



模型保存与加载

主讲：龙良曲

Outline

- save/load weights
 - save/load entire model
 - saved_model
-



```
# Save the weights
model.save_weights('./checkpoints/my_checkpoint')

# Restore the weights
model = create_model()
model.load_weights('./checkpoints/my_checkpoint')

loss, acc = model.evaluate(test_images, test_labels)
print("Restored model, accuracy: {:.2f}%".format(100*acc))
```



```
network.save_weights('weights.ckpt')
print('saved weights.')
del network

network = Sequential([layers.Dense(256, activation='relu'),
                      layers.Dense(128, activation='relu'),
                      layers.Dense(64, activation='relu'),
                      layers.Dense(32, activation='relu'),
                      layers.Dense(10)])
network.compile(optimizer=optimizers.Adam(lr=0.01),
               loss=tf.losses.CategoricalCrossentropy(from_logits=True),
               metrics=['accuracy'])
network.load_weights('weights.ckpt')
network.evaluate(ds_val)
```

**JUST
DO
IT.**





```
network.save('model.h5')  
print('saved total model.')  
del network  
  
print('load model from file')  
network = tf.keras.models.load_model('model.h5')  
  
network.evaluate(x_val, y_val)
```

**JUST
DO
IT.**

The text 'JUST DO IT.' is rendered in a bold, black, sans-serif font. The letters are heavily textured with a splatter or ink-blot effect, giving them a gritty, dynamic appearance. The background is a light gray with a subtle, repeating diamond-shaped grid pattern. The overall composition is centered and minimalist.



```
tf.saved_model.save(m, '/tmp/saved_model/')
```

```
imported = tf.saved_model.load(path)
```

```
f = imported.signatures["serving_default"]
```

```
print(f(x=tf.ones([1, 28, 28, 3])))
```

下一课时

Keras实战

Thank You.
