Homework #1 B

Spring 2020, CSE 446/546: Machine Learning
Dino Bektesevic

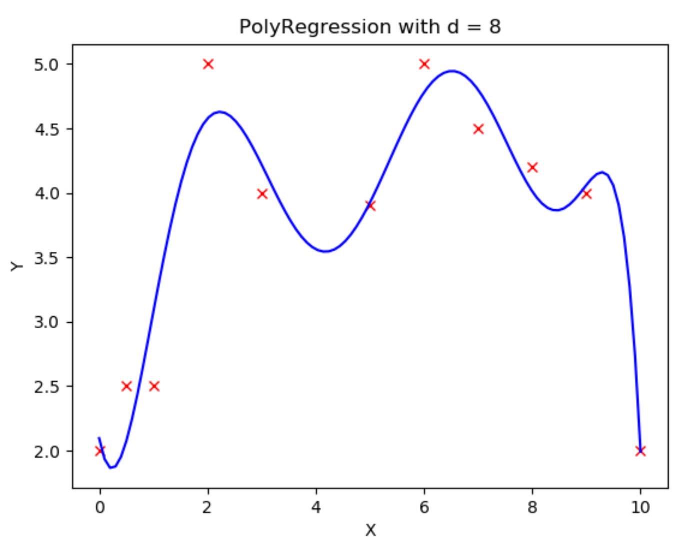
Ridge regression on MNIST

B.2

a. *[10 points]* We just fit a classifier that was linear in the pixel intensities to the MNIST data. For classification of digits the raw pixel values are very, very bad features: it's pretty hard to separate digits with linear functions in pixel space. The standard solution to this is to come up with some transform $h : \mathbb{R}^d \to \mathbb{R}^p$ of the original pixel values such that the transformed points are (more easily)linearly separable. In this problem, you'll use the feature transform:

$$h(x) = \cos(Gx + b)$$

where $G \in R^{p \times d}$, $b \in \mathbb{R}^p$, and the cosine function is applied element wise. We'll choose G to be a random matrix, with each entry sampled i.i.d. from a Gaussian with mean $\mu = 0$ and variance $\sigma^2 = 0.1$, and b to be a random vector sampled i.i.d. from the uniform distribution on $[0, 2\pi]$. The big question is: how do we choose p? Using cross-validation, of course! Randomly partition your training set into proportions 80/20 to use as a new training set and validation set, respectively. Using the train function you wrote above, train $\widehat{W}_p$ for different values of p and plot the classification training error and validation error on a single plot with p on the x-axis. Be careful, your computer may run out of memory and slow to a crawl if p is too large ($p \leq 6000$ should fit into 4 GB of memory that is a minimum for most computers, but if you're having trouble you can set p in the several hundreds). You can use the same value of $\lambda = 1e - 4$ as above but feel free to study the effect of using different values of λ and σ^2 for fun.



b. *[5 points]* Instead of reporting just the test error, which is an unbiased estimate of the true error, we would like to report a confidence interval around the test error that contains the true error.

Lemma 1. (Hoeffding's inequality) Fix $\delta \in (0,1)$. If for all $i = 1, \ldots, m$ we have that X_i are i.i.d.random variables with $X_i \in [a,b]$ and $\mathbb{E}[X_i] = \mu$ then

$$P\left(\left|\left(\frac{1}{m}\sum_{i=1}^{m} X_i\right) - \mu\right| \geq \sqrt{\frac{(b-a)^2 \log(2/\delta)}{2m}}\right) \leq \delta$$

We will use the above equation to construct a confidence interval around the true classification error $(\hat{f} = \mathbb{E}_{\text{test}}[\widehat{\epsilon}_{\text{test}}(\hat{f})]$ since the test error $\widehat{\epsilon}_{\text{test}}(\hat{f})$ is just the average of indicator variables taking values in $\{0,1\}$ corresponding to the i-th test example being classified correctly or not, respectively, where an error happens with probability $\mu = \epsilon(\hat{f}) = \mathbb{E}_{\text{test}}[\widehat{\epsilon}_{\text{test}}(\hat{f})]$, the true classification error. Let $\hat{p}$ be the value of p that approximately minimizes the validation error on the plot you just made and use $\hat{f}(x) = \text{argmax}_j\, x^T \widehat{W}_{\hat{p}} e_j$ to compute the classification test error $\widehat{\epsilon}_{\text{test}}(\hat{f})$. Use Hoeffding's inequality, above, to compute a confidence interval that contains $\mathbb{E}_{\text{test}}[\widehat{\epsilon}_{\text{test}}(\hat{f})]$ (i.e., the true error) with probability at least 0.95 (i.e. $\delta = 0.05$). Report $\widehat{\epsilon}_{\text{test}}(\hat{f})$ and the confidence interval.