October 17, 2017

The results below are generated from an R script.

```
# (c) 2015 by Guenter Klambauer
library(gmatrix)
setDevice(0)

## Now using device 0 - "GeForce 830M"

## Simulated data example
y <- c(rep(0,49),rep(1,51))
X <- matrix(NA,nrow=100,ncol=20)
for (i in 1:100){
   X[i, ] <- rnorm(n=20,sd=0.2)
}
X[,5] <- X[,5]+y
X[,7] <- X[,7]+y


source("nnet-gpu.R")
nnm <- nnetgpu(X,y,hiddenUnits=c(50,20),epochs=5,fineTuneEpochs=0,verbose=0,eta=0.2,batch=16,activationF

## ##------ Tue Oct 17 12:45:58 2017 ------##

## Using given strategy as initialization.

##
## #######
## Epoch: 1 Batch: 1
##  TRAIN:
##  Training loss: 0.6057969
## 1:0.606
##
## #######
## Epoch: 1 Batch: 2
##  TRAIN:
##  Training loss: 0.5759727
## 1:0.591
##
## #######
## Epoch: 1 Batch: 3
##  TRAIN:
##  Training loss: 0.6036865
## 1:0.595
##
## #######
## Epoch: 1 Batch: 4
```

```
##  TRAIN:
##  Training loss: 0.5324479
## 1:0.579
##
## #######
## Epoch: 1 Batch: 5
##  TRAIN:
##  Training loss: 0.5994346
## 1:0.583
##
## #######
## Epoch: 1 Batch: 6
##  TRAIN:
##  Training loss: 0.4820223
## 1:0.567
##
## #######
## Epoch: 1 Batch: 7
##  TRAIN:
##  Training loss: 0.5116873
## 1:0.559
## 1469 MB free out of 2002 MB total.
##
## #######
## Epoch: 2 Batch: 1
##  TRAIN:
##  Training loss: 0.5176625
## 1:0.518
##
## #######
## Epoch: 2 Batch: 2
##  TRAIN:
##  Training loss: 0.4688701
## 1:0.493
##
## #######
## Epoch: 2 Batch: 3
##  TRAIN:
##  Training loss: 0.4269529
## 1:0.471
##
## #######
## Epoch: 2 Batch: 4
##  TRAIN:
##  Training loss: 0.5331997
## 1:0.487
##
## #######
## Epoch: 2 Batch: 5
##  TRAIN:
##  Training loss: 0.3504838
## 1:0.459
##
## #######
```

```
## Epoch: 2 Batch: 6
##  TRAIN:
##  Training loss: 0.5141415
## 1:0.469
##
## #######
## Epoch: 2 Batch: 7
##  TRAIN:
##  Training loss: 0.3981275
## 1:0.458
## 1469 MB free out of 2002 MB total.
##
## #######
## Epoch: 3 Batch: 1
##  TRAIN:
##  Training loss: 0.5209044
## 1:0.521
##
## #######
## Epoch: 3 Batch: 2
##  TRAIN:
##  Training loss: 0.3556548
## 1:0.438
##
## #######
## Epoch: 3 Batch: 3
##  TRAIN:
##  Training loss: 0.374346
## 1:0.417
##
## #######
## Epoch: 3 Batch: 4
##  TRAIN:
##  Training loss: 0.4706955
## 1:0.43
##
## #######
## Epoch: 3 Batch: 5
##  TRAIN:
##  Training loss: 0.2527544
## 1:0.395
##
## #######
## Epoch: 3 Batch: 6
##  TRAIN:
##  Training loss: 0.4016292
## 1:0.396
##
## #######
## Epoch: 3 Batch: 7
##  TRAIN:
##  Training loss: 0.3747584
## 1:0.393
## 1469 MB free out of 2002 MB total.
```

```
##
## #######
## Epoch: 4 Batch: 1
##  TRAIN:
##  Training loss: 0.2442353
## 1:0.244
##
## #######
## Epoch: 4 Batch: 2
##  TRAIN:
##  Training loss: 0.3660507
## 1:0.305
##
## #######
## Epoch: 4 Batch: 3
##  TRAIN:
##  Training loss: 0.3163466
## 1:0.309
##
## #######
## Epoch: 4 Batch: 4
##  TRAIN:
##  Training loss: 0.3330096
## 1:0.315
##
## #######
## Epoch: 4 Batch: 5
##  TRAIN:
##  Training loss: 0.3323824
## 1:0.318
##
## #######
## Epoch: 4 Batch: 6
##  TRAIN:
##  Training loss: 0.4157573
## 1:0.335
##
## #######
## Epoch: 4 Batch: 7
##  TRAIN:
##  Training loss: 0.3918594
## 1:0.343
## 1469 MB free out of 2002 MB total.
##
## #######
## Epoch: 5 Batch: 1
##  TRAIN:
##  Training loss: 0.2523372
## 1:0.252
##
## #######
## Epoch: 5 Batch: 2
##  TRAIN:
##  Training loss: 0.3103623
```

```
## 1:0.281
##
## #######
## Epoch: 5 Batch: 3
##  TRAIN:
##  Training loss: 0.2748753
## 1:0.279
##
## #######
## Epoch: 5 Batch: 4
##  TRAIN:
##  Training loss: 0.2717896
## 1:0.277
##
## #######
## Epoch: 5 Batch: 5
##  TRAIN:
##  Training loss: 0.2827696
## 1:0.278
##
## #######
## Epoch: 5 Batch: 6
##  TRAIN:
##  Training loss: 0.299228
## 1:0.282
##
## #######
## Epoch: 5 Batch: 7
##  TRAIN:
##  Training loss: 0.2815584
## 1:0.282
## 1469 MB free out of 2002 MB total.

# check training
table(y,predictNNgpu(X,nnm,useBestWeights=TRUE)>0.5)

##
## y   FALSE TRUE
##   0    43    6
##   1     0   51

## IRIS data example
Y <- cbind(as.numeric(rep("setosa",nrow(iris))==iris$Species), as.numeric(rep("versicolor",nrow(iris))==
X <- as.matrix(iris[,1:4])
X <- t(t(X) - apply(X,2,mean) / apply(X,2,sd))

nnm <- nnetgpu(X,Y,hiddenUnits=c(100,100),epochs=5,fineTuneEpochs=0,verbose=0,eta=0.01,batch=64,activati

## ##------ Tue Oct 17 12:46:00 2017 ------##

## Using given strategy as initialization.

##
## #######
## Epoch: 1 Batch: 1
```

```
##  TRAIN:
##  Training loss: 1.035597
## 1:2.72 2:0.284 3:0.103
##
## #######
## Epoch: 1 Batch: 2
##  TRAIN:
##  Training loss: 0.5141431
## 1:1.641 2:0.324 3:0.36
##
## #######
## Epoch: 1 Batch: 3
##  TRAIN:
##  Training loss: 0.5325892
## 1:1.299 2:0.535 3:0.248
## 1469 MB free out of 2002 MB total.
##
## #######
## Epoch: 2 Batch: 1
##  TRAIN:
##  Training loss: 0.5339503
## 1:0.378 2:0.059 3:1.165
##
## #######
## Epoch: 2 Batch: 2
##  TRAIN:
##  Training loss: 0.3568857
## 1:0.214 2:0.505 3:0.617
##
## #######
## Epoch: 2 Batch: 3
##  TRAIN:
##  Training loss: 0.2080786
## 1:0.199 2:0.381 3:0.519
## 1469 MB free out of 2002 MB total.
##
## #######
## Epoch: 3 Batch: 1
##  TRAIN:
##  Training loss: 0.2353439
## 1:0.06 2:0.615 3:0.031
##
## #######
## Epoch: 3 Batch: 2
##  TRAIN:
##  Training loss: 0.1791621
## 1:0.099 2:0.353 3:0.171
##
## #######
## Epoch: 3 Batch: 3
##  TRAIN:
##  Training loss: 0.1447316
## 1:0.076 2:0.352 3:0.131
## 1469 MB free out of 2002 MB total.
```

```
##
## #######
## Epoch: 4 Batch: 1
##  TRAIN:
##  Training loss: 0.1211346
## 1:0.064 2:0.149 3:0.15
##
## #######
## Epoch: 4 Batch: 2
##  TRAIN:
##  Training loss: 0.1218113
## 1:0.064 2:0.142 3:0.158
##
## #######
## Epoch: 4 Batch: 3
##  TRAIN:
##  Training loss: 0.1512817
## 1:0.053 2:0.21 3:0.132
## 1469 MB free out of 2002 MB total.
##
## #######
## Epoch: 5 Batch: 1
##  TRAIN:
##  Training loss: 0.1664262
## 1:0.05 2:0.058 3:0.391
##
## #######
## Epoch: 5 Batch: 2
##  TRAIN:
##  Training loss: 0.2240706
## 1:0.039 2:0.345 3:0.201
##
## #######
## Epoch: 5 Batch: 3
##  TRAIN:
##  Training loss: 0.2087686
## 1:0.067 2:0.237 3:0.295
## 1469 MB free out of 2002 MB total.

# check training error
table(apply(Y,1,which.max),apply(predictNNgpu(X,nnm,useBestWeights=TRUE),1,which.max))

##
##      1  2  3
##   1 50  0  0
##   2  0 50  0
##   3  0 35 15
```

The R session information (including the OS info, R version and all packages used):

```
sessionInfo()

## R version 3.3.1 (2016-06-21)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 16.04.2 LTS
```

```
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C               LC_TIME=de_AT.UTF-8
##  [4] LC_COLLATE=en_US.UTF-8     LC_MONETARY=de_AT.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=de_AT.UTF-8       LC_NAME=C                  LC_ADDRESS=C
## [10] LC_TELEPHONE=C             LC_MEASUREMENT=de_AT.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] gmatrix_0.3 knitr_1.15
##
## loaded via a namespace (and not attached):
## [1] magrittr_1.5  tools_3.3.1   stringi_1.1.2 highr_0.6     stringr_1.2.0 evaluate_0.10
```

```
Sys.time()
```

```
## [1] "2017-10-17 12:46:02 CEST"
```