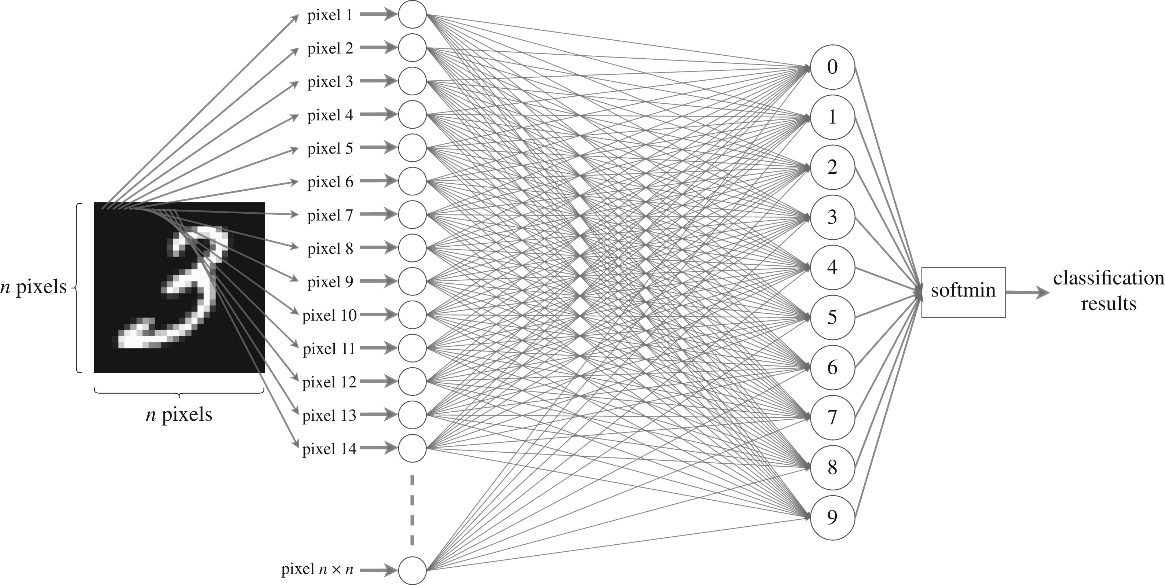
Praktikum Neuronale Netze in der Bildverarbeitung

Protokoll

Klassifizierung handgeschriebener Ziffern



Quelle: https://royalsocietypublishing.org/doi/10.1098/rsta.2019.0163

**Datum:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Matrikelnr.** | **Punkte Protokoll** |
| **Martin Blümel** | **4842445** | **/20** |

*Aufgabe 1: Training eines Neuronalen Netzes zur Ziffernklassifizierung*

Entnehmen Sie der Versuchsanleitung die zur Lösung der Aufgabe notwendigen Schritte 1 (Laden des Trainings- und Validierungsdatensatzes) und 2 (Erstellung einer geeigneten Netzarchitektur).  
Stellen Sie für beide Trainingsvorgänge (automatisierte und benutzerdefinierte Trainingsschleife) die Trainingskurven und in einem Diagramm dar.

Diagramm (5P):

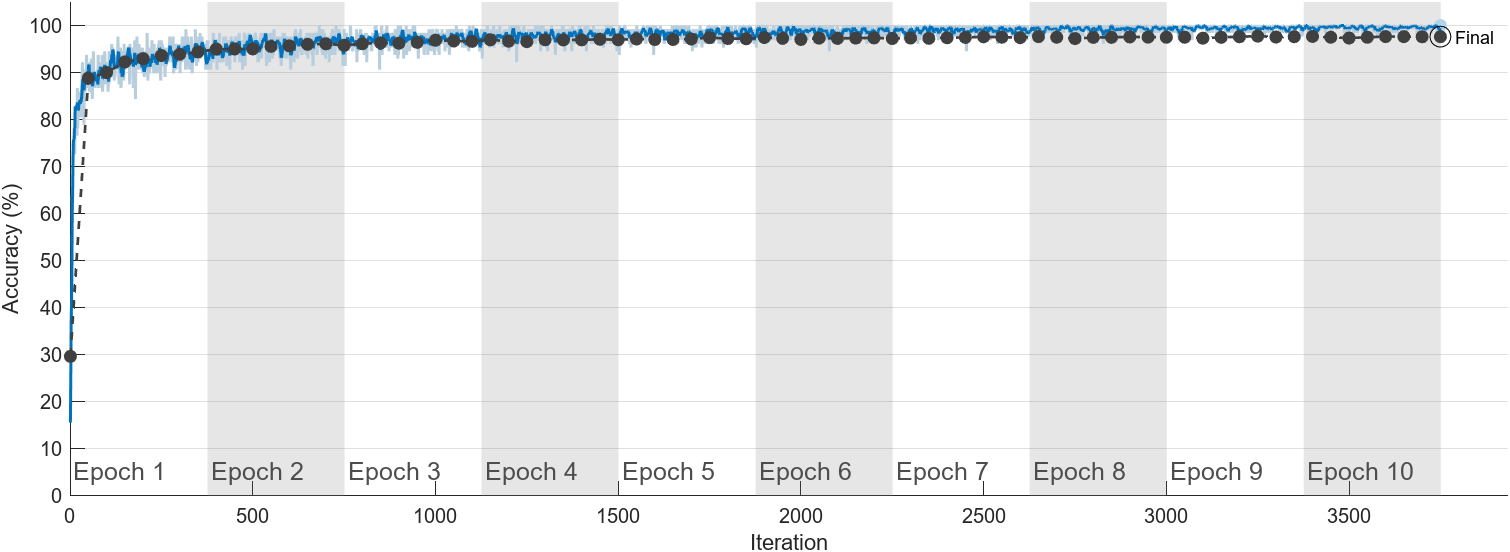


Diagramm 1: des Integrated-Trainings

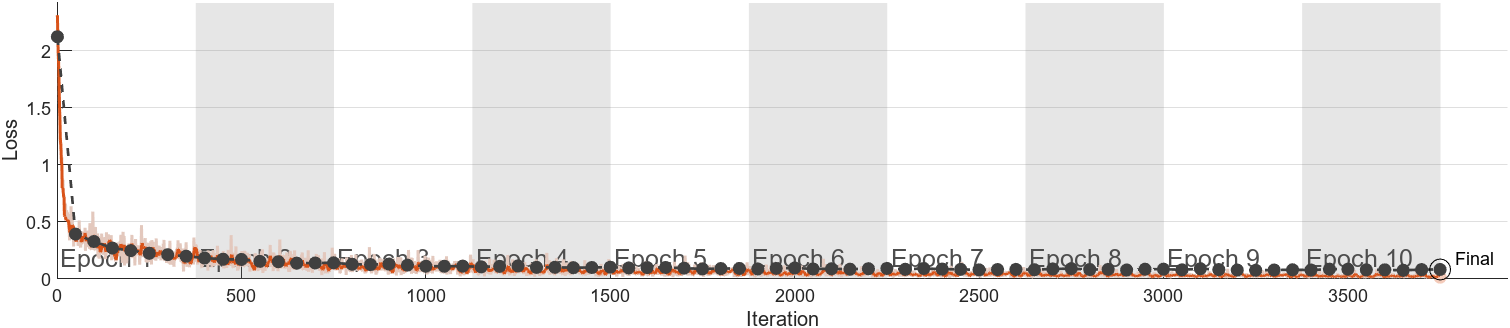


Diagramm 2: des Integrated-Trainings

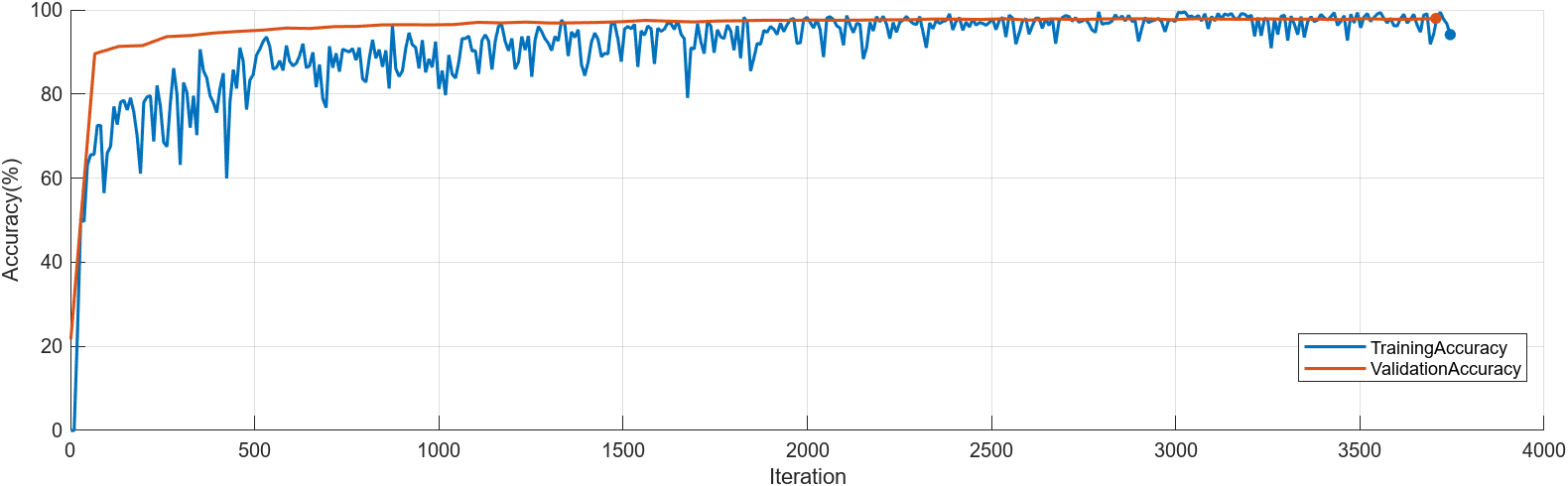


Diagramm 3: des Custom-Trainings

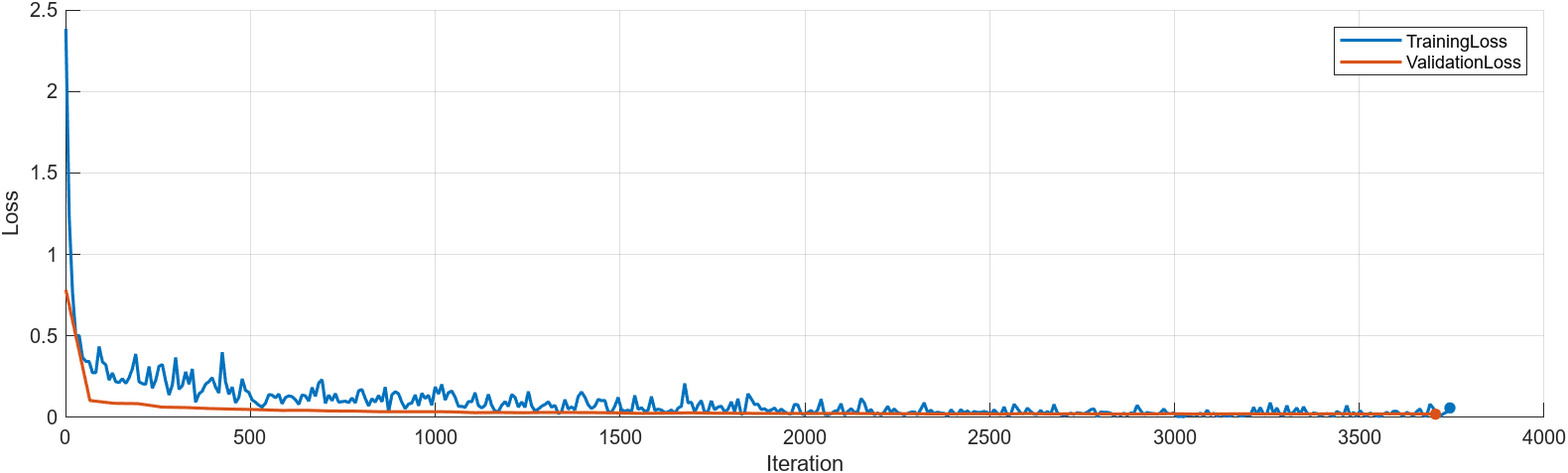


Diagramm 2: des Custom-Trainings

Beschreibung (1P):

Alle 4 Diagramme wurden mit dem in Matlab integrierten Prozessmointor erstellt.

Diagramm 1 und 2 zeigen die Trainingskurven für das automatisierte Training über 10 Epochen bzw. über 3750 Iterationen mit einer Lernrate von 0,001 und einer Minibatchgröße von 128.

Die Diagramme 3 und 4 zeigen die Trainingskurven für das benutzerdefinierte Training ebenso über 10 Epochen mit einer Lernrate von 0,001 und einer Minibatchgröße von 128. Der rote Graph zeigt jeweils die Validierungsdaten, der blaue die Testdaten, wobei die Validierungskurven alle 64 Iterationen aktualisiert werden und die Testdatenkurve alle 8 Iterationen.

Verwenden Sie für alle nachfolgenden Aufgaben die benutzerdefinierte Trainingsschleife!

*Aufgabe 2: Klassifizierungsgenauigkeit in Abhängigkeit der eingelesenen Ziffer*

Sortieren Sie hierfür die Testdaten nach den jeweiligen Ziffern und ermitteln Sie für jede Ziffer die mittlere Klassifizierungsgenauigkeit über den Mittelwert. Stellen Sie das Ergebnis in einem Diagramm dar. Welche Ziffer kann am besten erkannt werden? Markieren Sie diese im Diagramm!

Diagramm (3P):

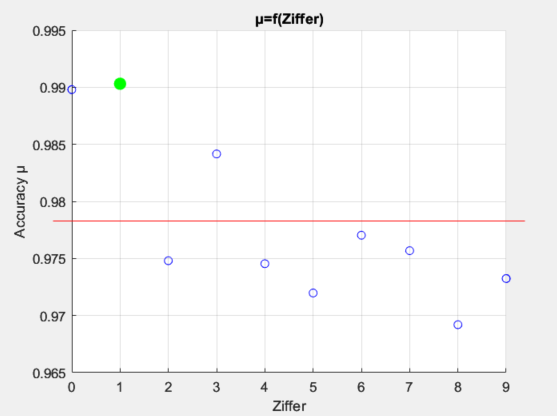


Diagramm 5: Accuracy , je Ziffer

Beschreibung (1P):

Im obigen Diagramm ist die Klassifizierungsgenauigkeit für jede Ziffer als blaue Kreise dargestellt (Custom-Training: Solver:Adam, MiniBatchsize=128, Epochenzahl=10, Lernrate=0,001). Die rote Linie kennzeichnet die Klassifizierungsgenauigkeit des gesamten Trainings, den Querschnitt der einzelnen Zifferngenauigkeit (hier ca. 97,83%). Der grüne Punkt markiert die Klassifizierungsgenauigkeit der am besten erkannten Ziffer hier die 1 (zu ca. 99,03%).

*Aufgabe 3: Klassifizierungsgenauigkeit in Abhängigkeit von Lernrate und Optimierungssalgorithmus*

Vergleichen Sie nun die beiden Optimierer *adam* und *sgdm* mit mindestens 6 verschiedenen *learning rates* zwischen 10−6 und 10−1. Stellen Sie die erreichte mittlere Klassifizierungsgenauigkeit in Abhängigkeit von der *learning rate* dar und verwenden Sie für die *learning rate* eine logarithmische Achse.

Diagramm (4P):

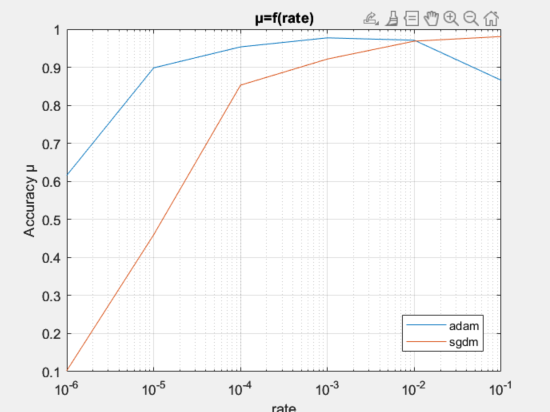


Diagramm 6: Accuracy , je Lernrate

Beschreibung (1P):

Diagramm 6 zeigt die Klassifizierungsgenauigkeit des Custom-Trainings (MiniBatchsize=128, Epochenzahl=10), bei verschiedenen Lernraten von 10−6 bis 10−1, über einer logarithmischen Achse. Der rote Graph zeigt hierbei den Verlauf für den Solver-Sgdm an und der blaue den Solver-Adam.

*Aufgabe 4: Klassifizierungsgenauigkeit und Trainingszeit in Abhängigkeit der Größe des Mini-Batch*

Testen Sie mindestens 5 verschiedene Größen des Mini-Batch zwischen 16 und 256. Nutzen Sie den solver *adam*. Wählen Sie aus der vorherigen Aufgabe eine *learning rate* aus, bei der der Trainingsprozess zu einer hohen Klassifizierungsgenauigkeit konvergiert. Stellen Sie die mittlere Klassifizierungsgenauigkeit und die benötigte Trainingszeit in Abhängigkeit von der Größe des Mini-Batches dar*:*  und .

Diagramme (3P):

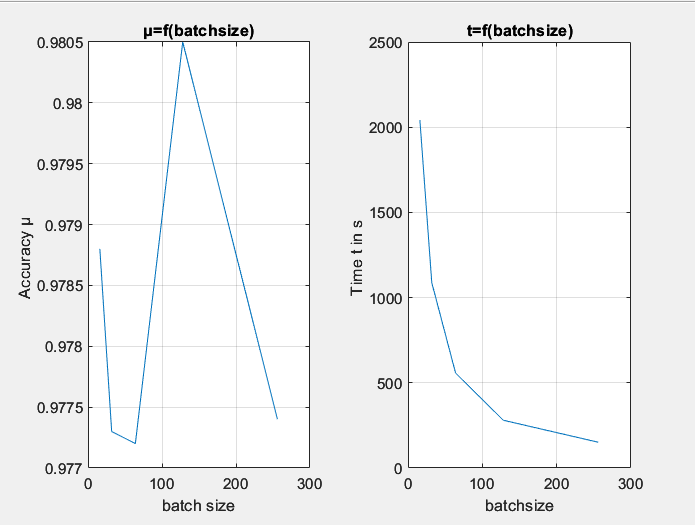


Diagramm 7: Accuracy , je Batchsize  
Diagramm 8: Time , je Batchsize

Beschreibung (1P):

Im linken Diagramm 7 ist die Klassifizierungsgenauigkeit des Custom-Trainings bei 5 verschiedenen MiniBatchsizes (ansteigend) von 16 bis 265 (16, 32, 64, 128, 256) dargestellt.

Im rechten Diagramm 8 ist die benötigte Trainingszeit des Custom-Trainings bei 5 verschiedenen MiniBatchsizes (ansteigend) von 16 bis 265 (16, 32, 64, 128, 256) dargestellt.

Das Training erfolge mit der in Aufgabe 3, für den Adam-Solver besten, bestimmten Lernrate: 0,001 und jeweils über 10 Epochen.

*Aufgabe 5: Diskussion (1P)*

Leiten Sie aus den Trainingsergebnissen Zusammenhänge zwischen den untersuchten Hyperparametern und der Trainingszeit, sowie der erreichten Klassifizierungsgenauigkeit ab. Formulieren Sie hierfür eine kurze Diskussion (Stichpunkte erlaubt):

In Aufgabe 3 (*Klassifizierungsgenauigkeit in Abhängigkeit der Lernrate*) sieht man das der Optimierer Sgdm mit kleiner werdender Lernrate immer mehr an Genauigkeit einbüßt. Bei einer Lernrate von 0,1 ist seine Klassifizierungsgenauigkeit maximal, bei einer Lernrate von 10−6 fällt sie auf nahe zu 10%. Bei Sgdm erweist sich eine größerer Lernrate als sinnvoll, andernfalls konvergiert das Training nicht rechtzeitig. Alternativ könnte man auch die Epochenzahl erhöhen damit das Training noch konvergiert.   
Der Optimierer Adam erreicht die maximale Klassifizierungsgenauigkeit dagegen erst bei einer Lernrate von 0,001. Bei kleineren Lernraten fällt die Genauigkeit auch ab auf ca 60% bei einer Lernrate von 10−6. Bei Adam sollte die Lernrate demzufolge weder zu klein noch zu groß sein damit das Training innerhalb der Epochenzahl zu einem gewünschten Maximum konvergiert.

In Aufgabe 4 (*Klassifizierungsgenauigkeit in Abhängigkeit der MiniBatchsize, Trainingszeit in Abhängigkeit der MiniBatchsize*) sieht man anhand von Diagramm 8 das mit zunehmender MiniBatchgröße eine geringe Trainingszeit erreicht werden kann.   
Die Klassifizierungsgenauigkeit erreicht ihr Maximum bei einer MiniBatchgröße von 128, mittig im untersuchten Intervall. Das Training divergiert bei konstanter Lernrate für zu kleine oder zu große MiniBatchgrößen.