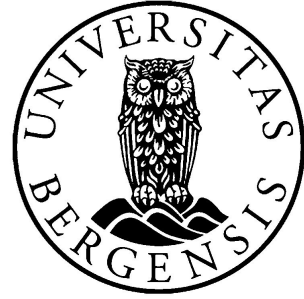

PHYS117



Klassifikasjon av sorte hull og sphaleron hendelser produsert i LHC kollisjoner med CMS detektoren

Navn (alle i lab.gruppen): Sindre K. Nesheim, Markus Cirotzki

Epost (alle i lab.gruppen): qes014@uib.no, mci008@uib.no

Veileder Nikolai Fomin, Anna Lipniacka

Institutt for fysikk og teknologi
Det matematiske-naturvitenskapelige fakultet

Abstrakt

Ved hjelp av Monte Carlo hendelsesgeneratorer og DELPHES rammeverket så kan man lage simulasjoner av detektorresponser fra CMS-detektoren på LHC, slik at man kan produsere datasett basert på ulike parametere. [1, 2, 3] På den måten vil man kunne simulere hvordan det kanskje faktisk ville sett ut dersom det var ekte hendelser med disse parameterene, og man kan analysere datasettene slik at man får verdifull innsikt i fysikken bak når man faktisk gjør eksperimentet. Noen av disse simulasjonene har blitt gjort med hypoteser som omhandler svarte hull (BH) og Sphaleroner (SP). Ved hjelp av datasettene nevnt ovenfor, maskinlæring og en kuttebasert metode har vi prøvd å klassifisere slike hendelser, og klart å skille vanlige standardmodell prosesser, BH, og SP med en nøyaktighet opp mot 90 %.

Innhold

1	Introduksjon	4
2	Problem/Bakgrunn	4
3	Prosedyre	7
3.1	Interessante variabler å se på	7
3.2	Manuell klassifisering av hendelser	10
3.3	Klassifisering av hendelser ved hjelp av maskinlæring	12
4	Resultater	14
5	Diskusjon	15
6	Konklusjon	17

1 Introduksjon

Det å analysere og gjøre beregninger av datasett som kommer fra proton-proton kollisjoner ved Large Hadron Collider, LHC, ved CERN og blir detektert av CMS-detektoren er fullt mulig å gjøre for hånd. Men er dette effektivt, og fins det en bedre metode for å gå gjennom enorme mengder data, som også er mer nøyaktig? Dette høres ut som en jobb for maskiner, og vi har derfor brukt nettopp maskinlæring for å analysere datasettene, nærmere bestemt maskinlæringsbiblioteket XGBoost [4].

I dette prosjektet har vi brukt simulerte datasett basert på teorier som prøver å løse hierarki problemet. Disse modellene forutser at det finnes mikroskopiske svarte hull og at det er mulig å lage disse ved proton-proton kollisjoner ved CERN, nærmere bestemt “Large Hadron Collider, LHC” [5]. Teorien forutser også noe kalt sphaleron, som er en partikkel-lignende løsning til den elektrosvake feltteorien [6].

Disse fenomenene nevnt ovenfor har ganske like kjennetegn [7]. Vi skal ved hjelp av datasettene finne disse kjennetegnene, og prøve å skille de, slik at om teorien faktisk stemmer og vi klarer å detektere disse fenomenene i CMS-detektoren på LHC, så kan vi også skille de. Dette skal vi gjøre ved hjelp av analyse av ulike variabler som vi får fra datasettene. Vi skal også bruke XGBoost, som er et maskinlærings bibliotek basert på beslutningstre, som vi “mater” med datasettene og trener det opp til å klare å skille mellom fenomenene.

2 Problem/Bakgrunn

Standardmodellen beskriver de tre naturkreftene fargekraft, svak kjernekraft og elektromagnetisme godt, men sliter med å forklare hvorfor gravitasjon er så mye svakere enn de andre naturkreftene. Denne store forskjellen i styrke mellom naturkreftene er det vi kaller hierarki problemet. Standardmodellen sliter også med å forklare hvorfor vi har en skeivfordeling av vanlig materie og antimaterie i universet, også kalt materie antimaterie assymetri. Løsninger til disse problemene har blitt foreslått, og blir kalt “Physics beyond the Standard Model (BSM)”.

Forklaringer på materie-antimaterie assymetrien i universet kan være at det finst prosesser som bryter lepton-og baryon bevaringslovene. Sphaleroner kan være en slik prosess som bryter bevaringslovene, der den endrer baryon- og leptonnummere, men holder endringen mellom de lik null: $\Delta(B - L) = 0$ [6, 8, 9].

Noen løsninger til hierarki problemet foreslår eksistensen av flere mindre romlige dimensjoner hvor kun tyngdekraften kan propagere. Det store volumet av ekstra

rom vil dempe den fundamentalt kraftige tyngdekraften og resulterer i den tilsynelatende svake tyngdekraften som vi kjenner til. En konsekvens av dette er altså mikroskopiske svarte hull, og en oppdagelse av dette vil kunne forklare hierarki problemet [10].

Disse to fenomenene er spekulert i å kunne bli produsert i proton-proton kollisjoner i LHC, med massesenterenergier opp mot $\sqrt{s} = 13$ TeV. Dersom de blir produsert vil de forfalle til en tilstand med høy multiplisitet av objekter, og vil derfor være vanskelige å skille [5]. Det er dette vi skal prøve å gjøre i dette prosjektet, ved hjelp av analyse av datasettene.

Datasettene vi har brukt, er vist i tabellene under:

Nr	Datasett	Kallenavn
1	ttbar.lhco	ttbar
2	ttbar_largejets.lhco	ttbar_lj

Tabell 1: ttbar datasett, hvor ttbar_lj har minst en jet med $p_T \geq 200$ GeV

Nr	Datasett	Kallenavn
1	PP13-Sphaleron-THR9-FRZ15-NB0-NSUBPALL.lhco	SP_1
2	PP13-Sphaleron-THR9-FRZ15-NB33-60-NSUBP50.lhco	SP_2
3	PP13-Sphaleron-THR9-FRZ15-NB33-71-NSUBP5.lhco	SP_3

Tabell 2: Sphaleron datasett hvor $\sqrt{s} = 13$ GeV.

Nr	Datasett	Kallenavn
1	BH_n4_M8.lhco	BH_n4_M8
2	BH_n4_M9.lhco	BH_n4_M9
3	BH_n4_M10.lhco	BH_n4_M10
4	BH_n4_M11.lhco	BH_n4_M11
5	BH_n4_M12.lhco	BH_n4_M12
6	BH_n5_M8.lhco	BH_n5_M8
7	BH_n5_M9.lhco	BH_n5_M9
8	BH_n5_M10.lhco	BH_n5_M10
9	BH_n5_M11.lhco	BH_n5_M11
10	BH_n5_M12.lhco	BH_n5_M12

Tabell 3: Svarte hull datasett. Tallet etter n står for antall ekstra dimensjoner og tallet etter M står for minimum massen til det svarte hullet i TeV.

Object	eta	phi	PT	jmass	ntrk	btag	hadem
photon	-0.268	2.684	22.47	0.0	0.0	0.0	0.0
jet	-1.143	-0.084	2359.53	164.5	33.0	0.0	0.01
jet	1.166	-2.964	1357.33	46.57	18.0	0.0	0.01
jet	0.253	2.126	966.97	20.59	9.0	0.0	0.04
jet	-0.166	-2.336	204.02	15.65	12.0	1.0	0.99
jet	-0.6	-2.05	147.13	14.61	17.0	0.0	0.33
jet	1.143	-2.139	66.13	9.31	9.0	0.0	0.0
jet	-0.488	-0.284	31.74	4.68	1.0	0.0	0.09
jet	2.828	2.049	26.18	6.2	1.0	0.0	2.65
MET	0.0	-2.784	255.79	0.0	0.0	0.0	0.0

Figur 1: Typisk hendelse fra BH fil. Det er også slik hendelsene i de andre filene er formattert.

I figur 1 ser vi en typisk event eller hendelse ifra data settet. Denne hendelsen kommer fra LHCO fil format og er plottet gjennom LHCO_reader python filen. LHCO_reader er orginalt skrevet på python 2, men ved bruk av pythons innebygde funksjon, 2to3, fikk vi konverteret den over til python 3 format som vi kjører på våres komputere.

En fil med data inneholder alt ifra 1000 til 100 000 hendelser alt etter hvilke fil som blir valgt. Hver hendelse inneholder et visst antall objekt som blir representert i hver sin rad, den siste raden i hver hendelse er MET, som står for “missing transverse momentum”, og er ikke et eget objekt. MET er den negative summen av all energien som står vinkelrett på protonstrålen. Summen av MET og all energien skal dermed teoretisk sett være lik 0, grunnet bevaring av energi og er dermed veldig interessant. Videre har vi 5 ulike objekt vi kan få. Disse er foton, jet, elektron, muon og tau.

En jet er signaturer av kvarker og gluoner som blir produsert i proton-proton kollisjonene. Fordi kvarker og gluoner ikke kan eksistere fritt, så kommer dei som konsentrerte stråler av hadroner, altså jets [11].

Muoner og tau er leptoner som ligner elektronet, og har lik ladning og spin men ulik masse. Tauene har mye større masse enn muonet som igjen har mye større masse enn elektronet.

De neste kolonnene forteller oss noe om egenskapene til de ulike objektene. Eta, også kalt pseudorapidity, er relatert til vinkelen θ i forhold til protonstråleaksen ved $\eta = \ln \tan(\theta/2)$.

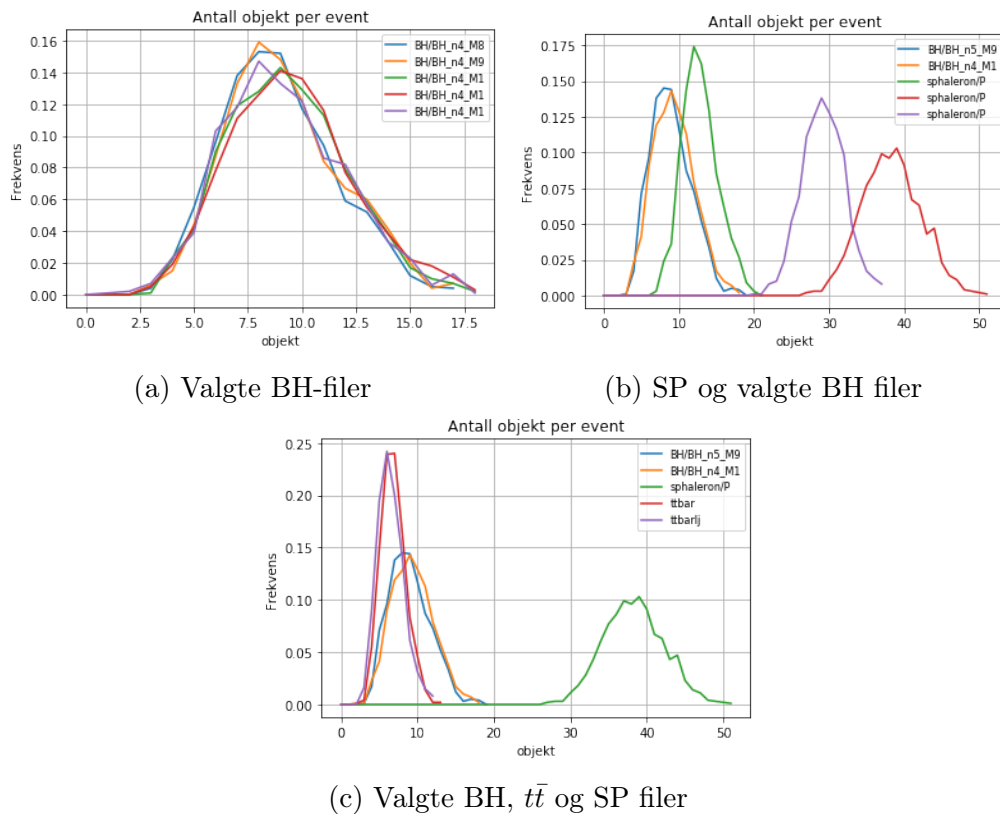
Phi er vinkelen rundt protonstrålen, slik at med eta og phi kan vi finne ut hvor et hvert objekt ble fanget opp. PT er momentet i ortogonal retning på protonstrålen, som vil si at summen av all PT-en skal bli MET. Dette er de mest interessante variablene å ta i bruk når vi skal prøve å skille svarte hull og sphaleron.

Variablene j_{mass} , n_{trk} , b_{tag} og h_{adem} har vi ikke brukt og er ikke så interessante for oss. Disse sier noe om henholdsvis den invariante massen til objektene, antall spor assosiert med objektene, om en jet har b-kvark, og forholdet mellom hadronisk og elektromagnetisk energi som er avgitt i kalorimetercellene i detektoren assosiert med objektene.

3 Prosedyre

3.1 Interessante variabler å se på

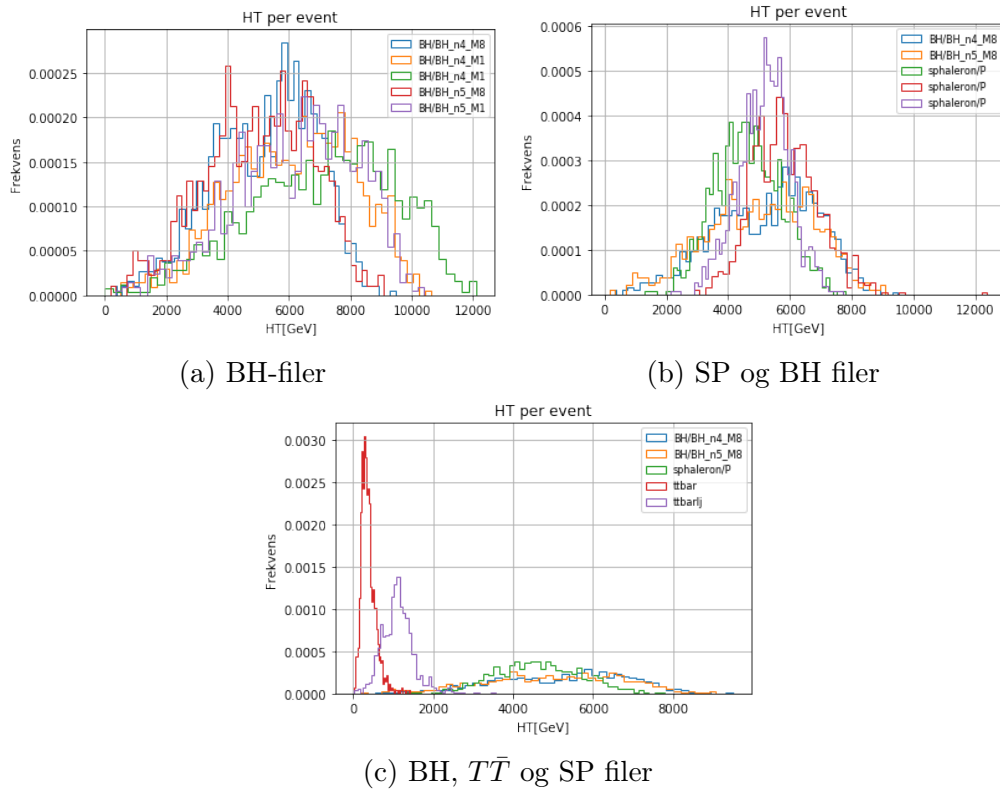
Vi finner antall objekt per hendelse ved bruk av `LHCO_reader` til å lese inn gitte filer og deretter teller hvor mange objekter det er for hver hendelse for alle hendelsene i filen. Vi plotter dette for en rekke filer som er vist i tabell 2a, 2b og 2c.



Figur 2: Antall objekt per hendelse for gitt filer

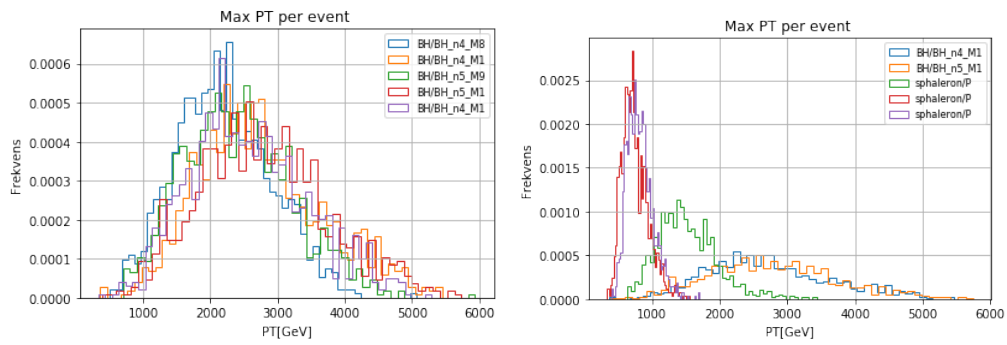
Videre ønsker vi å plotte HT for de samme filene som vi plottet for antall objekt.

Vi bruker igjen LHCO_reader for å plotte HT for alle hendelser og velger ut de filene som vi vil plotte. I dette tilfelle har vi lagret HT i bins for å gjøre det lettere å plotte, plottene er vist i figur 3.



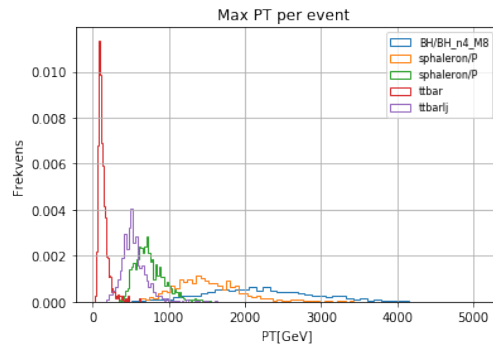
Figur 3: Viser HT per hendelse for gitte filer

Max PT er også en interessant variabel å se på. Max PT er det objektet i en hendelse som har størst PT. Vi plottes dette på samme måte som HT, ved bruk av LHCO_reader filen og bins. Plottene blir vist i figur 4.



(a) Max PT for BH filer

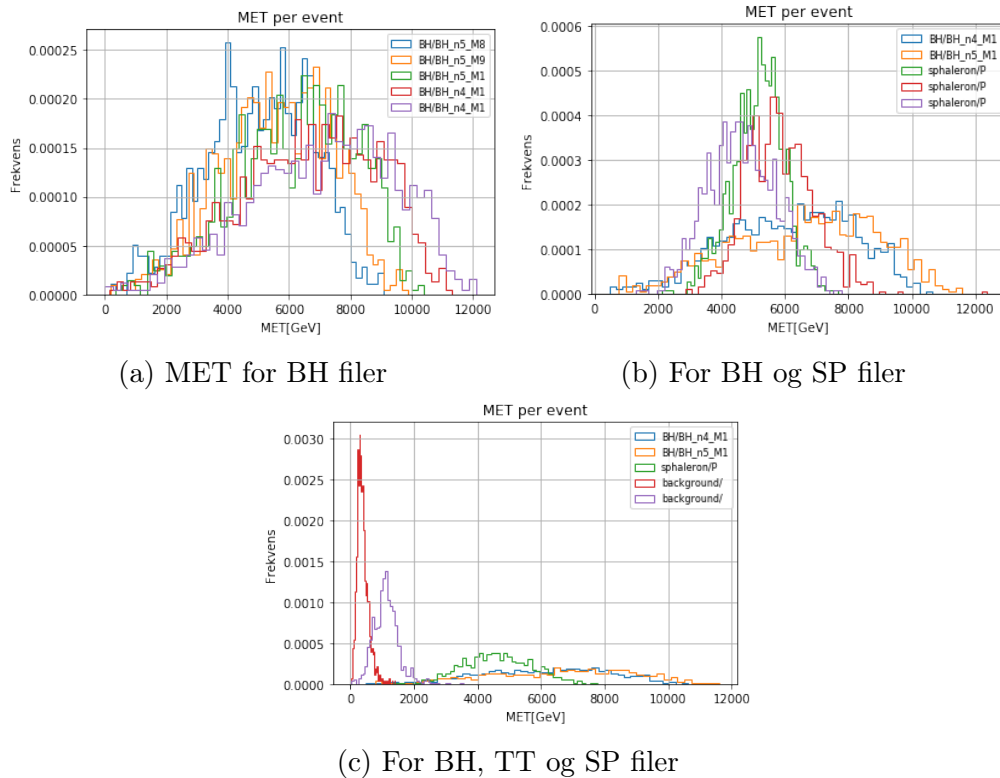
(b) For BH og SP filer



(c) For BH, TT og SP filer

Figur 4

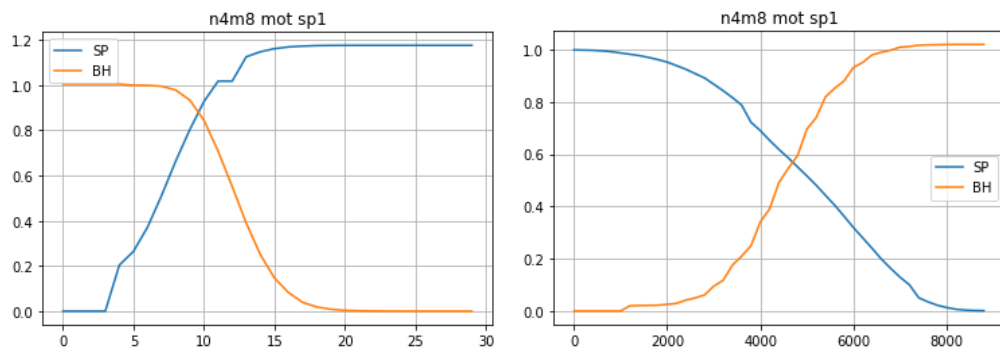
MET er også en variabel som kan være interessant å se på, vi plottet den på samme måte som HT og max PT, plottene er vist i figur 5.



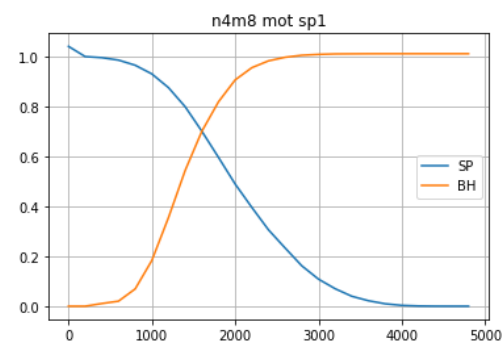
Figur 5

3.2 Manuell klassifisering av hendelser

For å finne gode skiller til å separere BH-filer og SP-filer tester vi effektiviteten av valgte variabler. Vi gjør dette ved bruk av Python. Her tester vi hvor mange av hendelsene til venstre og til høyre for variablene er BH og SP filer. I grafene som vi kommer frem til vil enten BH eller SP være til venstre eller høyre for oppgitt verdi. Dersom BH er til venstre vil det si at antall BH hendelser til venstre for denne verdien delt på antall SP hendelser til venstre for denne verdien. Det vil da være tilsvarende for SP filer til høyre for grafen. Grafene som ble plottet er vist i figur 6. Her har vi valgt de to SP og BH filene som vil være vanskeligst å separere. Det er dette vil vil bruke når vi videre skal finne de beste kuttene for det manuelle treet vårt. I plottene i figur 6 har vi brukt de tre variablene vi ser på som best for å finne gode kutt mellom SP og BH for de gitte filene. Ger har altså valgt å bruke antall hendelser, HT og det objektet med høyest PT for hver hendelse.



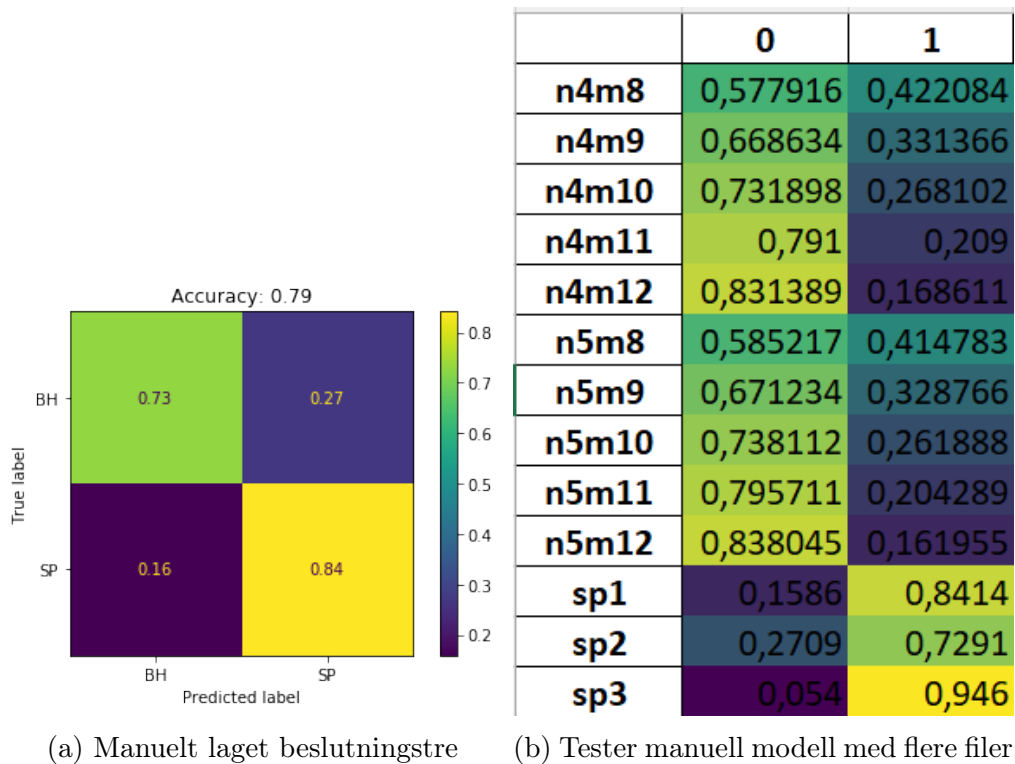
(a) BH til venstre og SP til høyre for (b) BH til høyre og SP til venstre for
antall hendelser HT



(c) BH til høyre og SP til venstre for
max PT

Figur 6: Effektivitet av kutt for gitte variabler og formler for BH fil n4m8 og sphaleron fil 1. x-aksen på disse grafene er energi [GeV].

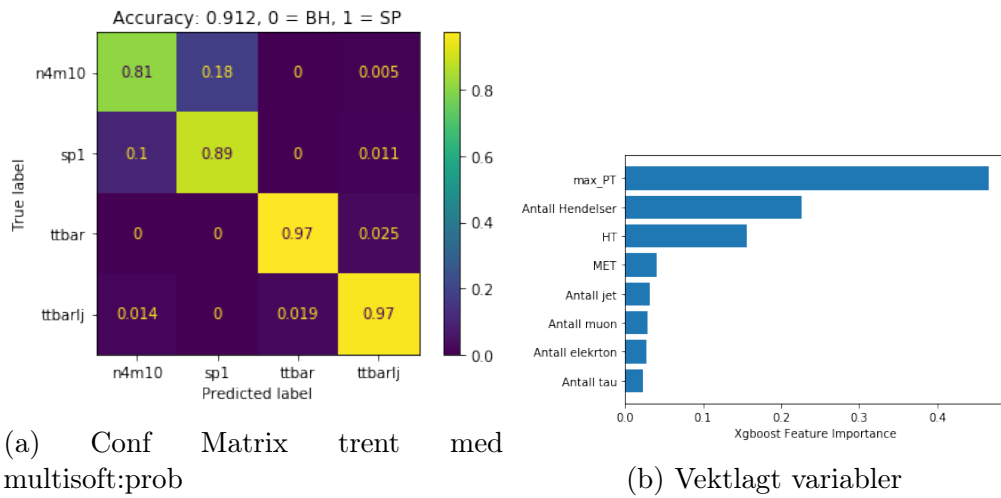
Videre har vi brukt plottene fra figur 6 til å lage vårt manuelle beslutningstre. Dette ble gjort i python og vi testet effektiviteten av dette ved å bruke den på alle BH filer med $n = 4$ og $n = 5$, i tillegg til alle SP filene. Under i figur 7a ligger de testede filene for n4m10 og SP1 filen, som er de filene denne modellen er basert på. Ved siden av i figur 7b ligger de resterende testene, det vil si samme modell testet på resterende filer.



Figur 7: Testing av manuelt laget beslutningstre

3.3 Klassifisering av hendelser ved hjelp av maskinl ring

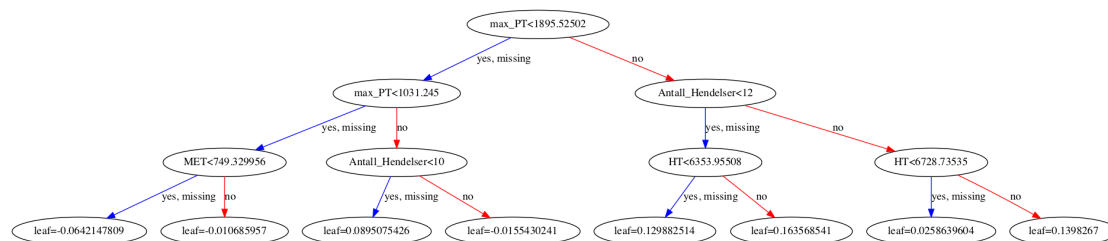
Vi brukte n  maskinl ring til   lage en ny modell som skal kunne separere BH, SP, ttbar og ttbarlj. Her har vi gitt XGBoost 4 filer med informasjon, de 4 filene representerer BH, SP, ttbar og ttbarlj filene. Informasjonen som er gitt til XGBoost er antall hendelser, MET, HT, det objektet med mest PT i en hendelse, antall jets, antall muon, antall tau, antall elektron og til slutt en true label som blir separert inni python og er med p  b de   trene modellen og teste den senere. N r vi trente modellen valgte vi de samme BH og SP filene som vi gjorde ved den manuelle modellen, alts  BH filen n4m8 og den f rste SP filen, de som var vanskeligst   separere. Vi velger i tillegg ttbar og ttbarlj til   trene og teste p  for   separere de ogs . Om vi hadde valgt SP2 eller SP3 ville XGBoost bare laget et kutt p  antall hendesler siden det er en veldig tydelig forskjell mellom SP2 og SP3 sammenlignet med BH filene. Dette ser vi tydelig i plottet vist i figur 2b. Vi tar n  de valgte filene, trener p  80% av filene og tester p  de resterende 20%. Resultatet av denne testen er vist i figur 8a, med de mest vektlagte variablene vist ved siden av i figur 8b. Vi tester deretter denne modellen p  de samme filene som vi testet med den manuelle modellen og resultatet av denne testen er gitt i figur 9.



	0	1	2	3
n4m8	0,789234	0,182488	0	0,028278
n4m9	0,808147	0,175396	0	0,016458
n4m10	0,832863	0,155667	0	0,01147
n4m11	0,860971	0,132352	0,000183	0,006494
n4m12	0,890775	0,103471	9,43E-05	0,005659
n5m8	0,797327	0,170304	0,000455	0,031915
n5m9	0,815512	0,165576	0,000182	0,018731
n5m10	0,846392	0,139412	0,000364	0,013832
n5m11	0,871021	0,119923	0,000274	0,008782
n5m12	0,900441	0,093178	9,38E-05	0,006287
sp1	0,093	0,905182	0	0,001818
sp2	0,019545	0,980455	0	0
sp3	0,037	0,963	0	0
ttbar	4,95E-05	3,96E-05	0,96999	0,029921
ttbarlj	0,006069	0,000822	0,033406	0,959703

Figur 9: Tester XGBoost model med flere filer

Selve modellen som XGBoost produserte er gitt under i figur 10.



Figur 10: Modellen produsert av XGBoost

4 Resultater

Vi ser i figur 2a at BH filene har relativt lik fordeling og det er ingen filer som skiller seg spesielt ut. I figur 2b ser vi at SP-filene er ganske ulike og spesielt den første SP-filen skiller seg godt ut fra de to andre SP-filene. Den er ikke enkel å skille ifra BH filene da det er stort overlapp mellom BH filene og den første SP-filen. Når vi ser på det siste plottet, figur 2c, ser vi at ttbar og ttbarlj er veldig like hverandre, men også ikke så lett å skille ifra BH filene.

Vi ser at BH filene er noe like med små variasjoner fra fil til fil, men ikke noe spesielt å merke seg som vist i figur 3a. Når vi ser på SP filer sammen med gitte BH filer, som vist i figur 3b, ser vi at det ikke er mye forskjell, men SP filene får en større topp rundt 5000 GeV. Vi ser på figur 3c at ttbar og ttbarlj er noe ulike hverandre og kan ha et fint kutt rundt 700 GeV. De skiller seg også godt ifra både SP og BH filene med et kutt rundt 2000 GeV.

Dersom vi ser på figur 4a som plotter gitte BH filer, ser vi at det ikke er mye forskjell på de ulike BH filene. I figur 4b ser vi 3 SP filer og 2 Bh filer. Her ser vi at det er 2 SP filer som overlapper veldig, mens den første SP-filen skiller seg ut bemerkelsesverdig. Vi ser igjen, som i plottene for antall hendelser, at det er lett å skille de to siste SP-filene fra BH-filene, men den første SP-filen er noe vanskeligere å finne et bra kutt til. I den siste figuren, figur 4c, ser vi ganske stor forskjell på ttbar, ttbarlj og BH pluss SP-filene. Det finnes et nokså godt kutt mellom SP og ttbarlj, og det er enda enklere å skille mellom ttbarlj og BH så dette kan vise seg å være en god variabel når vi skal separere de ulike filene senere.

I figur 5 ser vi at MET er noe lik HT når det kommer til forskjeller mellom BH, SP og ttbar-filene og denne variabelen kunne bli brukt til å separere filene.

Utifra plottene som er vist i del 3.2 bestemte vi oss for å basere modellen vår på fil n4m8 og SP1 som nevnt tidligere. Dette er basert på plottet i figur 2b hvor det er tydelig at vi kunne valgt et enkelt kutt på rundt 20 objekt per hendelse for å kutte mellom BH filene og SP2/3. Dermed er ikke disse filene like interessante når vi skal finne en modell som kan separere BH vs SP. Om vi går videre og ser på HT per hendelse ser vi som nevnt tidligere mye mer overlapp, så her kan det være noe vanskeligere å finne noen gode kutt for å separere BH og SP. Max PT er den med de tydeligste forskjellene mellom SP1 og BH filen n4m10 så vi velger å vektlegge denne variabelen mest, etterfulgt av antall hendelser og deretter HT. Når vi bruker plottene i figur 6 finner vi et godt kutt for max PT rundt 1500 GeV, for antall hendelser kan vi kutte på rett under 10 objekt og HT har et nokså greit kutt på rundt 4500 GeV. Med dette i grunnlaget for modellen endret vi litt på verdiene og prøvde oss litt frem til vi fant noen gode kutt som ga oss resultatet som er gitt i figur 7.

Vi ser på modellen som XGBoost ga oss og ser at den har vektlagt mange av de samme variablene som vi valgte i vår modell. Vi ga XGBoost en del flere variabler og vi ser at den har tatt de i bruk, men de var ikke like viktige som max PT, HT og antall objekter per hendelse i denne modellen heller. Dette ser vi i figur 8b og ved siden av i figur 8a ser vi hvordan XGBoost gjorde det på filene den trente på, det vil si de resterende 20%. Vi testet 2 forskjellige "Classifiers", en med "Multisoft:prob" og en med Binary". Den beste treffsikkerheten kom med "Multisoft:prob" og det er dermed denne vi har valgt å bruke i denne oppgaven.

5 Diskusjon

I fra figur 7a ser vi at det er lettere å finne SP enn det er å finne BH i den håndlagde kuttbaserte modellen. Det samme viser figur 7b, her ser vi også ganske store forskjeller mellom de ulike BH filene, mens SP filene er mye likere. BH filene går mellom 58% og 84% mens SP filene holder seg imellom 73% og 94%. Dette er da ikke en veldig god modell for å separere BH mot SP. Grunner til dette kan være noe unøyaktige kutt for valgte variabler og/eller for få valgte variabler. Vi prøvde å legge til noen flere variabler, men dette førte bare til dårligere treffsikkerhet eller nesten ingen forskjell. Vi kunne prøvd andre BH filer enn n4m10, men det ble dessverre ikke tid til dette. MET var en variabel som godt kunne blitt brukt til separasjon, men vi valgte å kun bruke 3 variabler for å holde beslutningstreet noe enklere. Når vi forsøkte med 4 variabler ble ikke treffsikkerheten noe bedre, som kan tilsi at vi testet for få kutt. Det er også noe å legge merke til at XGBoost heller ikke tok god nytte av MET.

Når det kommer til modellen ifra XGBoost kan vi utifra figur 8a se at den klarte å finne ttbar og ttbarlj veldig lett og vi får veldig gode resultater derifra. Dette gir god mening siden vi kan se ifra grafen i figur 4c at det er noen gode kutt mellom ttbar, ttbarlj og de resterende filene. Videre ser vi at SP1 filen er noen lettere å separere enn n4m10, som er det samme resultatet som vi fikk i vår modell, altså lettere å finne SP enn BH. Vi bommer altså mye sjeldnere når det er en SP enn når det er snakk om et BH. Treffsikkerheten er mye bedre i modellen ifra XGBoost enn den vi lagde for hånd, dette er nok fordi XGBoost tar med flere variabler som kan øke treffsikkerheten. Den har også testet flere kutt enn hva vi har, som skal gi et bedre resultat. Når vi ser på figur 9 ser vi at XGBoost sin modell er mye bedre på å konsekvent finne BH enn vår modell. Her er den dårligste treffsikkerheten 79% på n4m8 og den beste 90% på n5m12, som er mye mindre forskjell enn hva vi fikk, så denne modellen er en del bedre. Når vi ser på modellen som XGBoost har, som er vist i figur 10, ser vi at XGBoost har valgt mange kutt som er nokså lik som de vi har valgt. I figur 8b ser vi at XGBoost ikke har tatt for mye hensyn på MET selv om plottene i figur 5 kan tilsi at det finnes noen bra kutt her. Dette kan ha hatt en påvirkning på treffsikkerheten av modellen. Det er ikke mye problemer med overtilpassning siden vi ikke har best treffsikkerhet på den filen vi trente på. Vi prøvde ikke andre SP filer enn SP1 siden vi da bare hadde fått et kutt på rundt 20 objekt per hendelser som vi tydelig ser i figur 2b, dette hadde i tilfelle vært et godt eksempel på overtilpassning da det hadde passet bra på SP2 og SP3, men ikke for SP1.

Dersom vi sammenligner våre resultater med tidligere resultater ser vi noen forskjeller og noen likheter. Vi sammenligner med Jonas og Håvards oppgave i 2021 som fikk de samme filene. Dersom vi sammenligner treffsikkerheten til våres modell ifra XGBoost med modellen de fikk ser vi at treffsikkerheten når det kommer til BH hendelser er ganske likt. De hadde de fleste filer rundt 82% effektivitet med en fil som er ganske mye mindre på 64%. Vi hadde mye det samme, men en gjennomsnittlig treffsikkerhet som var bedre. Vår treffsikkerhet på SP filer var merkbart bedre, hvor de hadde på 83% mens vi ligger mellom 90 og 98% over alle 3 SP filene. For ttbar og ttbarlj har vi også fått mye bedre resultat, hvor vi ligger på 97 og 96%, mens de hadde på 91 og 89%. Det er derimot viktig å merke seg at de brukte en annen metode til å separere SP og BH filene enn den metoden vi har brukt her.

6 Konklusjon

Vi kan konkludere med at XGBoost lagde en klart bedre modell enn hva vi gjorde for hånd. Den modellen XGBoost kom med er også klart bedre enn tidligere modeller laget, her sammenlignet vi med en gruppe som hadde tilsvarende oppgave i 2021.

Referanser

- [1] Andreas Papaefstathiou, Simon Plätzer og Kazuki Sakurai. “On the phenomenology of sphaleron-induced processes at the LHC and beyond”. I: *Journal of High Energy Physics* 2019.12 (2019). DOI: 10.1007/jhep12(2019)017. URL: <https://doi.org/10.1007%2Fjhep12%282019%29017>.
- [2] Michael H. Seymour og Marilyn Marx. *Monte Carlo Event Generators*. 2013. DOI: 10.48550/ARXIV.1304.6677. URL: <https://arxiv.org/abs/1304.6677>.
- [3] Gregor Kasieczka mfl. “The LHC Olympics 2020 a community challenge for anomaly detection in high energy physics”. I: *Reports on Progress in Physics* 84.12 (2021), s. 124201. DOI: 10.1088/1361-6633/ac36b9. URL: <https://doi.org/10.1088%2F1361-6633%2Fac36b9>.
- [4] XGBoost utviklere. *XGBoost documentation*. URL: <https://xgboost.readthedocs.io/en/stable/>. (accessed: 28.11.2022).
- [5] Lisa Randall og Raman Sundrum. “Large Mass Hierarchy from a Small Extra Dimension”. I: *Phys. Rev. Lett.* 83 (17 1999), s. 3370–3373. DOI: 10.1103/PhysRevLett.83.3370. URL: <https://link.aps.org/doi/10.1103/PhysRevLett.83.3370>.
- [6] A. Ringwald. “High energy breakdown of perturbation theory in the electroweak instanton sector”. I: *Nuclear Physics B* 330.1 (1990), s. 1–18. ISSN: 0550-3213. DOI: [https://doi.org/10.1016/0550-3213\(90\)90300-3](https://doi.org/10.1016/0550-3213(90)90300-3). URL: <https://www.sciencedirect.com/science/article/pii/0550321390903003>.
- [7] A.M. Sirunyan mfl. “Search for black holes and other new phenomena in high-multiplicity final states in proton–proton collisions at s=13TeV”. I: *Physics Letters B* 774 (2017), s. 279–307. ISSN: 0370-2693. DOI: <https://doi.org/10.1016/j.physletb.2017.09.053>. URL: <https://www.sciencedirect.com/science/article/pii/S0370269317307657>.

- [8] Nima Arkani-Hamed, Savas Dimopoulos og Gia Dvali. “The hierarchy problem and new dimensions at a millimeter”. I: *Physics Letters B* 429.3 (1998), s. 263–272. ISSN: 0370-2693. DOI: [https://doi.org/10.1016/S0370-2693\(98\)00466-3](https://doi.org/10.1016/S0370-2693(98)00466-3). URL: <https://www.sciencedirect.com/science/article/pii/S0370269398004663>.
- [9] Vo Quoc Phong mfl. “Sphaleron in the first-order electroweak phase transition with the dimension-six Higgs field operator”. I: *Physical Review D* 101.11 (2020). DOI: 10.1103/physrevd.101.116010. URL: <https://doi.org/10.1103/PhysRevD.101.116010>.
- [10] Nima Arkani-Hamed, Savas Dimopoulos og John March-Russell. “Stabilization of submillimeter dimensions: The new guise of the hierarchy problem”. I: *Phys. Rev. D* 63 (6 2001), s. 064020. DOI: 10.1103/PhysRevD.63.064020. URL: <https://link.aps.org/doi/10.1103/PhysRevD.63.064020>.
- [11] Cern. *JETS AT CMS AND THE DETERMINATION OF THEIR ENERGY SCALE*. URL: <https://cms.cern/news/jets-cms-and-determination-their-energy-scale>. (accessed: 30.11.2022).