10601 Machine Learning
Hsueh Lin Huang
hsuehlih

Problem 1.

(a)

$$P(A\cap B) = P(A)P(B|A) = P(B)P(A|B)$$

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

(b)

$$P(B) = \sum_{i=1}^{n} P(B|A_i)P(A_i) \quad , \quad \cup_{i=1}^{n} A_i = \Omega$$

$$B = B\cap(\cup_i A_i) = \cup_i (B\cap A_i)$$

$$P(B) = P(B\cap(\cup_i A_i)) = \sum_i P(B\cap A_i)$$

Since $B\cap A_i$ are disjoint, we can write $P(B\cap A_i)$ as $P(B|A_i)P(A_i)$

(c)

$$P(A_i|B) = \frac{P(B|A_i)P(A_i)}{\Sigma_k P(B|A_k)P(A_k)}$$, $A_1 \ldots A_n$ are a partition of sample space Ω

$$\Sigma_k P(B|A_k)P(A_k) = P(B)$$

According to Baye's Rule, $P(A_i|B) = \dfrac{P(B|A_i)P(A_i)}{P(B)}$.

(d)

(1) $P(A,B,C) = P(A|B,C)P(B,C) = P(A|B,C)P(B|C)P(C)$

$\Rightarrow$ true, the definition of conditional probability.

(2) $P(A,B) = P(A|B)P(B|A)$

$\Rightarrow$ false, $P(A,B) = P(A|B)P(B)$

(3) $P(A,B,C) = P(B|A,C)P(C,A)$

$\Rightarrow$ true, the definition of conditional probability and $P(A,C) = P(C,A)$.

(4) $P(A,B,C) = P(B|A,C)P(C,A)P(C)$

$\Rightarrow$ false, the term $P(C,A)$ should be $P(A|C)$.

(5) $P(A,B) = P(A)P(B)$

$\Rightarrow$ false, only true if A and B are independent.

(e)

$E[X] = -1 \times P(A)$

$P(A) + E[X] = P(A) - P(A) = 0_{\#}$

The result will not hold if the value of X change from -1 to 1.
The result will become $2P(A)$ instead.

Problem 2

(a) $L(\hat{\theta}) = (1-\theta)^{X_1}\theta(1-\theta)^{X_2}\theta(1-\theta)^{X_3}\theta \ldots (1-\theta)^{X_n}\theta = \theta^n(1-\theta)^{\sum_{i=1}^{n}X_i}$

$\ell(\hat{\theta}) = \log L(\hat{\theta}) = n\log\theta + \left(\sum_{i=1}^{n}X_i\right)\log(1-\theta)$

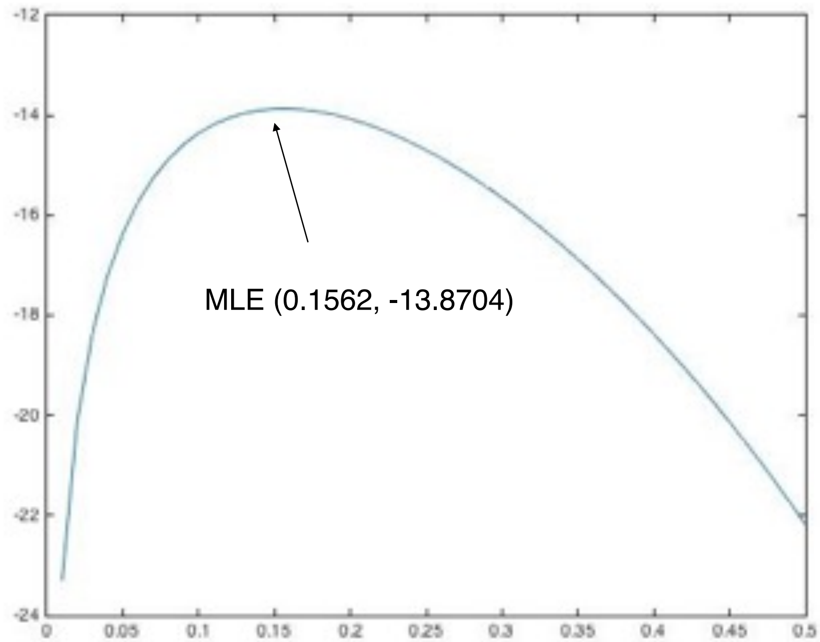
No, it doesn't depend on the order of random variables.

(b)

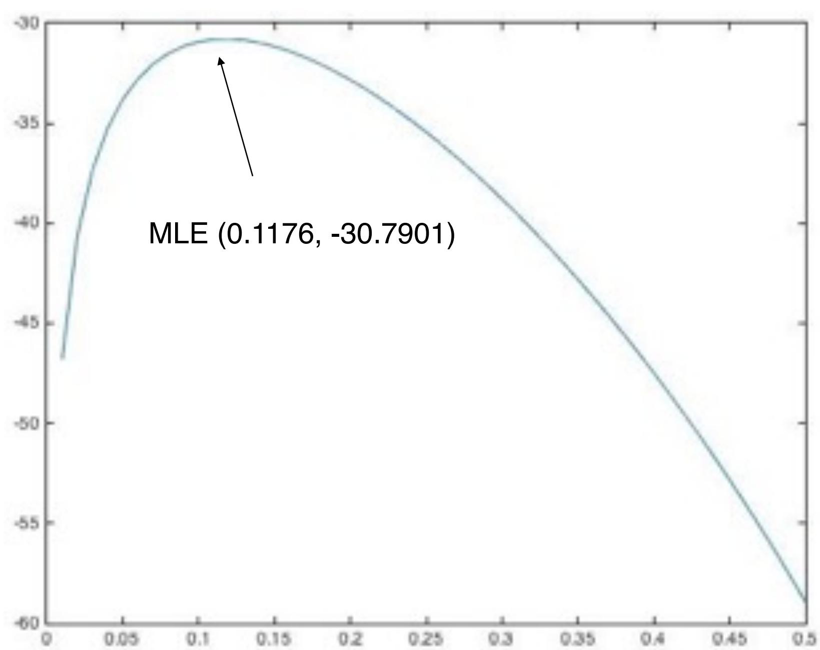
```
n=[5, 10, 15]
for i = 1:3
    log_likelihood = n(i)*log(theta) + sum(x(1:n(i)))*log(1-theta);
    figure;
    plot(theta, log_likelihood);
end
```

(c) $\dfrac{d[\log L(\hat{\theta})]}{d\hat{\theta}} = \dfrac{n}{\theta} - \dfrac{\left(\sum_{1}^{n}X_i\right)}{1-\theta} = 0 \Rightarrow \theta = \dfrac{n}{n+\sum_{1}^{n}X_i}$, it agrees with the plots.

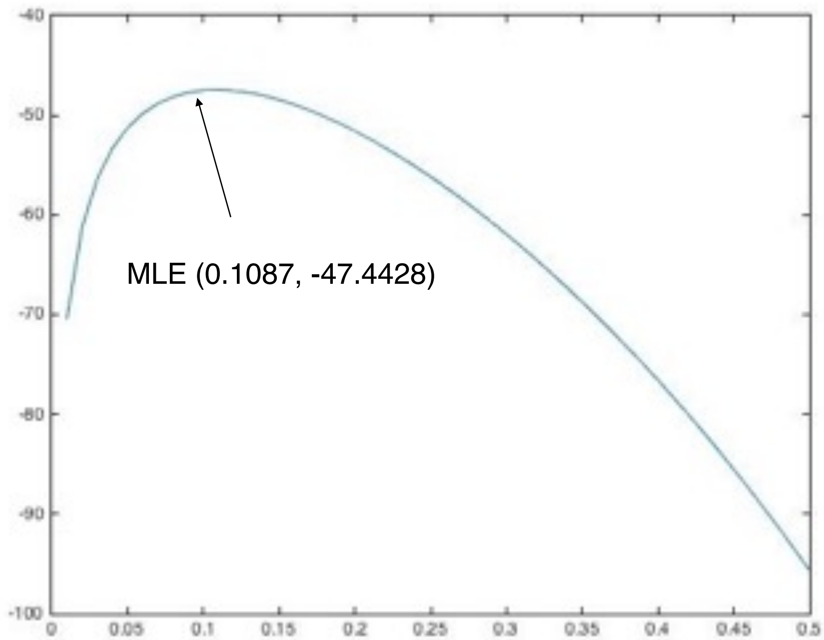
Problem 2: Maximum Likelihood Estimation
First five samples:



First ten samples:



First 15 samples:



code:

```
x = [1,0,3,5,18,14,5,7,13,9,0,17,4,24,3];
theta = 0.01:0.01:0.5;
n = [5, 10, 15];
p = zeros(1, 3);
log_likelihood = zeros(3, size(theta, 2));
for i = 1:3
    p(i) = n(i)/(n(i)+sum(x(1:(n(i)))));
    log_likelihood(i, :) = n(i)*log(theta)+sum(x(1:n(i)))*log(1-theta);
    figure;
    plot(theta, log_likelihood(i, :));
end
```

(d) Since the probability of each sample is between 0 and 1. and the $\log(P)$ is negative. Therefore, the more samples we get, the likelihood function becomes more negative.

Problem 3.

By Bayes Rule

(a)

$$= \arg\max_y \frac{P(X|Y=y)\,P(Y=y)}{P(X)} = \arg\max_y P(X|Y=y)\,P(Y=y)$$

$$= \arg\max_y \left(\prod_{w=1}^{V} P(X_w|Y=y)\right) P(Y=y) \quad \text{(Using Conditional Independence)}$$

(b)

Using Naive Bayes assumption, we need v parameters.

Not using Naive Bayes, we'll need 2^v parameters.

If v is large enough, we'll earn a big gain based on this assumption.

Problem 3: Implementing Naive Bayes

(c)

```
function [D] = NB_XGivenY(XTrain, yTrain)
% Implement your function here.
EconoRows = find(yTrain == 1);
OnionRows = find(yTrain == 2);
Economist = XTrain(EconoRows, :);
Onion = XTrain(OnionRows, :);
D = [(sum(Economist) + 0.001) / (length(EconoRows) + 0.901) ;
(sum(Onion) + 0.001) / (length(OnionRows) + 0.901)];
end
```

(d)

```
function [p] = NB_YPrior(yTrain)
% Implement your function here.
p = sum(yTrain==1)/length(yTrain);
end
```

(e)

```
function [yHat] = NB_Classify(D, p, XTest)
% Implement your function here.
yHat = zeros(size(XTest, 1), 1);
for i = 1:size(XTest, 1)
	econo_probs = XTest(i, :) .* D(1, :) + (1-XTest(i, :)) .* (1-D(1, :));
	onion_probs = XTest(i, :) .* D(2, :) + (1-XTest(i, :)) .* (1-D(2, :));

	econo_score = logProd([log(econo_probs), log(p)]);
	onion_score = logProd([log(onion_probs), log(1-p)]);

	if (econo_score > onion_score)
		yHat(i) = 1;
	else
		yHat(i) = 2;
	end
end
end
```

(f)

```
load("HW2Data.mat");
D = NB_XGivenY(XTrain, yTrain);
p = NB_YPrior(yTrain);
yHatTrain = NB_Classify(D, p, XTrain);
yHatTest = NB_Classify(D, p, XTest);
trainError = ClassificationError(yHatTrain, yTrain)
testError = ClassificationError(yHatTest, yTest)
```

Result: trainError = 0.0034, testError = 0.0276

- TrainError is smaller than testError as expected.

(g)

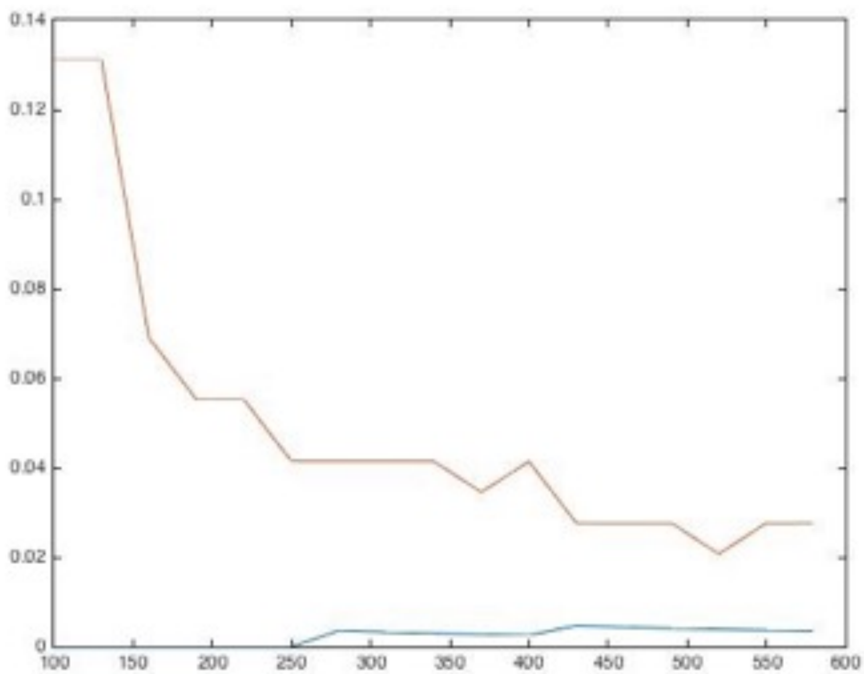
```
trainError = zeros(17, 1);
testError = zeros(17, 1);
i = 0;
for m = 100:30:580
    i = i+1;
    D = NB_XGivenY(XTrain(1:m, :), yTrain(1:m, :));
    p = NB_YPrior(yTrain(1:m));
    yHatTrain = NB_Classify(D, p, XTrain(1:m, :));
    yHatTest = NB_Classify(D, p, XTest);
    trainError(i) = ClassificationError(yHatTrain, yTrain(1:m));
    testError(i) = ClassificationError(yHatTest, yTest);
end
```

Result:

m	trainError	testError
100	0	0.1310
130	0	0.1310
160	0	0.0690
190	0	0.0552
220	0	0.0552
250	0	0.0414
280	0.0036	0.0414
310	0.0032	0.0414
340	0.0029	0.0414

370	0.0027	0.0345
400	0.0025	0.0414
430	0.0047	0.0276
460	0.0043	0.0276
490	0.0041	0.0276
520	0.0038	0.0207
550	0.0036	0.0276
580	0.0034	0.0276

Plot:



The training error of fewer samples is 0, which might imply that our prior is correct. As the number of sample grows, the training error increases. The test error decreased just as expected.

(h)

```
[~, I] = sort(D, 2, 'descend');
econo_most_likely = Vocabulary(I(1, 1:5), :)
```

```
onion_most_likely = Vocabulary(I(2, 1:5), :)
[~, I] = sort(D(1, :)./D(2, :), 2, 'descend');
econo_unique = Vocabulary(I(1:5), :)
[~, I] = sort(D(2, :)./D(1, :), 2, 'descend');
onion_unique = Vocabulary(I(1:5), :)
[~, I] = sort(D(1, :)./max(D(1, :)), 2, 'descend');
econo_max = Vocabulary(I(1:5), :)
[~, I] = sort(D(2, :)./max(D(2, :)), 2, 'descend');
onion_max = Vocabulary(I(1:5), :)
```

Result:

econoomist	onion	economist	onion	economist	onion
$P(X_w = 1 \vert Y = y)$		$\frac{P(X_w = 1 \vert Y = y)}{P(X_w = 1 \vert Y \neq y)}$.		$\frac{P(X_w = 1 \vert Y = y)}{\max_y P(X_v = 1 \vert Y = y)}$.	
'the'	'a'	'organis'	'4enlarg'	'the'	'a'
'to'	'and'	'reckon'	'5enlarg'	'to'	'and'
'of'	'the'	'favour'	'monday'	'of'	'the'
'in'	'to'	'centr'	'percent'	'in'	'to'
'a'	'of'	'labour'	'realiz'	'a'	'of'

$$\frac{P(X_w = 1 | Y = y)}{P(X_w = 1 | Y \neq y)}.$$

The second word list is the most informative about the class y, since the words like "the", "of", "a" appear in other magazines as well.