HW #4

Thomas Chow

SID 22215698

Problem 1

① $\mathcal{L}(\beta) = \lambda\|\beta\|_2^2 - \sum_{i=1}^{n}\left[y_i \log \mu_i + (1-y_i)\log(1-\mu_i)\right]$

We know $\mu_i = \frac{1}{1+e^{-\beta^T x_i}}$

$$\mathcal{L}(\beta) = \lambda\|\beta\|_2^2 - \sum_{i=1}^{n}\left[y_i \log\left(\frac{1}{1+e^{-\beta^T x_i}}\right) + (1-y_i)\log\left(1-\frac{1}{1+e^{-\beta^T x}}\right)\right.$$

$$\mathcal{L}(\beta) = \lambda\|\beta\|_2^2 - \sum_{i=1}^{n}\left[y_i \log\left(\frac{e^{\beta^T x}}{1+e^{\beta^T x}}\right) + (1-y_i)\log\left(\frac{e^{-\beta^T x}}{1+e^{-\beta^T x}}\right)\right]$$

$$\mathcal{L}(\beta) = \lambda\|\beta\|_2^2 - \sum_{i=1}^{n} y_i \log\left(\frac{e^{\beta^T x}}{1+e^{\beta^T x}}\right) + (1-y_i)\log\left(\frac{1}{1+e^{\beta^T x}}\right)$$

$$\mathcal{L}(\beta) = \lambda\|\beta\|_2^2 - \sum_{i=1}^{n} y_i\left[\log e^{\beta^T x} - \log(1+e^{\beta^T x})\right]$$

$$\mathcal{L}(\beta) = \lambda\|\beta\|_2^2 - \sum_{i=1}^{n} y_i\left[\beta^T x - \log(1+e^{\beta^T x})\right]$$

$$\frac{\partial \mathcal{L}(\beta)}{\partial \beta} = 2\lambda\beta - \sum_{i=1}^{n} y_i\left[x_i - \frac{x_i e^{\beta^T x_i}}{1+e^{\beta^T x}}\right]$$

$$\boxed{\frac{\partial \mathcal{L}(\beta)}{\partial \beta} = 2\lambda\beta = \sum_i x_i(y_i-\mu_i)}$$

② $$\frac{\partial \nabla_f \beta}{\partial \beta} = 2\lambda - \sum -x_i^2\left[\frac{e^{-\beta^T x_i}}{1+e^{-\beta^T x_i}}\right]\mu_i$$

$$= 2\lambda + \sum_{i=1}^{n} x_i^2(1-\mu_i)\mu_i$$

$$\boxed{\frac{\partial \nabla_f \beta}{\partial \beta} = 2\lambda + \sum_i x_i^2(1-\mu_i)\mu_i}$$

(3) Newton's: $\beta_{k+1} = \beta_k - H^{-1} \nabla f$

$$\boxed{\beta_{k+1} = \beta_k - \left[2\lambda + \sum_{i}^{N} x_i^2 (1-\mu_i)\mu_i\right]^{-1} \left[2\lambda\beta - \sum_i x_i (y_i - \mu_i)\right]}$$

(4) $\lambda = 0.07$, $\beta_0 = [2, 1, 0]^T$

(a). State value of μ_0.

$$\mu_0 = \frac{1}{1+e^{-\beta^T x_i}} = \boxed{[0.9526,\ 0.7311,\ 0.7311,\ 0.2689]}$$

calculated via MATLAB.

(b). β_1.

$$\beta_1 = \beta_0 - H^{-1} \nabla f(\beta)$$

$$\boxed{\beta_1 = [-0.3868,\ 1.4043,\ -2.2842]}$$

(c) μ_1.

$$\boxed{\mu_1 = [0.8731,\ 0.8258,\ 0.2932,\ 0.2198]}$$

(d). β_2

$$\boxed{\beta_2 = [-0.5122,\ 1.4527,\ -2.1627]}$$

CS 189 Problem Set 14
3/12/14

Thomas Chow
Login: by
Mukul Murthy
Login: dq

Problem 1 – see scan of problem 1

Problem 1.4 (retyped for clarity)

```
mu_0 =
   0.9526
   0.7311
   0.7311
   0.2689

B_1 =

  -0.3868
   1.4043
  -2.2842


mu_1 =

   0.8731
   0.8238
   0.2932
   0.2198



B_2 =

  -0.5122
   1.4527
  -2.1627
```

Problem 2

1. Derivations for batch gradient descent are attached (last page, handwritten).

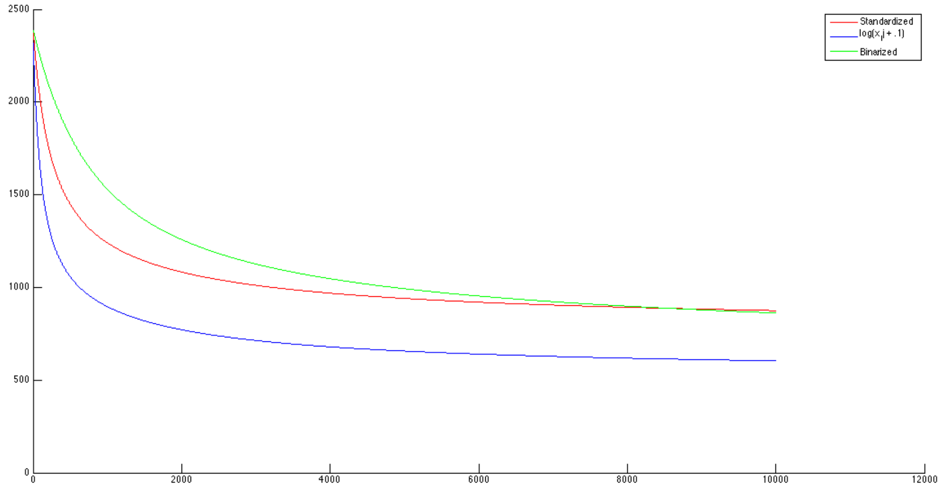

Figure 1. Loss values for batch gradient descent. 10000 Iterations. Lambda =1, Eta = 10^-6

2. Derivations for stochastic gradient descent are attached (last page, handwritten).

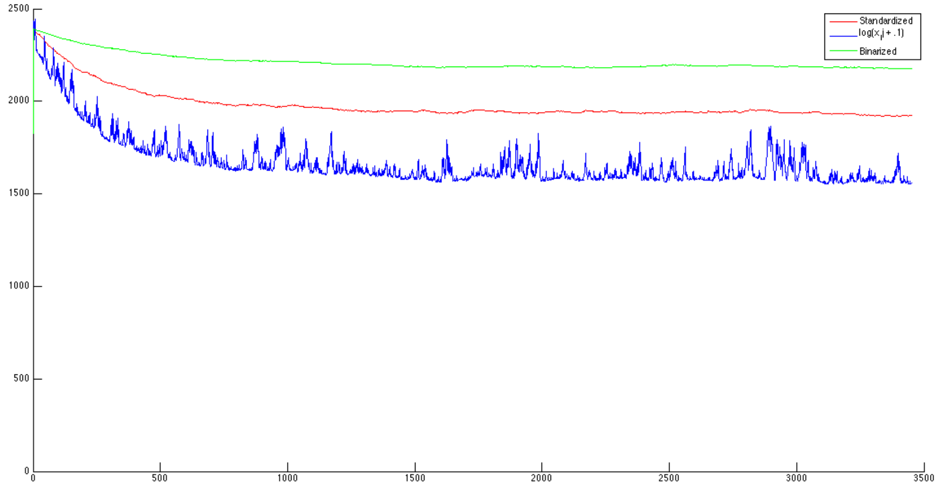

Figure 2. Loss values for stochastic gradient descent, with constant learning rate of 10^-3. 3450 iterations.

The negative log likelihood curve for stochastic gradient descent differs from that of batch gradient descent in the sense that it is not as smooth. This is due to the variation in individual data points that are used to compute the mean and beta vector. A mean and beta are computed based off a new data point during each iteration, which indicates that the change in the loss is decided by values that may vary drastically.

3.

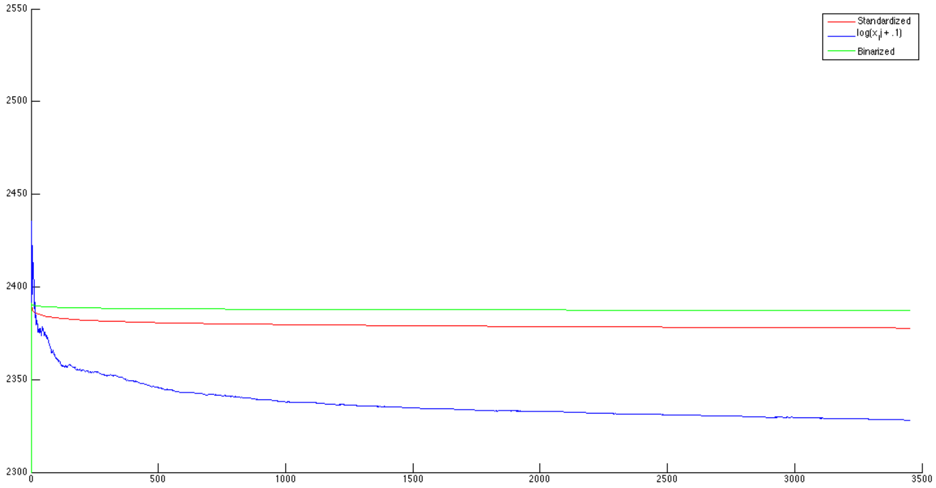

Figure 3. Loss values for stochastic gradient descent, with decreasing learning rate inversely proportional to the iteration count starting at 10^-3. 3450 iterations.

Using a decreasing learning rate improves the loss minimization. The plot indicates that the curve for the loss is much smoother than it was using a constant learning rate (for SGD). This occurs because we are giving less weight to the newly updated parameters as we increase iterations. Thus, each iteration tends to maintain the old parameter values, and the loss changes less.

4. We performed 10-fold cross validation and tuned values of lambda as well as tuning the preprocessing methods. We discovered that a lambda of 0.65 and the log transform preprocessing step gives the lowest average error over all the hold out sets.

Problem 2

1. Batch gradient descent:

$$\nabla \ell = 2\lambda \beta_k - X^T(y - \mu)$$

$$\mu = \frac{1}{1+e^{-\beta^T X}}$$

$$\beta_{k+1} = \beta_k - \eta \nabla$$

$$NLL = \lambda \|\beta\|_2^2 - \sum_{i=1}^{N} [y_i \log(\mu_i) + (1-y_i)\log(1-\mu_i)]$$

2.

$$\mu_i = \frac{1}{1+e^{-\beta^T x_i}}$$

$$\nabla = 2\lambda\beta - x_i^T(y_i - \mu)$$

$$\beta_{k+1} = \beta_k - \eta \nabla$$

$$NLL = \lambda \|\beta\|_2^2 - \sum_{i=1}^{N} y_i(\log(\mu_i)) + (1-y_i)\log(1-\mu_i)]$$