Sarah Telford a1810750

MATHS 2107 Statistics and Numerical Methods
Assignment 6

1.

a) Submitted to MATLAB grader

```
% a1810750 - Sarah Telford
% Modification Date: 28/10/21 function
[y, coeffs] = fitRational(t, tj, yj)
    % making vectors
    tj = tj(:) ;    % tj is vector containing data t_j for j=1,2,...,n  yj
    = yj(:) ;    % yj is vector containing data y_j for j=1,2,...,n
    % creating matrix A
     a = [ tj , tj.*-(yj) ] ;
     % creating matrix b
     b = [ yj ] ;
    % determining coefficients - creating a column vector of coefficients coeffs =
    a \ b ;
    % constructing vector y - containing values of y evaluated at each point of t y
    = (coeffs(1) .* t) ./ (1 + coeffs(2) .* t) ;
end
```

b)

```
% a1810750 - Sarah Telford
% Modification Date: 28/10/21

% Description: using fitRational function and data in fitRational.mat to
% compare the constructed fitted function to the origional data by %
plotting the graph.

% Read in data from fitRational.mat, getting vectors tj and yj load
fitRational.mat

% find y and coeffs using fitRational function
[y, coeffs] = fitRational(tj, tj, yj) ;

% print values of coefficient alpha
fprintf('value of Aplha: %d \n', coeffs(1)) ;
% print values of coefficient beta
fprintf('value of Beta: %d \n', coeffs(2)) ; %
print values of final equation
fprintf('the equation y(t) is: ( %d * t) / ( 1 + %d * t) \n\n', coeffs(1), coeffs(2)) ;

% plot fitted function figure
plot(tj, yj, 'o', tj, y, '-') ;
xlabel('t') ; ylabel('y')
;
legend('data', 'fitted function','location', 'southeast') ; title(sprintf('\\alpha
= %0.3f, \\beta = %0.3f',coeffs));
```

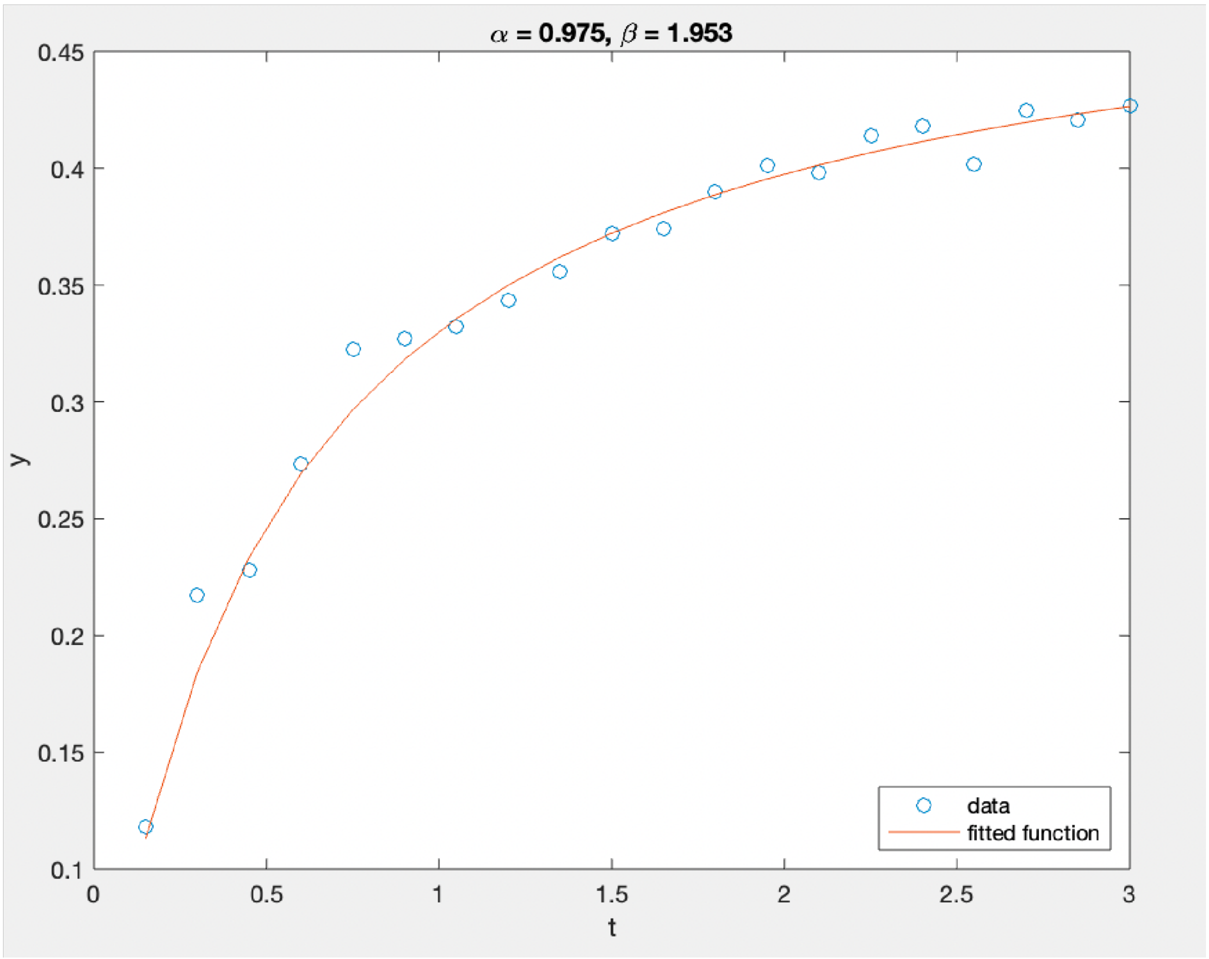

Figure 1: plot of a*t/(1 +* b*t) to data from fitRational.mat.*

2.

a) Submitted to MATLAB grader

```
% a1810750 - Sarah Telford %
Modification Date:  28/10/21
function [x, u] = solveHeatDirect(n, a, b, alpha, func)
    % find value of h
    h = 1/(n-1); %
    make vector x x
    = ones(n, 1); for
    i = 1:n
        x(i) = (i-1)*h;
    end
   % create sparse matrix A
    D = sparse(1:n-2,1:n-2,(-2-alpha*h^2)*ones(1,n-2),n-2,n-2);
    E  = sparse(2:n-2,1:n-3,ones(1,n-3),n-2,n-2);
    A = E+D+E';
```

```
        % making vector vecb
         vecb = zeros(n-2,1); vecb(1)
         = -h^2*func(x(2))-a; for k =
         3:n-1
            vecb(k-1) = -h^2*func(x(k)); end
         vecb(n-2) = -h^2*func(x(n-1))-b ;
        % finding vecor un un =
        A\vecb ; % constructing
        vector u u = zeros(n,1) ;
        for j = 2:n-1
            u(j) = un(j-1) ;
         end
        % setting boundary conditions
         u(1) = a ;
u(n) = b ;      end
```

b)

```
% a1810750 - Sarah Telford
% Modification Date: 28/10/21

% Description: using solveHeatDirect to solve  u" - u = -x*(1 - x), u(0) = 2, u(1), 3.

% Clear Workspace
 close all
 clear all

 % Set up functions  f
 = @(x) x.*(x - 1);
 u = @(x) sinh(x)/sinh(1) + 2 - x + x.^2;

% Use functions to get x and u using solveHeatDirect function
 [x1, u1] = solveHeatDirect(21, 2, 3, 1, f);
 [x2, u2] = solveHeatDirect(41, 2, 3, 1, f);

 % Create subplots x =
 linspace(0, 1, 100);
 subplot(2, 1, 1)
 plot(x1, u1, 'o', x2, u2, 'x', x, u(x), '-')
 xlabel('x') ylabel('u')
legend('n = 21', 'n=41', 'Exact', 'location', 'southeast')

 subplot(2, 1, 2)
 plot(x1, abs(u1 - u(x1)), 'o', x2, abs(u2 - u(x2)), 'x')
 xlabel('x') ylabel('abs(error)')
legend('n = 21', 'n=41', 'location', 'northwest')
```

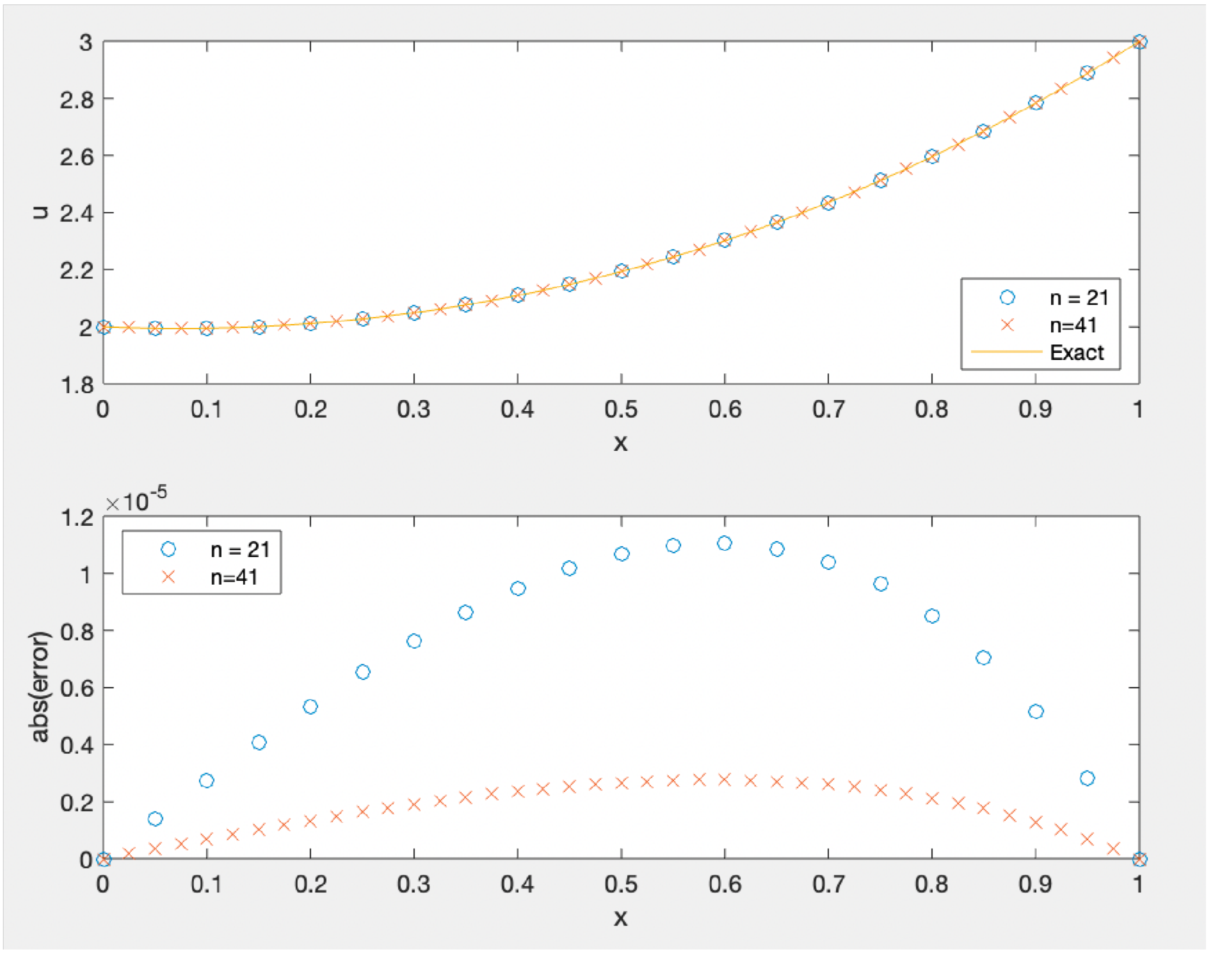

Figure 2: Numerical solutions (top) and corresponding errors (bottom)

c) **Is the error observed in part (b) consistent with an error that is $O(h^2)$. Explain your answer.**
From the graphs we can see that the grid space halves going from 21 to 41. If we are assuming that the error is $O(h^2)$, then halving the grid spacing will cause the error to decrease by a factor of 4. We can see that the peak error decreases by a factor of 3.9 which is approximately equal to the expected decreasing value of $O(h^2)$, 4 .

d) **Another way to solve the linear system (6) is to use Jacobi iteration. For what values of α would Jacobi iteration be guaranteed to converge to the solution? Explain your answer.**
We know that a Jacobi iteration converges for diagonally dominant systems.
Therefore, the linear system is diagonally dominant if values of a are:

$$| -2 - ah^2|$$
$$= 2 + ah^2 > |1| + |1|$$
$$= 2 \Rightarrow a > 0$$