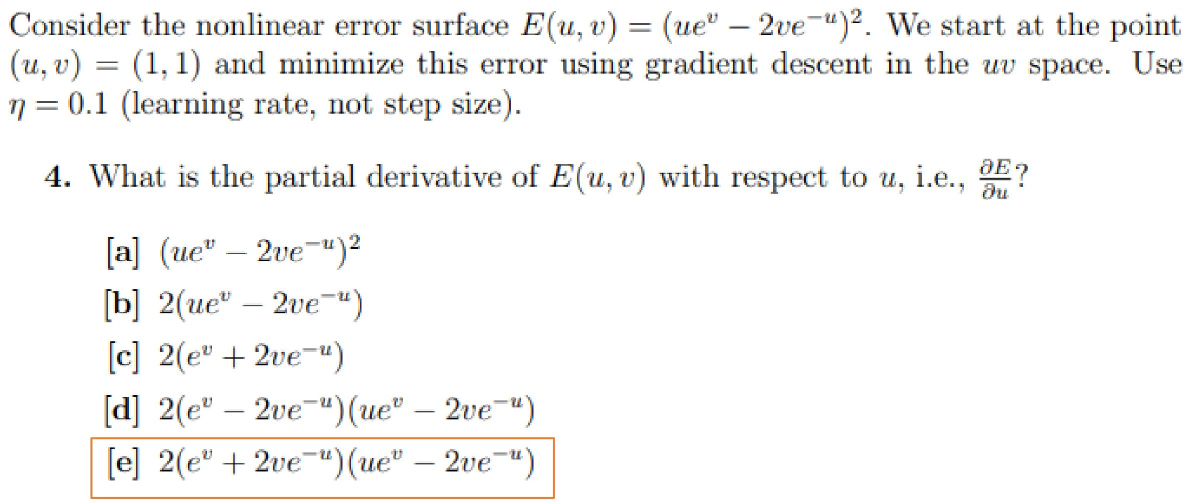
Introduction to Machine Learning Course - Homework 3-Harun Buyuktepe

## Part 1



Consider the nonlinear error surface $E(u, v) = (ue^v - 2ve^{-u})^2$. We start at the point $(u, v) = (1, 1)$ and minimize this error using gradient descent in the $uv$ space. Use $\eta = 0.1$ (learning rate, not step size).

**4.** What is the partial derivative of $E(u, v)$ with respect to $u$, i.e., $\frac{\partial E}{\partial u}$?

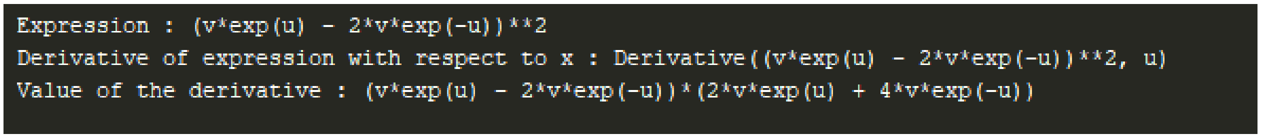
[a] $(ue^v - 2ve^{-u})^2$

[b] $2(ue^v - 2ve^{-u})$

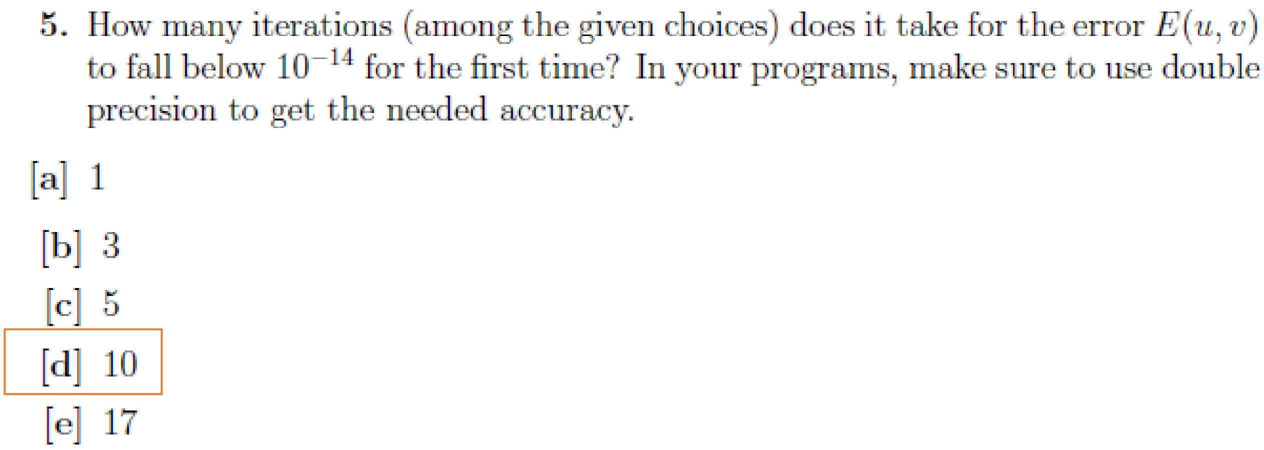
[c] $2(e^v + 2ve^{-u})$

[d] $2(e^v - 2ve^{-u})(ue^v - 2ve^{-u})$

[✓ [e] $2(e^v + 2ve^{-u})(ue^v - 2ve^{-u})$]



```
Expression : (v*exp(u) - 2*v*exp(-u))**2
Derivative of expression with respect to x : Derivative((v*exp(u) - 2*v*exp(-u))**2, u)
Value of the derivative : (v*exp(u) - 2*v*exp(-u))*(2*v*exp(u) + 4*v*exp(-u))
```



**5.** How many iterations (among the given choices) does it take for the error $E(u, v)$ to fall below $10^{-14}$ for the first time? In your programs, make sure to use double precision to get the needed accuracy.

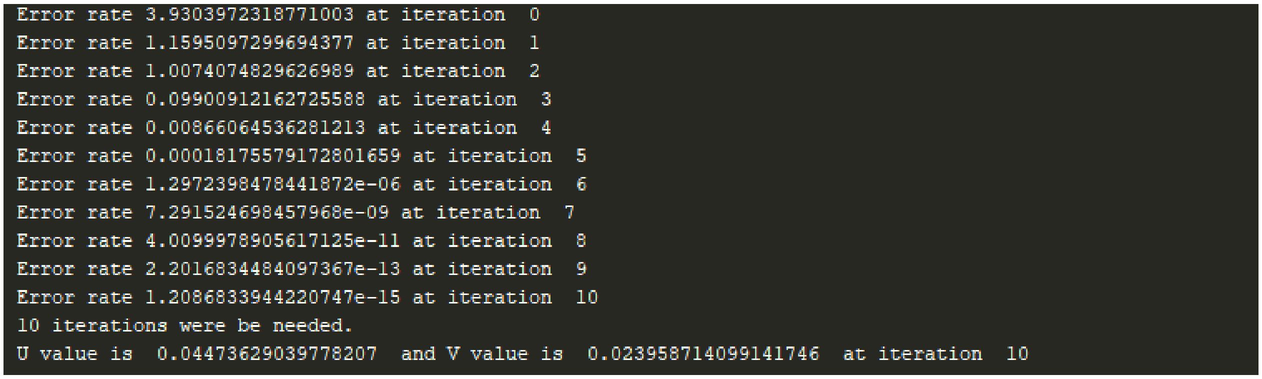
[a] 1

[b] 3

[c] 5

[✓ [d] 10]

[e] 17



```
Error rate 3.9303972318771003 at iteration  0
Error rate 1.1595097299694377 at iteration  1
Error rate 1.007407482962698 at iteration  2
Error rate 0.09900912162725588 at iteration  3
Error rate 0.008660645362812 13 at iteration  4
Error rate 0.00018175579172801659 at iteration  5
Error rate 1.2972398478441872e-06 at iteration  6
Error rate 7.291524698457968e-09 at iteration  7
Error rate 4.0099978905617125e-11 at iteration  8
Error rate 2.2016834484097367e-13 at iteration  9
Error rate 1.2086833944220747e-15 at iteration  10
10 iterations were be needed.
U value is  0.04473629039778207  and V value is  0.023958714099141746  at iteration  10
```

**6.** After running enough iterations such that the error has just dropped below $10^{-14}$, what are the closest values (in Euclidean distance) among the following choices to the final $(u, v)$ you got in Problem 5?

[a] $(1.000, 1.000)$

[b] $(0.713, 0.045)$

[c] $(0.016, 0.112)$

[d] $(-0.083, 0.029)$

[✓ [e] $(0.045, 0.024)$]

**7.** Now, we will compare the performance of "coordinate descent." In each iteration, we have two steps along the 2 coordinates. Step 1 is to move only along the $u$ coordinate to reduce the error (assume first-order approximation holds like in gradient descent), and step 2 is to reevaluate and move only along the $v$ coordinate to reduce the error (again, assume first-order approximation holds). Use the same learning rate of $\eta = 0.1$ as we did in gradient descent. What will the error $E(u, v)$ be closest to after 15 full iterations (30 steps)?

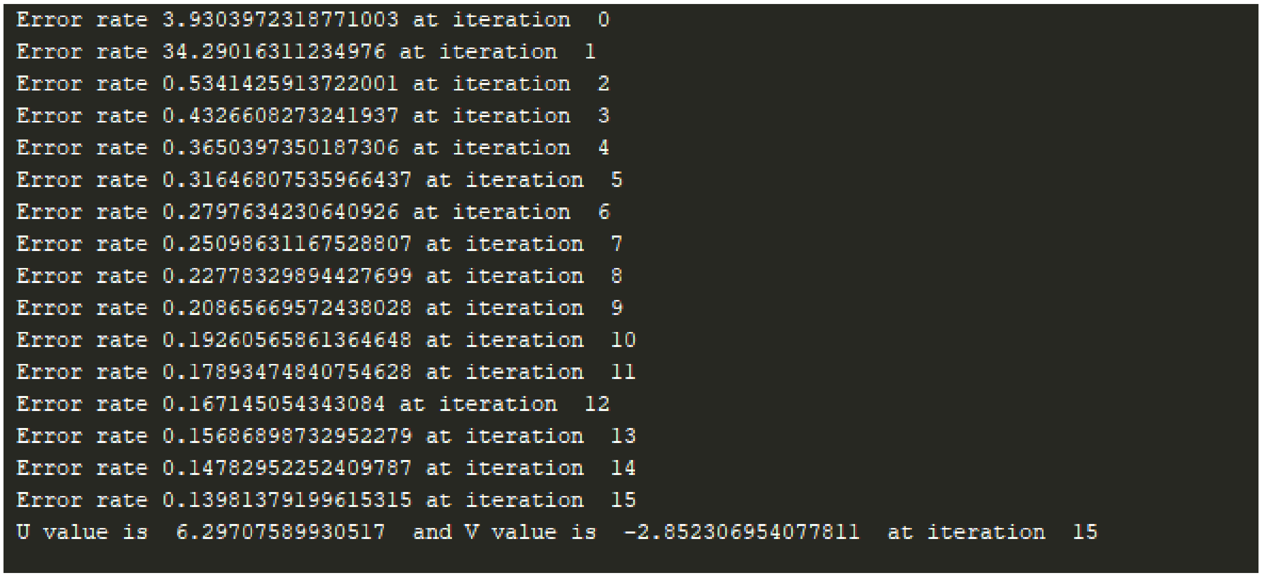
[✓ [a] $10^{-1}$]

[b] $10^{-7}$

[c] $10^{-14}$

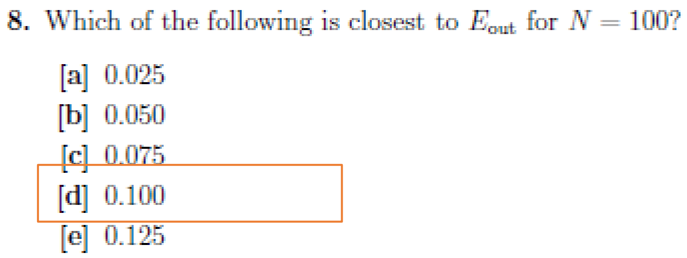
[d] $10^{-17}$

[e] $10^{-20}$



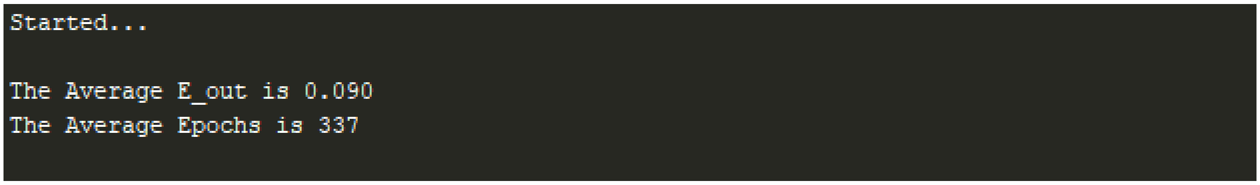
```
Error rate 3.9303972318771003 at iteration  0
Error rate 34.29016311234976 at iteration  1
Error rate 0.5341425913722001 at iteration  2
Error rate 0.4326608273241937 at iteration  3
Error rate 0.3650397350187306 at iteration  4
Error rate 0.31646807535966437 at iteration  5
Error rate 0.2797634230640926 at iteration  6
Error rate 0.25098631167528807 at iteration  7
Error rate 0.22778329894427699 at iteration  8
Error rate 0.20865669572438028 at iteration  9
Error rate 0.19260565861364648 at iteration  10
Error rate 0.17893474840754628 at iteration  11
Error rate 0.167145054343084 at iteration  12
Error rate 0.15686898732952279 at iteration  13
Error rate 0.14782952252409787 at iteration  14
Error rate 0.13981379199615315 at iteration  15
U value is  6.29707589930517  and V value is  -2.852306954077811  at iteration  15
```

## Part 2

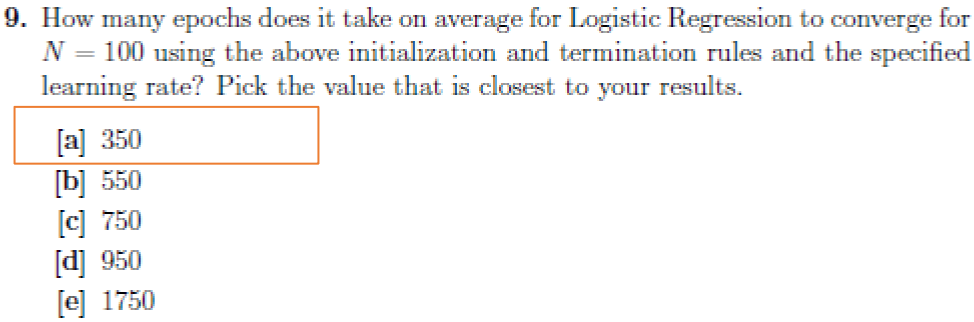


**8.** Which of the following is closest to $E_{\text{out}}$ for $N = 100$?

[a] 0.025
[b] 0.050
[c] 0.075
[✓ [d] 0.100]
[e] 0.125



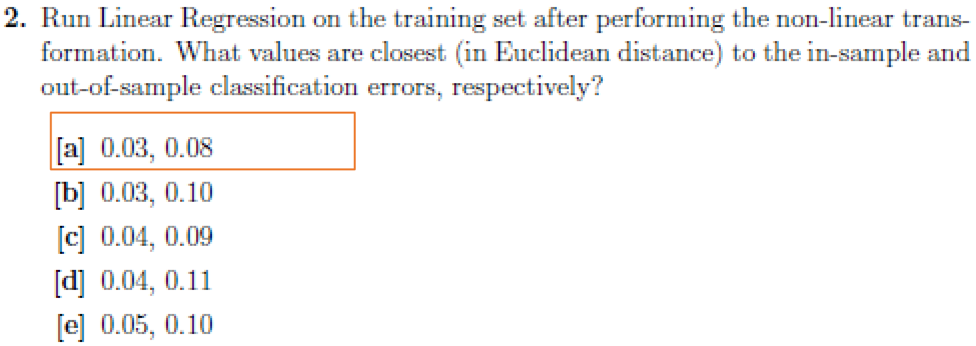
```
Started...

The Average E_out is 0.090
The Average Epochs is 337
```



**9.** How many epochs does it take on average for Logistic Regression to converge for $N = 100$ using the above initialization and termination rules and the specified learning rate? Pick the value that is closest to your results.

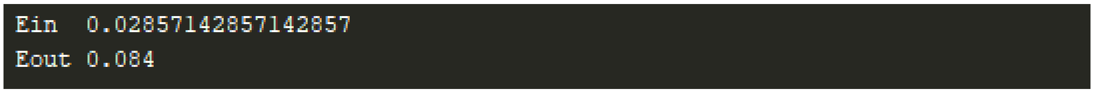
[✓ [a] 350]
[b] 550
[c] 750
[d] 950
[e] 1750

## Part 3



**2.** Run Linear Regression on the training set after performing the non-linear transformation. What values are closest (in Euclidean distance) to the in-sample and out-of-sample classification errors, respectively?

[✓ [a] 0.03, 0.08]
[b] 0.03, 0.10
[c] 0.04, 0.09
[d] 0.04, 0.11
[e] 0.05, 0.10



```
Ein  0.028571428571428571
Eout 0.084
```

**3.** Now add weight decay to Linear Regression, that is, add the term $\frac{\lambda}{N}\sum_{i=0}^{7} w_i^2$ to the squared in-sample error, using $\lambda = 10^k$. What are the closest values to the in-sample and out-of-sample classification errors, respectively, for $k = -3$? Recall that the solution for Linear Regression with Weight Decay was derived in class.

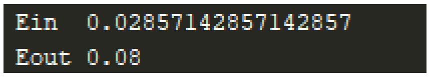
[a] 0.01, 0.02

[b] 0.02, 0.04

[c] 0.02, 0.06

[✓ [d] 0.03, 0.08]

[e] 0.03, 0.10



```
Ein  0.02857142857142857
Eout 0.08
```

**4.** Now, use $k = 3$. What are the closest values to the new in-sample and out-of-sample classification errors, respectively?

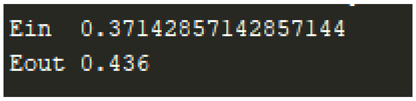
[a] 0.2, 0.2

[b] 0.2, 0.3

[c] 0.3, 0.3

[d] 0.3, 0.4

[✓ [e] 0.4, 0.4]



```
Ein  0.37142857142857144
Eout 0.436
```

**5.** What value of $k$, among the following choices, achieves the smallest out-of-sample classification error?

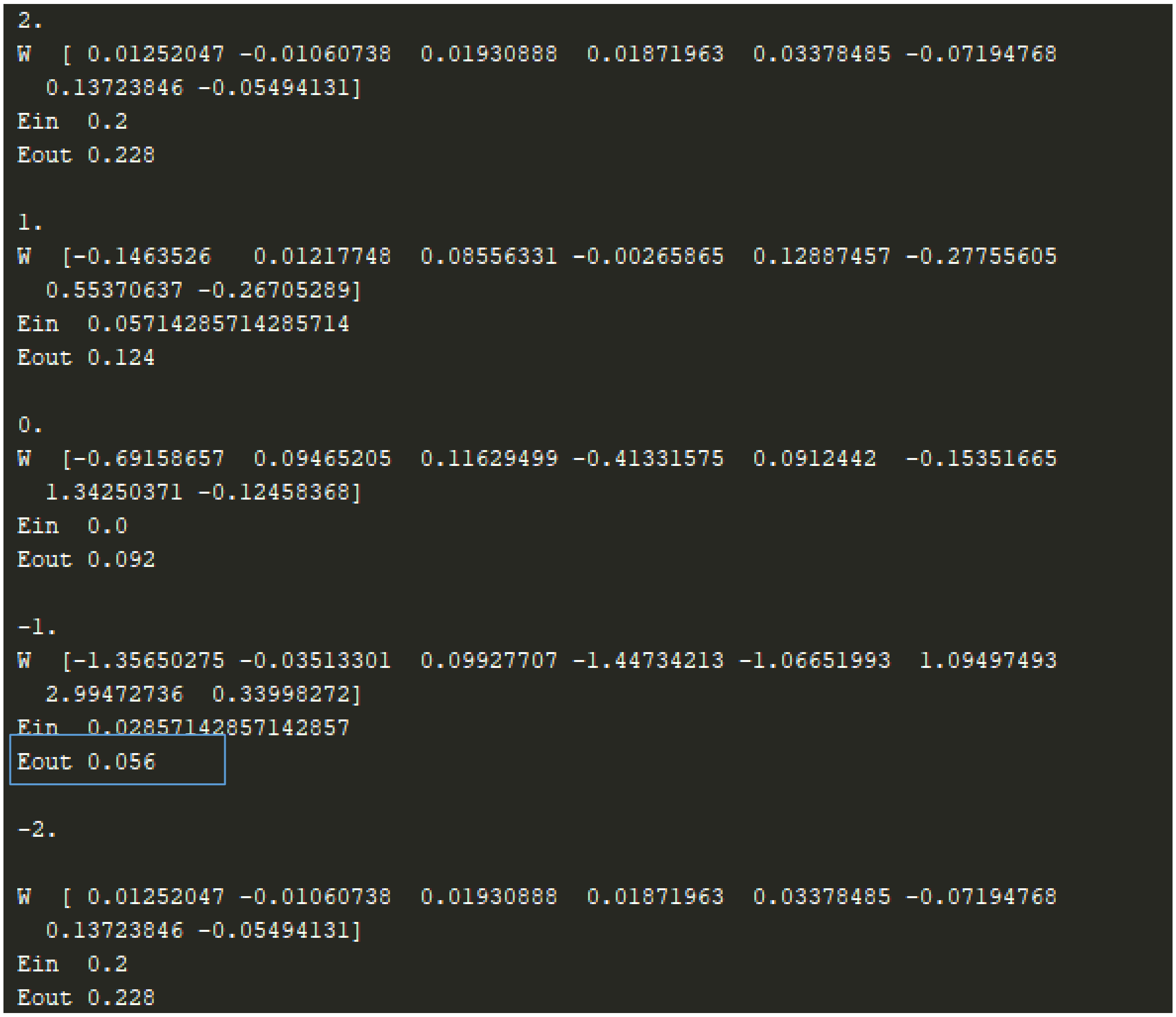
[a] 2

[b] 1

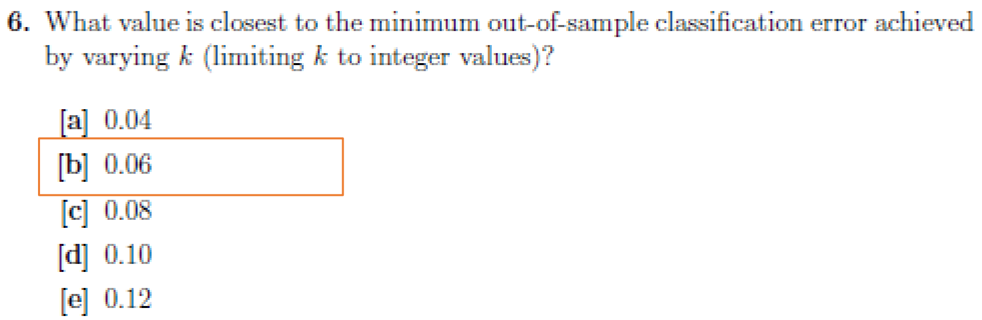
[c] 0

[✓ [d] −1]

[e] −2



```
2.
W  [ 0.01252047 -0.01060738  0.01930888  0.01871963  0.03378485 -0.07194768
  0.13723846 -0.05494131]
Ein  0.2
Eout 0.228

1.
W  [-0.1463526   0.01217748  0.08556331 -0.00265865  0.12887457 -0.27755605
  0.55370637 -0.26705289]
Ein  0.05714285714285714
Eout 0.124

0.
W  [-0.69158657  0.09465205  0.11629499 -0.41331575  0.0912442  -0.15351665
  1.34250371 -0.12458368]
Ein  0.0
Eout 0.092

-1.
W  [-1.35650275 -0.03513301  0.09927707 -1.44734213 -1.06651993  1.09497493
  2.99472736  0.33998272]
Ein  0.02857142857142857
Eout 0.056

-2.

W  [ 0.01252047 -0.01060738  0.01930888  0.01871963  0.03378485 -0.07194768
  0.13723846 -0.05494131]
Ein  0.2
Eout 0.228
```



**6.** What value is closest to the minimum out-of-sample classification error achieved by varying $k$ (limiting $k$ to integer values)?

[a] 0.04

[✓ [b] 0.06]

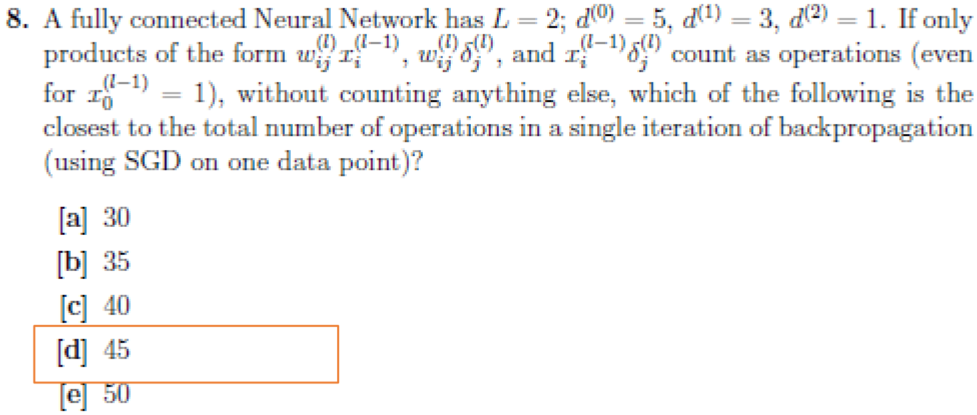
[c] 0.08

[d] 0.10

[e] 0.12

Respect problem 5's output.

## Part 4



8. A fully connected Neural Network has $L = 2$; $d^{(0)} = 5$, $d^{(1)} = 3$, $d^{(2)} = 1$. If only products of the form $w_{ij}^{(l)} x_i^{(l-1)}$, $w_{ij}^{(l)} \delta_j^{(l)}$, and $x_i^{(l-1)} \delta_j^{(l)}$ count as operations (even for $x_0^{(l-1)} = 1$), without counting anything else, which of the following is the closest to the total number of operations in a single iteration of backpropagation (using SGD on one data point)?

[a] 30

[b] 35

[c] 40

[✓ [d] 45]

[e] 50

$$L = 2, d^{(0)} = 5, d^{(1)} = 3, d^{(2)} = 1$$

$a)Forward\ Propagation$

$First\ Layer = 6 * 3 = 18\ w_{ij}^{(l)} * x_i^{(l-1)}$

$Second\ Layer = 4 * 1 = 4\ w_{ij}^{(l)} * x_i^{(l-1)}$

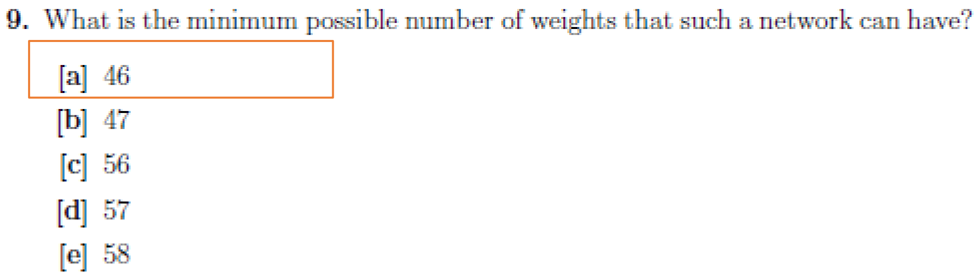
$b)Back\ Propagation$

$First\ Layer = 6 * 3 = 18\ \ x_i^{(l-1)} * \delta_j^{(l)}$

$Second\ Layer = 4 * 1 = 4\ \ x_i^{(l-1)} * \delta_j^{(l)}$

$= 3 * 1 = 3\ w_{ij}^{(l)} * x_i^{(l-1)}$

$47\ is\ the\ closest\ number\ of operations$



9. What is the minimum possible number of weights that such a network can have?

[✓ [a] 46]

[b] 47

[c] 56

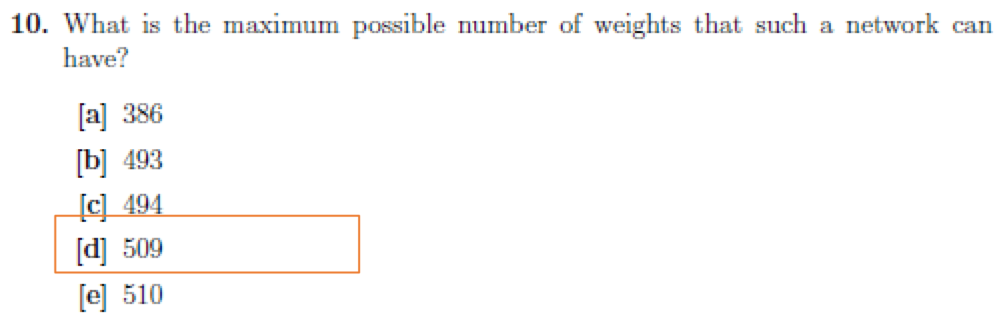
[d] 57

[e] 58

$10\ input - 1\ output$

$36\ hidden\ phase - 2\ nodes\ per\ layer\ So\ 18$

$36 + 10 = 46$



**10.** What is the maximum possible number of weights that such a network can have?

[a] 386

[b] 493

[c] 494

[✓ [d] 509]

[e] 510

$Assume, 23\ nodes\ on\ 1st\ layer, 13\ nodes\ on\ the\ 2nd\ layer$

$$23 \ast 12 + 10 \ast 22 + 13 = 509$$