GPU TECHNOLOGY CONFERENCE

October 6-7, 2016 | Seoul

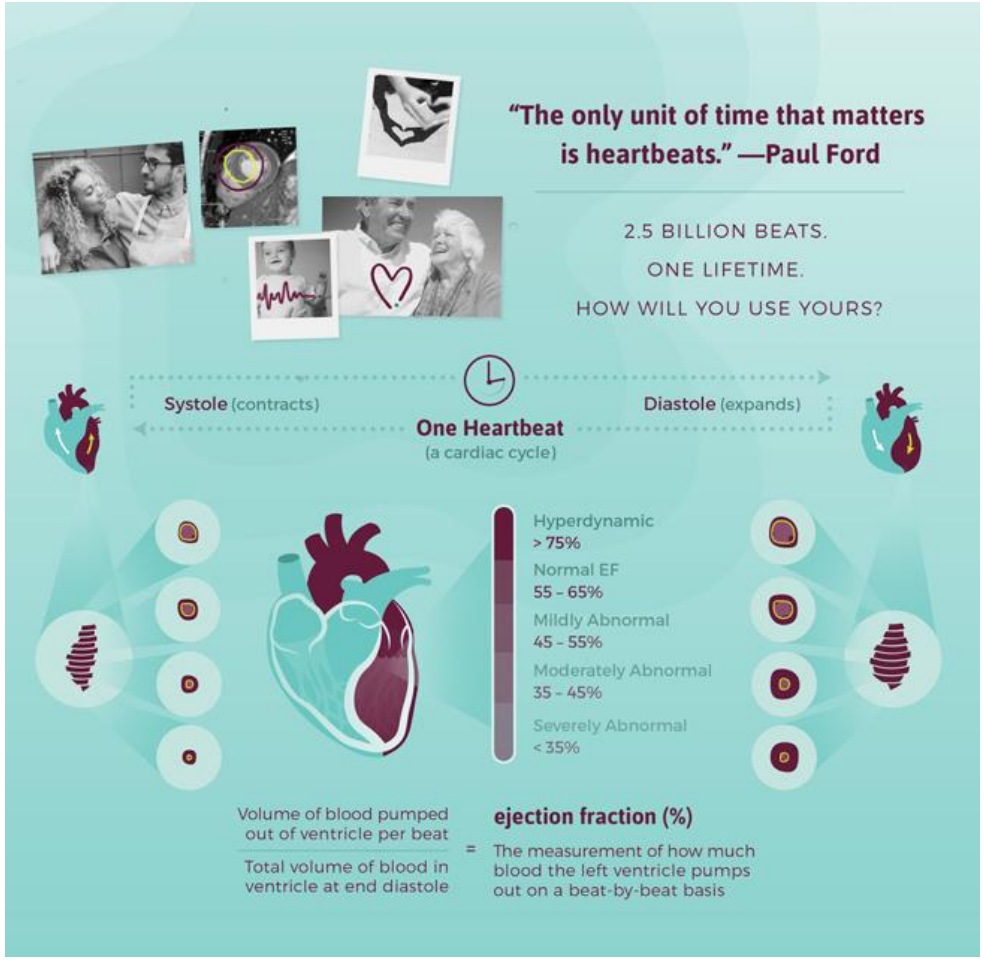
INTRODUCTION TO DEEP LEARNING WITH R AND MXNET

Seoul GTCx, Korea

PRESENTED BY

NVIDIA.

HEART DISEASE DIAGNOSE



CURRENT STATE

MRI based cardiac volume analysis

MANUAL

Skilled cardiologist

Long time, up to 20 minutes to complete

Cardiologist's time spend with the patients

Impediments for heart disease treatment research

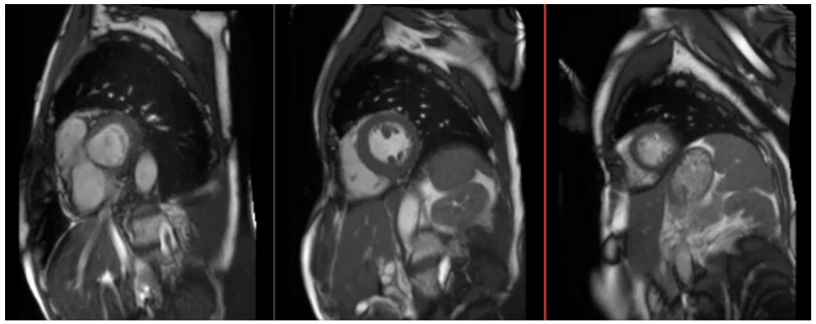
AUTOMATE

Easy diagnosis

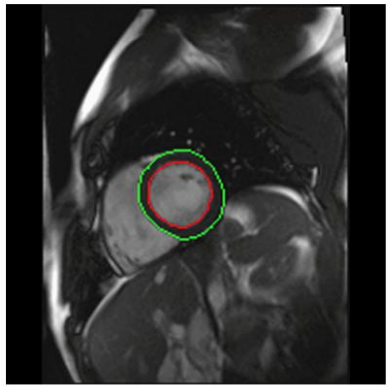
Shorten procedure time

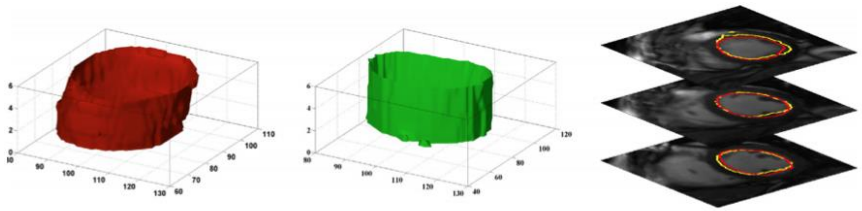
Advanced heart disease treatment

MEDICAL IMAGE ANALYSIS



Manual annotation





Software volume estimate

DATA SCIENCE BOWL CHALLENGE

Data science community and competition

Created and sponsored by Booz Allen and Kaggle

Providing an opportunity to advance the art of data science

DATASET

DATASET

https://www.kaggle.com/c/second-annual-data-science-bowl/data

The National Heart, Lung, and Blood Institute (NHLBI) provided the MRI images for this competition

End-systolic and end-diastolic volumes in cardiac MRIs from more than 500 studies

DICOM format image contains 30 images across the cardiac cycle

Data variations on age, hospitals, and normal or abnormal heart function

Various images views, number of slices, frames per time slice, and number of images

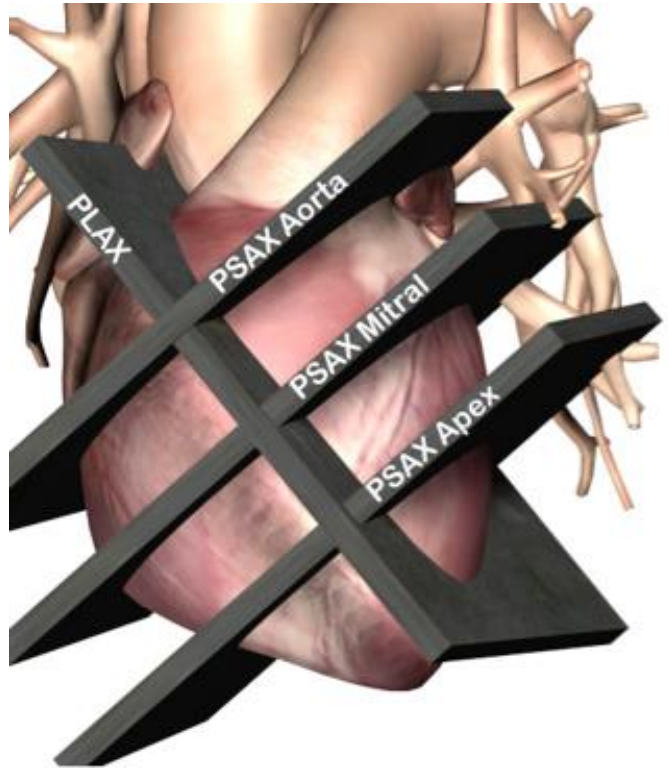
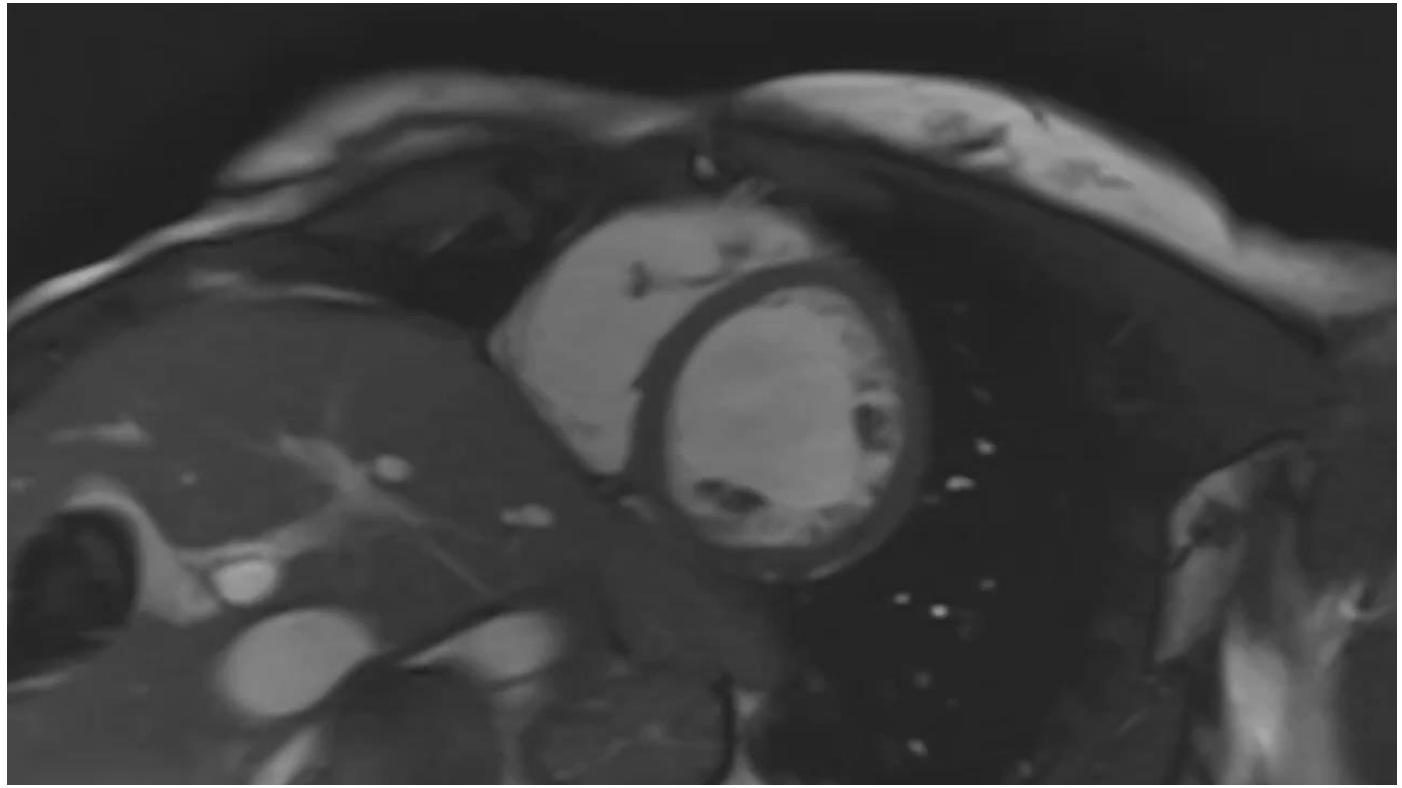


IMAGE EXAMPLE

Heartbeat sample



DATA DETAILS

500 training studies with on average 10 unique SAX observation

Each operation contain 30 DICOM images which capture an entire heartbeat

Packed image as 64 x 64

An observation has 64 x 64 x 30 tensor

500 labels, roughly 10 SAX observation is different view of same heart

EVALUATION

EJECTION FRACTION

$$100 \cdot \frac{V_D - V_S}{V_D}$$

The fraction of outbound blood pumped from the heart with each heart beat

Ejection fraction too low can signify a wide range of cardiac problems

ACCURACY SCORE

Continuous Ranked Probability Score

$$CRPS = \frac{1}{600 \cdot N} \sum_{m=1}^{N} \sum_{n=0}^{599} (P(y \leq n) - H(n - V_m))^2$$

P: predicted distribution

N: number of rows in the test set

V: actual volume

H(x): Heaviside step function ($H(x < 0) = 0$ and $H(x \geq 0) = 1$)

GENERAL OVERVIEW OF MODEL

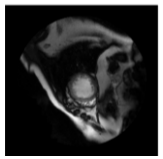
INPUT DATA

Detailed dataset description

x500 study

x10 observations (views)

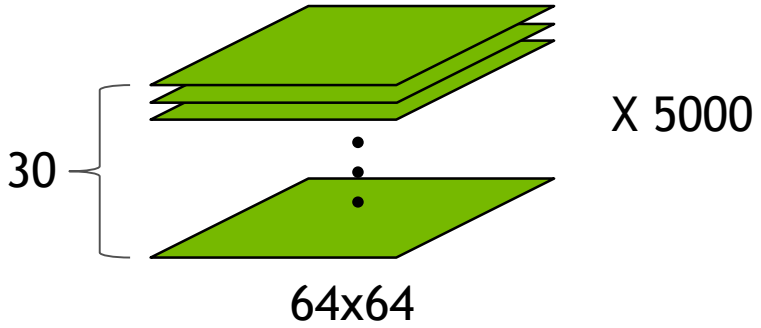
x30 DICOM images for 1 heart beat



64x64 pixel sized image

x500 labels for diastole

x500 labels for stytole



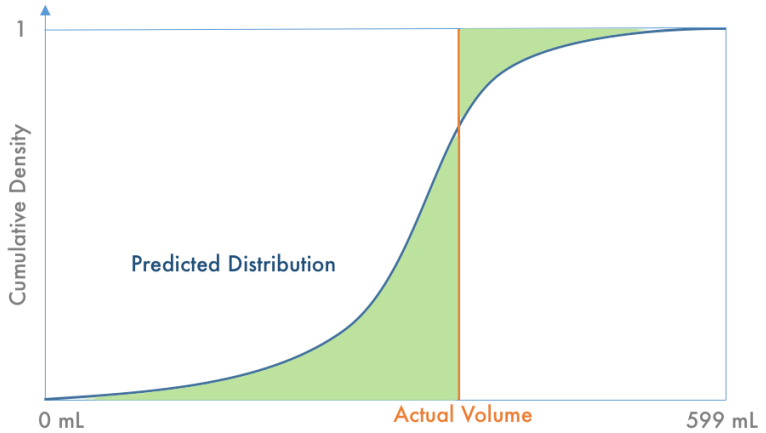
NETWORK OBJECTIVE

Predict CDF value of 600-data point

This problem is a regression problem

Our neural-net will produce 600x1 vector

Label (a single float-point value) is transformed into step function



$$H((x - label) > 0) = 1, H\big((x - label) < 0\big) = 0$$

PREPROCESSING

Packed training data into a csv file (`train-64x64-data.csv`)

5293 lines & each line has 64 x 64 x 30 tensor

To fit into memory for large data, we will use `CSVIter` from mxnet

Transformed labels (0 or 1) are stored in two separated file as `train-systole.csv` & `train-diastole.csv`

MXNET

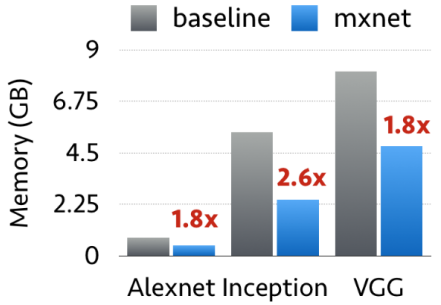
WHAT IS MXNET?

Created by dmlc, http://mxnet.io/

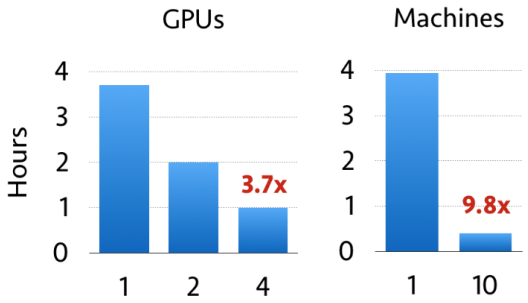
"Open source software library for efficiency and flexibility". Available on GitHub.

Portable

Efficient



Scalable



Scales up to multi GPUs and distributed setting with auto parallelism

Cloud-friendly and directly compatible with S3, HDFS, and Azure

RUNNING MXNET

Construct a graph—this happens before any real computation happens. Specify your neural network as a graph.

> `Variables`--characteristics of the graph that can change over time, i.e., learned weights.
>
> Operations—computations that combine the variables and the data, e.g., convolution, activation, matrix multiply, etc.

Define and train model through **`mx.model.FeedForward`**

SAMPLE WORKFLOW

Prepare input data — Interface using `mx.io` (e.g, `extract()`, `CSVIter()`)

Build the computation network: Create network, cost function, training nodes.

Train the model: `CSVIter()` feed data iteratively when training model. Set hyper-parameters and train the model using `mx.model.FeedForward`

Evaluate the model: run `predict` (same network as above) and then evaluate accuracy based on suitable metric.

INFERENCE NETWORK EXAMPLE

```
net <- mx.symbol.Convolution(source, kernel = c(5,5),
    num.filter = 40)

net <- mx.symbol.BatchNorm(net, fix.gamma = True)

net <- mx.symbol.Activation9net, act.type = “relu”)

net <- mx.symbol.Pooliing(net, pool.type = “max”,
    kernel = c(2,2), stride = c(2,2))
```

SETTING UP THE LAB

CONNECTION INSTRUCTIONS

Navigate to nvlabs.qwiklab.com

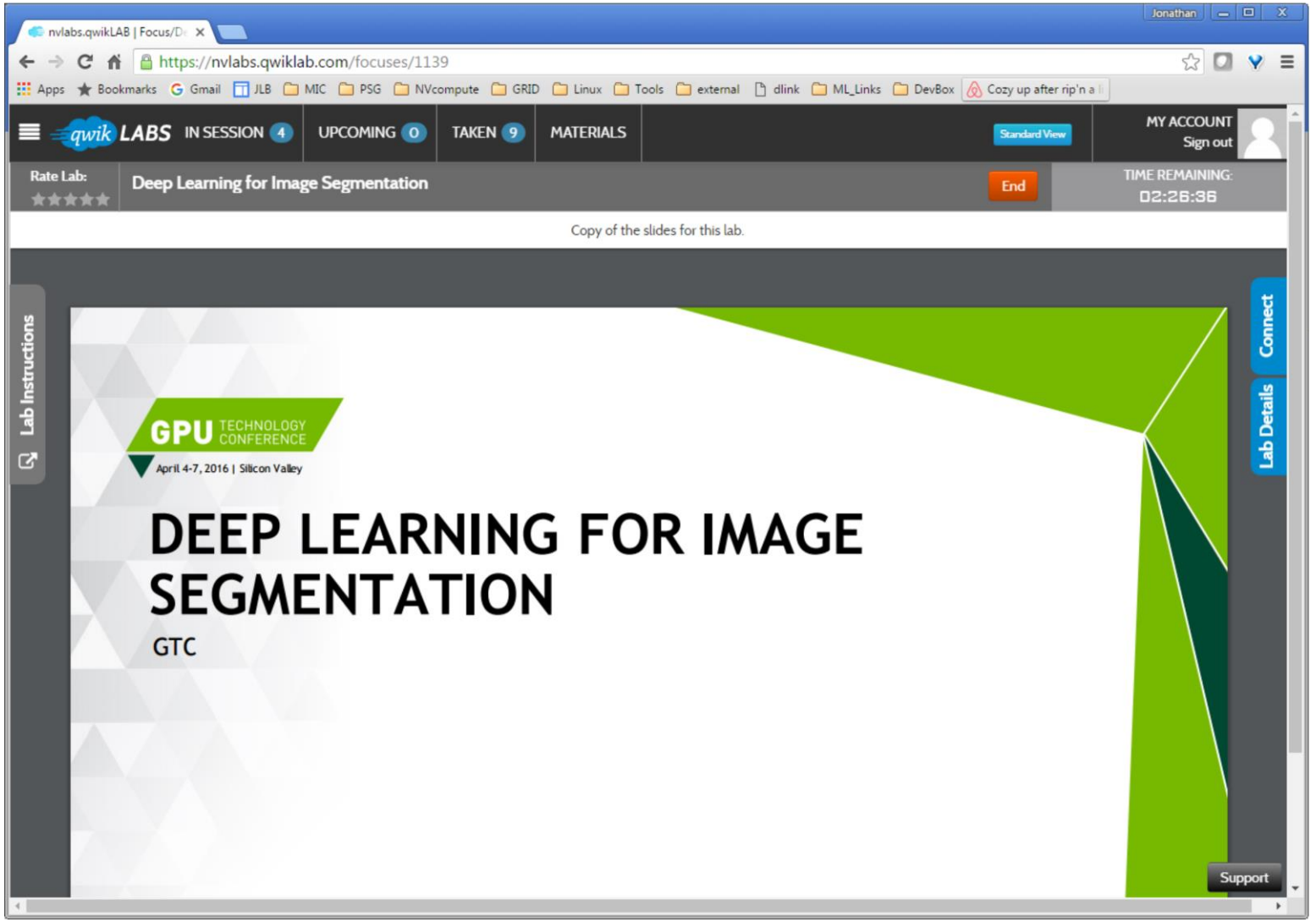
Login or create a new account

Select the “DLI Instructor-Led Labs” Class

Find the lab called “Introduction to Deep Learning with R and MXNet”, select it, click Select, and finally click Start

When prompted, enter your 16-digit token code
After a short wait, lab Connection information will be shown Please ask Lab Assistants for help!

Click here for handouts



Click here for Connection instructions

SOFTWARE FOR LAB

No need to other software

This lab uses R & MXNet over jupyter with IRKernel

NDSB COMPETITION RESULTS & CRPS IMPLICATION

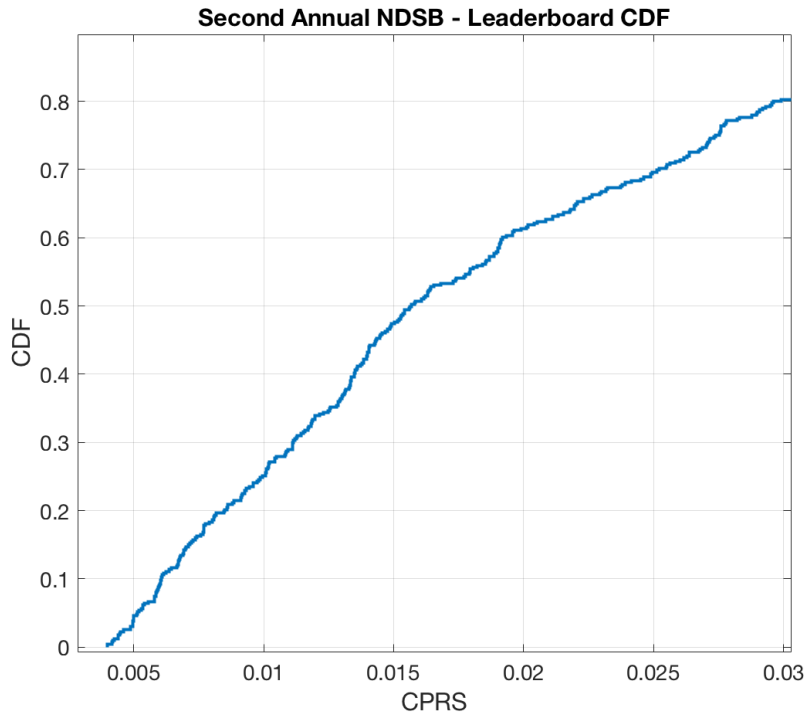
COMPETITION RESULT

For CRPS scores

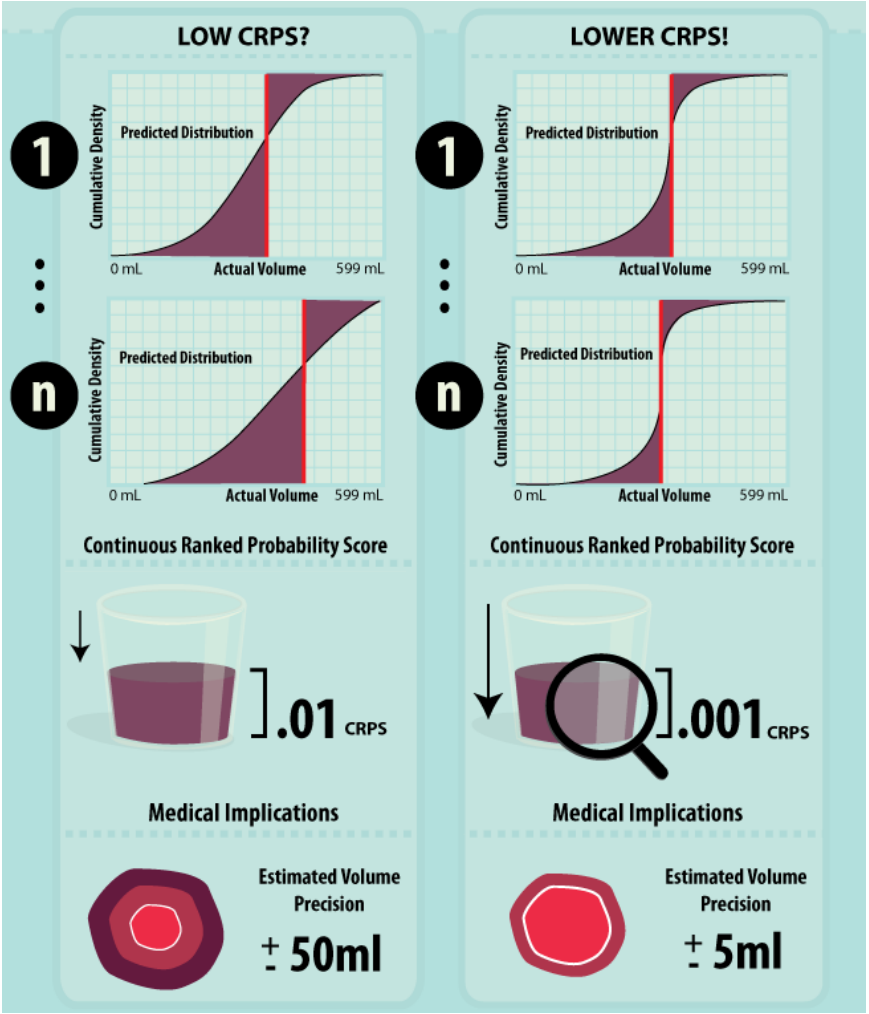
1st place: 0.009485

10th place: 0.012611

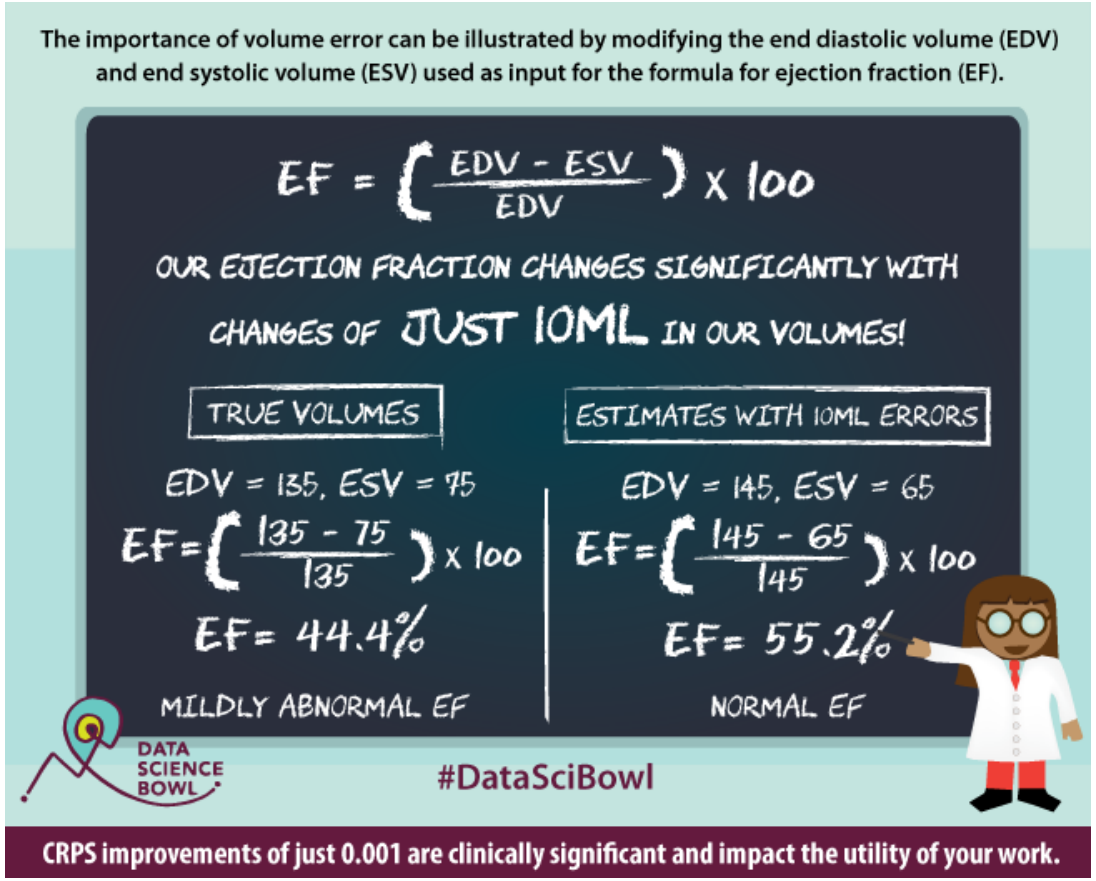
For CRPS 0.03 score ➔ 80th percentile



CRPS AND IMPLICATIONS



PREDICTION ERROR EFFECT



CODE

TRAINING DATA

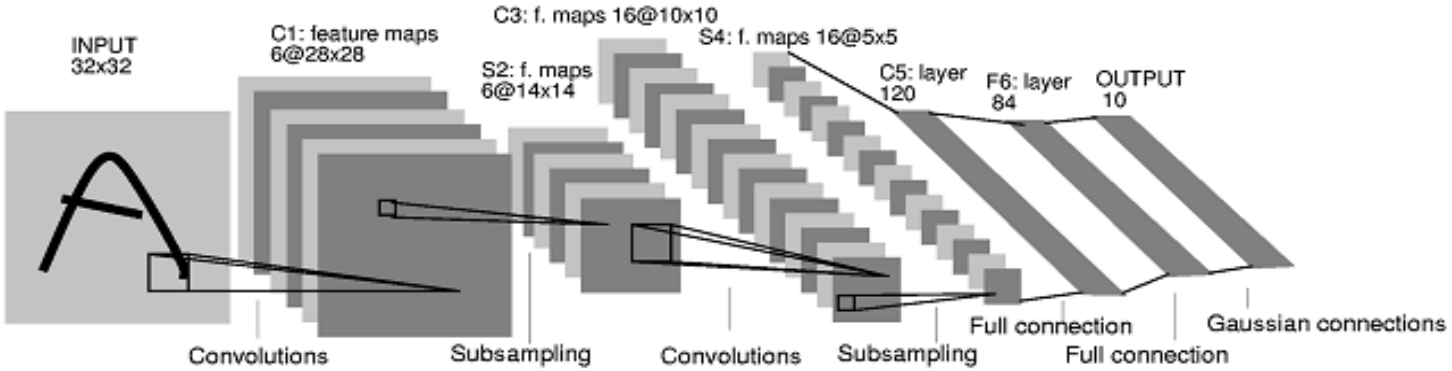
Normalize the pixel values: [0:255] ➔ [-128:127]

Use `mx.symbol.SliceChannel` to get individual frame

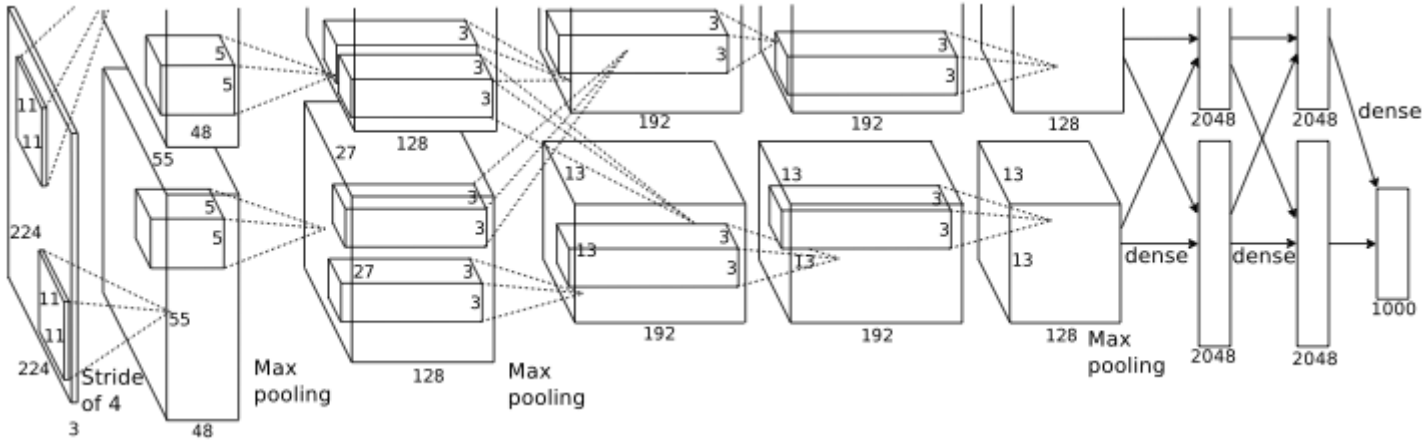
Obtain differential frame between successive frame

Use mx.varg.symbol.Concat to pack individual frame into 64x64x29 tensor

CNN



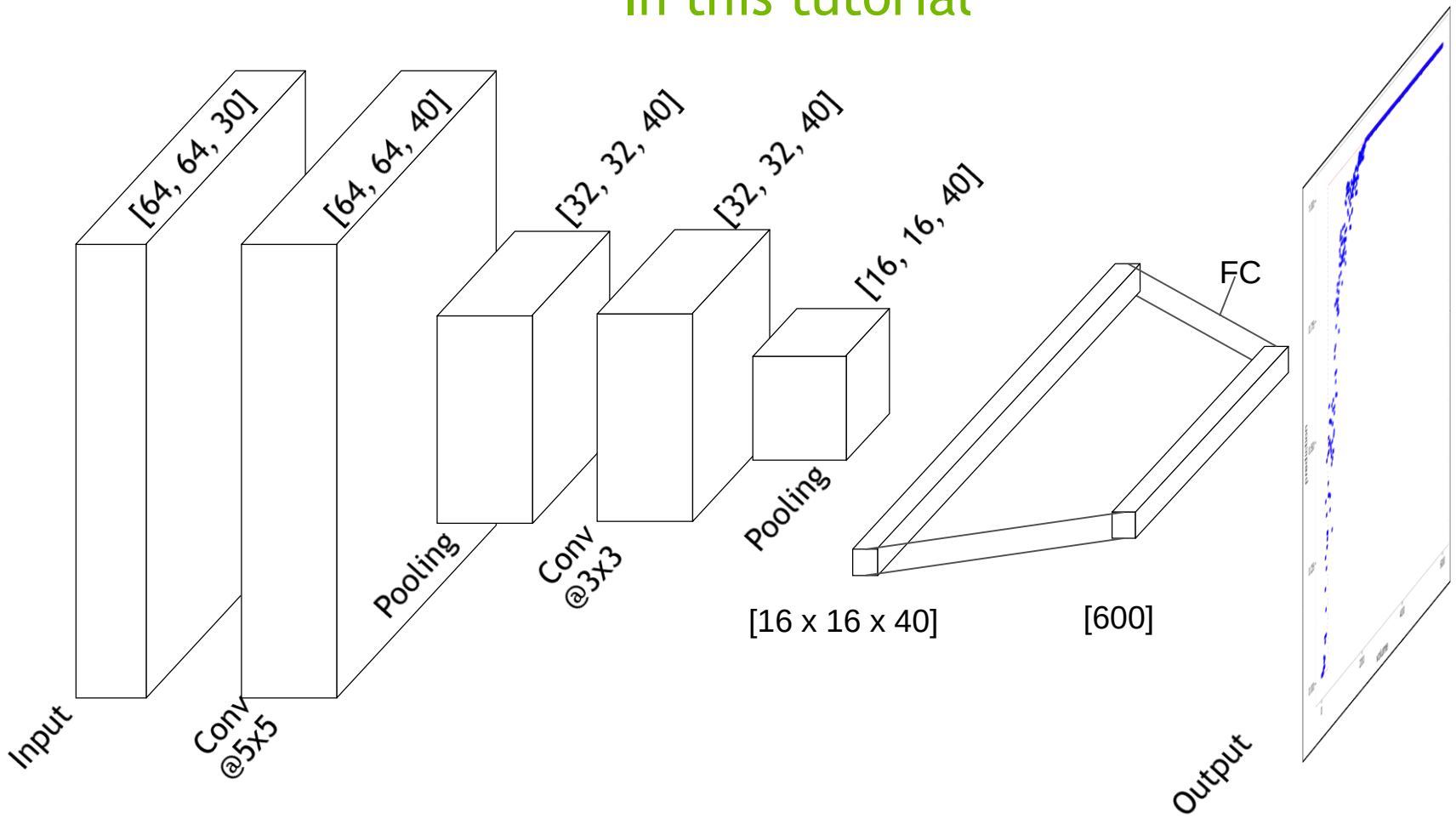
LeCun et al. 1989-1998: Handwritten Digit Recognition



Krizhevsky, Hinton et al., 2012: ImageNet Winner

LENET STYLE NETWORK

In this tutorial



GRAPH CONFIGURATION

Define LeNet Network

Initialize CSVIter with training data, label, and batch size

Initialize network

Define loss function(cost function)

```
costfun <- function(label, pred) {
  pred <- as.array(pred)
  label <- as.array(label)
  return(sum((label - pred) ^ 2) / length(label))
}
mx.metric.CRPS <- mx.metric.custom("CRPS", costfun)
```

TRAINING MODEL

Training the model

```
stytole_model <- mx.model.FeedForward.create(
  X                 = data_train,
  ctx               = mx.gpu(0),
  symbol            = network,
  run.round         = 5,
  learning.rate     = 0.01,
  wd                = 0.0001,
  momentum          = 0.9,
  Eval.metric       = mx.metric.CRPS
)
```

And save trained model,

```
mx.model.save(stytole_model,"stytole_model_BN",65)
```

OR INFERENCE

Load the pre-trained result

```
stytole_model <- mx.model.load("stytole_model_BN", 65)
```

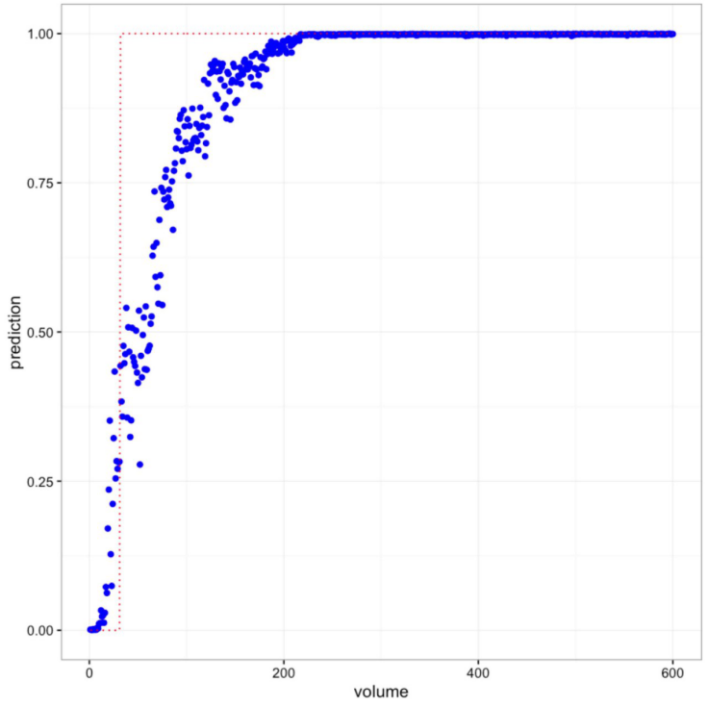
to save training time, to evaluation only

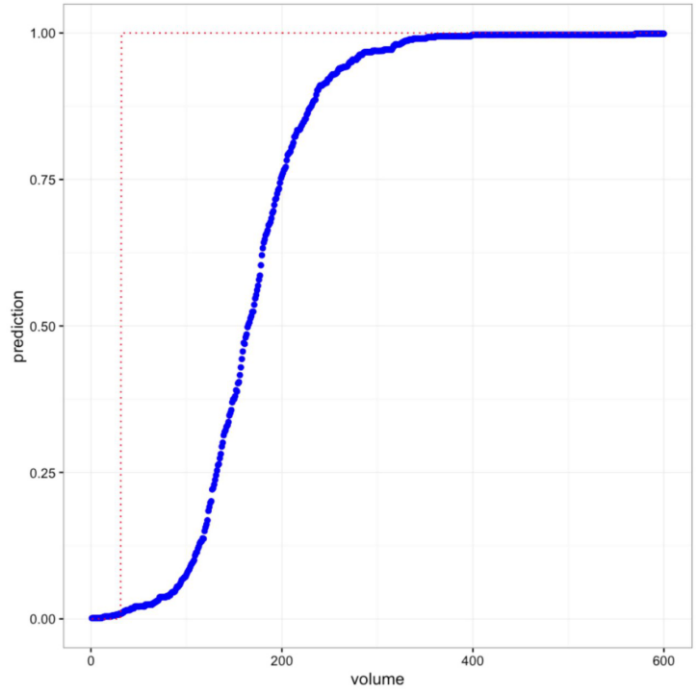
For stytole pre-trained model,

```
stytole_model <- mx.model.load("diastole_model", 65)
```

EVALUATION

stytole & diastole

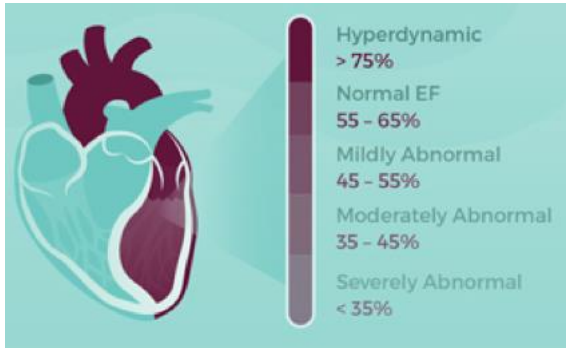




EVAPORATION FRACTION

Recall back,

$$EF = 100 \cdot \frac{V_D - V_S}{V_D}$$



For ix = 14,

$$V_S = 32, V_D = 69 \quad \rightarrow \quad EF = 100 \cdot \frac{69 - 32}{69} \cong 53.62\%$$

FURTHER ENHANCEMENTS

WHAT ELSE?

1. Try other test sample
2. Try removing batch normalization
3. Change number of features of convolution layer
4. Check batch size effect
5. Change learning rate & momentum
6. Change cost function (residual squared → abs)
7. Try other activation function rather than ReLU
8. Try different pooling function
9. Try dropout layer removal or rate change
10. Try double differencing data or not differencing data
11. Try other formulate the network output
12. Use other CNN network like GoogLeNet, resnet, or your own

LAB SUMMARY

Intro to image analysis and regression output

Used LeNet like network for image analysis

Used R and MXNet as framework

Explored data specific score(CRPS) and its implication

SUPPORTING INFO

MXNet documentation and tutorials with source code

http://mxnet.io/

http://mxnet.readthedocs.io/en/latest/packages/r/fiveMinutesNeuralNetwork.html

Same tutorial code in Train.R, Train.py

Transforming How We Diagnose heart Disease

https://www.kaggle.com/c/second-annual-data-science-bowl

GPU TECHNOLOGY CONFERENCE

April 4-7, 2016 | Silicon Valley

THANK YOU

JOIN THE CONVERSATION

#GTC16

PRESENTED BY

NVIDIA.