



EPIC DATA CONSULTING

25.01.2022

35 – 40 Minuten Pitch

Unser Team für Sie

Nicolas Mahn, 263020



Jan-Hendrik Hausner, 261056



Artur Erfurt, 262311



Kevin Hilzinger, 263023



Patrick Jäckle, 263032



Niko Kauz, 263616



„Information is the oil of the
21st century, and analytics is
the combustion engine“

—Peter Sondergaard, Gartner Research

SmartBuild

– Ziel: Produktion optimieren



Ist es möglich anhand der **Eingangsdaten** die Eigenschaften der **Endprodukte** vorherzusagen, um somit die **Produktion** zu optimieren?

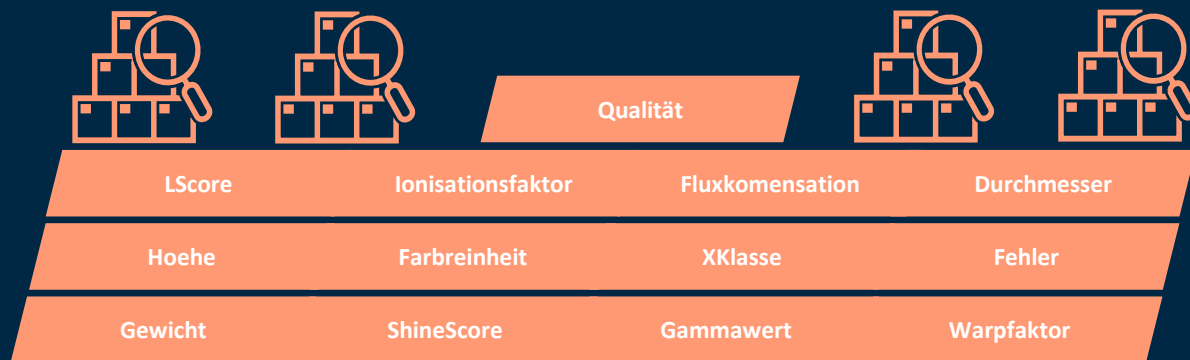
➡ Optimierung durch Machine Learning

Epic Data in der Auswertung

10.000 Datensätze

13 Parameter

Die Nummer 1 im Wettbewerb



Was die Daten in der Analyse sagen

Qualität maximieren

93% Genauigkeit bei der
Bestimmung der Qualität des
Endproduktes

LScore erhöhen

Rohmaterialien mit einem
Gewicht über 250 Gramm
maximieren LScore gut/mittel

Ausschuss minimieren

Rohmaterialien zw. 250
und 300 Gramm verringern
den Ausschuss auf ca. 3%

Vielseitige Analysen

Viele, teils gute, Vorhersagen
über Merkmale wie
Gammawert, Farbreinheit usw.

Weitere Daten

Zusätzliche Daten und
Merkmale können Modelle
und Prognose optimieren

Schnelle Integration

Mit R lassen sich Ergebnisse,
Visualisierung und Reports leicht in
die Unternehmens-IT integrieren

Agenda



■ Business



Technologie

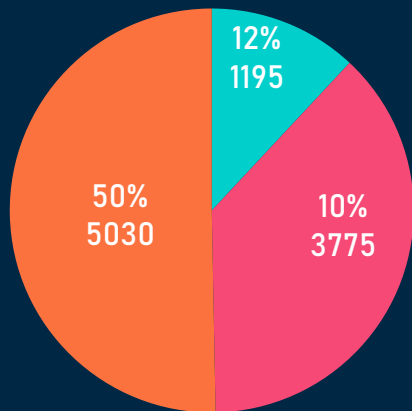


Business

LScore im Visier – Decision Tree



93.25% korrekte Vorhersage mit unserem Modell
 Gewicht größer als 250 Gramm führt zu guten und mittleren Ergebnissen.



gut

mittel

schlecht

Tatsächliche Zuordnung

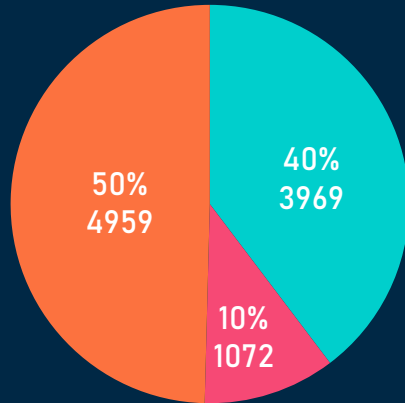
Vorhergesagt

| | Tatsächliche Zuordnung | | |
|----------|------------------------|--------|----------|
| | Gut | Mittel | Schlecht |
| Gut | 160 | 0 | 1 |
| Mittel | 41 | 731 | 61 |
| Schlecht | 12 | 20 | 974 |

Fehler unter der Lupe – AdaBoost



78.65% korrekte Vorhersage mit unserem Modell
Gewicht zwischen 250 und 300 Gramm minimiert den Ausschuss



■ Ausschuss ■ Fehler im Toleranzbereich ■ Nein

Vorhergesagt

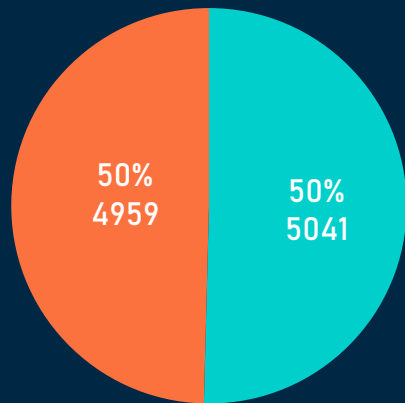
| | Tatsächliche Zuordnung | | |
|--------------|------------------------|--------------|------|
| | Ausschuss | Fehler im T. | Nein |
| Ausschuss | 775 | 204 | 187 |
| Fehler im T. | 0 | 0 | 0 |
| Nein | 27 | 9 | 798 |

Was hat Fehler? – AdaBoost



88.8% korrekte Vorhersage mit unserem Modell

Ob es zu einem Fehler kommt, kann sehr gut vorhergesagt werden



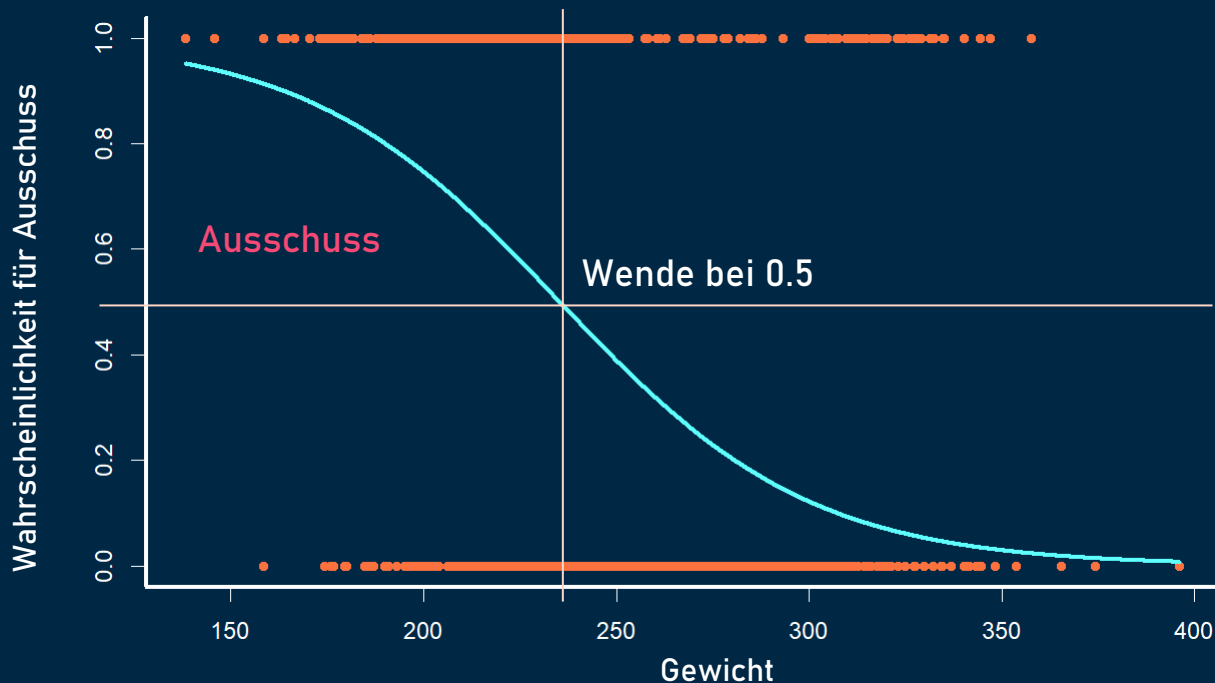
■ Fehler

■ Kein Fehler

| | | Tatsächliche Zuordnung | |
|--------------|--------|------------------------|------|
| | | Fehler | Nein |
| Vorhergesagt | Fehler | 979 | 187 |
| | Nein | 36 | 798 |

Ausschuss – Logistische Regression

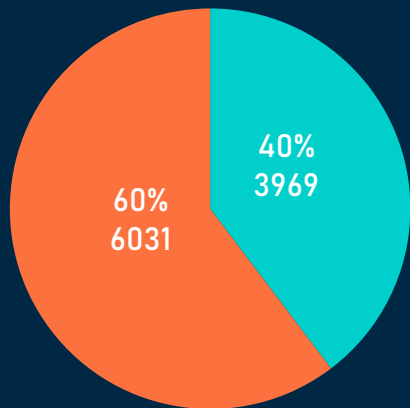
➔ **74%** korrekte Vorhersage mit unserem Model
jedoch Pauschalisierung bei Gewichten von >250 Gramm



Erkenntnis:
Eher Materialien
verwenden, die über
250 Gramm wiegen

Ausschuss – Decision Tree

➔ **78.85%** der Fälle wurden korrekt vorhergesagt
Gewicht zwischen 250 und 300 Gramm verringert den Ausschuss enorm



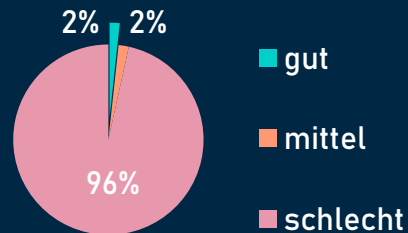
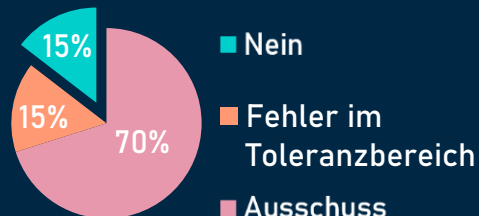
■ Ausschuss ■ Nein

| | | Tatsächliche Zuordnung | |
|--------------|------|------------------------|------|
| | | Ja | Nein |
| Vorhergesagt | Ja | 692 | 313 |
| | Nein | 110 | 885 |

Tradeoff Fehler «» LScore

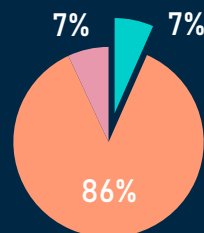
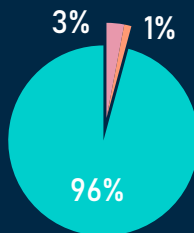
Gewicht
< 250
Gramm

führt zu schlechten
LScore (96%) und
hohen Ausschuss
(70%)



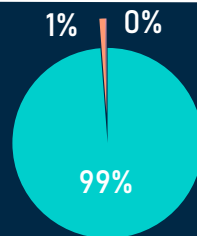
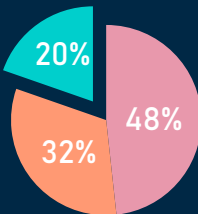
Gewicht
250 – 300
Gramm

führt zu wenigen,
keinen Fehlern (97%),
LScore ist mittel (86%)
und gut (7%)



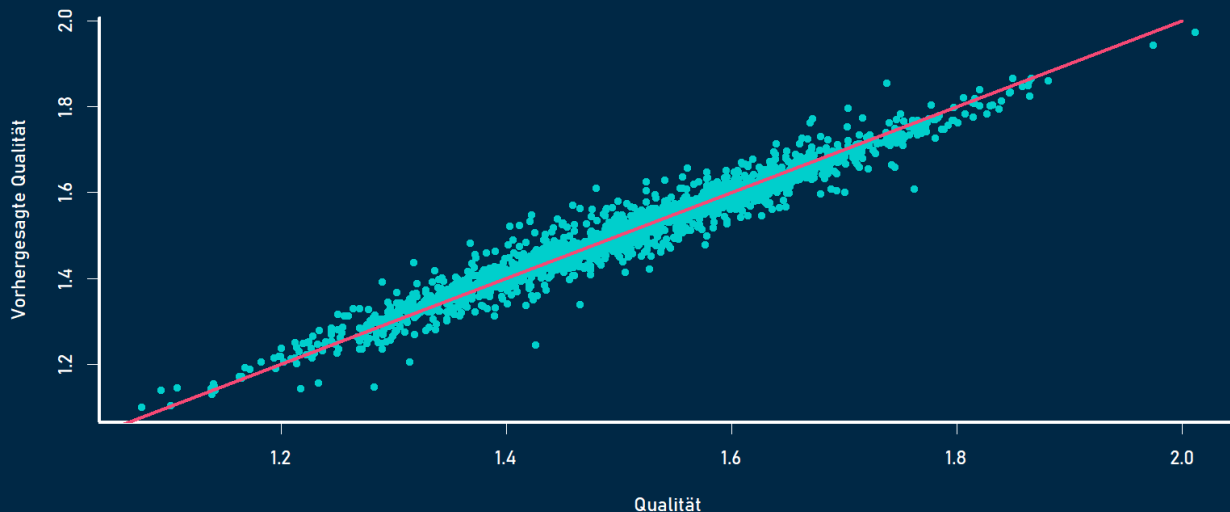
Gewicht
> 300
Gramm

Mindert Anteil von
Nicht-Ausschuss
(52%), jedoch ist
LScore gut/mittel
(~100%)



Qualität evaluiert

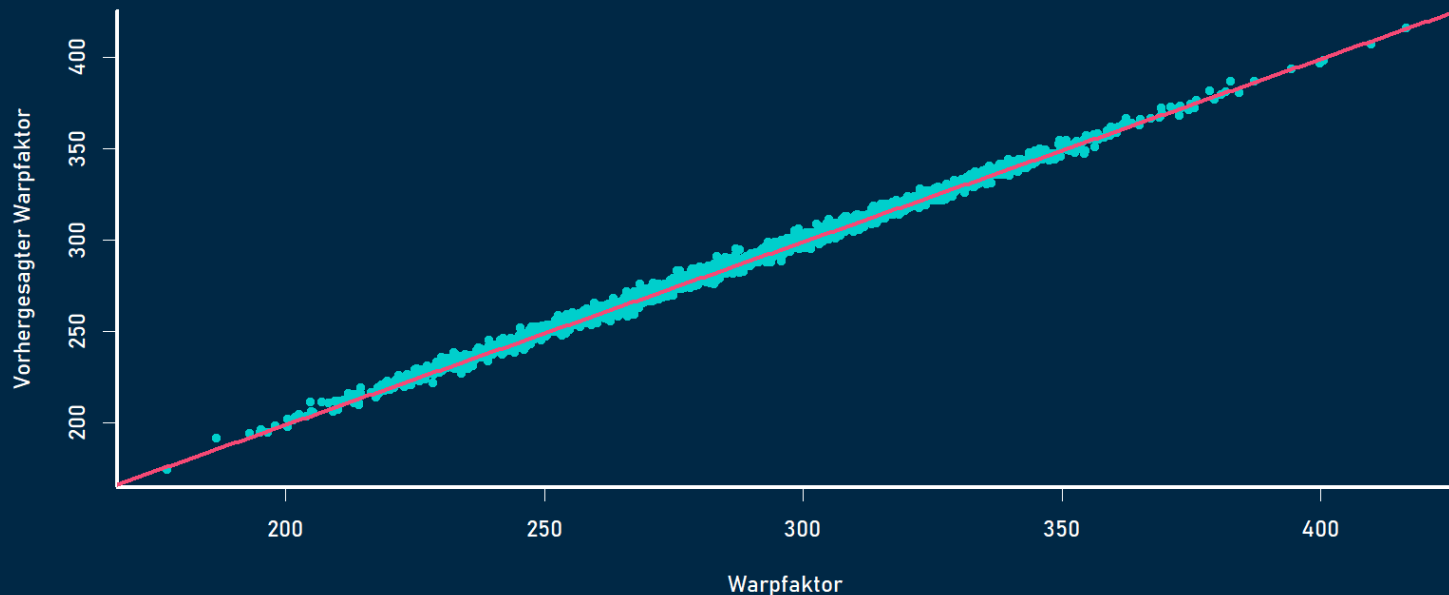
➔ 93.45% korrekte Vorhersage mit unserem Modell, eingestuft in Ist-Nicht-Qualitativ < 1.5 < Ist-Qualitativ



➔ Qualität abhängig von Eingangsattributen.
Durchschnittliche Abweichung: 0.02

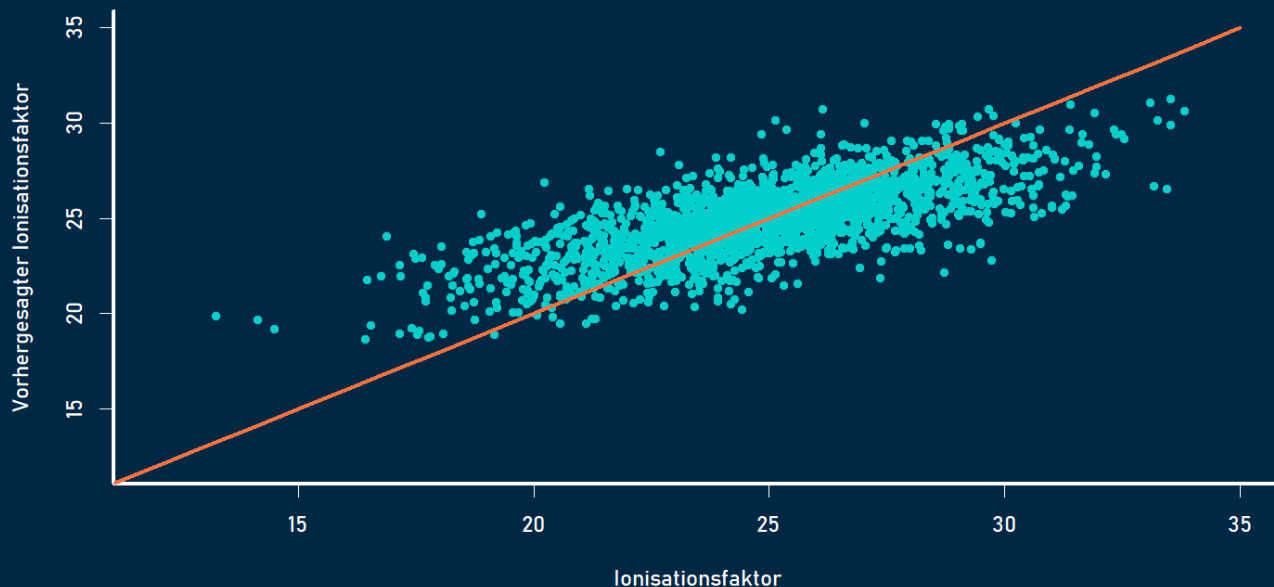
Warpfaktor im Visier

➔ Warpfaktor abhängig vom Gewicht
Durchschnittliche Abweichung: 1.861
Durchschnittliche Abweichung in Baseline: 28.391



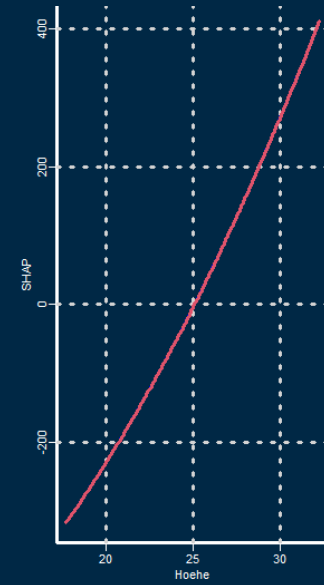
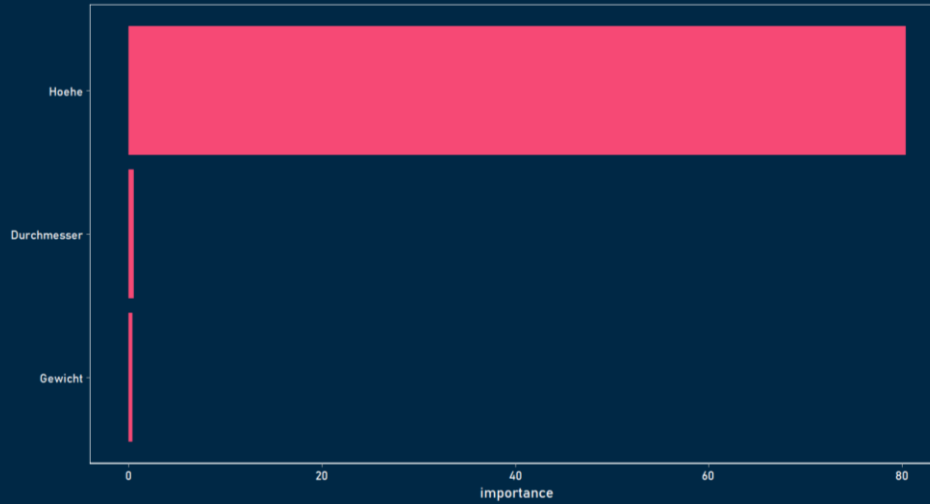
Ionisationsfaktor analysiert

➔ Ionisationsfaktor abhängig von der Höhe
Durchschnittliche Abweichung: 1.607



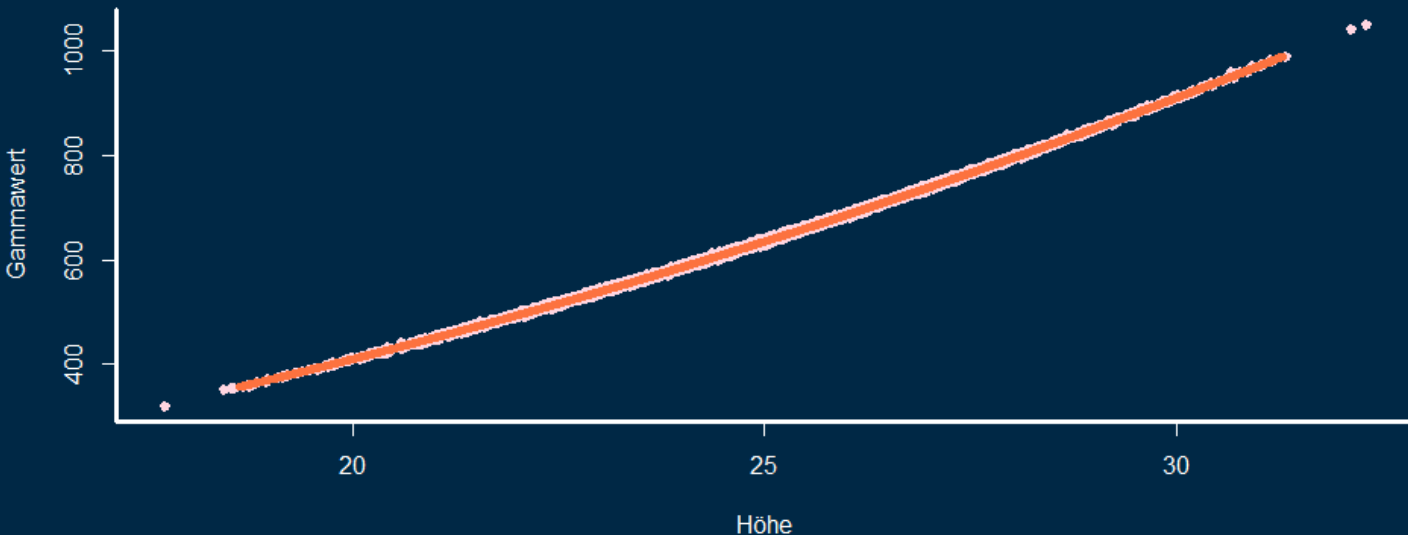
Gammawert – SHAP Value

➔ Gammawert abhängig von der Höhe
Regression Polynom zweiten Grades



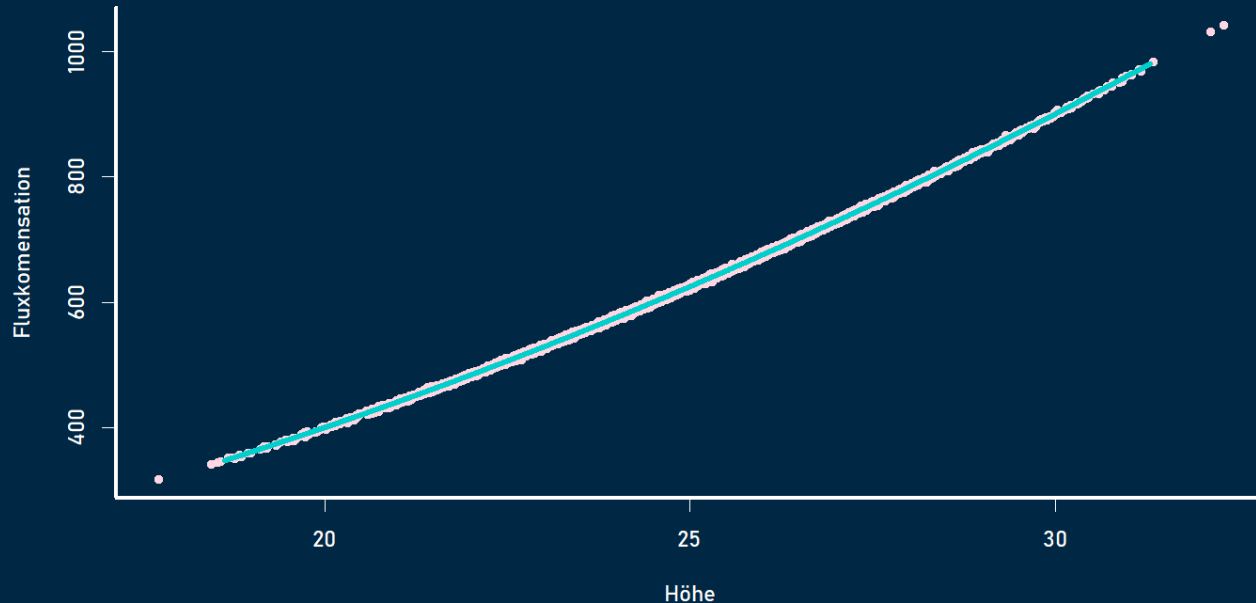
Gammawert - Polynomiale Regression

➔ Durchschnittliche Abweichung im Model: **1.767**
Durchschnittliche Abweichung in Baseline: **80.508**



Fluxkompensation- Poly. Regression

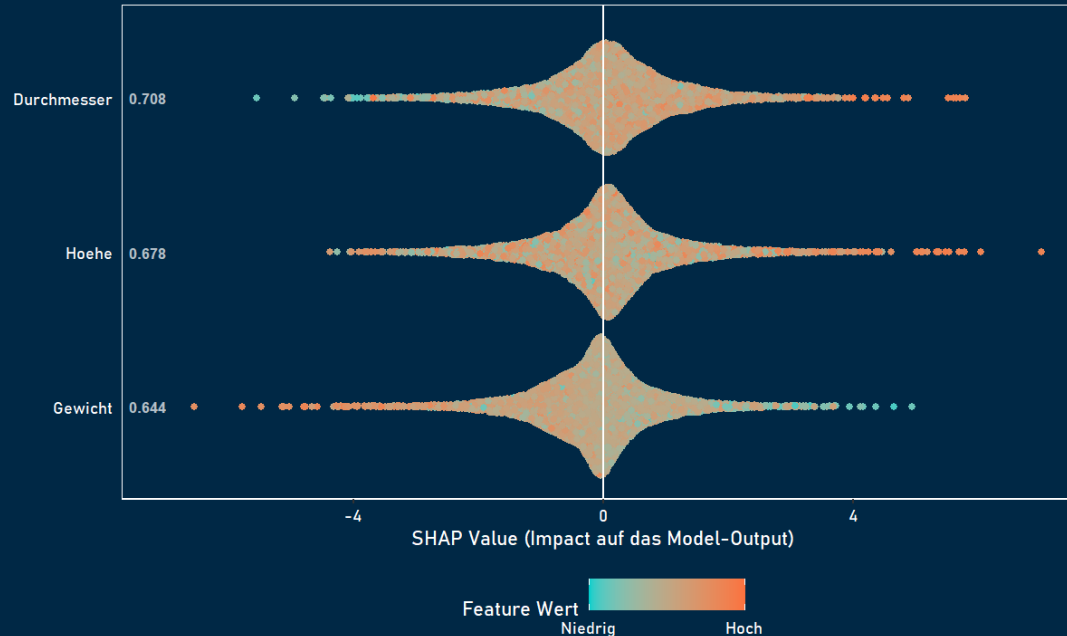
➔ Fluxkompensation abhängig von Höhe.
Durchschnittliche Abweichung: 1.619



ShineScore - Lineare Regression



ShineScore nicht abhängig von Eingangsattributen
Unterschied der Baseline- und Testdaten-Error < 0.001



ShineScore – Neuronale Netze

➔ **Starke Abweichungen** bei den Vorhersagen
Baseline nur minimal schlechter

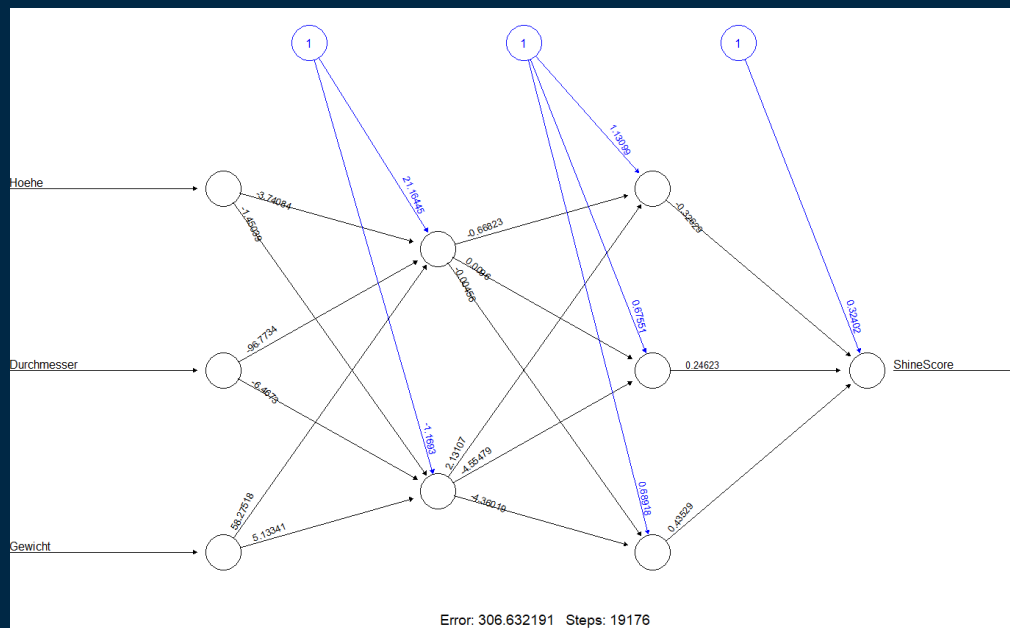
5.037 Abweichung

Durchschnittlich bei den Vorhersagen

Abweichung in der Baseline beträgt
5.038

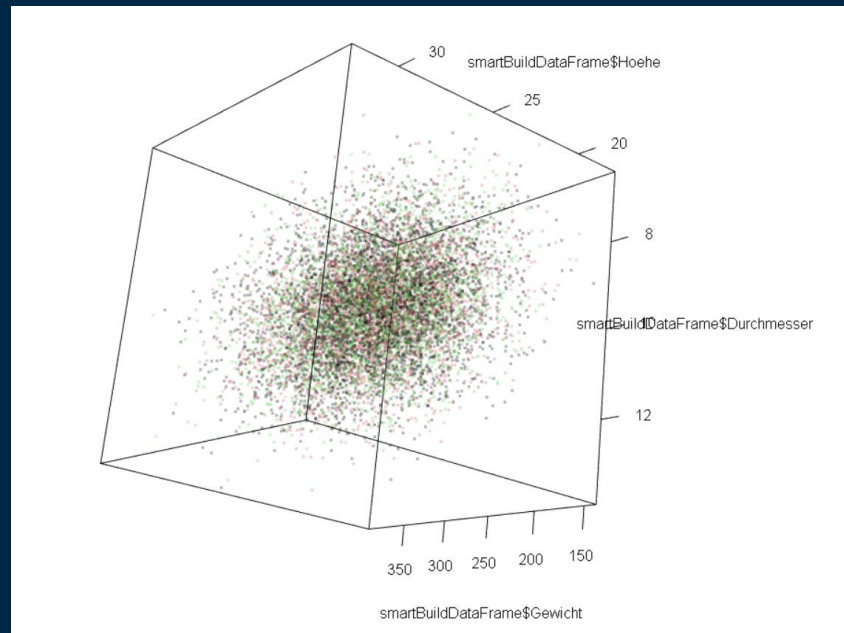
Bei einem Wertebereich zwischen
-0.409 und 20.436

Hängt nicht von den Eingangswerten ab



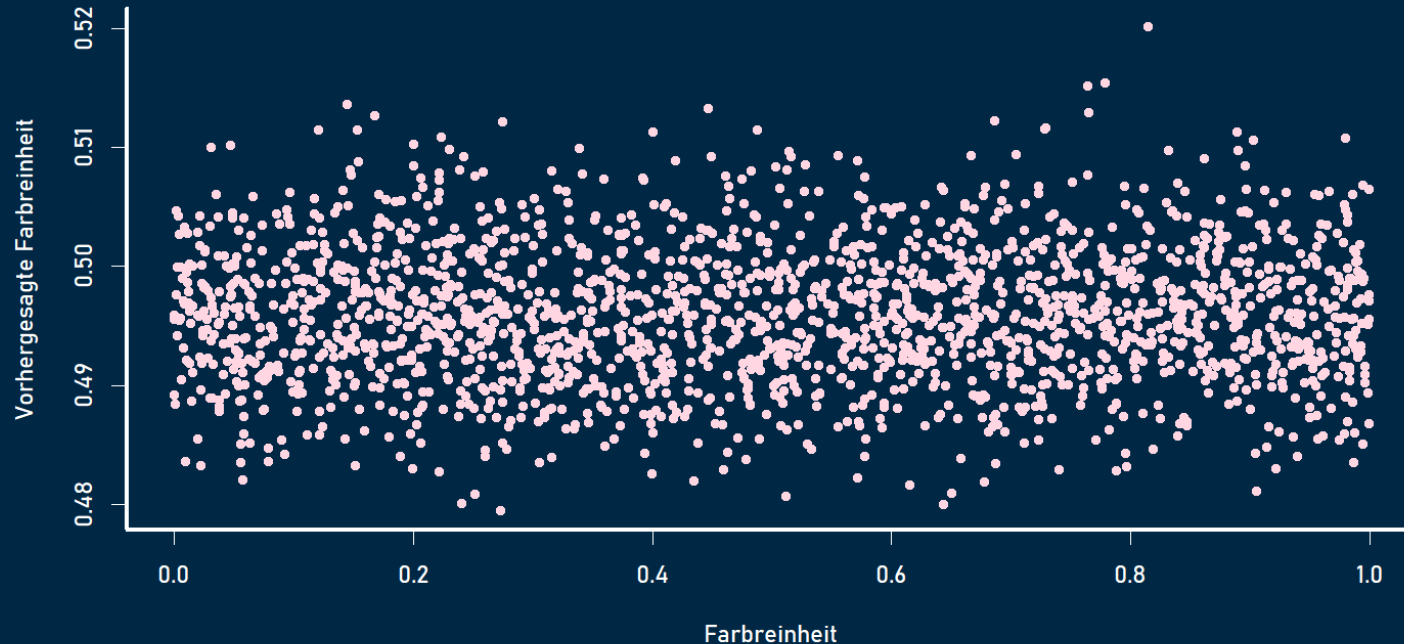
Clustering unter der Lupe

➔ Für sämtliche Label keine Clustergruppen möglich
Hoher Davies Bouldin Index aller Cluster



Farbreinheit visualisiert – Lineare R.

➔ Farbreinheit nicht abhängig von Eingangsattributen



XKlasse evaluiert – RandomForest



25.2% korrekte Vorhersage mit unserem Modell
Es konnten keine Muster erkannt werden

| | | Tatsächliche Zuordnung | | | |
|--------------|-----|------------------------|-----|-----|-----|
| | | I | II | III | IV |
| Vorhergesagt | I | 124 | 131 | 135 | 136 |
| | II | 136 | 154 | 151 | 133 |
| | III | 89 | 113 | 106 | 134 |
| | IV | 116 | 124 | 108 | 110 |

XKlasse nicht
abhängig von den
Eingangsattributen

Unsere Ergebnisse

Gute Vorhersagen

- Qualität
- Ist Qualitativ
- Hat Fehler
- Ausschuss (DT)
- LScore
- Warpfaktor
- Gammawert
- Fluxkomen.

Mittlere Vorhersagen

- Fehler
- Ionisationsfaktor
- Ausschuss (LogR)

Schlechte Vorhersagen

- XKlasse
- ShineScore
- Farbreinheit



Technologie



Die Technologie R



Starke
Funktionen



Vielseitige
Visualisierungen



Einfache
Nutzbarkeit



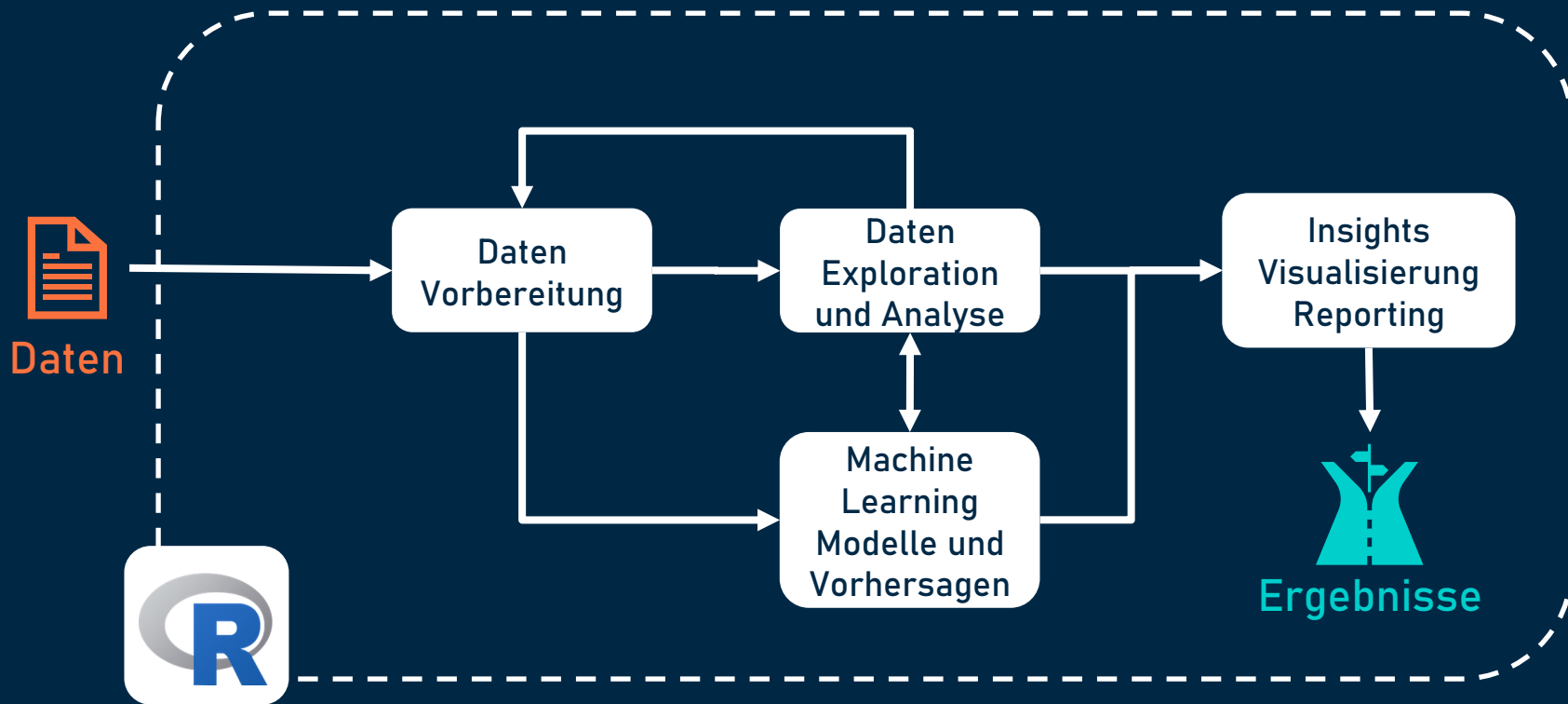
Sparsame
Ressourcennutzung



Stark
etabliert



Der Data Science Prozess





Die Daten genauer betrachtet

Eingangseigenschaften

| id | Durchmesser | Hoehe | Gewicht | Qualitaet | Fehler | Farbreinheit | ShineScore | |
|-------------------|------------------|----------------|---------------|---------------|--------------------------------|-------------------|-----------------|-----------|
| Min. : 1 | Min. : 6.329 | Min. :17.72 | Min. :138.3 | Min. :1.011 | Ausschuss :3969 | Min. :0.0001537 | Min. : -0.4087 | |
| 1st Qu.: 2501 | 1st Qu.: 9.327 | 1st Qu.:23.65 | 1st Qu.:228.7 | 1st Qu.:1.419 | Fehler im Toleranzbereich:1072 | 1st Qu.:0.2428698 | 1st Qu.: 4.8421 | |
| Median : 5000 | Median : 9.984 | Median :25.05 | Median :250.8 | Median :1.505 | nein :4959 | Median :0.4991127 | Median : 9.9733 | |
| Mean : 5000 | Mean : 9.993 | Mean :25.02 | Mean :251.9 | Mean :1.506 | | Mean :0.4968140 | Mean : 9.9371 | |
| 3rd Qu.: 7500 | 3rd Qu.:10.678 | 3rd Qu.:26.36 | 3rd Qu.:274.2 | 3rd Qu.:1.595 | | 3rd Qu.:0.7468785 | 3rd Qu.:14.9206 | |
| Max. :10000 | Max. :13.810 | Max. :32.30 | Max. :396.1 | Max. :2.011 | | Max. :0.9999420 | Max. :20.4356 | |
| Ionisationsfaktor | Fluxkompensation | Gammawert | XKlasse | Warpfaktor | Lscore | Ausschuss | IstQualitativ | HatFehler |
| Min. :13.25 | Min. : 317.1 | Min. : 320.8 | I :2501 | Min. :169.0 | gut :1195 | Ja :3969 | Ja :5158 | Ja :5041 |
| 1st Qu.:23.08 | 1st Qu.: 559.2 | 1st Qu.: 569.2 | II :2537 | 1st Qu.:262.3 | mittel :3775 | Nein:6031 | Nein:4842 | Nein:4959 |
| Median :25.00 | Median : 627.4 | Median : 637.3 | III:2490 | Median :285.8 | schlecht:5030 | | | |
| Mean :25.00 | Mean : 629.9 | Mean : 639.8 | IV :2472 | Mean :286.9 | | | | |
| 3rd Qu.:26.92 | 3rd Qu.: 695.5 | 3rd Qu.: 705.1 | | 3rd Qu.:310.5 | | | | |
| Max. :35.28 | Max. :1041.9 | Max. :1049.3 | | Max. :436.6 | | | | |

Abgeleitete Attribute

IstQualitativ:

Hier wird ein Endprodukt nach seiner Qualität unterschieden mit 1.5 als Trennung

Ausschuss:

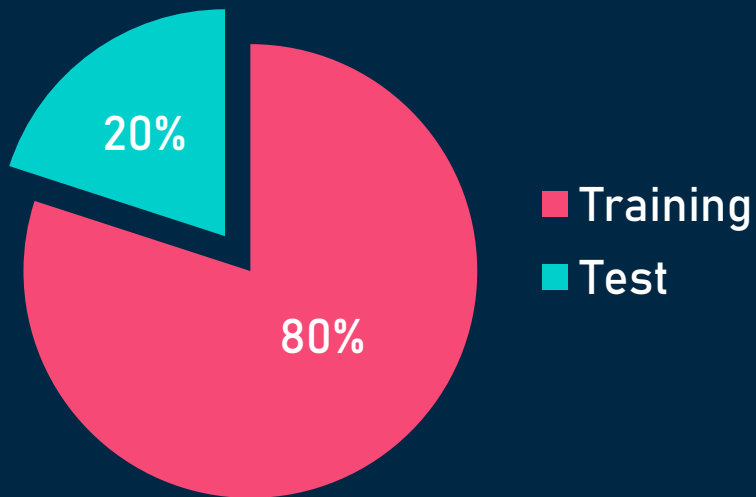
Hier wird aus Fehler Ausschuss gegen keine Fehler und Fehler im Toleranzbereich gestellt.

HatFehler:

Hier wird mit Fehler geprüft ob ein Fehler vorliegt (auch im Toleranzbereich) oder nicht



Aufteilung der Daten



Die Daten werden aufgeteilt in...

80% Trainings-Daten

Um die Modelle zu trainieren

20% Test-Daten

Um die Aussagekraft und Güte der Modelle zu testen und evaluieren

Supervised Learning

Die Trainings-Daten besitzen Labels, über wir analysieren und vorhersagen wollen

Mit einem Seed wird sichergestellt, dass wir immer mit den gleichen zufälligen Daten arbeiten



Machine Learning Modelle



Lineare
Regression



Neuronale
Netze

Qualitaet
Min.: 1.011
Median: 1.5
Mean: 1.5
Max.: 2.0

Regression

Vorhersage eines numerischen Wertes
u.a. Qualität

Klassifikation

Vorhersage einer Kategorie
u.a. Fehler

Fehler
A: 3969
FiT: 1072
nein: 4959



Decision
Tree



Random
Forest



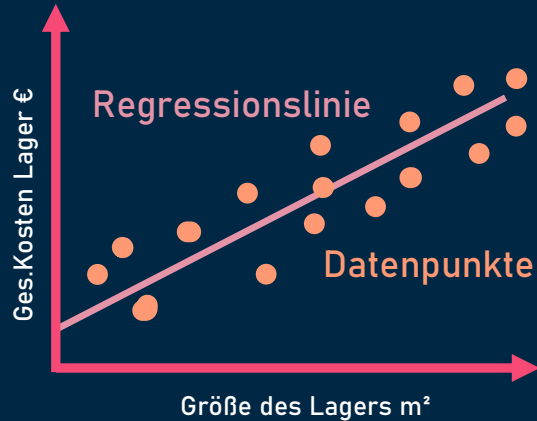
AdaBoost



Logistische
Regression



Lineare Regression



Vorhersage einer abhängigen numerischen Variablen mit Hilfe von unabhängigen Variablen mittels einer Regressionslinie

Stellt linearen Zusammenhang zwischen den Variablen dar

Je stärker die Korrelation der Variablen, desto genauer werden die Vorhersagen

Erweiterung mit der Multilinearen und Polynomialen Regression

Vorteile

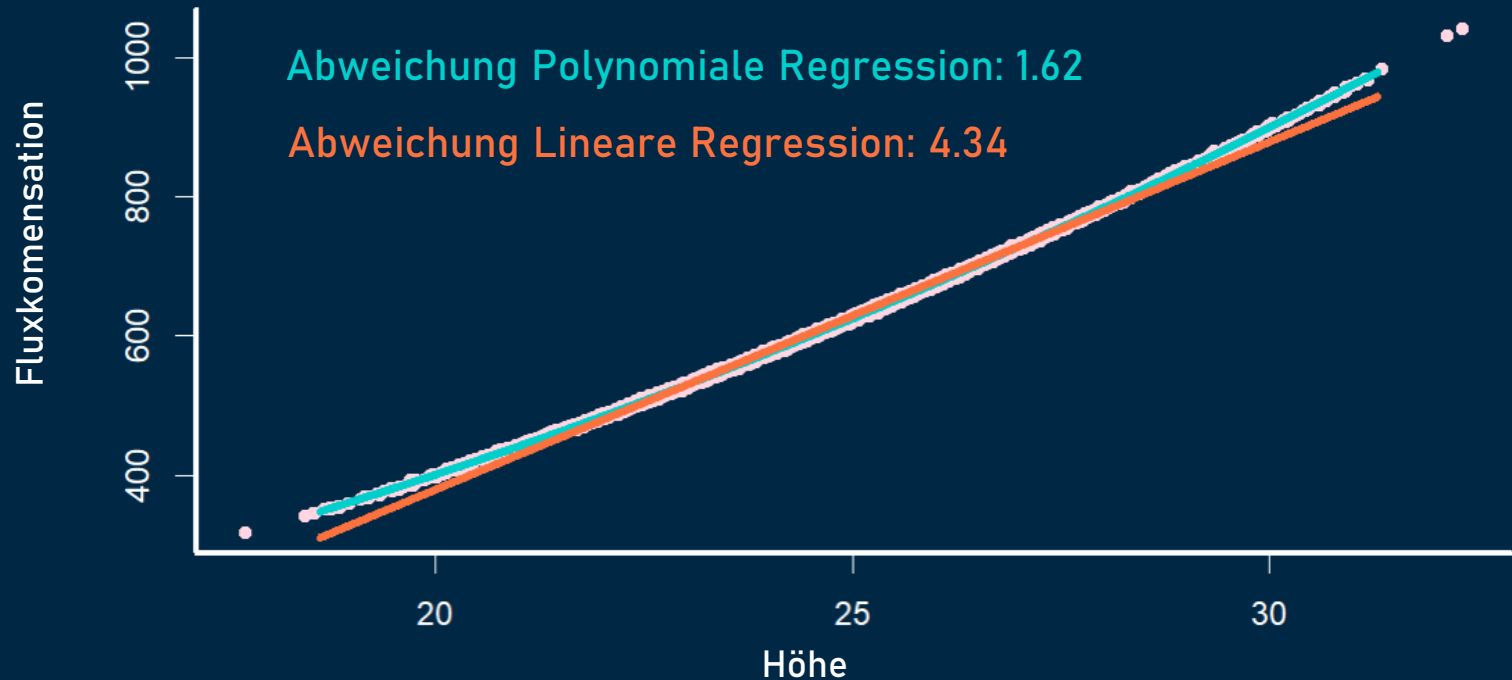
- Einfach und zuverlässig
- Wenig fehleranfällig
- Gute Visualisierung

Nachteile

- Nicht genau bei nicht-linearen Beziehungen
- Abweichung und Varianz
- Reagiert auf Ausreißer

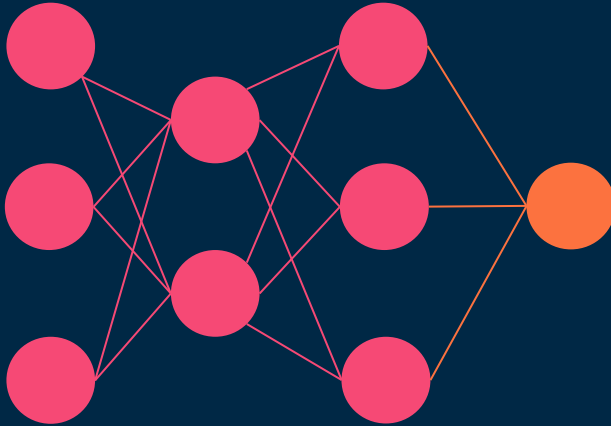


Regression an Fluxkompensation





Neuronale Netze



KI inspiriert vom menschlichen Gehirn

Nutzung von Neuronen und Connections in einem Netz, die Funktionen implementieren und Output aus Input generieren

Im Hidden Layer, zwischen Input und Output Neuronen, werden Inputs in Neuronen berechnet bis ein Output erfolgt

Die Neuronen im Hidden Layer sind Komponenten mit Einfluss, die dann das Neuronale Netz zu erkennen lernt

Vorteile

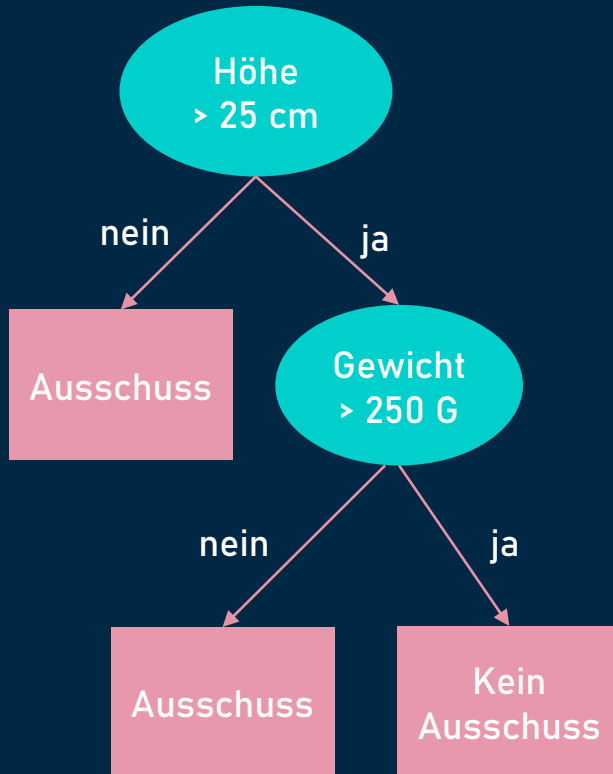
- Erkennt komplexe Muster
- Gut für viele Daten und mehrere Dimensionen
- Sehr gute Vorhersagen
- Multiple Outputs

Nachteile

- Benötigt viele qualitative Daten
- Performance
- Black Box



Decision Tree (Entscheidungsbaum)



Algorithmus zur Klassifikation von Objekten bzw. Datenpunkten mithilfe von bekannten Attributen und Werte

Ziel ist die Bildung einer hierarchischen Struktur mit möglichst wenigen Entscheidungswegen

Bildet mögliche Auswirkungen von aufeinanderfolgenden und zusammenhängenden Entscheidungen ab

Ermittlung erfolgt über direkte Klassifikation oder Wahrscheinlichkeiten

Vorteile

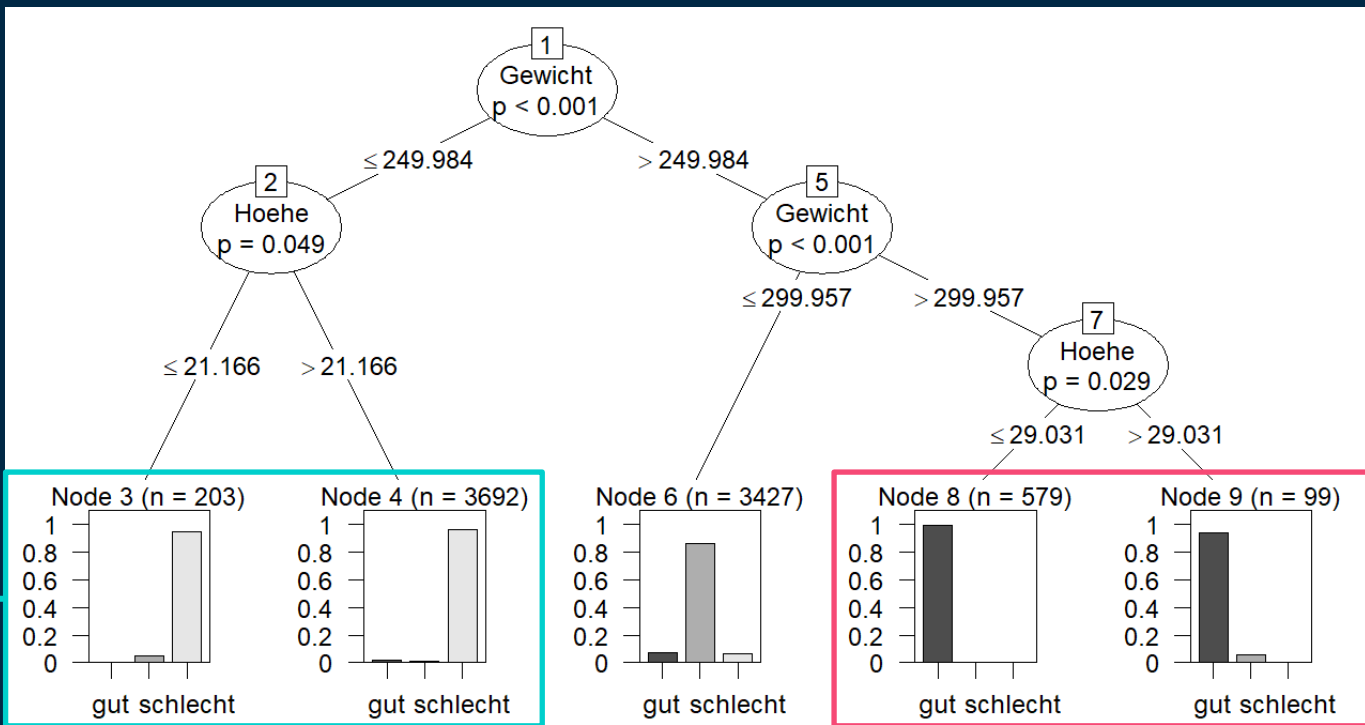
- Anschaulich, leicht verständlich
- Leicht implementierbar
- Identifizierung von Abhängigkeiten

Nachteile

- Viele Daten führen zu Überanpassung



Decision Tree und LScore

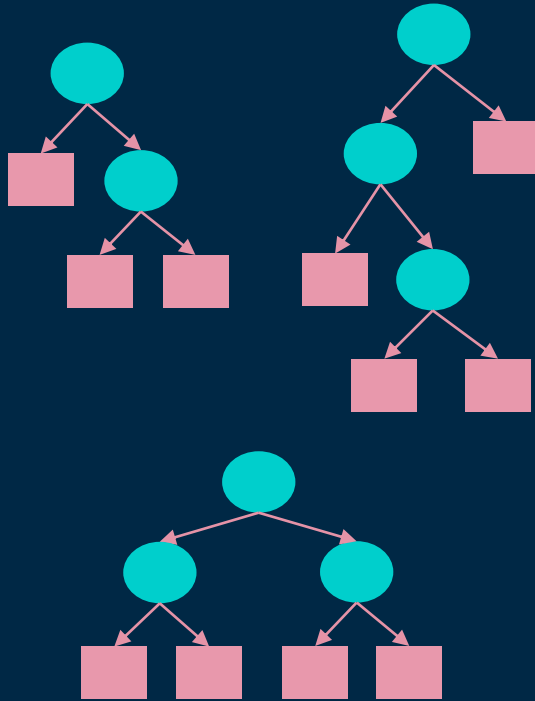


Einteilung
in schlecht

Einteilung
in gut



Random Forest



Daten werden in ein bootstrapped Subset aufgeteilt

Mit zufälligen Features wird ein Entscheidungsbaum gebaut

Dies wird oft parallel wiederholt mit zufälligen Subsets und Variablen → Vielfalt an Bäumen

Jeder Baum stimmt für die Klassifizierung ab. Mehrheit gewinnt

Vorteile

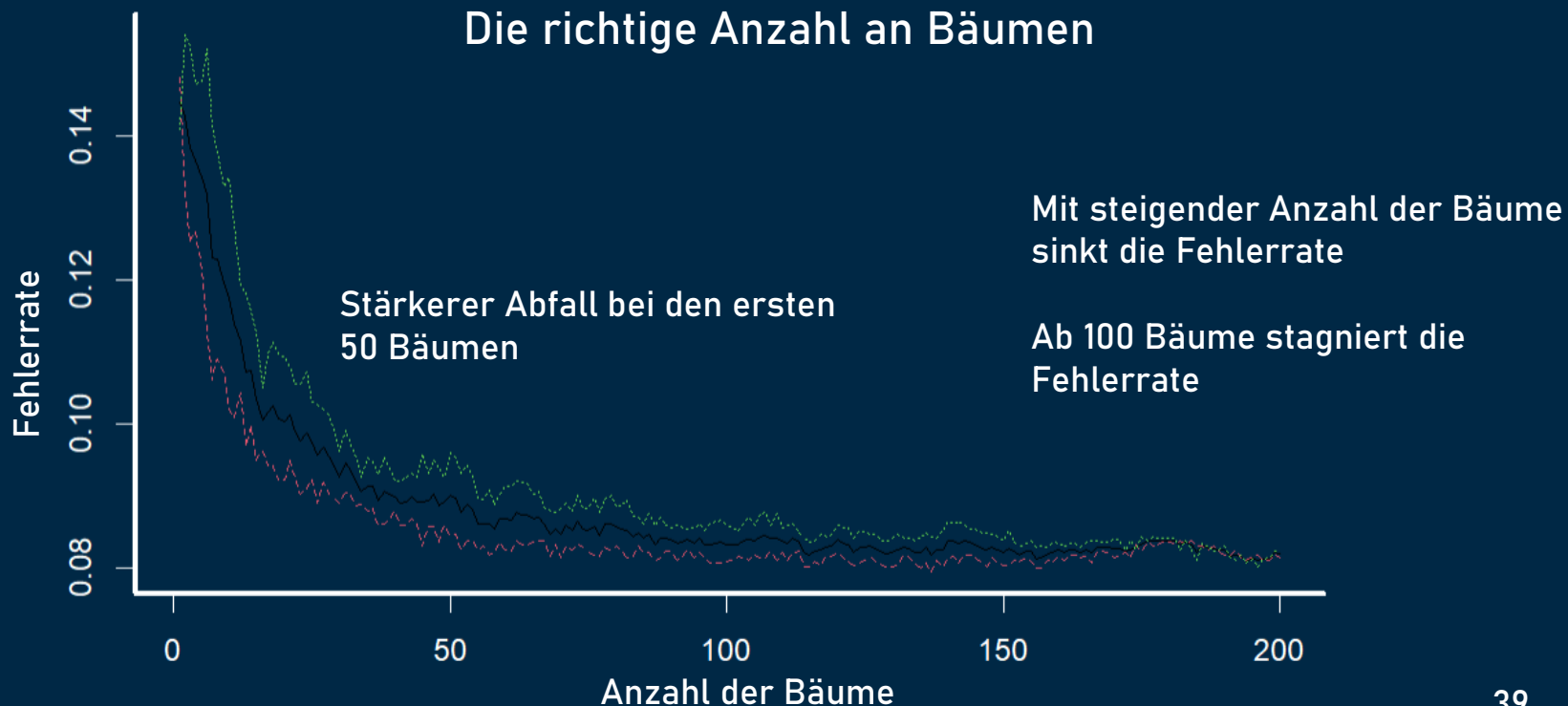
- Bessere Vorhersagen, da mehr Sichtweisen
- Unempfindlich ggü. Ausreißer
- Gut für viele Daten

Nachteile

- Hohe Varianz
- Langsam
- Komplex → Pruning nötig

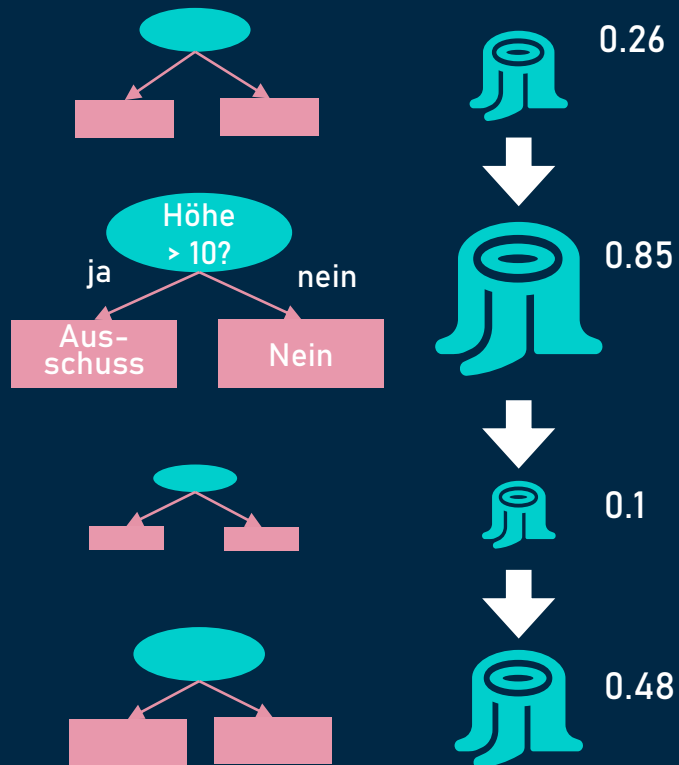


Random Forest und IstQualitativ





AdaBoost



Weak Learner

Stümpfe, die ein Feature und Subset nutzen, um Vorhersagen zu treffen, wodurch sie keine hohe Genauigkeit haben

Über Validierung und Testen wird die Gewichtung berechnet. Die Gewichtung sagt aus, wie viel Sagen ein Stumpf hat

Anhand der letzten Ergebnisse und Fehler werden die Stümpfe nacheinander beeinflusst und gebaut mit neuen Subsets

Jeder Stumpf stimmt ab mit seinen jeweiligen Gewichtungen. Die Klassifizierung mit der höchsten Gewichtung gewinnt

Vorteile

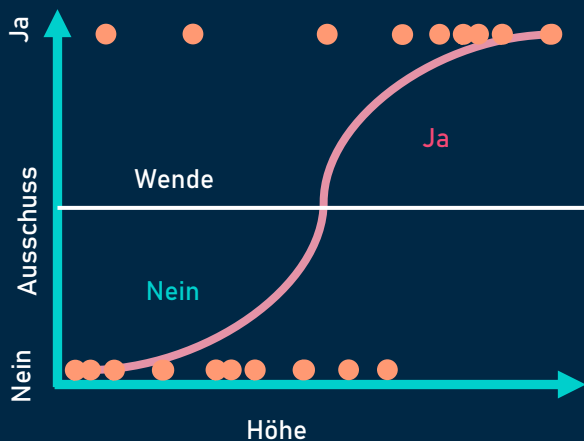
- Kombination von Weak Learner
- Bestrafung falscher Vorhersagen
- Weniger Varianz anfällig

Nachteile

- Langsam
- Benötigt qualitative Daten
- Reagiert auf Rauschen und Ausreißer



Logistische Regression



Binäre logistische Regression

Sagt vorher ob es zu einer Kategorie oder zur Anderen gehört

Mithilfe einer Sigmoid Kurve werden Wahrscheinlichkeiten bestimmt, ob es zu einer Kategorie gehört. Wende bei 50%

Anfang und Ende haben hohe Wahrscheinlichkeiten für die Klassifizierung, während in der Mitte es ungenau ist

Vorteile

- Effizient im Lernen
- Stark in simplen Zusammenhängen

Nachteile

- Benötigt viele Daten
- Pauschalisierung ab der Wende
- Nicht gut für komplexe Muster



Why should I trust you?

Erklärungs Modelle

... um das **wieso** Zu beantworten



LIME

Local Interpretable Model-agnostic Explanation

Für eine **lokale Interpretierbarkeit** der Features
und deren Auswirkungen auf die Vorhersage

SHAP

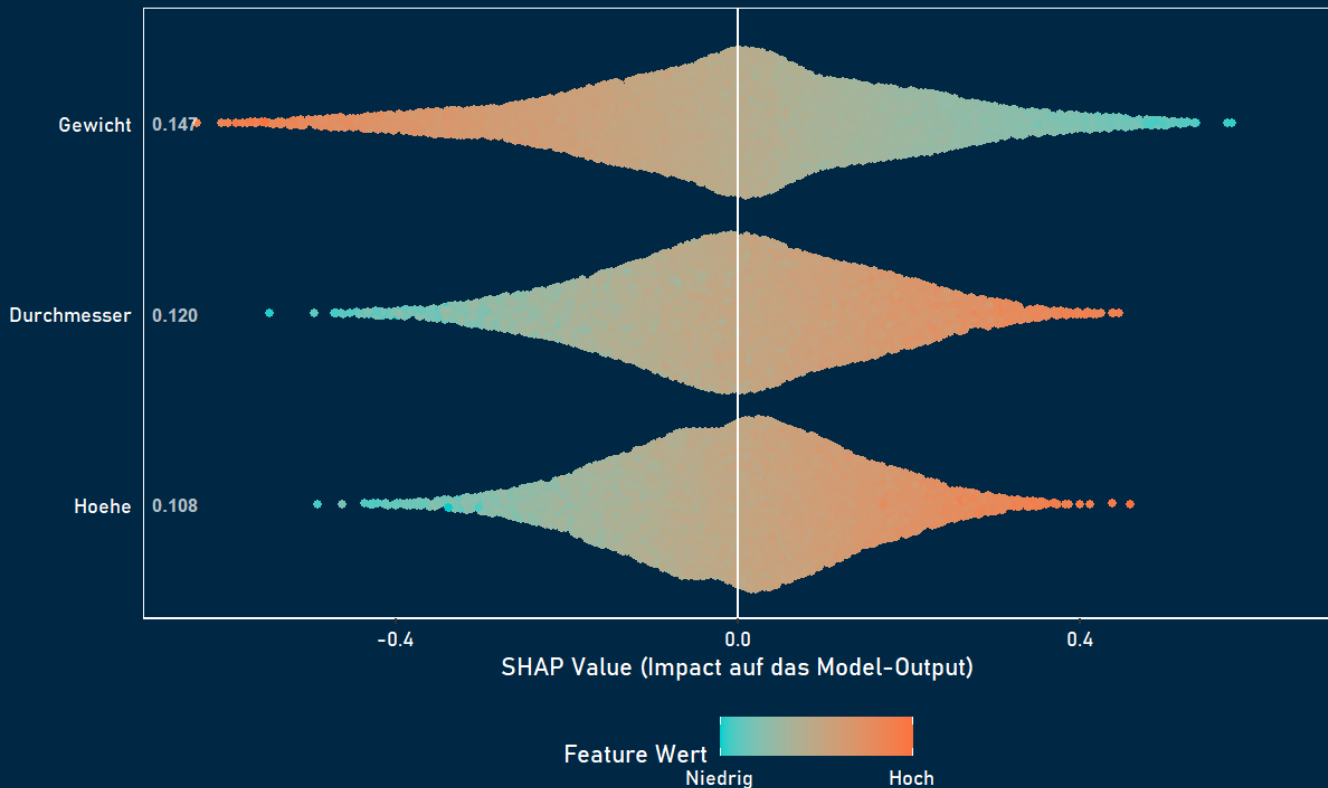
SHapley Additive exPlanation

Für die umfassende Interpretierbarkeit der
Auswirkungen **der Feature-Werte** auf die Vorhersagen





Regression erklärt an Qualität mit SHAP



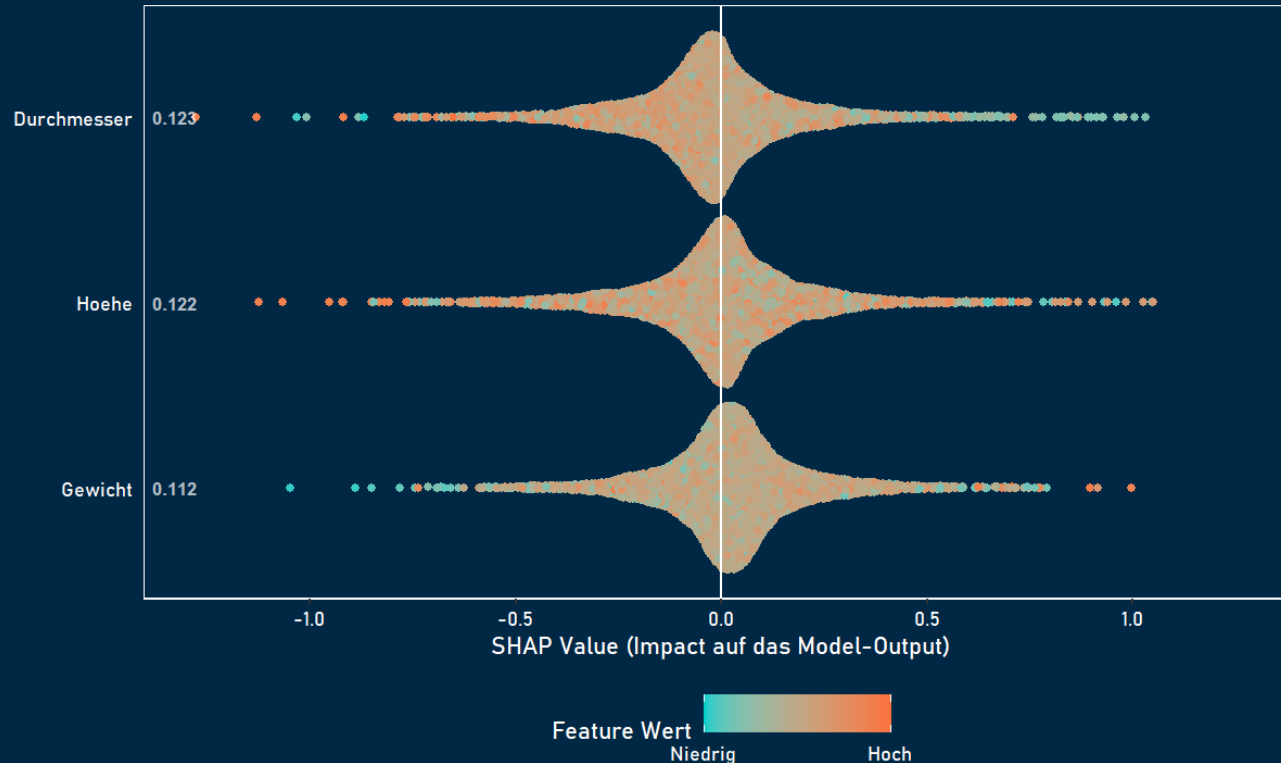
Ein hohes Gewicht
verringert die Qualität

Ein großer Durchmesser
und große Höhe wirken
sich positiv auf die
Qualität aus

Viele Datenpunkte
sammeln sich um 0



Regression erklärt an XKlasse mit SHAP



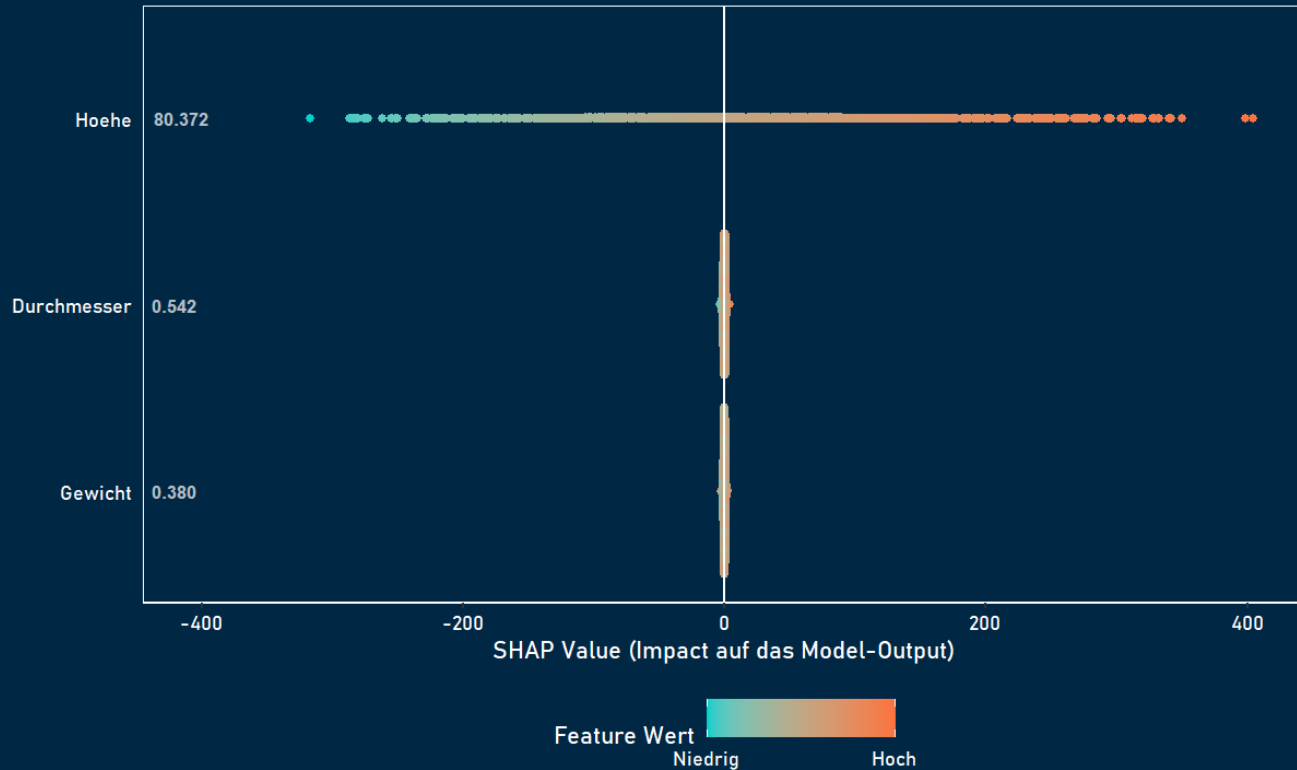
Viele Datenpunkte
sammeln sich um 0

Kein eindeutiges Muster
auffindbar

XKlasse nicht abhängig
von Attributen



Regression erklärt an Gammawert mit SHAP



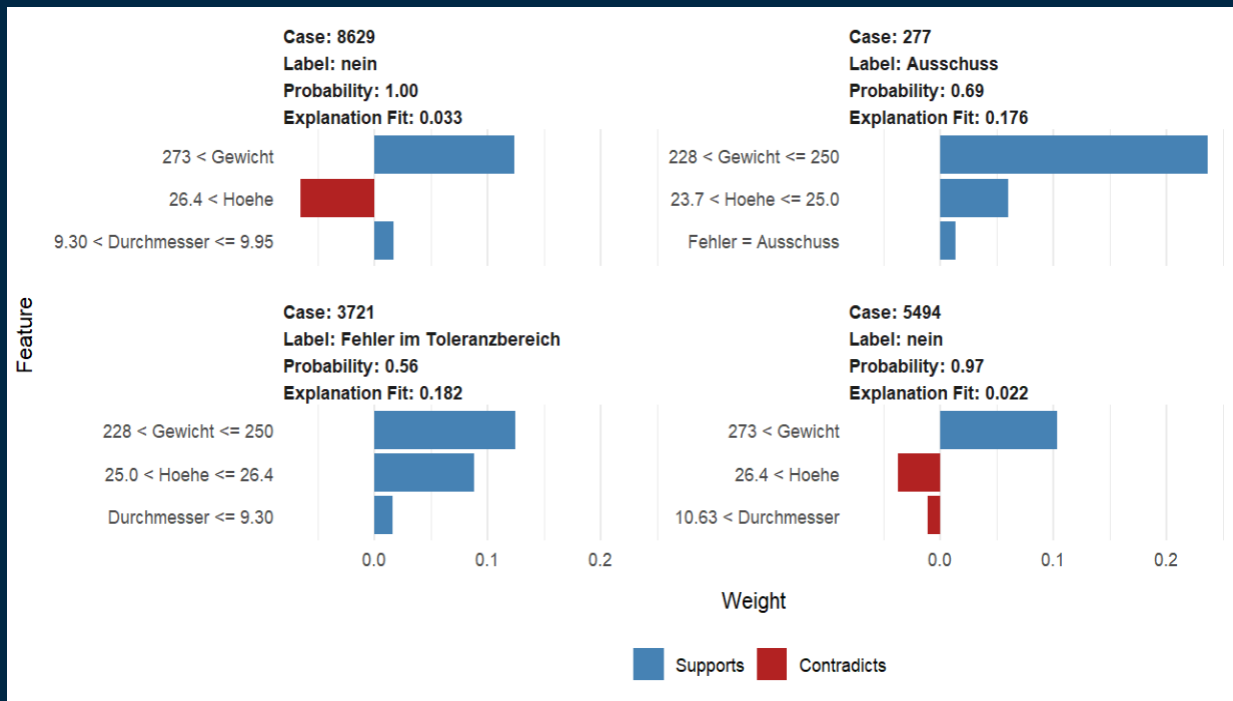
Gammawert nur
abhängig von Höhe
→ linearer
Zusammenhang

Große Höhe führt zu
hohem Gammawert

Nur minimales
Rauschen sichtbar bei
Durchmesser & Gewicht



Random Forest erklärt an Fehler mit LIME



Wahrscheinlichkeit der Vorhersage hier hoch

Blaue Balken unterstützen die Entscheidung

Rote sprechen gegen die Entscheidung

Jeder Einfluss hat eine Gewichtung



Evaluierung der Modelle

Metriken

... um das **wie** gut zu beantworten

93%

Accuracy

Anteil der richtig vorhergesagten
Klassifizierungen

Mean Absolute Error

Durchschnittliche Abweichung der Vorhersage
vom tatsächlichen Wert

0.02



Confusion Matrix

Gegenüberstellung der, auf Basis der Testdaten
vorhergesagten Werte mit den tatsächlichen Werten



Business



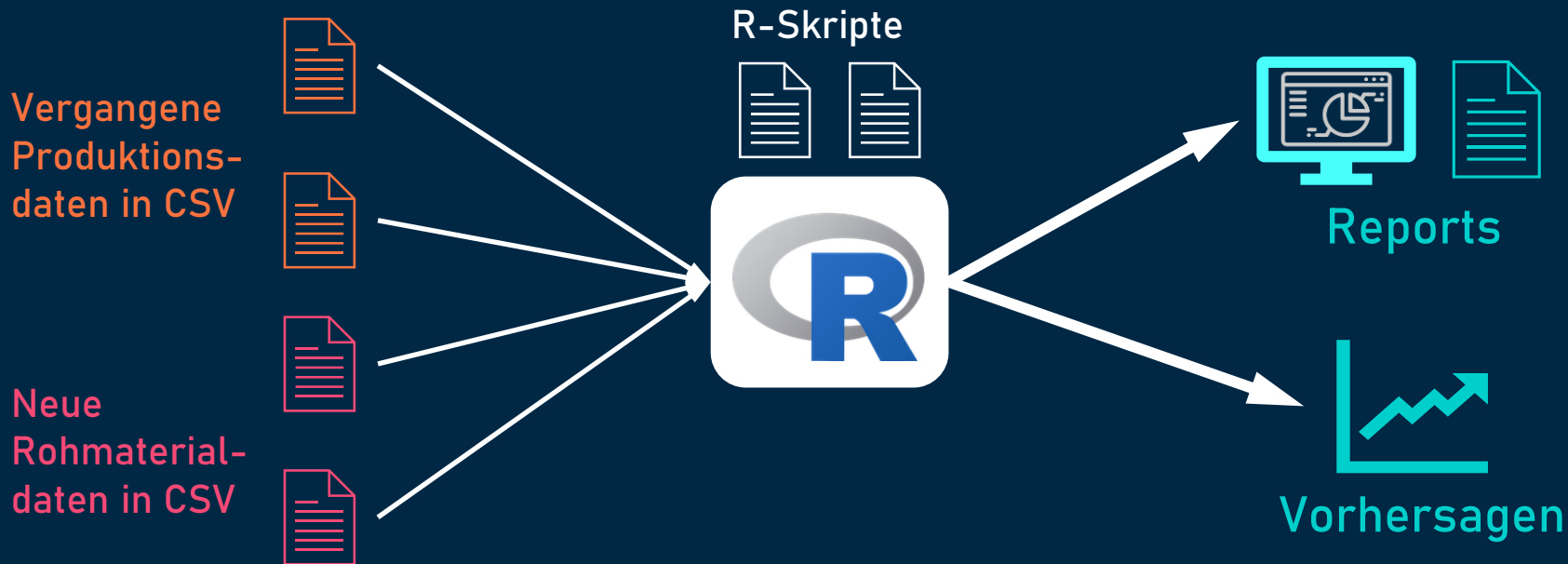
1010
1010

Technologie



Empfehlung für SmartBuild

Erstellen von Reports und Vorhersagen





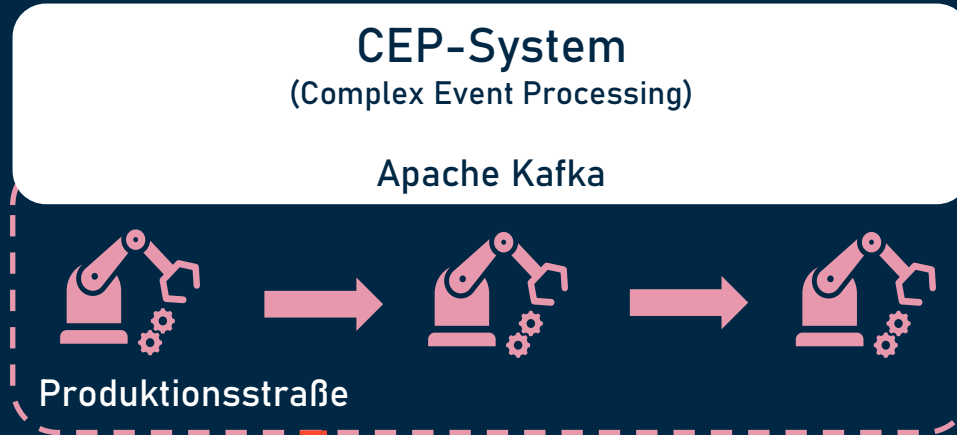
Empfehlung für SmartBuild

Daten in Echtzeit verarbeiten und Produkte steuern

Messung der Produkte
mit Echtzeitdatenstrom



Rohmaterial



Endprodukt



Aussortierung von fehlerhaften
Produkten, bevor es in die Produktion
kommt und währenddessen

Ihr Kontakt zum Erfolg



Niko Kauz
Senior Manager – Data Analytics

T +49 177 60358817

M +49 800 1111111

@ niko.kauz@epic-data.com

Zögern Sie nicht, uns zeitnah zu kontaktieren. Gerne besprechen wir in einem weiteren Termin die Details unserer Partnerschaft.

Unser Code zum Einsehen



<https://github.com/NicolasMahn/BusinessIntelligence2>



EPIC DATA CONSULTING

25.01.2022

Backup Slides



Clustering



Einteilung von Datenpunkten in homogene Gruppen (Cluster)

Durch wiederholende Berechnungen und Anpassungen von Centroiden werden Schwerpunkte für Gruppierungen gesetzt. Ändern sich die Centroiden nicht mehr im Cluster wurden die Schwerpunkte gefunden

Ein neuer Datenpunkt wird anhand seiner Attribute einem Cluster zugeordnet

Die gruppierten Datenpunkte sollten am Ende ähnlich sein

Vorteile

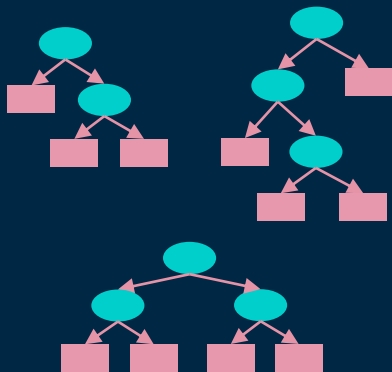
- Effizient für einfache Modelle
- Für verschiedene Szenarien und Datentypen nutzbar

Nachteile

- Reagiert empfindlich auf Ausreißer
- Clustern nicht immer möglich
- Kugel-\Kreisförmige Zusammenhänge nötig



Random Forest



Aufteilung der Daten in ein bootstrapped Subset. Der Out-Of-Bag Dataset, die nicht genutzten Daten, wird zur Evaluierung verwendet einmal für den Entscheidungsbaum, wo er nicht reinkam und ein Mal für die anderen Bäume zur Abstimmung. Bei solchen Evaluierungen kommt es zum Out-Of-Bag Error, dem Anteil der nicht korrekten Vorhersagen

Mit einigen zufällig gewählten Features wird ein Entscheidungsbaum gebaut. Je nach Einstellung werden X Features gewählt (Default ist Wurzel der Anzahl aller Features). Der Node mit dem größten Information Gain wird gewählt

Dies wird oft wiederholt mit zufälligen Subsets und Variablen, wodurch eine Vielfalt an Bäumen entsteht, die zusammen oft bessere Ergebnisse liefern als ein Baum, da diese alle abstimmen, welche Klassifizierung für einen neuen Datenpunkt gewählt wird und die Klassifizierung mit den meisten Stimmen gewählt wird

Bei der Optimierung des Random Forests kann die Anzahl der zu erstellenden Bäume und auch die Anzahl der gewählten Feature konfiguriert werden. Weiter ist Pruning und Boosting mit Weighting möglich



Benchmark

LScore Klassifizierung

Durchschnittliche Ausführungszeit bei 10 Versuchen und Generierung
von 50 Decision Trees / Decision Stumps

Random Forest



Ausführungs-
Zeit

0.4 Sekunden

Vorhersage
Genauigkeit

77.85%

AdaBoost



9.2 Sekunden

78.6%

FastAdaBoost

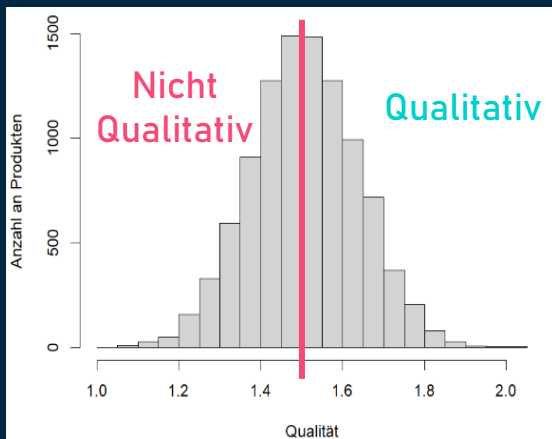


2.62 Sekunden

71.35%

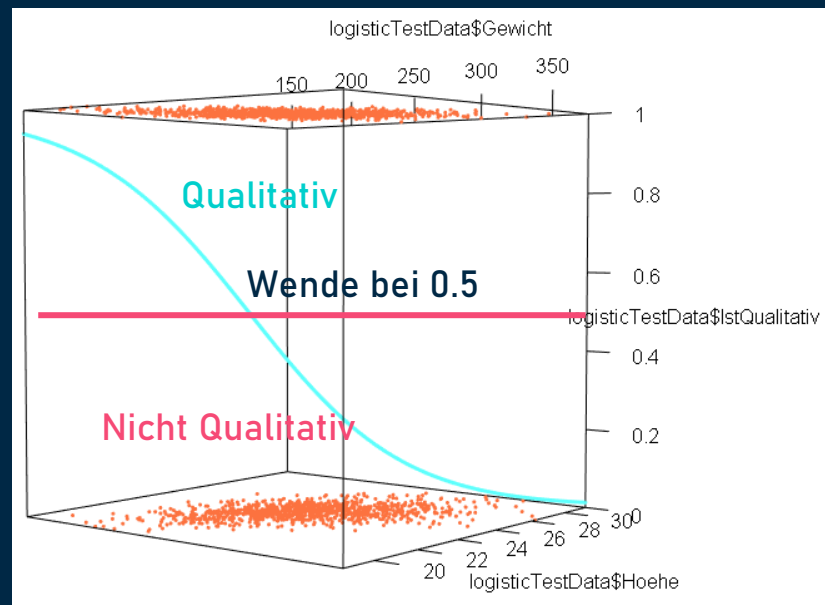
Qualität weiter im Visier

➔ 93.45% korrekte Vorhersage mit unserem Modell, wenn die Qualität eingestuft wird

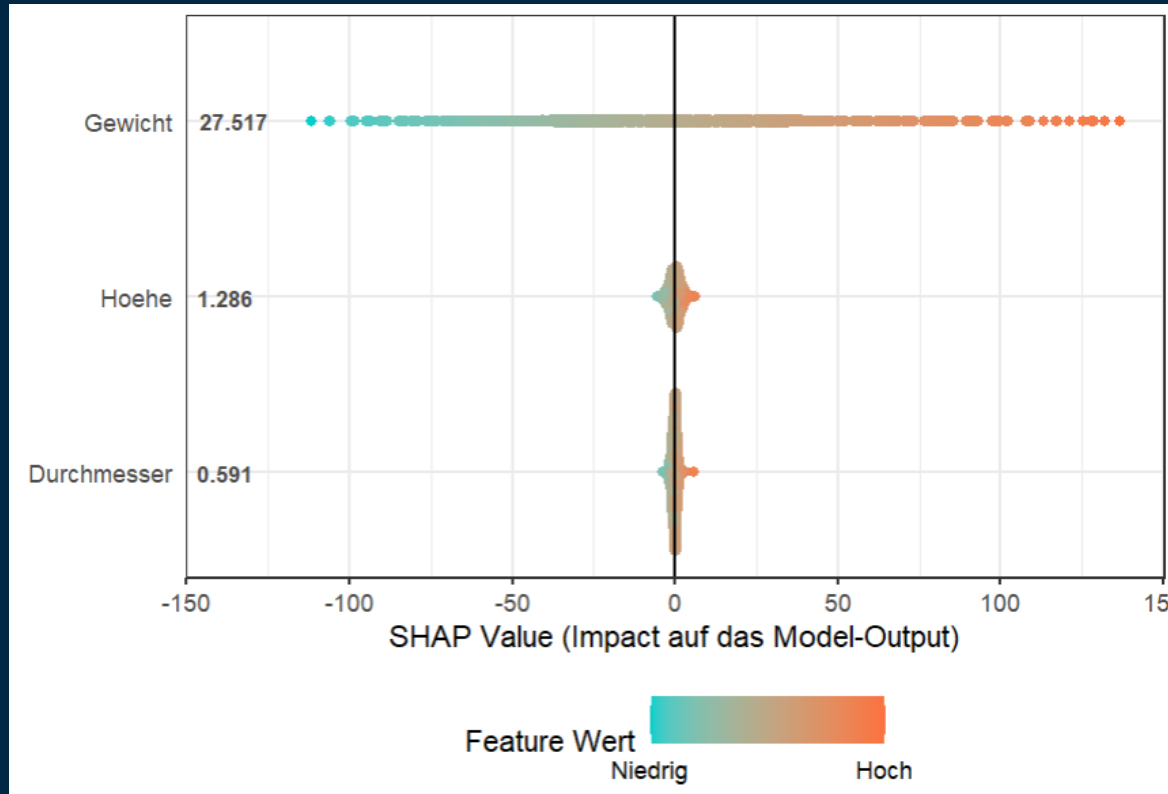


Vorhersage
ob Produkt
Qualität von
mind. 1.5 hat

Umsetzung mit weiterer
Kategorisierung IstQualitativ:
→ Qualität > 1.5 = Ja, sonst Nein



Regression erklärt an Warpfaktor



Nach 100 Durchläufen
train-rmse: 1.570859

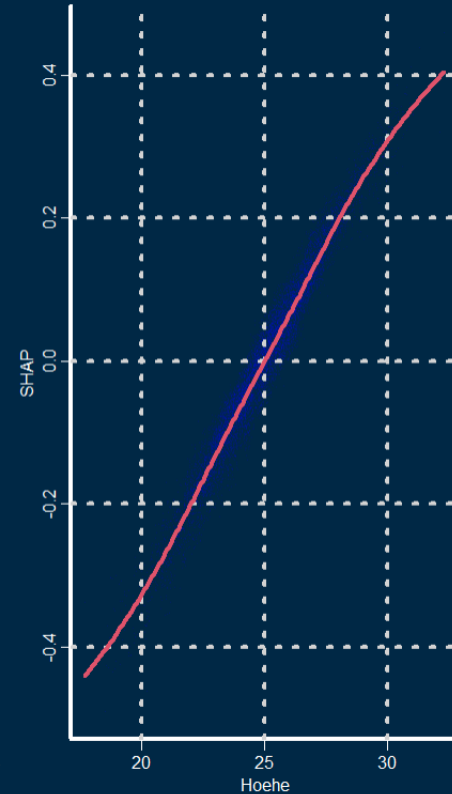
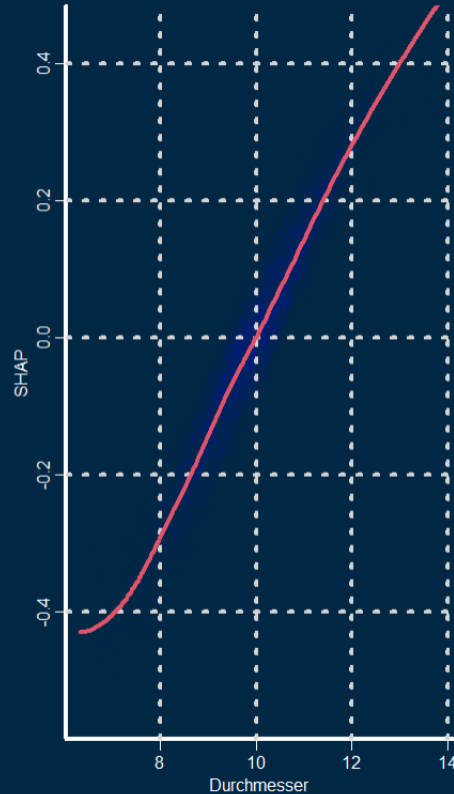
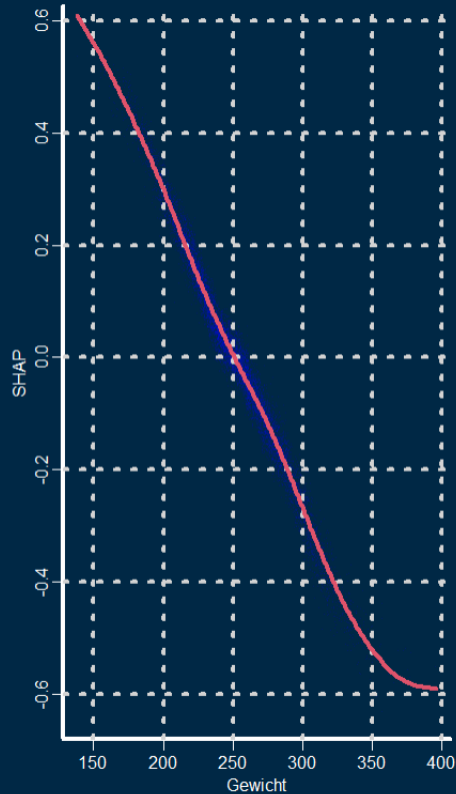
Gammawert nur abhängig
von Gewicht
→ linearer Zusammenhang

Große Höhe führt zu hohem
Gammawert

Leichtes Rauschen sichtbar
bei Durchmesser und
Gewicht



Regression erklärt an Qualität mit SHAP

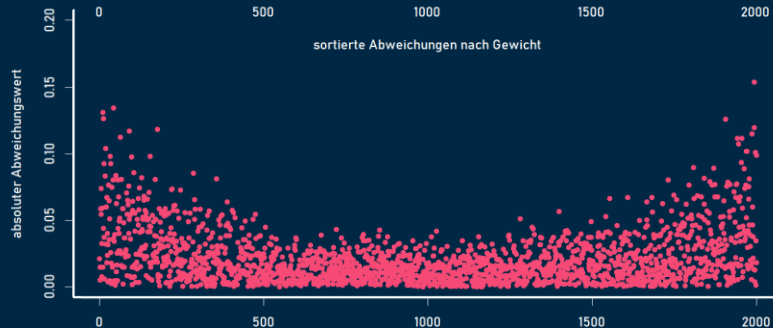
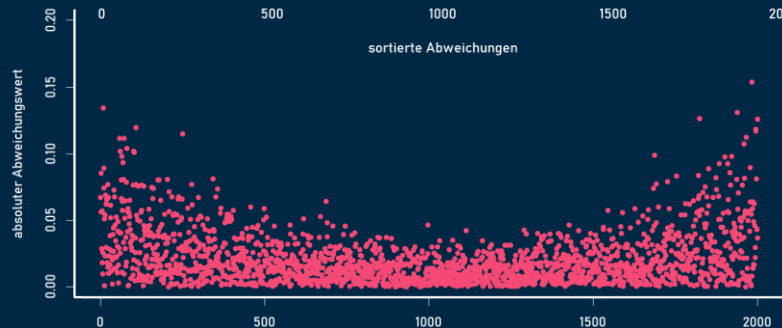
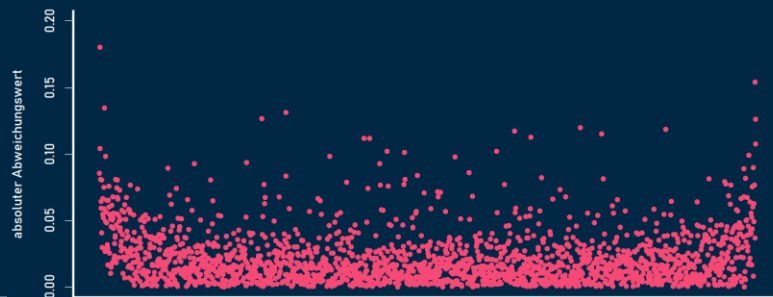
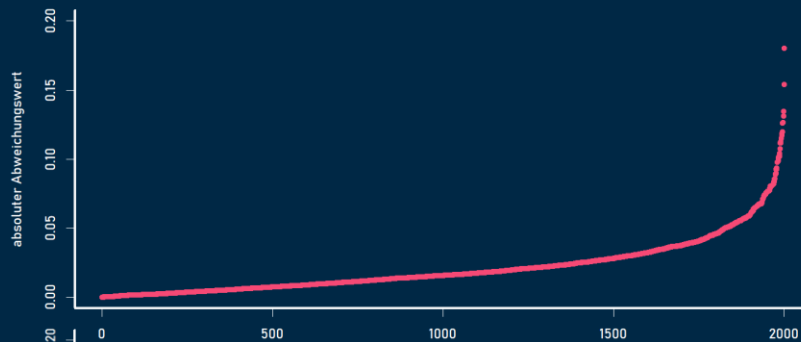




Qualität im Visier



Durchschnittliche Abweichung: 0.02





XKlasse im Visier – Decision Tree



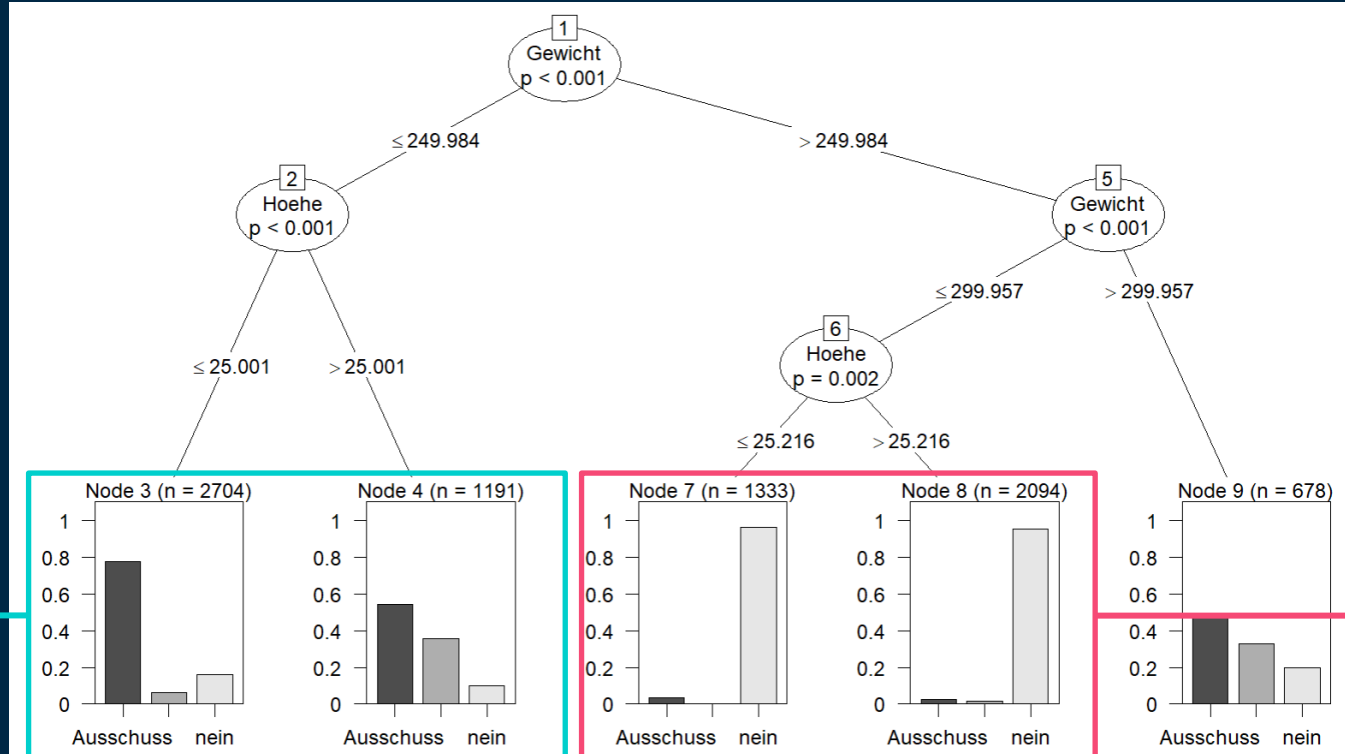
24,25% korrekte Vorhersage mit unserem Modell
Die XKlasse ist weitestgehend unabhängig von den Bemaßungen der Teile

| | | Tatsächliche Zuordnung | | | |
|--------------|-----|------------------------|-----|-----|-----|
| | | I | II | III | IV |
| Vorhergesagt | I | 273 | 295 | 290 | 301 |
| | II | 0 | 0 | 0 | 0 |
| | III | 0 | 0 | 0 | 0 |
| | IV | 192 | 227 | 210 | 212 |

Klasse II und III
konnten nicht
vorhergesagt
werden



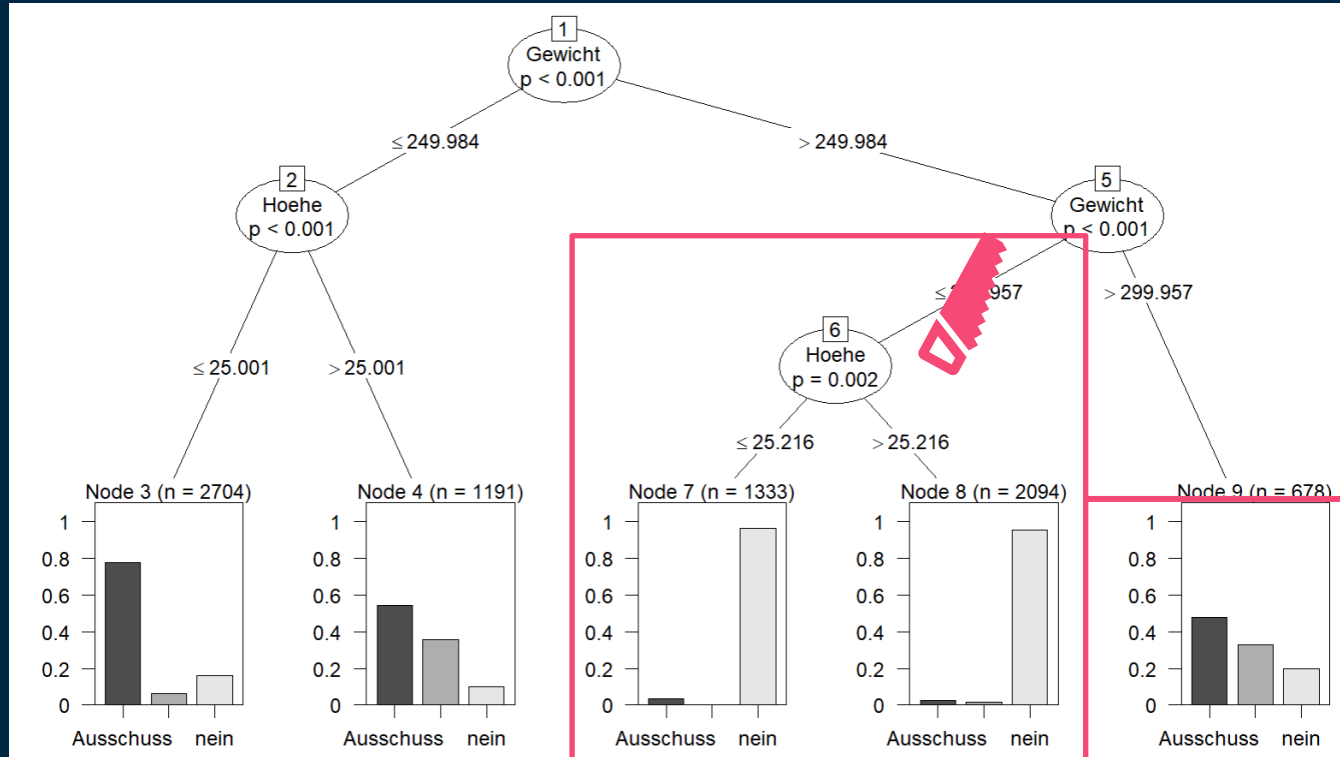
Decision Tree und Fehler



Einteilung
Ausschuss

Einteilung
in nein

Optimierung durch Pruning



Nicht-kritischen und redundante Zweige absägen, die eventuell auf Ausreißer reagierten

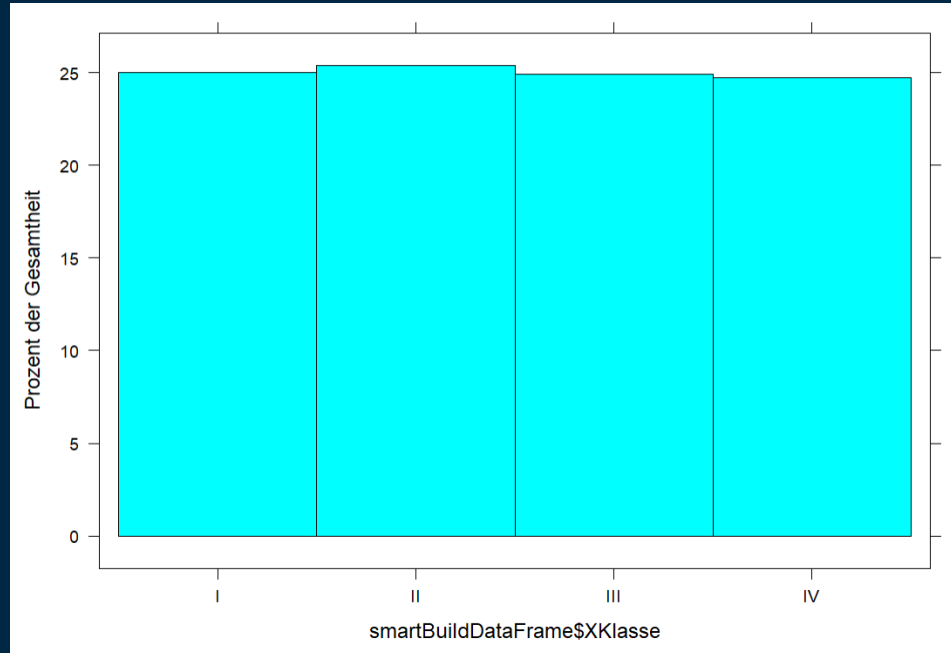
Damit wird das Risiko von Overfitting gesenkt und die Vorhersagen werden genauer



XKlasse exploriert

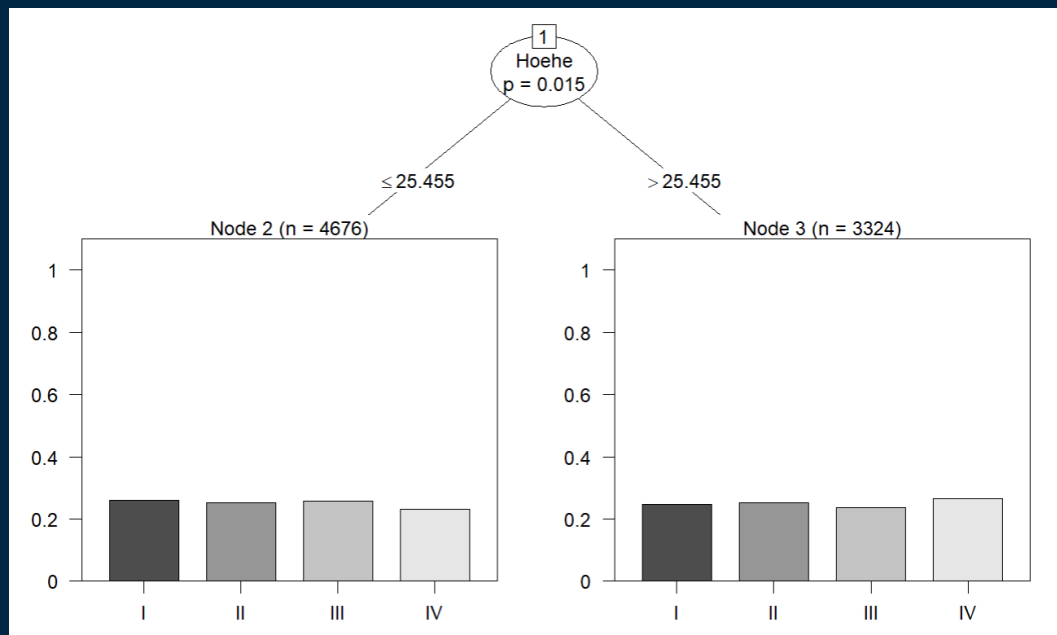


Gleichverteilung der XKlassen über die Daten



▶ XKlasse – Decision Tree

➡ Kein Zusammenhang mit Inputdaten gefunden



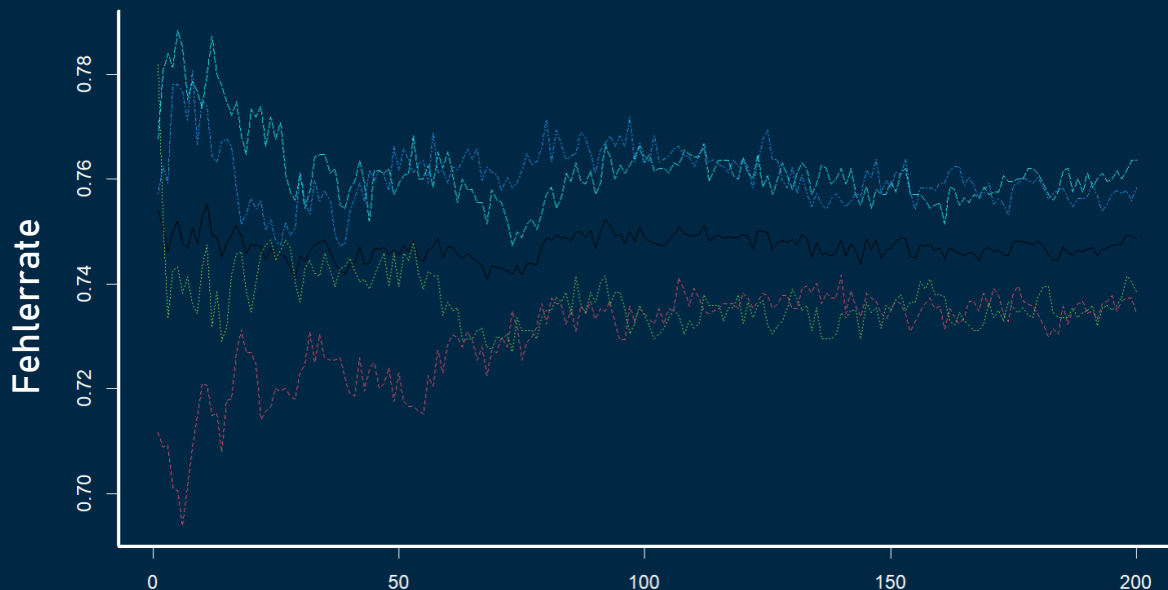
| | real | | | |
|--------------|------|-----|-----|-----|
| vorhergesagt | I | II | III | IV |
| I | 273 | 295 | 290 | 301 |
| II | 0 | 0 | 0 | 0 |
| III | 0 | 0 | 0 | 0 |
| IV | 192 | 227 | 210 | 212 |

24,25% der Vorhersagen waren Korrekt

Klasse II und III konnten nicht vorhergesagt werden



XKlasse im Visier



Anzahl der Bäume

| | real | | | | |
|------|------|-----|-----|-----|------------|
| pred | I | II | III | IV | Fehlerrate |
| I | 119 | 123 | 122 | 130 | 72,4% |
| II | 116 | 156 | 140 | 122 | 74,4% |
| III | 104 | 123 | 114 | 141 | 74,3% |
| IV | 126 | 120 | 124 | 120 | 76,9% |

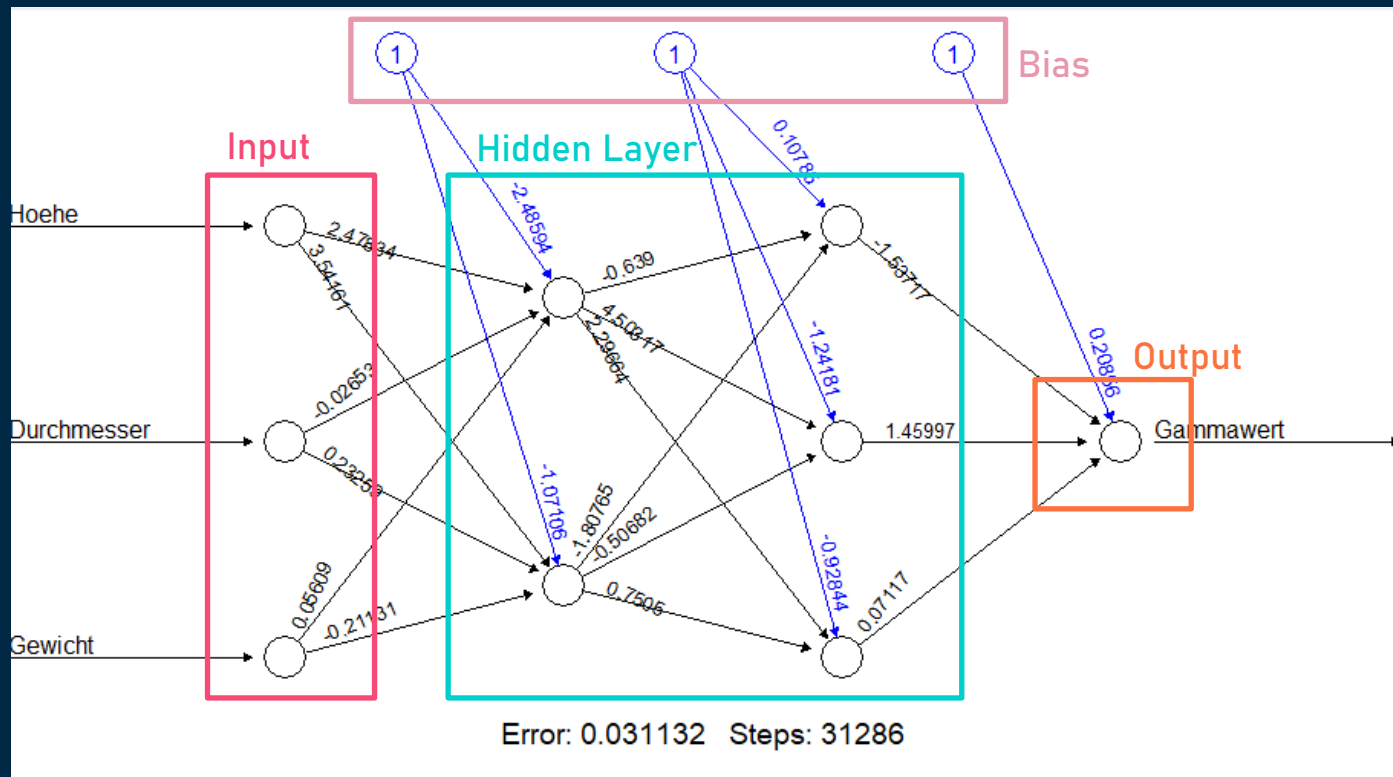
25,5% der Vorhersagen waren
Korrekt – etwas besser

Klasse II und III werden erkannt

Mit steigender Anzahl der Bäume
sinkt die Fehlerrate hier nur gering

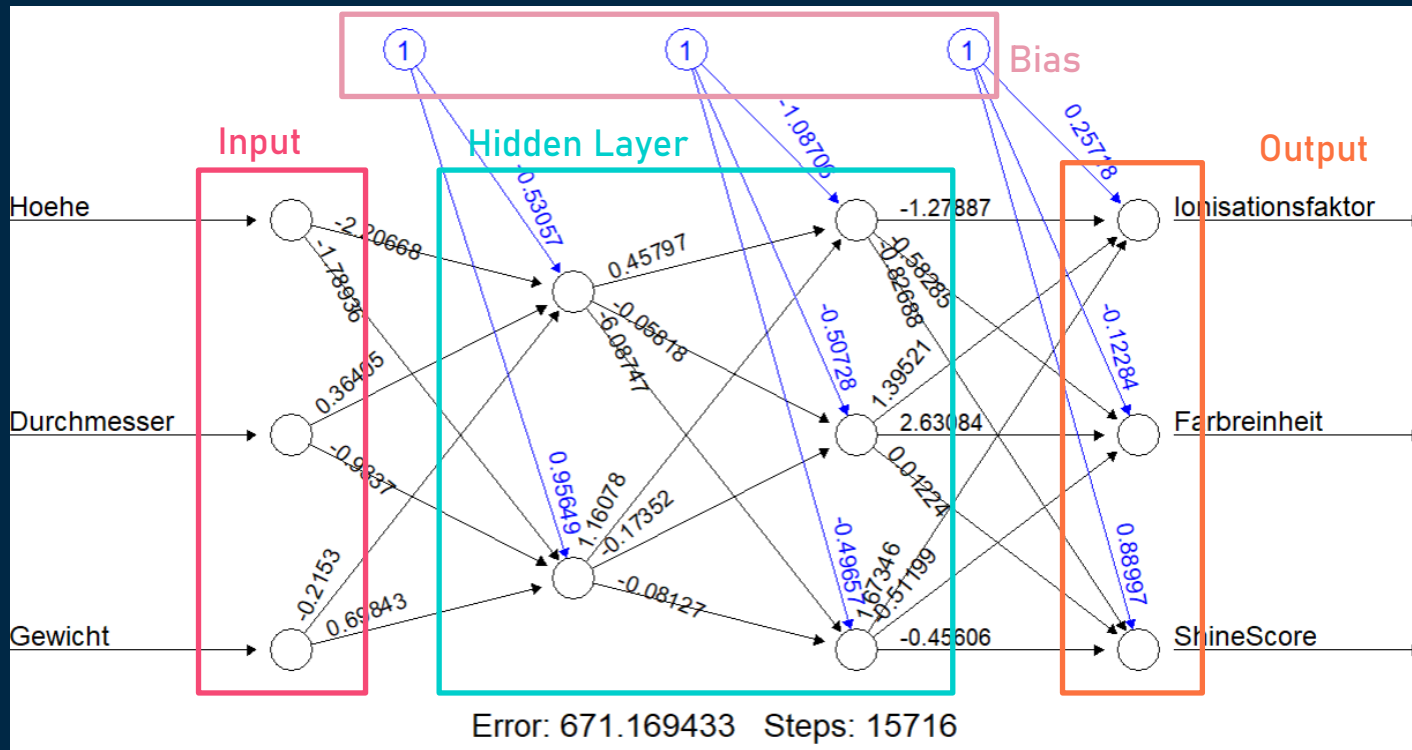


Neuronale Netze an Gammawert



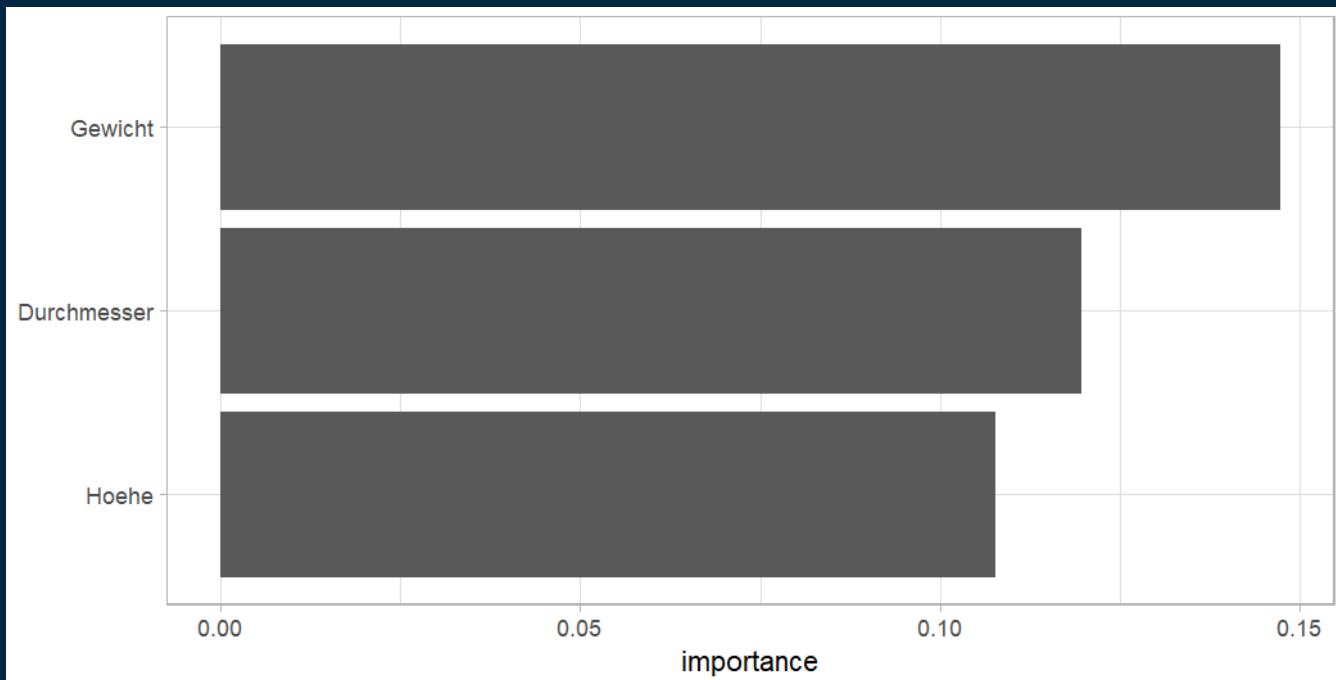


Multiple Outputs mit Neuronalen Netzen





Wichtigkeit der Attribute auf Qualität

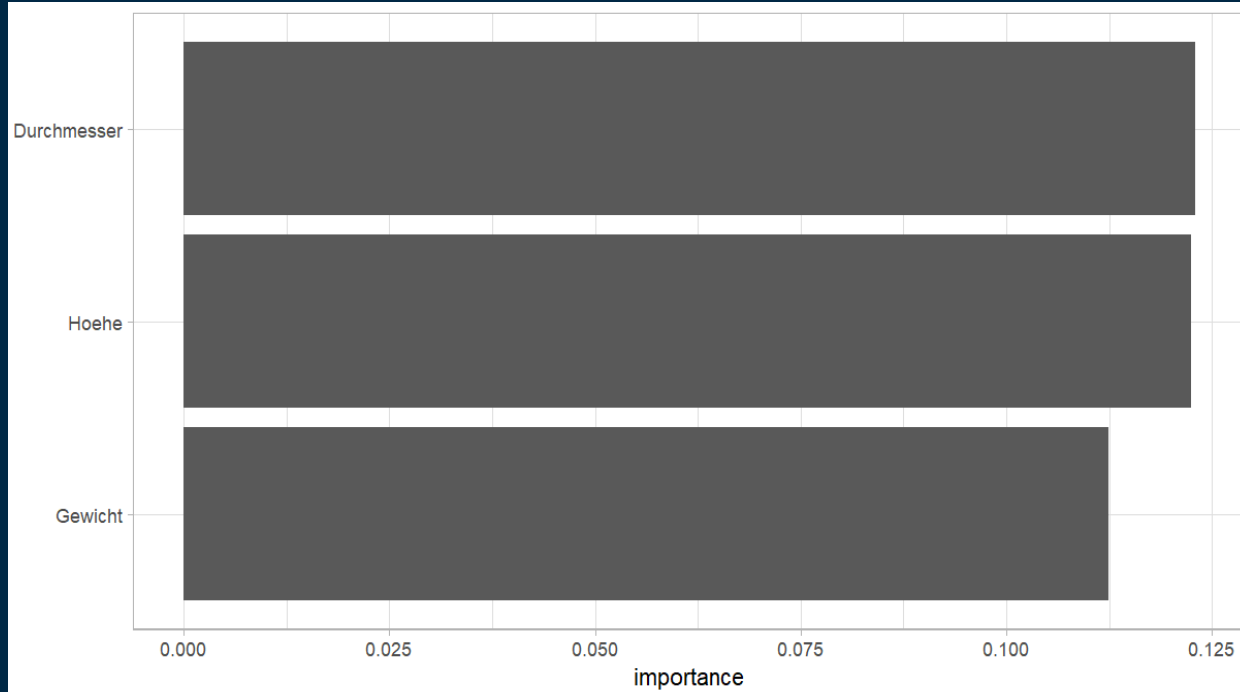


Gewicht am wichtigsten für Qualität

Durchmesser und Höhe ungefähr gleich



Wichtigkeit der Attribute auf XKlasse

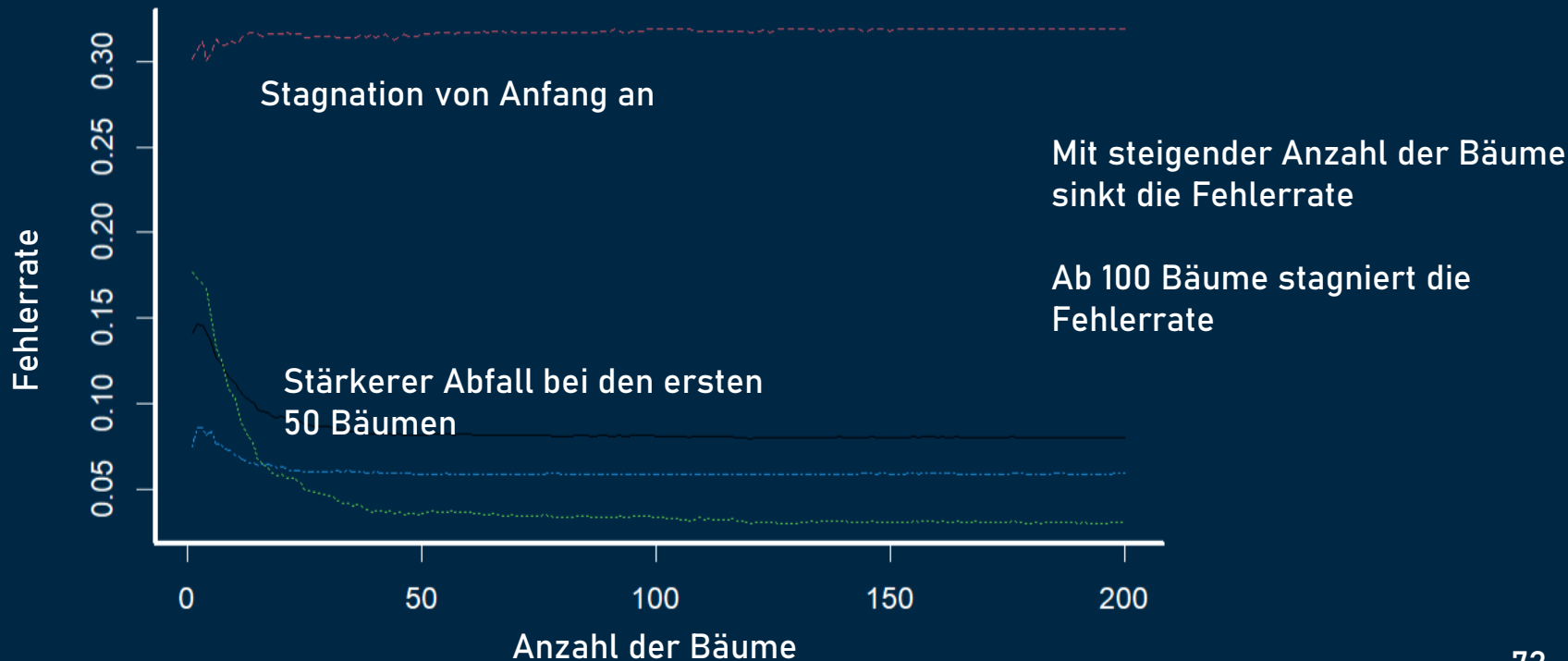


Alle Attribute sind fast
gleich wichtig für das
Merkmal XKlasse



LScore im Visier – RF

93% korrekte Vorhersage



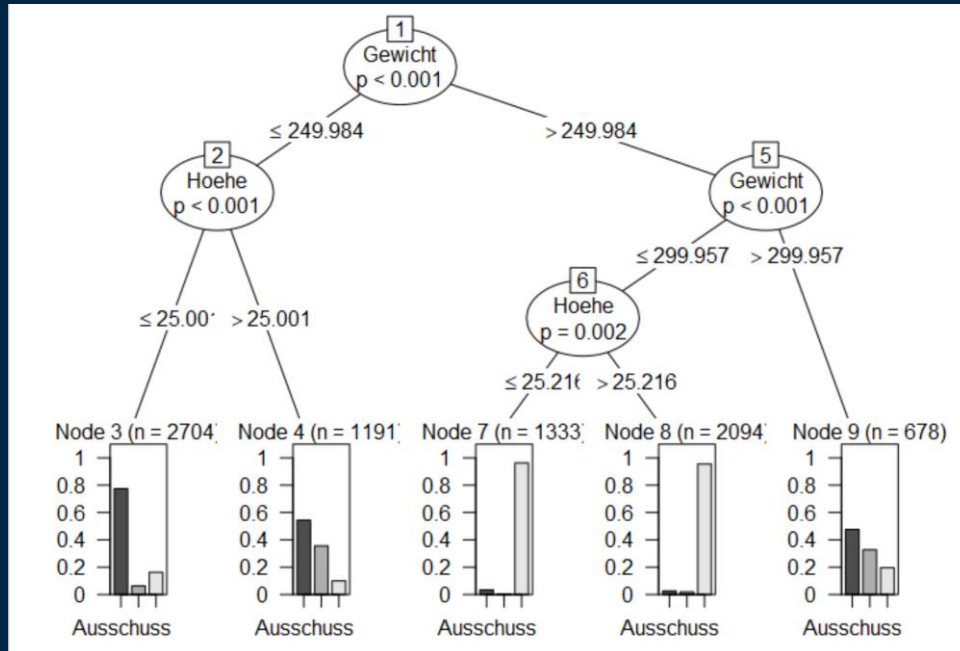


Fehler im Visier – Decision Tree

78.6% korrekte Vorhersage

| | real | | |
|--------------|-----------|-----|------|
| vorhergesagt | Ausschuss | FiT | nein |
| Ausschuss | 775 | 204 | 188 |
| FiT | 0 | 0 | 0 |
| nein | 27 | 9 | 797 |

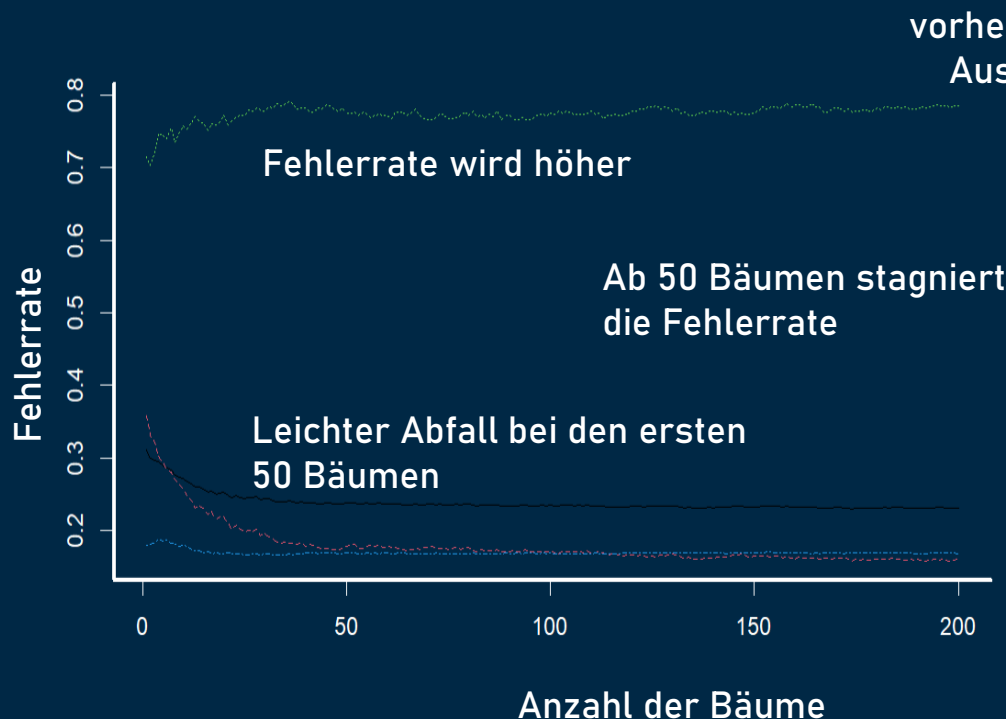
Fehler im Toleranzbereich
konnte nicht vorhergesagt
werden





Fehler im Visier – Random Forest

77% korrekte Vorhersage



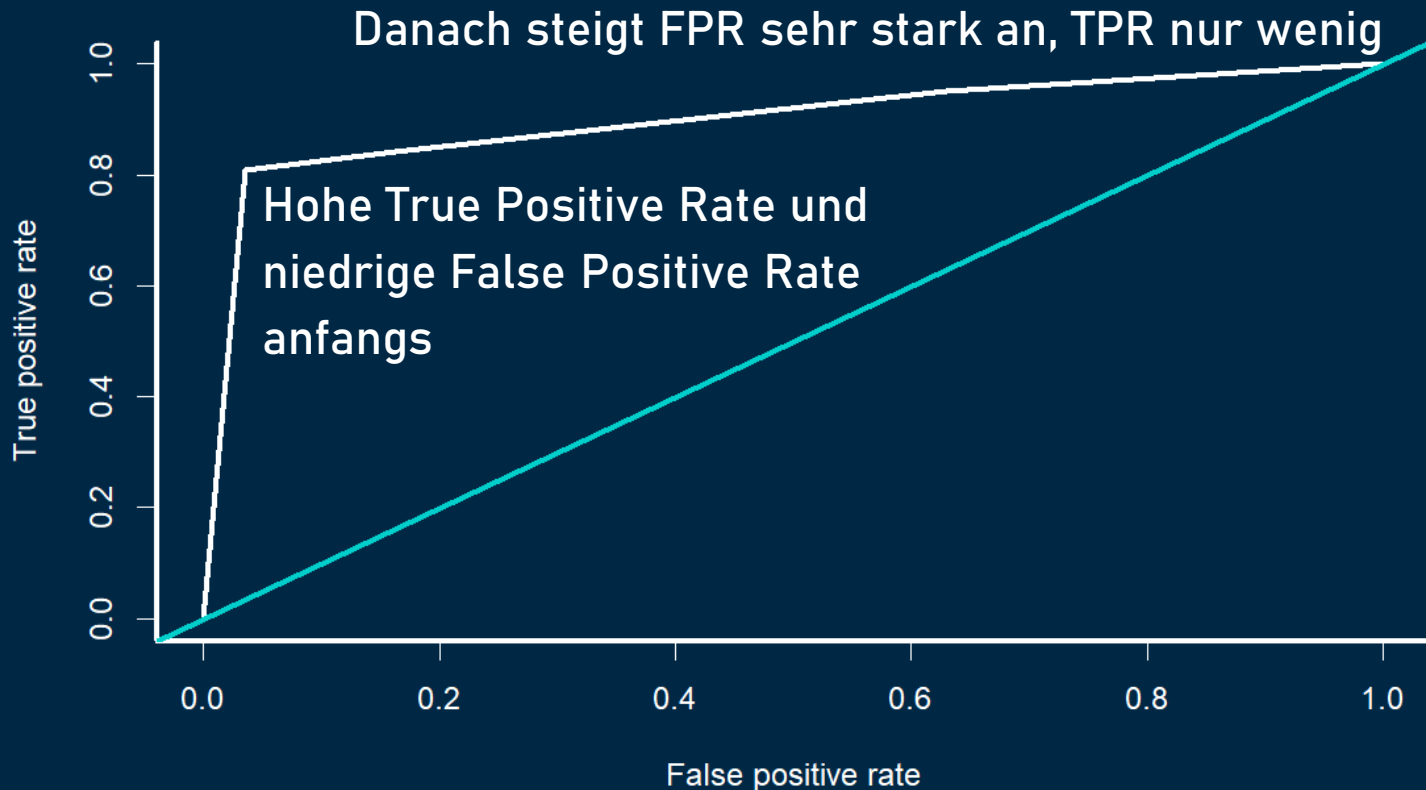
| | real | | | |
|--------------|-----------|-----|------|------------|
| vorhergesagt | Ausschuss | FiT | nein | Fehlerrate |
| Ausschuss | 686 | 149 | 170 | 16% |
| FiT | 74 | 52 | 13 | 78% |
| nein | 42 | 12 | 802 | 17% |

Fehler im Toleranzbereich konnten, wenn auch schlecht, vorhergesagt werden

Probleme bei der Erkennung von FiT bei Decision Tree, AdaBoost und Random Forest

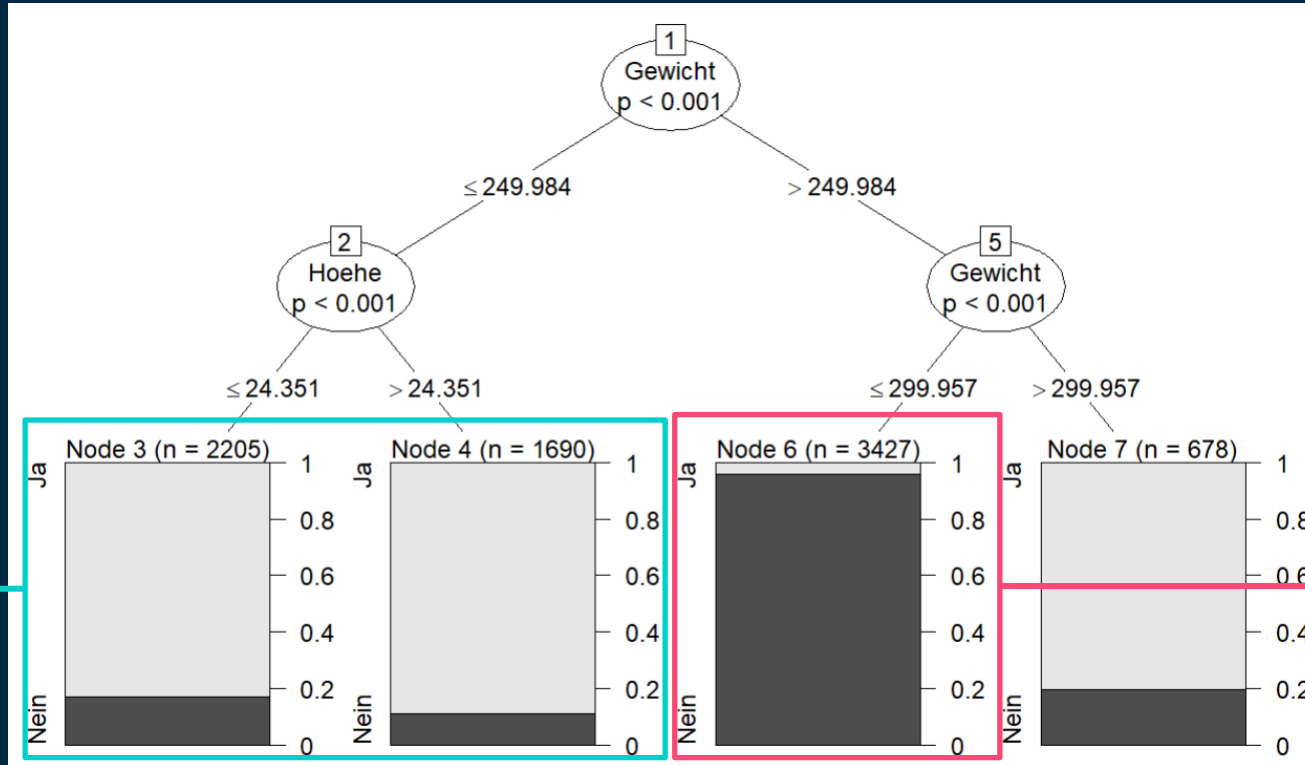


HatFehler – ROC Kurve





HatFehler – Decision Tree



Sehr viele
werden in Ja
zugeteilt

Fast alle
werden in
Nein zugeteilt