**Report**

**Dyuti Ballav Paul**

**IIT(ISM) Dhanbad**

* **Linear Regression**
* Hyperparameters - In machine learning, a **hyperparameter** is a parameter whose value is used to control the learning process. By contrast, the values of other parameters (typically node weights) are derived via training.
* Alpha and iterations
* Overflow: Due to larger value of learning rate
* Reduce overflow by giving less learning rate
* α = 10e-6 overflow, diverging
* α = 10e-7 diverging Iteration no. 200: cost inf
* Iteration no. 300: cost inf
* mean and sigma when used to rescale new input the it is giving an error

Variable(variable) mean: Any | Unbound

"mean" is possibly unboundPylance[reportUnboundVariable](https://github.com/microsoft/pyright/blob/main/docs/configuration.md#reportUnboundVariable)

* Manually finding the mean and standard deviation solved this
* After iter = 1000 and init w and b =0
* Initially no normalisaition was done due to which it is taking large number of iteraitons for the cost to saturate to a minimum value of 4.770e03.
* Very small steps: Due to un-normalised data, it requires alpha to be very small of the order 10e-08. For values of order > 10e-08, it leads to divergence of the cost.
* After normalisation, the data gets centered around zero, and with small values it we can have our alpha of the order < 0.1, due to which we can reach to saturation very quickly without divergence.
* Training

iterations: 100 | alpha: 0.1

-----------------------------------------------------------------------------

Iteration no. 0: cost 2.4952e+04

Iteration no. 10: cost 7.0072e+03

Iteration no. 20: cost 5.0411e+03

Iteration no. 30: cost 4.8030e+03

Iteration no. 40: cost 4.7739e+03

Iteration no. 50: cost 4.7703e+03

Iteration no. 60: cost 4.7698e+03

Iteration no. 70: cost 4.7698e+03

Iteration no. 80: cost 4.7698e+03

Iteration no. 90: cost 4.7698e+03

Iteration no. 90: cost 4.770e+03

Iteration no. 91: cost 4.770e+03

Iteration no. 92: cost 4.770e+03

Iteration no. 93: cost 4.770e+03

Iteration no. 94: cost 4.770e+03

Iteration no. 95: cost 4.770e+03

Iteration no. 96: cost 4.770e+03

Iteration no. 97: cost 4.770e+03

Iteration no. 98: cost 4.770e+03

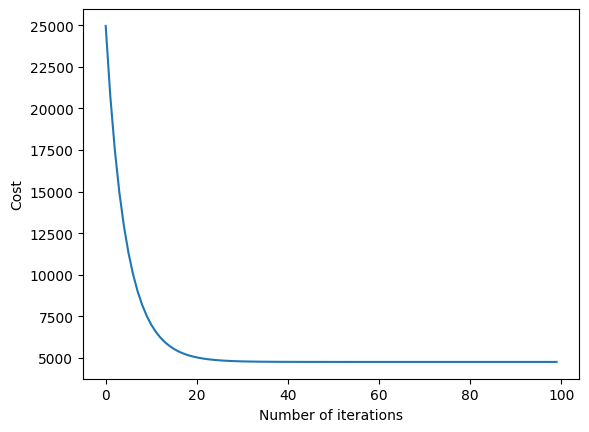
Iteration no. 99: cost 4.770e+03

b,w found by gradient descent: -0.37,[[72.99450661 66.24736555 97.43959016 1.7175908 17.36281484 24.81555849

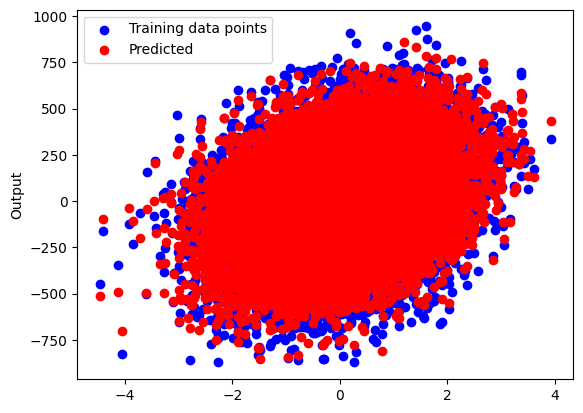
71.23162987 30.99471656 20.34997943 93.74647297 39.36806608 37.25428984

2.82788165 40.70113281 40.70113281 46.08605106 21.51549994 33.84497104

33.68303034 39.24713064]]



* To check with one feature, rough idea of fitting on the cross validation set
* **Runtime: 6 seconds**
* **R2 Score: 0.8162685626228243**



* **Polynomial Regression**
* **Training after applying regularisation**
* **Here, we can see the value of the w parameters is reduced**

**With degree = 5**

iterations: 5000 | alpha: 0.1 | deg: 5

-----------------------------------------------------------------------------------------

Iteration no. 0: cost 1.4409e+12

Iteration no. 500: cost 8.5247e+08

Iteration no. 1000: cost 6.4245e+07

Iteration no. 1500: cost 9.9815e+06

Iteration no. 2000: cost 2.8846e+06

Iteration no. 2500: cost 1.1570e+06

Iteration no. 3000: cost 5.5417e+05

Iteration no. 3500: cost 3.0846e+05

Iteration no. 4000: cost 1.9644e+05

Iteration no. 4500: cost 1.3873e+05

Iteration no. 4990: cost 1.055e+05

Iteration no. 4991: cost 1.054e+05

Iteration no. 4992: cost 1.054e+05

Iteration no. 4993: cost 1.053e+05

Iteration no. 4994: cost 1.052e+05

Iteration no. 4995: cost 1.052e+05

Iteration no. 4996: cost 1.051e+05

Iteration no. 4997: cost 1.051e+05

Iteration no. 4998: cost 1.050e+05

Iteration no. 4999: cost 1.050e+05

b,w found by gradient descent: 4757.59750808114, [[-1.73451597e+03 -5.27785727e+02 -8.87895741e+02 -2.03070238e+03

-7.85728256e+02 9.43933344e+02 4.68850758e+02 -1.19713426e+03

-5.79069671e+02 6.24919745e+02 4.83217534e+02 2.81149053e+02

4.15256958e+02 4.47726085e+01 -1.01393522e+02 1.35329357e+02

1.35277819e+02 4.07980775e+01 -1.66918636e+02 -1.45639113e+02

1.27357483e+02 -4.41004973e+03 -2.57648512e+03 1.92597280e+03

1.65725552e+03 5.31086198e+02 1.32854495e+03 1.59146834e+03

8.13746647e+02 3.37551198e+02 6.84582451e+02 4.41106889e+02

-1.52641585e+02 -2.63765763e+01 2.58819388e+03 4.97793849e+02

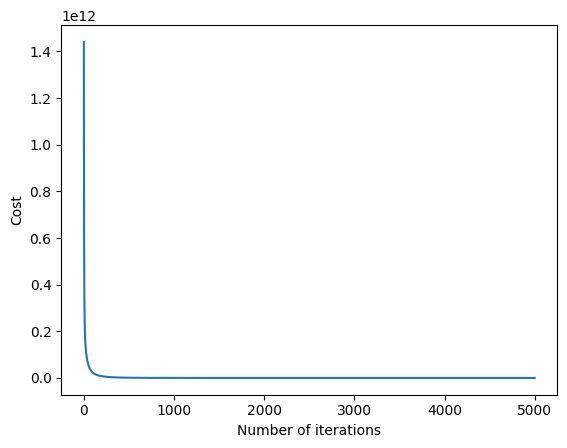
2.91179595e+03 5.37784493e+03 3.20266769e+03 4.85574733e+02

9.55950209e+02 7.83101239e+02 -6.88512742e+01 9.33095196e+02

1.15774027e+03 1.43749812e+03 3.01713081e+03 2.46843951e+03

-3.28973581e+02 4.09552131e+03 2.60918082e+04 -3.16544047e+02

5.19955103e+04 7.08278052e+04 2.56344311e+06]]

****

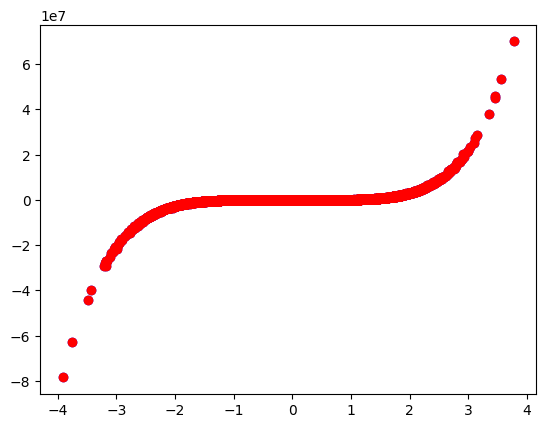
Deg = 5

Runtime: 12 seconds (approx.)

R2 Score: 0.9999999739983719

Error cv: 8.5019e+04

Just to check fitting with one of the feature



* Checking Other degrees to fing best degree

Deg = 2

Iteration no. 4999: cost 2.279e+12

R2 Score: -1.9584726448276193

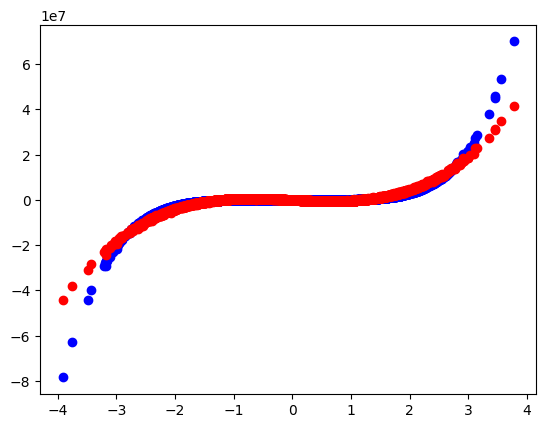
Error cv: 2.4404e+12

Deg = 3

Iteration no. 4999: cost 3.323e+11

R2 Score: 0.8736928731472738

Error cv: 3.5163e+11



Deg = 4

Iteration no. 4999: cost 3.304e+11

R2 Score: 0.8726822597241529

Error cv: 3.5466e+11

Deg = 6

Iteration no. 4999: cost 7.597e+07

R2 Score: 0.9999723091873058

Error cv: 9.0494e+07

Deg = 7

Iteration no. 4999: cost 4.474e+09

R2 Score: 0.9984448153041755

Error cv: 5.0958e+09

Deg = 8

Iteration no. 4999: cost 3.072e+09

R2 Score: 0.9987241759592421

Error cv: 4.1758e+09

Deg = 9

Iteration no. 4999: cost 2.269e+09

R2 Score: 0.9988932311188778

Error cv: 3.6357e+09

Deg = 10

Iteration no. 4999: cost 1.826e+09

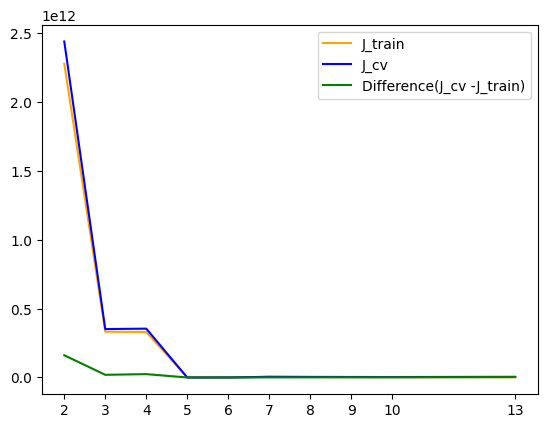
R2 Score: 0.9990085579067658

Error cv: 3.2550e+09

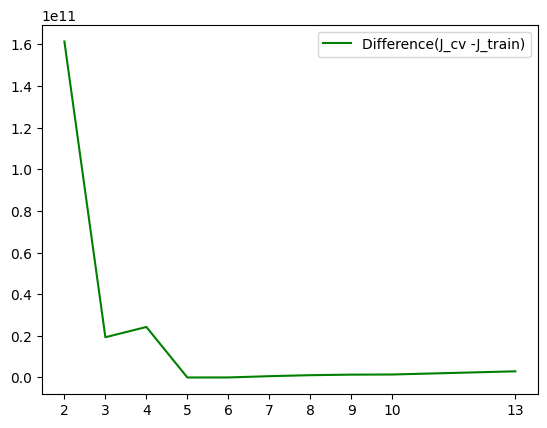
* Iterations = 5000 alpha = .1 (deg = 1 to 6) then diverge for higher

degrees for .1 so new alpha choosen alpha = .01(deg = 6 to 13)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Degree | J\_error | J\_cv | Difference  (J\_cv – J\_train) | R2 Score |
| 2 | 2.279e+12 | 2.4404e+12 | 1.6140e+11 | -1.9584726448276193 |
| 3 | 3.323e+11 | 3.5163e+11 | 1.9330e+10 | 0.8736928731472738 |
| 4 | 3.304e+11 | 3.5466e+11 | 2.4260e+10 | 0.8726822597241529 |
| 5 | 1.050e+05 | 8.5019e+04 | -1.9981e+04 (min magnitude of loss on dev set) | 0.9999999739983719 |
| 6 | 7.597e+07 | 9.0494e+07 | 1.4524e+07 | 0.9999723091873058 |
| 7 | 4.474e+09 | 5.0958e+09 | 6.2180e+08 | 0.9984448153041755 |
| 8 | 3.072e+09 | 4.1758e+09 | 1.1038e+09 | 0.9987241759592421 |
| 9 | 2.269e+09 | 3.6357e+09 | 1.3667e+09 | 0.9988932311188778 |
| 10 | 1.826e+09 | 3.2550e+09 | 1.4290e+09 | 0.9990085579067658 |
| 13 | 1.812e+09 | 4.7713e+09 | 2.9593e+09 | 0.9985548173582577 |



* Difference minimum for degree = 5



**Logistic Regression:**

* **Tried:** sigmoid as argument to cal\_grad\_lgstc() function but failed
* **Reason:** Error will occur while the gradient descent will run because this function takes cal\_gard\_lgstc() as argument where sigmoid is an undefined parameter
* So, better use mathematical form of the sigmoid in the cal\_grad\_lgstc()
* # Previously I used here -1 but the simplified loss function was for y = 1 or 0
* # for y = -1 or 1 think for another loss
* Training

iterations: 700 | alpha: 1e-06

-----------------------------------------------------------------------------------------

Iteration no. 0: cost 3.774e-01

Iteration no. 70: cost 1.560e-01

Iteration no. 140: cost 1.372e-01

Iteration no. 210: cost 1.307e-01

Iteration no. 280: cost 1.274e-01

Iteration no. 350: cost 1.252e-01

Iteration no. 420: cost 1.236e-01

Iteration no. 490: cost 1.224e-01

Iteration no. 560: cost 1.214e-01

Iteration no. 630: cost 1.206e-01

-----------------------------------------------------------------------------------------

Iteration no. 0: cost 3.096e-01

Iteration no. 70: cost 8.861e-02

Iteration no. 140: cost 6.948e-02

Iteration no. 210: cost 6.105e-02

Iteration no. 280: cost 5.609e-02

Iteration no. 350: cost 5.273e-02

Iteration no. 420: cost 5.026e-02

Iteration no. 490: cost 4.834e-02

Iteration no. 560: cost 4.678e-02

Iteration no. 630: cost 4.548e-02

-----------------------------------------------------------------------------------------

Iteration no. 0: cost 4.144e-01

Iteration no. 70: cost 2.161e-01

Iteration no. 140: cost 1.984e-01

Iteration no. 210: cost 1.900e-01

Iteration no. 280: cost 1.843e-01

Iteration no. 350: cost 1.801e-01

Iteration no. 420: cost 1.768e-01

Iteration no. 490: cost 1.741e-01

Iteration no. 560: cost 1.718e-01

Iteration no. 630: cost 1.699e-01

-----------------------------------------------------------------------------------------

Iteration no. 0: cost 3.501e-01

Iteration no. 70: cost 1.743e-01

Iteration no. 140: cost 1.468e-01

Iteration no. 210: cost 1.338e-01

Iteration no. 280: cost 1.260e-01

Iteration no. 350: cost 1.207e-01

Iteration no. 420: cost 1.170e-01

Iteration no. 490: cost 1.141e-01

Iteration no. 560: cost 1.118e-01

Iteration no. 630: cost 1.100e-01

-----------------------------------------------------------------------------------------

Iteration no. 0: cost 4.337e-01

Iteration no. 70: cost 2.455e-01

Iteration no. 140: cost 2.195e-01

Iteration no. 210: cost 2.055e-01

Iteration no. 280: cost 1.961e-01

Iteration no. 350: cost 1.894e-01

Iteration no. 420: cost 1.842e-01

Iteration no. 490: cost 1.802e-01

Iteration no. 560: cost 1.770e-01

Iteration no. 630: cost 1.744e-01

-----------------------------------------------------------------------------------------

Iteration no. 0: cost 2.265e-01

Iteration no. 70: cost 1.605e-01

Iteration no. 140: cost 1.420e-01

Iteration no. 210: cost 1.305e-01

Iteration no. 280: cost 1.222e-01

Iteration no. 350: cost 1.158e-01

Iteration no. 420: cost 1.108e-01

Iteration no. 490: cost 1.066e-01

Iteration no. 560: cost 1.031e-01

Iteration no. 630: cost 1.001e-01

-----------------------------------------------------------------------------------------

Iteration no. 0: cost 3.914e-01

Iteration no. 70: cost 2.646e-01

Iteration no. 140: cost 2.467e-01

Iteration no. 210: cost 2.367e-01

Iteration no. 280: cost 2.298e-01

Iteration no. 350: cost 2.247e-01

Iteration no. 420: cost 2.207e-01

Iteration no. 490: cost 2.174e-01

Iteration no. 560: cost 2.147e-01

Iteration no. 630: cost 2.124e-01

-----------------------------------------------------------------------------------------

Iteration no. 0: cost 2.562e-01

Iteration no. 70: cost 1.268e-01

Iteration no. 140: cost 1.138e-01

Iteration no. 210: cost 1.064e-01

Iteration no. 280: cost 1.010e-01

Iteration no. 350: cost 9.688e-02

Iteration no. 420: cost 9.352e-02

Iteration no. 490: cost 9.072e-02

Iteration no. 560: cost 8.833e-02

Iteration no. 630: cost 8.625e-02

-----------------------------------------------------------------------------------------

Iteration no. 0: cost 3.930e-01

Iteration no. 70: cost 1.719e-01

Iteration no. 140: cost 1.327e-01

Iteration no. 210: cost 1.160e-01

Iteration no. 280: cost 1.066e-01

Iteration no. 350: cost 1.005e-01

Iteration no. 420: cost 9.618e-02

Iteration no. 490: cost 9.293e-02

Iteration no. 560: cost 9.037e-02

Iteration no. 630: cost 8.828e-02

-----------------------------------------------------------------------------------------

Iteration no. 0: cost 3.542e-01

Iteration no. 70: cost 1.375e-01

Iteration no. 140: cost 1.174e-01

Iteration no. 210: cost 1.070e-01

Iteration no. 280: cost 1.003e-01

Iteration no. 350: cost 9.559e-02

Iteration no. 420: cost 9.202e-02

Iteration no. 490: cost 8.922e-02

Iteration no. 560: cost 8.694e-02

Iteration no. 630: cost 8.502e-02

-----------------------------------------------------------------------------------------

b,w found by gradient descent: [[-8.40705028e-06]

[-6.14149844e-06]

[-1.33832495e-05]

[-8.76401900e-06]

[-1.99795064e-05]

[ 8.67786764e-06]

[-1.13920487e-05]

[-8.22396459e-06]

[-1.63813445e-05]

[-1.55370924e-05]],[[ 4.92295472e-08 1.33566928e-07 4.05048712e-06 ... -4.76518828e-05

-1.73657243e-05 -5.46555707e-07]

[-7.10372445e-09 -2.73803987e-08 -4.32456289e-07 ... -1.23754474e-05

-5.83671730e-06 -2.77952317e-07]

[-6.19797002e-08 -1.33400888e-07 -7.74649323e-09 ... 1.13220049e-04

3.01601390e-05 6.69301019e-06]

...

[-1.63468190e-09 -2.05603028e-08 -2.47578766e-07 ... -2.14887181e-05

-9.35159417e-06 -4.52214446e-07]

[-1.00089933e-08 1.30898530e-07 -4.80361154e-06 ... -4.59349993e-05

-2.37054986e-05 -4.88485461e-06]

[-3.07954506e-09 1.19613014e-07 -3.04836452e-07 ... 1.42654888e-05

7.99685119e-06 2.36768066e-06]]

Class cv: [0. 1. 7. ... 9. 2. 0.]

* Train Accuracy: 80.250000
* Classes representing dresses
* Class 0



* Class 1



* Class 2



* Class 3



* Class 4



* Class 5



* Class 6



* Class 7



* Class 8



* Class 9



* **k Nearest Neighbours(kNN)**
* **In brief**
* Finding the distance from all the data points
* Sort them and take the first k(generally odd) distances from the sorted distances
* Counting the points and noting down the class these k points belong to
* --->The class with the max no. of points among the k points will be the predicted class of the unclassified point

Runtime: 18 mins

k-Value: 5

Train Accuracy: 80.700000

* **Neural Network**
* Here we have done the neural-networks for classification
* 2-layer neural network
* Training

Total iterations: 300

Number of layers: 2 | alpha: 0.001

Layer 1 : 784 neurons

Layer 2 : 10 neurons

Iteration no. 0: loss 1.5030e+01 acc: 0.066

Iteration no. 100: loss 2.3936e+00 acc: 0.720

Iteration no. 200: loss 1.8796e+00 acc: 0.738

Iteration no. 290: loss 1.6097e+00 acc: 0.748

Iteration no. 291: loss 1.6073e+00 acc: 0.748

Iteration no. 292: loss 1.6048e+00 acc: 0.748

Iteration no. 293: loss 1.6024e+00 acc: 0.748

Iteration no. 294: loss 1.6000e+00 acc: 0.748

Iteration no. 295: loss 1.5975e+00 acc: 0.748

Iteration no. 296: loss 1.5952e+00 acc: 0.749

Iteration no. 297: loss 1.5928e+00 acc: 0.749

Iteration no. 298: loss 1.5904e+00 acc: 0.749

Iteration no. 299: loss 1.5880e+00 acc: 0.749

Runtime: 15 minutes (approx.)

* We can increase the iterations to get more accuracy
* **n Layer neural network**
* Training
* Number of layers: 4 | alpha: 0.001
* Layer 1 : 20 neurons
* Layer 2 : 30 neurons
* Layer 3 : 40 neurons
* Layer 4 : 10 neurons
* Iteration no. 0: loss 1.3665e+01 acc: 0.100
* Iteration no. 100: loss 1.3809e+00 acc: 0.521
* Iteration no. 200: loss 1.1990e+00 acc: 0.575
* Iteration no. 240: loss 1.1500e+00 acc: 0.589
* Iteration no. 241: loss 1.1366e+00 acc: 0.581
* Iteration no. 242: loss 1.1480e+00 acc: 0.589
* Iteration no. 243: loss 1.1345e+00 acc: 0.581
* Iteration no. 244: loss 1.1458e+00 acc: 0.590
* Iteration no. 245: loss 1.1325e+00 acc: 0.582
* Iteration no. 246: loss 1.1438e+00 acc: 0.591
* Iteration no. 247: loss 1.1304e+00 acc: 0.583
* Iteration no. 248: loss 1.1418e+00 acc: 0.592
* Iteration no. 249: loss 1.1285e+00 acc: 0.583
* Runtime: 3 minutes
* Here also, we can increase the iterations to get more accuracy