CS 6350: Machine Learning Fall 2016

Gopal Menon

December 6, 2016

1 Warmup: Probabilities

For the following questions, suppose A_1, A_2, A_3, A_4 are events.
(Remember that no points will be awarded without explanations.)

1. [2 points] If $P(A_1) = P(A_2) = P(A_1 \mid A_2) = \frac{1}{2}$, then are the events A_1 and A_2 independent? Why?

   Events A_1 and A_2 are independent if

   $$P(A_1 \mid A_2) = P(A_1)$$

   and

   $$P(A_2 \mid A_1) = P(A_2)$$

   Using Bayes rule

   $$\begin{aligned} P(A_2 \mid A_1) &= \frac{P(A_1 \mid A_2)P(A_2)}{P(A_1)} \\ &= \frac{\frac{1}{2} \times \frac{1}{2}}{\frac{1}{2}} \\ &= \frac{1}{2} \\ &= P(A_2) \\ &= P(A_1 \mid A_2) \\ &= P(A_1) \end{aligned}$$

2. [3 points] Suppose A_1, A_2 and A_3 are mutually exclusive. If, for $i \in \{1, 2, 3\}$, we have $P(A_i) = \frac{1}{3}$ and $P(A_4 \mid A_i) = \frac{i}{6}$, then what is $P(A_4)$?

   Using the theorem of total probability in the above case

   $$\begin{aligned} P(A_4) &= \sum_{i=1}^{3} P(A_4 \mid A_i)P(A_i) \\ &= \sum_{i=1}^{3} \frac{i}{6} \times \frac{1}{3} = \frac{1}{3} \times \left(\frac{1}{6} + \frac{2}{6} + \frac{3}{6}\right) = \frac{1}{3} \times \frac{6}{6} = \frac{1}{3} \end{aligned}$$

3. [3 points] Let n be the number at the top when a fair six-sided die is tossed. If a fair coin is tossed n times, then what is the probability of exactly two heads?

Let H be the event of getting a head, $2H$ be the event of getting exactly two heads when tossing a coin, and let D_n be the event for the number at the top when a die is tossed. So the probability of exactly two heads is

$$
\begin{aligned}
&= \sum_{n=1}^{6} P(2H \mid D_n) \times P(D_n) \\
&= \sum_{n=1}^{6} P(2H \mid D_n) \times \frac{1}{6} \\
&= \sum_{n=1}^{6} \binom{n}{2} \times (P(H))^2 \times (1 - P(H))^{n-2} \times \frac{1}{6} \\
&= \frac{1}{6} \times [0 + 1 \times \left(\frac{1}{2}\right)^2 \times \left(\frac{1}{2}\right)^0 + 3 \times \left(\frac{1}{2}\right)^2 \times \left(\frac{1}{2}\right)^1 + 6 \times \left(\frac{1}{2}\right)^2 \times \left(\frac{1}{2}\right)^2 \\
&+ 10 \times \left(\frac{1}{2}\right)^2 \times \left(\frac{1}{2}\right)^3 + 15 \times \left(\frac{1}{2}\right)^2 \times \left(\frac{1}{2}\right)^4] \\
&= \frac{1}{6} \times [0 + \frac{1}{4} + \frac{3}{8} + \frac{6}{16} + \frac{10}{32} + \frac{15}{64}] \\
&= \frac{1}{6} \times [\frac{16 + 24 + 24 + 20 + 15}{64}] \\
&= \frac{1}{6} \times \frac{99}{64} \\
&= \frac{33}{128}
\end{aligned}
$$

4. [4 points] Prove or disprove: If $P(A_1) = a_1$ and $P(A_2) = a_2$, then $P(A_1|A_2) \geq \frac{a_1+a_2-1}{a_2}$.

From the product rule of probability, we know that

$$P(A_1 \wedge A_2) = P(A_1 \mid A_2)P(A_2) \tag{1}$$

From the sum rule of probability, we know that

$$P(A_1 \vee A_2) = P(A_1) + P(A_2) - P(A_1 \wedge A_2) \tag{2}$$

From equation 1,

$$P(A_1 \mid A_2) = \frac{P(A_1 \wedge A_2)}{P(A_2)}$$

Using equation 2,

$$P(A_1 \mid A_2) = \frac{P(A_1) + P(A_2) - P(A_1 \wedge A_2)}{P(A_2)}$$

Since we know that $P(A_1 \wedge A_2) \leq 1$,

$$\begin{aligned} P(A_1 \mid A_2) &\geq \frac{P(A_1) + P(A_2) - 1}{P(A_2)} \\ &\geq \frac{a_1 + a_2 - 1}{a_2} \end{aligned}$$

5. [8 points] If A_1 and A_2 are independent events, then show that

   (a) $E[A_1 + A_2] = E[A_1] + E[A_2]$
   The expected value (also known as the mean μ) of a random variable X is defined as

   $$E(X) = \sum_{e \in S} X(e)P(e)$$

   where e is a single event in probability space S.

   $$\begin{aligned} E(A_1 + A_2) &= \sum_{e \in S} \{A_1(e) + A_2(e)\} P(e) \\ &= \sum_{e \in S} A_1(e)P(e) + A_2(e)P(e) \\ &= E(A_1) + E(A_2) \end{aligned}$$

6. $var[A_1 + A_2] = var[A_1] + var[A_2]$

   Here $E[\cdot]$ and $var[\cdot]$ denote the mean and variance respectively.

   The variance of a random variable X is defined as

   $$\begin{aligned} var(X) &= E([X - E(X)]^2) \\ &= E(X^2 - 2XE(X) + E(X)^2) \\ &= E(X^2) - 2E(XE(X)) + E(E(X)^2) \end{aligned}$$

   In the above equations, I have represented $(E(X))^2$ as $E(X)^2$ in order to simplify the notation rather than use the explicit version with the extra parentheses.

   Based on the definition of $E(X)$,

   $$\begin{aligned} E(XE(X)) &= \sum_{e \in S} X(e)P(e)E(X) \\ &= E(X) \sum_{e \in S} X(e)P(e) \\ &= E(X)^2 \end{aligned}$$

The reason we can take $E(X)$ out of the summation above, is that it is just a number. By a similar argument, $E(E(X)^2) = E(X)^2$, since expected value of a number is that same number.

Going back to the expansion of $var(X)$,

$$\begin{aligned} var(X) &= E(X^2) - 2E(XE(X)) + E(E(X)^2) \\ &= E(X^2) - 2E(X)^2 + E(X)^2 \\ &= E(X^2) - E(X)^2 \end{aligned}$$

Now we can expand $var[A_1 + A_2]$

$$\begin{aligned} var[A_1 + A_2] &= E([A_1 + A_2]^2) - E(A_1 + A_2)^2 \\ &= E(A_1^2 + 2A_1A_2 + A_2^2) - (E(A_1) + E(A_2))^2 \\ &= E(A_1^2) + 2E(A_1A_2) + E(A_2^2) - (E(A_1)^2 + 2E(A_1)E(A_2) + E(A_2)^2) \end{aligned}$$

Based on he definition of $E(X)$

$$\begin{aligned} E(A_1A_2) &= \sum_{e \in S} A_1(e)A_2(e)P(e) \\ &= \sum_{x \in S, y \in S} A_1(x)A_2(y)P(A_1 = x, A_2 = y) \end{aligned}$$

Since A_1 and A_2 are independent, $P(A_1 = x, A_2 = y) = P(A_1 = x)P(A_2 = y)$. Going back to the expansion of $E(A_1A_2)$,

$$\begin{aligned} E(A_1A_2) &= \sum_{x \in S, y \in S} A_1(x)A_2(y)P(A_1 = x, A_2 = y) \\ &= \sum_{x \in S, y \in S} A_1(x)A_2(y)P(A_1 = x)P(A_2 = y) \\ &= \sum_{x \in S} A_1(x)P(A_1 = x) \sum_{y \in S} A_2(y)P(A_2 = y) \\ &= E(A_1)E(A_2) \end{aligned}$$

Going back to the expansion of $var[A_1 + A_2]$,

$$\begin{aligned} var[A_1 + A_2] &= E(A_1^2) + 2E(A_1A_2) + E(A_2^2) - (E(A_1)^2 + 2E(A_1)E(A_2) + E(A_2)^2) \\ &= E(A_1^2) + 2E(A_1)E(A_2) + E(A_2^2) - (E(A_1)^2 + 2E(A_1)E(A_2) + E(A_2)^2) \\ &= E(A_1^2) - E(A_1)^2 + E(A_2^2) - E(A_2)^2 \end{aligned}$$

Since we have already shown above that $var(X) = E(X^2) - E(X)^2$, this means that

$$var[A_1 + A_2] = var(A_1) + var(A_2)$$

2 Naive Bayes

1. [Part 1] Suppose we have a binary classification problem where the label y can either be -1 or 1. In the first case, consider the case where we have only one feature x_1 that can also be either -1 or 1. The generative distribution of the data is $P(x_1, y) = P(y)P(x_1 \mid y)$. Note that this satisfies the independence assumption of the naive Bayes model. All features are conditionally independent of each other given the label – of course, there is only one feature so this statement is trivially true.

   Suppose we know the true distribution that generated the data as follows:

   - $P(y = -1) = 0.1$ and $P(y = 1) = 0.9$
   - $P(x_1 = -1 \mid y = -1) = 0.8$, $P(x_1 = 1 \mid y = -1) = 0.2$, $P(x_1 = -1 \mid y = 1) = 0.1$ and $P(x_1 = 1 \mid y = 1) = 0.9$.

   (a) [2 points] If we have infinite data drawn from this distribution and we train a naive Bayes classifier, what would the values of $\hat{P}(x_1 \mid y)$ and $\hat{P}(y)$ be?

   According to *Hoeffdings Inequality* [1], the probability distribution of a random variable ν will be very close to its mean value μ for large samples. For a sample size of N,

   $$P\left[|\nu - \mu| > \epsilon\right] \leq 2e^{-2\epsilon^2 N}$$

   for any $\epsilon > 0$.

   When N is ∞, the values of $\hat{P}(x_1 \mid y)$ and $\hat{P}(y)$ will be the same as $P(x_1 \mid y)$ and $P(y)$.

   (b) [6 points] Use these learned values probabilities from the previous question to fill up the following table:

Input x_1	$\hat{P}(x_1, y = -1)$	$\hat{P}(x_1, y = 1)$	**Prediction:** $y' = arg\max_y \hat{P}(x_1, y)$
-1	$0.8 \times 0.1 = 0.08$	$0.1 \times 0.9 = 0.09$	1
1	$0.2 \times 0.1 = 0.02$	$0.9 \times 0.9 = 0.81$	1

   (c) [3 points] If the probabilities learned above were used to make predictions, what would the error of that classifier be? In other words, what is $P(y' \neq y)$?

   Hint: To answer this, you should use the fact that $P(y' \neq y) = P(y' \neq y, x_1 = -1) + P(y' \neq y, x_1 = 1)$.

$$\begin{aligned}
P(y' \neq y) &= P(y' \neq y, x_1 = -1) + P(y' \neq y, x_1 = 1) \\
P(y' \neq y) &= P(y = -1, x_1 = -1) + P(y = -1, x_1 = 1) \\
P(y' \neq y) &= P(x_1 = -1, y = -1) + P(x_1 = 1, y = -1) \\
P(y' \neq y) &= P(x_1 = -1 \mid y = -1)P(y = -1) + P(x_1 = 1 \mid y = -1)P(y = -1) \\
&= 0.8 \times 0.1 + 0.2 \times 0.1 \\
&= 0.08 + 0.02 \\
&= 0.10
\end{aligned}$$

2. [Part 2] Now, suppose we have a binary classification problem with two features x_1, x_2 both of which can be -1 or 1. However, the second feature x_2 is actually identical to the first feature x_1. And we have the same true probabilities $P(x_1 \mid y)$ and $P(y)$ as in Part 1 above.

   (a) [1 point] Are x_1 and x_2 conditionally independent given y? Prove your answer formally using the definition of conditional independence.

   Since the features x_1 and x_2 are identical,

$$P(x_1, x_2 \mid y) = P(x_1 \mid y) = P(x_2 \mid y)$$

   For x_1 and x_2 to be conditionally independent given y, the following should hold true

$$P(x_1, x_2 \mid y) = P(x_1 \mid y)P(x_2 \mid y)$$

   The only cases where the product of two probabilities is the same as the individual probabilities is when both are 0 or when both are 1. This means that the above two equations cannot be true for all cases of probability values and so x_1 and x_2 are not conditionally independent given y.

   (b) [8 points] Let $\hat{P}(x_1 \mid y), \hat{P}(x_2 \mid y)$ and $\hat{P}(y)$ represent the learned parameters of a naive Bayes classifier that is learned on infinite data generated according to the above distribution. Using these parameters, fill up the following table:

x_1	x_2	$\hat{P}(x_1, x_2, y = -1)$	$\hat{P}(x_1, x_2, y = 1)$	**Prediction:** $y' = arg\max_y \hat{P}(x_1, x_2, y)$
-1	-1	$.8 \times .8 \times .1 = .064$	$.1 \times .1 \times .9 = .009$	-1
-1	1	$.8 \times .2 \times .1 = .016$	$.1 \times .9 \times .9 = .081$	1
1	-1	$.2 \times .8 \times .1 = .016$	$.9 \times .1 \times .9 = .081$	1
1	1	$.2 \times .2 \times .1 = .004$	$.9 \times .9 \times .9 = .729$	1

   (c) [3 points] If the probabilities learned above were used to make predictions, what would the error of that classifier be? In other words, what is $P(y' \neq y)$?

   The following computations assume that x_1 and x_2 are independent given y.

$$\begin{aligned}
P(y' \neq y) &= P(y' \neq y, x_1 = -1, x_2 = -1) + P(y' \neq y, x_1 = -1, x_2 = 1) \\
&\quad + P(y' \neq y, x_1 = 1, x_2 = -1) + P(y' \neq y, x_1 = 1, x_2 = 1) \\
P(y' \neq y) &= P(x_1 = -1, x_2 = -1, y' \neq y) + P(x_1 = -1, x_2 = 1, y' \neq y) \\
&\quad + P(x_1 = 1, x_2 = -1, y' \neq y) + P(x_1 = 1, x_2 = 1, y' \neq y) \\
P(y' \neq y) &= P(x_1 = -1, x_2 = -1, y = 1) + P(x_1 = -1, x_2 = 1, y = -1) \\
&\quad + P(x_1 = 1, x_2 = -1, y = -1) + P(x_1 = 1, x_2 = 1, y = -1) \\
P(y' \neq y) &= P(x_1 = -1, x_2 = -1 \mid y = 1)P(y = 1) + P(x_1 = -1, x_2 = 1 \mid y = -1)P(y = -1) \\
&\quad + P(x_1 = 1, x_2 = -1 \mid y = -1)P(y = -1) + P(x_1 = 1, x_2 = 1 \mid y = -1)P(y = -1)
\end{aligned}$$

$$
\begin{aligned}
P(y' \neq y) &= P(x_1 = -1 \mid y = 1)P(x_2 = -1 \mid y = 1)P(y = 1) \\
&+ P(x_1 = -1 \mid y = -1)P(x_2 = 1 \mid y = -1)P(y = -1) \\
&+ P(x_1 = 1 \mid y = -1)P(x_2 = -1 \mid y = -1)P(y = -1) \\
&+ P(x_1 = 1 \mid y = -1)P(x_2 = 1 \mid y = -1)P(y = -1) \\
&= 0.1 \times 0.1 \times 0.9 + 0.8 \times 0.2 \times 0.1 + 0.2 \times 0.8 \times 0.1 + 0.2 \times 0.2 \times 0.1 \\
&= 0.009 + 0.016 + 0.016 + 0.004 \\
&= 0.45
\end{aligned}
$$

(d) [2 points] Do you expect a logistic regression classifier to have the same performance as the naive Bayes classifier when the variable is duplicated? Give an intuitive explanation (no more than 2 sentences) for your answer.

Given that both Naïve Bayes and Logistic Regression classifiers have a linear decision boundary, the decision boundary has to be the same. Since both classifiers will in effect learn the same linear decision boundary, they will predict the same output.

3 [25 points] Naïve Bayes and Linear Classifiers

In this problem you will show that a Gaussian naïve Bayes classifier is a linear classifier. We will denote inputs by d dimensional vectors, $\mathbf{x} = (x_1, x_2, \ldots, x_d)^T$. We will assume that each feature x_j is a real number. Our classifier will predict the label 1 if $\Pr(y = 1|\mathbf{x}) \geq \Pr(y = 0|\mathbf{x})$. Or equivalently, $\frac{\Pr(\mathbf{x}|y=1)\Pr(y=1)}{\Pr(\mathbf{x}|y=0)\Pr(y=0)} \geq 1$. Remember the naïve Bayes assumption we saw in class: $\Pr(\mathbf{x}|y) = \prod_{j=0}^{d} \Pr(x_j|y)$

Suppose each $P(x_j|y)$ is defined using a Gaussian/Normal probability density function, one for each value of y and j. Each Gaussian distribution has mean $\mu_{j,y}$ and variance σ^2 (Note that they will all have same variance). As a reminder, the Gaussian distribution is represented by the following probability density function: $f(x_j \mid \mu_{j,y}, \sigma) = \frac{1}{\sqrt{2\sigma^2\pi}} e^{-\frac{(x_j - \mu_{j,y})^2}{2\sigma^2}}$

Show that this naïve Bayes classifier has a linear decision boundary.

[Hint: Refer to the notes on the naive Bayes classifier and Linear models in the class website to see how to do this with binary features]

The classifier will predict a label of 1 if

$$P(y = 1 \mid \mathbf{x}) \geq P(y = 0 \mid \mathbf{x})$$

or equivalently if

$$
\begin{aligned}
\frac{P(y = 1 \mid \mathbf{x})}{P(y = 0 \mid \mathbf{x})} &\geq 1 \\
\frac{P(y = 1)P(\mathbf{x} \mid y = 1)}{P(y = 0)P(\mathbf{x} \mid y = 0)} &\geq 1
\end{aligned}
$$

Using the Naïve Bayes assumption

$$\frac{P(y=1)\prod_{j=0}^{d} P(\mathbf{x}_j \mid y=1)}{P(y=0)\prod_{j=0}^{d} P(\mathbf{x}_j \mid y=0)} \geq 1$$

As per the normal probability distribution assumption described above in the question, the classifier will predict a label of 1 if,

$$\frac{P(y=1)}{P(y=0)}\prod_{j=0}^{d}\frac{\frac{1}{\sqrt{2\sigma^2\pi}}\exp\left(\frac{-\left(x_j-\mu_{1_{j,y}}\right)^2}{2\sigma^2}\right)}{\frac{1}{\sqrt{2\sigma^2\pi}}\exp\left(\frac{-\left(x_j-\mu_{2_{j,y}}\right)^2}{2\sigma^2}\right)} \geq 1$$

$$\frac{P(y=1)}{P(y=0)}\prod_{j=0}^{d}\frac{\exp\left(\frac{-\left(x_j-\mu_{1_{j,y}}\right)^2}{2\sigma^2}\right)}{\exp\left(\frac{-\left(x_j-\mu_{2_{j,y}}\right)^2}{2\sigma^2}\right)} \geq 1$$

$$\ln\left(\frac{P(y=1)}{P(y=0)}\right)+\sum_{j=0}^{d}\left(-\frac{\left(x_j-\mu_{1_{j,y}}\right)^2}{2\sigma^2}+\frac{\left(x_j-\mu_{2_{j,y}}\right)^2}{2\sigma^2}\right) \geq 0$$

$$\ln\left(\frac{P(y=1)}{P(y=0)}\right)+\sum_{j=0}^{d}\left(\frac{-x_j^2+2x_j\mu_{1_{j,y}}-\mu_{1_{j,y}}^2+x_j^2-2x_j\mu_{2_{j,y}}+\mu_{2_{j,y}}^2}{2\sigma^2}\right) \geq 0$$

$$\ln\left(\frac{P(y=1)}{P(y=0)}\right)+\sum_{j=0}^{d}\left(\frac{\mu_{2_{j,y}}^2-\mu_{1_{j,y}}^2+2(\mu_{1_{j,y}}-\mu_{2_{j,y}})x_j}{2\sigma^2}\right) \geq 0$$

$$b+\sum_{j=0}^{d}x_jw_j \geq 0$$

where

$$b=\ln\left(\frac{P(y=1)}{P(y=0)}\right)+\sum_{j=0}^{d}\left(\frac{\mu_{2_{j,y}}^2-\mu_{1_{j,y}}^2}{2\sigma^2}\right)$$

and

$$w_j=\sum_{j=0}^{d}\frac{\mu_{1_{j,y}}-\mu_{2_{j,y}}}{\sigma^2}$$

This means that the classifier has a linear decision boundary.

4 Experiment

We looked maximum a posteriori learning of the logistic regression classifier in class. In particular, we showed that learning the classifier is equivalent to the following optimization problem: $\min_{\mathbf{w}}\left\{\sum_{i=1}^{m}\log(1+\exp(-y_i\mathbf{w}^T\mathbf{x}_i))+\frac{1}{\sigma^2}\mathbf{w}^T\mathbf{w}\right\}$.

In this question, you will derive the stochastic gradient descent algorithm for the logistic regression classifier, and also implement it with cross-validation.

1. [5 points] What is the derivative of the function $g(\mathbf{w})=\log(1+\exp(-y_i\mathbf{w}^T\mathbf{x}_i))$ with respect to the weight vector?

$$\begin{aligned}
\nabla g(\mathbf{w}) &= \nabla\log(1+\exp(-y_i\mathbf{w}^T\mathbf{x}_i))\\
&= \frac{1}{1+\exp(-y_i\mathbf{w}^T\mathbf{x}_i)}\exp(-y_i\mathbf{w}^T\mathbf{x}_i)(-y_i x_i)\\
&= \frac{-y_i\mathbf{x}_i\exp(-y_i\mathbf{w}^T\mathbf{x}_i)}{1+\exp(-y_i\mathbf{w}^T\mathbf{x}_i)} = \frac{-y_i\mathbf{x}_i}{1+\exp(y_i\mathbf{w}^T\mathbf{x}_i)}
\end{aligned}$$

2. [5 points] The inner most step in the SGD algorithm is the gradient update where we use a single example instead of the entire dataset to compute the gradient. Write down the objective where the entire dataset is composed of a single example, say $(\mathbf{x}_i, y_i)$. Derive the gradient of this objective with respect to the weight vector.

   We need to find the weight $\mathbf{w}$ that minimizes the expression given above in the question. The objective when the entire dataset consists of a single example $(\mathbf{x}_i, y_i)$ is $J(\mathbf{w})=\log(1+\exp(-y_i\mathbf{w}^T\mathbf{x}_i))+\frac{1}{\sigma^2}\mathbf{w}^T\mathbf{w}$

   The derivative of the first part has already been derived above. The gradient of this objective with respect to the weight vector is $\nabla J(\mathbf{w})=\frac{-y_i\mathbf{x}_i}{1+\exp(y_i\mathbf{w}^T\mathbf{x}_i)}+\frac{2\mathbf{w}}{\sigma^2}$

3. [10 points] Write down the pseudo code for the stochastic gradient algorithm using the gradient from previous part.

Algorithm 1 Stochastic Gradient Descent

```
procedure SGD(S = (x_i, y_i), x ∈ ℝ^n, y ∈ {−1, 1}, T, γ_0, σ)
    w = 0 ∈ ℝ^n, t = 0
    for epoch = 1 to T do
        Shuffle test data
        for (x_i, y_i) ∈ S do
            γ_t = γ_0 / (1 + γ_0 t / σ)
            w = w − γ_t ( −y_i x_i / (1 + exp(y_i w^T x_i)) + 2w / σ^2 )
            t = t + 1
        end for
    end for
    return w
end procedure
```

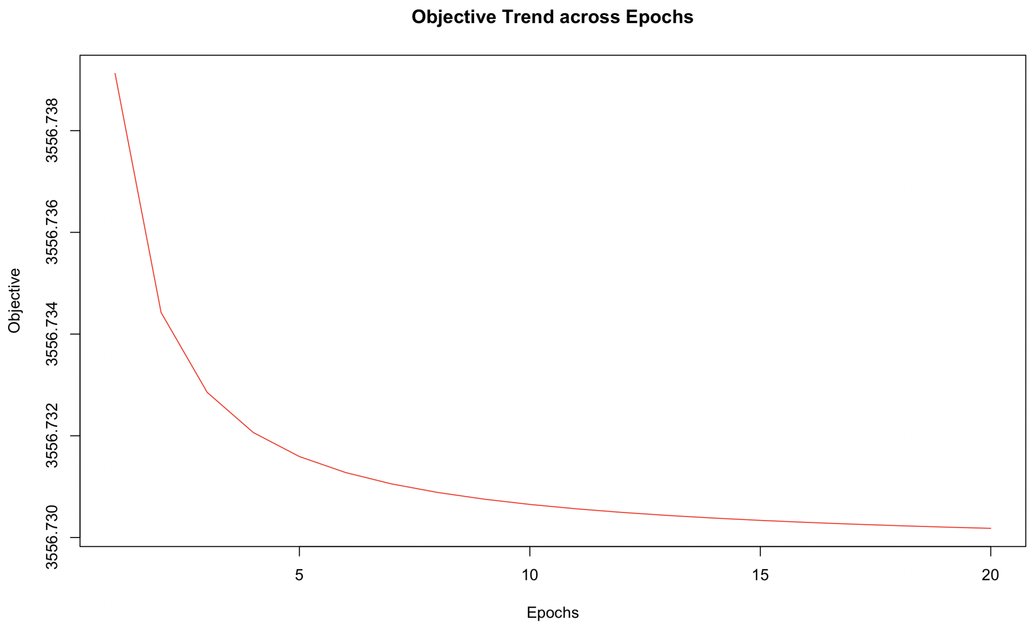

Figure 1: Plot of the objective

4. [20 points] The accuracy obtained after 5 fold cross validation was 76%. A learning rate of 1.0×10^{-6} and σ of 0.010000000000000002 was chosen based on the highest average accuracy value of 75% achieved during cross validation. During stochastic gradient descent, 20 epochs were used during 5 fold cross validation and this resulted in $20 \times 5 = 100$ epochs. A plot of the objective for 20 epochs is shown in figure 1.

References

[1] Abu-Mostafa, Yaser S., Malik Magdon-Ismail, and Hsuan-Tien Lin. *Learning from Data: A Short Course*. United States: AMLBook.com, 2012. Print.