Maschinelles Lernen
Aufgabenblatt 01b

Prof. Dr. Christoph Böhm
Hochschule München

5. April 2024

Aufgabe 1.1. Ihre Aufgabe ist es den klassischen (vorbereitenden) Workflow eines Data Scientists nachzuvollziehen. Konkret beinhaltet dies, die folgenden Aufgaben.

1. Laden der Daten:

   Laden Sie die Daten aus `adult.data` in einen Pandas DataFrame. Verfügbar über `https://archive.ics.uci.edu/dataset/2/adult`

2. Datenaufbereitung:

   (a) In den nominalen Daten sind noch unbekannte Werte gekennzeichnet durch `'?'` vorhanden. Bereinigen Sie die Daten, indem Sie alle Zeilen entfernen, die unbekannte Werte enthalten.

   (b) Entfernen Sie die Spalten `fnlwgt` und `income` als Features. Löschen Sie zudem kategoriale Features mit sehr vielen Kategorien.

   (c) Als Target soll das Feature `income` dienen, jedoch kommt nicht jeder Algorithmus mit nominalen Features klar. Konvertieren Sie das Target daher, sodass `income` den Wert 1 annimmt, falls das `income` ursprünglich den Wert `'>50K'` hat und 0 andernfalls. Speichern Sie dies in einem eigenen DataFrame oder Array `y`.

   (d) Wieviel Prozent der Personen haben ein Einkommen von mehr als 50.000$?

   (e) Was ist die Genauigkeit eines naiven Klassifikation-Modells, welches unabhängig von den tatsächlichen Features immer weniger als 50.000$ Einkommen zuweist? Dies ist das Mindestmaß an Genauigkeit, an dem sich ihre späteren Modelle messen müssen.

3. Wählen Sie geeignete Visualisierungen um interessante Aspekte im Datensatz zu beschreiben und zu plausibilisieren.

4. Datentransformation:

   Schreiben Sie eine Methode `transform(X)` welche einen Feature-DataFrame `X` als Parameter erhält und einen transformierten DataFrame zurückgibt, bei dem

   - Die im Wertebereich verzerrten Features `capital_gain` und `capital_loss` sollten durch Logarithmierung normalisiert werden. Verwenden Sie hierfür die Funktion

     $$f : \mathbb{R} \to \mathbb{R}, f(x) = \log(x + 1)$$

   - Anschließend sollen alle numerischen Features, d.h. `age`, `education_num`, `capital_gain`, `capital_loss`, `hours_per_week` auf den Wertebereich $[0, 1]$ normalisiert werden.
   - Transformieren Sie alle nominalen Features via *one-hot-encoding*. Informieren Sie sich hierzu über die Methode `get_dummies` (von Pandas).
   - Transformieren Sie ihre Features mit Hilfe der Methode `transform`.
   - Splitten Sie den Datensatz in einen Trainings- und Testdatensatz, wobei der Testdatensatz eine relative Größe von 20% haben soll. Verwenden Sie für die Reproduzierbarkeit einen `random_state=0`.