

**VISOKA ŠKOLA STRUKOVNIH NAUKA ZA INFORMACIONE
TEHNOLOGIJE**



Analiza velikih podataka (Big Data)

Projekat

**Replikacija analize podataka za
Higsov bozon: Raspad u četiri
leptona u ROOT Framework-u**

Predmetni nastavnik:

Dr Aleksandar Simović

Student:

Viktor Varkulja 469/23

Beograd

februar, 2025.

SADRŽAJ

UVOD.....	3
OBJAŠNJENJE FIZIKE	3
REPLIKACIJA ANALIZE PODATAKA	4
PRVI NIVO REPLIKACIJE.....	5
DRUGI NIVO REPLIKACIJE.....	6
TREĆI NIVO REPLIKACIJE.....	9
ANALIZA PYTHON I C++ FAJLOVA.....	11
ZAKLJUČAK.....	16
LITERATURA	17
LINKOVI	17

UVOD

U fizici elementarnih čestica potrebne su veoma zahtevne kompjuterske operacije da bi se izvršili eksperimenti koji mogu da utvrde razne teorije iz te oblasti. Tako su u CERN-u uspeli 2012. godine da utvrde postojanje Higsovog bozona, koji je do tad samo na teoretskoj bazi postojao. Pomoću Kompaktnog mionskog solenoida (Compact Muon Solenoid – CMS), detektor koji je napravljen za otkrivanje Higsovog bozona, koji je deo Velikog hadronskog sudarivača (Large Hadron Collider – LHC), uspeo je da detektuje da se Higsov bozon raspada u dva Z bozona i onda u četiri leptona (s tim da postoje i druga raspadanja).¹

Za postizanje ovih rezultata potrebno je bilo sudarati ogromnu količinu grupisanih protona jedan sa drugim da bi te kolizije bile snimljene pomoću CMS-a. Po zvaničnim podacima više od 1 milijarde kolizija se dešavalo svake sekunde.² Za ovaj eksperiment je bilo potrebno korišćene najboljih tehnologija za prenos, čišćene, obradu i analizu podataka. Zato za analizu i vizualizaciju podataka su koristili ROOT Framework. ROOT Framework je objektno-orijentisan softverski okvir za skladištenje, obradu, analizu i vizualizaciju podataka, koji su u CERN-u razvili tako da bude prikladan baš za ovakve eksperimente i skupove podataka.³

Ovaj projekat će replicirati dobijen rezultat tokom otkrića Higsovog bozona pomoću CERN-ove Open Data platforme, gde su pristupni odgovarajući podaci za ovaj eksperiment. Na ovoj platformi se nalazi uputstvo kako treba doći do ovih rezultata. Uputstvo nudi i izvorne kodove za postizanje željenih rezultata. Podaci su izabrani da budu što lakši i što brži za obradu, ali nude i verziju gde se sa svim podacima vrši analiza. Za to predviđena obrada traje skoro mesec dana. Od izvornog koda ima nekoliko C++ i Python fajlova, koji su zaduženi za analizu i ima više ROOT fajlova koji sadrže podatke potrebne za obradu. Cilj projekta je da replicira rezultat raspadanja Higsovog bozona u četiri leptona i da objasni izvorni kod, odnosno da objasni proces do rezultata i na kraju da objasni značaj rezultata.

OBJAŠNJENJE FIZIKE

Fiziku možemo podeliti po veličini i po brzini kretanja objekata. U slučaju da su objekti veliki i sporo se kreću, onda pričamo o klasičnoj fizici. Kada se objekti brzo kreću, otprilike brzinom svetlosti (c) onda pričamo o relativističkoj fizici, a kad su mali objekti, otprilike veličine atoma, onda o kvantnoj fizici. U slučaju kad je objekat mali i brzo se kreće, onda je reč o kvantnoj teoriji polja. Ova teorija spaja relativističku fiziku i kvantnu fiziku. Cilj ove teorije je da da okvir u kojem se može objasniti, pomoću dinamike elementarnih čestica, zakonitost ponašanja čestica. Relativistički deo teorije omogućava gubitak karakteristike objekata, npr. mase, ali obezbeđuje konzervaciju energije. Kvantni deo teorije, međutim, daje objektima takvu osobinu, da im ponašanje nije potpuno predvidljivo nego samo sa određenom verovatnoćom se može predvideti. Ovo utiče na pozicioniranje čestica ili na raspadanje raznih čestica. A kvantna teorija polja nudi

¹ Chatrchyan S., et al., *Observation of a New Boson at a Mass of 125 GeV with the CMS Experiment at the LHC*, [CMS], Phys. Lett. B 716 (2012), 30-61, str. 1

² <https://home.cern/resources/faqs/facts-and-figures-about-lhc>

³ Brun R., Rademakers F., ROOT—An object oriented data analysis framework. Nuclear instruments and methods in physics research section A: accelerators, spectrometers, detectors and associated equipment, 389(1-2), pp.81-86., 1997., str. 1-2

i druge osobine koje se ne pojavljuju u drugim oblicima fizike, kao što su postojanje antičestica ili teorema simetrije naelektrisanja, parnosti i vremenske inverzije.⁴

Trenutno prihvaćena teorija, koja objašnjava tri od četiri osnovne interakcije između čestica je Standardni Model. Sa Standardnim Modelom fizičari su uspjeli da dokažu i objasne tri osnovne interakcije između čestica, to su: elektromagnetna, jaka i slaba sila između čestica. Četvrta interakcija je gravitaciona sila, ali to još nisu uspjeli da dokažu. Ova teorija predlaže da postoje fermioni i bozoni. Fermioni su osobe materija, a bozoni su nosioci sile. Fermioni se mogu raspodeliti u kvarkove i leptone. Kvarkovi su subatomske čestice koje čine protone i neutrone pomoću jake sile. Postoji šest flejvora (ukusa) kvarkova, to su: u (up), d (down), s (strange), c (charm), t (top) i b (bottom). Leptoni su čestice na koje ne utiče jaka sila nego slaba i mogu samostalno da postoje. Leptoni isto postoje u šest ukusa: elektron (e), mion (μ), tau (τ), elektron-neutrino, mion-neutrino i tau-neutrino. Neutrini su neutralno naelektrisane čestice. Postoje dve vrste bozona: baždarni i skalarni bozoni. Baždarni bozoni su nosioci sile i medijatori 3 glavne interakcije između čestica koje je Standardni Model uspeo da objasni. Postoji foton koji je nosilac elektromagnetske sile, Z i W bozoni koji su nosioci slabe sile, a gluoni su nosioci jake sile. Od skalarnih bozona trenutno postoji samo Higsov bozon.⁵

Higsov bozon je čestica koja daje masu ostalim česticama. Međutim, fotoni i gluoni ne poseduju osobinu mase, znači oni nisu u interakciji ili su u minimalnoj interakciji sa Higsovim poljem. Do 2012. godine Higsov bozon je postojao samo u teoriji, ali te godine u CERN-u su uspjeli da dokažu da postoji ta čestica. Pomoću podataka iz LHC-a konstatovali su da postoji jedna čestica koja se raspada u dva fotona ili se raspada u dva Z bozona pa u četiri leptona (elektroni ili mioni). Ova čestica ima masu od 125 GeV (giga-elektronvolt – elektronvolt je energija koju poseduje slobodan elektron prolaskom kroz potencijalnu razliku od 1 volta, $1 \text{ eV} = 1,6 \times 10^{-19} \text{ J}$) i širinu od 100 MeV. Ovaj eksperiment je bio odrađen sa energijom sudara u centralnoj masi od 7 i 8 TeV, što znači da je sudaranje protona iznosilo ukupno 7 pa u sledećem krugu 8 TeV energije. A verovatnoća proizvodnje Higsovog bozona je bila između 23 i 14 pb (piko-barni – barn je jedinica površine koja izražava verovatnoću sudara ili interakcije čestica), što je veoma mala verovatnoća.⁶

REPLIKACIJA ANALIZE PODATAKA

Za replikaciju analize prvo je potrebno razumeti šta je bio rezultat eksperimenta. Na slici 1. se vidi rezultat eksperimenta. Crne tačke označavaju podatke koji su prikupljeni, plavo polje označava pozadinske signale, to su događaji koji ne potiču iz Higsovog bozona. Plavo polje odgovara predviđenoj distribuciji, gde je očekivano da se kod mase Z bozona, što je 91 GeV, nalazi vrh raspodele. Međutim, crvenom se vidi još jedan vrh koji se nalazi kod mase od 125 GeV. Ovaj detalj je dokaz da Higsov bozon ima masu od 125 GeV.⁷

Replikacija analize podataka se može odraditi prateći korake napisane na sledećem linku: <https://opendata.web.cern.ch/record/5500>. Za podatke se koriste Monte Karlo (algoritam za simuliranje događaja) simulacije, koji nisu originalni podaci ali su približni tome. Pored toga

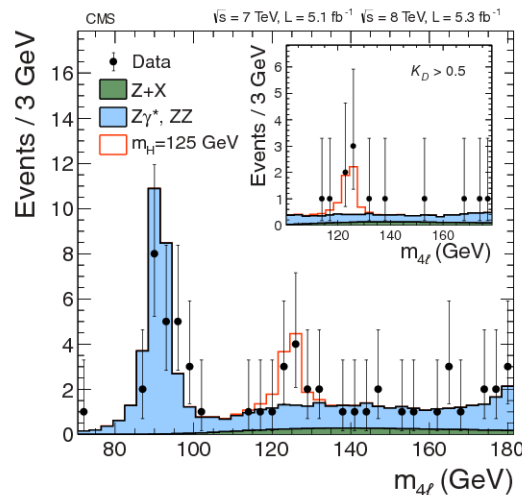
⁴ Griffiths D., *Introduction to Elementary Particles*, John Wiley & Sons, Protland, Oregon, USA, 2008., str. 1-4

⁵ Ibid, str. 46-48, 55-56

⁶ Chatrchyan S., et al., *Observation of a New Boson at a Mass of 125 GeV with the CMS Experiment at the LHC*, [CMS], Phys. Lett. B 716 (2012), 30-61, str. 1-2

⁷ Ibid, str. 12

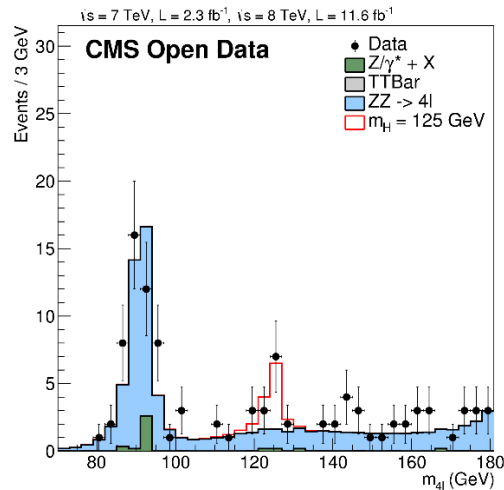
izvorni kod nije identičan originalnom, smanjen je i pojednostavljen, ali kod ipak daje dobar uvid u način obrade podataka. Strategija obrade podataka je da se uzmu poslednja stanja 4μ , $2e2\mu$ iz DoubleMuParked skupa podataka, a finalno stanje $4e$ se uzima iz DoubleElectron dataset-a. A nakon toga se vrši normalizacija podataka, gde su ZZ (dupli Z bozon) kontribucije skalirane na masovni opseg od 180-600 GeV, a Higsova kontribucija je skalirana podacima u signalnom opsegu (80-180 GeV).



Slika 1 - Objavljena vizualizacija rezultata (izvor: Chatrchyan S., et al., *Observation of a New Boson at a Mass of 125 GeV with the CMS Experiment at the LHC*, [CMS], Phys. Lett. B 716 (2012), 30-61, str. 13)

PRVI NIVO REPLIKACIJE

Replikacija ima 4 nivoa. Za projektni rad je odrađeno samo 3 nivoa, zbog ograničenja u vremenu. Prvi nivo je upoređenje dijagrama iz slike 1 i slike 2. Vidi se da su uslovi rezultata malo drugačiji, to jest energija centralne mase je ostala 7 i 8 TeV, ali luminoznost je različita. To označava količinu sudara pri odgovarajućoj energiji. Pored toga, pozadinski podaci su detaljnije označeni na prvom dijagramu.



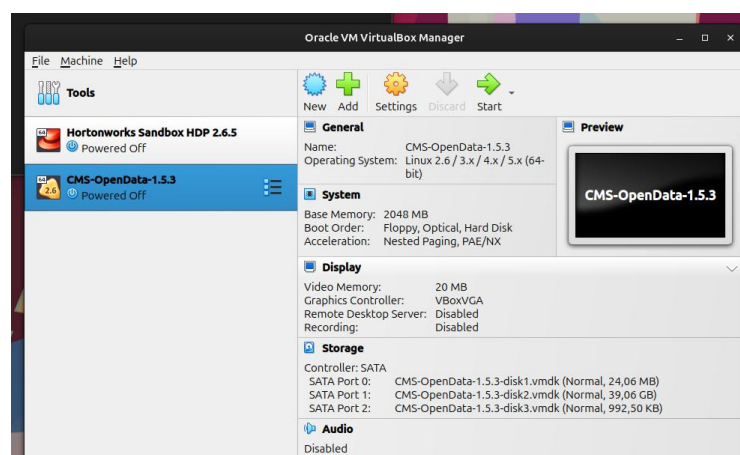
Slika 2 - Rezultat sa podacima iz Monte Karlo simulacije

DRUGI NIVO REPLIKACIJE

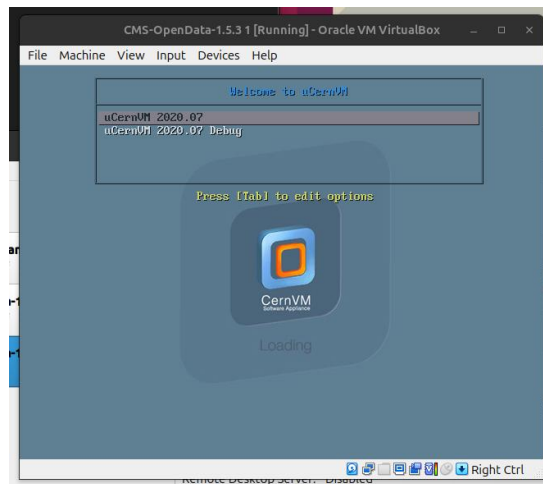
Drugi nivo replikacije je reprodukcija dijagrama sa slike 2. Ovo se postiže sa preuzimanjem ROOT fajlova u kojem se nalaze histogrami po kojem će se pomoću C++ makroa napraviti novi dijagram. Ima dva načina reprodukcije dijagrama:

- lokalno na računaru instaliranjem i korišćenjem ROOT Framework-a,
- ili pomoću virtualne mašine.

Za projektni rad je korištena virtualna mašina (VM), jer za treći nivo je VM obavezna za reprodukciju. Za virtualnu mašinu je potrebno instalirati VirtualBox i skinuti image sa CERN Open Data platforme (na sledećem linku se nalazi VM image: <https://opendata.cern.ch/record/252>). Potrebno je pratiti korake pokretanja virtualne mašine na sledećem linku: <https://opendata.cern.ch/docs/cms-virtual-machine-2011>.

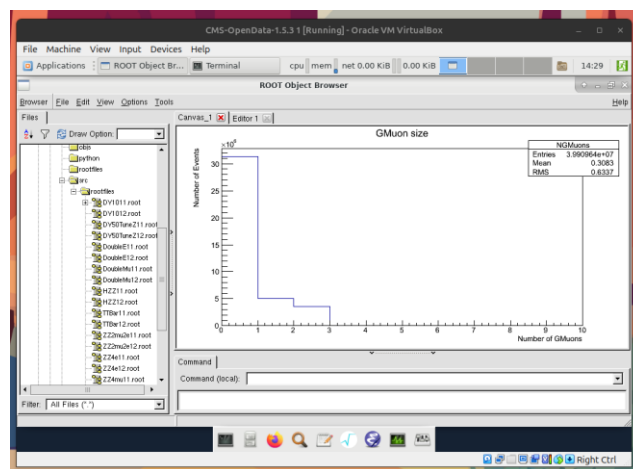


Slika 3 - VirtualBox sa CMS VM image-om



Slika 4 - - Pokretanje CSM virtualne mašine

Nakon pokretanja virtualne mašine potrebno je generisati CMSSW (CMS Software) okruženje koje omogućava dohvaćanje podataka sa mreže i softver za obradu i analizu podataka. Potrebno je u novom direktorijumu skinuti ROOT fajlove koji će se koristiti za analizu podataka. Važna napomena je da prvo treba ispraviti fajl *rootfilelist.txt* da u svakom redu umesto http-ja piše https, jer kada pokušava da dohvati ROOT fajlove onda izbacuje grešku *404 Not Found*. Na slici 5 se vidi kako izgledaju ROOT fajlovi.



Slika 5 - Skinuti ROOT fajlovi

Kada su skinuti ROOT fajlovi onda je potrebno skinuti jedan C++ makro koji će da eksportuje novi dijagram (slika 2).

```
→ #postavljenje okruženja
→ cmsrel CMSSW_5_3_32
→ cd CMSSW_5_3_32/src
→ cmsenv
→
→ #nova datoteka
```

```
→ mkdir rootfiles
→ cd rootfiles
→
→ #dohvatanje ROOT fajlova
→ wget https://opendata.web.cern.ch/record/5501/files/rootfilelist.txt
→ wget -i rootfilelist.txt
→
→ #preuzimanje i pokretanje C++ makroa
→ wget https://opendata.web.cern.ch/record/5500/files/M4Lnormdatall.cc
→ root -l M4Lnormdatall.cc
```

Makro se zove *M4Lnormdatall.cc* i sadržava neophodan izvorni kod za obradu podataka. Kada pregledamo kod, vidimo prvo da su postavljene promenljive koje označavaju razna podešavanja za analizu podataka. Prvo su određeni fajlovi koje će makro koristiti za kreiranje novog dijagrama. To su ROOT fajlovi koje smo preuzimali, odnosno definisan je fajl output-a. Nakon toga su definisane luminoznosti, preseki Monte Karlo simulacije, faktori skaliranja, skalirani preseki za Higsove kontribucije i broj događaja za svaki tip raspadanja.

Posle definisanja parametara dolazi kreiranje histograma koji će na kraju svi biti spojeni u jedan histogram. Prvo se pravi histogram za pozadinske podatke, tj. raspad dva Z bozona u 4 leptona (4μ , $2e2\mu$ i $4e$). Prvo pristupa ROOT fajlu, pa klonira histogram koji se nalazi u njemu u objekat TH1D, koji je objekat za histograme. Nakon toga skalira podatke sa određenom matematičkom funkcijom. U slučaju pozadinskih podataka, funkcija je proizvod luminoznosti, preseka i faktora skaliranja podeljeno sa brojem događaja. Svi ti podaci su odgovarajući za ovaj tip raspadanja i godine podataka. Ova procedura se ponavlja za tri načina raspadanja Z bosona za godine 2011 i 2012. Znači ukupno šest histograma treba da se preuzmu iz ROOT fajlova i potrebno je da se obrade i spoje u novi dijagram koji će biti pozadinski podaci.

Isti proces se izvršava za podatke sa Higsovom kontribucijom. Razlika je da matematička funkcija za skaliranje nema faktor skaliranja nego je u preseku već uračunato skaliranje. Ovo će biti novi vrh na dijagramu koji će biti označen sa crvenom linijom. Nakon toga će da ponovi isto za ostala dva pozadinska podatka ($Z/\gamma^* + X$, $TT\bar{B}$). A na kraju kloniraće histograme iz ROOT fajlova koji nisu iz Monte Karlo simulacije, nego podatke sa merenja. Te podatke nije potrebno skalirati. Kad je to gotovo onda je potrebno kombinovati te TH1D objekte u jedan poslednji dijagram koje se na kraju eksportuje.

//Primer obrade podataka sa parametrima

// Input file directory

string inDir = ".";

...

// Name of the input file for MC

string inFileZZ4mu12 = "ZZ4mu12.root";

...

// Luminosity 2012 and 2011

Double_t lumi12 = 11580.;

...

// MC cross section


```
Double_t xsecZZ412 = 0.077;
...
// Scale factor
Double_t sfZZ = 1.386;
...
// No. of event
Int_t nevtZZ4mu12 = 1499064;
...
// ZZ -> 4mu

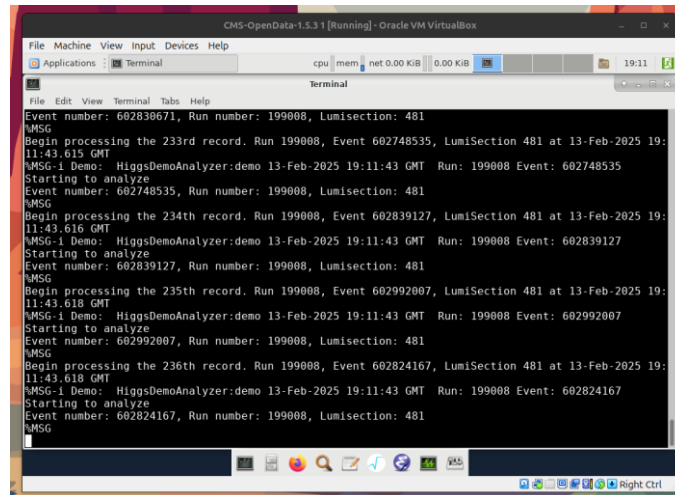
TFile *f2 = new TFile((inDir + inFileZZ4mu12).c_str());
TH1D *ZZto4mu12 = (TH1D*) f2->Get("demo/mass4mu_8TeV_low")->Clone();
ZZto4mu12->Scale((lumi12 * xsecZZ412 * sfZZ) / nevtZZ4mu12);
//((data lumi * xsec * scale factor) / no.of event b4 any cut
```

TREĆI NIVO REPLIKACIJE

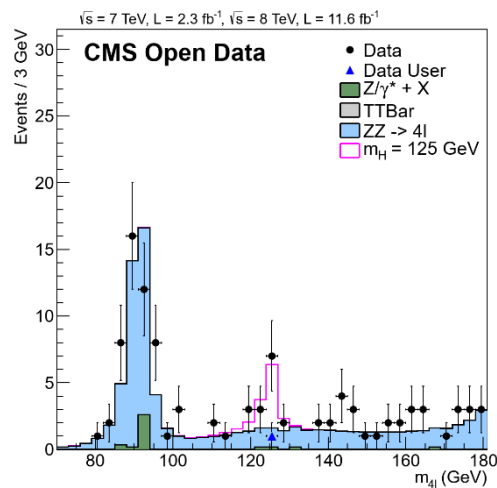
Za treći nivo replikacije je potrebno generisati ROOT fajlove. Jedan sadrži podatke o jednom kandidatu Higsove čestice, a drugi fajl sadrži simulirane Higsove signale sa smanjenom statistikom (zbog ubrzavanja analize). Pomoću tih fajlova treba napraviti novi dijagram koji će imati obeležen jedan Higsov kandidat, koji korisnik može da promeni.

Za početak je potrebno pokrenuti virtualnu mašinu i postaviti okruženje za rad sa podacima. To je već urađeno u drugom nivou. Preporučljivo je, ali nije obavezno, odraditi sekciju *Test & Validation* sa veb stranice gde su napisani koraci za podešavanje virtualne mašine. U ovom slučaju nije odradena ta sekcija. U direktorijumu `~/CMSSW_5_3_32/src` potrebno je napraviti novi folder. U ovaj novi folder je potrebno skinuti *BuildFile.xml* koji će da sadrži naziv biblioteka koje će C++ makro da koristi. Nakon toga je potrebno u novom *src* fajlu da se skine *HiggsDemoAnalyzer.cc*, makro koji će od definisanog skupa podataka da napravi ROOT fajlove. Potrebno je vratiti se u roditeljsku datoteku i kompajlirati ovaj C++ fajl. Nakon kompajliranja potrebno je preuzeti Python fajlove *demoanalyzer_cfg_level3data.py* i *demoanalyzer_cfg_level3MC.py*. Ovi fajlovi će da prikupe odgovarajuće podatke, da pokrenu *HiggsDemoAnalyzer.cc* i da proslede podatke njemu da izanalizira. Potrebno je preuzeti JSON validacioni dataset, koji obezbeđuje da se koriste sirovi podaci koji su najkvalitetniji. To su podaci koji su bili prikupljeni kad je sistem i svi detektori ispravno radili.

Sad je moguće pokretanje analize podataka pomoću komande *cmsRun*. Ova komanda će pokrenuti Python fajlove koji će pokrenuti C++ fajl koji će analizirati podatke i napravi nove ROOT fajlove. Novi ROOT fajlovi će biti *DoubleMuParked2012C_10000_Higgs.root* i *Higgs4L1file.root*. Prvi će sadržati podatke jednog izabranog Higs kandidata, a drugi fajl će sadržati skup podataka sa Monte Karlo simulacije Higsovih signala. Potrebno je prebaciti te fajlove u *rootfiles* datoteku iz drugog nivoa i preuzeti novi makro (*M4Lnormdata11_lv13.cc*). Ovaj makro ima isti zadatak kao makro iz drugog nivoa, tj. da skupi i spoji sve histograme u jedan konačan histogram. Jedina razlika je da sada uzima podatke i iz novih fajlova i obeležava izabrani Higs kandidat. Nakon pokretanja makroa u folderu će biti novi dijagram rezultata. Na slici 7 se vidi da je malim plavom trouglom obeležen Higs kandidat, tačno kod 125 GeV.



Slika 6 - Terminal nakon pokretanja `csmRun demoanalyzer_cfg_level3data.py`



Slika 7 - Novi dijagram nakon trećeg nivoa

```
→ #koraci za treći nivo
→ #preuzimanje BuildField.xml
→ cd ~/CMSSW_5_3_32/src
→ mkdir -p Demo/DemoAnalyzer
→ cd Demo/DemoAnalyzer
→ wget https://opendata.web.cern.ch/record/5500/files/BuildFile.xml
→
→ #preuzimanje i kompajliranje HiggsDemoAnalyzer.cc
→ mkdir src
→ cd src
→ wget https://opendata.web.cern.ch/record/5500/files/HiggsDemoAnalyzer.cc
→ cd ..
→ scram b
→
→ #preuzimanje Python fajlova
→ wget https://opendata.web.cern.ch/record/5500/files/demoanalyzer\_cfg\_level3data.py
```

```
→ wget https://opendata.web.cern.ch/record/5500/files/  
demoanalyzer_cfg_level3MC.py  
→  
→ #nova datoteka za json fajl i za nove root fajlove  
→ mkdir datasets  
→ cd datasets  
→ wget https://opendata.web.cern.ch/record/1002/files/  
Cert_190456-208686_8TeV_22Jan2013ReReco_Collisions12_JSON.txt  
→  
→ #pokretanje analize podataka  
→ cd ..  
→ cmsRun demoanalyzer_cfg_level3data.py  
→ cmsRun demoanalyzer_cfg_level3MC.py  
→  
→ #prebacivanje root fajlova u rootfiles  
→ mv Higgs4L1file.root ../../rootfiles/  
→ mv DoubleMuParked2012C_10000_Higgs.root ../../rootfiles/  
→ cd ../../rootfiles  
→  
→ #preuzimanje i pokretanje c++ makroa  
→ wget https://opendata.web.cern.ch/record/5500/files/M4Lnormdata11_lvl3.cc  
→ root -l M4Lnormdata11_lvl3.cc
```

ANALIZA PYTHON I C++ FAJLOVA

Python fajlovi *demoanalyzer_cfg_level3data.py* (data.py) i *demoanalyzer_cfg_level3MC.py* (MC.py) su izvorni kod koji prikuplja podatke i prosleđuje ih u C++ makro koji ih obrađuje. Suštinski oba fajla imaju isti kod ali razlikuju se od izvora podataka:

- MC.py prikuplja podatke sa Monte Karlo simulacije

```
# to speed up, read only first file with 7499 events  
process.source = cms.Source("PoolSource",  
fileNames =  
cms.untracked.vstring('root://eospublic.cern.ch/eos/opendata/cms/MonteCarlo2012/Summer12_DR53X/SMHiggsToZ  
ZTo4L_M-125_8TeV-powheg15-JHUGenV3-pythia6/AODSIM/PU_S10_START53_V19-v1/10000/029D759D-6CD9-  
E211-B3E2-1CC1DE041FD8.root')  
)
```

- data.py prikuplja originalne podatke eksperimenata za jedan Higgs kandidat i potreban mu je validacioni JSON fajl koji obezbeđuje da makro analizira kvalitetne podatke. Ako želimo da izaberemo drugi Higgs kandidat, onda je potrebno promeniti *fileNames* promenljivu.

```
# define JSON file for 2012 data  
goodJSON='/home/cms-opendata/CMSSW_5_3_32/src/Demo/DemoAnalyzer/datasets/Cert_190456-  
208686_8TeV_22Jan2013ReReco_Collisions12_JSON.txt'
```

```
myLumis = LumiList.LumiList(filename = goodJSON).getCMSSWString().split(',')
...
# to speed up, pick single example file with 1 nice 2mu2e Higgs candidate
# (9058 events)
process.source = cms.Source("PoolSource",
fileNames =
cms.untracked.vstring('root://eospublic.cern.ch/eos/opendata/cms/Run2012C/DoubleMuParked/AOD/22Jan2013-
v1/10000/F2878994-766C-E211-8693-E0CB4EA0A939.root')
)
```

U ovim fajlovima se priprema tzv. process objekat. Podešavaju se parametri, kao što je izvor podataka (process.source), i na kraju se pokreće EDAnalyzer objekat koji je u ovom slučaju:

```
process.demo = cms.EDAnalyzer('HiggsDemoAnalyzer')
```

i pokreće C++ fajl za analizu podataka.

Makro koji se pokreće je *HiggsDemoAnalyzer.cc* a to je C++ fajl koji obrađuje, analizira i pravi grafikone od podataka koji su mu prosleđeni. Na kraju od skupova podataka napravi novi ROOT fajl gde se nalaze novi podaci. Ovaj fajl počinje sa deklaracijom klase:

```
class HiggsDemoAnalyzer: public edm::EDAnalyzer {
public:
    explicit HiggsDemoAnalyzer(const edm::ParameterSet&);
    ~HiggsDemoAnalyzer();
private:
    virtual void beginJob();
    virtual void analyze(const edm::Event&, const edm::EventSetup&);
    virtual void endJob();
    bool providesGoodLumisecion(const edm::Event& iEvent);

    TH1D *h_globalmu_size;
    TH1D *h_recomu_size;
    TH1D *h_e_size;
    ...
}
```

U ovoj klasi se nalazi konstruktor, destruktork, metode i mnogo klasnih atributa. Ovi atributi su uglavnom histogramski objekti (TH1D) ili numerički tipovi. Nakon deklaracije klase sledi implementacija konstruktora i metoda. U konstruktoru su pripremanju svi histogrami koji će se nalaziti u ROOT fajlu na kraju. Inicijaliziraju se objekti i podešavaju se nazivi osa.

```
HiggsDemoAnalyzer::HiggsDemoAnalyzer(const edm::ParameterSet& iConfig) {  
    // now do what ever initialization is needed  
    edm::Service<TFileService> fs;  
    // *****  
    // book histograms and set axis labels  
    // (called once for initialization)  
    // *****  
    // Global Muon (GM) size  
    h_globalmu_size = fs->make<TH1D>("NGMuons", "GMuon size", 10, 0., 10.);  
    h_globalmu_size->GetXaxis()->SetTitle("Number of GMuons");  
    h_globalmu_size->GetYaxis()->SetTitle("Number of Events");  
    ...  
}
```

Metod koji se implementira se zove *analyze*. Ovaj metod se zove za svaki događaj koji se prosleđuje iz Python fajlova. U MC.py ima 7499 događaja a u data.py fajlu ima 9058. Na početku metode se odmah pokreće događaj koji je prosleđen. Nakon toga popunjava *Handle* objekte sa kolekcijom podataka sa određenom nazivom.

```
void HiggsDemoAnalyzer::analyze(const edm::Event& iEvent, const edm::EventSetup& iSetup) {  
    // *****  
    // here each relevant event will get analyzed  
    // *****  
    nRun = iEvent.run();  
    nEvt = (iEvent.id()).event(); // iEvent: no class named event()  
    nLumi = iEvent.luminosityBlock();  
    ...  
    edm::Handle<reco::TrackCollection> tracks;  
    iEvent.getByLabel("generalTracks", tracks);  
    edm::Handle<reco::TrackCollection> gmuons;  
    iEvent.getByLabel("globalMuons", gmuons);  
    edm::Handle<reco::MuonCollection> muons;  
    iEvent.getByLabel("muons", muons);  
    ...  
}
```

Ispod toga se inicializiraju numerički atributi i onda počinje čišćenje podataka. Pomoću for petlje prolazi se kroz kolekcije podataka i oni podaci se popunjavaju u odgovarajuće histograme. Oni podaci koji odgovaraju kriterijumima se čuvaju u paru sa svojim indeksom u *vector* klasi.

```
// Loop over muons size and select good muons  
for (unsigned u = 0; u < muons->size(); u++){
```

```
const reco::Muon &itMuon = (*muons)[u];
math::XYZPoint point(primvtx[0].position());
// select global particle flow muons
// some muons might not have valid track references
if (itMuon.isPFMuon() && itMuon.isPFIsoValid() && (itMuon.globalTrack()).isNonnull())
{
    //fill histograms
    h_p_reco->Fill(itMuon.p());
    h_pt_reco_b4->Fill(itMuon.pt());
    h_eta_reco_b4->Fill(itMuon.eta());
}
```

...

```
if (std::abs(SIP3d_mu) < 4. && std::abs((itMuon.globalTrack())->dxy(point)) < 0.5 &&
std::abs((itMuon.globalTrack())->dz(point)) < 1. && relPFIso_mu < 0.4){
    if (itMuon.pt() > 5. && std::abs(itMuon.eta()) < 2.4){
        vIdPtmu.push_back( std::make_pair(u, itMuon.pt()) );
    }
}
} // end of if (itMuon.isPFMuon())......
} // end muons loop
```

Čišćenje se izvršava za oba tipa leptona (mion, elektron). Posle čišćenja podataka se obavljaju kalkulacije podataka, sa kojima se odgovarajući histogrami dopunjuju. Kalkulacije su izvršene za sve vrste raspadanja: $Z \rightarrow 2e$, $Z \rightarrow 2\mu$, $ZZ \rightarrow 4e$, $ZZ \rightarrow 2e2\mu$ i $ZZ \rightarrow 2\mu$. Uzimaju se prve dve ili prve četiri čestice i sabiraju se njihove naelektrisanosti. U slučaju da je 0 onda se vrše dalja sabiranja impulsa čestica. U slučaju kada se uzima četiri čestice onda se gleda suma naelektrisanosti od svake moguće kombinacije para. Nakon sabiranja impulsa se vrše neke kalkulacije pa se napunjavaju podaci na grafikone. Sledeći primer je slučaj kad se Z bozon raspada u dva elektrona. Kalkulacije koje se vrše nad 4 čestice su slične, ali se gledaju još i svi mogući parovi sami u sebi.

```
//===== ZTo2Electron start
=====//
if (nGoodElectron >= 2)
{
    const reco::GsfElectron &elec1 = (*electrons)[vIdPte.at(0).first];
    const reco::GsfElectron &elec2 = (*electrons)[vIdPte.at(1).first];
    if (elec1.charge() + elec2.charge() == 0){
        for (unsigned i = 0; i < vIdPte.size(); i++){
            // These pT and eta are filled after all the cuts
            // access directly .second as the second pair is already pT
        }
    }
}
```

```

        h_pt_e_after_Zto2e->Fill(vIdPte.at(i).second);
        h_eta_e_after_Zto2e-> Fill((((*electrons)[vIdPte.at(i).first]).superCluster()->eta());
    }

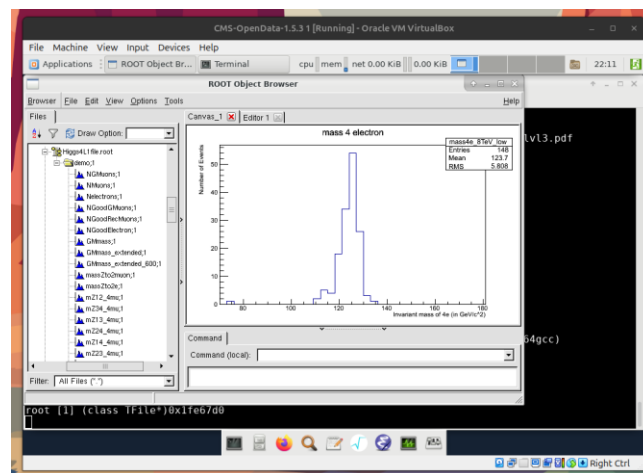
    s1 = sqrt(((elec1.p()) * (elec1.p()) + sqme) * ((elec2.p()) * (elec2.p()) + sqme));
    s2 = elec1.px() * elec2.px() + elec1.py() * elec2.py() + elec1.pz() * elec2.pz();
    s = sqrt(2.0 * (sqme + (s1 - s2)));
    h_mZ_2e->Fill(s);
}

}

//===== ZTo2Electron end =====//

```

Sa završetkom poslednje kalkulacije se završava i metoda i sa njim ceo makro.



Slika 8 - Izgled Higgs4L1file.root u ROOT Framework-u

Makro *M4Lnormdata1_lv13.cc* je zadužen da spoji sve histograme u jedan konačni. Ima isti kod kao u drugom nivou, ali se za Higs dijagrame iz 2012 godine koriste podaci iz *Higgs4L1file.root*. Druga razlika je da ima novi izvor podataka, gde je izabran samo jedan Higs kandidat. Ti podaci se učitaju u svoj histogram i kasnije se ukombinuju u celokupni histogram.

```

// Replace the input file for Higgs MC 2012
// with user's generated Higgs input file level 3
string inFileHZZ12 = "Higgs4L1file.root";

// Add the input file for data
// with user's generated data input file level 3
string inFileUser = "DoubleMuParked2012C_10000_Higgs.root";

...

////////// HISTOGRAM 1 DATA FILE FROM USER //////////

f3 = TFile::Open((inDir + inFileUser).c_str());
TH1D *DouMuser = (TH1D*) f3->Get("demo/mass2mu2e_8TeV_low")->Clone();
DouMuser->SetMarkerColor(kBlue);
DouMuser->SetMarkerStyle(22);
DouMuser->SetMarkerSize(2.0);

```

```
DouMuser->SetLineColor(kBlack);  
DouMuser->SetLineWidth(1);
```

Za četvrti nivo replikacije je predviđeno da korisnik sam generiše sve ROOT fajlove potrebne za histogram. To bi značilo više miliona događaja po svakom dataset-u. Zbog nedostatka vremena projekat nije predviđen za odrađivanje ovog zadatka.

ZAKLJUČAK

Projekat je uspešno pokazao proces repliciranja rezultata otkrića Higsovog bozona. Korišćenjem CERN Open Data platforme i alata kao što su ROOT Framework i CMSSW, omogućena nam je analiza velikih skupova podataka. Replikacija je izvršena u tri nivoa, koji su pokazali detaljni proces reprodukcije rezultata.

Prvi nivo je omogućio poređenje dijagrama originalnih rezultata sa rekonstruisanim dijagramima, dok je drugi nivo omogućio reprodukciju dijagrama pomoću ROOT Framework-a u virtuelnoj mašini. Treći nivo replikacije omogućio je generisanje ROOT fajlova za jednog Higs kandidata i kreiranje dijagrama koji sadrži obeleženog Higs kandidata.

Kroz analizu Python i C++ fajlova koji se koriste za obradu podataka, objašnjeni su procesi čišćenja podataka, selekcije događaja i skaliranja histograma. Dobijamo uvid u to kako može da bude komplikovan rad sa česticama.

Zbog vremenskih ograničenja, četvrti nivo replikacije, koji uključuje analizu celokupnog dataset-a, nije odrađen. Ali to bi bilo odlično za budući projekat, jer obuhvata generisanje svojih ROOT fajlova i generisanje preciznijih rezultata.

LITERATURA

1. Brun R., Rademakers F., *ROOT—An object oriented data analysis framework*. Nuclear instruments and methods in physics research section A: accelerators, spectrometers, detectors and associated equipment, 389(1-2), pp.81-86., 1997., [https://doi.org/10.1016/S0168-9002\(97\)00048-X](https://doi.org/10.1016/S0168-9002(97)00048-X)
2. Chatrchyan S., et al., *Observation of a New Boson at a Mass of 125 GeV with the CMS Experiment at the LHC*, [CMS], Phys. Lett. B 716 (2012), 30-61 [doi:10.1016/j.physletb.2012.08.021](https://doi.org/10.1016/j.physletb.2012.08.021) [arXiv:1207.7235 [hep-ex]].
3. Close F., *Particle physics: A very short introduction*, Oxford University Press, New York, USA, 2023.
4. Griffiths D., *Introduction to Elementary Particles*, John Wiley & Sons, Portland, Oregon, USA, 2008.
5. Wiese U. J., *The Standard Model of Particle Physics*, University of Bern, 2018.

LINKOVI

- <https://github.com/cms-opendata-analyses/HiggsExample20112012/tree/master?tab=readme-ov-file>
- <https://home.cern/resources/faqs/facts-and-figures-about-lhc>
- <https://home.cern/science/experiments/cms>
- <https://opendata.cern.ch/>
- <https://opendata.cern.ch/docs/cms-virtual-machine-2011>
- <https://opendata.cern.ch/record/252>
- <https://opendata.cern.ch/record/5210>
- <https://opendata.web.cern.ch/record/5500>
- https://pdg.lbl.gov/2024/listings/contents_listings.html
- <https://root.cern/>