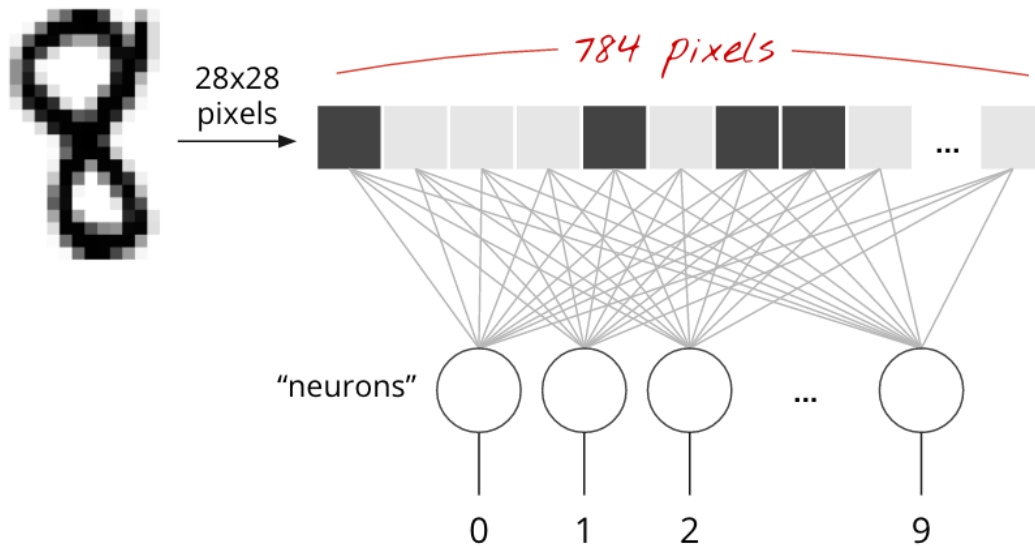
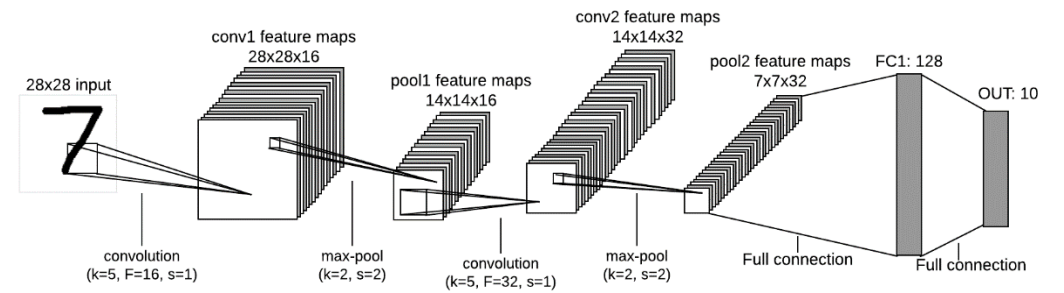


ANN and CNN

ANN



CNN



Difference Levels of coding

使用tf.estimator

```
nn = tf.estimator.Estimator(model_fn=model_fn, params=model_params)
```

高階，快速建立模型

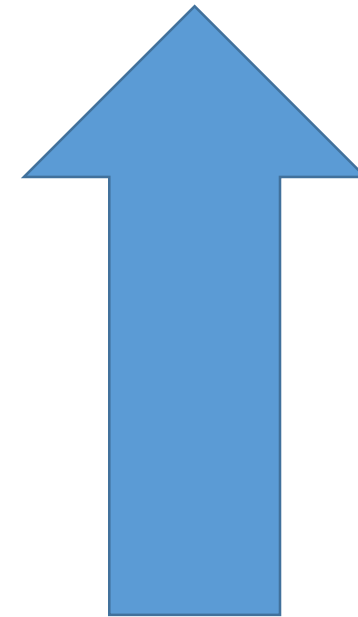
使用tf.keras

```
model = tf.keras.models.Sequential([  
    tf.keras.layers.Flatten(input_shape=(28, 28)),  
    tf.keras.layers.Dense(512, activation=tf.nn.relu),  
    tf.keras.layers.Dropout(0.2),  
    tf.keras.layers.Dense(10, activation=tf.nn.softmax)  
])
```

自己寫

```
layer_1 = tf.add(tf.matmul(x, weights['h1']), biases['b1'])
```

低階，自定度更高



Using Estimators in tensorflow

tf.estimator提供了方法讓你快速地建立regressors和classifiers

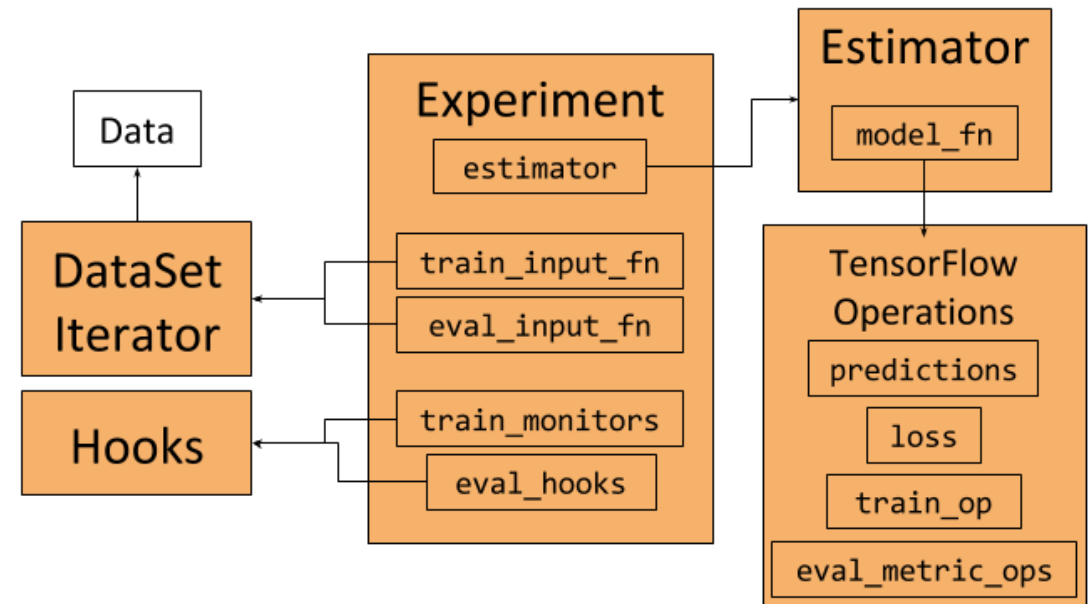
- tf.estimator.LinearClassifier
- tf.estimator.LinearRegressor
- tf.estimator.DNNClassifier
- tf.estimator.DNNRegressor
- tf.estimator.DNNLinearCombinedClassifier
- tf.estimator.DNNLinearCombinedRegressor

Using Estimators in tensorflow

如果tf.estimator提供的方法不敷使用，你也可以自訂estimator

```
nn =  
tf.estimator.Estimator(model_fn=model_fn, params=model_params)
```

- `model_fn`: 模型函數，它包含了 `training`, `evaluation`, `prediction`。這邊實作這些功能。
- `params`: hyper parameter dict (Ex: `learning_rate`, `dropout`)，它可以用來控制 `model_fn`。



'VALID' and 'SAME' padding

2x2 kernel, stride 2

VALID

1	2	3
4	5	6

<https://blog.csdn.net/wuzqChen>

SAME

1	2	3	0
4	5	6	0

<https://blog.csdn.net/wuzqChen>

這個是“賦值”

```
training=mode == tf.estimator.ModeKeys.TRAIN
```

這個是“等於”