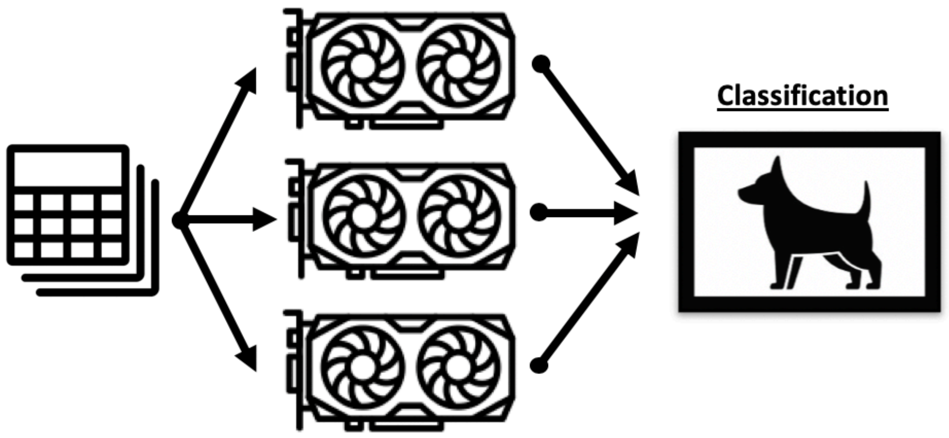
Accelerating Deep Neural Networks

Aumit Leon ‘19.5

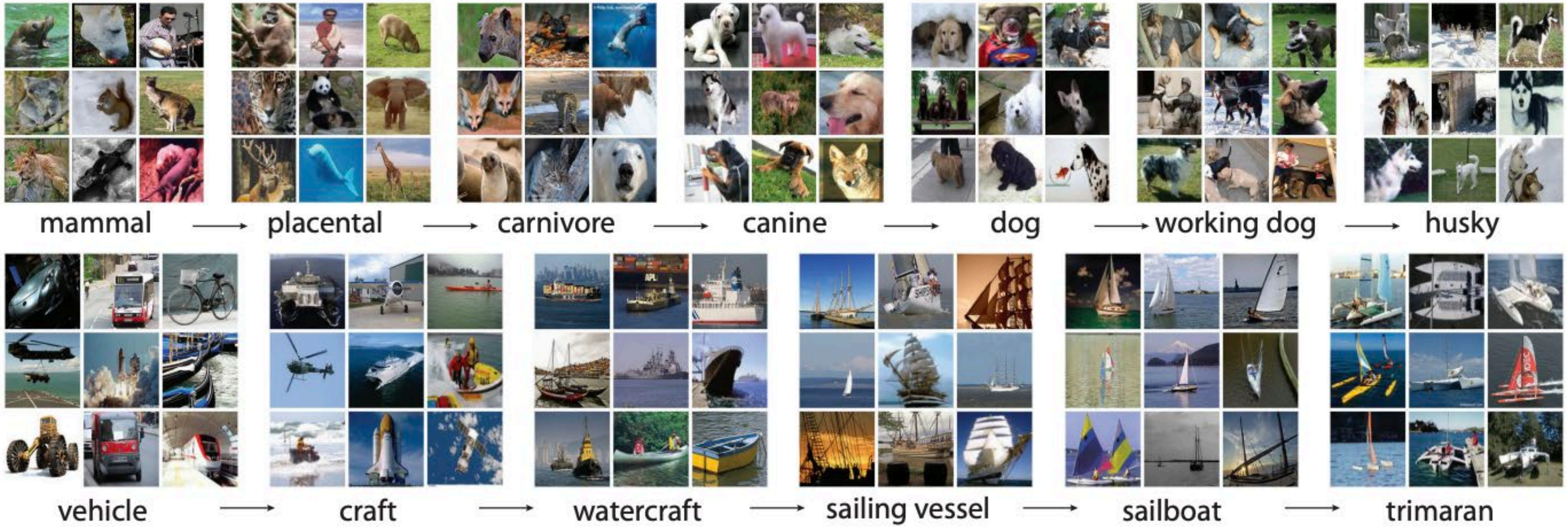
Advised by Prof. Andrea Vaccari



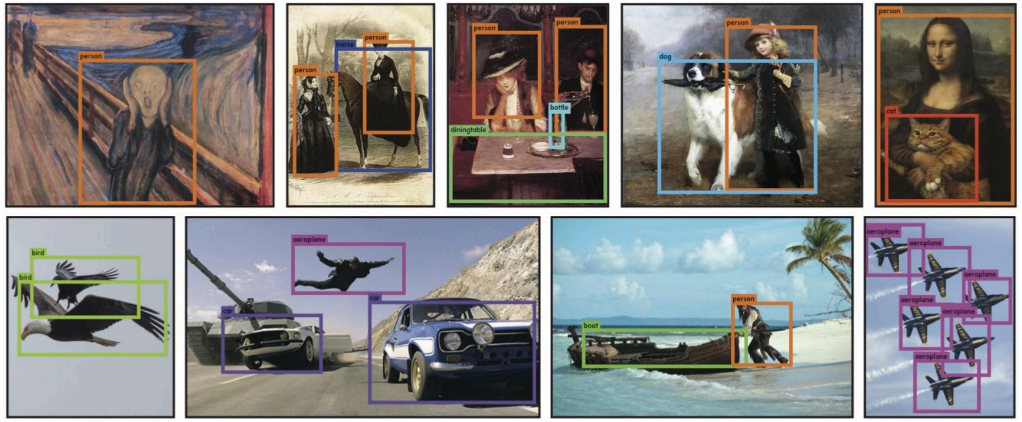
Overview

- Deep Learning and the State of the Art in Image Recognition
- The Need for Acceleration Through Parallelism
- Machine Learning Overview
- Parallelism within Deep Learning
- Image Classification with AlexNet
- Parallelizing AlexNet
- Conclusions

Deep Learning and the State-of-the-Art

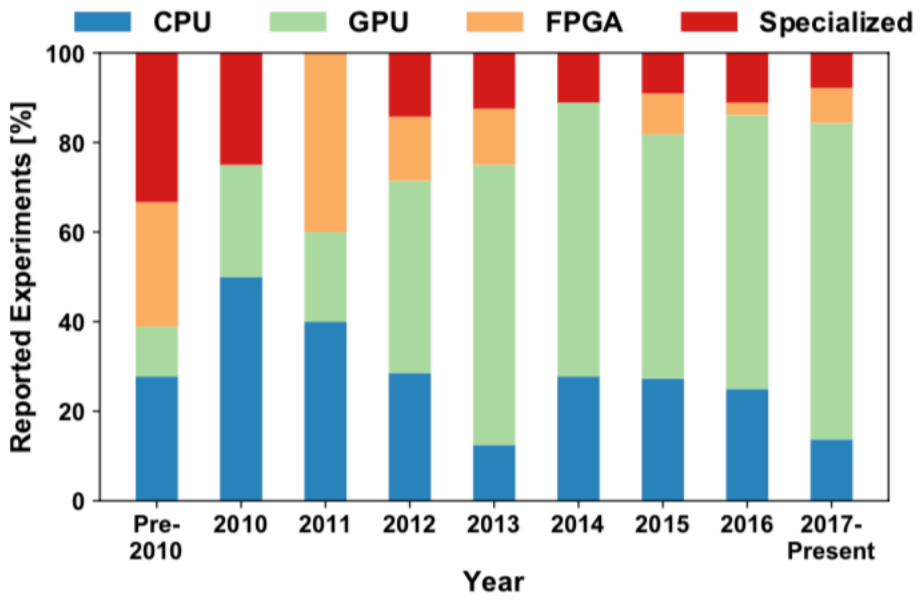


1

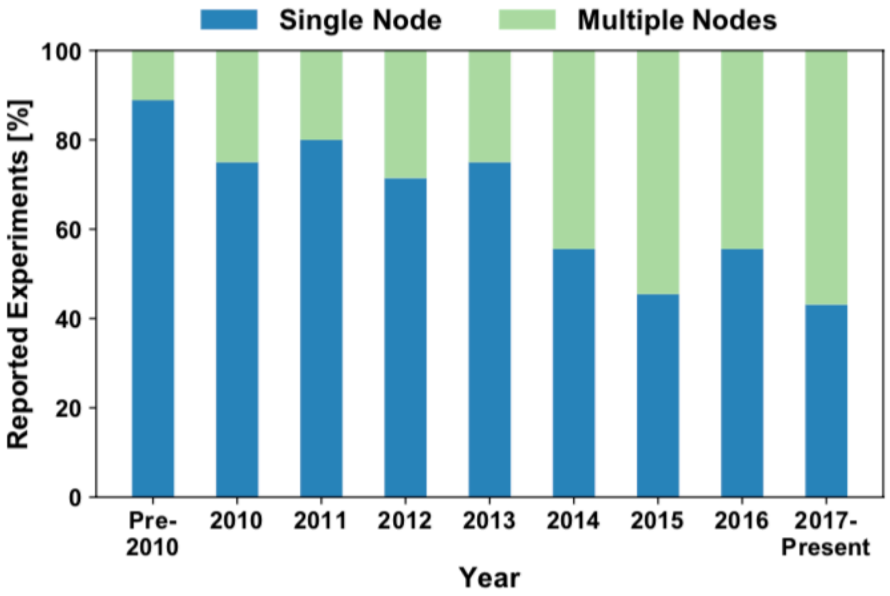


2

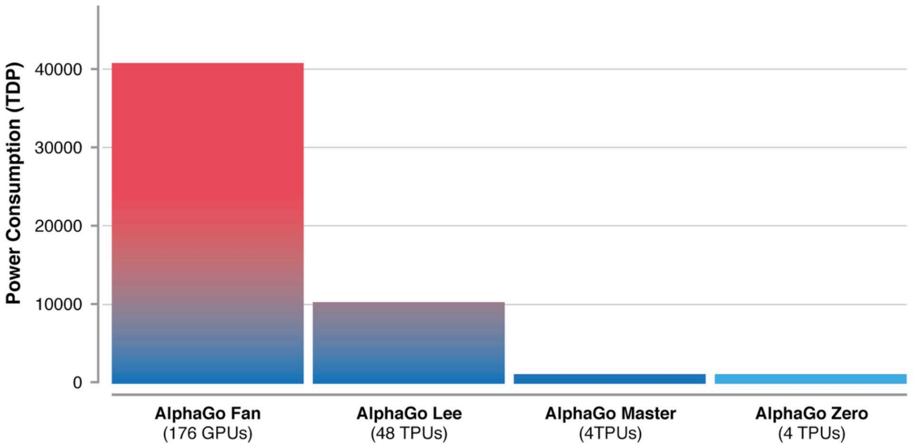
The Need for Parallelism Within Deep Learning



[3]



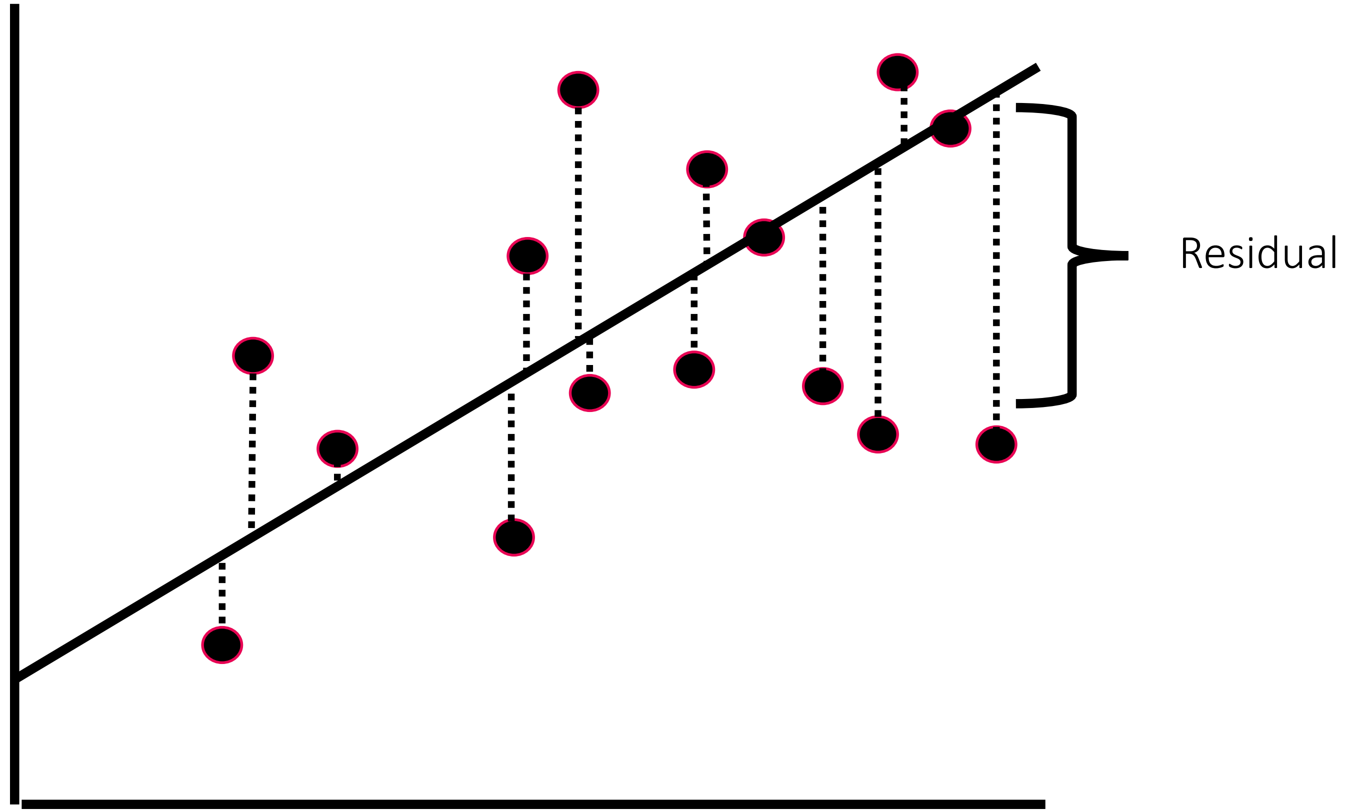
[3]



[4]

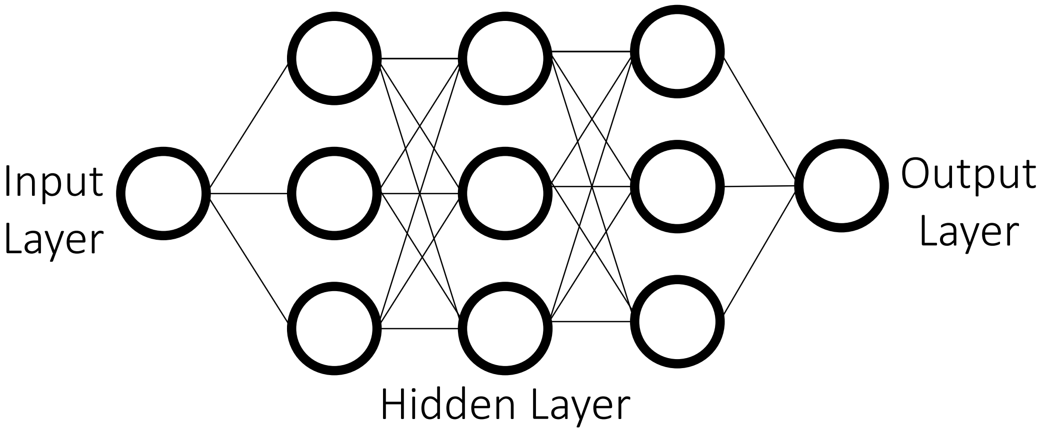
[4]

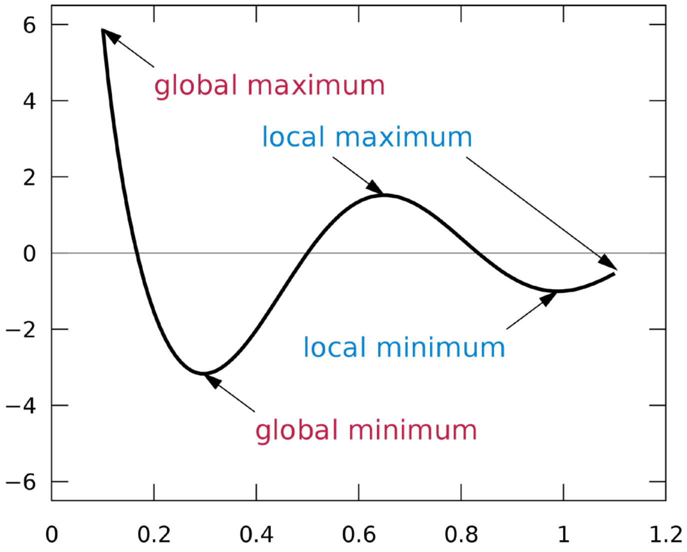
Background – Machine Learning Overview



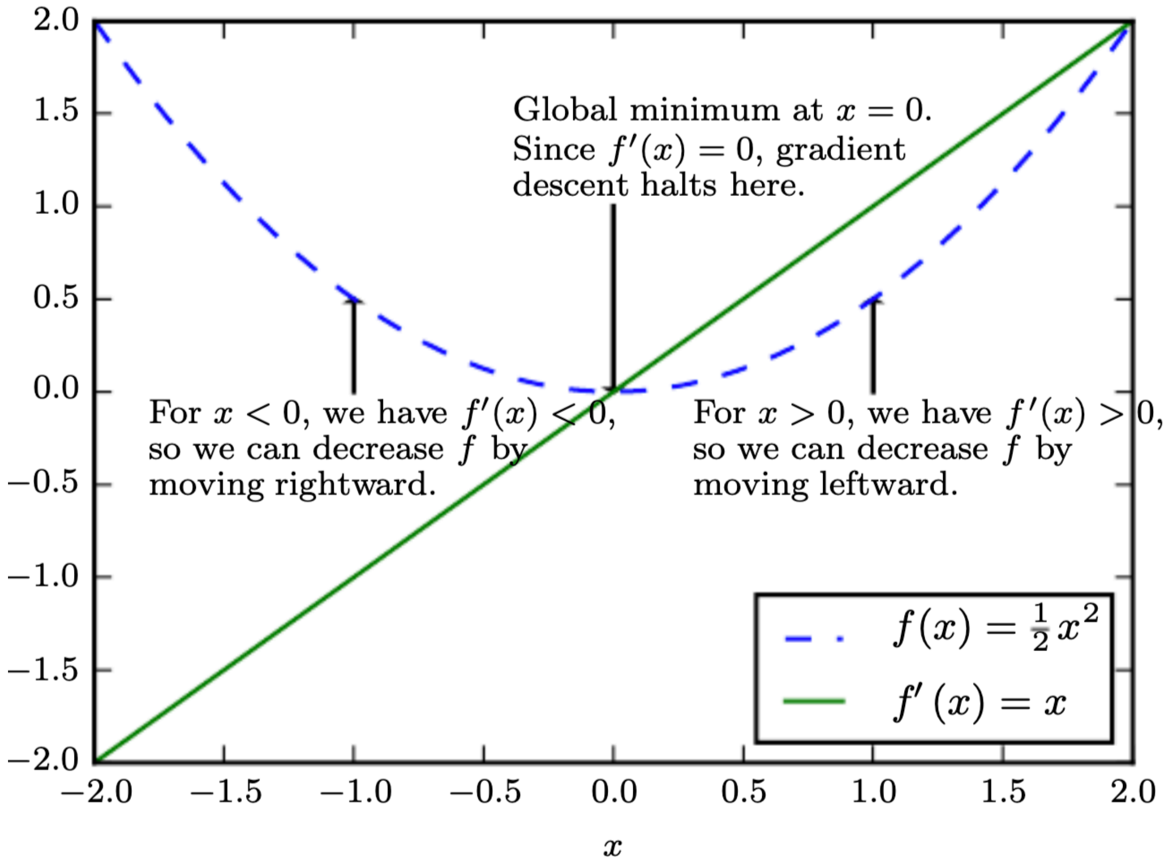
$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y - \hat{Y})^2$$

Background – Deep Neural Networks



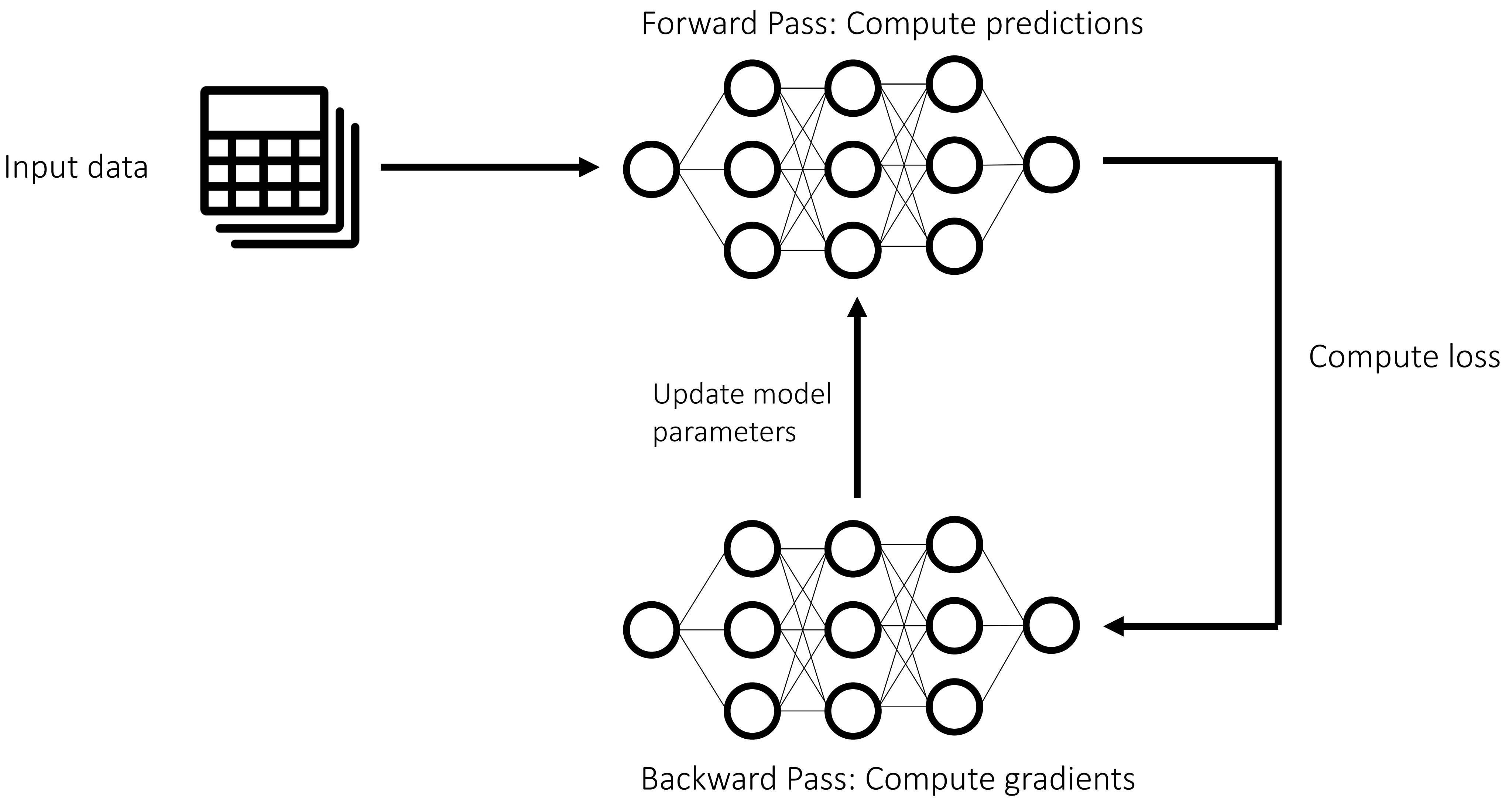


5



6

Background – Deep Neural Networks



Convolutional Neural Networks

2	3	2	2
1	5	3	7
4	5	2	1
2	5	9	4

4 x 4 x 3 image representation

$*$

1	0
0	1

2 x 2 kernel

$$2(1) + 3(0) + 1(0) + 5(1) = 7$$

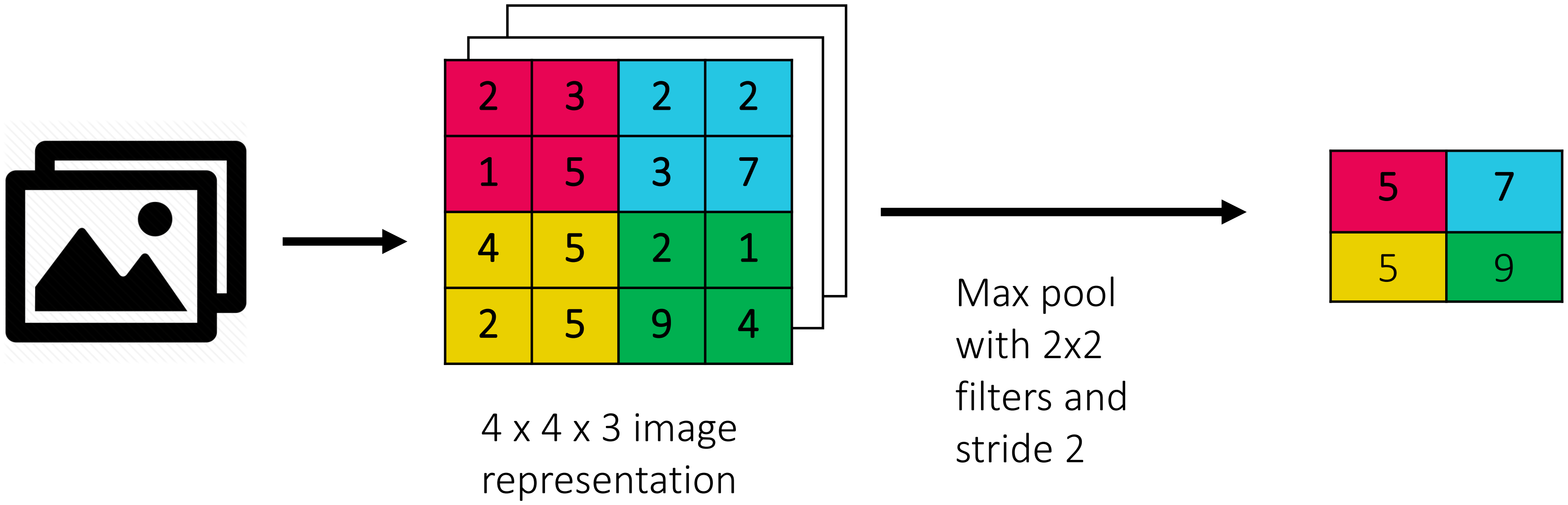
$$2(1) + 2(0) + 3(0) + 7(1) = 9$$

$$4(1) + 5(0) + 2(0) + 5(1) = 9$$

$$2(1) + 1(0) + 9(0) + 4(1) = 6$$

7	9
9	6

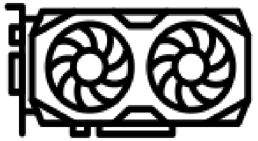
Convolutional Neural Networks – Max Pooling



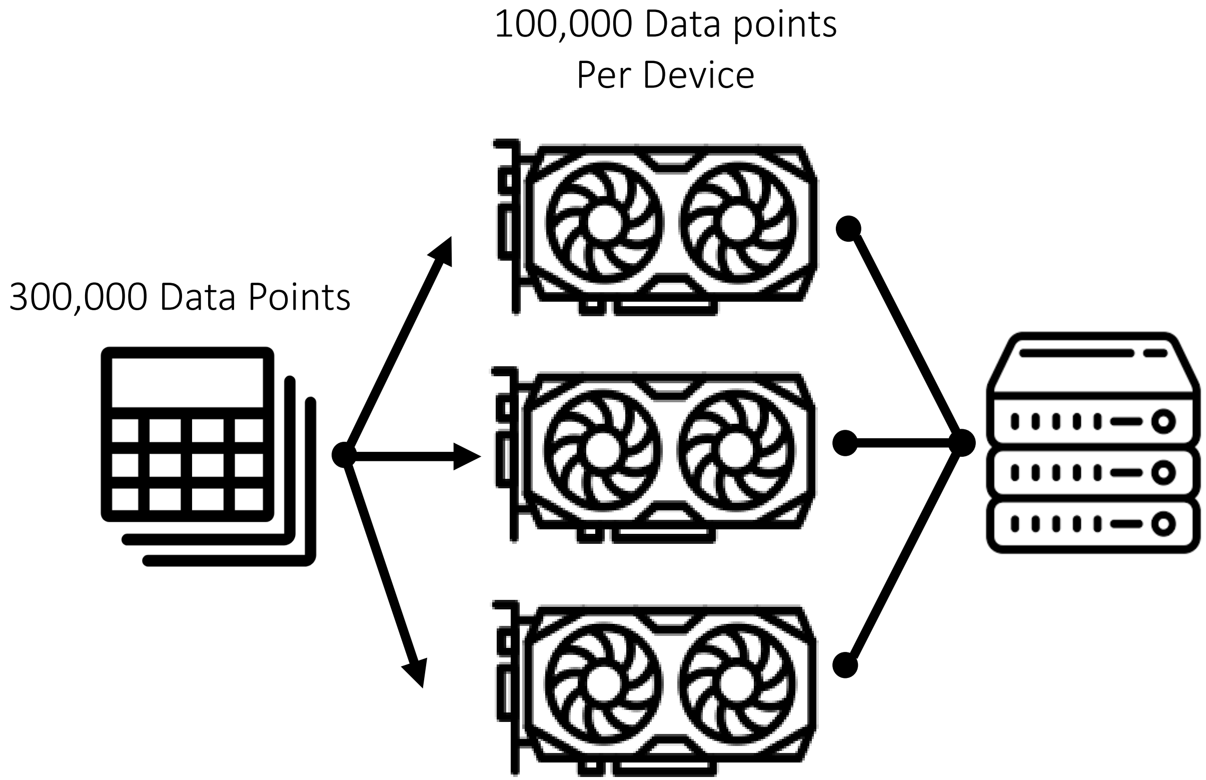
Data Parallelism

Equal sized subsets passed through each device

Each GPU has a replica of the model



Aggregate results from each subset

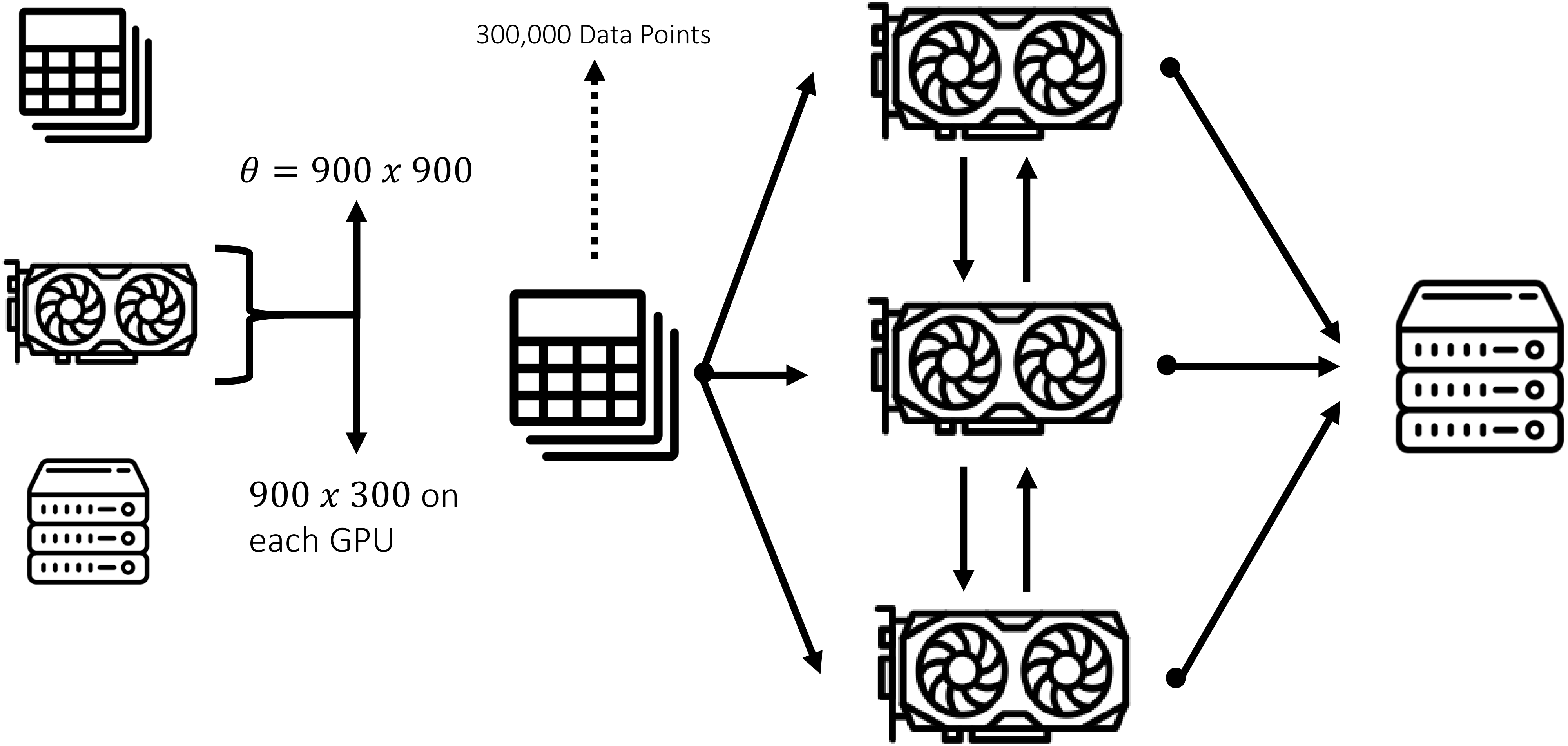


Model Parallelism

Full dataset passed through each device

Each GPU responsible for subset of weight matrix

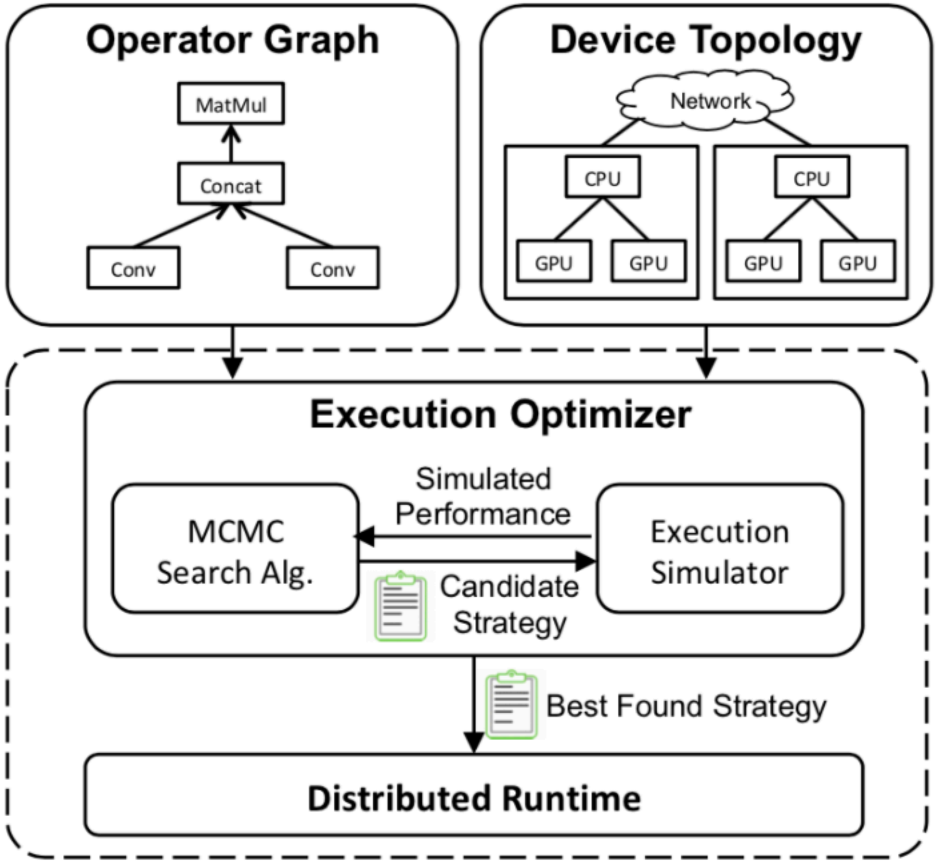
Aggregate results from each subset



Frameworks for Parallelism - FlexFlow

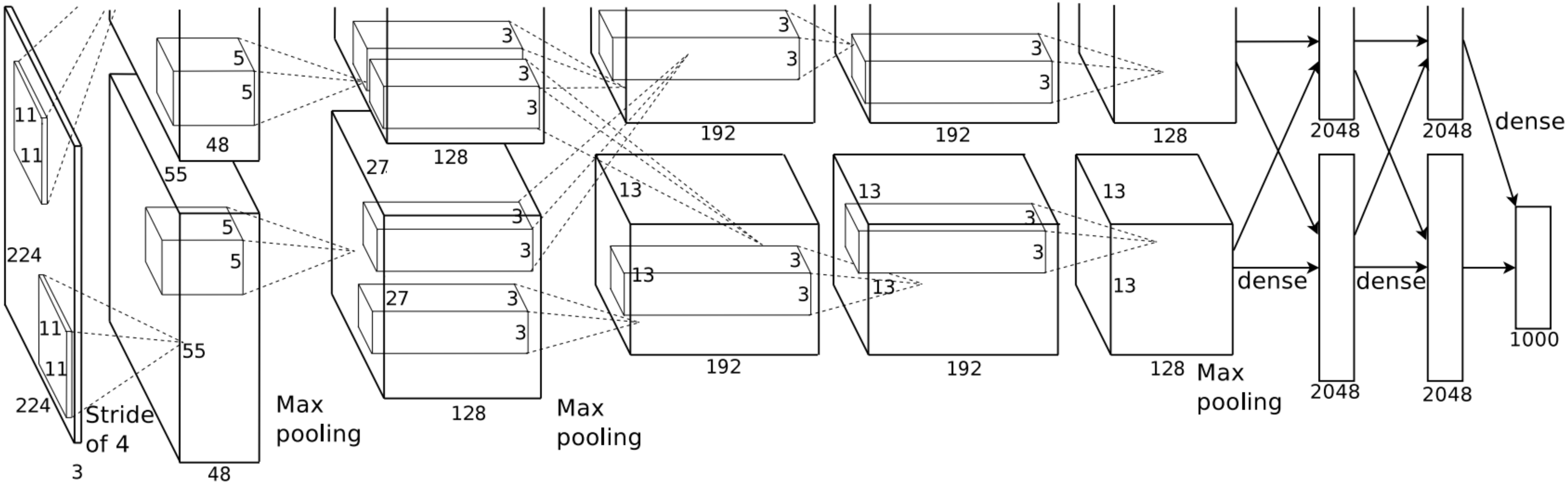
Approach	Dimensions	Hybrid	DNN Support
Data Parallelism	S		all
Model Parallelism	O, P		all
OWT	S, O, P	✓	AlexNet*
FlexFlow	S, O, A, P	✓	all

FlexFlow



9

AlexNet Architecture

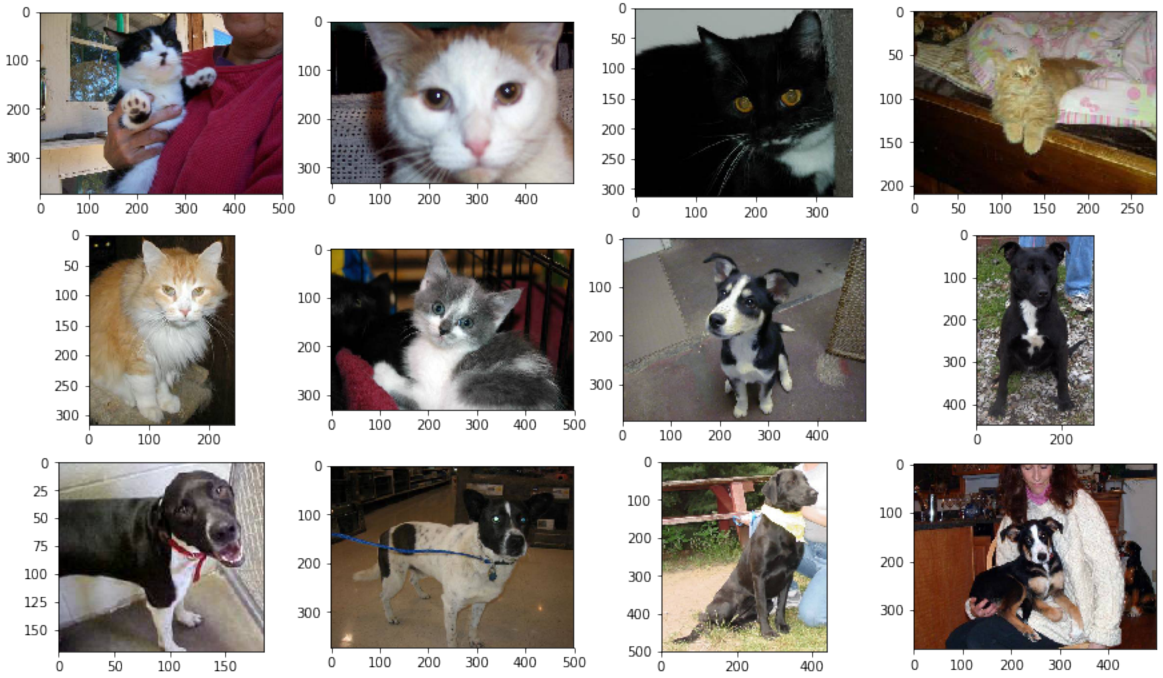


10

Parallelizing AlexNet - Datasets

Tiny ImageNet (Stanford)

Dogs vs. Cats (Kaggle)



Parallelizing AlexNet – Tiny ImageNet

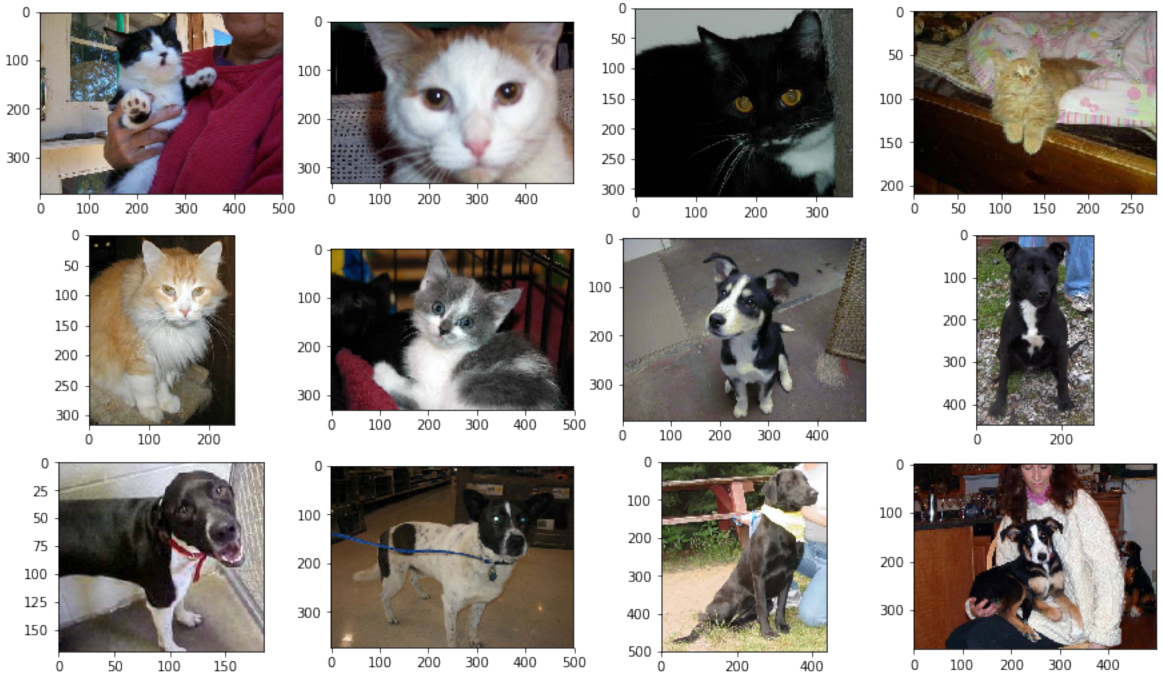
Tiny ImageNet (Stanford)

- Based on the original ImageNet dataset
- Developed for an image classification competition at Stanford
- 200 distinct classes based on Synsets
- 120,000 images (Each class has 500 training images, 50 validation images, and 50 test images)
- All images are 64 x 64 x 3

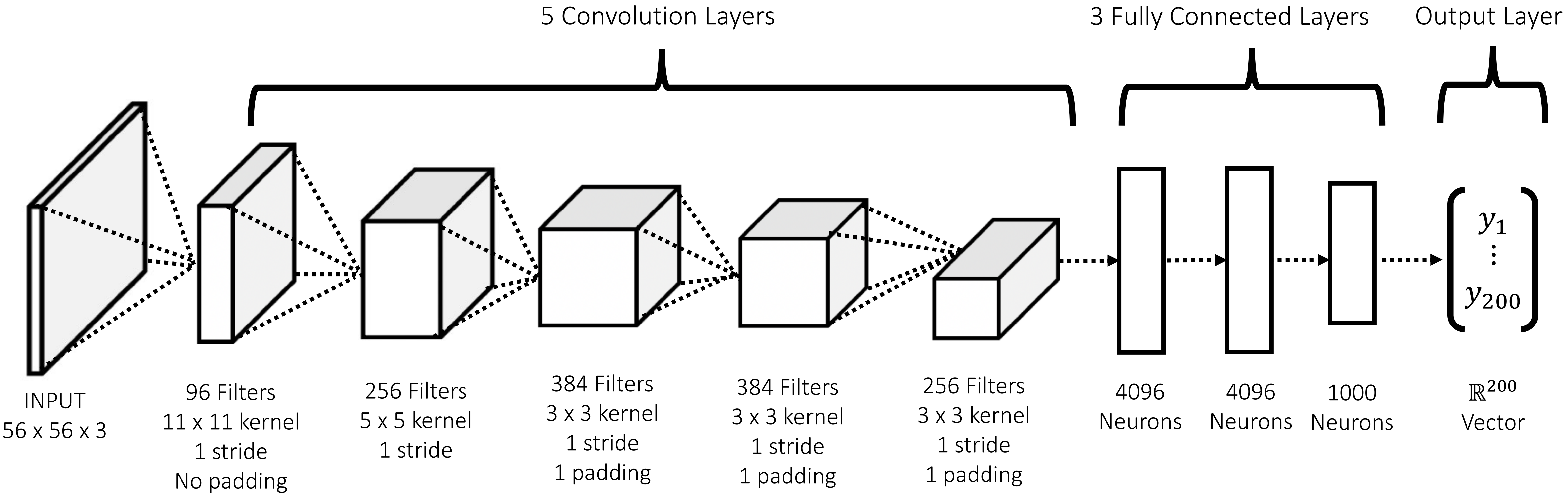
Parallelizing AlexNet – Dogs vs. Cats

- Subset of data from Asirra (Animal Species Image Recognition for Restricting Access)
- Microsoft Research + Asirra provide a subset from 3 million labelled images
- Kaggle competition in 2014
- 25,000 training images (2000 training, 800 validation)
- Images vary in size

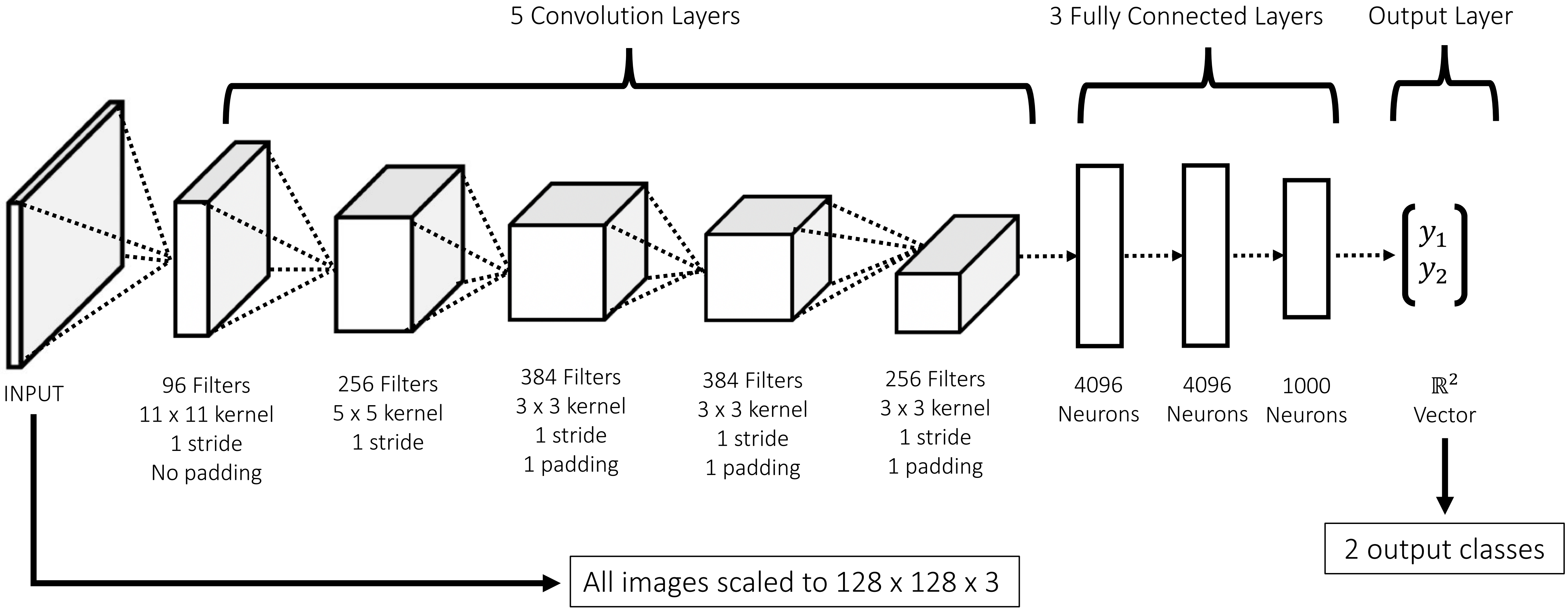
Dogs vs. Cats (Kaggle)



AlexNet – Tiny ImageNet



AlexNet – Dogs vs. Cats



Parallelization Experiments

- Train over 20 epochs on the same system (uniform resources)
- Training time as proxy for device efficiency
- Baseline (no parallelization configuration)
- Data Parallelism
- Model Parallelism
- One Weird Trick - expert designed method
- Compare with FlexFlow

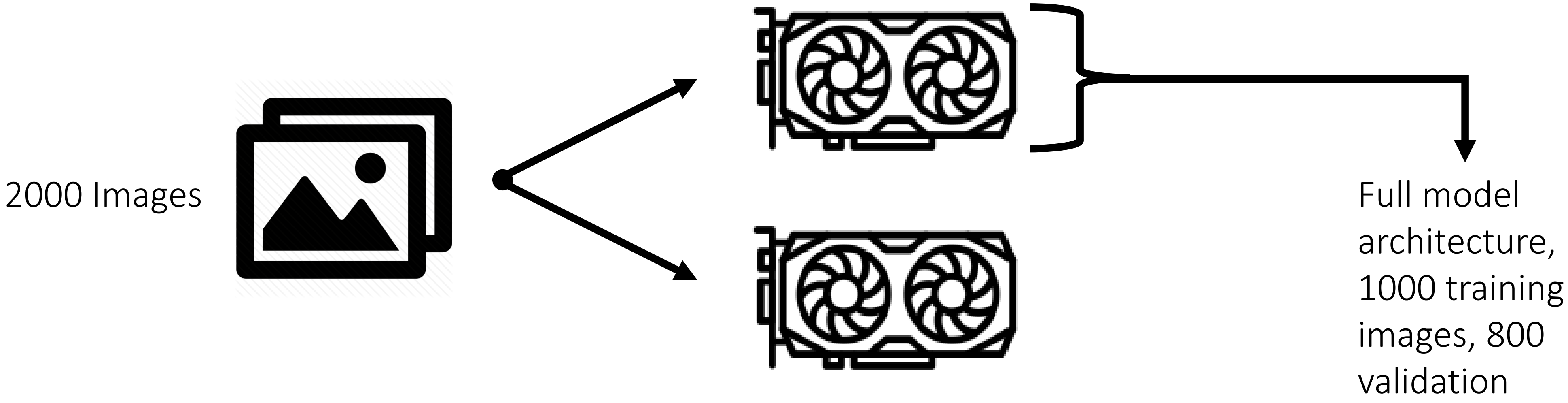
Parallelization Experiments – Technical Specifications

- Debian Image from Google Cloud Platform
- 2 vCPUs, 13 GB
- 2 Titan P100 Tesla GPUs
- CuDNN v10.0
- TensorFlow-gpu v.1.13.0
- Keras v2.2.4

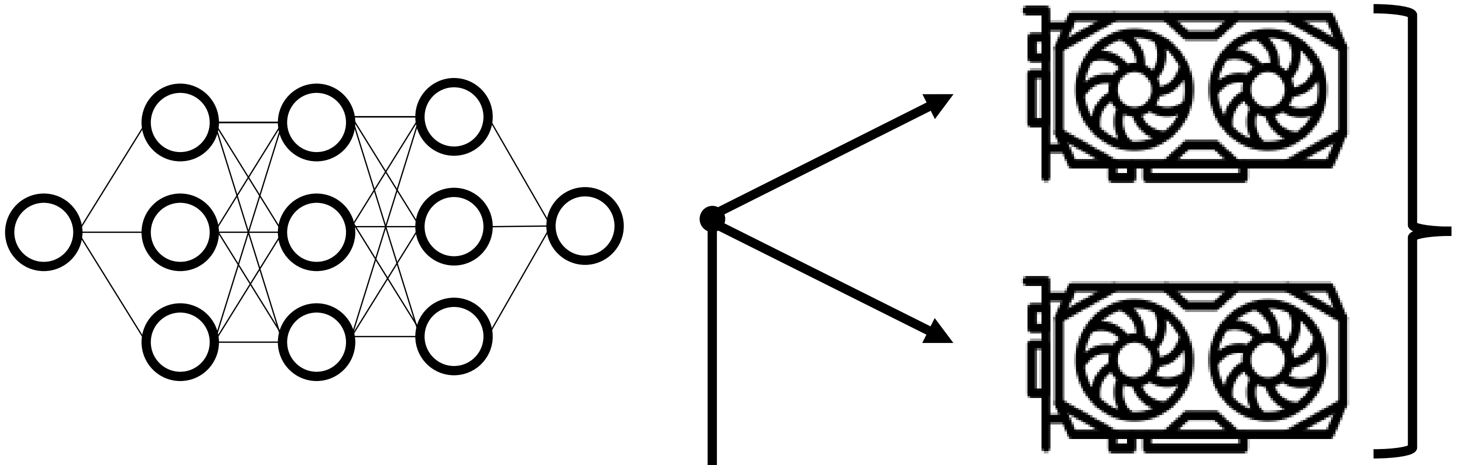
Google Cloud Platform

11, 12, 13, 14

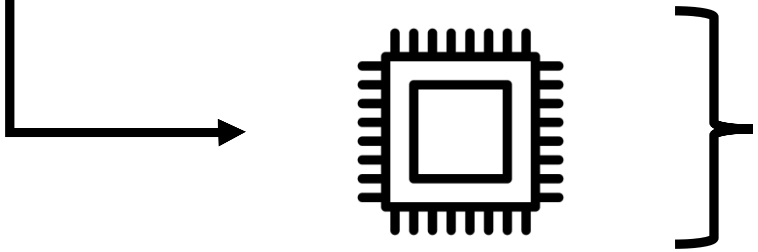
Parallelizing AlexNet – Data Parallelism



Parallelizing AlexNet – Model Parallelism

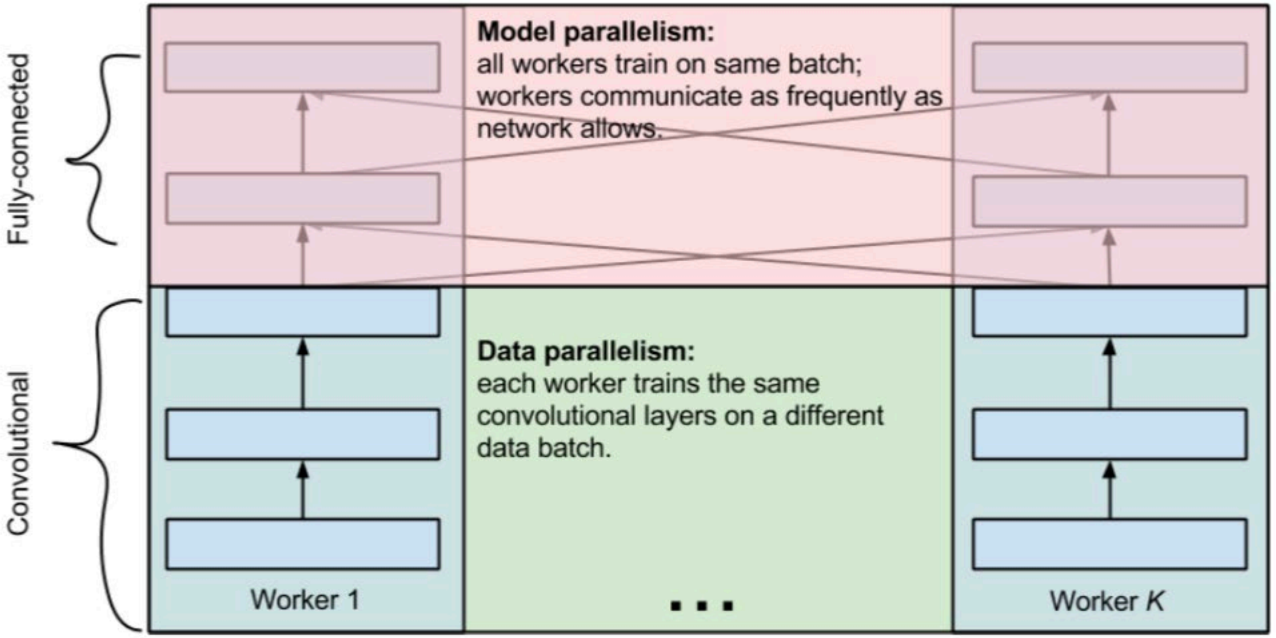


Convolution layers
Batch normalization
Fully connected layers



Padding
Max Pooling

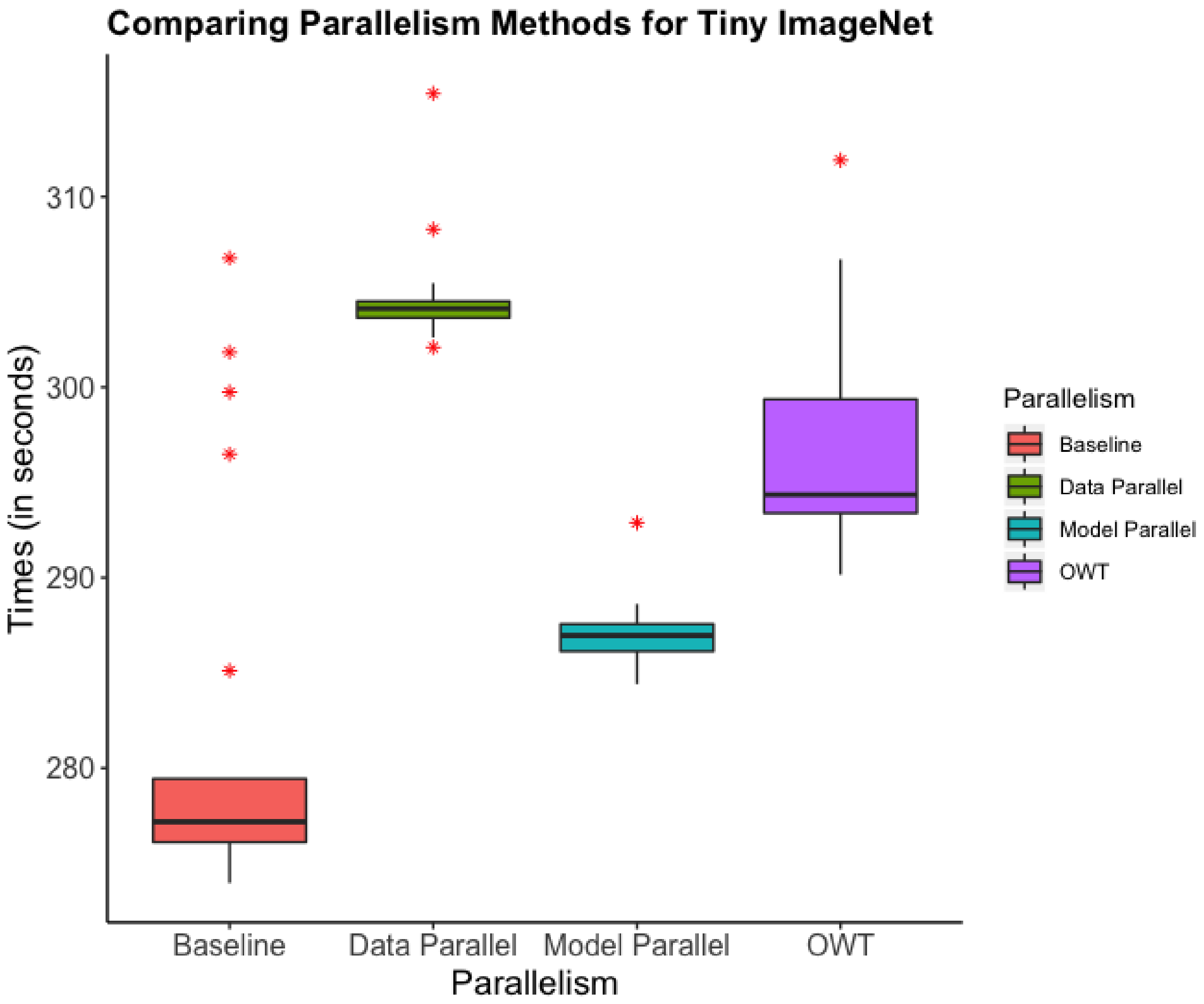
One Weird Trick for Parallelizing AlexNet



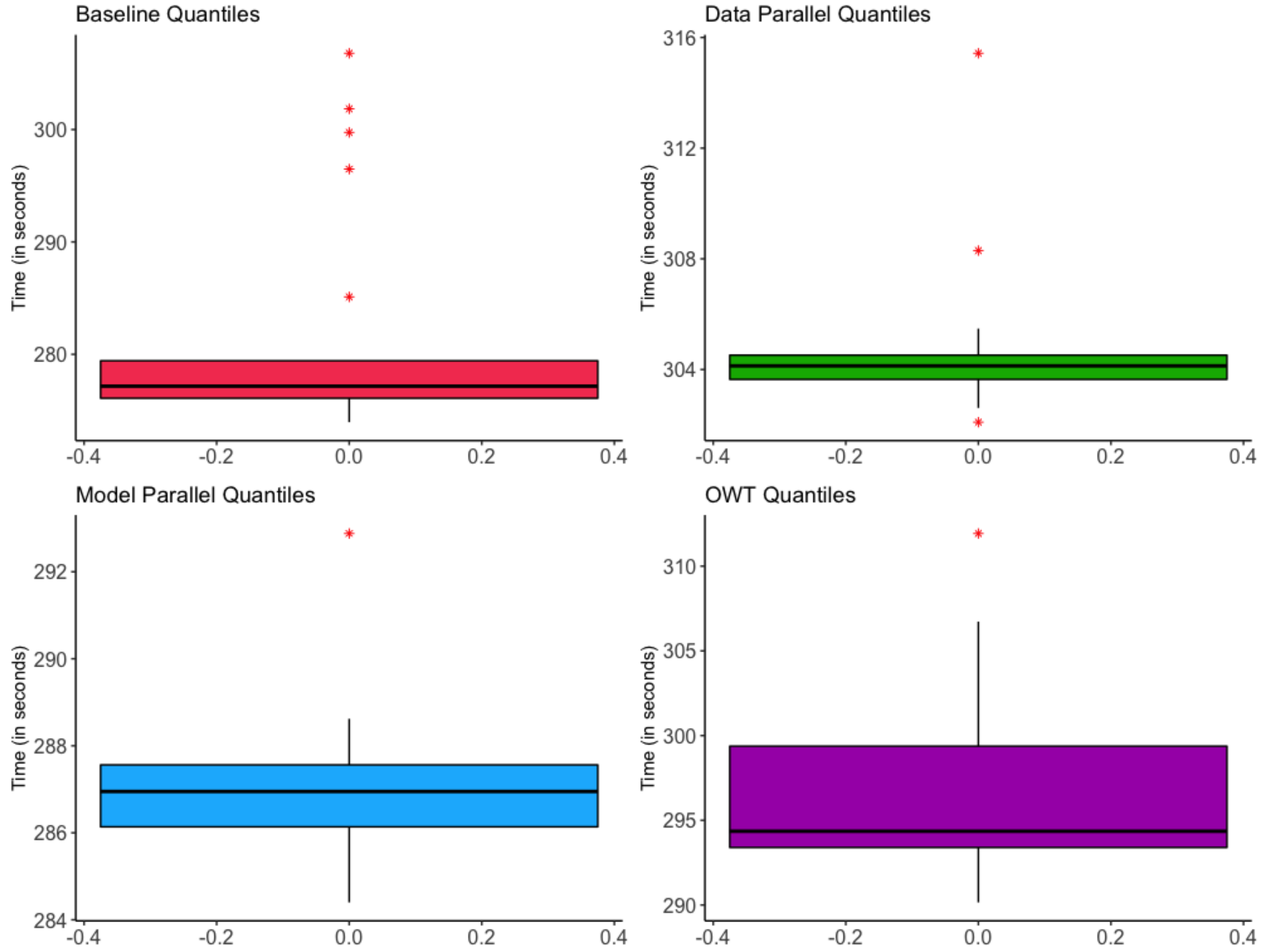
GPUs	Batch size	Cross-entropy	Top-1 error	Time	Speedup
1	(128, 128)	2.611	42.33%	98.05h	1x
2	(256, 256)	2.624	42.63%	50.24h	1.95x
2	(256, 128)	2.614	42.27%	50.90h	1.93x
4	(512, 512)	2.637	42.59%	26.20h	3.74x
4	(512, 128)	2.625	42.44%	26.78h	3.66x
8	(1024, 1024)	2.678	43.28%	15.68h	6.25x
8	(1024, 128)	2.651	42.86%	15.91h	6.16x

8

Parallelization Results – Tiny ImageNet



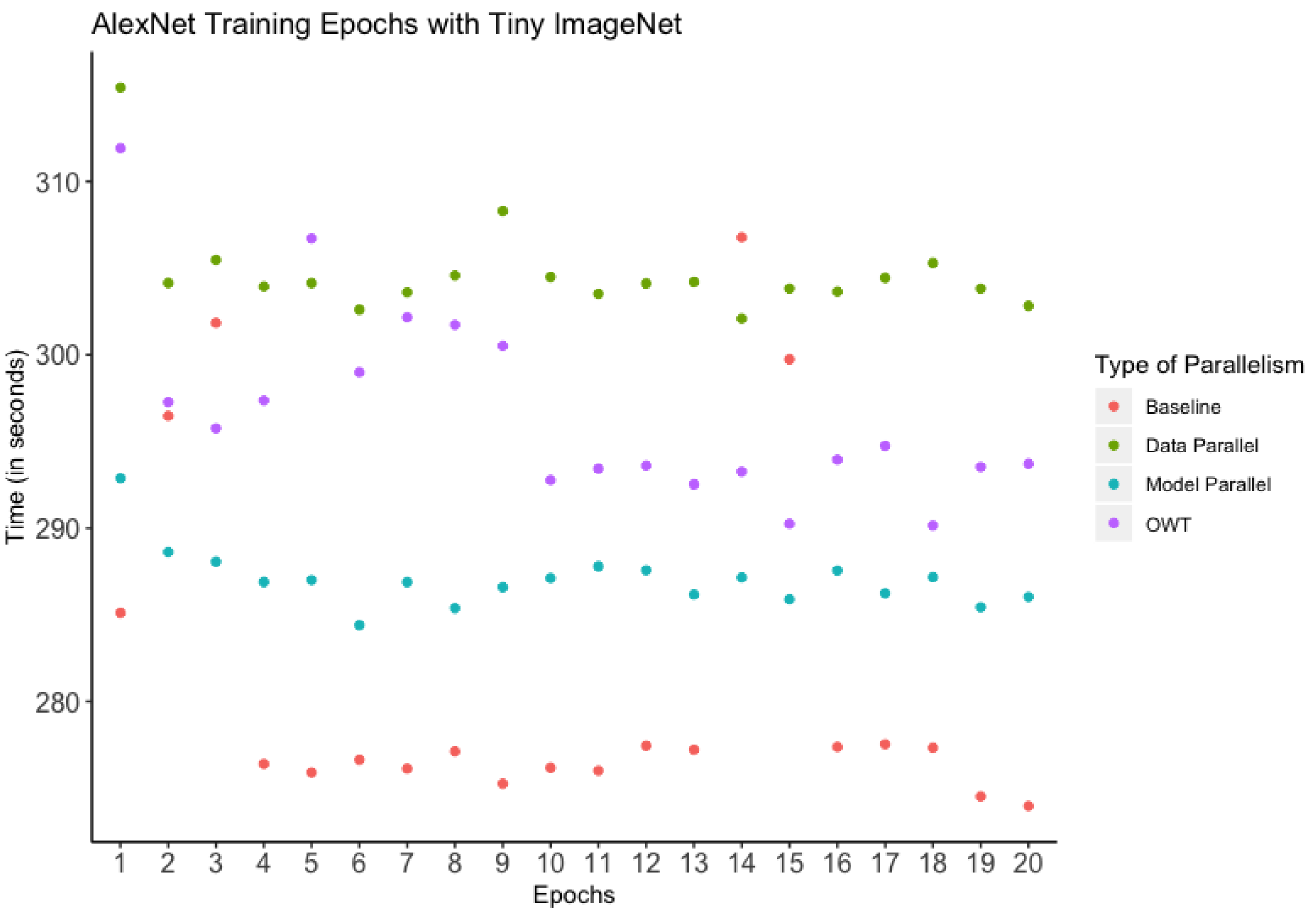
Parallelization Results – Tiny ImageNet



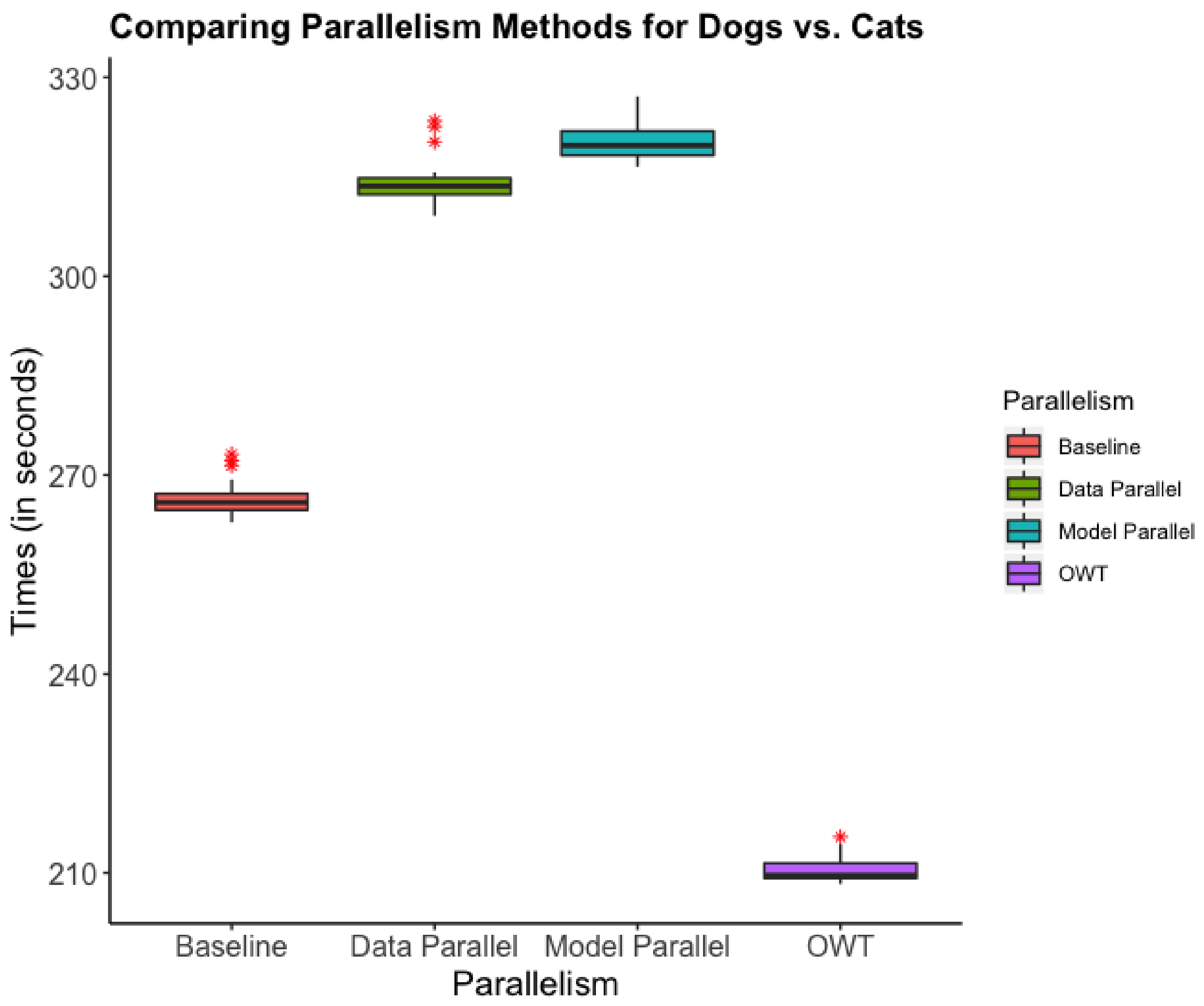
Parallelization Results – Tiny ImageNet

Tiny ImageNet

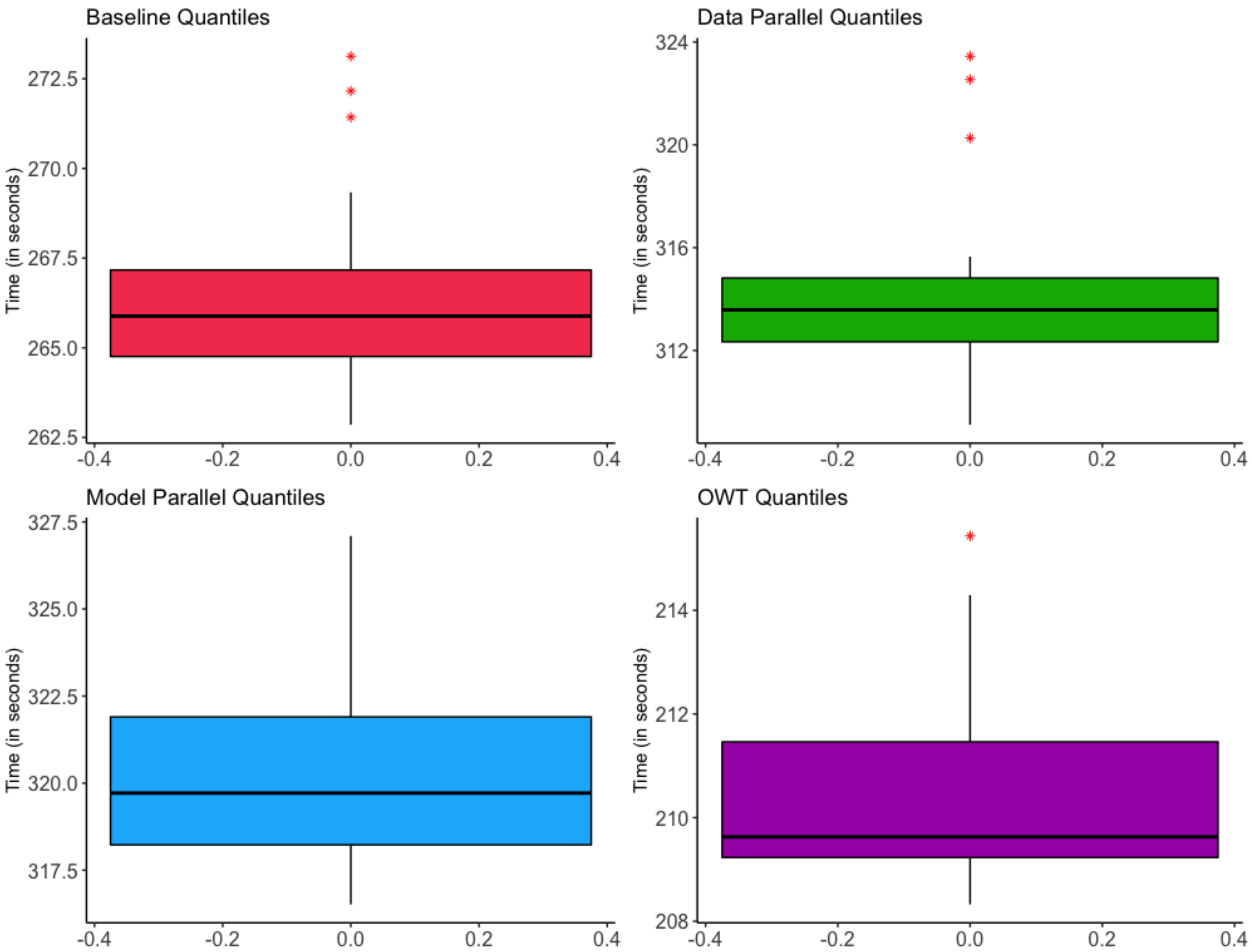
Benchmark	Average Epoch Time
Baseline	**281.7**
Data Parallel	304.7
Model Parallel	287.0*
OWT	296.7



Parallelization Results – Dogs vs. Cats



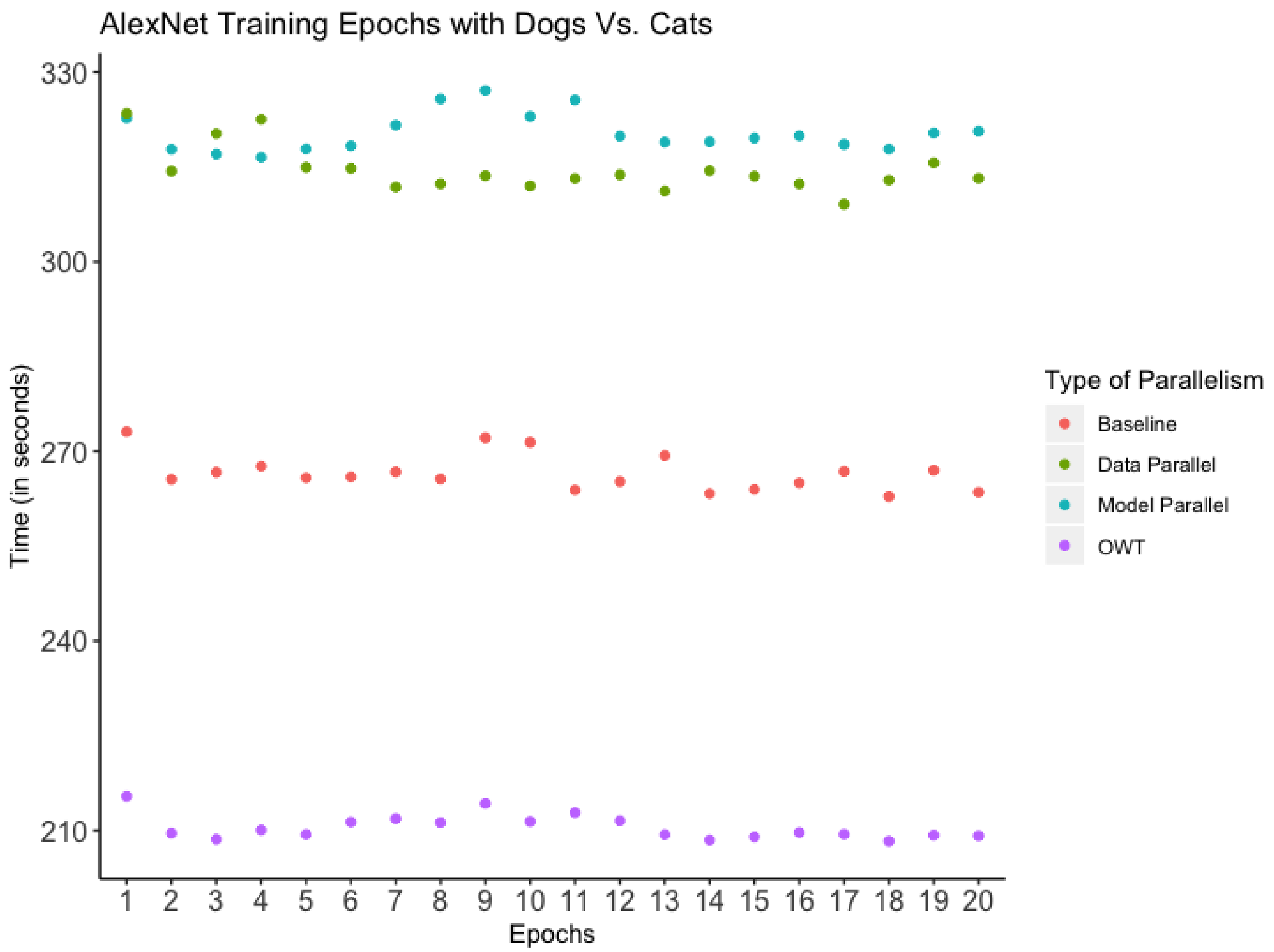
Parallelization Results – Dogs vs. Cats



Parallelization Results – Dogs vs. Cats

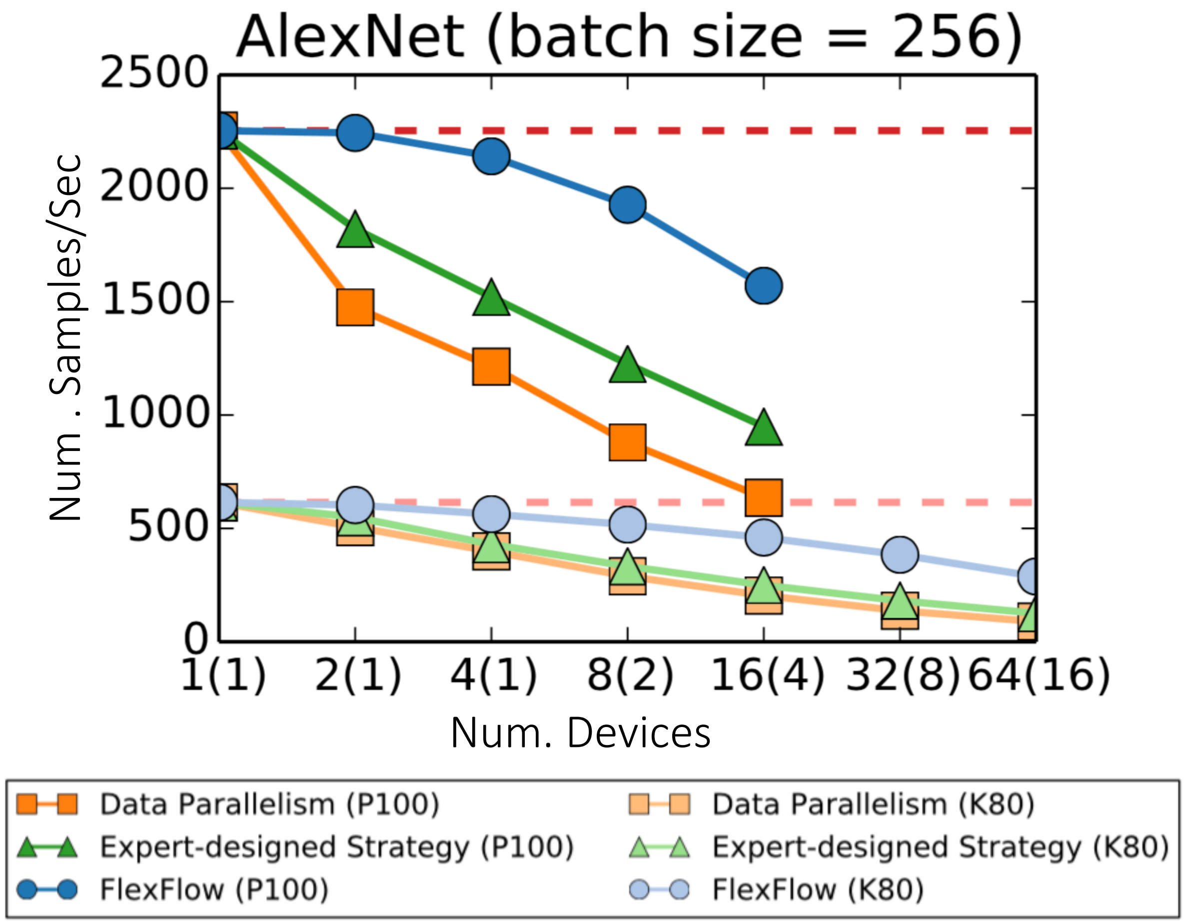
Dogs vs. Cats

Benchmark	Average Epoch Time
Baseline	266.6
Data Parallel	314.5
Model Parallel	320.4
OWT	210.5*



AlexNet and FlexFlow

Num. GPUs	AlexNet Full	AlexNet Delta	AlexNet Speedup
4	0.11	0.04	**2.9**×
8	0.40	0.13	**3.0**×
16	1.4	0.48	**2.9**×
32	5.3	1.8	**3.0**×
64	18	5.9	**3.0**×



9

Conclusions

- Deep neural networks can be accelerated through parallelism
- Expert designed methods are useful, but don’t scale
- Efficient device usage makes deep learning more accessible
- Frameworks are the future
- Parallelism as abstraction!

Acknowledgements

- Professor Andrea Vaccari
- Professor Michael Linderman
- Jonathan Kemp
- Friends, Family

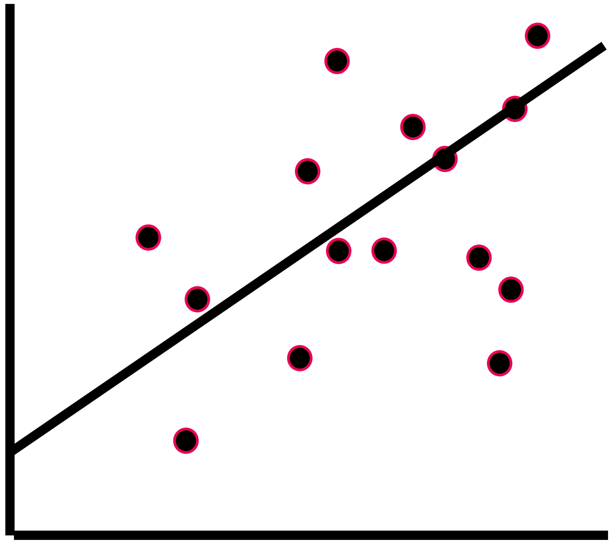
Sources

1. Deng, et al. *ImageNet: A Large-Scale Hierarchical Image Database*
2. Redmon, et al. *You Only Look Once: Unified, Real-Time Object Detection*
3. Ben-Nun, Hoefler, *You Only Look Once: Unified, Real-Time Object Detection*
4. Deepmind, https://deepmind.com/research/alphago/
5. Backpropogation, https://en.wikipedia.org/wiki/Backpropagation
6. Goodfellow, et al. *Deep Learning*
7. Convolution animations, https://github.com/vdumoulin/conv_arithmetic
8. Krizhevsky, *One weird trick for parallelizing convolutional neural networks*
9. Jia, et al. *Beyond Data and Model Parallelism for Deep Neural Networks*
10. Krizhevsky, et al. *ImageNet Classification with Deep Convolutional Neural Networks*
11. Google Cloud Platform, https://cloud.google.com/
12. TensorFlow, https://www.tensorflow.org/
13. Keras, https://keras.io/
14. Nvidia, https://www.nvidia.com/en-us/

Thank you! Questions?

Appendix A – ML Overview

Machine Learning Overview



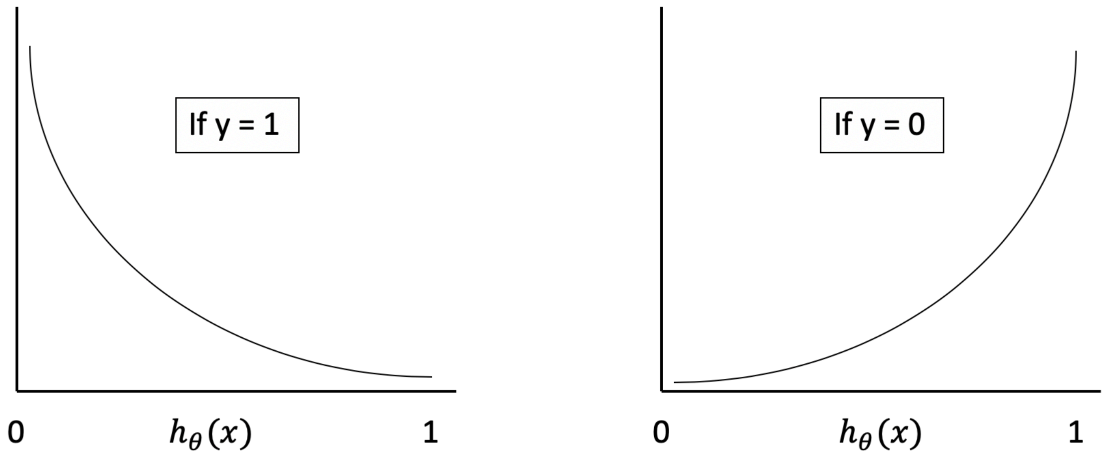
$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y - \hat{Y})^2$$

Hypothsis function: h_θ

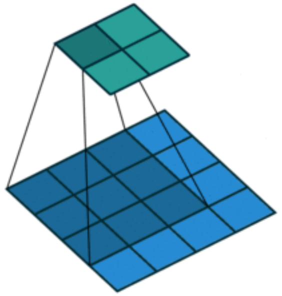
Training example: x_i

Prediction on example x_i: $h_\theta(x_i) = y_i$

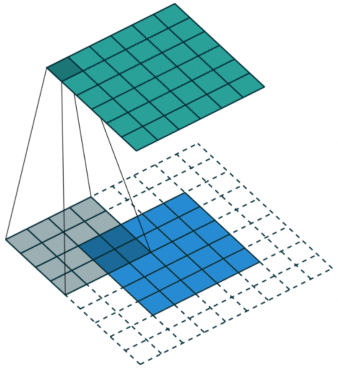
$$Cost(h_\theta(x_i), y_i) = \begin{cases} -\log(h_\theta(x_i)) & if\; y_i = 1 \\ -\log(1 - h_\theta(x_i)) & if\; y_i = 0 \end{cases}$$



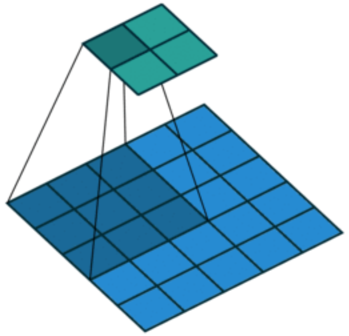
Convolutional Neural Networks – Padding & Stride



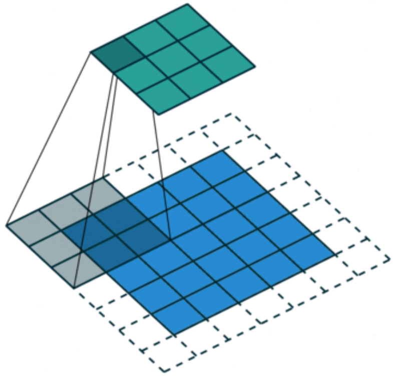
No Padding,
1 stride



Arbitrary Padding,
1 stride



No Padding,
2 stride



1 Padding,
2 stride

7

Appendix B – Results

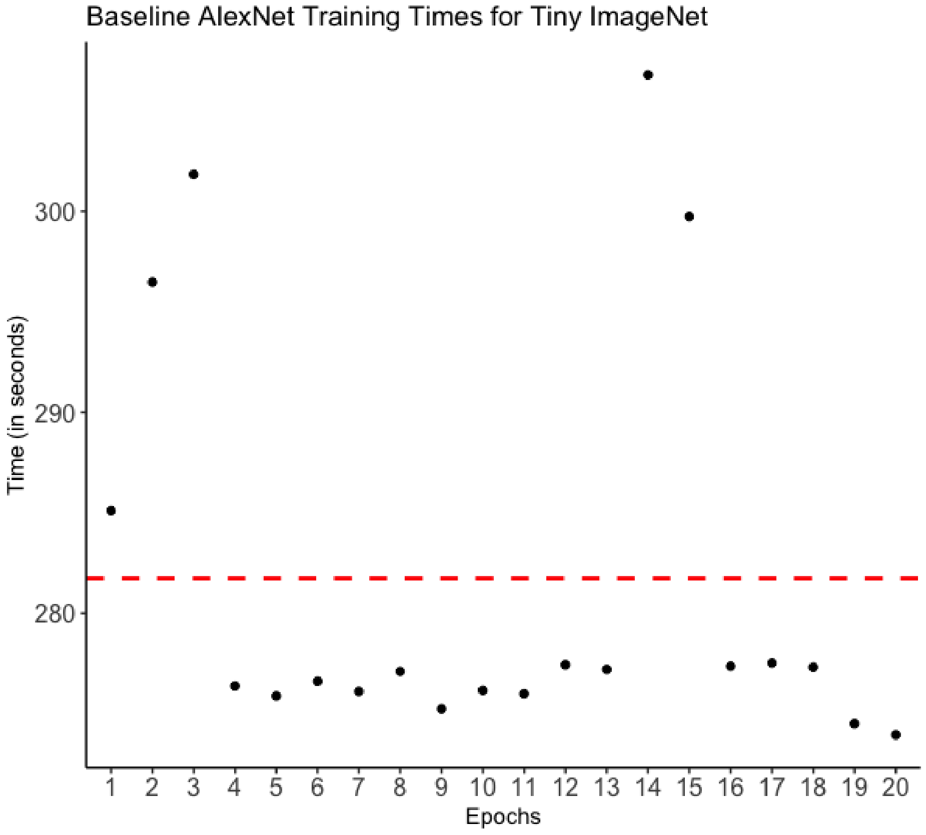
AlexNet Baseline with Tiny ImageNet

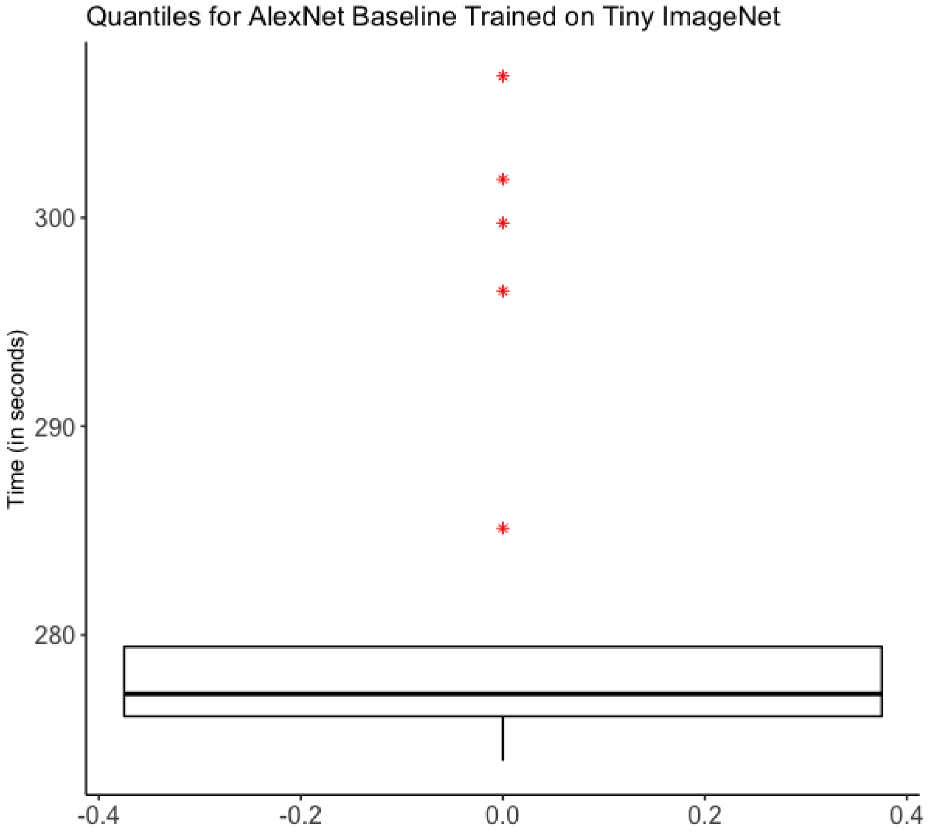
Epochs	1	2	3	4	5	6	7	8	9	10
Times	285.1	296.5	301.8	276.4	275.9	276.6	276.1	277.1	275.3	276.2

Epochs	11	12	13	14	15	16	17	18	19	20
Times	276.0	277.4	277.2	306.8	299.7	277.4	277.5	277.3	274.5	274.0

Minimum	1st Quartile	Median	Mean	3rd Quartile	Maximum
274.0	276.1	277.2	**281.7**	279.3	306.8

AlexNet Baseline with Tiny ImageNet





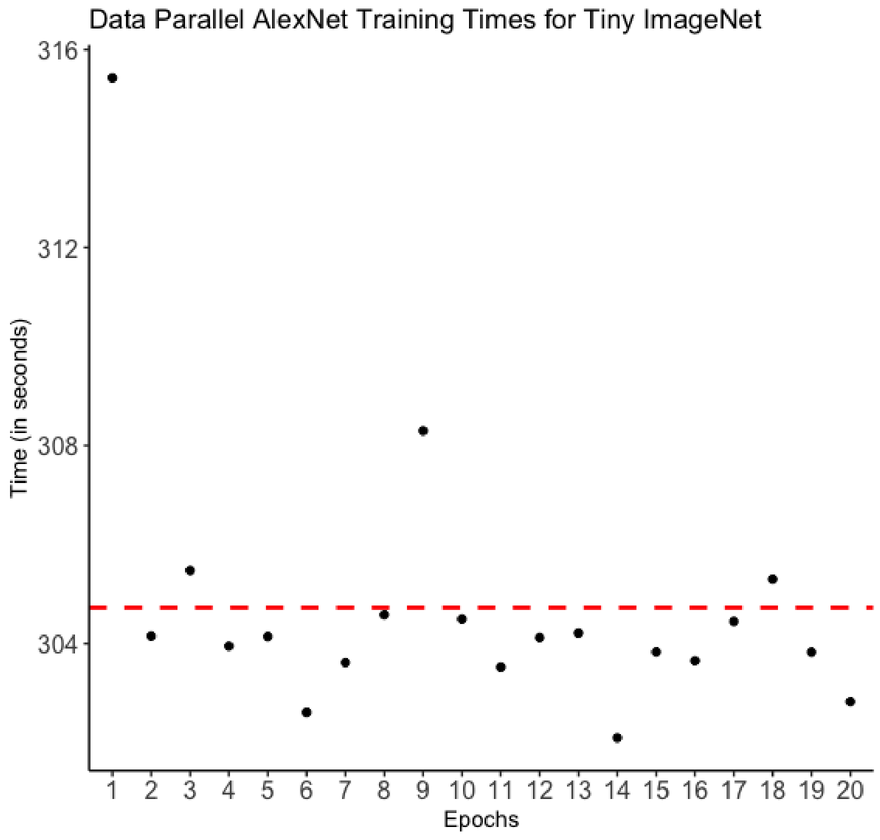
Data Parallel AlexNet with Tiny ImageNet

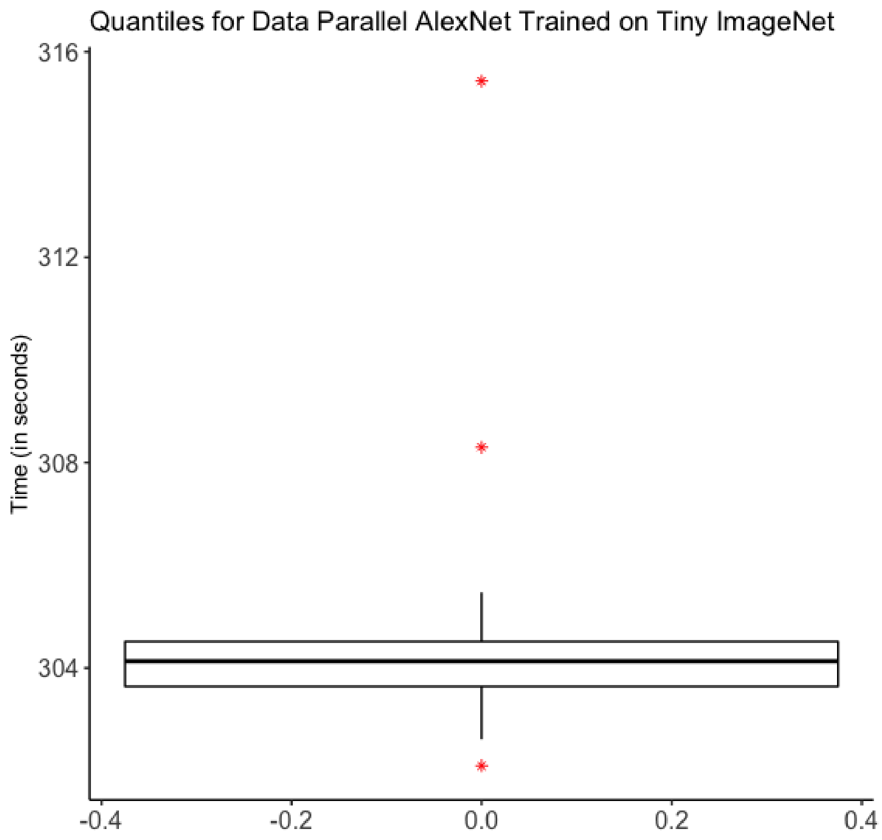
Epochs	1	2	3	4	5	6	7	8	9	10
Times	315.4	304.2	305.5	303.9	304.1	302.6	303.6	304.6	308.3	304.5

Epochs	11	12	13	14	15	16	17	18	19	20
Times	303.5	304.1	304.2	302.1	303.8	303.7	304.4	305.3	303.8	302.8

Minimum	1st Quartile	Median	Mean	3rd Quartile	Maximum
302.1	303.6	304.1	304.7	304.5	315.4

Data Parallel AlexNet with Tiny ImageNet





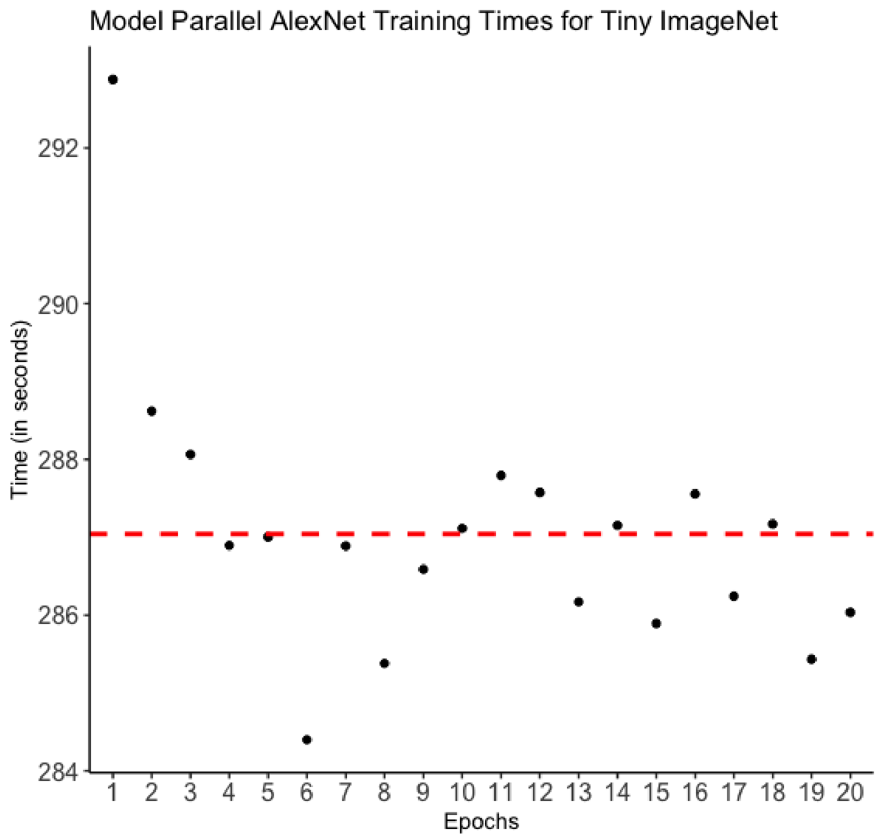
Model Parallel AlexNet with Tiny ImageNet

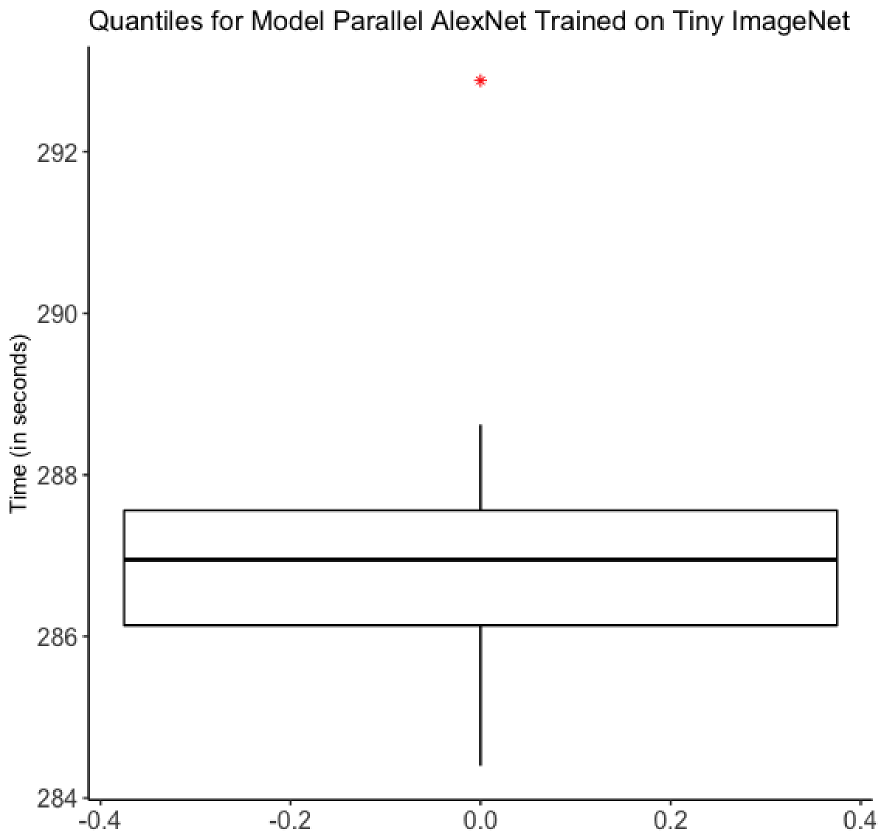
Epochs	1	2	3	4	5	6	7	8	9	10
Times	292.9	288.6	288.1	286.9	287.0	284.4	286.9	285.4	286.6	287.1

Epochs	11	12	13	14	15	16	17	18	19	20
Times	287.8	287.6	286.2	287.2	285.9	287.6	286.2	287.2	285.4	286.0

Minimum	1st Quartile	Median	Mean	3rd Quartile	Maximum
284.4	286.1	286.9	**287.0**	287.6	292.9

Model Parallel AlexNet with Tiny ImageNet





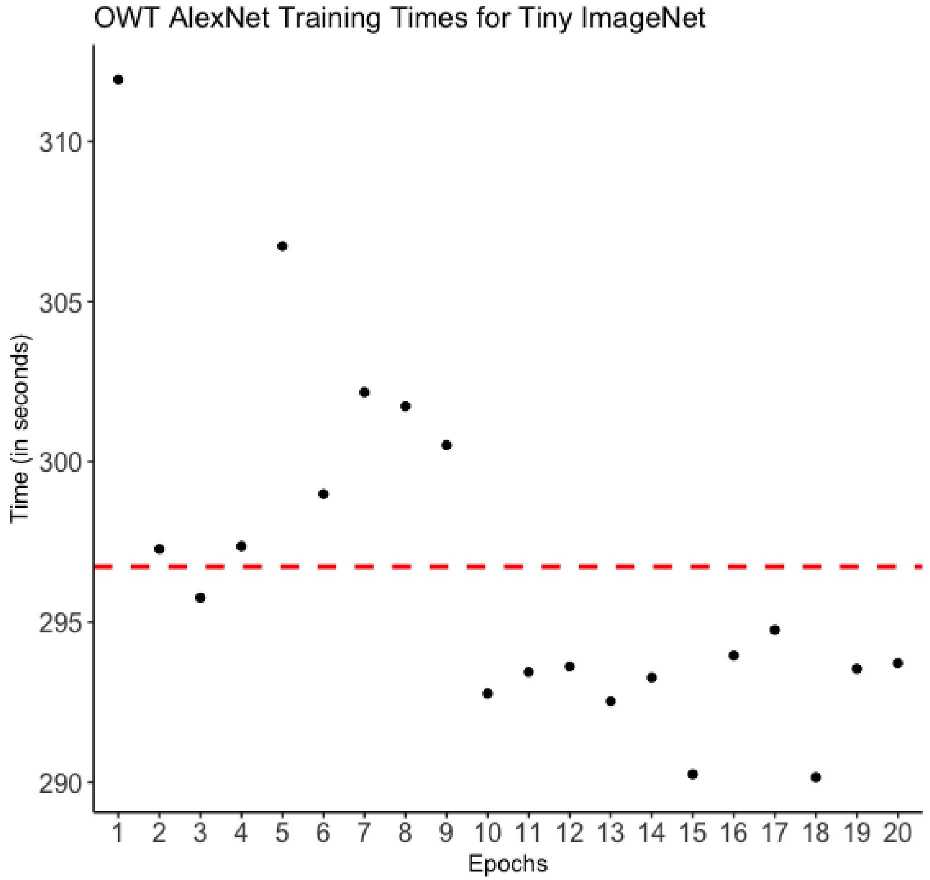
OWT AlexNet with Tiny ImageNet

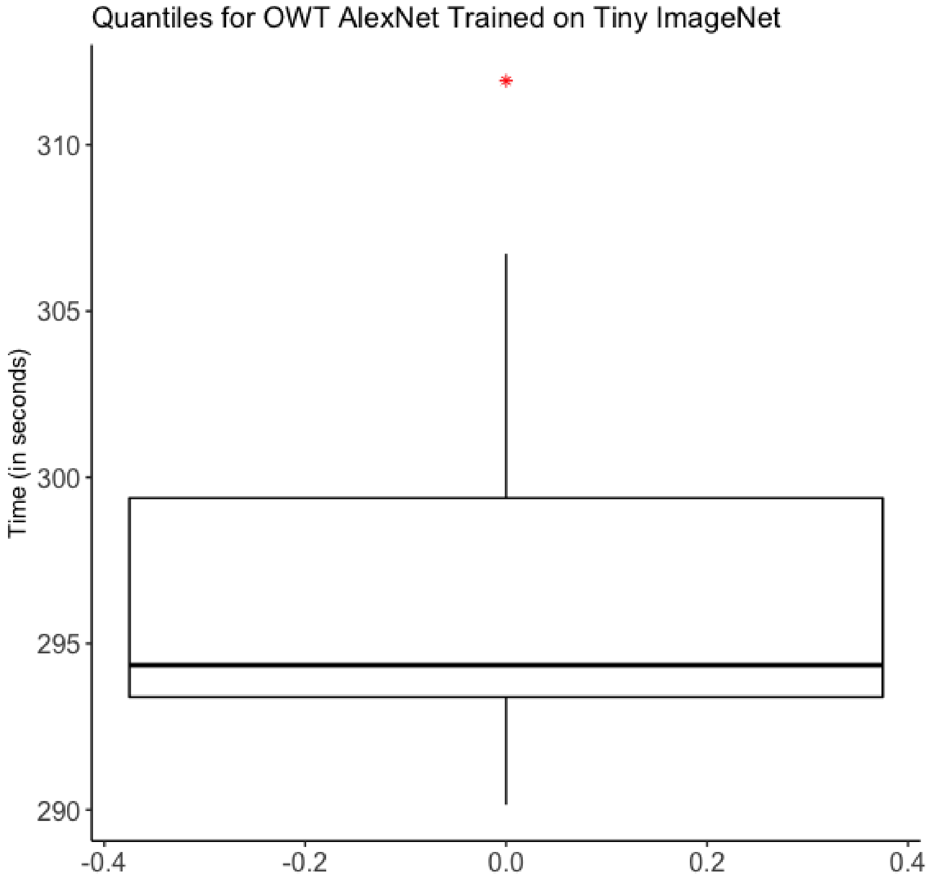
Epochs	1	2	3	4	5	6	7	8	9	10
Times	311.9	297.3	295.8	297.4	306.8	299.0	302.2	301.7	300.5	292.8

Epochs	11	12	13	14	15	16	17	18	19	20
Times	293.4	293.6	292.5	293.3	290.2	294.0	294.8	290.2	293.5	293.7

Minimum	1st Quartile	Median	Mean	3rd Quartile	Maximum
290.2	293.4	294.4	**296.7**	299.4	311.9

OWT AlexNet with Tiny ImageNet





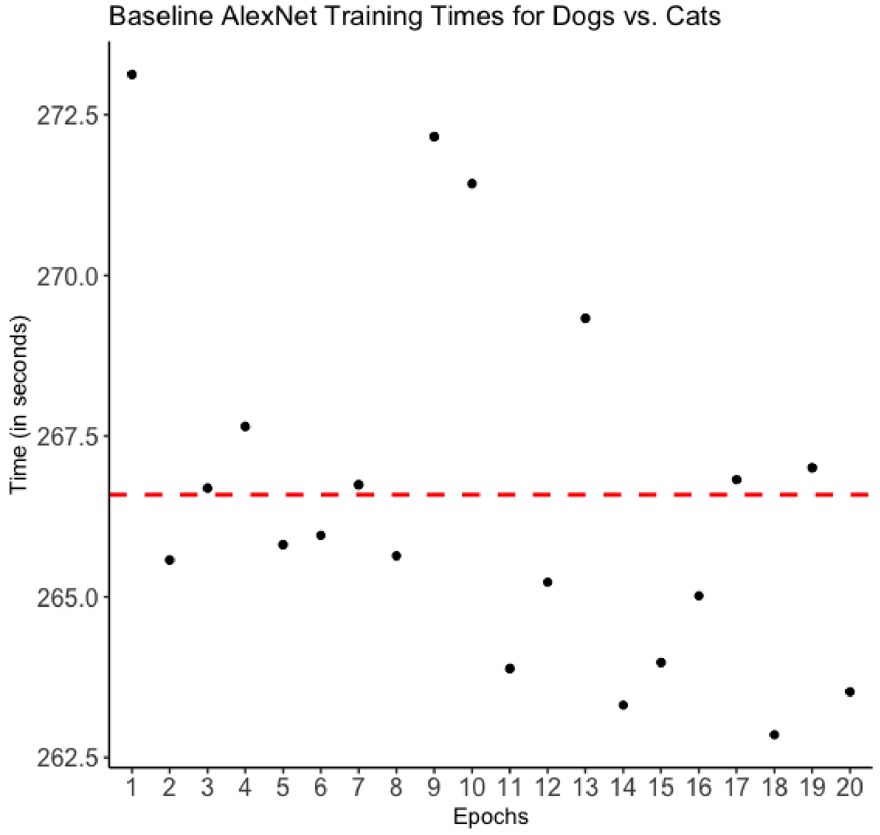
AlexNet Baseline with Dogs vs. Cats

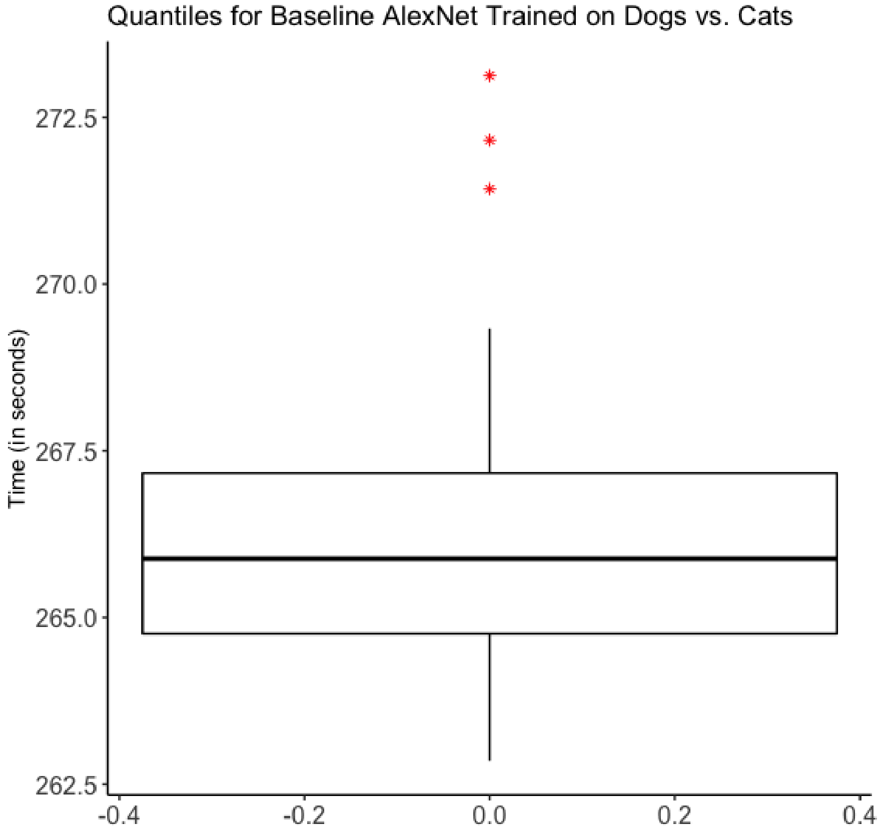
Epochs	1	2	3	4	5	6	7	8	9	10
Times	273.1	265.6	266.7	267.6	265.8	266.0	266.7	265.6	272.2	271.4

Epochs	11	12	13	14	15	16	17	18	19	20
Times	263.9	265.2	269.3	263.3	264.0	265.0	266.8	262.9	267.0	263.5

Minimum	1st Quartile	Median	Mean	3rd Quartile	Maximum
262.9	264.8	265.9	**266.6**	267.2	273.1

AlexNet Baseline with Dogs vs. Cats





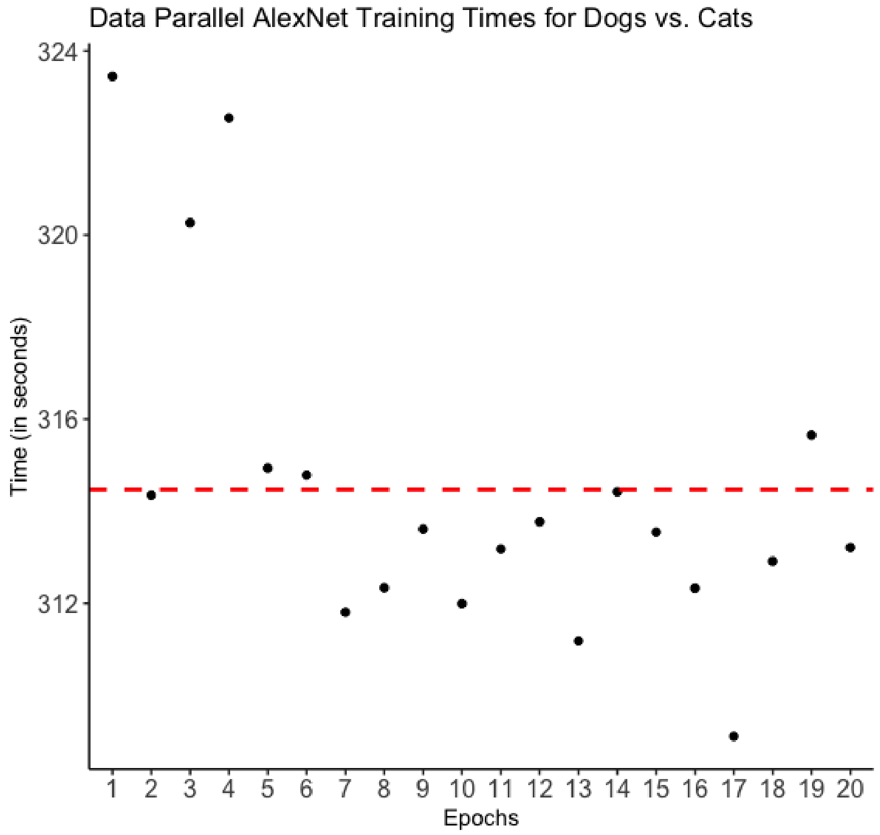
Data Parallel AlexNet with Dogs vs. Cats

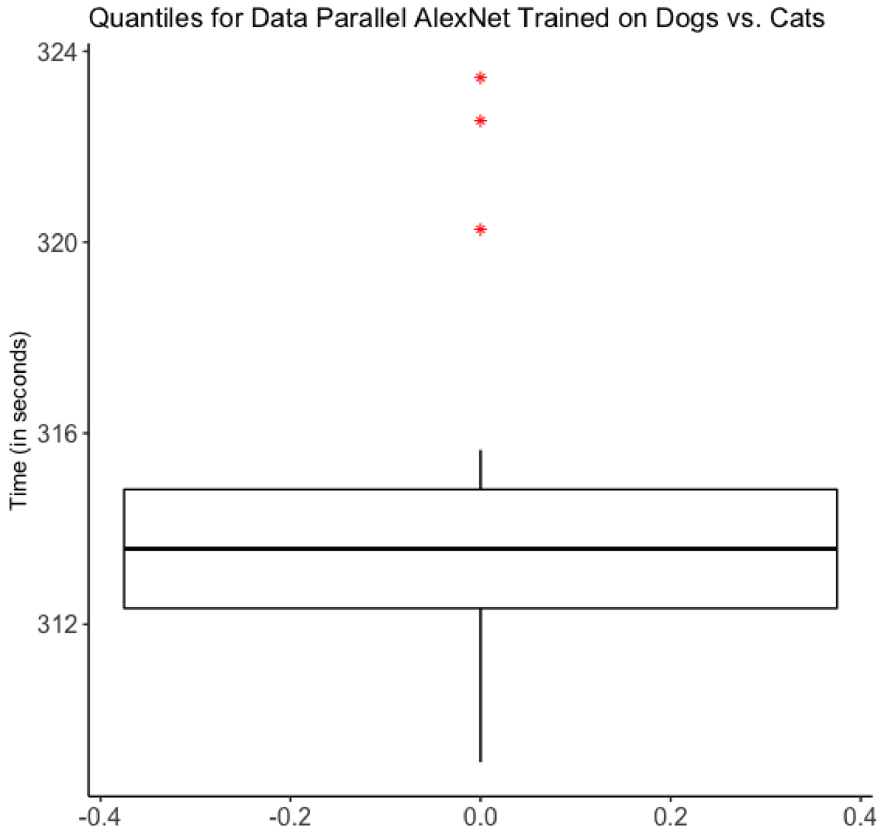
Epochs	1	2	3	4	5	6	7	8	9	10
Times	323.4	314.3	320.3	322.5	314.9	314.8	311.8	312.3	313.6	312.0

Epochs	11	12	13	14	15	16	17	18	19	20
Times	313.2	313.8	311.2	314.4	313.5	312.3	309.1	312.9	315.7	313.2

Minimum	1st Quartile	Median	Mean	3rd Quartile	Maximum
309.1	312.3	313.6	**314.5**	314.8	323.4

Data Parallel AlexNet with Dogs vs. Cats





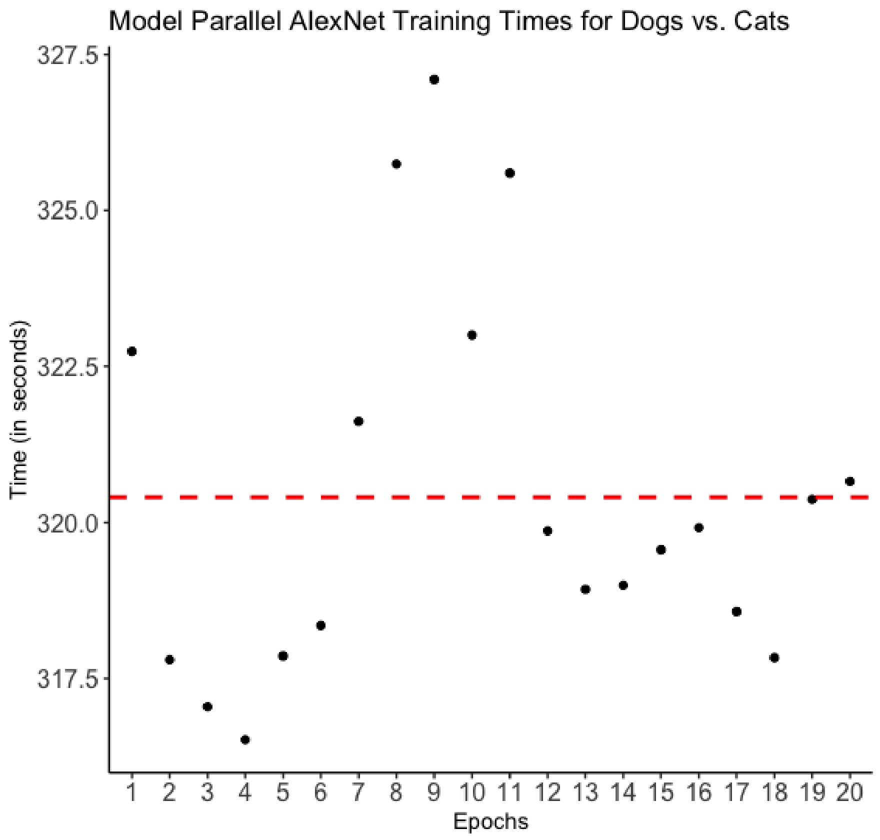
Model Parallel AlexNet with Dogs vs. Cats

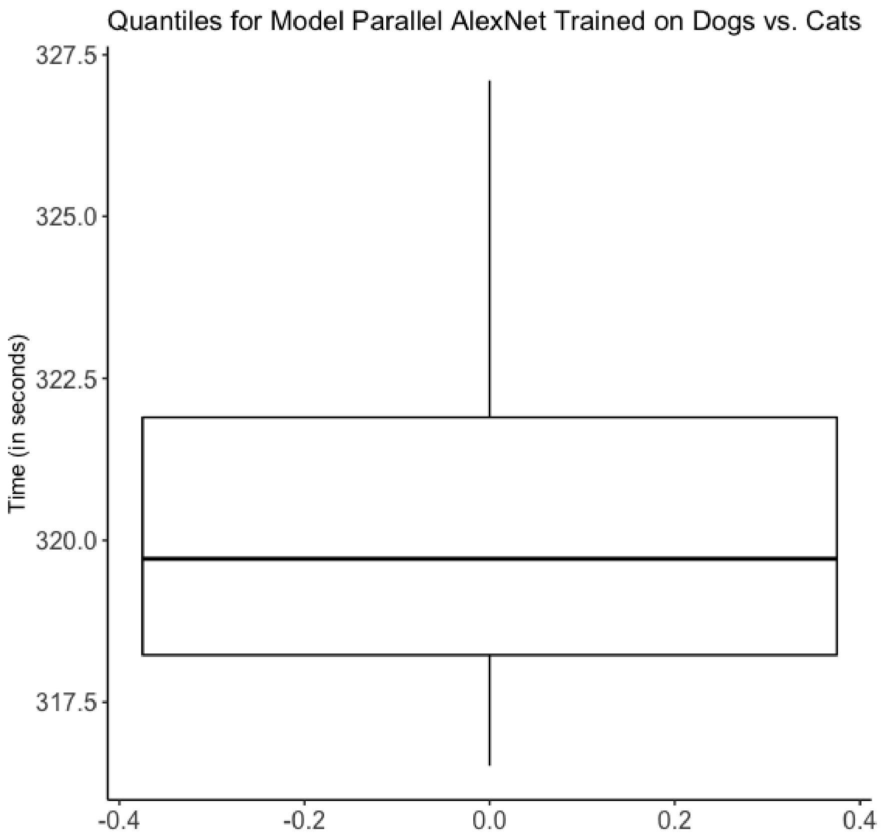
Epochs	1	2	3	4	5	6	7	8	9	10
Times	322.7	317.8	317.0	316.5	317.9	318.3	321.6	325.7	327.1	323.0

Epochs	11	12	13	14	15	16	17	18	19	20
Times	325.6	319.9	318.9	320.0	319.6	319.9	318.6	317.8	320.4	320.7

Minimum	1st Quartile	Median	Mean	3rd Quartile	Maximum
316.5	318.2	319.7	320.4	321.9	327.1

Model Parallel AlexNet with Dogs vs. Cats





OWT AlexNet with Dogs vs. Cats

Epochs	1	2	3	4	5	6	7	8	9	10
Times	215.4	209.6	208.6	210.1	209.4	211.3	211.9	211.3	214.3	211.4

Epochs	11	12	13	14	15	16	17	18	19	20
Times	212.8	211.6	209.4	208.5	209.0	209.7	209.4	208.3	209.3	209.2

Minimum	1st Quartile	Median	Mean	3rd Quartile	Maximum
208.3	209.2	209.6	**210.5**	211.5	215.4

OWT AlexNet with Dogs vs. Cats