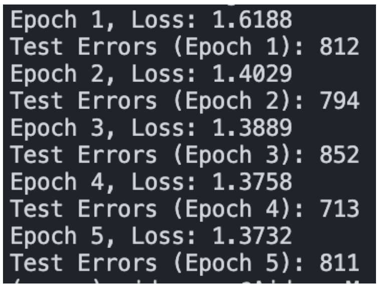
Dataset:
MNIST (60000 training cases, 10000 testing cases)

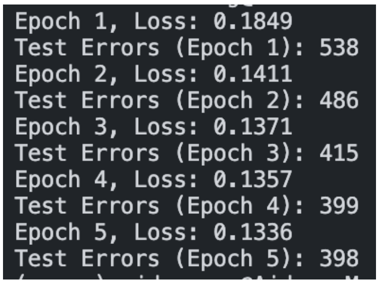
Note: Input and output layers are not hidden layers
Teacher model:

- 2 hidden layers of 1200 neurons via RELU on all 60000 MNIST training cases
- Regularized using dropout and weight-constraints described in another paper. (To prevent overfitting) (According to the other paper: 50% drop out with separate L2 constraints on the incoming weights of each hidden unit)
- Input images jittered up by 2 pixels in any direction.
- Aim 67 test errors

Model	Command	Results	% Optimization
Teacher	`python train_teacher.py --use_l2_regularization --jitter`	 Epoch 1, Loss: 1.6188 Test Errors (Epoch 1): 812 Epoch 2, Loss: 1.4029 Test Errors (Epoch 2): 794 Epoch 3, Loss: 1.3889 Test Errors (Epoch 3): 852 Epoch 4, Loss: 1.3758 Test Errors (Epoch 4): 713 Epoch 5, Loss: 1.3732 Test Errors (Epoch 5): 811	0.12%

Experiment 1: Student model:

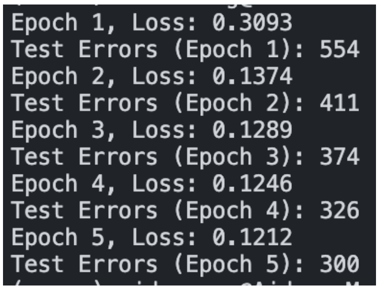
- 2 hidden layers of 800 neurons via RELU on all 60000 MNIST training cases
- No regularization
- Aim 146 test errors

Model	Command	Results	% Optimization
Student	`python train_student.py --jitter`	 Epoch 1, Loss: 0.1849 Test Errors (Epoch 1): 538 Epoch 2, Loss: 0.1411 Test Errors (Epoch 2): 486 Epoch 3, Loss: 0.1371 Test Errors (Epoch 3): 415 Epoch 4, Loss: 0.1357 Test Errors (Epoch 4): 399 Epoch 5, Loss: 0.1336 Test Errors (Epoch 5): 398	26.02%

Observation: Without distillation, student saw a 140 reduction in test errors from epoch 1 to 5

Experiment 2: Student model (regularized):

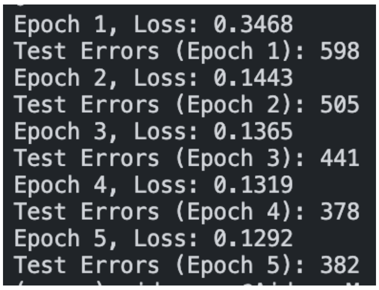
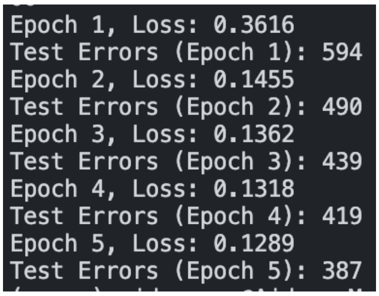
- 2 hidden layers of 800 neurons via RELU on all 60000 MNIST training cases
- Regularization: Matching the soft targets produced by teacher at T=20
- Aim 74 test errors

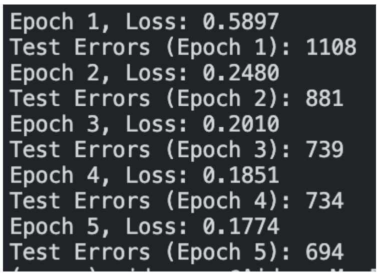
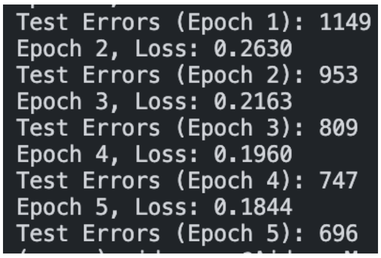
Model	Command	Results	% Optimization
Student	`python train_student.py --jitter --use_distillation True --temperature 20`	 Epoch 1, Loss: 0.3093 Test Errors (Epoch 1): 554 Epoch 2, Loss: 0.1374 Test Errors (Epoch 2): 411 Epoch 3, Loss: 0.1289 Test Errors (Epoch 3): 374 Epoch 4, Loss: 0.1246 Test Errors (Epoch 4): 326 Epoch 5, Loss: 0.1212 Test Errors (Epoch 5): 300	45.85%

Observation: With distillation, student saw a 254 reduction in test errors from epoch 1 to 5. This shows that distillation helped speed up the rate of reduction of errors, which is good. However, student with distillation started with more test errors (554) compared to the student without distillation (538).

Experiment 3: Student model (variations):

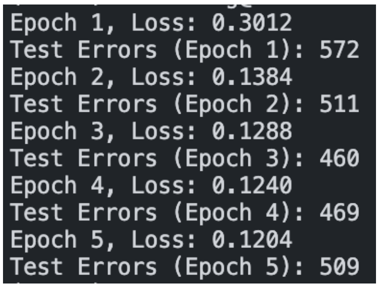
- 2 hidden layers of 300+ neurons via RELU, tested on T over 8: Aim similar results
- 2 hidden layers of 30 neurons via RELU, T from 2.5-4: Aim for results that are much better (lower test errors, higher optimization)

Model	Command	Results	% Optimization
Student (300 layer neurons, T = 9, regularized)	`python train_student.py --jitter --use_distillation True --temperature 9 --hidden_sizes 300 300`	 Epoch 1, Loss: 0.3468 Test Errors (Epoch 1): 598 Epoch 2, Loss: 0.1443 Test Errors (Epoch 2): 505 Epoch 3, Loss: 0.1365 Test Errors (Epoch 3): 441 Epoch 4, Loss: 0.1319 Test Errors (Epoch 4): 378 Epoch 5, Loss: 0.1292 Test Errors (Epoch 5): 382	36.12%
Student (300 layer neurons, T = 10, regularized)	`python train_student.py --jitter --use_distillation True --temperature 10 --hidden_sizes 300 300`	 Epoch 1, Loss: 0.3616 Test Errors (Epoch 1): 594 Epoch 2, Loss: 0.1455 Test Errors (Epoch 2): 490 Epoch 3, Loss: 0.1362 Test Errors (Epoch 3): 439 Epoch 4, Loss: 0.1318 Test Errors (Epoch 4): 419 Epoch 5, Loss: 0.1289 Test Errors (Epoch 5): 387	34.85%

Student (30 layer neurons, T = 2.4, regularized)	`python train_student.py --jitter --use_distillation True --temperature 2.4 --hidden_sizes 30 30`	 Epoch 1, Loss: 0.5897 Test Errors (Epoch 1): 1108 Epoch 2, Loss: 0.2480 Test Errors (Epoch 2): 881 Epoch 3, Loss: 0.2010 Test Errors (Epoch 3): 739 Epoch 4, Loss: 0.1851 Test Errors (Epoch 4): 734 Epoch 5, Loss: 0.1774 Test Errors (Epoch 5): 694	37.36%
Student (30 layer neurons, T = 4, regularized)	`python train_student.py --jitter --use_distillation True --temperature 4 --hidden_sizes 30 30`	 Test Errors (Epoch 1): 1149 Epoch 2, Loss: 0.2630 Test Errors (Epoch 2): 953 Epoch 3, Loss: 0.2163 Test Errors (Epoch 3): 809 Epoch 4, Loss: 0.1960 Test Errors (Epoch 4): 747 Epoch 5, Loss: 0.1844 Test Errors (Epoch 5): 696	39.43%

Experiment 4: Omit 3 from transfer set:

- Run student model (assume not regularized): Aim 206 test errors, of which 133 are on the 1010 3s in the test set

Model	Command	Results	% Optimization
Student (800 layer neurons, T = 20, regularized)	`python train_student.py --jitter --use_distillation True --temperature 20 --omit_class 3`	 Epoch 1, Loss: 0.3012 Test Errors (Epoch 1): 572 Epoch 2, Loss: 0.1384 Test Errors (Epoch 2): 511 Epoch 3, Loss: 0.1288 Test Errors (Epoch 3): 460 Epoch 4, Loss: 0.1240 Test Errors (Epoch 4): 469 Epoch 5, Loss: 0.1204 Test Errors (Epoch 5): 509	11.01%

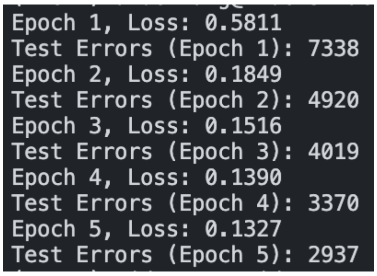
Experiment 5: Increased bias by 3.5:

- Aim 109 errors, 14 are on 3s
- No results for this yet, have not implemented arg to vary bias

Experiment 6: Transfer set only contains 7s and 8s from training set

- Aim student makes 47.3% more errors (unclear what this is relative to from paper)

Model	Command	Results	% Optimization

Student (800 layer neurons, T = 20, regularized)	`python train_student.py --jitter --use_distillation True --temperature 20 --classes 7 8`	
		`Epoch 1, Loss: 0.5811`
		`Test Errors (Epoch 1): 7338`
		`Epoch 2, Loss: 0.1849`
		`Test Errors (Epoch 2): 4920`
		`Epoch 3, Loss: 0.1516`
		`Test Errors (Epoch 3): 4019`
		`Epoch 4, Loss: 0.1390`
		`Test Errors (Epoch 4): 3370`
		`Epoch 5, Loss: 0.1327`
		`Test Errors (Epoch 5): 2937`
			59.98%

- After biases for 7 and 8 are reduced by 7.6, aim 13.2% test errors
- No results for this yet, have not implemented arg to vary bias