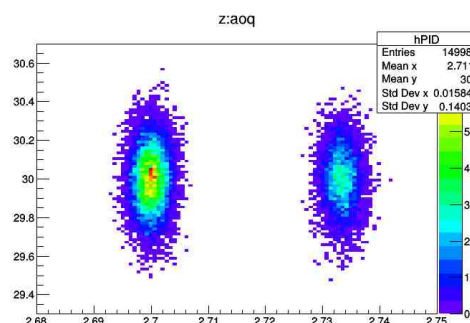
4. Comprobar si la estructura de “decayTree” es como la siguiente:

```
*****
*Tree      :decayTree :
*Entries   :    14998 : Total =          7955654 bytes File Size =    808087 *
*          :          : Tree compression factor =    9.87
*****
*Branch    :implant.
*Entries   :    14998 : BranchElement (see below)
*.....
*Br       0 :implant.t : Double_t
*Entries   :    14998 : Total Size=    120802 bytes File Size =    105210 *
*Baskets   :         4 : Basket Size=    32000 bytes Compression=    1.14 *
*.....
*Br       1 :implant.z : Double_t
*Entries   :    14998 : Total Size=    120802 bytes File Size =    106058 *
*Baskets   :         4 : Basket Size=    32000 bytes Compression=    1.13 *
*.....
*Br       2 :implant.aoq : Double_t
*Entries   :    14998 : Total Size=    120818 bytes File Size =    102517 *
*Baskets   :         4 : Basket Size=    32000 bytes Compression=    1.17 *
*.....
*Br       3 :implant.tbeta : vector<double>
*Entries   :    14998 : Total Size=    7592135 bytes File Size =    491297 *
*Baskets   :        241 : Basket Size=    32000 bytes Compression=    15.44 *
*.....
```

Este TTree es equivalente a la siguiente tabla:

TTree: decayTree			
TBranch: implant			
double t	double z	double aoq	std::vector<double> tbeta
15.5	30	2.66	{ 5.6, 8.2, 12.2, 19.7 }
...

- Usando el comando **Draw**, plotear en un histograma bidimensional las variables “z:aoq”. Redireccionar (>>) el histograma a un histograma de nombre “hPID”, y dar el bineado y rango apropiado en caso necesario. Una vez dibujado, cambiar las propiedades del histograma (usando el editor gráfico o la terminal) para que se parezca a éste:



- En el menú superior de la ventana del TCanvas, desplegar la pestaña “View”, y buscar el apartado “Toolbar” para añadir la barra de herramientas. El primer botón por la derecha será un símbolo de unas tijeras. Click sobre el botón para hacer dos cortes gráficos, cada uno que

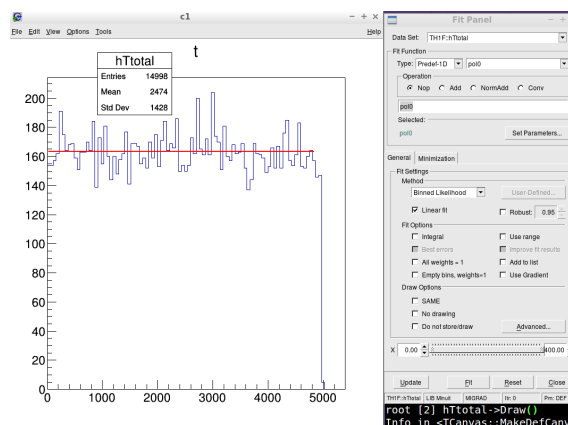
rodee a cada agrupación de datos (a cada isótopo). **Importante:** después de hacer un corte gráfico hay que cambiarle el nombre (por ejemplo “CUTGiso1”, “CUTGiso2”). Para ello, pasar el ratón sobre la línea poligonal, y en el momento en que la forma del cursor cambie, pulsar el botón derecho del ratón y buscar en el menú desplegable la opción “SetName” para cambiar el nombre de cada corte gráfico.

7. Guardar el histograma y los cortes gráficos en el fichero, usando la función “Write()”. Recordamos:

```
objeto.Write();
puntero->Write();
```

(todos los objetos de ROOT tienen un nombre, y dentro de ROOT funciona como un puntero).

8. Dibujar la hoja “t” de decayTree usando la función Draw, y redireccionar (>>) el histograma a un histograma de nombre “hTtotal”. Repetir el proceso pero usando como condiciones los cortes gráficos generados en el apartado 6. Redireccionar los histogramas a histogramas con nombres “hTiso1” y “hTiso2”.
9. Guardar en el fichero los histogramas del apartado 8, usando el método “Write()”.
10. Dibujar el histograma “hTtotal”, y pasar el cursor por encima de la línea del histograma. Cuando el cursor cambie de forma, click con el botón derecho del ratón y buscar en la lista desplegable “Fit panel”. Hacer un ajuste usando la función `pol0` y como método de ajuste “Binned likelihood” (por defecto es “Chi square”). A partir del ajuste, obtener el rate “total” y de cada isótopo (“iso1”, “iso2”).



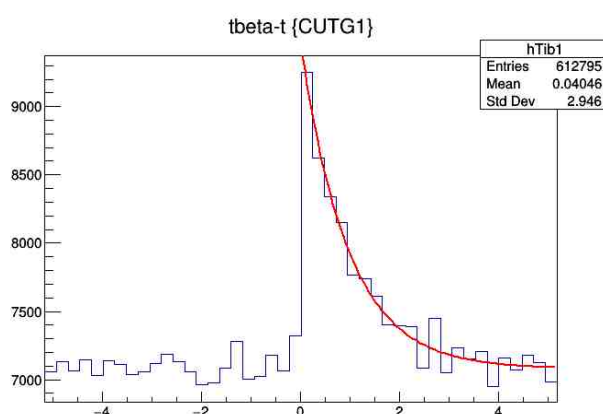
11. Guardar en el fichero los valores de los ajustes, usando la clase “TParameter”, con la siguiente sintaxis:

```
root[] TParameter<Double_t> par1("rateIso1", 1.1)
root[] par1.Write()
```

12. La rama `tbeta` es un vector de los tiempos de desintegraciones que están entre -10 y 10 s alrededor de cada implante. Usando la función Draw, dibujar la diferencia temporal entre esas desintegraciones beta y el momento de la implantación para obtener una curva de desintegración. Para ello, se puede dibujar “`tbeta - t`”. Para que las curvas tengan un sentido físico, hay que seleccionar un único isótopo. Para ello, usar como condición los cortes gráficos del apartado 6, y redireccionar los correspondientes plots a histogramas de nombres “hTib1”, “hTib2”. El rango y binned de los histogramas ha de ser adecuado para poder ver claramente una curva de desintegración.
13. Guardar los histogramas del apartado 12 usando la función “Write()”

14. Hacer un ajuste a las curvas de desintegración: dibujar el histograma `hTib` y abrir el “Fit panel”.

- Probamos con la función de ajuste `expo + [2]` y como método “Binned likelihood”. Si el fit es *malo*, intentar un ajuste en dos etapas: primero un ajuste exponencial `expo`, y seguidamente probar con la función `expo + [2]`.
- En el seminario de [JL Tain sobre “Estadística y análisis de datos”](#) se explicó cómo se hacen en realidad este tipo de ajustes. Primero hay que caracterizar el fondo (que en nuestro caso son correlaciones accidentales, y que contribuyen por igual a todo el espectro). Para ello podemos hacer un ajuste con `pol0` a la parte negativa del espectro, y así obtener un valor del fondo (por ejemplo, 3333 cuentas por canal). Seguidamente, podemos hacer un ajuste con la función `expo + 3333`.



15. La pendiente del ajuste se correspondería con la *constante de desintegración* λ . Definir una nueva variable tipo `Double_t`, y asignarle el valor correspondiente al periodo de semidesintegración $T_{1/2}$ calculado a partir de los ajustes del apartado 14.b). Para cada ajuste, guardar este valor usando la clase “Tparameter” (ver apartado 11).

16. Para finalizar, cerramos el fichero con el comando “Close()”,

```
ifile->Close();
```

Si abrimos el fichero (`root -l out1.root`) y mostramos lo que hay dentro con el comando `.ls`, debería parecerse a ésto:

```
root [1] .ls
TFile**          out1.root
TFile*           out1.root
KEY: TTree       decayTree;1
KEY: TH2F hPID;1 z:aoq
KEY: TCutG       CUTG1;1   Graph
KEY: TCutG       CUTG2;1   Graph
KEY: TH1F hTtotal;1 t
KEY: TH1F hTiso1;1 t {CUTG1}
KEY: TH1F hTiso2;1 t {CUTG2}
KEY: TParameter<double> rateTotal;1
KEY: TParameter<double> rateIso1;1
KEY: TParameter<double> rateIso2;1
KEY: TH1F hTib1;1  tbeta-t {CUTG1}
KEY: TH1F hTib2;1  tbeta-t {CUTG2}
KEY: TParameter<double> T12Iso1;1
KEY: TParameter<double> T12Iso2;1
```

Memoria de prácticas

Para la evaluación de la realización-comprensión de esta práctica, el alumno deberá presentar una memoria (con los puntos que se detallan más abajo) y el fichero ROOT donde se han guardado los espectros al email oficial de la asignatura: fnex2018@ific.uv.es

Para cualquier consulta, los correos de los tutores de la práctica son:

- Sonja Orrigo: Sonja.Orrigo@ific.uv.es
- Alvaro Tolosa: Alvaro.Tolosa@ific.uv.es

1. Plot de identificación Z vs A/Q .

1.a) Seguir las instrucciones anteriores para generar un plot de identificación Z vs A/Q . Insertar el plot y las instrucciones de terminal que se usaron para generarlo (si se hicieron cambios con el editor gráfico, enumerarlos).

1.b) ¿A qué isótopos se corresponde la identificación?

1.c)* En un separador de masas de una instalación *in-flight*, ¿qué tres magnitudes se necesitan principalmente para calcular Z & A/Q ? (Pista: [seminario de JL Tain, “ \$\beta\$ -delayed neutron emission: Measurement of emission probabilities”](#))

2. Rate de implantación

2.a) Seguir las instrucciones anteriores para generar un plot de la variable t . Insertar los plots y las instrucciones de terminal que se usaron para generarlo (si se hicieron cambios con el editor gráfico, enumerarlos).

2.b) Explicar el procedimiento para obtener los rates de los isótopos individuales y dar los resultados numéricos (valor y error).

2.c)* ¿Haciendo un ajuste donde haya un fondo, como por ejemplo en el apartado 14, es mejor dejar libre el fondo o fijarlo? ¿Y porqué?

3. Desintegración de los implantes

3.a) Seguir las instrucciones anteriores para generar los plots correspondientes a la curva de desintegración. Insertar los plots y las instrucciones de terminal que se usaron para generarlo (si se hicieron cambios con el editor gráfico, enumerarlos).

3.b) Explicar el procedimiento para obtener la constante “ λ ” a partir de los plots anteriores

3.c)* Cuando se detecta una partícula beta no se sabe de qué isótopo proviene. ¿A qué parte de la curva contribuyen las betas que no provienen de la desintegración del isótopo que se selecciona?

***Estas cuestiones son optativas.**