

Moderne Methoden der Datenanalyse

Multivariate Analyse: Higgs Challenge

| | |
|-------------------|---------|
| Lukas Fritz | 1686473 |
| Fabian Leven | 1638446 |
| Ali Deniz Özdemir | 1724032 |
| Lena Salfenmoser | 1723697 |
| Johannes Heizmann | 1725035 |

14. Juli 2016

Inhaltsverzeichnis

| | | |
|----------|---|----------|
| 1 | Einleitung | 3 |
| 2 | Auswahl einer geeigneten Untermenge an Variablen | 3 |
| 2.1 | Vorgehen | 3 |
| 2.2 | Auswertung | 4 |
| 3 | Transformation der Eingabevariablen | 4 |
| 3.1 | Vorgehen | 4 |
| 3.2 | Auswertung | 4 |
| 4 | Getting familiar with the project | 7 |
| 4.1 | Correlation of Variables | 7 |
| 4.2 | Choosing a Classifier | 8 |
| 5 | improvement approach/Methodik | 8 |
| 5.1 | Improving the Classifier | 8 |
| 5.2 | Choosing the right cut | 8 |
| 6 | Conclusion | 8 |

1 Einleitung

Die "Higgs-challenge" wurde im Jahr 2014 von einer Gruppe von Wissenschaftlern der ATLAS-Kollaboration ins Leben gerufen. Dabei sollen die Originaldaten des ATLAS-Experiments auf Zerfälle des Higgs-Bosons in zwei Tau-Leptonen untersucht werden. Hierfür wurden Simulations- und Testdaten zur Verfügung gestellt, mithilfe derer Klassifikationsalgorithmen trainiert werden konnten, um schließlich im Original-Datenset zwischen "Hintergrund" und "Tau-Tau-Zerfall eines Higgs" unterscheiden zu können. Die "challenge" besteht dabei darin, die verschiedenen Methoden des maschinellen Lernens optimal auf die gegebene Situation anzupassen, sodass die Klassifizierung möglichst effizient erfolgen kann.

Das Training sowie die Analyse der Daten erfolgt mit dem "Toolkit for Multivariate Data Analysis with ROOT" (kurz: TMVA), das verschiedene Methoden des maschinellen Lernens bereitstellt. Von diesen waren vier (Maximum Likelihood-Methode, Fisher-Diskriminante, Boosted Decision Trees und neuronales Netzwerk) in einem Template vorgegeben. Die Mess- bzw. Trainingsdaten umfassten 30 Variablen. Da nicht alle gemessenen Werte eine Aussagekraft bezüglich der Entscheidung "Hintergrund" oder "Signal" besitzen, galt es zunächst, aus diesen 30 Variablen eine geeignete Untermenge auszuwählen. Anschließend wurde untersucht, inwiefern eine Transformation der (übrigen) Eingabevariablen, sowie eine Anpassung der Parameter der Algorithmen des maschinellen Lernens die Effizienz der Klassifizierung verbessern konnte. Ausschlaggebendes Kriterium war hierbei jeweils der im Training erzielte approximate median significance-Wert (kurz: AMS-Wert). Sobald ein zufriedenstellendes Ergebnis erreicht wurde, konnte der Klassifizierungsalgorithmus auf die Originaldaten angewandt werden.

2 Auswahl einer geeigneten Untermenge an Variablen

2.1 Vorgehen

Um eine geeignete Untermenge an Variablen für die multivariate Analyse zu finden, haben wir die Relevanz der Variablen, die je nach verwendetem Trainingsalgorithmus (Likelihood, Fisher, BDT, MLP) unterschiedlich ausfallen kann, evaluiert. Hierzu haben wir verschiedene Verfahren angewandt, die im Folgenden beschrieben werden.

Im Zuge der Ausführung der Trainingsalgorithmen des vorgegebenen TMVA-Templates wird von diesem für jede Klassifizierungsmethode eine Rangliste bzgl. der Wichtigkeit der einzelnen Variablen erstellt. Um die Tauglichkeit dieser Rangliste zu überprüfen, betrachteten wir die Trainingsalgorithmen separat und trainierten den Likelihood-, Fisher-, sowie den BDT-Algorithmus jeweils insgesamt 30 mal, wobei entsprechend der vom TMVA-Template ausgegebenen Reihenfolge jeweils eine Variable entfernt wurde (beginnend mit der irrelevantesten). Bei jedem dieser insgesamt 90 Trainings wurde der ausgegebene AMS-Wert für das entsprechende Variablen-Subset protokolliert. Die Entwicklung der AMS-Werte mit abnehmender Variablenzahl sind in Abbildung 1 gezeigt. Es ist deutlich zu erkennen, dass der Likelihood-Algorithmus einen ähnlichen Verlauf wie der Fisher-Algorithmus aufweist. Dies lässt sich dadurch erklären, dass es sich hierbei

um ähnliche Algorithmen handelt. Eine sinnvolle Untermenge an Variablen liese sich demnach mithilfe des Maximums des AMS-Verlaufs bestimmen.

Da das Training des neuronalen Netzwerks mit Abstand die meiste Zeit benötigt, wurde hier ein anderes Verfahren angewandt: Es wurden 30 Trainingsläufe mit je einer entfernten Variablen durchgeführt und die Variablen entsprechend des jeweils erreichten AMS-Werts sortiert. Anschließend führten wir das zuvor beschriebene Verfahren gemäß dieses Rankings durch und erhielten den in Abbildung 1 gezeigten Verlauf der AMS-Werte.

Dieses Verfahren stellte den Startpunkt für eine tiefergehende Analyse der Relevanz der einzelnen Variablen dar: Für jede Methode wurde das Training 30 mal mit jeweils einer aus dem Parameterset entfernten Variable durchgeführt, wobei die irrelevanteste Variable (höchster AMS-Wert) ermittelt wurde. Diese wurde aus dem Variablen-Set entfernt und das Training 29 mal durchgeführt, wobei jeweils eine der übrigen Variablen entfernt wurde. Je nach erzieltm AMS-Wert wurde das Parameter-Subset wiederum um den unwichtigsten Parameter reduziert usw. Da dieses Vorgehen aufgrund der langen Trainingszeit des neuronalen Netzwerks hierfür nicht durchzuführen war, beschränkten wir uns hierbei auf die übrigen drei Verfahren. Insgesamt wurde zur Erstellung der entsprechenden Rankings das Training $3 \cdot 465 = 1395$ mal durchgeführt. Der Verlauf des AMS-Werts in Abhängigkeit der Anzahl verwendeter Parameter ist Abbildung 1 zu entnehmen. Da die AMS-Werte höher liegen als bei dem zuvor beschriebenen Verfahren, ist diese Rangliste als aussagekräftiger zu erachten.

2.2 Auswertung

Die Ergebnisse für den Verlauf der AMS-Werte bzgl. der verschiedenen Ranglisten sind in Abbildung 1 gezeigt. Ein senkrechter Strich markiert jeweils das Maximum, anhand dessen die Auswahl des geeigneten Variablen-Subsets festgelegt wurde: Alle Variablen, die zu diesem Zeitpunkt noch Teil des Trainingsprozesses waren, sind Teil des für die weitere Auswertung der Daten gewählten Parameter-Sets. Welche Variablen dies für die jeweiligen Trainings-Algorithmen sind, ist Tabelle 1 zu entnehmen. An oberster Stelle in der Tabelle stehen dabei die für die Analyse wichtigsten Parameter. Ein waagrechter Strich zeigt die Begrenzung des Variablen-Subsets an.

Eine deutliche Verbesserung des AMS-Werts durch eine Verkleinerung der Variablenanzahl zeigt sich lediglich bei der Likelihood-Methode (siehe Abbildung 1). Da allerdings hiermit auch eine Reduzierung der Trainingszeit einhergeht, was vor allem beim neuronalen Netzwerk aufgrund seiner langen Algorithmenlaufzeit vorteilhaft ist.

3 Transformation der Eingabevariablen

3.1 Vorgehen

Gemäß der Aufgabenstellung in der Template-Datei, experimentierten wir mit den Transformationen „Decorrelate“ (D), „Gauß“ (G), und „Normalise“ (N). Um heraus zu finden,

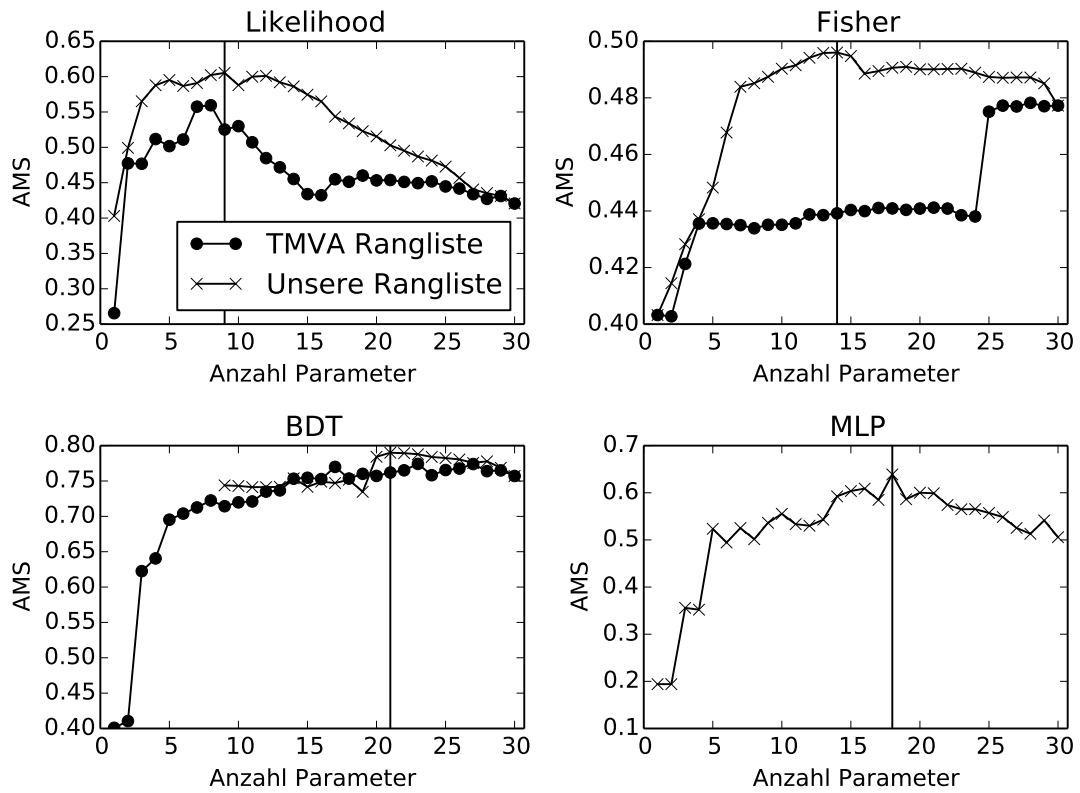


Abbildung 1: AMS über der Anzahl der Variablen.

Tabelle 1: Rangliste der Variablen bzgl. ihrer Relevanz für die Analyse. Die Wichtigkeit der Parameter nimmt von oben nach unten ab. Für jede Methode ist durch einen waagrechten Strich die durch das Maximum des AMS-Verlaufs gegebene Abgrenzung einer geeigneten Auswahl an Variablen gekennzeichnet. Bei der BDT-Methode wurde unter den besten acht Parametern kein Ranking vorgenommen.

| Likelihood | Fisher | BDT | MLP |
|---------------------------|---------------------------|----------------------|-------------------------|
| d_mass_transverse_met_lep | d_mass_transverse_met_lep | - | p_jet_num |
| d_mass_vis | d_pt_ratio_lep_tau | - | d_deltaeta_jet_jet |
| p_tau_pt | p_lep_pt | - | d_mass_MMC |
| d_deltar_tau_lep | d_mass_vis | - | p_jet_leading_pt |
| d_pt_ratio_lep_tau | d_deltar_tau_lep | - | d_mass_transverse_met_l |
| p_met | d_pt_h | - | p_jet_leading_eta |
| p_lep_pt | p_tau_pt | - | d_mass_jet_jet |
| p_lep_eta | d_mass_jet_jet | - | p_jet_leading_phi |
| d_mass_MMC | p_met | d_mass_MMC | d_pt_h |
| — | | | |
| p_jet_leading_phi | d_prodetta_jet_jet | d_lep_eta_centrality | d_mass_vis |
| d_met_phi_centrality | p_jet_subleading_pt | p_jet_subleading_phi | p_jet_subleading_phi |
| p_tau_phi | p_lep_phi | p_tau_pt | d_prodetta_jet_jet |
| p_met_phi | d_mass_MMC | d_mass_vis | p_jet_subleading_pt |
| p_tau_eta | d_deltaeta_jet_jet | p_tau_phi | p_tau_pt |
| — | | | |
| p_jet_subleading_pt | d_lep_eta_centrality | p_met_sumet | p_met_phi |
| d_pt_tot | p_jet_leading_phi | p_lep_pt | p_jet_all_pt |
| p_jet_subleading_eta | p_jet_leading_eta | p_lep_phi | p_met_sumet |
| d_prodetta_jet_jet | d_sum_pt | p_jet_subleading_pt | p_tau_eta |
| — | | | |
| p_jet_subleading_phi | p_jet_subleading_eta | p_jet_subleading_eta | p_met |
| p_met_sumet | p_tau_phi | d_pt_tot | d_met_phi_centrality |
| d_deltaeta_jet_jet | p_met_phi | p_jet_subleading_eta | d_pt_ratio_lep_tau |
| — | | | |
| d_mass_jet_jet | p_jet_leading_pt | p_jet_subleading_phi | p_tau_phi |
| d_lep_eta_centrality | p_tau_eta | p_jet_subleading_pt | p_jet_subleading_eta |
| p_lep_phi | p_met_sumet | d_pt_h | p_lep_phi |
| p_jet_num | p_jet_num | p_met | d_pt_tot |
| d_pt_h | d_pt_tot | d_prodetta_jet_jet | p_lep_eta |
| d_sum_pt | p_lep_eta | p_lep_pt | d_sum_pt |
| p_jet_leading_pt | p_jet_subleading_phi | d_pt_tot | d_lep_eta_centrality |
| p_jet_all_pt | p_jet_all_pt | p_met_phi | p_lep_pt |
| p_jet_leading_eta | d_met_phi_centrality | p_lep_phi | d_deltar_tau_lep |

ob sie einen Einfluss auf das Ergebnis haben, wendeten wir sie in allen Kombinationen und Reihenfolgen auf den Input an und evaluierten den AMS-Wert.

3.2 Auswertung

Die Ergebnisse sind in Abbildung 2 gezeigt. Man sieht, dass es für die Fischer-Methode Sinn macht, die Daten zu dekorrelieren und zu Normalisieren. Die Reihenfolge hat keine Auswirkungen. Bei der BDT-Methode verschlechtern die Transformationen das Ergebnis. Außerdem gab es bei dieser Methode bei manchen Transformationen einen Programmfehler, den wir nicht weiter untersucht haben. In Abbildung 2 sind nur die erfolgreichen Abläufe der BDT-Methode gelistet und deshalb weniger als bei der Fischer-Methode.

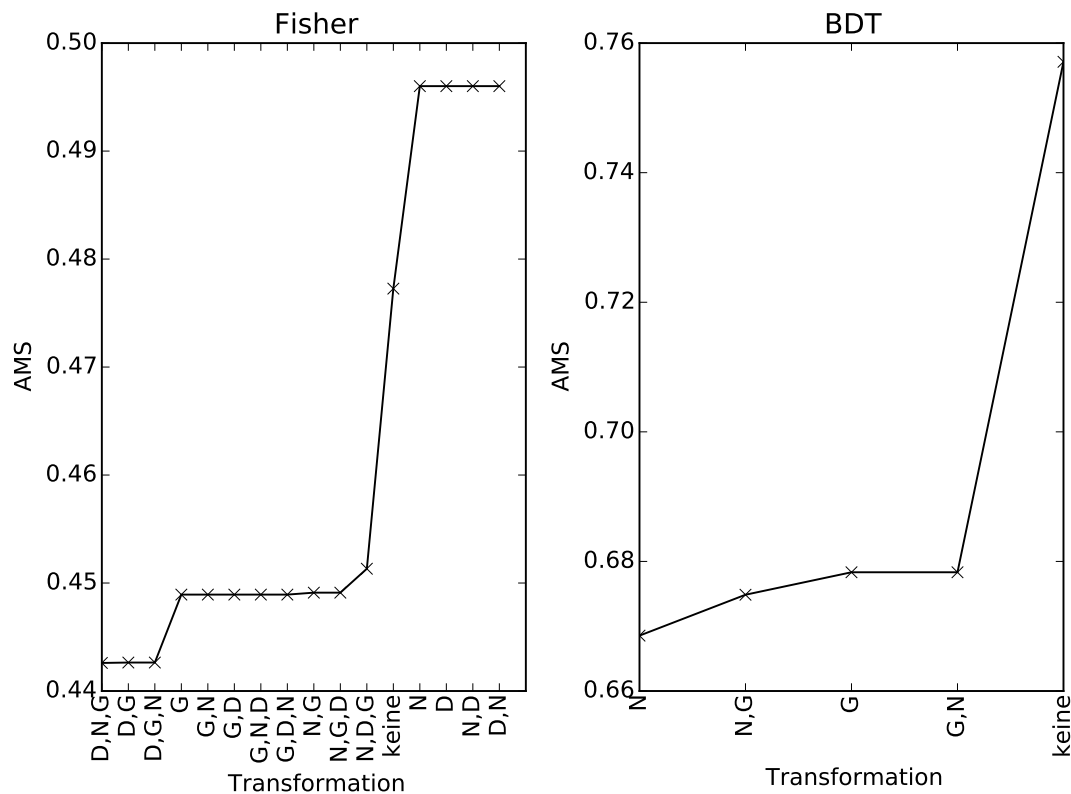


Abbildung 2: AMS über Transformation der Eingabevariablen.

4 Getting familiar with the project

4.1 Correlation of Variables

- determination of differences in correlations of signal and background

- removal of non relevant variables

4.2 Choosing a Classifier

5 improvement approach/Methodik

5.1 Improving the Classifier

5.2 Choosing the right cut

6 Conclusion