

## 1: Getting Started

In this lesson you're going to hit the ground running – in the first five minutes you'll see a complete end-to-end example of training and using a model that's so advanced it was considered at the cutting edge of research capabilities in 2015. So, let's get started!

### Video

Click the video below to play. Once it's playing, you'll see a little rectangle in the bottom-right—click it to make the video full-screen. Press Esc to remove full-screen view. Press c to turn subtitles on/off.

This lesson is based partly on chapter 1 of the book.

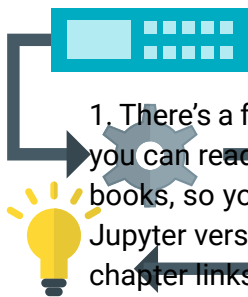
### How to complete lesson 1

Every lesson includes lots of hands-on exercises for you to try. Most of these are run in interactive notebooks, all of which are available on Kaggle. If you don't work through the notebooks yourself, you're not going to get nearly as much out of this course—so that means you need to get set up on Kaggle. We have a page to help you get going with Kaggle: [click here to go there now](#). Instead of using Kaggle, another great option is Paperspace Gradient. If you don't have a Paperspace account yet, sign up with this link to get \$10 credit (and we get a credit too).

Once you've got your Kaggle account set up, you'll need to get familiar with Jupyter Notebook, which is the platform we use for most of this course (and which most deep learning researchers and engineers use for their work). Jupyter is the most popular tool for doing data science in Python, for good reason. It is powerful, flexible, and easy to use. We think you will love it! Since the most important thing for learning deep learning is writing code and experimenting, it's important that you have a great platform for experimenting with code. If you haven't used it before, we've provided this to help you get started: Jupyter Notebook 101.

OK, now that you have your Kaggle account and know how to use Jupyter, you're ready to open the notebook for this lesson: [here it is](#). For every lesson, you can find links to all notebooks used in the Resources section of the lesson web page. For instance, for lesson 1, you'll see that section immediately below this one.

As well as watching the video and working through the notebooks, you should also read the relevant chapter(s) of the fast.ai book, *Practical Deep Learning for Coders*. Each lesson will tell you what chapter you need to read, just below the video. For this lesson, it's chapter



1. There's a few ways to read the book – you can buy it as a paper book or Kindle ebook, or you can read it for free as a Jupyter notebook. The whole book is written as Jupyter notebooks, so you can also execute all the code in the book yourself. To go to the interactive Jupyter version of any chapter, click The book in the left sidebar, where you'll find a list of chapter links. You'll also find links to read-only versions of each chapter there.

## Resources

- Kaggle notebooks for this lesson:
  - Is it a bird? Creating a model from your own data
  - Jupyter Notebook 101
- The fastai book:
  - Published version
  - Free notebook version
  - Chapter 1 notebook
- Repo containing all lesson notebooks
- Solutions to chapter 1 questions from the book

## Links

You'll see that fast.ai's way of teaching is very different to what you might be used to, if you did a technical degree at university. Nearly all technical subjects at university are taught "bottom up": start with basic foundations, and gradually work up to complete useful solutions to real world problems. But we go "top down": start with complete useful solutions to real world problems, and gradually work down to the basic foundations. Education experts recommend this approach for more effective learning. For more information, have a look at this article that discusses the fast.ai teaching philosophy: [Providing a Good Education in Deep Learning](#).

- How to learn - highly recommended books for fast.ai students
  - Meta Learning
  - A Mathematician's Lament by Paul Lockhart
  - Making Learning Whole by David Perkins
- Jupyter
  - Presentations: RISE
  - Blogging: fastpages
  - The notebooks used to create the fastai library
  - nbdev - the system we built to create Python libraries using Jupyter
- Fastai: A Layered API for Deep Learning paper: [Information Journal](#) or [arxiv](#) or [fast.ai](#)
- Dall-e 2 illustrations of Twitter bios
- timm: PyTorch Image Models



If you need help

There's lot of helpful people, and helpful answers to past questions, on the fast.ai forums.



There are special help topics for beginner questions, to ensure that your questions aren't missed:

- Help: Setup
- Help: Creating a dataset, and using Gradio / Spaces
- Help: Using Colab or Kaggle
- Help: Python, git, bash, etc
- Help: SGD and Neural Net foundations
- Help: Basics of fastai, PyTorch, numpy, etc
- Help: Beginner questions that don't fit elsewhere

Bibliography