

Uporaba strojnega učenja za iskanje nove fizike na LHC

Simon Bukovšek

Mentor: prof. dr. Borut Paul Kerševan

Seminar, 3. letnik

Oddelek za fiziko, FMF, UL, 2023/2024

Povzetek

Strojno učenje je vedno pomembnejše orodje v fiziki visokih energij. V tem seminarju je predstavljeno delovanje detektorja ATLAS na Velikem hadronskem trkalniku, osnovni koncepti strojnega učenja in dve vrsti uporabe nevronske mreže za odkrivanje nove fizike iz meritev eksperimenta ATLAS. Prva je uporaba klasifikatorjev za ločevanje signala od ozadja, druga pa uporaba avtoenkoderjev za iskanje morebitnih odstopanj od Standardnega modela.

Kazalo

1	Uvod	1
2	Teoretično ozadje	1
2.1	Nova fizika	2
3	LHC in ATLAS	2
3.1	Sestava detektorja	2
3.2	Simulacije trkov	3
3.3	Signal in ozadje	4
4	Strojno učenje	4
5	Nevronske mreže	5
5.1	Učenje nevronske mreže	6
5.2	Mere za uspešnost modela	6
5.3	Uporaba nevronske mreže v fiziki delcev	7
6	Avtoenkoderji	10
6.1	Uporaba avtoenkoderjev za iskanje nove fizike	10
7	Zaključek	12

1 Uvod

Dandanes je v eksperimentalni fiziki osnovnih delcev eden od glavnih načinov odkrivanja nove fizike trkanje delcev v pospeševalnikih in opazovanje nastalih produktov. Večina nastalih težkih delcev razpade na manjše, še preden bi jih lahko zaznali z detektorji. Pogosto se zgodi kaskada zaporednih razpadov na več manjših produktov. Naloga fizikov je, da iz gibalnih količin in smeri končnih delcev – elektronov, mionov, protonov, nevtronov, nevtrinov in fotonov – sklepaajo o osnovnih produktih trka [1]. To je podobno, kot opazovati valove na obali in iz tega sklepati, koliko in kako težke kamne je nekdo vrgel na sredino ribnika. K sreči nam pomagata dve dejstvi: trke lahko z izjemno natančnostjo simuliramo, kar nam da veliko količino podatkov za primerjavo, obenem pa je samo število trkov zelo veliko, kar nam da dobro statistično moč za sklepanje [2, 3]. Obe dejstvi pa tudi namigujeta, da bi za analizo lahko uporabili strojno učenje. Za to namreč potrebujemo veliko testnih podatkov, naučeni modeli pa so vsaj v teoriji lahko sposobni prepoznave vzorcev, ki jih ljudje ne moremo [4]. V tem seminarju si bomo ogledali, kako pridobimo podatke iz pospeševalnika, kako delujejo nevronske mreže in kako jih lahko uporabimo na pridobljenih podatkih za iskanje nove fizike.

2 Teoretično ozadje

Fiziko osnovnih delcev in njihovih interakcij opisuje *Standardni model* (SM) [5]. Ta model je zelo uspešen, saj z veliko natančnostjo napove rezultate mnogih eksperimentov, ki jih z današnjo opremo uspejo izvesti. Model opisuje vrsto osnovnih delcev, ki med seboj interagirajo preko treh osnovnih interakcij – elektromagnetne, šibke in močne. Posredniki teh interakcij so posebne vrste delci, tako imenovani *umeritveni bozoni*. Standardni model opiše vsako vrsto delec kot eksitacijo v svojem polju, njihove interakcije pa kot sklopitve med temi polji. Delce lahko razdelimo med kvarke, leptone in bozone, kvarke in leptone pa še dodatno delimo na tri generacije. Razvrstitev osnovnih delcev po Standardnem modelu je prikazana na Sliki 1.

	tri generacije fermionov			nosilci interakcij	
	I	II	III		
masa naboj spin	$\approx 2,2 \text{ MeV}$ $2/3$ $1/2$	$\approx 1,28 \text{ GeV}$ $2/3$ $1/2$	$\approx 173,1 \text{ GeV}$ $2/3$ $1/2$	0 0 1	$\approx 125 \text{ GeV}$ 0 0
	u gor	c čar	t vrh	g gluon	H Higgsov bozon
kvarki	$\approx 4,7 \text{ MeV}$ $-1/3$ $1/2$	$\approx 96 \text{ MeV}$ $-1/3$ $1/2$	$\approx 4,18 \text{ GeV}$ $-1/3$ $1/2$	0 0 1	
	d dol	s čudo	b dno	γ foton	
	$\approx 0,511 \text{ MeV}$ -1 $1/2$	$\approx 106 \text{ MeV}$ -1 $1/2$	$\approx 1,78 \text{ GeV}$ -1 $1/2$	$\approx 91,2 \text{ GeV}$ 0 1	
	e elektron	μ mion	τ tauon	Z bozon Z	
leptoni	$\approx 0 \text{ eV}$ 0 $1/2$	$\approx 0 \text{ eV}$ 0 $1/2$	$\approx 0 \text{ eV}$ 0 $1/2$	$\approx 80,4 \text{ GeV}$ ± 1 1	
	ν_e elektronski nevtrino	ν_μ mionski nevtrino	ν_τ tauonski nevtrino	W^\pm bozona W^+ in W^-	umeritveni bozoni

Slika 1: Standardni model osnovnih delcev. Vrednosti so vzete iz [6]. Mase nevtrinov so neničelne, vendar so izjemno majhne in so zato označene kot skoraj ničelne.

Število dogodkov N opazovanega procesa izračunamo po enačbi

$$N = \sigma \int \mathcal{L} dt = \sigma L, \quad (1)$$

kjer je \mathcal{L} *luminoznost* (angl. luminosity) pospeševalnika, L integrirana luminoznost, σ pa *sipalni presek* reakcije, ki ga dobimo iz teoretičnih izračunov. Integrirana luminoznost se meri v enotah recipročne površine in je povezana s številom preletov delcev drug mimo drugega ter njihove številske gostote na mestu trka. Je izključno lastnost pospeševalnika in je neodvisna od procesa, ki ga opazujemo. [5, 7].

2.1 Nova fizika

Četudi je Standardni model najnatančnejša fizikalna teorija, kar jih današnja fizika pozna, je hkrati povsem jasno, da je nepopolna. Poleg tega da ne opisuje gravitacije, pušča neodgovorjenih nekaj pomembnih vprašanj, kot so: zakaj je vesolje sestavljeno iz materije in ne antimaterije, kaj sta temna energija in temna snov, zakaj imajo osnovni delci tako različne mase in kako se v teorijo vključi gravitacija. Posamezne probleme lahko rešimo z dodajanjem novih delcev in interakcij, kar nam odpre množico novih možnosti in teorij. Te teorije označujemo s kratico BSM (angl. beyond the Standard model), saj presegajo Standardni model. Skušamo jih potrditi tako, da v množici nastalih delcev v trkalnikih iščemo take, ki jih Standardni model ne napove [8]. Kot je že iz uvoda jasno, je to metaforično zelo podobno iskanju igle v kopici sena.

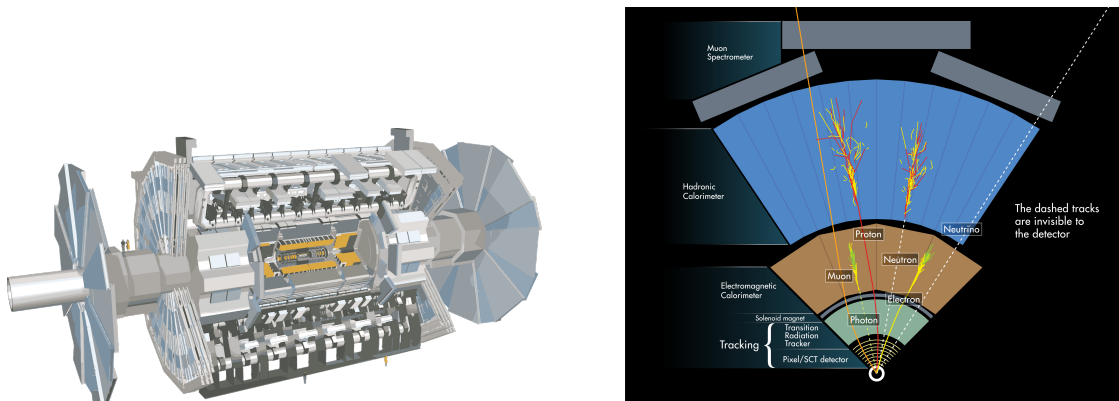
3 LHC in ATLAS

Z iskanjem novih delcev se ukvarjajo številni pospeševalniki po svetu, največji in najmočnejši med njimi je *Veliki hadronski trkalnik* (angl. Large Hadron Collider, LHC) v Evropskem centru za jedrske raziskave (fr. Conseil Européen pour la Recherche Nucléaire, CERN) v Ženevi. LHC je krožni pospeševalnik z obsegom 27 km, v katerem v ultra-visokem vakuumu pospešujejo dva snopa hadronov (tipično protonov) v nasprotni smeri do energij okrog 7 TeV. To ustreza Lorentzovemu faktorju reda 10^4 . Po obsegu je razporejenih več detektorjev, kjer lahko delci iz obeh cevi med seboj trkajo. Poudarek tega seminarja je na detektorju ATLAS (angl. A Toroidal LHC ApparatuS) [9]. Protone v presledkih trkajo več let, da pridobijo kar se da veliko integrirano luminoznost in posledično čim več trkov. Med letoma 2015 in 2018 so pri invariantni masi $\sqrt{s} = 13$ TeV pridobili integrirano luminoznost okrog 140 fb^{-1} [2]. Podatki, pridobljeni v tem času, se imenujejo "Run 2". Invariantna masa je pri teh energijah kar enaka dvakratniku energije posameznega protona pri trku. Enota barn (b) ustreza površini 10^{-28} m^2 . Po Enačbi (1) lahko pričakujemo, da se je proces s sipalnim presekom $\sigma = 1 \text{ fb}$ zgodil približno 140-krat.

3.1 Sestava detektorja

Pri trku dveh protonov lahko nastane cela množica delcev. Mnogo je zelo kratkoživih in razpadejo v druge delce, še preden lahko priletijo do senzorjev. Detektor je sestavljen tako, da lahko vsaj približno ugotovi smeri in energije tistih delcev, ki jih lahko zazna. Sestavljen je iz več komponent (glej Sliko **2a**), ki si čebulasto sledijo od osi trka navzven. Najbližje osi trka so sledilni detektorji, ki s pomočjo silicijevih polprevodnih ploščic zaznavajo nabite delce. Nahajajo se v močnem magnetnem polju (približno 2 T) in iz krivinskega polmera sledi delcev z natančnostjo nekaj odstotkov izračunajo njihove gibalne

količine. V naslednjem sloju sta hadronski in elektromagnetni kalorimeter, ki zaznavata energijo hadronov in fotonov. V zunanjem sloju so nameščeni detektorji za zaznavanje mionov, ki so poleg nevtrinov in morebitnih eksotičnih delcev edini, ki lahko preletijo vse prejšnje sloje (glej Sliko 2b). Nevtrinov se ne da enostavno zaznati, vendar se njihovo prečno energijo in azimutalno smer lahko določi kot manjkajoči del energije vseh zaznanih delcev [10].



(a) Prikaz detektorja ATLAS. Vzeto iz [1]. (b) Prečni prerez detektorja. Vzeto iz [11].

Slika 2: Detektor ATLAS. Na obeh slikah se vidi čebulna struktura, ki je ključna za zaznavanje delcev pri trku.

Skupine bližnjih delcev se s pomočjo algoritmov združi v *pljuske* (angl. jets), ki nam dajo bolj neposredne informacije o nastalih delcih takoj po trku. Pljuskom se lahko določi smer in energijo, z algoritmi strojnega učenja pa lahko pa tudi ali so nastali iz kvarka b ali c (tako imenovana b -označevanje in c -označevanje, angl. tagging) [12].

3.2 Simulacije trkov

Rekonstrukcija dogodkov iz končnih razpadnih produktov je izjemno zahtevna naloga. Del rešitve je uporaba simulacij trkov. S tem lahko rekonstruirajo posamezne pljuske in tako zmanjšajo količino relevantnih podatkov za posamezen dogodek, obenem pa simulacije lahko služijo za treniranje algoritmov strojnega učenja. Večino simulacij temelji na metodi Monte Carlo (MC), ki s pomočjo žrebanja naključnih števil simulira razpadne procese. Velika ovira predstavlja dejstvo, da protoni niso osnovni delci, in da se njihovih lastnosti ne da perturbativno izračunati iz kvantne teorije polja. Mnogi algoritmi tako temeljijo na rezultatih preteklih eksperimentov in k sreči se najnaprednejši algoritmi dobro ujemajo med sabo. Po simulaciji trkov, ki izračuna vektorje četverce gibalnih količin vseh nastalih delcev, je potrebno simulirati, kako bi detektor delce zaznal. Ta ne zazna nujno vseh nastalih produktov, zato se s tem zadnjim korakom simulacije poustvari realistična meritev, kot jo dobijo iz detektorja. Na koncu lahko dodajo še šum in trke prekrijejo enega čez drugega ter s tem lahko sklepajo o sestavi dejanskih meritev. Prednost simulacij je tudi ta, da lahko oblikujejo hipotetične odzive na procese, ki vsebuje kake neodkritne delce, denimo supersimetrične delce, kvarke četrte generacije, nove umeritvene bozone, ali pa celo procese, ki se odvijajo v višjih dimenzijah [3, 13].

Kljub temu da lahko s pomočjo simulacij na podlagi končnih meritev sklepamo o sestavnih delih prvotne reakcije, je to še vedno zapleten proces. V zadnjih letih je razvoj

računalnikov in grafičnih kartic omogočil, da lahko del tega sklepanja prepustimo samo-učečim računalniškim algoritmom.

3.3 Signal in ozadje

Ko iščemo nove procese v trkalniku, je naša glavna naloga, da jih razločimo od kopice drugih procesov, ki dajo zelo podobne razpadne produkte. Iskan proces, oziroma njegove produkte, imenujemo *signal*, vse ostale že poznane procese pa *ozadje*. Novi delci lahko razpadejo na več različnih načinov, imenovanih *razpadni kanali*, na vsakega s svojo verjetnostjo. Pri preučevanju novega delca se običajno omejimo na iskanje samo enega razpadnega kanala. Pri analizi želimo, da je verjetnost za ta razpad čim večja, obenem pa tudi, da se vzporedno ne dogaja še kak drug, že znan proces z enakimi razpadnimi produkti in podobno energijo. Ločevanje signala od ozadja je torej ključnega pomena, hkrati pa je to nekaj, kar se lahko računalniki z nekaj zunanje pomoči naučijo sami [12].

4 Strojno učenje

Strojno učenje je široko področje računalništva in matematike in zanj ne obstaja enotna definicija. V kontekstu tega seminarja je nemara najbolj primerno, če si izposodim izjavo Arthurja Samuela iz leta 1959: “Strojno učenje je področje, ki se ukvarja z omogočanjem računalnikov, da se učijo, ne da bi jih za to eksplicitno programirali.” To zavzema vse od najenostavnejše linearne regresije do najmodernejših generativnih jezikovnih modelov. V tem poglavju si bomo ogledali nekaj osnovnih konceptov strojnega učenja, v nadaljevanju pa se bomo posvetili dvema konkretnima realizacijama, ki nam lahko pomagata pri odkrivanju nove fizike: klasifikatorji ter avtoenkoderji.

Cilj strojnega učenja je, da iz nabora vhodnih podatkov, imenovanih *značilke* (angl. features), napove izhodne podatke, imenovane *oznake* (angl. labels). Značilke so lahko kakršnikoli številski podatki, denimo svetlosti slikovnih točk na sliki. V Ref. [12] so za značilke uporabili smeri in gibalne količine pljuskov ter leptonov po trku. Oznake nam lahko predstavljajo različne stvari – pri klasifikaciji so to različne kategorije, recimo tip oblačila na sliki, lahko je to verjetnost za posamezni dogodek, ali pa je celotno besedilo, ki ga vrne jezikovni model [4]. V Ref. [12] oznaka predstavlja napoved modela, ali je pri trku prisoten signal ali ne.

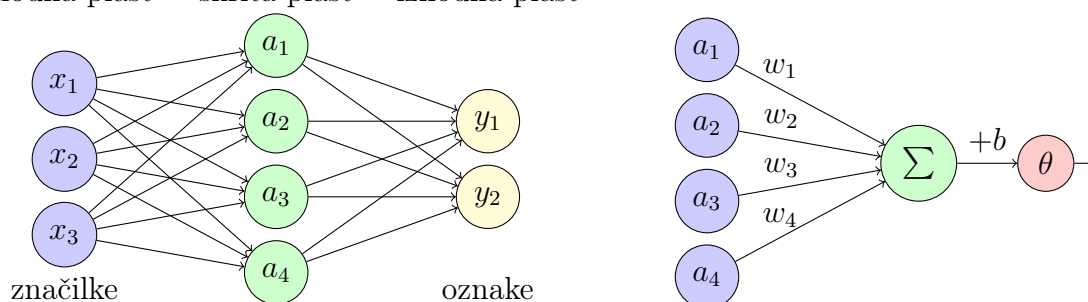
Preden pa je računalnik lahko sposoben smiselno tvoriti oznake, ga je potrebno to naučiti. V preprostem modelu podajamo sistemu značilke skupaj s pripadajočimi oznakami. Sistem nato preveri, kakšno oznako bi dal podanim značilkam in nato popravi svoje parametre tako, da se čim bolj približa pravilnim oznakam. Tako začne model sam povezovati določene lastnosti značilk, ne da bi mu to bilo eksplicitno povedano. Večji kot je model in več kot ima parametrov, bolj zapletene povezave lahko poišče in vrne natančnejše rezultate. Seveda pa z večanjem modela naletimo na druge težave, kot so velika prostorska in časovna zahtevnost.

Pomembno se je zavedati, da različni modeli strojnega učenja delujejo najboljše na različnih tipih podatkov in pogosto je na danih podatkih potrebno preizkusiti več različnih modelov, da se najde najprimernejši [4]. V fizikalnih uporabah strojnega učenja lahko močno koristijo, da značilke odražajo fizikalne lastnosti – denimo, da poleg gibalne količine curka podamo še njen kvadrat, ki je Lorentzovo invarianten. Prav tako je pomembno, da se številske vrednosti značilk ne raztezajo preko več redov velikosti, zato se fizikalne podatke pogosto normalizira, preden se jih poda modelu.

5 Nevronske mreže

Nevronska mreža (angl. neural network, NN) je računalniški model, ki posnema biološke mreže nevronov. Sestavljena je iz več plasti nevronov (glej Sliko 3a). Začetna plast se imenuje *vhodna* (na sliki označeno modro) in predstavlja značilke modela \mathbf{x} , zadnja pa se imenuje *izhodna* (označeno rdeče) in predstavlja oznake \mathbf{y} . Vmes se nahajajo *skrite plasti* (označeno zeleno). Če je skritih plasti več, se nevronske mreže reče *globoka* (DNN). Posamezen nevron je povezan z vsemi nevroni v prejšnji plasti, pri čemer vsaka povezava predstavlja neko utež. Ko v vhodno plast vstavimo nabor značilk, to povzroči aktivacijo nevronov po celotni mreži. Vsak nevron zavzame neko številsko vrednost, imenovano *aktivacija*. Aktivacije vhodnega sloja nevronov so kar vrednosti značilk in narekujejo aktivacije nevronov v nadaljnjih slojih. Aktivacija izhodnega sloja predstavlja napoved modela. Med fazo učenja se uteži modela po majhnih korakih spreminjajo.

vhodna plast skrita plast izhodna plast



(a) Primer enostavne nevronske mreže. Mreža vsebuje tri značilke in dve oznaki ter ima en skriti sloj z dimenzijo 4. Vsaka puščica predstavlja utež. Značilke lahko zberemo v vektor $\mathbf{x} = (x_1, x_2, x_3)$, oznake pa v vektor $\mathbf{y} = (y_1, y_2)$.

(b) Prikaz delovanja enega nevrona. Vsaka povezava predstavlja utež w , utežene aktivacije prejšnjega sloja se seštejejo, doda se pristranskost b in se nato prenese skozi aktivacijsko funkcijo θ .

Slika 3: Dva prikaza komponent nevronske mreže.

Naj vektor $\mathbf{x} \in \mathbb{R}^n$ predstavlja značilke in vektor $\mathbf{a} \in \mathbb{R}^m$ aktivacije naslednjega sloja tako kot na Sliki 3a. Naj bo velikost vektorja \mathbf{x} enaka n , velikost vektorja \mathbf{a} pa enaka m . Število puščic, ki jih lahko narišemo med slojema, in hkrati število uteži je enako $n \cdot m$, zato lahko vse uteži zapišemo v obliki matrike $\mathbf{W} \in \mathbb{R}^{m \times n}$. Vsak nevron v skriti plasti se aktivira z uteženim seštevkom aktivacij prejšnjega sloja, zraven pa se doda še *statistična pristranskost* (angl. bias), ki ga označimo z b . To je le dodaten parameter nevronske mreže, ki se spreminja z učenjem. S tem smo dosegli linearno kombinacijo značilk skupaj s premikom b . Ker pa s kopičenjem takih transformacij ne moremo modelirati zapletenih vzorcev, moramo na koncu vsako tako izračunano vrednost prenesti še skozi neko nelinearno funkcijo θ , ki ji rečemo *aktivacijska funkcija*. Vrednost, ki jo poda ta funkcija predstavlja aktivacijo nevrona. Zaradi enostavnejše praktične implementacije je priporočljivo, če ima ta funkcija dobro definiran odvod in je enostavna za računanje. Najpogosteje se uporabljata logistična funkcija¹ in *usmerjena linearna enota*² (angl. rectified linear unit, ReLU). Za posamezen nevron velja

$$a_i = \theta(x_1 w_{i1} + x_2 w_{i2} + \cdots + x_n w_{in} + b_i), \quad (2)$$

¹Logistična funkcija ali *sigmoida* ima predpis $\sigma(x) = 1/(1 + e^{-x})$.

² $\text{ReLU}(x) = \max\{x, 0\}$.

skupaj pa lahko aktivacijo celotnega sloja zapišemo z vektorsko enačbo

$$\mathbf{a} = \boldsymbol{\theta}(\mathbf{W}\mathbf{x} + \mathbf{b}), \quad (3)$$

kjer je $\mathbf{b} \in \mathbb{R}^m$ vektor pristranskosti. Tako deluje tudi vsaka naslednja plast, pri čemer se vektor \mathbf{a} iz prejšnje plasti uporabi kot vektor \mathbf{x} za naslednjo plast. Aktivacijske funkcije se lahko med sloji razlikujejo, zlasti pri izhodnem sloju, kjer pogosto želimo da so oznake normirane na interval $[0, 1]$. Velikost in število skritih plasti in vrste aktivacijskih funkcij so lastnosti modela, ki jih določimo pred začetkom učenja. Pogosto je potrebno preizkusiti več različnih konfiguracij, da lahko najdemo najboljšo [4].

5.1 Učenje nevronske mreže

Preden lahko z modelom napovedujemo oznake, ga je potrebno naučiti. V primeru klasifikatorja uporabljamo nadzorovano učenje, kjer modelu ob značilkah podamo tudi pravilne oznake. Učenje predstavlja algoritem *optimizer*, ki na podlagi napake napovedi modela popravi uteži in pristranskosti. Najprej je potrebno definirati neko *funkcijo izgube* L (angl. loss function), ki meri razliko med napovedmi modela in pravilnimi oznakami. Najenostavnejša taka funkcija je kvadrat razlike med napovedanimi in pravimi oznakami, ki jo zapišemo kot

$$L = \|\mathbf{y} - \hat{\mathbf{y}}\|^2 = \sum_{i=1}^n (y_i - \hat{y}_i)^2. \quad (4)$$

Pri tem $\hat{\mathbf{y}}$ predstavlja vektor pravih oznak, \mathbf{y} vektor napovedanih oznak, y_i in \hat{y}_i pa posamezne oznake, katerih skupno število je n . Cilj optimizacijskega algoritma je poiskati minimum funkcije izgube, ki je odvisna od uteži \mathbf{W} in pristranskosti \mathbf{b} . Število prostih parametrov modela (uteži in pristranskosti) se lahko meri v milijonih, funkcija izgube pa je odvisna od vseh teh parametrov. V vsakem koraku učenja želimo spremeniti parametre tako, da se funkcija izgube čimbolj zmanjša. Najenostavnejši optimizator je *vzratna propagacija napake* (angl. backpropagation), ki išče minimum s pomočjo majhnih korakov v smeri gradienta funkcije izgube po parametrih. Ker so nevronske mreže praktično linearne transformacije, opremljene z nelinearnimi aktivacijskimi funkcijami, je gradient funkcije izgube glede na uteži mogoče izračunati z verižnim pravilom odvajanja. To je tudi razlog, zakaj so aktivacijske funkcije izbrane tako, da so njihovi odvodi enostavni. Posamezen korak optimizacije lahko shematsko zapišemo kot

$$W_{\text{novi}} = W_{\text{stari}} - \alpha \nabla L(W), \quad (5)$$

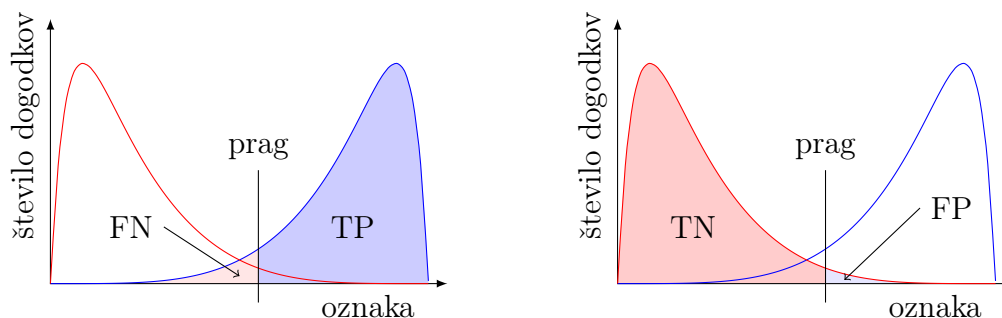
kjer W predstavlja nabor vseh uteži in pristranskosti v celotni nevronske mreži, α pa je *korak učenja* (angl. learning rate). Ta določa, kako velik korak naredimo v smeri gradienta. Na začetku je smiselno imeti dovolj velik korak, da se model hitro približa minimumu, kasneje pa je potrebno korak zmanjševati, da se model lahko ustali [4]. Obstajajo tudi naprednejši in učinkovitejši optimizatorji; dandanes je med najpopularnejšimi *Adam* [14], ki ga uporabljajo tako v Ref. [12] kot v Ref. [15].

5.2 Mere za uspešnost modela

Ko končamo z učenjem nevronske mreže, potrebujemo neko mero uspešnosti modela. Za lažjo predstavo privzemimo, da mreža vrne samo eno oznako, ki je zvezna spremenljivka na intervalu $[0, 1]$. Denimo, da to v konkretnem primeru pomeni prisotnost Higgsovega bozona

v reakciji. Ker je to zvezna spremenljivka, lahko izberemo poljuben prag (angl. treshold), ki ločuje prisotnost in odsotnost bozona. Ko si izberemo neko vrednost praga, dobimo štiri vrste primerov (glej Sliko 4):

- *pravilno pozitivni* (TP): model je pravilno napovedal prisotnost bozona,
- *lažno pozitivni* (FP): model je napovedal prisotnost bozona, v resnici pa ga ni bilo,
- *pravilno negativni* (TN): model je pravilno napovedal odsotnost bozona,
- *lažno negativni* (FN): model je napovedal odsotnost bozona, v resnici pa je bil.



Slika 4: Prikaz pravilno pozitivnih in negativnih ter lažno pozitivnih in negativnih primerov. Rdeča krivulja predstavlja porazdelitev negativnih primerov po oznaki, modra pa pozitivnih. Pokončna črna črta označuje izbran prag.

Pri določanju uspešnosti modela nas predvsem zanima, kako so te štiri količine odvisne od izbire meje. V ta namen vpeljemo dve novi količini, imenovani *delež pravilno pozitivnih primerov* (angl. true positive rate, TPR) in *delež lažno pozitivnih primerov* (angl. false positive rate, FPR). Definiramo ju kot

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}. \quad (6)$$

TPR predstavlja delež dejanskih pozitivnih primerov med vsemi, ki jih je model napovedal kot pozitivne. FPR pa predstavlja delež dejanskih negativnih primerov med vsemi, ki jih je model napovedal kot negativne. Obe količini sta odvisni od izbire praga, zato narišemo parametrično krivuljo obeh količin, ko prag preteče vse vrednosti med najnižjo in najvišjo. Nastali krivulji rečemo ROC (angl. receiver operating characteristic). Nekaj primerov je prikazanih na Sliki 6. Popoln model, ki bi znal pravilo klasificirati vse primere, bi imel ROC krivuljo, ki bi šla skozi točko $(0, 1)$ – Heavisideova funkcija. Popolnoma naključen model bi imel ROC krivuljo, ki bi bila enaka premici $y = x$ (zeleno krivuljo na Sliki 6). Bolj kot se ROC krivulja približa levemu zgornjemu kotu, boljši je model. Za enostavno primerjavo med modeli se pogosto vzame površina pod ROC krivuljo (AUC, angl. area under curve). Zavzame lahko vrednosti med $1/2$ in 1 – večja kot je vrednost, boljši je model [4]. To je izredno uporabna mera za primerjavo različnih modelov na istih podatkih in jo uporabljajo tudi v Ref. [12].

5.3 Uporaba nevronske mreže v fiziki delcev

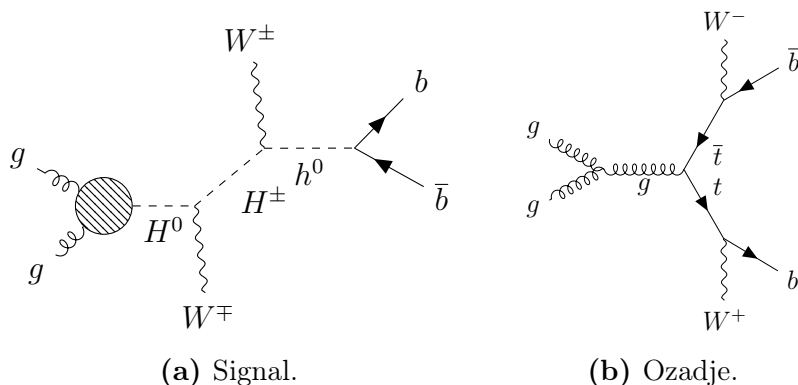
Strojno učenje dandanes sili že v vsak korak v procesu obdelave podatkov iz ATLAS detektorja, od simulacij dogodkov do združevanja razpadnih produktov v pljuske, b -označevanje

Mreža	Površina pod ROC krivuljo		(AUC) Vse značilke
	Osnovne	Sestavljene	
Plitva	$0,733 \pm 0,007$	$0,777 \pm 0,001$	$0,816 \pm 0,004$
Globoka	$0,880 \pm 0,001$	$0,800 \pm 0,001$	$0,885 \pm 0,002$

Tabela 1: Vrednosti AUC pri uporabi različnih vrst značilke za plitve in globoke nevronske mreže [12].

in odkrivanje novih delcev [16, 17, 18, 15]. V tem razdelku se bomo posvetili primeru uporabe strojnega učenja za iskanje novih procesov iz meritev detektorja ATLAS. Obdelava podatkov eksperimenta ATLAS s pomočjo nevronskih mrež za zdaj še ni vodila v revolucij odkritij nove fizike, se pa vlaga veliko truda v demonstracijo prednosti nevronskih mrež pred običajnimi metodami.

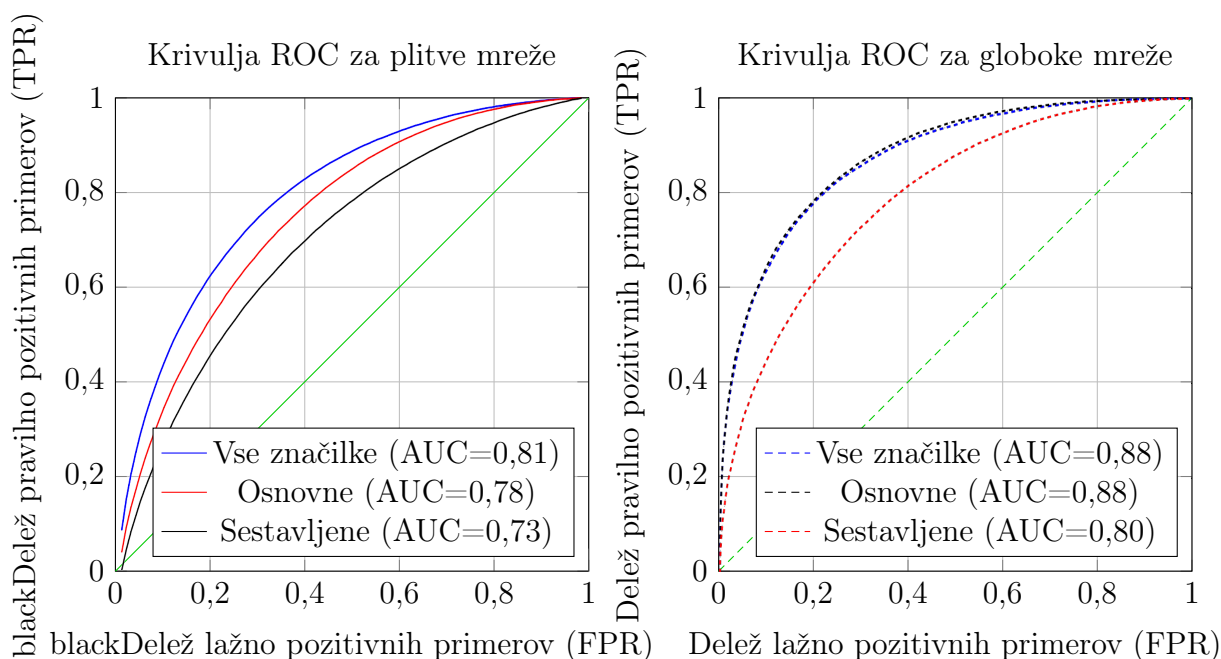
V Ref. [12] so se osredotočili na hipotetičen razpad težkega nevtralnega Higgsovega bozona H_0 preko hipotetičnega nabitega težkega Higgsovega bozona H^\pm in potrjenega lahkega Higgsovega bozona h_0 na dva W bozona in dva b kvarka. Razpadni proces je prikazan na Sliki 5a. Ta proces je v tem primeru signal. Proces ozadja, ki se najbolj prekriva z iskanim razpadom, je prikazan na Sliki 5b. Začetni in končni produkti so v tem primeru popolnoma enaki, edina razlika je kinematiki: signal bo imel resonančni vrh pri masi H_0 , ozadje pa ne.



Slika 5: Feynmanova diagrama za razpad težkega Higgsovega bozona H^0 na $W^+W^-b\bar{b}$ (signal) in razpad gluona na iste produkte (ozadje). Šrafirana pika predstavlja nepoznan in nepomemben del procesa.

S pomočjo prej naštetih orodij (glej podpoglavje 3.2) so simulirali okrog pol milijona primerov signala in ozadja pri $m_{H^0} = 425$ GeV in $m_{H^\pm} = 325$ GeV. Za vsak primer razpada je simulacija priskrbel 21 osnovnih (angl. low-level) in 7 sestavljenih (angl. high-level) značilke [19]. Osnovne značilke predstavljajo velikosti in smeri gibalnih količin štirih pljuskov, njihove b -oznake, smer in gibalno količino nastalega leptona ter smer nastalega nevtrina. Sestavljene značilke so preračunane iz osnovnih in predstavljajo skupno invariantno maso nekaterih komponent, kot je denimo skupna invariantna masa nevtrina in leptona $m_{\ell\nu}$, ki bi morala imeti resonanco pri masi W bozona. To so zelo relevantne fizikalne količine, ki močno pospešijo učenje plitvih nevronskih mrež. Kaj pa če obstajajo katere pomembne izpeljane količine, za katere morda še ne vemo? Bilo bi zelo koristno, če bi model znal sam izluščiti najpomembnejše lastnosti podatkov. V Ref. [12] so preverili, kako dobro se na posameznih značilkah uči "plitva" nevronska mreža z enim samim skritim slojem in globoka nevronska mreža z do 6 skritimi sloji. Primerjali so vrednosti površine pod ROC krivuljo (AUC) pri različnih modelih, rezultati so prikazani v Tabeli 1, krivulje ROC pa na Sliki 6.

Ugotovili so, da imajo globoke nevronske mreže v vseh kategorijah višjo velikost površine pod krivuljo (AUC) kot plitve mreže, kar pomeni, da so uspešnejše pri ločevanju



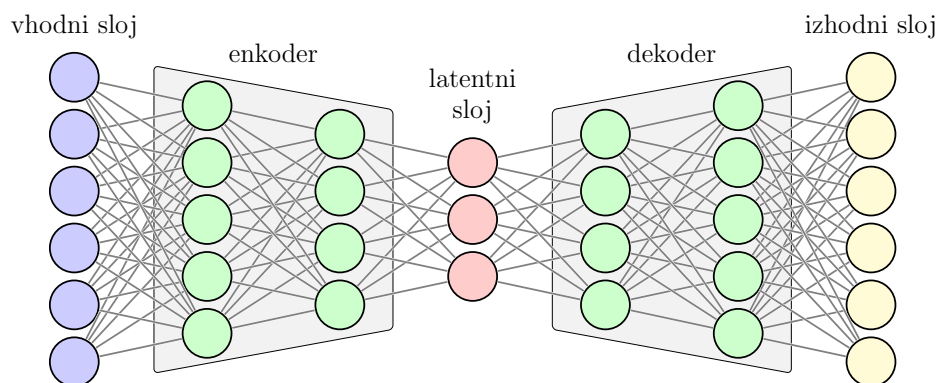
Slika 6: Krivulji ROC za plitvo in globoko nevronska mrežo pri različnih vrstah značilk. Zraven so označene vrednosti površine pod krivuljo (AUC). Večja kot je ta številka, boljši je model. Grafa nam razkrivata, da plitvi nevronske mreži močno koristi dodajanje izpeljanih značilk, globoka mreža pa je dovolj uspešna že samo z osnovnimi značilkami. Zeleni črti predstavljata popolnoma netreniran model. Povzeto iz [12].

signala od ozadja. Iz prve vrstice Tabele 1 vidimo, da je uspešnost plitve mreže pri osnovnih značilkah enaka $AUC = 0,733 \pm 0,007$, če pa poleg osnovnih dodamo še sestavljene značilke, se ta vrednost poveča na $AUC = 0,816 \pm 0,004$. To pomeni, da mreža ni sposobna sama izpeljati sestavljenih značilk, če ji jih ne podamo. Pri globoki mreži (druga vrstica v Tabeli 1 in Slika 6 desno) je razlika med AUC vrednostmi praktično ničelna. Ker globoka mreža pri dodanih izpeljanih značilkah ne razlikuje signala od ozadja bistveno bolje, lahko sklepamo, da se jih je naučila sama. Slabost takega pristopa se odraža v tem, da nam globoka nevronska mreža ne poda nobenega fizikalnega vpogleda v morebitne sestavljene značilke, saj celotna mreža deluje kot črna škatla.

Teh modelov niso preizkusili na pravih podatkih iz ATLASa, temveč le na hipotetičnih simulacijah, zato nam nič ne povedo o obstoju obravnavanega procesa $gg \rightarrow H_0 \rightarrow WWb\bar{b}$. Vseeno pa so s to analizo demonstrirali pomembno lastnost globokih nevronske mreže – sposobnost samostojnega učenja sestavljenih značilk in s tem povezovanja kompleksnejših vzorcev v podatkih. Za zdaj lahko s pomočjo fizikalnega znanja sami izračunamo sestavljene značilke, ko pa se enkrat odpravimo iskati novo fiziko, se morda pojavijo nove izpeljane količine, za katere ne vemo. V takem primeru je ključno, da imamo sistem, ki je sposoben iskati zapletene vzorce. V Ref. [12] se niso trudili neposredno iskati nove fizike, so pa podlagi simuliranih podatkov demonstrirali, da so lahko globoke nevronske mreže pri tem ključne.

6 Avtoenkoderji

Avtoenkoderji so posebna vrsta nevronske mreže, primarno namenjeni zmanjševanju dimenzije podatkov. Omogočajo nam nenadzorovano učenje, kar ima v fizikalnem kontekstu nekaj prednosti. V osnovni različici je avtoenkoder nevronska mreža, kjer skrite sloje lahko razdelimo v tri dele. Prvi del je *enkoder*, ki ga sestavljajo postopoma manjši sloji nevronov. Drugi del je *latentni sloj*, ki ima manj nevronov kot vhodni sloj in tako z manj parametri predstavlja vhodne podatke. Tretji del je *dekoder*, ki ga sestavljajo postopoma večji sloji. Tipično je dekoder kar zrcalna slika enkoderja (glej Sliko 7). V avtoenkoderjih ima izhodni sloj vedno enako dimenzijo kot vhodni.



Slika 7: Shematska sestava preprostega avtoenkoderja. Vhodni in izhodni sloj imata šest parametrov, latentni sloj pa samo tri.

Uspešnost celotne mreže merimo kot razliko med vhodnimi in izhodnimi podatki. Cilj je, da se model nauči čim boljše poustvariti vhodne podatke, pri čemer jih mora vmes zmanjšati na manjše število dimenzij. S tem se nauči izluščiti najpomembnejše lastnosti podatkov.

Če se tak model uči samo na eni vrsti podatkov – recimo na procesih ozadja v pospeševalniku – in se jih nauči dobro rekonstruirati, potem tega morda ne zna dobro storiti za procese signala. Če torej v naučen model podamo nek dogodek in model vrne približno iste oznake nazaj, lahko sklepamo, da je dogodek podoben tistim, na katerih se je model učil – torej ozadju. Če pa model vrne bistveno drugačne oznake, je to lahko znak, da je dogodek nekaj posebnega – signal. To je en primer implementacije zaznave odstopanj. Pri trkih v eksperimentu ATLAS nam tak pristop lahko koristi, saj se z njim ne zavežemo posebni vrsti teorije, ki opisuje novo fiziko, ampak lahko zaznavamo odstopanja od Standardnega modela v kateremkoli (tudi še neteoretiziranem) procesu [4].

6.1 Uporaba avtoenkoderjev za iskanje nove fizike

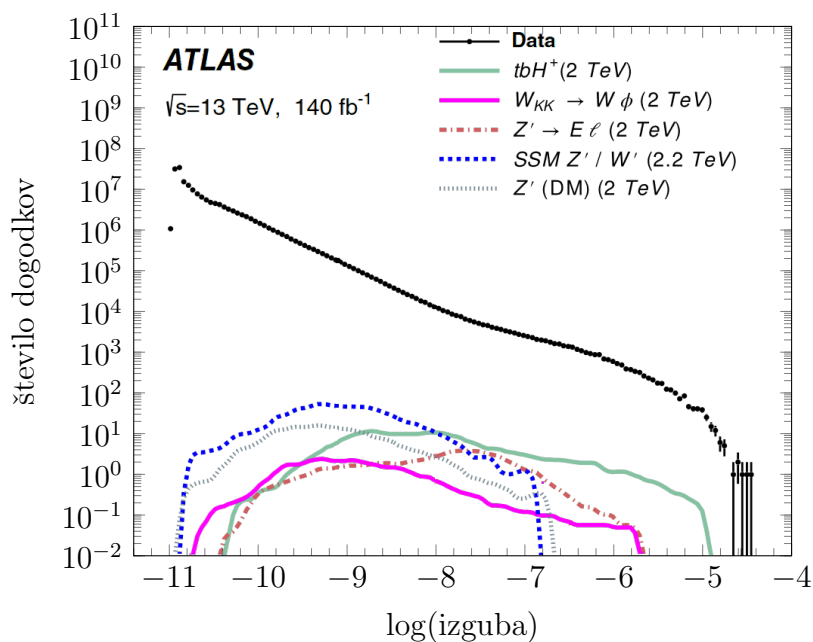
Poglejmo si primer uporabe avtoenkoderjev za iskanje nove fizike na eksperimentu ATLAS. V tem primeru ne gre za osredotočanje na kakšen poseben nov delec ali proces, temveč za zaznavo *kakršnihkoli* odstopanj od Standardnega modela. V Ref. [15] so preizkusili, kako deluje ta pristop. Sestavili so avtoenkoder z vhodnim velikosti 1278. Izhodni sloj je imel prav tako velikost 1287, saj morata biti v avtoenkoderju vhodni in izhodni sloj enakih dimenzij. Za vsak izmerjen trk so izbrali 1278 podatkov, ki so zelo podobni tistim, ki so opisani v podpoglavju 5.3 (smeri in velikosti gibalnih količin nastalih pljuskov in leptonov), le da so tokrat uporabili več kombinacij invariantnih mas. Latentni sloj je imel 200 nevronov, kar pomeni, da se je nevronska mreža učila stisniti 1278 vrednosti v zgolj 200 in nato iz njih nazaj sestaviti začetne podatke. Za funkcijo izgube so uporabili kvadrat

razlike med vhodnim in izhodnim slojem – tako kot v Enačbi (4), le da so pravilne oznake $\hat{\mathbf{y}}$ kar enake značilkam značilke \mathbf{x} :

$$\mathcal{L} = \|\mathbf{x} - \mathbf{y}\|^2 \quad (7)$$

Cilj avtoenkoderja je namreč, da se z izhodnimi podatki čim bolj približa vhodnim. Mrežo so trenirali so na simuliranih podatkih, kot je opisano v podpoglavju 3.2.

Med procesom učenja je mreža skušala vsak trk karakterizirati z 200 parametri (kolikor je dimenzija latentnega sloja) in hkrati iz teh parametrov nazaj poustvariti vsak trk. Funkcija izgube meri kvadrat razlike med vhodnimi in izhodnimi podatki (Enačbi (7)). Med procesom učenja se jo model trudi zmanjšati za dogodke iz Standardnega modela. Po učenju pa nam podaja mero za verjetnost, ali posamezen dogodek pripada Standardnemu modelu ali ne – večja kot je vrednost funkcije izgube za dogodek, večja je verjetnost, da je dogodek del nove fizike. Na Sliki 8 je s črno prikazana porazdelitev funkcije izgube za 166 055 597 trkov, izmerjenih na eksperimentu ATLAS. Gre za histogram, pri čemer je širina posameznega stolpca zelo majhna. Vidimo lahko, da ima več kot 90 % dogodkov vrednost logaritma funkcije izgube manj -10 od zato lahko za njih z določeno gotovostjo trdimo, da so procesi Standardnega modela. Število dogodkov z večjimi vrednostmi izgube pa strmo pada. V barvah so prikazane porazdelitve funkcije izgube za simulirane dogodke nekaterih prevladujočih teorij nove fizike. Pomembna razlika med pravimi podatki (črno) in teorijami onkraj Standardnega modela (barvno) je ta, da ima funkcija izgube za prave podatke v večini primerov relativno majhne vrednosti z vrhom pri -11 , vse ostale teorije pa imajo porazdelitve izgub pomaknjene proti višjim vrednostim. Za primer si lahko pogledamo polno zeleno krivuljo, ki predstavlja simulirane trke, kjer je vključen hipotetičen težki Higgsov bozon H^+ (isti kot v podpoglavju 5.3). Vidimo, da za take trke logaritem funkcije izgube najverjeteneje zavzame vrednosti med -10 in -6 , kar predstavlja bistveno odstopanje od -11 , kolikor je značilno za prave meritve (črno). Na žalost pa bi bilo število takih trkov, če obstajajo, kvečjemu nekaj deset, kar je bistveno premalo za kakršnokoli statistično sklepanje.



Slika 8: Histogram vrednosti funkcije izgube za 166 055 597 podatkov iz detektorja ATLAS (črno). Na isti graf so narisane porazdelitve funkcije izgube za simulacije podobnih dogodkov nekaterih prevladujočih teorij onkraj standardnega (barvno). Podrobnosti presegajo obseg tega seminarja. Povzeto po [15].

Vseeno analiza iz [15] dokazuje, da avtoenkoderji lahko vsaj v principu delujejo kot orodje za odkrivanje nove fizike, na žalost pa je število takih dogodkov za zdaj še premajhno. Zaradi tega v Ref. [15] niso s primerno gotovostjo uspeli potrditi kakršnegakoli novega procesa.

7 Zaključek

Uporaba strojnega učenja vstopa z različnih koncev v tok obdelave podatkov z detektorja ATLAS. Razvoj računalniške opreme je v zadnjem desetletju omogočil hitrejšo uporabo strojnega učenja, kar je mnoge pritegnilo, da si z njim pomagajo pri odkrivanju procesov znotraj detektorja. V tem seminarju smo si ogledali splošno delovanje nevronske mreže in njihovo uporabo pri klasificiranju posebnega razpada hipotetičnega težkega Higgsovega bozona. Globoke nevronske mreže so sposobne same določiti najpomembnejše lastnosti podatkov in jim za uspešno delovanje ni potrebno dodajati izračunanih sestavljenih značilk. Ker večina osnovnih nevronske mreže temelji na poznavanju alternativne fizikalne teorije, so nekateri avtorji poskusili uporabiti avtoenkoderje za zaznavo kakršnihkoli odstopanj. Takšni pristopi za zdaj še niso uspeli dokazati signifikantnih odstopanj od Standardnega modela. Nemara ljudje še niso preizkusili pravega pristopa, morda so trenutne energije trkov premajhne, ali pa preprosto še ni bilo izmerjenih dovolj trkov, da bi na podlagi izjemno redkih dogodkov lahko z dovoljšnjo gotovostjo zaznali novo fiziko.

Literatura

- [1] *Detector & Technology*. CERN. 2024. URL: <https://atlas.cern/Discover/Detector> (pridobljeno 15. 3. 2024).
- [2] ATLAS Collaboration. »Luminosity determination in pp collisions at $\sqrt{s} = 13$ TeV using the ATLAS detector at the LHC«. V: *The European Physical Journal C* 982.83 (2023). DOI: <https://doi.org/10.48550/arXiv.2212.09379>.
- [3] *Monte Carlo production chain*. ATLAS Open Data Documentation. 2024. URL: https://opendata.atlas.cern/docs/documentation/monte_carlo/MC_production_chain/ (pridobljeno 15. 3. 2024).
- [4] A. Géron. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. Ur. R. Roumeliotis in N. Tache. 2. izd. O'Reilly Media, Inc., 2019. ISBN: 9781492032649.
- [5] D. J. Griffiths. *Introduction to elementary particles; 2nd rev. version*. Physics textbook. New York, NY: Wiley, 2008. ISBN: 9783527406012. URL: <https://cds.cern.ch/record/111880>.
- [6] Particle Data Group in sod. »Review of Particle Physics«. V: *Progress of Theoretical and Experimental Physics* 2022.8 (avg. 2022), str. 083C01. ISSN: 2050-3911. DOI: 10.1093/ptep/ptac097. eprint: https://academic.oup.com/ptep/article-pdf/2022/8/083C01/49175558/ptac097_19_miscellaneous_searches_and_searches_in_other_sections.pdf. URL: <https://doi.org/10.1093/ptep/ptac097>.
- [7] B. Povh in sod. *Particles and Nuclei: An Introduction to the Physical Concepts*. Springer Berlin Heidelberg, 2008. ISBN: 9783540793687. URL: <https://books.google.si/books?id=XyW97WGyVbkC>.
- [8] B. C. Allanach. »Beyond the Standard Model«. V: *CERN Yellow Rep. School Proc.* 6 (2019), str. 113–144. DOI: 10.23730/CYRSP-2019-006.113. URL: <https://cds.cern.ch/record/2702850>.
- [9] *The Large Hadron Collider*. CERN. 2024. URL: <https://home.cern/science/accelerators/large-hadron-collider> (pridobljeno 15. 3. 2024).
- [10] I. Efthymiopoulos. *Overview of the ATLAS detector at LHC*. Teh. poročilo 7. version revised com-conf-99-002_version1 on Tue Jun 22 21:18:36 MET DST 1999. Geneva: CERN, 1999. URL: <https://cds.cern.ch/record/409257>.
- [11] J. Pequeno in P. Schaffner. »How ATLAS detects particles: diagram of particle paths in the detector«. 2013. URL: <https://cds.cern.ch/record/1505342> (pridobljeno 24. 3. 2024).
- [12] P. Baldi, P. Sadowski in D. Whiteson. »Searching for exotic particles in high-energy physics with deep learning«. V: *Nature Communications* 5.1 (jul. 2014). ISSN: 2041-1723. DOI: 10.1038/ncomms5308. URL: <http://dx.doi.org/10.1038/ncomms5308>.
- [13] C. Bierlich in sod. *A comprehensive guide to the physics and usage of PYTHIA 8.3*. Stran 46. 2022. arXiv: 2203.11601 [hep-ph].
- [14] D. P. Kingma in J. Ba. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: 1412.6980 [cs.LG].

- [15] »Search for New Phenomena in Two-Body Invariant Mass Distributions Using Un-supervised Machine Learning for Anomaly Detection at $\sqrt{s} = 13$ TeV with the ATLAS Detector«. V: *Phys. Rev. Lett.* 132 (8 feb. 2024), str. 081801. DOI: 10.1103/PhysRevLett.132.081801. URL: <https://link.aps.org/doi/10.1103/PhysRevLett.132.081801>.
- [16] S. Otten in sod. *Event Generation and Statistical Sampling for Physics with Deep Generative Models and a Density Information Buffer*. 2021. arXiv: 1901.00875 [hep-ph].
- [17] ATLAS Collaboration in Irinel Caprini. »A neural network clustering algorithm for the ATLAS silicon pixel detector«. V: *Journal of Instrumentation* 9 (jun. 2014). DOI: 10.1088/1748-0221/9/09/P09009.
- [18] Michela Paganini. *Machine Learning Algorithms for b-Jet Tagging at the ATLAS Experiment*. 2017. arXiv: 1711.08811 [hep-ex].
- [19] D. Whiteson. *HIGGS*. UCI Machine Learning Repository. 2014. DOI: <https://doi.org/10.24432/C5V312>.