

## Aufgabenblatt 2: Fashion-MNIST Klassifikation

### Ziel

Implementiere Modell bestehend aus einem Softmax Layer und maximal einem zusätzlichen Hidden Layer und evaluiere dessen Performance.

### Arbeitsschritte

1. Aufsetzen des DL-Frameworks Deiner Wahl.
2. Implementiere Funktionalität um den [Fashion-MNIST Datensatz](#)<sup>1</sup> zu laden. Untersuche, wie Du den Datensatz in Train-Validate-Test aufsplitten kannst. Normalisiere die Daten.
3. Implementiere zuerst ein Modell bestehend nur aus einem Softmax Layer und verwende fürs Training die Cross-Entropy-Kostenfunktion. Stelle sicher, dass Du (mindestens) die Kostenfunktion und die Accuracy sowohl für die Train und Validate pro Epoche ausgeben kannst und damit Du nachher beim Training die Lernkurven plotten kannst.
4. Führe nun das Training durch:
  - a. Initialisiere die Gewichte in geeigneter Weise: Bei  $n$  Input-Features wähle  $w_{i,j} \sim N(0, 1/\sqrt{n})$  und  $b_j = 0$ . Warum ist diese Wahl "geeignet"?
  - b. Führe zuerst Batch-Gradient Descent durch. Wie wählst Du die Lernrate? Über wie viele Epochen lässt Du das Training laufen? Nach welchen Kriterien entscheidest Du das?
  - c. Überprüfe, dass der Wert der Kostenfunktion am Anfang des Trainings am "richtigen" Ort startet (konsistent mit der Initialisierung der Gewichte). Wie bestimmst Du diesen Startpunkt?
  - d. Führe mehrere (5-10) Trainings mit unabhängig gezogenen Anfangsgewichte durch. Plote die resultierenden Lernkurven. Ermittle daraus eine mittlere Lernkurve und einen Fehler (ausgedrückt durch die Standardabweichung).
  - e. Führe nun Mini-Batch-Gradient (z.B. batchsize=32) aus und beantworte auch hier wieder die Fragen aus b. Vergleiche Anzahl Update-Schritte und Anzahl Epochen mit der Anzahl Epochen aus b. Hast Du eine andere Lernrate gewählt als in b.?
  - f. Entscheide Dich für ein finales bestes Modell.
5. Für Dein finales bestes Modell erstelle die Confusion-Matrix und berechne
  - a. Class Accuracy, Precision, Recall, F1-Score
  - b. System Accuracy, Precision, Recall, F1-Score
6. Verwende Matplotlib (o.ä.), um die gelernten Gewichte pro Klasse als Bild zu plotten. Beachte, dass für jede Klasse ein Array von Gewichten definiert ist in der Länge der Anzahl in den Fashion MNIST enthaltenen Pixel. Das erlaubt hier diese Klassen-Gewichte in Bilder zu "reshapen".

---

<sup>1</sup> Alternativ, kannst Du auch ein anderes, für Klassifikation geeignetes, nicht zu grosses Datenset wählen, z.B. CIFAR10.

7. Wiederhole nun die Schritte 3-5, dieses Mal mit einem zusätzlichen Hidden Layer.  
Dabei untersuche verschiedene Grössen ( $n$ ) des Hidden Layers:  $n = 10, 20, 50, 100, 200, \dots$  Mit welcher Anzahl  $n$  erhältst Du die beste Performance?  
Entscheide Dich auch hier wieder für eine finales Modell.
8. Beurteile schliesslich die erzielten Ergebnisse für nur Softmax und Softmax + 1 Hidden Layer auch anhand der Accuracy, Precision und Recall.

## Materialien

- Slides aus Vorlesung MSE Deep Learning.
- Nach Bedarf ein Jupyter-Notebook mit Gerüst für ein mögliches Vorgehen.