Introduction to PDEs, Fall 2024

Homework 1 Solutions

Name:______________

1. Let us revisit the random walk. Consider the discrete model of the random walk and assume for the sake of simplicity that $\Delta x = \Delta t = 1$. Then we know that the discrete equation over the whole grid is
$$u(x, t+1) = \frac{1}{2}\big(u(x-1, t) + u(x+1, t)\big), x = \pm 1, \pm 2, \pm 3, ..., t = 0, 1, 2, ... \qquad (0.1)$$
Suppose we start by placing a number of $(4 - x^2)^+$ particles at location x, where "+" means the positive part, i.e., $\max\{\cdot, 0\}$. That is, the initial data are given such that $u(x, 0) = 4 - x^2$ for $|x| \leq 2$ and $u(x, 0) \equiv 0$ for $|x| \geq 2$.

For example, consider $u(\pm 3, t)$ at time $t = 2$. It is easy to see that we need the values of $u(\pm 4, 1)$, which in turn require the values of $u(\pm 5, 0)$. Similarly, you go back to $u(\pm 6, 0)$ to evaluate $u(\pm 3, 3)$. Some of you may have recognized the mechanism/scheme as a binomial tree, while I hope this gives you some intuition that the particles moving locally (by only one step) to their neighbors at each time will eventually spread out over the whole region (we will see later in this course that the speed is infinite when $\Delta x \to 0^+$). Or you can imagine that u is the number of infected living organisms, and the disease will eventually dominate the whole space if it spreads randomly.

(i) plot $u(x, t)$ for $x = \pm 3, \pm 2, \pm 1, 0$ at time $t = 0, 1, ..., 6$, and connect the neighbouring dots with straight lines;

(ii) Now set $\Delta x = \Delta t = 0.5$ and plot $u(x, t)$ for $x = \pm 3, \pm 2.5, \pm 2, \pm 1.5, \pm 1.0$ at time $t = 0, 1, ..., 6$. I suggest that you use MATLAB or other computational software to do the calculations. Now you see, the discrete problem is, to be frank, simple but computationally tedious; do the same for $\Delta x = \Delta t = 0.01$. It gets tedious with your bare hands, but not if you use a computer program. 1) This is why it makes good sense to study the continuous case, which approximates the discrete case when Δx and Δt are small; 2) now you have just learnt the finite difference method of solving the heat equation without even realizing it;

(iii) Now let us return to the same problem with $\Delta x = \Delta t = 1$ and the same initial condition $u(x, 0) = (4 - x^2)^+$, but now over the finite interval $(-5, 5)$. We are well set for $u(x, 1)$ at all x except $x = \pm 5$, because $x = \pm 6$ is not considered in this finite interval. Therefore, we need to set special conditions for $u(\pm 5, t)$ for some time t in order to calculate $u(\pm 4, t+1)$, and so on. This condition is called a boundary condition and must be set for any PDE over a finite interval. One type is the so-called Dirichlet boundary condition for which we set $u(\pm 5, t) = 0$ (or some other constant) for any $t > 0$. Suppose $u(\pm 5, t) = 0$ for all $t \geq 0$. Plot $u(x, t)$ for $x = \pm 5, ..., \pm 1, 0$ at time $t = 0, 1, ..., 6$ and connect the neighboring points with straight lines.

(iv) do the same as in (iii) with $u(\pm 5, t) = 2$ for any $t > 0$;

(v) do the same as in (iii) with $u(-5, t) = 1$ and $u(5, t) = 3$ for any $t > 0$. Now you can see that different boundary conditions can have different effects on the solution's behavior;

(vi) do the same as in (iii) with $u(-5, t) = 1$ and $u(5, t) = 3$ for any $t > 0$, but $u_0(x) = (x^2 - 4)^+$. Now you can see that different initial conditions can have different effects on the solution's behavior;

Note: You can always, for your amusement but no need to show me, set Δx and Δt to a different size, say 10^{-3}, to see how the discrete solution approaches the continuum solution;

Solution 1. *While I plotted these graphs using MATLAB, you are free to use any programming language that you prefer. However, I would like to note each one of you should know at least one of the basic computational techniques/skills to avoid brawling with your bare hands.*

I am pleased to see that most of the students have done a pretty good job of programming. For those who are struggling, I have included my own code for your reference and study, and I wish this can help you make progress in programming by the end of this course.

```
%% Sample MATLAB code for Problem 1
%% ------- plot u(x, t) for x =  3 ,  2 ,  1 , 0 at time t = 0,
    1,..,6 ---------
dx1=1;                              % space step size (can be divided
    by 1)
dt1=1;                              % time step size  (can be divided
    by 1)
x2=-3-6/dt1*dx1:dx1:3+6/dt1*dx1;   % initial interval
m1=1+6/dt1;                         % total time steps
n1=length(x2);                      % total space steps
u2=zeros(m1,n1);                    % the matrix for calculation
ut2=zeros(7,7);                     % final result matrix
u2(1,:)=max((4-x2.^2),0);                % initial data

% iteration for the discrete equation
for i=2:m1
    for j=i:n1+1-i
        u2(i,j)=0.5*(u2(i-1,j-1)+u2(i-1,j+1));
    end
end
ut2=u2(:,[(n1+1)/2-3/dx1:1/dx1:(n1+1)/2+3/dx1]);

xt2=-3:3;
figure
plot(xt2,ut2(1,:),'k.:','markersize',10)
hold on
plot(xt2,ut2(2,:),'k.--','markersize',10)
hold on
plot(xt2,ut2(3,:),'k.-','markersize',10)
hold on
plot(xt2,ut2(4,:),'b.--','markersize',10)
hold on
plot(xt2,ut2(5,:),'b.-','markersize',10)
hold on
plot(xt2,ut2(6,:),'b.--','markersize',10)
hold on
plot(xt2,ut2(7,:),'r.-','markersize',10)
xlabel('location $x$','Fontname', 'Times New Roman','Fontsize',14);
h1=legend('$u(x,0)$','$u(x,1)$','$u(x,2)$','$u(x,3)$','$u(x,4)$','$u
    (x,5)$','$u(x,6)$');
set(h1,'Fontsize',14,'Location','best')
title('$\Delta x=\Delta t=1$','Fontsize',16)
annotation('arrow',[0.1303 0.1303],[0.11 0.98]);
annotation('arrow',[0.13 1],[0.1103 0.1103]);
set(gca,'xtick',[-3:3],'ytick',[0:9],'xticklabel',[-3:3],'yticklabel
    ',[0:9],'Fontname', 'Times New Roman')

%% ------ plot u(x, t) for x =  3 ,  2 .5,  2 ,  1 .5,  1 , 0 .5, 0
    at time t = 0,1,...,6---------
dx2=0.5;                            % space step size (can be divided
    by 0.5)
dt2=0.5;                            % time step size  (can be divided
    by 0.5)
```

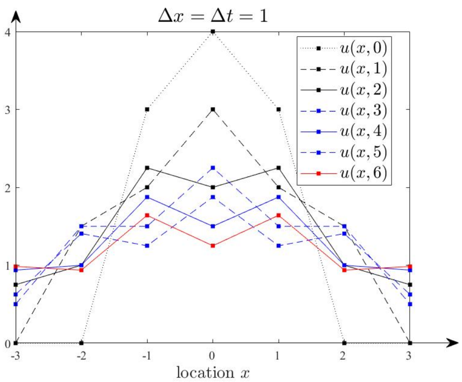
```
x2=-3-6/dt2*dx2:dx2:3+6/dt2*dx2;  % initial interval
m2=1+6/dt2;                       % total time steps
n2=length(x2);                    % total space steps
u2=zeros(m2,n2);                  % the matrix for calculation
ut2=zeros(7,13);                  % final result matrix
u2(1,:)=max((4-x2.^2),0);         % initial data

% iteration for the discrete equation
for i=2:m2
    for j=i:n2+1-i
        u2(i,j)=0.5*(u2(i-1,j-1)+u2(i-1,j+1));
    end
end
ut2=u2(:,[(n2+1)/2-3/dx2:0.5/dx2:(n2+1)/2+3/dx2]);

xt2=-3:dx2:3;
figure
plot(xt2,ut2(1,:),'k.:','markersize',10)
hold on
plot(xt2,ut2(2,:),'k.--','markersize',10)
hold on
plot(xt2,ut2(3,:),'k.-','markersize',10)
hold on
plot(xt2,ut2(4,:),'b.--','markersize',10)
hold on
plot(xt2,ut2(5,:),'b.-','markersize',10)
hold on
plot(xt2,ut2(6,:),'b.--','markersize',10)
hold on
plot(xt2,ut2(7,:),'r.-','markersize',10)
xlabel('location $x$','Fontname', 'Times New Roman','Fontsize',14);
h1=legend('$u(x,0)$','$u(x,1)$','$u(x,2)$','$u(x,3)$','$u(x,4)$','$u
   (x,5)$','$u(x,6)$');
set(h1,'Fontsize',14,'Location','best')
title('$\Delta x=\Delta t=0.5$','Fontsize',16)
annotation('arrow',[0.1303 0.1303],[0.11 0.98]);
annotation('arrow',[0.13 1],[0.1103 0.1103]);
set(gca,'xtick',[-3:3],'ytick',[0:9],'xticklabel',[-3:0.5:3],'
   yticklabel',[0:9],'Fontname', 'Times New Roman')

%% ------  the same problem with delta_x=delta_t=1 and DBC ---------
dx3=1;                            % space step size (can be divided
   by 1)
dt3=1;                            % time step size  (can be divided
   by 1)
x3=-5:dx3:5;                      % initial interval
m3=1+6/dt3;                       % total time steps
n3=length(x3);                    % total space steps
u3=zeros(m3,n3);                  % the matrix for calculation
u3(1,:)=max((4-x3.^2),0);         % initial data
u3(2:m3,[1 n3])=0;                    % boundary condition

% iteration for the discrete equation
for i=2:m3
    for j=2:n3-1
        u3(i,j)=0.5*(u3(i-1,j-1)+u3(i-1,j+1));
    end
```

```
end


figure
plot(x3,u3(1,:),'k.:','markersize',10)
hold on
plot(x3,u3(2,:),'k.--','markersize',10)
hold on
plot(x3,u3(3,:),'k.-','markersize',10)
hold on
plot(x3,u3(4,:),'b.--','markersize',10)
hold on
plot(x3,u3(5,:),'b.-','markersize',10)
hold on
plot(x3,u3(6,:),'b.--','markersize',10)
hold on
plot(x3,u3(7,:),'r.-','markersize',10)
xlabel('location $x$','Fontname', 'Times New Roman','Fontsize',14);
h1=legend('$u(x,0)$','$u(x,1)$','$u(x,2)$','$u(x,3)$','$u(x,4)$','$u
    (x,5)$','$u(x,6)$');
set(h1,'Fontsize',14,'Location','best')
title('$\Delta x=\Delta t=1\;u(\pm 5,t)=0$','Fontsize',16)
annotation('arrow',[0.1303 0.1303],[0.11 0.98]);
annotation('arrow',[0.13 1],[0.1103 0.1103]);
set(gca,'xtick',[-5:5],'ytick',[0:9],'xticklabel',[-5:5],'yticklabel
    ',[0:9],'Fontname', 'Times New Roman')

%% -------------------------------- F4
    --------------------------------------------------------
dx3=1;                              % space step size (can be divided
    by 1)
dt3=1;                              % time step size  (can be divided
    by 1)
x3=-5:dx3:5;                        % initial interval
m3=1+6/dt3;                         % total time steps
n3=length(x3);                      % total space steps
u3=zeros(m3,n3);                    % the matrix for calculation
u3(1,:)=max((4-x3.^2),0);           % initial data
u3(2:m3,[1 n3])=2;                     % boundary condition

% iteration for the discrete equation
for i=2:m3
    for j=2:n3-1
        u3(i,j)=0.5*(u3(i-1,j-1)+u3(i-1,j+1));
    end
end


figure
plot(x3,u3(1,:),'k.:','markersize',10)
hold on
plot(x3,u3(2,:),'k.--','markersize',10)
hold on
plot(x3,u3(3,:),'k.-','markersize',10)
hold on
plot(x3,u3(4,:),'b.--','markersize',10)
hold on
```

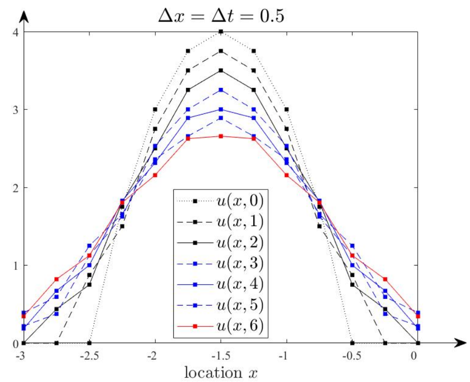
```
plot(x3,u3(5,:),'b.-','markersize',10)
hold on
plot(x3,u3(6,:),'b.--','markersize',10)
hold on
plot(x3,u3(7,:),'r.-','markersize',10)
xlabel('location $x$','Fontname', 'Times New Roman','Fontsize',14);
h1=legend('$u(x,0)$','$u(x,1)$','$u(x,2)$','$u(x,3)$','$u(x,4)$','$u
    (x,5)$','$u(x,6)$');
set(h1,'Fontsize',14,'Location','best')
title('$\Delta x=\Delta t=1\;u(\pm 5,t)=2$','Fontsize',16)
annotation('arrow',[0.1303 0.1303],[0.11 0.98]);
annotation('arrow',[0.13 1],[0.1103 0.1103]);
set(gca,'xtick',[-5:5],'ytick',[0:9],'xticklabel',[-5:5],'yticklabel
    ',[0:9],'Fontname', 'Times New Roman')

%% -------------------------------- F5
    --------------------------------------------------------
dx3=1;                              % space step size (can be divided
    by 1)
dt3=1;                              % time step size  (can be divided
    by 1)
x3=-5:dx3:5;                        % initial interval
m3=1+6/dt3;                         % total time steps
n3=length(x3);                      % total space steps
u3=zeros(m3,n3);                    % the matrix for calculation
u3(1,:)=max((4-x3.^2),0);           % initial data
u3(2:m3,1)=1;                     % boundary condition
u3(2:m3,n3)=3;                     % boundary condition

% iteration for the discrete equation
for i=2:m3
    for j=2:n3-1
        u3(i,j)=0.5*(u3(i-1,j-1)+u3(i-1,j+1));
    end
end


figure
plot(x3,u3(1,:),'k.:','markersize',10)
hold on
plot(x3,u3(2,:),'k.--','markersize',10)
hold on
plot(x3,u3(3,:),'k.-','markersize',10)
hold on
plot(x3,u3(4,:),'b.--','markersize',10)
hold on
plot(x3,u3(5,:),'b.-','markersize',10)
hold on
plot(x3,u3(6,:),'b.--','markersize',10)
hold on
plot(x3,u3(7,:),'r.-','markersize',10)
xlabel('location $x$','Fontname', 'Times New Roman','Fontsize',14);
h1=legend('$u(x,0)$','$u(x,1)$','$u(x,2)$','$u(x,3)$','$u(x,4)$','$u
    (x,5)$','$u(x,6)$');
set(h1,'Fontsize',14,'Location','best')
title('$\Delta x=\Delta t=1\;u(-5,t)=1\;u(+5,t)=3$','Fontsize',16)
annotation('arrow',[0.1303 0.1303],[0.11 0.98]);
```

```
annotation('arrow',[0.13 1],[0.1103 0.1103]);
set(gca,'xtick',[-5:5],'ytick',[0:9],'xticklabel',[-5:5],'yticklabel
    ',[0:9],'Fontname', 'Times New Roman')

%% --------------------------------- F6
    ---------------------------------------------------------
dx3=1;                                % space step size (can be divided
    by 1)
dt3=1;                                % time step size  (can be divided
    by 1)
x3=-5:dx3:5;                          % initial interval
m3=1+6/dt3;                           % total time steps
n3=length(x3);                        % total space steps
u3=zeros(m3,n3);                      % the matrix for calculation
u3(1,:)=max((-4+x3.^2),0);             % initial data
u3(2:m3,1)=1;                       % boundary condition
u3(2:m3,n3)=3;                       % boundary condition

% iteration for the discrete equation
for i=2:m3
    for j=2:n3-1
        u3(i,j)=0.5*(u3(i-1,j-1)+u3(i-1,j+1));
    end
end

figure
plot(x3,u3(1,:),'k.:','markersize',10)
hold on
plot(x3,u3(2,:),'k.--','markersize',10)
hold on
plot(x3,u3(3,:),'k.-','markersize',10)
hold on
plot(x3,u3(4,:),'b.--','markersize',10)
hold on
plot(x3,u3(5,:),'b.-','markersize',10)
hold on
plot(x3,u3(6,:),'b.--','markersize',10)
hold on
plot(x3,u3(7,:),'r.-','markersize',10)
xlabel('location $x$','Fontname', 'Times New Roman','Fontsize',14);
h1=legend('$u(x,0)$','$u(x,1)$','$u(x,2)$','$u(x,3)$','$u(x,4)$','$u
    (x,5)$','$u(x,6)$');
set(h1,'Fontsize',14,'Location','best')
title('$\Delta x=\Delta t=1\;u(-5,t)=1\;u(+5,t)=3$','Fontsize',16)
annotation('arrow',[0.1303 0.1303],[0.11 0.98]);
annotation('arrow',[0.13 1],[0.1103 0.1103]);
set(gca,'xtick',[-5:5],'ytick',[0:9],'xticklabel',[-5:5],'yticklabel
    ',[0:9],'Fontname', 'Times New Roman')
```

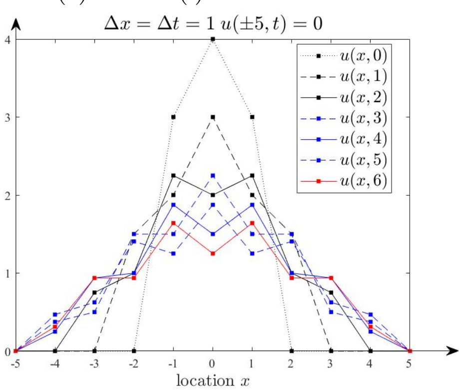
2. Now let us go back to our baby example in the 1D lattice/grid: each particle at x moves to either $x-\Delta x$ or $x+\Delta x$ with probability $\frac{1}{2}$, at each time t. As I mentioned in class, some students may have concerns that it is unfair or unrealistic to assume that every particle only moves to $x\pm\Delta x$ at the next time step, and it is possible that, for example, the particle can move to $x\pm\Delta x$ with probability $\frac{1}{4}$, and move to $x\pm 2\Delta x$ with a small probability, say $\frac{1}{8}$, and move to $x\pm 3\Delta x$ with an even smaller probability, say $\frac{1}{16}$, and move to $x\pm 4\Delta x$, $x\pm 5\Delta x$. ... and so on, with all probabilities adding up to 1. I have no objection to this possibility! However, this problem is designed to show that this case also leads to the classical heat equation, with only a variation in the diffusion rate.



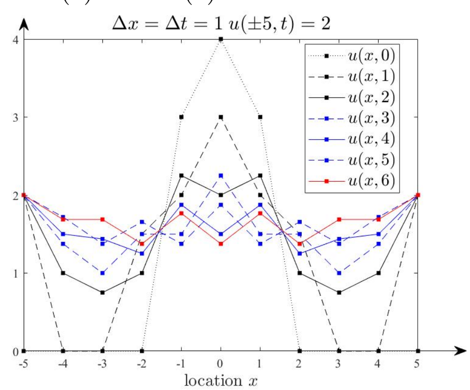
(a) Plot in (i) of Problem 1



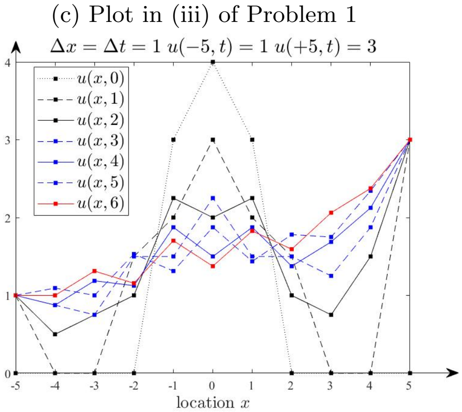
(b) Plot in (ii) of Problem 1



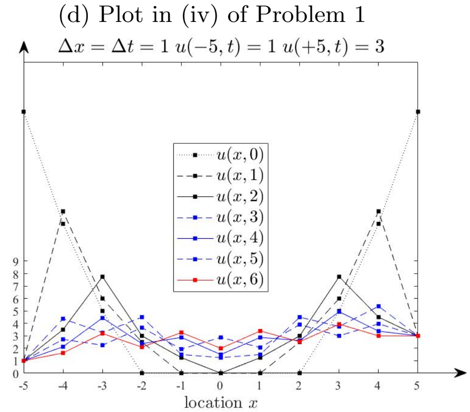
(c) Plot in (iii) of Problem 1



(d) Plot in (iv) of Problem 1



(e) Plot in (v) of Problem 1



(f) Plot in (vi) of Problem 1

Figure 1: All the plots to the variation of boundary/initial conditions

To simplify our mathematical analysis, and without losing our generality, let us assume that each particle can only move to $x \pm \Delta x$, $x \pm 2\Delta x$, with probability

$$p(x \to x \pm \Delta x, t) = \frac{\alpha}{2}, p(x \to x \pm 2\Delta x, t) = \frac{1-\alpha}{2},$$

so that $p(x \to x \pm 3\Delta x, t) = p(x \to x \pm 4\Delta x, t) = ... = 0$. Derive the PDE for $u(x,t)$ using the microscopic approach.

Solution 2. *First of all, the difference equation is*

$$u(x, t+\Delta t) = \frac{\alpha}{2}\big(u(x+\Delta x, t) + u(x-\Delta x, t)\big) + \frac{1-\alpha}{2}\big(u(x+2\Delta x, t) + u(x-2\Delta x, t)\big).$$

By expanding their Taylor series up to the order of Δx^2 *and* Δ, *we have that*

$$u_t\Delta t + O(\Delta t^2) + \Big(\frac{\alpha}{2}\Delta x^2 + 2(1-\alpha)\Delta^2\Big)u_{xx} + O(\Delta x^3),$$

therefore the PDE is $u_t = \frac{4-3\alpha}{2}Du_{xx}$ *after we send the limit.*

3. Assume that each particle can also move along the diagonal, so that $p((x,y) \to (x \pm \Delta x, y), t) = p((x,y) \to (x, y \pm \Delta y), t) = \alpha$ and $p((x,y) \to (x \pm \Delta x, y \pm \Delta y), t) = 1/4 - \alpha$, $\alpha \in (0, 1/4)$. Derive the PDE.

   Solution 3. *The argument is the same as that for the classic left-right-up-down movement. At time* $t + \Delta t$, *the population at location* (x,y) *consists of those migrated from North* $(x, y+\Delta y)$, *South* $(x, y-\Delta y)$, *West* $(x-\Delta x, y)$, *East* $(x+\Delta x, y)$, *and the four grids Northeast* $(x+\Delta x, y+\Delta y)$, *Northwest* $(x-\Delta x, y+\Delta y)$, *Southwest* $(x-\Delta x, y-\Delta y)$ *and Southeast* $(x+\Delta x, y-\Delta y)$. *In terms of mathematics, we can write, using the transition probability* P, *that*

$$\begin{aligned}
&u(x,y,t+\Delta t)\\
=&u(x,y+\Delta y,t)P\big((x,y+\Delta y) \to (x,y),t\big) + u(x,y-\Delta y,t)P\big((x,y-\Delta y) \to (x,y),t\big)\\
&+ u(x+\Delta x,y,t)P\big((x+\Delta x,y) \to (x,y),t\big) + u(x-\Delta x,y,t)P\big((x-\Delta x,y) \to (x,y),t\big)\\
&+ u(x+\Delta x,y+\Delta y,t)P\big((x+\Delta x,y+\Delta y) \to (x,y),t\big) + u(x-\Delta x,y+\Delta y,t)P\big((x-\Delta x,y+\Delta y) \to (x,y),t\big)\\
&+ u(x-\Delta x,y-\Delta y,t)P\big((x-\Delta x,y-\Delta y) \to (x,y),t\big) + u(x+\Delta x,y-\Delta y,t)P\big((x+\Delta x,y-\Delta y) \to (x,y),t\big)\\
=&\alpha\Big(u(x,y+\Delta y,t) + u(x,y-\Delta y,t) + u(x+\Delta x,y,t) + u(x-\Delta x,y,t)\Big)\\
&+ (1/4-\alpha)\Big(+u(x+\Delta x,y+\Delta y,t) + u(x-\Delta x,y+\Delta y,t) + u(x-\Delta x,y-\Delta y,t) + u(x+\Delta x,y-\Delta y,t)\Big).
\end{aligned}$$

   The rest calculations are the same as earlier, except that one needs to take care of the mixed derivatives in the following expansion(s)

$$u(x+\Delta x, y+\Delta y, t) = u(x,y,t) + u_x\Delta x + u_y\Delta y + \frac{1}{2}u_{xx}(\Delta x)^2 + u_{xy}\Delta x\Delta y + \frac{1}{2}u_{yy}(\Delta y)^2 + H.O.T.,$$

   where all the partial derivatives are evaluated at (x,y,t) *for notational simplicity. Then we can eventually have that*

$$u_t = (\frac{1}{2} - \alpha)D(u_{xx} + u_{yy})$$

   is the desired PDE, if we set $\Delta x = \Delta y$ *and* $\frac{\Delta x^2}{\Delta t} = D$.

   I would like to comment that when $\alpha = 1/4$, *this PDE reduces to the classical 2D equation that we have seen; moreover, one can also choose different scaling limits* D_i, *i.e.,* $\frac{\Delta x^2}{\Delta t} = D_1$ *and* $\frac{\Delta y^2}{\Delta t} = D_2$ *as some your classmates have done.*

4. Denote $u_n(x,t) := e^{-D(\frac{n\pi}{L})^2 t}\sin\frac{n\pi x}{L}$.

   i) show that, by formal calculation, for each $N < \infty$, the series $\sum_{n=1}^N c_n u_n(x,t)$ is a solution of the following baby heat equation

$$\frac{\partial u}{\partial t} = D\frac{\partial^2 u}{\partial x^2},$$

where c_n are constants. Note: The series is also a solution for $N = \infty$ as long as it converges, and we will discuss it later in this course;

ii) Assume that $L = \pi$ and $D = 0.05$. Use MATLAB or other software to plot $u_1(x,t)$ over $x \in (0,\pi)$ with $t = 0$, $t = 0.5$, $t = 2$ in the same coordinate. Do the same for $D = 0.1$ and $D = 1$. What are your observations and explain them intuitively;

iii) Again for $D = 1$, use MATLAB or other software to plot $1u_1(x,t) + 0.5u_2(x,t)$ over $x \in (0,\pi)$ with $t = 0$, $t = 0.5$, $t = 2$ in the same coordinate;

Solution 4. *We find that $\partial_t u_n = -D(\frac{n\pi}{L})^2 u_n$ and $\partial_{xx} u_n = -(\frac{n\pi}{L})^2 u_n$, therefore, u_n satisfies the PDE. There are several observations we have from these plots. First of all, for each D, $u_1(x,t)$ converges to zero as t increases; the profile of such convergence takes the sine function. This is sometimes called the mode of the pattern (think about blowing a balloon and the shape/pattern how the balloon grows is the mode). In the last plot, we investigate the effect of diffusion size D on the decay rate of $u_1(x,2)$. It demonstrates that a larger D results in a faster decay to zero of $u_1(x,2)$;*

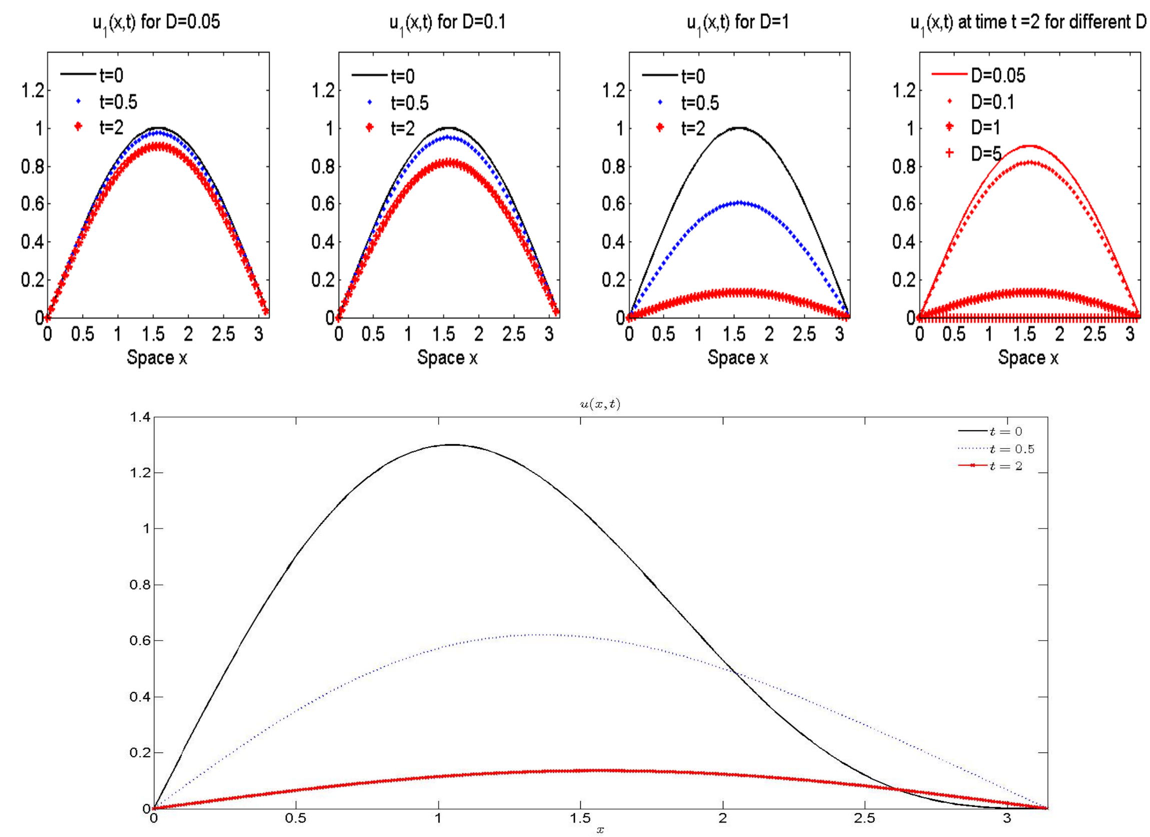


Figure 2: **Top**: The first three plots show the evolution of $u(x,t)$ at time $t = 0, 0.5$ and 2 for each fixed diffusion rate D; the last plot shows the effect of diffusion rate D on the profile of $u(x,t)$ for time $t = 2$. **Bottom:** The plots of $1u_1(x,t) + 0.5u_2(x,t)$ at the specified times.

indeed this is true, not only for $t = 2$ but also any time $t > 0$. For example, when $D = 0.01$, $u_1(x,2)$ has a profile of sine function, however when $D = 5$, $u_1(x,2)$ is almost zero. I wish this problem gives you some pictures of how the diffusion rate affects the evolution of solutions of heat equations. Generally speaking, for a single heat equation or reaction-diffusion in general, if the diffusion rate is sufficiently large, then the solution $u(x,t)$ always converges to a constant solution (depending on the initial data and boundary condition). This is called the stabilizing effect of the diffusion rate. I would like to elaborate a little bit here that, for multidimensional heat equation over a bounded domain on the other hand, usually one can not write a clean formula as $u_n(x,t)$ above (unless for some very geometries such as a rectangle or a disk), then the how fast the solutions decay *depends not only on the diffusion rate but also the shape of the domain (or the principal eigenvalue of Laplace operator in general). However, the stabilizing effect is no longer true when it comes to a system of reaction-diffusion equations (i.e., when there are more than two equations), and sometimes the system may admit a solution that converges to nonconstant solutions even when the diffusion rate(s) is(are) large. This phenomenon is called Turing's instability and such nonconstant stationary solutions are called Turing's patterns. This was first discovered by and named after the British mathematician Alan Turing, who is widely considered to be the father of theoretical computer science and artificial intelligence. Since then, mathematicians have developed reaction-diffusion models to study the emergence of patterns on animals such as stripes of zebra, dots on fish, etc., concerning the chemical reaction and diffusion of two substances, called activator and inhibitor. The study of Turing's patterns has emerged as one of the most extensively studied topics over the past few decades. Very recently, mathematical models of this fashion have been proposed and studied to investigate the well-observed hotspots in urban criminal activities. We might come back to these topics (reaction-diffusion systems) if time allows.*

```matlab
%% Sample Matlab code for problem 4
figure()
L = pi;
D = 1;
x = 0:0.01:L;
t1 = 0;
t2 = 0.5;
t3 = 2;
u1t1 = exp(-D*(1*pi/L)^2*t1)*sin(1*pi*x/L);
u2t1 = exp(-D*(2*pi/L)^2*t1)*sin(2*pi*x/L);
u1t2 = exp(-D*(1*pi/L)^2*t2)*sin(1*pi*x/L);
u2t2 = exp(-D*(2*pi/L)^2*t2)*sin(2*pi*x/L);
u1t3 = exp(-D*(1*pi/L)^2*t3)*sin(1*pi*x/L);
u2t3 = exp(-D*(2*pi/L)^2*t3)*sin(2*pi*x/L);
ut1 = 1*u1t1 +0.5*u2t1;
ut2 = 1*u1t2 +0.5*u2t2;
ut3 = 1*u1t3 +0.5*u2t3;
plot(x,ut1,'k-',x,ut2,'b:',x,ut3,'r-x','LineWidth',2)
legend({'$t=0$','$t=0.5$','$t=2$'},'Location','NorthEast','FontSize'
    ,16)
legend('boxoff')
title('$u(x,t)$','FontSize',16)
xlabel('$x$','FontSize',16)
xlim([0 L])
set(gca,'fontsize',16);
```

5. Go to review the following topics: gradient; directional derivative; multivariate integral; surface integral; divergence theorem. No need to turn in your review.

Solution 5. *I assume that you have done so, and it is important that you have done so.*