Desarrollo de herramientas para análisis de física con datos abiertos del experimento ATLAS en un entorno multi-cloud

Universidad Nacional de Colombia
Facultad de Ciencias, Departamento de Física
Beyond Research
Profesor: Carlos Eduardo Sandoval Usme
Miguel Ángel García Ruíz
miagarciaru@unal.edu.co

11 de Febrero de 202

Departamento de Física, Universidad Nacional de Colombia, Bogotá, Colombia.

Resumen

El siguiente informe tiene como objetivo introducir y dar una descripción de la implementación de notebooks desarrollados durante el programa Beyond Research en el semestre 2021-2. Éstos notebooks fueron desarrollados para el análisis de física de altas energías usando los datos abiertos del experimento ATLAS y están implementados tanto en un lenguaje de programación en c++ como en python, haciendo uso de algunas herramientas del framework ROOT y herramientas alternativas a éste. La descripción de cada notebook contiene información de los cortes implementados para la selección de los eventos de interés, herramientas usadas de ROOT, librerías adicionales de c++ o python, así como también un repositorio en GitHub en el cual están almacenados. En primer lugar, se detallan los notebooks implementados en c++, con un total de 4 notebooks que varían en nivel de dificultad dependiendo del análisis y herramientas implementadas. Finalmente, se procede con la descripción de los notebooks implementados en python, los cuales contendrán 3 notebooks adicionales para un total de 7 análisis implementados.

Palabras claves: ROOT, notebooks, c++, python, HEP, pandas.

Notebooks implementados en c++

- **Reconstrucción de la masa invariante del bosón Z decayendo en 2 leptones**

  Éste notebook es la reproducción del análisis de la reconstrucción de la masa invariante del bosón Z en python, el cual se encuentra en la carpeta python en el repositorio de notebooks del ATLAS Open-Data. Sin embargo, el notebook que reproduce el análisis previo es implementado en c++ y se añaden otras herramientas adicionales que mejoran significativamente el análisis.

  Dentro de las herramientas de ROOT usadas en la implementación se encuentran:

  - **TFile:** Usado para leer la información de los eventos del archivo root.
  - **TTree:** Usado para guardar la información de los eventos del tree llamado "mini" del archivo root.
  - **TH1F:** Usado para crear los histogramas que se llenan una vez termina la selección de eventos.
  - **TLorentzVector:** Usado para almacenar la información del pT, η, ϕ y la energía para cada lepton. Adicionalmente, es usado para calcular la masa del sistema de dos leptones.
  - **TF1:** Usado para definir una función que pueda ajustarse al histograma, calculando la masa del bosón Z y la desviación estándar del ajuste.
  - **TLegend y TLatex:** Son usadas para asignar un legend a las muestras graficadas en un histograma y agregar texto en latex dentro de la gráfica respectivamente.

  De ésta manera, al análisis que se encuentra en la página oficial del ATLAS Open-Data de la reconstrucción de la masa invariante del bosón Z, se añade la implementación en c++ con un ajuste gaussiano que se puede implementar en general para cualquier histograma producto de cualquier otro análisis en HEP que tenga una distribución parecida. Sin embargo, también se puede cambiar la función de ajuste por cualquier otra función (como una polinómica). La gráfica resultante puede verse en la gráfica [1].

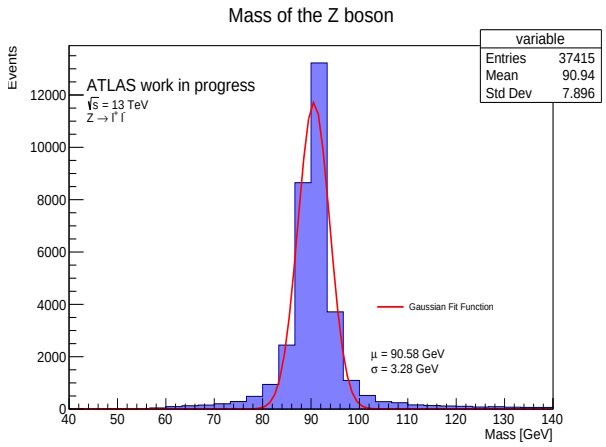

Figure 1: *Reconstrucción de la Masa Invariante del Bosón Z y su ajuste con TF1.*

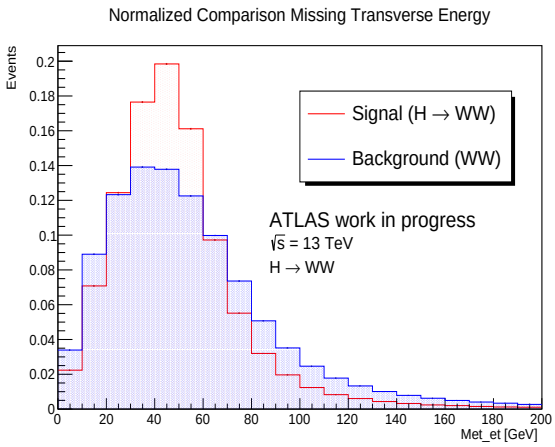

Figure 3: *Comparación normalizada del MET en cada evento teniendo el cuenta el weight.*

- **Comparación de la señal $H \to WW$ con otros procesos de producción de dibosones WW**

  En éste notebook se toman dos archivos root que tiene información sobre la producción de bosones WW de procesos en el modelo estándar y la señal del $H \to WW$. Para ello, se definen dos objetos TFile y TTree, uno para la señal y el otro para el background. Adicionalmente, se definen 4 histogramas en total, los cuales son definidos para llenar información del número de jets y la energía perdida transversa (MET) para cada evento.

  Así, ésta implementación no solamente reproduce el análisis de la comparación de ambos archivos root de la página del ATLAS Open-Data, sino que además se implementa en c++ y se añade al análisis el efecto de considerar el weight de cada evento al momento de generar ambas gráficas.

  Los resultados obtenidos al implementar éste notebook pueden verse en las figuras [2] y [3] (el número de jets y el MET respectivamente).

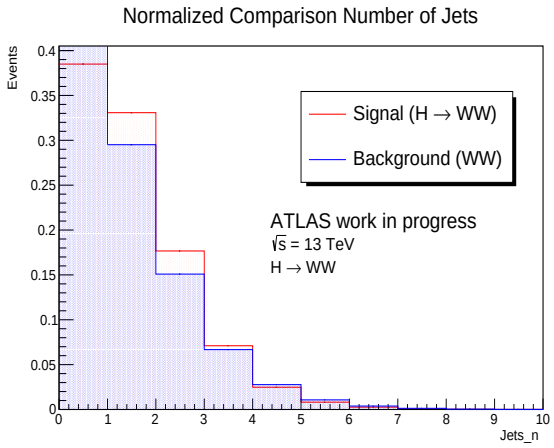

Figure 2: *Comparación normalizada del número de jets en cada evento teniendo el cuenta el weight.*

Hasta aquí, todos los notebooks implementados mantienen un nivel introductorio-medio. Cada uno de ellos reproduce el análisis de algunos notebooks en la página del ATLAS Open-Data que estaban desarrollados en python, pero con la condición de que ahora se implementan en c++ y se añaden herramientas adicionales (como ajustes, el event weight, etc) para mejorar el análisis.

- **Análisis del Graviton**

  En éste análisis se busca reproducir los resultados del notebook de la carpeta uproot que contiene el análisis del graviton. Debido a que este notebook usa uproot como herramienta alternativa de ROOT en python, el análisis que se desea obtener debe ser implementado en c++ y usando el framework de ROOT. Para ello, se usan las mismas muestras del notebook de uproot del análisis del graviton y se define una cadena de archivos root para cada background.

  Éste background contiene muestras MC de la producción del Z, $t\bar{t}$, $tt\bar{V}$, ZZ y la señal del graviton.

  Al análisis se implementan únicamente dos cortes:

  - **Corte 1 (lep_charge_cut):** Éste corte selecciona los eventos para los cuales la suma de las cargas de todos los 4 leptones es igual a cero.
  - **Corte 2 (cut_lep_type):** Éste corte selecciona eventos en los cuales cada par de leptones son del mismo sabor.

  Así, al aplicar ambos cortes se seleccionan eventos en los cuales los leptones sean $eeee$, o $ee\mu\mu$, o $\mu\mu\mu\mu$.

  Las funciones definidas en el análisis son las siguientes:

  - **set_branch_address:** Esta función se encargar de asignar los branches de cada archivo root a la cadena de los datos o las muestras MC. Al hacer una función, recibe como argumento el nombre

de la cadena y se encarga de asignar los valores de los branches a cada cadena por separado. Así, se evita repetir las mismas líneas de código para cada cadena, ya que tanto los datos como los MC usan las mismas variables.

- **fill_histograms:** Esta función se encarga de llenar los histogramas de los datos y las muestras MC. Para ello, recibe el nombre de la muestra, su peso, y el valor de la masa del sistema de 4 leptones. Así, también se evita repetir las mismas líneas de código para los diferentes histogramas.
- **get_xsec_weight:** Esta función define los valores del peso de cada evento debido a la sección eficaz, la eficiencia del detector y la suma de los pesos *sumw*. Luego, almacena éstos pesos en un vector donde cada componente representa una muestra de un archivo root usado para el análisis.
- **calc_weight:** Esta función calcula el peso total (variable weight) a partir del peso calculado en la función *get_xsec_weight*, los scaleFactors de cada evento, y el mcWeight almacenado en los trees para cada evento, dando como resultado el peso total que se usa para llenar los histogramas. Tal función recibe entonces adicionalmente el nombre de la cadena, pues realiza éste procedimiento por separado para cada cadena, evitando repetir líneas de código.
- **calc_mllll:** Esta función calcula la masa del sistema de los 4 leptones en el estado final. Recibe como argumentos el pt, η ,ϕ y la Energía de los leptones para cada evento.
- **analysis_of_samples:** Esta función define la selección de los eventos dada una cadena de muestras. Recibe como argumentos el nombre de la cadena y el vector que contiene los pesos asociados a la función *get_xsec_weight*.
- **graviton_analysis:** Esta función genera el análisis para todas las muestras de datos y MC. Imprime el nombre de la cadena, el número de eventos iniciales y el número de eventos que pasaron todos los cortes.

Para éste análisis se graficó la masa del sistema de los 4 leptones en el estado final. Este resultado se puede observar en la gráfica [4].

- $H \rightarrow WW$ **Analysis**

En éste análisis se busca implementar un notebook que ejecute el mismo análisis del framework de 13TeV de la página del ATLAS Open-Data usando todas las muestras que éste contiene. Por lo tanto, se implementan varios cortes con el objetivo de mejorar la selección de eventos. Entre éstos cortes podemos encontrar:

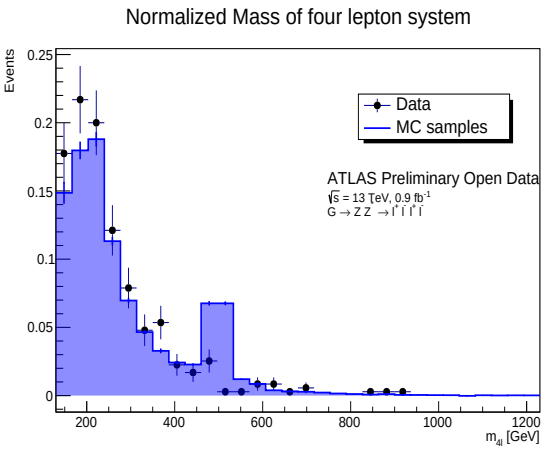

Figure 4: *Comparación normalizada del cálculo de la masa del sistema de 4 leptones entre los Datos y los MC para el Graviton Analysis notebook.*

- **Corte 1 (good_leptons_n_cut):** Éste corte selecciona eventos que contengan únicamente 2 buenos leptones, es decir: que tengan un $pT > 22GeV$ para el leadlepton y un $pt > 15GeV$ para segundo lepton, que el lep_isTightID sea verdadero, que estos leptones estén bien aislados, y que cumpla las condiciones para el η si es un muón o un electrón.
- **Corte 2 (flavour_leptons_cut):** Selecciona eventos en los cuales los dos buenos leptones sean de diferente sabor.
- **Corte 3 (opposite_charges_cut):** Selecciona eventos en los cuales los dos buenos leptones tienen cargas opuestas.
- **Corte 4 (selection_goodjets_and_zero_bjets_cut):** Selecciona eventos en los cuales hay por lo mucho un buen jet y ningún buen bjet. Para que sea un buen jet se exige que cada jet tenga un $pt > 20GeV$, que se encuentre en $|\eta| < 2.5$, que cumpla con el requisito del JVT para jets con $pt < 60GeV$ se debe cumplir que $|\eta| < 2.4$ y que su $jvt > 0.59$.
- **Corte 5 (low_mass_meson_resonances_DY_and_ggF_regions_cut):** Este corte selecciona eventos para los cuales el ángulo azimutal entre la energía perdida y el sistema de dos leptones es mayor a $\pi/2$. Así como también eventos en los cuales el ángulo azimutal entre los dos leptones es menor a 1.8. También elimina eventos para los cuales $MET < 20GeV$ y la masa del sistema de dos leptones no está entre 10-55GeV.

Adicionalmente, se agregan las funciones de **set_branch_address**, **fill_histograms**, analysis_of_samples y *HWW_analysis* que cumplen un papel análogo a las funciones del mismo nombre dadas en el notebook del análisis del graviton previamente definido.

Para éste análisis se graficaron la masa y el pT del sistema de dos leptones, la masa reconstruida o transversa y el MET. Éstas gráficas pueden verse en las figuras [5], [6], [7] y [8] respectivamente.

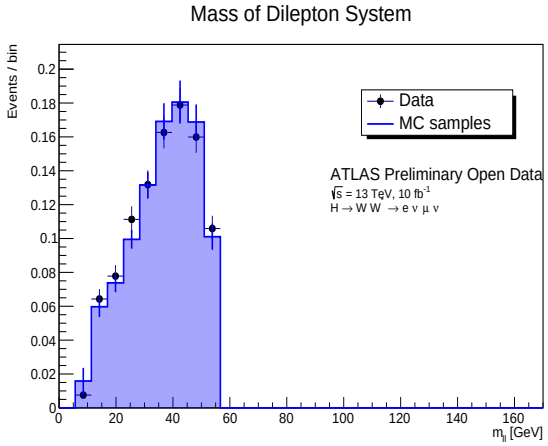

Figure 5: *Comparación normalizada del cálculo de la masa del sistema de 2 buenos leptones entre los Datos y los MC para el análisis de* $H \rightarrow WW$.

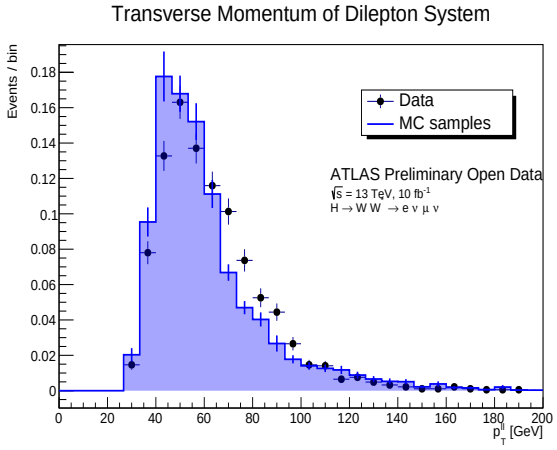

Figure 6: *Comparación normalizada del cálculo del pt del sistema de 2 leptones entre los Datos y los MC para el análisis de* $H \rightarrow WW$

Notebooks implementados en python

- **Reconstrucción de la masa del bosón Z usando pandas**

  Este notebook consiste en reproducir el resultado de la reconstrucción de la masa invariante del bosón Z que es implementado en python en la página del ATLAS Open-Data, pero ahora usando herramientas alternas a pyROOT. Para ello, se implementan las siguientes herramientas:

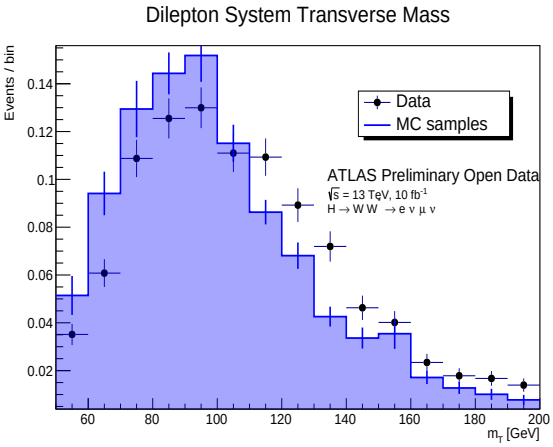

Figure 7: *Comparación normalizada del cálculo de la masa transversa entre los Datos y los MC para el análisis de* $H \rightarrow WW$.

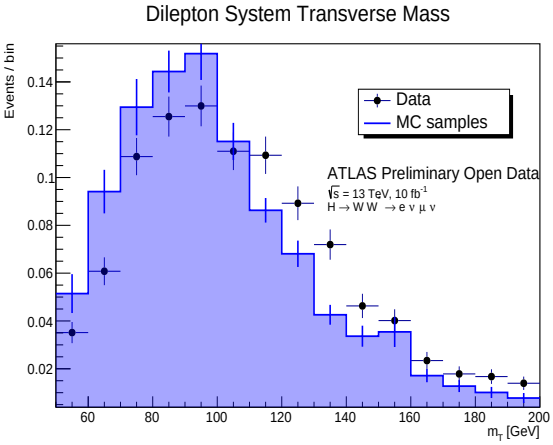

Figure 8: *Comparación normalizada del cálculo del MET entre los Datos y los MC para el análisis de* $H \rightarrow WW$

  - **Uproot:** Esta herramienta nos sirve para leer y abrir archivos .root. De ésta manera es posible acceder a la información de cada evento almacenado en las muestras de MC.
  - **Numpy:** Es la librería numérica de python y es utilizada para realizar operaciones numéricas en las funciones, como por ejemplo, el cálculo de la masa invariante.
  - **Scipy:** Es una librería con herramientas para el análisis científico. En éste caso, es usada para realizar un ajuste de los datos y obtener los parámetros de un ajuste gaussiano.
  - **Pandas:** Es usado para almacenar información de los eventos como DataFrames, permitiendo un mejor y visualización de los datos de los eventos de interés.
  - **Matplotlib:** Es usado para plotear y visualizar las gráficas resultantes del análisis una vez se tengan todos los eventos de interés.

Así, éste notebook realiza un análisis relativamente sencillo ya que se centra en introducir herramientas básicas del uso de pandas, y como ésta es usada específicamente para visualizar información de los eventos, almacenamiento de éstos en dataframes y selección de los eventos de interés por aplicación de los cortes. Adicionalmente, se muestra como realizar un ajuste gaussiano a las gráficas resultantes a través de la librería scipy.optimize.

Los cortes implementados en el análisis son tres:

- **Corte 1 (at_least_two_leptons_cut):** Éste corte selecciona eventos para los cuales se tienen por lo manos 2 leptones en el estado final.
- **Corte 2 (opposite_charges_cut):** Éste corte selecciona los eventos para los cuales se tiene que los dos leptones en el estado final poseen cargas opuestas. Así debe ser ya que sabemos que el bosón Z es neutro y por lo tanto ambos leptones deben tener cargas opuestas.
- **Corte 3 (lep_type_cut):** Éste corte selecciona los eventos para los cuales los dos leptones resultantes son leptones del mismo sabor, siendo así los posibles casos e^-e^+ o $\mu^-\mu^+$.

  Con éstos tres cortes principales se realizó el análisis y se almacenó la información de los eventos seleccionados, añadiendo el valor calculado de la masa invariante del sistema de dos leptones, para posteriormente realizar las gráficas resultantes.

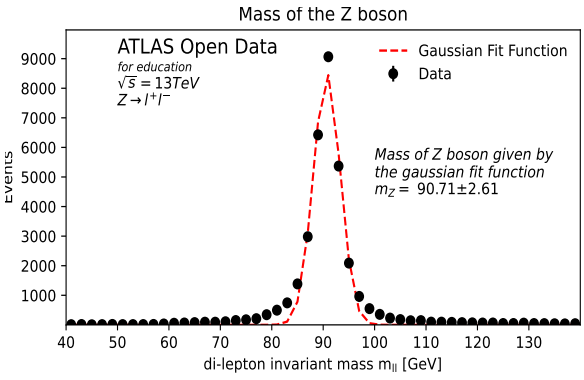

Figure 9: *Reconstrucción de la masa invariante del bosón Z usando pandas (visualización en forma de puntos)*

Finalmente, los resultados del análisis son mostrados de dos maneras diferentes: En primer lugar, se grafica la masa invariante del bosón Z tomando los puntos medios de cada bin en el histograma, de manera que la figura no muestra un histograma sino una sucesión de puntos que están ajustados a una función gaussiana. En segundo lugar, se muestra el histograma con el ajuste. En ambas gráficas se imprime el valor de la masa del Z proveniente del ajuste gaussiano junto con la desviación estándar. Éstas gráficas son representadas en las figuras [9] y [10].

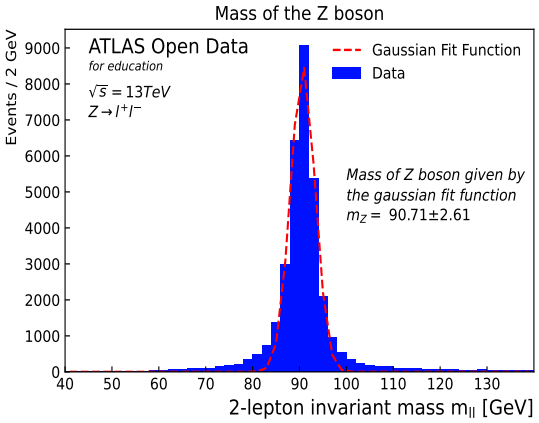

Figure 10: *Reconstrucción de la masa invariante del bosón Z usando pandas (visualización en forma de histograma)*

- **Análisis del $H \to ZZ$**

  Éste notebook implementa el análisis del $H \to ZZ$ usando herramientas alternas a pyROOT, como aquellas que fueron mencionadas en el notebook anterior, especialmente pandas. Las fuentes de background consideradas en el análisis son:

  - $Z, t\bar{t}$ **:** Se usan muestras MC de la producción de Z + jets y $t\bar{t}$ ya que éstos pueden producir en el estado final 4 leptones.
  - ZZ^***:** Producción de dibosones ZZ ya que el Z decae en dos leptones de cargas opuestas, produciendo 4 leptones en el estado final.
  - $H \to ZZ$**:** Aquí se toman muestras de producción del $H \to ZZ$ ya sea por procesos de gluon-gluon fusion, vector-boson fusion, etc.

  Se implementan en total dos simples cortes los cuales son:

  - **Corte 1 (cut_lep_charge):** Selecciona eventos para los cuales la suma de las cargas de todos los 4 leptones en el estado final dan igual a cero, manteniendo así la neutralidad de carga.
  - **Corte 2 (cut_lep_type):** Seleciona eventos en los cuales los 4 leptones se agrupan en dos pares de leptones del mismo sabor. Por lo cuál, se tienen las posibles combinaciones de leptones en el estado final: $e^-e^+e^-e^+$, $e^-e^+\mu^-\mu^+$ o $\mu^-\mu^+\mu^-\mu^+$.

  Para el análisis se definieron las siguientes funciones:

  - **get_data_from_files:** Esta función permite guardar la información de todas las muestras y

almacenarlas en dataframes. En ella se llama a las demás funciones que realizan el análisis, las cuales serán explicadas más adelante. Finalmente, ésta función retorna objetos dataframes que serán los que contienen la información de los eventos seleccionados.

- **get_xsec_weight:** En ésta función se calculan los pesos de los eventos teniendo en cuenta la sección eficaz, la eficiencia del detector y la suma de los pesos *sumw*. Así, importando el archivo *infofile.py*, es posible acceder a la información de todos éstos parámetros de cada muestra.
- **calc_weight:** Ésta función calcula el peso total al multiplicar el peso asociado a la función *get_xsec_weight* junto con los scaleFactors y el mcWeight asociado a todos los eventos. Éste *total_weight* es el usado al momento de graficar los histogramas.
- **calc_mllll:** Ésta función calcula la masa del sistema de 4 leptones en el estado final a partir de recibir como argumentos el pT, η, ϕ y E de cada lepton.
- **read_file:** En ésta función se asocian el peso total a todas las muestras MC, se aplican los cortes y se selecciona los eventos que pasan la selección. Por último, añade la información del cálculo de la masa de los 4 leptones y retorna un dataframe con toda la información relevante que se usó para el análisis. También imprime el nombre de la muestra que está siendo procesada, junto con el número de eventos iniciales y el número de eventos que pasaron la selección.

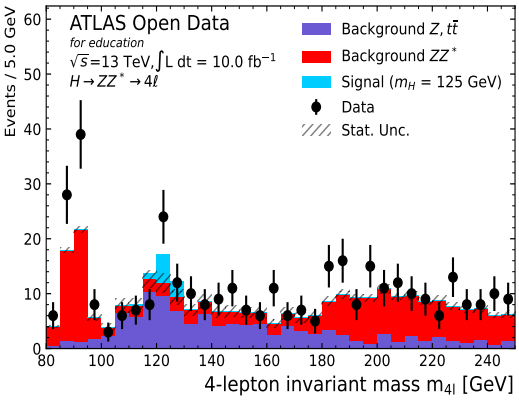

Figure 11: *Masa del sistema de 4 leptones en el estado final para el análisis del $H \to ZZ$ usando pandas*

Así, teniendo la información de la masa del sistema de 4 leptones se procede a graficar en un histograma el resultado final del análisis. Ésta gráfica puede verse en la figura [11]. La selección de los eventos puede mejorarse aplicando otros cortes, pero la idea de éste notebook consiste además, en introducir herramientas de pandas a un nivel intermedio, y a su vez, reproducir el resultado obtenido por otros métodos del notebook que implementa el $H \to ZZ$ del ATLAS Open-Data.

- **Análisis $Z \to \tau\tau$**

  Éste notebook implementa el análisis del $Z \to \tau\tau$ de uno de los doce frameworks del ATLAS Open-Data, donde uno de los leptones τ decae hadronicamente y el otro leptónicamente en un lepton más ligero (muón o electrón).

  La idea es implementar un análisis que fue implementado en un framework únicamente en un notebook y usando herramientas alternas de pyROOT. Así, se usaron las herramientas descritas previamente en los anteriores dos notebooks, incluyendo pandas.

  Además, se usaron todas las muestras que se usan en el framework de análisis, por lo que éste notebook tarda un tiempo considerable en ejecutarse debido a la cantidad de muestras a analizar.

  Dentro de las fuentes de background consideradas tenemos:

  - *Single_top*, $t\bar{t}$, **DY y dibosones:** En éste background juntamos el aporte debido a procesos de single top y $t\bar{t}$, junto con los procesos de producciones de Drell-Yan y dibosones ZZ, WZ y WW.
  - $W + jets$**:** En este background se encuentra un bosón W junto con un leptón e, μ o τ y neutrino.
  - $Z \to ee, \mu\mu$**:** En éste background separamos el decaimiento del Z en leptones ligeros, como el electrón o muón, de la señal de interés ($Z \to \tau\tau$).

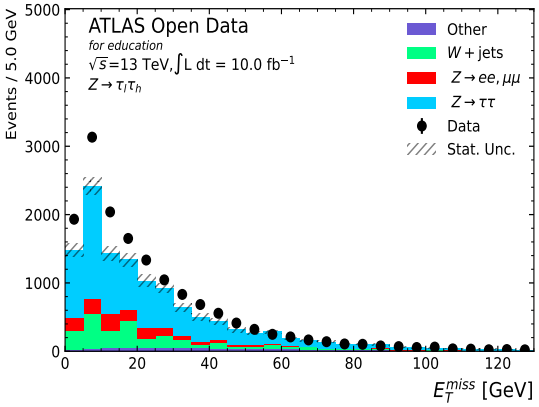

Figure 12: *Energía transversa perdida para el análisis del $Z \to \tau\tau$* usando pandas.

La selección de eventos se hace aplicando los siguientes cortes:

- **trigger:** Se requiere que se active el trigger para un lepton, sea un electrón o un muón.

- **goodlep_cut:** Selecciona eventos para los cuales hay únicamente un buen lepton, teniendo en cuenta que el pT de éstos debe ser mayor a 30 GeV, que se cumpla con la condición de que sea $lep\_isTIGHTID$ verdadero, y que dependiendo de si es un electrón o muón se encuentre dentro de las regiones para η permitidos.
- **goodtau_cut:** Selecciona eventos para los cuales solamente hay un buen lepton τ, teniendo en cuenta que el pT del lepton τ debe ser mayor a 25GeV, que debe cumplirse que $tau\_isTIGHTID$ sea verdadero, y que $|\eta| < 2.5$.
- **opposite_charge_cut:** Se requiere que los dos buenos leptones (el τ y el lepton ligero) tengan cargas opuestas.
- **transverse_mass_cut:** Se requiere que la masa transversa del W calculada a partir del MET y el leptón ligero sea menor a 30 GeV.
- **visible_mass_cut:** En el caso de la masa visible, que se calcula a partir del sistema de dos leptones, el τ y el lepton ligero, debe estar en el rango de $35GeV < m_{vis} < 75GeV$.
- **total_sum_dPhi_cut:** Selecciona eventos para los cuales la separación angular entre el MET y el τ, junto con la separación del MET y el leptón ligero, sumados den menor a 3.5.

Así, de manera similar al desarrollo y ejecución de las funciones definidas en el análisis del notebook anterior, se calcula el peso total para todas las muestras MC, teniendo en cuenta la sección eficaz y la información adicional suministrada en el archivo *infofile.py*. También se calculan las variables de la masa transversa, la masa visible y las separaciones angulares entre los leptones y el MET.

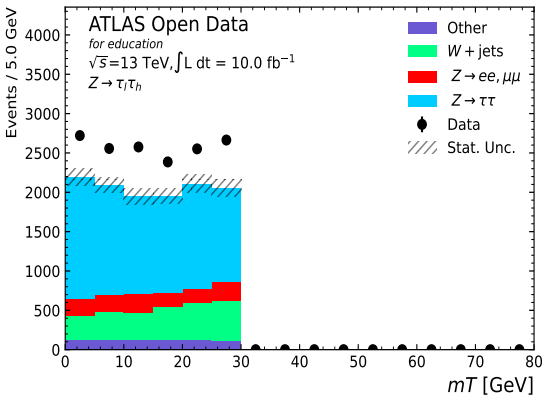

Figure 13: *Reconstrucción de la masa transversa del bosón W en el análisis $Z \to \tau\tau$ usando pandas*

Al finalizar el análisis, se tiene un dataframe con la información relevante de cada evento, así como también el calculo de las variables mencionadas previamente. También se grafican en el análisis la masa transversa del W, la masa visible y la energía transversa perdida. Éstas gráficas son mostradas en las figuras [12], [13] y [14] para el MET, la masa transversa y la masa visible respectivamente.

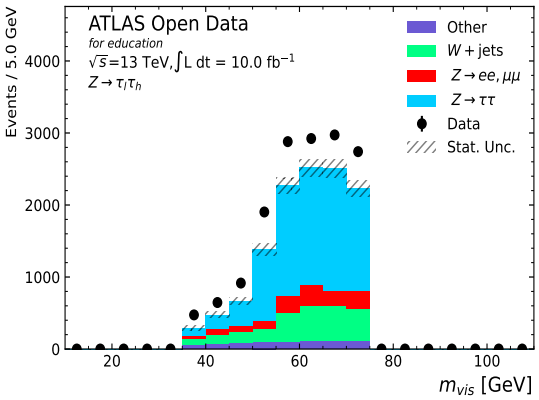

Figure 14: *Reconstrucción de la masa visible entre los dos buenos leptones en el análisis del $Z \to \tau\tau$ usando pandas*

Todos los notebooks descritos en éste informe se pueden encontrar en un repositorio en github [1], el cual está actualizado hasta la fecha de rea
lizado el informe. Allí se podrán encontrar dos carpetas principales, las cuales separan los notebooks en c++ y python. También se adjuntan enlaces directos al repositorio de la coleccioón de notebooks del ATLAS Open-Data [2], su página oficial [3] y la documentación del framework de análisis de 13 TeV [4].

Referencias

[1] Repositorio en GitHub del Beyond Research
`https://github.com/Miagarciaru/Beyond_Research_2021_2`

[2] Repositorio en GitHub de la colleción de ATLAS Open-Data
`https://github.com/atlas-outreach-data-tools/notebooks-collection-opendata`

[3] Website oficial del ATLAS Open-Data
`http://opendata.atlas.cern/release/2020/documentation/index.html`

[4] Documentación del framework de análisis de 13 TeV
`https://cds.cern.ch/record/2707171/files/ANA-OTRC-2019-01-PUB-updated.pdf`