

Übungsblatt 7

(24. Dezember 2010)

Projektaufgabe 20 Punkte

Aufgabe 1 Sie sollen in diesem Übungsblatt ein kleines C++ Projekt realisieren. Es handelt sich hierbei um eine Anwendung neuronaler Netze. Der GUI-Anteil ist mit der Bibliothek Qt4 zu erstellen.

Um die Aufgabenstellung umfassend verstehen zu können, ist es hilfreich, sich zunächst in das Thema Neuronale Netze (wir beschränken uns auf einfache Feedforward-Netze, sogenannte Perceptrons) einzulesen. Dazu steht eine Kurzeinführung auf der Internetseite bereit.

Aufgabe 1

Laden Sie sich die Klassen für die Erzeugung von Perceptron-Netzwerken herunter (<http://panitz.name/neural.tar.gz>) herunter und machen Sie sich zunächst mit der Klasse Perceptron vertraut. Wie kann die Klasse instanziiert werden? Wie könnte man eine eigene Aktivierungsfunktion und deren Ableitung angeben?

Aufgabe 2

Es befinden sich in der heruntergeladenen Datei auch Beispiele für die Implementierung einfacher boolescher Funktionen (AND, OR) welche keinen Hiddenlayer verwenden. Entwerfen Sie nun ein Perceptron, welches mit Hilfe mindestens eines Hiddenlayers die XOR-Funktion nachbildet. Dabei steht es Ihnen frei, die Gewichtung per Hand durchzuführen oder das Netz entsprechend zu trainieren. Beim Training kann es hilfreich sein, für einzelne Layer ein zusätzliches Bias-Neuron zu aktivieren.

Aufgabe 3

Erzeugen Sie nun eine eigene Netz-Klasse, die von der Klasse Perceptron erbt. Statten Sie Ihre Klasse mit folgenden Funktionen aus:

- Rücksetzen aller vorhandenen Gewichte auf jeweils einen Zufallswert zwischen $[-1.0; 1.0]$.

- Implementieren Sie eine Methode, mit der es möglich ist, ein komplettes Perceptron (Ihre Klasse) inkl. aller Gewichte in eine Datei zu schreiben.
- Implementieren Sie eine Methode, mit der es möglich ist, ein komplettes Perceptron (Ihre Klasse) inkl. aller Gewichte aus einer Datei zu rekonstruieren.

Aufgabe 4

Erzeugen Sie nun ein Perceptron, um ein komplexeres Problem zu lösen. Dabei ist eine der folgenden Anwendungen möglich:

- Erkennen von einzelnen Buchstaben oder Ziffern, mit der Maus gezeichnet oder aus Bildschirmschriften.
- Erkennen von Gesichtern (nur einzelne Merkmale) z.B. Lange Haare/kurze Haare, Geschlecht, Brille/keine Brille etc.
- Unterscheidung primitiver (nicht rotierende) geometrische Objekte (Kreis, Dreieck, Rechteck) ggf. Erkennung in einem „gestörten“ Bild.

Grundsätzlich sind auch gerne eigene Ideen möglich. Wichtig ist bei dieser Aufgabe jedoch ein geeignetes Training. Dabei spielt die Qualität der Trainingsdaten eine große Rolle. Informieren Sie sich ggf. in der Literatur über *common* Werte für epsilon (Lernrate) und die Anzahl der Trainingszyklen. Experimentieren Sie! Sollte das Netzwerk für Ihre Anwendung nur schlechte Ergebnisse liefern (kein Problem), dokumentieren sie dies nur entsprechend.

Aufgabe 5

Entwerfen Sie eine kleine GUI mit QT, mit der Sie das neuronale Netz „füttern“ können. Stellen Sie ausserdem die Netzausgabe entsprechend dar. Erzeugen Sie Menüpunkte zum Laden und Speichern und Zurücksetzen des Netzes, sowie zum Öffnen einer Aboutbox (Name, MatrikelNr, Programmname etc.)

Aufgabe 6 (Zusatzaufgabe)

Erweitern Sie Ihre GUI so, dass der Verlauf des Trainings in irgendeiner Form einsehbar wird. Beispielsweise können Sie ein Diagramm über den *Fehler des Netztes* nach jedem Trainingszyklus erstellen. Messen Sie dabei auch die Performance Ihres Perceptrons.

Abzugeben ist eine tar-Datei, die einen Ordner mit ihren Namen (nachnameVorname) enthält. Dieser Ordner soll enthalten:

- ein Ordner `src` mit den Quelltext Ihres Programms
- Ein Makefile, mit dem die Applikation gebaut werden kann.
- Ein kleines PDF-Dokument (2-5 Seiten), in dem Ihr Name, Ihre Matrikelnummer, das einen Überblick über die Funktionalität und der Gesamtarchitektur ihres Programms gibt.

Die Abgabe für die Übungsgruppen bei Herrn Panitz hat datenelektronisch an an die Emailadresse `abgabe@panitz.name` zu erfolgen. Die übrigen Übungsgruppen bekommen den Abgabemodus von ihren Dozenten mitgeteilt. Beachten Sie, dass keine ausführbaren Dateien oder Objektdateien in dem abgegebenen Ordner enthalten sein sollen. Ansonsten wird die Mail zurückgewiesen.

Das Projekt ist an einem gemeinsamen Termin in einen 5-minütigen Kurzvortrag vorzuführen, hierbei sind eine bis maximal 3 Präsentationsfolien zu erstellen.

Der genauen Präsentations- und Abgabetermin wird in den einzelnen Übungsstunden festgelegt.

Es sind Gruppenarbeiten in maximal Zweiergruppen erlaubt. Abgaben von Zweiergruppen müssen deutlich machen, welcher der beiden Mitarbeiter für welche Programmteile primär verantwortlich ist. Abgaben von Zweiergruppen müssen in Umfang und Qualität Einzelabgaben übertreffen und werden entsprechend schärfer bewertet. Präsentationen von Zweiergruppen müssen entsprechend auch von beiden teilnehmern gemeinsam gehalten werden und haben die doppelte Präsentationszeit zur Verfügung.

Es wird bewertet nach folgenden Kriterien:

- objektorientierter Klassenentwurf
- adäquate Benutzung von Zeiger- und Referenztypen
- korrekte Speicherverwaltung
- allgemeine Code-Qualität (Layout, Bezeichner, Funktionslängen)
- Dokumentation (Doxygen)
- Code Kommentierung
- Funktionalität/Ausbaustufe
- Benutzerführung
- Präsentation, Stil und Auftreten