1.4.2 誤差逆伝搬

手書き文字(digit)認識をします。データは下記からダウンロードしてください。
http://www.kaggle.com/c/digit-recognizer (http://www.kaggle.com/c/digit-recognizer)

```
setwd("C:/Users/k-harada/Desktop/kdd/digit/ORG")
library(data.table)
```

```
## Warning: package ‘data.table’ was built under R version 3.1.3
```

```
library(dplyr)
```

```
## Warning: package ‘dplyr’ was built under R version 3.1.3
```

```
## 
## Attaching package: ‘dplyr’
```

```
## The following objects are masked from ‘package:data.table’:
## 
##     between, last
```

```
## The following objects are masked from ‘package:stats’:
## 
##     filter, lag
```

```
## The following objects are masked from ‘package:base’:
## 
##     intersect, setdiff, setequal, union
```

```
traindata <- fread("train.csv")
```

```
## 
Read 47.6% of 42000 rows
Read 95.2% of 42000 rows
Read 42000 rows and 785 (of 785) columns from 0.072 GB file in 00:00:04
```

```
testdata <- fread("test.csv")

train_label <- traindata$label[1:28000]
valid_label <- traindata$label[28001:42000]

# do not resize here
# /255 so that range in 0-1
train_mat <- as.matrix(traindata[1:28000, ])[, -1]/255
valid_mat <- as.matrix(traindata[28001:42000, ])[,-1]/255
```

サンプルのニューラルネット

中間層1層のニューラルネットを作成します

```
# 答えの用意
answer_mat <- matrix(0, nrow = 28000, ncol = 10)
for (i in seq(10)) {
    answer_mat[train_label == (i - 1), i] <- 1
}

# learn weights
set.seed(0)
W1 <- matrix(rnorm(28*28*50), ncol=28*28)
intercept1 <- rep(-0.5, length = 50)

W2 <- matrix(0, nrow = 10, ncol = 50)
intercept2 <- rep(0, length = 10)

# learn rate
eta1 <- 0.01
eta2 <- 0.01
for (loop in seq(5)){
  for (i in seq(28000)) {
      # feed forward
      output1 <- 1/(1 + exp(-1 * (W1 %*% train_mat[i, ] + intercept1)))
      output2 <- 1/(1 + exp(-1 * (W2 %*% output1 + intercept2)))
      # back propagation
      W2 <- W2 + eta2 * (answer_mat[i, ] - output2) %*% t(output1)
      intercept2 <- intercept2 + eta2 * (answer_mat[i, ] - output2)
      W1 <- W1 + eta1 * ((output1 * (1 - output1)) * (t(W2) %*% (answer_mat[i, ] - output2))) %
*% t(train_mat[i, ])
      intercept1 <- intercept1 + eta1 * (output1 * (1 - output1)) * (t(W2) %*% (answer_mat[i, ]
- output2))
  }
}
```

学習結果の確認

```
output_mat1 <- 1/(1 + exp(-1 * (train_mat %*% t(W1) + matrix(1, nrow = 28000, ncol = 1) %*% matrix(intercept1, ncol = 50))))
output_mat2 <- 1/(1 + exp(-1 * (output_mat1 %*% t(W2) + matrix(1, nrow = 28000, ncol = 1) %*% matrix(intercept2, ncol = 10))))

output_mat1_v <- 1/(1 + exp(-1 * (valid_mat %*% t(W1) + matrix(1, nrow = 14000, ncol = 1) %*% matrix(intercept1, ncol = 50))))
output_mat2_v <- 1/(1 + exp(-1 * (output_mat1_v %*% t(W2) + matrix(1, nrow = 14000, ncol = 1) %*% matrix(intercept2, ncol = 10))))

# 出力
trainres <- max.col(as.matrix(output_mat2)) - 1
validres <- max.col(as.matrix(output_mat2_v)) - 1

table(trainres, train_label)
```

```
##         train_label
## trainres    0    1    2    3    4    5    6    7    8    9
##        0 2601    0   19    7    5   25   17    4    7   14
##        1    0 3020    6    8    8    8    4   19   19   12
##        2   16   15 2545   57   16   21   23   47   24   13
##        3   10   18   47 2616    5  150    4   28   73   43
##        4    6    2   32    3 2555   23   26   26   10   93
##        5   28    9    9   78    5 2131   37   11   43   19
##        6   28    3   43   14   32   57 2637    7   21    2
##        7    1    7   42   30    8    6    7 2690   10   79
##        8   26   30   71   84   25   82   22   22 2456   42
##        9    5    3   10   12   97   18    2   70   31 2448
```

```
mean(trainres == train_label)
```

```
## [1] 0.9178214
```

```
table(validres, valid_label)
```

```
##         valid_label
## validres    0    1    2    3    4    5    6    7    8    9
##        0 1337    0    7    2    0   12   16    3    2   10
##        1    0 1529    1    5   11    8    4    8   22    7
##        2    8    7 1209   44   17   14   17   22   18   10
##        3   10   12   31 1276    2   93    1    9   36   23
##        4    3    0   14    2 1184   16   18   21    4   59
##        5   22    3    9   45    2 1035   17    8   22   13
##        6   15    1   18   14   16   26 1268    0    8    2
##        7    2    4   14   14    5    7    1 1354    7   54
##        8   14   20   44   31   11   45   16    6 1238   26
##        9    0    1    6    9   68   18    0   46   12 1219
```

```
mean(validres == valid_label)
```

```
## [1] 0.9035
```