Data Science 1

Vorlesung 5 - Intro

Prof. Dr. Christian Bockermann

Hochschule Bochum

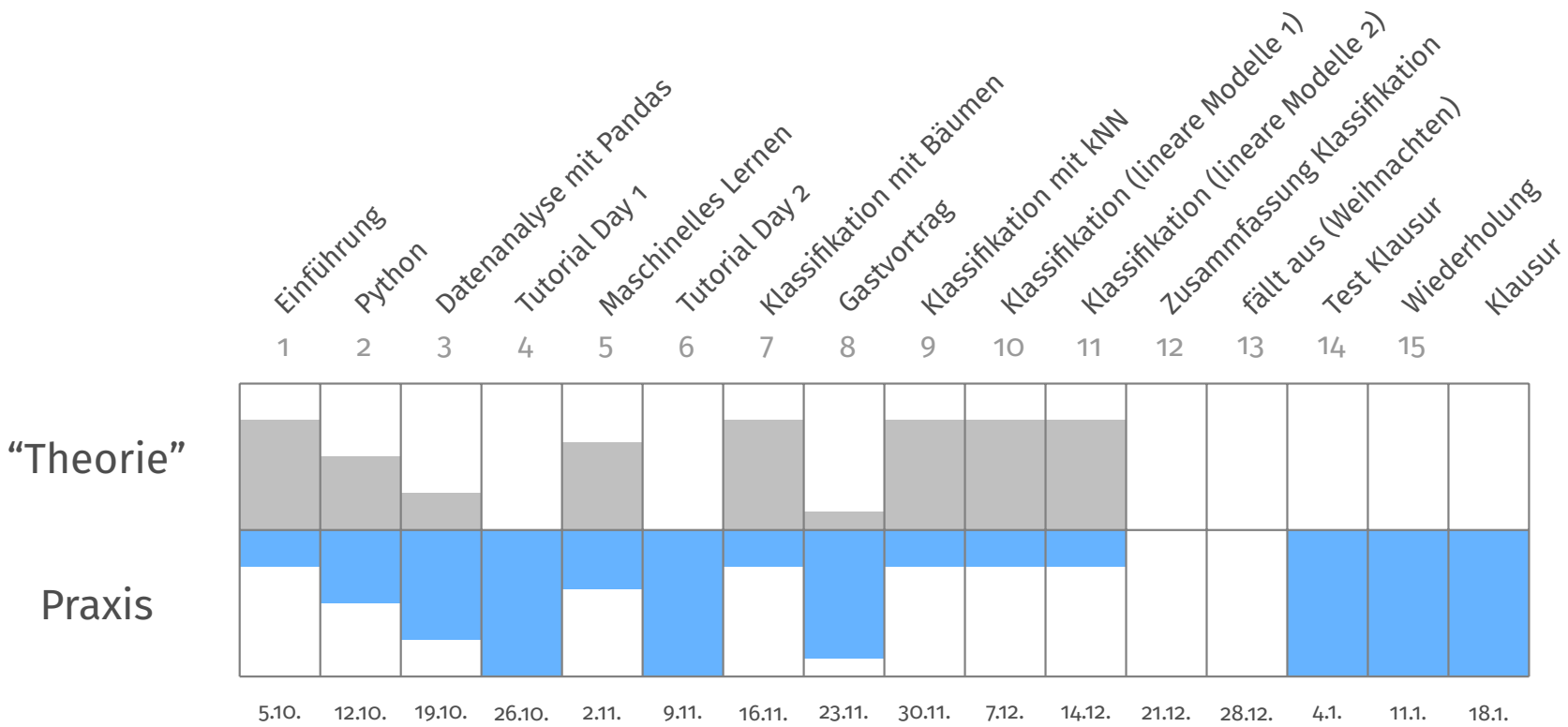
Wintersemester 2021/2022

Was geschah zuletzt?

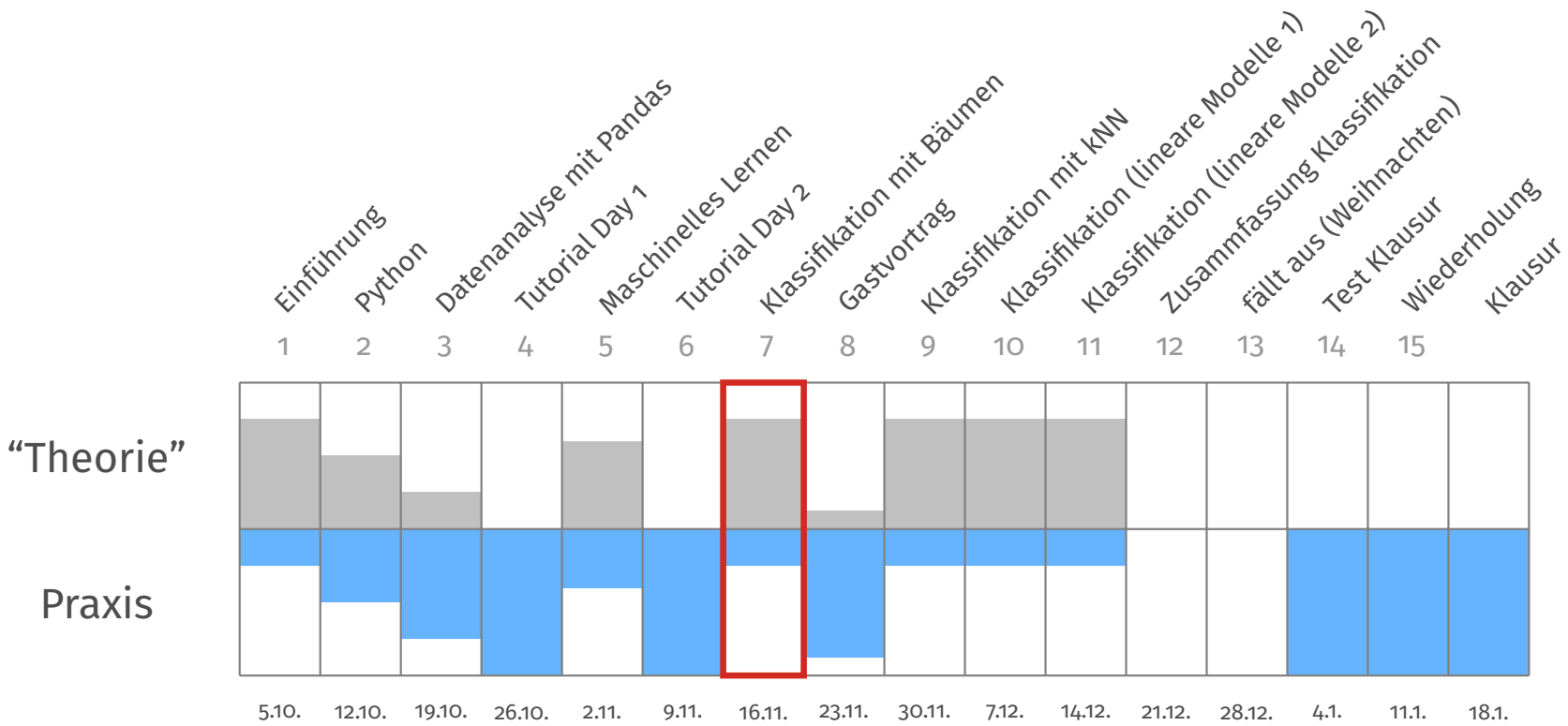
Wir sprachen über das Maschinelle Lernen!

- Grundlagen des Maschinellen Lernens
- Lernaufgaben als Fokussierung auf spezialisierte Tasks
- Formulierung von Modell-Training als Optimierungsaufgabe

Wo sind wir heute (Vorlesung 5) ?



Wo sind wir heute (Vorlesung 5) ?



Inhalt Vorlesung 5 - Worum geht's?

- Entscheidungsbäume als einfaches Lernverfahren
- Training/Erstellen von Entscheidungsbäumen
- Klassifikationsfehler und confusion matrix
- Modellierung/Training mit SciKit Learn

Bisheriger Classifier: Zufall

- Zufällig Klasse aus Trainingsdaten $\mathbf{X}_{train} \times \mathbf{y}_{train}$ wählen
- Wahrscheinlichkeit für Vorhersage der Klasse C

$$P(\hat{y} = C) = \frac{\text{Häufigkeit von } C \text{ in } \mathbf{y}_{train}}{|\mathbf{y}_{train}|}$$

Bisheriger Classifier: Zufall

- Zufällig Klasse aus Trainingsdaten $\mathbf{X}_{train} \times \mathbf{y}_{train}$ wählen
- Wahrscheinlichkeit für Vorhersage der Klasse C

$$P(\hat{y} = C) = \frac{\text{Häufigkeit von } C \text{ in } \mathbf{y}_{train}}{|\mathbf{y}_{train}|}$$

- Binäre Klassifikation, bei Gleichverteilung der Klassen führt zu durchschnittlichem Fehler von ~ 0.5

Bisheriger Classifier: Zufall

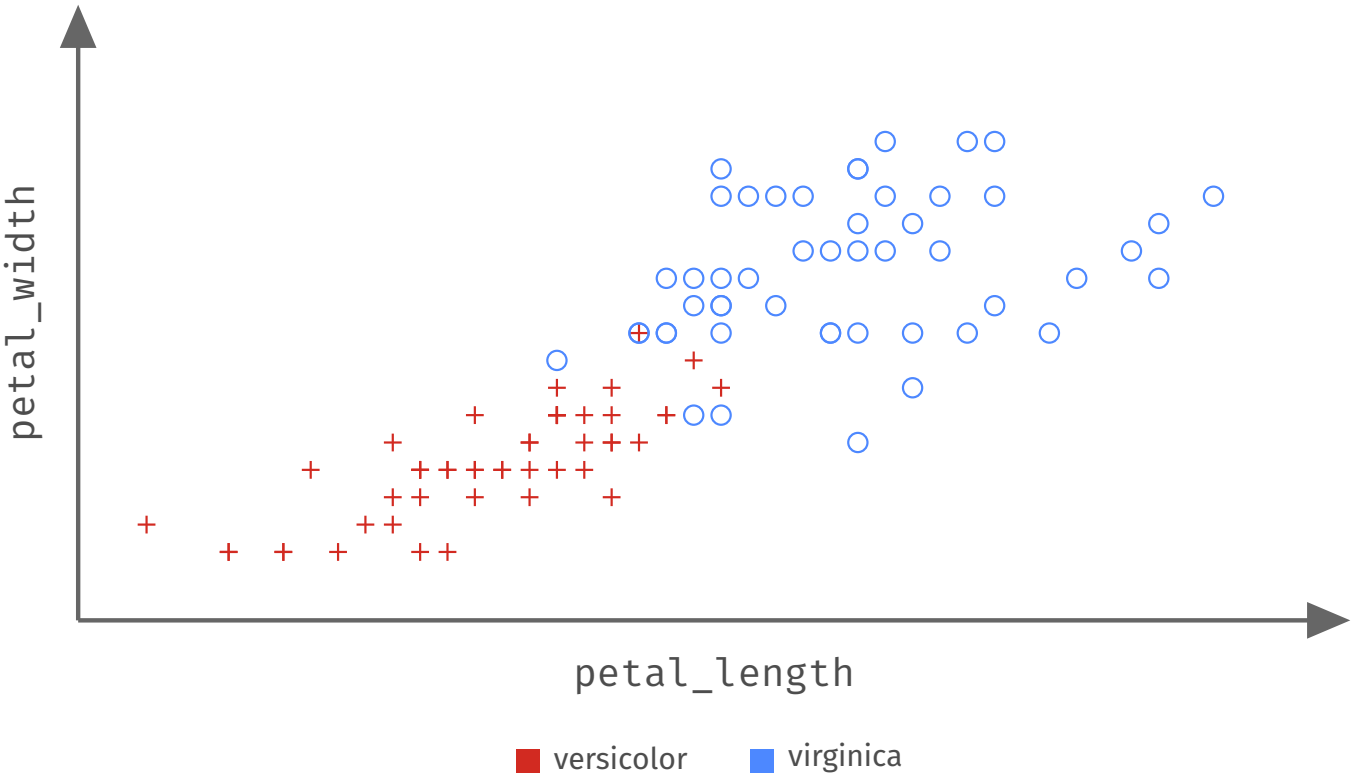
- Zufällig Klasse aus Trainingsdaten $\mathbf{X}_{train} \times \mathbf{y}_{train}$ wählen
- Wahrscheinlichkeit für Vorhersage der Klasse C

$$P(\hat{y} = C) = \frac{\text{Häufigkeit von } C \text{ in } \mathbf{y}_{train}}{|\mathbf{y}_{train}|}$$

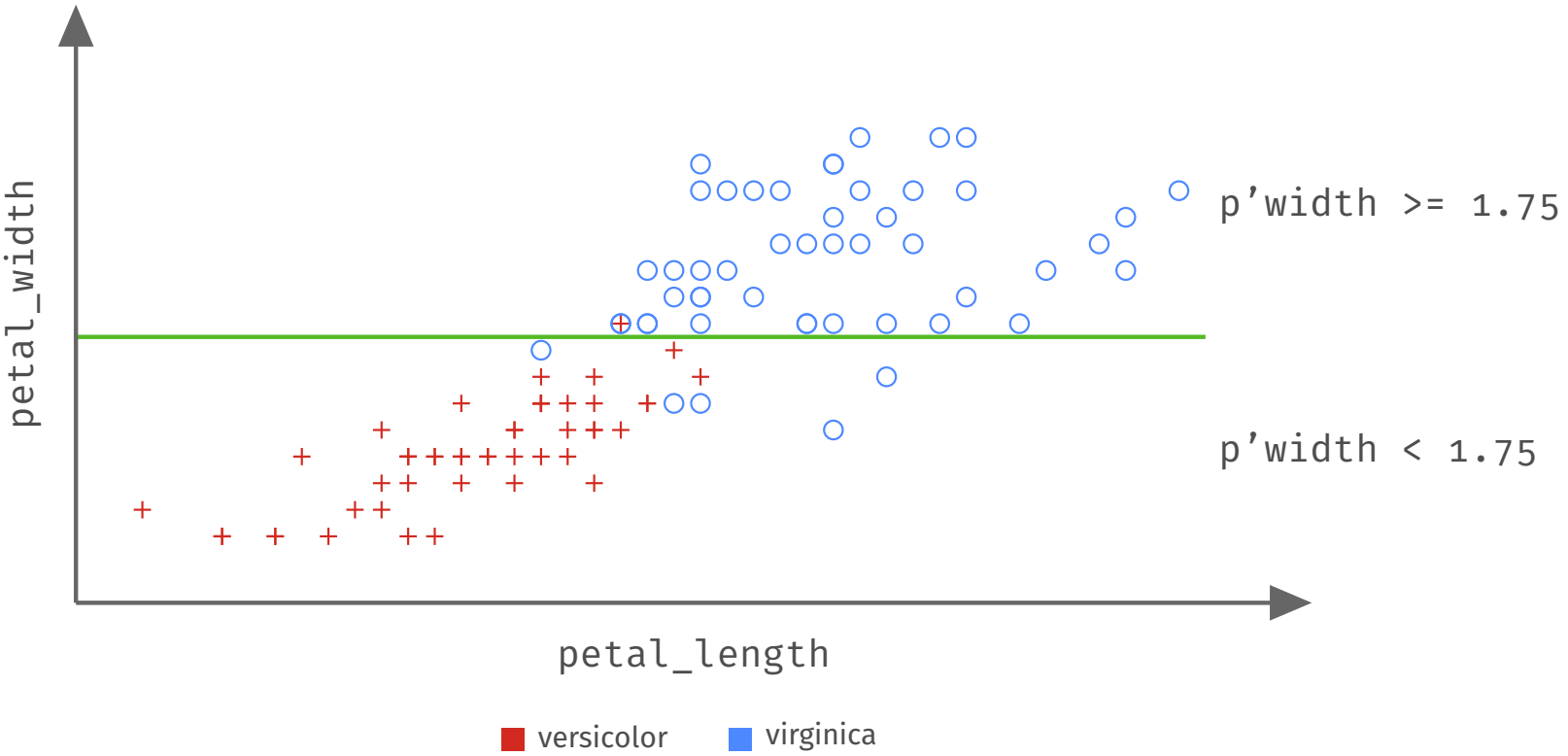
- Binäre Klassifikation, bei Gleichverteilung der Klassen führt zu durchschnittlichem Fehler von ~ 0.5

Wie können wir die Vorhersage verbessern?

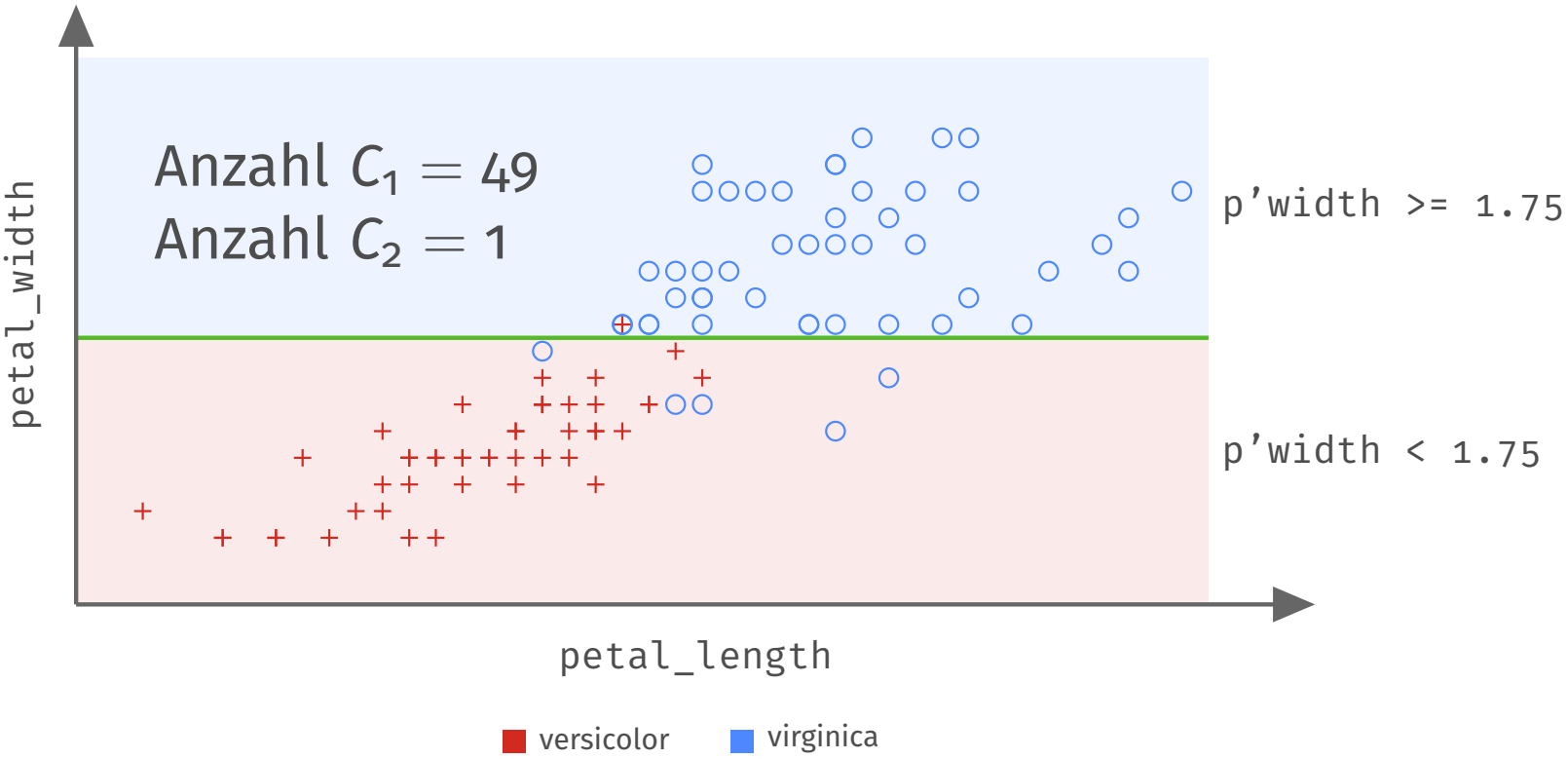
Idee: Daten teilen und W'keit für korrekte Vorhersage erhöhen



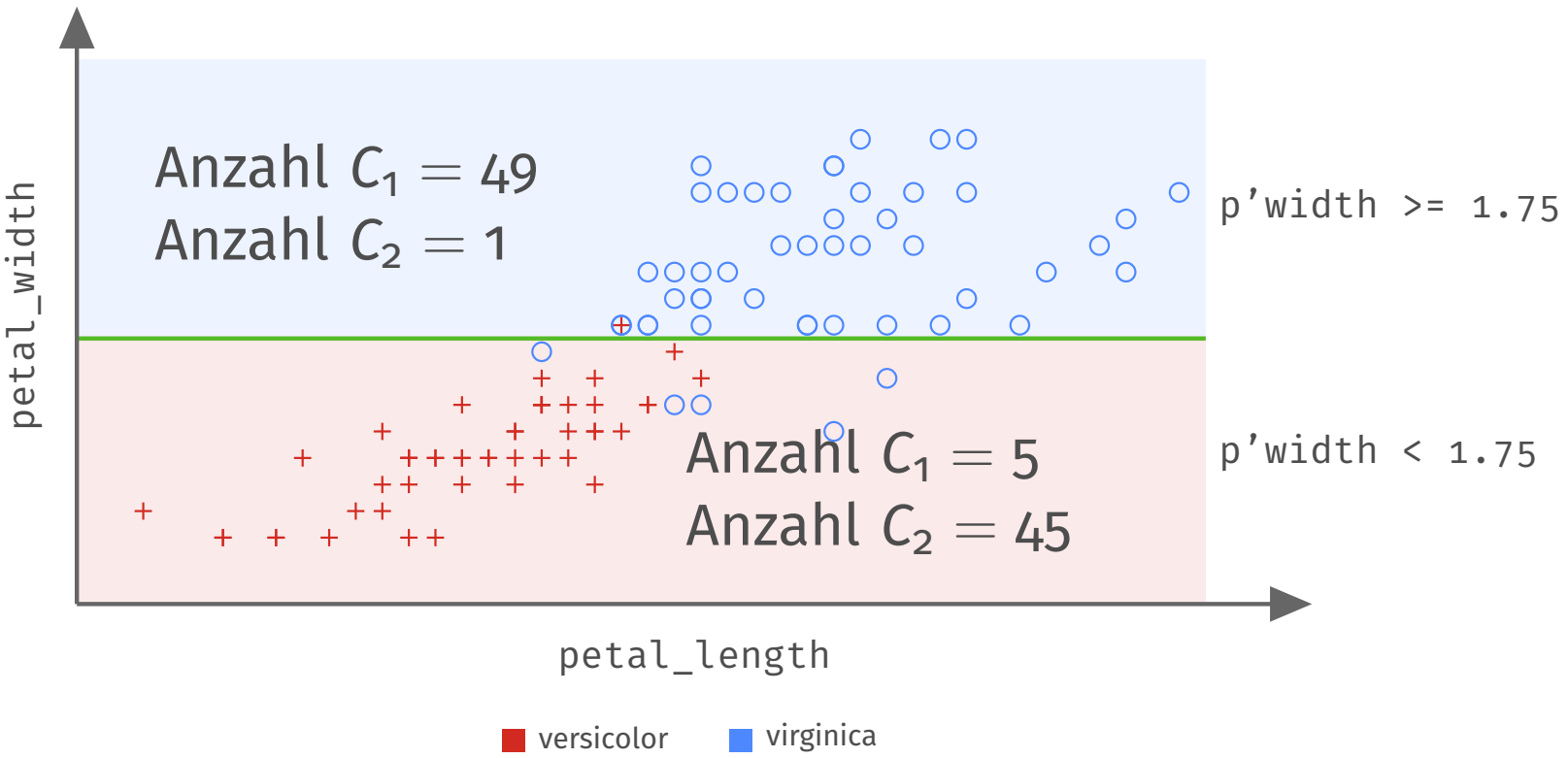
Idee: Daten teilen und W'keit für korrekte Vorhersage erhöhen



Idee: Daten teilen und W'keit für korrekte Vorhersage erhöhen



Idee: Daten teilen und W'keit für korrekte Vorhersage erhöhen



Einfache Vorhersage-Funktion

Die folgende Funktion berechnet die Vorhersage mit der Entscheidungsfunktion

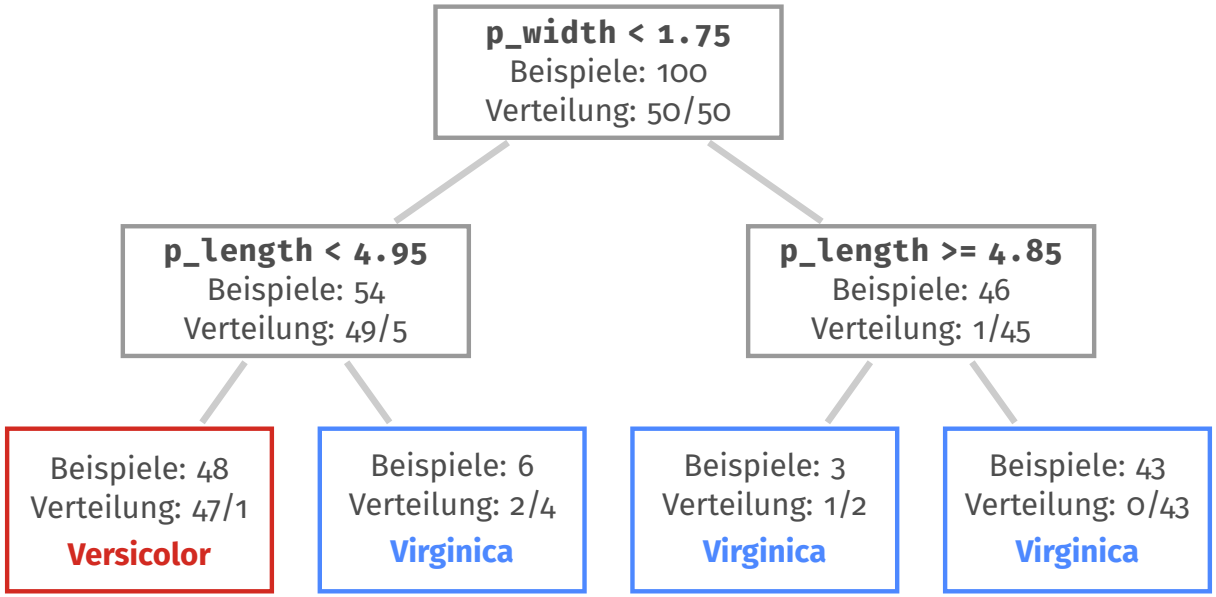
$$f(x) = \begin{cases} \text{versicolor} & \text{, falls } x[petal\_width] >= 1.75 \\ \text{virginica} & \text{, sonst.} \end{cases}$$

```
def _predict(row):
  if row["petal_width"] >= 1.75:
    return "versicolor"
  else:
    return "virginica"

def simple_predict(X):
  return [_predict(x) for i,x in X.iterrows()]
```

Entscheidungsbäume sind einfaches Modell

- Innere Knoten sind Entscheidungsknoten
- Blätter stellen Vorhersage dar



Frage: Wie finden wir gute Aufteilung der Daten?

- Gegeben sind die Trainingsdaten $\mathbf{X} = \mathbf{X}_{train} \times \mathbf{y}_{train}$
- Wir wollen einen Baum erstellen, der den Trainingsfehler minimiert
- Schrittweises Vorgehen (rekursiv):

Training eines Baumes:

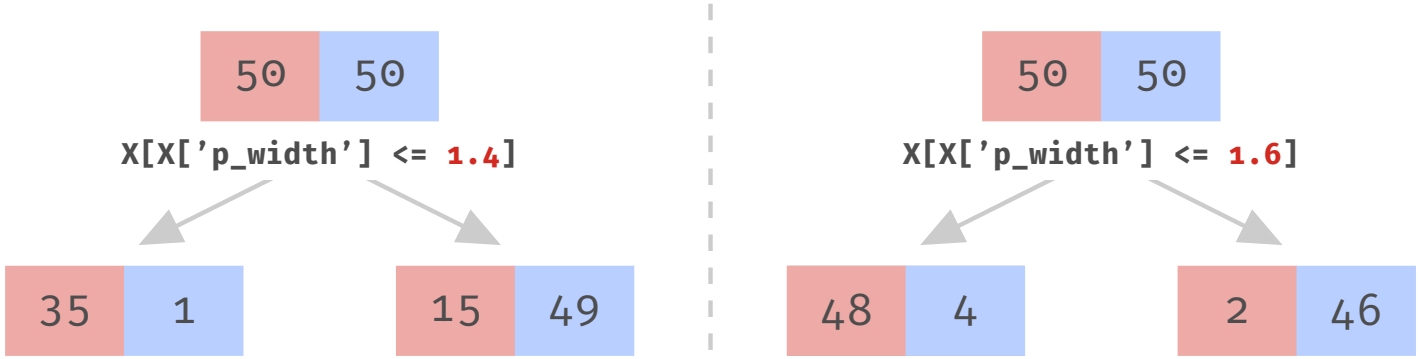
1. Finde bestes Attribut a^* und Wert t zum Aufteilen von $\mathbf{X}$
2. Teile $\mathbf{X}$ in $\mathbf{X}_{\mathbf{a} \leq \mathbf{t}}$ und $\mathbf{X}_{\mathbf{a} > \mathbf{t}}$

Frage: Wie finden wir das beste Attribut a und Wert t ?

- Wir probieren alle Attribute und deren Werte aus!

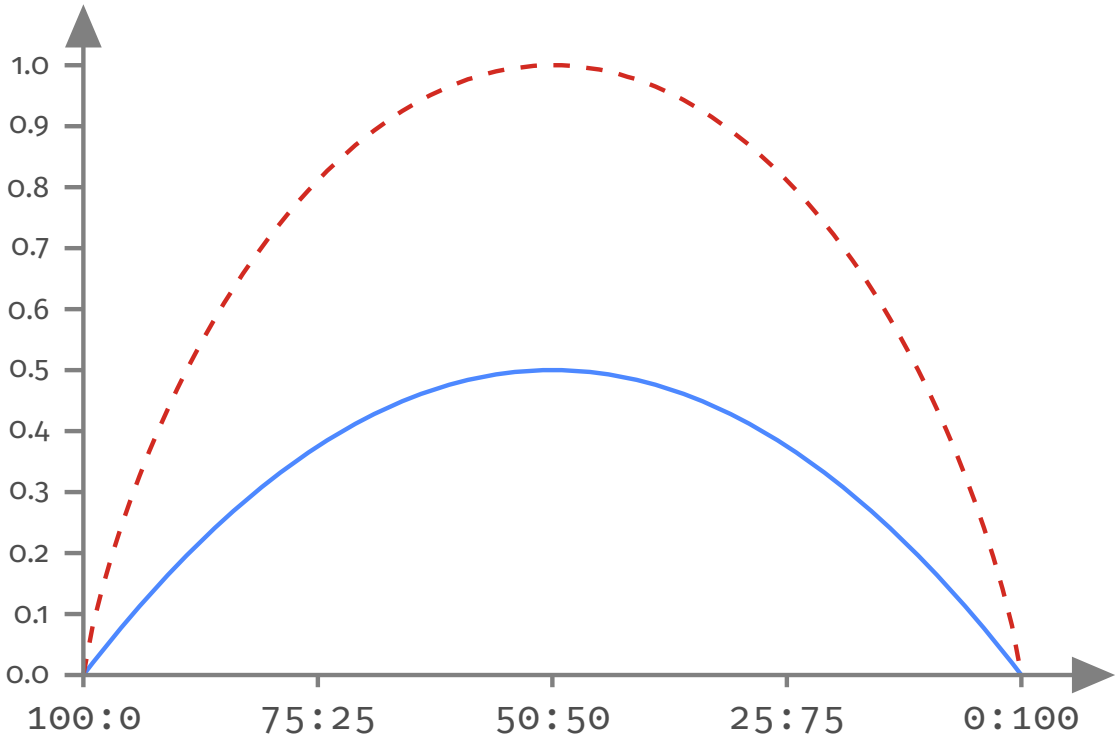
Beispiel: Iris Daten mit Klassen *virginica* und *versicolor*

- Datensatz hat Klassenverhältnis `50:50`
- Aufteilung nach Attribute `p_width` mit `t=1.4` und `t=1.6`



Aber welche Aufteilung ist besser?

Verlauf von Gini und Entropie



Die Grafik zeigt den Verlauf des Gini Index (blau) und der Entropy (rot) für verschiedene Klassenverhältnisse. Je *reiner* die Klassenaufteilung, desto kleiner sind die Werte für Gini, bzw. die Entropie.

Modell-Evaluation

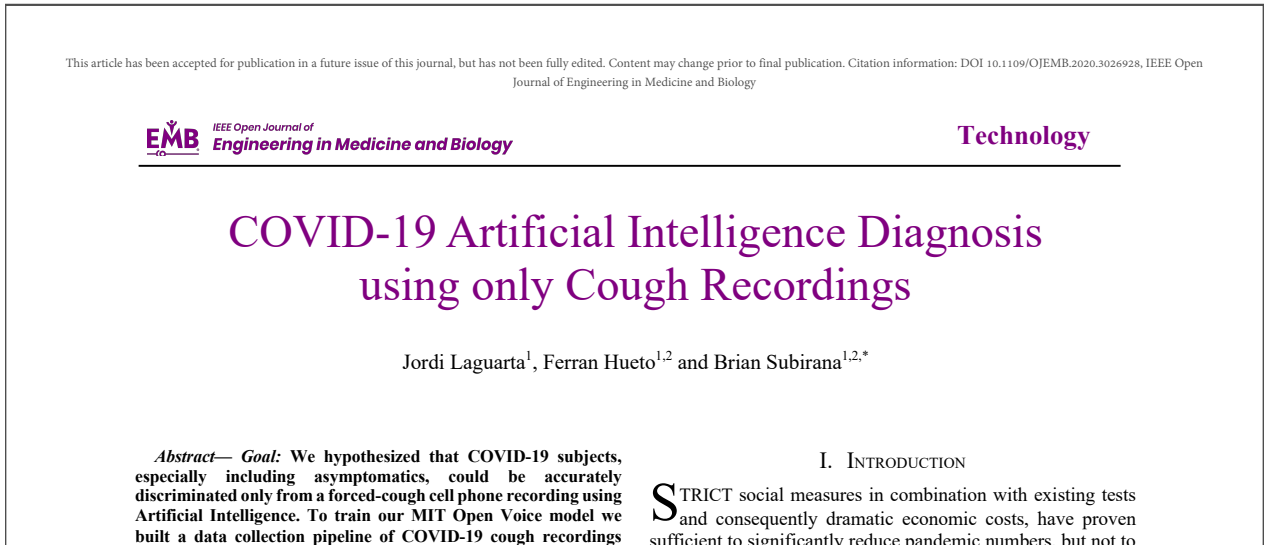
Wie gehen wir mit Klassifizierungsfehlern um?

- Evaluierung von Klassifizierern über confusion matrix
- Statistische Maße aus Fehlermatrix ableiten

Wie gehen wir mit Klassifizierungsfehlern um?

- Evaluierung von Klassifizierern über confusion matrix
- Statistische Maße aus Fehlermatrix ableiten

Beispiel: COVID-19 Artificial Intelligence Diagnosis...



This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJEMB.2020.3026928, IEEE Open Journal of Engineering in Medicine and Biology

IEEE Open Journal of Engineering in Medicine and Biology

Technology

COVID-19 Artificial Intelligence Diagnosis using only Cough Recordings

Jordi Laguarta[1], Ferran Hueto[1,2] and Brian Subirana[1,2,*]

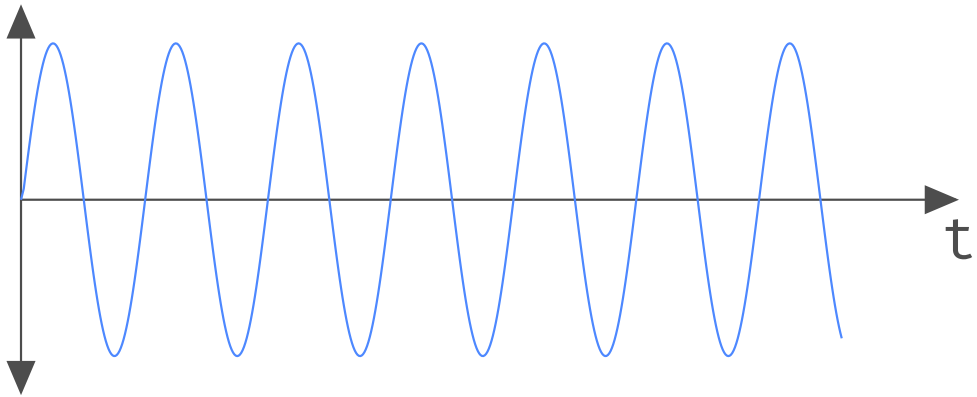
Abstract*— *Goal: **We hypothesized that COVID-19 subjects, especially including asymptomatics, could be accurately discriminated only from a forced-cough cell phone recording using Artificial Intelligence. To train our MIT Open Voice model we built a data collection pipeline of COVID-19 cough recordings**

I. Introduction

STRICT social measures in combination with existing tests and consequently dramatic economic costs, have proven sufficient to significantly reduce pandemic numbers, but not to

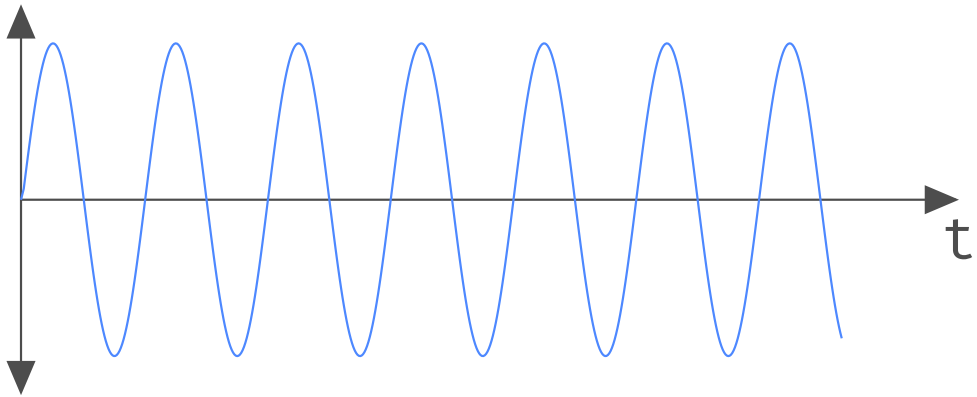
Datenquelle: Audio-Signal

Sinus-Welle bei 440 Hz



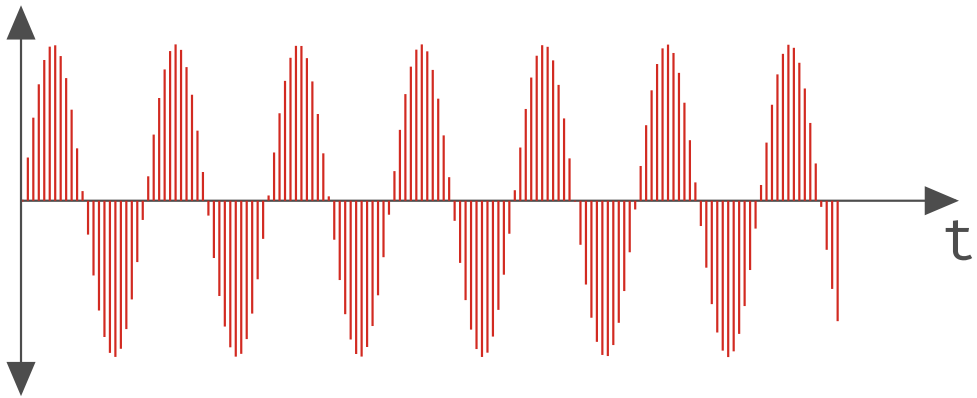
Datenquelle: Audio-Signal

Sinus-Welle bei 440 Hz – Kammerton “c”



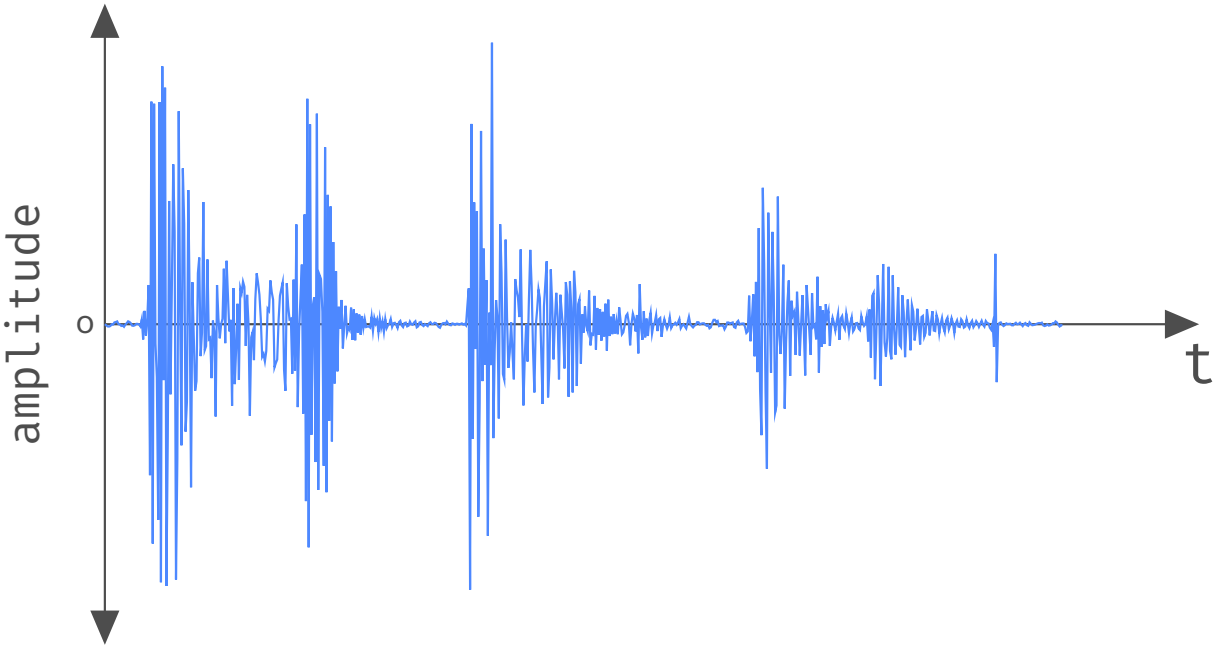
Datenquelle: Audio-Signal

Sinus-Welle bei 440 Hz – Kammerton “c”

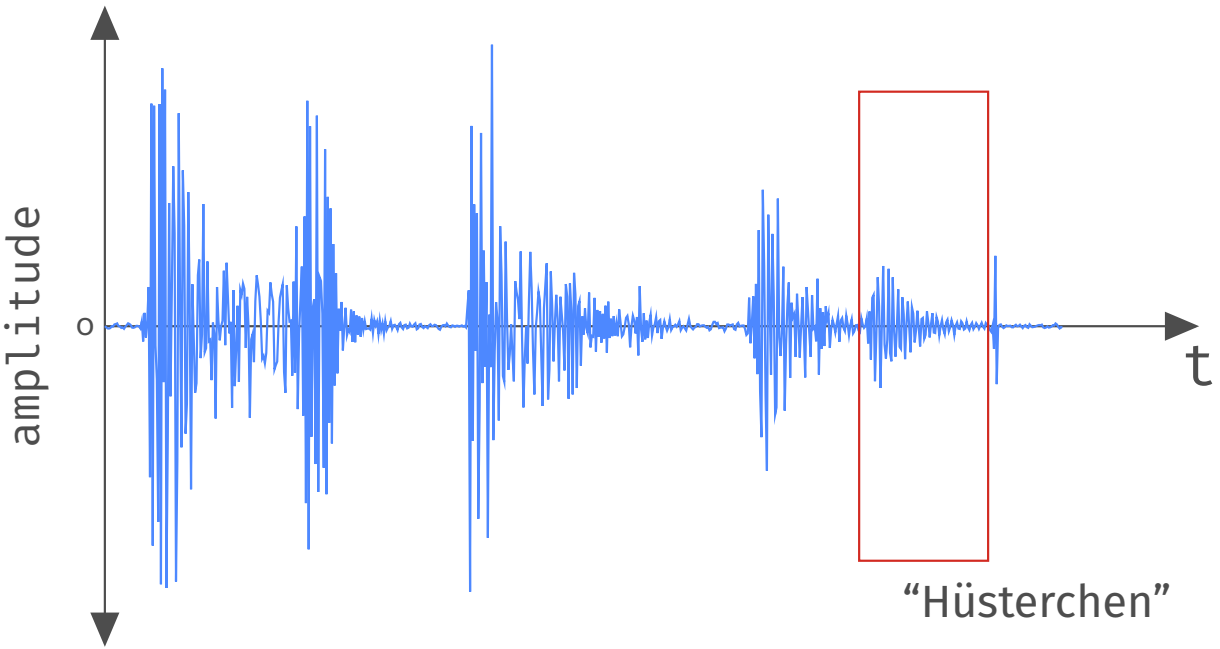


- Sampling Rate häufig 44.1 kHz, d.h. 44100 Werte $0 \leq x \leq 255$
- WAV Dateien enthalten Sampling Werte
- MP3 komprimiert Werte für kleinere Dateien

Audio-Signal für Sprache / Laute



Audio-Signal für Sprache / Laute



Husten-Erkennung

Wie erkennt man nun den “Husten”-Teil?

1. Definiere Husten-Muster / Form
2. Suche in den Daten (Samples) nach dem Muster

Husten-Erkennung

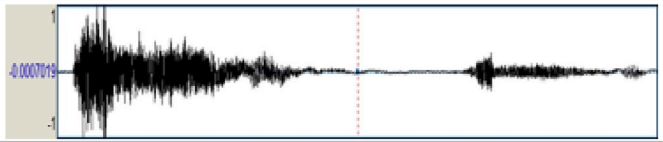
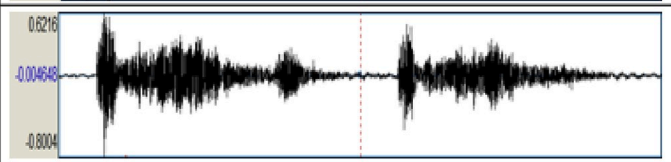
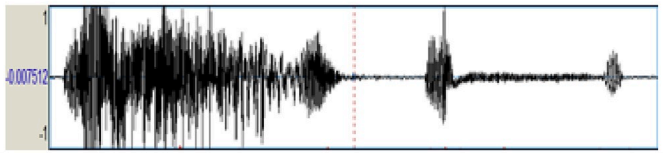
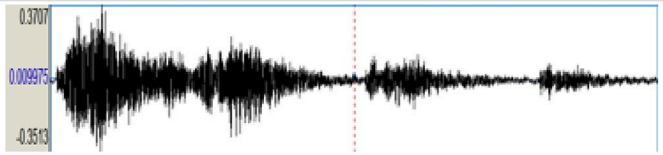
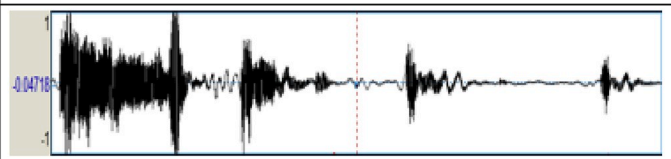
Wie erkennt man nun den “Husten”-Teil?

1. Definiere Husten-Muster / Form
2. Suche in den Daten (Samples) nach dem Muster

Und danach?

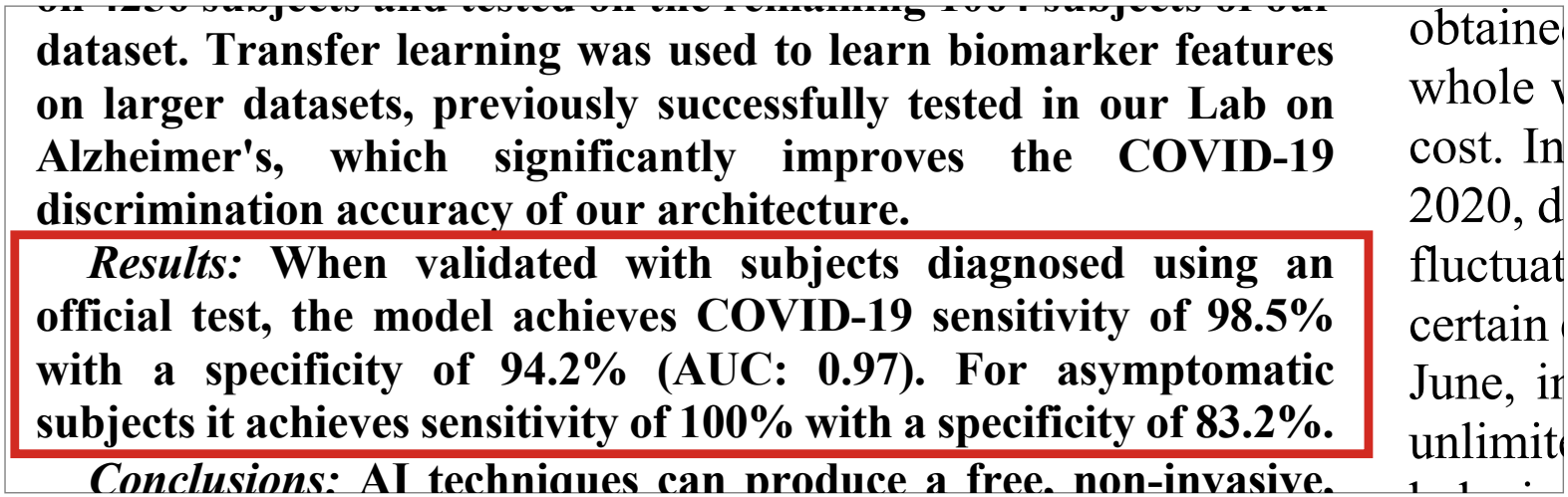
- Patienten mit Atemwegserkranken haben häufig unterschiedliche Charakteristiken beim Husten/Sprechen
- Stimmband-Eigenschaften ändern sich (z.B. Anfänglicher Luftdruck bei bestimmten Geräuschen)

Husten von Patienten mit unterschiedlichen Symptomen

Condition	Cough patterns	Duration
Normal airways		0.79 seconds
Narrowed airways (obstruction)		0.85 seconds
Widened airways (obstruction)		0.82 seconds
Scared lungs (restriction)		1.41 seconds
Fluid filled lungs (restriction)		1.82 seconds

Aus: *Cough sound analysis and objective correlation with spirometry and clinical diagnosis*, G. Rudraraju et.al., *Informatics in Medicine Unlocked 19 (2020)*

Zurück zur COVID-19 Erkennung



dataset. Transfer learning was used to learn biomarker features on larger datasets, previously successfully tested in our Lab on Alzheimer's, which significantly improves the COVID-19 discrimination accuracy of our architecture.

Results: **When validated with subjects diagnosed using an official test, the model achieves COVID-19 sensitivity of 98.5% with a specificity of 94.2% (AUC: 0.97). For asymptomatic subjects it achieves sensitivity of 100% with a specificity of 83.2%.**

Conclusions: **AI techniques can produce a free, non-invasive,**

obtaine
whole v
cost. In
2020, d
fluctuat
certain
June, ir
unlimit

Aus: *COVID-19 Artificial Intelligence Diagnosis using only Cough Recordings*, J.Laguarta, F.Hueto and B.Subirana, *Engineering in Medicine and Biology (Pre-Print)*

Paper gibt an: Sensitivity 98.5% und Specificity 94.2%

Was war damit gleich noch gemeint?

	Vorhersage ŷ		
"Wahrheit" y	**Klasse Pos**	**Klasse Neg**	
Klasse Pos	True Pos (TP)	False Neg (FN)	TP / (TP + FN)
Klasse Neg	False Pos (FP)	True Neg (TN)	TN / (FP + TN)
	TP / (TP + FP)	TN / (TN + FN)	

Paper gibt an: Sensitivity 98.5% und Specificity 94.2%

Was war damit gleich noch gemeint?

	Vorhersage ŷ		
"Wahrheit" y	**Klasse Pos**	**Klasse Neg**	
Klasse Pos	True Pos (TP)	False Neg (FN)	TP / (TP + FN) **Sensitivity**
Klasse Neg	False Pos (FP)	True Neg (TN)	TN / (FP + TN)
	TP / (TP + FP)	TN / (TN + FN)	

Paper gibt an: Sensitivity 98.5% und Specificity 94.2%

Was war damit gleich noch gemeint?

	Vorhersage ŷ			
"Wahrheit" y	**Klasse Pos**	**Klasse Neg**		
Klasse Pos	True Pos (TP)	False Neg (FN)	TP / (TP + FN)	**Sensitivity**
Klasse Neg	False Pos (FP)	True Neg (TN)	TN / (FP + TN)	**Specificity**
	TP / (TP + FP)	TN / (TN + FN)		

Paper gibt an: Sensitivity 98.5% und Specificity 94.2%

Was war damit gleich noch gemeint?

Vorhersage ŷ

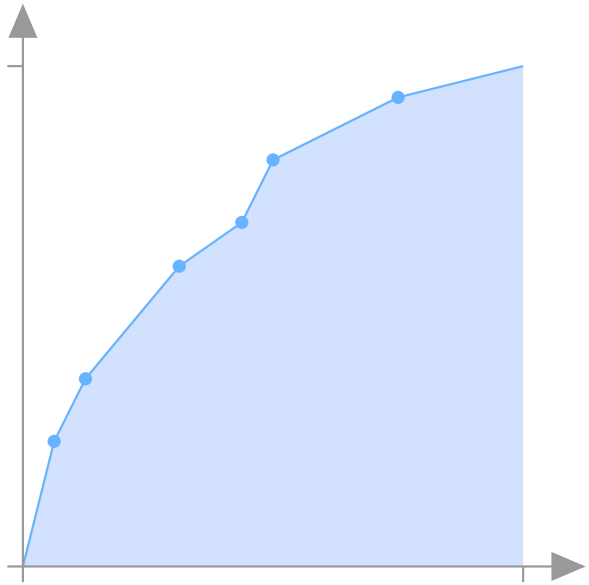
"Wahrheit" y	Klasse Pos	Klasse Neg		
Klasse Pos	True Pos (TP)	False Neg (FN)	TP / (TP + FN)	**Sensitivity**
Klasse Neg	False Pos (FP)	True Neg (TN)	TN / (FP + TN)	**Specificity**
	TP / (TP + FP)	TN / (TN + FN)		

Warum sind diese beiden so wichtig?

Kombination von Specificity und Sensitivity: AUC

True Positive Rate:

$$TPR = \frac{TP}{TP+FN}$$



False Positive Rate:

$$FPR = \frac{FP}{FP+TN}$$

Area under the ROC Curve

Wie geht's weiter?

Vorschau auf Vorlesung 6:

- **Gastvortrag** von **Jonas Rashedi**,
  Parfümerie Douglas (eCom)

Virtueller Raum für die Vorlesung 6:

`https://glight.hs-bochum.de/b/chr-nti-sjt`