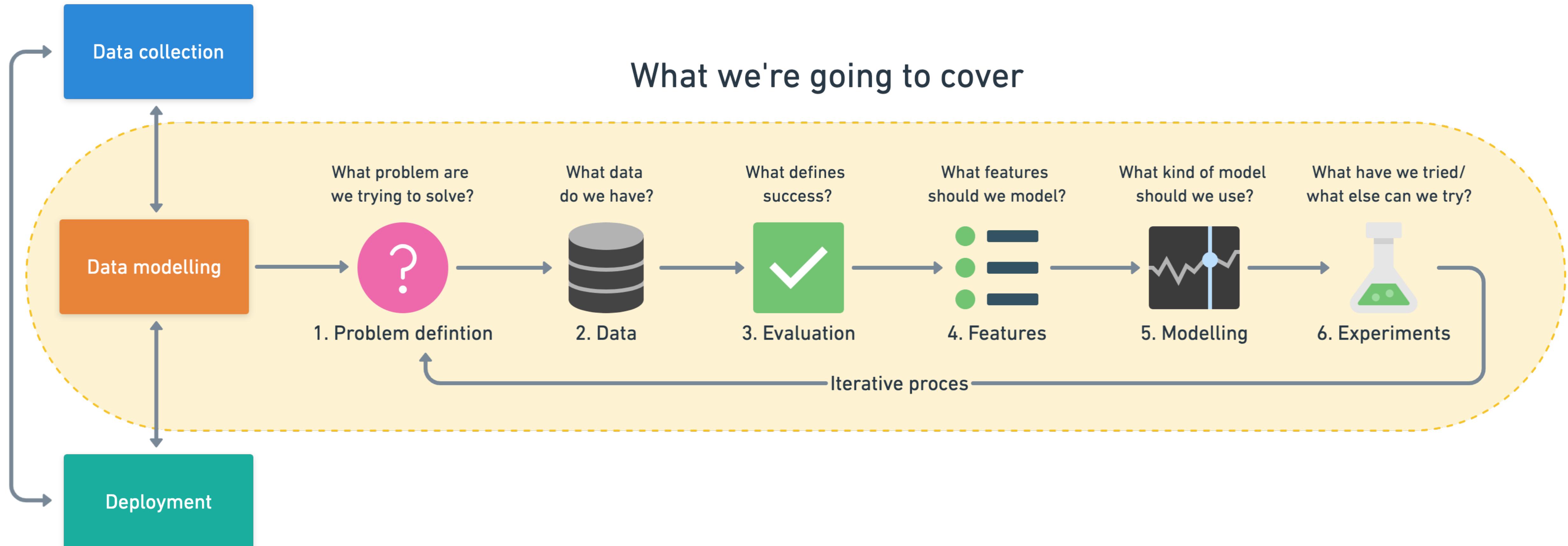
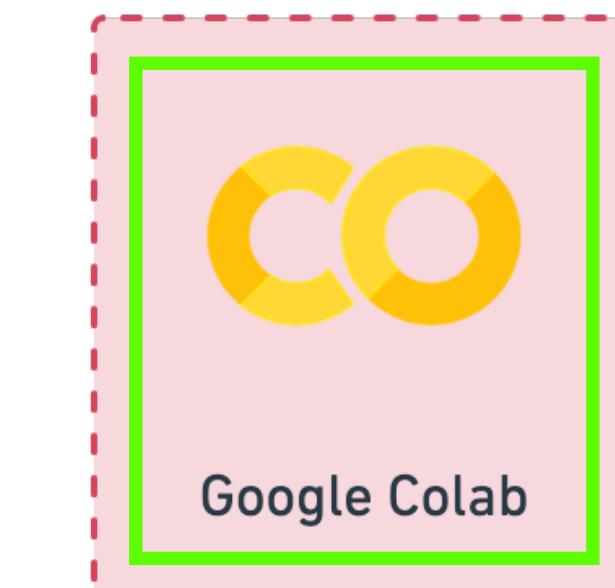
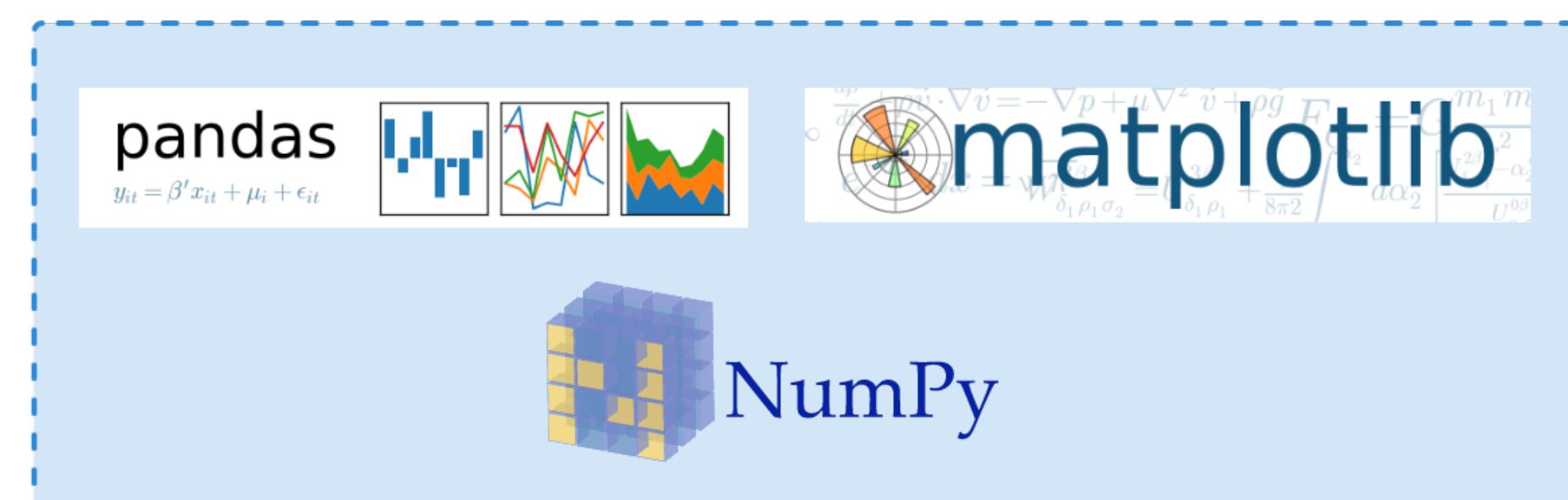
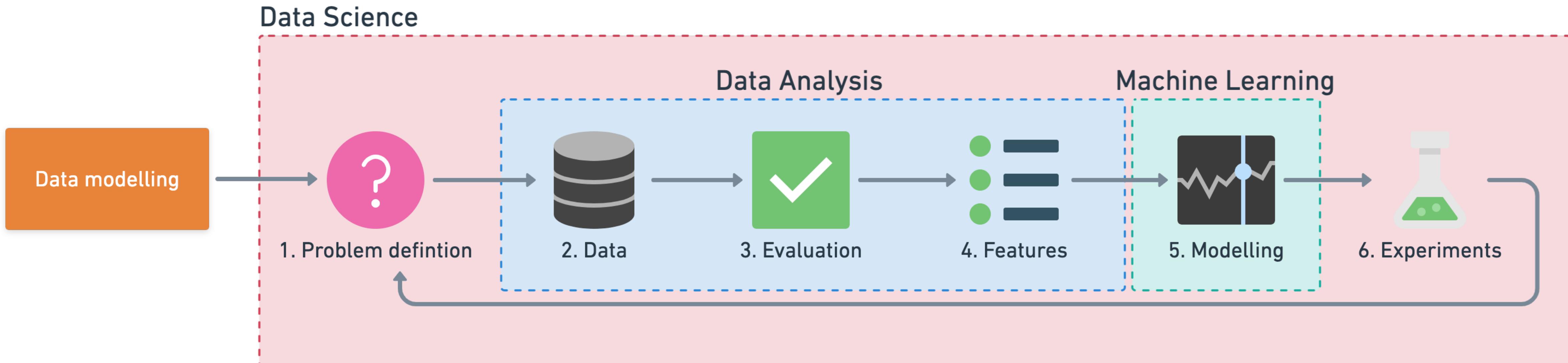


Steps in a full machine learning project

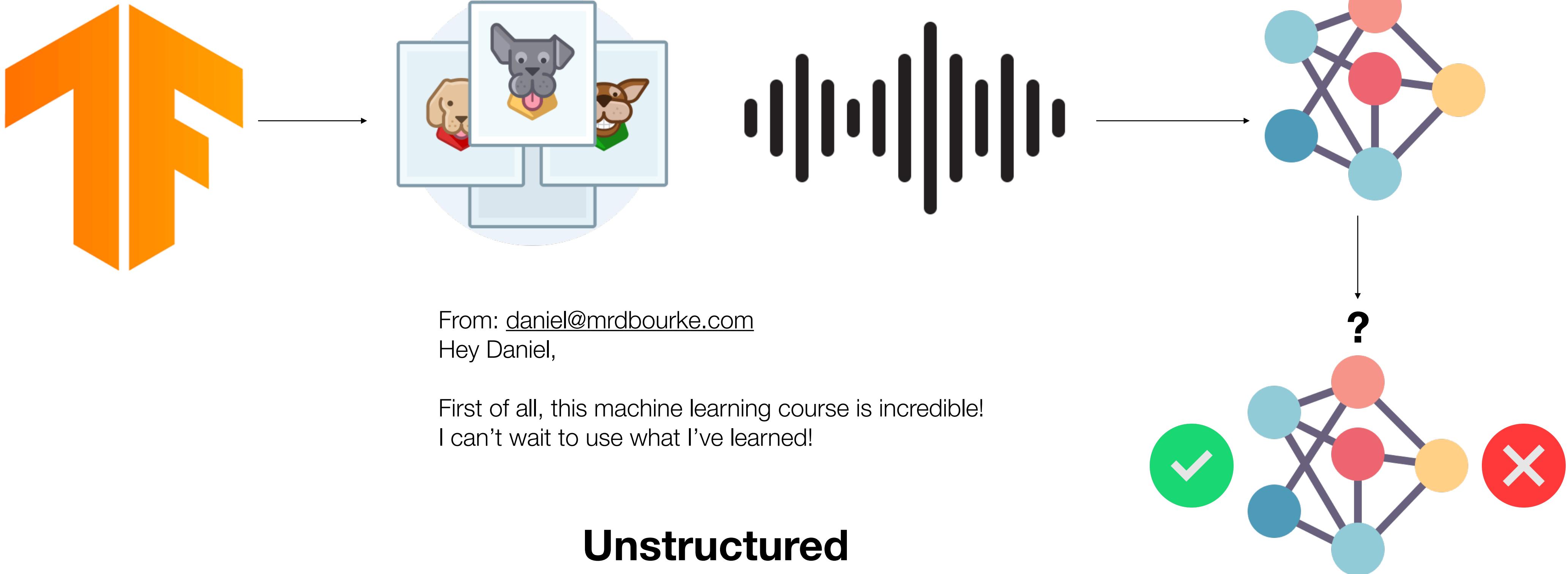


Tools you can use



A new contender!

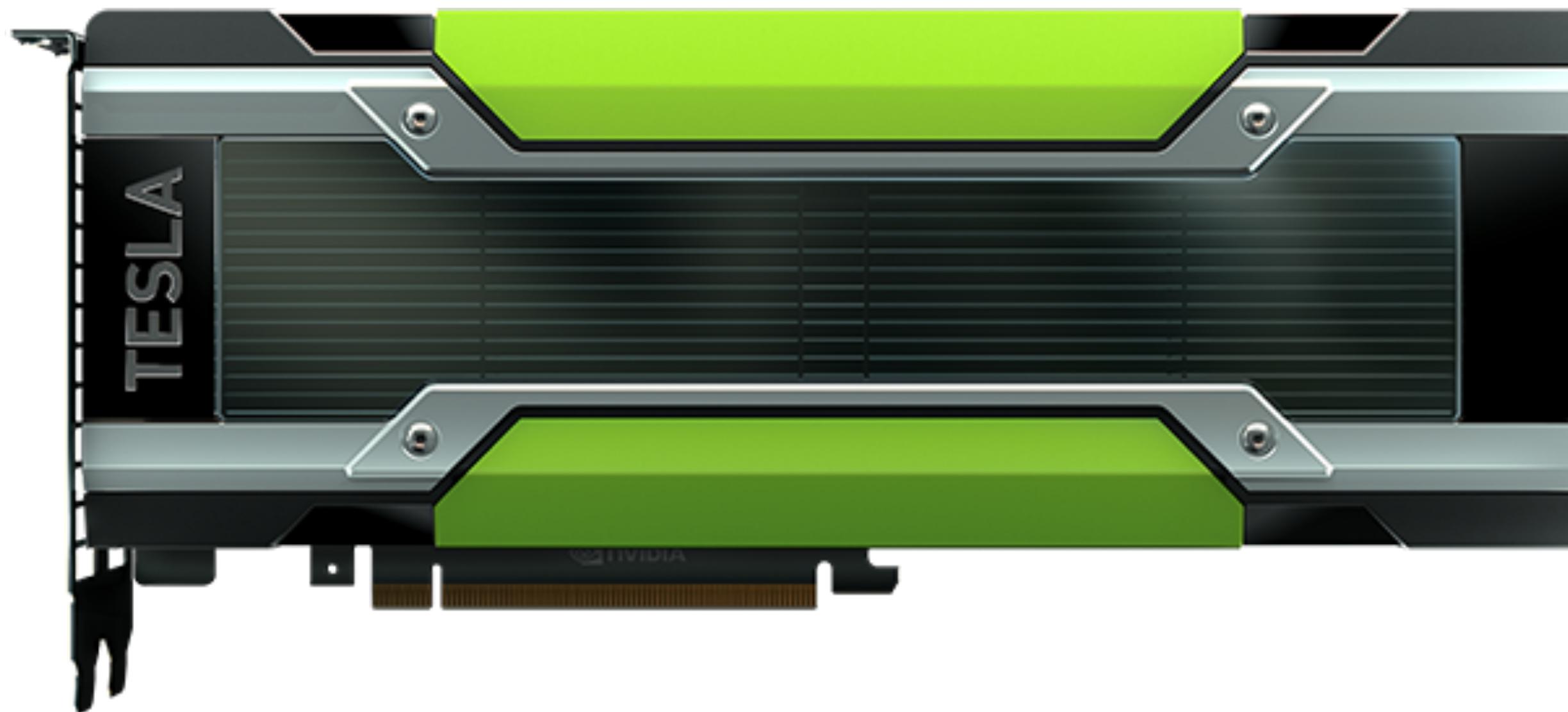
What is TensorFlow?



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)

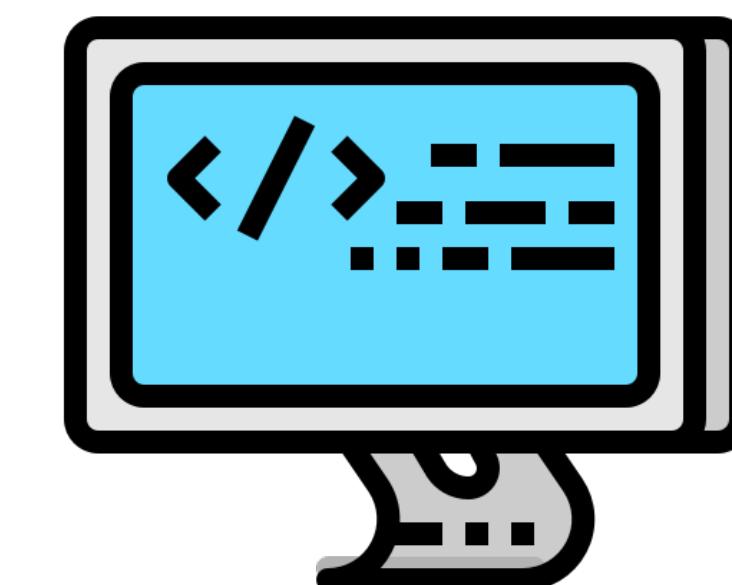
What is a GPU?



Choosing a model (throwback)

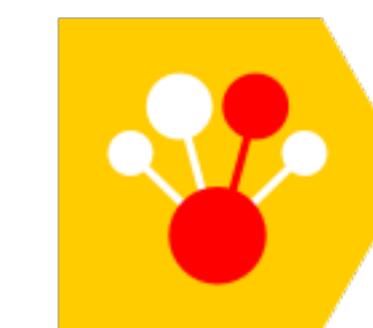


Problem 1 (structured)



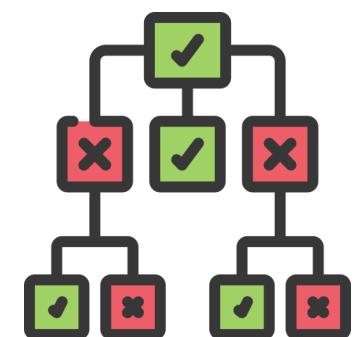
Model 1

Structured Data



CatBoost

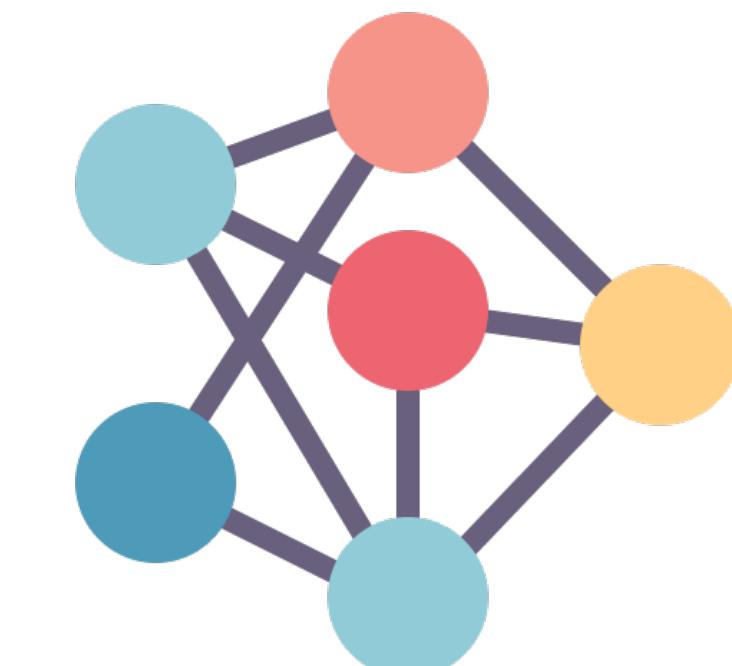
dmlc
XGBoost



Random Forest



Problem 2 (unstructured)

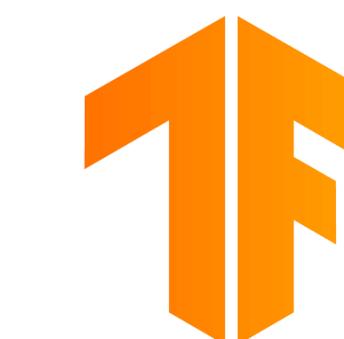


Model 2

Unstructured Data

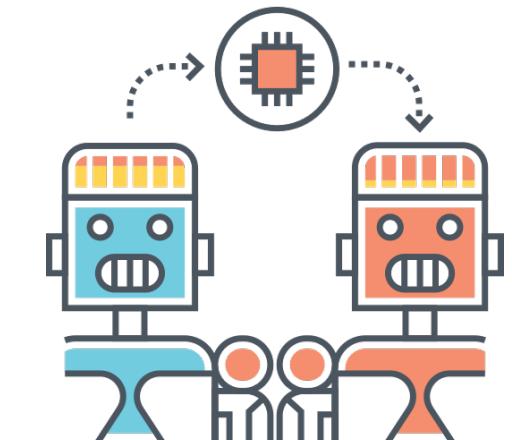


Deep Learning

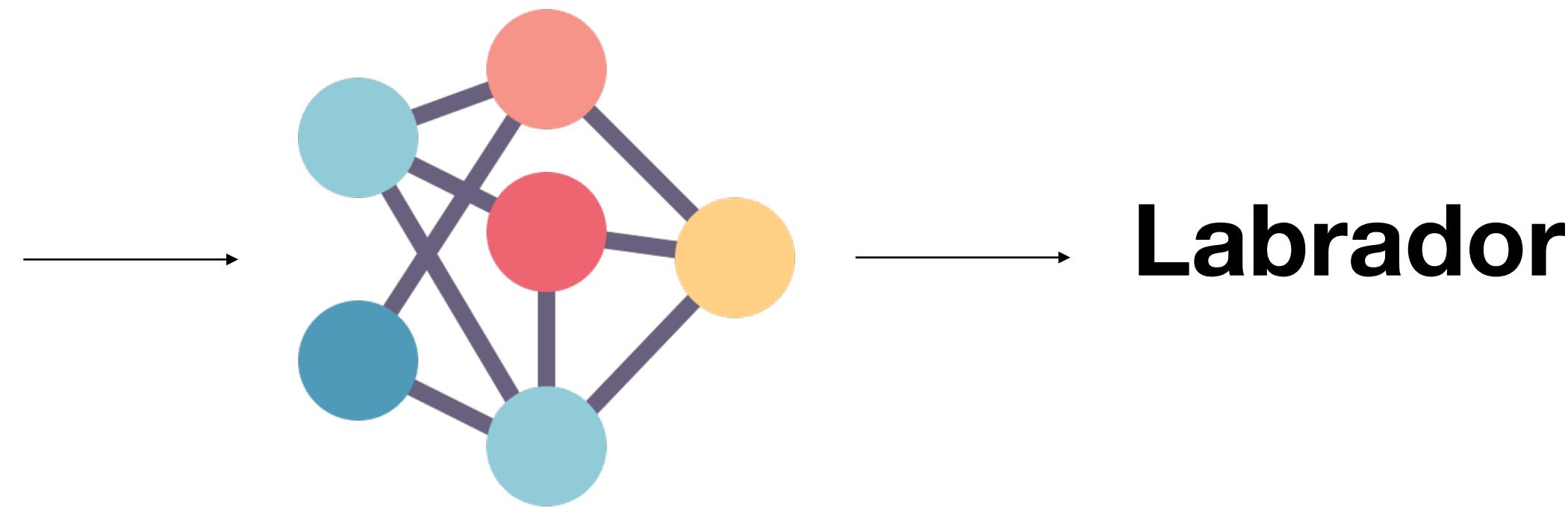


Transfer Learning

**TensorFlow
Hub**

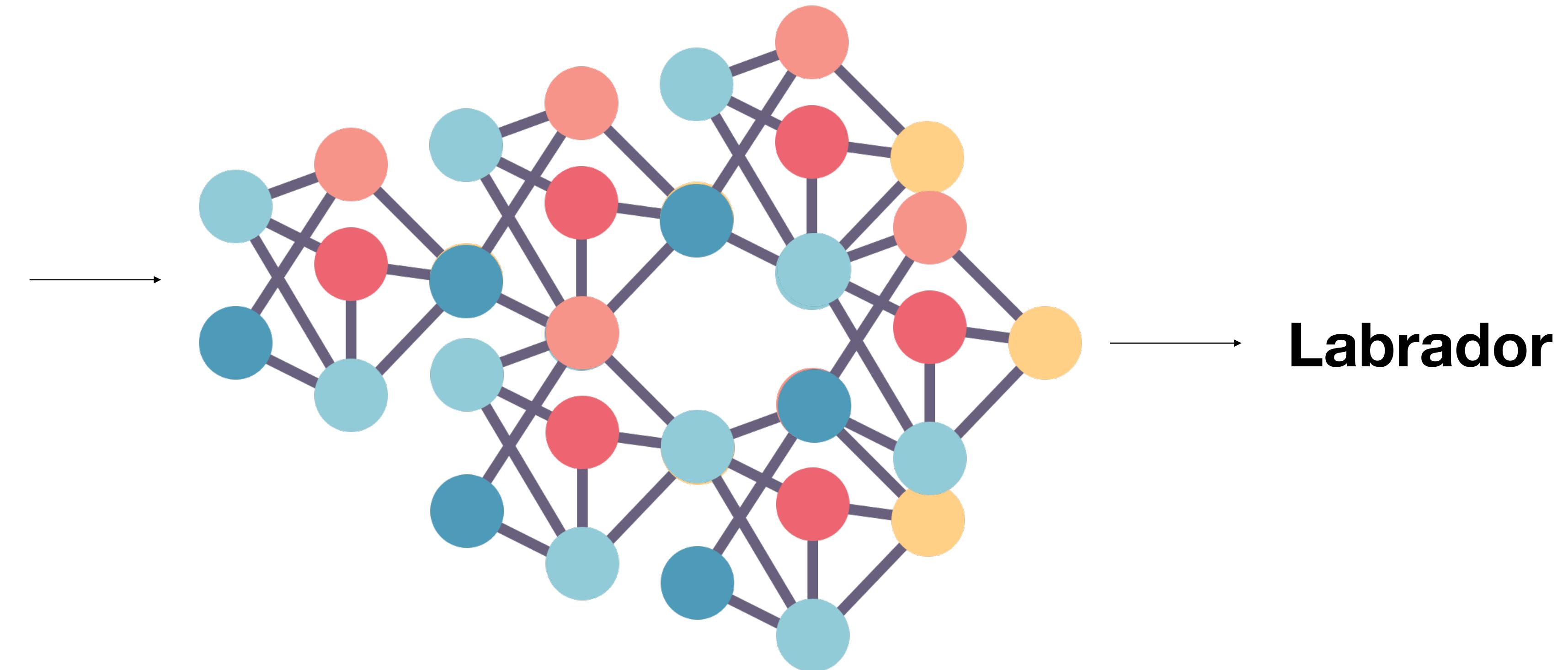
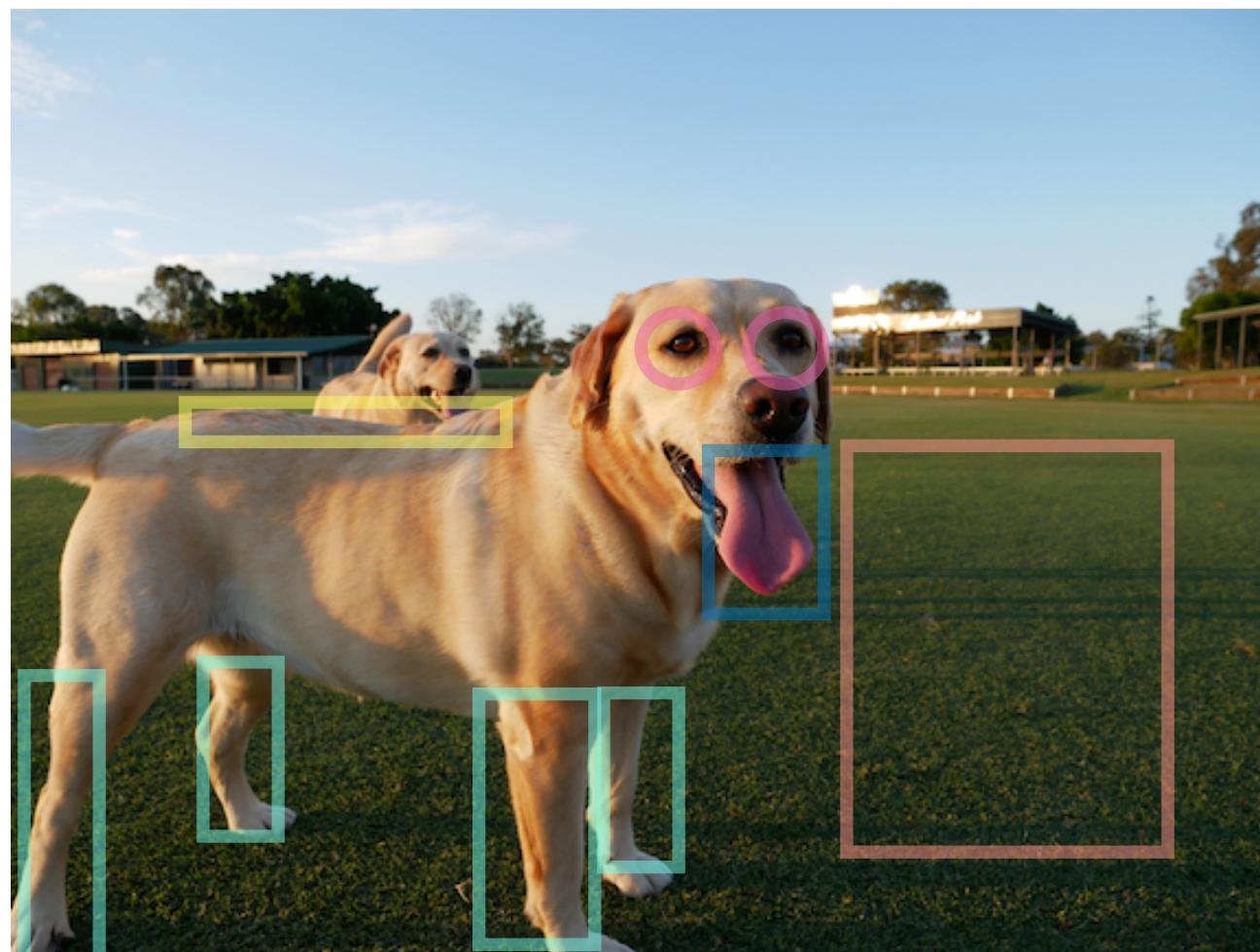


What is deep learning? What are neural networks?

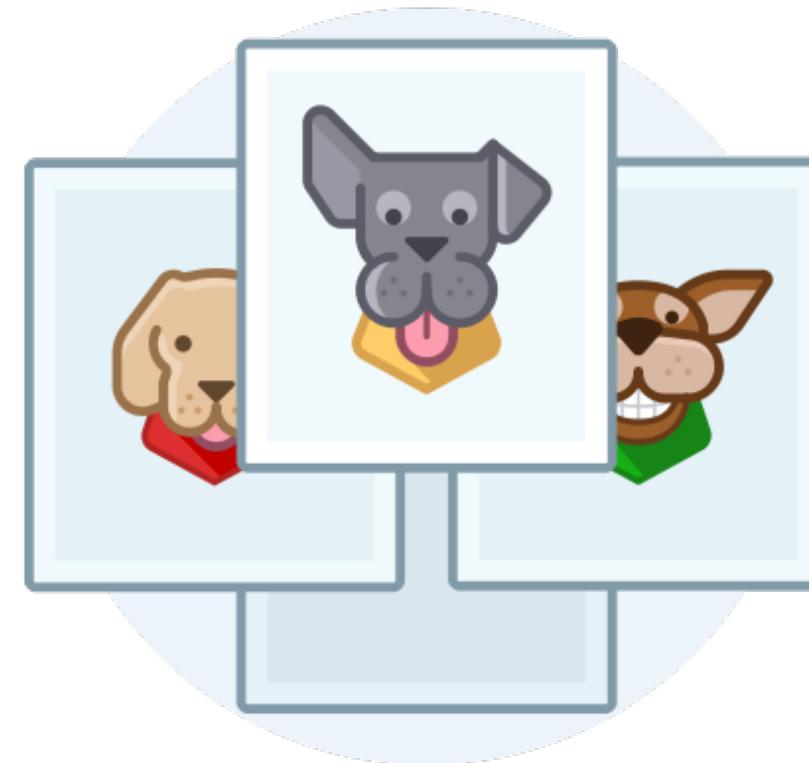


Labrador

What is deep learning? What are neural networks?



What kind of deep learning problems are there?



From: daniel@mrbourke.com

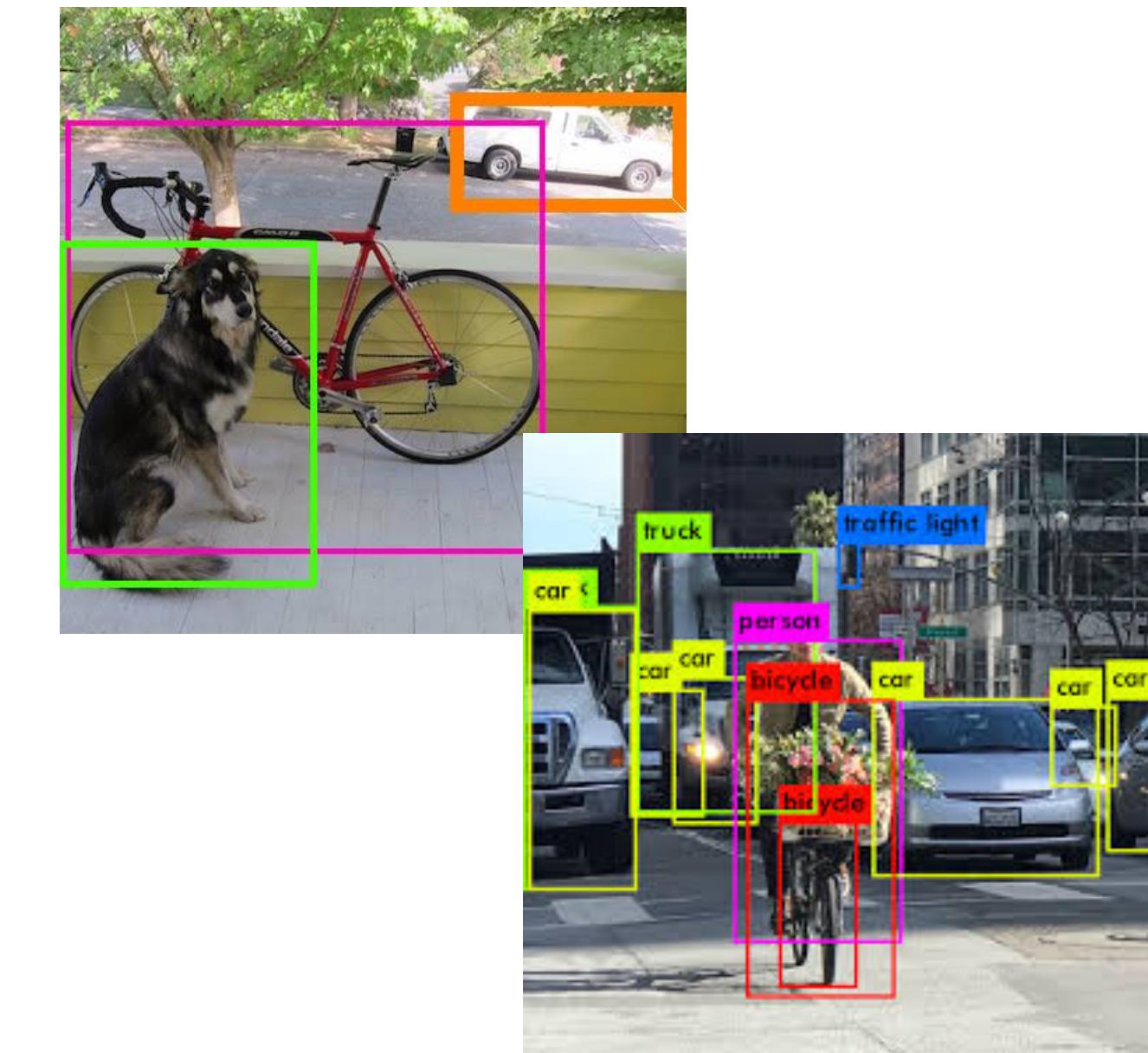
Hey Daniel,

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

Classification



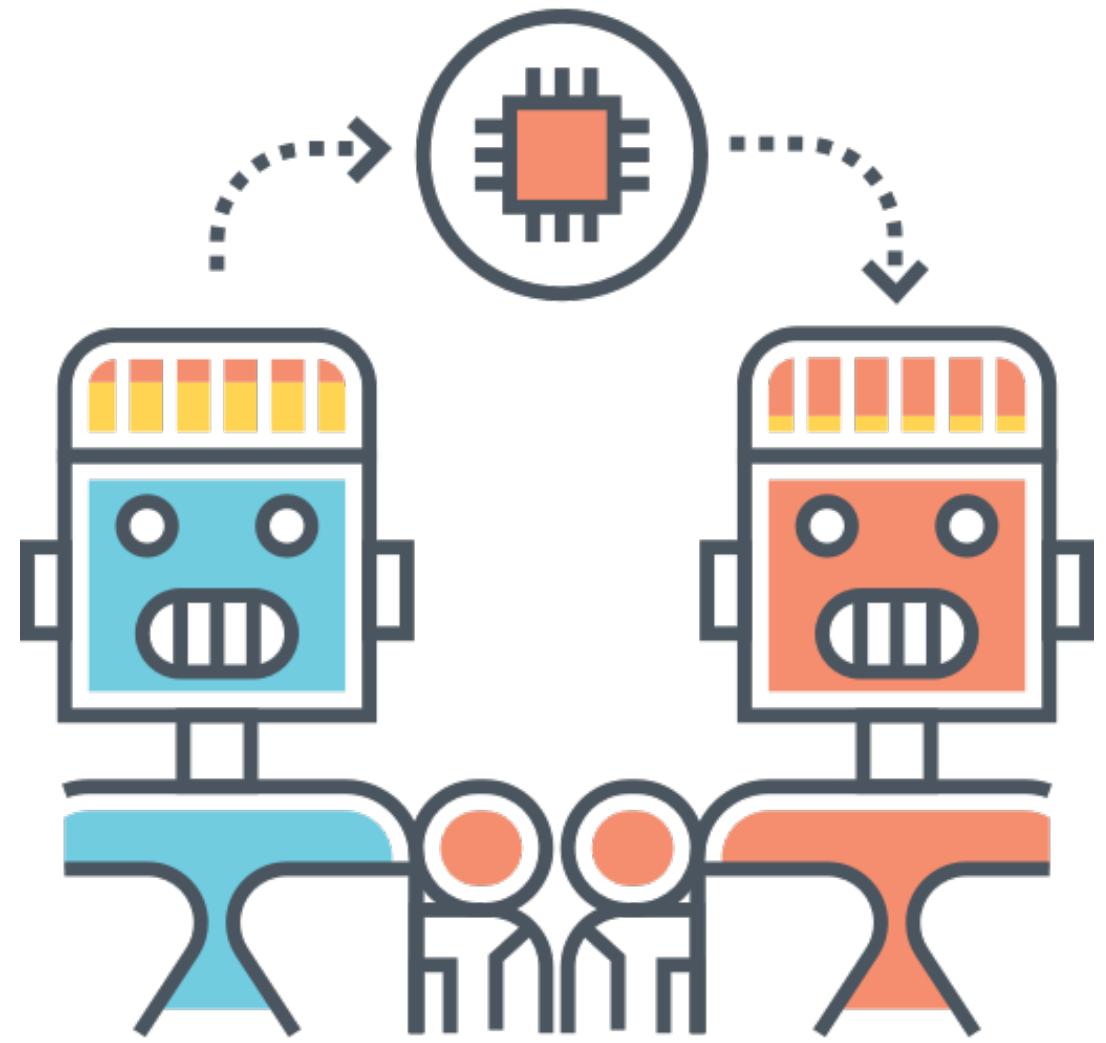
Hey Siri, where's the nearest cafe?



**Sequence to sequence
(seq2seq)**

Object detection

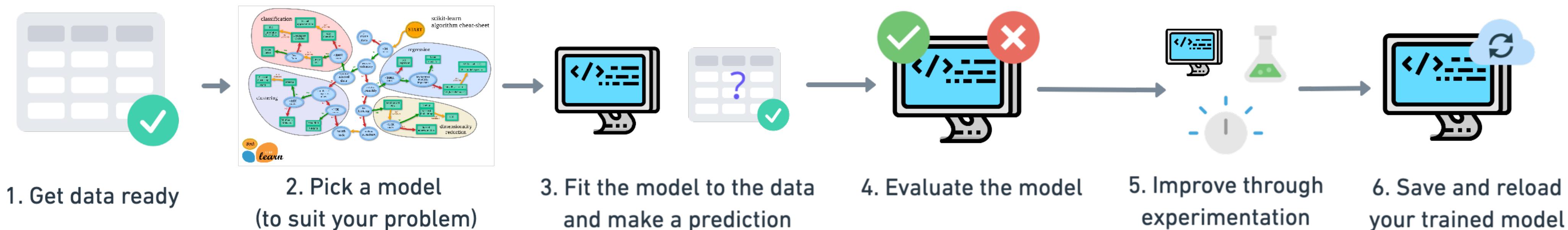
What is transfer learning? Why use transfer learning?



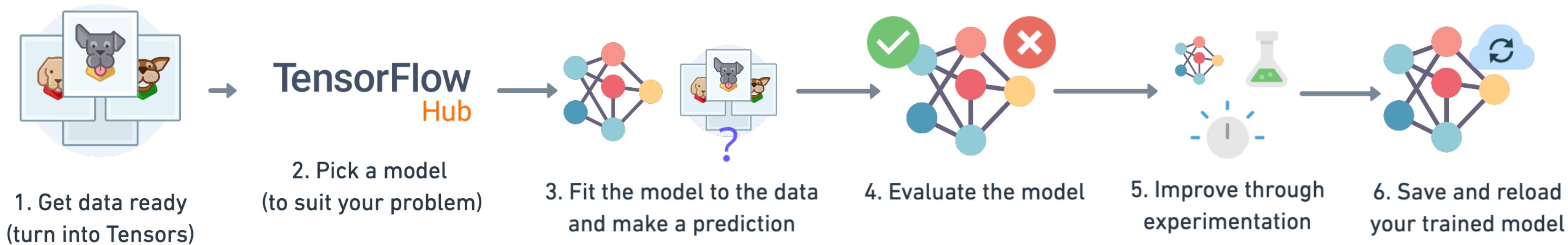
- Take what you know in one domain and apply it to another.
- Starting from scratch can be expensive and time consuming.
- Why not take advantage of what's already out there?

What are we going to cover?

A Scikit-Learn workflow



A TensorFlow workflow



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**

A screenshot of a web browser displaying the Kaggle Dog Breed Identification competition page. The page features a large image of a dog's head, the title "Dog Breed Identification", and the subtitle "Determine the breed of a dog in an image". The navigation bar includes links for Overview, Data, Notebooks, Discussion, Leaderboard, Rules, Team, My Submissions, and Late Submission (which is highlighted). The "Description" section contains text about the competition, mentioning ImageNet and the goal of determining dog breeds. The "Evaluation" section contains a brief description of the evaluation process.

Where can you get help?

- Follow along with the code



```
Using Transfer Learning and TensorFlow 2.0 to Classify Different Dog Breeds

Who's that doggy in the window?
Dogs are incredible. But have you ever been walking down the street, seen a dog and not known what breed it is? I have. And then someone says, "it's an English Terrier" and you think, how did they know that?

In this project we're going to be using machine learning to help us identify different breeds of dogs.

To do this, we'll be using data from the Kaggle dog breed identification competition. It consists of a collection of 10,000+ labelled images of 120 different dog breeds.

This kind of problem is called multi-class image classification. It's multi-class because we're trying to classify multiple different breeds of dog. If we were only trying to classify dogs versus cats, it would be called binary classification.

Multi-class image classification is an important problem because it's the same kind of technology Tesla uses in their self-driving cars or Airbus uses in automatically adding information to their listings.

Since the most important step in a deep learning problem is getting the data ready (turning it into numbers), that's what we're going to start with.

We're going to go through the following TensorFlow/Deep Learning workflow:
```

- Try it for yourself

- Press SHIFT + CMD + SPACE to read the docstring

- Search for it
- Try again

- Ask (don't forget the Discord chat!)



```
# Create a function which builds a Keras model
def create_model(input_shape=INPUT_SHAPE, output_shape=OUTPUT_SHAPE, model_url=MODEL_URL):
    print("Building model with:", MODEL_URL)

    # Setup the model layers
    model = tf.keras.Sequential([
        hub.KerasLayer(MODEL_URL),
        tf.keras.layers.Dense(10)
    ])

    # Compile the model
    model.compile(
        loss=tf.keras.losses.BinaryCrossentropy(),
        optimizer=tf.keras.optimizers.Adam(),
        metrics=['accuracy']
    )

    # Build the model
    model.build(INPUT_SHAPE)

    return model
```

What's happening here?
Setting up the model layers

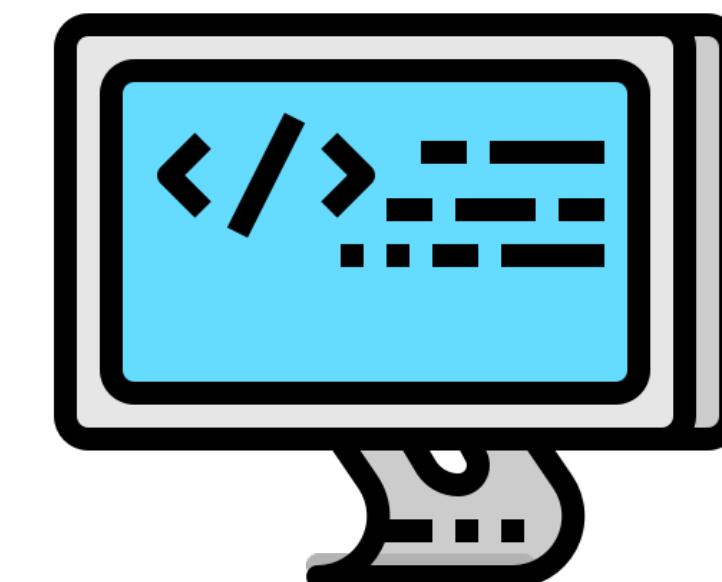
Let's find those doggos!



Choosing a model (throwback)



Problem 1 (structured)



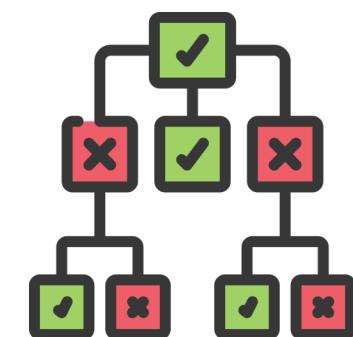
Model 1

Structured Data



CatBoost

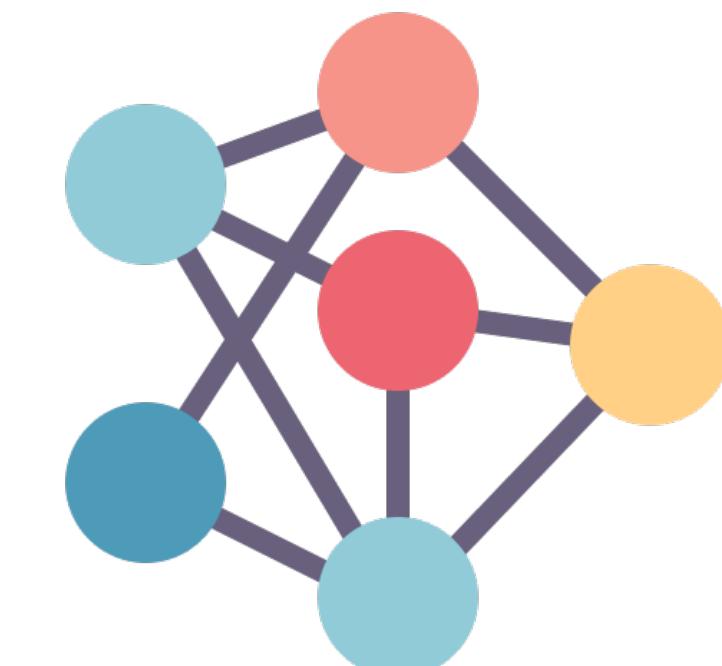
dmlc
XGBoost



Random Forest



Problem 2 (unstructured)



Model 2

Unstructured Data

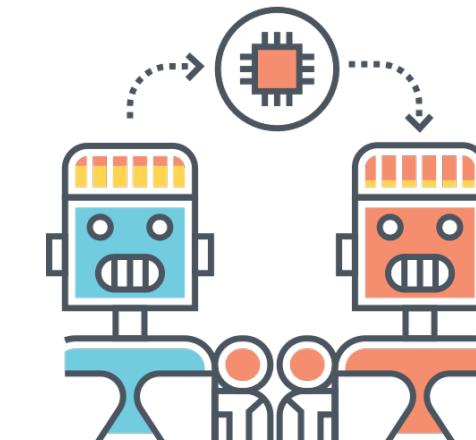


Deep Learning

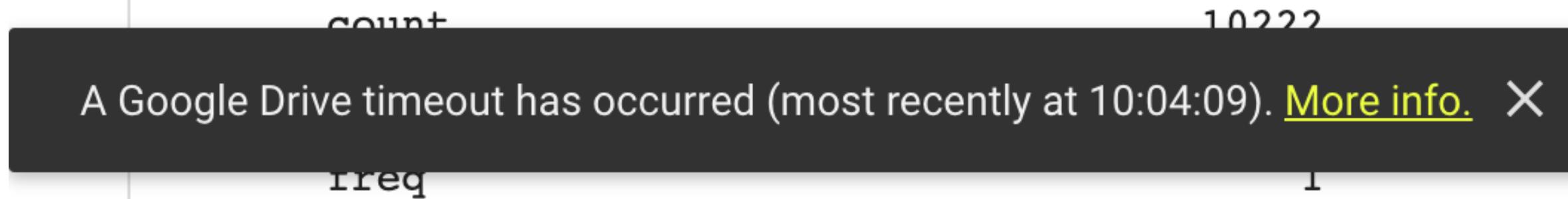


Transfer Learning

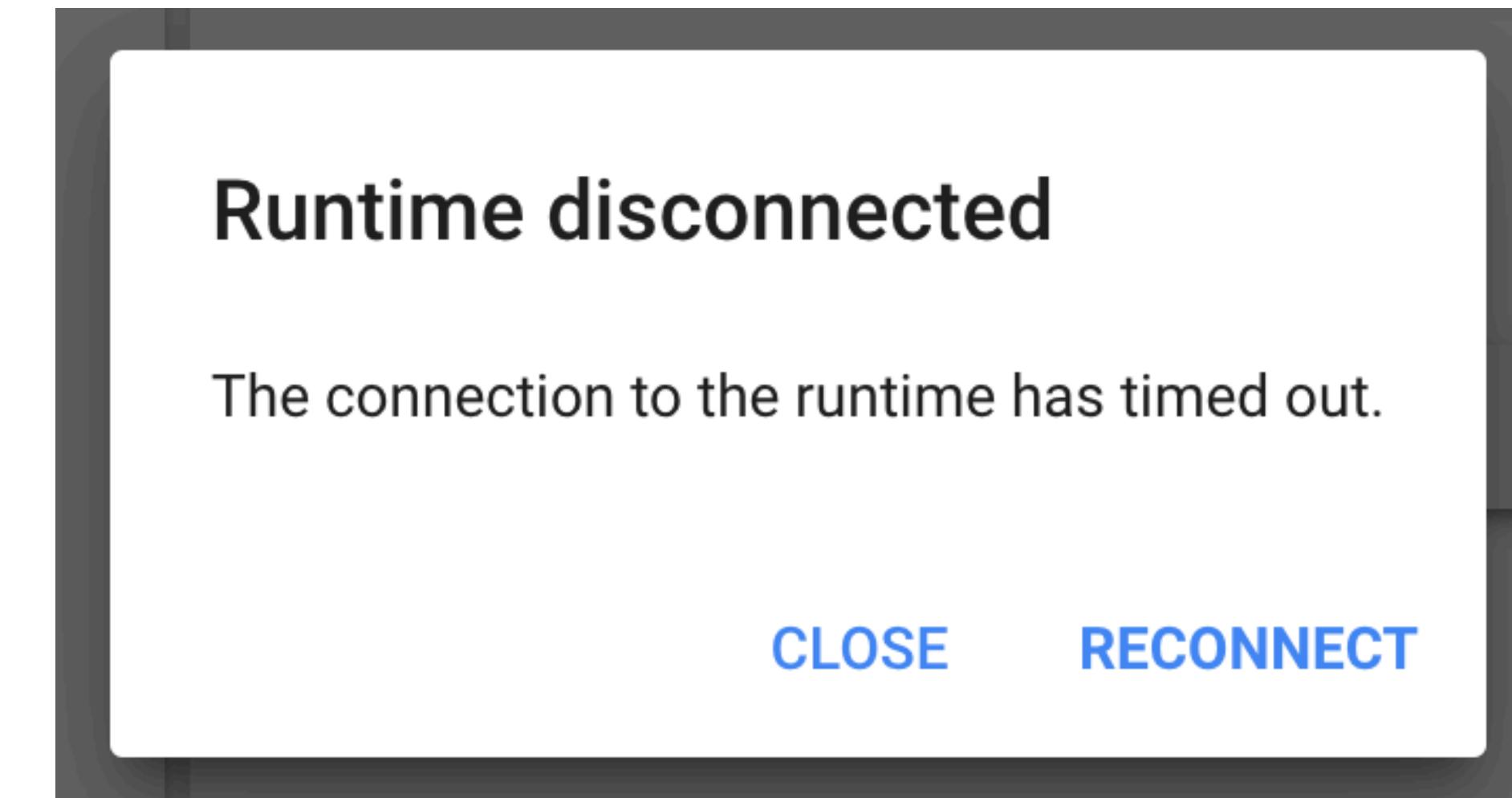
TensorFlow
Hub



Things you might see in Colab



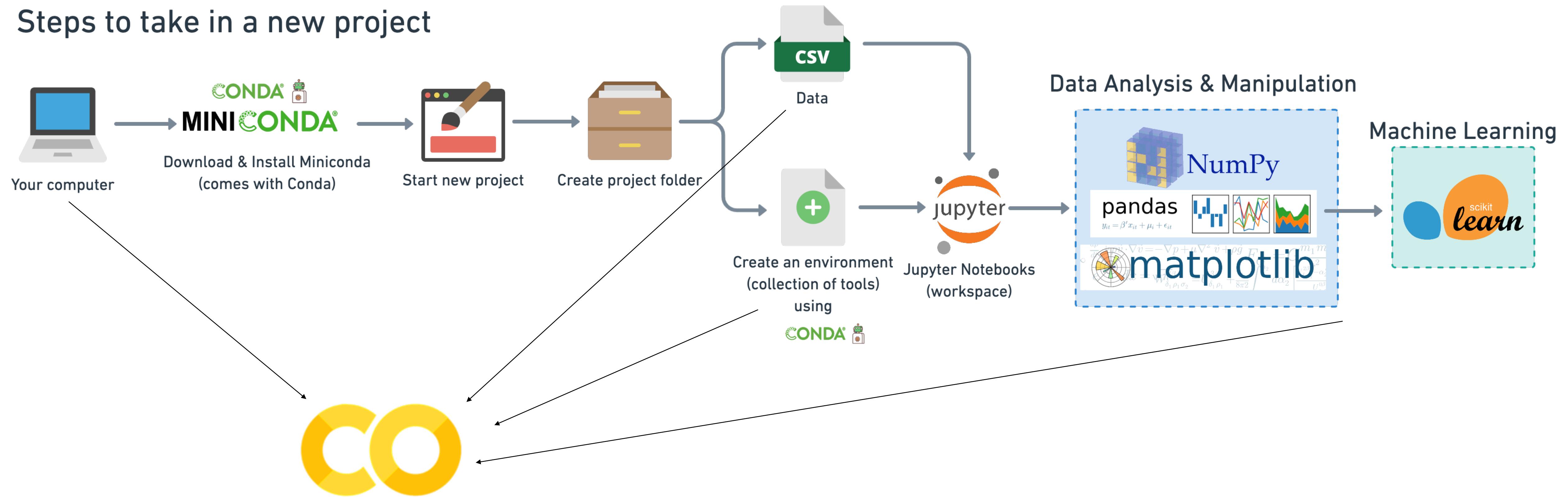
Usually fixes itself, may need to reconnect Google Drive to Colab.



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).

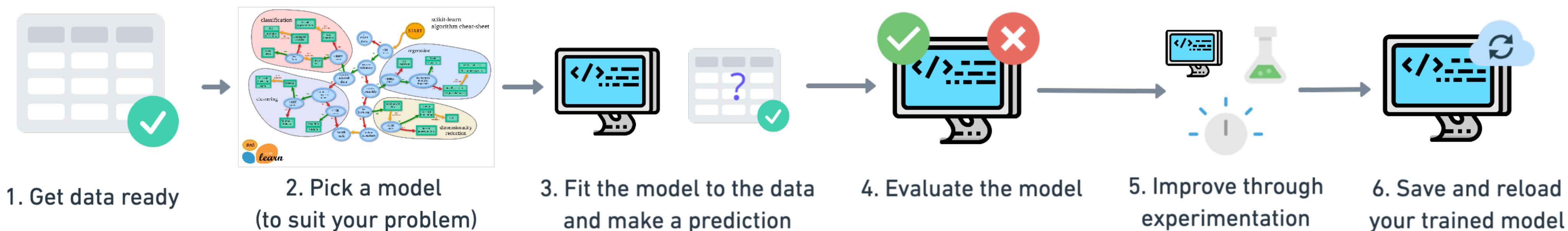
For more information and fixes, refer to the Google Colab FAQ: <https://research.google.com/colaboratory/faq.html>

Steps to take in a new project

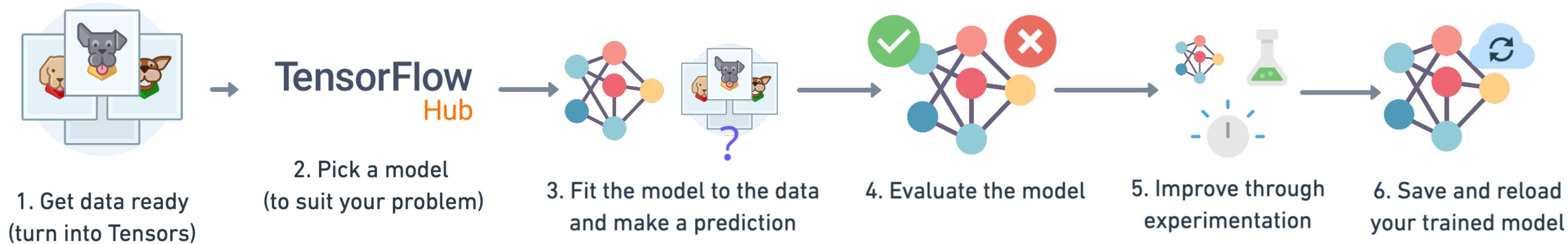


What are we going to cover?

A Scikit-Learn workflow

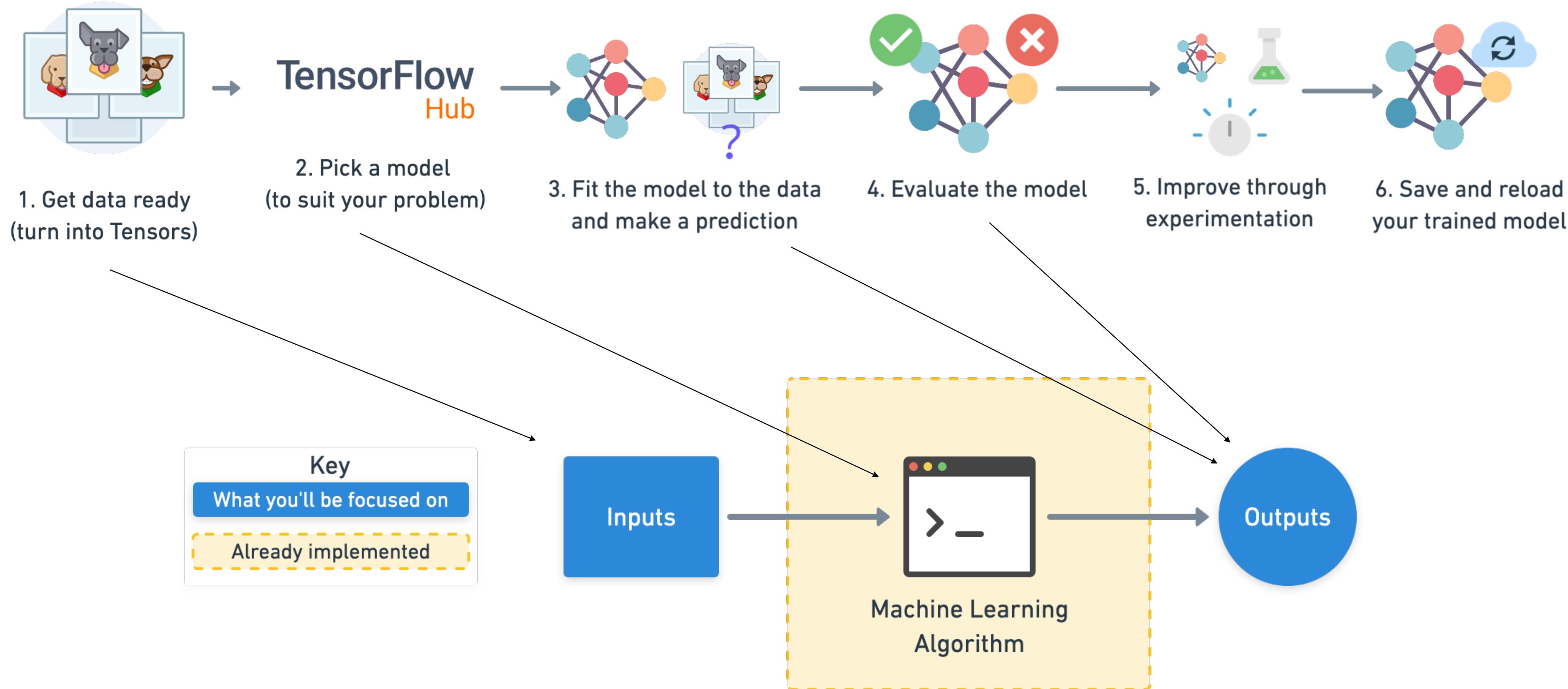


A TensorFlow workflow



What we're focused on

A TensorFlow workflow



Which activation? Which loss?

Binary classification

Activation: Sigmoid

Multi-class classification

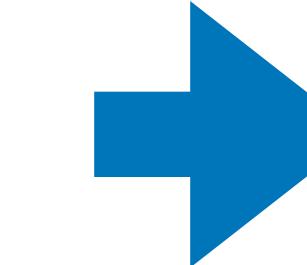
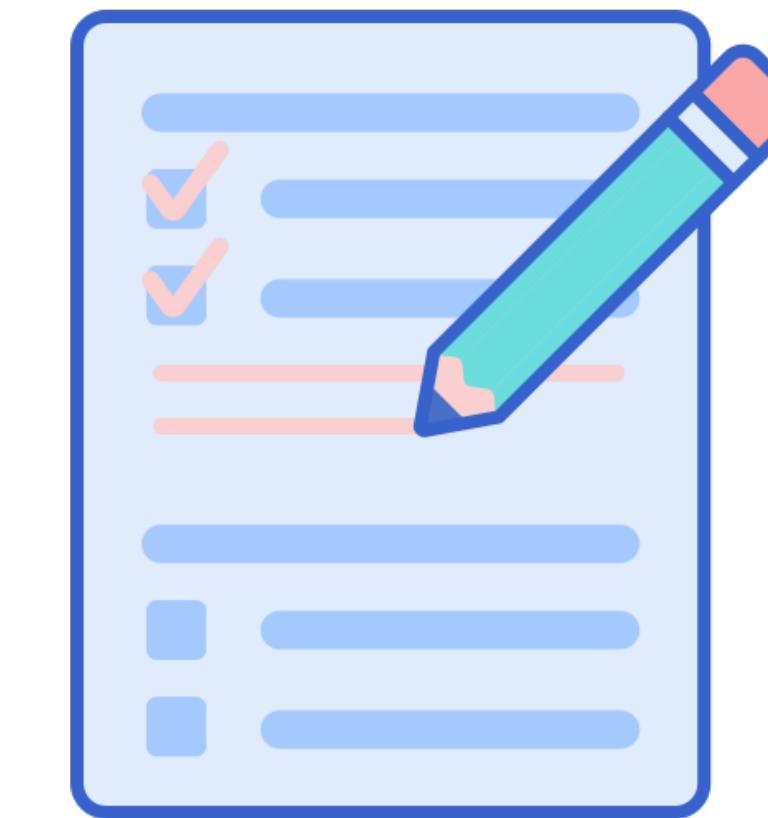
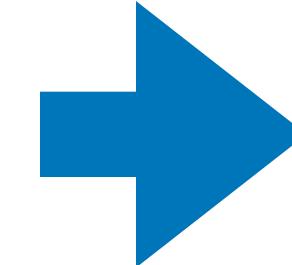
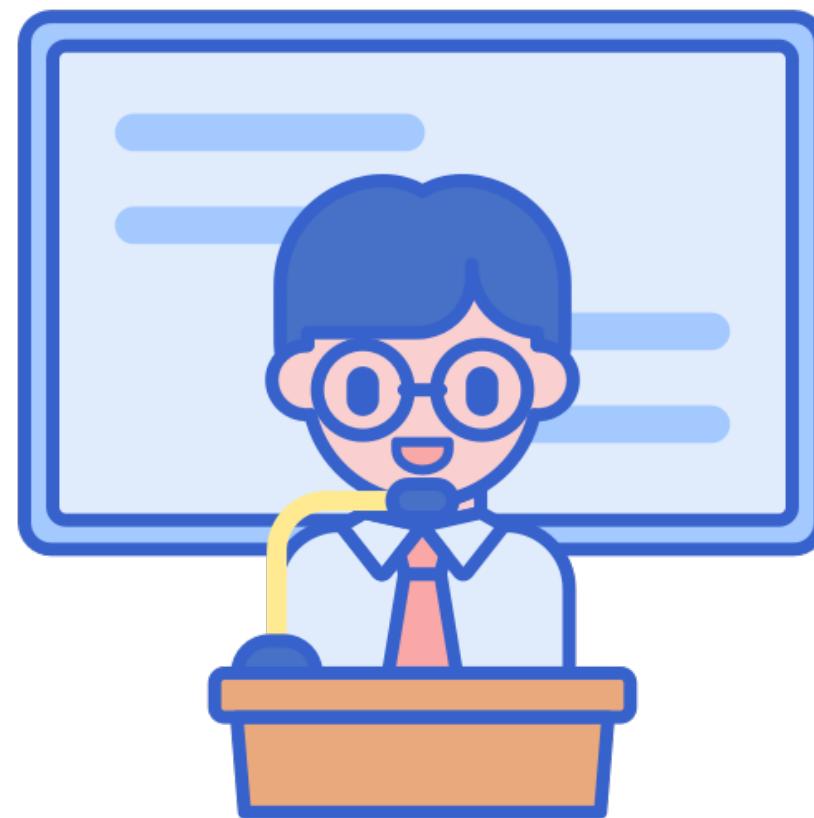
Activation: Softmax

Loss: Binary Crossentropy

Loss: Categorical Crossentropy

The most important concept in machine learning

(the 3 sets)



**Course materials
(training set)**

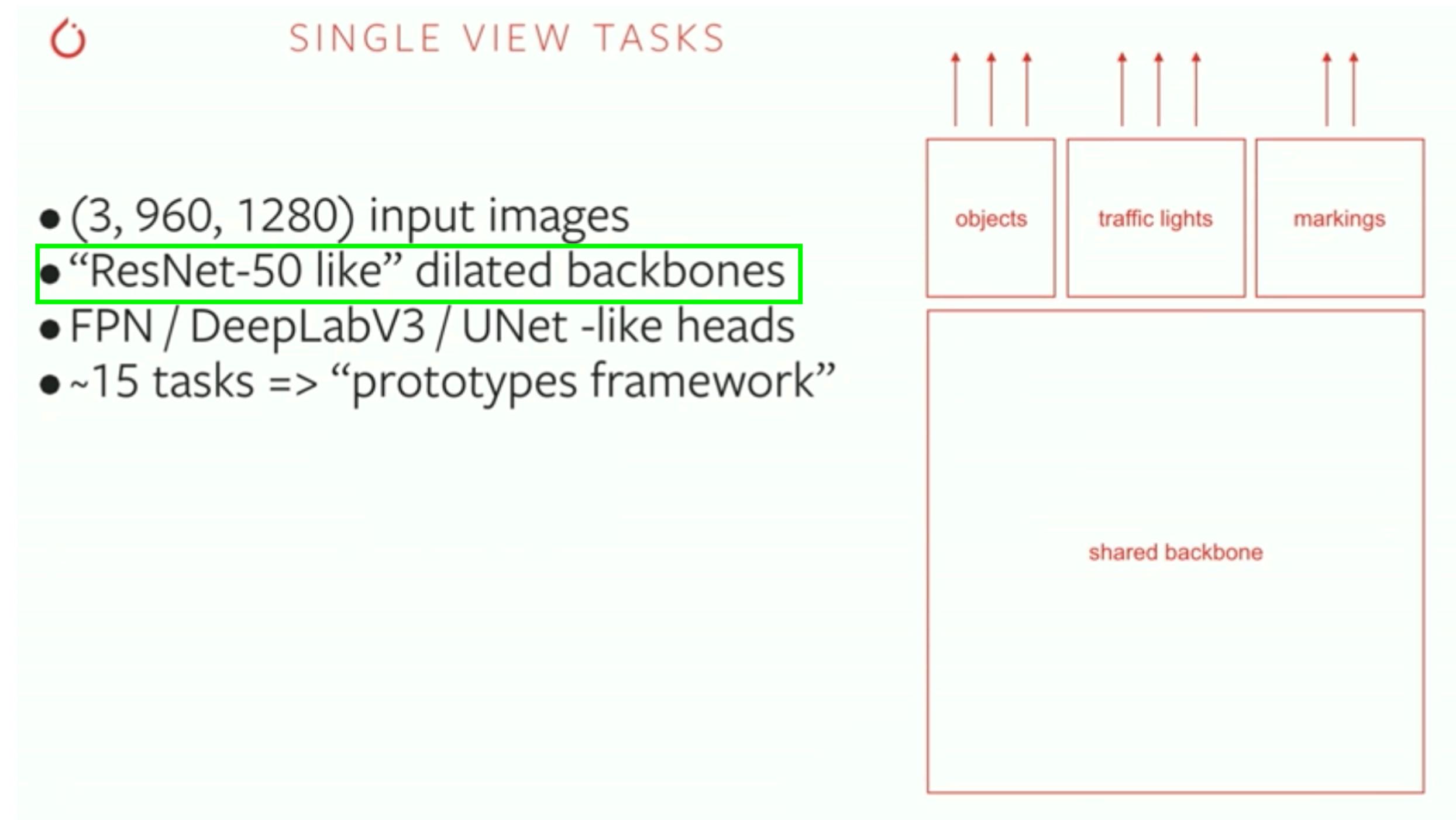
**Practice exam
(validation set)**

**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



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