k-NN *class*

Wingmor99

2/20/2019

exploration and preparation

```
## 'data.frame':    569 obs. of  32 variables:
##  $ id                : int  87139402 8910251 905520 868871 9012568 906539 925291 87880 862989 89827 .
##  $ diagnosis         : chr  "B" "B" "B" "B" ...
##  $ radius_mean       : num  12.3 10.6 11 11.3 15.2 ...
##  $ texture_mean      : num  12.4 18.9 16.8 13.4 13.2 ...
##  $ perimeter_mean    : num  78.8 69.3 70.9 73 97.7 ...
##  $ area_mean         : num  464 346 373 385 712 ...
##  $ smoothness_mean   : num  0.1028 0.0969 0.1077 0.1164 0.0796 ...
##  $ compactness_mean : num  0.0698 0.1147 0.078 0.1136 0.0693 ...
##  $ concavity_mean    : num  0.0399 0.0639 0.0305 0.0464 0.0339 ...
##  $ points_mean       : num  0.037 0.0264 0.0248 0.048 0.0266 ...
##  $ symmetry_mean     : num  0.196 0.192 0.171 0.177 0.172 ...
##  $ dimension_mean    : num  0.0595 0.0649 0.0634 0.0607 0.0554 ...
##  $ radius_se         : num  0.236 0.451 0.197 0.338 0.178 ...
##  $ texture_se        : num  0.666 1.197 1.387 1.343 0.412 ...
##  $ perimeter_se      : num  1.67 3.43 1.34 1.85 1.34 ...
##  $ area_se           : num  17.4 27.1 13.5 26.3 17.7 ...
##  $ smoothness_se     : num  0.00805 0.00747 0.00516 0.01127 0.00501 ...
##  $ compactness_se    : num  0.0118 0.03581 0.00936 0.03498 0.01485 ...
##  $ concavity_se      : num  0.0168 0.0335 0.0106 0.0219 0.0155 ...
##  $ points_se         : num  0.01241 0.01365 0.00748 0.01965 0.00915 ...
##  $ symmetry_se       : num  0.0192 0.035 0.0172 0.0158 0.0165 ...
##  $ dimension_se      : num  0.00225 0.00332 0.0022 0.00344 0.00177 ...
##  $ radius_worst      : num  13.5 11.9 12.4 11.9 16.2 ...
##  $ texture_worst     : num  15.6 22.9 26.4 15.8 15.7 ...
##  $ perimeter_worst   : num  87 78.3 79.9 76.5 104.5 ...
##  $ area_worst        : num  549 425 471 434 819 ...
##  $ smoothness_worst : num  0.139 0.121 0.137 0.137 0.113 ...
##  $ compactness_worst: num  0.127 0.252 0.148 0.182 0.174 ...
##  $ concavity_worst   : num  0.1242 0.1916 0.1067 0.0867 0.1362 ...
##  $ points_worst      : num  0.0939 0.0793 0.0743 0.0861 0.0818 ...
##  $ symmetry_worst    : num  0.283 0.294 0.3 0.21 0.249 ...
##  $ dimension_worst   : num  0.0677 0.0759 0.0788 0.0678 0.0677 ...

##
##   B   M
## 357 212
```

labels inside the *factor* can rename the variable.

```
wbcd$diagnosis <- factor(wbcd$diagnosis, levels = c("B", "M"),
                         labels = c("Benign", "Malignant"))
```

prop.table returns a propotion table.

```r
round(prop.table(table(wbcd$diagnosis)) * 100, digits = 1)
```

```
##
##    Benign Malignant
##      62.7      37.3
```

k-NN is heavily dependent upon the measurement scale of the input features. So we need to normalizing numeric data.

```r
# create normalization function
normalize <- function(x) {
  return ((x - min(x)) / (max(x) - min(x)))
}
wbcd_n <- as.data.frame(lapply(wbcd[2:31], normalize))
```

```r
# create training and test data
wbcd_train <- wbcd_n[1:469, ]
wbcd_test <- wbcd_n[470:569, ]
# create labels for training and test data

wbcd_train_labels <- wbcd[1:469, 1]
wbcd_test_labels <- wbcd[470:569, 1]
```

Training a k-NN model

k-NN is in **class** library.

```r
# load the "class" library
library(class)

wbcd_test_pred <- knn(train = wbcd_train, test = wbcd_test,
                      cl = wbcd_train_labels, k = 21)
```

Evaluating model

CrossTable in *gmodels*

```r
library(gmodels)

# Create the cross tabulation of predicted vs. actual
CrossTable(x = wbcd_test_labels, y = wbcd_test_pred,
           prop.chisq = FALSE)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
```

```
## |           N / Row Total |
## |           N / Col Total |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  100
##
##
##                  | wbcd_test_pred
## wbcd_test_labels |    Benign | Malignant | Row Total |
## -----------------|-----------|-----------|-----------|
##           Benign |        61 |         0 |        61 |
##                  |     1.000 |     0.000 |     0.610 |
##                  |     0.968 |     0.000 |           |
##                  |     0.610 |     0.000 |           |
## -----------------|-----------|-----------|-----------|
##        Malignant |         2 |        37 |        39 |
##                  |     0.051 |     0.949 |     0.390 |
##                  |     0.032 |     1.000 |           |
##                  |     0.020 |     0.370 |           |
## -----------------|-----------|-----------|-----------|
##     Column Total |        63 |        37 |       100 |
##                  |     0.630 |     0.370 |           |
## -----------------|-----------|-----------|-----------|
##
##
```

Improving model performance

Alternative method for erscaling numeric features

Z-score standardization

scale can standardlize a vetor.

```
# use the scale() function to z-score standardize a data frame
wbcd_z <- as.data.frame(scale(wbcd[-1]))
# confirm that the transformation was applied correctly
summary(wbcd_z$area_mean)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -1.4532 -0.6666 -0.2949  0.0000  0.3632  5.2459
```

```
# create training and test datasets
wbcd_train <- wbcd_z[1:469, ]
wbcd_test <- wbcd_z[470:569, ]

# re-classify test cases
wbcd_test_pred <- knn(train = wbcd_train, test = wbcd_test,
                      cl = wbcd_train_labels, k = 21)
```

```r
# Create the cross tabulation of predicted vs. actual
CrossTable(x = wbcd_test_labels, y = wbcd_test_pred,
           prop.chisq = FALSE)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  100
##
##
##                  | wbcd_test_pred
## wbcd_test_labels |    Benign | Malignant | Row Total |
## -----------------|-----------|-----------|-----------|
##           Benign |        61 |         0 |        61 |
##                  |     1.000 |     0.000 |     0.610 |
##                  |     0.924 |     0.000 |           |
##                  |     0.610 |     0.000 |           |
## -----------------|-----------|-----------|-----------|
##        Malignant |         5 |        34 |        39 |
##                  |     0.128 |     0.872 |     0.390 |
##                  |     0.076 |     1.000 |           |
##                  |     0.050 |     0.340 |           |
## -----------------|-----------|-----------|-----------|
##     Column Total |        66 |        34 |       100 |
##                  |     0.660 |     0.340 |           |
## -----------------|-----------|-----------|-----------|
##
##
```

Try different K.

```r
# try several different values of k
wbcd_train <- wbcd_n[1:469, ]
wbcd_test <- wbcd_n[470:569, ]


kValue <- c(1, 5, 11, 15, 21, 27)
for (i in kValue) {
  wbcd_test_pred <- knn(train = wbcd_train, test = wbcd_test, cl = wbcd_train_labels, k=1)
  CrossTable(x = wbcd_test_labels, y = wbcd_test_pred, prop.chisq=FALSE)
}
```