Review

DI Librarie

Introduction

Programming Graphs Neurons

Fitting Models

Eight Data Points

Dense Nets

ConvNote

Project Improvement

Where We Are

Up Next

Deep Learning with TensorFlow Deep Learning — Units 7 & 8

Dr. Jon Krohn
jon@untapt.com

December 2nd, 2017



Review

DL Librarie

Introduction Symbolic Programming Graphs

Fitting Model

Eight Data Points

Fight Million Points

Dense Net

20..00

Project Improvemen

Where We Are Ten Tunings

Up Nex

- Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- Dense Nets
- 6 Convolutional Nets
- 7 Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



- DL Librarie
- Introductio
 Symbolic
 Programming
 Graphs
- Fitting Mode

 Eight Data Points

 Fight Million Point
- Dense Net
- 20..00

Project Improveme

Where We Are Ten Tunings

Up Ne:

- 1 Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- Dense Nets
- 6 Convolutional Nets
- 7 Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



- _____
- DL Librarie
- Introductio
 Symbolic
 Programming
 Graphs
 Neurons
- Fitting Model

 Eight Data Points

 Eight Million Points
- Dense Net

Project Improvement Where We Are

Up Ne

- 1 Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- Dense Nets
- 6 Convolutional Nets
- 7 Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



- TEVIEW
- DL Librarie
- Symbolic Programming Graphs
- Fitting Model

 Eight Data Points

 Eight Million Points
- D01100 140
- Project Improvement Where We Are
- Up Nex

- Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- 5 Dense Nets
- 6 Convolutional Nets
- 7 Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



- DL Librarie
- Introductio
 Symbolic
 Programming
 Graphs
- Fitting Model

 Eight Data Points

 Eight Million Points
- Delise Ne

ConvNets

Project Improvemer Where We Are Ten Tunings

Up Nex

- 1 Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow
 Symbolic Programming
 Programming TensorFlow Graphs
 Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- 5 Dense Nets
- 6 Convolutional Nets
- 7 Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



- ICVICW
- Introduction Symbolic Programming
- Fitting Mode Eight Data Points Eight Million Point
- Delise Me

Project Improveme Where We Are

Up Nex

- Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- 5 Dense Nets
- 6 Convolutional Nets
- Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



Review

L Librarie

Introduction Symbolic Programming Graphs Neurons

Fitting Model
Eight Data Points
Eight Million Point

20..00 . .

ConvNet

Project mprovemer Where We Are Ten Tunings

Up Ne:

- Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow
 Symbolic Programming
 Programming TensorFlow Graphs
 Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- 5 Dense Nets
- 6 Convolutional Nets
- Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



- I IGVIGW
- Introduction
- Symbolic Programming Graphs Neurons
- Fitting Model

 Eight Data Points

 Eight Million Point
- Delise Ne

Project Improveme Where We Are

Up Ne

- 1 Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- **5** Dense Nets
- 6 Convolutional Nets
- Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



Review

DL Librarie

Introduction Symbolic Programming Graphs

Fitting Mode Eight Data Points

Dense Net

Delise Met

Project Improveme

Ten Tunings

1 Review Take-Home Exercise

- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- 5 Dense Nets
- 6 Convolutional Nets
- 7 Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



Review

A Libraria

Introduction

Symbolic Programmin Graphs

Fitting Model

Eight Data Points

Donco Not

Project Improveme Where We Are

Un Nex

Assessing Your Deep Learning Project III





Review

DL Librarie

Introduction Symbolic Programming

Fitting Model:
Eight Data Points
Eight Million Points

_ -----

Project Improvemen

Ten Tunings

Up Nex

Assessing Your Deep Learning Project III

- Splitting your data
 - training set (80% for optimizing parameters)
 - validation set (10% for hyperparameters)
 - test set (10% don't touch yet!)
- ② Building and assessing architecture
 - get above chance (simplifying problem, if necessary)
 - do existing performance benchmarks exist?
 - if not, use a simple architecture as benchmark



Review

DL Librari

Introduction Symbolic Programming Graphs

Fitting Models
Eight Data Points
Eight Million Points

_ ----

Project Improvemen Where We Are

Up Nex

Assessing Your Deep Learning Project III

- Splitting your data
 - training set (80% for optimizing parameters)
 - validation set (10% for hyperparameters)
 - test set (10% don't touch yet!)
- 2 Building and assessing architecture
 - get above chance (simplifying problem, if necessary)
 - do existing performance benchmarks exist?
 - if not, use a simple architecture as benchmark



Review

DL Librari

Introductio
Symbolic
Programming
Graphs

Fitting Models
Eight Data Points
Eight Million Points

Dense Nei

ConvNote

Project Improvement Where We Are Ten Tunings

Up Nex

Assessing Your Deep Learning Project III

- Splitting your data
 - training set (80% for optimizing parameters)
 - validation set (10% for hyperparameters)
 - test set (10% don't touch yet!)
- 2 Building and assessing architecture
 - get above chance (simplifying problem, if necessary)
 - do existing performance benchmarks exist?
 - if not, use a simple architecture as benchmark



Review

Assessing Your Deep Learning Project III

- Splitting your data
 - training set (80% for optimizing parameters)
 - validation set (10% for hyperparameters)
 - test set (10% don't touch yet!)



Review

DL Librarie

Introductio
Symbolic
Programming
Graphs

Fitting Models

Eight Data Points

Eight Million Points

Dense Net

20..00 . 101

Project Improvemen Where We Are Ten Tunings

Up Nex

Assessing Your Deep Learning Project III

- Splitting your data
 - training set (80% for optimizing parameters)
 - validation set (10% for hyperparameters)
 - test set (10% don't touch yet!)
- 2 Building and assessing architecture
 - get above chance (simplifying problem, if necessary)
 - do existing performance benchmarks exist?
 - if not, use a simple architecture as benchmark



Review

Assessing Your Deep Learning Project III

- Splitting your data
 - training set (80% for optimizing parameters)
 - validation set (10% for hyperparameters)
 - test set (10% don't touch yet!)
- 2 Building and assessing architecture
 - get above chance (simplifying problem, if necessary)



Review

DL Librario

Introductio
Symbolic
Programming
Graphs

Fitting Models

Eight Data Points

Eight Million Points

Dense Net

Delise Net

Project Improvemen Where We Are Ten Tunings

Up Next

Assessing Your Deep Learning Project III

- Splitting your data
 - training set (80% for optimizing parameters)
 - validation set (10% for hyperparameters)
 - test set (10% don't touch yet!)
- 2 Building and assessing architecture
 - get above chance (simplifying problem, if necessary)
 - do existing performance benchmarks exist?
 - if not, use a simple architecture as benchmark



Review

DL Librarie

Introduction Symbolic Programming Graphs

Fitting Models
Eight Data Points
Eight Million Points

Dense Net

Canullata

Project Improvemer Where We Are Ten Tunings

Up Next

Assessing Your Deep Learning Project III

- Splitting your data
 - training set (80% for optimizing parameters)
 - validation set (10% for hyperparameters)
 - test set (10% don't touch yet!)
- 2 Building and assessing architecture
 - get above chance (simplifying problem, if necessary)
 - do existing performance benchmarks exist?
 - if not, use a simple architecture as benchmark



DI Libraria

DL Libraries

Introduction
Symbolic
Programming
Graphs
Neurons

Fitting Model

Eight Data Points

Fight Million Points

Dense Net

ConvNets

Project Improvemei Where We Are

Up Nex

- Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- Dense Nets
- 6 Convolutional Nets
- 7 Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



Fitting Models

Eight Data Points

Dense Ne

ConvNote

Project Improvement Where We Are

Up Ne

Leading DL Libraries

	Caffe	Torch	MXNet	TensorFlow
Language	Python, Matlab	Lua, C	Python, R, C++ Julia, Matlab JavaScript, Go Scala, Perl	Python, R, C++ C, Java, Go
Programming Style	Symbolic	Imperative	Imperative	Symbolic
Parallel GPUs: Data	Yes	Yes	Yes	Yes
Parallel GPUs: Model		Yes	Yes	Yes
Pre-Trained Models	Model Zoo	ModelZoo	Model Zoo	github.com/tensorflow/ models
For RNNs				Best
High-Level APIs		PyTorch	in-built	Keras, TFLearn



Pearson, Inc.

......

DL Librarie

Introduction

Symbolic Programming Graphs Neurons

Fitting Model: Eight Data Points

Dense Net

D01100 1400

Project Improvement

Where We Are Ten Tunings

Up Nex

Review Take-Home Exercise

- 2 Comparison of the Leading Deep Learning Libraries
 - 3 Introduction to TensorFlow
 Symbolic Programming
 Programming TensorFlow Graphs
 Neurons in TensorFlow
 - 4 Fitting Models
 Eight Data Points
 Eight Million Points
 - 5 Dense Nets
 - 6 Convolutional Nets
 - 7 Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
 - 8 Up Next: Advanced Topics



I TOVIOW

DL Librarie

Introduction Symbolic Programming Graphs

Fitting Model

Eight Data Points

Eight Million Points

Dense Net

_

Project Improvemen Where We Are

Up Nex

- 1 Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- 5 Dense Nets
- 6 Convolutional Nets
- 7 Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



Review

DI Libraria

Introduction

Symbolic Programming Graphs

Fitting Model Eight Data Points

Light Willion Foll

Project Improvemen Where We Are

Up Next

TensorFlow Graphs

- build graph
- 2 initialize session
- g fetch and feed data



Review

DI Librarie

Introduction

Symbolic Programming Graphs

Fitting Models Eight Data Points

Eight Million Point

ConvNets

Project Improvemen Where We Are

Up Nex

TensorFlow Graphs

- 1 build graph
- 2 initialize session
- g fetch and feed data



Review

DI Librarie

Introduction

Symbolic Programming Graphs

Fitting Models Eight Data Points

Dense Ne

Project Improvemen

Ten Tunings

Up Nex

TensorFlow Graphs

- 1 build graph
- 2 initialize session
- 3 fetch and feed data



A Familiar Equation

Review

DI Librario

Introduction Symbolic

Programming Graphs

Fitting Model
Eight Data Points

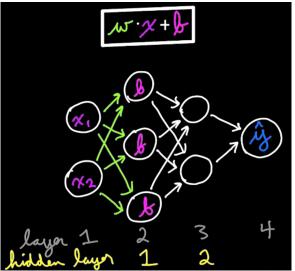
Dance Not

Delise Met

Project Improvemen

Where We Are Ten Tunings

Up Next







Review

DI Libraria

Introductio Symbolic

Programming Graphs

Fitting Model
Eight Data Points

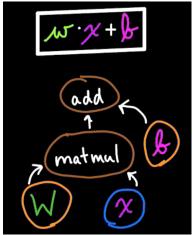
Dansa Nat

Project

Where We Are

Up Next

TensorFlow Graphs



©2018 Pearson, Inc.



DL Librarie

Introduction
Symbolic
Programming
Graphs
Neurons

Fitting Model

Eight Data Points

Eight Million Points

Dense Net

ConvNets

Improvemer
Where We Are
Ten Tunings

Up Ne

- 1 Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- 5 Dense Nets
- 6 Convolutional Nets
- 7 Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



Review

DL Librarie

Introduction

Symbolic

Graphs

Fitting Models

Eight Data Points

Danca Nat

Project

Where We Are

Up Next

TensorFlow Graph Programming

[first TensorFlow graphs notebook]



Neurons

- Introduction to TensorFlow Programming TensorFlow Graphs Neurons in TensorFlow
- Eight Million Points

- Where We Are Ten Hyperparameter-Tuning Steps



A Familiar Equation

Review

DL Libraria

Introduction

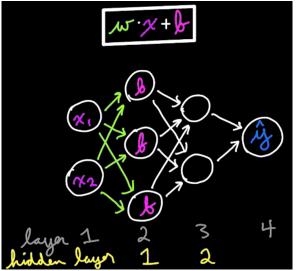
Programmin Graphs Neurons

Fitting Model:
Eight Data Points

Dense Net

Project Improvemen Where We Are

Up Next







Review

DI Libraria

Introduction

Symbolic Programmi Graphs Neurons

Fitting Models

Eight Data Points

Dense Net

. . .

Project Improvemen Where We Are

Up Next

Neurons in TensorFlow Programming

[first TensorFlow neurons notebook]



Review

DL Librarie

Introduction Symbolic Programming Graphs

Fitting Models
Eight Data Points

Dense Net

Delise Net

Project Improvemer

Un Nex

- 1 Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models

 Eight Data Points

 Eight Million Points
- **5** Dense Nets
- 6 Convolutional Nets
- 7 Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



.....

DL Librarie

Introduction Symbolic Programming Graphs

Fitting Model
Eight Data Points
Eight Million Point

Dense Net

Delise Net

Project Improvemen

Ten Tunings

l In Ne

- 1 Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- 5 Dense Nets
- 6 Convolutional Nets
- 7 Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



Review

DL Librari

Introduction

Symbolic Programming Graphs

Fitting Models

Eight Data Points

Dense Net

Delise Met

Camullata

Project Improvemen

Ten Tunings

Fitting Eight Points

[point by point intro to TensorFlow notebook]



Review

DL Librari

Introduction

Symbolic Programming Graphs

Fitting Model

Dance Nets

2 ...

Project Improvemen

Ten Tunings

Up Next

Fitting Eight Points with Tensors

[tensor-fied intro to TensorFlow notebook]



Outline

TIGVIGW

DL Librarie

Introduction Symbolic Programming Graphs Neurons

Fitting Models

Eight Data Points

Eight Million Points

Dense Net

Delise Net

Project Improvement Where We Are

Up Nex

- 1 Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
 - Dense Nets
- 6 Convolutional Nets
- Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



Reviev

DI Libraria

Introduction

Programming Graphs

Fitting Models

Eight Data Points

Eight Million Points

Dense Net

Delibe 140

Project Improvemen

Where We Are Ten Tunings

Up Next

Fitting Eight Million Points

[intro to TensorFlow times a million notebook]



Outline

TICVICW

DL Librarie

Introduction Symbolic Programming Graphs

Fitting Model

Eight Data Points

Fight Million Points

Dense Nets

Project Improvemen

Ten Tunings

Un Ne

- 1 Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- 5 Dense Nets
- 6 Convolutional Nets
- 7 Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



Dense Nets

Review

DI Librari

Introduction

Programming
Graphs

Fitting Models

Eight Data Points

Dense Nets

Camullata

Project Improvement Where We Are

Up Next

[intermediate net in TensorFlow notebook]



DL Librario

Introduction

Programming Graphs

Fitting Models
Eight Data Points

Dense Nets

CamulAlata

Project Improvemen

Ten Tunings

Up Next

Dense Nets

[dense net in TensorFlow notebook]



Outline

DI I de contra

DL Librarie

Introduction Symbolic Programming Graphs

Fitting Model

Eight Data Points

Eight Million Points

ConvNets

Project Improvemer Where We Are

Up Nex

- Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- 5 Dense Nets
- 6 Convolutional Nets
- 7 Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



Review

DI Libraria

Introduction

Symbolic Programming Graphs

Fitting Models

Eight Data Points

Dense Nets

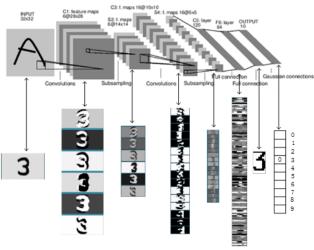
ConvNets

Project Improvemer Where We Are

Up Next

LeNet-5

LeCun et al. (1998)





Review

DI Libraria

Introduction

Symbolic Programming Graphs

Fitting Models

Eight Data Points

Dense Net

ConvNets

Project Improvemen

Where We Are Ten Tunings

Up Next

LeNet-5 LeCun et al. (1998)

[LeNet in TensorFlow notebook]



Outline

I ICVICW

DL Librarie

Introduction Symbolic Programming Graphs

Fitting Model

Eight Data Points

Fight Million Points

Dense Net

Project Improvement

Where We Are Ten Tunings

I In Ne

- Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- Dense Nets
- 6 Convolutional Nets
- 7 Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



Outline

......

DL Librarie

Introduction Symbolic Programming Graphs

Fitting Model

Eight Data Points

Fight Million Points

Dense Net

20...00 . 100.

Project Improvemen

Where We Are Ten Tunings

Un Ne

- Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- Dense Nets
- 6 Convolutional Nets
- Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



Review

M. I. Hawawi.

Introductio
Symbolic
Programming

Fitting Model

Dense Ne

Project Improveme Where We Are

Up Nex





Review

DL Librarie

Introduction

Symbolic Programming Graphs

Fitting Model: Eight Data Points Eight Million Points

Dense Net

Project Improvemer Where We Are

Up Nex

- Splitting your data
 - training set (80% for optimizing parameters)
 - validation set (10% for hyperparameters)
 - test set (10% don't touch yet!)
- ② Building and assessing architecture
 - get above chance (simplifying problem, if necessary)
 - do existing performance benchmarks exist?
 - if not, use a simple architecture as benchmark
- Improving performance & tuning hyperparameters in ten steps...



Review

DI Librario

Introduction Symbolic Programming

Fitting Models

Eight Data Points

Eight Million Points

Dense Nets

Project Improvement Where We Are

Up Nex

- Splitting your data
 - training set (80% for optimizing parameters)
 - validation set (10% for hyperparameters)
 - test set (10% don't touch yet!)
- ② Building and assessing architecture
 - get above chance (simplifying problem, if necessary)
 - do existing performance benchmarks exist?
 - if not, use a simple architecture as benchmark
- Improving performance & tuning hyperparameters in ten steps...



Review

DL Librari

Introduction Symbolic Programming Graphs

Fitting Models

Eight Data Points

Eight Million Points

Dense Nets

O - --- M - 4 -

Project Improvemer Where We Are

Up Nex

Improving

Your Deep Learning Project IV

- Splitting your data
 - training set (80% for optimizing parameters)
 - validation set (10% for hyperparameters)
 - test set (10% don't touch yet!)
- ② Building and assessing architecture
 - get above chance (simplifying problem, if necessary)
 - do existing performance benchmarks exist?
 - if not, use a simple architecture as benchmark
- Improving performance & tuning hyperparameters in ten steps...



Review

DL Librari

Introduction
Symbolic
Programming
Graphs

Fitting Models

Eight Data Points

Eight Million Points

Dense Nets

ConvNets

Project Improvemer Where We Are Ten Tunings

Up Ne

- Splitting your data
 - training set (80% for optimizing parameters)
 - validation set (10% for hyperparameters)
 - test set (10% don't touch yet!)
- ② Building and assessing architecture
 - get above chance (simplifying problem, if necessary)
 - do existing performance benchmarks exist?
 - if not, use a simple architecture as benchmark
- (3) Improving performance & tuning hyperparameters in ten steps...



Review

DL Librarie

Introduction Symbolic Programming Graphs

Fitting Models
Eight Data Points
Eight Million Points

Dense Net

Project Improveme Where We Are

Up Ne

- Splitting your data
 - training set (80% for optimizing parameters)
 - validation set (10% for hyperparameters)
 - test set (10% don't touch yet!)
- 2 Building and assessing architecture
 - · get above chance (simplifying problem, if necessary)
 - do existing performance benchmarks exist?
 - if not, use a simple architecture as benchmark
- 3 Improving performance & tuning hyperparameters in ten steps...



Review

DL Librarie

Introduction Symbolic Programming Graphs

Fitting Models
Eight Data Points
Eight Million Points

Delise Net

Project Improvement Where We Are

Up Ne

- Splitting your data
 - training set (80% for optimizing parameters)
 - validation set (10% for hyperparameters)
 - test set (10% don't touch yet!)
- 2 Building and assessing architecture
 - get above chance (simplifying problem, if necessary)
 - do existing performance benchmarks exist?
 - if not, use a simple architecture as benchmark
- Improving performance & tuning hyperparameters in ten steps...



Review

DL Librarie

Introduction Symbolic Programming Graphs

Fitting Models
Eight Data Points
Eight Million Points

Dense Net

Project

Improvemen
Where We Are
Ten Tunings

Up Ne:

- Splitting your data
 - training set (80% for optimizing parameters)
 - validation set (10% for hyperparameters)
 - test set (10% don't touch yet!)
- 2 Building and assessing architecture
 - · get above chance (simplifying problem, if necessary)
 - do existing performance benchmarks exist?
 - if not, use a simple architecture as benchmark
- Improving performance & tuning hyperparameters in ten steps...



Review

DL Librarie

Introduction Symbolic Programming Graphs Neurons

Fitting Models
Eight Data Points
Eight Million Points

Delise Net

Project Improvement Where We Are

Up Ne

- Splitting your data
 - training set (80% for optimizing parameters)
 - validation set (10% for hyperparameters)
 - test set (10% don't touch yet!)
- 2 Building and assessing architecture
 - · get above chance (simplifying problem, if necessary)
 - do existing performance benchmarks exist?
 - if not, use a simple architecture as benchmark
- Improving performance & tuning hyperparameters in ten steps...



Review

DL Librarie

Introductio
Symbolic
Programming
Graphs
Neurons

Fitting Models

Eight Data Points

Eight Million Points

Dense Nets

Project Improveme

Up Ne

- Splitting your data
 - training set (80% for optimizing parameters)
 - validation set (10% for hyperparameters)
 - test set (10% don't touch yet!)
- 2 Building and assessing architecture
 - get above chance (simplifying problem, if necessary)
 - do existing performance benchmarks exist?
 - if not, use a simple architecture as benchmark
- Improving performance & tuning hyperparameters in ten steps...



Outline

TICVICW

DL Librarie

Introduction Symbolic Programming Graphs

Fitting Model: Eight Data Points Eight Million Points

Dense Net

Project Improvement

Ten Tunings

- Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- Dense Nets
- 6 Convolutional Nets
- 7 Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



1. Initialization

Reviev

DI Librari

Introductio

Programming
Graphs
Nourons

Fitting Models

Eight Data Points

Fight Million Points

Dense Ne

ConvNets

Project Improvemer Where We Are Ten Tunings

Up Next

- number of layers
- type of layers
- layer width (by powers of two)



1. Initialization

Review

DI Librarie

Introductio

Symbolic Programming Graphs Neurons

Fitting Model

Eight Million Point

. . . .

CONVINCE

Project Improvemer Where We Are

Where We Are Ten Tunings

Up Next

- number of layers
- type of layers
- layer width (by powers of two)



1. Initialization

Review

DI Librarie

Introduction

Symbolic Programming Graphs Neurons

Eight Data Points

Eight Million Poin

ConvNets

Project Improvemen Where We Are Ten Tunings

Up Next

- number of layers
- type of layers
- layer width (by powers of two)



Review

OL Librarie

Introduction

Symbolic Programming Graphs Neurons

Fitting Models

Eight Data Points

Dense Net

Delise Net

Project Improveme Where We Are

Ten Tunings
Up Next

1. Initialization

...in lenet_in_tensorflow.ipynb:

```
Set neural network hyperparameters

epochs = 20
batch_size = 128
display progress = 40 # after this many batches, output progress to screen
wt_init = tf.contrib.layers.xavier_initializer() # weight initializer
```



Review

DL Librari

Introductio

Symbolic Programming Graphs

Fitting Models

Eight Data Points

Eight Million Points

Delise IV

ConvNet

Project Improvemen Where We Are Ten Tunings

Up Next

2. Get Above Chance

If your accuracy is below chance, try:

- simplifying the problem
- simplifying the network architecture
- reducing your training set size (to iterate more quickly)



Review

DI Librari

Introductio

Symbolic Programming Graphs

Fitting Model

Eight Data Points

Fight Million Points

Dense N

ConvNet

Project Improvemen Where We Are Ten Tunings

Up Next

2. Get Above Chance

If your accuracy is below chance, try:

- simplifying the problem
- simplifying the network architecture
- reducing your training set size (to iterate more quickly)



Review

DL Librario

Introductio Symbolic Programming Graphs

Fitting Models
Eight Data Points
Eight Million Points

Dense N

ConvNet

Project Improvemer Where We Are Ten Tunings

Up Next

2. Get Above Chance

If your accuracy is below chance, try:

- simplifying the problem
- simplifying the network architecture
- reducing your training set size (to iterate more quickly)



Review

DL Librari

Introductio

Programming Graphs Neurons

Fitting Model Eight Data Points

Eight Million Point

. . . .

CONVINCE

Improvemer Where We Are Ten Tunings

Up Next

- number of layers
- type of layers
- layer width (by powers of two)



3. Layers

Review

DL Librari

Introductio

Programming
Graphs
Neurons

Eight Data Points
Eight Million Points

_

Project Improvemen Where We Are Ten Tunings

Up Next

- number of layers
- type of layers
- layer width (by powers of two)



Reviev

DL Librario

Introduction

Programming Graphs Neurons

Eight Data Points Fight Million Points

. . . .

CONVINCE

Improvement Where We Are

Up Next

- number of layers
- type of layers
- layer width (by powers of two)



4 Cost

Units 7 and 8 — Tensorflow

Reviev

DI Librari

Introduction

Programming Graphs Neurons

Fitting Models

Eight Data Points

Eight Million Points

Delise Net

0----

Project Improvement

Ten Tunings

...in lenet in keras.ipynb:

```
Configure model

model.compile(loss=['categorical_crossentropy'] optimizer='adam', metrics=['accuracy'])
```

...in lenet in tensorflow.ipynb:

Define model's loss and its optimizer

cost = tf.reduce_mean(tf.nn.softmax_cross_entropy_with_logits(logits=predictions, labels=y)
optimizer = tf.train.AdamOptimizer(..minimize(cost)



Reviev

DL Librari

Introduction Symbolic Programming Graphs

Fitting Models

Eight Data Points

Eight Million Points

Delise IV

ConvNet

Project Improvemen Where We Are Ten Tunings

Up Next

5. Avoid Overfitting

If validation cost begins to increase or validation accuracy begins to decrease, consider:

- stopping training earlier
- dropout



Reviev

DL Librari

Introduction Symbolic Programming Graphs

Fitting Model

Eight Data Points

Eight Million Points

Delise IV

ConvNet

Project Improvemen Where We Are Ten Tunings

Up Next

5. Avoid Overfitting

If validation cost begins to increase or validation accuracy begins to decrease, consider:

- stopping training earlier
- dropout



Review

OL Librarie

Introduction

Symbolic Programming Graphs

Fitting Models Eight Data Points Fight Million Points

Dense Nets

Project Improvement

Ten Tunings
Up Next

5. Avoid Overfitting

...in lenet in keras.ipynb:

```
model = Sequential()
model.add(Conv2D(32, kernel_size=(3, 3), activation='relu', input_shape=(28, 28, 1)))
model.add(Conv2D(64, kernel_size=(3, 3), activation='relu'))
model.add(Dropout(0.25))
model.add(Dropout(0.25))
model.add(Piatten())
model.add(Dropout(0.5))
model.add(Dropout(0.5))
model.add(Dropout(0.5))
model.add(Dropout(0.5))
model.add(Dropout(0.5))
model.add(Dropout(0.5))
```

...in lenet in tensorflow.ipynb:

```
# max pooling layer:
pool_size = 2
mp_layer dropout = 0.25

# dense layer:
n_dense = 128
dense_layer_dropout = 0.5

# dense_layer_dropout = 0.5

# dense layer:
n_dense_layer_dropout = 0.5

# dense_layer_dropout = 0.5

# dense_layer_dropo
```



Ten Tuninas

6. Learning Rate

...in lenet in keras.ipynb:

Configure model

model.compile(loss='categorical crossentropy', optimizer='adam', metrics=['accuracy'])

...in lenet in tensorflow.ipynb:

Define model's loss and its optimizer

cost = tf.reduce mean(tf.nn.softmax cross entropy with logits(logits=predictions, labels=y)) optimizer = tf.train.AdamOptimizer().minimize(cost)



Review

DI Libraria

Introduction

Symbolic Programming Graphs Neurons

Fitting Models Eight Data Points

Dense Net

Delise Ne

Project Improvemen

Where We Are Ten Tunings

Up Nex

7. Epochs

...in lenet in keras.ipynb:

```
Train!

model.fit(X_train, y_train, batch_size=128, epochs=20, verbose=1, validation_data=(X_test, y_test))
```

...in lenet in tensorflow.ipynb:

```
Set neural network hyperparameters

| pochs = 20 |
| batcn_Bixe = 128 |
| display_progress = 40 # after this many batches, output progress to screen
| wt_init = tf.contrib.layers.xavier_initializer() # weight initializer

# loop over epochs;
| for epoch in range epoch in range epoch in range epoch in range epoch epoc
```



Units 7 and 8 — Tensorflow

Review

DL Librari

Introduction

Programming
Graphs
Neurone

Fitting Models

Eight Data Points

Fight Million Points

Delise IV

ConvNets

Project Improvemen Where We Are Ten Tunings

Up Next

8. Regularization λ

If using L1 or L2 regularization, consider:

- adjusting $\boldsymbol{\lambda}$ by orders of magnitude



Units 7 and 8
— Tensorflow

Reviev

DL Libraria

Introduction

Symbolic Programming Graphs

Fitting Models Eight Data Points

Dense Net

Donoc No.

Project Improvemen

Where We Are Ten Tunings

Up Ne

9. Batch Size

...in lenet in keras.ipynb:

```
Train!

model.fit(X_train, y_train, batch_size=128, epochs=20, verbose=1, validation_data=(X_test, y_test))
```

...in lenet in tensorflow.ipynb:

```
epochs = 20
batch_size = 128
display_progress = 40 # after this i
wt init = tf.contrib.layers.xavier :
```

Set neural network hyperparameters

```
# loop over all batches of the epoch:
n batches = int(mist.train.num examples / batch_size)
for i in range(n batches):
    # to reassure you something's happening!
if i % display progress == 0:
    print("Step", 'i.t.,' of ", n_batches, " in epoch ", epoch+1, ".", sep='')
batch_x, batch_y = mnist.train.next_batch_batch_size
```



Reviev

DL Librari

Introductio

Symbolic Programming Graphs Neurons

Fitting Models

Eight Data Points

_

Duningt

Improvemen
Where We Are
Ten Tunings

Up Next

10. Automation

For grid search of hyperparameters, consider:

- sampling values instead of looping over fixed values
- using [Spearmint]



10. Automation

Review

DL Librari

Introduction

Programming
Graphs
Neurons

Fitting Model

Eight Data Points

ConvNets

Project Improvemer Where We Are Ten Tunings

Up Next

For grid search of hyperparameters, consider:

- sampling values instead of looping over fixed values
- using [Spearmint]



Review

DL Librarie

Introduction

Symbolic Programming Graphs

Fitting Mode Eight Data Points

Daniel Mate

Project Improvemen

Where We Are
Ten Tunings

- 1 Initialization
- ② Get Above Chance
- 3 Layers
- 4 Cost
- Avoid Overfitting
- 6 Learning Rate
- Epochs
- $oxed{8}$ Regularization λ
- 9 Batch Size
- Automation



Review

DL Librari

Introduction

Symbolic Programming Graphs

Fitting Model

Eight Data Points

Fight Million Points

Dense Ne

ConvNets

Project Improvemen Where We Are Ten Tunings

- 1 Initialization
- ② Get Above Chance
- 3 Layers
- 4 Cost
- Avoid Overfitting
- 6 Learning Rate
- Epochs
- f 8 Regularization λ
- 9 Batch Size
- Automation



Review

DL Librari

Introduction

Programming
Graphs
Neurons

Fitting Mode

Eight Data Points

Dense Ne

ConvNets

Project Improvemen Where We Are Ten Tunings

- 1 Initialization
- 2 Get Above Chance
- 3 Layers
- 4 Cos
- Avoid Overfitting
- 6 Learning Rate
- Epochs
- 8 Regularization λ
- 9 Batch Size
- Automation



Review

DL Librari

Introduction Symbolic

Programming
Graphs
Neurons

Fitting Model: Eight Data Points Eight Million Points

20..00 ...

Project Improvemer Where We Are Ten Tunings

- Initialization
- ② Get Above Chance
- 3 Layers
- 4 Cost
- 6 Avoid Overfitting
- **6** Learning Rate
- Epochs
- 8 Regularization λ
- 9 Batch Size
- Automation



Review

DL Librari

Introduction

Symbolic Programming Graphs Neurons

Fitting Model Eight Data Points Fight Million Points

Donco Note

ConvNot

Project Improvemen Where We Are Ten Tunings

- Initialization
- ② Get Above Chance
- 3 Layers
- 4 Cost
- 6 Avoid Overfitting
- 6 Learning Rate
- Epochs
- 8 Regularization λ
- 9 Batch Size
- Automation



Review

DL Librari

Introduction Symbolic Programming

Graphs
Neurons

itting Mode

Eight Data Points
Eight Million Points

20..00 . . .

ConvNets

Project Improvemer Where We Are Ten Tunings

- 1 Initialization
- ② Get Above Chance
- 3 Layers
- 4 Cost
- 6 Avoid Overfitting
- 6 Learning Rate
- Epochs
- 8 Regularization λ
- 9 Batch Size
- Automation



Review

DL Librari

Introduction Symbolic Programming

Fitting Model: Eight Data Points Eight Million Points

Delise IV

Project Improvement Where We Are

Ten Tunings Up Next

- Initialization
- 2 Get Above Chance
- 3 Layers
- 4 Cost
- 6 Avoid Overfitting
- **6** Learning Rate
- Epochs
- 8 Regularization λ
- 9 Batch Size
- Automation



Review

DL Librari

Introduction Symbolic Programming Graphs

Fitting Model: Eight Data Points Eight Million Points

Dense Ne

ConvNets

Project Improvement Where We Are Ten Tunings

- Initialization
- ② Get Above Chance
- 3 Layers
- 4 Cost
- 6 Avoid Overfitting
- 6 Learning Rate
- Epochs
- 8 Regularization λ
- Batch Size
- Automation



Review

DL Librario

Introductio Symbolic Programming Graphs

Fitting Model
Eight Data Points
Eight Million Points

Delise ive

Project

Improvemer
Where We Are
Ten Tunings

- Initialization
- ② Get Above Chance
- 3 Layers
- 4 Cost
- 6 Avoid Overfitting
- 6 Learning Rate
- Epochs
- 8 Regularization λ
- Batch Size
- Automation



Review

DL Librari

Introductio Symbolic Programming Graphs

Fitting Model Eight Data Points Eight Million Points

ConvNet

Project Improvement Where We Are Ten Tunings

- Initialization
- 2 Get Above Chance
- 3 Layers
- 4 Cost
- 6 Avoid Overfitting
- **6** Learning Rate
- Epochs
- 8 Regularization λ
- Batch Size
- 40 Automation



Outline

TICVICW

DL Librarie

Introduction Symbolic Programming Graphs

Fitting Model

Eight Data Points

Fight Million Points

Dense Net

D01100 1400

Project Improvemen

Where We Are Ten Tunings

- 1 Review Take-Home Exercise
- 2 Comparison of the Leading Deep Learning Libraries
- 3 Introduction to TensorFlow Symbolic Programming Programming TensorFlow Graphs Neurons in TensorFlow
- 4 Fitting Models
 Eight Data Points
 Eight Million Points
- 5 Dense Nets
- 6 Convolutional Nets
- Deep Learning Project IV: Improving Where We Are Ten Hyperparameter-Tuning Steps
- 8 Up Next: Advanced Topics



Units 7 and 8 — Tensorflow

Review

DI Libraria

Introductio

Symbolic Programming Graphs

Fitting Models
Eight Data Points

Danca Not

Dense Net

Project Improvement Where We Are

Up Next

Generative Adversarial Networks

December 16th





Units 7 and 8

— Tensorflow

Review

DI Libraria

Introductio

Symbolic Programming Graphs

Fitting Model Eight Data Points

Dance No

Donoc 140

Project Improvemen

Where We Are Ten Tunings

Up Next

Reinforcement Learning December 16th



