

# Experimente gestalten fürs Maschinelle Lernen

## Übung 2: It's time for Machine Learning!

9. November 2021

Zeit für die Bearbeitung: Ein Tag vor dem nächsten Seminartermin, den *16.11. um 9 Uhr*.  
**Abgabe bei GitLab**

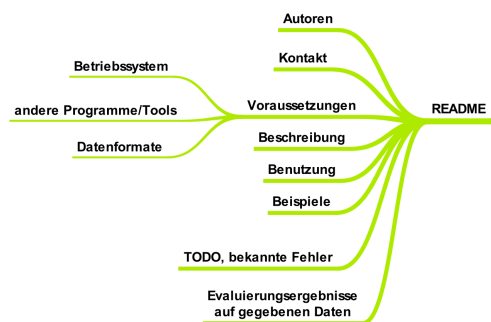
### Hinweise zur Gruppenarbeit (falls zutreffend)

Aufgabenteile mit **(A)** und **(B)** markiert müssen von unterschiedlichen Gruppenmitgliedern auf GitLab commitet werden.

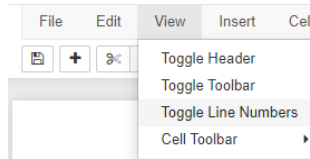
### Übung

In diesem Blatt werden wir (neben dem Arbeiten mit git) Folgendes lernen:

- fremden Code zu verstehen und uns davor nicht zu erschrecken
  - Messungen sind nicht ganz einfach
  - Machine Learning Algorithmen unterscheiden sich (u.A.) sehr in ihrer geometrischen Darstellungskraft.
1. Arbeite weiterhin im GitLab Repository, welches wir im 1. Übungsblatt erstellt haben. Erstelle darin einen neuen Ordner `02_übung` und bearbeite die folgenden Punkte ausschließlich in diesem Ordner.
  2. Nicht vergessen:
    - (a) die Code Guidelines aus der Vorlesung (siehe Slides mit Mindmap) zu beachten.
    - (b) für diesen Ordner ein `README.md` zu schreiben!



3. Auf Moodle finden sich ein Jupyter Notebook und eine Python Datei. Suche Dir eins der Formate aus um damit zu arbeiten. Hinweis: Auf Jupyter Notebook aktiviert man die Zeilennummerierung wie folgt:



- (a) **(A)** Mache Dich mit dem Code vertraut, identifiziere die wichtigen Schritte des Codes und halte diese im **README** fest mit Referenz auf die Zeilenintervalle (die Zeilennummern entsprechen der Version des Codes auf Moodle). Versuche dabei Dich nicht von der Komplexität des Codes oder der Unbekanntheit der Verfahren des Maschinellen Lernen erschrecken zu lassen.

Hinweis: In diesem Seminar ist der Code aus der Aufgabenstellungen weder perfekt, noch extrem einfach, noch ganz verständlich; wie im echten Leben so oft auch.

Hinweis: Beim Maschinellen Lernen ist der Workflow normalerweise wie folgt (nicht unbedingt genau in dieser Reihenfolge im Code):

- Daten und/oder Features laden oder erzeugen
- Daten in Train und Test Set aufteilen
- Machine Learning Algorithmus / Algorithmen (Classifier) definieren
- Classifier trainieren (auf Train Set) und evaluieren (auf Test Set)
- Ergebnisse visualisieren

Schiebe die Änderung auf den Server (**commit**, **push**).

- (b) **(B)** Nun beschäftigen wir uns mit der Zeitmessung der Ausführung des Codes:
- i. Mit dem Python Modul **time** oder mit einem Profiler Deiner Wahl, füge die notwendigen Zeilen Code ein, die die Zeit des Ausführens messen.
  - ii. Führe das Skript aus und halte im **README** fest, wie lange das Ausführen gedauert hat, mit den Angaben zu den *relevanten* Hardware Komponenten:
    - Prozessor (Frage für Experten<sup>1</sup>: Auf wie vielen Kernen lief der Prozess?)
    - Grafikkarte (Frage für Experten: Wie findet man heraus, ob ein Prozess die Grafikkarte benutzt?)
    - RAM (Direktzugriffsspeicher)
    - Daten von der SSD oder Festplatte (Hard Drive)?
  - iii. Reicht das einmalige Messen der Laufzeit? Wenn nein, wieso? Wenn notwendig, verbessere Deine Zeitangabe entsprechend; dabei nicht vergessen genau zu sagen was die Zeitangabe bedeutet (also auch wie (oft) sie ermittelt wurde). Was könnte uns diese Einsicht über das Testen von Machine Learning Algorithmen sagen?

Schiebe die Änderung auf den Server (**commit**, **push**).

- (c) **(A)** Das Skript gibt eine Visualisierung aus. Diese enthält das Ergebnis von 7 verschiedenen Machine Learning Algorithmen auf 3 unterschiedlichen Datensätzen. Halte im **README** fest:
- i. Was charakterisiert, bzw. differenziert die unterschiedlichen Datensätze?
  - ii. Aus den 7 Algorithmen werden wir im Seminar nur den Decision Tree (bzw. Random Forest) und den Naive Bayes näher kennen lernen. Wie unterscheiden sich die 7 Algorithmen voneinander, bezogen auf deren Geometrie / Form der "decision boundary", also der Linie, die die blauen von den roten Daten trennen sollte?
  - iii. Was bedeutet "linearly separable"(Google)? Welcher der 3 Datensätze ist linear separabel?

Schiebe die Änderung auf den Server (**commit**, **push**).

Hinweis: Wir werden diese Übung zum nächsten Seminartermin kurz besprechen. Danach dürft / solltet Ihr Euer git Repository noch verbessern.

#### 4. Mache Dir schon Gedanken zu Daten und möglichen Projektaufgaben.

---

<sup>1</sup>d.h. freiwillig