Problem_14_3

November 7, 2021

[ ]:
```
using LinearAlgebra

include("iris_flower_data.jl")
include("iris_multiclass_helpers.jl")
```

[ ]:
```
confusion_matrix (generic function with 1 method)
```

[ ]:
```
X_train = X[:,1:100]'
X_test = X[:,101:150]'
y_train = y[1:100]'
y_test = y[101:150]'
```

[ ]:
```
1×50 adjoint(::Vector{Float64}) with eltype Float64:
 2.0  3.0  2.0  2.0  1.0  1.0  3.0  3.0  …  1.0  3.0  1.0  2.0  1.0  2.0  1.0
```

[ ]:
```
y1_train = 1*(y_train .== 1)'
y1_test = 1*(y_test .== 1)'
y2_train = 1*(y_train .== 2)'
y2_test = 1*(y_test .== 2)'
y3_train = 1*(y_train .== 3)'
y3_test = 1*(y_test .== 3)'
```

[ ]:
```
50×1 Matrix{Int64}:
 0
 1
 0
 0
 0
 0
 1
 1
 1
 0
 0
 0
 1

 0
```

```
0
0
0
0
0
1
0
0
0
0
0
```

[ ]:
```
A_train = [ones(100) X_train]
```

[ ]:
```
100×5 Matrix{Float64}:
 1.0  4.9  3.1  1.5  0.1
 1.0  4.8  3.4  1.6  0.2
 1.0  6.0  2.7  5.1  1.6
 1.0  6.4  3.2  4.5  1.5
 1.0  5.1  3.8  1.9  0.4
 1.0  6.7  3.1  4.7  1.5
 1.0  4.3  3.0  1.1  0.1
 1.0  6.8  3.2  5.9  2.3
 1.0  6.3  2.8  5.1  1.5
 1.0  6.7  3.1  5.6  2.4
 1.0  4.9  3.6  1.4  0.1
 1.0  4.4  3.2  1.3  0.2
 1.0  4.6  3.6  1.0  0.2

 1.0  6.4  3.2  5.3  2.3
 1.0  5.5  2.4  3.8  1.1
 1.0  6.0  2.9  4.5  1.5
 1.0  6.3  2.3  4.4  1.3
 1.0  4.9  3.1  1.5  0.2
 1.0  4.8  3.0  1.4  0.1
 1.0  6.7  3.1  4.4  1.4
 1.0  5.1  3.8  1.6  0.2
 1.0  7.3  2.9  6.3  1.8
 1.0  5.1  3.4  1.5  0.2
 1.0  6.9  3.1  5.1  2.3
 1.0  4.9  2.4  3.3  1.0
```

[ ]:
```
theta1 = A_train \ (2*y1_train .- 1)
```

[ ]:
```
5×1 Matrix{Float64}:
 -0.9947466749775586
  0.09153586094062746
```

```
 0.5698447506712916
-0.3371231542607084
-0.2950531091716144
```

[ ]:
```
theta2 = A_train \ (2*y2_train .- 1)
```

[ ]:
```
5×1 Matrix{Float64}:
  2.63355046469394
  0.08917974435999566
 -1.1278629884847786
  0.1602264688097815
 -0.5514903082277343
```

[ ]:
```
theta3 = A_train \ (2*y3_train .- 1)
```

[ ]:
```
5×1 Matrix{Float64}:
 -2.638803789716383
 -0.18071560530062303
  0.5580182378134874
  0.17689668545092702
  0.8465434173993485
```

[ ]:
```
y1_hat_train = A_train*theta1 .> 0
y2_hat_train = A_train*theta2 .> 0
y3_hat_train = A_train*theta3 .> 0
```

[ ]:
```
100×1 BitMatrix:
 0
 0
 1
 1
 0
 0
 0
 1
 0
 1
 0
 0
 0

 1
 0
 0
 0
 0
 0
```

```
0
0
1
0
1
0
```

[ ]:
```
A_test = [ones(50) X_test]
y1_hat_test = A_test*theta1 .> 0
y2_hat_test = A_test*theta2 .> 0
y3_hat_test = A_test*theta3 .> 0
```

[ ]:
```
50×1 BitMatrix:
 0
 1
 0
 0
 0
 0
 1
 1
 1
 0
 0
 0
 1

 0
 0
 0
 1
 0
 0
 1
 0
 0
 0
 0
 0
```

[ ]:
```
using Statistics
error_rate(y, yhat) = mean(y .!= yhat)
```

[ ]:
```
error_rate (generic function with 1 method)
```

[ ]:
```
error1_train = error_rate(y1_train, y1_hat_train)
error2_train = error_rate(y2_train, y2_hat_train)
```

```
error3_train = error_rate(y3_train, y3_hat_train)
@show error1_train
@show error2_train
@show error3_train
```

```
error1_train = 0.0
error2_train = 0.28
error3_train = 0.1
```

[ ]: 0.1

[ ]:
```
error1_test = error_rate(y1_test, y1_hat_test)
error2_test = error_rate(y2_test, y2_hat_test)
error3_test = error_rate(y3_test, y3_hat_test)
@show error1_test
@show error2_test
@show error3_test
```

```
error1_test = 0.0
error2_test = 0.24
error3_test = 0.02
```

[ ]: 0.02