iris_data_classification_report

seunghun_Oh

2022-10-28

```
setwd("/Users/sh_oh/Dropbox/iris")
```

```
# 1.데이터읽기 및 수동 One-Hot Encoding, 데이터 설정

iris = read.csv(file = "iris.csv", stringsAsFactors = TRUE)

iris$Species.setosa[iris$Species=="setosa"]=    1
iris$Species.setosa[iris$Species!="setosa"]=    0
iris$Species.versicolor[iris$Species=="versicolor"]=    1
iris$Species.versicolor[iris$Species!="versicolor"]=    0
iris$Species.virginica[iris$Species=="virginica"]=  1
iris$Species.virginica[iris$Species!="virginica"]=  0

x   <-  matrix(c(iris$Sepal.Length, iris$Sepal.Width, iris$Petal.Length,iris$Petal.Wi
dth), nrow=150)
x
```

```
##        [,1] [,2] [,3] [,4]
##   [1,]  5.1  3.5  1.4  0.2
##   [2,]  4.9  3.0  1.4  0.2
##   [3,]  4.7  3.2  1.3  0.2
##   [4,]  4.6  3.1  1.5  0.2
##   [5,]  5.0  3.6  1.4  0.2
##   [6,]  5.4  3.9  1.7  0.4
##   [7,]  4.6  3.4  1.4  0.3
##   [8,]  5.0  3.4  1.5  0.2
##   [9,]  4.4  2.9  1.4  0.2
##  [10,]  4.9  3.1  1.5  0.1
##  [11,]  5.4  3.7  1.5  0.2
##  [12,]  4.8  3.4  1.6  0.2
##  [13,]  4.8  3.0  1.4  0.1
##  [14,]  4.3  3.0  1.1  0.1
##  [15,]  5.8  4.0  1.2  0.2
##  [16,]  5.7  4.4  1.5  0.4
##  [17,]  5.4  3.9  1.3  0.4
##  [18,]  5.1  3.5  1.4  0.3
##  [19,]  5.7  3.8  1.7  0.3
##  [20,]  5.1  3.8  1.5  0.3
##  [21,]  5.4  3.4  1.7  0.2
##  [22,]  5.1  3.7  1.5  0.4
##  [23,]  4.6  3.6  1.0  0.2
##  [24,]  5.1  3.3  1.7  0.5
##  [25,]  4.8  3.4  1.9  0.2
##  [26,]  5.0  3.0  1.6  0.2
##  [27,]  5.0  3.4  1.6  0.4
##  [28,]  5.2  3.5  1.5  0.2
##  [29,]  5.2  3.4  1.4  0.2
##  [30,]  4.7  3.2  1.6  0.2
##  [31,]  4.8  3.1  1.6  0.2
##  [32,]  5.4  3.4  1.5  0.4
##  [33,]  5.2  4.1  1.5  0.1
##  [34,]  5.5  4.2  1.4  0.2
##  [35,]  4.9  3.1  1.5  0.2
##  [36,]  5.0  3.2  1.2  0.2
##  [37,]  5.5  3.5  1.3  0.2
##  [38,]  4.9  3.6  1.4  0.1
##  [39,]  4.4  3.0  1.3  0.2
##  [40,]  5.1  3.4  1.5  0.2
##  [41,]  5.0  3.5  1.3  0.3
##  [42,]  4.5  2.3  1.3  0.3
##  [43,]  4.4  3.2  1.3  0.2
##  [44,]  5.0  3.5  1.6  0.6
##  [45,]  5.1  3.8  1.9  0.4
##  [46,]  4.8  3.0  1.4  0.3
##  [47,]  5.1  3.8  1.6  0.2
##  [48,]  4.6  3.2  1.4  0.2
##  [49,]  5.3  3.7  1.5  0.2
##  [50,]  5.0  3.3  1.4  0.2
##  [51,]  7.0  3.2  4.7  1.4
##  [52,]  6.4  3.2  4.5  1.5
##  [53,]  6.9  3.1  4.9  1.5
##  [54,]  5.5  2.3  4.0  1.3
```

```
##  [55,]  6.5  2.8  4.6  1.5
##  [56,]  5.7  2.8  4.5  1.3
##  [57,]  6.3  3.3  4.7  1.6
##  [58,]  4.9  2.4  3.3  1.0
##  [59,]  6.6  2.9  4.6  1.3
##  [60,]  5.2  2.7  3.9  1.4
##  [61,]  5.0  2.0  3.5  1.0
##  [62,]  5.9  3.0  4.2  1.5
##  [63,]  6.0  2.2  4.0  1.0
##  [64,]  6.1  2.9  4.7  1.4
##  [65,]  5.6  2.9  3.6  1.3
##  [66,]  6.7  3.1  4.4  1.4
##  [67,]  5.6  3.0  4.5  1.5
##  [68,]  5.8  2.7  4.1  1.0
##  [69,]  6.2  2.2  4.5  1.5
##  [70,]  5.6  2.5  3.9  1.1
##  [71,]  5.9  3.2  4.8  1.8
##  [72,]  6.1  2.8  4.0  1.3
##  [73,]  6.3  2.5  4.9  1.5
##  [74,]  6.1  2.8  4.7  1.2
##  [75,]  6.4  2.9  4.3  1.3
##  [76,]  6.6  3.0  4.4  1.4
##  [77,]  6.8  2.8  4.8  1.4
##  [78,]  6.7  3.0  5.0  1.7
##  [79,]  6.0  2.9  4.5  1.5
##  [80,]  5.7  2.6  3.5  1.0
##  [81,]  5.5  2.4  3.8  1.1
##  [82,]  5.5  2.4  3.7  1.0
##  [83,]  5.8  2.7  3.9  1.2
##  [84,]  6.0  2.7  5.1  1.6
##  [85,]  5.4  3.0  4.5  1.5
##  [86,]  6.0  3.4  4.5  1.6
##  [87,]  6.7  3.1  4.7  1.5
##  [88,]  6.3  2.3  4.4  1.3
##  [89,]  5.6  3.0  4.1  1.3
##  [90,]  5.5  2.5  4.0  1.3
##  [91,]  5.5  2.6  4.4  1.2
##  [92,]  6.1  3.0  4.6  1.4
##  [93,]  5.8  2.6  4.0  1.2
##  [94,]  5.0  2.3  3.3  1.0
##  [95,]  5.6  2.7  4.2  1.3
##  [96,]  5.7  3.0  4.2  1.2
##  [97,]  5.7  2.9  4.2  1.3
##  [98,]  6.2  2.9  4.3  1.3
##  [99,]  5.1  2.5  3.0  1.1
## [100,]  5.7  2.8  4.1  1.3
## [101,]  6.3  3.3  6.0  2.5
## [102,]  5.8  2.7  5.1  1.9
## [103,]  7.1  3.0  5.9  2.1
## [104,]  6.3  2.9  5.6  1.8
## [105,]  6.5  3.0  5.8  2.2
## [106,]  7.6  3.0  6.6  2.1
## [107,]  4.9  2.5  4.5  1.7
## [108,]  7.3  2.9  6.3  1.8
## [109,]  6.7  2.5  5.8  1.8
## [110,]  7.2  3.6  6.1  2.5
```

```
## [111,]  6.5  3.2  5.1  2.0
## [112,]  6.4  2.7  5.3  1.9
## [113,]  6.8  3.0  5.5  2.1
## [114,]  5.7  2.5  5.0  2.0
## [115,]  5.8  2.8  5.1  2.4
## [116,]  6.4  3.2  5.3  2.3
## [117,]  6.5  3.0  5.5  1.8
## [118,]  7.7  3.8  6.7  2.2
## [119,]  7.7  2.6  6.9  2.3
## [120,]  6.0  2.2  5.0  1.5
## [121,]  6.9  3.2  5.7  2.3
## [122,]  5.6  2.8  4.9  2.0
## [123,]  7.7  2.8  6.7  2.0
## [124,]  6.3  2.7  4.9  1.8
## [125,]  6.7  3.3  5.7  2.1
## [126,]  7.2  3.2  6.0  1.8
## [127,]  6.2  2.8  4.8  1.8
## [128,]  6.1  3.0  4.9  1.8
## [129,]  6.4  2.8  5.6  2.1
## [130,]  7.2  3.0  5.8  1.6
## [131,]  7.4  2.8  6.1  1.9
## [132,]  7.9  3.8  6.4  2.0
## [133,]  6.4  2.8  5.6  2.2
## [134,]  6.3  2.8  5.1  1.5
## [135,]  6.1  2.6  5.6  1.4
## [136,]  7.7  3.0  6.1  2.3
## [137,]  6.3  3.4  5.6  2.4
## [138,]  6.4  3.1  5.5  1.8
## [139,]  6.0  3.0  4.8  1.8
## [140,]  6.9  3.1  5.4  2.1
## [141,]  6.7  3.1  5.6  2.4
## [142,]  6.9  3.1  5.1  2.3
## [143,]  5.8  2.7  5.1  1.9
## [144,]  6.8  3.2  5.9  2.3
## [145,]  6.7  3.3  5.7  2.5
## [146,]  6.7  3.0  5.2  2.3
## [147,]  6.3  2.5  5.0  1.9
## [148,]  6.5  3.0  5.2  2.0
## [149,]  6.2  3.4  5.4  2.3
## [150,]  5.9  3.0  5.1  1.8
```

```
y   <-  matrix(c(iris$Species.setosa,   iris$Species.versicolor,    iris$Species.virg
inica), nrow=150)
y
```

```
##        [,1] [,2] [,3]
##   [1,]    1    0    0
##   [2,]    1    0    0
##   [3,]    1    0    0
##   [4,]    1    0    0
##   [5,]    1    0    0
##   [6,]    1    0    0
##   [7,]    1    0    0
##   [8,]    1    0    0
##   [9,]    1    0    0
##  [10,]    1    0    0
##  [11,]    1    0    0
##  [12,]    1    0    0
##  [13,]    1    0    0
##  [14,]    1    0    0
##  [15,]    1    0    0
##  [16,]    1    0    0
##  [17,]    1    0    0
##  [18,]    1    0    0
##  [19,]    1    0    0
##  [20,]    1    0    0
##  [21,]    1    0    0
##  [22,]    1    0    0
##  [23,]    1    0    0
##  [24,]    1    0    0
##  [25,]    1    0    0
##  [26,]    1    0    0
##  [27,]    1    0    0
##  [28,]    1    0    0
##  [29,]    1    0    0
##  [30,]    1    0    0
##  [31,]    1    0    0
##  [32,]    1    0    0
##  [33,]    1    0    0
##  [34,]    1    0    0
##  [35,]    1    0    0
##  [36,]    1    0    0
##  [37,]    1    0    0
##  [38,]    1    0    0
##  [39,]    1    0    0
##  [40,]    1    0    0
##  [41,]    1    0    0
##  [42,]    1    0    0
##  [43,]    1    0    0
##  [44,]    1    0    0
##  [45,]    1    0    0
##  [46,]    1    0    0
##  [47,]    1    0    0
##  [48,]    1    0    0
##  [49,]    1    0    0
##  [50,]    1    0    0
##  [51,]    0    1    0
##  [52,]    0    1    0
##  [53,]    0    1    0
##  [54,]    0    1    0
```

```
##  [55,]    0    1    0
##  [56,]    0    1    0
##  [57,]    0    1    0
##  [58,]    0    1    0
##  [59,]    0    1    0
##  [60,]    0    1    0
##  [61,]    0    1    0
##  [62,]    0    1    0
##  [63,]    0    1    0
##  [64,]    0    1    0
##  [65,]    0    1    0
##  [66,]    0    1    0
##  [67,]    0    1    0
##  [68,]    0    1    0
##  [69,]    0    1    0
##  [70,]    0    1    0
##  [71,]    0    1    0
##  [72,]    0    1    0
##  [73,]    0    1    0
##  [74,]    0    1    0
##  [75,]    0    1    0
##  [76,]    0    1    0
##  [77,]    0    1    0
##  [78,]    0    1    0
##  [79,]    0    1    0
##  [80,]    0    1    0
##  [81,]    0    1    0
##  [82,]    0    1    0
##  [83,]    0    1    0
##  [84,]    0    1    0
##  [85,]    0    1    0
##  [86,]    0    1    0
##  [87,]    0    1    0
##  [88,]    0    1    0
##  [89,]    0    1    0
##  [90,]    0    1    0
##  [91,]    0    1    0
##  [92,]    0    1    0
##  [93,]    0    1    0
##  [94,]    0    1    0
##  [95,]    0    1    0
##  [96,]    0    1    0
##  [97,]    0    1    0
##  [98,]    0    1    0
##  [99,]    0    1    0
## [100,]    0    1    0
## [101,]    0    0    1
## [102,]    0    0    1
## [103,]    0    0    1
## [104,]    0    0    1
## [105,]    0    0    1
## [106,]    0    0    1
## [107,]    0    0    1
## [108,]    0    0    1
## [109,]    0    0    1
## [110,]    0    0    1
```

```
## [111,]    0    0    1
## [112,]    0    0    1
## [113,]    0    0    1
## [114,]    0    0    1
## [115,]    0    0    1
## [116,]    0    0    1
## [117,]    0    0    1
## [118,]    0    0    1
## [119,]    0    0    1
## [120,]    0    0    1
## [121,]    0    0    1
## [122,]    0    0    1
## [123,]    0    0    1
## [124,]    0    0    1
## [125,]    0    0    1
## [126,]    0    0    1
## [127,]    0    0    1
## [128,]    0    0    1
## [129,]    0    0    1
## [130,]    0    0    1
## [131,]    0    0    1
## [132,]    0    0    1
## [133,]    0    0    1
## [134,]    0    0    1
## [135,]    0    0    1
## [136,]    0    0    1
## [137,]    0    0    1
## [138,]    0    0    1
## [139,]    0    0    1
## [140,]    0    0    1
## [141,]    0    0    1
## [142,]    0    0    1
## [143,]    0    0    1
## [144,]    0    0    1
## [145,]    0    0    1
## [146,]    0    0    1
## [147,]    0    0    1
## [148,]    0    0    1
## [149,]    0    0    1
## [150,]    0    0    1
```

```
# 2.데이터분리(생략): 훈련데이터와 테스트데이터를 나누지 않음
```

```
# 3. 활성화함수 설정
 # activation_finction1-(tau: sigmoid): 입력층 -> 은닉층
 # activation_finction2-(softmax: softmax): 은닉층 -> 출력층
 # tau1d: tau의 일계도함수

tau <-  function(x) 1/(1    +   exp(-x))
tau1d   <-  function(x) tau(x)*(1   -   tau(x))

softmax <-  function(x) {
  nDim  =   length(x)
  res   =   rep(0,  nDim)
  res   =   matrix(res, nrow    =   nrow(x),    byrow   =   FALSE)

  ExpMatr <- matrix(0, nrow(x), ncol(x))
  SumExpMatr <- rep(0, nrow(x))
  for   (k  in  1:ncol(x))  {
    for (h  in  1:nrow(x))  {
      ExpMatr[h, k] =   exp(x[h, k])
    }
    SumExpMatr[k] <- sum(ExpMatr[,k])
  }

  for   (i  in  1:nrow(x))  {
    for (j  in  1:ncol(x))  {
      res[i,j]  =   exp(x[i,j])/SumExpMatr[j]
    }
  }
  return(res)
}
```

```
# 4.인공신경망(NN) 모형학습-SGD Version

SGD.NN1 <-  function(X, Y,  hidden, rho = NULL, tol = NULL, max.epoch = NULL)
{
  set.seed(123)
  X <- as.matrix(X)
  Y <- as.matrix(Y)
  n <-  nrow(X)
  d <-  ncol(X)

  if (is.null(rho)) rho <- 1/n
  if (is.null(max.epoch)) max.epoch <- 500
  if (is.null(tol)) tol <- 5e-6

  ones  <-  rep(1,  n)

  # 난수 발생하여(rnorm) W1, W2 초기값 설정(출력층의 Node수: 3)

  W1    <-  matrix(rnorm(hidden*(d  +   1)),    hidden, d   +   1)
  W2    <-  matrix(rnorm(3*(hidden  +   1)),    3,  hidden  +   1)

  # Permute the data set & while문 설정

  epoch <-  0
  loss.trace    <-  rep(NA, max.epoch)

  while (epoch  <   max.epoch)  {
    epoch   <-  epoch   +   1
    ind <-  sample(1:n, n,  replace =   FALSE)

    # 가중치 업데이트를 위한 matrix 틀 설정

    X.tr    <-  as.matrix(X[ind,])
    Y.tr    <-  as.matrix(Y[ind,])
    Y.tilde <-  matrix(0,n,3)

    x <- matrix(0,n,d+1)
    s <- matrix(0,n,hidden)
    z <- matrix(0,hidden+1,n)

    for (   i   in  1:n )   {
      x[i,] <-  c(1,    X.tr[i,])
      s[i,] <-  drop(W1%*%x[i,])
      z[,i] <-  c(1 ,tau(s[i,]))
    }
    gr1.sum <-  matrix(0,   hidden, d   +   1)
    gr2.sum <-  matrix(0,   3, hidden   +   1)

    # 오차 역전파를 통한 가중치 업데이트 및 softmax를 이용한 최종 출력값 Y.tilde 출력

    Y.tilde <-  softmax(W2%*%z)
    delta   <-  t(Y.tr) -   Y.tilde
    gr2 <-  delta%*%matrix(z,   n, hidden   +   1)
    gr2.sum <-  gr2.sum +   gr2
    W2  <-  W2  +   rho*gr2
```

```
    gr1 <- matrix(0, hidden, d + 1)
    eta.k <- matrix(0, hidden, 1)
    for (   k   in  1 : hidden  )   {
      eta.k[k,] <-  sum(t(delta)%*%W2[, k   +   1]%*%tau1d(s[,k]))
      for   (   j   in  1:(d    +   1)  )   {
        gr1[,j] <-  eta.k[k,]*x[j]
        gr1.sum[k,  j]  <-  gr1.sum[k,  j]  +   gr1[k, j]
        W1[k,   j]  <-  W1[k,   j]  +   rho*gr1[k, j]
      }
    }
        loss.trace[epoch]   <-  (-1)*sum(Y%*%log(Y.tilde))
  }

  # 분류 문제이므로 손실함수로 cross-entropy를 이용 및 결측치 제거

  loss.trace    <-  na.omit(loss.trace)

  return(list(W1 = W1, W2   =   W2, Z   =   z, y.pred   =   Y.tilde, loss   =   loss.
trace, epoch   =   epoch))
}
```

```
# 5. 모델 성능 평가 with confusion_matrix & ROC curve

SGD <- SGD.NN1(X=x, Y=y, hidden=15, rho =0.001 , tol = 4e-4, max.epoch = 890)


 # confusionMatrix()에 factor값을 넣기 위해 자료의 형태 변환-1: 실제값

  factor.y <- rep(0, nrow(y))
  for(i in 1:nrow(y)){
    for(j in 1:ncol(y)){
     if( y[i,j]==1 ){ factor.y[i]<- j-1}
    }
  }
  factor.y
```

```
##   [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [38] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
##  [75] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2
## [112] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [149] 2 2
```

```
# confusionMatrix()에 factor값을 넣기 위해 자료의 형태 변환-2: 예측값

factor.ypred <- t(SGD$y.pred)
for( i in 1:nrow(factor.ypred)) {
  for( j in 1:ncol(factor.ypred)){
    if( factor.ypred[i,j]==max(factor.ypred[i,]) ){factor.ypred[i,j]<- 1}
    else{factor.ypred[i,j]<-0}
  }
}
factor.ypred <- as.matrix(factor.ypred)
factor.ypred
```

```
##        [,1] [,2] [,3]
##   [1,]    1    0    0
##   [2,]    1    0    0
##   [3,]    0    1    0
##   [4,]    0    1    0
##   [5,]    0    0    1
##   [6,]    0    1    0
##   [7,]    1    0    0
##   [8,]    0    0    1
##   [9,]    1    0    0
##  [10,]    1    0    0
##  [11,]    0    0    1
##  [12,]    1    0    0
##  [13,]    0    0    1
##  [14,]    1    0    0
##  [15,]    1    0    0
##  [16,]    1    0    0
##  [17,]    0    0    1
##  [18,]    0    1    0
##  [19,]    1    0    0
##  [20,]    0    1    0
##  [21,]    0    0    1
##  [22,]    0    0    1
##  [23,]    0    1    0
##  [24,]    0    0    1
##  [25,]    0    1    0
##  [26,]    1    0    0
##  [27,]    0    1    0
##  [28,]    0    0    1
##  [29,]    1    0    0
##  [30,]    0    0    1
##  [31,]    1    0    0
##  [32,]    1    0    0
##  [33,]    0    1    0
##  [34,]    0    1    0
##  [35,]    1    0    0
##  [36,]    1    0    0
##  [37,]    0    1    0
##  [38,]    1    0    0
##  [39,]    0    1    0
##  [40,]    1    0    0
##  [41,]    0    1    0
##  [42,]    0    0    1
##  [43,]    0    0    1
##  [44,]    0    1    0
##  [45,]    0    1    0
##  [46,]    0    1    0
##  [47,]    0    0    1
##  [48,]    0    0    1
##  [49,]    1    0    0
##  [50,]    0    1    0
##  [51,]    0    0    1
##  [52,]    1    0    0
##  [53,]    1    0    0
##  [54,]    0    1    0
```

```
##  [55,]    0    0    1
##  [56,]    0    0    1
##  [57,]    0    0    1
##  [58,]    1    0    0
##  [59,]    0    1    0
##  [60,]    0    0    1
##  [61,]    0    1    0
##  [62,]    0    0    1
##  [63,]    1    0    0
##  [64,]    0    0    1
##  [65,]    0    0    1
##  [66,]    1    0    0
##  [67,]    1    0    0
##  [68,]    1    0    0
##  [69,]    0    0    1
##  [70,]    0    0    1
##  [71,]    0    0    1
##  [72,]    1    0    0
##  [73,]    0    1    0
##  [74,]    1    0    0
##  [75,]    0    1    0
##  [76,]    1    0    0
##  [77,]    1    0    0
##  [78,]    0    0    1
##  [79,]    0    1    0
##  [80,]    0    1    0
##  [81,]    0    1    0
##  [82,]    0    0    1
##  [83,]    0    1    0
##  [84,]    0    0    1
##  [85,]    1    0    0
##  [86,]    0    0    1
##  [87,]    1    0    0
##  [88,]    0    0    1
##  [89,]    0    0    1
##  [90,]    1    0    0
##  [91,]    0    0    1
##  [92,]    0    1    0
##  [93,]    0    0    1
##  [94,]    0    1    0
##  [95,]    0    1    0
##  [96,]    0    0    1
##  [97,]    1    0    0
##  [98,]    0    1    0
##  [99,]    1    0    0
## [100,]    0    0    1
## [101,]    1    0    0
## [102,]    0    1    0
## [103,]    0    0    1
## [104,]    0    0    1
## [105,]    1    0    0
## [106,]    0    1    0
## [107,]    0    1    0
## [108,]    0    1    0
## [109,]    0    0    1
## [110,]    0    1    0
```

```
## [111,]    0    0    1
## [112,]    1    0    0
## [113,]    0    1    0
## [114,]    0    1    0
## [115,]    1    0    0
## [116,]    1    0    0
## [117,]    1    0    0
## [118,]    0    0    1
## [119,]    0    1    0
## [120,]    0    0    1
## [121,]    1    0    0
## [122,]    0    1    0
## [123,]    0    1    0
## [124,]    0    1    0
## [125,]    0    1    0
## [126,]    0    0    1
## [127,]    1    0    0
## [128,]    1    0    0
## [129,]    0    1    0
## [130,]    0    0    1
## [131,]    0    1    0
## [132,]    1    0    0
## [133,]    0    0    1
## [134,]    0    1    0
## [135,]    0    1    0
## [136,]    0    1    0
## [137,]    0    0    1
## [138,]    1    0    0
## [139,]    1    0    0
## [140,]    0    0    1
## [141,]    0    1    0
## [142,]    1    0    0
## [143,]    0    0    1
## [144,]    0    1    0
## [145,]    0    0    1
## [146,]    0    0    1
## [147,]    1    0    0
## [148,]    0    0    1
## [149,]    0    1    0
## [150,]    1    0    0
```

```
predf.y <- rep(0, nrow(factor.ypred))
for(i in 1:nrow(factor.ypred)){
 for(j in 1:ncol(factor.ypred)){
   if( factor.ypred[i,j]==1 ){ predf.y[i]<- j-1}
 }
}
predf.y
```

```
##   [1] 0 0 1 1 2 1 0 2 0 0 2 0 2 0 0 0 2 1 0 1 2 2 1 2 1 0 1 2 0 2 0 0 1 1 0 0 1
##  [38] 0 1 0 1 2 2 1 1 1 2 2 0 1 2 0 0 1 2 2 2 0 1 2 1 2 0 2 2 0 0 0 2 2 2 0 1 0
##  [75] 1 0 0 2 1 1 1 2 1 2 0 2 0 2 2 0 2 1 2 1 1 2 0 1 0 2 0 1 2 2 0 1 1 1 2 1 2
## [112] 0 1 1 0 0 0 2 1 2 0 1 1 1 1 2 0 0 1 2 1 0 2 1 1 1 2 0 0 2 1 0 2 1 2 2 0 2
## [149] 1 0
```

```
# confusionMatrix(혼동행렬) 계산

library(caret)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```

```
confusionMatrix(as.factor(predf.y), as.factor(factor.y))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1  2
##          0 19 16 15
##          1 17 13 20
##          2 14 21 15
##
## Overall Statistics
##
##                Accuracy : 0.3133
##                  95% CI : (0.2402, 0.3941)
##     No Information Rate : 0.3333
##     P-Value [Acc > NIR] : 0.7257
##
##                   Kappa : -0.03
##
##  Mcnemar's Test P-Value : 0.9931
##
## Statistics by Class:
##
##                      Class: 0 Class: 1 Class: 2
## Sensitivity            0.3800  0.26000   0.3000
## Specificity            0.6900  0.63000   0.6500
## Pos Pred Value         0.3800  0.26000   0.3000
## Neg Pred Value         0.6900  0.63000   0.6500
## Prevalence             0.3333  0.33333   0.3333
## Detection Rate         0.1267  0.08667   0.1000
## Detection Prevalence   0.3333  0.33333   0.3333
## Balanced Accuracy      0.5350  0.44500   0.4750
```

```
# ROC Curve 그리기
 #install.packages("pROC")

library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

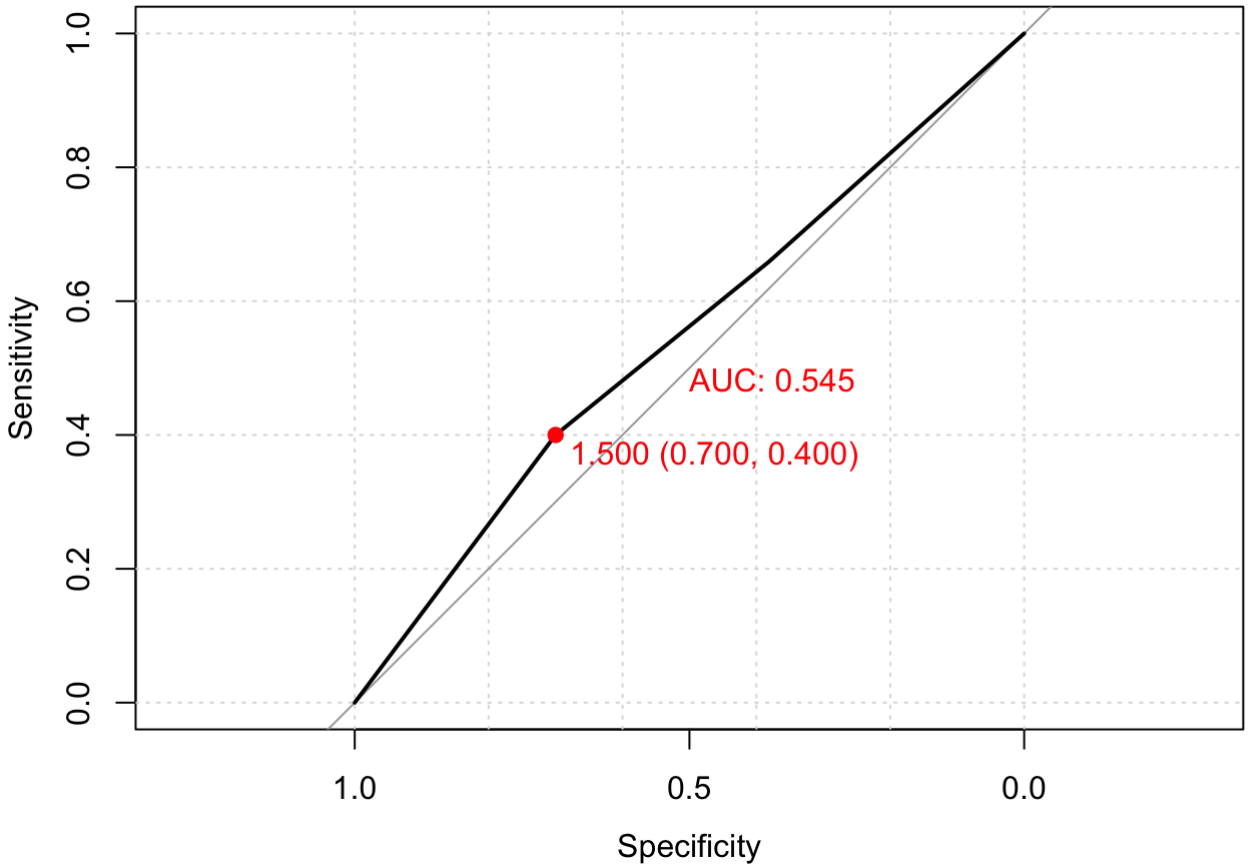
```
 iris.roc <- roc(predf.y, factor.y)
```

```
## Warning in roc.default(predf.y, factor.y): 'response' has more than two levels.
## Consider setting 'levels' explicitly or using 'multiclass.roc' instead
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
 plot.roc(iris.roc,

         col='black', # 선의 색

         print.auc=TRUE,  #auc 출력

         print.auc.col='red', #auc 색

         print.thres=TRUE, # theshold 출력

         print.thres.pch=19, #theshold 점 모양

         print.thres.col = "red", #threhold 색

         grid=c(0.2, 0.2)) #격자
```


Sensitivity
0.0
0.2
0.4
0.6
0.8
1.0
AUC: 0.545
1.500 (0.700, 0.400)
1.0
0.5
0.0
Specificity