

Deep Learning

Wintersemester 2024/25

Übungsblatt 4

Besprechung am 05.11.2024

Aufgabe 1 – EDF und Cross Entropy Loss

In der letzten Vorlesung haben Sie gelernt, dass der Binary Cross Entropy Loss (BCE Loss) ein Spezialfall des allgemeinen Cross Entropy Loss (CE Loss) ist. Das EDF enthält bereits eine Implementierung eines Nodes für den CE Loss, nämlich `CrossEntropyLoss`.

Allerdings erwartet dieser Node als Input eine diskrete Wahrscheinlichkeitsverteilung, also eine Matrix p mit Dimensionen $B \times 2$, wobei jede Zeile $p[i]$ einem Vektor mit diesen Wahrscheinlichkeiten für einen Datenpunkt \mathbf{x}_i entspricht, also:

- $p[i][0] = P(Y = 0 | X = \mathbf{x}_i)$
- $p[i][1] = P(Y = 1 | X = \mathbf{x}_i)$

Bisher war der Output unserer Computation Graphs für Klassifizierungen die Ausgabe eines Sigmoid Node, also ein Vektor mit B Einträgen zwischen 0 und 1.

Damit wir also den bereitgestellten CE Loss Node nutzen können, müssen wir einen neuen Computation Node `SingleProbToProbVector` erstellen, der folgende Konvertierung durchführt:

- Eingabe: $B \times 1$ Matrix mit den Werten $P(Y = 1 | X = \mathbf{x}_i)$
- Ausgabe: $B \times 2$ Matrix mit den Werten $P(Y = 0 | X = \mathbf{x}_i)$ und $P(Y = 1 | X = \mathbf{x}_i)$

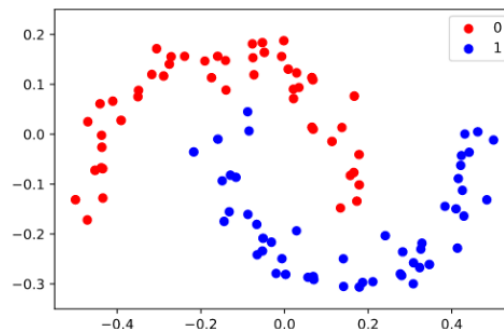
für alle $i \in \{1, \dots, b\}$. Wenn die Eingabe z also eine $B \times 1$ Matrix ist, soll die Ausgabe o eine $B \times 2$ Matrix sein mit $o[i][0] = z[i][0]$ und $o[i][1] = 1 - z[i][0]$ für $i \in \{0, \dots, B - 1\}$.

Im Notebook `logistic_regression.ipynb`:

- Implementieren Sie die Klasse `SingleProbToProbVector` nach den oben genannten Vorgaben
- Definieren Sie Computation Graphs für die verschiedenen Auflösungen der Datensätze (also 1×1 , 4×4 , 8×8 und volle Auflösung), welche die Helligkeitswerte als Eingabewerte nimmt und logistische Regression durchführt.

Aufgabe 2 – Binäre Klassifikation von 2D-Punkten

Jetzt wechseln wir von Daten mit einer Dimension zu zwei Dimensionen. Dazu finden Sie im Jupyter Notebook `pcl_binary_classification.ipynb` den Code zum Laden eines 2D-Datensatzes, der folgendermaßen aussieht:



Sie sollen dazu ein Modell mit logistischer Regression trainieren, welches alle Datenpunkte korrekt klassifiziert. Die Angaben im Notebook sollten ausreichen, aber im Folgenden werden noch mal die Aufgaben explizit genannt:

- Trainieren Sie ein einfaches Modell mit logistischer Regression auf dem Datensatz und nutzen Sie wieder den Cross Entropy Loss wie in Aufgabe 1.
- Visualisieren Sie die gefundene Lösung. Der Code zum Plotten ist bereits vorgegeben.
- Sie sollten sehen, dass Ihr Modell die Daten nicht korrekt abbilden kann. (Falls Sie nicht wissen wieso, schauen Sie sich noch einmal die Vorlesungsunterlagen an.)
Konstruieren Sie nun ein Multilayer Perceptron (MLP) mit einem Hidden Layer, welches die Aufgabe lösen soll. Machen Sie sich dazu klar, wie der Computation Graph aussehen sollte und fügen Sie dann die entsprechenden Nodes zusammen.

Es ist noch weiterer Code vorgegeben, um das Ergebnis zu visualisieren.

Wie immer gilt: Sollten Fragen auftauchen, melden Sie sich bitte im Forum.

Viel Erfolg!