ECE 340 (Spring 2013)

PROBABILISTIC METHODS IN ENGINEERING

SOLUTIONS TO HOMEWORK #1

1.5. Consider flipping a fair coin thrice and recording the sequence of heads and tails. Here, we can choose $\Omega=\{H,T\}^3$. Define the events $E_1=\{(\omega_1, \omega_2, \omega_3)\in\Omega: \omega_1=H\}$, $E_2=\{(T,H,H),(T,T,H)\}$, $E_3=\{(T,H,T)\}$, and $E_4=\{(T,T,T)\}$.

(a) Now under the assumption of having a fair coin and that one coin flip does not influence the outcome of another coin flip, it would be reasonable to assume that all outcomes are equally likely. Hence, the probability of the event E_1 is $|E_1|/|\Omega|=4/8=1/2$. Similarly, the probability of event E_2 is $|E_2|/|\Omega|=2/8=1/4$; the probability of event E_3 is $|E_3|/|\Omega|=2/8=1/4$; and finally the probability of event E_4 is $|E_4|/|\Omega|=1/8=1/8$.

(b) Consider drawing a ball from an urn that has 4 white balls, 2 black balls, 2 green balls, and one red ball. Here, we can choose $\Omega=\{W1, W2, W3, W4, B1, B2, G1, G2, R\}$. And, define $E_1=\{W1, \ldots, W4\}$, $E_2=\{B1,B2\}$, $E_2=\{G1,G2\}$, and $E_4=\{R\}$.

(c) Assuming a 52-card deck of cards, we can consider an experiment where we draw a card, record it, then replace it and draw a second card. The outcome is the ordered recordings of the drawn cards. Now we define Ω as the set of all ordered pairs of cards (allowing repetition). Define $E_1=\{\omega\in\Omega$: only one of the cards is red$\}$, $E_2=\{\omega\in\Omega$: both cards are red$\}$, $E_3=\{\omega\in\Omega$: both cards are black$\}$, and $E_4=\{\omega\in\Omega$: the first card is black and the second card is a heart$\}$. Assume that each outcome is equally likely.

1.6.
a) $\Omega=\{(B1, B2), (B2, B1), (W, B1), (W, B2), (B1, W), (B2, W)\}$.

b) $\Omega=\{(B1,B2), (B2,B1), (W,B1), (W,B2), (B1,W), (B2,W), (W,W), (B1, B1), (B2, B2)\}$.

c) 0 for part (a) and 1/9 in part b.

d) In part (a) the outcome of the second draw will depend upon the outcome of the first draw. (Give an example.) This is not the case in part (b).

1.7. If we repeat the experiment n times and assume that from the n trials exactly k of them lead to the occurrence of event A (i.e., $f_A(n) = k/n$), then it must be true that exactly n-k trials must have resulted in the event A not occurring, i.e., they must have resulted in the event B occurring. So, $f_B(n) = (n-k)/n=1-(k/n)=1- f_A(n)$.

1.8. Suppose we repeat an experiment n times and assume that from the n trials exactly k of them lead to the occurrence of event A (and necessarily not B nor C), exactly m of them lead to the occurrence of event B (and necessarily not A nor C), and exactly j of them lead to the occurrence of event C (and necessarily not A nor B). Then, A or B or C occurred precisely k+m+j times. Hence, $f_{A \text{ or } B \text{ or } C}(n) = (k+m+j)/n= (k/n)+ (m/n)+ (j/n) = f_A(n) + f_B(n) + f_C(n)$.

2.1 Solution

a) The sample space consists of the twelve hours:
S={1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}

b) A={1, 2, 3, 4} B={2, 3, 4, 5, 6, 7, 8} D={1, 3, 5, 7, 9, 11}

c) A ∩ B ∩ D = {3}

A^C ∩ B = {5, 6, 7, 8}

A U (B ∩ D^C) = {1, 2, 3, 4} U ({2, 3, 4, 5, 6, 7, 8} ∩ { 2, 4, 6, 8, 10, 12})
= {1, 2, 3, 4, 6, 8}

(A U B) ∩ D^C = {2, 4, 6, 8}

2.2 Solution

a) Sample space:

$$S = \begin{Bmatrix} (1,1),(1,2),(1,3),(1,4),(1,5),(1,6), \\ (2,1),(2,2),(2,3),(2,4),(2,5),(2,6), \\ (3,1),(3,2),(3,3),(3,4),(3,5),(3,6), \\ (4,1),(4,2),(4,3),(4,4),(4,5),(4,6), \\ (5,1),(5,2),(5,3),(5,4),(5,5),(5,6), \\ (6,1),(6,2),(6,3),(6,4),(6,5),(6,6) \end{Bmatrix}$$

b)

$$A = \begin{Bmatrix} (1,1) & & & & & \\ (2,1) & (2,2) & & & & \\ (3,1) & (3,2) & (3,3) & & & \\ (4,1) & (4,2) & (4,3) & (4,4) & & \\ (5,1) & (5,2) & (5,3) & (5,4) & (5,5) & \\ (6,1) & (6,2) & (6,3) & (6,4) & (6,5) & (6,6) \end{Bmatrix}$$

c) B={(6,1), (6,2), (6,3), (6,4), (6,5), (6,6)}

d) B is a subset of A so when B occurs then A also occurs, thus B implies A. But A does not imply B.

e)

$$A \cap B^C = \left\{ \begin{array}{lllll} (1,1) & & & & \\ (2,1) & (2,2) & & & \\ (3,1) & (3,2) & (3,3) & & \\ (4,1) & (4,2) & (4,3) & (4,4) & \\ (5,1) & (5,2) & (5,3) & (5,4) & (5,5) \end{array} \right\}$$

The event is “number of dots in first toss is not 6 and not less than the number of dots in the second toss.”

f) C={(1,3), (2,4), (3,1), (3,5), (4,2), (4,6), (5,3), (6,4)}.

Hence, $A \cap C$ = {(3,1), (4,2), (5,3), (6,4)}

2.5 Solution

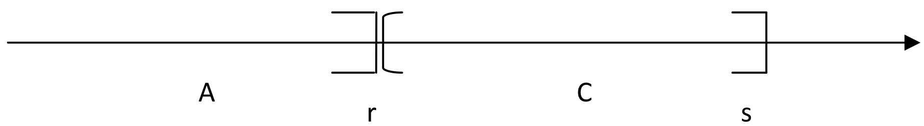
a) Each testing of a pen has two possible outcomes: “pend good” (g) or “pen bad” (b). The experiment consists of testing pens until a good pen is found. Therefore, each outcome of the experiment consists of a string of “b’s” ended by a “g.” We assume that each pen is not put back in the drawer after being tested. Thus,
S={g, bg, bbg, bbbg, bbbbg}

b) We now simply record the number of pens tested, so S={1, 2, 3, 4, 5}

c) S={gg, bgg, gbg, gbbg, bbgg, bgbg, gbbbg, bgbbg, bbgbg, bbbgg, gbbbbg, bgbbbg, bbgbbg, bbbgbg, bbbbgg}

d) S={2, 3, 4, 5, 6}

2.9 If we sketch the events A and B we see that B=A U C. We also see that the intervals corresponding to A and C have no points in common, so $A \cap C = \emptyset$.



We also see that $(r,s] = (r,\infty) \cap (-\infty,s] = (-\infty,r]^C \cap (-\infty,s]$; that is $C = A^C \cap B$. If we sketch the events A and B we see that B=A U C. More formally,

$$A \cup C = A \cup (A^C \cap B) = (A \cup A^C) \cap (A \cup B) = \Re \cap (A \cup B) = A \cup B = B.$$

Or,

$$B = B \cap \Omega = B \cap (A \cup A^C) = (B \cap A) \cup (B \cap A^C) = (B \cap A) \cup C = A \cup C.$$

To see that $A \cap C = \emptyset$, we do the following

$$A \cap C = A \cap (A^C \cap B) = (A \cap A^C) \cap B = \emptyset \cap B = \emptyset.$$

2.11.

a) the event A implies B means $A \subset B$, and the event B implies C means $B \subset C$. We need to show that $A \subset C$, which is totally clear since whatever is in A is automaticallyin B and, in turn, automatically in C.

2.12 Solution:

Note that if A U B = A, then $B \subset A$.

To prove this by contradiction: if *b* is an element of B but not an element of A (contrary to the conclusion $B \subset A$), then necessarily A is a *proper* subset of A U B (this means A U B has at least one element that is not in A), which is a contradiction to the assumption that A U B = A. So $B \subset A$. (*)

Similarly, we can prove if $A \cap B = A$ then $A \subset B$. To prove this, note that if A has a member, say *a*, that is not in B, then clearly, *a* cannot be in $A \cap B$ either, which implies that $A \cap B \neq A$, a contradiction. So $A \subset B$. (**)

Now we know that the definition of set equality tells us that A=B if and only if (also written as "iff") $A \subset B$ and $B \subset A$. Then from (*) and (**), we conclude that that A=B.

2.13. $(A \cup B) \setminus (A \cap B)$, which can also be written as $(A \setminus B) \cup (B \setminus A)$, or $(A \cap B^c) \cup (B \cap A^c)$

2.14 Solution:

a) $(A \cap B^c \cap C^c) \cup (B \cap A^c \cap C^c) \cup (C \cap A^c \cap B^c) = Ea$;

b) $(A \cap B \cap C^c) \cup (B \cap A^c \cap C) \cup (C \cap A \cap B^c) = Eb$;

c) AUBUC, or equivalently, Ea U Eb U (A∩B∩C);

d) Eb U (A∩B∩C);

e) $A^c \cap B^c \cap C^c$, or equivalently $(AUBUC)^c$.

MATLAB assignment

a) Important comments are provided in the following

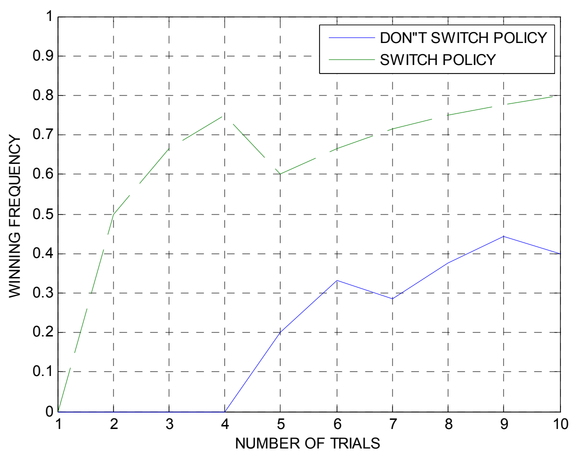
```
    % START OF PROGRAM
%   game.m
%   CREATED BY M. HAYAT ON 8/30/99 FOR ECE211
%
%   THIS PROGRAM SIMULATES THE GAME STRATEGY PROBLEM
%   DISCUSSED IN CLASS
%
%   NOTE: THE PERCENT SYMBOL % IS USED IN MATLAB TO
%   COMMENT OUT THE TEXT THAT FOLLOWS IT
clear
close(figure(1))
Max=input('What is the maximum number of trials?')
f=0;
sf2=0;

for T=1:Max;
    x=rand;%generate a number x that is uniformly distributed between 0 and 1.
    t1=1/3;
    t2=2/3;
    s=zeros(1,3);
    if (x<t1)
        s(1)=1;% 33% of the time, put it behind door 1
    elseif (x<t2)
        s(2)=1;% 33% of the time, put it behind door 2
    else
        s(3)=1;% 33% of the time, put it behind door 3
    end

    y=rand;% generate player's choice
    if (y<t1)
        i=1;% 33% of the time, guess door 1
    elseif (y<t2)
        i=2;% 33% of the time, guess door 2
    else
        i=3;% 33% of the time, guess door 3
    end

    f=f+s(i);% if the guess is correct, then count 1

    mean=f/T;% find the updated mean
    p(T)=mean;% put the mean to the vector p as a function of T = 1:Max

    %now consider the switch policy
    x=rand;
    s=zeros(1,3);
    if (x<t1)
        s(1)=1;
    elseif (x<t2)
        s(2)=1;
    else
        s(3)=1;
    end
    k=find(s);

    y=rand;
    if (y<t1)
        i=1;
    elseif (y<t2)
        i=2;
    else
        i=3;
```

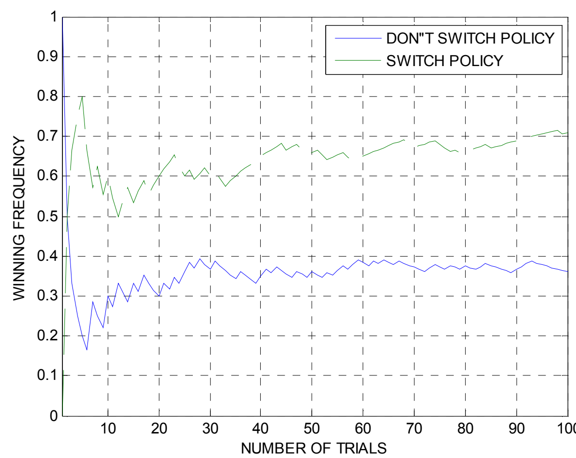
```
    end
    if (k == i)% if the pick is correct
        if (k == 1)% and if the prize is behind door 1
            j=2;   % the player switches to door 2
        else% if the prize is not behind door 1
            j = 1;% the player switches to door 1
        end
    else % if the initial pick is not correct
        j=k; % then the player switch to the correct door
    end
    sf2=sf2+s(j); % is the choice j is correct, add the counter

    smean=sf2/T;  % find the updated mean
    p2(T)=smean;
end
TT=[1:1:Max];

plot(TT,p,'-',TT,p2,'--')
grid
xlabel('NUMBER OF TRIALS')
ylabel('WINNING FREQUENCY')
legend('DON"T SWITCH POLICY','SWITCH POLICY')
axis([1 Max 0 1]);

%END OF PROGRAM
```

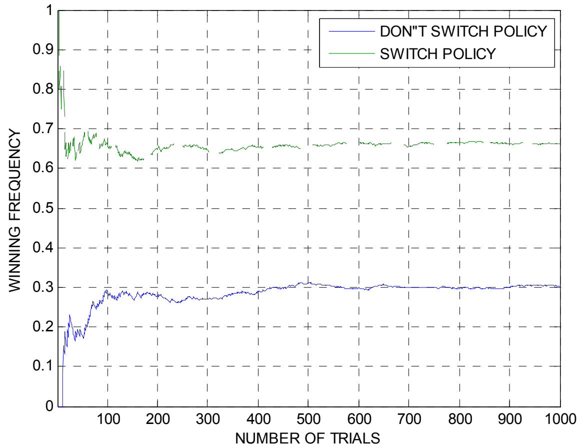
b) Estimates of the probability of winning as a function of the number of trials

	Winning frequency (%)	
Number of Trials	**Don't switch**	**Switch**
10	0.5	0.8
100	0.41	0.7
1000	0.303	0.662

As the number of trials increases the winning probability of winning for the "don't switch policy" approaches 1/3 and for the "switch policy" it approaches 2/3.







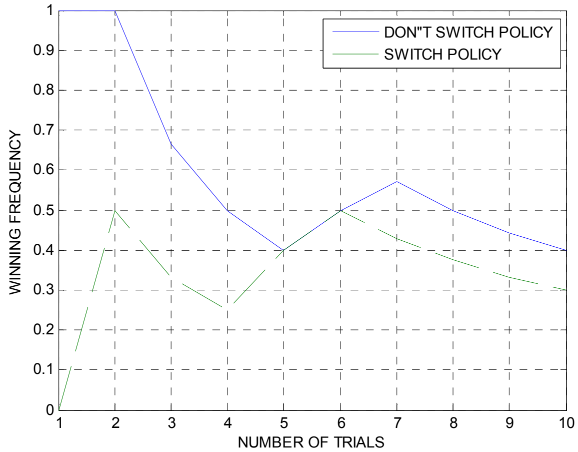
c) To simulate the fact that the dealer is biased in placing the prize behind door #1 50% of the time, the lines of code shown in **bold** need to be modified or added.

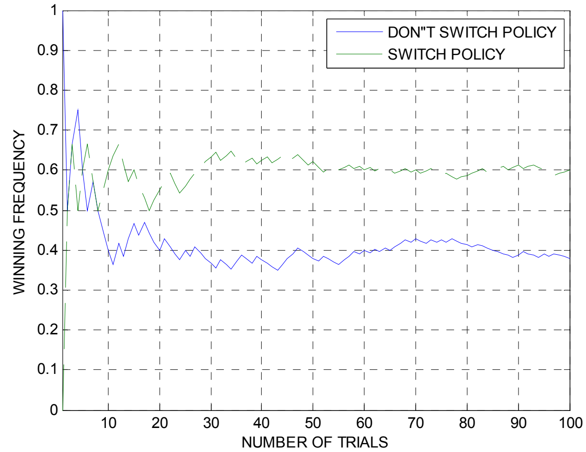
```
% START OF PROGRAM
%   game.m
%   CREATED BY M. HAYAT ON 8/30/99 FOR ECE211
%
%   THIS PROGRAM SIMULATES THE GAME STRATEGY PROBLEM
%   DISCUSSED IN CLASS
%
%   NOTE: THE PERCENT SYMBOL % IS USED IN MATLAB TO
%   COMMENT OUT THE TEXT THAT FOLLOWS IT
clear
close(figure(1))
Max=input('What is the maximum number of trials?')
f=0;
sf2=0;

for T=1:Max;
    x=rand;%generate a number x that is uniformly distributed between 0 and 1.
    t1=1/2;
    t2=1/4;
    s=zeros(1,3);
    if (x<t1)
        s(1)=1;% 33% of the time, put it behind door 1
    elseif (x<t2)
        s(2)=1;% 33% of the time, put it behind door 2
    else
        s(3)=1;% 33% of the time, put it behind door 3
    end
    t1=1/3;
    t2=2/3;
    y=rand;% generate player's choice
    if (y<t1)
        i=1;% 33% of the time, guess door 1
    elseif (y<t2)
        i=2;% 33% of the time, guess door 2
    else
        i=3;% 33% of the time, guess door 3
    end

    f=f+s(i);% if the guess is correct, then count 1

    mean=f/T;% find the updated mean
    p(T)=mean;% put the mean to the vector p as a function of T = 1:Max

    %now consider the switch policy
    t1=1/2;
    t2=1/4;
    x=rand;
    s=zeros(1,3);
    if (x<t1)
        s(1)=1;
    elseif (x<t2)
```

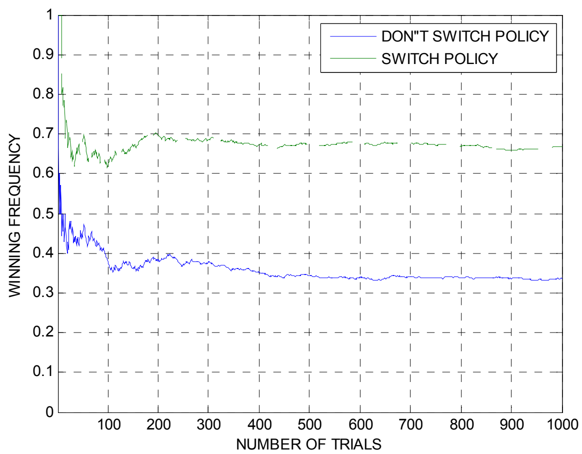
```
        s(2)=1;
    else
        s(3)=1;
    end
    k=find(s);

    t1=1/3;
    t2=2/3;
    y=rand;
    if (y<t1)
        i=1;
    elseif (y<t2)
        i=2;
    else
        i=3;
    end
    if (k == i)% if the pick is correct
        if (k == 1)% and if the prize is behind door 1
            j=2;   % the player switches to door 2
        else% if the prize is not behind door 1
            j = 1;% the player switches to door 1
        end
    else % if the initial pick is not correct
        j=k; % then the player switch to the correct door
    end
    sf2=sf2+s(j); % is the choice j is correct, add the counter

    smean=sf2/T;  % find the updated mean
    p2(T)=smean;
end
TT=[1:1:Max];

plot(TT,p,'-',TT,p2,'--')
grid
xlabel('NUMBER OF TRIALS')
ylabel('WINNING FREQUENCY')
legend('DON"T SWITCH POLICY','SWITCH POLICY')
axis([1 Max 0 1]);

%END OF PROGRAM
```

The new winning frequencies and plots are (for example):

	Winning frequency (%)	
Number of Trials	**Don't switch**	**Switch**
10	0.30	0.70
100	0.43	0.63
1000	0.336	0.669







d) From the results of part c it seems that the winning probabilities are not dependent on the distribution used to place the prize initially behind either. Even the prize is hidden in any one of the doors 100%, the player is not assumed to have this information, this leads to the same probability of winning either policy. A rigorous justification of this answer will be given once we cover the topic of conditional probabilities.