

```

1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3  """
4  Example 2.5
5  Early Stopping
6  Machine Learning for Engineering Problem Solving
7  @author: Austin Downey
8  """
9
10 import IPython as IP
11 IP.get_ipython().run_line_magic('reset', '-sf')
12
13 import numpy as np
14 import matplotlib.pyplot as plt
15 import sklearn as sk
16
17
18 plt.close('all')
19
20 %% build the data sets
21
22 # use 6 to help give a smooth curve that makes the case for early stopping
23 np.random.seed(6)
24
25 m = 20
26 X = 6 * np.random.rand(m, 1) - 3
27 Y = 0.5 * X**2 + X + 2 + np.random.randn(m, 1)
28
29 # plot the data
30 plt.figure()
31 plt.grid(True)
32 plt.scatter(X,Y,color='gray')
33 plt.xlabel('x')
34 plt.ylabel('y')
35
36 X_model = np.linspace(-3,3,num=1000)
37 X_model = np.expand_dims(X_model,axis=1)
38
39 X_train, X_val, y_train, y_val = sk.model_selection.train_test_split(X, Y, test_size=0.2)
40
41 %% perform early stopping
42
43
44 # prepare the data
45 poly_scaler = sk.pipeline.Pipeline([("poly_features", sk.preprocessing.PolynomialFeatures
46 (
47     degree=90, include_bias=False)), ("std_scaler", sk.preprocessing.StandardScaler())])
48 X_train_poly_scaled = poly_scaler.fit_transform(X_train)
49 X_val_poly_scaled = poly_scaler.fit_transform(X_val)
50
51 # set up the model, not that by setting max_iter=1 it will only train one epoch
52 model = sk.linear_model.SGDRegressor(max_iter=1, tol=0, learning_rate="constant"
53 ,eta0=0.0005,penalty=None,warm_start=True)
54
55 # Train the model in a loop to build the data set to investigate the benefit of early
56 stopping
57 val_errors = []
58 train_errors = []
59 for epoch in range(1000):
60     model.fit(X_train_poly_scaled, y_train.ravel()) # continues where it left off
61     y_val_predict = model.predict(X_val_poly_scaled) # Predict the target values
62     y_train_predict = model.predict(X_train_poly_scaled) # Predict the target values
63     val_error = sk.metrics.mean_squared_error(y_val, y_val_predict) # Calculate error
64     train_error = sk.metrics.mean_squared_error(y_train.ravel(), y_train_predict) #
65     Calculate error
66     val_errors.append(val_error)

```

```
65     train_errors.append(train_error)
66
67     # plot the early learning curves, you may have to plot this a few times to get
68     # a set of curves that shows strong results
69     plt.figure()
70     plt.grid(True)
71     plt.plot(val_errors, label='validation data')
72     plt.plot(train_errors, '--', label='training data')
73     plt.ylabel('RMSE')
74     plt.xlabel('epoch')
75     plt.legend()
76
77
78
79
80
81
```