Deep Learning HW1 – Report

Saar Ben David – 305560591

Ido Rabia – 201181450

**Summary:**

We’ve Implemented a NN framework using vectorization. We’ve tested our framework on the MNIST dataset. We trained our network twice: one time without Batchnorm and one time with Batchnorm. Both in terms of performance (i.e. number of iterations) and accuracy, training with Batchnorm outperformed training without Batchnorm.

In addition, we’ve implemented a Dropout mechanism and tested it with and without Batchnorm. Both cases performed poorly compared to training without Dropout.

**Note** – The default settings of our code are use\_batchnorm=True, use\_dropout=False (can be changed in the main.py).

**Results:**

At first, we used stop-criterion of “100 iterations without improvement”, which didn’t allow the training without Batchnorm to converge (it finished after 2782 iterations, with accuracy of ~60%).

We then raised the stop-criterion to “350 iterations without improvement”, and we got the results shown in the table below. It’s clear that training the network with Batchnorm reduced the number of iterations needed (by ~46%) and improved accuracy (by ~1.5%).

|  |  |  |
| --- | --- | --- |
|  | Without Batchnorm | With Batchnorm |
| Batch Size | 50 | 50 |
| Learning Rate | 0.009 | 0.009 |
| Total Iterations | 15000 (hit the maximum) | 8224 |
| Stop Criterion | No improvement for 350 iterations | No improvement for 350 iterations |
| Epochs | 15 | 8 |
| Validation | 0.8915 | 0.9029 |
| Train | 0.8971 | 0.9042 |
| Test | 0.8894 | 0.9039 |

In the graph below we can see the costs of both training executions along the training iterations.

**Note** - in order for the two lines (Batchnorm & w/o Batchnorm) to be of the same length, we enforced the execution with Batchnorm to train for 15000 iterations (normally it would take 8224). We marked the additional (enforcecd) iterations in yellow in the costs table below (*see Appendix*).

Timeline

Description automatically generated with medium confidenceThe graph suggests that training with Batchnorm converges faster - after ~300 iterations there’s a major drop in the cost, compared to No-Batchnorm line.

In addition, it is pretty clear that starting at iteration ~300 the cost of Batchnorm is lower/equal to the cost of No-Batchnorm.

**Bonus:**

In order to implement the Dropout mechanism, we’ve added a code section to the L\_model\_forward function. We defined the keep\_probability to be 0.9 (hard-coded), and we randomly generated a 0/1 masking-matrix in each iteration. The masking-matrix has the same shape as the current activation layer. Lastly, we multiply the masking-matrix with the current activation layer and normalized by the keep\_probability.

When added the Dropout mechanism we inspected a dramatic decrease in performance (with or without Batchnorm). In our opinion, the reason for that decrease is due to the fact that the network is not deep enough to justify the usage of dropout. In addition, it’s not wide enough (every node is too important to its layer). Other regularization methods might suit for this specific network, such as L2-norm. See results table below.

|  |  |  |
| --- | --- | --- |
|  | Without Batchnorm | With Batchnorm |
| Batch Size | 50 | 50 |
| Learning Rate | 0.009 | 0.009 |
| Total Iterations | 2545 | 4801 |
| Stop Criterion | No improvement for 350 iterations | No improvement for 350 iterations |
| Epochs | 2 | 5 |
| Validation | 0.3156 | 0.7971 |
| Train | 0.2567 | 0.7001 |
| Test | 0.2138 | 0.6905 |

**Appendix:**

|  |  |  |
| --- | --- | --- |
| iteration | Batchnorm | No Batchnorm |
| 0 | 2.78540394 | 2.307143223 |
| 100 | 2.65916479 | 2.212297171 |
| 200 | 2.347815779 | 2.19542124 |
| 300 | 1.97086282 | 2.11484286 |
| 400 | 1.8416731 | 2.07970183 |
| 500 | 1.77795576 | 1.99192528 |
| 600 | 1.68654497 | 2.05690274 |
| 700 | 1.72141726 | 1.88167496 |
| 800 | 1.4559581 | 1.94583345 |
| 900 | 1.71338693 | 1.79571772 |
| 1000 | 1.5356181 | 1.88182757 |
| 1100 | 1.36593593 | 1.86921478 |
| 1200 | 1.46806342 | 1.71623897 |
| 1300 | 1.3302365 | 1.68844466 |
| 1400 | 1.39834706 | 1.77313537 |
| 1500 | 1.02791348 | 1.52182841 |
| 1600 | 1.10758172 | 1.56490381 |
| 1700 | 1.0098612 | 1.58591657 |
| 1800 | 1.16337484 | 1.57717157 |
| 1900 | 1.19662236 | 1.36887989 |
| 2000 | 0.89919449 | 1.35876269 |
| 2100 | 0.86852341 | 1.36353158 |
| 2200 | 1.01026182 | 1.42694152 |
| 2300 | 1.0287075 | 1.40032013 |
| 2400 | 0.71869883 | 1.0652689 |
| 2500 | 1.01504026 | 1.16677904 |
| 2600 | 0.87984494 | 1.2441739 |
| 2700 | 0.8497869 | 1.16139484 |
| 2800 | 0.86782933 | 1.44407924 |
| 2900 | 0.69509367 | 1.16181164 |
| 3000 | 0.92007513 | 1.21287908 |
| 3100 | 0.76824972 | 1.10929954 |
| 3200 | 0.8490991 | 1.30831384 |
| 3300 | 0.70091343 | 1.01457902 |
| 3400 | 0.6145565 | 0.9705818 |
| 3500 | 0.54949809 | 0.90443037 |
| 3600 | 0.92926653 | 1.24622259 |
| 3700 | 0.60501046 | 1.00351566 |
| 3800 | 0.72904384 | 0.95939295 |
| 3900 | 0.63632745 | 0.96018284 |
| 4000 | 0.52556534 | 0.81447507 |
| 4100 | 0.56687654 | 0.762656 |
| 4200 | 0.50667407 | 0.95794351 |
| 4300 | 0.57099386 | 0.77893928 |
| 4400 | 0.72316575 | 1.40208506 |
| 4500 | 0.64802313 | 1.00782768 |
| 4600 | 0.56888548 | 0.71131232 |
| 4700 | 0.59991267 | 1.10113041 |
| 4800 | 0.52298739 | 0.91725275 |
| 4900 | 0.81776664 | 1.14107407 |
| 5000 | 0.45965513 | 0.91410929 |
| 5100 | 0.52253019 | 0.91057873 |
| 5200 | 0.54186957 | 1.00371004 |
| 5300 | 0.38580843 | 0.57804832 |
| 5400 | 0.72142797 | 0.88052387 |
| 5500 | 0.51441581 | 0.67178064 |
| 5600 | 0.47553785 | 0.79282163 |
| 5700 | 0.63951584 | 0.85328397 |
| 5800 | 0.68700695 | 0.88700039 |
| 5900 | 0.41841968 | 0.88279108 |
| 6000 | 0.41564646 | 0.74151278 |
| 6100 | 0.43886873 | 0.76559039 |
| 6200 | 0.6719369 | 0.99646372 |
| 6300 | 0.32853508 | 0.59607677 |
| 6400 | 0.51737919 | 0.64737793 |
| 6500 | 0.45796139 | 0.86087474 |
| 6600 | 0.55982486 | 0.64098472 |
| 6700 | 0.44093007 | 0.68727318 |
| 6800 | 0.32638969 | 0.60247673 |
| 6900 | 0.43427162 | 0.70262762 |
| 7000 | 0.41815625 | 0.80493033 |
| 7100 | 0.54258887 | 0.60735987 |
| 7200 | 0.37936004 | 0.59734311 |
| 7300 | 0.53744073 | 0.82691829 |
| 7400 | 0.45244177 | 0.70245254 |
| 7500 | 0.53635617 | 0.69994169 |
| 7600 | 0.56990986 | 0.87880035 |
| 7700 | 0.3337982 | 0.58511865 |
| 7800 | 0.41815399 | 0.74346761 |
| 7900 | 0.44770654 | 0.69142261 |
| 8000 | 0.62733913 | 0.94427276 |
| 8100 | 0.41544665 | 0.61374233 |
| 8200 | 0.29874423 | 0.63691339 |
| 8300 | 0.3192404 | 0.45383293 |
| 8400 | 0.68793172 | 0.70564664 |
| 8500 | 0.45665712 | 0.588695 |
| 8600 | 0.52005905 | 0.52259456 |
| 8700 | 0.46439673 | 0.74336739 |
| 8800 | 0.35211929 | 0.47116173 |
| 8900 | 0.38793763 | 0.4582736 |
| 9000 | 0.32335152 | 0.56317808 |
| 9100 | 0.37752916 | 0.46849526 |
| 9200 | 0.54427209 | 1.07067885 |
| 9300 | 0.51163174 | 0.72805591 |
| 9400 | 0.27689213 | 0.45966281 |
| 9500 | 0.44130063 | 0.78021101 |
| 9600 | 0.37059574 | 0.5576674 |
| 9700 | 0.70590203 | 0.67299821 |
| 9800 | 0.36885169 | 0.72354801 |
| 9900 | 0.40589288 | 0.63845378 |
| 10000 | 0.3585854 | 0.66324052 |
| 10100 | 0.24646743 | 0.34766954 |
| 10200 | 0.52866566 | 0.73350577 |
| 10300 | 0.36009357 | 0.38407222 |
| 10400 | 0.39630266 | 0.53251725 |
| 10500 | 0.48965511 | 0.5357839 |
| 10600 | 0.63846833 | 0.76552041 |
| 10700 | 0.30443527 | 0.63575779 |
| 10800 | 0.29898013 | 0.57137484 |
| 10900 | 0.38151646 | 0.5899765 |
| 11000 | 0.57007302 | 0.81104678 |
| 11100 | 0.26897028 | 0.41398633 |
| 11200 | 0.42576022 | 0.5201872 |
| 11300 | 0.35235007 | 0.54874019 |
| 11400 | 0.47559302 | 0.44895563 |
| 11500 | 0.33086516 | 0.39012485 |
| 11600 | 0.21698669 | 0.45341418 |
| 11700 | 0.39580193 | 0.51563538 |
| 11800 | 0.30275617 | 0.60441751 |
| 11900 | 0.41390273 | 0.3308317 |
| 12000 | 0.24282569 | 0.47274729 |
| 12100 | 0.50987144 | 0.61440806 |
| 12200 | 0.3486532 | 0.52784268 |
| 12300 | 0.45328779 | 0.62196311 |
| 12400 | 0.47556838 | 0.64533067 |
| 12500 | 0.31469447 | 0.39470275 |
| 12600 | 0.29120984 | 0.52010965 |
| 12700 | 0.36494484 | 0.43959077 |
| 12800 | 0.58963538 | 0.89838483 |
| 12900 | 0.36724517 | 0.43965417 |
| 13000 | 0.23585624 | 0.48184353 |
| 13100 | 0.25596294 | 0.27609989 |
| 13200 | 0.5623705 | 0.52529631 |
| 13300 | 0.35867053 | 0.41105408 |
| 13400 | 0.39940327 | 0.42439299 |
| 13500 | 0.35513513 | 0.51475018 |
| 13600 | 0.28316312 | 0.29499855 |
| 13700 | 0.35589853 | 0.35791364 |
| 13800 | 0.28842941 | 0.35758297 |
| 13900 | 0.32737193 | 0.28595428 |
| 14000 | 0.54693318 | 0.75589486 |
| 14100 | 0.46770782 | 0.56120319 |
| 14200 | 0.18227805 | 0.2813179 |
| 14300 | 0.34795699 | 0.58163671 |
| 14400 | 0.28370352 | 0.49585705 |
| 14500 | 0.58975457 | 0.5555593 |
| 14600 | 0.30330235 | 0.56564281 |
| 14700 | 0.33482521 | 0.4812242 |
| 14800 | 0.3282743 | 0.51636084 |
| 14900 | 0.17540035 | 0.26460519 |
| 15000 | 0.47075046 | 0.66585292 |