

CSE 575 Homework 1

Andrew Dudley (addudley@asu.edu)

February 15, 2018

Question 1.

Suppose that in your coin flip experiment, you observed a set of α_H heads and α_T tails. Let θ denote the probability of observing heads, whose prior distribution follows $Beta(\beta_H, \beta_T)$, where β_H and β_T are two positive parameters.

- (a) Prove that the posterior distribution $P(\theta|D)$ follows $Beta(\beta_H + \alpha_H, \beta_T + \alpha_T)$

The posterior distribution $P(\theta|D) = \frac{P(D|\theta)P(\theta)}{P(D)}$. We're told that

$$P(\theta) = \frac{\theta^{\beta_H-1}(1-\theta)^{\beta_T-1}}{B(\beta_H, \beta_T)} \sim Beta(\beta_H, \beta_T),$$

where $B(\beta_H, \beta_T)$ is the beta function.

Note that $P(D|\theta)$ is the likelihood function, which for a Bernoulli experiment is

$$\theta^{\alpha_H}(1-\theta)^{\alpha_T}.$$

Putting these together, we get

$$P(\theta|D) = \frac{\theta^{\beta_H+\alpha_H-1}(1-\theta)^{\beta_T+\alpha_T-1}}{P(D)B(\beta_H, \beta_T)},$$

and because both $P(D)$ and $B(\beta_H, \beta_T)$ are normalizing constants (they don't rely on θ), we can rewrite the equation as

$$P(\theta|D) = \frac{\theta^{\beta_H+\alpha_H-1}(1-\theta)^{\beta_T+\alpha_T-1}}{B(\beta_H + \alpha_H, \beta_T + \alpha_T)},$$

From here, we can see that

$$P(\theta|D) \sim Beta(\beta_H + \alpha_H, \beta_T + \alpha_T).$$

(b) What is the mean of $P(\theta|D)$?

For notational convenience, let $a = \beta_H + \alpha_H$ and $b = \beta_T + \alpha_T$. As they say, statistics is the practice of replacing expectations with averages, so the mean of $P(\theta|D)$ is the same as $\mathbb{E}[P(\theta|D)]$. And, of course, because this value is a probability, we can bound the integral by $[0, 1]$.

$$\begin{aligned}\mathbb{E}[P(\theta|D)] &= \int_0^1 \theta P(\theta|D) d\theta \\ &= \frac{1}{B(a, b)} \int_0^1 \theta^a (1 - \theta)^{b-1} d\theta\end{aligned}$$

Here, we can see that the integral is, itself, a beta function.

$$\begin{aligned}&= \frac{B(a+1, b)}{B(a, b)} \\ &= \frac{\Gamma(a+1)\Gamma(b)\Gamma(a+b)}{\Gamma(a)\Gamma(b)\Gamma(a+b+1)} \\ &= \frac{a}{a+b} \\ &= \frac{\beta_H + \alpha_H}{\beta_H + \alpha_H + \beta_T + \alpha_T}\end{aligned}$$

(c) What is the MAP estimator $\hat{\theta}_{MAP}$ of θ ?

$$\begin{aligned}\frac{d}{d\theta} \mathcal{L}(\theta) &= \frac{d}{d\theta} \ln \left(\theta^{\beta_H + \alpha_H - 1} (1 - \theta)^{\beta_T + \alpha_T - 1} \right) \\ &= (\beta_H + \alpha_H - 1) \left[\frac{d}{d\theta} \ln \theta \right] + (\beta_T + \alpha_T - 1) \left[\frac{d}{d\theta} \ln(1 - \theta) \right] \\ &= \frac{\beta_H + \alpha_H - 1}{\theta} - \frac{\beta_T + \alpha_T - 1}{1 - \theta}\end{aligned}$$

And to find a critical point (the minimum value), we set the derivative to 0.

$$\begin{aligned}\frac{\beta_H + \alpha_H - 1}{\theta} - \frac{\beta_T + \alpha_T - 1}{1 - \theta} &= 0 \\ \hat{\theta}_{MAP} &= \frac{\beta_H + \alpha_H - 1}{\beta_H + \alpha_H + \beta_T + \alpha_T - 2}\end{aligned}$$

Question 2.

For this question, assume that $x_1, \dots, x_N \in \mathbb{R}$ are i.i.d. from a normal distribution.

- (a) Let $\hat{\mu}_{MLE}$ denote the MLE of μ . Prove that $\hat{\mu}_{MLE}$ is unbiased.

We'll first need to determine the equation for $\hat{\mu}_{MLE}$, then we'll need to compare its expectation to the population mean μ . Let $D = \{x_1, \dots, x_n\}$. Then, because the data is assumed independent, the likelihood of μ with respect to D can be written as

$$P(D|\mu) = \prod_{i=1}^n p(x_i|\mu)$$

Furthermore, we know that the samples are from a normal distribution, giving us

$$p(x|\mu) = \frac{1}{(2\pi\sigma^2)^{\frac{1}{2}}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

To simplify finding the derivative, take the log of the likelihood function

$$\mathcal{L}(P(D|\mu)) = \sum_{i=1}^n \ln(p(x_i|\mu))$$

Now we'll find the MLE of μ by taking the derivative of the log-likelihood, setting it to 0, and solving for μ

$$\begin{aligned} \frac{d\mathcal{L}}{d\mu} &= \sum_{i=1}^n \frac{d}{d\mu} \ln(p(x_i|\mu)) &&= 0 \\ &= - \sum_{i=1}^n \frac{d}{d\mu} \left(\frac{1}{2} \ln(2\pi\sigma^2) + \frac{1}{2\sigma^2} (x_i - \mu)^2 \right) &&= 0 \\ &= \sum_{i=1}^n (x_i - \hat{\mu}) &&= 0 \\ \hat{\mu} &= \frac{1}{n} \sum_{i=1}^n x_i \end{aligned}$$

Now that we know the equation for $\hat{\mu}$, we'll find its expectation.

$$\begin{aligned}\mathbb{E}[\hat{\mu}] &= \mathbb{E}\left[\frac{1}{n} \sum_{i=1}^n x_i\right] \\ &= \frac{1}{n} \sum_{i=1}^n \mathbb{E}[x_i] \\ &= \frac{1}{n} \sum_{i=1}^n \mu \\ &= \mu\end{aligned}$$

Therefore, $\hat{\mu}_{MLE}$ is unbiased.

(b) If the true value of μ is unknown, then the MLE estimate of σ^2 is

$$\sigma_{MLE}^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \hat{\mu}_{MLE})^2$$

Prove that σ_{MLE}^2 is biased.

Before we begin, there are a few derivations that we'll want to have on hand when completing this proof. To start, we'll derive the definition of variance σ^2 .

$$\begin{aligned}\sigma^2 &= \mathbb{E}[(X - \mu)^2] \\ &= \mathbb{E}[X^2 - 2X\mu + \mu^2] \\ &= \mathbb{E}[X^2] - 2\mu\mathbb{E}[X] + \mu^2 \\ &= \mathbb{E}[X^2] - \mu^2 \\ &= \mathbb{E}[X^2] - (\mathbb{E}[X])^2\end{aligned}$$

A little bit of algebraic manipulation of these equations gives us

$$\mathbb{E}[X^2] = \sigma^2 + \mu^2, \tag{1}$$

Also note that σ^2 is a function of X . If we pass into it $\hat{\mu}$, we get

$$\begin{aligned}\sigma_{\hat{\mu}}^2 &= \mathbb{E}[\hat{\mu}^2] - (\mathbb{E}[\hat{\mu}])^2 \\ &= \mathbb{E}[\hat{\mu}^2] - \mu^2\end{aligned} \tag{2}$$

We can also derive the variance of $\hat{\mu}$ as follows

$$\begin{aligned}
 \sigma_{\hat{\mu}}^2 &= \text{Var}\left(\frac{x_1 + \cdots + x_n}{n}\right) \\
 &= \frac{1}{n^2} \text{Var}(x_1 + \cdots + x_n) \\
 &= \frac{1}{n^2} [\text{Var}(x_1) + \cdots + \text{Var}(x_n)] \\
 &= \frac{\sigma^2}{n}
 \end{aligned} \tag{3}$$

Using (2) and (3), we arrive at the equation

$$\mathbb{E}[\hat{\mu}^2] = \frac{\sigma^2}{n} + \mu^2 \tag{4}$$

With these derivations in our toolbox, we can now begin the proof that σ_{MLE}^2 is biased. Again, we'll do so by comparing the expectation of this value to the population variance.

$$\begin{aligned}
 \mathbb{E}[\sigma_{MLE}^2] &= \mathbb{E}\left[\frac{1}{n} \sum (x_i - \hat{\mu})^2\right] \\
 &= \frac{1}{n} \mathbb{E}\left[\sum (x_i^2 - 2x_i\hat{\mu} + \hat{\mu}^2)\right] \\
 &= \frac{1}{n} \mathbb{E}\left[\sum x_i^2 - 2\hat{\mu} \sum x_i + n\hat{\mu}^2\right] \\
 &= \frac{1}{n} \mathbb{E}\left[\sum x_i^2 - 2n\hat{\mu}^2 + n\hat{\mu}^2\right] \\
 &= \frac{1}{n} \left(\sum \mathbb{E}[x_i^2] - n\mathbb{E}[\hat{\mu}^2]\right) \\
 &= \frac{1}{n} (n\mathbb{E}[X^2] - n\mathbb{E}[\hat{\mu}^2]) \\
 &= \mathbb{E}[X^2] - \mathbb{E}[\hat{\mu}^2]
 \end{aligned} \tag{5}$$

Do the expectations in (5) look familiar? They should! Indeed, they are the two expectations that we derived earlier. By plugging the expressions from (1) and (4) into (5), we get

$$\begin{aligned}
 \mathbb{E}[\sigma_{MLE}^2] &= \sigma^2 + \mu^2 - \frac{\sigma^2}{n} - \mu^2 \\
 &= \sigma^2 - \frac{\sigma^2}{n} \\
 &= \left(\frac{n-1}{n}\right) \sigma^2
 \end{aligned}$$

$\left(\frac{n-1}{n}\right) \sigma^2 \neq \sigma^2$, therefore σ_{MLE}^2 is biased.

Question 3.

- (a) How many independent parameters would there be for the Naive Bayes classifier trained with the given data? What are they?

There would be **thirteen independent parameters**. Note first that RID is a nominal attribute and buys_computer is the dependent (target) attribute, leaving us with age, income, student, and credit_rating as the attributes of interest.

An assumption is made with Naive Bayes that

$$P(X_i | X_{1,\dots,i-1}, x_{i+1}, \dots, n, Y) = P(X_i | Y) \forall i \in \{1, \dots, n\}$$

A Bayes classifier can be represented as

$$P(Y_c | X_{1..n}) = \frac{P(X_{1..n} | Y_c) P(Y_c)}{\sum_Y P(X_{1..n} | Y) P(Y)}$$

Given the conditional independence assumption of Naive Bayes, this can be simplified to

$$P(Y_c | X_{1..n}) = \frac{\prod_i P(X_i | Y_c) P(Y_c)}{\sum_Y \prod_i P(X_i | Y) P(Y)}$$

Let m_i represent the number of discrete categories in the variable x_i and C represent the number of discrete classes of the target attribute Y . Then, given $P(X = X_{ij} | Y = y_c)$ where $j \in \{1, \dots, m_i\}$ and $c \in \{1, \dots, C\}$, each independent variable x_i contributes $m_i - 1$ independent parameters for each class y_c . Note that, given $m_i - 1$ parameters for the class conditional of an attribute, the final parameter of that class conditional is simply the difference between 1 and the sum of those parameters, and is therefore not independent.

Similarly, we must account for the independent parameters contributed by the prior $P(Y)$, which will be $C - 1$ parameters.

Thus, the equation for the total number of independent parameters in a Naive Bayes model will be

$$C * \sum_{i=1}^n (m_i - 1) + C - 1$$

Plugging in the attributes from the data provided, the number of independent parameters is **thirteen**.

(b)

$$\begin{aligned}
 P(\text{age} = \text{"youth"} | \text{buys_computer} = \text{"no"}) &= 3/5, P(\text{age} = \text{"middle_aged"} | \text{buys_computer} = \text{"no"}) = 0 \\
 P(\text{age} = \text{"youth"} | \text{buys_computer} = \text{"yes"}) &= 2/9, P(\text{age} = \text{"middle_aged"} | \text{buys_computer} = \text{"yes"}) = 4/9 \\
 P(\text{income} = \text{"low"} | \text{buys_computer} = \text{"no"}) &= 1/5, P(\text{income} = \text{"medium"} | \text{buys_computer} = \text{"no"}) = 2/5 \\
 P(\text{income} = \text{"low"} | \text{buys_computer} = \text{"yes"}) &= 3/9, P(\text{income} = \text{"medium"} | \text{buys_computer} = \text{"yes"}) = 4/9 \\
 P(\text{student} = \text{"no"} | \text{buys_computer} = \text{"no"}) &= 4/5 \\
 P(\text{student} = \text{"no"} | \text{buys_computer} = \text{"yes"}) &= 3/9 \\
 P(\text{credit_rating} = \text{"fair"} | \text{buys_computer} = \text{"no"}) &= 2/5 \\
 P(\text{credit_rating} = \text{"fair"} | \text{buys_computer} = \text{"yes"}) &= 6/9 \\
 P(\text{buys_computer} = \text{"yes"}) &= 9/14
 \end{aligned}$$

(c) Given a new person with features $x = (\text{youth}, \text{medium}, \text{yes}, \text{fair})$, what is $P(Y = \text{yes} | x)$?

$$\frac{\frac{2}{9} * \frac{4}{9} * \frac{6}{9} * \frac{6}{9} * \frac{9}{14}}{\frac{2}{9} * \frac{4}{9} * \frac{6}{9} * \frac{6}{9} * \frac{9}{14} + \frac{3}{5} * \frac{2}{5} * \frac{1}{5} * \frac{2}{5} * \frac{5}{14}} = 0.80451$$

Therefore, NB would classify this input at $Y = \text{yes}$.

Question 4.

Suppose we have two positive examples $x_1 = (1, 0)$ and $x_2 = (0, -1)$, and two negative samples $x_3 = (0, 1)$ and $x_4 = (-1, 0)$. Apply standard gradient ascent method to train a logistic regression classifier (without any regularization term). Initialize the weight vector with two different value and set $w_0^0 = 0$. Would the final weight vector (w^*) be the same for the two different initial values? What are the values? You may assume the learning rate to be a positive real constant η .

For notational and implementation convenience, first append a 1 to each sample vector

$$X = \left[\begin{array}{ccc} 1 & 0 & 1 \\ 0 & -1 & 1 \\ 0 & 1 & 1 \\ -1 & 0 & 1 \end{array} \right] \Bigg\}_m, \quad \theta = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\mathbf{z} = X\theta = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad \hat{\mathbf{p}} = \text{sig}((z)) = \frac{1}{1 + e^{-z}} = \begin{bmatrix} 0.5 \\ 0.5 \\ 0.5 \\ 0.5 \end{bmatrix}$$

\hat{p} represents the model's estimated probabilities of a positive class in vectorized form. When performing prediction, a threshold value of 0.5 is set such that the predicted class is 1 when $p > 0.5$ and 0 when $p \leq 0.5$. This threshold effectively draws a discriminant line in the input space. When training, however, The cross-entropy loss is used as a cost function on the estimated probability. When training with batches of data, the total loss is simply the average cost over the training data. This allows us to write the cost function as

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^m y^{(i)} \log(\hat{p}^{(i)}) + (1 - y^{(i)}) \log(1 - \hat{p}^{(i)})$$

To minimize the loss, we can calculate the gradient of $J(\theta)$ with respect to θ and then update θ in the negative direction of that gradient. Iterating over this process ad infinitum will converge to the optimal parameters (ignoring issues of perfect separation, etc).

$$\frac{\delta J(\theta)}{\delta \theta_j} = \frac{\delta J}{\delta \hat{p}} \frac{\delta \hat{p}}{\delta z} \frac{\delta z}{\delta \theta_j}$$

$$\begin{aligned} \frac{\delta J}{\delta \hat{p}} &= -\frac{1}{m} \sum_{i=1}^m \left[\frac{y^{(i)}}{\hat{p}^{(i)}} - \frac{1 - y^{(i)}}{1 - \hat{p}^{(i)}} \right] \\ &= -\frac{1}{m} \sum_{i=1}^m \left[\frac{y^{(i)} - \hat{p}^{(i)}}{\hat{p}^{(i)}(1 - \hat{p}^{(i)})} \right] \end{aligned}$$

$$\begin{aligned} \frac{\delta \hat{p}}{\delta z} &= \text{sig}(z)(1 - \text{sig}(z)) \\ &= \hat{p}(1 - \hat{p}) \end{aligned}$$

$$\frac{\delta J}{\delta \hat{p}} \frac{\delta \hat{p}}{\delta z} = \frac{1}{m} \sum_{i=1}^m \hat{p}^{(i)} - y^{(i)}$$

$$\frac{\delta z}{\delta \theta_j} = x_j$$

$$\frac{\delta J}{\delta \hat{p}} \frac{\delta \hat{p}}{\delta z} \frac{\delta z}{\delta \theta_j} = \frac{1}{m} \sum_{i=1}^m (\hat{p}^{(i)} - y^{(i)}) x_j^{(i)}$$

From this result, it is clear that the only calculation we need to complete prior to updating our weights is the sigmoid function. It's also worth noting

that this result can be written in matrix form, which allows for efficient computation when implementing

$$\frac{\delta J}{\delta \hat{p}} \frac{\delta \hat{p}}{\delta z} \frac{\delta z}{\delta \theta} = \frac{1}{m} X^T (\hat{p} - y)$$

Now that we know the process of logistic regression, what can we say about the parameters? What do they represent?

Unfortunately, interpreting the parameters learned isn't nearly as straight forward as it is in linear regression models, even though what we're ultimately fitting is a discriminant line in the feature space. However, if we consider the threshold value 0.5 and its relation of the the input to the sigmoid function z , we know that when $z > 0$, $\text{sig}(z) > 0.5$. Likewise, when $z \leq 0$, $\text{sig}(z) \leq 0.5$. From this, it is obvious that the parameters we are learning are for a line in the feature space at form at $z = 0$. Scaling the coefficients of z by any scalar value would result in the same line, so the there are infinite values that θ could converge to, and this will depend on the the values that the vector is initiated with.

In this example, the values will ultimately converge to $\theta_1 = -\theta_2$ (s.t. $\theta_1 \in \mathcal{R}$), and $b = 0$

$$\theta_1 x_1 + \theta_2 x_2 + b = 0$$

Question 5.

Naive Bayes Classifier and Logistic Regression

- (a) - How many independent parameters are there in this Gaussian Naive Bayes classifier? What are they?

Unlike Question 3, the class-conditionals in this problem aren't discrete, they're probability distribution functions. There are infinite possible values of x_i , and therefore no notion have having " $m_i - 1$ " independent variables per attribute. Instead, each attribute has a corresponding μ and σ^2 parameter for each of the two possible class labels. However, because the target attribute is still discrete (boolean), one of the priors constitutes an independent parameter and the other does not.

Thus, there are 9 independent parameters in this problem.

- Can we translate w in to the parameters of an equivalent Gaussian Naive Bayes classifier without any extra assumption?

Note: Tom Mitchell's Machine Learning book was referenced for this problem. *Machine Learning, Tom Mitchell, McGraw Hill, 1997.*

To determine a relationship between Gaussian Naive Bayes and logistic regression, it would help to first have them in a similar form.

The posterior probability of the positive class in Naive Bayes is, by definition

$$\frac{P(D|Y=1)P(Y=1)}{P(D|Y=1)P(Y=1) + P(D|Y=0)P(Y=0)}$$

By dividing the numerator and denominator by the numerator, we get

$$\frac{1}{1 + \frac{P(X|Y=0)P(Y=0)}{P(X|Y=1)P(Y=1)}}$$

Which can be written as

$$\frac{1}{1 + e^{\ln\left(\frac{P(X|Y=0)P(Y=0)}{P(X|Y=1)P(Y=1)}\right)}} = \frac{1}{1 + e^{\ln\left(\frac{P(Y=0)}{P(Y=1)}\right) + \sum_i \ln\left(\frac{P(X_i|Y=0)}{P(X_i|Y=1)}\right)}}$$

Notice that $\frac{P(Y=0)}{P(Y=1)}$ is the definition of odds, and can be written as $\frac{p}{1-p}$.

$$\frac{1}{1 + e^{\ln\left(\frac{p}{1-p}\right) + \sum_i \ln\left(\frac{P(X_i|Y=0)}{P(X_i|Y=1)}\right)}}$$

Now, let's unpack the summation to reveal the parameters of the NB classifier...

$$\begin{aligned} \sum_i \ln \frac{P(X_i|Y=0)}{P(X_i|Y=1)} &= \sum_i \ln \frac{\frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(\frac{-(X_i - \mu_{i0})^2}{2\sigma_i^2}\right)}{\frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(\frac{-(X_i - \mu_{i1})^2}{2\sigma_i^2}\right)} \\ &= \sum_i \left(\frac{\mu_{i0} - \mu_{i1}}{\sigma_i^2} X_i + \frac{\mu_{i1}^2 - \mu_{i0}^2}{2\sigma_i^2} \right) \end{aligned}$$

Plugging the summation back into the original equation, we have

$$\frac{1}{1 + e^{\ln\left(\frac{p}{1-p}\right) + \sum_i \left(\frac{\mu_{i0} - \mu_{i1}}{\sigma_i^2} X_i + \frac{\mu_{i1}^2 - \mu_{i0}^2}{2\sigma_i^2}\right)}}$$

From here, we can see in the exponent that we have a linear combination of the X_i vectors in terms of the parameters of Naive Bayes, along with a constant value

$$w_0 = \ln\left(\frac{p}{1-p}\right) + \sum_i \frac{\mu_{i1}^2 - \mu_{i0}^2}{2\sigma_i^2}$$

$$w_i = \frac{\mu_{i0} - \mu_{i1}}{\sigma_i^2}$$

(b) Implementation of Naive Bayes and Logistic Regression. Compare the two approaches on the bank note authentication dataset. Implement a Gaussian Naive Bayes classifier and a logistic regression classifier. Please write your code from scratch. For the NB classifier, assume the $P(x_i|y) \sim \mathcal{N}(\mu_{i,k}, \sigma_{i,k})$ where x_i is a feature in the bank note data and y is the class label. Use three-fold cross-validation to split the data and train/test your models.

- 5.1)** For each algorithm: briefly describe how you implement it by giving the pseudocode. The pseudocode must include equations for estimating the model parameters and for classifying a new sample.
- 5.2)** Plot a learning curve: The accuracy vs the size of the training set. Plot 6 points for the curve, using $[0.01, 0.01, 0.05, 0.1, 0.625, 1]$ random fractions of your training set, and testing on the full test set each time. Average your results over 5 runs using each random fraction of the training set. Plot both the Naive Bayes and the logistic regression learning curves on the same figure.

Algorithm 1 Logistic regression

```

procedure TRAIN( $X, y, \theta$ ) ▷ Train the model
  for  $i \leftarrow 1, 1000$  do
     $\mathbf{z} \leftarrow X\theta$ 
     $\mathbf{p} \leftarrow \text{sigmoid}(\mathbf{z})$ 
     $\Delta\theta = \text{mean}(\text{diag}(\mathbf{p} - \mathbf{y})X)$ 
     $\theta = \theta - \Delta\theta$ 
  end for
  return  $\theta$ 
end procedure

procedure PREDICT( $x, \theta$ )
   $z \leftarrow x \cdot \theta$ 
   $p \leftarrow \text{sigmoid}(z)$ 
  if  $p > 0.5$  then
    return 1
  else
    return 0
  end if
end procedure

```

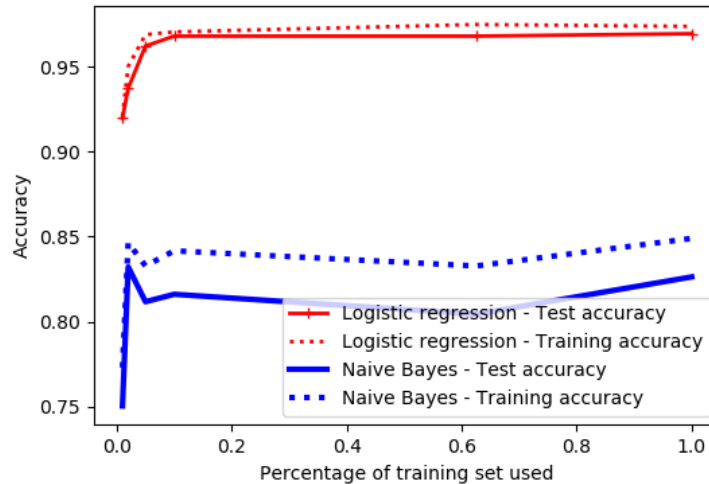
Algorithm 2 Naive Bayes

```

procedure TRAIN( $X, y$ )
   $\mu_{j,c} \leftarrow \text{mean}(\{x_{i,j} | y_i = c\}) \quad \forall c \in \text{unique}(y)$ 
   $\sigma_{j,c}^2 \leftarrow \text{variance}(\{x_{i,j} | y_i = c\}) \quad \forall c \in \text{unique}(y)$ 
   $\mathbf{p} \leftarrow \text{length}(\{y_i | y_i = c\}) / \text{length}(\mathbf{y}) \quad \forall c \in \text{unique}(y)$ 
  return  $\mu, \sigma, p$ 
end procedure

procedure PREDICT( $x, \mu, \sigma, \mathbf{p}$ )
   $\mathcal{L}_c = \prod_j \mu_{j,c}$ 
  return  $\text{argmax}_{\mathbf{c}}(\mathcal{L}_{\mathbf{c}} * \mathbf{p}_{\mathbf{c}})$ 
end procedure

```



- 5.3)** Show the power of the generative model: Use your trained Naive Bayes classifier to generate 500 examples from class $y = 1$. Report the mean and variance of the generated examples and the corresponding training data (for each fold, over 1 run). Compare the results with the parameters learned from the training set.

My implementation uses the unbiased variance estimator (sample variance). The learned parameters and the parameters of the generated data are very similar, which is to be expected, as the generated data is generated using the learned parameters.

```
=====
Naive Bayes -- Fold 1
-----
Learned parameters
      0      1      2      3
mean -1.705562 -1.131651  2.002437 -1.155452
var   3.155517 27.664670 26.316116  4.087141
-----
Generated data parameters
      0      1      2      3
mean -1.518889 -1.175661  2.102247 -1.285719
var   3.052246 29.534087 24.755123  3.625868
-----
=====
```

Naive Bayes -- Fold 2

 Learned parameters

	0	1	2	3
mean	-1.817263	-1.210659	2.218032	-1.265630
var	3.241277	29.411025	28.369520	4.246753

 Generated data parameters

	0	1	2	3
mean	-1.813054	-0.814665	1.896032	-1.317391
var	3.042551	27.059326	25.419083	4.314348

=====

Naive Bayes -- Fold 3

 Learned parameters

	0	1	2	3
mean	-1.722668	-1.175853	2.033345	-1.213837
var	3.200858	27.628852	27.450893	4.258903

 Generated data parameters

	0	1	2	3
mean	-1.663900	-0.964194	2.216354	-1.252897
var	3.286815	27.380846	28.067771	4.036993
