Digit Recognizer EDA

Anandan Balaji

27 October 2016

Contents

Digit Recognizer - Exploratory Data Analysis.

The Digit Recognizer dataset was downloaded from kaggle. It's based on MNIST database.

As usual, let's understand the data first.

Read the Data

```r
library(data.table)

train <- fread("../dataset/train.csv", header = TRUE)
```

```
##
Read 95.2% of 42000 rows
Read 42000 rows and 785 (of 785) columns from 0.072 GB file in 00:00:03
```

```r
str(train)
```

```
## Classes 'data.table' and 'data.frame':   42000 obs. of  785 variables:
##  $ label   : int  1 0 1 4 0 0 7 3 5 3 ...
##  $ pixel0  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel1  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel2  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel3  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel4  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel5  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel6  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel7  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel8  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel9  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel10 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel11 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel12 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel13 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel14 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel15 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel16 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel17 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel18 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel19 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel20 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel21 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel22 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel23 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel24 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel25 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel26 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel27 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel28 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel29 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel30 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel31 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel32 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel33 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel34 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel35 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel36 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel37 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel38 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel39 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel40 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel41 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel42 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel43 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel44 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel45 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel46 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel47 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel48 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel49 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel50 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel51 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel52 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel53 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel54 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel55 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel56 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel57 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel58 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel59 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel60 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel61 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel62 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel63 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel64 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel65 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel66 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel67 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel68 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel69 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel70 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel71 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel72 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel73 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel74 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel75 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel76 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel77 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel78 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel79 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel80 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel81 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel82 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel83 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel84 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel85 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel86 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel87 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel88 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel89 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel90 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel91 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel92 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel93 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel94 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel95 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel96 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ pixel97 : int  0 0 0 0 0 0 0 0 0 0 ...
##   [list output truncated]
##  - attr(*, ".internal.selfref")=<externalptr>
```

```r
nrow(train)
```

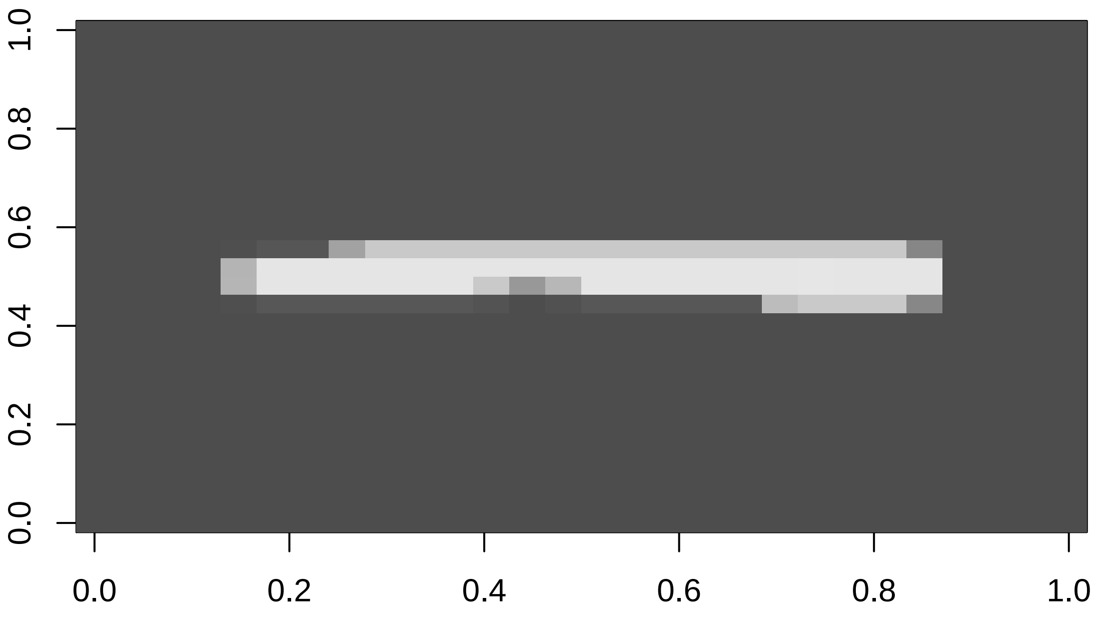
```
## [1] 42000
```

```r
ncol(train)
```

```
## [1] 785
```

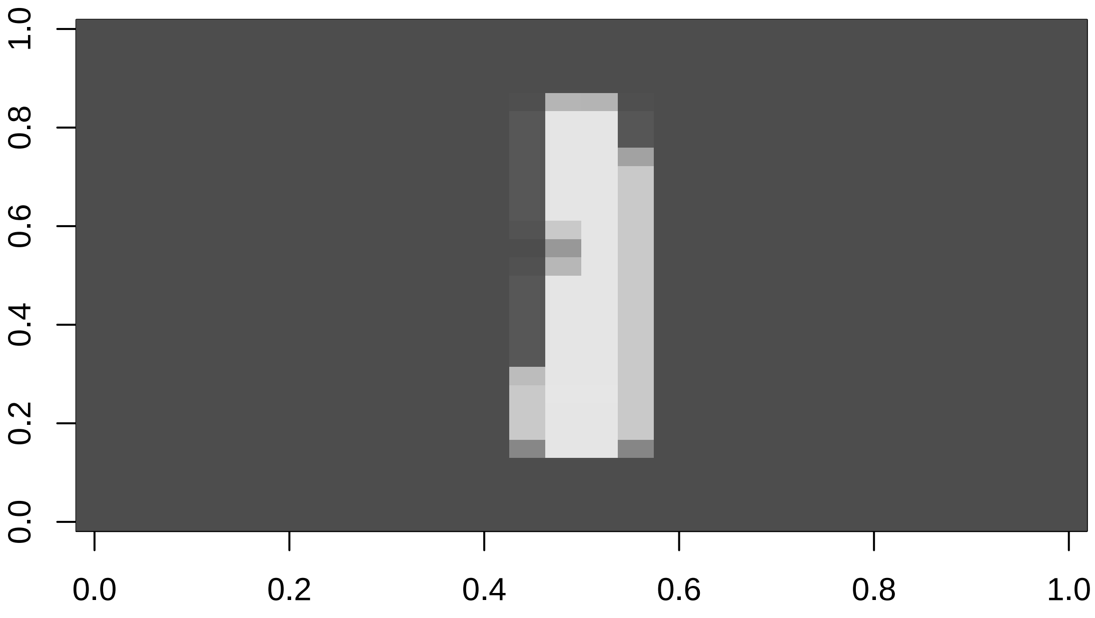
Display one of the row.

```r
# each row has 785 columns, ie., 28 * 28 -> 784 + 1 target variable.
label_idx <- 1
# each row is a 28x28 -> 784 pixels. create a matrix for a given row
row3 <- matrix(unlist(train[3, -label_idx, with=FALSE]), nrow = 28, byrow = TRUE)

#plot row 3
image(row3, col=grey.colors(255))
```



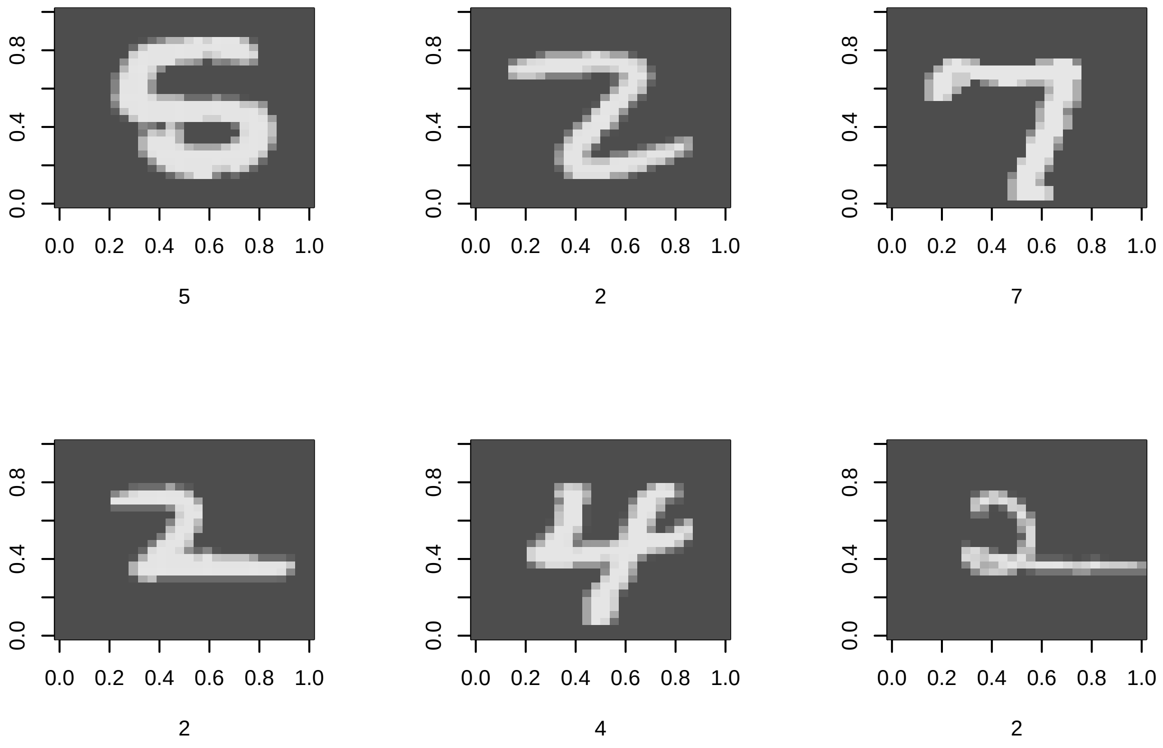
Rotate the image

```r
# define funciton rotate
rotate <- function(x) {
  # transpose after reversing the input
  t(apply(x,2, rev))
}

# try the  example
image(rotate(row3), col=grey.colors(255))
```



Plot few more

```r
# plot few more images
# set the params for image
par(mfrow=c(2,3))

lapply(
    c(20,25,30,45,50,56),
      function (x) image(
                            rotate(matrix(unlist(train[x,-1, with=FALSE]), nrow = 28, byrow=T)),
                            col = grey.colors(255),
                            xlab = train[x,1, with=FALSE])
    )
```



```
## [[1]]
## NULL
##
## [[2]]
## NULL
##
## [[3]]
## NULL
##
## [[4]]
## NULL
##
## [[5]]
## NULL
##
```

```
## [[6]]
## NULL
```