**training backpropagation using pso-Copy1**

# Forward propagation

def forward\_prop(params):

"""Forward propagation as objective function

This computes for the forward propagation of the neural network, as

well as the loss. It receives a set of parameters that must be

rolled-back into the corresponding weights and biases.

Inputs

------

params: np.ndarray

The dimensions should include an unrolled version of the

weights and biases.

Returns

-------

float

The computed negative log-likelihood loss given the parameters

"""

# Neural network architecture

n\_inputs = 4

n\_hidden = 20

n\_classes = 3

# Roll-back the weights and biases

W1 = params[0:80].reshape((n\_inputs,n\_hidden))

b1 = params[80:100].reshape((n\_hidden,))

W2 = params[100:160].reshape((n\_hidden,n\_classes))

b2 = params[160:163].reshape((n\_classes,))

# Perform forward propagation

z1 = X.dot(W1) + b1 # Pre-activation in Layer 1

a1 = np.tanh(z1) # Activation in Layer 1

z2 = a1.dot(W2) + b2 # Pre-activation in Layer 2

logits = z2 # Logits for Layer 2

# Compute for the softmax of the logits

exp\_scores = np.exp(logits) # np.exp means exponential function

probs = exp\_scores / np.sum(exp\_scores, axis=1, keepdims=True) #axis=1 means columns

# Compute for the negative log likelihood

N = 150 # Number of samples

corect\_logprobs = -np.log(probs[range(N), y])

loss = np.sum(corect\_logprobs) / N

print(X.shape)

print(W1.shape)

print(b1.shape)

print(z1.shape)

print(a1.shape)

print(W2.shape)

print(b2.shape)

print(z2.shape)

print((X, " "))

print((W1, " "))

print((b1, " "))

print((a1, " "))

print((z1, " "))

print((W2, " "))

print((b2, " "))

print((z2, " "))

return loss

#########################################################################

X = data.data

y = data.target

print(X.shape) (150, 4)

print(y.shape) (150,)

print(X.shape) (150, 4)

print((X, " "))

(array([[5.1, 3.5, 1.4, 0.2],

[4.9, 3. , 1.4, 0.2],

[4.7, 3.2, 1.3, 0.2],

[4.6, 3.1, 1.5, 0.2],

[5. , 3.6, 1.4, 0.2],

[5.4, 3.9, 1.7, 0.4],

[4.6, 3.4, 1.4, 0.3],

[5. , 3.4, 1.5, 0.2],

[4.4, 2.9, 1.4, 0.2],

[4.9, 3.1, 1.5, 0.1],

[5.4, 3.7, 1.5, 0.2],

[4.8, 3.4, 1.6, 0.2],

[4.8, 3. , 1.4, 0.1],

[4.3, 3. , 1.1, 0.1],

[5.8, 4. , 1.2, 0.2],

[5.7, 4.4, 1.5, 0.4],

[5.4, 3.9, 1.3, 0.4],

[5.1, 3.5, 1.4, 0.3],

[5.7, 3.8, 1.7, 0.3],

[5.1, 3.8, 1.5, 0.3],

[5.4, 3.4, 1.7, 0.2],

[5.1, 3.7, 1.5, 0.4],

[4.6, 3.6, 1. , 0.2],

[5.1, 3.3, 1.7, 0.5],

[4.8, 3.4, 1.9, 0.2],

[5. , 3. , 1.6, 0.2],

[5. , 3.4, 1.6, 0.4],

[5.2, 3.5, 1.5, 0.2],

[5.2, 3.4, 1.4, 0.2],

[4.7, 3.2, 1.6, 0.2],

[4.8, 3.1, 1.6, 0.2],

[5.4, 3.4, 1.5, 0.4],

[5.2, 4.1, 1.5, 0.1],

[5.5, 4.2, 1.4, 0.2],

[4.9, 3.1, 1.5, 0.1],

[5. , 3.2, 1.2, 0.2],

[5.5, 3.5, 1.3, 0.2],

[4.9, 3.1, 1.5, 0.1],

[4.4, 3. , 1.3, 0.2],

[5.1, 3.4, 1.5, 0.2],

[5. , 3.5, 1.3, 0.3],

[4.5, 2.3, 1.3, 0.3],

[4.4, 3.2, 1.3, 0.2],

[5. , 3.5, 1.6, 0.6],

[5.1, 3.8, 1.9, 0.4],

[4.8, 3. , 1.4, 0.3],

[5.1, 3.8, 1.6, 0.2],

[4.6, 3.2, 1.4, 0.2],

[5.3, 3.7, 1.5, 0.2],

[5. , 3.3, 1.4, 0.2],

[7. , 3.2, 4.7, 1.4],

[6.4, 3.2, 4.5, 1.5],

[6.9, 3.1, 4.9, 1.5],

[5.5, 2.3, 4. , 1.3],

[6.5, 2.8, 4.6, 1.5],

[5.7, 2.8, 4.5, 1.3],

[6.3, 3.3, 4.7, 1.6],

[4.9, 2.4, 3.3, 1. ],

[6.6, 2.9, 4.6, 1.3],

[5.2, 2.7, 3.9, 1.4],

[5. , 2. , 3.5, 1. ],

[5.9, 3. , 4.2, 1.5],

[6. , 2.2, 4. , 1. ],

[6.1, 2.9, 4.7, 1.4],

[5.6, 2.9, 3.6, 1.3],

[6.7, 3.1, 4.4, 1.4],

[5.6, 3. , 4.5, 1.5],

[5.8, 2.7, 4.1, 1. ],

[6.2, 2.2, 4.5, 1.5],

[5.6, 2.5, 3.9, 1.1],

[5.9, 3.2, 4.8, 1.8],

[6.1, 2.8, 4. , 1.3],

[6.3, 2.5, 4.9, 1.5],

[6.1, 2.8, 4.7, 1.2],

[6.4, 2.9, 4.3, 1.3],

[6.6, 3. , 4.4, 1.4],

[6.8, 2.8, 4.8, 1.4],

[6.7, 3. , 5. , 1.7],

[6. , 2.9, 4.5, 1.5],

[5.7, 2.6, 3.5, 1. ],

[5.5, 2.4, 3.8, 1.1],

[5.5, 2.4, 3.7, 1. ],

[5.8, 2.7, 3.9, 1.2],

[6. , 2.7, 5.1, 1.6],

[5.4, 3. , 4.5, 1.5],

[6. , 3.4, 4.5, 1.6],

[6.7, 3.1, 4.7, 1.5],

[6.3, 2.3, 4.4, 1.3],

[5.6, 3. , 4.1, 1.3],

[5.5, 2.5, 4. , 1.3],

[5.5, 2.6, 4.4, 1.2],

[6.1, 3. , 4.6, 1.4],

[5.8, 2.6, 4. , 1.2],

[5. , 2.3, 3.3, 1. ],

[5.6, 2.7, 4.2, 1.3],

[5.7, 3. , 4.2, 1.2],

[5.7, 2.9, 4.2, 1.3],

[6.2, 2.9, 4.3, 1.3],

[5.1, 2.5, 3. , 1.1],

[5.7, 2.8, 4.1, 1.3],

[6.3, 3.3, 6. , 2.5],

[5.8, 2.7, 5.1, 1.9],

[7.1, 3. , 5.9, 2.1],

[6.3, 2.9, 5.6, 1.8],

[6.5, 3. , 5.8, 2.2],

[7.6, 3. , 6.6, 2.1],

[4.9, 2.5, 4.5, 1.7],

[7.3, 2.9, 6.3, 1.8],

[6.7, 2.5, 5.8, 1.8],

[7.2, 3.6, 6.1, 2.5],

[6.5, 3.2, 5.1, 2. ],

[6.4, 2.7, 5.3, 1.9],

[6.8, 3. , 5.5, 2.1],

[5.7, 2.5, 5. , 2. ],

[5.8, 2.8, 5.1, 2.4],

[6.4, 3.2, 5.3, 2.3],

[6.5, 3. , 5.5, 1.8],

[7.7, 3.8, 6.7, 2.2],

[7.7, 2.6, 6.9, 2.3],

[6. , 2.2, 5. , 1.5],

[6.9, 3.2, 5.7, 2.3],

[5.6, 2.8, 4.9, 2. ],

[7.7, 2.8, 6.7, 2. ],

[6.3, 2.7, 4.9, 1.8],

[6.7, 3.3, 5.7, 2.1],

[7.2, 3.2, 6. , 1.8],

[6.2, 2.8, 4.8, 1.8],

[6.1, 3. , 4.9, 1.8],

[6.4, 2.8, 5.6, 2.1],

[7.2, 3. , 5.8, 1.6],

[7.4, 2.8, 6.1, 1.9],

[7.9, 3.8, 6.4, 2. ],

[6.4, 2.8, 5.6, 2.2],

[6.3, 2.8, 5.1, 1.5],

[6.1, 2.6, 5.6, 1.4],

[7.7, 3. , 6.1, 2.3],

[6.3, 3.4, 5.6, 2.4],

[6.4, 3.1, 5.5, 1.8],

[6. , 3. , 4.8, 1.8],

[6.9, 3.1, 5.4, 2.1],

[6.7, 3.1, 5.6, 2.4],

[6.9, 3.1, 5.1, 2.3],

[5.8, 2.7, 5.1, 1.9],

[6.8, 3.2, 5.9, 2.3],

[6.7, 3.3, 5.7, 2.5],

[6.7, 3. , 5.2, 2.3],

[6.3, 2.5, 5. , 1.9],

[6.5, 3. , 5.2, 2. ],

[6.2, 3.4, 5.4, 2.3],

[5.9, 3. , 5.1, 1.8]]), ' ')

z1 = X.dot(W1) + b1

print(W1.shape) (4, 20) [0:80]

print((W1, " "))

(array([[0.69703159, 0.09576439, 0.44383653, 0.96359829, 0.02636632,

0.07372328, 0.82734034, 0.72268481, 0.43289626, 0.85653455,

0.59429846, 0.88190552, 0.59235855, 0.06805411, 0.60225421,

0.11703041, 0.49008022, 0.43526799, 0.04859478, 0.93300345],

[0.32445939, 0.71907542, 0.53107874, 0.61553875, 0.81016251,

0.65983495, 0.26297115, 0.8512136 , 0.8710699 , 0.82631511,

0.65860164, 0.62696638, 0.11552657, 0.5128526 , 0.75101402,

0.53833173, 0.90301555, 0.4387422 , 0.28066631, 0.50717405],

[0.16577625, 0.06748108, 0.91508073, 0.36739907, 0.5796949 ,

0.20796088, 0.71958829, 0.13777653, 0.60913279, 0.47722185,

0.14802514, 0.52610868, 0.04060913, 0.06766715, 0.24606694,

0.99594544, 0.64844235, 0.99049594, 0.19680018, 0.73019522],

[0.73213845, 0.33646159, 0.0430223 , 0.18419285, 0.17523724,

0.52220322, 0.35004571, 0.69874152, 0.13589839, 0.72982751,

0.57336553, 0.62432825, 0.94912662, 0.6259226 , 0.1043612 ,

0.97534379, 0.53348318, 0.92021259, 0.4325179 , 0.56006283]]), ' ‘)

print(b1.shape) (20,) [80:100]

print((b1, " "))

(array([0.91311914, 0.72654706, 0.08708389, 0.56837573, 0.1965857 ,

0.31416947, 0.13062386, 0.08207791, 0.25190015, 0.89855585,

0.60802452, 0.40491978, 0.44705823, 0.76755838, 0.93752138,

0.25567155, 0.67670553, 0.40227706, 0.00467584, 0.47101572]), ' ')

print(z1.shape) (150, 20)

print((z1, " "))

(array([[ 6.59463657, 5.01524143, 6.56065346, ..., 2.69041009,

6.55716276, 8.19048888],

[ 6.07274245, 4.54911898, 6.17524159, ..., 2.57043625,

6.26440124, 7.61970536],

[ 6.08405539, 4.65473156, 6.10347619, ..., 2.52367297,

6.11635319, 7.59414569],

...,

[10.07555633, 5.36315609, 9.02393099, ..., 4.89793386,

8.33342336, 12.62730585],

[10.56867757, 5.65416733, 9.1564022 , ..., 5.0686341 ,

8.21289839, 13.10928239],

[ 9.54836991, 5.17195677, 8.54170427, ..., 4.67956609,

7.7348238 , 12.09735996]]), ' ')

a1 = np.tanh(z1) # Activation in Layer 1

print(a1.shape) (150, 20)

print((a1, " "))

(array([[0.99998726, 0.99917011, 0.99996654, ..., 0.99997885, 0.92119531,

0.99999983],

[0.99997672, 0.99823094, 0.99993204, ..., 0.99996096, 0.8950609 ,

0.99999959],

[0.99997206, 0.99860233, 0.99992119, ..., 0.99995248, 0.90025199,

0.99999944],

...,

[0.99999995, 0.99976761, 0.99999999, ..., 1. , 0.99553267,

1. ],

[0.99999996, 0.99988985, 0.99999999, ..., 1. , 0.99737755,

1. ],

[0.99999984, 0.99969771, 0.99999997, ..., 1. , 0.99414803,

1. ]]), ' ')

z2 = a1.dot(W2) + b2 # Pre-activation in Layer 2

print(W2.shape) (20, 3) [100:160]

print((W2, " "))

(array([[0.71827051, 0.75912465, 0.78250809],

[0.67563777, 0.23389726, 0.37918579],

[0.9440568 , 0.8556592 , 0.5342471 ],

[0.68848822, 0.38150576, 0.36192862],

[0.27647883, 0.29571141, 0.97102891],

[0.0359247 , 0.69711724, 0.53778783],

[0.01217349, 0.89430731, 0.05424005],

[0.8632876 , 0.53113927, 0.69991055],

[0.23913838, 0.92811309, 0.64756441],

[0.84337225, 0.47216991, 0.9696701 ],

[0.75357756, 0.78676443, 0.19914582],

[0.18056629, 0.96101007, 0.25393367],

[0.48661745, 0.05000003, 0.74853264],

[0.04008546, 0.22370093, 0.84671098],

[0.1268119 , 0.87460361, 0.51543094],

[0.62882331, 0.96955363, 0.60255715],

[0.98518311, 0.75837017, 0.47828754],

[0.19261889, 0.28075122, 0.76003383],

[0.64697548, 0.21933127, 0.89165754],

[0.8181524 , 0.53356788, 0.55552901]]), ' ')

print(b2.shape) (3,) [160:163]

(array([0.54015616, 0.14186906, 0.67218656]), ' ')

print(z2.shape) (150, 3)

print((z2, " "))

(array([[10.69415342, 11.84530777, 12.45450814],

[10.69333662, 11.84434335, 12.45219509],

[10.69299904, 11.84391378, 12.45120529],

[10.69321362, 11.84422798, 12.45196144],

[10.69412327, 11.84527255, 12.45443482],

[10.69523808, 11.8467091 , 12.45800647],

[10.69352414, 11.84459178, 12.45284514],

[10.69413091, 11.84529869, 12.4544933 ],

[10.6921707 , 11.84301434, 12.44916835],

[10.69352852, 11.84455712, 12.4526924 ],

[10.69474908, 11.84605436, 12.45634006],

[10.69407408, 11.84525115, 12.45439732],

[10.6929701 , 11.84387384, 12.45107831],

[10.69056129, 11.84097009, 12.4445896 ],

[10.69484372, 11.84615006, 12.45656583],

[10.69541009, 11.84691865, 12.45853886],

[10.69479558, 11.84612387, 12.45653051],

[10.69428182, 11.84548626, 12.45495491],

[10.69529398, 11.84676922, 12.45814548],

[10.69465658, 11.84595841, 12.45612825],

[10.69481996, 11.84616802, 12.45662371],

[10.69469313, 11.84601905, 12.4562831 ],

[10.69244854, 11.8431993 , 12.44959751],

[10.69479131, 11.84618677, 12.45670745],

[10.69456537, 11.84589282, 12.45597498],

[10.69392093, 11.84508482, 12.45397426],

[10.69453617, 11.84584664, 12.4558592 ],

[10.69443137, 11.845662 , 12.45537219],

[10.69418082, 11.84534059, 12.45457773],

[10.693733 , 11.84485514, 12.45344688],

[10.69376173, 11.8448929 , 12.45352682],

[10.69475343, 11.8460918 , 12.45644432],

[10.69472769, 11.84601863, 12.45627038],

[10.6949755 , 11.84633757, 12.4570593 ],

[10.69352852, 11.84455712, 12.4526924 ],

[10.69325848, 11.84419054, 12.45181595],

[10.69440854, 11.84560493, 12.45520872],

[10.69352852, 11.84455712, 12.4526924 ],

[10.69201346, 11.84278601, 12.44864574],

[10.69424532, 11.84543642, 12.45482211],

[10.69397892, 11.84510529, 12.45403412],

[10.6908747 , 11.8416952 , 12.44628188],

[10.69237627, 11.84319528, 12.44959094],

[10.69480486, 11.84620114, 12.45675408],

[10.6952056 , 11.84668747, 12.45796233],

[10.69335116, 11.84439847, 12.45235448],

[10.69469524, 11.84600064, 12.45622713],

[10.69308942, 11.84404964, 12.45154129],

[10.6946651 , 11.84595019, 12.4560856 ],

[10.69383891, 11.84492926, 12.45359188],

[10.69633772, 11.84820702, 12.46192227],

[10.69631215, 11.84818152, 12.46185738],

[10.69633789, 11.84821211, 12.46193771],

[10.69596275, 11.84792541, 12.46125352],

[10.69628721, 11.84817167, 12.4618398 ],

[10.69620391, 11.8481003 , 12.46166404],

[10.69632556, 11.8481965 , 12.46189634],

[10.69555627, 11.84748425, 12.46012763],

[10.69629392, 11.84817159, 12.4618361 ],

[10.69604956, 11.84795479, 12.46129967],

[10.69532099, 11.84742189, 12.46005463],

[10.69624631, 11.84812066, 12.46170507],

[10.69598786, 11.84793622, 12.46127267],

[10.696273 , 11.84815897, 12.46180819],

[10.69611311, 11.84797091, 12.46131719],

[10.69631022, 11.84817847, 12.46184901],

[10.69624131, 11.84812692, 12.46172645],

[10.69612735, 11.84801392, 12.46143939],

[10.69613919, 11.84808959, 12.46166373],

[10.69601882, 11.84793035, 12.46124026],

[10.69631218, 11.84819095, 12.4618861 ],

[10.69620417, 11.84807888, 12.46159895],

[10.69624503, 11.84815842, 12.46182047],

[10.69624856, 11.84813947, 12.46176086],

[10.69626578, 11.84813972, 12.46175316],

[10.6962979 , 11.84816947, 12.46182775],

[10.69630702, 11.84818999, 12.46188567],

[10.69633335, 11.84821299, 12.46194271],

[10.69626056, 11.84814527, 12.46177276],

[10.69598488, 11.8478521 , 12.46102068],

[10.69593813, 11.8478673 , 12.46108961],

[10.69589791, 11.84781851, 12.46096187],

[10.69612654, 11.84800886, 12.46142495],

[10.69626686, 11.84817341, 12.46185533],

[10.6962222 , 11.8481119 , 12.4616902 ],

[10.69631155, 11.8481787 , 12.46184906],

[10.69632524, 11.84819839, 12.46190221],

[10.69615615, 11.84808511, 12.46164148],

[10.69619317, 11.84806553, 12.46156381],

[10.69604338, 11.84796728, 12.46134035],

[10.69611085, 11.84803288, 12.46150541],

[10.69627921, 11.84815855, 12.46180381],

[10.69611401, 11.8480108 , 12.46143697],

[10.69553004, 11.84747972, 12.46012673],

[10.69614306, 11.84804357, 12.46152207],

[10.6962046 , 11.84807712, 12.46159321],

[10.69619681, 11.84807735, 12.46159789],

[10.69625121, 11.84812719, 12.46172231],

[10.69563307, 11.84749366, 12.46012061],

[10.69616836, 11.84805336, 12.46153924],

[10.69636656, 11.84824649, 12.46202832],

[10.69626566, 11.84817679, 12.4618663 ],

[10.69636448, 11.84824525, 12.46202553],

[10.69632737, 11.84821892, 12.46196382],

[10.69635173, 11.84823713, 12.46200719],

[10.69637639, 11.84825513, 12.46204971],

[10.69602979, 11.84800897, 12.46147347],

[10.69636428, 11.84824689, 12.46203064],

[10.69631054, 11.84821704, 12.46196645],

[10.69638411, 11.84825729, 12.46205244],

[10.69634743, 11.84822468, 12.46197134],

[10.6963095 , 11.84820769, 12.4619385 ],

[10.69635373, 11.84823562, 12.46200155],

[10.69622111, 11.84815451, 12.4618208 ],

[10.69629923, 11.84820111, 12.46192367],

[10.69635388, 11.84823319, 12.46199407],

[10.69633985, 11.84822461, 12.46197492],

[10.69638961, 11.84826168, 12.46206309],

[10.69636718, 11.848253 , 12.4620478 ],

[10.69614325, 11.84811164, 12.46172885],

[10.69636844, 11.84824609, 12.46202614],

[10.69626193, 11.84816862, 12.46184328],

[10.69637142, 11.84825332, 12.46204668],

[10.69628859, 11.84818608, 12.4618831 ],

[10.69636613, 11.84824314, 12.46201829],

[10.69637052, 11.84824749, 12.46202934],

[10.69629059, 11.84818302, 12.46187279],

[10.69630873, 11.8481939 , 12.46189685],

[10.6963303 , 11.84822405, 12.46197801],

[10.69635904, 11.84823912, 12.46200954],

[10.69636129, 11.84824509, 12.46202662],

[10.69638891, 11.84826046, 12.46205971],

[10.69633253, 11.84822597, 12.46198277],

[10.69629521, 11.84818864, 12.46188753],

[10.69626786, 11.84818447, 12.4618885 ],

[10.69637626, 11.84825388, 12.46204596],

[10.69636491, 11.84824213, 12.4620158 ],

[10.69634277, 11.84822543, 12.46197594],

[10.6963 , 11.84818543, 12.46187542],

[10.69635903, 11.84823756, 12.46200482],

[10.6963615 , 11.84824194, 12.46201695],

[10.69635784, 11.84823527, 12.46199844],

[10.69626566, 11.84817679, 12.4618663 ],

[10.69636864, 11.84824728, 12.46202967],

[10.69637046, 11.84824751, 12.46202946],

[10.69635019, 11.84823168, 12.46199133],

[10.69626889, 11.84818169, 12.46187959],

[10.69633843, 11.84822162, 12.46196653],

[10.69635905, 11.84823602, 12.46200011],

[10.69630493, 11.84819486, 12.46190168]]), ' ')