Aalto-yliopisto
Perustieteiden
korkeakoulu

Gradient based optimization methods and learning rates in Feedforward Neural Networks

Julius Lind

04.12.2019

Advisor: Prof. *Harri Ehtamo*

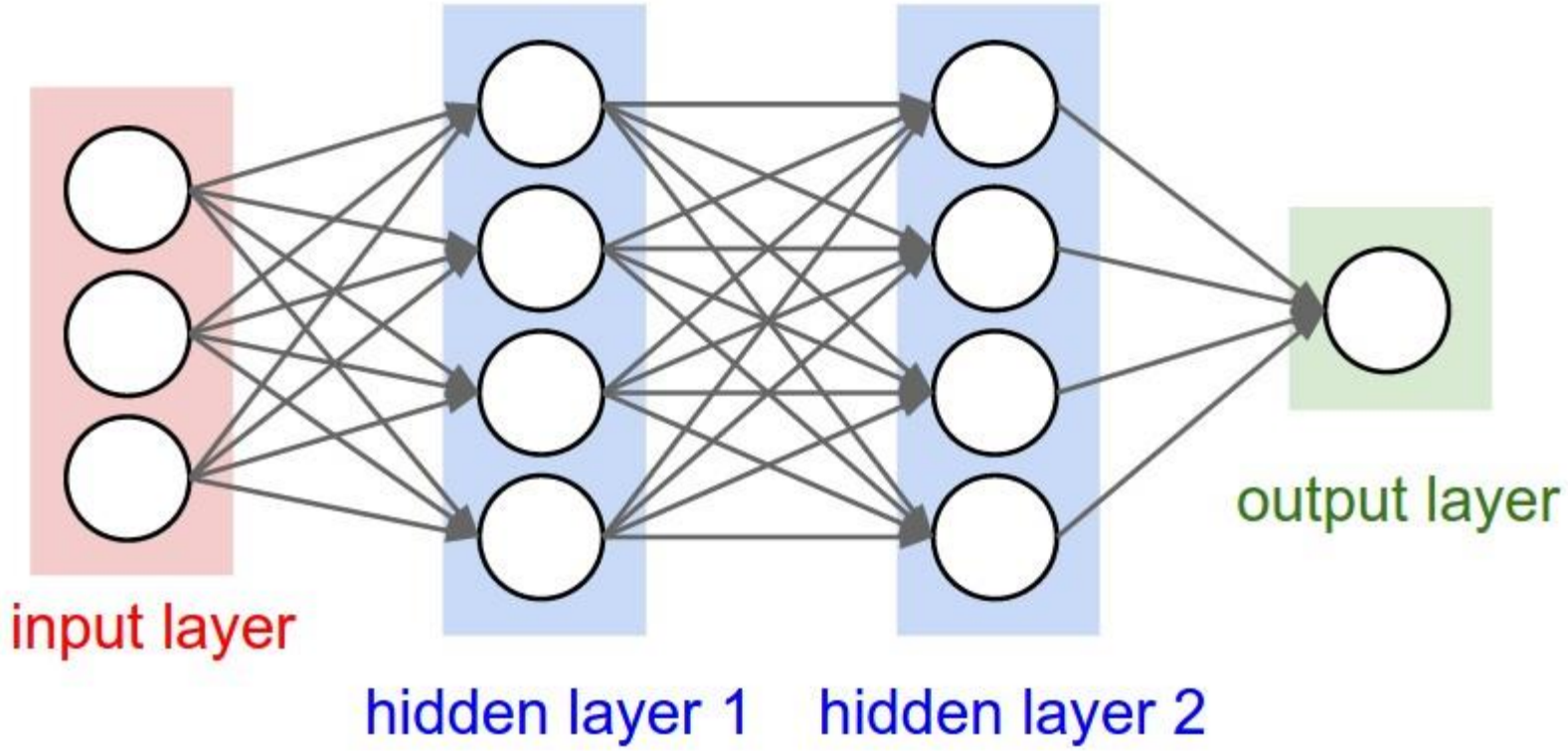
Supervisor: Prof. *Harri Ehtamo*

Systeemianalyysin
laboratorio

Tausta

- Eteenpäinsyöttävät neuroverkot ovat neuroverkkojen alaluokka
- Neuroverkkoja käytetään:
  - Kuvien luokitteluun
  - Konekääntäminen
  - Regressioon
- Eteenpäinsyöttävät neuroverkot koostuvat syötekerroksista, ulostulokerroksista ja näiden välissä on piilotettuja kerroksia
- Syötedata: (X, y)
- Jokainen kerros muuntaa syötteen ja tätä käytetään seuraavan kerroksen syötteenä

$$z = \sigma(AX + b)$$



Lähde: Stanford course CS231n Convolutional Neural Networks for Visual Recognition

Tausta

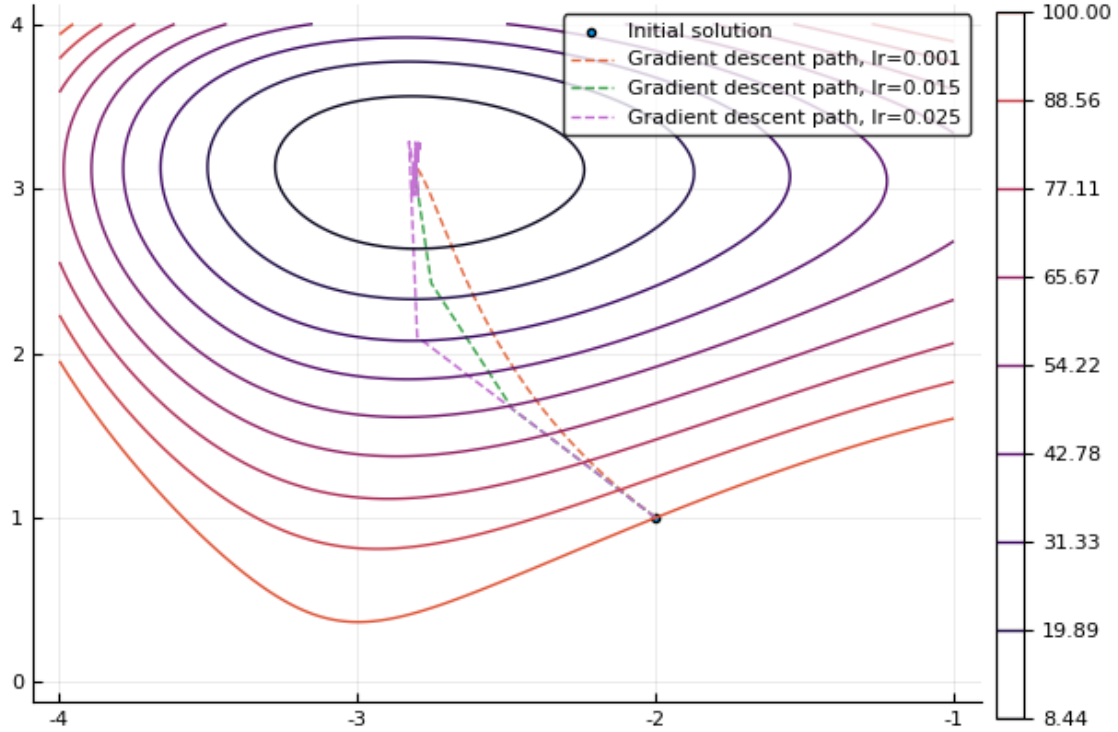
- Verkon ulostuloa $\bar{y}$ verrataan y odotettuun ulostuloon y
  - Logistinen sakkofunktio= $-\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{k} y_{i,j}\log(\bar{y}_{i,j})$
- Optimointitehtävä
  - Minimoi sakkofunktio
- Vastavirta-algoritmi
  - Käytä ketjusääntöä sakkofunktion gradienttien laskemiseen
- Sakkofunktion voi minimoida gradienttimenetelmillä

Tavoitteet

- Tutkia eri gradienttimetodien ominaisuuksia neuroverkkojen kouluttamiseen
- Metodit:
  - Gradienttimetodi
  - Stokastinen gradienttimetodi
  - Momentum/Nesterov accerelated gradient (NAG)
  - RPROP/RMSPROP
  - Adam
- Kouluttaa neuroverkko luokittelemaan kuvia ja verrata metodien tehokkuutta:
  - Sakkofunktion arvo
  - Luokittelutarkkuus
  - Kesto

Menetelmät

- Gradienttimenetelmä: $w_i = w_{i-1} - \lambda \nabla_w J(y, f(x, w_{i-1}))$.
  - Askelkoolla suuri vaikutus vaadittujen iteraatioiden määrään



Menetelmät

- Momentum:

$$m_i = \beta m_{i-1} + \lambda \nabla_w J(y, f(x, w_{i-1}))$$
$$w_i = w_{i-1} - m_i$$

  - Käytetään aikaisempia gradientteja määrittämään seuraavan askeleen suunta
  - Askelkoko kasvaa jos peräkkäiset askeleet ovat samaan suuntaan
  - Askelkoko pienenee jos suunta vaihtuu iteraatioiden välissä

- NAG:

$$m_i = \beta m_{i-1} + \lambda \nabla_w J(y, f(x, w_{i-1} - \beta m_{i-1}))$$
$$w_i = w_{i-1} - m_i$$

  - Käytetään aikaisempia gradientteja kuten momentumissa
  - Gradientti arvioidaan pisteessä johon siirryttäisiin edellisten gradienttien perusteella

Menetelmät

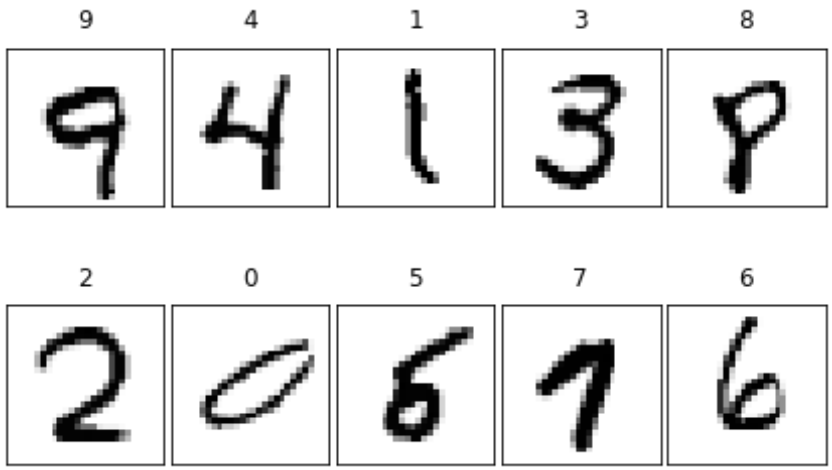
- Adaptiiviset menetelmät (RPROP, RMSPROP, ADAM)
  - Erilliset askelkoot jokaiselle parametrille
  - Askelkokoja muutetaan aikasempien gradienttien/gradienttien neliöiden perusteella
- RPROP:
  - Askelkokoja kasvatetaan jos edellisen gradientin merkki oli sama
  - Jos merkki vaihtuu askelkokoa pienennetään
- RMSPROP/ADAM:
  - Askelkoot skaalataan aikaisempien gradienttien neliön magnitudilla
  - Käytetään aikaisempia gradientteja kuten momentumissa (ADAM)

Menetelmät

- Stokastinen gradienttimenetelmä
  - Sakkofunktio ja gradientit lasketaan vain yhden datapisteen perusteella
  - Iteraatiot nopeutuvat
  - Gradienteissa on korkea varianssi
- Osajoukkoalgoritmi (Mini-Batch Gradient Descent)
  - Sakkofunktio ja gradientit lasketaan datan osajoukolla
  - Iteraatiot nopeutuvat
  - Pienempi varianssi kuin stokastisella gradienttimenetelmällä

Työkalut ja data

- PyTorch ja Scikit-learn
  - Avoimen lähdekoodin koneoppimispaketteja pythonille
- MNIST
  - Kuvanluokitusdatasetti joka sisältää 70000 28x28 kuvaa käsinkirjoitetuista 0-9 numeroista

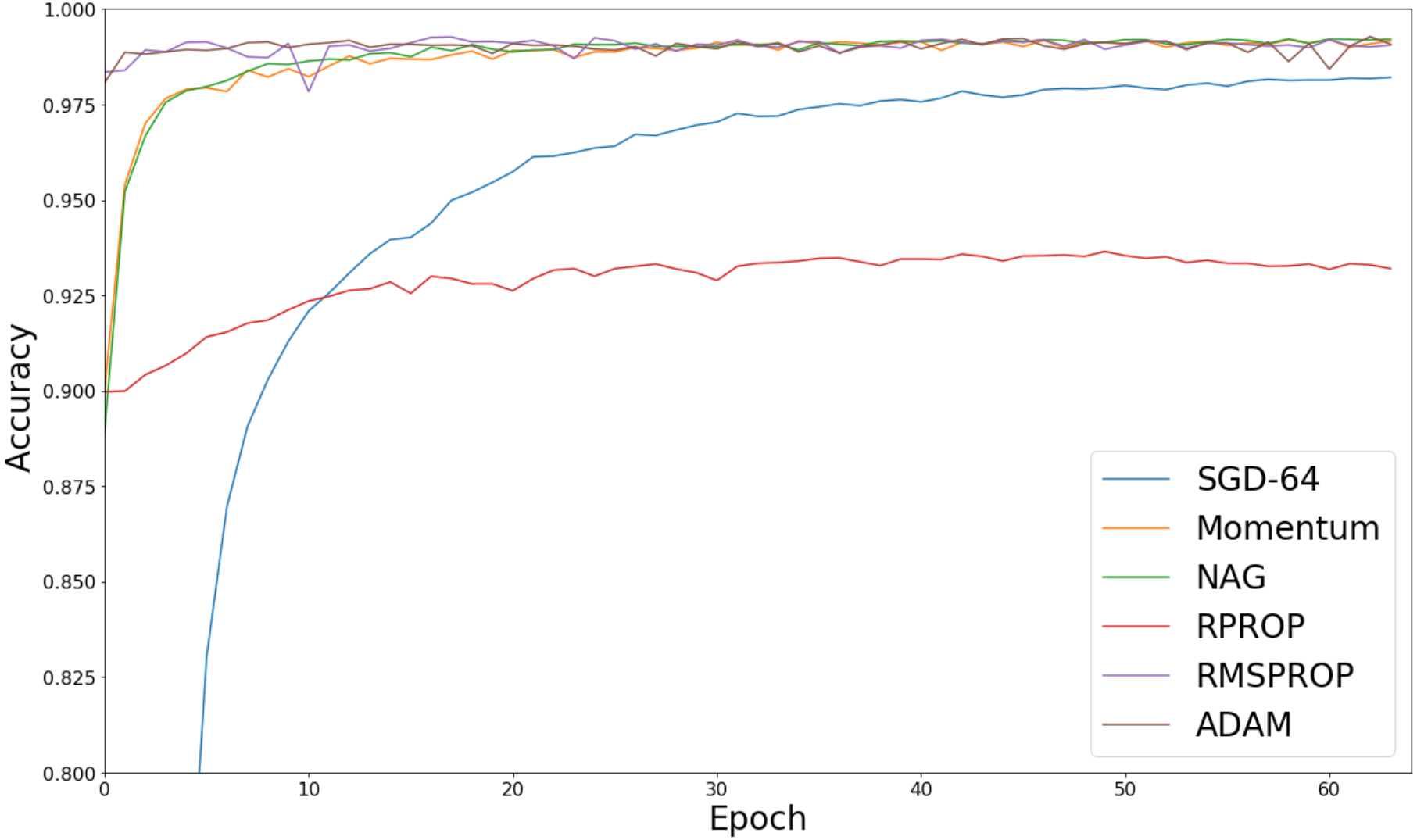


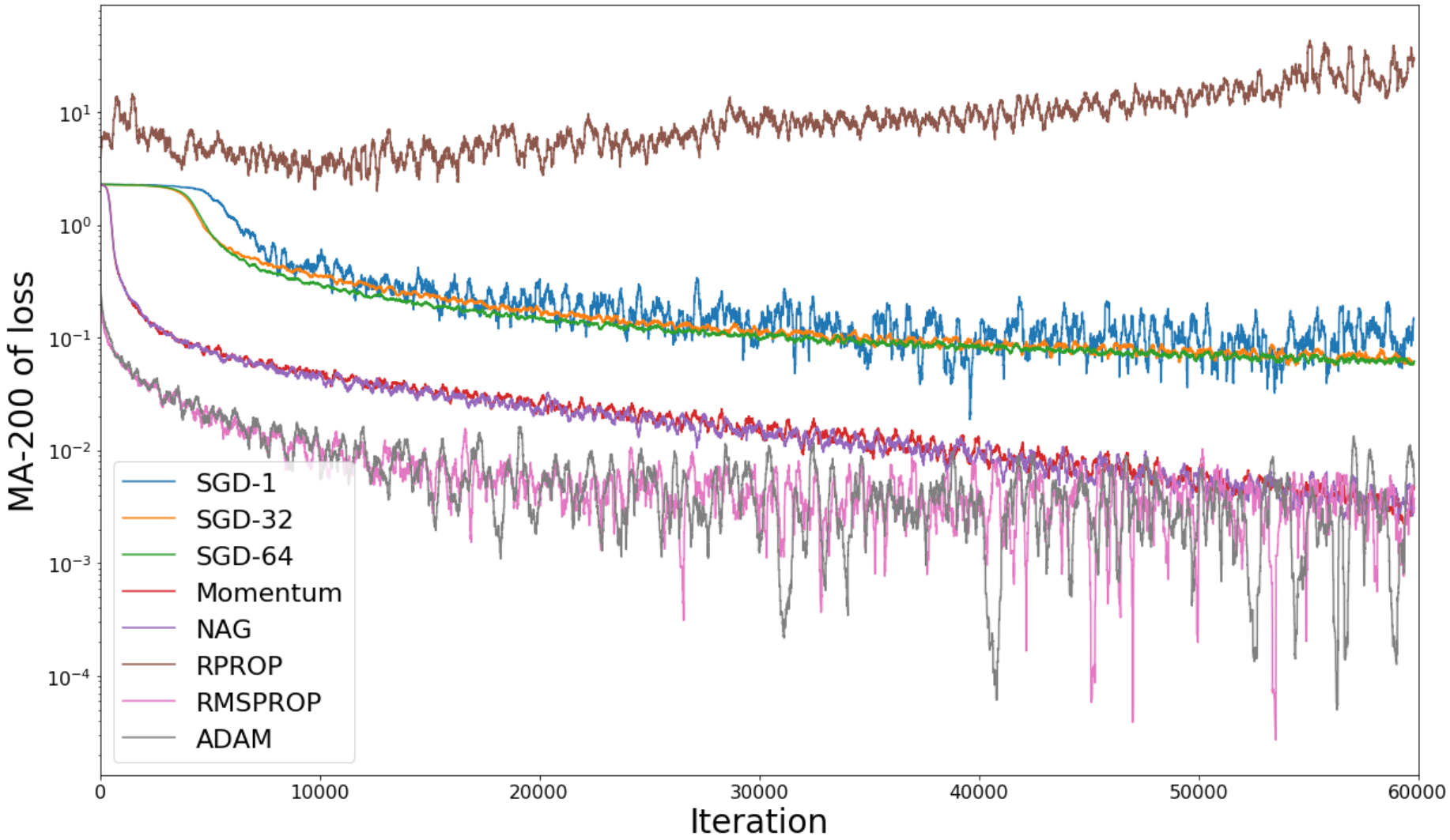
Menetelmät

- Koulutetaan LeNet-5 neuroverkko luokittelemaan kuvia
  - 60000 Kuvaa käytetään koulutukseen
- Eräkoot 1,32,64 stokastisessa gradienttimenetelmässä
- Eräkoot 64 muissa menetelmissä
- $\lambda = 10^{-6}$
- Jokaisella menetelmällä 60000 iteraatiota
- Luokitellaan jäljelle jäävät 10000 kuvaa koulutetulla neuroverkolla ja lasketaan luokittelutarkkuus

Tulokset

Menetelmä	Sakkofunktion arvo	Luokittelutarkkuus (%)	Aika (s)
SGD-1	0.15	97.8	85s
SGD-32	0.06	98.08	479s
SGD-64	0.06	98.21	876s
Momentum	0.005	99.17	863s
NAG	0.003	99.22	858s
RPROP	29.8	93.2	1084s
RMSPROP	0.004	99.06	860s
ADAM	0.05	99.08	865s


1.000
0.975
0.950
0.925
0.900
0.875
0.850
0.825
0.800
Accuracy
0
10
20
30
40
50
60
Epoch
SGD-64
Momentum
NAG
RPROP
RMSPROP
ADAM


MA-200 of loss
Iteration
SGD-1
SGD-32
SGD-64
Momentum
NAG
RPROP
RMSPROP
ADAM
10^{1}
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
0
10000
20000
30000
40000
50000
60000

Johtopäätökset

- Eräkoolla suurin vaikutus koulutusaikaan
- Sakkofunktion arvolla ei suoraa yhteyttä luokittelutarkkuuteen
- Gradienttimenetelmä jää helpommin lokaaleihin minimeihin
- ADAM:in ja RMSPROP:in parametrikohtaiset askelkoot nopeuttavat koulutusta alkuvaiheissa
- Metodien yhdistämisellä voisi saavuttaa parempia tuloksia