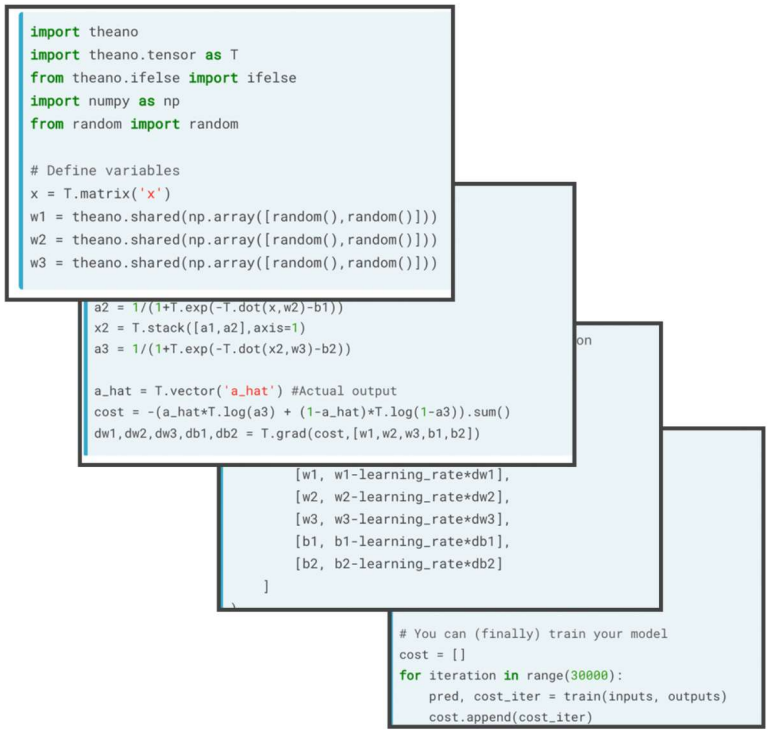
Theano vs Keras



```
import theano
import theano.tensor as T
from theano.ifelse import ifelse
import numpy as np
from random import random

# Define variables
x = T.matrix('x')
w1 = theano.shared(np.array([random(),random()]))
w2 = theano.shared(np.array([random(),random()]))
w3 = theano.shared(np.array([random(),random()]))
```

```
a2 = 1/(1+T.exp(-T.dot(x,w2)-b1))
x2 = T.stack([a1,a2],axis=1)
a3 = 1/(1+T.exp(-T.dot(x2,w3)-b2))

a_hat = T.vector('a_hat') #Actual output
cost = -(a_hat*T.log(a3) + (1-a_hat)*T.log(1-a3)).sum()
dw1,dw2,dw3,db1,db2 = T.grad(cost,[w1,w2,w3,b1,b2])
```

```
        [w1, w1-learning_rate*dw1],
        [w2, w2-learning_rate*dw2],
        [w3, w3-learning_rate*dw3],
        [b1, b1-learning_rate*db1],
        [b2, b2-learning_rate*db2]
    ]
)
```

```
# You can (finally) train your model
cost = []
for iteration in range(30000):
    pred, cost_iter = train(inputs, outputs)
    cost.append(cost_iter)
```

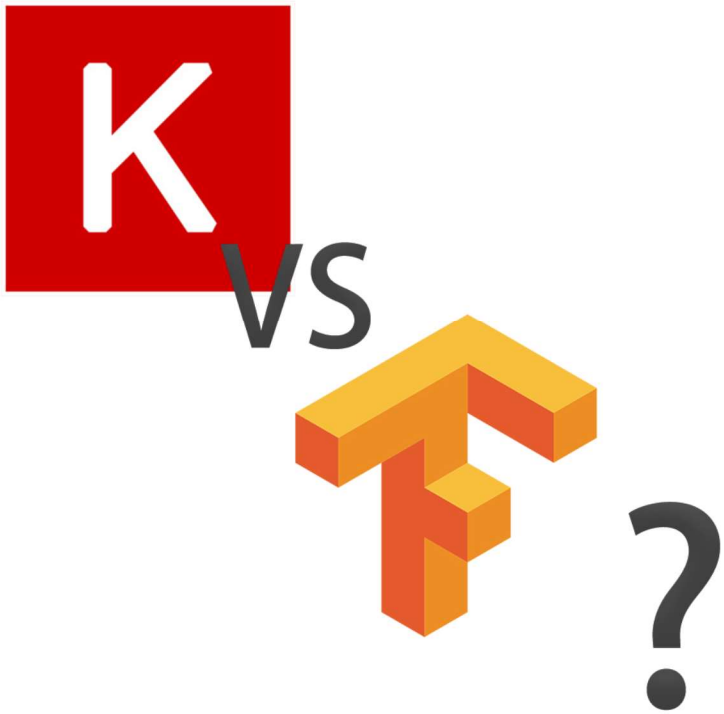
```
from keras.layers import Dense
from keras.models import Sequential

# Define model and add layers
model = Sequential()
model.add(Dense(2,input_shape=(2,),activation='sigmoid'))
model.add(Dense(1,activation='sigmoid'))

model.compile(optimizer='adam',loss='categorical_crossentropy')

# Train model
model.fit(inputs,outputs)
```

Keras

- Deep Learning Framework
- Enables fast experimentation
- Runs on top of other frameworks
- Written by François Chollet

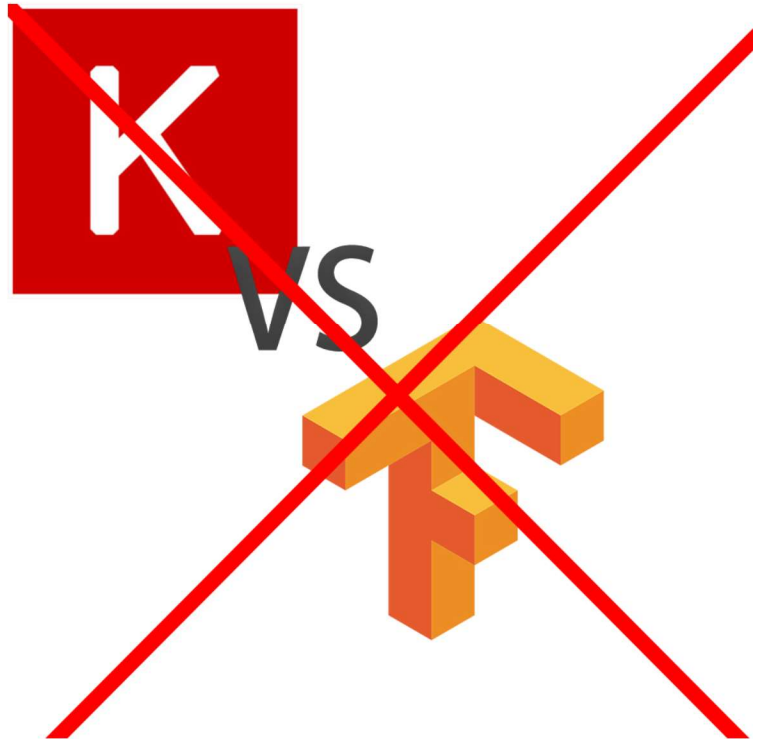
Why use Keras?

- Fast industry-ready models
- For beginners and experts
- Less code
- Build any architecture
- Deploy models in multiple platforms

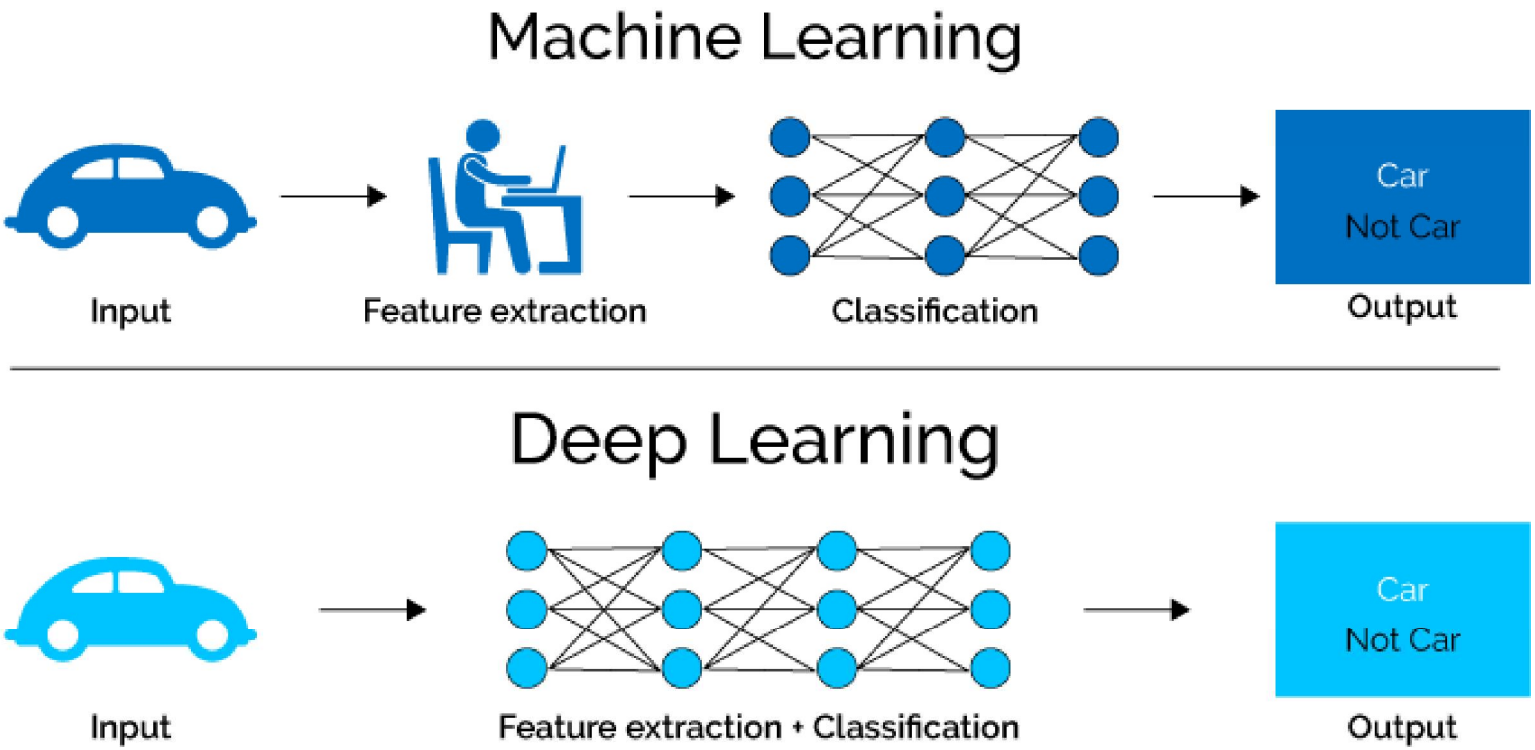


Keras + TensorFlow

- TensorFlow's high level framework of choice
- Keras is complementary to TensorFlow
- You can use TensorFlow for low level features

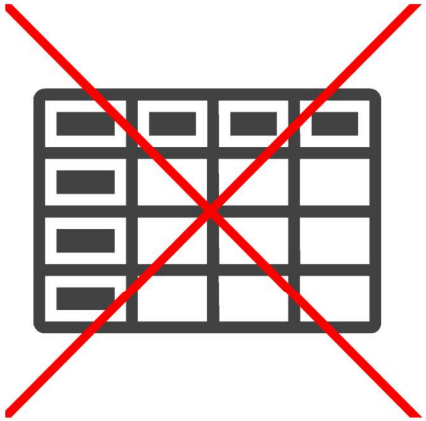


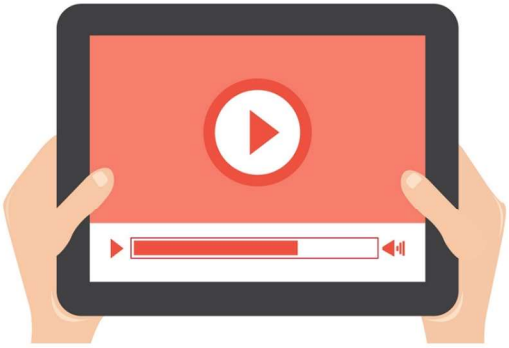
Feature Engineering



[1] Towards Data Science

Unstructured data



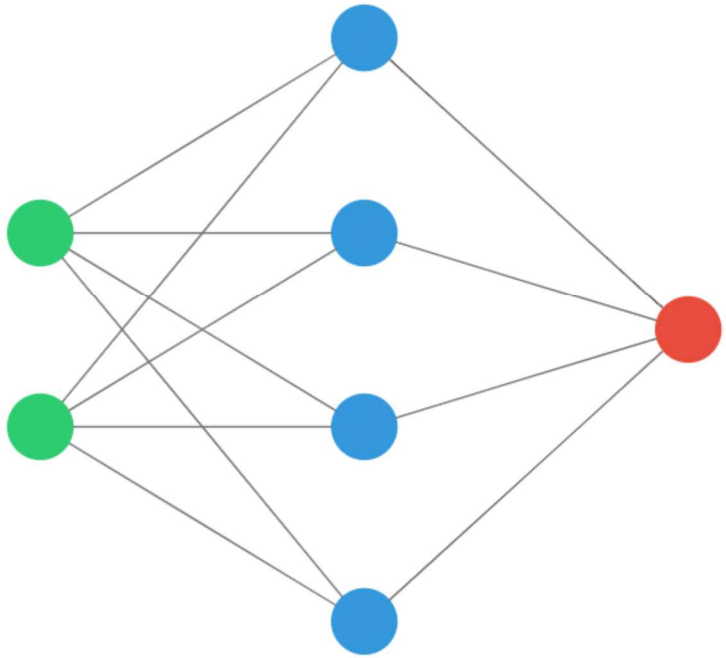


So, when to use neural networks?

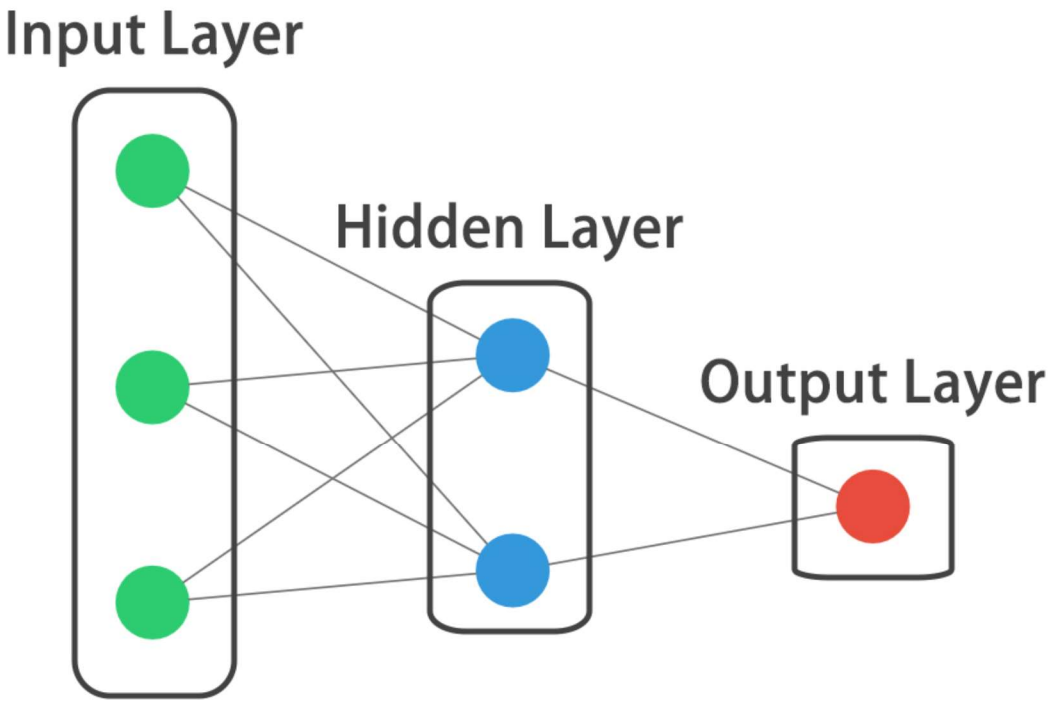
- Dealing with unstructured data
- Don't need easily interpretable results
- You can benefit from a known architecture

Example: Classify images of cats and dogs

- **Images -> Unstructured data**
- You don't care about why the network knows it's a cat or a dog
- You can benefit from convolutional neural networks



A neural network?



Parameters