



DEEP
LEARNING
INSTITUTE



ODTÜ
METU

Image Classification with DIGITS

Dr. Alptekin Temizel

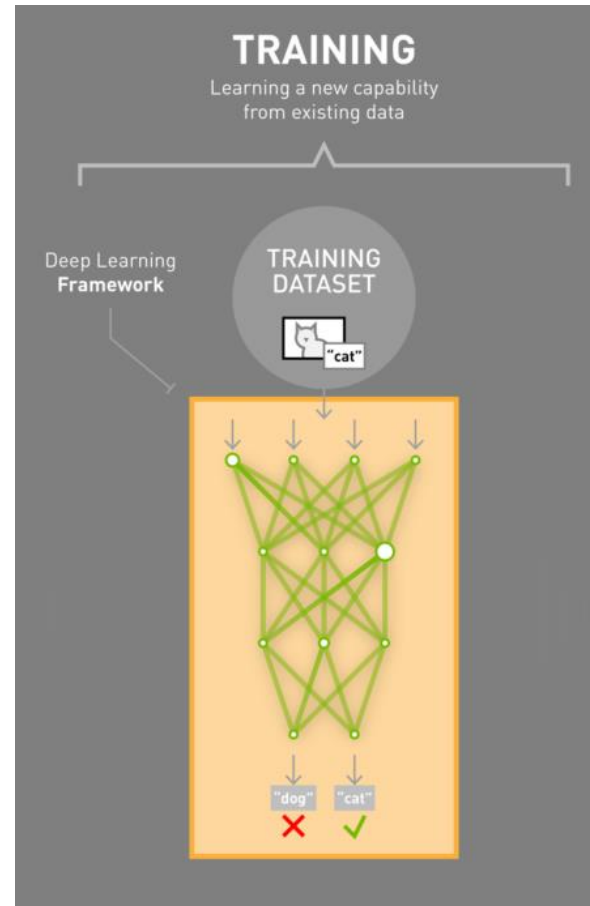
DLI Certified Instructor

Associate Professor, Graduate School of Informatics, METU

8 January 2018

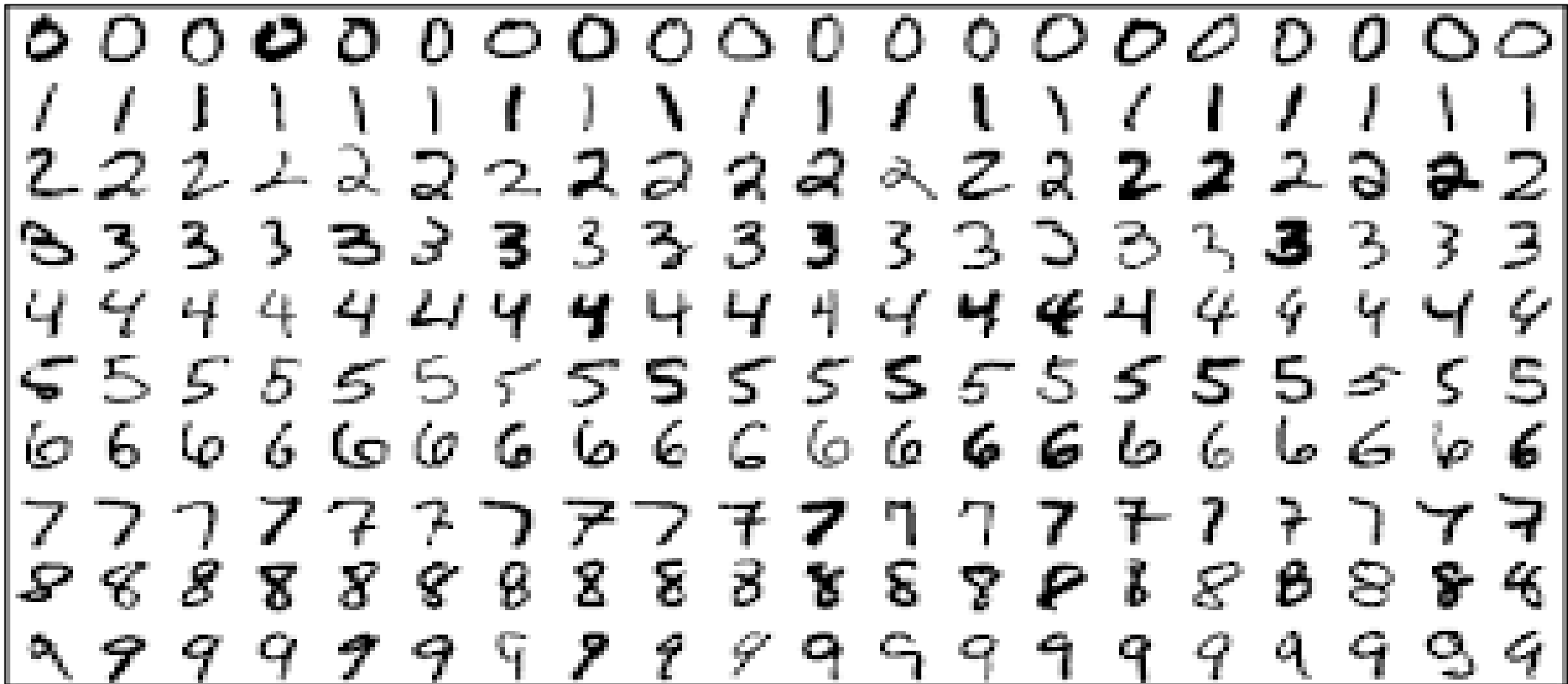
Training a network with data

Lab



HANDWRITTEN DIGIT RECOGNITION

HELLO WORLD of machine learning



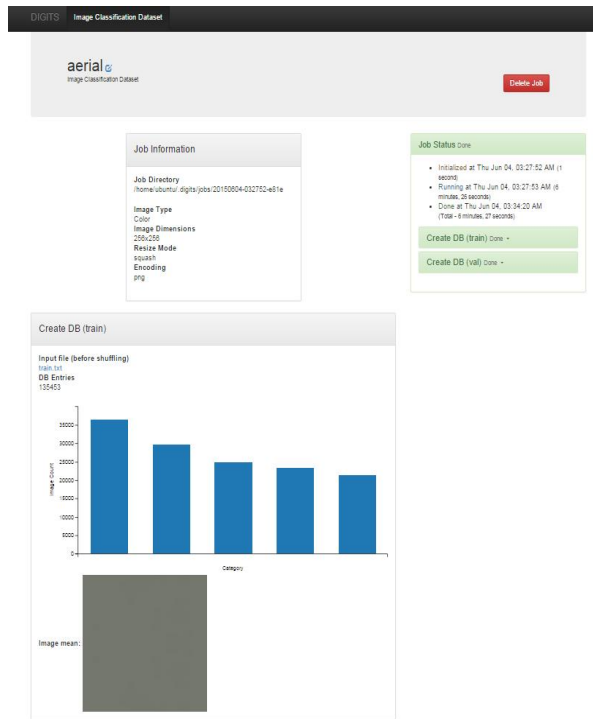
WHAT THIS LAB IS

- An introduction to:
 - Workflow of training a network
 - Understanding the results
- Hands-on exercises using DIGITS for computer vision and classification

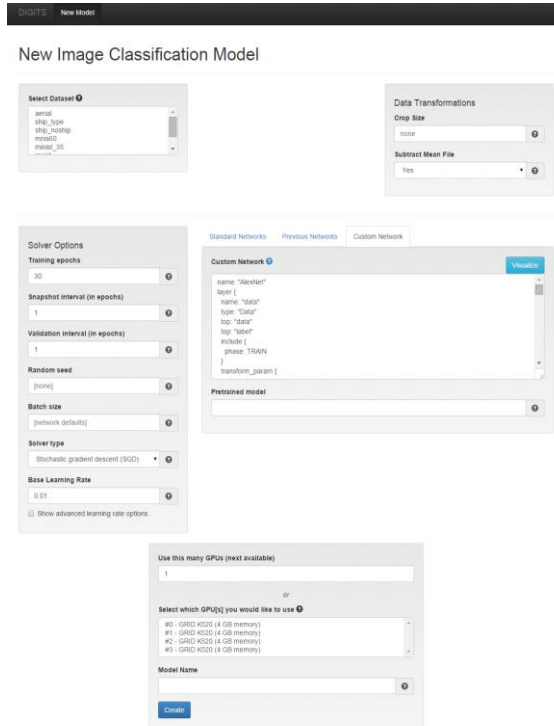
NVIDIA DIGITS

Interactive Deep Learning GPU Training System

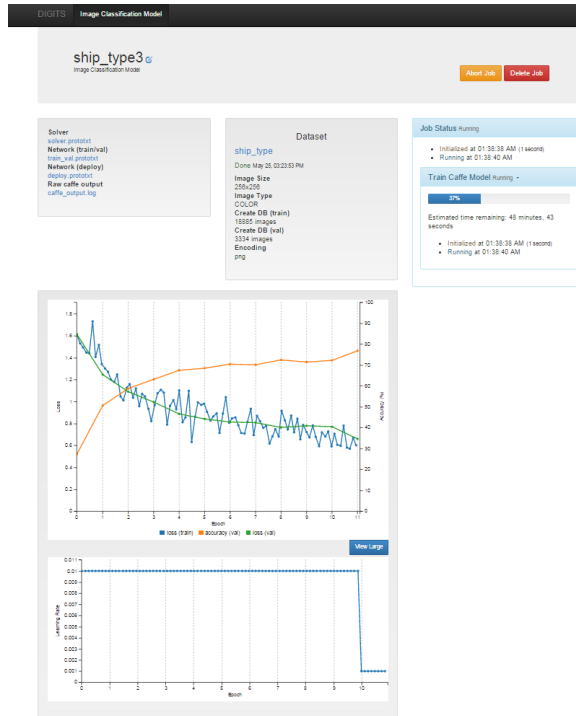
Process Data



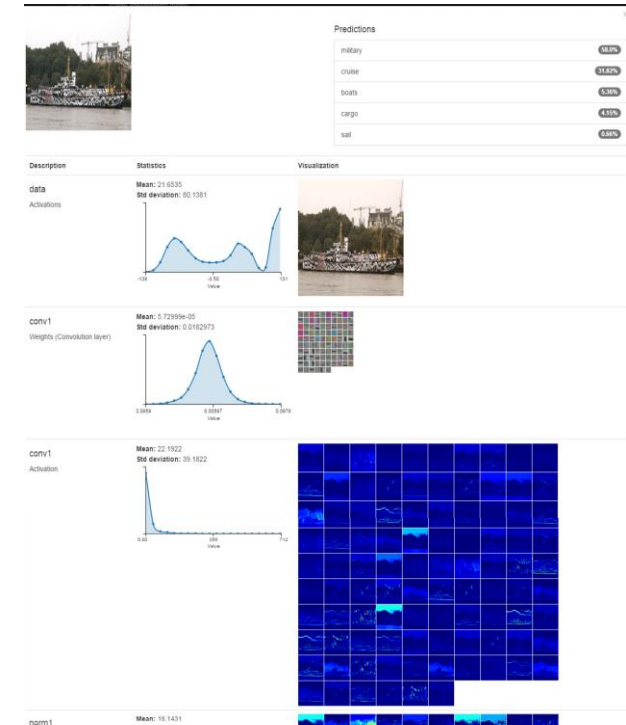
Configure DNN



Monitor Progress



Visualization



WHAT THIS LAB IS NOT

- Intro to machine learning from first principles
- Rigorous mathematical formalism of neural networks
- Survey of all the features and options of tools and frameworks

LAB OVERVIEW

- Learn about the workflow of Deep Learning
 - Load data
 - Expose a network to data
 - Evaluate model results
 - Try different techniques to improve initial results

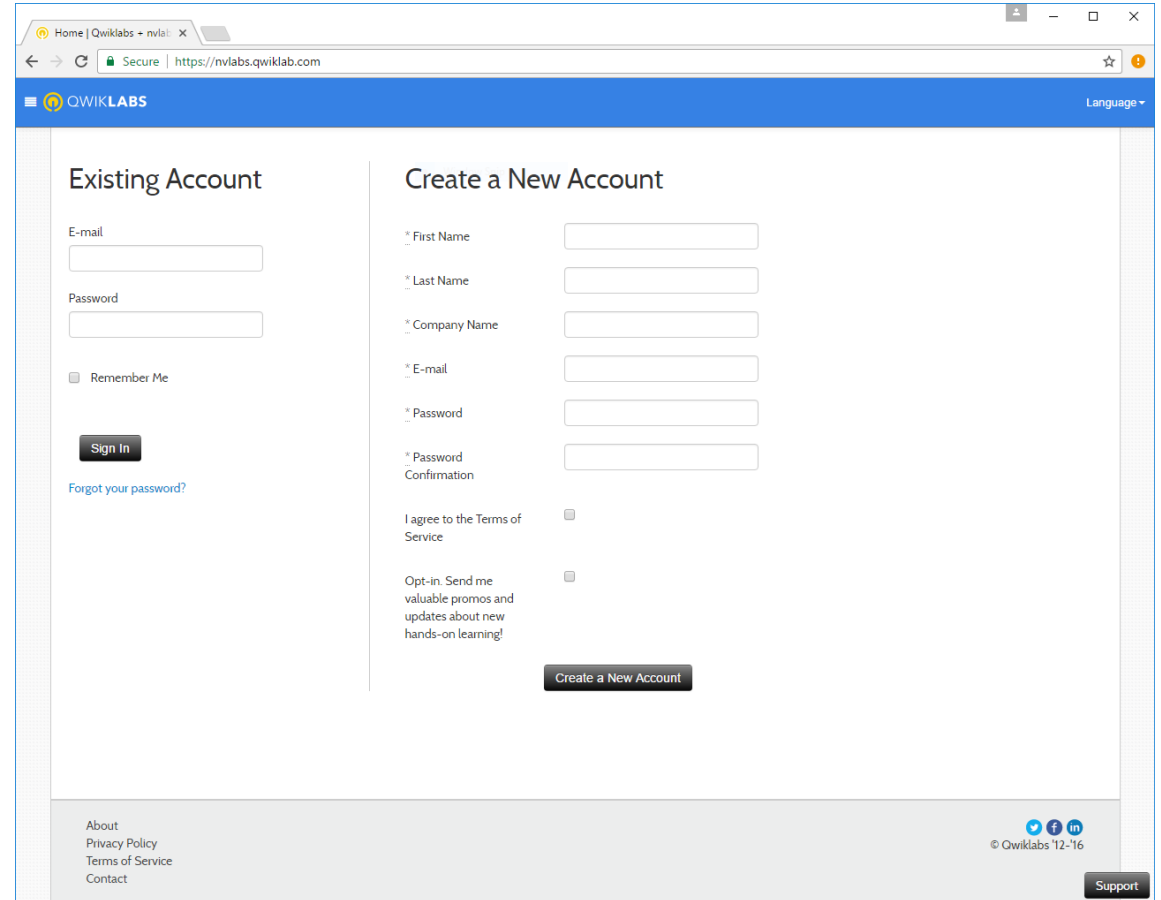
LAB OVERVIEW

- We have clean, labeled data
- Our task is “supervised image classification”
- We will train a live network on GPU

LAUNCHING THE LAB

NAVIGATING TO QWIKLABS

1. Navigate to:
<https://nvlabs.qwiklab.com>
1. Login or create a new account



The screenshot shows a web browser window with the URL <https://nvlabs.qwiklab.com>. The page has a blue header with the Qwiklabs logo and a language dropdown. The main content area is divided into two columns: 'Existing Account' and 'Create a New Account'.

Existing Account:

- E-mail:
- Password:
- ☐ Remember Me
-
- [Forgot your password?](#)

Create a New Account:

- * First Name:
- * Last Name:
- * Company Name:
- * E-mail:
- * Password:
- * Password Confirmation:
- ☐ I agree to the Terms of Service
- ☐ Opt-in. Send me valuable promos and updates about new hands-on learning!
-

Footer:

- About
- Privacy Policy
- Terms of Service
- Contact
- © Qwiklabs '12-'16
- [Support](#)

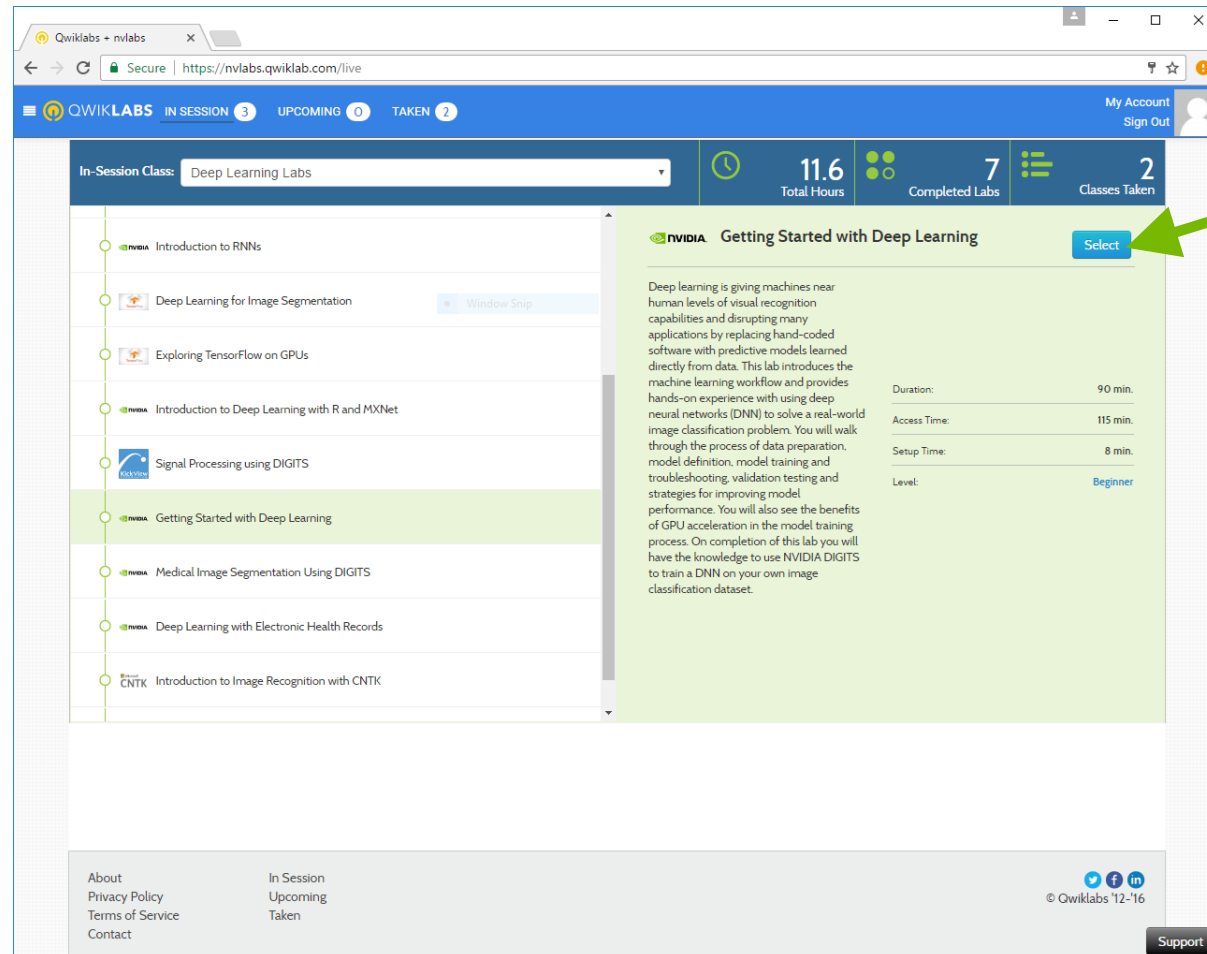
ACCESSING LAB ENVIRONMENT

3. Select the event specific In-Session Class in the upper left

3. Click the “Image Classification with DIGITS” Class from the list

The screenshot shows the Qwiklabs interface. At the top, there's a navigation bar with 'QWIKLABS', 'IN SESSION 3', 'UPCOMING 0', and 'TAKEN 2'. On the right, there's a user profile section with 'My Account' and 'Sign Out'. Below the navigation bar, there's a section for 'In-Session Class' with a dropdown menu set to 'Deep Learning Labs'. To the right of this, there are statistics: '11.6 Total Hours', '7 Completed Labs', and '2 Classes Taken'. Below the 'In-Session Class' dropdown, there's a list of available labs. The 'Getting Started with Deep Learning' lab is highlighted in green. To the right of this list, there's a detailed view of the selected lab, including a description, duration (90 min), access time (115 min), setup time (8 min), and level (Beginner). At the bottom of the page, there's a footer with links for 'About', 'Privacy Policy', 'Terms of Service', 'Contact', 'In Session', 'Upcoming', 'Taken', and social media icons.

LAUNCHING THE LAB ENVIRONMENT

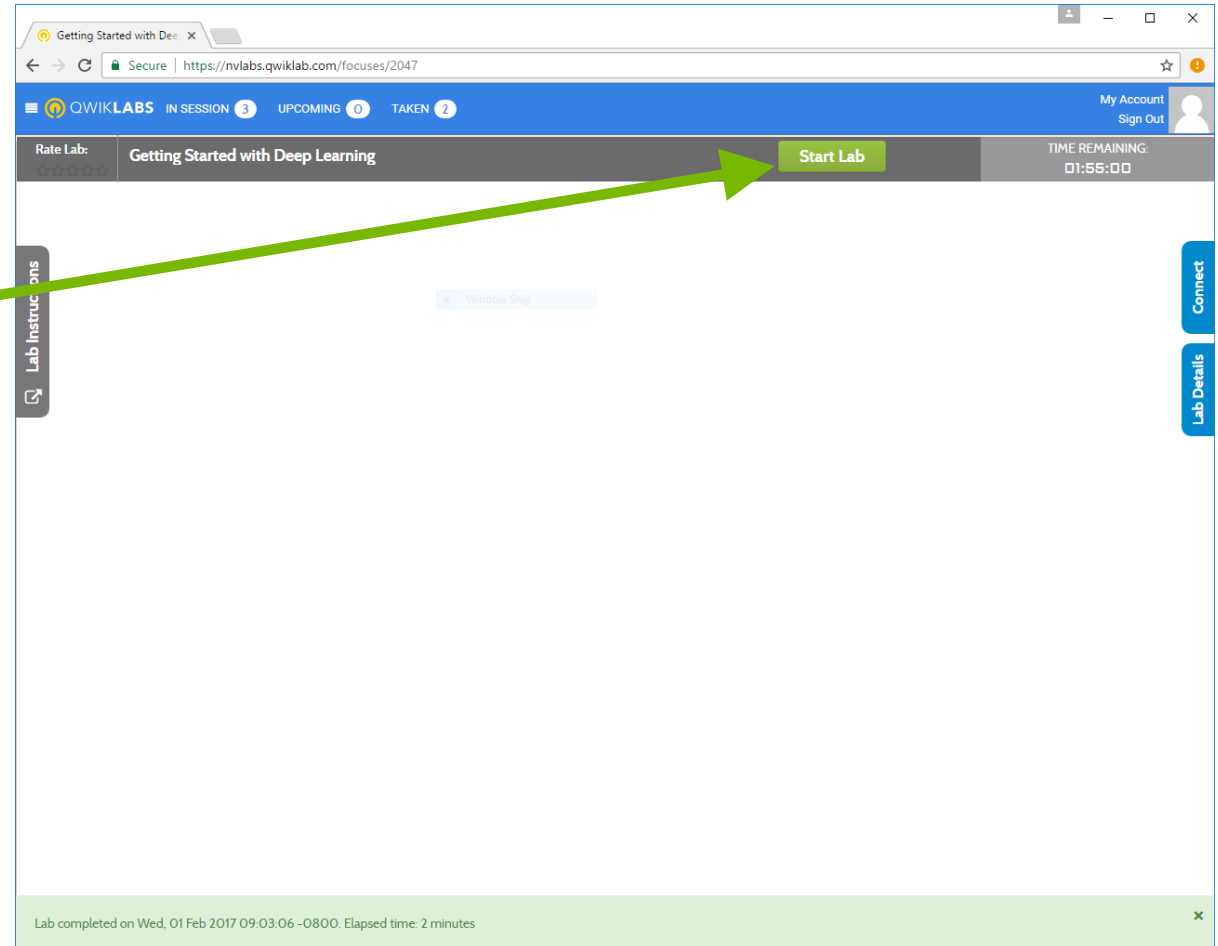


5. Click on the Select button to launch the lab environment

- After a short wait, lab Connection information will be shown
- Please ask Lab Assistants for help!

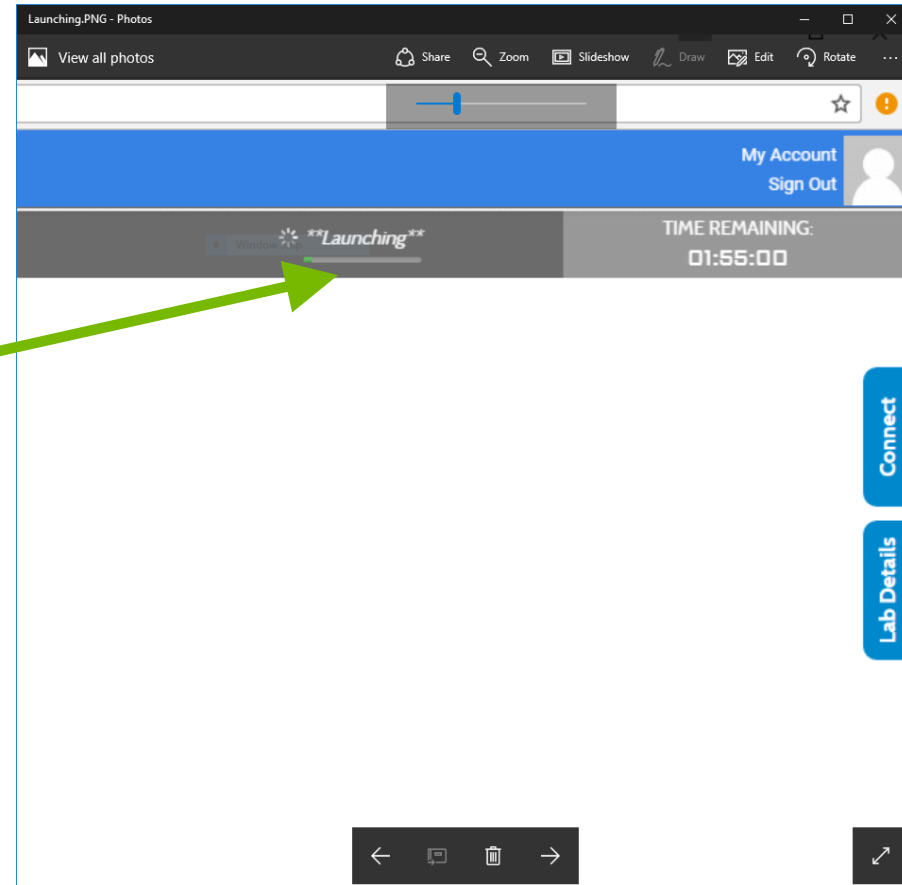
LAUNCHING THE LAB ENVIRONMENT

6. Click on the Start Lab button



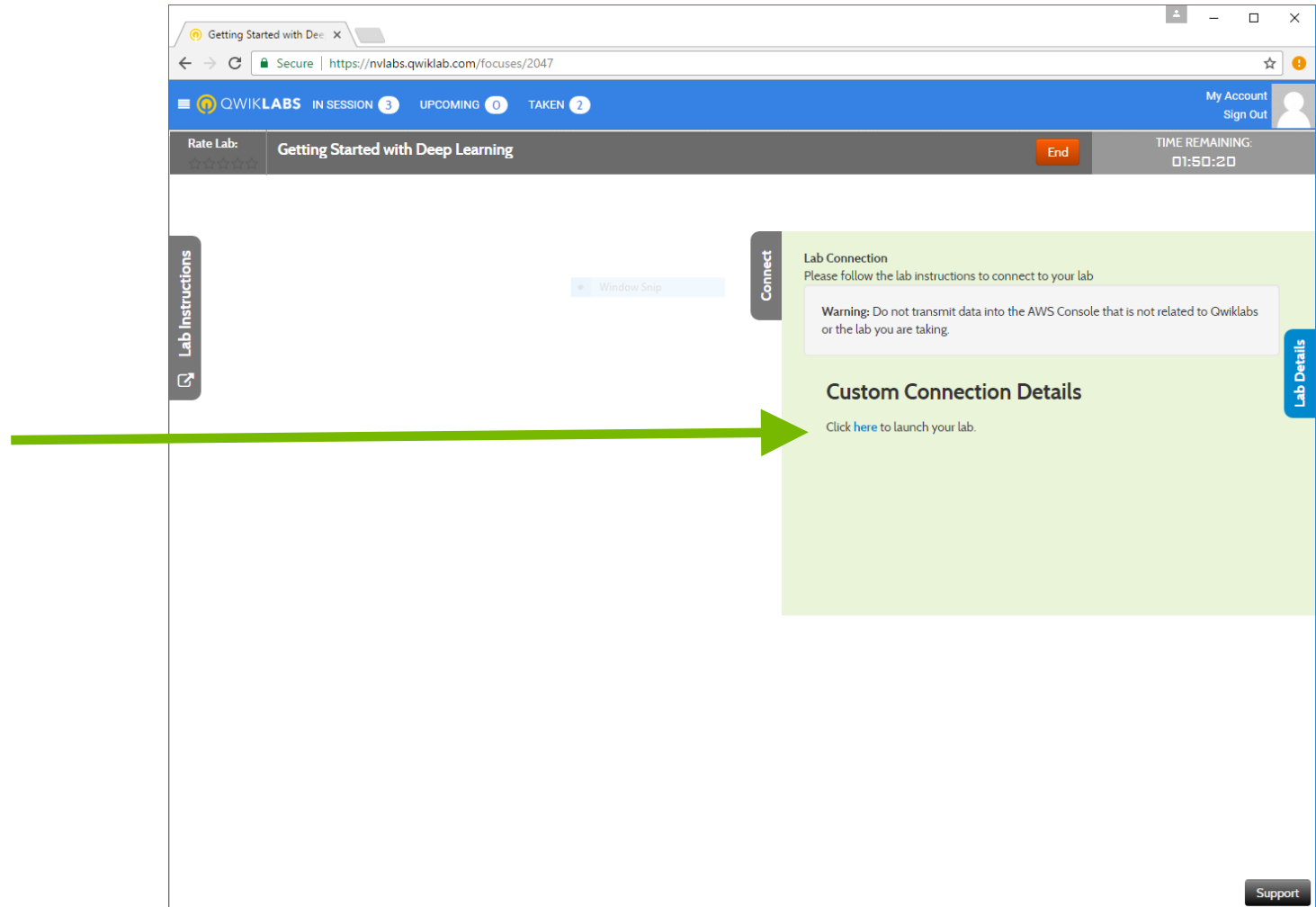
LAUNCHING THE LAB ENVIRONMENT

You should see that the lab environment is “launching” towards the upper-right corner



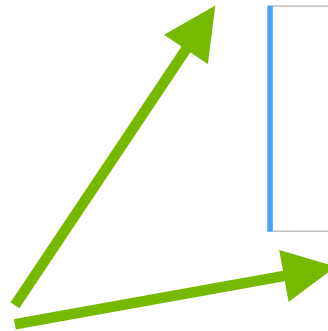
CONNECTING TO THE LAB ENVIRONMENT

7. Click on “here” to access your lab environment / Jupyter notebook



CONNECTING TO THE LAB ENVIRONMENT

You should see your
“Image Classification
with DIGITS” Jupyter
notebook



jupyter Image Classification with DIGITS - Training a model Last Checkpoint: Last Wednesday at 9:17 AM (unsaved changes)

File Edit View Insert Cell Kernel Help Not Trusted Python 2 C

Markdown





Image Classification with DIGITS

An introduction to Deep Learning

In this lab, you'll learn to **train** a **neural network** using clean **labeled data**. We'll introduce deep learning through the task of **supervised image classification**, where, given a large number of images and their labels, you'll build a tool that can *predict* labels of *new* images.

The intent is to build the skills to start experimenting with deep learning. You'll examine:

- What it means to *train* vs. to *program*
- The role of data in artificial intelligence
- How to load data for training a neural network
- The role of a *network* in deep learning
- How to train a model with data

At the end of this lab, you'll have a trained neural network that can successfully classify images to solve a classic deep learning challenge:

How can we digitize handwriting?

Training vs. programming

The fundamental difference between artificial intelligence (AI) and traditional programming is that AI *learns* while traditional algorithms are *programmed*. Let's examine the difference through an example:

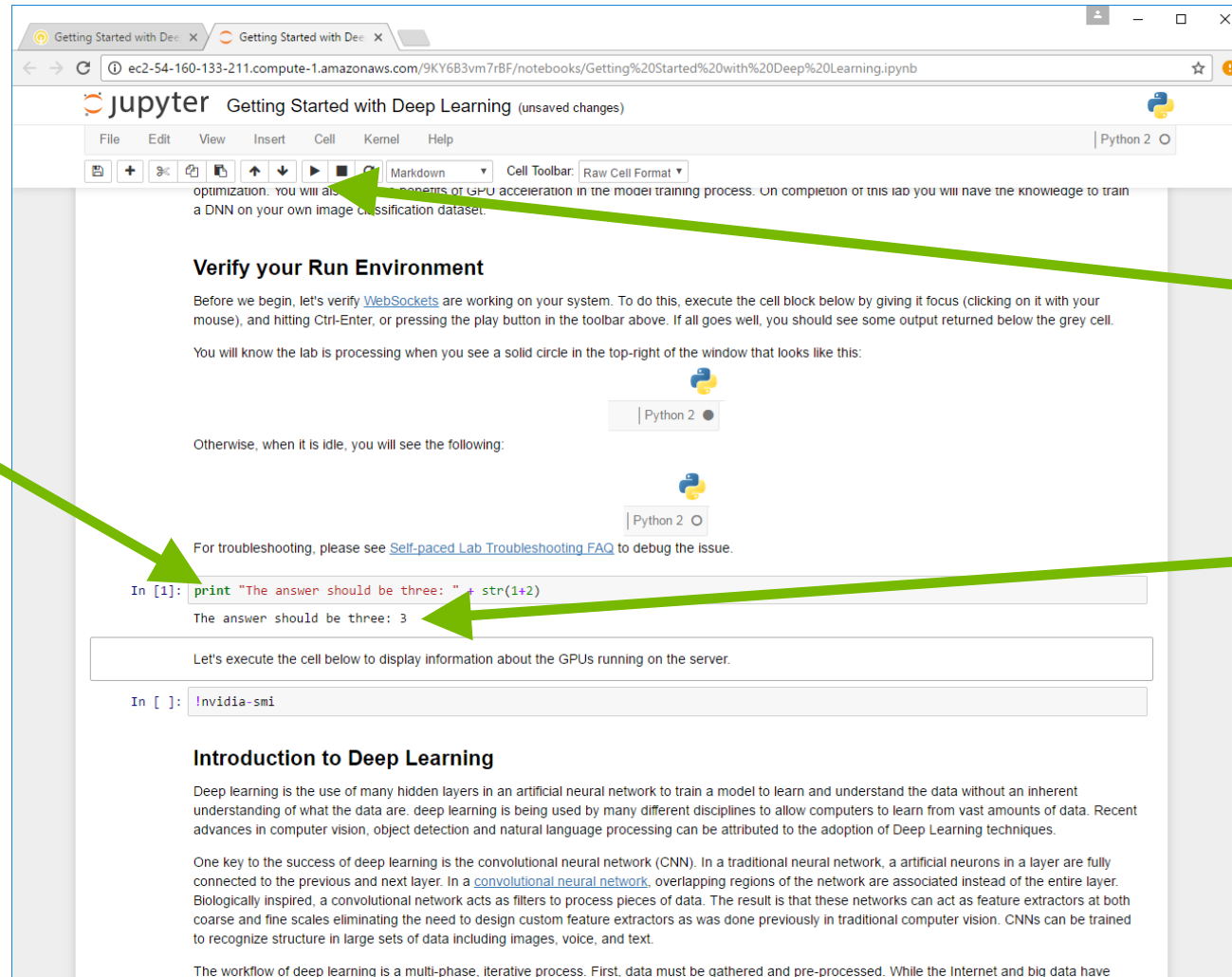
Imagine you were asked to give a robot instructions to make a sandwich using traditional computer programming, instruction by instruction. How might you start?

JUPYTER NOTEBOOK

1. Place your cursor in the code

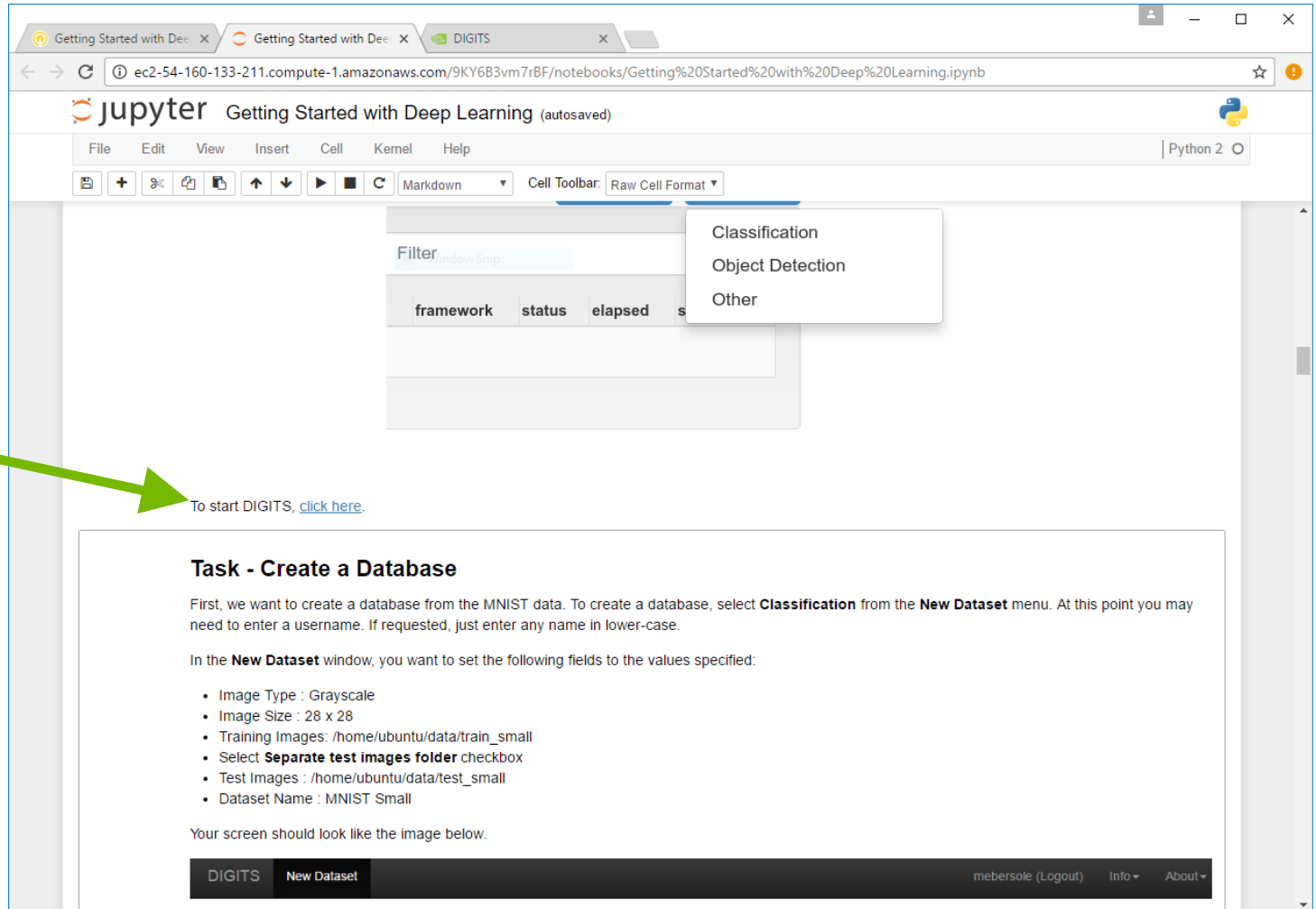
2. Click the “run cell” button

2. Confirm you receive the same result



STARTING DIGITS

Instruction in
Jupyter notebook
will link you to
DIGITS



The screenshot shows a Jupyter notebook titled "Getting Started with Deep Learning (autosaved)" running on a Python 2 kernel. The interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Help) and a toolbar with various icons. A dropdown menu is open, showing options: "Classification", "Object Detection", and "Other". Below the menu, there is a table with columns: "framework", "status", "elapsed", and "s". A green arrow points from the text "Instruction in Jupyter notebook will link you to DIGITS" to a link in the notebook content that says "To start DIGITS, [click here](#)."

To start DIGITS, [click here](#).

Task - Create a Database

First, we want to create a database from the MNIST data. To create a database, select **Classification** from the **New Dataset** menu. At this point you may need to enter a username. If requested, just enter any name in lower-case.

In the **New Dataset** window, you want to set the following fields to the values specified:

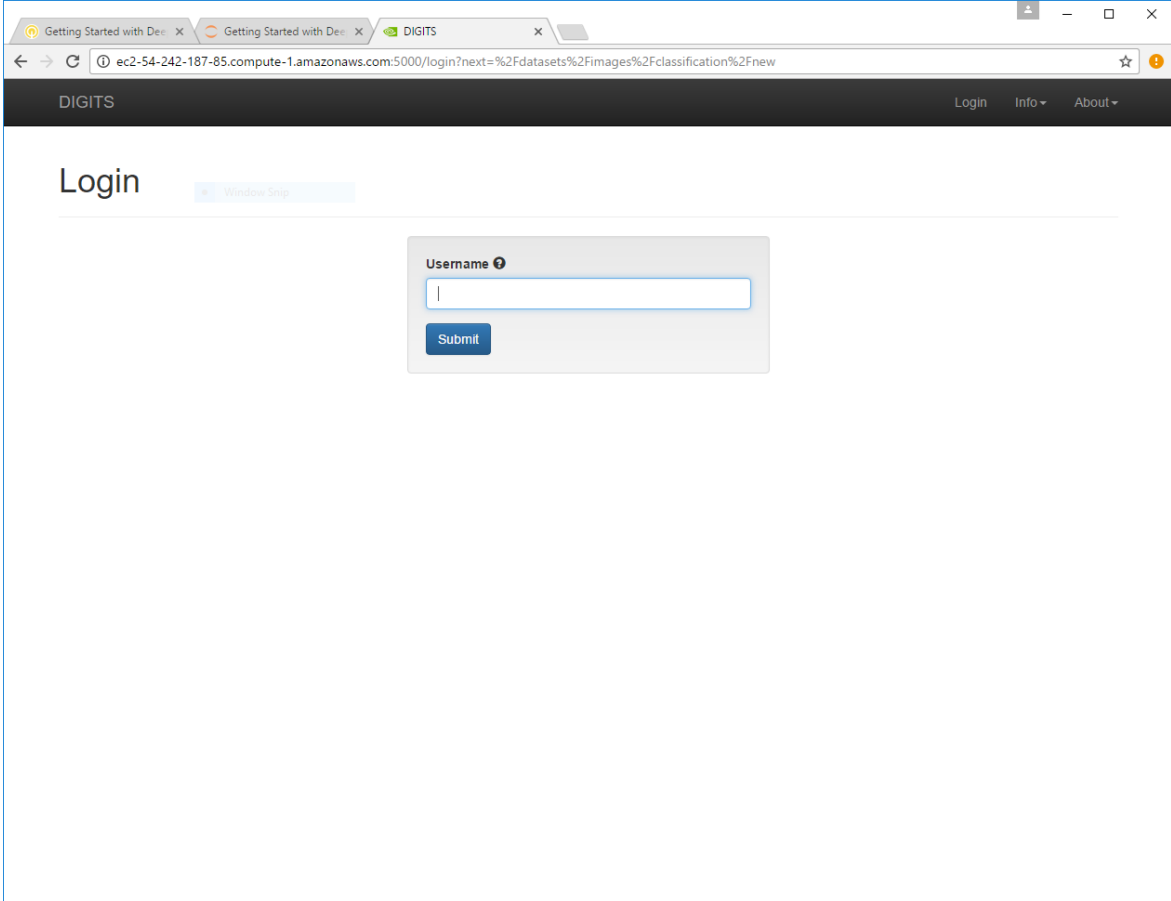
- Image Type : Grayscale
- Image Size : 28 x 28
- Training Images : /home/ubuntu/data/train_small
- Select **Separate test images folder** checkbox
- Test Images : /home/ubuntu/data/test_small
- Dataset Name : MNIST Small

Your screen should look like the image below.

DIGITS New Dataset mebersole (Logout) Info About

ACCESSING DIGITS

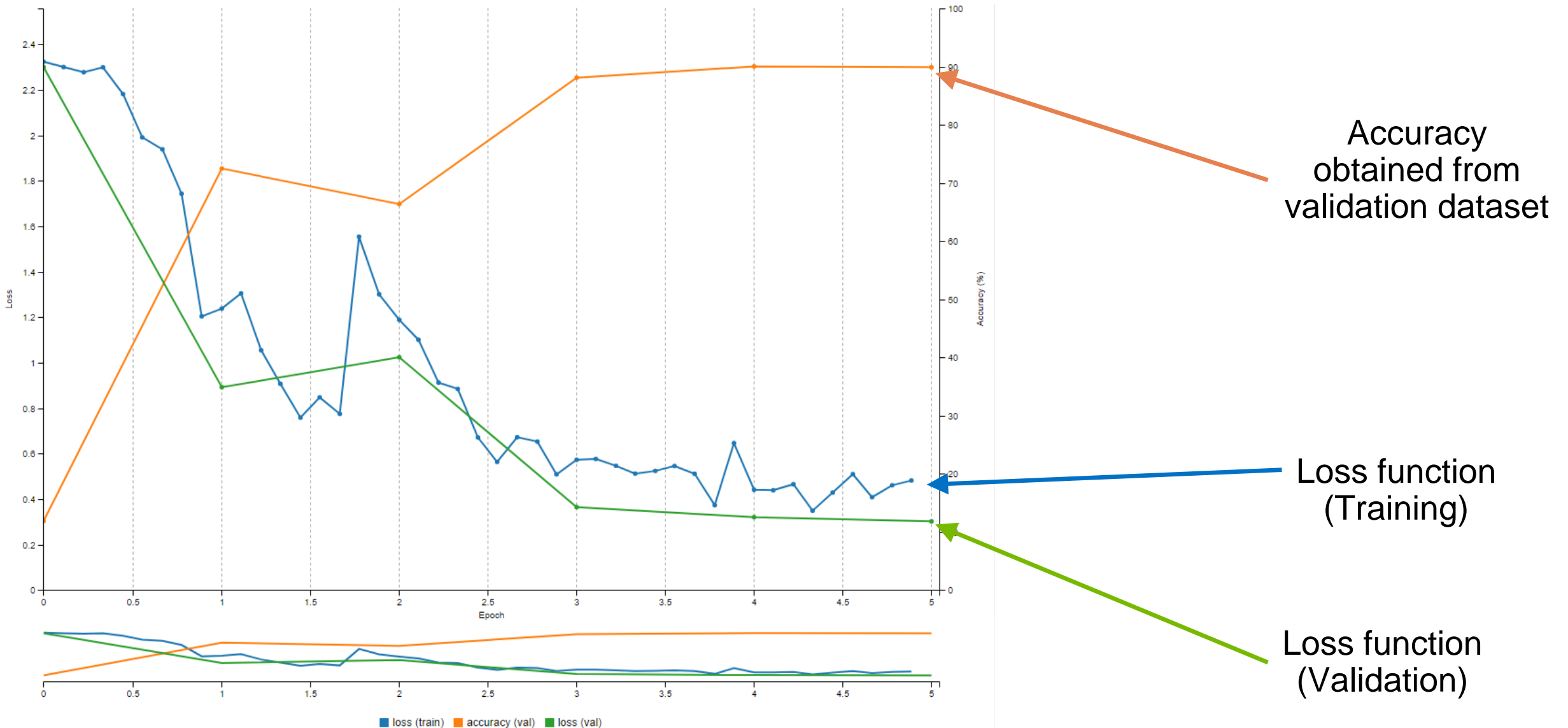
- Will be prompted to enter a username to access DIGITS
 - Can enter any username
 - Use lower case letters



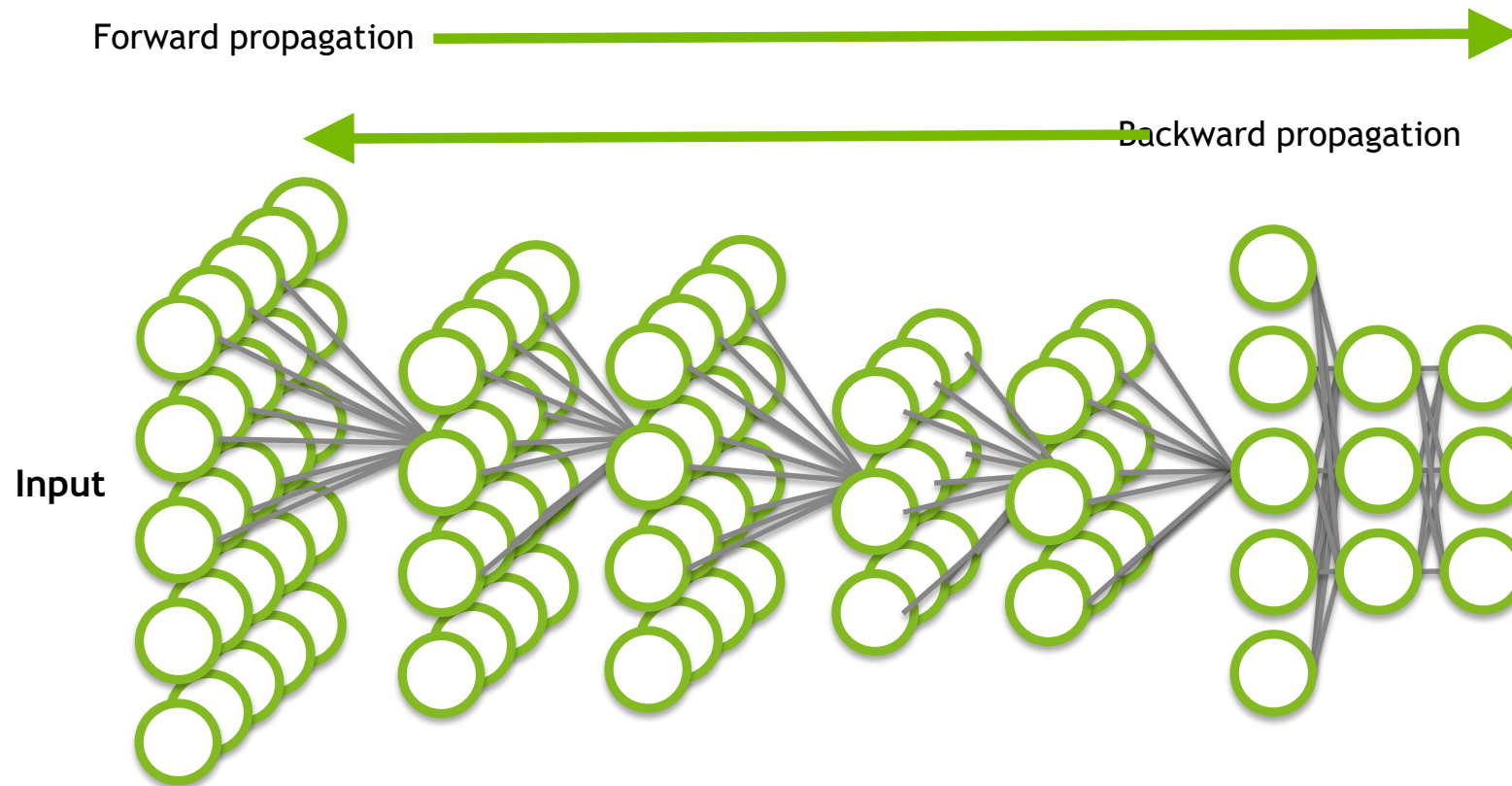
The screenshot shows a web browser window with the address bar displaying the URL: `ec2-54-242-187-85.compute-1.amazonaws.com:5000/login?next=%2Fdatasets%2Fimages%2Fclassification%2Fnew`. The browser has three tabs open: "Getting Started with De...", "Getting Started with De...", and "DIGITS". The page title is "DIGITS" and the navigation bar includes "Login", "Info", and "About". The main content area is titled "Login" and features a "Window Snip" button. Below this is a "Username" label with a question mark icon, a text input field, and a "Submit" button.

Evaluating Performance

EVALUATE THE MODEL



DEEP LEARNING APPROACH - TRAINING



Process

- Forward propagation yields an inferred label for each training image
- Loss function used to calculate difference between known label and predicted label for each image
- Weights are adjusted during backward propagation
- Repeat the process

Next Challenges

Ideas?



- Increase accuracy and confidence with similar data





- Generalize performance to more diverse data

Lab Review

More data








Full dataset (10 epochs)

- 99% of accuracy achieved
- No improvements in recognizing real-world images

	Defaults	Training+Data 
	1 : 99.90 %	0 : 93.11 %
	2 : 69.03 %	2 : 87.23 %
	8 : 71.37 %	8 : 71.60 %
	8 : 85.07 %	8 : 79.72 %
	0 : 99.00 %	0 : 95.82 %
	8 : 99.69 %	8 : 100.0 %
	8 : 54.75 %	2 : 70.57 %

DATA AUGMENTATION

Adding inverted images (10 epochs)

	SMALL DATASET	FULL DATASET	+INVERTED
	1 : 99.90 %	0 : 93.11 %	1 : 90.84 %
	2 : 69.03 %	2 : 87.23 %	2 : 89.44 %
	8 : 71.37 %	8 : 71.60 %	3 : 100.0 %
	8 : 85.07 %	8 : 79.72 %	4 : 100.0 %
	0 : 99.00 %	0 : 95.82 %	7 : 82.84 %
	8 : 99.69 %	8 : 100.0 %	8 : 100.0 %
	8 : 54.75 %	2 : 70.57 %	2 : 96.27 %

DATA AUGMENTATION

Adding Inverted Images

















DIGITS Image Classification Dataset smorino (Logout) Info

Exploring MNIST invert (train_db) images

Show all images or filter by class: 0 1 2 3 4 5 6 7 8 9

Items per page: 10 - 25 - 50 - 100








« 0 1 2 3 4 5 ... 3600 »

			
2	9	7	3
			
1	4	6	5
			
5	3	8	2
			
3	1	8	6

```
keras.preprocessing.image.ImageDataGenerator(featurewise_center=False,
samplewise_center=False,
featurewise_std_normalization=False,
samplewise_std_normalization=False,
zca_whitening=False,
zca_epsilon=1e-6,
rotation_range=0.,
width_shift_range=0.,
height_shift_range=0.,
shear_range=0.,
zoom_range=0.,
channel_shift_range=0.,
fill_mode='nearest',
cval=0.,
horizontal_flip=False,
vertical_flip=False,
rescale=None,
preprocessing_function=None,
data_format=K.image_data_format())
```

MODIFIED NETWORK

Adding filters and ReLU layer (10 epochs)

	SMALL DATASET	FULL DATASET	+INVERTED	ADDING LAYER
	1 : 99.90 %	0 : 93.11 %	1 : 90.84 %	1 : 59.18 %
	2 : 69.03 %	2 : 87.23 %	2 : 89.44 %	2 : 93.39 %
	8 : 71.37 %	8 : 71.60 %	3 : 100.0 %	3 : 100.0 %
	8 : 85.07 %	8 : 79.72 %	4 : 100.0 %	4 : 100.0 %
	0 : 99.00 %	0 : 95.82 %	7 : 82.84 %	2 : 62.52 %
	8 : 99.69 %	8 : 100.0 %	8 : 100.0 %	8 : 100.0 %
	8 : 54.75 %	2 : 70.57 %	2 : 96.27 %	8 : 70.83 %

MODIFY THE NETWORK

Necessary for less “solved” challenges.

```
layer {
  name: "pool1"
  type: "Pooling"
  ...
}

layer {
  name: "reluP1"
  type: "ReLU"
  bottom: "pool1"
  top: "pool1"
}

layer {
  name: "reluP1"
```

```
layer {
  name: "conv1"
  type: "Convolution"
  ...
  convolution_param {
    num_output: 75
  }
  ...
}

layer {
  name: "conv2"
  type: "Convolution"
  ...
  convolution_param {
    num_output: 100
  }
  ...
}
```

Next Steps

- Experiment with Image Classification
 - Different datasets
 - Increase performance
- Learn to train existing networks with data for other challenges
- Learn about network construction
- Learn about how to create an image classifier with other frameworks
 - Caffe/Keras
 - Tensorflow
 - Etc.