welcome to

Mission:

To offer a AI education to anyone on Earth for free. Our doors are open to all those who wish to learn. We are a learning community that spans almost every country dedicated to teaching our students how to make a positive impact in the world using AI technology, whether that's through employment or entrepreneurship.

Values:

1. Embrace the Weird
2. Inspire and Educate
3. Data Driven Optimism
4. Rapid Experimentation
5. Be Frugal
6. Choose Love, not Fear

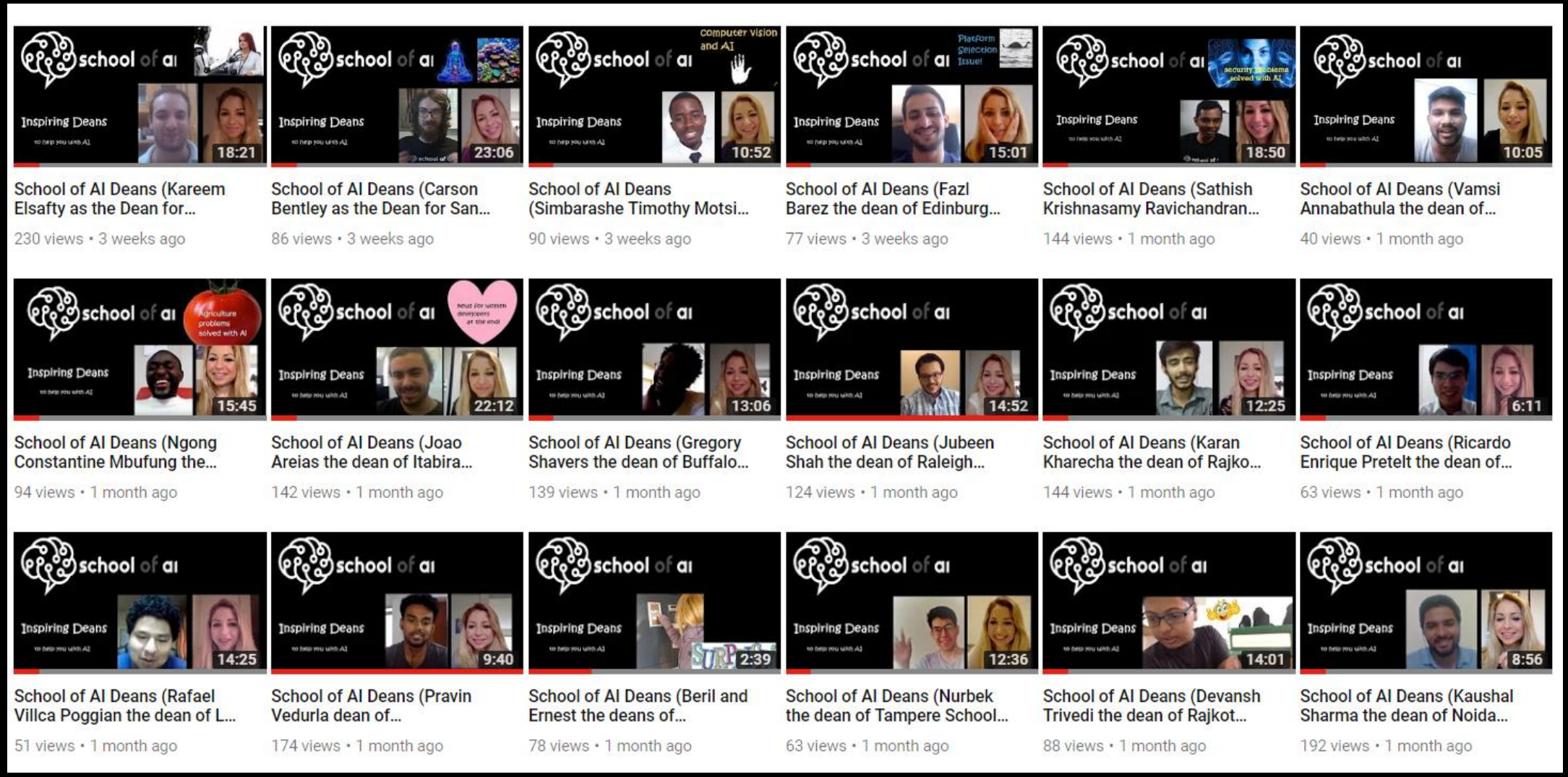
Beril Sirmacek www.BerilSirmacek.com

- Artificial Intelligence
- Augmented Reality
- Computer Vision

- Remote Sensing
- Medical Applications

- create4D www.create4D.com
- farmAR

https://www.youtube.com/user/DrSirmacek/videos



Review questions:

Which learning algorithm would you choose in order to solve these problems?

Classify different fruits

Predict weather

based on various sensor data

Learn to play chess by trial and error

Review questions:

Which learning algorithm would you choose in order to solve these problems?

Classify different fruits

UNSUPERVISED

Predict weather

based on various sensor data

SUPERVISED

Learn to play chess by trial and error

REINFORCEMENT

Goals for today:

1. Features
2. Dimensionality Reduction
3. Autoencoders & PCA
4. Tensors
5. TensorFlow Session
6. TensorBoard

What are good features?

What are good features?

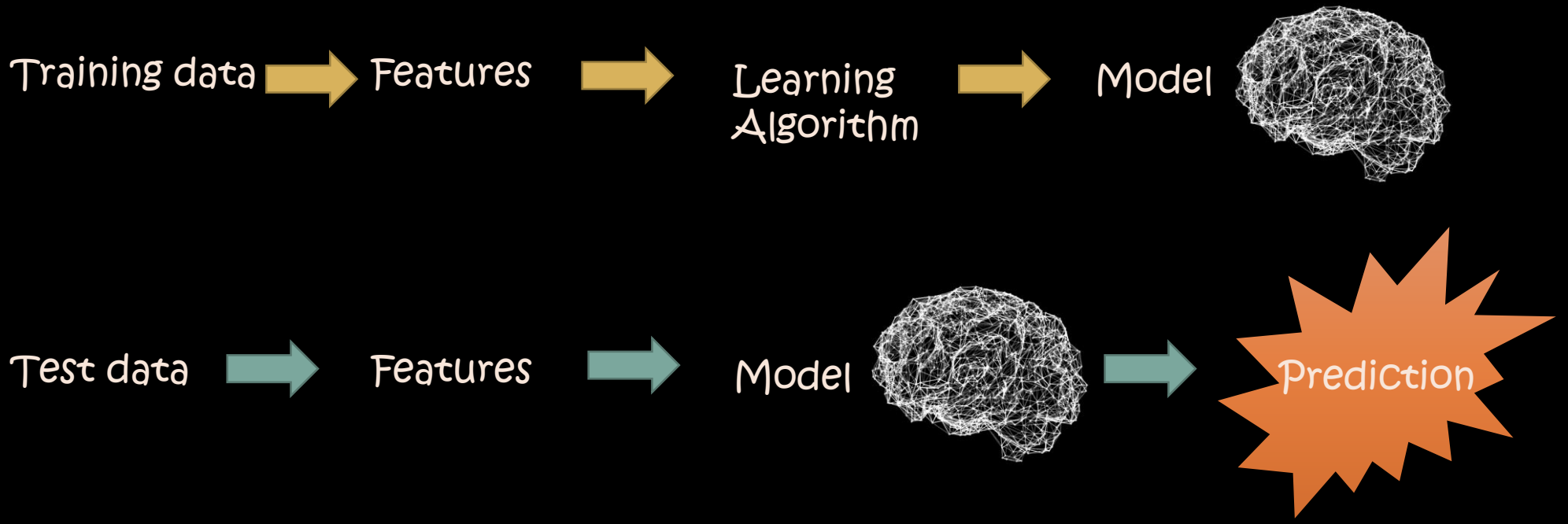
Lots of features

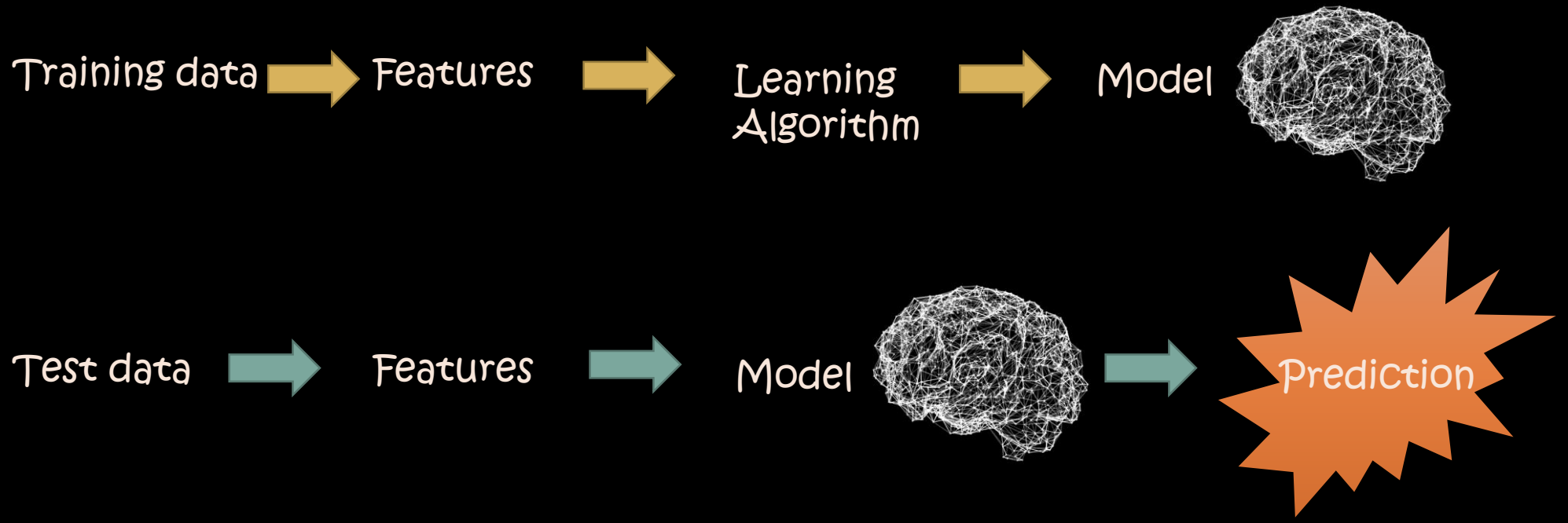
= Lots of information

?!?!?!?!?!!!!!!

Few things to consider;

- Noisy? (reliably distinguishing the classes?)
- Causing over-fitting?
- Highly correlated with other features?
- Slowing down the system?


Training data
Features
Learning Algorithm
Model
Test data
Features
Model
Prediction

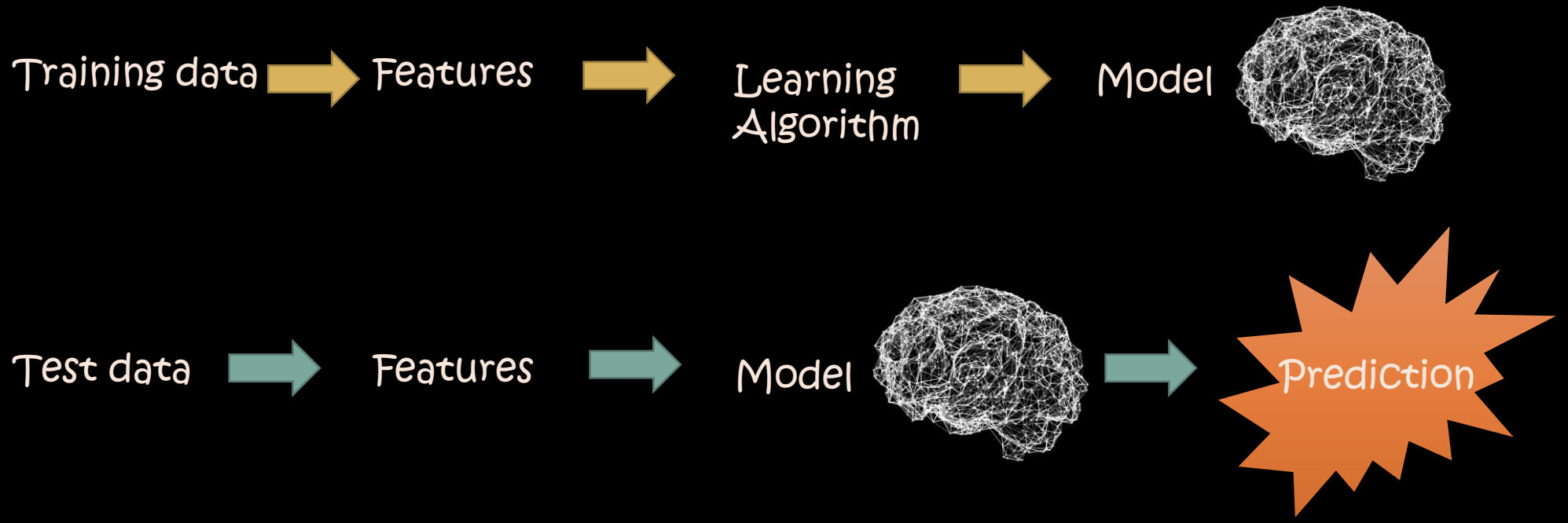


The distance between the ground truth and the prediction is called the cost.

The essence of a supervised machine-learning algorithm to find the parameters of a model which result in the least cost.

Optimization problem: Looking for a Θ that minimizes the cost among all data points $x \in X$

$\text{Cost}(\Theta|X) = \sum \|g(x|\Theta) - f(x)\|$ (distance between the prediction and the groundtruth)



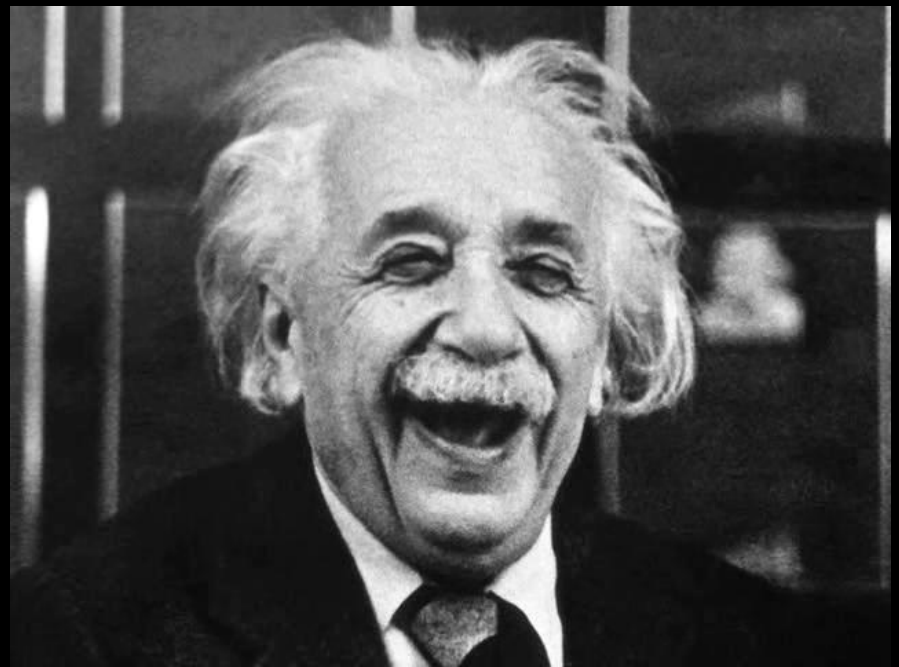
Cost on the training set is very low,
but Cost on the test set is very high,

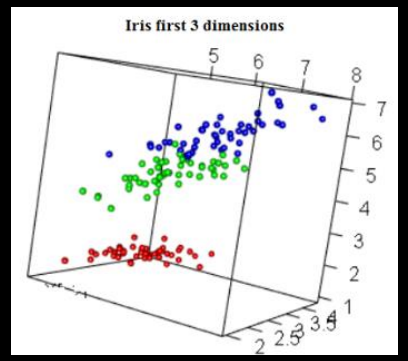
means you have overfitting problem!

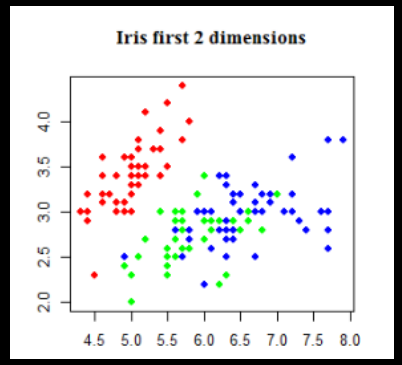
??

Make everything as simple as possible,

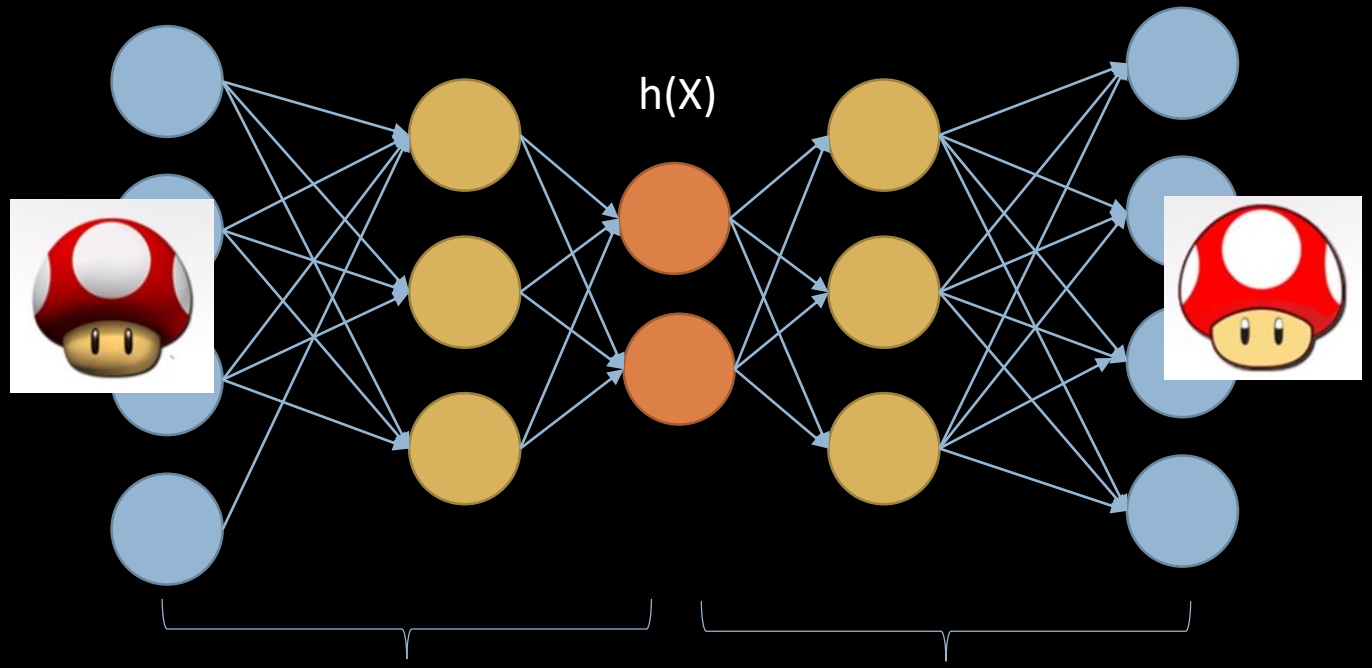
But not simpler.







https://rpubs.com/sandipan/197468



Encoder:
h(X) = sigm(b+Wx)

Decoder:
$\bar{\bar{X}}$ = sigm(b+W^{out}h(x))

Goal:

Find small number of "directions" that explain variation, re-represent data by projecting along those directions

Important assumption:

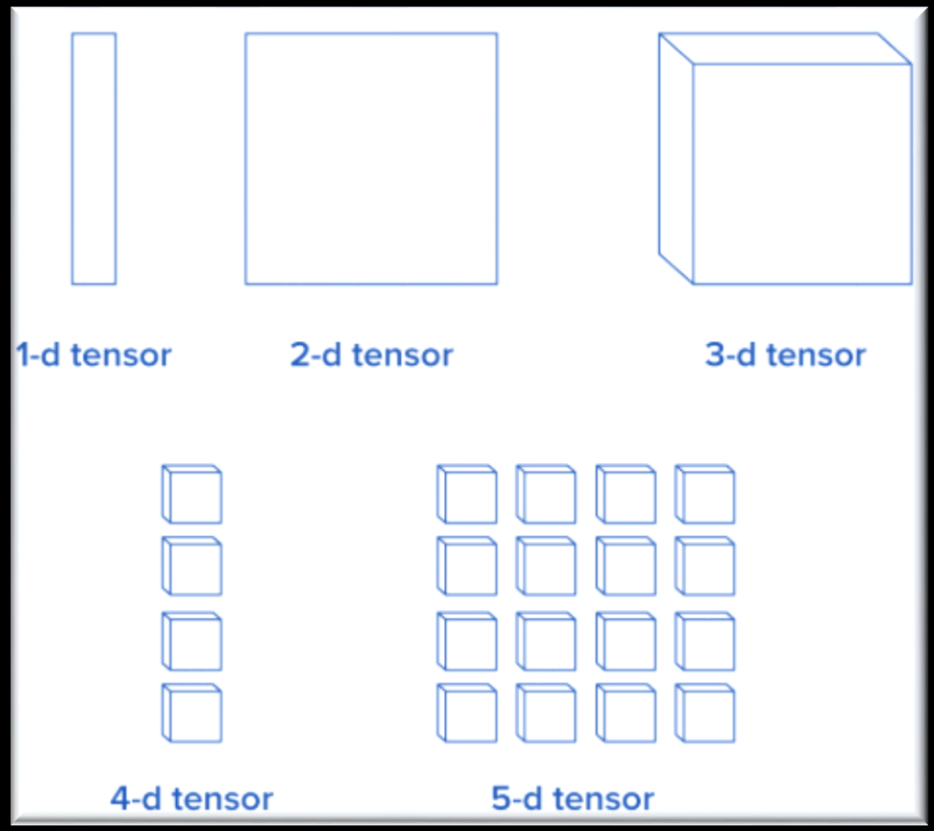
Variation contains information

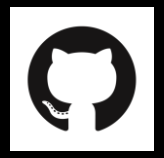
- Input and output layers have the same number of neurons
- Feedforward
- Linear Autoencoders are directly related to PCA
- Weights are not equal to the principal components, and are generally not orthogonal, yet the principal components may be recovered from them https://youtu.be/H1AllrJ-_30

Matrix

Tensor

A **tensor** is a generalization of vectors and matrices to potentially higher dimensions. Internally, TensorFlow represents tensors as n-dimensional arrays of base datatypes.





https://github.com/bsirmacek/schoolofAI_enschede

```
import tensorflow as tf

x = tf.constant([[1.,2.]])
negMatrix = tf.negative(x)

print(x)

with tf.Session() as sess:
    result = sess.run(negMatrix)

print(result)
```

```
Tensor("Const_1:0", shape=(1, 2), dtype=float32)
[[-1. -2.]]
```

Defining a tensor is different than running it.
You have to Run the Session to fill the values of the defined tensor.

```
import tensorflow as tf

x = tf.constant([[1.,2.]])
negMatrix = tf.negative(x)

num1 = tf.constant(5)
num2 = tf.constant(10)
num3 = tf.multiply(num1,num2)


with tf.Session() as sess:
    result = sess.run([negMatrix, num3])

print(result)
```

```
[array([[-1., -2.]], dtype=float32), 50]
```

A session not only configures where your code will be computed on your machine,

but also crafts how the computation will be laid out in order to parallelize the computation.

tf.add(x,y)

tf.substract(x,y)

tf.pow(x,y)

tf.exp(x)

tf.sqrt(x)

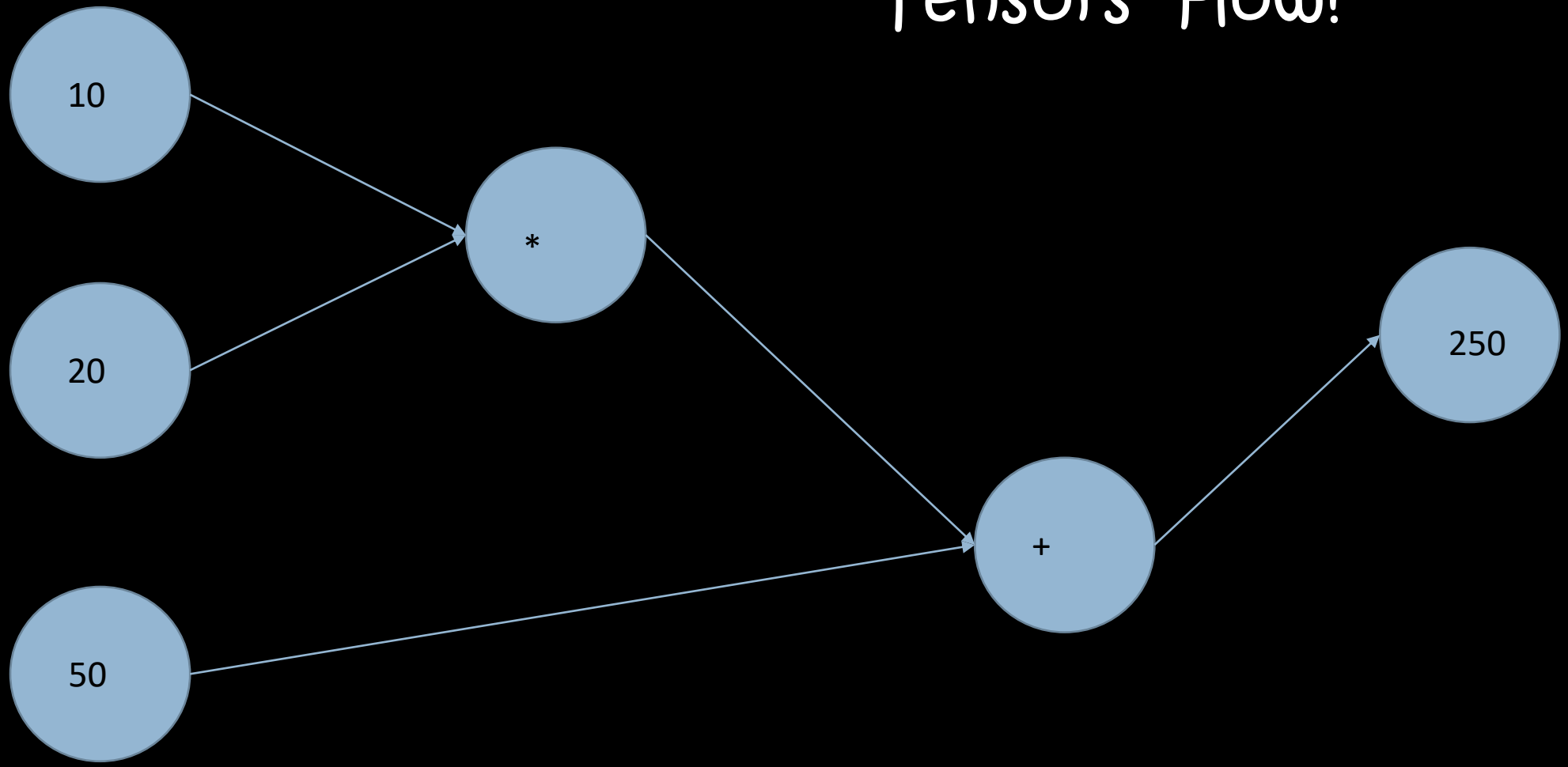
tf.div(x,y)

tf.mod(x,y)

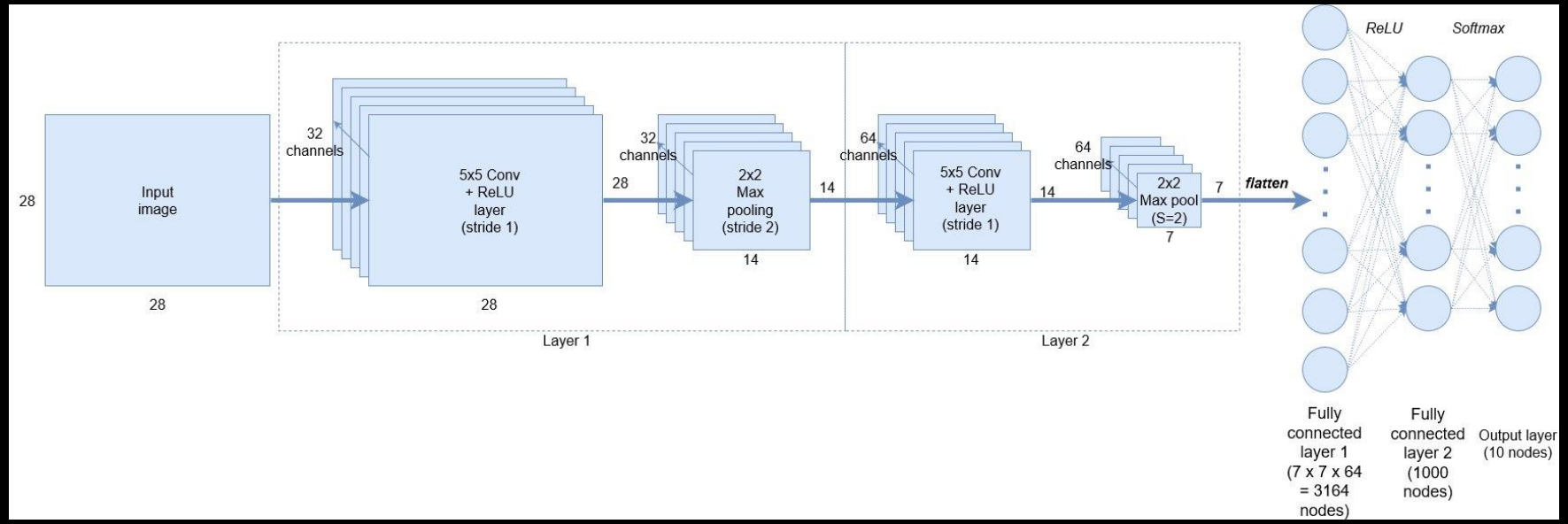
tf.zeros(shape)
tf.ones(shape)

Question: Initialize a 100x100 size tensor with all elements equal to 2

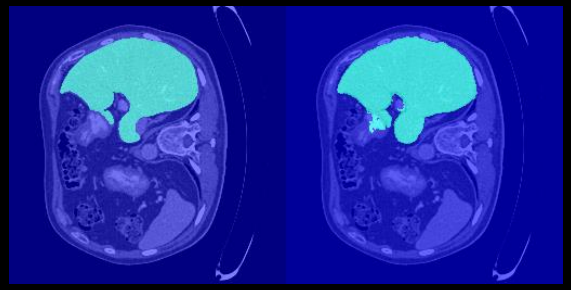
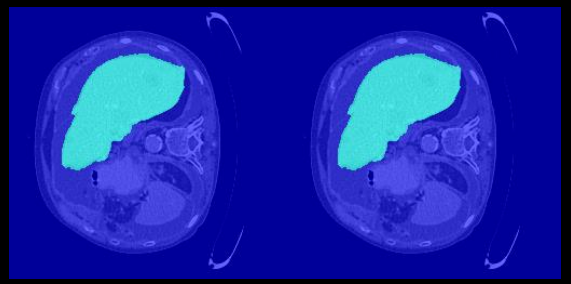
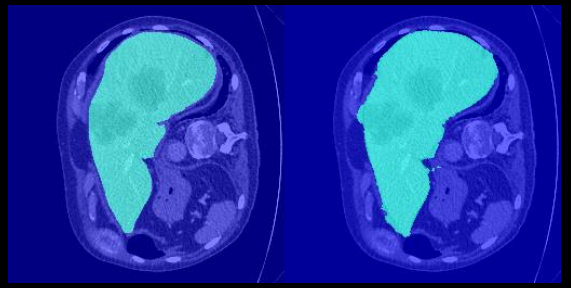
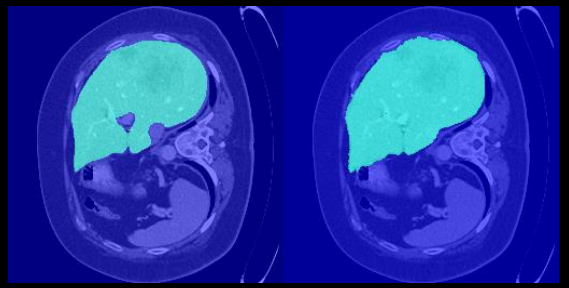
Tensors Flow!

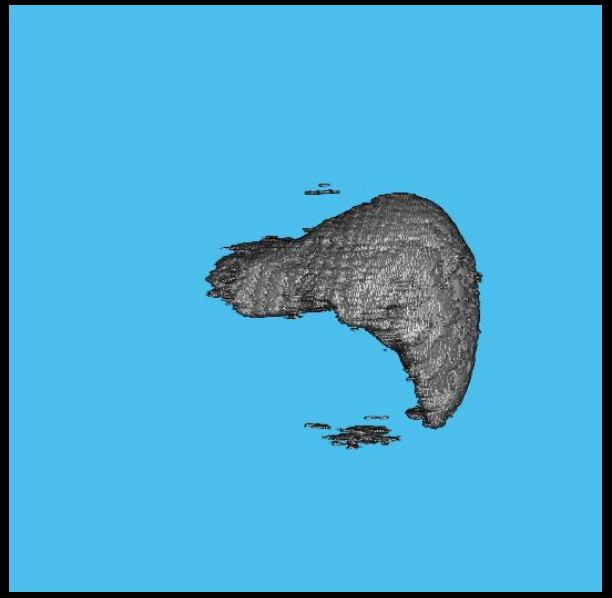
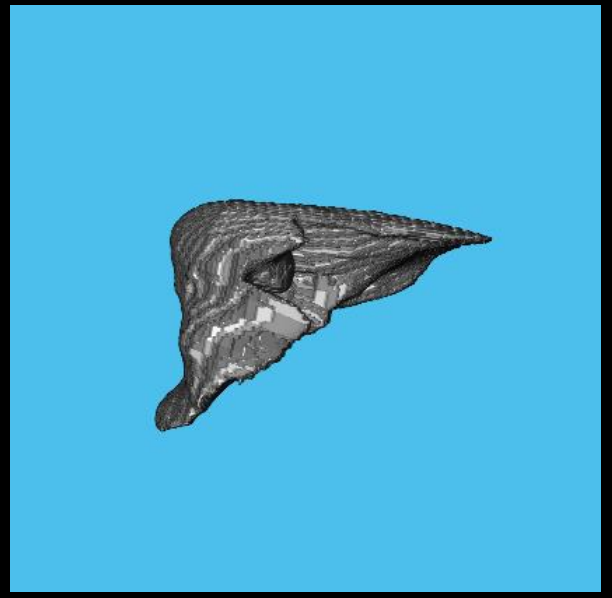


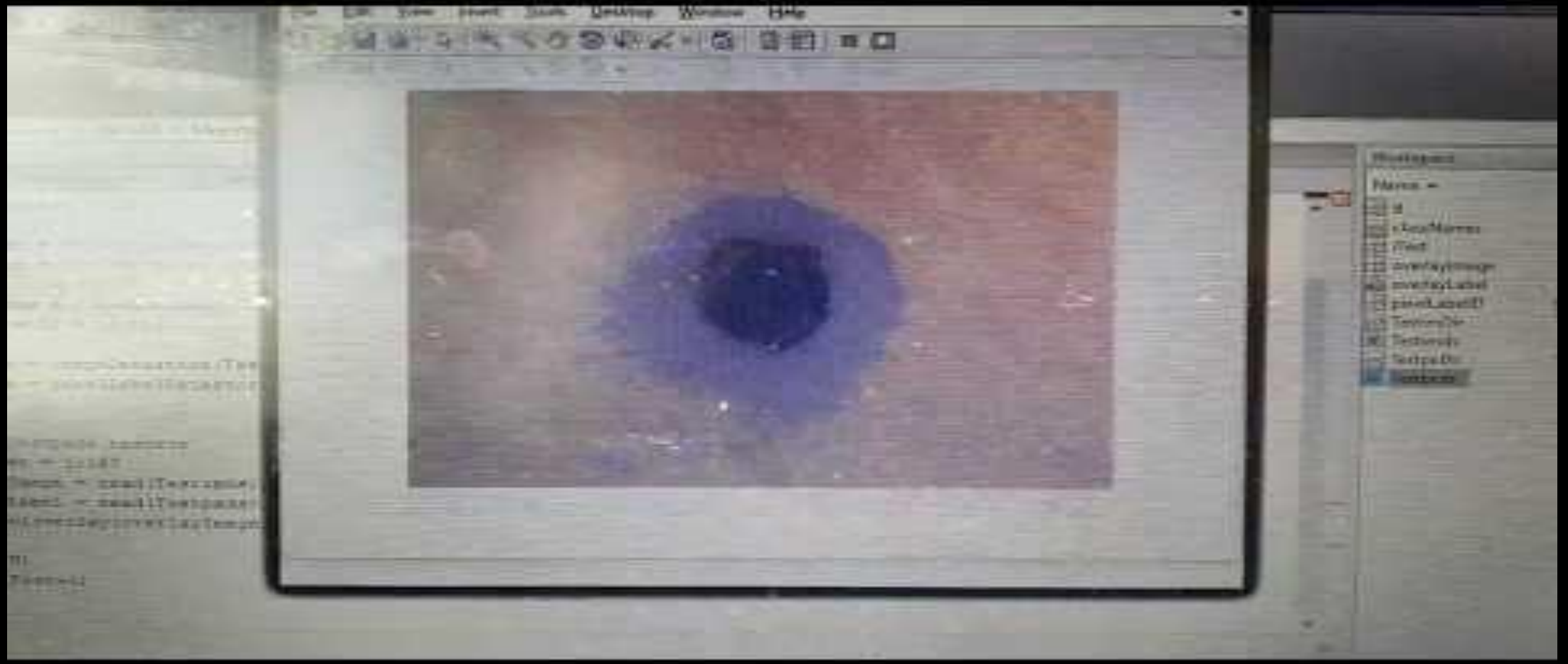
Easy to visualize




```
model = Sequential()
model.add(Conv2D(32, kernel_size=(5, 5), strides=(1, 1), activation='relu', input_shape=input_shape))
model.add(MaxPooling2D(pool_size=(2, 2), strides=(2, 2)))
model.add(Conv2D(64, (5, 5), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Flatten())
model.add(Dense(1000, activation='relu'))
model.add(Dense(num_classes, activation='softmax'))

model.compile(loss=keras.losses.categorical_crossentropy, optimizer=keras.optimizers.SGD(lr=0.01), metrics=['accuracy'])

model.fit(x_train, y_train, batch_size=batch_size, epochs=epochs, verbose=1, validation_data=(x_test, y_test), callbacks=[history])

score = model.evaluate(x_test, y_test, verbose=0)
print('Test loss:', score[0])
print('Test accuracy:', score[1])
```

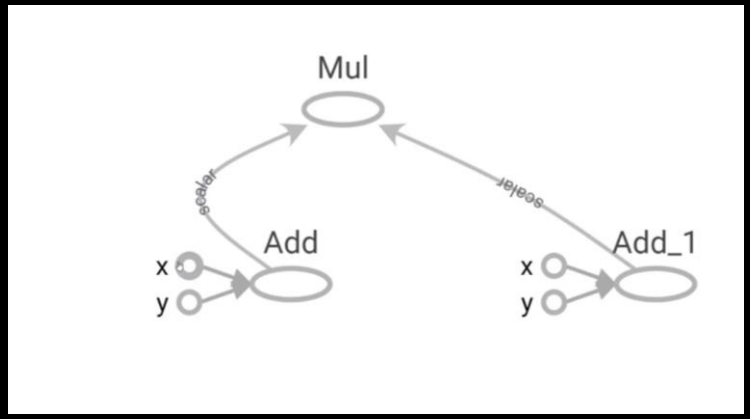







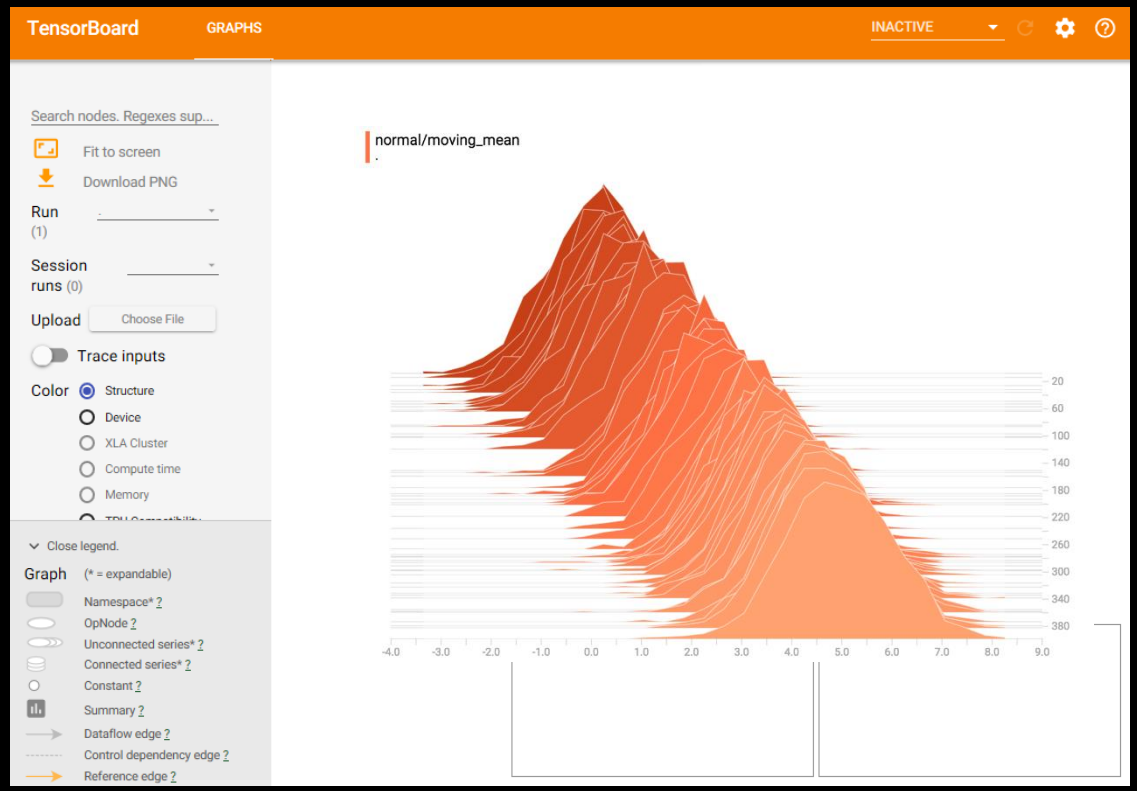
Interactive visualization environment

A flow chart of the way data transforms



>>tensorboard –logdir=./output

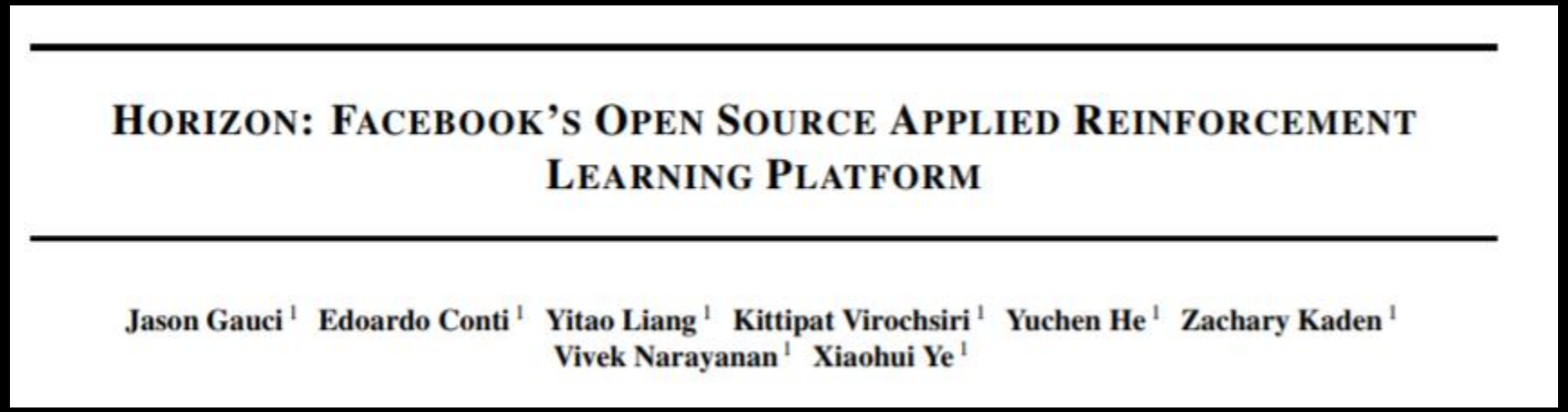
http://copymetothebrowser



https://www.tensorflow.org/guide/summaries_and_tensorboard

https://www.tensorflow.org/guide/tensorboard_histograms

Discussion



HORIZON: FACEBOOK'S OPEN SOURCE APPLIED REINFORCEMENT LEARNING PLATFORM

Jason Gauci [1] Edoardo Conti [1] Yitao Liang [1] Kittipat Virochsiri [1] Yuchen He [1] Zachary Kaden [1]
Vivek Narayanan [1] Xiaohui Ye [1]

Historically, we have used supervised learning models for predicting click through rate (CTR) and likelihood that the notification leads to meaningful interactions. These predictions are combined into a score that is used to filter the notifications.

https://arxiv.org/pdf/1811.00260.pdf

Discussion

https://experiments.withgoogle.com/ai/ai-duet/view/