

1 Backpropagation: Hidden Layer

(1 Punkt)

In der Vorlesung wurde(n) die Delta-Regel bzw. die Gewichtsupdates bei der Backpropagation für die Ausgabeschicht und den davor liegenden letzten Hidden-Layer unter Verwendung der Transferfunktion $g(a) = 1/(1 + \exp(-a))$ hergeleitet.

Leiten Sie die Gewichtsupdates für die erste versteckte Schicht (für ein Netz mit zwei echten versteckten Schichten) her.

Thema: Verständnis Backpropagation

2 Backpropagation: Alternative Transferfunktion

(1 Punkt)

In der Vorlesung wurde(n) die Delta-Regel bzw. die Gewichtsupdates bei der Backpropagation für die Ausgabeschicht und den davor liegenden letzten Hidden-Layer unter Verwendung der Transferfunktion $g(a) = 1/(1 + \exp(-a))$ hergeleitet.

Leiten Sie die Gewichtsupdates für die letzte und die vorletzte Schicht unter Verwendung der alternativen Transferfunktion

$$g(a) = \tanh(a) = \frac{\exp(a) - \exp(-a)}{\exp(a) + \exp(-a)}$$

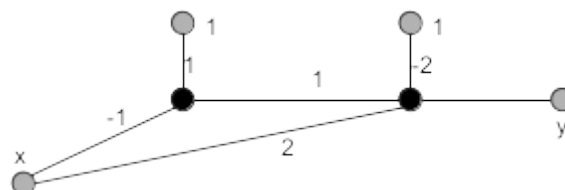
her.

Thema: Verständnis Backpropagation

3 MLP Feedforward und Backpropagation

(1 Punkt)

Betrachten Sie das folgende MLP, welches aus zwei Neuronen besteht. Die Gewichte sind an den Kanten angegeben. Das Netz erhält den skalaren Input x und berechnet daraus die Ausgabe y . Beide Neurone verwenden die Transferfunktion $g(a) = 1/(1 + \exp(-a))$.



- Berechnen Sie die Ausgabe y für den Input $(x, y_t) = (0, 0.5)$. Wie groß ist der Fehler?
- Berechnen Sie die partiellen Ableitungen für die Gewichte. Wie lauten die Gewichtsupdates für das obige Trainingsbeispiel? Setzen Sie $\eta = 0.01$.

Thema: Verständnis MLP und Backpropagation

4 MLP und Backpropagation

(4 Punkte)

Implementieren Sie ein Feedforward MLP mit mindestens einer versteckten Schicht. Nutzen Sie die in der Vorlesung besprochene quadratische Fehlerfunktion und die Transferfunktion $g(a) = 1/(1 + \exp(-a))$.

Implementieren Sie das Backpropagations-Verfahren zum Verändern der Gewichte. Achten Sie insbesondere darauf, die bereits berechneten δ_k der jeweils hinteren Schicht wieder zu verwenden (und nicht jeweils erneut zu berechnen!), d.h. propagieren Sie die Fehler von hinten nach vorn durch das Netz.

Trainieren Sie das Netz für den Iris-Datensatz (`iris.csv`) aus dem AIMA-Repository¹ und nutzen Sie dabei die Variante des stochastischen Gradientenabstiegs. Messen Sie pro Epoche (also nach jedem Durchlauf durch den kompletten Datensatz) den Trainingsfehler. Zeichnen Sie den Trainingsfehler als Diagramm über den Epochen auf.

Falls der Trainingsfehler nach einigen tausend Epochen nicht gegen einen Wert nahe Null strebt, erweitern Sie Ihr Netz (beispielsweise eine versteckte Schicht mehr oder mehr Neurone in der schon existierenden versteckten Schicht, ...) und trainieren Sie es erneut. Nach wievielen Epochen ist der Trainingsfehler fast Null?

Hinweis: Fragen Sie sich, ob ein objektorientierter Ansatz hier wirklich hilfreich ist!

Thema: Verständnis MLP und Backpropagation, Gefühl für nötige Größe des Netzes

5 Crossvalidation

(2 Punkte)

Mangels Testdatensatz konnte bisher nur der Trainingsfehler gemessen werden. Implementieren Sie das in der VL besprochene Crossvalidierungs-Verfahren und wenden Sie es auf Ihr Netz an. Teilen Sie dabei die Daten in 10 gleich große Teilmengen und nutzen Sie jeweils 9 Teile als Trainingsdaten und 1 Teil als Testdaten (in wechselnden Kombinationen).

Messen Sie analog zur vorigen Aufgabe in jeder Epoche Trainings- und Generalisierungsfehler und zeichnen Sie diese in ein Diagramm (d.h. Sie erhalten letztlich 10 leicht verschiedene Diagramme).

Nach wievielen Schritten wird nun jeweils der Wendepunkt beim *Generalisierungsfehler* erreicht? Was ergibt sich daraus für die Anzahl der Epochen zum abschließenden Training des Netzes auf dem gesamten Datensatz?

Trainieren Sie mit der ermittelten Zahl der Epochen das Netz für den gesamten Datensatz. Wie groß ist der Trainingsfehler nun?

Thema: Umgang mit Crossvalidation

6 Weight Decay

(1 Punkt)

Erweitern Sie Ihre Implementierung um das Verfahren *Weight Decay*.

Trainieren Sie das Netz mit der in der vorigen Aufgabe ermittelten Zahl von Epochen für den gesamten Datensatz. Vergleichen Sie die Gewichte und den Trainingsfehler mit den entsprechenden Resultaten der vorigen Aufgabe.

Thema: Verfahren zum Verbessern von MLPs

7 MLP und Training

(1 Punkt)

Sie haben ein MLP mit Hilfe eines Trainingsdatensatzes trainiert. Dabei haben Sie den Trainingsfehler gemessen, welcher mit jeder Iteration (Trainingsepoche) kleiner wurde. Die Testergebnisse für einen unabhängigen Datensatz sind allerdings bedeutend schlechter als die gemessenen Trainingsfehler.

Welche der folgenden Maßnahmen würden Sie vorschlagen, um die Leistung des MLP zu verbessern? Begründen Sie Ihre Antwort!

- a) Trainingsdatensatz vergrößern
- b) Anzahl der Trainingsepochen vergrößern
- c) Anzahl der Trainingsepochen verkleinern
- d) Anzahl der Hidden-Neurone vergrößern
- e) Anzahl der Hidden-Neurone verkleinern
- f) Weight-Decay-Parameter vergrößern

Thema: Trainingsfehler vs. Generalisierungsfehler, Maßnahmen

¹github.com/aimacode/aima-data

8 Gradientenabstieg und Momentum anwenden

(1 Punkt)

Führen Sie vier Iterationen des Gradientenabstieg mit Momentum durch, um das Minimum der Funktion $f(u) = \frac{1}{3}u^3 + 50u^2 - 100u - 30$ zu finden. Beginnen Sie mit $u = 20$. Die Lernrate sei $\eta = 0.01$ und der Momentumparameter $\alpha = 0.1$. Beim ersten Update sei $\Delta u^{(t-1)} = 0$.

Vergleichen Sie das Ergebnis mit einem Gradientenabstieg ohne Momentum.

Thema: Gradientenabstieg und Momentum mal anders :-)

Frohe Weihnachten und einen guten Start ins nächste Jahr!