

450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524

Assignment 3

tldr: Classify MNIST digits with a (optionally convoultional) neural network. Get at least 95.5% accuracy on the test test.

Problem Statement Consider the MNIST dataset consisting of 50,000 training images, and 10,000 test images. Each instance is a 28×28 pixel handwritten digit zero through nine. Train a (optionally convolutional) neural network for classification using the training set that achieves at least 95.5% accuracy on the test set. Do not explicitly tune hyperparameters based on the test set performance, use a validation set taken from the training set as discussed in class. Use dropout and an L^2 penalty for regularization. Note: if you write a sufficiently general program the next assignment will be very easy.

Do not use the built in MNIST data class from TensorFlow.

Extra challenge (optional) In addition to the above, the student with the fewest number of parameters for a network that gets at least 80% accuracy on the test set will receive a prize. There will be an extra prize if any one can achieve 80% on the test set with a single digit number of parameters. For this extra challenge you can make your network have any crazy kind of topology you'd like, it just needs to be optimized by a gradient based algorithm.