

Title of thesis

Second title line

Name

Bachelorarbeit

Name

An der Fakultät für Physik
Institut für Experimentelle Kernphysik (IEKP)

Erstgutachter: ?
Zweitgutachter: ?

Karlsruhe, XX. Monat 20XX

Inhaltsverzeichnis

1. Einleitung	1
2. Theoretische Grundlagen	3
2.1. Das Standardmodell der Teilchenphysik	3
2.2. Assoziierte Higgs-Boson-Top-Quark-Paar-Produktion (t <bar>t>H)</bar>	4
3. Experimentelle Grundlagen	7
3.1. Der Large Hadron Collider (LHC)	7
3.2. Der Compact-Muon-Solenoid-Detektor (CMS)	8
3.3. t <bar>t>H-Analyse</bar>	9
4. Algorithmen zur multivariaten Analyse	11
4.1. Grundlagen zur multivariaten Datenanalyse	11
4.2. Boosted Decision Trees (BDTs)	12
4.2.1. Entscheidungsbäume	12
4.2.2. Verstärken von Entscheidungsbäumen (Boosting)	12
4.2.3. Überanpassung (overtraining)	14
4.2.4. Variieren der Trainingsereignisse (Bagging)	15
4.3. Verwendete Algorithmen zur multivariaten Analyse	15
4.3.1. Toolkit for Multivariate Analysis in ROOT (TMVA)	16
4.3.2. Scikit-Learn – machine learning in python	16
4.3.3. Extreme Gradient Boosting (XGBoost)	16
5. Vergleich der multivariaten Algorithmen	17
5.1. Vergleichbarkeit der Algorithmen	17
5.1.1. ROC-Kurve	18
5.2. Verwendete Datensätze	18
5.3. Anwendung und Vergleich der Algorithmen zur ttH Analyse	18
Literaturverzeichnis	19
Anhang	21
A. Anhang 1	21

1. Einleitung

[B⁺12] ...

2. Theoretische Grundlagen

Das Standardmodell der Teilchenphysik beschreibt die bisher bekannten Bausteine der Materie, sowie deren Wechselwirkungen. In Kapitel 2.1 wird ein kurzer Überblick über gegeben, woraufhin in Abschnitt 2.2 genauer auf die Produktion von Higgs-Boson und Topquark eingegangen wird.

2.1. Das Standardmodell der Teilchenphysik

Der folgende Abschnitt soll einen kurzen Überblick über das Standardmodell der Elementarteilchenphysik geben, dabei bezieht er sich meist auf [Pov14].

Das Standardmodell der Elementarteilchenphysik ist eine Quantenfeldtheorie, die die Theorie der elektroschwachen Wechselwirkung mit der Quantenchromodynamik zusammenfasst und vereinheitlicht. Das Universum besteht aus einigen grundlegenden Bausteinen, die durch vier elementare Kräfte beeinflusst werden [O'L12h]. Die bislang beste Beschreibung dieses Aufbaus liefert das Standardmodell, wenngleich es die vierte Kraft, die Gravitation, nicht erklärt kann. Dennoch war es mit diesem Modell möglich fast alle experimentellen Ergebnisse zu bestätigen, sowie sehr präzise Vorhersagen über verschiedene Phänomene zu treffen.

Die bislang entdeckte Materie besteht aus zwei Arten von Elementarteilchen, den Leptonen sowie den Quarks. Diese lassen sich jeweils in drei Familien unterteilen. Jede Quark-Familie besteht jeweils aus einem Quarkpaar und deren Antiteilchen, diese sind Up- und Down-, Strange- und Charm-, sowie Bottom- und Topquark.

Leptonen bilden jeweils zusammen mit dem dazugehörigen Neutrino und den jeweiligen Antiteilchen eine Familie. Im Gegensatz zu den Quarks unterliegen Leptonen nicht der starken Wechselwirkung.

Die dritte elementare Wechselwirkung, die im Standardmodell beschrieben ist, ist die elektromagnetische Wechselwirkung. Ihr unterliegen alle geladenen Teilchen. Diese Wechselwirkungen sind in ihrer Struktur sehr ähnlich und werden durch den Austausch von Vektorbosonen vermittelt. Diese sind die Gluonen der starken Wechselwirkung, die W- und Z-Bosonen der schwachen Wechselwirkung und die Photonen (γ) der elektromagnetischen. Während die Fermionen aus denen die Materie besteht, einen halbzahligen Spin besitzen, haben die Bosonen einen ganzzahligen Spin.

Der letzte fehlende Baustein im Standardmodell ist ein elementares Spin-0 Teilchen, ohne das keine konsistente Erklärung für die W und Z⁰ Massen möglich wäre. Dieses ist das

Higgs-Boson, welches 2012 am CERN entdeckt wurde. Die Kopplung zwischen Higgs-Boson und anderen Elementarteilchen ist proportional zur Fermionenmasse. In 2.1 sind alle elementaren Bosonen und Fermionen dargestellt.

Drei Generationen der Materie (Fermionen)				
	I	II	III	
Massen →	2,3 MeV	1,275 GeV	173,07 GeV	0
Ladung →	$\frac{2}{3}$	$\frac{2}{3}$	$\frac{2}{3}$	0
Spin →	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	0
Name →	u up	c charm	t top	γ Photon
Quarks				
	4,8 MeV $-\frac{1}{3}$ $\frac{1}{2}$	95 MeV $-\frac{1}{3}$ $\frac{1}{2}$	4,18 GeV $-\frac{1}{3}$ $\frac{1}{2}$	0 0 1 g Gluon
	d down	s strange	b bottom	
Leptonen				
	<2 eV 0 $\frac{1}{2}$	<0,19 MeV 0 $\frac{1}{2}$	<18,2 MeV 0 $\frac{1}{2}$	Z^0 91,2 GeV 0 1 Z Boson
	ν_e Elektron-Neutrino	ν_μ Myon-Neutrino	ν_τ Tau-Neutrino	
Eichbosonen				
	0,511 MeV -1 $\frac{1}{2}$	105,7 MeV -1 $\frac{1}{2}$	1,777 GeV -1 $\frac{1}{2}$	W^\pm 80,4 GeV ± 1 1 W Boson
	e Elektron	μ Myon	τ Tau	

Abbildung 2.1.: Die 12 fundamentalen Fermionen und 5 fundamentalen Bosonen des Standardmodells der Teilchenphysik,
Quelle: [Wik10]

Insgesamt stimmen die experimentellen Ergebnisse gut mit den Vorhersagen des Standardmodells überein. Dennoch reicht das Modell nicht aus, um sämtliche Phänomene zu erklären. Im Modell werden beispielsweise masselose Neutrinos gefordert, allerdings ist durch die Beobachtung von Neutrinooszillationen erwiesen, dass massive Neutrinos existieren.

2.2. Assozierte Higgs-Boson-Top-Quark-Paar-Produktion ($t\bar{t}H$)

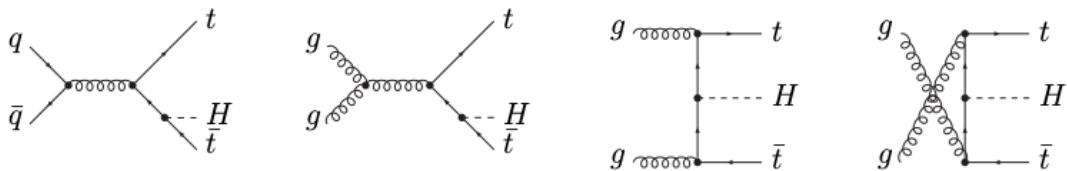
Da die Kopplungskonstante des Higgs-Mechanismus im Standardmodell von der Fermionenmasse abhängt, ist eine Untersuchung der Kopplung zwischen Top-Quark und Higgs-Boson aufgrund der hohen Masse des Top-Quarks verglichen mit anderen Quarkmassen, besonders interessant. In Tabelle 2.1 sind zum Vergleich die Quarkmassen aufgelistet.

Tabelle 2.1.: Tabelle mit Quarkmassen [O⁺14]

Quark	Symbol	Masse
Up	u	$2,3^{+0,7}_{-0,5}$
Down	d	$4,8^{+0,5}_{-0,3}$ MeV
Strange	s	95 ± 5 MeV
Charm	c	$1,275 \pm 0,025$ GeV
Bottom	b	$4,18 \pm 0,03$ GeV
Top	t	$173,07 \pm 0,52 \pm 0,72$ GeV

Diese Kopplung kann während der assoziierten Produktion eines Higgs-Bosons mit einem Paar aus Top-Quark und Anti-Top-Quark untersucht werden.

Wechselwirkungen zwischen Teilchen können durch Feynman-Diagramme visualisiert werden. In Abbildung 2.2 sind exemplarisch einige Feynmandiagramme zur $t\bar{t}H$ -Produktion in führender Ordnung abgebildet.

Abbildung 2.2.: Feynman-Diagramme für die $t\bar{t}H$ -Produktion aus Hadronenkollisionen in führender Ordnung [BDK⁺02]

3. Experimentelle Grundlagen

Zur Untersuchung der theoretischen Vorhersagen des Standardmodells, werden weltweit Experimente durchgeführt. In Kapitel 3 werden die experimentellen Grundlagen anhand des Compact-Muon-Solenoid-Experiment (CMS, Abschnitt 3.2) am Large-Hadron-Collider (LHC, Abschnitt 3.1) des CERN vorgestellt. Anschließend wird der Ablauf einer Hochenergiephysik-Analyse am Beispiel der $t\bar{t}H$ -Analyse vorgestellt.

3.1. Der Large Hadron Collider (LHC)

Der Large-Hadron-Collider (LHC) ist ein Teilchenbeschleuniger der Europäischen Organisation für Kernforschung (CERN). Er befindet sich in einem 26,7 km langen Tunnel im Grenzgebiet zwischen Frankreich und der Schweiz bei Genf zwischen 45 m und 170 m unter der Erdoberfläche. Es ist der zur Zeit leistungsstärkste Teilchenbeschleuniger der Welt.

Der LHC kann mit Protonen oder Bleiionen betrieben werden. Wenn Protonen genutzt werden, durchlaufen diese zunächst eine Reihe von Vorbeschleunigern in denen sie präpariert werden. Als erstes wird Wasserstoffgas ionisiert. Die so entstehenden Protonen werden im LINAC2, einem Linearbeschleuniger in Bündeln (bunches) mit je $1,1 \cdot 10^{11}$ Protonen auf 50 MeV beschleunigt [O'L12e]. Anschließend werden die Protonenbunches im Proton-Synchrotron-Booster (PSB), im Proton-Synchrotron (PS) und im Super-Proton-Synchrotron (SPS) erst auf 1,4 GeV, dann auf 25 GeV und schließlich auf 450 GeV beschleunigt [O'L12g]. Im LHC selbst werden in zwei Strahlröhren jeweils maximal 2808 bunches in entgegengesetzte Richtungen auf maximal 7 TeV beschleunigt [Lef09]. Die Protonen werden an bestimmten Punkten zur Kollision gebracht.

Es gibt sieben Experimente, die die bei den Kollisionen entstehenden Teilchen untersuchen. Diese Experimente werden von Kollaborationen von Wissenschaftlern aus aller Welt durchgeführt. Die beiden größten sind ATLAS und CMS, die mit ihren universellen Teilchendetektoren alle entstehenden Kollisionsprodukte untersuchen können, während bei dem Schwerionen-Experiment ALICE (A Large Ion Collider Experiment) und dem Large-Hadron-Collider-beauty-Experiment (LHC-B) spezifische Phänomene untersucht werden. ALICE wurde entwickelt um durch Kollisionen von Bleiionen ein Quark-Gluon-Plasma zu erzeugen, was den Bedingungen kurz nach dem Urknall entspricht. LHC-B untersucht Unterschiede zwischen Materie und Antimaterie mithilfe von b-Quarks. Die kleineren Experimente sind TOTEM (Total, elastic and diffractive cross-section measurement) und LHCf (Large Hadron Collider forward), die beide in Richtung des Strahls ("vorwärts") gestreute Ereignisse, also Ereignisse mit sehr kleinem Streuwinkel untersuchen, sowie MoEDAL

(Monopole and Exotics Detector at the LHC), das nach magnetischen Monopolen sucht [O'L12b, O'L12a, O'L12c, O'L12i, O'L12d, O'L12f].

In Abbildung 3.1 ist der schematische Aufbau des LHC mit den Experimenten CMS, ATLAS, LHC-B und ALICE abgebildet. Die Vorbeschleuniger sind dabei außer dem SPS nicht berücksichtigt.

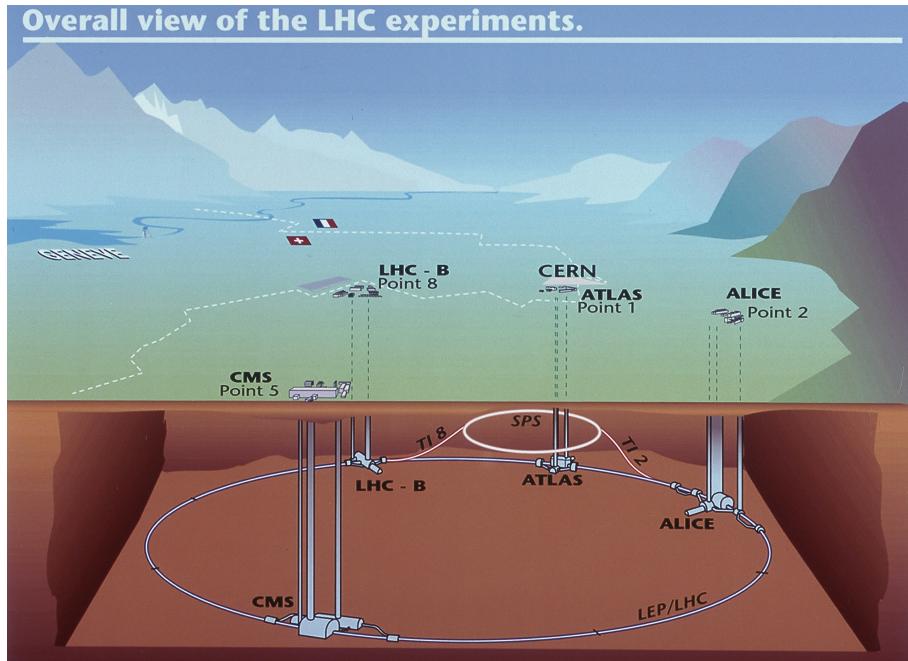


Abbildung 3.1.: Der LHC mit den vier Hauptexperimenten und dem Super-Proton-Synchrotron [Tea99]

3.2. Der Compact-Muon-Solenoid-Detektor (CMS)

Der Compact-Muon-Solenoid (CMS) ist einer der beiden universellen Teilchendetektoren am LHC. Sein Aufbau ist in Abbildung 3.2 zu sehen. Der Detektor ist mit etwa 14 000 Tonnen und einem Durchmesser von 15 Metern auf circa 28,7 Meter Länge recht kompakt gebaut. Die einzelnen Teile des Detektors wurden überirdisch gefertigt und erst in einer Kaverne in ungefähr 100 Metern Tiefe zusammengebaut.

Der CMS-Detektor ist kein einzelner Detektor, sondern besteht aus mehreren Detektoren, die in zylinderförmigen Lagen, ähnlich einer Zwiebel, übereinander geschichtet sind. Herz- und namensgebendes Stück ist der große Magnet (solenoid), der ein bis zu 4 Tesla starkes homogenes Magnetfeld erzeugen kann, durch das geladene Teilchen abgelenkt werden um ihren Impuls bestimmen zu können. Die innerste Lage bildet der Spurdetektor. Er besteht aus Silizium-Pixeldetektoren und Silizium-Streifendetektoren. Sie bestimmen die Spuren der bei der Kollision erzeugten Teilchen. Als nächste Schicht ist ein elektromagnetisches Kalorimeter (ECAL) aus Blei-Wolframat-Kristallen ($PbWO_4$) eingebaut, mit dem Elektronen, Positronen und Photonen nachgewiesen werden können sowie ihre Energie bestimmt werden kann. Das ECAL wird umschlossen von einem hadronischen Kalorimeter (HCAL), das zur Identifizierung und Energiemessung der Hadronen verwendet wird. An diese Lage schließt sich die Magnetspule an. Außerhalb des Magneten befinden sich das eiserne Rückführjoch, das von Myonenkammern zum Nachweis von Myonen durchzogen ist.

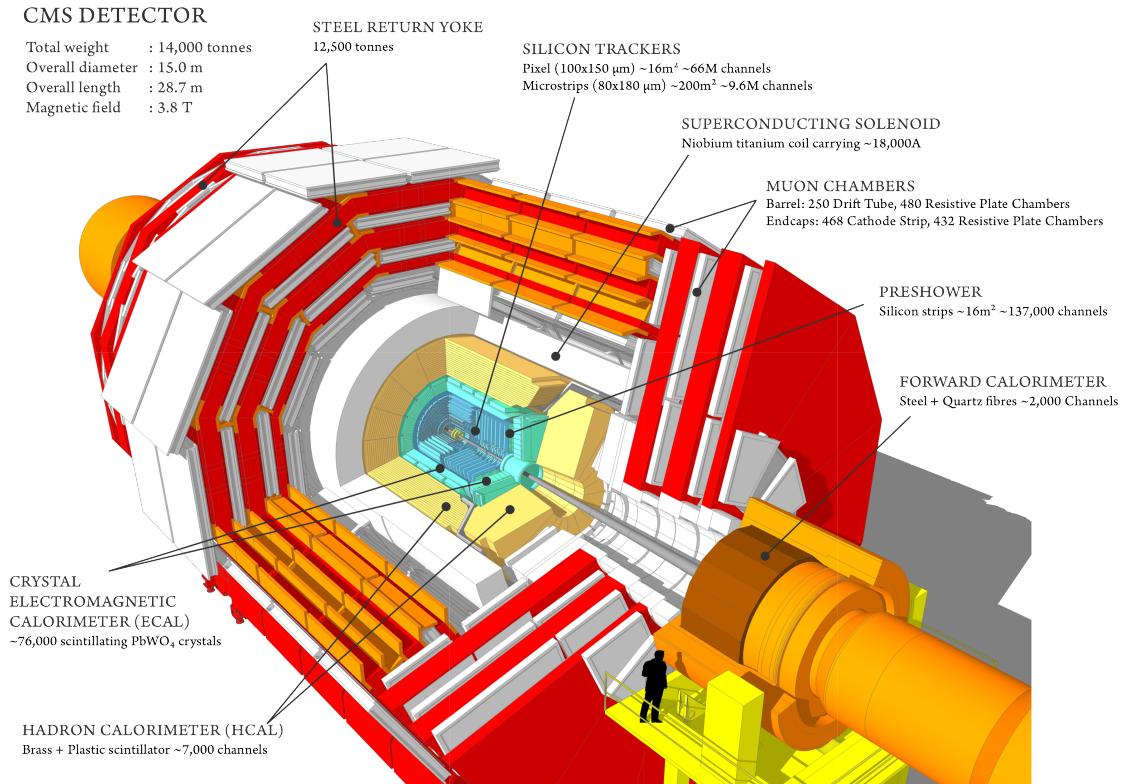


Abbildung 3.2.: Der CMS-Detektor im Querschnitt [CMS11a]

In Abbildung 3.3 ist nochmals ein Ausschnitt des CMS-Detektors mit den Trajektorien der einzelnen Teilchen abgebildet. Im linken oberen Bereich ist der komplette Detektor mit auseinandergefahrenen Segmenten zu sehen. Darunter ist der Querschnitt gezeigt, die hervorgehobene Sektion ist rechts daneben größer dargestellt. Man erkennt die einzelnen Schichten des Detektors. Die rote Linie zeigt beispielhaft das Verhalten eines Elektrons, das während es den Spurdetektor durchläuft vom Magnetfeld abgelenkt wird und schließlich im elektromagnetischen Kalorimeter einen Schauer erzeugt wodurch es detektiert werden kann. In grün sind die Trajektorien von Hadronen gezeigt. Die gestrichelte Linie entspricht einem ungeladenen, die durchgezogene einem positiv geladenen. Beide erzeugen im Hadronenkalorimeter Schauer. Wie ein ungeladenes Hadron wird auch ein Photon (blau gestrichelte Linie) nicht vom Magnetfeld abgelenkt, allerdings erzeugt es bereits im ECAL einen Schauer. Ein Anti-Myon (blaue durchgezogene Linie) interagiert nicht mit den inneren Schichten des Detektors und wird daher erst in den Myonenkammern detektiert.

3.3. $t\bar{t}H$ -Analyse

Der Wirkungsquerschnitt der $t\bar{t}H$ -Produktion ist klein und es gibt viele andere Prozesse, die als Untergrund fungieren. Deshalb ist es anspruchsvoll die Produktionsrate zu messen. Im folgenden soll die $t\bar{t}H$ -Analyse am CMS-Experiment kurz erläutert werden. Einige der verwendeten und darüberhinausführende Informationen finden sich in [K⁺¹⁴].

Der $t\bar{t}H$ -Analyse liegen durch das Standardmodell motivierte Berechnungen zugrunde. Anhand der verschiedenen vorhergesagten Endzustände werden die einzelnen Produktionskanäle in verschiedene Kategorien unterteilt, die sich in wesentlichen Punkten unterscheiden. Mithilfe von Teilchenphysikalischen Methoden wie dem Erkennen von Bottom-Quarks (b-tagging) auf die hier nicht näher eingegangen wird, werden für die verschiedenen Kol-

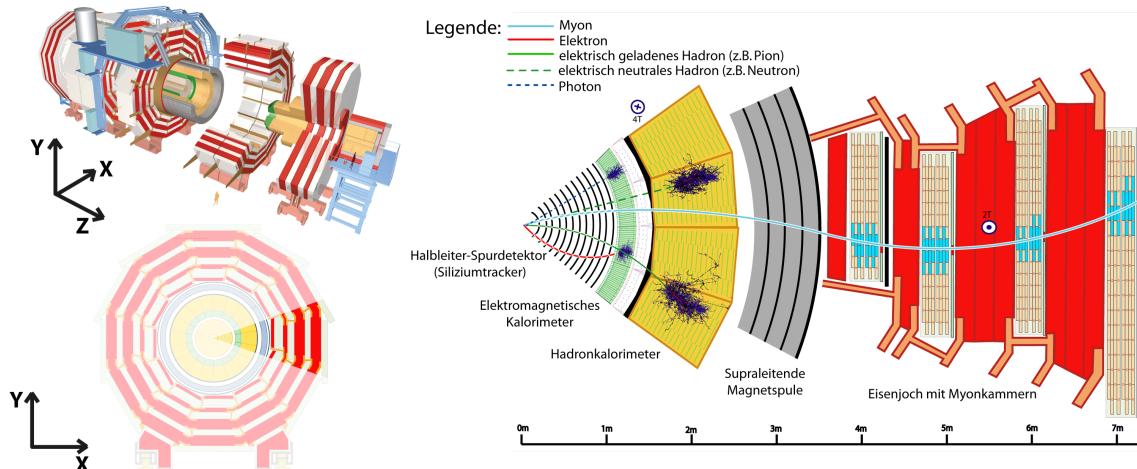


Abbildung 3.3.: Ausschnitt des CMS-Detektors mit verschiedenen Teilchenbahnen [CMS11b]

lisionsprodukte physikalische Eigenschaften bestimmt. Diese können beispielsweise Transversalimpuls oder rekonstruierte Teilchenmasse sein, aber teilweise auch Kombinationen von Eigenschaften mehrerer rekonstruierter Teilchen, etwa Winkelverteilungen zwischen den Teilchen. Da die meisten bei Proton-Proton-Kollisionen produzierten Ereignisse keine $t\bar{t}H$ -Ereignisse sind, müssen sie von den viel häufigeren Untergrundereignissen unterscheiden werden. Hierfür werden multivariate Analysemethoden (MVA-Methoden) verwendet, welche die oben genannten Eigenschaften kombinieren.

Dazu ist es entscheidend die Struktur der aufgenommenen Daten bereits zu kennen. Aus diesem Grund werden mithilfe von Monte-Carlo-Methoden, die auf theoretische Berechnungen sowie Wissen über das Detektorverhalten basieren, Simulationsdaten erstellt. Die gewünschten und gesuchten Ereignisse sind in den Simulationsdaten daher ebenso wie die unerwünschten Untergrunddaten bekannt. Auf die simulierten Daten werden nun multivariate Algorithmen angewandt, die anhand der konstruierten Eigenschaften erlernen die Signal- von den Untergrundereignissen zu unterscheiden. Dadurch lernen die Algorithmen selbstständig (machine learning) verschiedenartige Daten zu unterscheiden.

Zuletzt werden die trainierten Algorithmen auf die im echten Detektor gemessenen Daten angewandt. Die somit erhaltenen Ergebnisse werden dann mit den simulierten Daten verglichen und die Verträglichkeit mit der Vorhersage des Standardmodells geprüft.

4. Algorithmen zur multivariaten Analyse

Multivariate Datenanalyse spielt in der experimentellen Hochenergiephysik eine entscheidende Rolle um die großen gemessenen Datenmengen untersuchen und auswerten zu können. In diesem Kapitel 4 werden zunächst einige Grundlagen der multivariaten Datenanalyse genannt, um später genauer auf verschiedene Algorithmen und ihre Implementationen anhand kleinerer Beispiele einzugehen.

4.1. Grundlagen zur multivariaten Datenanalyse

Datenanalyse bezeichnet statistische Verfahren, mit deren Hilfe aus numerischen Daten Informationen gewonnen werden. Bei multivariaten Analysemethoden werden mehrere Ein-gabegrößen zugleich statistisch untersucht, dadurch ist eine Berechnung sehr aufwändig und somit manuell praktisch nicht zu bewerkstelligen. Mithilfe der zunehmenden Rechenleistung aktueller Computer ist dies jedoch möglich und wird in vielen Bereichen immer wichtiger, beispielsweise im Finanzwesen, bei Studien zum Konsumverhalten, oder der Sprach-, Schrift- und Bilderkennung. Die dazu verwendeten Algorithmen bezeichnet man auch als maschinelles Lernen (machine learning), da mit ihrer Hilfe versucht wird, die zugrunde liegenden Eigenschaften der Daten zu Lernen und Vorhersagen zu treffen. Man unterscheidet zwischen Regression (regression), bei der eine kontinuierliche Ausgangsgröße gesucht wird und Klassifikation (classification), bei der eine diskrete Antwort gesucht wird.[Sut16] Im Fall der $t\bar{t}H$ -Analyse werden Regressionsmodelle verwendet um physikalische Größen wie beispielsweise die Higgsbosonmasse zu rekonstruieren. Bei der Klassifikation wird dagegen versucht, ein Ereignis einer Klasse zuzuordnen, also entweder Signal (signal) oder Untergrund (background). Im Folgenden werden ausschließlich Klassifikationsprobleme behandelt.

Es existieren verschiedene Ansätze zur Klassifikation. Beispiele sind die Stützvektormethode, wobei jedoch die englische Bezeichnung support vector machine (SVM) gebräuchlich ist, Random Forest (RF), was Zufälliger Wald bedeutet und mehrere zufällig erstellte Entscheidungsbäume (Abschnitt 4.2.1) bezeichnet, oder Neuronale Netze. Ein weiteres Beispiel sind verstärkte Entscheidungsbäume (Boosted Decision Trees (BDTs)). Da hiervon verschiedene Implementationen im Kapitel 5 untersucht und getestet werden sollen, werden sie im folgenden Abschnitt 4.2 genauer beschrieben.

4.2. Boosted Decision Trees (BDTs)

Boosted Decision Trees sind eine häufig genutzte Methode der multivariaten Datenanalyse. Im folgenden werden sie anhand eines einfachen Beispiels erklärt. Bei diesem handelt es sich um zwei zweidimensionale Gaußverteilungen die sich überlappen. Der Erwartungswert der Signalverteilung ist bei $X = Y = 0$, der der Untergrundverteilung bei $X = Y = 1$. Beide haben eine Standardabweichung von 1. Eine stellt das Signal dar, die andere dient als Untergrund. In Abbildung ?? ist ein Streudiagramm (scatterplot) der Datenpunkte dargestellt.

ToDo

Die folgenden Abschnitte sind größtenteils an [Has09] angelehnt. (**scatterplot einfügen**)

4.2.1. Entscheidungsbäume

Entscheidungsbäume unterteilen den Bereich der zu klassifizierenden Objekte anhand gerader Schnitte auf dessen Eigenschaften (Variablen) in mehrere Sequenzen. Wieviele dieser Sequenzen ab dem Wurzelknoten erstellt werden, wird durch die Tiefe (depth) des Baumes angegeben. Man unterscheidet zwischen zwei Arten, binären Bäumen mit diskreten Rückgabewerten zur Unterscheidung mehrerer Klassen (classification trees), zum Beispiel Signal und Untergrund, sowie denjenigen mit kontinuierlicher Antwort (regression trees). [Sut16] Eine häufige Implementation von Entscheidungsbäumen ist CART (classification and regression trees), so implementierte Bäume eignen sich sowohl für Klassifikationen als auch für Regressionen.

In Abbildung ?? ist ein Beispiel eines Baumes mit der Tiefe zwei zu sehen. An jedem Knoten werden die Objekte aufgrund ihrer Eigenschaften und Kriterien in Signal und Untergrund unterteilt. Im Wurzelknoten ist der erste diskriminierende Schnitt angegeben. Alle Objekte mit einem Y-Wert größer als 0.33 werden als eher Hintergrundartig eingestuft. In der nächsten Stufe des Baumes werden für jede der beiden zuvor getrennten Mengen Schnitte auf den X-Wert angewendet. In Abbildung 4.1(b) ist die Ausgabe (output) dieses Baumes abgebildet. Das heißt jedem Punkt wird entsprechend seiner X- und Y-Koordinaten ein Wert zugeordnet, der angibt, ob der Punkt eher als Signal oder als Untergrund klassifiziert wurde. Höhere Werte (rote Färbung) werden für signalartige Punkte verwendet, niedrigere (blaue Färbung) für hintergrundartige. Man erkennt deutlich die verschiedenen Schnitte des Baumes.

ToDo

(inkscape tree zeichnen und einfuegen)

Die Trennung ist allerdings noch sehr grob, selbst wenn man die Tiefe des Baumes deutlich erhöht, wie in 4.1(a) mit der Tiefe 100 zu sehen, wird diese nicht deutlich besser. Eine Verbesserung ist beispielsweise möglich, indem man mehrere Entscheidungsbäume so miteinander verknüpft, dass sie zusammen eine starke Klassifikation ermöglichen. Eine dieser Methoden ist das Verstärken von Entscheidungsbäumen (Boosting).

4.2.2. Verstärken von Entscheidungsbäumen (Boosting)

Durch Boosting soll die Güte der Klassifikation eines einzelnen Baumes erhöht werden. Dazu werden mehrere Bäume hintereinander trainiert. Damit diese sich voneinander unterscheiden, werden bei nachfolgenden Bäumen die falsch klassifizierten Ereignisse anders behandelt

Die einfachste Methode ist, das Gewicht jedes falsch klassifizierten Ereignisses auf die gleiche Weise anzupassen. Eine weitere Verbesserung lässt sich erzielen, indem man eine Ausgleichsfunktion (loss function) einführt. Diese ordnet jedem Ereignis ausgehend von der aktuellen BDT-Ausgabe einen Wert zu, der bei richtiger Klassifikation minimal ist. Dadurch ist es möglich, die Gewichte so anzupassen, dass die Ausgleichsfunktion minimiert wird.

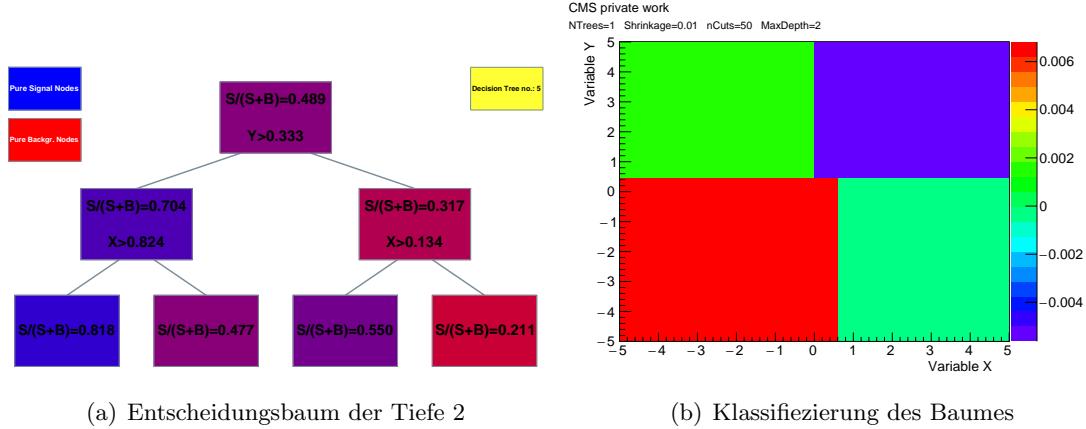


Abbildung 4.1.: Links ist eine schematische Abbildung eines Entscheidungsbaumes der Tiefe 2 abgebildet. X und Y sind die Variablen anhand denen durch Schnitte (Zahlen nach den Variablen) zwischen Untergrund und Signal unterscheiden werden soll. Die rechte Graphik zeigt die Klassifikation die mithilfe des Baumes erstellt wurde.

Wenn die zu trennenden Klassen mit $y = \pm 1$ bezeichnet werden und f die Vorhersage des BDTs ist, also $\text{sign}(f)$ die jeweilige Klasse vorhersagt, so kann man die einfache Methode der Neugewichtung mathematisch mit

$$L = I(\text{sign}(f) \neq y) \quad (4.1)$$

beschreiben. Dabei nimmt die Funktion den Wert 1 an, wenn die Vorhersage nicht der wahren Klasse y entspricht oder 0, wenn die Vorhergesagte mit der wahren Klasse übereinstimmt.

Eine etwas kompliziertere Ausgleichsfunktion berücksichtigt die quadratischen Fehler

$$L = (y - f)^2. \quad (4.2)$$

Diese Funktion bildet eine Parabel um den wahren Wert. Je weiter die Vorhersage davon abweicht, desto höher wird das Ereignis gewichtet. Nachteilig ist dabei, dass sowohl negative Differenzen, als auch positive gleich stark korrigiert werden, also beispielsweise werden für $y = 1$ BDT-Ausgaben von $f = 0$ und $f = 2$ gleich stark korrigiert, obwohl $f = 0$ gar keine Klassifikation zulässt, während $f = 2$ deutlich auf $y = 1$ hinweist. Daher verwendet man meist Funktionen, die negative Abweichungen stärker umgewichtet, wie beispielsweise die exponentielle Ausgleichsfunktion

$$L = \exp(-y \cdot f). \quad (4.3)$$

Diese Ausgleichsfunktion wird beispielsweise von AdaBoost (kurz für Adaptive Boosting) (**quelle f ADA**), dem ersten entwickelten Boosting-Algorithmus verwendet. Problematisch an der exponentiellen Ausgleichsfunktion ist, dass sie nicht so robust gegenüber Ausreißern ist.

To Do

Dieses Problem behebt der Gradient-Boosting-Algorithmus. Dabei wird eine zusätzliche Ausgleichsfunktion definiert, die nicht mithilfe der üblichen Vorgehensweise minimiert werden kann sondern über einen Ansatz der steilsten Abnahme (steepest-descent) minimiert werden muss. Dazu wird zunächst der negative Gradient der Ausgleichsfunktion gebildet.

$$r_m = \left| \frac{\partial L(y, f(x))}{\partial f(x)} \right|_{f=f_{m-1}} \quad (4.4)$$

Diese nennt man auch Pseudo-Residuen. Hierbei bezeichnet m die jeweilige Boosting-Iteration. Danach wird ein zusätzlicher Regressionsbaum trainiert, für den als Zielwerte die Pseudo-Residuen anstatt der Klassen y verwendet werden [HSS⁺07].

Insgesamt erhält man eine BDT-Ausgabe von

$$\hat{f}(x) = f_M(x), \quad (4.5)$$

wobei jede Iteration wie

$$f_m(x) = f_{m-1} + \nu \cdot \sum_{j=1}^J \gamma_{jm} I(x \in R_{jm}) \quad (4.6)$$

berechnet wird.

Dabei ist ν die Lernrate (shrinkage oder learning rate) ein Parameter, der Werte von Null bis Eins annehmen kann. Man kann sie als Parameter auffassen, mit der die Boosting Prozedur kontrolliert werden kann. Je kleiner der Wert ist, desto geringer werden die neu trainierten Bäume gewichtet. Somit kann einem Erlernen von statistischen Fluktuationen entgegengewirkt werden (4.2.3). Die Gesamtanzahl der trainierten Bäume ist M . Bei kleiner Lernrate sollte die Anzahl an Bäumen höher gewählt werden als bei größerer. Die Bezeichnungen dieser und weiterer Optionen mit kurzer Erklärung sind in 5.1 beschrieben.

In Abbildung 4.2 sind zum Vergleich die Rückgabewerte von einem einzelnen Entscheidungsbaum der Tiefe 100 (4.2(a)) sowie diejenigen von BDTs mit zwei (4.2(b)), zehn (4.2(c)) und hundert (4.2(d)) Boosting-Schritten gezeigt. Man erkennt, dass bei diesem einfachen Beispiel die Ausgabe der geboosteten Entscheidungsbäume schon ab zehn Einzelbäumen deutlich glatter wird und so eine stärkere Unterscheidung ermöglichen. Alle diese Klassifikatoren wurden mit dem Gradient-Boosting-Algorithmus von TMVA 4.3.1 erstellt.

4.2.3. Überanpassung (overtraining)

Überanpassung tritt auf, wenn die MVA-Methode zu wenige Freiheitsgrade zur Verfügung hat, weil zu viele Modellparameter an zu wenige Datenpunkte angepasst werden. So werden Statistiken des Trainingsdatensatzes vom Algorithmus gelernt. Dadurch wird zwar die Klassifikation der Trainingsdaten sehr gut, aber die Vorhersage von unbekannten Daten wird deutlich schlechter. Dies bezeichnet man auch als Generalisierungsfehler.

Dies kann beispielsweise auftreten, wenn nur einzelne Ereignisse auf den Knoten eines Entscheidungsbaumes fallen.

ToDo

In Abbildung (**figure scatterplot overtraining or not in training/testing**) ist ein Streudiagramm mit der Klassifikation eines stark überangepassten BDTs im Vergleich zu einer realistischeren Vorhersage abgebildet. In Abb... und Abb... (**BDT output**) sind außerdem die BDT-Ausgaben für unbekannte Testdaten zu sehen.

ToDo

Es gibt verschiedene Ansätze Überanpassung zu vermeiden. Dazu zählt das Verwerfen von Entscheidungsbäumästen mit zu wenigen Einträgen, gewissermaßen das "Abschneiden" (pruning) des Astes am letzten Knoten mit genügend Ereignissen. Außerdem ist es üblich, den Trainingsdatensatz nochmals in zwei Teile zu spalten. Man spricht dann von Trainings- und Validierungsdatensatz. Mithilfe des Trainingsdatensatzes wird der MVA-Algorithmus trainiert, dann wird die Güte des Trainings anhand des Validierungsdatensatzes bestimmt. Sobald man mit der Güte der Klassifikation zufrieden ist, kann man den trainierten Algorithmus nutzen um Vorhersagen für unbekannte Daten, zum Beispiel die experimentell gemessenen Daten zu machen.

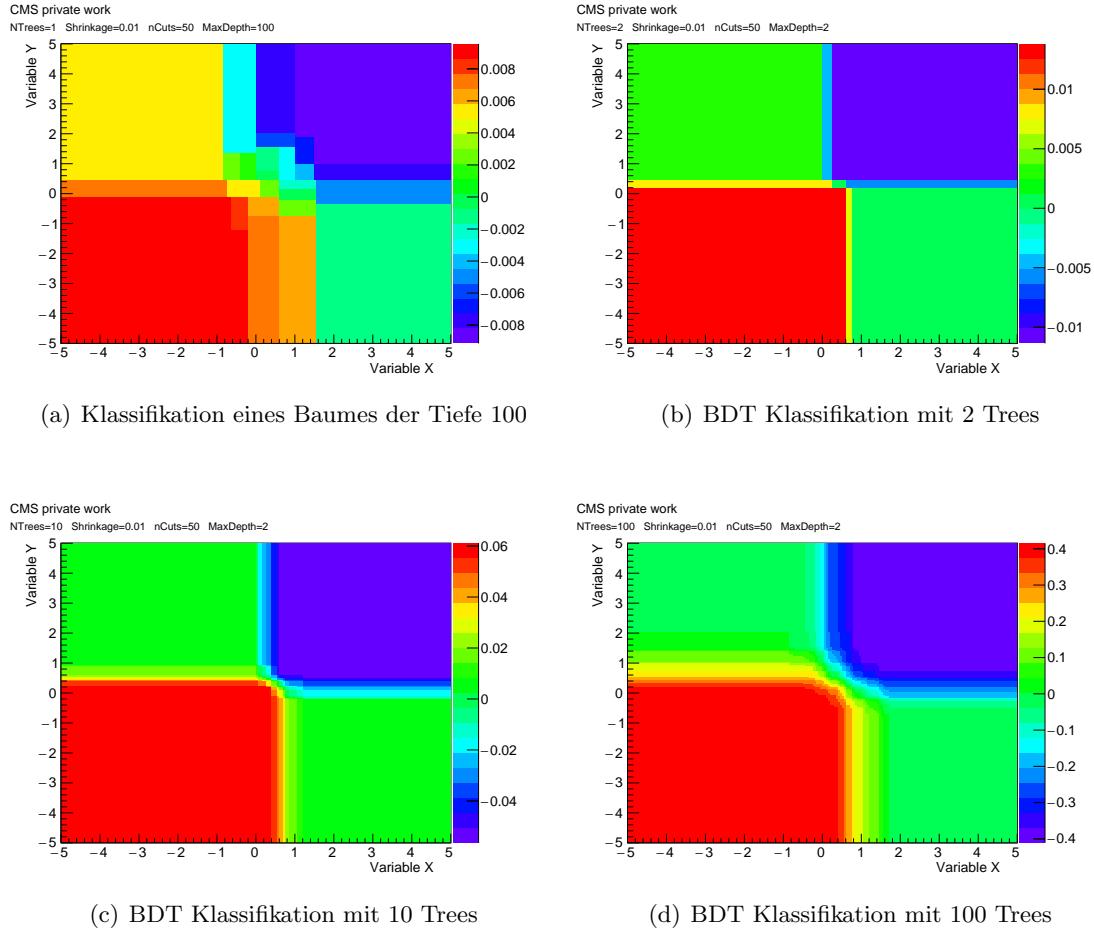


Abbildung 4.2.: (a) zeigt die Klassifikation, die mit einem einzelnen Entscheidungsbaum der Tiefe 100 erreicht wird. Signalartige Regionen sind mit positiver Ausgabe in Rottönen dargestellt, untergrundartige ergeben eine negative Ausgabe und sind in Blautönen dargestellt. In den übrigen Grafiken ist jeweils die Klassifikation eines BDT mit (b) 2 Bäumen, (c) 10 Bäumen und (d) 100 Bäumen zu sehen.

4.2.4. Variieren der Trainingsereignisse (Bagging)

Bagging (aus dem Englischen von bootstrap aggregation abgeleitet) bezeichnet eine Technik die Trainingsereignisse zu variieren. Dabei wird für das Training jedes Baumes eine zufällige Teilmenge der für das Training verwendbaren Daten benutzt. Dadurch unterscheiden sich die einzelnen Bäume voneinander und der Gesamtklassifikator bildet einen Mittelwert der einzelnen schwachen Klassifikatoren. Da Bagging nicht die Güte eines Klassifikators verbessern soll, sondern vor allem zum Stabilisieren der Antwort gedacht ist, handelt es sich beim Bagging nicht um einen Boosting-Algorithmus im klassischen Sinn. [HSS⁺07] Durch Bagging wird der Generalisierungsfehler ebenfalls reduziert. Der Ansatz des stochastischen Gradient-Boosting vereint Boosting und Bagging.

4.3. Verwendete Algorithmen zur multivariaten Analyse

In diesem Abschnitt werden kurz verschiedene Implementationen von multivariaten Algorithmen vorgestellt, die im weiteren Verlauf der Arbeit miteinander verglichen werden.

4.3.1. Toolkit for Multivariate Analysis in ROOT (TMVA)

Das Toolkit für multivariate Datenanalyse in ROOT (TMVA) ist ein Softwarepaket, das ins Analyseframework ROOT integriert ist und eine Vielzahl an multivariaten Analysealgorithmen zur Verfügung stellt. Die TMVA-Algorithmen sind speziell für eine Anwendung in der Hochenergiephysik ausgelegt.

Im Vergleich wird der BDT-Algorithmus mit Gradient-Boosting verwendet. TMVA BDTs nutzen die binomiale Log-Likelihood-Ausgleichsfunktion

$$L(f, y) = \ln(1 + \exp(-2F(x)y)) \quad (4.7)$$

4.3.2. Scikit-Learn – machine learning in python

Scikit-Learn ist ein Python-Programm-Paket. Es bietet ebenfalls eine Reihe von verschiedenen Klassifikatoren. Scikit-Learn stellt eine Großzahl von machine learning Algorithmen zur Verfügung. Im Gegensatz zu TMVA ist das Programm Paket scikit-learn nicht speziell für physikalische Problemstellungen entwickelt, sondern ist auf eine breite Nutzergruppe in allen Bereichen des maschinellen Lernens ausgerichtet.

Der zum Vergleich verwendete GradientBoostingClassifier nutzt die "Deviance"-Ausgleichsfunktion

$$L(y, f) = -2(y \cdot f - \ln(1 + \exp f)) \quad (4.8)$$

4.3.3. Extreme Gradient Boosting (XGBoost)

Extreme Gradient Boosting (XGBoost) [CG16] ist ein Gradient-Boosting-Algorithmus, der sich stark am theoretischen Modell des Gradient-Boosting von Jerome H. Friedman [Fri00] orientiert. Er ist für mehrere Programmierplattformen implementiert, beispielsweise R und python.

Es werden CART trees verwendet, die in jedem Knoten nicht nur die Trennung der Klassen speichern, sondern jedem Knoten auch einen kontinuierlichen Ausgabewert zuweisen um die Vorhersage quantitativer zu machen.

XGBoost verwendet als Ausgleichsfunktion die mittleren Fehlerquadrate, außerdem ist ein Zusatzterm mit Regularisierungsfunktion implementiert. Insgesamt erhält man eine zu minimierende Zielfunktion von

$$F = \sum_{i=1}^n [2(\hat{y}_i^{t-1} - y_i) f_t + f_t^2] + \Omega(f_t) + \text{Konstante}. \quad (4.9)$$

Dabei ist

$$\Omega(f_t) = \gamma T + \frac{1}{2} \lambda \sum_{j_1}^T w_t^2 \quad (4.10)$$

die Komplexität des Baumes mit der Anzahl Endknoten T und den kontinuierlichen Ausgabewerten der Bäume w_t . ($\lambda\gamma$)

ToDo

Die zum Vergleich genutzte Version von XGBoost ist in Python implementiert und wird mithilfe eines Transformationsskriptes für Scikit-Learn aufgerufen.

5. Vergleich der multivariaten Algorithmen

In diesem Kapitel wird zunächst erläutert, anhand welcher Kriterien die verwendeten Algorithmen miteinander verglichen werden können. Anschließend werden verschiedene Datensätze mithilfe der Algorithmen untersucht und die Ergebnisse verglichen.

5.1. Vergleichbarkeit der Algorithmen

Bevor die verschiedenen Implementationen der Algorithmen verglichen werden können, müssen zunächst einige Vergleichskriterien festgelegt werden und überprüft werden inwie weit sich die Parameter der Algorithmen unterscheiden.

In Tabelle 5.1 sind die Einstellungsmöglichkeiten der drei Algorithmen dargestellt.

Tabelle 5.1.: Tabelle mit einstellbaren Parametern der verschiedenen Algorithmen

TMVA	scikit-learn	XGBoost	Funktion
NTrees	n_estimators	n_estimators	Anzahl der Entscheidungsbäume
Shrinkage	learning_rate	learning_rate	Lernrate des Gradient Boosting
MaxDepth	max_depth	max_depth	Tiefe der Entscheidungsbäume
nCuts	—	—	Anzahl an getesteten Schnitten
MinNodeSize	min_samples_leaf	—	Minimalanzahl Ereignisse pro Knoten
BaggedSampleFraction	subsampling	—	Größe der Teilmengen des Trainingsdatensatzes

Die Lernrate, die Anzahl an Entscheidungsbäumen sowie die Tiefe der Bäume haben bei allen drei Algorithmen die gleiche Funktion. Die Anzahl der zu testenden Schnitte ist nur in TMVA regelbar. Dies könnte daran liegen, dass in TMVA viele Berechnungen mithilfe von den in ROOT implementierten Histogrammen durchgeführt werden, während Scikit-Learn Arrays verwendet, die keine Schnitte benötigen sondern eine kontinuierliche Überprüfung der Ausgabe ermöglichen.

Die minimale Anzahl an Ereignissen pro Knoten legt fest, ab wann ein Entscheidungsbaum beschnitten werden soll. In TMVA wird dies über einen Prozentsatz des Trainingsdatensatzes festgelegt, während in Scikit-Learn und XGBoost ein Absolutwert genutzt wird.

Um nur die Größe der zufälligen Teilmenge einzustellen, die jeder Entscheidungsbaum durch Bagging zum Training nutzt, dient die BaggedSampleFraction sowie das subsampling. Beide sind Parameter im Bereich zwischen 0 und 1 und werden mit der Gesamtanzahl der Trainingsereignisse multipliziert um die Anzahl der Ereignisse pro Baum zu erhalten.

Außerdem müssen zunächst Kriterien gefunden werden anhand deren die BDT-Ausgaben miteinander verglichen werden können. In dieser Arbeit werden ROC-Kurve (Abschnitt 5.1.1) sowie das Integral der ROC-Kurven zum Vergleich der Klassifikationsqualität und der Kolmogorov-Smirnoff-Test (Abschnitt ??) zur Untersuchung ob ein Generalisierungsfehler vorliegt verwendet.

5.1.1. ROC-Kurve

Die Receiver Operating Characteristic

Tabelle 5.2.: *Tabelle mit TMVA BDT-Ausgaben*

Trainingszeit in s	ROC-Integral	KS-Test Signal	KS-Test Untergrund
179.72	0.7292	0.20	0.34
176.39	0.7315	0.28	0.97
177.61	0.7329	0.21	0.99
177.02	0.7337	0.22	0.99
174.75	0.7342	0.11	1.0
201.52	0.7294	0.23	0.86
204.64	0.7319	0.23	0.99
195.36	0.7332	0.19	0.99
193.92	0.7339	0.17	0.99
200.38	0.7343	0.17	0.98
211.11	0.7297	0.26	0.78
218.64	0.7321	0.26	0.99
219.69	0.7335	0.20	0.98
226.88	0.7341	0.16	0.99
215.05	0.7344	0.23	0.98
246.53	0.7300	0.23	0.72
243.78	0.7325	0.23	0.99
253.18	0.7337	0.17	0.98
244.71	0.7342	0.18	0.99
248.4	0.7345	0.29	0.94
286.38	0.7302	0.26	0.90
281.77	0.7327	0.21	0.98
286.48	0.7339	0.15	1.0
285.86	0.7344	0.20	1.0
289.66	0.7345	0.20	0.97

5.2. Verwendete Datensätze

5.3. Anwendung und Vergleich der Algorithmen zur ttH Analyse

Literaturverzeichnis

- [B⁺12] O Brüning *et al.*: *LHC design report*. CERN 2004-003, June 27, 2012.
- [BDK⁺02] W. Beenakker, S. Dittmaier, M. Krämer, B. Plümper, M. Spira und P. M. Zerwas: *NLO QCD corrections to t anti- t H production in hadron collisions*. 2002.
- [CG16] Tianqi Chen und Carlos Guestrin: *XGBoost: A Scalable Tree Boosting System*. CoRR, abs/1603.02754, 2016. <http://arxiv.org/abs/1603.02754>.
- [CMS11a] CMS: *CMS detector*, 2011. https://cms-docdb.cern.ch/cgi-bin/PublicDocDB/RetrieveFile?docid=11514&version=1&filename=cms_120918_03.png, besucht: aufgerufen am 15. Juli 2016.
- [CMS11b] CMS: *large transverse and 3-D views, German, png file (slice_white_v3_Deutsch.png, 523.2 kB)*, 2011. https://cms-docdb.cern.ch/cgi-bin/PublicDocDB/RetrieveFile?docid=5697&filename=slice_white_v3_Deutsch.png&version=2, besucht: aufgerufen am 20. Juli 2016.
- [Fri00] Jerome H. Friedman: *Greedy Function Approximation: A Gradient Boosting Machine*. Annals of Statistics, 29:1189–1232, 2000.
- [Has09] Trevor Hastie: *The Elements of Statistical Learning : Data Mining, Inference, and Prediction*, 2009, ISBN 978-0-387-84858-7. <http://dx.doi.org/10.1007/978-0-387-84858-7>.
- [HSS⁺07] Andreas Hoecker, Peter Speckmayer, Joerg Stelzer, Jan Therhaag, Eckhard von Toerne und Helge Voss: *TMVA: Toolkit for Multivariate Data Analysis*. PoS, ACAT:040, 2007.
- [K⁺14] Vardan Khachatryan *et al.*: *Search for the associated production of the Higgs boson with a top-quark pair*. JHEP, 09:087, 2014. [Erratum: JHEP10,106(2014)].
- [Lef09] C Lefevre: *LHC: the guide (English version)*. *Guide du LHC (version anglaise)*. <http://cds.cern.ch/record/1165534>, Feb 2009.
- [O⁺14] K. A. Olive *et al.*: *Review of Particle Physics*. Chin. Phys., C38:090001, 2014.
- [O’L12a] Cian O’Luanaigh: *ALICE*. <div class=“field-headline”> ALICE: A Large Ion Collider Experiment </div>. Feb 2012. <http://cds.cern.ch/record/1997265>.
- [O’L12b] Cian O’Luanaigh: *Experiments*. Jul 2012. <http://cds.cern.ch/record/1997374>.
- [O’L12c] Cian O’Luanaigh: *LHCb*. <div class=“field-headline”> LHCb: The Large Hadron Collider beauty experiment </div>. Feb 2012. <http://cds.cern.ch/record/1997262>.

-
- [O'L12d] Cian O'Luanaigh: *LHCf*. <div class="field-headline"> *LHCf* </div>. Jul 2012. <http://cds.cern.ch/record/1997373>.
- [O'L12e] Cian O'Luanaigh: *Linear accelerator 2*. Sep 2012. <http://cds.cern.ch/record/1997427>.
- [O'L12f] Cian O'Luanaigh: *MOEDAL*. <div class="field-headline"> *The Monopole and Exotics Detector at the LHC* </div>. Nov 2012. <http://cds.cern.ch/record/1997527>.
- [O'L12g] Cian O'Luanaigh: *The accelerator complex*. Jan 2012. <http://cds.cern.ch/record/1997193>.
- [O'L12h] Cian O'Luanaigh: *The Standard Model*. Jan 2012. <http://cds.cern.ch/record/1997201>.
- [O'L12i] Cian O'Luanaigh: *TOTEM*. <div class="field-headline"> *TOTEM: Measuring the proton* </div>. Feb 2012. <http://cds.cern.ch/record/1997259>.
- [Pov14] Bogdan Povh: *Teilchen und Kerne : Eine Einführung in die physikalischen Konzepte*, 2014, ISBN 978-364-23782-2-5. <http://swbplus.bsz-bw.de/bsz39819646xcov.htm> <http://dx.doi.org/10.1007/978-3-642-37822-5>.
- [Sut16] Shan Suthaharan: *Machine Learning Models and Algorithms for Big Data Classification : Thinking with Examples for Effective Learning*, 2016, ISBN 978-1-4899-7641-3. <http://swbplus.bsz-bw.de/bsz455193959cov.htm> <http://dx.doi.org/10.1007/978-1-4899-7641-3>.
- [Tea99] AC Team: *The four main LHC experiments*. <http://cds.cern.ch/record/40525>, Jun 1999.
- [Wik10] Wikipedia: *Graphik Standardmodell der Teilchenphysik*, 2010. <http://en.wikipedia.org/w/index.php?title=Plagiarism&oldid=5139350>, aufgerufen am 5. Juli 2016.

Anhang

A. Anhang 1

ein Bild

Abbildung A.1.: A figure

...