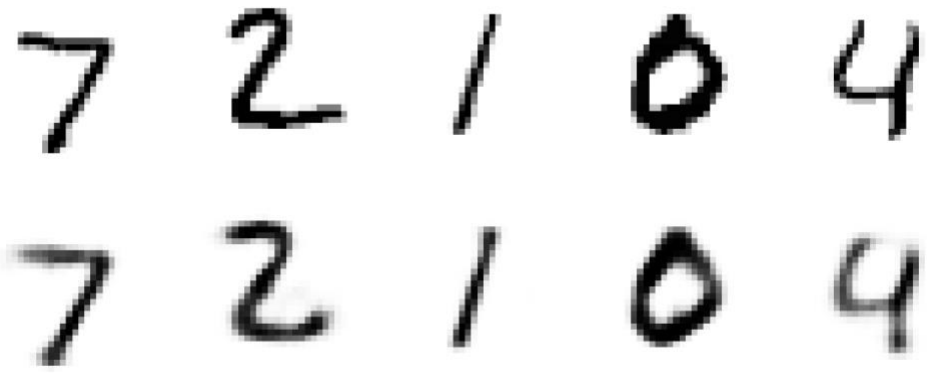
Loaded MNIST dataset of handwritten digits:

```
# import data
(X_train, y_train), (X_test, y_test) = mnist.load_data()

# scale data
X_train = X_train / 255
X_test = X_test / 255
```

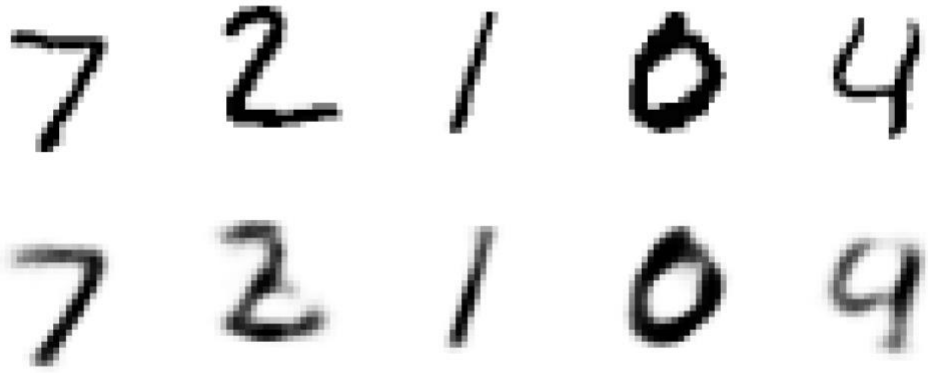
Fully connected feedforward undercomplete autoencoder built to reconstruct recognizable images:

```
# build fully connectd feedforward autoencoder
n_codings = 3

stacked_encoder = models.Sequential([
    layers.Flatten(input_shape=[28, 28]),
    layers.Dense(100, activation='selu'),
    layers.Dense(n_codings, activation='selu')
    ])

stacked_decoder = models.Sequential([
    layers.Dense(100, activation='selu', input_shape=[n_codings]),
    layers.Dense(28 * 28, activation='sigmoid'),
    layers.Reshape([28, 28])
    ])

stacked_ae = models.Sequential([stacked_encoder, stacked_decoder])
stacked_ae.compile(loss='binary_crossentropy', optimizer=SGD(lr=1.5))
history = stacked_ae.fit(X_train, X_train, epochs=10,
                         validation_data=(X_test, X_test))
```

Randomly selected test images (top) and corresponding reconstructions (bottom) using 30 codings:



Using 15 codings, digits remain recognizable, though the 4 starts to resemble a 9:



Even with just 9 codings, the digits appear quite similar to reconstructions using 15 encodings shown above.

3 codings produce results that are now clearly erroneous: