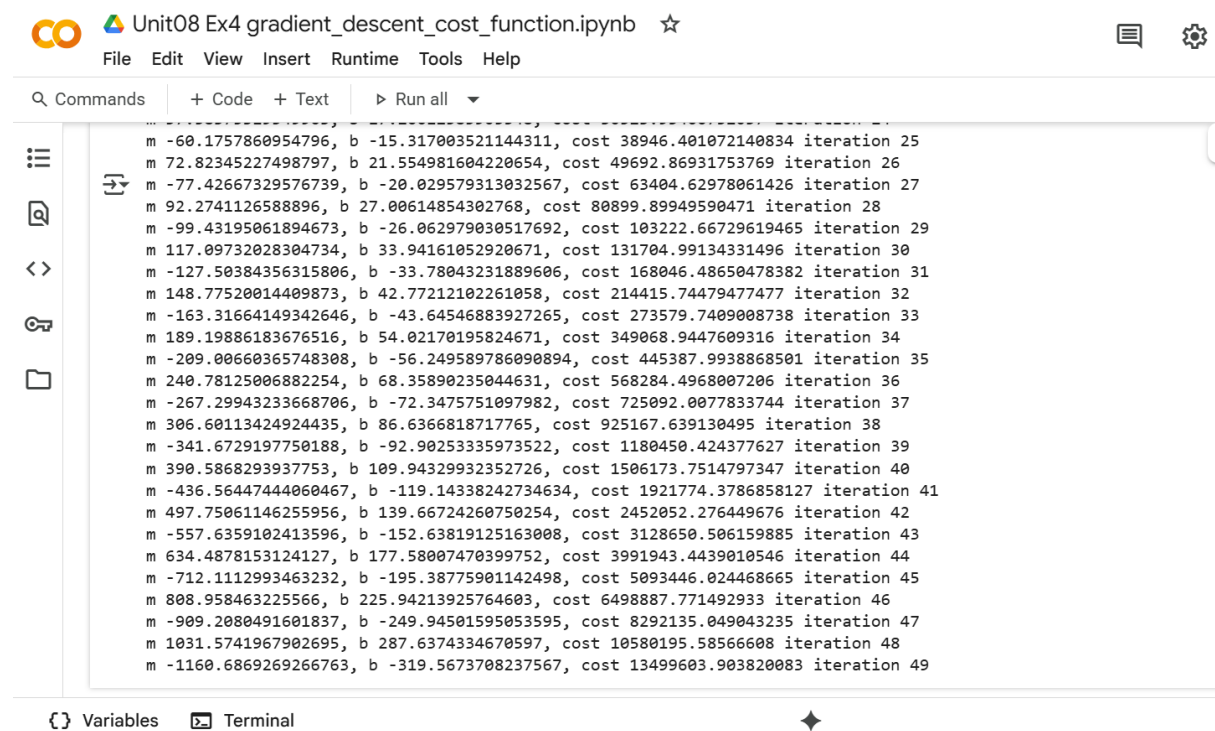


Activity 4

The tutorial illustrated the practical workings of gradient descent by visualizing the decrease in cost over time. We concentrated on fine-tuning the learning rate to help the model converge effectively without overshooting the optimal point. This hands-on exercise clarified how hyperparameters such as the learning rate impact training dynamics and overall model accuracy. It also highlighted the importance of balancing speed and stability during the optimization process.

50 Iterations and 0.09 Learning rate



The screenshot shows a Jupyter Notebook titled "Unit08 Ex4 gradient_descent_cost_function.ipynb". The interface includes a menu bar (File, Edit, View, Insert, Runtime, Tools, Help), a toolbar with icons for commands, code, text, and running, and a sidebar with icons for file explorer, search, and other functions. The main area displays a list of 50 iterations of training results, each showing the model parameters (m and b) and the cost function value. The results are as follows:

Iteration	m	b	cost
25	-60.1757860954796	-15.317003521144311	38946.401072140834
26	72.82345227498797	21.554981604220654	49692.86931753769
27	-77.42667329576739	-20.029579313032567	63404.62978061426
28	92.2741126588896	27.00614854302768	80899.89949590471
29	-99.43195061894673	-26.062979030517692	103222.66729619465
30	117.09732028304734	33.94161052920671	131704.99134331496
31	-127.50384356315806	-33.78043231889606	168046.48650478382
32	148.77520014409873	42.77212102261058	214415.74479477477
33	-163.31664149342646	-43.64546883927265	273579.7409008738
34	189.19886183676516	54.02170195824671	349068.9447609316
35	-209.00660365748308	-56.249589786090894	445387.9938868501
36	240.78125006882254	68.35890235044631	568284.4968007206
37	-267.29943233668706	-72.3475751097982	725092.0077833744
38	306.60113424924435	86.6366818717765	925167.639130495
39	-341.6729197750188	-92.90253335973522	1180450.424377627
40	390.5868293937753	109.94329932352726	1506173.7514797347
41	-436.56447444060467	-119.14338242734634	1921774.3786858127
42	497.75061146255956	139.66724260750254	2452052.276449676
43	-557.6359102413596	-152.63819125163008	3128650.506159885
44	634.4878153124127	177.58007470399752	3991943.4439010546
45	-712.1112993463232	-195.38775901142498	5093446.024468665
46	808.958463225566	225.94213925764603	6498887.771492933
47	-909.2080491601837	-249.94501595053595	8292135.049043235
48	1031.5741967902695	287.6374334670597	10580195.58566608
49	-1160.6869269266763	-319.5673708237567	13499603.903820083

150 Iterations and 0.09 Learning rate

Unit08 Ex4 gradient_descent_cost_function.ipynb

File Edit View Insert Runtime Tools Help

Q Commands + Code + Text ▶ Run all

≡

🔍

<>

🔗

📁

```
m -12216629.379201535, b -3383808.0475572473, cost 1490040682164294.5 iteration 125
m 13799558.717298418, b 3822258.885771886, cost 1901189715363038.5 iteration 126
m -15587581.761269271, b -4317507.8010082, cost 2425787682892033.5 iteration 127
m 17607289.918588314, b 4876939.374258682, cost 3095138709682664.0 iteration 128
m -19888685.802316237, b -5508844.6491455715, cost 3949184712140546.5 iteration 129
m 22465693.776808526, b 6222639.3409514, cost 5038888836165807.0 iteration 130
m -25376599.565386105, b -7028908.759896457, cost 6429276560597804.0 iteration 131
m 28664683.884422477, b 7939660.202193402, cost 8203315936635229.0 iteration 132
m -32378801.135918472, b -8968406.311789546, cost 1.0466868507208286e+16 iteration 133
m 36574170.10156646, b 10130461.057728548, cost 1.3355006340536628e+16 iteration 134
m -41313130.09070854, b -11443072.167508483, cost 1.7040072131692852e+16 iteration 135
m 46666132.03934896, b 12925772.691625651, cost 2.174196333939201e+16 iteration 136
m -52712721.07203984, b -14600576.07411541, cost 2.774125403919312e+16 iteration 137
m 59542783.31062135, b 16492398.618126873, cost 3.539593750821627e+16 iteration 138
m -67257817.31819744, b -18629334.5008715, cost 4.516278861494436e+16 iteration 139
m 75972507.1823041, b 21043168.681111995, cost 5.762462076346161e+16 iteration 140
m -85816362.5464585, b -23769753.93993238, cost 7.352506388487237e+16 iteration 141
m 96935708.00309281, b 26849639.16434304, cost 9.381293876908842e+16 iteration 142
m -109495793.4117762, b -30328576.58690882, cost 1.1969887566875669e+17 iteration 143
m 123683314.48047148, b 34258297.26109393, cost 1.527275557546602e+17 iteration 144
m -139709123.13185275, b -38697184.445357576, cost 1.948698862580981e+17 iteration 145
m 157811425.84970886, b 43711236.86600727, cost 2.4864060963069104e+17 iteration 146
m -178259259.6603586, b -49374954.10871683, cost 3.172483647660192e+17 iteration 147
m 201356555.26585853, b 55772539.467445835, cost 4.0478715482641626e+17 iteration 148
m -227446589.89296216, b -62999055.86025804, cost 5.164806470580601e+17 iteration 149
```

{ } Variables

📄 Terminal

295 Iterations and 0.18 Learning rate

Unit08 Ex4 gradient_descent_cost_function.ipynb

File Edit View Insert Runtime Tools Help

Q Commands + Code + Text ▶ Run all

≡

🔍

<>

🔗

📁

```
m 2.956950015611448e+139, b 8.190277394783913e+138, cost 1.0485852261554812e+279 iteration 270
m -9.637122004846549e+139, b -2.669328263594194e+139, cost 1.1138083830332609e+280 iteration 271
m 3.1408755659027524e+140, b 8.69972167653399e+139, cost 1.1830884921615508e+281 iteration 272
m -1.023656161613782e+141, b -2.8353634238767977e+140, cost 1.2566779004421393e+282 iteration 273
m 3.33624148815549e+141, b 9.240853954147695e+140, cost 1.3348446510322561e+283 iteration 274
m -1.0873287031988205e+142, b -3.011726154142477e+141, cost 1.4178734596689636e+284 iteration 275
m 3.543759386115897e+142, b 9.815645255896074e+141, cost 1.506066751722e+285 iteration 276
m -1.1549617470539832e+143, b -3.19905884062782e+142, cost 1.5997457637524505e+286 iteration 277
m 3.7641851260675953e+143, b 1.0426189210181212e+143, cost 1.699251713589587e+287 iteration 278
m -1.2268016407859652e+144, b -3.398043826701405e+143, cost 1.8049470431877702e+288 iteration 279
m 3.998321590010209e+144, b 1.1074709671399525e+144, cost 1.917216738790428e+289 iteration 280
m -1.3031100550941364e+145, b -3.609405898241456e+144, cost 2.036469733209683e+290 iteration 281
m 4.2470216000887215e+145, b 1.1763568820142142e+145, cost 2.1631403953293197e+291 iteration 282
m -1.3841649368837967e+146, b -3.8339149236067226e+145, cost 2.2976901122564834e+292 iteration 283
m 4.511191024925565e+146, b 1.24952757672367e+146, cost 2.4406089698849485e+293 iteration 284
m -1.4702615216641234e+147, b -4.072388657816461e+146, cost 2.592417537991285e+294 iteration 285
m 4.791792079169983e+147, b 1.3272495692969998e+147, cost 2.753668766365963e+295 iteration 286
m -1.5617134089183906e+148, b -4.325695721153502e+147, cost 2.9249499988859197e+296 iteration 287
m 5.0898468282830145e+148, b 1.4098059554780377e+148, cost 3.106885112864636e+297 iteration 288
m -1.6588537043634005e+149, b -4.5947587630397113e+148, cost 3.30013679147217e+298 iteration 289
m 5.406440911323954e+149, b 1.4974974398779312e+149, cost 3.505408937502201e+299 iteration 290
m -1.762036233258707e+150, b -4.880557822707993e+149, cost 3.7234492372780577e+300 iteration 291
m 5.742727495298235e+150, b 1.5906434312660921e+150, cost 3.955051884036525e+301 iteration 292
m -1.8716368291850153e+151, b -5.184133898911795e+150, cost 4.201060470709116e+302 iteration 293
m 6.09993147547012e+151, b 1.6895832059894622e+151, cost 4.4623710626375486e+303 iteration 294
```

{ } Variables

📄 Terminal

The experiment demonstrated that adjusting the learning rate and iteration count demands a precise balance. Increasing these parameters without observing the cost trend can cause

overshooting or unstable training, while reducing them too much may hinder the model from reaching the optimal minimum. Our findings showed that the most effective strategy is to monitor the cost function's decline and extend training only when necessary—emphasizing the value of adaptive optimization guided by convergence behavior rather than relying on fixed settings.