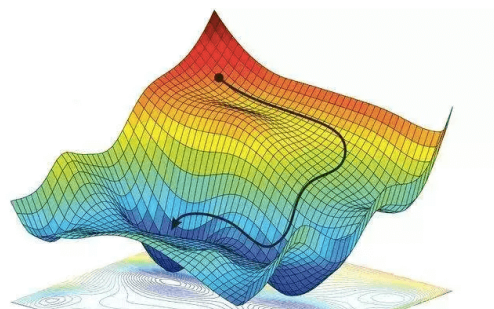
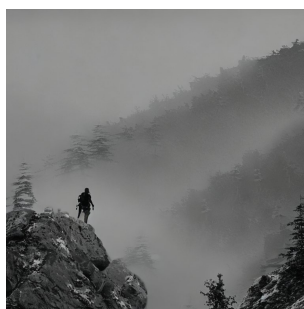


Mini Challenge

Gradient Descent



Das Ziel dieser Aufgabe besteht darin, dass Sie ein grundlegendes Verständnis für numerische Näherungsverfahren in höheren Dimensionen erlangen, insbesondere für den Gradient Descent und dessen praktische Anwendung. Hierfür sollen Sie ein Jupyter Notebook erstellen und das MNIST Dataset laden und erkunden. Anschließend sollen Sie ein neuronales Netzwerk erstellen und trainieren, um die Bilder korrekt zu klassifizieren. Es dürfen nur die angegebenen Python packages verwendet werden.

Ziel dieser Aufgabe ist nicht nur, Ihre mathematischen Kenntnisse unter Beweis zu stellen, sondern auch die entsprechende Kommunikation und Präsentation Ihrer Ergebnisse. Ihre Abgaben sollen also nicht nur **mathematisch korrekt**, sondern auch **leicht verständlich** und **reproduzierbar** sein. Genauere Angaben zu den Erwartungen an die Abgabe finden Sie in den Auswertungskriterien. Dokumentieren Sie ihren Arbeitsfortschritt und Erkenntnisgewinn in Form eines Lerntagebuchs, um Lernfortschritte, Schwierigkeiten und Erkenntnisse festzuhalten.

Die folgenden Aufgabenstellungen präzisieren die einzelnen Bearbeitungsschritte und geben die Struktur des Notebooks vor.

Aufgabe 1.

Laden Sie das MNIST-Dataset mithilfe des torchvision-Pakets (Verwenden Sie das torchvision Paket für diese Aufgabe) und verwenden Sie matplotlib, um sich einen Überblick über die Daten zu verschaffen. Beschreiben Sie das grundlegenden Eigenschaften des Datensets, z.B. wie viele und welche Daten es enthält.

Aufgabe 2.

Erstellen Sie eine Klasse für ein lineares Layer mit beliebig vielen Knoten. Implementieren Sie die Methoden forward, backward und update mithilfe von numpy. Schreiben sie geeignete Unittests, um die Funktionsweise der Funktion zu prüfen.

Aufgabe 3.

Erstellen Sie ein neuronales Netzwerk in numpy mit einem Hidden Linear Layer und einem Output Knoten. Trainieren Sie das Netzwerk darauf, die Ziffer 4 korrekt zu identifizieren (d.h. der Output soll 1 für diese Ziffer und 0 für alle anderen Ziffern sein). Trainieren Sie das Netzwerk auf den Trainingsdaten und evaluieren Sie es anhand von Testdaten. Verwenden Sie eine geeignete Loss-Funktion sowie Accuracy-Funktion und geben Sie deren mathematische Definition an. Begründen Sie Ihre Wahl mit einer Abwägung der Vor- und Nachteile. Diskutieren Sie kurz weitere Optionen für Loss und Accuracy.

Aufgabe 4.

Trainieren Sie das Netzwerk mit verschiedenen Lernraten und Größen des Hidden Layers. Verfolgen Sie während des Trainings die Entwicklung der Loss- und Accuracy-Funktionen auf Trainings- und Testdatensätzen und entscheiden Sie, welche Wahl von Lernrate und Hidden Layer-Größe die besten Ergebnisse in geringster Zeit liefert.

Aufgabe 5.

Erweitern Sie das Netzwerk auf 3 Hidden Layer mit gleicher Größe und 10 Outputs. Das Ziel ist die korrekte Klassifizierung aller Ziffern. Verwenden Sie eine geeignete Loss-Funktion sowie Accuracy-Funktion und geben Sie deren mathematische Definition an. Begründen Sie Ihre Wahl und diskutieren Sie kurz weitere Möglichkeiten. Variieren Sie die Lernrate und die Größe der Hidden Layer und wählen Sie das beste Ergebnis aus.

Chatgpt:

Für diese Aufgaben kann ChatGPT als Ressource genutzt werden, um Fragen zu stellen oder bei Problemen Unterstützung zu erhalten. Voraussetzung ist, dass Sie transparent kommunizieren, wo und wie Sie ChatGPT eingesetzt haben und welche Verbesserungen nötig waren. Sie können ChatGPT wie folgt nutzen:

- Stellen Sie Ihre Fragen klar und präzise.
- Formulieren Sie Ihre Fragen so, dass sie spezifisch auf Ihre Probleme abgestimmt sind.
- Seien Sie geduldig und geben Sie ChatGPT genügend Zeit, um eine sinnvolle Antwort zu generieren.
- Verwenden Sie die Antworten von ChatGPT als eine Art von Hilfestellung und überprüfen Sie diese stets auf ihre Richtigkeit.

Wenn Sie eine Frage stellen, die auf eine bestimmte Stelle in Ihrem Code Bezug nimmt, können Sie den Code zusammen mit Ihrer Frage bereitstellen.