6.036: Machine Learning

Ryan Lacey <rlacey@mit.edu>

February 28, 2014

1. (a) For each next x there is no component of θ in the direction of x, so the dot product is zero. Perceptron classifies this as a mistake. This will occur d times, each time fixing updating θ to correct the new mistake, which will not have to be fixed again. The feature vector ordering does matter because the dot product of any previously unseen x with θ is zero until θ converges to a vector of ones.

   (b) θ converges to a vector of ones, ie. $[1_0, 1_1, 1_2, ..., 1_d]$

   (c) Bound on number of mistakes is $\frac{R^2}{\gamma^2}$

   $R \geq ||x^{(i)}||$
   $||x^{(i)}|| = 1$

   $\gamma = ||\theta||^{-1}$
   $\gamma = \sqrt{d}^{-1}$

   $$\frac{R^2}{\gamma^2} = \frac{1}{\sqrt{d}^{-2}} = d$$

   We respect our mistakes bound established in class and hit the maximum number of possible mistakes in this arrangement.

   (d) No, the number of mistakes will not depend upon the feature vector ordering. The passive aggressive function will update and thereafter correctly classify one point at a time, thereby requiring d mistakes before all points are correctly classified. In the end we will converge to the same θ as perceptron. The hinge loss will be one until this convergence, for any ordering of the feature vectors, so passive aggressive will run analogous to the perceptron algorithm.

2. (a)

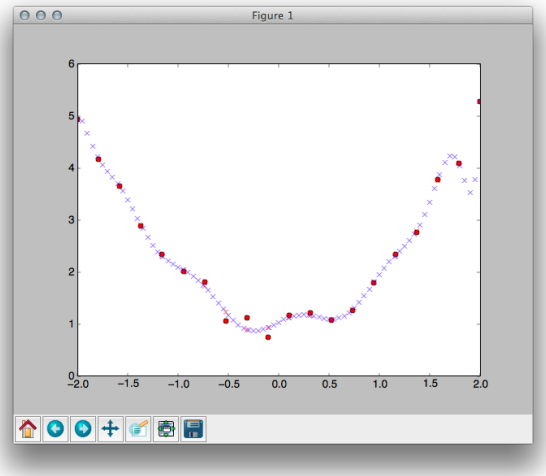

Figure 1: Line overfitting to training data points when regression parameters set to `noisy_quad_fit(13, 0.00001)`

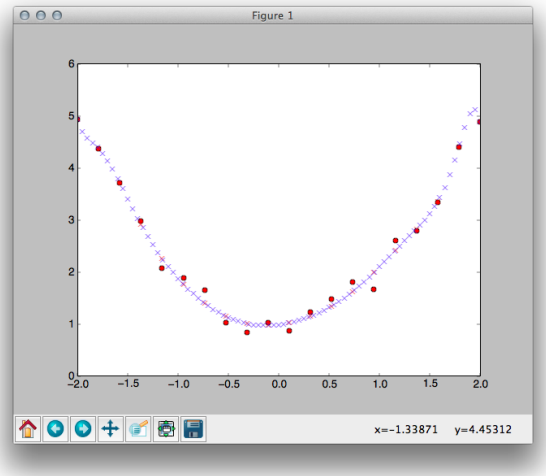

Figure 2: Line that generalizes well to test data when regression parameters set to `noisy_quad_fit(13, 0.1)`

(b)

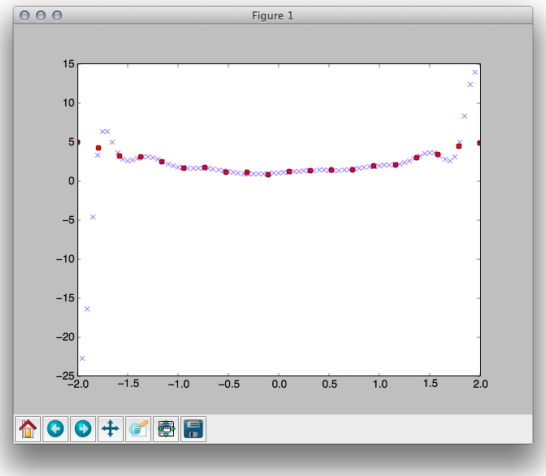

Figure 3: Minimum training error = 0.0827554694891 at `order=18` and `regularization = 1e-06`

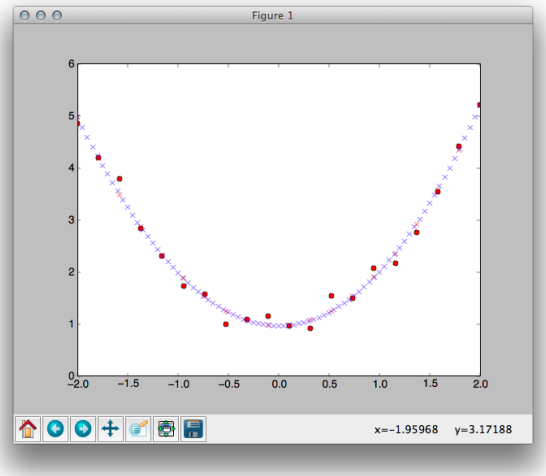

Figure 4: Minimum testing error = 0.0397647479675 at `order=3` and `regularization = 1e-06`

(c) Sinusoid with noise

Minimum training error = 0.0702075371943 at `order=17` and `regularization = 5e-06`
Minimum testing error = 0.141655026216 at `order=10` and `regularization = 0.01`

(d) No noise regressions

Polynomial
Minimum training error = 7.00892770489e-08 at `order=2` and `regularization = 1e-06`
Minimum testing error = 7.02414684195e-08 at `order=2` and `regularization = 1e-06`

Sinusoid
Minimum training error = 0.0000484999916992 at `order=13` and `regularization = 1e-06`
Minimum testing error = 0.000132456013403 at `order=13` and `regularization = 1e-06`

Without noise the training and test data sets had the same optimal order and regularization parameter. This make sense because the points for both data sets were created in the same manner (an even distribution over some predefined space). The small, arguably negligible, differences in the error between the test and the training sets arises from the fact that they contained different quantities of points. Note the order of the polynomial is 2, which should be expected because the `vander` operation that determined the point locations results in a parabolic curve. It is also worth noting that although the error for the sinusoid is small, it is several magnitudes greater than that of the polynomial. This is because we use a polynomial to emulate the curvature of a sinusoid, which quickly plateaus in its ability to trace without being of a very high order.