## Assignment 1

**Team member 1:**

**Name:** Varun Vaddi

**PeopleSoft number:** 2347481

**Name on Kaggle leaderboard:** Vamsi\_n\_Varun

**Contribution Description:** Developed the code along with Vamsi for both perceptron and SoftMax. Ran 3 models of Perceptron & 4 models of SoftMax (but when merging both the code models, ran into issue where the name of all 8 files of SoftMax are named as vyeruban, couldn’t re-run due to time constraints). Prepared the Report document for submission.

**Team member 2:**

**Name:** Vamsi Krishna Yerubandi

**PeopleSoft number:** 2351142

**Name on Kaggle leaderboard:** Vamsi\_n\_Varun

**Contribution Description:** Developed the code along with Varun for both perceptron and SoftMax. Ran 3 models of Perceptron & 4 models of SoftMax. Prepared the Kaggle submission and Zip file.

**Theory**

|  |  |
| --- | --- |
| A1 | * Perceptron * Logistic Regression * Stochastic Gradient Descent |
| A2 | **Gradient Descent:**  -It computes Gradient of Loss function for the whole training dataset.  - As a result, convergence is slow, as it need to run whole dataset for each iteration.  - Expensive and takes more time to process for large datasets.  **Stochastic Gradient Descent (SGD):**  -It computes Gradient of Loss function for the randomly selected subset of the training dataset.  - Convergence is quicker.  -Less time to process large datasets, as it only considers a small subset of data.  In simple terms, Gradient Descent is more stable but slower, while SGD is quicker but can be a bit more unpredictable. |
| A3 | Binary Cross-entropy Loss/ Log Loss  A white paper with writing on it  Description automatically generated |
| A4 | Derive the SGD for Log Reg  A white paper with writing on it  Description automatically generated  A piece of paper with writing on it  Description automatically generated |

**Perceptron:**

You should also mention whether adding a learning rate decay helped and how you implemented this decay.

**Experimentation**

|  |  |
| --- | --- |
| Experiment # | 1 |
| Hyperparameters | Learning rate = 0.01, epoch = 200, decay = 0.8 |
| Training accuracy | 39.47 |
| Validation accuracy | 28.1 |
| Test accuracy | 30.94 |
| Loss vs number of iterations plot using matplotlib | A graph with numbers and a line  Description automatically generated |
| Training and validation accuracy vs number of iterations plot | A graph with numbers and a line  Description automatically generated |

|  |  |
| --- | --- |
| Experiment # | 2 |
| Hyperparameters | Learning rate = 0.1, epoch = 200, decay = 0.8 |
| Training accuracy | 39.38 |
| Validation accuracy | 27.8 |
| Test accuracy | 30.98 |
| Loss vs number of iterations plot using matplotlib | A graph with numbers and a line  Description automatically generated |
| Training and validation accuracy vs number of iterations plot | A graph with numbers and lines  Description automatically generated |

|  |  |
| --- | --- |
| Experiment # | 3 |
| Hyperparameters | Learning rate = 0.5, epoch = 200, decay = 0.8 |
| Training accuracy | 39.57 |
| Validation accuracy | 29.7 |
| Test accuracy | 30.92 |
| Loss vs number of iterations plot using matplotlib | A graph with numbers and a line  Description automatically generated |
| Training and validation accuracy vs number of iterations plot | A graph with numbers and a line  Description automatically generated |

|  |  |
| --- | --- |
| Experiment # | 4 |
| Hyperparameters | Learning rate = 0.01, epoch = 100, decay = 0.2 |
| Training accuracy | 40.25 |
| Validation accuracy | 28.3 |
| Test accuracy | 30.18 |
| Loss vs number of iterations plot using matplotlib | A graph with numbers and a line  Description automatically generated |
| Training and validation accuracy vs number of iterations plot | A graph with numbers and lines  Description automatically generated |

|  |  |
| --- | --- |
| Experiment # | 5 |
| Hyperparameters | Learning rate = 0.1, epoch = 100, decay = 0.2 |
| Training accuracy | 40.17 |
| Validation accuracy | 28.6 |
| Test accuracy | 30.56 |
| Loss vs number of iterations plot using matplotlib | A graph with numbers and a line  Description automatically generated |
| Training and validation accuracy vs number of iterations plot | A graph with numbers and lines  Description automatically generated |

|  |  |
| --- | --- |
| Experiment # | 6 |
| Hyperparameters | Learning rate = 0.5, epoch = 100, decay = 0.2 |
| Training accuracy | 40 |
| Validation accuracy | 29.7 |
| Test accuracy | 29.92 |
| Loss vs number of iterations plot using matplotlib | A graph with numbers and a line  Description automatically generated |
| Training and validation accuracy vs number of iterations plot | A graph with numbers and lines  Description automatically generated |

**Hyperparameters of Kaggle model**

|  |  |
| --- | --- |
| Optimal hyperparameters | Learning rate = 0.1, epoch = 200, decay = 0.8 |
| Training accuracy | 39.38 |
| Validation accuracy | 27.8 |
| Test accuracy | 30.98 |
| Loss vs number of iterations plot | A graph with numbers and a line  Description automatically generated |
| Training and validation accuracy vs number of iterations plot | A graph with numbers and lines  Description automatically generated |

**Softmax:**

Also report your training, validation, and testing accuracy with your optimal hyperparameter setting.

**Experimentation**

|  |  |
| --- | --- |
| Experiment # | 1 |
| Hyperparameters | Learning rate = 0.01, epoch = 100, regularization constant = 0.01 |
| Training accuracy | 24.25 |
| Validation accuracy | 23.59 |
| Test accuracy | 23.3 |
| Loss vs number of iterations plot using matplotlib |  |
| Training and validation accuracy vs number of iterations plot |  |

|  |  |
| --- | --- |
| Experiment # | 2 |
| Hyperparameters | Learning rate = 0.01, epoch = 100, regularization constant = 0.05 |
| Training accuracy | 23.98 |
| Validation accuracy | 26 |
| Test accuracy | 23.7 |
| Loss vs number of iterations plot using matplotlib |  |
| Training and validation accuracy vs number of iterations plot |  |

|  |  |
| --- | --- |
| Experiment # | 3 |
| Hyperparameters | Learning rate = 0.01, epoch = 200, regularization constant = 0.01 |
| Training accuracy | 24.14 |
| Validation accuracy | 24.09 |
| Test accuracy | 22.48 |
| Loss vs number of iterations plot using matplotlib |  |
| Training and validation accuracy vs number of iterations plot |  |

|  |  |
| --- | --- |
| Experiment # | 4 |
| Hyperparameters | Learning rate = 0.01, epoch = 200, Regularization constant = 0.05 |
| Training accuracy | 23.72 |
| Validation accuracy | 21.3 |
| Test accuracy | 22.54 |
| Loss vs number of iterations plot using matplotlib |  |
| Training and validation accuracy vs number of iterations plot |  |

|  |  |
| --- | --- |
| Experiment # | 5 |
| Hyperparameters | Learning rate = 0.1, epoch = 100, regularization constant = 0.01 |
| Training accuracy | 22.6 |
| Validation accuracy | 23.59 |
| Test accuracy | 23.02 |
| Loss vs number of iterations plot using matplotlib |  |
| Training and validation accuracy vs number of iterations plot |  |

|  |  |
| --- | --- |
| Experiment # | 6 |
| Hyperparameters | Learning rate = 0.1, epoch = 100, regularization constant = 0.05 |
| Training accuracy | 22.07 |
| Validation accuracy | 24.8 |
| Test accuracy | 21.6 |
| Loss vs number of iterations plot using matplotlib |  |
| Training and validation accuracy vs number of iterations plot |  |

|  |  |
| --- | --- |
| Experiment # | 7 |
| Hyperparameters | Learning rate = 0.1, epoch = 200, regularization constant = 0.01 |
| Training accuracy | 26.51 |
| Validation accuracy | 25.1 |
| Test accuracy | 25.31 |
| Loss vs number of iterations plot using matplotlib |  |
| Training and validation accuracy vs number of iterations plot |  |

|  |  |
| --- | --- |
| Experiment # | 8 |
| Hyperparameters | Learning rate = 0.1, epoch = 200, regularization constant = 0.05 |
| Training accuracy | 25.85 |
| Validation accuracy | 25.5 |
| Test accuracy | 26.27 |
| Loss vs number of iterations plot using matplotlib |  |
| Training and validation accuracy vs number of iterations plot |  |

**Hyperparameters of Kaggle model**

|  |  |
| --- | --- |
| Optimal hyperparameters | Learning rate = 0.1, epoch = 200, regularization constant = 0.05 |
| Training accuracy | 25.85 |
| Validation accuracy | 25.5 |
| Test accuracy | 26.27 |
| Loss vs number of iterations plot |  |
| Training and validation accuracy vs number of iterations plot |  |