David Felipe Alvear Goyes

1. Binary logistic regression with SGD.
In this exercise, we consider a logistic regression classifier of a two-class subset of the Iris data set. In logistic regression, we use a linear predictor of the form $x^T w + b$, which we write as $x^T\theta$. Here $\theta = [b\ w]^T$, where we assume that the bias parameter has been absorbed in θ and the set of features x of the sample data includes a corresponding "1" feature. The prediction $x^T\theta$ is transformed into the probability of belonging to the $y = 1$ class through the logistic function.

You will find on Blackboard a starter file that generates a data set of $N = 100$ samples, extracted from the original Iris data set we have used previously.

- write down the expression for the cross entropy loss function and, in a short sentence, justify why this is a reasonable loss function.
- divide the samples from the two classes randomly into a training and testing subsets (80/20).
- write a loss function for a minibatch of data points.
- using the training data, set up an appropriate SGD optimization loop to find the optimal parameters of the logistic regression classifier. Use minibatches of size $B = 10$ random data samples for generating the stochastic gradient at every iteration.
- plot (semilog) the history of the loss on the training and testing data subsets vs iteration count.
- evaluate the quality of the classifier by generating a confusion matrix of the testing data. Comment.

Logistic Regression clasifier

Predictor $w^T x + b \rightarrow x^T\theta$

→ write down expression for cross entropy loss:

$$H(y_n, \hat{y}_n) = -\left(y_n \log(\hat{y}_n) + (1-y_n)\log(1-\hat{y}_n)\right)$$

* y_n → true label

* $\hat{y}_n$ → Predicted label

→ The cross entropy defined as: $H(p,q) = -\sum_i p_i \log(q_i)$, where p_i is the true probability of event i occurring. q_i is the estimated probabity. $H(p,q)$ measure the surprise of an event. Then if the true distribution match the estimated probability then the cross entropy is minimal, Meaning that the surprise of that event is minimal. However, if the true and estimated distributions differ there will be a big Surprise resulting in a high cross entropy.

→ $\log(y_n)$ or $\log(q)$ cuantify the surprise of an event. Then it's weighted multiplying by the estimated probability. Taking the negative sign ensure that incorrect predictions (high surprise) are penalized.

2. Multiclass logistic regression.
In this exercise we generalize the binary logistic classification to the 3-class classification of the complete Iris dataset.

- the loss/objective function for multiclass logistic regression may be written as

$$J(W) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{c=1}^{C} y_{nc}\log \hat{y}_{nc}$$

  where W is the $C \times D$ matrix of unknown coefficients (where we assume that a "1" feature has been added to every data sample to obtain $D = 5$ features), $\hat{y}_n = \mathcal{S}(Wx_n)$, and $\mathcal{S}$ is the softmax function. Explain very briefly the rationale for this loss function.

- write an SGD optimization loop to find the parameters W of the multiclass logistic regression for the Iris data set. Can you give an interpretation for the 3 rows of W?
- perform a validation study of your classifier by generating and evaluating a confusion matrix on test data (not used in training)
- suppose we are interested in finding out which features are most important to our classification. We may do so by adding a regularization term that attempts to select the fewest features to use in the classification. Write a mathematical expression for such a regularization term, and describe in a few words what it attempts to do.

Multiclass Logistic Regression

→ $C \times D$: C classes, D Features

→ $$\hat{y}_{nc} = P_c(x_n, \theta) = \frac{\exp(\theta_{c,:} x_n)}{\sum_{l=1}^{C} \exp(\theta_{l,:} x_n)}$$

$$J(\theta) = \frac{1}{N}\sum_{n=1}^{N} \mathbb{H}(y_n, \hat{y}_n) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{c=1}^{C} y_{nc} \log P_c(x_n;\theta)$$ → Loss Function

→ The loss function for multiclass logistic regression extend the loss Fn of binary logistic Regression to handle multiple classes. This loss Measure how well the predicted probability distribution aligns with the true distribution for each class. The value of the loss will be high Penalizing when the model assigns a low probability to a correct class.

→ Interpretation of Rows in W

- → Each row of W Represent the weights for each class. Multiplying W with the data will give the score or log-odds for that particular class. The rows of W help to determine how likely is that Point to a Particular class.

→ LASSO Regularization

$$L1 = \lambda \|w\|_1 = \lambda \sum_{I=1}^{D} |w_I|$$

- → L1 Regularization encorage many of the feature weights to be zero. It serve as a feature selector for weights that are not beneficial, relevant.
- → This regularization also produce sparse solutions.