Multiway Classification with Real Data

Matthew Asisgress

2025-10-24

MNIST

Download

```r
# load library
library(dslabs)

# download MNIST data and subset training set to digits of 2 or 3
mnist <- read_mnist()
inde <- which(mnist$train$labels %in% c(2, 3))
images <- mnist$train$images[inde, ]
labels <- mnist$train$labels[inde]

# restructure data into a three-way array and prepare labels
X0 <- array(images, c(nrow(images), 28, 28))
y0 <- as.factor(labels - 2)

# subset to first 300 observations
ind <- 1:3e2
X <- X0[ind, , ]
y <- y0[ind]
```

Analysis

```r
# load library
library(cpfa)
```

```
## Loading required package: multiway
```

```
## Loading required package: CMLS
```

```
## Loading required package: quadprog
```

```
## Loading required package: parallel
```

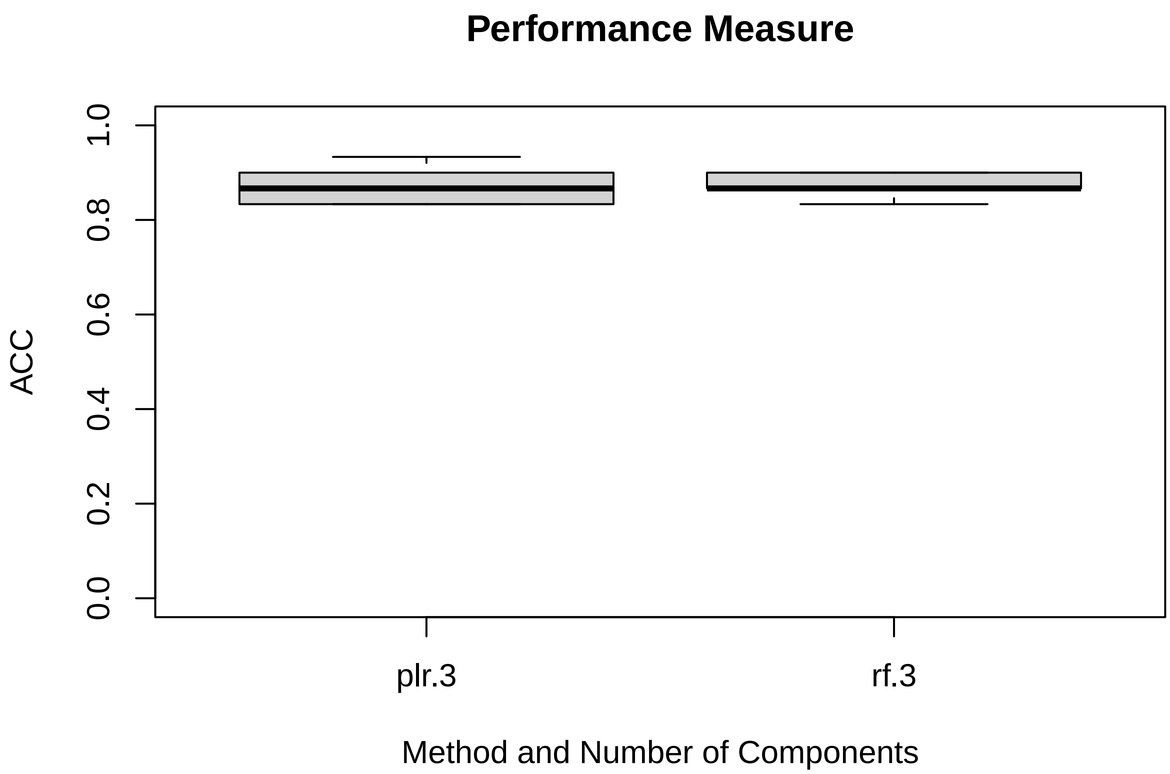
```r
# set seed
set.seed(500)

# initialize alpha, number of trees, and node size for PLR and RF
alpha <- seq(0, 1, length = 8)
ntree <- c(400, 600, 800, 1000)
nodesize <- c(4, 8, 16, 32)
parameters <- list(alpha = alpha, ntree = ntree, nodesize = nodesize)
```

```r
# initialize inputs
method <- c("PLR", "RF")
family <- "binomial"
model <- "parafac"
nfolds <- 10
nstart <- 10
const <- c("uncons", "uncons", "uncons")

# implement train-test splits with inner k-fold CV to optimize classification
outputR <- cpfa(x = X, y = y, model = model, nfac = 3, nrep = 5,
                ratio = 0.9, nfolds = nfolds, method = method, family = family,
                parameters = parameters, cmode = 1, type.out = "descriptives",
                seeds = NULL, plot.out = TRUE, parallel = FALSE, const = const,
                nstart = nstart)
```

```
## nrep = 1
## nfac = 3 model = parafac
##   |                                                                              |
## nfac = 3 method = plr
## nfac = 3 method = rf
## nrep = 2
## nfac = 3 model = parafac
##   |                                                                              |
## nfac = 3 method = plr
## nfac = 3 method = rf
## nrep = 3
## nfac = 3 model = parafac
##   |                                                                              |
## nfac = 3 method = plr
## nfac = 3 method = rf
## nrep = 4
## nfac = 3 model = parafac
##   |                                                                              |
## nfac = 3 method = plr
## nfac = 3 method = rf
## nrep = 5
## nfac = 3 model = parafac
##   |                                                                              |
## nfac = 3 method = plr
## nfac = 3 method = rf
```



Results

Classification Metrics

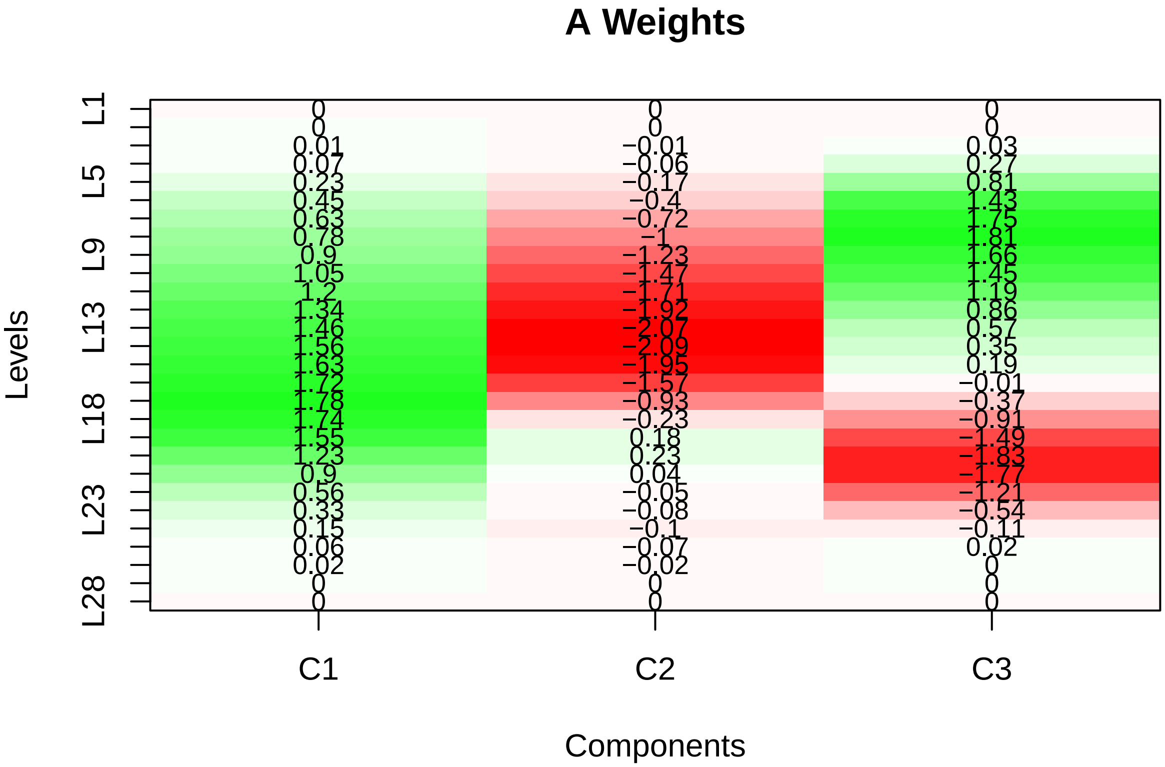
```
# examine classification performance measures - median across train-test splits
outputR$descriptive$median[, 1:2]
```

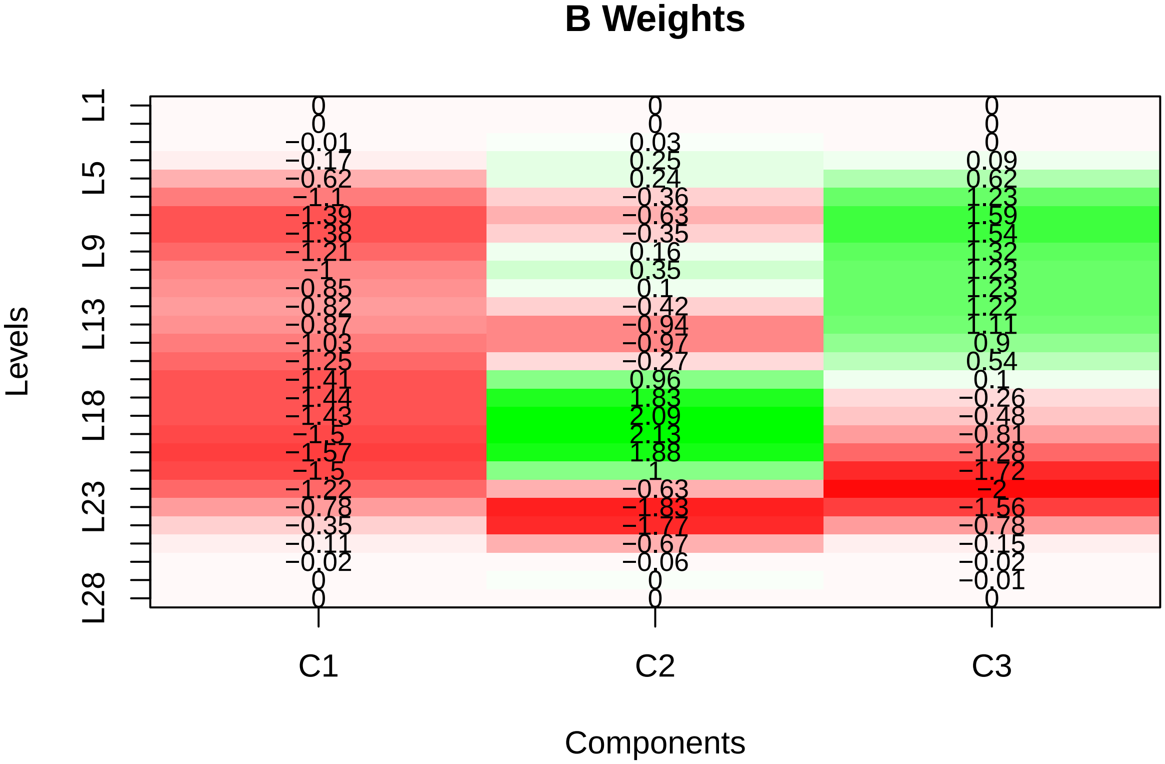
```
##                  err       acc
## fac.3plr 0.1333333 0.8666667
## fac.3rf  0.1333333 0.8666667
```

```
# examine optimal tuning parameters averaged across train-test splits
outputR$mean.opt.tune
```

```
##   nfac alpha   lambda gamma cost ntree nodesize size decay rda.alpha delta eta
## 1    3   0.2 54.15575    NA   NA   560        4   NA    NA        NA    NA  NA
##   max.depth subsample nrounds
## 1        NA        NA      NA
```

Plots





(note: this work is in progress with more updates anticipated in November 2025)