

Lab 10: Linear and Logistic Regression

Im heutigen Praktikum werden Sie ein lineares Regressionsmodell und einen binären Klassifikator trainieren.

1. Lineare Regression mittels Linear Least Squares

Lösen Sie die folgende (alte Klausur-) Aufgabe:

Aufgabe 5: Kurzfragen.

6 Punkte

- a) Jemand hat die Schallgeschwindigkeit c in Luft bei verschiedenen Temperaturen T gemessen und folgende Werte erhalten:

n	T_n	C_n
1	5°C	333.7 m/s
2	10°C	337.7 m/s
3	15°C	339.4 m/s
4	20°C	342.5 m/s
5	25°C	346.6 m/s

- a₁) (2P) Bestimmen Sie mit Hilfe der LS-Koeffizienten b_0 und b_1 die «exakte» lineare Abhängigkeit der Schallgeschwindigkeit $c = b_0 + b_1 \cdot T$ von der Temperatur T .
- a₂) (1P) Wie gross ist dann die «exakte» Schallgeschwindigkeit c bei 20°C?

2. Binäre Klassifikation mittels Logistische Regression

Im Folgenden implementieren Sie eine binäre Klassifikation mithilfe eines logistischen Regressionsmodells. Dabei wird ein Klassifikator trainiert, der anhand der beiden Features «hoursStudiedforSEP» und «midtermGrade» vorhersagt, ob ein Studierender die Semesterendprüfung (SEP) bestehen wird.

Gegeben sei das File `SEP_Training_Data.csv`, welches drei Spalten enthält: die Anzahl Stunden, die für die Semesterendprüfung gelernt wurde, die Vornote der Zwischenprüfung, sowie das Label «pass», also ob die SEP bestanden wurde.

- a) Visualisieren Sie die Trainingsdaten mittels eines Scatterplots. Aus diesem Plot soll das Label des zugehörigen Datenpunkts ersichtlich sein.
- b) Trainieren Sie von Hand (Gradient Descent Formeln aus den Slides) ein logistisches Regressionsmodell mit den gegebenen Trainingsdaten.
- c) Wenden Sie das gelernte Modell auf den Testdaten an und vergleichen Sie die Vorhersagen Ihres Regressionsmodells mit den echten Labels.

- d) Verwenden Sie nun `sklearn.linear_model.LogisticRegression` aus der `scikit-learn` Library, um ein logistisches Regressionsmodell zu fitten.
- e) Falls Sie noch Zeit haben: Probieren Sie weitere Klassifikationsmodelle der `scikit-learn` Library aus (z.B. Random Forests, Support Vector Machines, Fully-connected Neural Networks, ...).