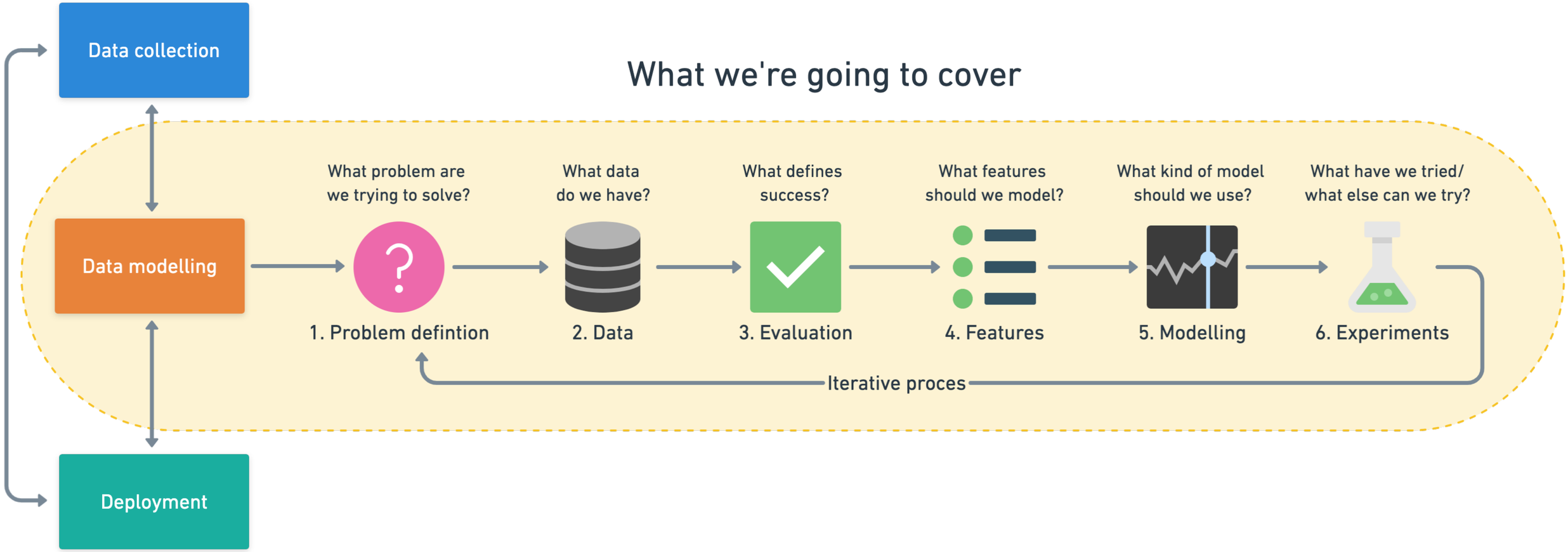
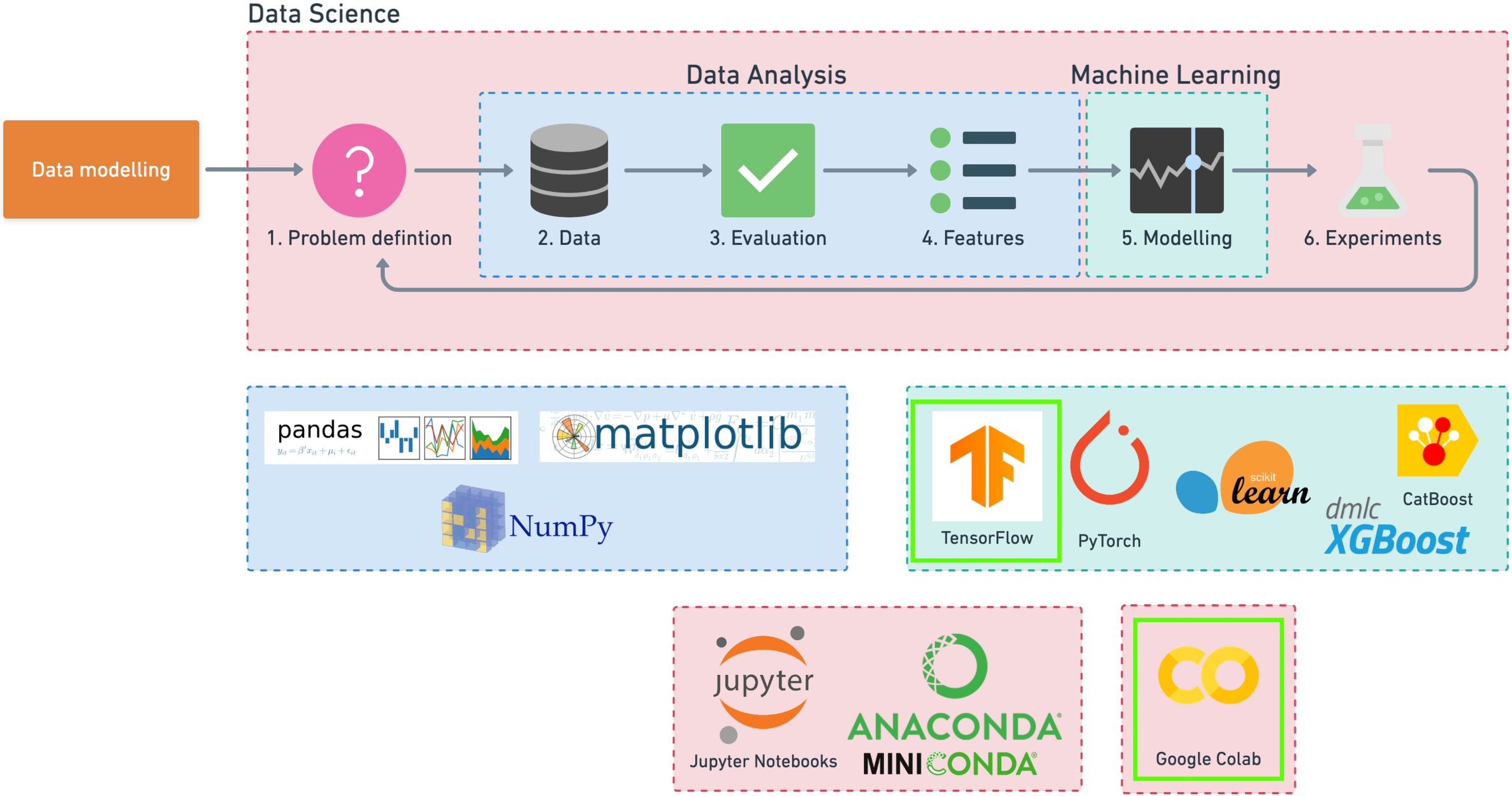
Steps in a full machine learning project

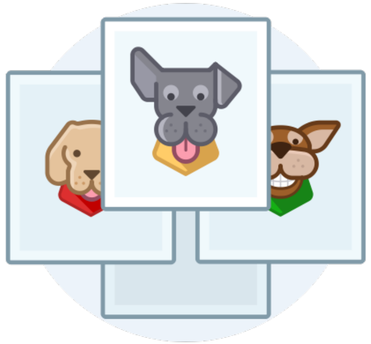


Tools you can use



A new contender!

What is TensorFlow?



From: daniel@mrdbourke.com
Hey Daniel,

First of all, this machine learning course is incredible!
I can't wait to use what I've learned!

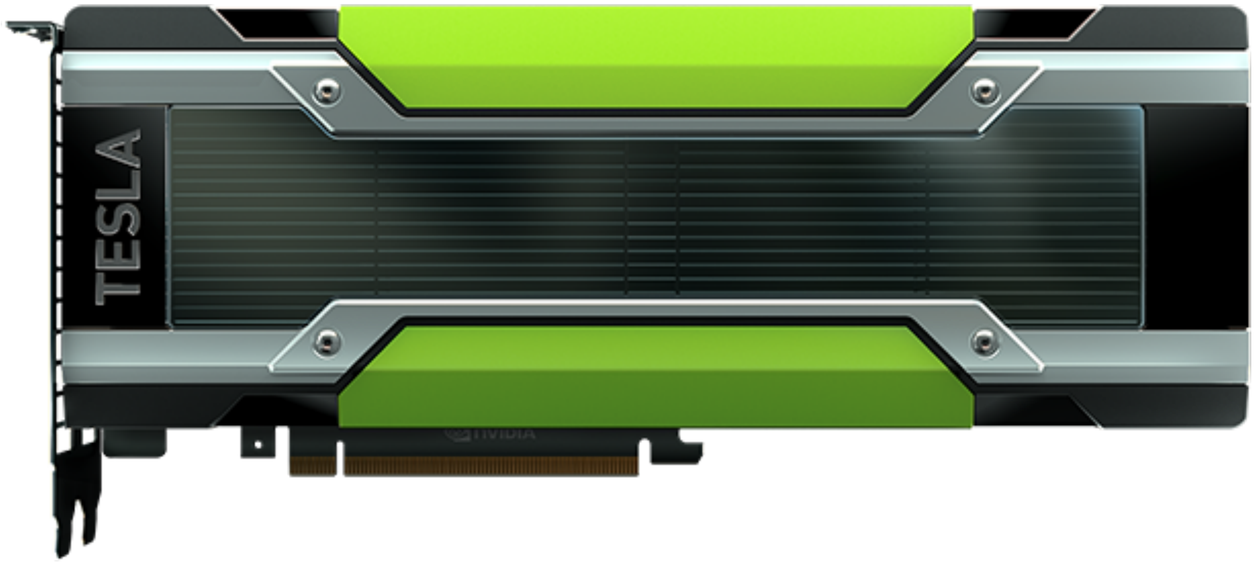
Unstructured

?

Why TensorFlow?

- Write fast deep learning code in Python (able to run on a GPU)
- Able to access many pre-built deep learning models
- Whole stack: preprocess, model, deploy
- Originally designed and used in-house by Google (now open-source)

What is a GPU?



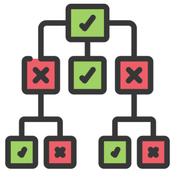
Choosing a model (throwback)

Problem 1 (structured)

Model 1

Structured Data

CatBoost



Random Forest

Problem 2 (unstructured)

Model 2

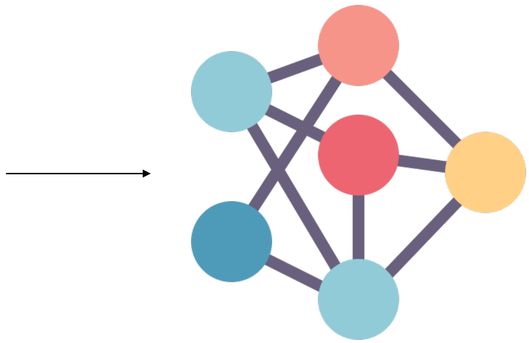
Unstructured Data

Deep Learning

Transfer Learning

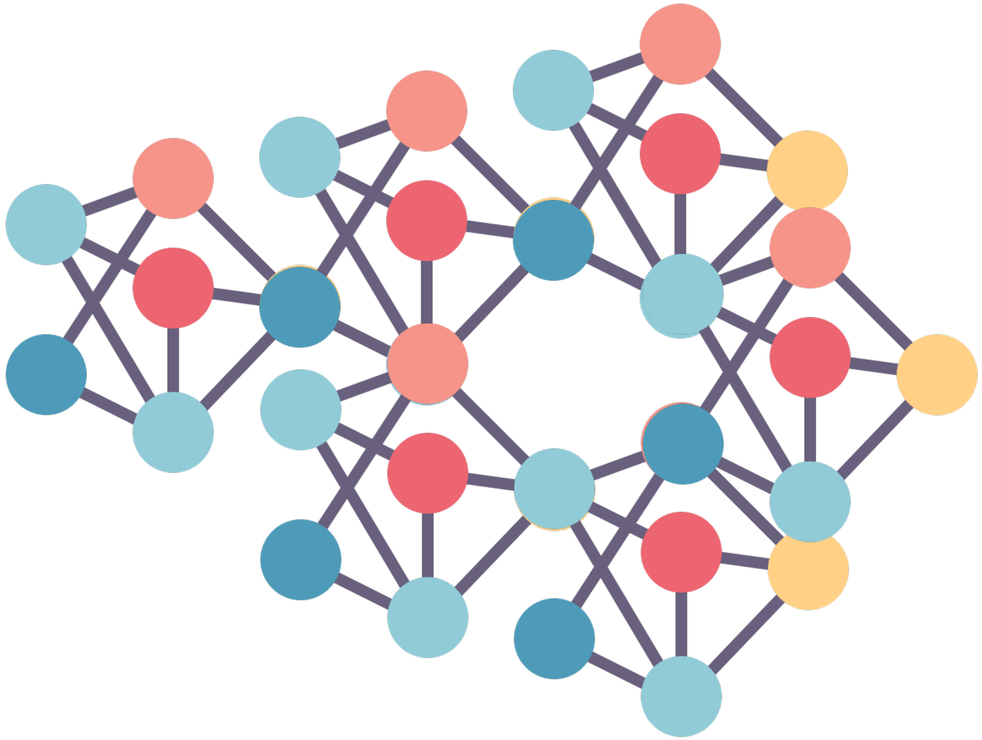
TensorFlow Hub

What is deep learning? What are neural networks?



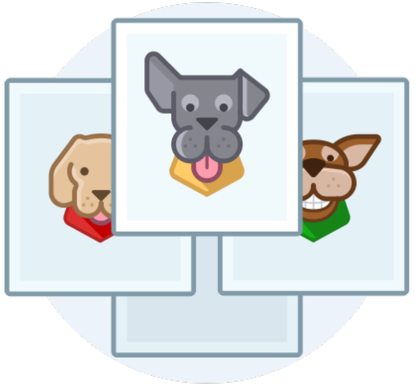
Labrador

What is deep learning? What are neural networks?



Labrador

What kind of deep learning problems are there?



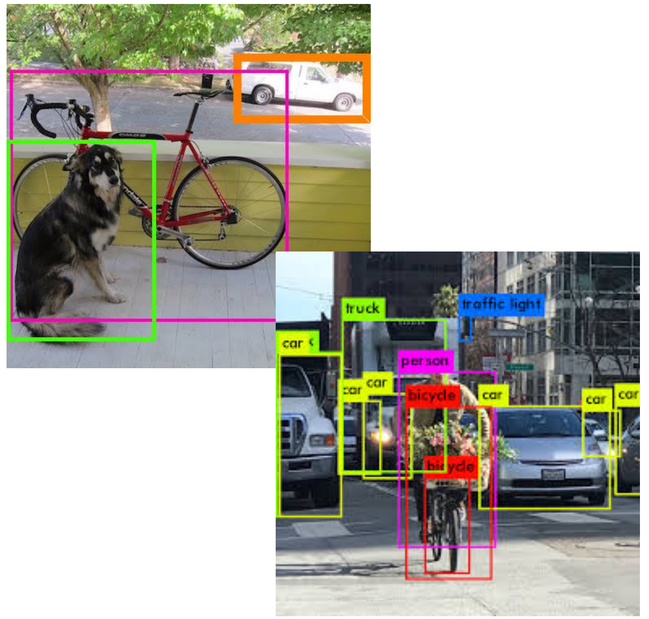
From: daniel@mrdbourke.com
Hey Daniel,

First of all, this machine learning course is incredible!
I can't wait to use what I've learned!

Classification

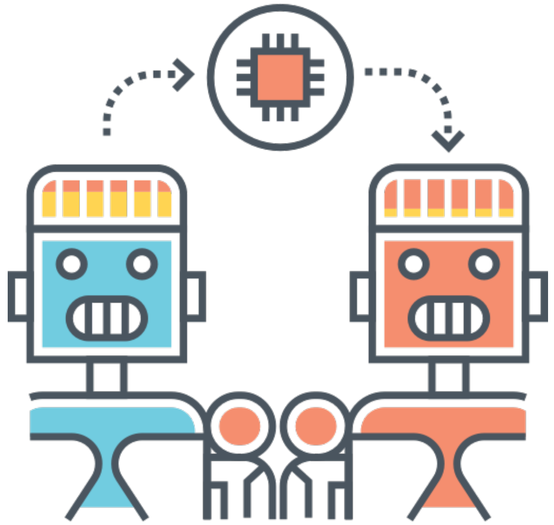
Hey Siri, where's the nearest cafe?

Sequence to sequence (seq2seq)



Object detection

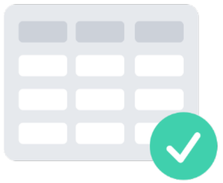
What is transfer learning? Why use transfer learning?



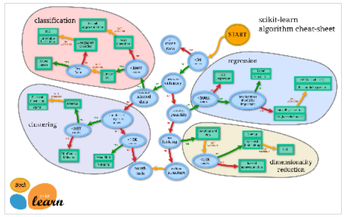
- Take what you know in one domain and apply it to another.
- Starting from scratch can be expensive and time consuming.
- Why not take advantage of what's already out there?

What are we going to cover?

A Scikit-Learn workflow



1. Get data ready



2. Pick a model
(to suit your problem)

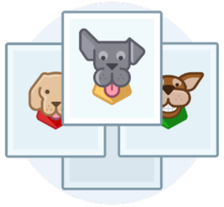
3. Fit the model to the data
and make a prediction

4. Evaluate the model

5. Improve through
experimentation

6. Save and reload
your trained model

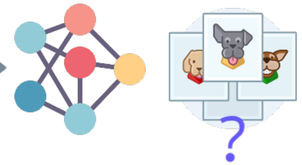
A TensorFlow workflow



1. Get data ready
(turn into Tensors)

TensorFlow
Hub

2. Pick a model
(to suit your problem)



3. Fit the model to the data
and make a prediction

4. Evaluate the model

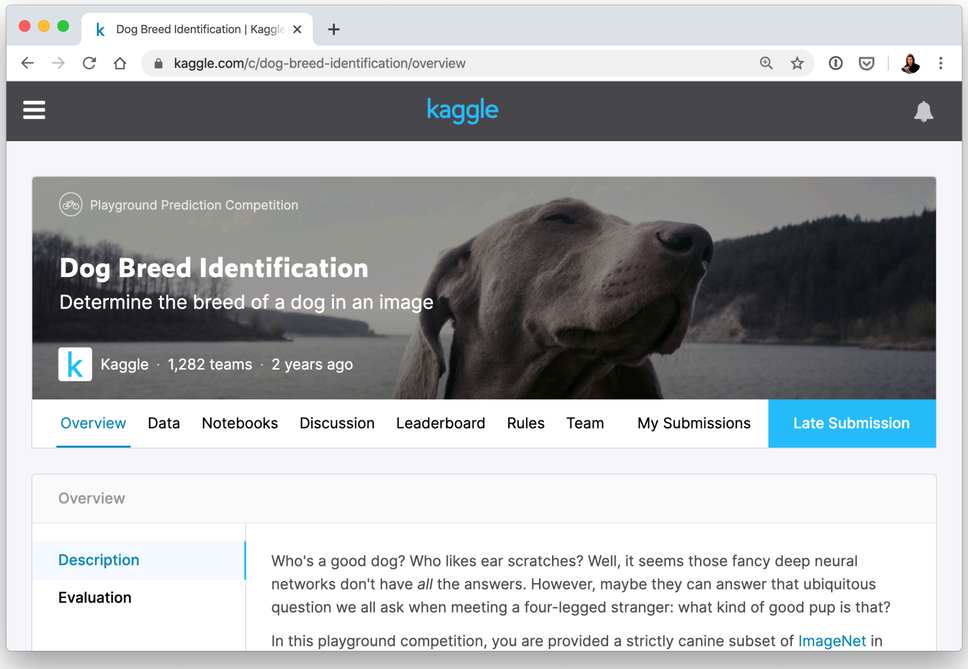
5. Improve through
experimentation

6. Save and reload
your trained model

What are we going to cover?

- An end-to-end multi-class classification workflow with TensorFlow
- Preprocessing image data (getting it into Tensors)
- Choosing a deep learning model
- Fitting a model to the data (learning patterns)
- Making predictions with a model (using patterns)
- Evaluating model predictions
- Saving and loading models
- Using a trained model to make predictions on custom data

The problem we're going to work on

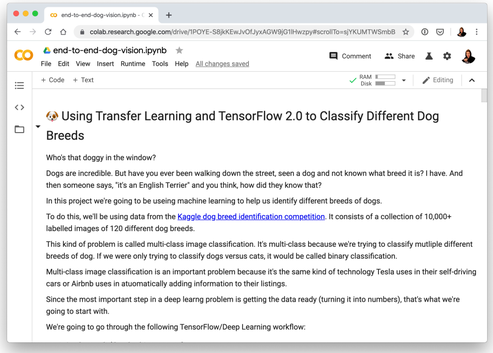


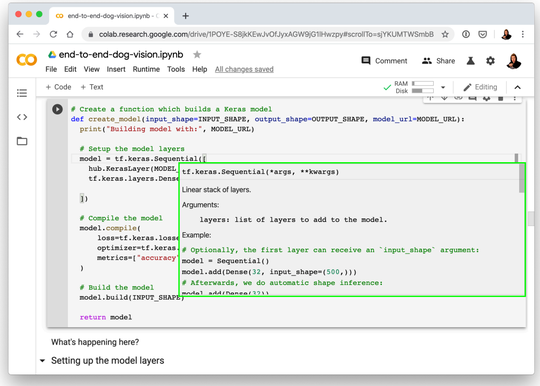
Classification

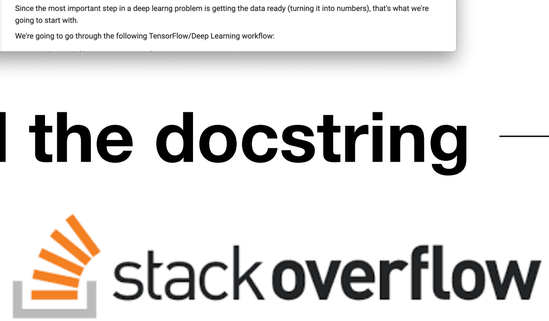
- "Is this example one thing or another?"
- Binary classification = two options
- Multi-class classification = more than two options

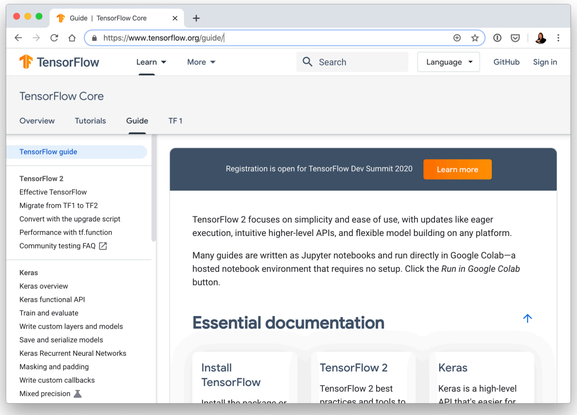
Where can you get help?

- Follow along with the code
- Try it for yourself
- Press SHIFT + CMD + SPACE to read the docstring
- Search for it
- Try again
- Ask (don't forget the Discord chat!)

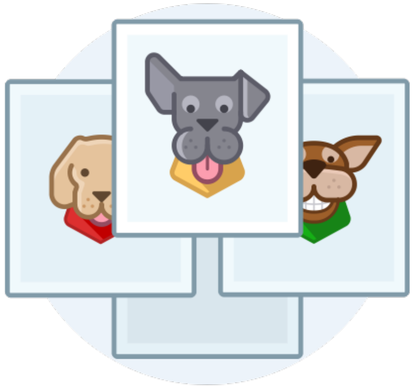








Let's find those doggos!



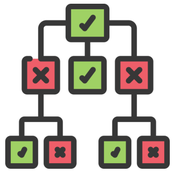
Choosing a model (throwback)

Problem 1 (structured)

Model 1

Structured Data

CatBoost



Random Forest

Problem 2 (unstructured)

Model 2

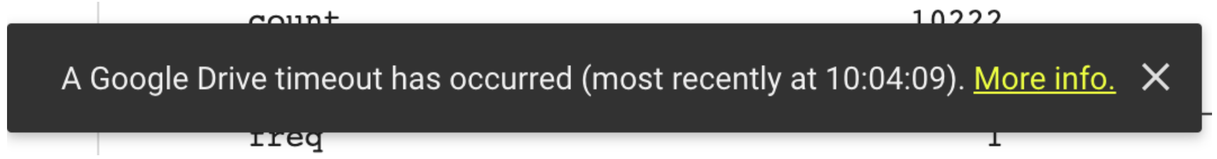
Unstructured Data

Deep Learning

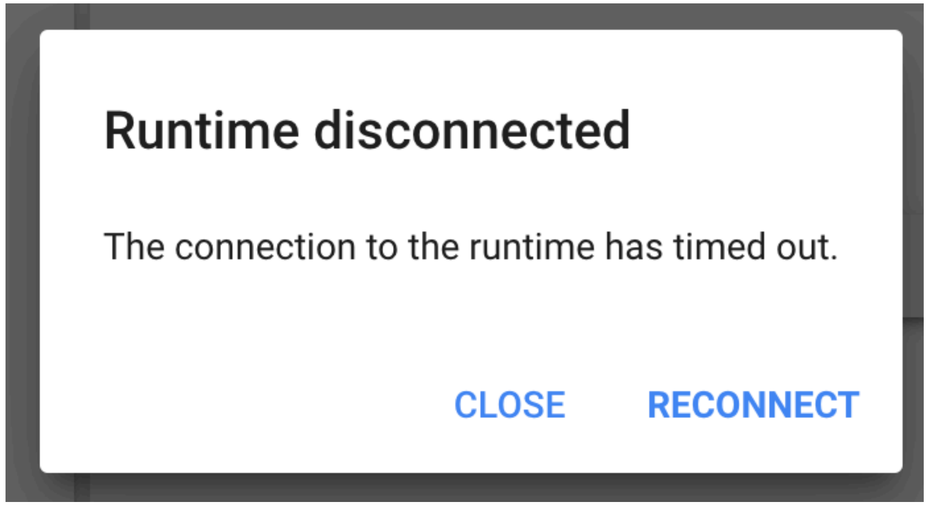
Transfer Learning

TensorFlow Hub

Things you might see in Colab



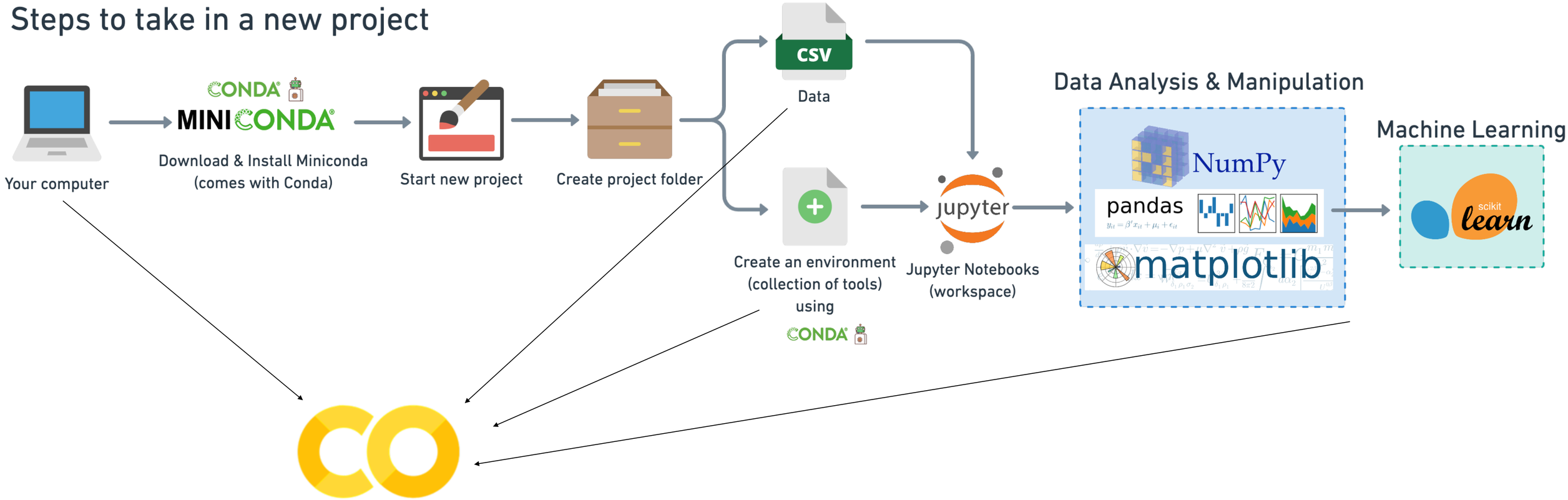
Usually fixes itself, may need to reconnect Google Drive to Colab.



A Colab connection will sometimes drop out (it's hard to tell when). If it does, you'll need to reconnect and potentially rerun all of your cells (disconnecting = variables lost).

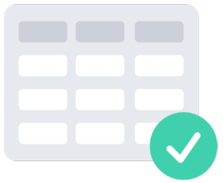
For more information and fixes, refer to the Google Colab FAQ: https://research.google.com/colaboratory/faq.html

Steps to take in a new project

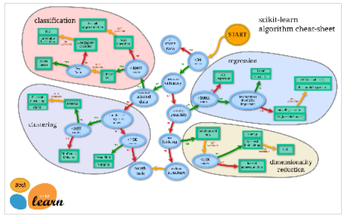


What are we going to cover?

A Scikit-Learn workflow



1. Get data ready



2. Pick a model
(to suit your problem)

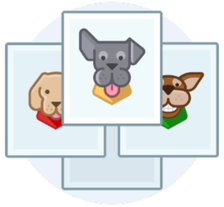
3. Fit the model to the data
and make a prediction

4. Evaluate the model

5. Improve through
experimentation

6. Save and reload
your trained model

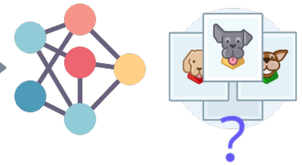
A TensorFlow workflow



1. Get data ready
(turn into Tensors)

TensorFlow
Hub

2. Pick a model
(to suit your problem)



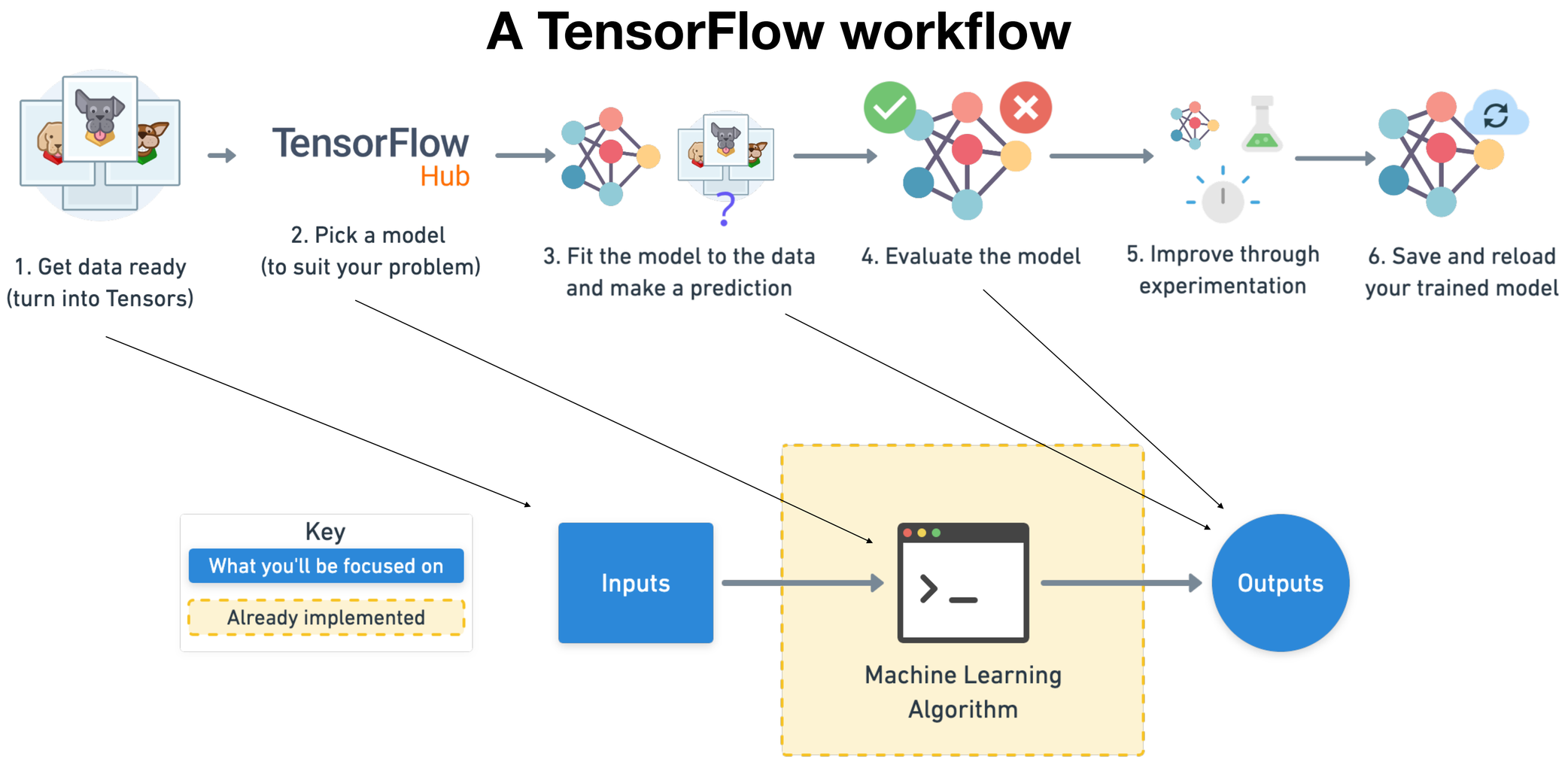
3. Fit the model to the data
and make a prediction

4. Evaluate the model

5. Improve through
experimentation

6. Save and reload
your trained model

What we're focused on

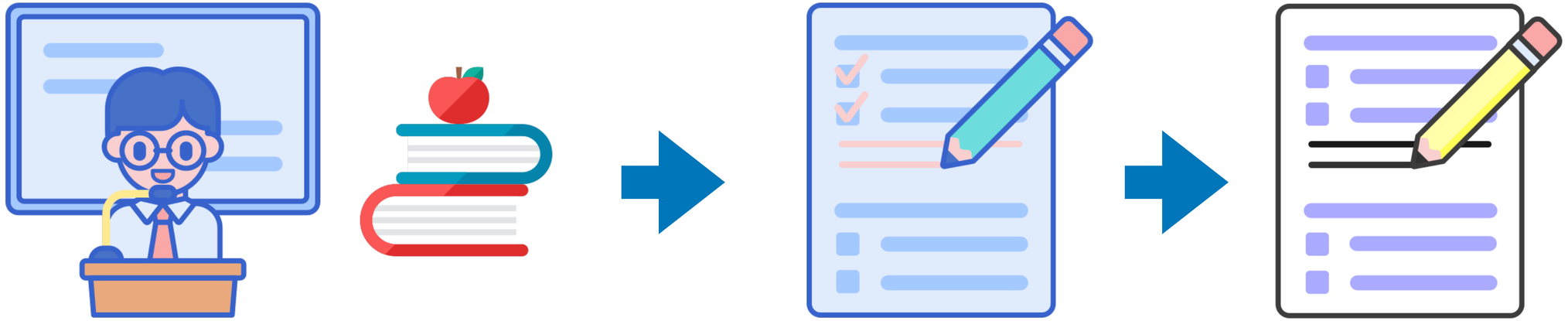


Which activation? Which loss?

Binary classification	Multi-class classification
Activation: Sigmoid	Activation: Softmax
Loss: Binary Crossentropy	Loss: Categorical Crossentropy

The most important concept in machine learning

(the 3 sets)



Course materials (training set)

Practice exam (validation set)

Final exam (test set)

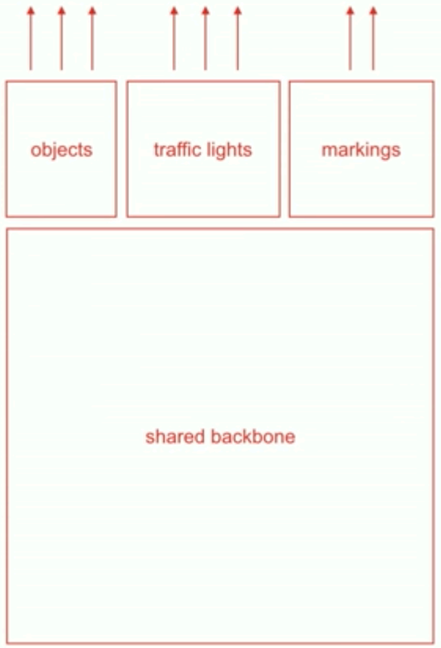
Generalization

The ability for a machine learning model to perform well on data it hasn't seen before.

Tesla using ResNet50 backbones

SINGLE VIEW TASKS

- (3, 960, 1280) input images
- “ResNet-50 like” dilated backbones
- FPN / DeepLabV3 / UNet -like heads
- ~15 tasks => “prototypes framework”



Source: “PyTorch at Tesla by Andrei Karpathy”
https://youtu.be/oBklltKXtDE?t=173