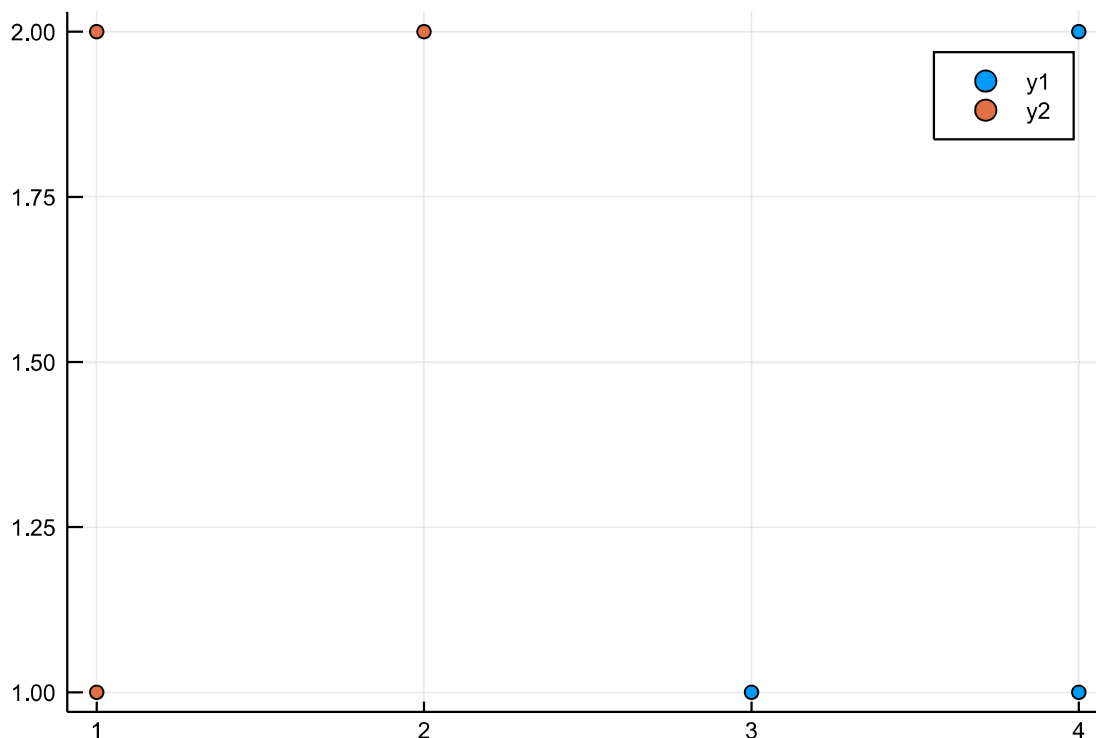In [8]:

```
# READ AND PLOT DATA
using Plots
data = readdlm("data.txt")
x = data[:,1:2]
y = data[:,3]
scatter(x[y.==0,1],x[y.==0,2])
scatter!(x[y.==1,1],x[y.==1,2])
```

Out[8]:

In [9]:

```
# CALCULATE W
using Knet
sigmoid(z) = 1./(1+exp.(-z))
predict(w,x) = sigmoid(w[1]*x .+ w[2])
J(w,x,y) = - sum(y .* log.(predict(w,x)) + (1 .- y) .* log.(1 .- predict(w,x)))
dJ = grad(J)

m,n = size(x)
alfa = 0.1
w = Any[ 0.1*randn(1,n), 0.1*randn(1,1) ]
for t=1:500
    #println(J(w,x',y'))
    w = w - alfa * dJ(w,x',y')
end
println(w)
#x = [1 1.125;]'; println(predict(w,x).>0.5)
#x = [4 1.125;]'; println(predict(w,x).>0.5)
```

Array{Float64,2}[[-4.2933 4.73519], [3.30719]]

In [10]:

```
# RESULT PLOTTING
yy = 0:0.1:3
xx = (0:0.1:5)'
m, n = length(yy), length(xx)
yy = mat(repmat(yy, 1, n))
xx = mat(repmat(xx, m, 1))
coorX = [xx[:]';yy[:]']
coorY = vec(predict(w,coorX).>0.5)
scatter(coorX'[coorY,1],coorX'[coorY,2])
scatter!(coorX'[~coorY,1],coorX'[~coorY,2])
scatter!(x[y.==0,1],x[y.==0,2],markersize=10)
scatter!(x[y.==1,1],x[y.==1,2],markersize=10)
```

Out[10]: