CS 6350: Machine Learning – Homework 6

Clinton Fernandes, u1016390 12/06/2016

1 Warmup: Probabilities

1. We are given that $P(A_1) = P(A_2) = P(A_1/A_2) = \frac{1}{2}$

   From bayes theorm: $P(A_1 \bigcap A_2) = P(A_1 \mid A_2)P(A_2)$

   From the above two expressions, we have $P(A_1 \bigcap A_2) = P(A_1)P(A_2)$

   We see that the probability of occurrence of both the events (A_1 and A_2) is equal to the product of the probabilities of occurrence of each event. This means that occurrence of A_1 does not affect the probability of A_2 occurring. Hence the events A_1 and A_2 are independent.

2. $P(A_1) = P(A_2) = P(A_3) = 1/3$

   $P(A_4 \mid A_1) = 1/6 \qquad P(A_4 \mid A_2) = 2/6 \qquad P(A_4 \mid A_3) = 3/6$

   $P(A_4 \bigcap A_1) = P(A_4 \mid A_1)P(A_1) = (1/6) * (1/3) = 1/18$

   $P(A_4 \bigcap A_2) = P(A_4 \mid A_2)P(A_2) = (2/6) * (1/3) = 2/18$

   $P(A_4 \bigcap A_3) = P(A_4 \mid A_3)P(A_3) = (3/6) * (1/3) = 3/18$

   $P(A_4) = P(A_4 \bigcap A_1) + P(A_4 \bigcap A_2) + P(A_4 \bigcap A_3) = 6/18 = 1/3 = 0.333\bar{3}$

3. If a six sided die is tossed than the possible numbers that can occur on the top face of the die are either 1 or 2 or 3 or 4 or 5 or 6. n = {1, 2, 3, 4, 5, 6}

   Also, the probability of any of the 6 numbers appearing on the top face of the die is $P(n) = 1/6$

   n is the number of times the coin is tossed

   $P(heads = 2 \mid n = 1) = 0$

   $P(heads = 2 \mid n = 2) = 1/4$

   $P(heads = 2 \mid n = 3) = 3/8$

   $P(heads = 2 \mid n = 4) = 6/16$

   $P(heads = 2 \mid n = 5) = 10/32$

   $P(heads = 2 \mid n = 6) = 15/64$

   Probability of exactly 2 heads $= P(heads = 2) = \sum_{n=1}^{6} P(heads = 2 \mid n)P(n)$

   $= \frac{1}{6} \sum_{n=1}^{6} P(heads = 2 \mid n)$

$P(heads = 2) = \frac{1}{6}(1/4 + 3/8 + 6/16 + 12/32 + 15/64) = 33/128 = 0.2578125$

4. $P(A_1) = a_1$ and $P(A_2) = a_2$

   $P(A_1 \cup A_2) \leq 1$

   $P(A_1) + P(A_2) - P(A_1 \mid A_2)P(A_2) \ \leq \ 1$

   $P(A_1 \mid A_2) \geq \frac{P(A_1)+P(A_2)-1}{P(A_2)}$

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

5. (a) If A_1 and A_2 are independent events, show that $E[A_1 + A_2] = E[A_1] + E[A_2]$

   Consider that the random variable A_1 takes the values $a_{11}, a_{12}, a_{13}, ..., a_{1m}$

   Consider that the random variable A_2 takes the values $a_{21}, a_{22}, a_{23}, ..., a_{2n}$

   $E[X] = \sum_{i=1}^{n} x_i P(X = x_i)$

$$
\begin{aligned}
E[A_1 + A_2] &= \sum_{i=1}^{m}\sum_{j=1}^{n}(a_{1i} + a_{2j})P(A_1 = a_{1i}, A_2 = a_{2j}) \\
&= \sum_{i=1}^{m}\sum_{j=1}^{n} a_{1i}P(A_1 = a_{1i}, A_2 = a_{2j}) + \sum_{i=1}^{m}\sum_{j=1}^{n} a_{2j}P(A_1 = a_{1i}, A_2 = a_{2j}) \\
&= \sum_{i=1}^{m}\Big(a_{1i}\sum_{j=1}^{n} P(A_1 = a_{1i}, A_2 = a_{2j})\Big) + \sum_{j=1}^{n}\Big(a_{2j}\sum_{i=1}^{m} P(A_1 = a_{1i}, A_2 = a_{2j})\Big) \\
&\qquad \text{Since the random variables are independent} \\
&= \sum_{i=1}^{m}\Big(a_{1i}\sum_{j=1}^{n} P(A_1 = a_{1i})P(A_2 = a_{2j})\Big) + \sum_{j=1}^{n}\Big(a_{2j}(\sum_{i=1}^{m} P(A_1 = a_{1i})P(A_2 = a_{2j})\Big) \\
&= \sum_{i=1}^{m}\Big(a_{1i}P(A_1 = a_{1i})\sum_{j=1}^{n} P(A_2 = a_{2j})\Big) + \sum_{j=1}^{n}\Big(a_{2j}P(A_2 = a_{2j})\sum_{i=1}^{m} P(A_1 = a_{1i})\Big) \\
&\qquad \text{we know that} \quad \sum_{j=1}^{n} P(A_2 = a_{2j}) = 1 \quad \text{and} \quad \sum_{i=1}^{m} P(A_1 = a_{1i}) = 1 \\
&= \sum_{i=1}^{m}(a_{1i})P(A_1 = a_{1i}) + \sum_{j=1}^{n}(a_{2j})P(A_2 = a_{2j}) \\
&= E[A_1] + E[A_2]
\end{aligned}
\tag{1}
$$

   (b) If A_1 and A_2 are independent events, show that $var[A_1 + A_2] = var[A_1] + var[A_2]$

   $var[X] = \sum_{i=1}^{n}(x_i - E[X])^2 P(X = x_i)$

$$
\begin{aligned}
var[A_1 + A_2] &= \sum_{i=1}^{m}\sum_{j=1}^{n}(a_{1i} - E[A_1] + a_{2j} - E[A_2])^2 P(A_1 = a_{1i}, A_2 = a_{2j}) \\
&= \sum_{i=1}^{m}\sum_{j=1}^{n}(a_{1i} - E[A_1])^2 P(A_1 = a_{1i}, A_2 = a_{2j}) \\
&\quad + \sum_{i=1}^{m}\sum_{j=1}^{n}(a_{2j} - E[A_2])^2 P(A_1 = a_{1i}, A_2 = a_{2j}) \\
&\quad - 2\sum_{i=1}^{m}\sum_{j=1}^{n}(a_{1i} - E[A_1])(a_{2j} - E[A_2]) P(A_1 = a_{1i}, A_2 = a_{2j})
\end{aligned} \tag{2}
$$

Consider the first two terms of the above equation (2)

$$
\sum_{i=1}^{m}\sum_{j=1}^{n}(a_{1i} - E[A_1])^2 P(A_1 = a_{1i}, A_2 = a_{2j}) + \sum_{i=1}^{m}\sum_{j=1}^{n}(a_{2j} - E[A_2])^2 P(A_1 = a_{1i}, A_2 = a_{2j})
$$

$$
\begin{aligned}
&= \sum_{i=1}^{m}\sum_{j=1}^{n}(a_{1i} - E[A_1])^2 P(A_1 = a_{1i}, A_2 = a_{2j}) \\
&\qquad + \sum_{i=1}^{m}\sum_{j=1}^{n}(a_{2j} - E[A_2])^2 P(A_1 = a_{1i}, A_2 = a_{2j}) \\
&= \sum_{i=1}^{m}(a_{1i} - E[A_1])^2 \sum_{j=1}^{n} P(A_1 = a_{1i}, A_2 = a_{2j}) \\
&\qquad + \sum_{j=1}^{n}(a_{2j} - E[A_2])^2 \sum_{i=1}^{m} P(A_1 = a_{1i}, A_2 = a_{2j}) \\
&\qquad \text{(Since the random variables are independent)} \\
&= \sum_{i=1}^{m}(a_{1i} - E[A_1])^2 \sum_{j=1}^{n} P(A_1 = a_{1i}) P(A_2 = a_{2j}) \\
&\qquad + \sum_{j=1}^{n}(a_{2j} - E[A_2])^2 \sum_{i=1}^{m} P(A_1 = a_{1i}) P(A_2 = a_{2j}) \\
&= \sum_{i=1}^{m}(a_{1i} - E[A_1])^2 P(A_1 = a_{1i}) \sum_{j=1}^{n} P(A_2 = a_{2j}) \\
&\qquad + \sum_{j=1}^{n}(a_{2j} - E[A_2])^2 P(A_2 = a_{2j}) \sum_{i=1}^{m} P(A_1 = a_{1i}) \\
&\qquad \text{(we know that } \sum_{j=1}^{n} P(A_2 = a_{2j}) = 1 \quad \text{and} \quad \sum_{i=1}^{m} P(A_1 = a_{1i}) = 1) \\
&= \sum_{i=1}^{m}(a_{1i} - E[A_1])^2 P(A_1 = a_{1i}) + \sum_{j=1}^{n}(a_{2j} - E[A_2])^2 P(A_2 = a_{2j}) \\
&= var[A_1] + var[A_2]
\end{aligned} \tag{3}
$$

Now, consider the third term of the equation (2)

$$-2\sum_{i=1}^{m}\sum_{j=1}^{n}(a_{1i}-E[A_1])(a_{2j}-E[A_2])P(A_1=a_{1i},A_2=a_{2j})$$

$$\begin{aligned}
&= -2\sum_{i=1}^{m}\sum_{j=1}^{n}(a_{1i}-E[A_1])(a_{2j}-E[A_2])P(A_1=a_{1i})P(A_2=a_{2j}) \\
&= -2\sum_{i=1}^{m}\Big((a_{1i}-E[A_1])P(A_1=a_{1i})\sum_{j=1}^{n}\Big((a_{2j}-E[A_2])P(A_2=a_{2j})\Big)\Big) \\
&= -2\sum_{i=1}^{m}\Big((a_{1i}-E[A_1])P(A_1=a_{1i})\sum_{j=1}^{n}\Big((a_{2j}P(A_2=a_{2j})-E[A_2]P(A_2=a_{2j})\Big)\Big) \\
&= -2\sum_{i=1}^{m}\Big((a_{1i}-E[A_1])P(A_1=a_{1i})\Big(\sum_{j=1}^{n}(a_{2j}P(A_2=a_{2j})-\sum_{j=1}^{n}E[A_2]P(A_2=a_{2j})\Big)\Big) \\
&= -2\sum_{i=1}^{m}\Big((a_{1i}-E[A_1])P(A_1=a_{1i})\Big(E[A_2]-E[A_2]\sum_{j=1}^{n}P(A_2=a_{2j})\Big)\Big) \\
&= -2\sum_{i=1}^{m}\Big((a_{1i}-E[A_1])P(A_1=a_{1i})\Big(E[A_2]-E[A_2]*1\Big)\Big) \\
&= -2\sum_{i=1}^{m}\Big((a_{1i}-E[A_1])P(A_1=a_{1i})\Big(0\Big)\Big)=0
\end{aligned} \tag{4}$$

Now, from equation 2, equation 3 and equation 4, $var[A_1+A_2]=var[A_1]+var[A_2]$

2 Naive Bayes

1. (a) If we have infinite data drawn from the distribution, then the distribution of the sampled data will be the same as the true distribution and hence the predicted values will be equal to true values. $\hat{P}(x_1 \mid y)=P(x_1 \mid y)$ and $\hat{P}(y)=P(y)$.

(b)

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	0.08	0.09	1
1	0.02	0.81	1

(c) Error of classifier is 10%:
$P(y'\neq y)=P(y'\neq y, x_1=-1)+P(y'\neq y, x_1=1)=0.08+0.02=0.1=10\%$.

2. (a) It is given that x_2 is identical to x_1, hence, $P(x_1,x_2 \mid y)=P(x_1 \mid y)$
If x_1 and x_2 were conditionally independent, then $P(x_1,x_2 \mid y)=P(x_1 \mid y)P(x_2 \mid y)$
For the above equations to agree, we need to have $P(x_2 \mid y)=1$, But this may not be possible.
Hence x_1 and x_2 are not conditionally independent.

(b)

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	0.064	0.009	-1
-1	1	0.016	0.081	1
1	-1	0.016	0.081	1
1	1	0.004	0.729	1

(c) Error of the classifier is 11%:

$$P(y' \neq y) = P(y' \neq y, x_1 = -1, x_2 = -1) + P(y' \neq y, x_1 = -1, x_2 = 1) + P(y' \neq y, x_1 = 1, x_2 = 1) + P(y' \neq y, x_1 = 1, x_2 = -1)$$

It is given that x_1 and x_2 are identical, hence, ($x_1 = -1$ and $x_2 = 1$) cannot occur simultaneously. And ($x_1 = 1$ and $x_2 = -1$) cannot occur simultaneously.

Also, $P(y' \neq y, x_1 = -1, x_2 = -1) = P(y' \neq y, x_1 = -1) = P(y' \neq y, x_2 = -1)$

Similarly, $P(y' \neq y, x_1 = 1, x_2 = 1) = P(y' \neq y, x_1 = 1) = P(y' \neq y, x_2 = 1)$

Then, $P(y' \neq y) = P(y' \neq y, x_1 = -1) + 0 + 0 + P(y' \neq y, x_1 = 1)$

$$P(y' \neq y) = 0.09 + 0.02 = 0.11 = 11\%$$

(d) From the above problem, the performance of the naive bayse classifier decreases when the variables were duplicated. Naive bayse classifier assumes that the features are independent, but logistic regression does not make any such assumption. Logistic regression will perform better than naive bayse since it is not affected by duplication of features.

3 Naïve Bayes and Linear Classifiers

The classifier will predict the label 1 if $\Pr(y = 1|\mathbf{x}) \geq \Pr(y = 0|\mathbf{x})$. Or equivalently,

$$\frac{\Pr(\mathbf{x}|y = 1)\Pr(y = 1)}{\Pr(\mathbf{x}|y = 0)\Pr(y = 0)} \geq 1$$

The Gaussian distribution is represented by the following probability density function:

$$f(x_j \mid \mu_{j,y}, \sigma) = \frac{1}{\sqrt{2\sigma^2\pi}} e^{-\frac{(x_j - \mu_{j,y})^2}{2\sigma^2}}$$

From the naïve Bayes assumption: $\Pr(\mathbf{x}|y) = \prod_{j=0}^{d} \Pr(x_j|y)$.

$$\frac{\Pr(y = 1)}{\Pr(y = 0)} \prod_{j=0}^{d} \frac{\Pr(x_j|y = 1)}{\Pr(x_j|y = 0)} \geq 1$$

To simplify the notations, let $P(y = 1) = p$, $P(y = 0) = 1 - p$, and also substitute the probability density function in the above equation.

Let μ_{0j} be the mean of the jth variable when $y = 0$

Let μ_{1j} be the mean of the jth variable when $y = 1$

$$\frac{p}{1-p}\prod_{j=0}^{d}\frac{\frac{1}{\sqrt{2\sigma^2\pi}}e^{-\frac{(x_j-\mu_{1j})^2}{2\sigma^2}}}{\frac{1}{\sqrt{2\sigma^2\pi}}e^{-\frac{(x_j-\mu_{0j})^2}{2\sigma^2}}} \geq 1$$

$$\frac{p}{1-p}\prod_{j=0}^{d}e^{\frac{-(x_j-\mu_{1j})^2}{2\sigma^2}+\frac{(x_j-\mu_{0j})^2}{2\sigma^2}} \geq 1$$

$$\frac{p}{1-p}\prod_{j=0}^{d}e^{\frac{-(x_j-\mu_{1j})^2+(x_j-\mu_{0j})^2}{2\sigma^2}} \geq 1$$

$$\frac{p}{1-p}\prod_{j=0}^{d}e^{\frac{-x_j^2-\mu_{1j}^2+2x_j\mu_{1j}+x_j^2+\mu_{0j}^2-2x_j\mu_{0j}}{2\sigma^2}} \geq 1$$

$$\frac{p}{1-p}\prod_{j=0}^{d}e^{\frac{(\mu_{0j}^2-\mu_{1j}^2)+x_j(2\mu_{1j}-2\mu_{0j})}{2\sigma^2}} \geq 1$$

$$\frac{p}{1-p}\prod_{j=0}^{d}e^{\frac{(\mu_{0j}^2-\mu_{1j}^2)}{2\sigma^2}}e^{\frac{x_j(2\mu_{1j}-2\mu_{0j})}{2\sigma^2}} \geq 1$$

$$\frac{p}{1-p}\prod_{j=0}^{d}e^{\frac{(\mu_{0j}^2-\mu_{1j}^2)}{2\sigma^2}}\prod_{j=0}^{d}e^{\frac{x_j(2\mu_{1j}-2\mu_{0j})}{2\sigma^2}} \geq 1$$

$$log\Big(\frac{p}{1-p}\prod_{j=0}^{d}e^{\frac{(\mu_{0j}^2-\mu_{1j}^2)}{2\sigma^2}}\Big)+\sum_{j=0}^{d}x_j\frac{(2\mu_{1j}-2\mu_{0j})}{2\sigma^2} \geq 0$$

For any input **x**, the first term in this summation is a constant because it does not have any x_j terms. Let us denote it by $b = log\Big(\frac{p}{1-p}\prod_{j=0}^{d}e^{\frac{(\mu_{0j}^2-\mu_{1j}^2)}{2\sigma^2}}\Big)$. Further, let us denote $\frac{(2\mu_{1j}-2\mu_{0j})}{2\sigma^2}$ by w_j. Substituting these, we get the following expression:

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

We obtained this condition for predicting that the label is 1. This means that our classifier is a linear classifier.

4 Experiment

1. $g(\mathbf{w}) = \log(1+\exp(-y_i\mathbf{w}^T\mathbf{x}_i))$

   $\frac{dg(\mathbf{w})}{d\mathbf{w}} = \frac{1}{1+\exp(-y_i\mathbf{w}^T\mathbf{x}_i)}\exp(-y_i\mathbf{w}^T\mathbf{x}_i)(-y_i\mathbf{x}_i)$

   $\frac{dg(\mathbf{w})}{d\mathbf{w}} = \frac{-y_i\mathbf{x}_i}{\exp(y_i\mathbf{w}^T\mathbf{x}_i)+1}$

2. If the entire dataset is composed of a single example, then the onjective function will be:

$$J = log(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i)) + \frac{1}{\sigma^2}\mathbf{w}^T\mathbf{w}$$

The gradient of this objective with respect to the weight vector is:

$$\frac{dJ(\mathbf{w})}{d\mathbf{w}} = \frac{-y_i \mathbf{x}_i}{\exp(y_i \mathbf{w}^T \mathbf{x}_i) + 1} + \frac{2\mathbf{w}}{\sigma^2}$$

3. Pseudo code for the stochastic gradient algorithm using the gradient from previous part:

initialize w

for *each epoch, t = 0, 1, 2, . . . T)* **do**

  Shuffle the data

  for *each training example* (x_i, y_i)*)* **do**

    $\mathbf{w}^{t+1} = \mathbf{w}^t - r(\frac{-y_i x_i}{\exp(y_i \mathbf{w}^T \mathbf{x}_i)+1} + \frac{2\mathbf{w}^t}{\sigma^2})$

  end

end

return w

4. Results of 10-fold Cross validation with 5 epochs and learning rate of 0.01 are:

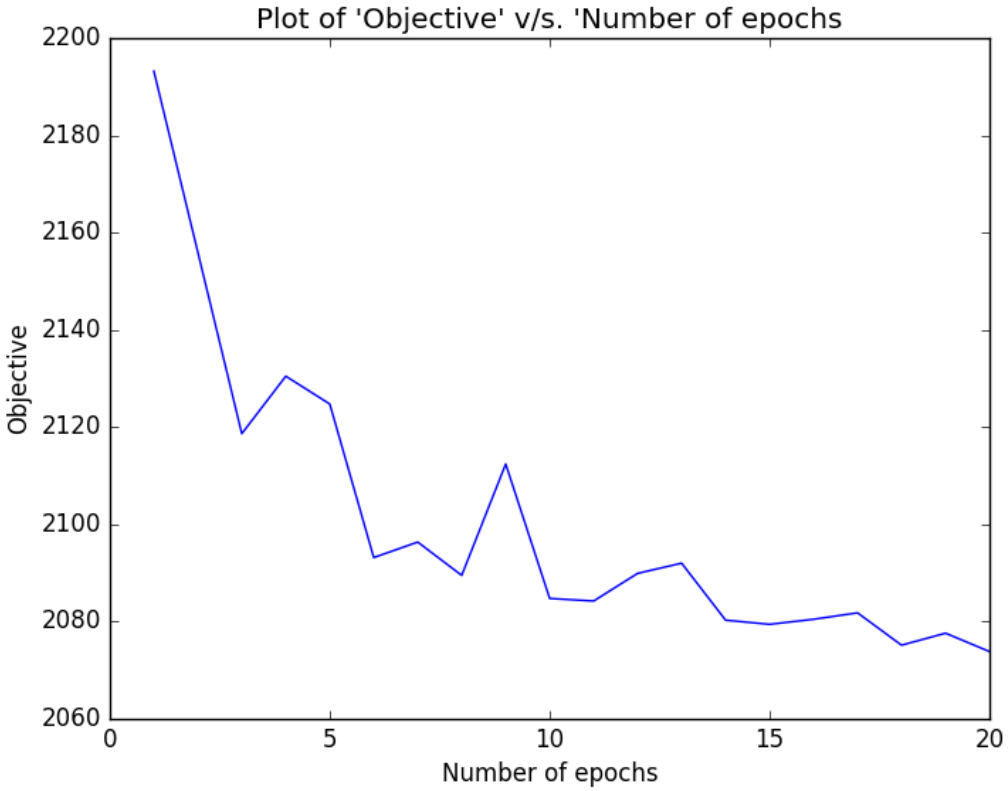
σ	Cross validation Accuracy(%)
1	75.586
50	84.019
100	84.316
125	84.051
150	84.393
175	84.126
200	84.390
225	84.191
250	84.003

Best Sigma through Cross Validation is 150, with Average accuracy of 84.393%

Number of Epochs used for final run: 20

Accuracy on training data: 85.27%

Accuracy on test data: 84.64%



From the above figure, it is seen that the Magnitude of the Objective decreases with increase in the number of epochs.