Assignment 2
Report

An introduction into logistic regression, support vector machines, and k-fold cross-validation.

Student:

Amaan Makhani

[redacted]

Teacher:

Nishant Mehta

Course:

Seng 474 - Data Mining

1. Introduction

In this report, we will be using the fashion-mst data set. To reduce the number of samples, only class 5 (sneaker) and class 7 (sandal) were used. Each class has approximately 6000 data points for training and 1000 data points for testing. As part of pre-processing, as suggested by Nishant, the pixel values were scaled by 1/255. This report uses 2000 training examples for each class to reduce the training time of the models being used. This training size is used throughout the tasks completed in this report. An eight k-fold cross-validation technique was used in multiple experiments to validate and tune parameters.

2. Task One

This task consisted of varying the l2 regularization parameter C of the logistic regression model that was implemented. The graph of the test error and training error is shown below in ***figure 1***.

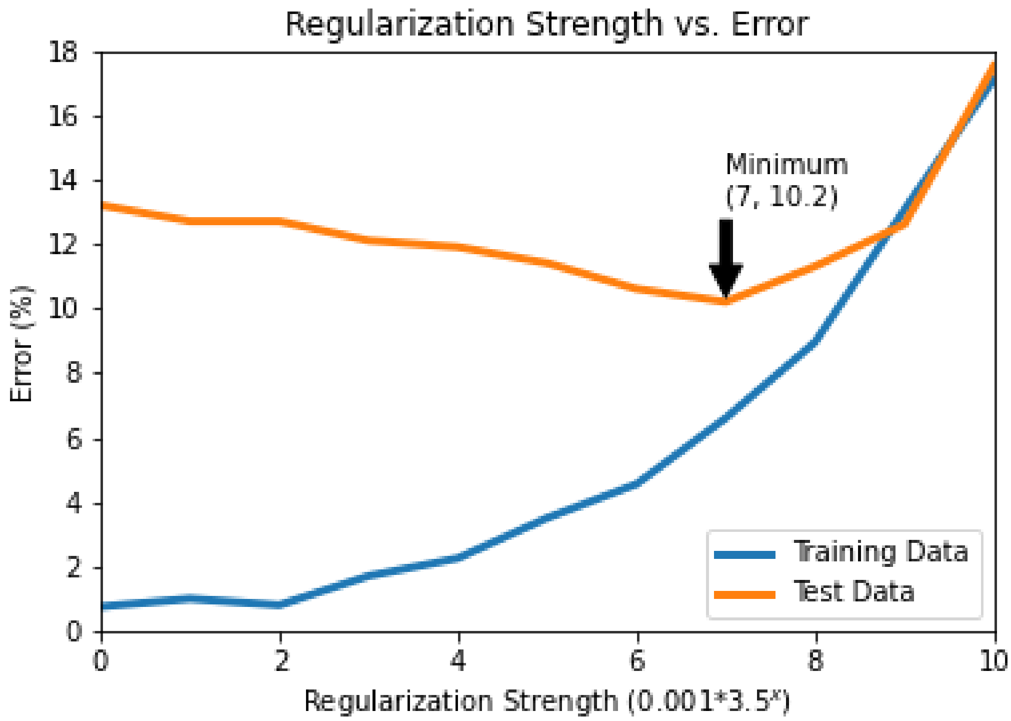


Figure 1: The effect of varying the regularization strength for logistic regression.

The regularization strength shown in the figure above is inversely proportional to the value of C used in the training objective equation below. Therefore, the higher the regularization strength, the more regularization applied, i.e. less emphasis on the training error. Since it is more intuitive to discuss regularization in terms of strength, this report will be using regularization strength rather than referring to the value of C.

$$\min_{w\in\mathbb{R}^d, b\in\mathbb{R}} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \ell\Big(y_i, \sigma(\langle w, x_i\rangle + b)\Big),$$

The values used for the regularization strength were exponentially spaced with an initial value of 0.001 and a base of 3.5. The exponent for the base was then varied.

As shown in ***figure 1***, we can see that the training set's initial error was roughly one percent until the exponent was higher than two. Too little regularization results in very low training error while the test error is relatively high; this can be seen within this first region of the graph. Before an exponent of two, you can see the test and training errors flatten with small variations as the exponent gradually increases. That is why this region is one of too little regularization as more regularization is needed to generalize the dataset better.

After this point in the training error increases and the test error decreases. The test error is seen to decrease at a lower rate than the training error increase. This change in both errors continues until the minimum test error at an exponent value of 7 is reached. The distance between both values is seen to be getting smaller as the exponent increases. This region of the graph can be seen as one in which the regularization strength allows for better generalization of the dataset, thus preventing the overfitting of the training examples.

After this range, both errors continually increase at a similar rate, and their error values do not decrease after this. This region is an error of underfitting the training data as the regularization strength prevents enough emphasis to be placed on the training examples. Regularization strength decreases the emphasis placed on the training examples; however, fitting and training the model is needed otherwise the model won't be able to converge/separate the data precisely. For this dataset and this regularization parameter value range, the best regularization strength is 6.43. This is equivalent to a C value of 0.15.

3. Task Two

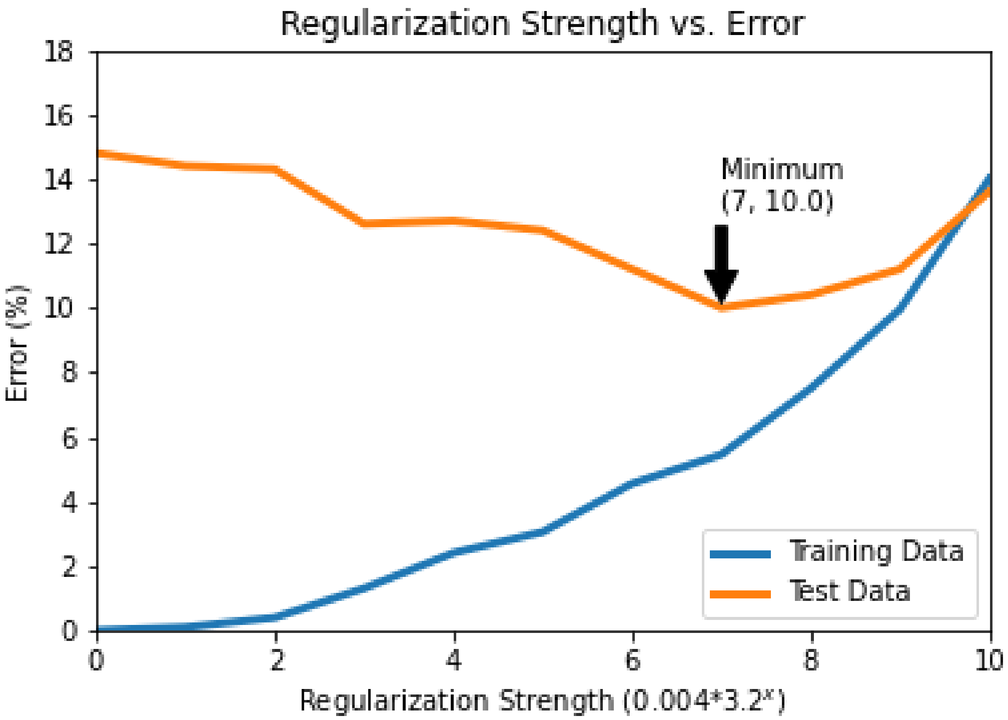


Figure 2: The effect of varying the regularization strength for SVM's.

Similar to task one, the values used for the regularization strength were exponentially spaced with an initial value of 0.004 and a base of 3.2. The exponent on the base was then varied. As shown in ***figure 2***, we can see the initial error of the training set was roughly zero percent until the exponent was higher than two. Too little regularization results in very low training error while the test error is relatively high; this can be seen within this first region of the graph. Before an exponent of two, you can see the test and training errors flatten with small variations as the exponent gradually increases. That is why this region is one of too little regularization as more regularization is needed to generalize the dataset better.

After this point in the training, error increases and the test error decreases; the test error is seen to decrease in a step-like fashion indicating it is possibly reaching local maximums and continuing its quest for a global maximum. Knowing this, we can infer there are more local minimums that an SVM can reach compared to a logistic regression model. This decline in both errors continues until the minimum test error at an exponent value of 7 is reached. This region of the graph is one in which the regularization strength is allowing for better generalization of the dataset, thus preventing the overfitting of the training examples.

After this range, both errors continually increase at a similar rate starting around an exponent value of nine. Both the training and test error do not decrease after this. This region is an error of underfitting of the training data as the regularization strength restricts the model to place enough emphasis on the training examples. Regularization strength decreases the emphasis placed on the training examples; however, fitting and training the model is needed, or else it won't be able to converge/separate the data precisely. For this dataset and this regularization parameter value range, the best regularization strength is 15.46. This is equivalent to a C value of 0.06. This minimum test error is reached at a regularization strength that is nearly double the one used in logistic regression model for task one. This implies that more regularization is needed for linear SVM's. More regularization will generally create simpler decision functions. In the case of SVM's, this is even more significant as the soft support vectors in a linear model tradeoff misclassification for a larger margin. If the soft vectors are to sensitive to the training data, they could move away from the optimal support vector locations. The regularization strength will allow for a greater misclassification rate of the training example.

4. Task Three

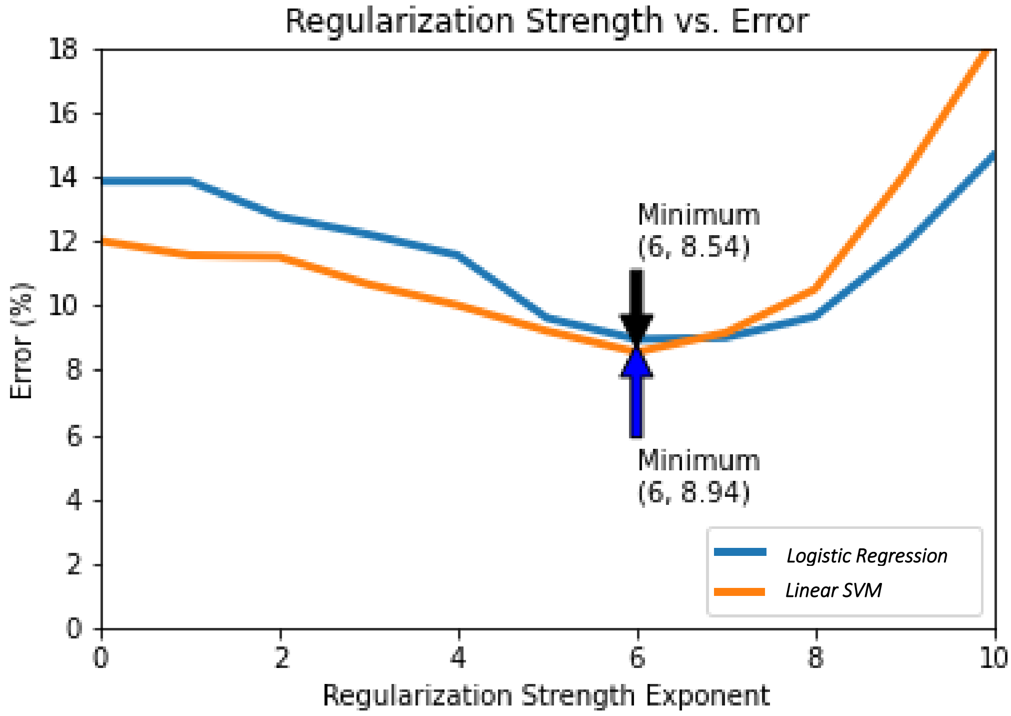
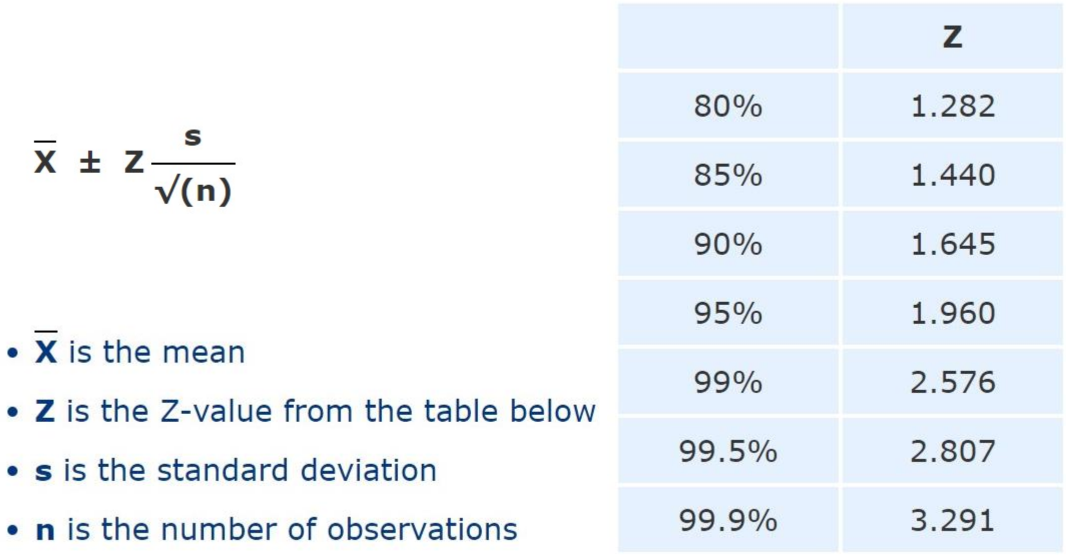


Figure 3: K-fold cross-validation of regularization parameters for logistic regression and SVM's, where k-fold was 8.

In task 3, k-fold cross-validation was performed for both the logistic regression model and SVM model in order to determine their optimal regularization strength. The regularization strength values used the same exponentially spaced values described in task one and two for their respective models. A k-fold of 8 was used, and their optimal were then used on the entire test and training data to determine their optimized error. The optimized logistic regression model produced a test error of 10.5%, and the optimized SVM model produced an error of 11.2%. These error values are relatively close to one another and seem to produce a fair comparison. When compared to previous results obtained in task one and two, the k-fold cross-validation technique found the optimized values of the regularization strength were lower than previously found. This can be seen ***in figure 3***, which plots the k-fold cross-validation error for each regularization strength value. The minimum test error in both ***figure 1 and 2*** is higher than the test error obtained through k-fold cross-validation. However, the optimized values produced a higher test error than the error value achieved in task one and two. This regularization value may produce the best k-fold cross-validation error but not necessarily the lowest error on the full dataset. This could be because the test data not used for k-fold cross-validation has a different distribution than the training data.



$$\overline{X} \pm Z\frac{s}{\sqrt{(n)}}$$

- $\overline{\mathbf{X}}$ is the mean
- **Z** is the Z-value from the table below
- **s** is the standard deviation
- **n** is the number of observations

	Z
80%	1.282
85%	1.440
90%	1.645
95%	1.960
99%	2.576
99.5%	2.807
99.9%	3.291

Figure 4: Confidence interval relationship to test size.

In terms of confidence intervals, since the test size is 4000/8 = 500, we can see that if we had a larger test size to represent the population, we could construct a more accurate confidence interval. To view this, if we want to build a 95% confidence level with a confidence interval of 4%, we will need 522 samples. However, if we create a similar confidence level with an interval of 3%, we will need 843 samples. Since the difference in optimal test errors is small, a more accurate confidence interval is important to accurately represent the population. A larger test size can also be obtained by reducing the number folds used. The fold value required to achieve a test set of 843 samples is roughly four which is not ideal. So, a better solution to decreasing the confidence interval is to increase the samples from the reduced dataset. This means you could increase the 4000 samples in total to around 6750.

5. Task Four

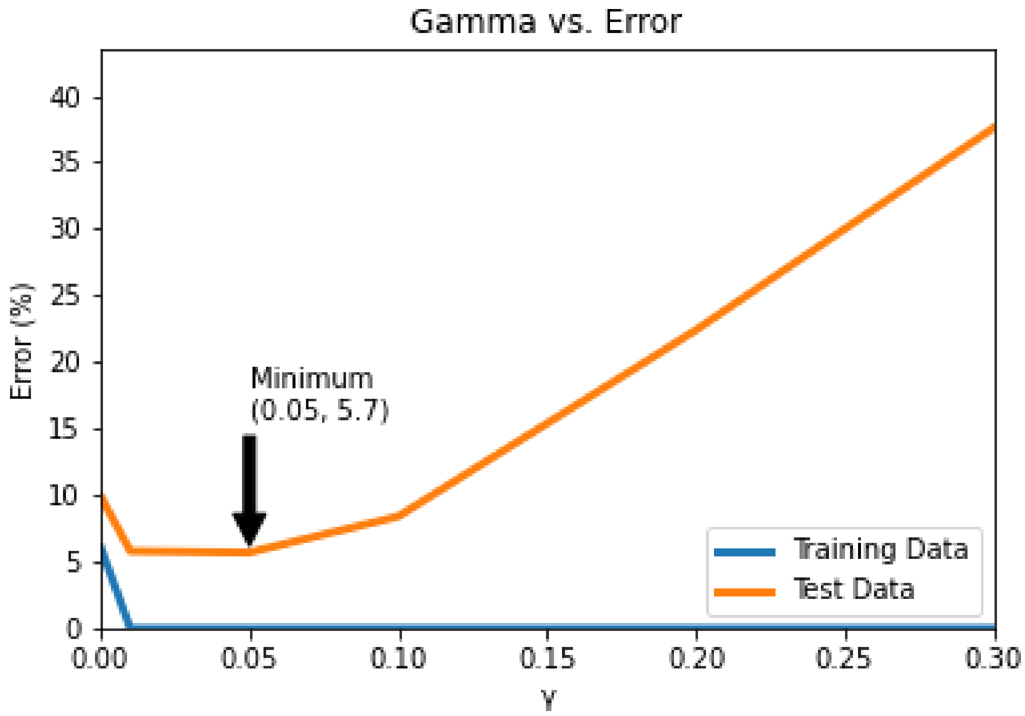


Figure 5: SVM with gaussian kernel using gamma values and optimized regularization strength.

This task used a set of gammas and determined its optimal regularization strength for each using k-fold cross-validation, where k was 8. This SVM used a Gaussian kernel rather than a linear kernel. Once the corresponding pairs were found, the pairs were used to find the test error on the entire dataset. Those values were then plotted in ***figure 5***. Compared to a linear kernel, the minimum test error was 3% lower. This is a substantial difference. This difference is a result of the Gaussian kernel allowing a more curved decision function and the impact gamma has on these support vectors. I was surprised as to how low the best gamma values for this dataset were. The default value used with many datasets in online experiments was around 1.0, however, in a few reports when the grid search function was run it found the optimal gamma tended to be 0.1. The best values of gamma were chosen because of multiple prior experiences. As seen in ***figure 5***, after a gamma value of 0.1, the test error grows significantly at a high rate. The best gamma value was found to be 0.05. Gamma defines how far the influence of a training example reaches. A smaller value of gamma indicates an examples influence is far while a larger gamma value indicates a close influence. This minimum error indicates due to gamma the span of the points that influence the support vectors is far. The overfitting due to gamma started around 0.1 as the training error didn't change while the test error decreased at a high linear rate. In other online experiments, I was aware gamma values greater than one often leads to overfitting, but the gamma values in my experiments began overfitting at a value of 0.1.

Gamma	Optimal Regularization Strength
0.0001	0.0045
0.01	0.0461
0.05	0.0045
0.1	0.4718
0.2	0.0045
0.3	0.0045

Figure 6: Table of gamma values and its corresponding optimal regularization strength.