

Javier de León Gómez

201603068

Documentación Tarea 2.II.1

Se nos pide analizar un set de datos para crear un histograma de las tasas de conteos por minuto y realizar un fit de Poisson para verificar si el histograma presenta ese comportamiento esperado. Para llenar los datos de nuestro histograma se realiza una escala de segundos a minutos.

Luego de tener los datos se realiza el histograma con los mismos y se crea una implementación de la función de Poisson para poder realizar un fit con la misma y la base de datos. Para terminar se grafica el histograma conjunto con el fit de Poisson y se imprimen en consola los parámetros del mismo.

Para realizar lo que se nos pide primero creamos un metodo que puede analizar el archivo de la base de datos y depurarlo para poder llenar nuestra lista de valores deseados para el histograma.

Además se realizaron distintas configuraciones de histogramas respecto al número de bins y se obtiene el cual produce el resultado de χ^2 sobre grados de libertad más cercano a uno sin aumentar considerablemente la cantidad de bins del fenómeno.

Código del script:

```
/*
Tarea 2
Realizar un histograma y un fit de posion
con la tasa de conteos por minuto de un archivo
*/

// Vamos a importar todas las librerias del ejemplo por si acaso

#include <iostream>
#include <fstream>
#include <string>
#include "TGraph.h"
#include "TCanvas.h"
#include <cassert>
#include <cmath>

#include "TH1.h"
#include "TH1D.h"
#include "TCanvas.h"
#include "TRandom.h"

#include <cstdlib>
#include <iostream>
#include <fstream>
#include <stdio.h>
```

```
#include <stdlib.h>

std::list<double> lista_valores_histograma;

//aqui hacemos el histo y el fit

void Histograma()
{
    //esta cantidad de bins me deja mejor el chi 2

    TH1D *h3 = new TH1D("Histograma", "Tasa de Conteo Promedio por Minuto;Tasa por Minuto;Frecuencia",40, lista_valores_histograma.front(), lista_valores_histograma.back());

    for (auto val : lista_valores_histograma)
    {
        h3->Fill(val);
    }

    // dibujamos el histograma
    TCanvas* c = new TCanvas;
    c->cd(1);
    h3->Draw();

    //hacemos el fit
    //esta funcion nos da una dist de poison default

    TF1 *f = new TF1("f", "[0] * TMath::Poisson((x/[2]), ([1]/[2]))",
    h3->GetXaxis()->GetXmin(), h3->GetXaxis()->GetXmax());
    // maximum( Poisson( expected value = variance = 5. ) ) = 0.18
    f->SetParameters(h3->GetBinContent(h3->GetMaximumBin()) / 0.18, // "Constant"
                    h3->GetMean(), // "Mean"
                    h3->GetMean() / 5.); // "XScaling"
    f->SetParNames("Constant", "Mean", "XScaling");
    f->SetNpx(1000);
    TFitResultPtr r = h3->Fit(f, "S");
    f->GetNDF();
    f->GetChisquare();
    f->GetProb();
    h3->Draw();
    c->Update();

    //pedimos el chi2 y los grados de lib

    r->Print("V");
}

void Programa()
{
```

```
lista_valores_histograma.clear();
//obtenemos los datos que necesitamos para el histograma de los logs
std::ifstream indata("wcd_lago_006-run.log");

//depuramos las lineas que no nos interezan
std::string indice;
while (std::getline(indata, indice))
{

    //le pedimos que solo guarde los datos de indice M
    if (indice[0] == 'M')
    {
        //con este tipo de variable podemos sacar facil la info de cada fila

        std::istringstream line_stream(indice);
        //estamos poniendole los nombres de las etiquetas de las columnas
        //para dejar en orden nuestros datos procesados
        std::string id, fecha, hora;
        int acumulado;
        double promedios, maxs, mins;

        line_stream >> id >> fecha >> hora >> acumulado >> promedios >> maxs >>
mins;

        //metemos a la lista el valor que nos interesa y lo escalamos
        lista_valores_histograma.push_back(promedios*60);

    }

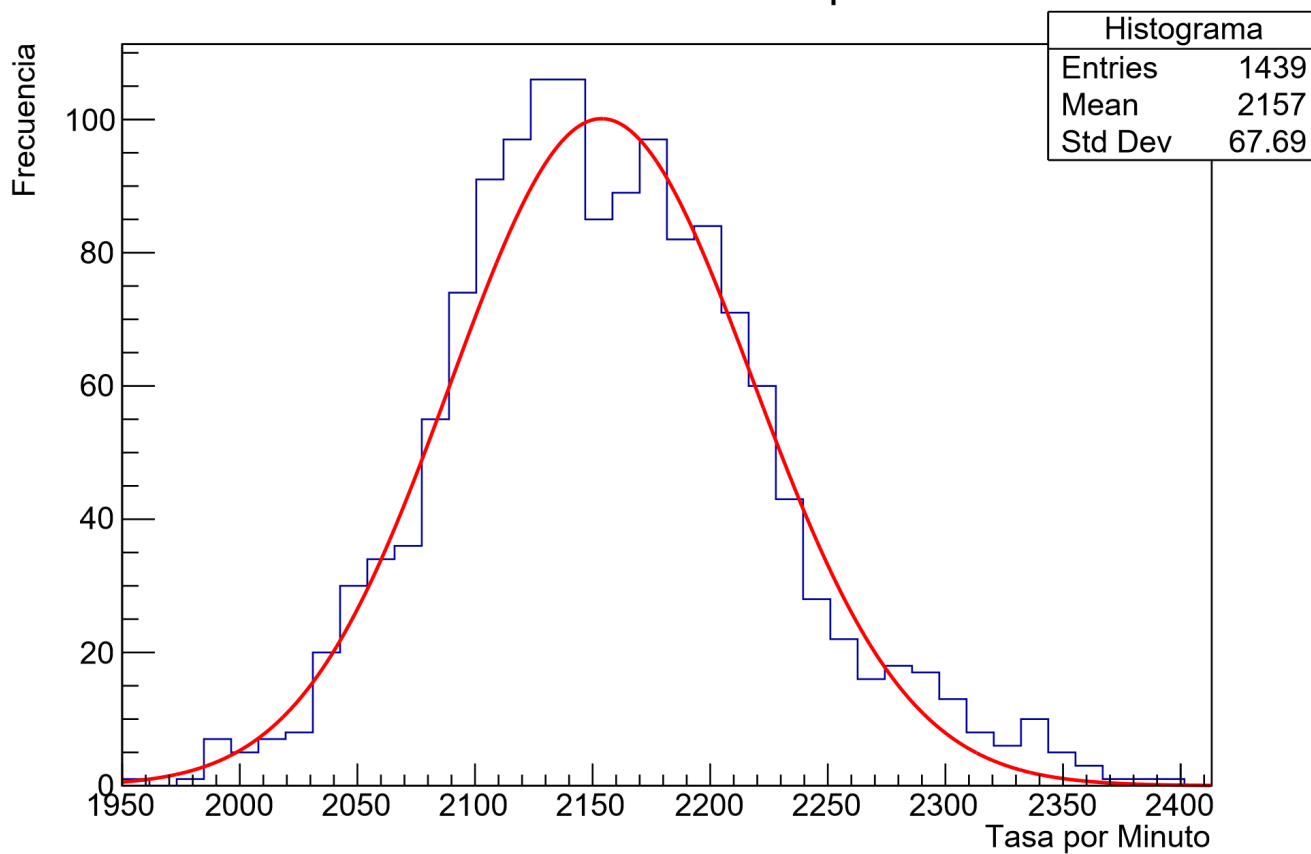
    indata.close();

    lista_valores_histograma.sort();
    Histograma();
}
```

Resultados Histograma 40 Bins:

Se obtiene el siguiente histograma y fit de la distribución:

Tasa de Conteo Promedio por Minuto



Se obtienen los siguientes parámetros para el fit:

Chi2 = 49.3031

NDf = 35

Edm = 4.16531e-08

NCalls = 922

Constant = 8424 +/- 439.634

Mean = 2154.75 +/- 1.80022

XScaling = 1.91203 +/- 0.0861593