Joker Conference, October 19-20, 2018 - St. Petersburg, Russia

TensorFlow for Java Developers

Derek Ferguson
Head of Engineering, Chase CB
derek.ferguson@mail.ru
derek.ferguson@chase.com

The Request

- Your business wants you to investigate machine learning
- They're looking for you to propose a suitable problem…
- And then they're looking for you to solve it

Our Starting Point

- We assume no prior background in machine learning
- We assume no prior background in advanced math / data

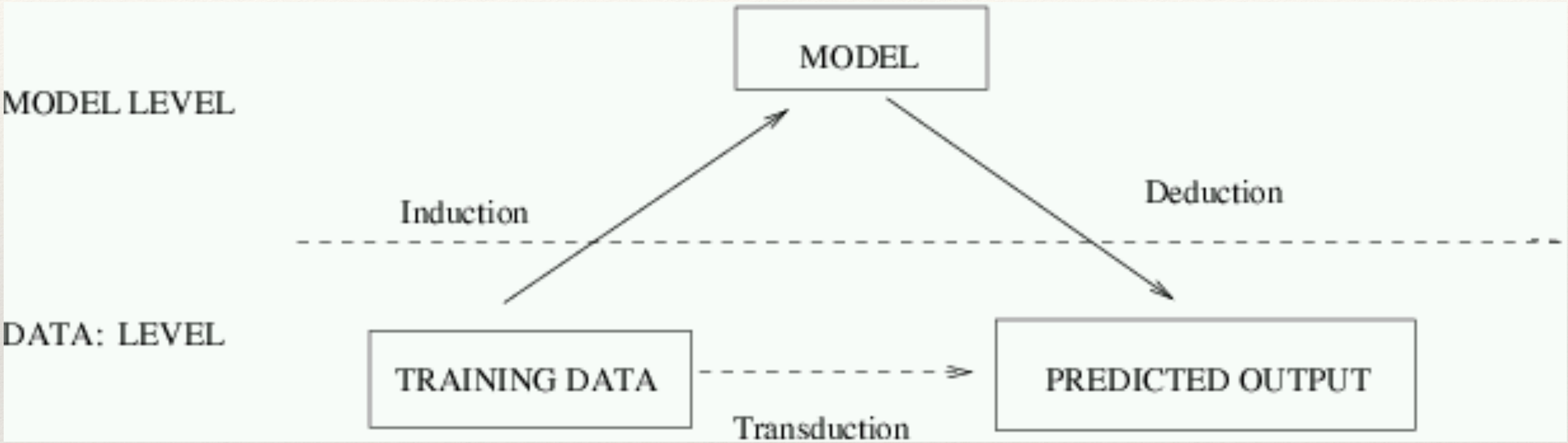
The Solution (aka “the Plan”)

- A very pragmatic, quick intro to machine learning
- An equally pragmatic, quick intro to TensorFlow
- 4 Java coding demonstrations

What is Machine Learning?

- Teach a computer to solve 2 + 2 - 1 = __ <— arithmetic
- Teach a computer to solve 2 + __ - 1 = 4 <— algebra
- Teach a computer to solve 2 __ 3 __ 1 = 4 <— ML

Supervised Learning



Induction

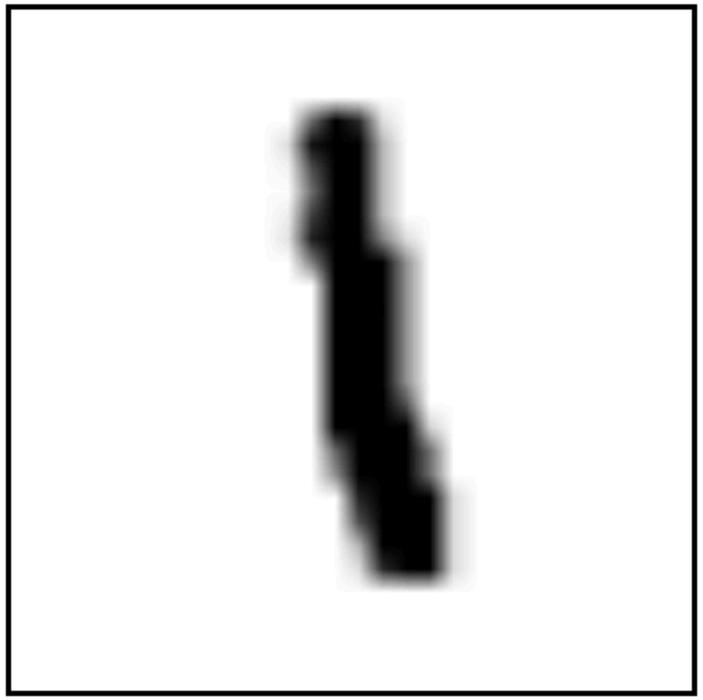
- $0 \_\_ 0 = 0$
- $1.5 \_\_ 3 = 4.5$
- $6 \_\_ 7 = 42$
- $1 \_\_ 2 \_\_ 1 \_\_ 3 \_\_ 2 \_\_ 1 \_\_ 7 \_\_ 4 \_\_ 3 \_\_ 7 \_\_ 8 \_\_ 4 \_\_ 4 = -2$

Induction

No Such Thing as Non-Numeric Data

- Do you smoke? (0 = No, 1 = Yes)
- Images and time data are just number matrices

No Such Thing as Non-Numeric Data



$$\approx \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & .6 & .8 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & .7 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & .7 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & .5 & 1 & .4 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & .4 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & .4 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & .7 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & .9 & 1 & .1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & .3 & 1 & .1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}$$

TensorFlow in 1 slide

- TensorFlow is an Open Source, C++ library for ML
- Its primary (think 90%+) wrapper is Python
- Java integration requires… craftiness!

What are Tensors?

- Have types (float, int, etc.)
- Have ranks — which is a fancy word for “number of dimensions”
- Have shapes — which is their size in each rank (aka dimension)

Tensor Definition Examples

- Rank 0 = Scalar
- Rank 1 / Shape {5} = 5 element array
- Rank 2 / Shape {3,3} = 3x3 Matrix

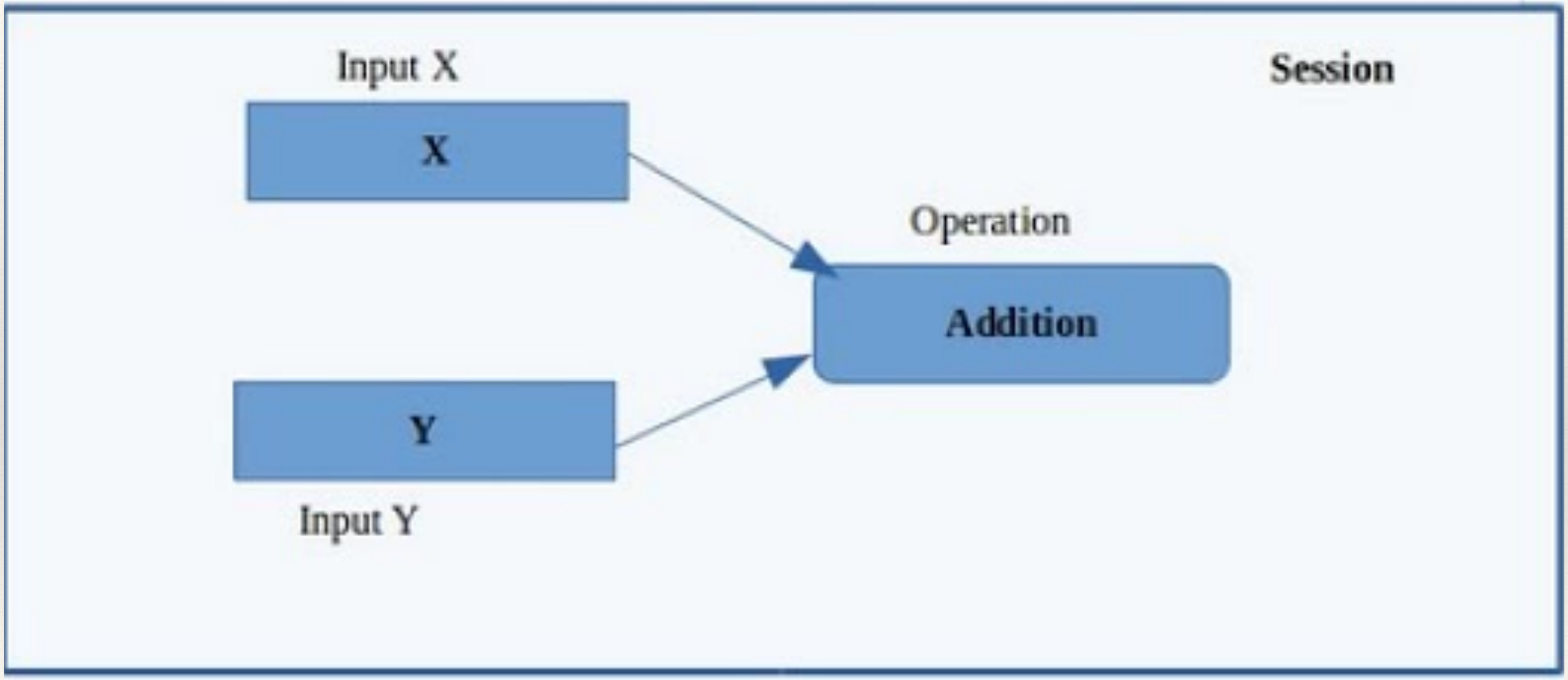
More About Tensors…

- Tensors may be contained by Variables
- Tensors may be mutated by Operations
- Graphs contain Tensors, Variables and Operations

Why is it called TensorFlow?

- “stateful data**flow** graphs”
- Sessions provides execution environments for Graphs to flow
- Tensors trained via mutation of Variables
- Models asked to make Predictions after Training

Simple Addition

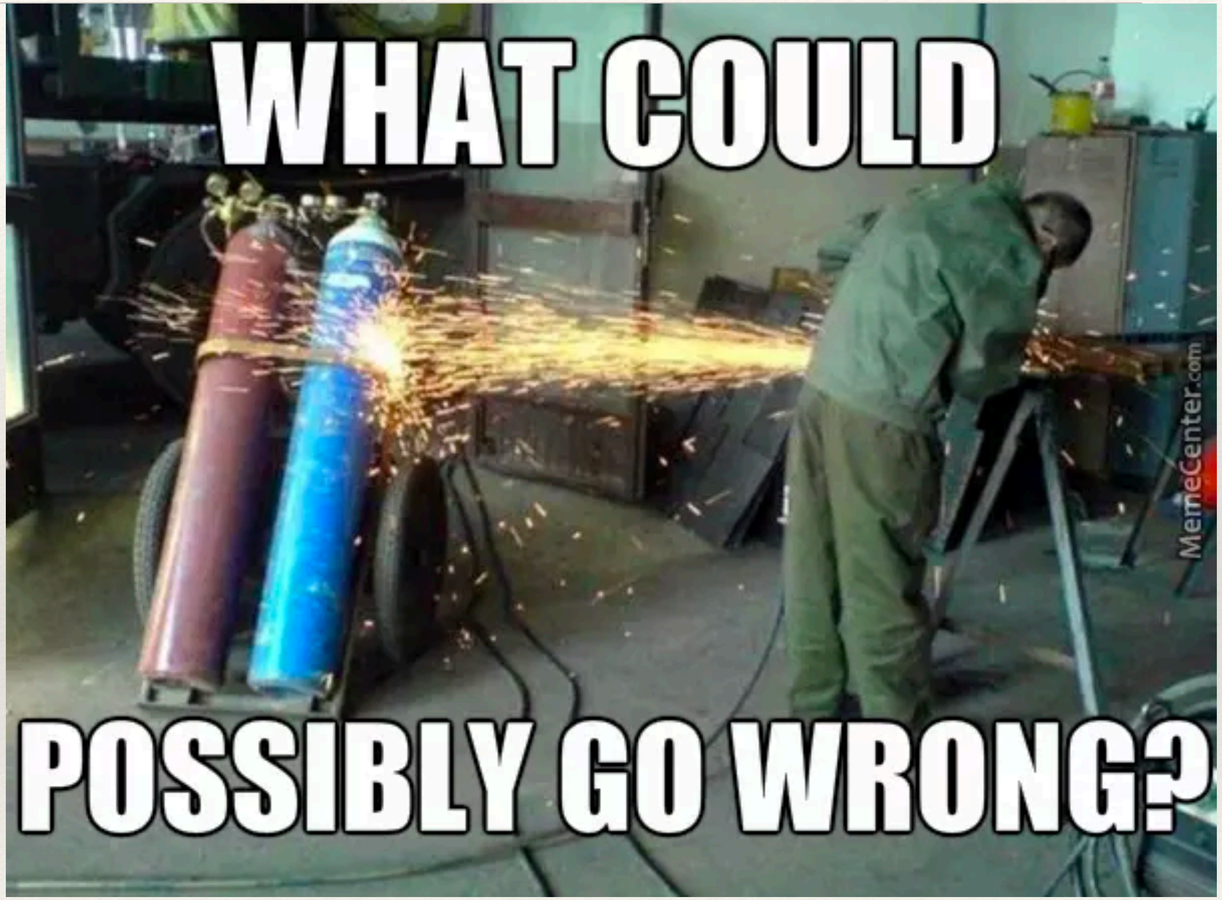


The TensorFlow Java SDK

- Available as a maven dependency (and other formats)
- TensorFlow Lite allows basic TensorFlow on Android Java
- Wraps maybe 20% of the TensorFlow classes

What's in the 20%?

- Basic model creation
- Model training
- Inline predictions
- Predictions across the network



Essential Classes

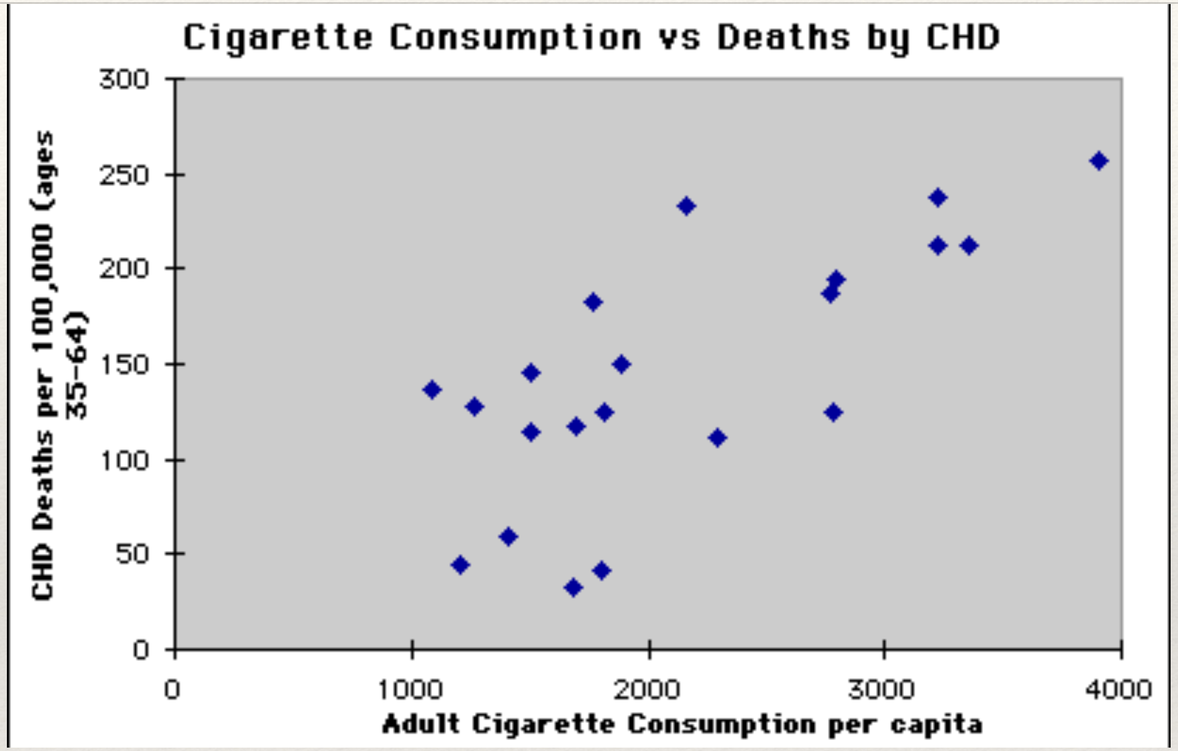
- `org.tensorflow.Tensor<T>`
- `org.tensorflow.Output`
- `org.tensorflow.Graph`
- `org.tensorflow.Session`

Hands On Lab

Demo 1

Adding 2 numbers

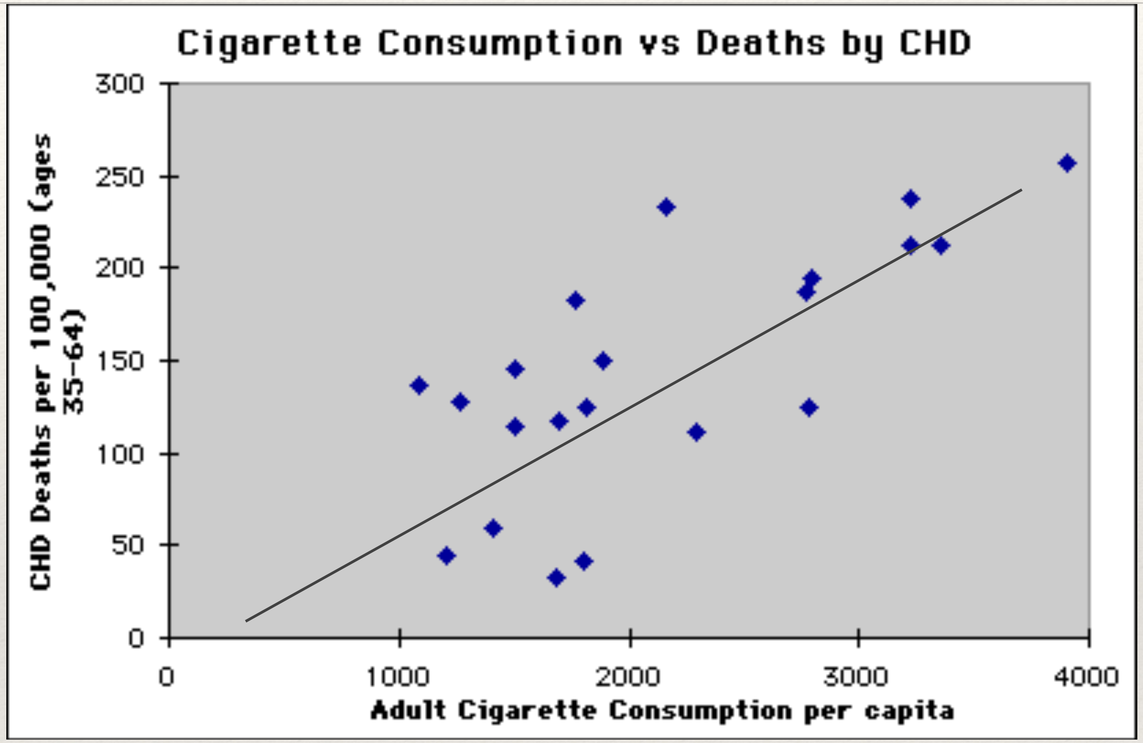
Start with Raw Data



Raw Data Points

Predict via Linear Regression

- $y = mX + b$



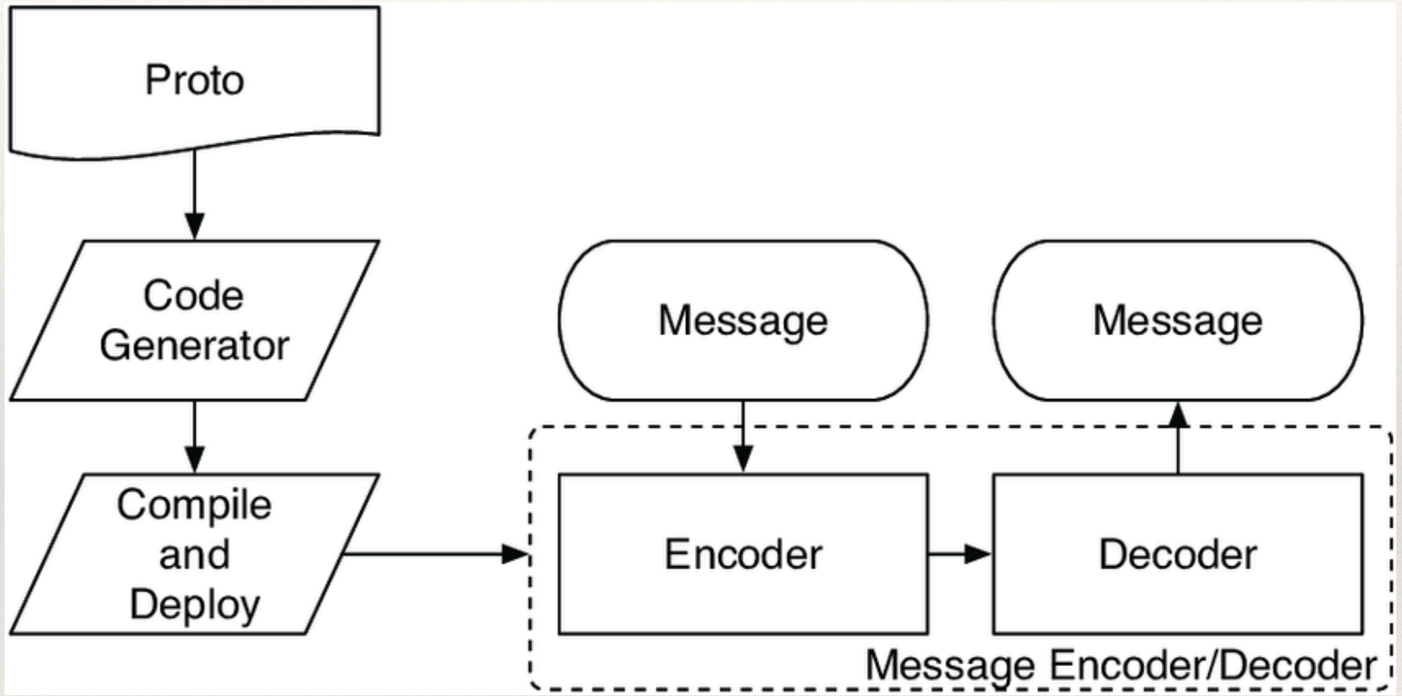
Fit Line

Evolving a model

- We want a class that adjusts the model to fit the data
- TF provides the GradientDescentOptimizer
- Of course - this isn't in the 20% :-(

Getting the GradientDescentOptimizer

- You can download a binary of an untrained GDO model online
- Stored as Protobufs

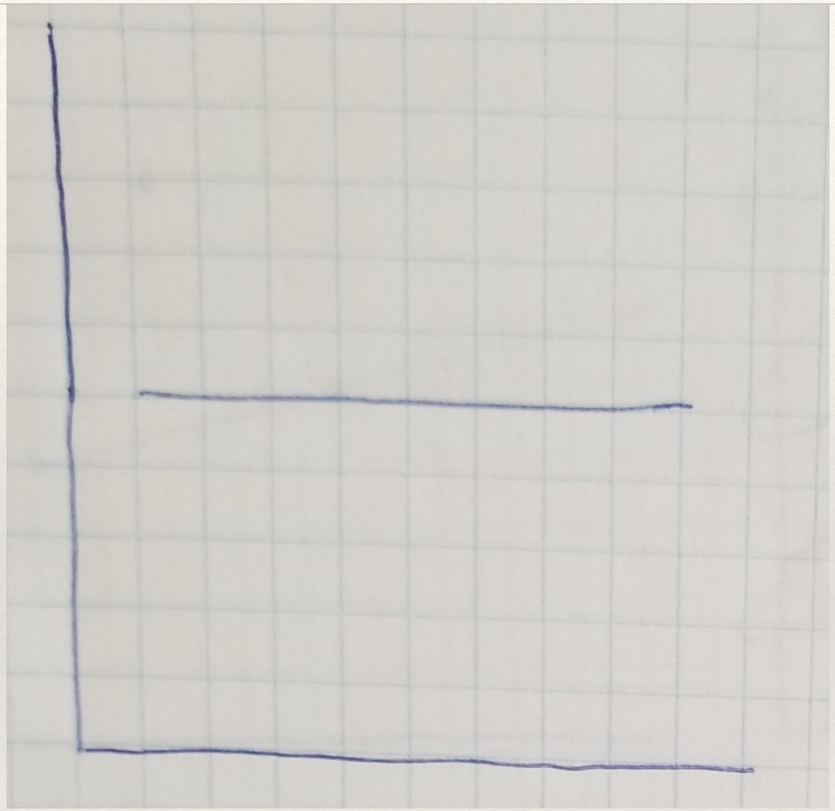


Getting the GradientDescentOptimizer

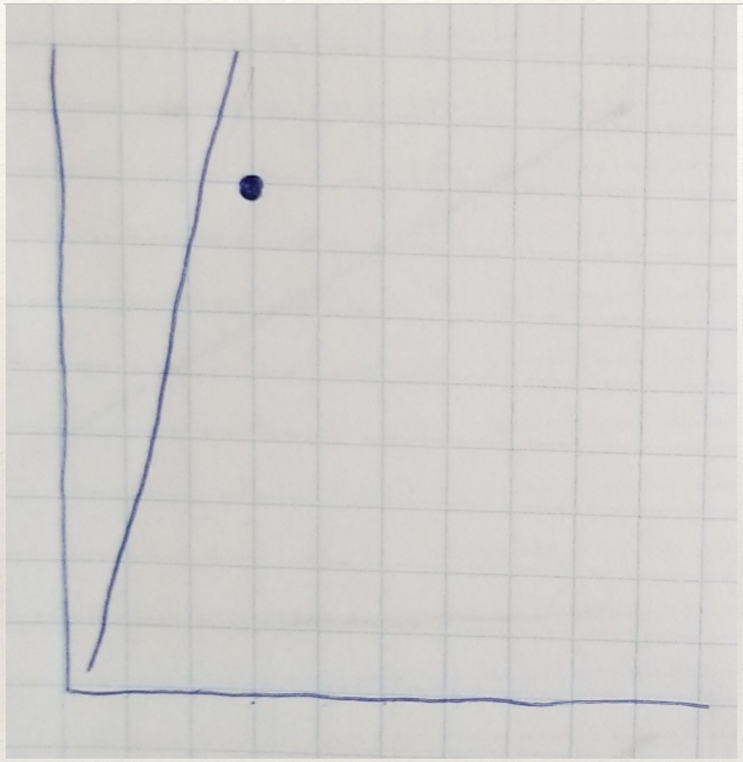
http://derekferguson-6c9ab059-eval-test.apigee.net/hello?w=3.0&b=2.0&lr=0.1

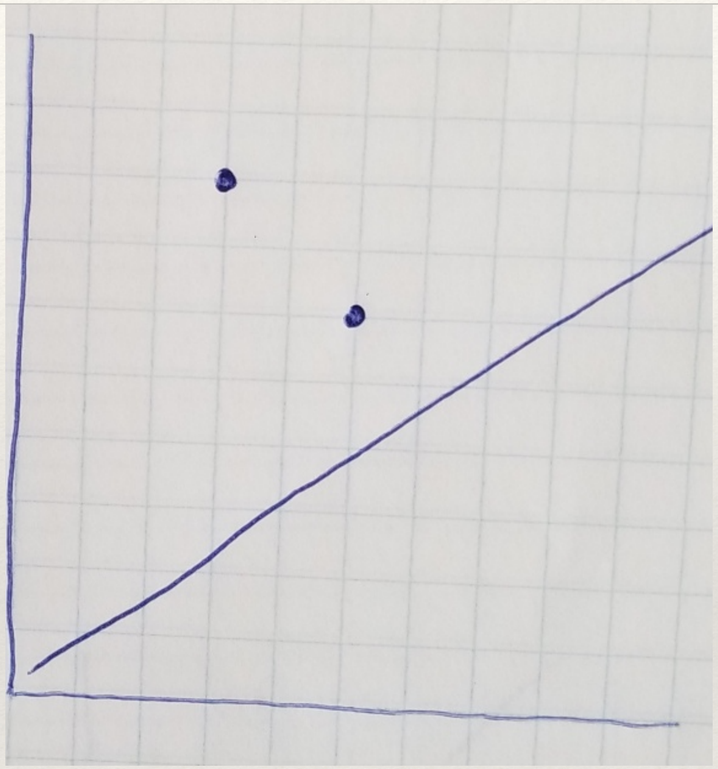
- w — Provide the initial “slope”
- b — Provide the initial “y intercept”
- lr — Provide the “learning rate”

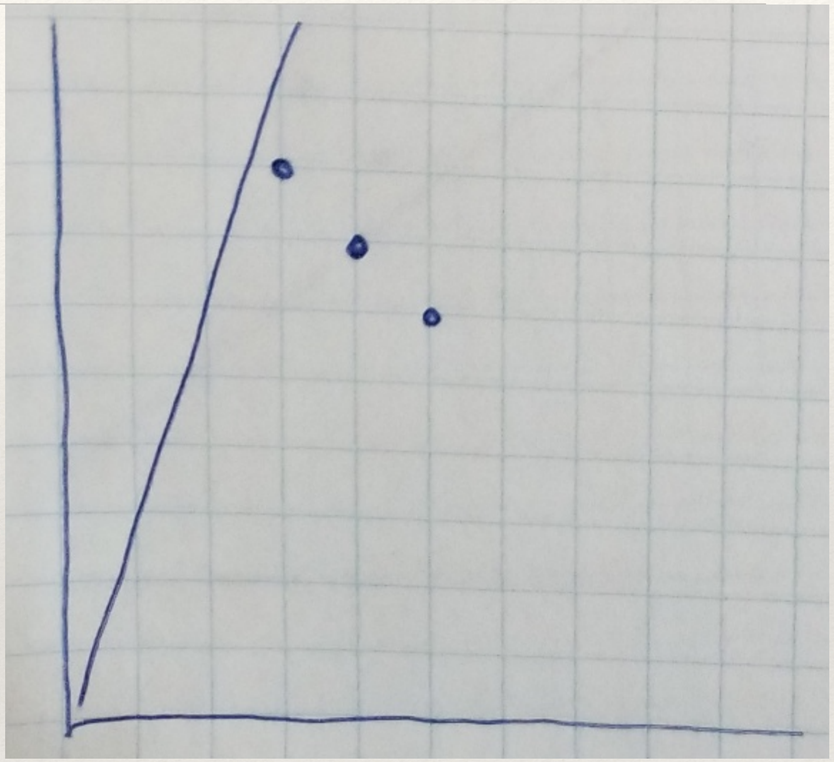
Where to start?



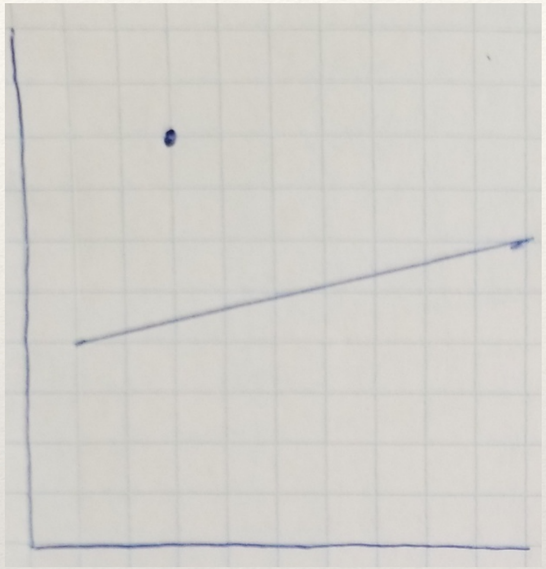
Learning Rate - don't set too large!

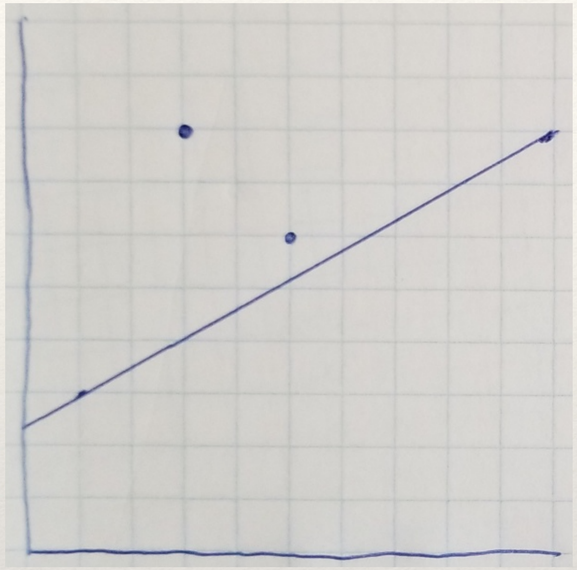


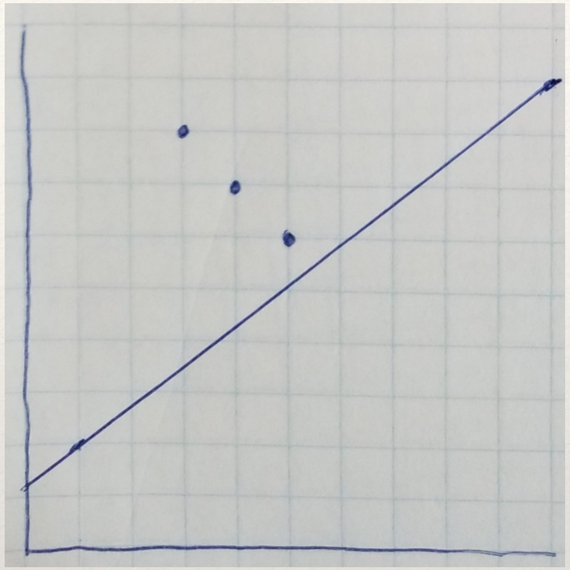


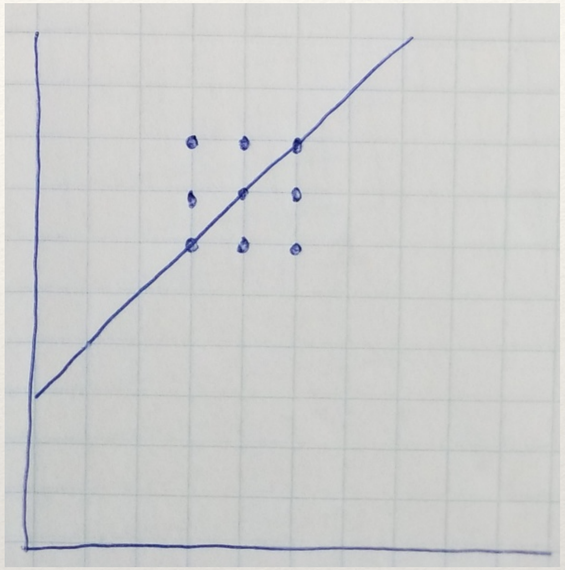


Learning Rate - set low and have lots of data!









New TensorFlow Concepts

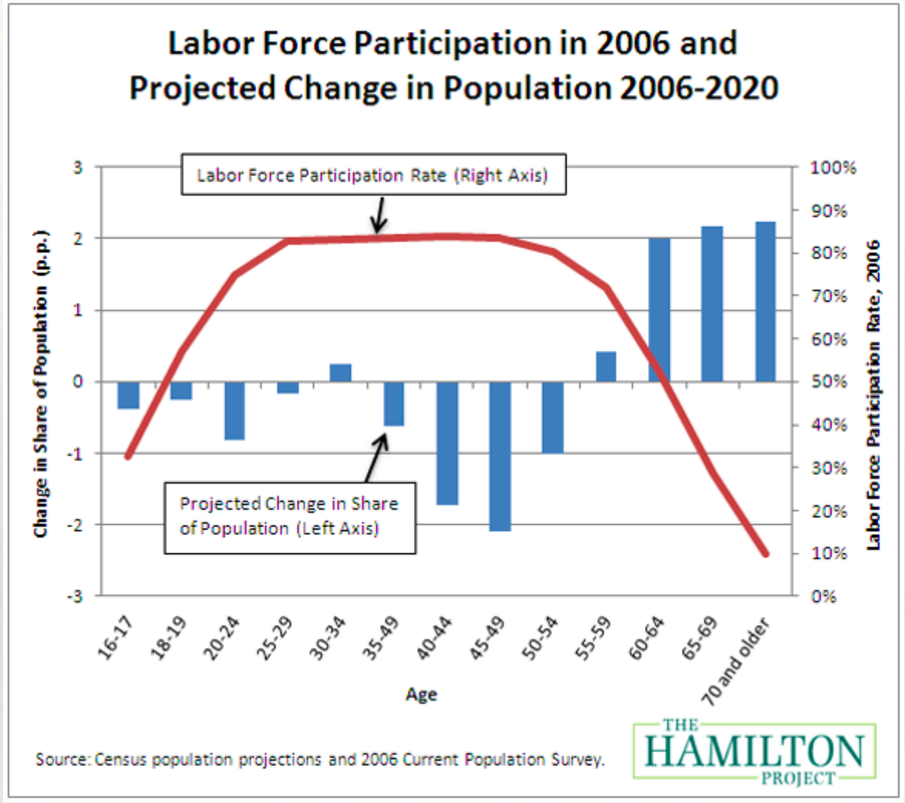
- Graph importation
- Place holders
- Feeds

Hands On Lab

Demo 2

Training a model

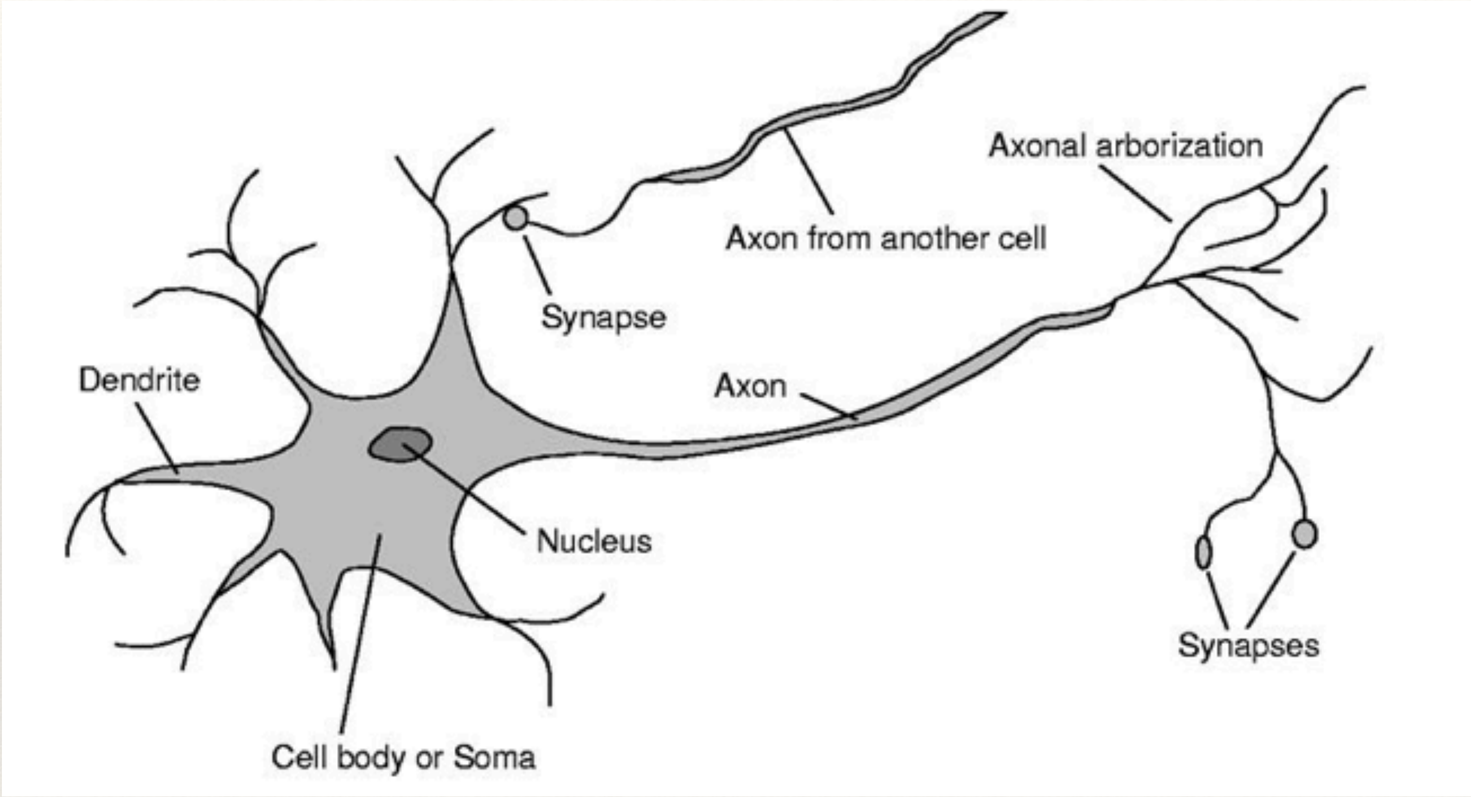
Is everything a line? (Clue: No)



Linear Regression Isn't Enough

- Training to recognise addition vs. subtraction =
  - Given A, B, and C, "predict" that A and B were either added or subtracted to produce C
- Surveys = given answers A, B, C, etc., predict outcome Z
- Recognizing images = given matrix of points, "predict" identity
- The key: Neural Networks!

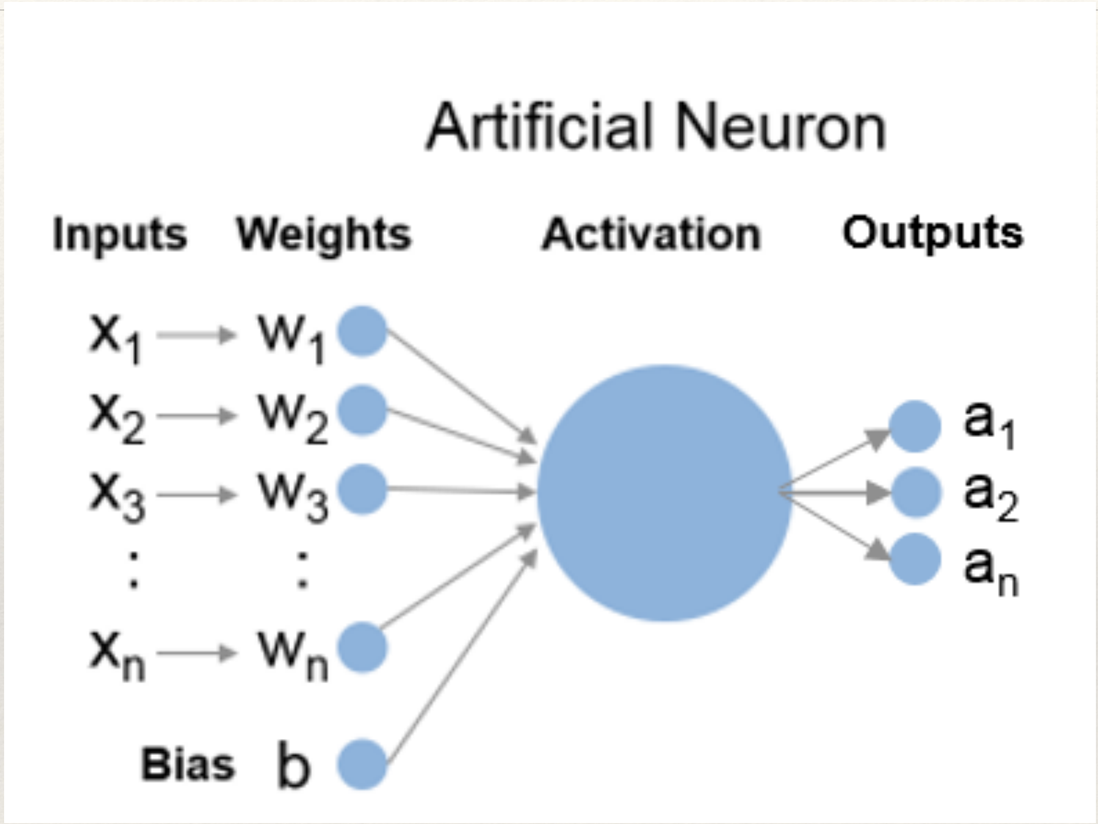
Constant 86B Neurons, Variable Connections



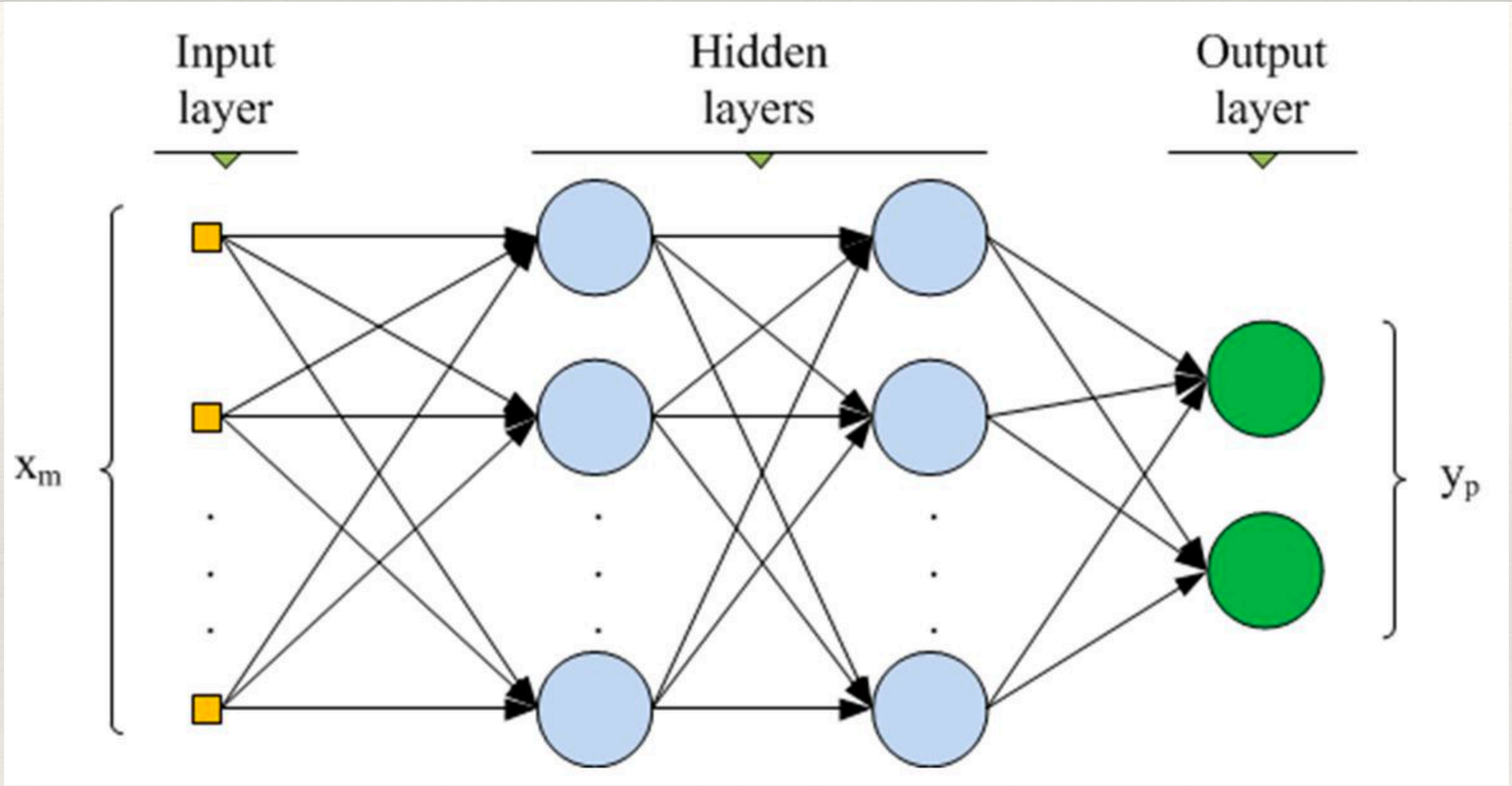
Neural Networks in TensorFlow - pt 1

- In TensorFlow, this is modeled/trained as…
  - Every “neuron” is a tensor
  - Weights - sensitivity to each input (for now, data features)
  - Biases - constant added to output

Neural Networks in TensorFlow - pt 1



Neural Networks in TensorFlow - pt 2



DNNClassifier to the Rescue… almost!

- TensorFlow provides a DNNClassifier for neural networks
- Of course, this isn't in the 20%, either…
- For this, we'll use a pre-trained model (like output of Demo 2)
  - Fits the most common real-world team structures
  - See me in the discussion area to see the Python code

Demo 3 - How It Was Previously Trained

Input A	Input B	Input C	Label
58	18	40	Subtraction
2	100	102	Addition
3	1	2	Subtraction
578212	312000	890212	Addition

Essential Class

- `org.tensorflow.SavedModelBundle`

Hands On Lab

Demo 3

Inline Prediction: Recognizing Add and Subtract

Deploying Trained Models

- Before TF 1.8, Python users used Flask
- Since TF 1.8, TensorFlow Serving is available

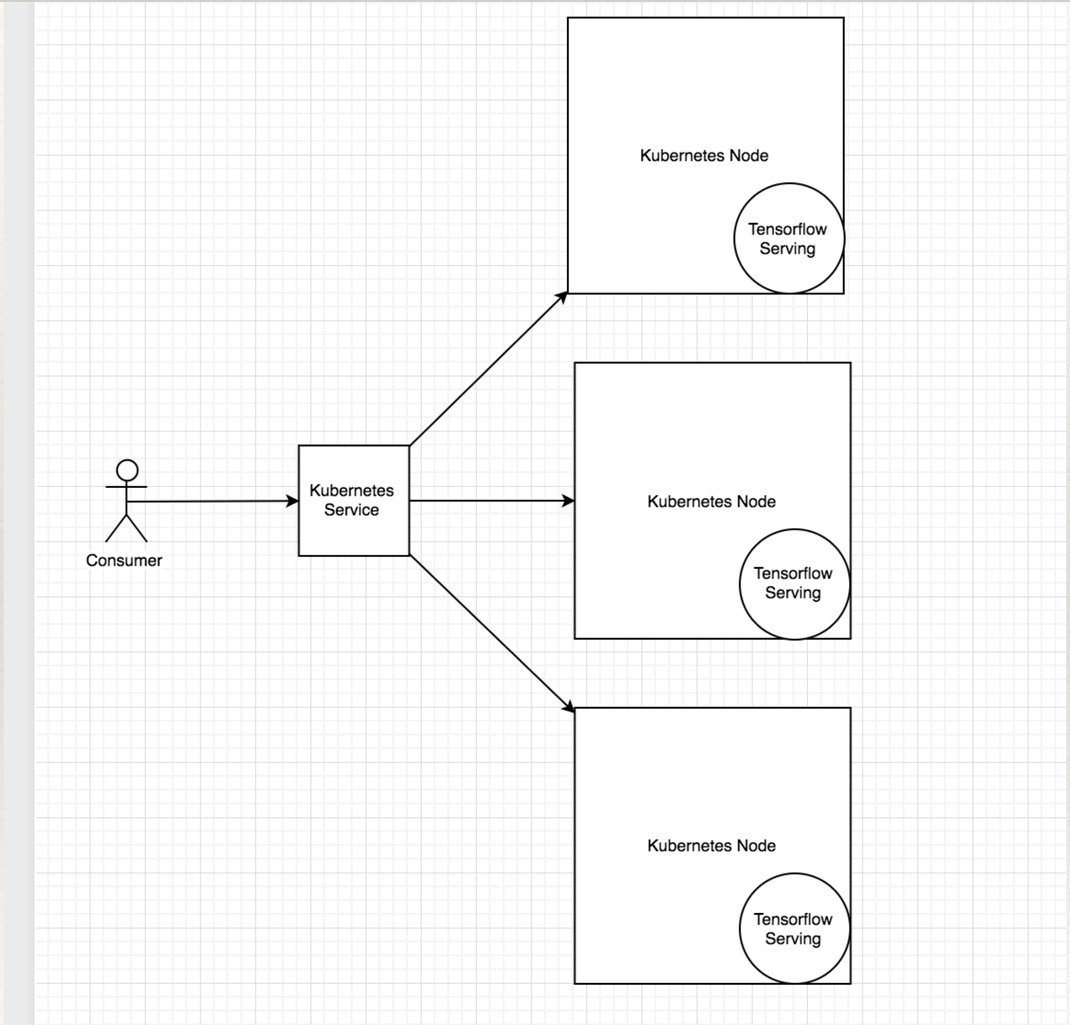
```
FROM tensorflow/serving

COPY exports/1530401406 /models/model

EXPOSE 8500 (Protobuf)

EXPOSE 8501 (REST)
```

Executing your Docker predictor

```
docker build -t predictor .
```

```
docker run
  -p 8500:8500 -p 8501:8501
  predictor
```

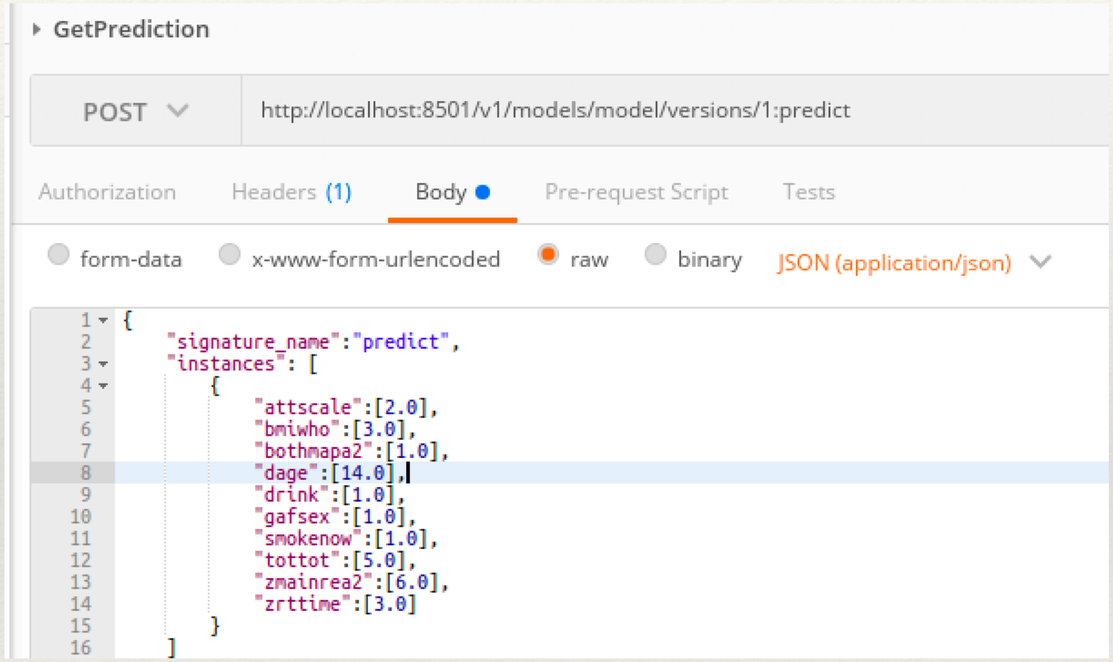
Orchestrating your Docker predictor



Kubernetes with TF Serving

Protobuf on 8500, but… let's REST

```
saved_model_cli.py show –dir DirectoryOfModel –all
```



10 Input Features above (X)

Hands On Lab

Demo 4

TensorFlow Serving via REST

What we learned…

- How machine learning works
- What TensorFlow is
- Where to get the required libraries
- How to obtain, train and invoke TF models

Where to from here?

- More of the missing 80%
- Advanced model creation
- Unsupervised learning
- Remember: the whole thing is Open Source!

Q&A

Contact me! English или по-русски!
https://github.com/JavaDerek/JokerTfJava

–Derek Ferguson — derek.ferguson@mail.ru or derek.ferguson@chase.com