Artificial Intelligence Nanodegree

Recurrent Neural Network Projects

Welcome to the Recurrent Neural Network Project in the Artificial Intelligence Nanodegree! In this notebook, some template code has already been provided for you, and you will need to implement additional functionality to successfully complete this project. You will not need to modify the included code beyond what is requested. Sections that begin with **'Implementation'** in the header indicate that the following block of code will require additional functionality which you must provide. Instructions will be provided for each section and the specifics of the implementation are marked in the code block with a 'TODO' statement. Please be sure to read the instructions carefully!

In addition to implementing code, there will be questions that you must answer which relate to the project and your implementation. Each section where you will answer a question is preceded by a **'Question X'** header. Carefully read each question and provide thorough answers in the following text boxes that begin with **'Answer:'**. Your project submission will be evaluated based on your answers to each of the questions and the implementation you provide.

> **Note:** Code and Markdown cells can be executed using the **Shift + Enter** keyboard shortcut. In addition, Markdown cells can be edited by typically double-clicking the cell to enter edit mode.

Implementation TODOs in this notebook

This notebook contains two problems, cut into a variety of TODOs. Make sure to complete each section containing a TODO marker throughout the notebook. For convinence we provide links to each of these sections below.

TODO #1: Implement a function to window time series

TODO #2: Create a simple RNN model using keras to perform regression

TODO #3: Finish cleaning a large text corpus

TODO #4: Implement a function to window a large text corpus

TODO #5: Create a simple RNN model using keras to perform multiclass classification

TODO #6: Generate text using a fully trained RNN model and a variety of input sequences

Problem 1: Perform time series prediction

In this project you will perform time series prediction using a Recurrent Neural Network regressor. In particular you will re-create the figure shown in the notes - where the stock price of Apple was forecasted (or predicted) 7 days in advance. In completing this exercise you will learn how to construct RNNs using Keras, which will also aid in completing the second project in this notebook.

The particular network architecture we will employ for our RNN is known as Long Term Short Memory (LTSM) (https://en.wikipedia.org/wiki/Long_short-term_memory), which helps significantly avoid technical problems with optimization of RNNs.

1.1 Getting started

First we must load in our time series - a history of around 140 days of Apple's stock price. Then we need to perform a number of pre-processing steps to prepare it for use with an RNN model. First off, it is good practice to normalize time series - by normalizing its range. This helps us avoid serious numerical issues associated how common activation functions (like tanh) transform very large (positive or negative) numbers, as well as helping us to avoid related issues when computing derivatives.

Here we normalize the series to lie in the range [0,1] using this scikit function (http://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html), but it is also commonplace to normalize by a series standard deviation.

```
In [1]:  ### Load in necessary libraries for data input and normalization
         %matplotlib inline
         import numpy as np
         import matplotlib.pyplot as plt

         ### load in and normalize the dataset
         dataset = np.loadtxt('datasets/normalized_apple_prices.csv')
```

Lets take a quick look at the (normalized) time series we'll be performing predictions on.

```
In [2]:   # lets take a look at our time series
          plt.plot(dataset)
          plt.xlabel('time period')
          plt.ylabel('normalized series value')
```

Out[2]:   <matplotlib.text.Text at 0x7fbc589cf8d0>
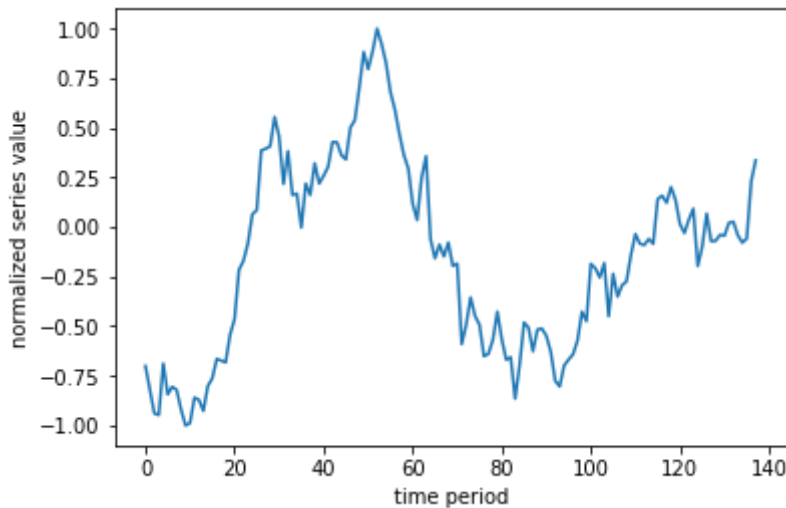
1.2 Cutting our time series into sequences

Remember, our time series is a sequence of numbers that we can represent in general mathematically as

$$s_0, s_1, s_2, \ldots, s_P$$

where s_p is the numerical value of the time series at time period p and where P is the total length of the series. In order to apply our RNN we treat the time series prediction problem as a regression problem, and so need to use a sliding window to construct a set of associated input/output pairs to regress on. This process is animated in the gif below.

For example - using a window of size T = 5 (as illustrated in the gif above) we produce a set of input/output pairs like the one shown in the table below

Input	Output
$\langle s_1, s_2, s_3, s_4, s_5 \rangle$	s_6
$\langle s_2, s_3, s_4, s_5, s_6 \rangle$	s_7
$\vdots$	$\vdots$
$\langle s_{P-5}, s_{P-4}, s_{P-3}, s_{P-2}, s_{P-1} \rangle$	s_P

Notice here that each input is a sequence (or vector) of length 4 (and in general has length equal to the window size T) while each corresponding output is a scalar value. Notice also how given a time series of length P and window size T = 5 as shown above, we created P - 5 input/output pairs. More generally, for a window size T we create P - T such pairs.

Now its time for you to window the input time series as described above!

TODO: Fill in the function below - called **window_transform_series** - that runs a sliding window along the input series and creates associated input/output pairs. Note that this function should input a) the series and b) the window length, and return the input/output subsequences. Make sure to format returned input/output as generally shown in table above (where window_size = 5), and make sure your returned input is a numpy array.

```
In [3]:  ### Fill out the function below that transforms the input series and win
         dow-size into a set of input/output pairs for use with our RNN model
         def window_transform_series(series,window_size):
             # containers for input/output pairs
             X = []
             y = []

             # Slice the series into windows
             for i in range(window_size, len(series)):
                 X.append(series[i-window_size:i])
                 y.append(series[i])

             # reshape each
             X = np.asarray(X)
             X.shape = (np.shape(X)[0:2])
             y = np.asarray(y)
             y.shape = (len(y),1)

             return X,y
```

You can test your function on the list of odd numbers given below

```
In [4]:  odd_nums = np.array([1,3,5,7,9,11,13])
```

To window this sequence with a window_size = 2 using the **window_transform_series** you should get the following input/output pairs

```
In [5]:  # run a window of size 2 over the odd number sequence and display the re
         sults
         window_size = 2
         X,y = window_transform_series(odd_nums,window_size)

         # print out input/output pairs --> here input = X, corresponding output
          = y
         print ('--- the input X will look like ----')
         print (X)

         print ('--- the associated output y will look like ----')
         print (y)

         print ('the shape of X is ' + str(np.shape(X)))
         print ('the shape of y is ' + str(np.shape(y)))
         print('the type of X is ' + str(type(X)))
         print('the type of y is ' + str(type(y)))
```

```
--- the input X will look like ----
[[ 1  3]
 [ 3  5]
 [ 5  7]
 [ 7  9]
 [ 9 11]]
--- the associated output y will look like ----
[[ 5]
 [ 7]
 [ 9]
 [11]
 [13]]
the shape of X is (5, 2)
the shape of y is (5, 1)
the type of X is <class 'numpy.ndarray'>
the type of y is <class 'numpy.ndarray'>
```

Again - you can check that your completed **window_transform_series** function works correctly by trying it on the odd_nums sequence - you should get the above output.

(remember to copy your completed function into the script *my_answers.py* function titled *window_transform_series* before submitting your project)

With this function in place apply it to the series in the Python cell below. We use a window_size = 7 for these experiments.

```
In [6]:  # window the data using your windowing function
         window_size = 7
         X,y = window_transform_series(series = dataset,window_size =
         window_size)
```

1.3 Splitting into training and testing sets

In order to perform proper testing on our dataset we will lop off the last 1/3 of it for validation (or testing). This is that once we train our model we have something to test it on (like any regression problem!). This splitting into training/testing sets is done in the cell below.

Note how here we are **not** splitting the dataset *randomly* as one typically would do when validating a regression model. This is because our input/output pairs *are related temporally*. We don't want to validate our model by training on a random subset of the series and then testing on another random subset, as this simulates the scenario that we receive new points *within the timeframe of our training set*.

We want to train on one solid chunk of the series (in our case, the first full 2/3 of it), and validate on a later chunk (the last 1/3) as this simulates how we would predict *future* values of a time series.

```
In [7]:  # split our dataset into training / testing sets
         train_test_split = int(np.ceil(2*len(y)/float(3)))   # set the split poi
         nt

         # partition the training set
         X_train = X[:train_test_split,:]
         y_train = y[:train_test_split]

         # keep the last chunk for testing
         X_test = X[train_test_split:,:]
         y_test = y[train_test_split:]

         # NOTE: to use keras's RNN LSTM module our input must be reshaped to [sa
         mples, window size, stepsize]
         X_train = np.asarray(np.reshape(X_train, (X_train.shape[0], window_size,
          1)))
         X_test = np.asarray(np.reshape(X_test, (X_test.shape[0], window_size,
         1)))
```

1.4 Build and run an RNN regression model

Having created input/output pairs out of our time series and cut this into training/testing sets, we can now begin setting up our RNN. We use Keras to quickly build a two hidden layer RNN of the following specifications

- layer 1 uses an LSTM module with 5 hidden units (note here the input_shape = (window_size,1))
- layer 2 uses a fully connected module with one unit
- the 'mean_squared_error' loss should be used (remember: we are performing regression here)

This can be constructed using just a few lines - see e.g., the general Keras documentation (https://keras.io/getting-started/sequential-model-guide/) and the LTSM documentation in particular (https://keras.io/layers/recurrent/) for examples of how to quickly use Keras to build neural network models. Make sure you are initializing your optimizer given the keras-recommended approach for RNNs (https://keras.io/optimizers/)

(given in the cell below). (remember to copy your completed function into the script *my_answers.py* function titled *build_part1_RNN* before submitting your project)

```
In [8]:  ### create required RNN model
         # import keras network libraries
         from keras.models import Sequential
         from keras.layers import Dense
         from keras.layers import LSTM
         import keras

         # given - fix random seed - so we can all reproduce the same results on
          our default time series
         np.random.seed(0)


         # build an RNN to perform regression on our time series input/output dat
         a
         model = Sequential()
         model.add(LSTM(5, input_shape=(window_size,1)))
         model.add(Dense(1))


         # build model using keras documentation recommended optimizer initializa
         tion
         optimizer = keras.optimizers.RMSprop(lr=0.001, rho=0.9, epsilon=1e-08, d
         ecay=0.0)

         # compile the model
         model.compile(loss='mean_squared_error', optimizer=optimizer)
```

Using TensorFlow backend.

With your model built you can now fit the model by activating the cell below! Note: the number of epochs (np_epochs) and batch_size are preset (so we can all produce the same results). You can choose to toggle the verbose parameter - which gives you regular updates on the progress of the algorithm - on and off by setting it to 1 or 0 respectively.

```
In [9]:  # run your model!
         model.fit(X_train, y_train, epochs=1000, batch_size=50, verbose=1)
```

```
Epoch 1/1000
88/88 [==============================] - 0s - loss: 0.1422
Epoch 2/1000
88/88 [==============================] - 0s - loss: 0.1343
Epoch 3/1000
88/88 [==============================] - 0s - loss: 0.1290
Epoch 4/1000
88/88 [==============================] - 0s - loss: 0.1245
Epoch 5/1000
88/88 [==============================] - 0s - loss: 0.1206
Epoch 6/1000
88/88 [==============================] - 0s - loss: 0.1170
Epoch 7/1000
88/88 [==============================] - 0s - loss: 0.1139
Epoch 8/1000
88/88 [==============================] - 0s - loss: 0.1106
Epoch 9/1000
88/88 [==============================] - 0s - loss: 0.1079
Epoch 10/1000
88/88 [==============================] - 0s - loss: 0.1051
Epoch 11/1000
88/88 [==============================] - 0s - loss: 0.1021
Epoch 12/1000
88/88 [==============================] - 0s - loss: 0.0994
Epoch 13/1000
88/88 [==============================] - 0s - loss: 0.0968
Epoch 14/1000
88/88 [==============================] - 0s - loss: 0.0944
Epoch 15/1000
88/88 [==============================] - 0s - loss: 0.0917
Epoch 16/1000
88/88 [==============================] - 0s - loss: 0.0893
Epoch 17/1000
88/88 [==============================] - 0s - loss: 0.0869
Epoch 18/1000
88/88 [==============================] - 0s - loss: 0.0846
Epoch 19/1000
88/88 [==============================] - 0s - loss: 0.0823
Epoch 20/1000
88/88 [==============================] - 0s - loss: 0.0803
Epoch 21/1000
88/88 [==============================] - 0s - loss: 0.0781
Epoch 22/1000
88/88 [==============================] - 0s - loss: 0.0763
Epoch 23/1000
88/88 [==============================] - 0s - loss: 0.0740
Epoch 24/1000
88/88 [==============================] - 0s - loss: 0.0721
Epoch 25/1000
88/88 [==============================] - 0s - loss: 0.0702
Epoch 26/1000
88/88 [==============================] - 0s - loss: 0.0685
Epoch 27/1000
88/88 [==============================] - 0s - loss: 0.0667
Epoch 28/1000
88/88 [==============================] - ETA: 0s - loss: 0.054 - 0s - l
oss: 0.0651
```

```
Epoch 29/1000
88/88 [==============================] - 0s - loss: 0.0635
Epoch 30/1000
88/88 [==============================] - 0s - loss: 0.0621
Epoch 31/1000
88/88 [==============================] - 0s - loss: 0.0608
Epoch 32/1000
88/88 [==============================] - 0s - loss: 0.0592
Epoch 33/1000
88/88 [==============================] - 0s - loss: 0.0580
Epoch 34/1000
88/88 [==============================] - 0s - loss: 0.0568
Epoch 35/1000
88/88 [==============================] - 0s - loss: 0.0557
Epoch 36/1000
88/88 [==============================] - 0s - loss: 0.0546
Epoch 37/1000
88/88 [==============================] - 0s - loss: 0.0536
Epoch 38/1000
88/88 [==============================] - 0s - loss: 0.0528
Epoch 39/1000
88/88 [==============================] - 0s - loss: 0.0519
Epoch 40/1000
88/88 [==============================] - 0s - loss: 0.0511
Epoch 41/1000
88/88 [==============================] - 0s - loss: 0.0504
Epoch 42/1000
88/88 [==============================] - 0s - loss: 0.0498
Epoch 43/1000
88/88 [==============================] - 0s - loss: 0.0491
Epoch 44/1000
88/88 [==============================] - 0s - loss: 0.0486
Epoch 45/1000
88/88 [==============================] - 0s - loss: 0.0481
Epoch 46/1000
88/88 [==============================] - 0s - loss: 0.0477
Epoch 47/1000
88/88 [==============================] - 0s - loss: 0.0473
Epoch 48/1000
88/88 [==============================] - 0s - loss: 0.0469
Epoch 49/1000
88/88 [==============================] - 0s - loss: 0.0466
Epoch 50/1000
88/88 [==============================] - 0s - loss: 0.0465
Epoch 51/1000
88/88 [==============================] - 0s - loss: 0.0462
Epoch 52/1000
88/88 [==============================] - 0s - loss: 0.0457
Epoch 53/1000
88/88 [==============================] - 0s - loss: 0.0456
Epoch 54/1000
88/88 [==============================] - 0s - loss: 0.0452
Epoch 55/1000
88/88 [==============================] - 0s - loss: 0.0451
Epoch 56/1000
88/88 [==============================] - 0s - loss: 0.0448
Epoch 57/1000
```

```
88/88 [==============================] - 0s - loss: 0.0447
Epoch 58/1000
88/88 [==============================] - 0s - loss: 0.0443
Epoch 59/1000
88/88 [==============================] - 0s - loss: 0.0443
Epoch 60/1000
88/88 [==============================] - 0s - loss: 0.0441
Epoch 61/1000
88/88 [==============================] - 0s - loss: 0.0438
Epoch 62/1000
88/88 [==============================] - 0s - loss: 0.0436
Epoch 63/1000
88/88 [==============================] - 0s - loss: 0.0434
Epoch 64/1000
88/88 [==============================] - 0s - loss: 0.0432
Epoch 65/1000
88/88 [==============================] - 0s - loss: 0.0429
Epoch 66/1000
88/88 [==============================] - 0s - loss: 0.0427
Epoch 67/1000
88/88 [==============================] - 0s - loss: 0.0426
Epoch 68/1000
88/88 [==============================] - 0s - loss: 0.0423
Epoch 69/1000
88/88 [==============================] - 0s - loss: 0.0421
Epoch 70/1000
88/88 [==============================] - 0s - loss: 0.0419
Epoch 71/1000
88/88 [==============================] - 0s - loss: 0.0417
Epoch 72/1000
88/88 [==============================] - 0s - loss: 0.0414
Epoch 73/1000
88/88 [==============================] - ETA: 0s - loss: 0.039 - 0s - l
oss: 0.0413
Epoch 74/1000
88/88 [==============================] - ETA: 0s - loss: 0.038 - 0s - l
oss: 0.0411
Epoch 75/1000
88/88 [==============================] - 0s - loss: 0.0408
Epoch 76/1000
88/88 [==============================] - 0s - loss: 0.0406
Epoch 77/1000
88/88 [==============================] - 0s - loss: 0.0404
Epoch 78/1000
88/88 [==============================] - 0s - loss: 0.0403
Epoch 79/1000
88/88 [==============================] - 0s - loss: 0.0400
Epoch 80/1000
88/88 [==============================] - 0s - loss: 0.0399
Epoch 81/1000
88/88 [==============================] - 0s - loss: 0.0396
Epoch 82/1000
88/88 [==============================] - 0s - loss: 0.0393
Epoch 83/1000
88/88 [==============================] - 0s - loss: 0.0392
Epoch 84/1000
88/88 [==============================] - 0s - loss: 0.0391
```

```
Epoch 85/1000
88/88 [==============================] - 0s - loss: 0.0389
Epoch 86/1000
88/88 [==============================] - 0s - loss: 0.0387
Epoch 87/1000
88/88 [==============================] - 0s - loss: 0.0384
Epoch 88/1000
88/88 [==============================] - 0s - loss: 0.0383
Epoch 89/1000
88/88 [==============================] - 0s - loss: 0.0380
Epoch 90/1000
88/88 [==============================] - 0s - loss: 0.0382
Epoch 91/1000
88/88 [==============================] - 0s - loss: 0.0376
Epoch 92/1000
88/88 [==============================] - ETA: 0s - loss: 0.027 - 0s - l
oss: 0.0376
Epoch 93/1000
88/88 [==============================] - 0s - loss: 0.0375
Epoch 94/1000
88/88 [==============================] - 0s - loss: 0.0372
Epoch 95/1000
88/88 [==============================] - 0s - loss: 0.0374
Epoch 96/1000
88/88 [==============================] - 0s - loss: 0.0370
Epoch 97/1000
88/88 [==============================] - 0s - loss: 0.0370
Epoch 98/1000
88/88 [==============================] - 0s - loss: 0.0367
Epoch 99/1000
88/88 [==============================] - 0s - loss: 0.0367
Epoch 100/1000
88/88 [==============================] - 0s - loss: 0.0364
Epoch 101/1000
88/88 [==============================] - 0s - loss: 0.0363
Epoch 102/1000
88/88 [==============================] - 0s - loss: 0.0361
Epoch 103/1000
88/88 [==============================] - 0s - loss: 0.0360
Epoch 104/1000
88/88 [==============================] - 0s - loss: 0.0358
Epoch 105/1000
88/88 [==============================] - 0s - loss: 0.0357
Epoch 106/1000
88/88 [==============================] - 0s - loss: 0.0355
Epoch 107/1000
88/88 [==============================] - 0s - loss: 0.0354
Epoch 108/1000
88/88 [==============================] - 0s - loss: 0.0352
Epoch 109/1000
88/88 [==============================] - 0s - loss: 0.0350
Epoch 110/1000
88/88 [==============================] - 0s - loss: 0.0350
Epoch 111/1000
88/88 [==============================] - 0s - loss: 0.0346
Epoch 112/1000
88/88 [==============================] - 0s - loss: 0.0345
```

```
Epoch 113/1000
88/88 [==============================] - 0s - loss: 0.0343
Epoch 114/1000
88/88 [==============================] - 0s - loss: 0.0341
Epoch 115/1000
88/88 [==============================] - 0s - loss: 0.0342
Epoch 116/1000
88/88 [==============================] - 0s - loss: 0.0339
Epoch 117/1000
88/88 [==============================] - 0s - loss: 0.0338
Epoch 118/1000
88/88 [==============================] - 0s - loss: 0.0336
Epoch 119/1000
88/88 [==============================] - 0s - loss: 0.0336
Epoch 120/1000
88/88 [==============================] - 0s - loss: 0.0332
Epoch 121/1000
88/88 [==============================] - 0s - loss: 0.0332
Epoch 122/1000
88/88 [==============================] - 0s - loss: 0.0329
Epoch 123/1000
88/88 [==============================] - 0s - loss: 0.0328
Epoch 124/1000
88/88 [==============================] - 0s - loss: 0.0326
Epoch 125/1000
88/88 [==============================] - 0s - loss: 0.0325
Epoch 126/1000
88/88 [==============================] - 0s - loss: 0.0323
Epoch 127/1000
88/88 [==============================] - 0s - loss: 0.0322
Epoch 128/1000
88/88 [==============================] - 0s - loss: 0.0320
Epoch 129/1000
88/88 [==============================] - 0s - loss: 0.0320
Epoch 130/1000
88/88 [==============================] - 0s - loss: 0.0318
Epoch 131/1000
88/88 [==============================] - 0s - loss: 0.0315
Epoch 132/1000
88/88 [==============================] - 0s - loss: 0.0313
Epoch 133/1000
88/88 [==============================] - 0s - loss: 0.0314
Epoch 134/1000
88/88 [==============================] - 0s - loss: 0.0315
Epoch 135/1000
88/88 [==============================] - 0s - loss: 0.0309
Epoch 136/1000
88/88 [==============================] - 0s - loss: 0.0308
Epoch 137/1000
88/88 [==============================] - 0s - loss: 0.0308
Epoch 138/1000
88/88 [==============================] - 0s - loss: 0.0306
Epoch 139/1000
88/88 [==============================] - 0s - loss: 0.0304
Epoch 140/1000
88/88 [==============================] - 0s - loss: 0.0303
Epoch 141/1000
```

```
88/88 [==============================] - 0s - loss: 0.0301
Epoch 142/1000
88/88 [==============================] - 0s - loss: 0.0301
Epoch 143/1000
88/88 [==============================] - ETA: 0s - loss: 0.031 - 0s - l
oss: 0.0299
Epoch 144/1000
88/88 [==============================] - 0s - loss: 0.0297
Epoch 145/1000
88/88 [==============================] - 0s - loss: 0.0295
Epoch 146/1000
88/88 [==============================] - 0s - loss: 0.0294
Epoch 147/1000
88/88 [==============================] - 0s - loss: 0.0295
Epoch 148/1000
88/88 [==============================] - 0s - loss: 0.0291
Epoch 149/1000
88/88 [==============================] - 0s - loss: 0.0290
Epoch 150/1000
88/88 [==============================] - 0s - loss: 0.0289
Epoch 151/1000
88/88 [==============================] - 0s - loss: 0.0287
Epoch 152/1000
88/88 [==============================] - 0s - loss: 0.0286
Epoch 153/1000
88/88 [==============================] - 0s - loss: 0.0287
Epoch 154/1000
88/88 [==============================] - 0s - loss: 0.0284
Epoch 155/1000
88/88 [==============================] - 0s - loss: 0.0282
Epoch 156/1000
88/88 [==============================] - 0s - loss: 0.0280
Epoch 157/1000
88/88 [==============================] - 0s - loss: 0.0280
Epoch 158/1000
88/88 [==============================] - 0s - loss: 0.0278
Epoch 159/1000
88/88 [==============================] - 0s - loss: 0.0276
Epoch 160/1000
88/88 [==============================] - 0s - loss: 0.0277
Epoch 161/1000
88/88 [==============================] - 0s - loss: 0.0275
Epoch 162/1000
88/88 [==============================] - 0s - loss: 0.0275
Epoch 163/1000
88/88 [==============================] - 0s - loss: 0.0272
Epoch 164/1000
88/88 [==============================] - 0s - loss: 0.0270
Epoch 165/1000
88/88 [==============================] - 0s - loss: 0.0270
Epoch 166/1000
88/88 [==============================] - 0s - loss: 0.0268
Epoch 167/1000
88/88 [==============================] - 0s - loss: 0.0268
Epoch 168/1000
88/88 [==============================] - 0s - loss: 0.0266
Epoch 169/1000
```

```
88/88 [==============================] - 0s - loss: 0.0265
Epoch 170/1000
88/88 [==============================] - 0s - loss: 0.0265
Epoch 171/1000
88/88 [==============================] - 0s - loss: 0.0263
Epoch 172/1000
88/88 [==============================] - 0s - loss: 0.0264
Epoch 173/1000
88/88 [==============================] - 0s - loss: 0.0262
Epoch 174/1000
88/88 [==============================] - 0s - loss: 0.0259
Epoch 175/1000
88/88 [==============================] - 0s - loss: 0.0259
Epoch 176/1000
88/88 [==============================] - 0s - loss: 0.0258
Epoch 177/1000
88/88 [==============================] - 0s - loss: 0.0256
Epoch 178/1000
88/88 [==============================] - 0s - loss: 0.0256
Epoch 179/1000
88/88 [==============================] - 0s - loss: 0.0254
Epoch 180/1000
88/88 [==============================] - 0s - loss: 0.0254
Epoch 181/1000
88/88 [==============================] - 0s - loss: 0.0253
Epoch 182/1000
88/88 [==============================] - 0s - loss: 0.0251
Epoch 183/1000
88/88 [==============================] - 0s - loss: 0.0252
Epoch 184/1000
88/88 [==============================] - 0s - loss: 0.0249
Epoch 185/1000
88/88 [==============================] - 0s - loss: 0.0248
Epoch 186/1000
88/88 [==============================] - 0s - loss: 0.0248
Epoch 187/1000
88/88 [==============================] - 0s - loss: 0.0245
Epoch 188/1000
88/88 [==============================] - 0s - loss: 0.0244
Epoch 189/1000
88/88 [==============================] - 0s - loss: 0.0244
Epoch 190/1000
88/88 [==============================] - 0s - loss: 0.0242
Epoch 191/1000
88/88 [==============================] - 0s - loss: 0.0241
Epoch 192/1000
88/88 [==============================] - 0s - loss: 0.0240
Epoch 193/1000
88/88 [==============================] - 0s - loss: 0.0239
Epoch 194/1000
88/88 [==============================] - 0s - loss: 0.0241
Epoch 195/1000
88/88 [==============================] - 0s - loss: 0.0238
Epoch 196/1000
88/88 [==============================] - 0s - loss: 0.0236
Epoch 197/1000
88/88 [==============================] - 0s - loss: 0.0235
```

```
Epoch 198/1000
88/88 [==============================] - 0s - loss: 0.0234
Epoch 199/1000
88/88 [==============================] - 0s - loss: 0.0233
Epoch 200/1000
88/88 [==============================] - 0s - loss: 0.0231
Epoch 201/1000
88/88 [==============================] - 0s - loss: 0.0231
Epoch 202/1000
88/88 [==============================] - 0s - loss: 0.0229
Epoch 203/1000
88/88 [==============================] - 0s - loss: 0.0229
Epoch 204/1000
88/88 [==============================] - 0s - loss: 0.0228
Epoch 205/1000
88/88 [==============================] - 0s - loss: 0.0228
Epoch 206/1000
88/88 [==============================] - 0s - loss: 0.0226
Epoch 207/1000
88/88 [==============================] - 0s - loss: 0.0224
Epoch 208/1000
88/88 [==============================] - 0s - loss: 0.0223
Epoch 209/1000
88/88 [==============================] - 0s - loss: 0.0222
Epoch 210/1000
88/88 [==============================] - 0s - loss: 0.0223
Epoch 211/1000
88/88 [==============================] - 0s - loss: 0.0221
Epoch 212/1000
88/88 [==============================] - 0s - loss: 0.0219
Epoch 213/1000
88/88 [==============================] - 0s - loss: 0.0220
Epoch 214/1000
88/88 [==============================] - 0s - loss: 0.0217
Epoch 215/1000
88/88 [==============================] - 0s - loss: 0.0219
Epoch 216/1000
88/88 [==============================] - 0s - loss: 0.0216
Epoch 217/1000
88/88 [==============================] - 0s - loss: 0.0216
Epoch 218/1000
88/88 [==============================] - 0s - loss: 0.0217
Epoch 219/1000
88/88 [==============================] - 0s - loss: 0.0215
Epoch 220/1000
88/88 [==============================] - 0s - loss: 0.0213
Epoch 221/1000
88/88 [==============================] - 0s - loss: 0.0213
Epoch 222/1000
88/88 [==============================] - 0s - loss: 0.0211
Epoch 223/1000
88/88 [==============================] - 0s - loss: 0.0211
Epoch 224/1000
88/88 [==============================] - 0s - loss: 0.0211
Epoch 225/1000
88/88 [==============================] - 0s - loss: 0.0209
Epoch 226/1000
```

```
88/88 [==============================] - ETA: 0s - loss: 0.020 - 0s - l
oss: 0.0212
Epoch 227/1000
88/88 [==============================] - 0s - loss: 0.0208
Epoch 228/1000
88/88 [==============================] - 0s - loss: 0.0207
Epoch 229/1000
88/88 [==============================] - 0s - loss: 0.0207
Epoch 230/1000
88/88 [==============================] - 0s - loss: 0.0206
Epoch 231/1000
88/88 [==============================] - 0s - loss: 0.0205
Epoch 232/1000
88/88 [==============================] - 0s - loss: 0.0205
Epoch 233/1000
88/88 [==============================] - 0s - loss: 0.0203
Epoch 234/1000
88/88 [==============================] - 0s - loss: 0.0206
Epoch 235/1000
88/88 [==============================] - 0s - loss: 0.0202
Epoch 236/1000
88/88 [==============================] - 0s - loss: 0.0201
Epoch 237/1000
88/88 [==============================] - 0s - loss: 0.0201
Epoch 238/1000
88/88 [==============================] - 0s - loss: 0.0200
Epoch 239/1000
88/88 [==============================] - 0s - loss: 0.0199
Epoch 240/1000
88/88 [==============================] - 0s - loss: 0.0202
Epoch 241/1000
88/88 [==============================] - 0s - loss: 0.0198
Epoch 242/1000
88/88 [==============================] - 0s - loss: 0.0197
Epoch 243/1000
88/88 [==============================] - 0s - loss: 0.0198
Epoch 244/1000
88/88 [==============================] - 0s - loss: 0.0196
Epoch 245/1000
88/88 [==============================] - 0s - loss: 0.0197
Epoch 246/1000
88/88 [==============================] - 0s - loss: 0.0196
Epoch 247/1000
88/88 [==============================] - 0s - loss: 0.0195
Epoch 248/1000
88/88 [==============================] - 0s - loss: 0.0196
Epoch 249/1000
88/88 [==============================] - 0s - loss: 0.0194
Epoch 250/1000
88/88 [==============================] - 0s - loss: 0.0196
Epoch 251/1000
88/88 [==============================] - 0s - loss: 0.0192
Epoch 252/1000
88/88 [==============================] - 0s - loss: 0.0192
Epoch 253/1000
88/88 [==============================] - 0s - loss: 0.0192
Epoch 254/1000
```

```
88/88 [==============================] - 0s - loss: 0.0191
Epoch 255/1000
88/88 [==============================] - 0s - loss: 0.0193
Epoch 256/1000
88/88 [==============================] - 0s - loss: 0.0190
Epoch 257/1000
88/88 [==============================] - 0s - loss: 0.0190
Epoch 258/1000
88/88 [==============================] - 0s - loss: 0.0190
Epoch 259/1000
88/88 [==============================] - 0s - loss: 0.0189
Epoch 260/1000
88/88 [==============================] - 0s - loss: 0.0188
Epoch 261/1000
88/88 [==============================] - 0s - loss: 0.0191
Epoch 262/1000
88/88 [==============================] - 0s - loss: 0.0190
Epoch 263/1000
88/88 [==============================] - 0s - loss: 0.0187
Epoch 264/1000
88/88 [==============================] - 0s - loss: 0.0187
Epoch 265/1000
88/88 [==============================] - 0s - loss: 0.0186
Epoch 266/1000
88/88 [==============================] - 0s - loss: 0.0188
Epoch 267/1000
88/88 [==============================] - 0s - loss: 0.0186
Epoch 268/1000
88/88 [==============================] - 0s - loss: 0.0188
Epoch 269/1000
88/88 [==============================] - 0s - loss: 0.0186
Epoch 270/1000
88/88 [==============================] - ETA: 0s - loss: 0.015 - 0s - l
oss: 0.0184
Epoch 271/1000
88/88 [==============================] - 0s - loss: 0.0187
Epoch 272/1000
88/88 [==============================] - 0s - loss: 0.0184
Epoch 273/1000
88/88 [==============================] - 0s - loss: 0.0184
Epoch 274/1000
88/88 [==============================] - 0s - loss: 0.0183
Epoch 275/1000
88/88 [==============================] - 0s - loss: 0.0183
Epoch 276/1000
88/88 [==============================] - 0s - loss: 0.0183
Epoch 277/1000
88/88 [==============================] - 0s - loss: 0.0186
Epoch 278/1000
88/88 [==============================] - 0s - loss: 0.0182
Epoch 279/1000
88/88 [==============================] - 0s - loss: 0.0183
Epoch 280/1000
88/88 [==============================] - 0s - loss: 0.0181
Epoch 281/1000
88/88 [==============================] - 0s - loss: 0.0181
Epoch 282/1000
```

```
88/88 [==============================] - 0s - loss: 0.0183
Epoch 283/1000
88/88 [==============================] - 0s - loss: 0.0183
Epoch 284/1000
88/88 [==============================] - 0s - loss: 0.0181
Epoch 285/1000
88/88 [==============================] - 0s - loss: 0.0180
Epoch 286/1000
88/88 [==============================] - 0s - loss: 0.0182
Epoch 287/1000
88/88 [==============================] - 0s - loss: 0.0180
Epoch 288/1000
88/88 [==============================] - 0s - loss: 0.0180
Epoch 289/1000
88/88 [==============================] - 0s - loss: 0.0180
Epoch 290/1000
88/88 [==============================] - 0s - loss: 0.0180
Epoch 291/1000
88/88 [==============================] - 0s - loss: 0.0180
Epoch 292/1000
88/88 [==============================] - 0s - loss: 0.0179
Epoch 293/1000
88/88 [==============================] - 0s - loss: 0.0179
Epoch 294/1000
88/88 [==============================] - 0s - loss: 0.0179
Epoch 295/1000
88/88 [==============================] - 0s - loss: 0.0179
Epoch 296/1000
88/88 [==============================] - 0s - loss: 0.0180
Epoch 297/1000
88/88 [==============================] - 0s - loss: 0.0179
Epoch 298/1000
88/88 [==============================] - 0s - loss: 0.0179
Epoch 299/1000
88/88 [==============================] - 0s - loss: 0.0177
Epoch 300/1000
88/88 [==============================] - 0s - loss: 0.0177
Epoch 301/1000
88/88 [==============================] - 0s - loss: 0.0178
Epoch 302/1000
88/88 [==============================] - 0s - loss: 0.0177
Epoch 303/1000
88/88 [==============================] - 0s - loss: 0.0179
Epoch 304/1000
88/88 [==============================] - 0s - loss: 0.0176
Epoch 305/1000
88/88 [==============================] - 0s - loss: 0.0177
Epoch 306/1000
88/88 [==============================] - 0s - loss: 0.0178
Epoch 307/1000
88/88 [==============================] - 0s - loss: 0.0178
Epoch 308/1000
88/88 [==============================] - 0s - loss: 0.0176
Epoch 309/1000
88/88 [==============================] - 0s - loss: 0.0176
Epoch 310/1000
88/88 [==============================] - 0s - loss: 0.0178
```

```
Epoch 311/1000
88/88 [==============================] - ETA: 0s - loss: 0.015 - 0s - l
oss: 0.0175
Epoch 312/1000
88/88 [==============================] - ETA: 0s - loss: 0.014 - 0s - l
oss: 0.0176
Epoch 313/1000
88/88 [==============================] - 0s - loss: 0.0176
Epoch 314/1000
88/88 [==============================] - 0s - loss: 0.0175
Epoch 315/1000
88/88 [==============================] - ETA: 0s - loss: 0.013 - 0s - l
oss: 0.0176
Epoch 316/1000
88/88 [==============================] - 0s - loss: 0.0175
Epoch 317/1000
88/88 [==============================] - 0s - loss: 0.0178
Epoch 318/1000
88/88 [==============================] - 0s - loss: 0.0175
Epoch 319/1000
88/88 [==============================] - 0s - loss: 0.0175
Epoch 320/1000
88/88 [==============================] - 0s - loss: 0.0177
Epoch 321/1000
88/88 [==============================] - 0s - loss: 0.0177
Epoch 322/1000
88/88 [==============================] - 0s - loss: 0.0175
Epoch 323/1000
88/88 [==============================] - 0s - loss: 0.0175
Epoch 324/1000
88/88 [==============================] - 0s - loss: 0.0174
Epoch 325/1000
88/88 [==============================] - ETA: 0s - loss: 0.015 - 0s - l
oss: 0.0179
Epoch 326/1000
88/88 [==============================] - 0s - loss: 0.0174
Epoch 327/1000
88/88 [==============================] - 0s - loss: 0.0174
Epoch 328/1000
88/88 [==============================] - 0s - loss: 0.0175
Epoch 329/1000
88/88 [==============================] - 0s - loss: 0.0175
Epoch 330/1000
88/88 [==============================] - 0s - loss: 0.0176
Epoch 331/1000
88/88 [==============================] - 0s - loss: 0.0174
Epoch 332/1000
88/88 [==============================] - 0s - loss: 0.0174
Epoch 333/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 334/1000
88/88 [==============================] - 0s - loss: 0.0174
Epoch 335/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 336/1000
88/88 [==============================] - 0s - loss: 0.0175
Epoch 337/1000
```

```
88/88 [==============================] - 0s - loss: 0.0176
Epoch 338/1000
88/88 [==============================] - 0s - loss: 0.0174
Epoch 339/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 340/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 341/1000
88/88 [==============================] - 0s - loss: 0.0175
Epoch 342/1000
88/88 [==============================] - 0s - loss: 0.0174
Epoch 343/1000
88/88 [==============================] - 0s - loss: 0.0172
Epoch 344/1000
88/88 [==============================] - ETA: 0s - loss: 0.014 - 0s - l
oss: 0.0172
Epoch 345/1000
88/88 [==============================] - 0s - loss: 0.0176
Epoch 346/1000
88/88 [==============================] - 0s - loss: 0.0174
Epoch 347/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 348/1000
88/88 [==============================] - 0s - loss: 0.0172
Epoch 349/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 350/1000
88/88 [==============================] - 0s - loss: 0.0172
Epoch 351/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 352/1000
88/88 [==============================] - 0s - loss: 0.0172
Epoch 353/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 354/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 355/1000
88/88 [==============================] - 0s - loss: 0.0172
Epoch 356/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 357/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 358/1000
88/88 [==============================] - 0s - loss: 0.0172
Epoch 359/1000
88/88 [==============================] - 0s - loss: 0.0175
Epoch 360/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 361/1000
88/88 [==============================] - 0s - loss: 0.0172
Epoch 362/1000
88/88 [==============================] - 0s - loss: 0.0174
Epoch 363/1000
88/88 [==============================] - 0s - loss: 0.0172
Epoch 364/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 365/1000
```

```
88/88 [==============================] - 0s - loss: 0.0172
Epoch 366/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 367/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 368/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 369/1000
88/88 [==============================] - ETA: 0s - loss: 0.019 - 0s - l
oss: 0.0171
Epoch 370/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 371/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 372/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 373/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 374/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 375/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 376/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 377/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 378/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 379/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 380/1000
88/88 [==============================] - 0s - loss: 0.0172
Epoch 381/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 382/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 383/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 384/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 385/1000
88/88 [==============================] - 0s - loss: 0.0172
Epoch 386/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 387/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 388/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 389/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 390/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 391/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 392/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 393/1000
```

```
88/88 [==============================] - 0s - loss: 0.0170
Epoch 394/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 395/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 396/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 397/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 398/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 399/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 400/1000
88/88 [==============================] - 0s - loss: 0.0172
Epoch 401/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 402/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 403/1000
88/88 [==============================] - 0s - loss: 0.0175
Epoch 404/1000
88/88 [==============================] - ETA: 0s - loss: 0.020 - 0s - l
oss: 0.0170
Epoch 405/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 406/1000
88/88 [==============================] - ETA: 0s - loss: 0.017 - 0s - l
oss: 0.0169
Epoch 407/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 408/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 409/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 410/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 411/1000
88/88 [==============================] - 0s - loss: 0.0175
Epoch 412/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 413/1000
88/88 [==============================] - 0s - loss: 0.0172
Epoch 414/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 415/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 416/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 417/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 418/1000
88/88 [==============================] - 0s - loss: 0.0173
Epoch 419/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 420/1000
88/88 [==============================] - 0s - loss: 0.0169
```

```
Epoch 421/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 422/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 423/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 424/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 425/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 426/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 427/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 428/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 429/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 430/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 431/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 432/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 433/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 434/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 435/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 436/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 437/1000
88/88 [==============================] - 0s - loss: 0.0176
Epoch 438/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 439/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 440/1000
88/88 [==============================] - ETA: 0s - loss: 0.012 - 0s - l
oss: 0.0171
Epoch 441/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 442/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 443/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 444/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 445/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 446/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 447/1000
88/88 [==============================] - 0s - loss: 0.0172
Epoch 448/1000
88/88 [==============================] - 0s - loss: 0.0169
```

```
Epoch 449/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 450/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 451/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 452/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 453/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 454/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 455/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 456/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 457/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 458/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 459/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 460/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 461/1000
88/88 [==============================] - 0s - loss: 0.0172
Epoch 462/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 463/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 464/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 465/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 466/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 467/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 468/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 469/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 470/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 471/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 472/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 473/1000
88/88 [==============================] - 0s - loss: 0.0172
Epoch 474/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 475/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 476/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 477/1000
```

```
88/88 [==============================] - 0s - loss: 0.0167
Epoch 478/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 479/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 480/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 481/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 482/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 483/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 484/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 485/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 486/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 487/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 488/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 489/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 490/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 491/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 492/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 493/1000
88/88 [==============================] - 0s - loss: 0.0172
Epoch 494/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 495/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 496/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 497/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 498/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 499/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 500/1000
88/88 [==============================] - ETA: 0s - loss: 0.019 - 0s - l
oss: 0.0167
Epoch 501/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 502/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 503/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 504/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 505/1000
```

```
88/88 [==============================] - 0s - loss: 0.0168
Epoch 506/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 507/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 508/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 509/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 510/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 511/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 512/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 513/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 514/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 515/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 516/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 517/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 518/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 519/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 520/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 521/1000
88/88 [==============================] - 0s - loss: 0.0170
Epoch 522/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 523/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 524/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 525/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 526/1000
88/88 [==============================] - ETA: 0s - loss: 0.012 - 0s - l
oss: 0.0167
Epoch 527/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 528/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 529/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 530/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 531/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 532/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 533/1000
```

```
88/88 [==============================] - 0s - loss: 0.0170
Epoch 534/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 535/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 536/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 537/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 538/1000
88/88 [==============================] - 0s - loss: 0.0171
Epoch 539/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 540/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 541/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 542/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 543/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 544/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 545/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 546/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 547/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 548/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 549/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 550/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 551/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 552/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 553/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 554/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 555/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 556/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 557/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 558/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 559/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 560/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 561/1000
88/88 [==============================] - 0s - loss: 0.0166
```

```
Epoch 562/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 563/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 564/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 565/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 566/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 567/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 568/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 569/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 570/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 571/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 572/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 573/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 574/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 575/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 576/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 577/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 578/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 579/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 580/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 581/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 582/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 583/1000
88/88 [==============================] - ETA: 0s - loss: 0.019 - 0s - l
oss: 0.0166
Epoch 584/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 585/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 586/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 587/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 588/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 589/1000
88/88 [==============================] - 0s - loss: 0.0165
```

```
Epoch 590/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 591/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 592/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 593/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 594/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 595/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 596/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 597/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 598/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 599/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 600/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 601/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 602/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 603/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 604/1000
88/88 [==============================] - ETA: 0s - loss: 0.018 - 0s - l
oss: 0.0166
Epoch 605/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 606/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 607/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 608/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 609/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 610/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 611/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 612/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 613/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 614/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 615/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 616/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 617/1000
88/88 [==============================] - 0s - loss: 0.0167
```

```
Epoch 618/1000
88/88 [==============================] - ETA: 0s - loss: 0.013 - 0s - l
oss: 0.0164
Epoch 619/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 620/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 621/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 622/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 623/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 624/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 625/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 626/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 627/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 628/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 629/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 630/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 631/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 632/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 633/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 634/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 635/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 636/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 637/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 638/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 639/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 640/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 641/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 642/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 643/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 644/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 645/1000
88/88 [==============================] - 0s - loss: 0.0164
```

```
Epoch 646/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 647/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 648/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 649/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 650/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 651/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 652/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 653/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 654/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 655/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 656/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 657/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 658/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 659/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 660/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 661/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 662/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 663/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 664/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 665/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 666/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 667/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 668/1000
88/88 [==============================] - ETA: 0s - loss: 0.017 - 0s - l
oss: 0.0165
Epoch 669/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 670/1000
88/88 [==============================] - ETA: 0s - loss: 0.016 - 0s - l
oss: 0.0165
Epoch 671/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 672/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 673/1000
```

```
88/88 [==============================] - 0s - loss: 0.0164
Epoch 674/1000
88/88 [==============================] - ETA: 0s - loss: 0.019 - 0s - l
oss: 0.0164
Epoch 675/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 676/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 677/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 678/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 679/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 680/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 681/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 682/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 683/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 684/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 685/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 686/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 687/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 688/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 689/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 690/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 691/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 692/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 693/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 694/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 695/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 696/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 697/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 698/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 699/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 700/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 701/1000
```

```
88/88 [==============================] - 0s - loss: 0.0167
Epoch 702/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 703/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 704/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 705/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 706/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 707/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 708/1000
88/88 [==============================] - ETA: 0s - loss: 0.018 - 0s - l
oss: 0.0163
Epoch 709/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 710/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 711/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 712/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 713/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 714/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 715/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 716/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 717/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 718/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 719/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 720/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 721/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 722/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 723/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 724/1000
88/88 [==============================] - 0s - loss: 0.0169
Epoch 725/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 726/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 727/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 728/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 729/1000
```

```
88/88 [==============================] - 0s - loss: 0.0164
Epoch 730/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 731/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 732/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 733/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 734/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 735/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 736/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 737/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 738/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 739/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 740/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 741/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 742/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 743/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 744/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 745/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 746/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 747/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 748/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 749/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 750/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 751/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 752/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 753/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 754/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 755/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 756/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 757/1000
88/88 [==============================] - 0s - loss: 0.0163
```

```
Epoch 758/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 759/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 760/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 761/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 762/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 763/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 764/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 765/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 766/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 767/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 768/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 769/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 770/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 771/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 772/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 773/1000
88/88 [==============================] - ETA: 0s - loss: 0.017 - 0s - l
oss: 0.0167
Epoch 774/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 775/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 776/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 777/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 778/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 779/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 780/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 781/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 782/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 783/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 784/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 785/1000
88/88 [==============================] - 0s - loss: 0.0166
```

```
Epoch 786/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 787/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 788/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 789/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 790/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 791/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 792/1000
88/88 [==============================] - ETA: 0s - loss: 0.019 - 0s - l
oss: 0.0164
Epoch 793/1000
88/88 [==============================] - ETA: 0s - loss: 0.015 - 0s - l
oss: 0.0164
Epoch 794/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 795/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 796/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 797/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 798/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 799/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 800/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 801/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 802/1000
88/88 [==============================] - ETA: 0s - loss: 0.014 - 0s - l
oss: 0.0162
Epoch 803/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 804/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 805/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 806/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 807/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 808/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 809/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 810/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 811/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 812/1000
88/88 [==============================] - 0s - loss: 0.0162
```

```
Epoch 813/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 814/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 815/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 816/1000
88/88 [==============================] - 0s - loss: 0.0167
Epoch 817/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 818/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 819/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 820/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 821/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 822/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 823/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 824/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 825/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 826/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 827/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 828/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 829/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 830/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 831/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 832/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 833/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 834/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 835/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 836/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 837/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 838/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 839/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 840/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 841/1000
```

```
88/88 [==============================] - 0s - loss: 0.0162
Epoch 842/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 843/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 844/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 845/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 846/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 847/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 848/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 849/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 850/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 851/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 852/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 853/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 854/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 855/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 856/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 857/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 858/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 859/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 860/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 861/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 862/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 863/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 864/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 865/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 866/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 867/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 868/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 869/1000
88/88 [==============================] - 0s - loss: 0.0163
```

```
Epoch 870/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 871/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 872/1000
88/88 [==============================] - ETA: 0s - loss: 0.019 - 0s - l
oss: 0.0162
Epoch 873/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 874/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 875/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 876/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 877/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 878/1000
88/88 [==============================] - 0s - loss: 0.0168
Epoch 879/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 880/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 881/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 882/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 883/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 884/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 885/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 886/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 887/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 888/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 889/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 890/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 891/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 892/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 893/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 894/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 895/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 896/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 897/1000
88/88 [==============================] - 0s - loss: 0.0163
```

```
Epoch 898/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 899/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 900/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 901/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 902/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 903/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 904/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 905/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 906/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 907/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 908/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 909/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 910/1000
88/88 [==============================] - ETA: 0s - loss: 0.016 - 0s - l
oss: 0.0162
Epoch 911/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 912/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 913/1000
88/88 [==============================] - ETA: 0s - loss: 0.020 - 0s - l
oss: 0.0161
Epoch 914/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 915/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 916/1000
88/88 [==============================] - ETA: 0s - loss: 0.020 - 0s - l
oss: 0.0162
Epoch 917/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 918/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 919/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 920/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 921/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 922/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 923/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 924/1000
88/88 [==============================] - 0s - loss: 0.0161
```

```
Epoch 925/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 926/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 927/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 928/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 929/1000
88/88 [==============================] - 0s - loss: 0.0165
Epoch 930/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 931/1000
88/88 [==============================] - 0s - loss: 0.0160
Epoch 932/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 933/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 934/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 935/1000
88/88 [==============================] - 0s - loss: 0.0160
Epoch 936/1000
88/88 [==============================] - 0s - loss: 0.0164
Epoch 937/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 938/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 939/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 940/1000
88/88 [==============================] - 0s - loss: 0.0166
Epoch 941/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 942/1000
88/88 [==============================] - 0s - loss: 0.0160
Epoch 943/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 944/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 945/1000
88/88 [==============================] - 0s - loss: 0.0160
Epoch 946/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 947/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 948/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 949/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 950/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 951/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 952/1000
88/88 [==============================] - 0s - loss: 0.0160
Epoch 953/1000
```

```
88/88 [==============================] - 0s - loss: 0.0164
Epoch 954/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 955/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 956/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 957/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 958/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 959/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 960/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 961/1000
88/88 [==============================] - ETA: 0s - loss: 0.017 - 0s - l
oss: 0.0160
Epoch 962/1000
88/88 [==============================] - 0s - loss: 0.0160
Epoch 963/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 964/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 965/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 966/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 967/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 968/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 969/1000
88/88 [==============================] - 0s - loss: 0.0160
Epoch 970/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 971/1000
88/88 [==============================] - 0s - loss: 0.0160
Epoch 972/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 973/1000
88/88 [==============================] - ETA: 0s - loss: 0.014 - 0s - l
oss: 0.0160
Epoch 974/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 975/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 976/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 977/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 978/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 979/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 980/1000
88/88 [==============================] - 0s - loss: 0.0166
```

```
Epoch 981/1000
88/88 [==============================] - 0s - loss: 0.0163
Epoch 982/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 983/1000
88/88 [==============================] - 0s - loss: 0.0160
Epoch 984/1000
88/88 [==============================] - 0s - loss: 0.0160
Epoch 985/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 986/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 987/1000
88/88 [==============================] - 0s - loss: 0.0160
Epoch 988/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 989/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 990/1000
88/88 [==============================] - 0s - loss: 0.0160
Epoch 991/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 992/1000
88/88 [==============================] - 0s - loss: 0.0160
Epoch 993/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 994/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 995/1000
88/88 [==============================] - 0s - loss: 0.0160
Epoch 996/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 997/1000
88/88 [==============================] - 0s - loss: 0.0160
Epoch 998/1000
88/88 [==============================] - 0s - loss: 0.0161
Epoch 999/1000
88/88 [==============================] - 0s - loss: 0.0162
Epoch 1000/1000
88/88 [==============================] - 0s - loss: 0.0162
```

Out[9]: `<keras.callbacks.History at 0x7fbc3cd1cb38>`

1.5 Checking model performance

With your model fit we can now make predictions on both our training and testing sets.

```
In [10]: # generate predictions for training
         train_predict = model.predict(X_train)
         test_predict = model.predict(X_test)
```

In the next cell we compute training and testing errors using our trained model - you should be able to achieve at least

training_error < 0.02

and

testing_error < 0.02

with your fully trained model.

If either or both of your accuracies are larger than 0.02 re-train your model - increasing the number of epochs you take (a maximum of around 1,000 should do the job) and/or adjusting your batch_size.

```
In [11]:  # print out training and testing errors
          training_error = model.evaluate(X_train, y_train, verbose=0)
          print('training error = ' + str(training_error))

          testing_error = model.evaluate(X_test, y_test, verbose=0)
          print('testing error = ' + str(testing_error))
```

```
training error = 0.0159912935712
testing error = 0.0139957363826
```

Activating the next cell plots the original data, as well as both predictions on the training and testing sets.

```
In [12]:   ### Plot everything - the original series as well as predictions on trai
           ning and testing sets
           import matplotlib.pyplot as plt
           %matplotlib inline

           # plot original series
           plt.plot(dataset,color = 'k')

           # plot training set prediction
           split_pt = train_test_split + window_size
           plt.plot(np.arange(window_size,split_pt,1),train_predict,color = 'b')

           # plot testing set prediction
           plt.plot(np.arange(split_pt,split_pt +
           len(test_predict),1),test_predict,color = 'r')

           # pretty up graph
           plt.xlabel('day')
           plt.ylabel('(normalized) price of Apple stock')
           plt.legend(['original series','training fit','testing fit'],loc='center
            left', bbox_to_anchor=(1, 0.5))
           plt.show()
```
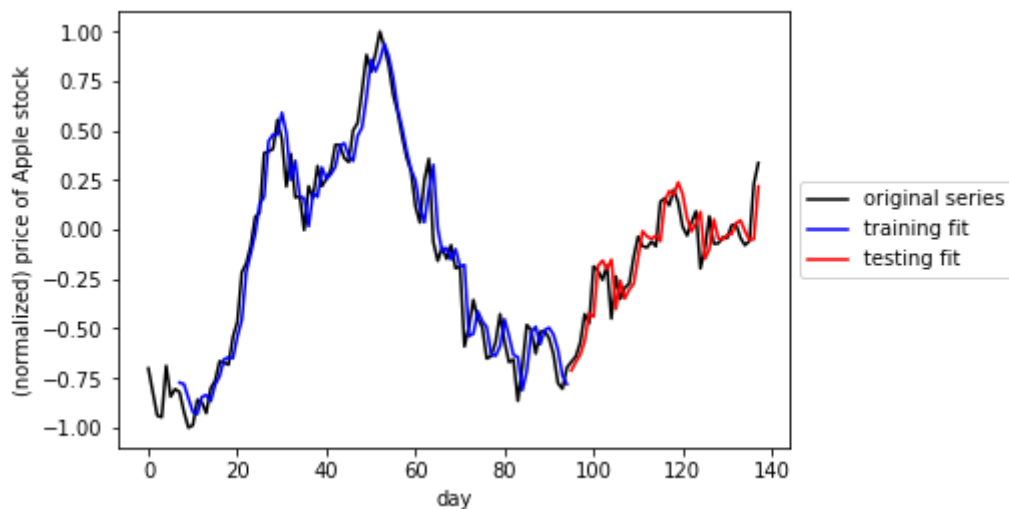

Note: you can try out any time series for this exercise! If you would like to try another see e.g., this site containing thousands of time series (https://datamarket.com/data/list/?q=provider%3Atsdl) and pick another one!

Problem 2: Create a sequence generator

2.1 Getting started

In this project you will implement a popular Recurrent Neural Network (RNN) architecture to create an English language sequence generator capable of building semi-coherent English sentences from scratch by building them up character-by-character. This will require a substantial amount amount of parameter tuning on a large training corpus (at least 100,000 characters long). In particular for this project we will be using a complete version of Sir Arthur Conan Doyle's classic book The Adventures of Sherlock Holmes.

How can we train a machine learning model to generate text automatically, character-by-character? *By showing the model many training examples so it can learn a pattern between input and output.* With this type of text generation each input is a string of valid characters like this one

dogs are grea

whlie the corresponding output is the next character in the sentence - which here is 't' (since the complete sentence is 'dogs are great'). We need to show a model many such examples in order for it to make reasonable predictions.

Fun note: For those interested in how text generation is being used check out some of the following fun resources:

- Generate wacky sentences (http://www.cs.toronto.edu/~ilya/rnn.html) with this academic RNN text generator
- Various twitter bots that tweet automatically generated text likethis one (http://tweet-generator-alex.herokuapp.com/).
- the NanoGenMo (https://github.com/NaNoGenMo/2016) annual contest to automatically produce a 50,000+ novel automatically
- Robot Shakespeare (https://github.com/genekogan/RobotShakespeare) a text generator that automatically produces Shakespear-esk sentences

2.2 Preprocessing a text dataset

Our first task is to get a large text corpus for use in training, and on it we perform a several light pre-processing tasks. The default corpus we will use is the classic book Sherlock Holmes, but you can use a variety of others as well - so long as they are fairly large (around 100,000 characters or more).

```
In [103]:   # read in the text, transforming everything to lower case
            text = open('datasets/holmes.txt').read().lower()
            print('our original text has ' + str(len(text)) + ' characters')

            our original text has 581864 characters
```

Next, lets examine a bit of the raw text. Because we are interested in creating sentences of English words automatically by building up each word character-by-character, we only want to train on valid English words. In other words - we need to remove all of the other junk characters that aren't words!

```
In [104]:   ### print out the first 1000 characters of the raw text to get a sense o
            f what we need to throw out
            text[:2000]
```

```
Out[104]:   "\ufeffproject gutenberg's the adventures of sherlock holmes, by arthur
            conan doyle\n\nthis ebook is for the use of anyone anywhere at no cost
             and with\nalmost no restrictions whatsoever.  you may copy it, give it
            away or\nre-use it under the terms of the project gutenberg license inc
            luded\nwith this ebook or online at www.gutenberg.net\n\n\ntitle: the a
            dventures of sherlock holmes\n\nauthor: arthur conan doyle\n\nposting d
            ate: april 18, 2011 [ebook #1661]\nfirst posted: november 29, 2002\n\nl
            anguage: english\n\n\n*** start of this project gutenberg ebook the adv
            entures of sherlock holmes ***\n\n\n\n\nproduced by an anonymous projec
            t gutenberg volunteer and jose menendez\n\n\n\n\n\n\n\n\nthe adventur
            es of sherlock holmes\n\nby\n\nsir arthur conan doyle\n\n\n\n   i. a sc
            andal in bohemia\n  ii. the red-headed league\n iii. a case of identity
            \n  iv. the boscombe valley mystery\n   v. the five orange pips\n  vi.
             the man with the twisted lip\n vii. the adventure of the blue carbuncl
            e\nviii. the adventure of the speckled band\n  ix. the adventure of the
            engineer's thumb\n   x. the adventure of the noble bachelor\n  xi. the
             adventure of the beryl coronet\n xii. the adventure of the copper beec
            hes\n\n\n\nadventure i. a scandal in bohemia\n\ni.\n\nto sherlock hol
            mes she is always the woman. i have seldom heard\nhim mention her under
            any other name. in his eyes she eclipses\nand predominates the whole of
            her sex. it was not that he felt\nany emotion akin to love for irene ad
            ler. all emotions, and that\none particularly, were abhorrent to his co
            ld, precise but\nadmirably balanced mind. he was, i take it, the most p
            erfect\nreasoning and observing machine that the world has seen, but as
            a\nlover he would have placed himself in a false position. he never\nsp
            oke of the softer passions, save with a gibe and a sneer. they\nwere ad
            mirable things for the observer--excellent for drawing the\nveil from m
            en's motives and actions. but for the trained reasoner\nto admit such i
            ntrusions into his own delicate and finely\nadjusted temperament was to
            introduce a dist"
```

Wow - there's a lot of junk here (i.e., weird uncommon character combinations - as this first character chunk contains the title and author page, as well as table of contents)! e.g., all the carriage return and newline sequences '\n' and '\r' sequences. We want to train our RNN on a large chunk of real english sentences - we don't want it to start thinking non-english words or strange characters are valid! - so lets clean up the data a bit.

First, since the dataset is so large and the first few hundred characters contain a lot of junk, lets cut it out. Lets also find-and-replace those newline tags with empty spaces.

```
In [105]:   ### find and replace '\n' and '\r' symbols - replacing them
            text = text[1195:]
            text = text.replace('\n',' ')     # replacing '\n' with '' simply removes
             the sequence
            text = text.replace('\r',' ')
```

Lets see how the first 1000 characters of our text looks now!

```
In [106]:   ### print out the first 1000 characters of the raw text to get a sense o
            f what we need to throw out
            text[:1000]
```

Out[106]: " to sherlock holmes she is always the woman. i have seldom heard him m
           ention her under any other name. in his eyes she eclipses and predomina
           tes the whole of her sex. it was not that he felt any emotion akin to l
           ove for irene adler. all emotions, and that one particularly, were abho
           rrent to his cold, precise but admirably balanced mind. he was, i take
            it, the most perfect reasoning and observing machine that the world ha
           s seen, but as a lover he would have placed himself in a false positio
           n. he never spoke of the softer passions, save with a gibe and a sneer.
           they were admirable things for the observer--excellent for drawing the
            veil from men's motives and actions. but for the trained reasoner to a
           dmit such intrusions into his own delicate and finely adjusted temperam
           ent was to introduce a distracting factor which might throw a doubt upo
           n all his mental results. grit in a sensitive instrument, or a crack in
           one of his own high-power lenses, would not be more disturbing than a s
           trong em"

TODO: finish cleaning the text

Lets make sure we haven't left any other non-English/proper punctuation (commas, periods, etc., are ok) characters lurking around in the depths of the text. You can do this by ennumerating all the text's unique characters, examining them, and then replacing any unwanted (non-english) characters with empty spaces! Once we find all of the text's unique characters, we can remove all of the non-English/proper punctuation ones in the next cell. Note: don't remove necessary punctuation marks! (given in the cell below).

(remember to copy your completed function into the script *my_answers.py* function titled *clean_text* before submitting your project)

```
In [107]:   ### ist all unique characters in the text and remove any non-english one
            s
            # find all unique characters in the text
            uniques = sorted(set(text))
            print("Characters in the original text:")
            print(uniques)

            # remove as many non-english characters and character sequences as you c
            an

            # get all valid characters (lowercase letters, punctuation, space)
            import string
            valid_characters = list(string.ascii_lowercase)
            valid_characters = valid_characters + list([' ', '!', '"', '&', "'",
            '(', ')', ',', '-', '.', ':', ';', '?'])
            valid_characters = sorted(valid_characters)
            print("Characters considered to be valid:")
            print(valid_characters)

            # determine characters to be removed
            invalid_characters = set(uniques).difference(valid_characters)
            print("Characters considered to be invalid which will be removed from th
            e text:")
            print(invalid_characters)
            for c in invalid_characters:
                text = text.replace(c, '')

            # shorten any extra dead space created above
            text = text.replace('  ',' ')
```

```
Characters in the original text:
[' ', '!', '"', '$', '%', '&', "'", '(', ')', '*', ',', '-', '.', '/',
 '0', '1', '2', '3', '4', '5', '6', '7', '8', '9', ':', ';', '?', '@',
 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n',
 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z', 'à', 'â',
 'è', 'é']
Characters considered to be valid:
[' ', '!', '"', '&', "'", '(', ')', ',', '-', '.', ':', ';', '?', 'a',
 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o',
 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z']
Characters considered to be invalid which will be removed from the tex
t:
{'2', '$', '6', '@', '0', 'è', '*', '/', '9', 'à', '5', '3', '%', '1',
 '7', '4', 'â', '8', 'é'}
```

With your chosen characters removed print out the first few hundred lines again just to double check that everything looks good.

In [108]: *### print out the first 2000 characters of the raw text to get a sense o f what we need to throw out*
`text[:2000]`

Out[108]: `" to sherlock holmes she is always the woman. i have seldom heard him m`
`ention her under any other name. in his eyes she eclipses and predomina`
`tes the whole of her sex. it was not that he felt any emotion akin to l`
`ove for irene adler. all emotions, and that one particularly, were abho`
`rrent to his cold, precise but admirably balanced mind. he was, i take`
`  it, the most perfect reasoning and observing machine that the world ha`
`s seen, but as a lover he would have placed himself in a false positio`
`n. he never spoke of the softer passions, save with a gibe and a sneer.`
`they were admirable things for the observer--excellent for drawing the`
`  veil from men's motives and actions. but for the trained reasoner to a`
`dmit such intrusions into his own delicate and finely adjusted temperam`
`ent was to introduce a distracting factor which might throw a doubt upo`
`n all his mental results. grit in a sensitive instrument, or a crack in`
`one of his own high-power lenses, would not be more disturbing than a s`
`trong emotion in a nature such as his. and yet there was but one woman`
`  to him, and that woman was the late irene adler, of dubious and questi`
`onable memory. i had seen little of holmes lately. my marriage had drif`
`ted us away from each other. my own complete happiness, and the home-ce`
`ntred interests which rise up around the man who first finds himself ma`
`ster of his own establishment, were sufficient to absorb all my attenti`
`on, while holmes, who loathed every form of society with his whole bohe`
`mian soul, remained in our lodgings in baker street, buried among his o`
`ld books, and alternating from week to week between cocaine and ambitio`
`n, the drowsiness of the drug, and the fierce energy of his own keen na`
`ture. he was still, as ever, deeply attracted by the study of crime, an`
`d occupied his immense faculties and extraordinary powers of observatio`
`n in following out those clues, and clearing up those mysteries which h`
`ad been abandoned as hopeless by the official police. from time to time`
`i heard some v"`

Now that we have thrown out a good number of non-English characters/character sequences lets print out some statistics about the dataset - including number of total characters and number of unique characters.

In [109]: ```
# count the number of unique characters in the text
chars = sorted(list(set(text)))

print some of the text, as well as statistics
print ("this corpus has " + str(len(text)) + " total number of characte
rs")
print ("this corpus has " + str(len(chars)) + " unique characters")
```

`this corpus has 577178 total number of characters`
`this corpus has 39 unique characters`

# 2.3 Cutting data into input/output pairs

Now that we have our text all cleaned up, how can we use it to train a model to generate sentences automatically? First we need to train a machine learning model - and in order to do that we need a set of input/output pairs for a model to train on. How can we create a set of input/output pairs from our text to train on?

Remember in part 1 of this notebook how we used a sliding window to extract input/output pairs from a time series? We do the same thing here! We slide a window of length $T$ along our giant text corpus - everything in the window becomes one input while the character following becomes its corresponding output. This process of extracting input/output pairs is illustrated in the gif below on a small example text using a window size of T = 5.



Notice one aspect of the sliding window in this gif that does not mirror the analaogous gif for time series shown in part 1 of the notebook - we do not need to slide the window along one character at a time but can move by a fixed step size $M$ greater than 1 (in the gif indeed $M = 1$). This is done with large input texts (like ours which has over 500,000 characters!) when sliding the window along one character at a time we would create far too many input/output pairs to be able to reasonably compute with.

More formally lets denote our text corpus - which is one long string of characters - as follows
$$s_0, s_1, s_2, \ldots, s_P$$

where $P$ is the length of the text (again for our text $P \approx 500,000$!). Sliding a window of size T = 5 with a step length of M = 1 (these are the parameters shown in the gif above) over this sequence produces the following list of input/output pairs

| Input | Output |
|-------|--------|
| $\langle s_1, s_2, s_3, s_4, s_5 \rangle$ | $s_6$ |
| $\langle s_2, s_3, s_4, s_5, s_6 \rangle$ | $s_7$ |
| $\vdots$ | $\vdots$ |
| $\langle s_{P-5}, s_{P-4}, s_{P-3}, s_{P-2}, s_{P-1} \rangle$ | $s_P$ |

Notice here that each input is a sequence (or vector) of 4 characters (and in general has length equal to the window size T) while each corresponding output is a single character. We created around P total number of input/output pairs (for general step size M we create around ceil(P/M) pairs).

Now its time for you to window the input time series as described above!

**TODO:** Create a function that runs a sliding window along the input text and creates associated input/output pairs. A skeleton function has been provided for you. Note that this function should input a) the text b) the window size and c) the step size, and return the input/output sequences. Note: the return items should be *lists* - not numpy arrays.

(remember to copy your completed function into the script *my_answers.py* function titled *window_transform_text* before submitting your project)

```
In [110]:   ### fill out the function below that transforms the input text and windo
            w-size into a set of input/output pairs for use with our RNN model
            import numpy as np
            def window_transform_text(text,window_size,step_size):
                # containers for input/output pairs
                inputs = []
                outputs = []

                # Slice the series into windows
                for i in range(window_size, len(text), step_size):
                    inputs.append(text[i-window_size:i])
                    outputs.append(text[i])

                return inputs,outputs
```

With our function complete we can now use it to produce input/output pairs! We employ the function in the next cell, where the window_size = 50 and step_size = 5.

```
In [111]:   # run your text window-ing function
            window_size = 100
            step_size = 5
            inputs, outputs = window_transform_text(text,window_size,step_size)
```

Lets print out a few input/output pairs to verify that we have made the right sort of stuff!

```
In [112]:   # print out a few of the input/output pairs to verify that we've made th
            e right kind of stuff to learn from
            print('input = ' + inputs[2])
            print('output = ' + outputs[2])
            print('--------------')
            print('input = ' + inputs[100])
            print('output = ' + outputs[100])
```

```
            input = ck holmes she is always the woman. i have seldom heard him ment
            ion her under any other name. in his
            output = e
            --------------
            input = never spoke of the softer passions, save with a gibe and a snee
            r. they were admirable things for the
            output =
```

Looks good!

2.4 Wait, what kind of problem is text generation again?

In part 1 of this notebook we used the same pre-processing technique - the sliding window - to produce a set of training input/output pairs to tackle the problem of time series prediction *by treating the problem as one of regression*. So what sort of problem do we have here now, with text generation? Well, the time series prediction was a regression problem because the output (one value of the time series) was a continuous value. Here - for character-by-character text generation - each output is a *single character*. This isn't a continuous value - but a distinct class - therefore **character-by-character text generation is a classification problem**.

How many classes are there in the data? Well, the number of classes is equal to the number of unique characters we have to predict! How many of those were there in our dataset again? Lets print out the value again.

```
In [113]:  # print out the number of unique characters in the dataset
           chars = sorted(list(set(text)))
           print ("this corpus has " +  str(len(chars)) + " unique characters")
           print ('and these characters are ')
           print (chars)
```

```
this corpus has 39 unique characters
and these characters are
[' ', '!', '"', '&', "'", '(', ')', ',', '-', '.', ':', ';', '?', 'a',
 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o',
 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z']
```

Rockin' - so we have a multi-class classification problem on our hands!

2.5 One-hot encoding characters

There's just one last issue we have to deal with before tackle: machine learning algorithm deal with numerical data and all of our input/output pairs are characters. So we just need to transform our characters into equivalent numerical values. The most common way of doing this is via a 'one-hot encoding' scheme. Here's how it works.

We transform each character in our inputs/outputs into a vector with length equal to the number of unique characters in our text. This vector is all zeros except one location where we place a 1 - and this location is unique to each character type. e.g., we transform 'a', 'b', and 'c' as follows

$$a \longleftarrow \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \quad b \longleftarrow \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \quad c \longleftarrow \begin{bmatrix} 0 \\ 0 \\ 1 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \dots$$

where each vector has 32 entries (or in general: number of entries = number of unique characters in text).

The first practical step towards doing this one-hot encoding is to form a dictionary mapping each unique character to a unique integer, and one dictionary to do the reverse mapping. We can then use these dictionaries to quickly make our one-hot encodings, as well as re-translate (from integers to characters) the results of our trained RNN classification model.

```
In [114]:  # this dictionary is a function mapping each unique character to a uniqu
           e integer
           chars_to_indices = dict((c, i) for i, c in enumerate(chars))   # map each
            unique character to unique integer

           # this dictionary is a function mapping each unique integer back to a un
           ique character
           indices_to_chars = dict((i, c) for i, c in enumerate(chars))   # map each
            unique integer back to unique character
```

Now we can transform our input/output pairs - consisting of characters - to equivalent input/output pairs made up of one-hot encoded vectors. In the next cell we provide a function for doing just this: it takes in the raw character input/outputs and returns their numerical versions. In particular the numerical input is given as X, and numerical output is given as the y

```
In [115]:  # transform character-based input/output into equivalent numerical versi
           ons
           def encode_io_pairs(text,window_size,step_size):
               # number of unique chars
               chars = sorted(list(set(text)))
               num_chars = len(chars)

               # cut up text into character input/output pairs
               inputs, outputs = window_transform_text(text,window_size,step_size)

               # create empty vessels for one-hot encoded input/output
               X = np.zeros((len(inputs), window_size, num_chars), dtype=np.bool)
               y = np.zeros((len(inputs), num_chars), dtype=np.bool)

               # loop over inputs/outputs and tranform and store in X/y
               for i, sentence in enumerate(inputs):
                   for t, char in enumerate(sentence):
                       X[i, t, chars_to_indices[char]] = 1
                   y[i, chars_to_indices[outputs[i]]] = 1

               return X,y
```

Now run the one-hot encoding function by activating the cell below and transform our input/output pairs!

```
In [116]:  # use your function
           window_size = 100
           step_size = 5
           X,y = encode_io_pairs(text,window_size,step_size)
```

2.6 Setting up our RNN

With our dataset loaded and the input/output pairs extracted / transformed we can now begin setting up our RNN for training. Again we will use Keras to quickly build a single hidden layer RNN - where our hidden layer consists of LTSM modules.

Time to get to work: build a 3 layer RNN model of the following specification

- layer 1 should be an LSTM module with 200 hidden units --> note this should have input_shape = (window_size,len(chars)) where len(chars) = number of unique characters in your cleaned text
- layer 2 should be a linear module, fully connected, with len(chars) hidden units --> where len(chars) = number of unique characters in your cleaned text
- layer 3 should be a softmax activation (since we are solving a *multiclass classification*)
- Use the **categorical_crossentropy** loss

This network can be constructed using just a few lines - as with the RNN network you made in part 1 of this notebook. See e.g., the general Keras documentation (https://keras.io/getting-started/sequential-model-guide/) and the LTSM documentation in particular (https://keras.io/layers/recurrent/) for examples of how to quickly use Keras to build neural network models.

```
In [117]:  ### necessary functions from the keras library
           from keras.models import Sequential
           from keras.layers import Dense, Activation, LSTM
           from keras.optimizers import RMSprop
           from keras.utils.data_utils import get_file
           import keras
           import random


           # build the required RNN model: a single LSTM hidden layer with softmax
            activation, categorical_crossentropy loss
           model = Sequential()
           model.add(LSTM(200, input_shape=(window_size,len(chars))))
           model.add(Dense(len(chars)))
           model.add(Activation('softmax'))

           # initialize optimizer
           optimizer = keras.optimizers.RMSprop(lr=0.001, rho=0.9, epsilon=1e-08, d
           ecay=0.0)

           # compile model --> make sure initialized optimizer and callbacks - as d
           efined above - are used
           model.compile(loss='categorical_crossentropy', optimizer=optimizer)
```

2.7 Training our RNN model for text generation

With our RNN setup we can now train it! Lets begin by trying it out on a small subset of the larger version. In the next cell we take the first 10,000 input/output pairs from our training database to learn on.

```
In [35]:  # a small subset of our input/output pairs
          Xsmall = X[:10000,:,:]
          ysmall = y[:10000,:]
```

Now lets fit our model!

```
In [36]:  # train the model
          model.fit(Xsmall, ysmall, batch_size=500, epochs=40,verbose = 1)

          # save weights
          model.save_weights('model_weights/best_RNN_small_textdata_weights.hdf5')
```

```
Epoch 1/40
10000/10000 [==============================] - 76s - loss: 3.1232
Epoch 2/40
10000/10000 [==============================] - 4s - loss: 2.9581
Epoch 3/40
10000/10000 [==============================] - 4s - loss: 2.9377
Epoch 4/40
10000/10000 [==============================] - 4s - loss: 2.9153
Epoch 5/40
10000/10000 [==============================] - 4s - loss: 2.8741
Epoch 6/40
10000/10000 [==============================] - 4s - loss: 2.8171
Epoch 7/40
10000/10000 [==============================] - 4s - loss: 2.7476
Epoch 8/40
10000/10000 [==============================] - 4s - loss: 2.6716
Epoch 9/40
10000/10000 [==============================] - 4s - loss: 2.6158
Epoch 10/40
10000/10000 [==============================] - 4s - loss: 2.5567
Epoch 11/40
10000/10000 [==============================] - 4s - loss: 2.5091
Epoch 12/40
10000/10000 [==============================] - 4s - loss: 2.4669
Epoch 13/40
10000/10000 [==============================] - 4s - loss: 2.4281
Epoch 14/40
10000/10000 [==============================] - 4s - loss: 2.3946
Epoch 15/40
10000/10000 [==============================] - 4s - loss: 2.3608
Epoch 16/40
10000/10000 [==============================] - 4s - loss: 2.3331
Epoch 17/40
10000/10000 [==============================] - 4s - loss: 2.3047
Epoch 18/40
10000/10000 [==============================] - 4s - loss: 2.2815
Epoch 19/40
10000/10000 [==============================] - 4s - loss: 2.2609
Epoch 20/40
10000/10000 [==============================] - 4s - loss: 2.2345
Epoch 21/40
10000/10000 [==============================] - 4s - loss: 2.2198
Epoch 22/40
10000/10000 [==============================] - 4s - loss: 2.1996
Epoch 23/40
10000/10000 [==============================] - 4s - loss: 2.1793
Epoch 24/40
10000/10000 [==============================] - 4s - loss: 2.1609
Epoch 25/40
10000/10000 [==============================] - 4s - loss: 2.1385
Epoch 26/40
10000/10000 [==============================] - 4s - loss: 2.1230
Epoch 27/40
10000/10000 [==============================] - 4s - loss: 2.1032
Epoch 28/40
10000/10000 [==============================] - 4s - loss: 2.0829
Epoch 29/40
```

```
10000/10000 [==============================] - 4s - loss: 2.0638
Epoch 30/40
10000/10000 [==============================] - 4s - loss: 2.0467
Epoch 31/40
10000/10000 [==============================] - 4s - loss: 2.0290
Epoch 32/40
10000/10000 [==============================] - 4s - loss: 2.0060
Epoch 33/40
10000/10000 [==============================] - 4s - loss: 1.9884
Epoch 34/40
10000/10000 [==============================] - 4s - loss: 1.9732
Epoch 35/40
10000/10000 [==============================] - 4s - loss: 1.9479
Epoch 36/40
10000/10000 [==============================] - 4s - loss: 1.9333
Epoch 37/40
10000/10000 [==============================] - 4s - loss: 1.9055
Epoch 38/40
10000/10000 [==============================] - 4s - loss: 1.8896
Epoch 39/40
10000/10000 [==============================] - 4s - loss: 1.8694
Epoch 40/40
10000/10000 [==============================] - 4s - loss: 1.8448
```

How do we make a given number of predictions (characters) based on this fitted model?

First we predict the next character after following any chunk of characters in the text of length equal to our chosen window size. Then we remove the first character in our input sequence and tack our prediction onto the end. This gives us a slightly changed sequence of inputs that still has length equal to the size of our window. We then feed in this updated input sequence into the model to predict the another character. Together then we have two predicted characters following our original input sequence. Repeating this process N times gives us N predicted characters.

In the next Python cell we provide you with a completed function that does just this - it makes predictions when given a) a trained RNN model, b) a subset of (window_size) characters from the text, and c) a number of characters to predict (to follow our input subset).

```
In [118]:  # function that uses trained model to predict a desired number of future
           characters
           def predict_next_chars(model,input_chars,num_to_predict):
               # create output
               predicted_chars = ''
               for i in range(num_to_predict):
                   # convert this round's predicted characters to numerical input

                   x_test = np.zeros((1, window_size, len(chars)))
                   for t, char in enumerate(input_chars):
                       x_test[0, t, chars_to_indices[char]] = 1.

                   # make this round's prediction
                   test_predict = model.predict(x_test,verbose = 0)[0]

                   # translate numerical prediction back to characters
                   r = np.argmax(test_predict)                        # predict
               class of each test input
                   d = indices_to_chars[r]

                   # update predicted_chars and input
                   predicted_chars+=d
                   input_chars+=d
                   input_chars = input_chars[1:]
               return predicted_chars
```

With your trained model try a few subsets of the complete text as input - note the length of each must be exactly equal to the window size. For each subset us the function above to predict the next 100 characters that follow each input.

```
In [119]:  # choose an input sequence and use the prediction function in the previo
           us Python cell to predict 100 characters following it
           # get an appropriately sized chunk of characters from the text
           start_inds = [2048]

           # load in weights
           model.load_weights('model_weights/best_RNN_small_textdata_weights.hdf5')
           for s in start_inds:
               start_index = s
               input_chars = text[start_index: start_index + window_size]

               # use the prediction function
               predict_input = predict_next_chars(model,input_chars,num_to_predict
           = 100)

               # print out input characters
               print('------------------')
               input_line = 'input chars = ' + '\n' +  input_chars + '"' + '\n'
               print(input_line)

               # print out predicted characters
               line = 'predicted chars = ' + '\n' +  predict_input + '"' + '\n'
               print(line)
```

```
------------------
input chars =
essa in the case of the trepoff murder, of his clearing up of the singu
lar tragedy of the atkinson b"

predicted chars =
e the sistor in the tored of the couthe in the the wish chich serter in
the stored in the stored in "
```

This looks ok, but not great. Now lets try the same experiment with a larger chunk of the data - with the first 100,000 input/output pairs.

Tuning RNNs for a typical character dataset like the one we will use here is a computationally intensive endeavour and thus timely on a typical CPU. Using a reasonably sized cloud-based GPU can speed up training by a factor of 10. Also because of the long training time it is highly recommended that you carefully write the output of each step of your process to file. This is so that all of your results are saved even if you close the web browser you're working out of, as the processes will continue processing in the background but variables/output in the notebook system will not update when you open it again.

In the next cell we show you how to create a text file in Python and record data to it. This sort of setup can be used to record your final predictions.

```
In [59]:  ### A simple way to write output to file
          f = open('my_test_output.txt', 'w')              # create an output file
           to write too
          f.write('this is only a test ' + '\n')           # print some output tex
          t
          x = 2
          f.write('the value of x is ' + str(x) + '\n')    # record a variable val
          ue
          f.close()

          # print out the contents of my_test_output.txt
          f = open('my_test_output.txt', 'r')              # create an output file
           to write too
          f.read()
```

Out[59]: 'this is only a test \nthe value of x is 2\n'

With this recording devices we can now more safely perform experiments on larger portions of the text. In the next cell we will use the first 100,000 input/output pairs to train our RNN model.

First we fit our model to the dataset, then generate text using the trained model in precisely the same generation method applied before on the small dataset.

Note: your generated words should be - by and large - more realistic than with the small dataset, but you won't be able to generate perfect English sentences even with this amount of data. A rule of thumb: your model is working well if you generate sentences that largely contain real English words.

```
In [ ]:    # a small subset of our input/output pairs
           Xlarge = X[:100000,:,:]
           ylarge = y[:100000,:]

           # fit to our larger dataset
           model.fit(Xlarge, ylarge, batch_size=500, epochs=700,verbose = 1)

           # save weights
           model.save_weights('model_weights/best_RNN_large_textdata_weights.hdf5')
```

```
Epoch 1/700
100000/100000 [==============================] - 43s - loss: 2.7872
Epoch 2/700
100000/100000 [==============================] - 43s - loss: 2.3589
Epoch 3/700
100000/100000 [==============================] - 43s - loss: 2.1887
Epoch 4/700
100000/100000 [==============================] - 43s - loss: 2.0767
Epoch 5/700
100000/100000 [==============================] - 43s - loss: 1.9917
Epoch 6/700
100000/100000 [==============================] - 43s - loss: 1.9238
Epoch 7/700
100000/100000 [==============================] - 43s - loss: 1.8669
Epoch 8/700
100000/100000 [==============================] - 43s - loss: 1.8180
Epoch 9/700
100000/100000 [==============================] - 43s - loss: 1.7749
Epoch 10/700
100000/100000 [==============================] - 43s - loss: 1.7354
Epoch 11/700
100000/100000 [==============================] - 43s - loss: 1.6990
Epoch 12/700
100000/100000 [==============================] - 43s - loss: 1.6640
Epoch 13/700
100000/100000 [==============================] - 43s - loss: 1.6309
Epoch 14/700
100000/100000 [==============================] - 43s - loss: 1.5993
Epoch 15/700
100000/100000 [==============================] - 43s - loss: 1.5695
Epoch 16/700
100000/100000 [==============================] - 43s - loss: 1.5403
Epoch 17/700
100000/100000 [==============================] - 43s - loss: 1.5115
Epoch 18/700
100000/100000 [==============================] - 43s - loss: 1.4839
Epoch 19/700
100000/100000 [==============================] - 43s - loss: 1.4569
Epoch 20/700
100000/100000 [==============================] - 43s - loss: 1.4301
Epoch 21/700
100000/100000 [==============================] - 43s - loss: 1.4036
Epoch 22/700
100000/100000 [==============================] - 43s - loss: 1.3768
Epoch 23/700
100000/100000 [==============================] - 43s - loss: 1.3515
Epoch 24/700
100000/100000 [==============================] - 43s - loss: 1.3249
Epoch 25/700
100000/100000 [==============================] - 43s - loss: 1.2996
Epoch 26/700
100000/100000 [==============================] - 43s - loss: 1.2732
Epoch 27/700
100000/100000 [==============================] - 43s - loss: 1.2469
Epoch 28/700
100000/100000 [==============================] - 43s - loss: 1.2217
Epoch 29/700
```

```
100000/100000 [==============================] - 43s - loss: 1.1953
Epoch 30/700
100000/100000 [==============================] - 43s - loss: 1.1691
Epoch 31/700
100000/100000 [==============================] - 43s - loss: 1.1445
Epoch 32/700
100000/100000 [==============================] - 43s - loss: 1.1192
Epoch 33/700
100000/100000 [==============================] - 43s - loss: 1.0935
Epoch 34/700
100000/100000 [==============================] - 43s - loss: 1.0687
Epoch 35/700
100000/100000 [==============================] - 43s - loss: 1.0451
Epoch 36/700
100000/100000 [==============================] - 43s - loss: 1.0203
Epoch 37/700
100000/100000 [==============================] - 43s - loss: 0.9979
Epoch 38/700
100000/100000 [==============================] - 43s - loss: 0.9746
Epoch 39/700
100000/100000 [==============================] - 43s - loss: 0.9525
Epoch 40/700
100000/100000 [==============================] - 43s - loss: 0.9317
Epoch 41/700
100000/100000 [==============================] - 43s - loss: 0.9103
Epoch 42/700
100000/100000 [==============================] - 43s - loss: 0.8898
Epoch 43/700
100000/100000 [==============================] - 43s - loss: 0.8712
Epoch 44/700
100000/100000 [==============================] - 43s - loss: 0.8517
Epoch 45/700
100000/100000 [==============================] - 43s - loss: 0.8343
Epoch 46/700
100000/100000 [==============================] - 43s - loss: 0.8154
Epoch 47/700
100000/100000 [==============================] - 43s - loss: 0.7996
Epoch 48/700
100000/100000 [==============================] - 43s - loss: 0.7851
Epoch 49/700
100000/100000 [==============================] - 43s - loss: 0.7694
Epoch 50/700
100000/100000 [==============================] - 43s - loss: 0.7539
Epoch 51/700
100000/100000 [==============================] - 43s - loss: 0.7389
Epoch 52/700
100000/100000 [==============================] - 43s - loss: 0.7270
Epoch 53/700
100000/100000 [==============================] - 43s - loss: 0.7130
Epoch 54/700
100000/100000 [==============================] - 43s - loss: 0.7001
Epoch 55/700
100000/100000 [==============================] - 43s - loss: 0.6890
Epoch 56/700
100000/100000 [==============================] - 43s - loss: 0.6774
Epoch 57/700
100000/100000 [==============================] - 43s - loss: 0.6656
```

```
Epoch 58/700
100000/100000 [==============================] - 43s - loss: 0.6564
Epoch 59/700
100000/100000 [==============================] - 43s - loss: 0.6448
Epoch 60/700
100000/100000 [==============================] - 43s - loss: 0.6363
Epoch 61/700
100000/100000 [==============================] - 43s - loss: 0.6256
Epoch 62/700
100000/100000 [==============================] - 43s - loss: 0.6155
Epoch 63/700
100000/100000 [==============================] - 43s - loss: 0.6070
Epoch 64/700
100000/100000 [==============================] - 43s - loss: 0.5990
Epoch 65/700
100000/100000 [==============================] - 43s - loss: 0.5908
Epoch 66/700
100000/100000 [==============================] - 43s - loss: 0.5825
Epoch 67/700
100000/100000 [==============================] - 43s - loss: 0.5747
Epoch 68/700
100000/100000 [==============================] - 43s - loss: 0.5670
Epoch 69/700
100000/100000 [==============================] - 43s - loss: 0.5596
Epoch 70/700
100000/100000 [==============================] - 43s - loss: 0.5537
Epoch 71/700
100000/100000 [==============================] - 43s - loss: 0.5449
Epoch 72/700
100000/100000 [==============================] - 43s - loss: 0.5395
Epoch 73/700
100000/100000 [==============================] - 43s - loss: 0.5336
Epoch 74/700
100000/100000 [==============================] - 43s - loss: 0.5265
Epoch 75/700
100000/100000 [==============================] - 43s - loss: 0.5202
Epoch 76/700
100000/100000 [==============================] - 43s - loss: 0.5140
Epoch 77/700
100000/100000 [==============================] - 43s - loss: 0.5080
Epoch 78/700
100000/100000 [==============================] - 43s - loss: 0.5042
Epoch 79/700
100000/100000 [==============================] - 43s - loss: 0.4958
Epoch 80/700
100000/100000 [==============================] - 43s - loss: 0.4912
Epoch 81/700
100000/100000 [==============================] - 43s - loss: 0.4838
Epoch 82/700
100000/100000 [==============================] - 43s - loss: 0.4813
Epoch 83/700
100000/100000 [==============================] - 43s - loss: 0.4772
Epoch 84/700
100000/100000 [==============================] - 43s - loss: 0.4721
Epoch 85/700
100000/100000 [==============================] - 43s - loss: 0.4683
Epoch 86/700
```

```
100000/100000 [==============================] - 43s - loss: 0.4631
Epoch 87/700
100000/100000 [==============================] - 43s - loss: 0.4561
Epoch 88/700
100000/100000 [==============================] - 43s - loss: 0.4530
Epoch 89/700
100000/100000 [==============================] - 43s - loss: 0.4483
Epoch 90/700
100000/100000 [==============================] - 43s - loss: 0.4427
Epoch 91/700
100000/100000 [==============================] - 43s - loss: 0.4384
Epoch 92/700
100000/100000 [==============================] - 43s - loss: 0.4364
Epoch 93/700
100000/100000 [==============================] - 43s - loss: 0.4312
Epoch 94/700
100000/100000 [==============================] - 43s - loss: 0.4267
Epoch 95/700
100000/100000 [==============================] - 43s - loss: 0.4226
Epoch 96/700
100000/100000 [==============================] - 43s - loss: 0.4178
Epoch 97/700
100000/100000 [==============================] - 43s - loss: 0.4155
Epoch 98/700
100000/100000 [==============================] - 43s - loss: 0.4106
Epoch 99/700
100000/100000 [==============================] - 43s - loss: 0.4070
Epoch 100/700
100000/100000 [==============================] - 43s - loss: 0.4026
Epoch 101/700
100000/100000 [==============================] - 43s - loss: 0.3998
Epoch 102/700
100000/100000 [==============================] - 43s - loss: 0.3984
Epoch 103/700
100000/100000 [==============================] - 43s - loss: 0.3922
Epoch 104/700
100000/100000 [==============================] - 43s - loss: 0.3904
Epoch 105/700
100000/100000 [==============================] - 43s - loss: 0.3907
Epoch 106/700
100000/100000 [==============================] - 43s - loss: 0.3827
Epoch 107/700
100000/100000 [==============================] - 43s - loss: 0.3807
Epoch 108/700
100000/100000 [==============================] - 43s - loss: 0.3769
Epoch 109/700
100000/100000 [==============================] - 43s - loss: 0.3766
Epoch 110/700
100000/100000 [==============================] - 43s - loss: 0.3726
Epoch 111/700
100000/100000 [==============================] - 43s - loss: 0.3705
Epoch 112/700
100000/100000 [==============================] - 43s - loss: 0.3696
Epoch 113/700
100000/100000 [==============================] - 43s - loss: 0.3637
Epoch 114/700
100000/100000 [==============================] - 43s - loss: 0.3612
```

```
Epoch 115/700
100000/100000 [==============================] - 43s - loss: 0.3570
Epoch 116/700
100000/100000 [==============================] - 43s - loss: 0.3559
Epoch 117/700
100000/100000 [==============================] - 43s - loss: 0.3524
Epoch 118/700
100000/100000 [==============================] - 43s - loss: 0.3534
Epoch 119/700
100000/100000 [==============================] - 43s - loss: 0.3475
Epoch 120/700
100000/100000 [==============================] - 43s - loss: 0.3435
Epoch 121/700
100000/100000 [==============================] - 43s - loss: 0.3430
Epoch 122/700
100000/100000 [==============================] - 43s - loss: 0.3395
Epoch 123/700
100000/100000 [==============================] - 43s - loss: 0.3367
Epoch 124/700
100000/100000 [==============================] - 43s - loss: 0.3335
Epoch 125/700
100000/100000 [==============================] - 43s - loss: 0.3306
Epoch 126/700
100000/100000 [==============================] - 43s - loss: 0.3298
Epoch 127/700
100000/100000 [==============================] - 43s - loss: 0.3250
Epoch 128/700
100000/100000 [==============================] - 43s - loss: 0.3256
Epoch 129/700
100000/100000 [==============================] - 43s - loss: 0.3230
Epoch 130/700
100000/100000 [==============================] - 43s - loss: 0.3196
Epoch 131/700
100000/100000 [==============================] - 43s - loss: 0.3169
Epoch 132/700
100000/100000 [==============================] - 43s - loss: 0.3147
Epoch 133/700
100000/100000 [==============================] - 43s - loss: 0.3138
Epoch 134/700
100000/100000 [==============================] - 43s - loss: 0.3078
Epoch 135/700
100000/100000 [==============================] - 43s - loss: 0.3074
Epoch 136/700
100000/100000 [==============================] - 43s - loss: 0.3081
Epoch 137/700
100000/100000 [==============================] - 43s - loss: 0.3038
Epoch 138/700
100000/100000 [==============================] - 43s - loss: 0.3022
Epoch 139/700
100000/100000 [==============================] - 43s - loss: 0.2994
Epoch 140/700
100000/100000 [==============================] - 43s - loss: 0.2987
Epoch 141/700
100000/100000 [==============================] - 43s - loss: 0.2972
Epoch 142/700
100000/100000 [==============================] - 43s - loss: 0.2937
Epoch 143/700
```

```
100000/100000 [==============================] - 43s - loss: 0.2921
Epoch 144/700
100000/100000 [==============================] - 43s - loss: 0.2926
Epoch 145/700
100000/100000 [==============================] - 43s - loss: 0.2888
Epoch 146/700
100000/100000 [==============================] - 43s - loss: 0.2881
Epoch 147/700
100000/100000 [==============================] - 43s - loss: 0.2833
Epoch 148/700
100000/100000 [==============================] - 43s - loss: 0.2822
Epoch 149/700
100000/100000 [==============================] - 43s - loss: 0.2803
Epoch 150/700
100000/100000 [==============================] - 43s - loss: 0.2777
Epoch 151/700
100000/100000 [==============================] - 43s - loss: 0.2783
Epoch 152/700
100000/100000 [==============================] - 43s - loss: 0.2760
Epoch 153/700
100000/100000 [==============================] - 43s - loss: 0.2722
Epoch 154/700
100000/100000 [==============================] - 43s - loss: 0.2736
Epoch 155/700
100000/100000 [==============================] - 43s - loss: 0.2704
Epoch 156/700
100000/100000 [==============================] - 43s - loss: 0.2705
Epoch 157/700
100000/100000 [==============================] - 43s - loss: 0.2681
Epoch 158/700
100000/100000 [==============================] - 43s - loss: 0.2659
Epoch 159/700
100000/100000 [==============================] - 43s - loss: 0.2635
Epoch 160/700
100000/100000 [==============================] - 43s - loss: 0.2625
Epoch 161/700
100000/100000 [==============================] - 43s - loss: 0.2615
Epoch 162/700
100000/100000 [==============================] - 43s - loss: 0.2621
Epoch 163/700
100000/100000 [==============================] - 43s - loss: 0.2592
Epoch 164/700
100000/100000 [==============================] - 43s - loss: 0.2552
Epoch 165/700
100000/100000 [==============================] - 43s - loss: 0.2548
Epoch 166/700
100000/100000 [==============================] - 43s - loss: 0.2529
Epoch 167/700
100000/100000 [==============================] - 43s - loss: 0.2530
Epoch 168/700
100000/100000 [==============================] - 43s - loss: 0.2488
Epoch 169/700
100000/100000 [==============================] - 43s - loss: 0.2485
Epoch 170/700
100000/100000 [==============================] - 43s - loss: 0.2485
Epoch 171/700
100000/100000 [==============================] - 43s - loss: 0.2484
```

```
Epoch 172/700
100000/100000 [==============================] - 43s - loss: 0.2466
Epoch 173/700
100000/100000 [==============================] - 43s - loss: 0.2444
Epoch 174/700
100000/100000 [==============================] - 43s - loss: 0.2446
Epoch 175/700
100000/100000 [==============================] - 43s - loss: 0.2406
Epoch 176/700
100000/100000 [==============================] - 43s - loss: 0.2386
Epoch 177/700
100000/100000 [==============================] - 43s - loss: 0.2380
Epoch 178/700
100000/100000 [==============================] - 43s - loss: 0.2380
Epoch 179/700
100000/100000 [==============================] - 43s - loss: 0.2354
Epoch 180/700
100000/100000 [==============================] - 43s - loss: 0.2358
Epoch 181/700
100000/100000 [==============================] - 43s - loss: 0.2356
Epoch 182/700
100000/100000 [==============================] - 43s - loss: 0.2345
Epoch 183/700
100000/100000 [==============================] - 43s - loss: 0.2293
Epoch 184/700
100000/100000 [==============================] - 43s - loss: 0.2298
Epoch 185/700
100000/100000 [==============================] - 43s - loss: 0.2310
Epoch 186/700
100000/100000 [==============================] - 43s - loss: 0.2292
Epoch 187/700
100000/100000 [==============================] - 43s - loss: 0.2281
Epoch 188/700
100000/100000 [==============================] - 43s - loss: 0.2250
Epoch 189/700
100000/100000 [==============================] - 43s - loss: 0.2240
Epoch 190/700
100000/100000 [==============================] - 43s - loss: 0.2243
Epoch 191/700
100000/100000 [==============================] - 43s - loss: 0.2213
Epoch 192/700
100000/100000 [==============================] - 43s - loss: 0.2229
Epoch 193/700
100000/100000 [==============================] - 43s - loss: 0.2177
Epoch 194/700
100000/100000 [==============================] - 43s - loss: 0.2180
Epoch 195/700
100000/100000 [==============================] - 43s - loss: 0.2181
Epoch 196/700
100000/100000 [==============================] - 43s - loss: 0.2144
Epoch 197/700
100000/100000 [==============================] - 43s - loss: 0.2144
Epoch 198/700
100000/100000 [==============================] - 43s - loss: 0.2141
Epoch 199/700
100000/100000 [==============================] - 43s - loss: 0.2136
Epoch 200/700
```

```
100000/100000 [==============================] - 43s - loss: 0.2155
Epoch 201/700
100000/100000 [==============================] - 43s - loss: 0.2113
Epoch 202/700
100000/100000 [==============================] - 43s - loss: 0.2108
Epoch 203/700
100000/100000 [==============================] - 43s - loss: 0.2090
Epoch 204/700
100000/100000 [==============================] - 43s - loss: 0.2121
Epoch 205/700
100000/100000 [==============================] - 43s - loss: 0.2096
Epoch 206/700
100000/100000 [==============================] - 43s - loss: 0.2092
Epoch 207/700
100000/100000 [==============================] - 43s - loss: 0.2078
Epoch 208/700
100000/100000 [==============================] - 43s - loss: 0.2047
Epoch 209/700
100000/100000 [==============================] - 43s - loss: 0.2056
Epoch 210/700
100000/100000 [==============================] - 43s - loss: 0.2016
Epoch 211/700
100000/100000 [==============================] - 43s - loss: 0.2035
Epoch 212/700
100000/100000 [==============================] - 43s - loss: 0.2030
Epoch 213/700
100000/100000 [==============================] - 43s - loss: 0.1989
Epoch 214/700
100000/100000 [==============================] - 43s - loss: 0.1983
Epoch 215/700
100000/100000 [==============================] - 43s - loss: 0.1988
Epoch 216/700
100000/100000 [==============================] - 43s - loss: 0.1961
Epoch 217/700
100000/100000 [==============================] - 43s - loss: 0.1964
Epoch 218/700
100000/100000 [==============================] - 43s - loss: 0.1934
Epoch 219/700
100000/100000 [==============================] - 43s - loss: 0.1970
Epoch 220/700
100000/100000 [==============================] - 43s - loss: 0.1958
Epoch 221/700
100000/100000 [==============================] - 43s - loss: 0.1949
Epoch 222/700
100000/100000 [==============================] - 43s - loss: 0.1899
Epoch 223/700
100000/100000 [==============================] - 43s - loss: 0.1943
Epoch 224/700
100000/100000 [==============================] - 43s - loss: 0.1913
Epoch 225/700
100000/100000 [==============================] - 43s - loss: 0.1925
Epoch 226/700
100000/100000 [==============================] - 43s - loss: 0.1892
Epoch 227/700
100000/100000 [==============================] - 43s - loss: 0.1875
Epoch 228/700
100000/100000 [==============================] - 43s - loss: 0.1887
```

```
Epoch 229/700
100000/100000 [==============================] - 43s - loss: 0.1864
Epoch 230/700
100000/100000 [==============================] - 43s - loss: 0.1882
Epoch 231/700
100000/100000 [==============================] - 43s - loss: 0.1848
Epoch 232/700
100000/100000 [==============================] - 43s - loss: 0.1852
Epoch 233/700
100000/100000 [==============================] - 43s - loss: 0.1869
Epoch 234/700
100000/100000 [==============================] - 43s - loss: 0.1841
Epoch 235/700
100000/100000 [==============================] - 43s - loss: 0.1809
Epoch 236/700
100000/100000 [==============================] - 43s - loss: 0.1797
Epoch 237/700
100000/100000 [==============================] - 43s - loss: 0.1795
Epoch 238/700
100000/100000 [==============================] - 43s - loss: 0.1788
Epoch 239/700
100000/100000 [==============================] - 43s - loss: 0.1781
Epoch 240/700
100000/100000 [==============================] - 43s - loss: 0.1778
Epoch 241/700
100000/100000 [==============================] - 43s - loss: 0.1795
Epoch 242/700
100000/100000 [==============================] - 43s - loss: 0.1775
Epoch 243/700
100000/100000 [==============================] - 43s - loss: 0.1782
Epoch 244/700
100000/100000 [==============================] - 43s - loss: 0.1777
Epoch 245/700
100000/100000 [==============================] - 43s - loss: 0.1773
Epoch 246/700
100000/100000 [==============================] - 43s - loss: 0.1763
Epoch 247/700
100000/100000 [==============================] - 43s - loss: 0.1749
Epoch 248/700
100000/100000 [==============================] - 43s - loss: 0.1721
Epoch 249/700
100000/100000 [==============================] - 43s - loss: 0.1750
Epoch 250/700
100000/100000 [==============================] - 43s - loss: 0.1725
Epoch 251/700
100000/100000 [==============================] - 43s - loss: 0.1740
Epoch 252/700
100000/100000 [==============================] - 43s - loss: 0.1713
Epoch 253/700
100000/100000 [==============================] - 43s - loss: 0.1701
Epoch 254/700
100000/100000 [==============================] - 43s - loss: 0.1689
Epoch 255/700
100000/100000 [==============================] - 43s - loss: 0.1688
Epoch 256/700
100000/100000 [==============================] - 43s - loss: 0.1697
Epoch 257/700
```

```
100000/100000 [==============================] - 43s - loss: 0.1731
Epoch 258/700
100000/100000 [==============================] - 43s - loss: 0.1701
Epoch 259/700
100000/100000 [==============================] - 43s - loss: 0.1680
Epoch 260/700
100000/100000 [==============================] - 43s - loss: 0.1717
Epoch 261/700
100000/100000 [==============================] - 43s - loss: 0.1663
Epoch 262/700
100000/100000 [==============================] - 43s - loss: 0.1677
Epoch 263/700
100000/100000 [==============================] - 43s - loss: 0.1674
Epoch 264/700
100000/100000 [==============================] - 43s - loss: 0.1664
Epoch 265/700
100000/100000 [==============================] - 43s - loss: 0.1641
Epoch 266/700
100000/100000 [==============================] - 43s - loss: 0.1632
Epoch 267/700
100000/100000 [==============================] - 43s - loss: 0.1650
Epoch 268/700
100000/100000 [==============================] - 43s - loss: 0.1637
Epoch 269/700
100000/100000 [==============================] - 43s - loss: 0.1636
Epoch 270/700
100000/100000 [==============================] - 43s - loss: 0.1618
Epoch 271/700
100000/100000 [==============================] - 43s - loss: 0.1620
Epoch 272/700
100000/100000 [==============================] - 43s - loss: 0.1590
Epoch 273/700
100000/100000 [==============================] - 43s - loss: 0.1636
Epoch 274/700
100000/100000 [==============================] - 43s - loss: 0.1616
Epoch 275/700
100000/100000 [==============================] - 43s - loss: 0.1601
Epoch 276/700
100000/100000 [==============================] - 43s - loss: 0.1582
Epoch 277/700
100000/100000 [==============================] - 43s - loss: 0.1580
Epoch 278/700
100000/100000 [==============================] - 43s - loss: 0.1582
Epoch 279/700
100000/100000 [==============================] - 43s - loss: 0.1560
Epoch 280/700
100000/100000 [==============================] - 43s - loss: 0.1544
Epoch 281/700
100000/100000 [==============================] - 43s - loss: 0.1559
Epoch 282/700
100000/100000 [==============================] - 43s - loss: 0.1565
Epoch 283/700
100000/100000 [==============================] - 43s - loss: 0.1572
Epoch 284/700
100000/100000 [==============================] - 43s - loss: 0.1557
Epoch 285/700
100000/100000 [==============================] - 43s - loss: 0.1561
```

```
Epoch 286/700
100000/100000 [==============================] - 43s - loss: 0.1537
Epoch 287/700
100000/100000 [==============================] - 43s - loss: 0.1537
Epoch 288/700
100000/100000 [==============================] - 43s - loss: 0.1550
Epoch 289/700
100000/100000 [==============================] - 43s - loss: 0.1522
Epoch 290/700
100000/100000 [==============================] - 43s - loss: 0.1499
Epoch 291/700
100000/100000 [==============================] - 43s - loss: 0.1496
Epoch 292/700
100000/100000 [==============================] - 43s - loss: 0.1521
Epoch 293/700
100000/100000 [==============================] - 43s - loss: 0.1471
Epoch 294/700
100000/100000 [==============================] - 43s - loss: 0.1476
Epoch 295/700
100000/100000 [==============================] - 43s - loss: 0.1507
Epoch 296/700
100000/100000 [==============================] - 43s - loss: 0.1490
Epoch 297/700
100000/100000 [==============================] - 43s - loss: 0.1499
Epoch 298/700
100000/100000 [==============================] - 43s - loss: 0.1489
Epoch 299/700
100000/100000 [==============================] - 43s - loss: 0.1508
Epoch 300/700
100000/100000 [==============================] - 43s - loss: 0.1493
Epoch 301/700
100000/100000 [==============================] - 43s - loss: 0.1481
Epoch 302/700
100000/100000 [==============================] - 43s - loss: 0.1489
Epoch 303/700
100000/100000 [==============================] - 43s - loss: 0.1501
Epoch 304/700
100000/100000 [==============================] - 43s - loss: 0.1459
Epoch 305/700
100000/100000 [==============================] - 43s - loss: 0.1477
Epoch 306/700
100000/100000 [==============================] - 43s - loss: 0.1475
Epoch 307/700
100000/100000 [==============================] - 43s - loss: 0.1464
Epoch 308/700
100000/100000 [==============================] - 43s - loss: 0.1454
Epoch 309/700
100000/100000 [==============================] - 43s - loss: 0.1475
Epoch 310/700
100000/100000 [==============================] - 43s - loss: 0.1458
Epoch 311/700
100000/100000 [==============================] - 43s - loss: 0.1457
Epoch 312/700
100000/100000 [==============================] - 43s - loss: 0.1440
Epoch 313/700
100000/100000 [==============================] - 43s - loss: 0.1470
Epoch 314/700
```

```
100000/100000 [==============================] - 43s - loss: 0.1451
Epoch 315/700
100000/100000 [==============================] - 43s - loss: 0.1447
Epoch 316/700
100000/100000 [==============================] - 43s - loss: 0.1450
Epoch 317/700
100000/100000 [==============================] - 43s - loss: 0.1418
Epoch 318/700
100000/100000 [==============================] - 43s - loss: 0.1420
Epoch 319/700
100000/100000 [==============================] - 43s - loss: 0.1412
Epoch 320/700
100000/100000 [==============================] - 43s - loss: 0.1413
Epoch 321/700
100000/100000 [==============================] - 43s - loss: 0.1399
Epoch 322/700
100000/100000 [==============================] - 43s - loss: 0.1427
Epoch 323/700
100000/100000 [==============================] - 43s - loss: 0.1411
Epoch 324/700
100000/100000 [==============================] - 43s - loss: 0.1424
Epoch 325/700
100000/100000 [==============================] - 43s - loss: 0.1421
Epoch 326/700
100000/100000 [==============================] - 43s - loss: 0.1392
Epoch 327/700
100000/100000 [==============================] - 43s - loss: 0.1399
Epoch 328/700
100000/100000 [==============================] - 43s - loss: 0.1402
Epoch 329/700
100000/100000 [==============================] - 43s - loss: 0.1390
Epoch 330/700
100000/100000 [==============================] - 43s - loss: 0.1396
Epoch 331/700
100000/100000 [==============================] - 43s - loss: 0.1387
Epoch 332/700
100000/100000 [==============================] - 43s - loss: 0.1384
Epoch 333/700
100000/100000 [==============================] - 43s - loss: 0.1404
Epoch 334/700
100000/100000 [==============================] - 43s - loss: 0.1350
Epoch 335/700
100000/100000 [==============================] - 43s - loss: 0.1359
Epoch 336/700
100000/100000 [==============================] - 43s - loss: 0.1331
Epoch 337/700
100000/100000 [==============================] - 43s - loss: 0.1362
Epoch 338/700
100000/100000 [==============================] - 43s - loss: 0.1353
Epoch 339/700
100000/100000 [==============================] - 43s - loss: 0.1339
Epoch 340/700
100000/100000 [==============================] - 43s - loss: 0.1329
Epoch 341/700
100000/100000 [==============================] - 43s - loss: 0.1359
Epoch 342/700
100000/100000 [==============================] - 43s - loss: 0.1324
```

```
Epoch 343/700
100000/100000 [==============================] - 43s - loss: 0.1312
Epoch 344/700
100000/100000 [==============================] - 43s - loss: 0.1369
Epoch 345/700
100000/100000 [==============================] - 43s - loss: 0.1340
Epoch 346/700
100000/100000 [==============================] - 43s - loss: 0.1322
Epoch 347/700
100000/100000 [==============================] - 43s - loss: 0.1315
Epoch 348/700
100000/100000 [==============================] - 43s - loss: 0.1332
Epoch 349/700
100000/100000 [==============================] - 43s - loss: 0.1324
Epoch 350/700
100000/100000 [==============================] - 43s - loss: 0.1290
Epoch 351/700
100000/100000 [==============================] - 43s - loss: 0.1321
Epoch 352/700
100000/100000 [==============================] - 43s - loss: 0.1332
Epoch 353/700
100000/100000 [==============================] - 43s - loss: 0.1293
Epoch 354/700
100000/100000 [==============================] - 43s - loss: 0.1317
Epoch 355/700
100000/100000 [==============================] - 43s - loss: 0.1307
Epoch 356/700
100000/100000 [==============================] - 43s - loss: 0.1307
Epoch 358/700
100000/100000 [==============================] - 43s - loss: 0.1312
Epoch 359/700
100000/100000 [==============================] - 43s - loss: 0.1300
Epoch 360/700
100000/100000 [==============================] - 43s - loss: 0.1300
Epoch 361/700
100000/100000 [==============================] - 43s - loss: 0.1269
Epoch 362/700
100000/100000 [==============================] - 43s - loss: 0.1310
Epoch 363/700
100000/100000 [==============================] - 43s - loss: 0.1306
Epoch 364/700
100000/100000 [==============================] - 43s - loss: 0.1289
Epoch 365/700
100000/100000 [==============================] - 43s - loss: 0.1271
Epoch 366/700
100000/100000 [==============================] - 43s - loss: 0.1305
Epoch 367/700
100000/100000 [==============================] - 43s - loss: 0.1287
Epoch 368/700
100000/100000 [==============================] - 43s - loss: 0.1262
Epoch 369/700
100000/100000 [==============================] - 43s - loss: 0.1252
Epoch 370/700
100000/100000 [==============================] - 43s - loss: 0.1269
Epoch 371/700
100000/100000 [==============================] - 43s - loss: 0.1278
Epoch 372/700
```

```
100000/100000 [==============================] - 43s - loss: 0.1253
Epoch 373/700
100000/100000 [==============================] - 43s - loss: 0.1272
Epoch 374/700
100000/100000 [==============================] - 43s - loss: 0.1279
Epoch 375/700
100000/100000 [==============================] - 43s - loss: 0.1245
Epoch 376/700
100000/100000 [==============================] - 43s - loss: 0.1236
Epoch 377/700
100000/100000 [==============================] - 43s - loss: 0.1238
Epoch 378/700
100000/100000 [==============================] - 43s - loss: 0.1252
Epoch 379/700
100000/100000 [==============================] - 43s - loss: 0.1241
Epoch 380/700
100000/100000 [==============================] - 43s - loss: 0.1273
Epoch 381/700
100000/100000 [==============================] - 43s - loss: 0.1226
Epoch 382/700
100000/100000 [==============================] - 43s - loss: 0.1231
Epoch 383/700
100000/100000 [==============================] - 43s - loss: 0.1228
Epoch 384/700
100000/100000 [==============================] - 43s - loss: 0.1252
Epoch 385/700
100000/100000 [==============================] - 43s - loss: 0.1250
Epoch 386/700
100000/100000 [==============================] - 43s - loss: 0.1253
Epoch 387/700
100000/100000 [==============================] - 43s - loss: 0.1199
Epoch 388/700
100000/100000 [==============================] - 43s - loss: 0.1218
Epoch 389/700
100000/100000 [==============================] - 43s - loss: 0.1230
Epoch 390/700
100000/100000 [==============================] - 43s - loss: 0.1200
Epoch 391/700
100000/100000 [==============================] - 43s - loss: 0.1240
Epoch 392/700
100000/100000 [==============================] - 43s - loss: 0.1218
Epoch 393/700
100000/100000 [==============================] - 43s - loss: 0.1196
Epoch 394/700
100000/100000 [==============================] - 43s - loss: 0.1213
Epoch 395/700
100000/100000 [==============================] - 43s - loss: 0.1205
Epoch 396/700
100000/100000 [==============================] - 43s - loss: 0.1223
Epoch 397/700
100000/100000 [==============================] - 43s - loss: 0.1219
Epoch 398/700
100000/100000 [==============================] - 43s - loss: 0.1189
Epoch 399/700
100000/100000 [==============================] - 43s - loss: 0.1219
Epoch 400/700
100000/100000 [==============================] - 43s - loss: 0.1202
```

```
Epoch 401/700
100000/100000 [==============================] - 43s - loss: 0.1214
Epoch 402/700
100000/100000 [==============================] - 43s - loss: 0.1191
Epoch 403/700
100000/100000 [==============================] - 43s - loss: 0.1224
Epoch 404/700
100000/100000 [==============================] - 43s - loss: 0.1192
Epoch 405/700
100000/100000 [==============================] - 43s - loss: 0.1191
Epoch 406/700
100000/100000 [==============================] - 43s - loss: 0.1172
Epoch 407/700
100000/100000 [==============================] - 43s - loss: 0.1152
Epoch 408/700
100000/100000 [==============================] - 43s - loss: 0.1175
Epoch 409/700
100000/100000 [==============================] - 43s - loss: 0.1197
Epoch 410/700
100000/100000 [==============================] - 43s - loss: 0.1196
Epoch 411/700
100000/100000 [==============================] - 43s - loss: 0.1187
Epoch 412/700
100000/100000 [==============================] - 43s - loss: 0.1162
Epoch 413/700
100000/100000 [==============================] - 43s - loss: 0.1183
Epoch 414/700
100000/100000 [==============================] - 43s - loss: 0.1179
Epoch 415/700
100000/100000 [==============================] - 43s - loss: 0.1192
Epoch 416/700
100000/100000 [==============================] - 43s - loss: 0.1169
Epoch 417/700
100000/100000 [==============================] - 43s - loss: 0.1166
Epoch 418/700
100000/100000 [==============================] - 43s - loss: 0.1164
Epoch 419/700
100000/100000 [==============================] - 43s - loss: 0.1174
Epoch 420/700
100000/100000 [==============================] - 43s - loss: 0.1188
Epoch 421/700
100000/100000 [==============================] - 43s - loss: 0.1160
Epoch 422/700
100000/100000 [==============================] - 43s - loss: 0.1144
Epoch 423/700
100000/100000 [==============================] - 43s - loss: 0.1132
Epoch 424/700
100000/100000 [==============================] - 43s - loss: 0.1138
Epoch 425/700
100000/100000 [==============================] - 43s - loss: 0.1147
Epoch 426/700
100000/100000 [==============================] - 43s - loss: 0.1133
Epoch 427/700
100000/100000 [==============================] - 43s - loss: 0.1148
Epoch 428/700
100000/100000 [==============================] - 43s - loss: 0.1144
Epoch 429/700
```

```
100000/100000 [==============================] - 43s - loss: 0.1147
Epoch 430/700
100000/100000 [==============================] - 43s - loss: 0.1107
Epoch 431/700
100000/100000 [==============================] - 43s - loss: 0.1114
Epoch 432/700
100000/100000 [==============================] - 43s - loss: 0.1145
Epoch 433/700
100000/100000 [==============================] - 43s - loss: 0.1119
Epoch 434/700
100000/100000 [==============================] - 43s - loss: 0.1133
Epoch 435/700
100000/100000 [==============================] - 43s - loss: 0.1124
Epoch 436/700
100000/100000 [==============================] - 43s - loss: 0.1129
Epoch 437/700
100000/100000 [==============================] - 43s - loss: 0.1136
Epoch 438/700
100000/100000 [==============================] - 43s - loss: 0.1140
Epoch 439/700
100000/100000 [==============================] - 43s - loss: 0.1118
Epoch 440/700
100000/100000 [==============================] - 43s - loss: 0.1117
Epoch 441/700
100000/100000 [==============================] - 43s - loss: 0.1145
Epoch 442/700
100000/100000 [==============================] - 43s - loss: 0.1127
Epoch 443/700
100000/100000 [==============================] - 43s - loss: 0.1124
Epoch 444/700
100000/100000 [==============================] - 43s - loss: 0.1096
Epoch 445/700
100000/100000 [==============================] - 43s - loss: 0.1122
Epoch 446/700
100000/100000 [==============================] - 43s - loss: 0.1094
Epoch 447/700
100000/100000 [==============================] - 43s - loss: 0.1110
Epoch 448/700
100000/100000 [==============================] - 43s - loss: 0.1113
Epoch 449/700
100000/100000 [==============================] - 43s - loss: 0.1120
Epoch 450/700
100000/100000 [==============================] - 43s - loss: 0.1103
Epoch 451/700
100000/100000 [==============================] - 43s - loss: 0.1099
Epoch 452/700
100000/100000 [==============================] - 43s - loss: 0.1072
Epoch 453/700
100000/100000 [==============================] - 43s - loss: 0.1117
Epoch 454/700
100000/100000 [==============================] - 43s - loss: 0.1086
Epoch 455/700
100000/100000 [==============================] - 43s - loss: 0.1106
Epoch 456/700
100000/100000 [==============================] - 43s - loss: 0.1100
Epoch 457/700
100000/100000 [==============================] - 43s - loss: 0.1116
```

```
Epoch 458/700
100000/100000 [==============================] - 43s - loss: 0.1083
Epoch 459/700
100000/100000 [==============================] - 43s - loss: 0.1089
Epoch 460/700
100000/100000 [==============================] - 43s - loss: 0.1110
Epoch 461/700
100000/100000 [==============================] - 43s - loss: 0.1101
Epoch 462/700
100000/100000 [==============================] - 43s - loss: 0.1092
Epoch 463/700
100000/100000 [==============================] - 43s - loss: 0.1073
Epoch 464/700
100000/100000 [==============================] - 43s - loss: 0.1095
Epoch 465/700
100000/100000 [==============================] - 43s - loss: 0.1105
Epoch 466/700
100000/100000 [==============================] - 43s - loss: 0.1089
Epoch 467/700
100000/100000 [==============================] - 43s - loss: 0.1094
Epoch 468/700
100000/100000 [==============================] - 43s - loss: 0.1093
Epoch 469/700
100000/100000 [==============================] - 43s - loss: 0.1081
Epoch 470/700
100000/100000 [==============================] - 43s - loss: 0.1098
Epoch 471/700
100000/100000 [==============================] - 43s - loss: 0.1085
Epoch 472/700
100000/100000 [==============================] - 43s - loss: 0.1064
Epoch 473/700
100000/100000 [==============================] - 43s - loss: 0.1076
Epoch 474/700
100000/100000 [==============================] - 43s - loss: 0.1084
Epoch 475/700
100000/100000 [==============================] - 43s - loss: 0.1080
Epoch 476/700
100000/100000 [==============================] - 43s - loss: 0.1081
Epoch 477/700
100000/100000 [==============================] - 43s - loss: 0.1046
Epoch 478/700
100000/100000 [==============================] - 43s - loss: 0.1069
Epoch 479/700
100000/100000 [==============================] - 43s - loss: 0.1079
Epoch 480/700
100000/100000 [==============================] - 43s - loss: 0.1048
Epoch 481/700
100000/100000 [==============================] - 43s - loss: 0.1046
Epoch 482/700
100000/100000 [==============================] - 43s - loss: 0.1064
Epoch 483/700
100000/100000 [==============================] - 43s - loss: 0.1067
Epoch 484/700
100000/100000 [==============================] - 43s - loss: 0.1055
Epoch 485/700
100000/100000 [==============================] - 43s - loss: 0.1069
Epoch 486/700
```

```
100000/100000 [==============================] - 43s - loss: 0.1057
Epoch 487/700
100000/100000 [==============================] - 43s - loss: 0.1046
Epoch 488/700
100000/100000 [==============================] - 43s - loss: 0.1054
Epoch 489/700
100000/100000 [==============================] - 43s - loss: 0.1061
Epoch 490/700
100000/100000 [==============================] - 43s - loss: 0.1027
Epoch 491/700
100000/100000 [==============================] - 43s - loss: 0.1019
Epoch 492/700
100000/100000 [==============================] - 43s - loss: 0.1048
Epoch 493/700
100000/100000 [==============================] - 43s - loss: 0.1042
Epoch 494/700
100000/100000 [==============================] - 43s - loss: 0.1052
Epoch 495/700
100000/100000 [==============================] - 43s - loss: 0.1067
Epoch 496/700
100000/100000 [==============================] - 43s - loss: 0.1050
Epoch 497/700
100000/100000 [==============================] - 43s - loss: 0.1027
Epoch 498/700
100000/100000 [==============================] - 43s - loss: 0.1066
Epoch 499/700
100000/100000 [==============================] - 43s - loss: 0.1041
Epoch 500/700
100000/100000 [==============================] - 43s - loss: 0.1031
Epoch 501/700
100000/100000 [==============================] - 43s - loss: 0.1042
Epoch 502/700
100000/100000 [==============================] - 43s - loss: 0.1051
Epoch 503/700
100000/100000 [==============================] - 43s - loss: 0.1045
Epoch 504/700
100000/100000 [==============================] - 43s - loss: 0.1051
Epoch 505/700
100000/100000 [==============================] - 43s - loss: 0.1041
Epoch 506/700
100000/100000 [==============================] - 43s - loss: 0.1005
Epoch 507/700
100000/100000 [==============================] - 43s - loss: 0.1056
Epoch 508/700
100000/100000 [==============================] - 43s - loss: 0.1020
Epoch 509/700
100000/100000 [==============================] - 43s - loss: 0.1032
Epoch 510/700
100000/100000 [==============================] - 43s - loss: 0.1015
Epoch 511/700
100000/100000 [==============================] - 43s - loss: 0.1019
Epoch 512/700
100000/100000 [==============================] - 43s - loss: 0.1013
Epoch 513/700
100000/100000 [==============================] - 43s - loss: 0.1016
Epoch 514/700
100000/100000 [==============================] - 43s - loss: 0.1026
```

```
Epoch 515/700
100000/100000 [==============================] - 43s - loss: 0.1021
Epoch 516/700
100000/100000 [==============================] - 43s - loss: 0.1040
Epoch 517/700
100000/100000 [==============================] - 43s - loss: 0.1038
Epoch 518/700
100000/100000 [==============================] - 43s - loss: 0.1030
Epoch 519/700
100000/100000 [==============================] - 43s - loss: 0.1003
Epoch 520/700
100000/100000 [==============================] - 43s - loss: 0.1011
Epoch 521/700
100000/100000 [==============================] - 43s - loss: 0.1035
Epoch 522/700
100000/100000 [==============================] - 43s - loss: 0.1012
Epoch 523/700
100000/100000 [==============================] - 43s - loss: 0.1007
Epoch 524/700
100000/100000 [==============================] - 43s - loss: 0.0994
Epoch 525/700
100000/100000 [==============================] - 43s - loss: 0.1009
Epoch 526/700
100000/100000 [==============================] - 43s - loss: 0.1008
Epoch 527/700
100000/100000 [==============================] - 43s - loss: 0.1018
Epoch 528/700
100000/100000 [==============================] - 43s - loss: 0.1011
Epoch 529/700
100000/100000 [==============================] - 43s - loss: 0.1013
Epoch 530/700
100000/100000 [==============================] - 43s - loss: 0.0966
Epoch 531/700
100000/100000 [==============================] - 43s - loss: 0.1004
Epoch 532/700
100000/100000 [==============================] - 43s - loss: 0.1000
Epoch 533/700
100000/100000 [==============================] - 43s - loss: 0.0980
Epoch 534/700
100000/100000 [==============================] - 43s - loss: 0.0986
Epoch 535/700
100000/100000 [==============================] - 43s - loss: 0.0991
Epoch 536/700
100000/100000 [==============================] - 43s - loss: 0.1008
Epoch 537/700
100000/100000 [==============================] - 43s - loss: 0.1012
Epoch 538/700
100000/100000 [==============================] - 43s - loss: 0.0981
Epoch 539/700
100000/100000 [==============================] - 43s - loss: 0.1003
Epoch 540/700
100000/100000 [==============================] - 43s - loss: 0.0969
Epoch 541/700
100000/100000 [==============================] - 43s - loss: 0.0995
Epoch 542/700
100000/100000 [==============================] - 43s - loss: 0.0987
Epoch 543/700
```

```
100000/100000 [==============================] - 43s - loss: 0.0991
Epoch 544/700
100000/100000 [==============================] - 43s - loss: 0.0969
Epoch 545/700
100000/100000 [==============================] - 43s - loss: 0.0989
Epoch 546/700
100000/100000 [==============================] - 43s - loss: 0.0963
Epoch 547/700
100000/100000 [==============================] - 43s - loss: 0.0980
Epoch 548/700
100000/100000 [==============================] - 43s - loss: 0.0993
Epoch 549/700
100000/100000 [==============================] - 43s - loss: 0.0989
Epoch 550/700
100000/100000 [==============================] - 43s - loss: 0.0965
Epoch 551/700
100000/100000 [==============================] - 43s - loss: 0.0951
Epoch 552/700
100000/100000 [==============================] - 43s - loss: 0.0986
Epoch 553/700
100000/100000 [==============================] - 43s - loss: 0.0972
Epoch 554/700
100000/100000 [==============================] - 43s - loss: 0.0964
Epoch 555/700
100000/100000 [==============================] - 43s - loss: 0.0980
Epoch 556/700
100000/100000 [==============================] - 43s - loss: 0.0976
Epoch 557/700
100000/100000 [==============================] - 43s - loss: 0.0960
Epoch 558/700
100000/100000 [==============================] - 43s - loss: 0.0957
Epoch 559/700
100000/100000 [==============================] - 43s - loss: 0.0996
Epoch 560/700
100000/100000 [==============================] - 43s - loss: 0.0952
Epoch 561/700
100000/100000 [==============================] - 43s - loss: 0.0973
Epoch 562/700
100000/100000 [==============================] - 43s - loss: 0.0963
Epoch 563/700
100000/100000 [==============================] - 43s - loss: 0.0976
Epoch 564/700
100000/100000 [==============================] - 43s - loss: 0.0980
Epoch 565/700
100000/100000 [==============================] - 43s - loss: 0.0977
Epoch 566/700
100000/100000 [==============================] - 43s - loss: 0.0980
Epoch 567/700
100000/100000 [==============================] - 43s - loss: 0.0956
Epoch 568/700
100000/100000 [==============================] - 43s - loss: 0.0959
Epoch 569/700
100000/100000 [==============================] - 43s - loss: 0.0964
Epoch 570/700
100000/100000 [==============================] - 43s - loss: 0.0963
Epoch 571/700
100000/100000 [==============================] - 43s - loss: 0.0971
```

```
Epoch 572/700
100000/100000 [==============================] - 43s - loss: 0.0956
Epoch 573/700
100000/100000 [==============================] - 43s - loss: 0.0928
Epoch 574/700
100000/100000 [==============================] - 43s - loss: 0.0942
Epoch 575/700
100000/100000 [==============================] - 43s - loss: 0.0965
Epoch 576/700
100000/100000 [==============================] - 43s - loss: 0.0970
Epoch 577/700
100000/100000 [==============================] - 43s - loss: 0.0951
Epoch 578/700
100000/100000 [==============================] - 43s - loss: 0.0947
Epoch 579/700
100000/100000 [==============================] - 43s - loss: 0.0948
Epoch 580/700
100000/100000 [==============================] - 43s - loss: 0.0973
Epoch 581/700
100000/100000 [==============================] - 43s - loss: 0.0950
Epoch 582/700
100000/100000 [==============================] - 43s - loss: 0.0968
Epoch 583/700
100000/100000 [==============================] - 43s - loss: 0.0952
Epoch 584/700
100000/100000 [==============================] - 43s - loss: 0.0932
Epoch 585/700
100000/100000 [==============================] - 43s - loss: 0.0968
Epoch 586/700
100000/100000 [==============================] - 43s - loss: 0.0955
Epoch 587/700
100000/100000 [==============================] - 43s - loss: 0.0930
Epoch 588/700
100000/100000 [==============================] - 43s - loss: 0.0920
Epoch 589/700
100000/100000 [==============================] - 43s - loss: 0.0913
Epoch 590/700
100000/100000 [==============================] - 43s - loss: 0.0935
Epoch 591/700
100000/100000 [==============================] - 43s - loss: 0.0926
Epoch 592/700
100000/100000 [==============================] - 43s - loss: 0.0933
Epoch 593/700
100000/100000 [==============================] - 43s - loss: 0.0947
Epoch 594/700
100000/100000 [==============================] - 43s - loss: 0.0925
Epoch 595/700
100000/100000 [==============================] - 43s - loss: 0.0943
Epoch 596/700
100000/100000 [==============================] - 43s - loss: 0.0933
Epoch 597/700
100000/100000 [==============================] - 43s - loss: 0.0951
Epoch 598/700
100000/100000 [==============================] - 43s - loss: 0.0956
Epoch 599/700
100000/100000 [==============================] - 43s - loss: 0.0896
Epoch 600/700
```

```
100000/100000 [==============================] - 43s - loss: 0.0930
Epoch 601/700
100000/100000 [==============================] - 43s - loss: 0.0909
Epoch 602/700
100000/100000 [==============================] - 43s - loss: 0.0927
Epoch 603/700
100000/100000 [==============================] - 43s - loss: 0.0944
Epoch 604/700
100000/100000 [==============================] - 43s - loss: 0.0920
Epoch 605/700
100000/100000 [==============================] - 43s - loss: 0.0950
Epoch 606/700
100000/100000 [==============================] - 43s - loss: 0.0934
Epoch 607/700
100000/100000 [==============================] - 43s - loss: 0.0921
Epoch 608/700
100000/100000 [==============================] - 43s - loss: 0.0920
Epoch 609/700
100000/100000 [==============================] - 43s - loss: 0.0924
Epoch 610/700
100000/100000 [==============================] - 43s - loss: 0.0915
Epoch 611/700
100000/100000 [==============================] - 43s - loss: 0.0929
Epoch 612/700
100000/100000 [==============================] - 43s - loss: 0.0903
Epoch 613/700
100000/100000 [==============================] - 43s - loss: 0.0915
Epoch 614/700
100000/100000 [==============================] - 43s - loss: 0.0905
Epoch 615/700
100000/100000 [==============================] - 43s - loss: 0.0925
Epoch 616/700
100000/100000 [==============================] - 43s - loss: 0.0904
Epoch 617/700
100000/100000 [==============================] - 43s - loss: 0.0915
Epoch 618/700
100000/100000 [==============================] - 43s - loss: 0.0925
Epoch 619/700
100000/100000 [==============================] - 43s - loss: 0.0902
Epoch 620/700
100000/100000 [==============================] - 43s - loss: 0.0920
Epoch 621/700
100000/100000 [==============================] - 43s - loss: 0.0917
Epoch 622/700
100000/100000 [==============================] - 43s - loss: 0.0901
Epoch 623/700
100000/100000 [==============================] - 43s - loss: 0.0915
Epoch 624/700
100000/100000 [==============================] - 43s - loss: 0.0913
Epoch 625/700
100000/100000 [==============================] - 43s - loss: 0.0917
Epoch 626/700
100000/100000 [==============================] - 43s - loss: 0.0892
Epoch 627/700
100000/100000 [==============================] - 43s - loss: 0.0905
Epoch 628/700
100000/100000 [==============================] - 43s - loss: 0.0911
```

```
Epoch 629/700
100000/100000 [==============================] - 43s - loss: 0.0907
Epoch 630/700
100000/100000 [==============================] - 43s - loss: 0.0905
Epoch 631/700
100000/100000 [==============================] - 43s - loss: 0.0895
Epoch 632/700
100000/100000 [==============================] - 43s - loss: 0.0924
Epoch 633/700
100000/100000 [==============================] - 43s - loss: 0.0895
Epoch 634/700
100000/100000 [==============================] - 43s - loss: 0.0912
Epoch 635/700
100000/100000 [==============================] - 43s - loss: 0.0891
Epoch 636/700
100000/100000 [==============================] - 43s - loss: 0.0896
Epoch 637/700
100000/100000 [==============================] - 43s - loss: 0.0899
Epoch 638/700
100000/100000 [==============================] - 43s - loss: 0.0901
Epoch 639/700
100000/100000 [==============================] - 43s - loss: 0.0901
Epoch 640/700
100000/100000 [==============================] - 43s - loss: 0.0878
Epoch 641/700
100000/100000 [==============================] - 43s - loss: 0.0880
Epoch 642/700
100000/100000 [==============================] - 43s - loss: 0.0899
Epoch 643/700
100000/100000 [==============================] - 43s - loss: 0.0904
Epoch 644/700
100000/100000 [==============================] - 43s - loss: 0.0897
Epoch 645/700
100000/100000 [==============================] - 43s - loss: 0.0887
Epoch 646/700
100000/100000 [==============================] - 43s - loss: 0.0880
Epoch 647/700
100000/100000 [==============================] - 43s - loss: 0.0905
Epoch 648/700
100000/100000 [==============================] - 43s - loss: 0.0857
Epoch 649/700
100000/100000 [==============================] - 43s - loss: 0.0894
Epoch 650/700
100000/100000 [==============================] - 43s - loss: 0.0886
Epoch 651/700
100000/100000 [==============================] - 43s - loss: 0.0868
Epoch 652/700
100000/100000 [==============================] - 43s - loss: 0.0887
Epoch 653/700
100000/100000 [==============================] - 43s - loss: 0.0876
Epoch 654/700
100000/100000 [==============================] - 43s - loss: 0.0877
Epoch 655/700
100000/100000 [==============================] - 43s - loss: 0.0898
Epoch 656/700
100000/100000 [==============================] - 43s - loss: 0.0867
Epoch 657/700
```

```
100000/100000 [==============================] - 43s - loss: 0.0875
Epoch 658/700
100000/100000 [==============================] - 43s - loss: 0.0904
Epoch 659/700
100000/100000 [==============================] - 43s - loss: 0.0888
Epoch 660/700
100000/100000 [==============================] - 43s - loss: 0.0884
Epoch 661/700
100000/100000 [==============================] - 43s - loss: 0.0883
Epoch 662/700
100000/100000 [==============================] - 43s - loss: 0.0913
Epoch 663/700
100000/100000 [==============================] - 43s - loss: 0.0863
Epoch 664/700
100000/100000 [==============================] - 43s - loss: 0.0899
Epoch 665/700
100000/100000 [==============================] - 43s - loss: 0.0885
Epoch 666/700
100000/100000 [==============================] - 43s - loss: 0.0884
Epoch 667/700
100000/100000 [==============================] - 43s - loss: 0.0853
Epoch 668/700
100000/100000 [==============================] - 43s - loss: 0.0870
Epoch 669/700
100000/100000 [==============================] - 43s - loss: 0.0862
Epoch 670/700
100000/100000 [==============================] - 43s - loss: 0.0876
Epoch 671/700
100000/100000 [==============================] - 43s - loss: 0.0871
Epoch 672/700
100000/100000 [==============================] - 43s - loss: 0.0873
Epoch 673/700
100000/100000 [==============================] - 43s - loss: 0.0857
Epoch 674/700
100000/100000 [==============================] - 43s - loss: 0.0868
Epoch 675/700
100000/100000 [==============================] - 43s - loss: 0.0883
Epoch 676/700
100000/100000 [==============================] - 43s - loss: 0.0868
Epoch 677/700
100000/100000 [==============================] - 43s - loss: 0.0855
Epoch 678/700
100000/100000 [==============================] - 43s - loss: 0.0876
Epoch 679/700
100000/100000 [==============================] - 43s - loss: 0.0900
Epoch 680/700
100000/100000 [==============================] - 43s - loss: 0.0840
Epoch 681/700
100000/100000 [==============================] - 43s - loss: 0.0869
Epoch 682/700
100000/100000 [==============================] - 43s - loss: 0.0854
Epoch 683/700
100000/100000 [==============================] - 43s - loss: 0.0867
Epoch 684/700
100000/100000 [==============================] - 43s - loss: 0.0890
Epoch 685/700
100000/100000 [==============================] - 43s - loss: 0.0855
```

```
Epoch 686/700
100000/100000 [==============================] - 43s - loss: 0.0863
Epoch 687/700
100000/100000 [==============================] - 43s - loss: 0.0893
Epoch 688/700
100000/100000 [==============================] - 43s - loss: 0.0853
Epoch 689/700
100000/100000 [==============================] - 43s - loss: 0.0879
Epoch 690/700
100000/100000 [==============================] - 43s - loss: 0.0842
Epoch 691/700
100000/100000 [==============================] - 43s - loss: 0.0864
Epoch 692/700
100000/100000 [==============================] - 43s - loss: 0.0832
Epoch 693/700
100000/100000 [==============================] - 43s - loss: 0.0841
Epoch 694/700
100000/100000 [==============================] - 43s - loss: 0.0850
Epoch 695/700
100000/100000 [==============================] - 43s - loss: 0.0854
Epoch 696/700
100000/100000 [==============================] - 43s - loss: 0.0850
Epoch 697/700
100000/100000 [==============================] - 43s - loss: 0.0859
Epoch 698/700
100000/100000 [==============================] - 43s - loss: 0.0847
Epoch 699/700
100000/100000 [==============================] - 43s - loss: 0.0848
Epoch 700/700
100000/100000 [==============================] - 43s - loss: 0.0852
```

```python
In [121]:  # choose an input sequence and use the prediction function in the previo
           us Python cell to predict 100 characters following it
           # get an appropriately sized chunk of characters from the text
           start_inds = [1006, 2005, 3006]

           # save output
           f = open('text_gen_output/RNN_large_textdata_output.txt', 'w')  # create
            an output file to write too

           # load weights
           model.load_weights(weights_file)
           for s in start_inds:
               start_index = s
               input_chars = text[start_index: start_index + window_size]

               # use the prediction function
               predict_input = predict_next_chars(model,input_chars,num_to_predict
           = 100)

               # print out input characters
               line = '-------------------' + '\n'
               print(line)
               f.write(line)

               input_line = 'input chars = ' + '\n' +  input_chars + '"' + '\n'
               print(input_line)
               f.write(input_line)

               # print out predicted characters
               predict_line = 'predicted chars = ' + '\n' +  predict_input + '"' +
           '\n'
               print(predict_line)
               f.write(predict_line)
           f.close()
```

----

input chars =
in a nature such as his. and yet there was but one woman to him, and th
at woman was the late irene a"

predicted chars =
 smill shoned from his byouks. it was about for sele never scide, from
 the inger to wark all and a c"

----

input chars =
account of his doings: of his summons to odessa in the case of the trep
off murder, of his clearing u"

predicted chars =
p inntrattent vinion. "then, what afairs have any find. you may not alm
es wat once as me ashented. t"

----

input chars =
against the blind. he was pacing the room swiftly, eagerly, with his he
ad sunk upon his chest and hi"

predicted chars =
s righ from his keep and flough." i here the known pontle, and me, at o
pen!' leary, with a buling ov"

In [ ]: