Classification I

COMP9417, 23T1

1. Bayes Rule

2. 1 (a, b, c)

3. Naive Bayes Classification

4. 2 (a, b, c, d, e)

5. Logistic Regression

6. 3 (a, b, c)

Section 1

Bayes Rule

Bayes Rule

The basic Bayes rule lets us express conditional probabilities in a different way:

$$P(A|B) = \frac{P(A) \cdot P(B|A)}{P(B)}$$

Also, recall the law of total probability, if we have an event A which is conditionally dependent on a sample space B, we can calculate the marginal probability $P(A)$:

$$P(A) = \sum_i P(B_i) \cdot P(A|B_i)$$

Section 2

1 (a, b, c)

1 (a, b, c)

Assume that the probability of a certain disease is 0.01. The probability of testing positive given that a person is infected with the disease is 0.95, and the probability of testing positive given that the person is not infected with the disease is 0.05.

a) Calculate the probability of testing positive.

If we define D to represent the disease being present, T to represent a positive test.

From the problem we have the probabilities:

- $P(D) = 0.01$
- $P(T|D) = 0.95$
- $P(T|\neg D) = 0.05$

we need to calculate $P(T)$.

We can simply apply the law of total probability:

$$\begin{aligned} P(T) &= P(D) \cdot P(T|D) + P(\neg D) \cdot P(T|\neg D) \\ &= 0.01 \cdot 0.95 + (1 - 0.01) \cdot 0.05 \\ &= 0.059 \end{aligned}$$

b) Calculate the probability of being infected with the disease, given that the test is positive.

We need to find $P(D|T)$.

We can use what we know and apply Bayes rule:

$$\begin{aligned} P(D|T) &= \frac{P(T|D)P(D)}{P(T)} \\ &= \frac{0.95 \times 0.01}{0.059} \\ &= 0.16 \end{aligned}$$

c) Now assume that you test the individual a second time, and the test comes back positive (so two tests, two positives). Assume that conditional on having the disease, the outcomes of the two tests are independent, what is the probability that the individual has the disease? (note, conditional independence in this case means that $P(TT|D) = P(T|D)P(T|D)$, and not $P(TT) = P(T)P(T)$.) You may also assume that the test outcomes are conditionally independent given not having the disease.

We are trying to find $P(D|TT)$.

Let's get this into a nicer form,

$$
\begin{aligned}
P(D|TT) &= \frac{P(TT|D)P(D)}{P(TT)} \\
&= \frac{P(T|D)^2 P(D)}{P(TT)}
\end{aligned}
$$

we see that we don't have the value $P(TT)$.

Apply the law of total probability:

$$
\begin{aligned}
P(TT) &= P(D) \cdot P(TT|D) + P(\neg D) \cdot P(TT|\neg D) \\
&= 0.01 \cdot (0.95)^2 + (1 - 0.01) \cdot (0.05)^2 \\
&= 0.0115
\end{aligned}
$$

Now, we can sub our new values in for the answer:

$$\begin{aligned}P(D|TT) &= \frac{P(T|D)^2 P(D)}{P(TT)}\\ &= \frac{(0.95)^2 \times 0.01}{0.0115}\\ &= 0.7848\end{aligned}$$

Section 3

Naive Bayes Classification

Naive Bayes Classification

The naive Bayes classifier solves the problem:

$$\begin{aligned}\hat{y}_i &= \underset{k \in \{1,\ldots,K\}}{\arg\max}\; p(C_k | x_i) \\ &= \underset{k \in \{1,\ldots,K\}}{\arg\max}\; p(C_k) \prod_{i=1}^{p} p(x_i | C_k)\end{aligned}$$

we are essentially trying to estimate the class of a data point based on the prior and posterior probabilities estimated from our data.

Section 4

2 (a, b, c, d, e)

2 (a, b, c, d, e)

a) What is probabilistic classification? How does it differ from non-probabalistic classification methods?

In probabilistic methods, we estimate probabilities from our dataset and use these to learn a possible distribution for our data. When using probabilities, the most optimal choice of parameters are the parameters which occur with the highest probability/likelihood. In contrast, non-parametric methods mean that we define some empirical 'loss' function to estimate the error of our estimate in comparison to the assumed true pattern of the data. Then, our solution is the one which minimises this loss.

b) What is the Naive Bayes assumption and why do we need it?

The naive Bayes assumption is the assumption that our data is conditionally independent $x_i \perp x_j | c_k$ for all $i \neq j$.

What does this let us do?

Remember, we are trying to maximise $p(c_k|\mathbf{x})$, so:

$$p(c_k|\mathbf{x}) = \frac{p(c_k)p(\mathbf{x}|c_k)}{p(\mathbf{x})} = \frac{p(c_k)p(\mathbf{x}|c_k)}{\sum_{k=1}^{K} p(c_k)p(\mathbf{x}|c_k)}$$

once we model the numerator, the denominator can be calculated for the entire dataset.

So, how do we model the numerator?

We can use the product rule and decompose the probabilities:

$$\begin{aligned}
p(\mathbf{x}|c_k)p(c_k) &= p(\mathbf{x}, c_k) \\
&= p(x_1, \ldots, x_p, c_k) \\
&= p(x_1|x_2 \ldots, x_p, c_k)p(x_2, \ldots, x_p, c_k) \\
&= p(x_1|x_2 \ldots, x_p, c_k)p(x_2|x_3, \ldots, x_p, c_k)p(x_3|x_4, \ldots, x_p, c_k) \cdots p(x_p|c_k)p(c_k)
\end{aligned}$$

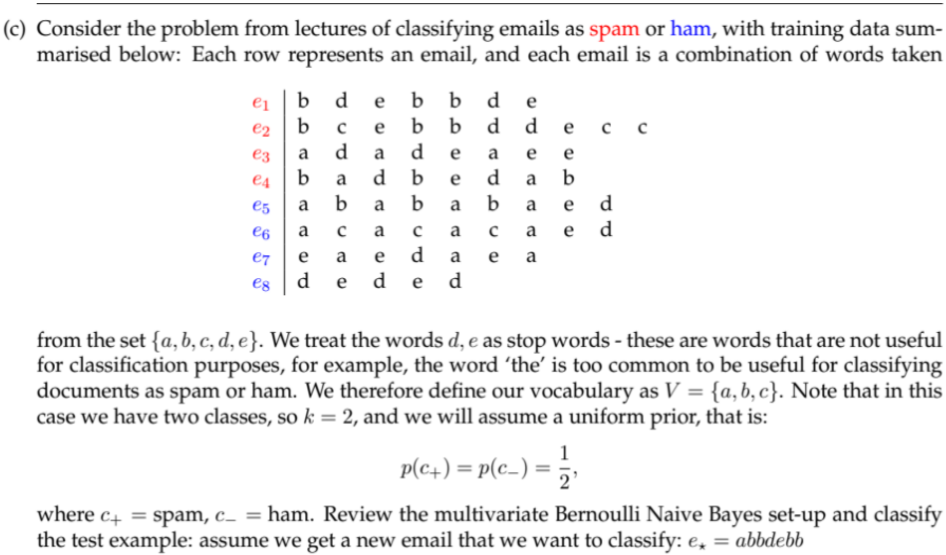
We have:

$$p(\mathbf{x}|c_k)p(c_k) = p(x_1|x_2 \ldots, x_p, c_k)p(x_2|x_3, \ldots, x_n, c_k)p(x_3|x_4, \ldots, x_p, c_k) \cdots p(x_p|c_k)p(c_k)$$

How do we apply the naive Bayes assumption?

$$\begin{aligned} p(\mathbf{x}|c_k)p(c_k) &= p(x_1|c_k)p(x_2|c_k)p(x_3|c_k) \cdots p(x_p|c_k)p(c_k) \\ &= p(c_k) \prod_{i=1}^{p} p(x_i|c_k) \end{aligned}$$

so, instead of estimating an p dimensional distribution as we did with conditionals, applying the assumption means that we now have p independent 1-dimensional distributions to estimate.



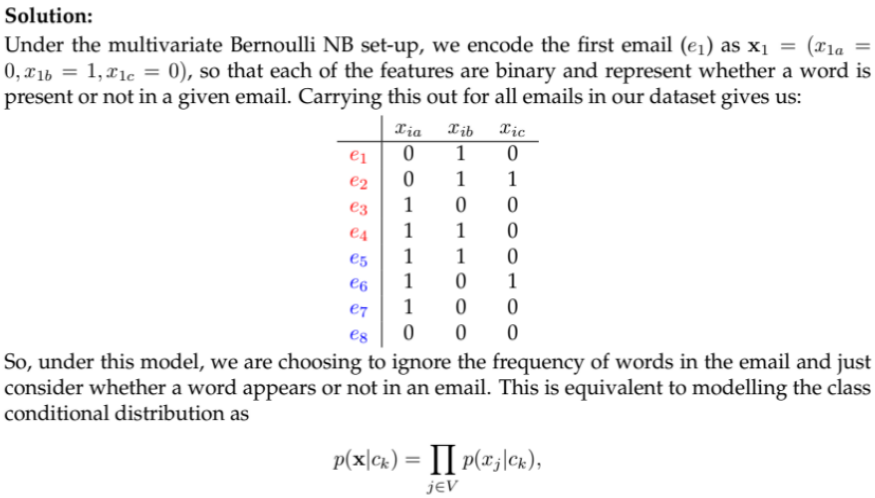
(c) Consider the problem from lectures of classifying emails as spam or ham, with training data summarised below: Each row represents an email, and each email is a combination of words taken

e_1	b	d	e	b	b	d	e			
e_2	b	c	e	b	b	d	d	e	c	c
e_3	a	d	a	d	e	a	e	e		
e_4	b	a	d	b	e	d	a	b		
e_5	a	b	a	b	a	b	a	e	d	
e_6	a	c	a	c	a	c	a	e	d	
e_7	e	a	e	d	a	e	a			
e_8	d	e	d	e	d					

from the set $\{a, b, c, d, e\}$. We treat the words d, e as stop words - these are words that are not useful for classification purposes, for example, the word 'the' is too common to be useful for classifying documents as spam or ham. We therefore define our vocabulary as $V = \{a, b, c\}$. Note that in this case we have two classes, so $k = 2$, and we will assume a uniform prior, that is:

$$p(c_+) = p(c_-) = \frac{1}{2},$$

where $c_+ =$ spam, $c_- =$ ham. Review the multivariate Bernoulli Naive Bayes set-up and classify the test example: assume we get a new email that we want to classify: $e_\star = abbdebb$



Solution:

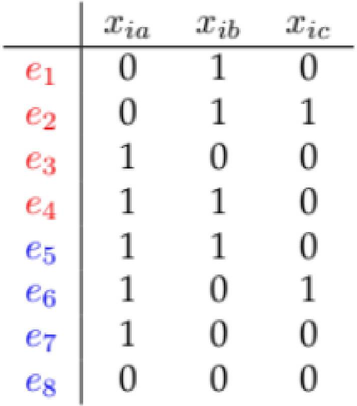
Under the multivariate Bernoulli NB set-up, we encode the first email (e_1) as $\mathbf{x}_1 = (x_{1a} = 0, x_{1b} = 1, x_{1c} = 0)$, so that each of the features are binary and represent whether a word is present or not in a given email. Carrying this out for all emails in our dataset gives us:

	x_{ia}	x_{ib}	x_{ic}
e_1	0	1	0
e_2	0	1	1
e_3	1	0	0
e_4	1	1	0
e_5	1	1	0
e_6	1	0	1
e_7	1	0	0
e_8	0	0	0

So, under this model, we are choosing to ignore the frequency of words in the email and just consider whether a word appears or not in an email. This is equivalent to modelling the class conditional distribution as

$$p(\mathbf{x}|c_k) = \prod_{j \in V} p(x_j|c_k),$$

To estimate our probabilities:



	x_{ia}	x_{ib}	x_{ic}
e_1	0	1	0
e_2	0	1	1
e_3	1	0	0
e_4	1	1	0
e_5	1	1	0
e_6	1	0	1
e_7	1	0	0
e_8	0	0	0

$$p_j^k = \frac{\text{no. docs. in class } k \text{ that contain } j}{\text{no. docs. in class } k}$$

$$p_a^+ = \frac{2}{4} \quad p_a^- = \frac{3}{4}$$
$$p_b^+ = \frac{3}{4} \quad p_b^- = \frac{1}{4}$$
$$p_c^+ = \frac{1}{4} \quad p_c^- = \frac{1}{4}$$

Now, we need to calculate our probabilities to solve

$$\hat{c} = \arg\max_{k \in \{-,+\}} p(c_k) p(\mathbf{x}^e | c_k)$$

For the positive class:

$$\begin{aligned}
p(c_+)p(\mathbf{x}^e|c_+) &= p(c_+) \prod_{j \in \{a,b,c\}} p(x_j = x_j^e | c_+) \\
&= p(c_+) \times p(x_a = 1|c_+) \times p(x_b = 1|c_+) \times p(x_c = 0|c_+) \\
&= p(c_+) \times p_a^+ \times p_b^+ \times (1 - p_c^+) \\
&= \frac{1}{2} \times \frac{2}{4} \times \frac{3}{4} \times (1 - \frac{1}{4}) \\
&= \frac{9}{64}
\end{aligned}$$

For the negative case:

$$
\begin{aligned}
p(c_-)p(\mathbf{x}^e|c_-) &= p(c_-) \prod_{j\in\{a,b,c\}} p(x_j = x_j^e|c_-) \\
&= p(c_-) \times p(x_a = 1|c_-) \times p(x_b = 1|c_-) \times p(x_c = 0|c_-) \\
&= p(c_-) \times p_a^- \times p_b^- \times (1 - p_c^-) \\
&= \frac{1}{2} \times \frac{3}{4} \times \frac{1}{4} \times (1 - \frac{1}{4}) \\
&= \frac{9}{128}
\end{aligned}
$$

Therefore, we pick $c+$ as it has a larger probability of occuring.

b) Repeat a) but with smoothing.

How does smoothing work?

$$p_j^k = \frac{\text{no. docs. in class } k \text{ that contain } j + 1}{\text{no. docs. in class } k + \text{no. possible values of } x}$$

$$\begin{aligned} p_a^+ &= \frac{2+1}{4+2} & p_a^- &= \frac{3+1}{4+2} \\ p_b^+ &= \frac{3+1}{4+2} & p_b^- &= \frac{1+1}{4+2} \\ p_c^+ &= \frac{1+1}{4+2} & p_c^- &= \frac{1+1}{4+2} \end{aligned}$$

We then use these values to again find that c_+ is the most probable class.

c) The same as a), though now using a multinomial distribution instead of a Bernoulli. For an email $e = abbdebbcc$.

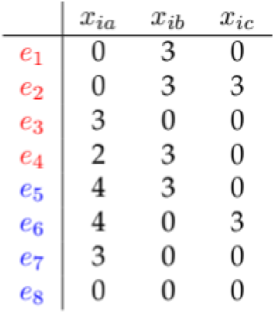
Form

The classic multinomial distribution is:

$$P(X = (x_1, \cdots, x_n)) = \frac{n!}{\prod_{i=1}^{k} x_i!} \prod_{i=1}^{k} \theta_i^{x_i}$$

Applied to a naive Bayes classifier:

$$p(\mathbf{x}|c_k) = \frac{n!}{\prod_{j\in V} x_j!} \prod_{j\in V} p(x_j|c_k)^{x_j}$$



	x_{ia}	x_{ib}	x_{ic}
e_1	0	3	0
e_2	0	3	3
e_3	3	0	0
e_4	2	3	0
e_5	4	3	0
e_6	4	0	3
e_7	3	0	0
e_8	0	0	0

$$\theta_j^k = \frac{\text{no. of times word } j \text{ appears in class } k}{\text{no. of words that appear in class } k}$$

$$\theta_a^+ = \frac{5}{17} \quad \theta_a^- = \frac{11}{17}$$
$$\theta_b^+ = \frac{9}{17} \quad \theta_b^- = \frac{3}{17}$$
$$\theta_c^+ = \frac{3}{17} \quad \theta_c^- = \frac{3}{17}$$

Our data is $x^e = (1, 4, 2)$, if we substitute our values:

$$\begin{aligned} p(c_+)p(\mathbf{x}|c_+) &= p(c_+)\frac{n!}{\prod_{j\in V} x_j!}\prod_{j\in V} p(x_j|c_k)^{x_j} \\ &= \frac{1}{2} \times \frac{7!}{1! \times 4! \times 2!}\left((\theta_a^+)^1 \times (\theta_b^+)^4 \times (\theta_c^+)^2\right) \\ &= \frac{1}{2} \times \frac{7!}{1! \times 4! \times 2!}\left(\frac{5}{17} \times \left(\frac{9}{17}\right)^4 \times \left(\frac{3}{17}\right)^2\right) \\ &= 0.0377 \end{aligned}$$

We can apply the same logic for the negative case and find that,

$$p(c_-)p(\mathbf{x}|c_-) = 0.001$$

therefore, c_+ is the most probable class to assign to this email.

d) Repeat c) but with smoothed probabilities for the multinomial.

$$\theta_j^k = \frac{\text{no. of times word } j \text{ appears in class } k + 1}{\text{no. of words that appear in class } k + |V|}$$

here, $V = \{a, b, c\}$ so the cardinality $|V| = 3$.

$$\begin{aligned}
\theta_a^+ &= \frac{5+1}{17+3} & \theta_a^- &= \frac{11+1}{17+3} \\
\theta_b^+ &= \frac{9+1}{17+3} & \theta_b^- &= \frac{3+1}{17+3} \\
\theta_c^+ &= \frac{3+1}{17+3} & \theta_c^- &= \frac{3+1}{17+3}
\end{aligned}$$

Section 5

Logistic Regression

Logistic Regression

Often called *logit* **model**. A way for us to use a linear combination $w^T x$ to predict probabilities of a binary classification problem.

For a data point (x_i, y_i) the model will predict:

$$P(y_i = 1 | x_i)$$

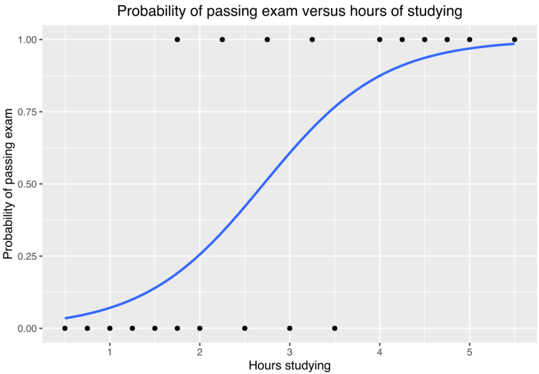
Simply, the probability that the target belongs to class 1 given the datapoint at index i.

The logistic regression is defined as the following function:

$$\sigma(w^T x_i) = \frac{1}{1 + e^{-w^T x_i}}$$

In the basic case where we only have one feature:

$$\sigma(w^T x_i) = \frac{1}{1 + e^{-w_0 - w_1 x_i}}$$



Section 6

3 (a, b, c)

3 (a, b, c)

If we define the binary prediction problem as a probability:

$$P(y = 1|x) = p(x)$$

We write the logistic regression prediction as:

$$\hat{p}(x) = \sigma(\hat{w}^T x)$$

$$\text{where } \sigma(z) = \frac{1}{1 + e^{-z}}$$

where we predict the class of an input x to be 1 if $\hat{p}(x) \geq 0.5$.

3a

What is the role of the sigmoid function here?

In a linear model, we can't simply predict probabilities or classes with the classic equation $\hat{p}(x) = \hat{w}^T x$. The sigmoid $\sigma(z)$ us model probabilities in a valid interval ($[0, 1]$).

3b

Consider the statistical view of the binary classification problem $y_i|x_i \sim \text{Bernoulli}(p_i^*)$ where $p_i^* = \sigma(x_i^T w)$ is our logistic regression model.

By definition of the Bernoulli:

$$P(y|x) = p^y(1-p)^{1-y}$$

So, we can estimate p using MLE:

$$\ln L(w) = \ln\left(\prod_{i=1}^{n} P(y_i|x_i)\right)$$

$$= \sum_{i=1}^{n} \ln P(y_i|x_i)$$

$$= \sum_{i=1}^{n} \ln\left(p_i^{y_i}(1-p_i)^{1-y_i}\right)$$

$$= \sum_{i=1}^{n} [y_i \ln p_i + (1-y_i)\ln(1-p_i)]$$

$$= \sum_{i=1}^{n} \left[y_i \ln\left(\sigma(w^T x_i)\right) + (1-y_i)\ln\left(1-\sigma(w^T x_i)\right)\right]$$

$$= \sum_{i=1}^{n} \left[y_i \ln\left(\frac{\sigma(w^T x_i)}{1-\sigma(w^T x_i)}\right) + \ln\left(1-\sigma(w^T x_i)\right)\right]$$

So, our solution is then:

$$\hat{w} = \arg\max_{w} \sum_{i=1}^{n} \left[y_i \ln \left(\frac{\sigma(w^T x_i)}{1 - \sigma(w^T x_i)} \right) + \ln\Big(1 - \sigma(w^T x_i)\Big) \right]$$

we can then solve this using optimisation methods (i.e gradient descent).

c) An alternative approach to the logistic regression problem is to view it purely from the optimisation perspective. This requires us to pick a loss function and solve for the corresponding minimizer. Write down the MSE objective for logistic regression and discuss whether you think this loss is appropriate.

The MSE objective would be:

$$\hat{w} = \arg\min_{w} \|y - \sigma(Xw)\|_2^2$$

is this appropriate?

This is not an appropriate choice as y is binary (class 0 or 1) and our prediction is real valued. This means we're comparing real class values with probabilities which doesn't make direct sense. The maximum likelihood derivation using a log-loss is the most intuitive and applies in this case as logistic regression predicts probabilities.