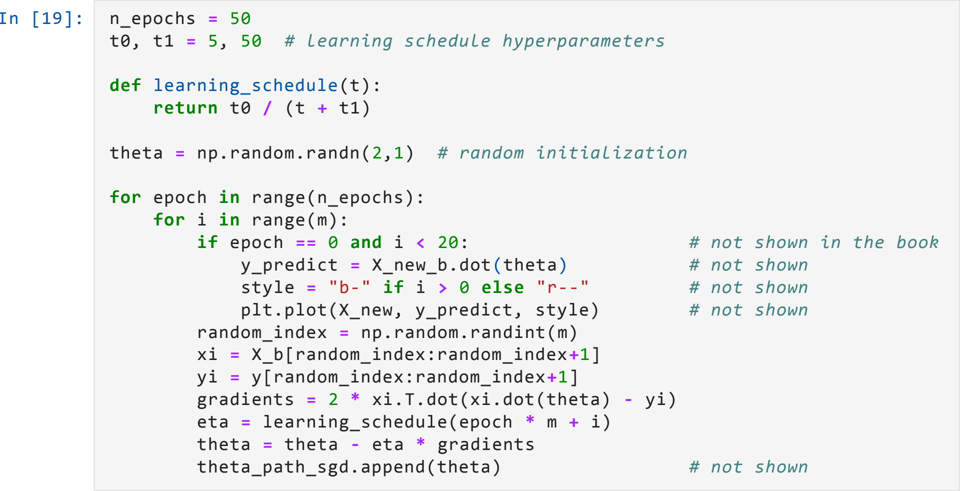

```
In [19]:  n_epochs = 50
          t0, t1 = 5, 50  # learning schedule hyperparameters

          def learning_schedule(t):
              return t0 / (t + t1)

          theta = np.random.randn(2,1)  # random initialization

          for epoch in range(n_epochs):
              for i in range(m):
                  if epoch == 0 and i < 20:                    # not shown in the book
                      y_predict = X_new_b.dot(theta)           # not shown
                      style = "b-" if i > 0 else "r--"         # not shown
                      plt.plot(X_new, y_predict, style)        # not shown
                  random_index = np.random.randint(m)
                  xi = X_b[random_index:random_index+1]
                  yi = y[random_index:random_index+1]
                  gradients = 2 * xi.T.dot(xi.dot(theta) - yi)
                  eta = learning_schedule(epoch * m + i)
                  theta = theta - eta * gradients
                  theta_path_sgd.append(theta)                 # not shown
```