**6.1** Given the following data:

| $x$ | 0.398 | 0.399 | 0.400 | 0.401 | 0.402 |
|---|---|---|---|---|---|
| $f(x)$ | 0.0630 | 0.06375 | 0.0649 | 0.0654 | 0.0658 |

find the first derivative $f'(x)$ at the point $x = 0.399$.
($a$) Use the three-point forward difference formula.
($b$) Use the two-point central difference formula.

**Solution**

($a$) The three point forward difference formula is given by (see Table 6-1):

$$f'(x_i) = \frac{-3f(x_i) + 4f(x_{i+1}) - f(x_{i+2})}{2h}$$

Setting $x_i = 0.399$ implies that $x_{i+1} = 0.4$ and $x_{i+2} = 0.401$. Correspondingly, $f(x_i) = 0.06375$, $f(x_{i+1}) = 0.0649$, and $f(x_{i+2}) = 0.0654$. In this problem, $h = 0.001$. Substituting into the above formula yields:

$$f'(0.399) = \frac{-3(0.06375) + 4(0.0649) - (0.0654)}{2(0.001)} = 1.475$$

($b$) The two-point central difference formula is given by (see Table 6-1):

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_{i-1})}{2h}$$

Again, setting $x_i = 0.399$ implies that $x_{i-1} = 0.398$ and $x_{i+1} = 0.400$. Correspondingly, $f(x_{i-1}) = 0.063$, and $f(x_{i+1}) = 0.0649$. Substituting into the above formula yields:

$$f'(0.399) = \frac{0.0649 - 0.063}{2(0.001)} = 0.95$$

**6.2** The following data shows the population of Nepal in selected years between 1980 and 2005.

| Year | 1980 | 1985 | 1990 | 1995 | 2000 | 2005 |
|---|---|---|---|---|---|---|
| Population (millions) | 15 | 17 | 19.3 | 22 | 24.5 | 27.1 |

Calculate the rate of growth of the population in millions per year for 2005.
(*a*) Use two-point backward difference formula.
(*b*) Use three-point backward difference formula.
(*c*) Using the slope in 2005 from part (*b*), apply the two-point central difference formula to extrapolate and predict the population in the year 2010.

**Solution**

(*a*) From Table 6-1, the two-point backward difference formula is:

$$f'(x_i) = \frac{f(x_i) - f(x_{i-1})}{h}$$

Here, $x_i = 2005$, $h = 5$, $f(x_i) = 27.1$ (million), and $f(x_{i-1}) = 24.5$ (million). Substituting these values into the above formula yields:

$$f'(2005) = \frac{27.1 - 24.5}{5} = 0.52\frac{million}{year}$$

Thus, the rate of growth of the population in 2005 is 0.52 million/year or 520,000 per year.

(*b*) From Table 6-1, three-point backward difference formula is:

$$f'(x_i) = \frac{f(x_{i-2}) - 4f(x_{i-1}) + 3f(x_i)}{2h}$$

Substituting $x_i = 2005$, $h = 5$, $f(x_i) = 27.1$ (million), $f(x_{i-1}) = 24.5$ (million), and $f(x_{i-2}) = 22$ (million) into the above expression yields:

$$f'(2005) = \frac{22 - 4(24.5) + 3(27.1)}{2(5)} = 0.53\frac{million}{year}$$

Thus, the rate of growth of the population in 2005 is 0.53 million/year or 530,000 per year. Note how this answer differs from that of part (*a*).

(*c*) The central difference formula is:

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_{i-1})}{2h}$$

Substituting $x_i = 2005$, $h = 5$, $f'(x_i) = 0.53\frac{million}{year}$ from part (*b*), and $f(x_{i-1}) = 24.5$ (million) into the above equation, and solving for $f(x_{i+1})$ yields:

$$f(x_{i+1}) = f(2010) = 2hf'(x_i) + f(x_{i-1}) = 2(5)(0.53) + 24.5 = 29.8 \text{ million}$$

**6.3** The following data is given for the stopping distance of a car versus the speed at which it begins braking:

| Speed (mph) | 20 | 30 | 40 | 50 | 60 |
|---|---|---|---|---|---|
| Stopping distance (ft) | 18 | 32 | 78 | 112 | 154 |

(*a*) Calculate the rate of change of the stopping distance at a speed of 60 mph using (*i*) the two-point backward difference formula, and (*ii*) the three-point backward difference formula.
(*b*) Calculate an estimate for the stopping distance at 70 mph by using the results from part (*a*) for the slope and the two-point central difference formula applied at the speed of 60 mph.

**Solution**

(*a*) (i) The two-point backward difference formula for the rate of change of stopping distance with respect to speed is given by (see Table 6-1):

$$x'(v_i) = \frac{x(v_i) - x(v_{i-1})}{h}$$

For $v_i = 60\text{ mph}$, $h = 10\text{ mph}$, $v_{i-1} = 50\text{ mph}$, $x(v_i) = 154\text{ ft}$, and $x(v_{i-1}) = 112\text{ ft}$, the change in stopping distance with respect to speed, at 60 mph is:

$$x'(60\text{ mph}) = \frac{154 - 112}{10} = 4.2 \quad \text{ft/mph}$$

(ii) With the three-point backward difference formula, the change in stopping distance with respect to speed, at 60 mph is:

$$x'(60\text{ mph}) = \frac{78 - 4(112) + 3(154)}{2(10)} = 4.6\text{ ft/mph}$$

which is close but not equal to the answer obtained in part (i).

(*b*) Using the slopes at 60 mph from part (*a*), the two-point central difference formula can be applied to extrapolate the data to find the stopping distance at 70 mph:

$$x(v_{i+1}) = 2hx'(v_i) + x(v_{i-1})$$

For the slope of $4.2\text{ ft/mph}$ obtained in part (*a*) (i), the extrapolated value is $x(70\text{ mph}) = 2(10)(4.2) + 112 = 196\text{ ft}$. For the slope of $4.5$ obtained in part (*a*) (ii), the extrapolated value is $x(70\text{ mph}) = 2(10)(4.6) + 112 = 204\text{ ft}$.

**6.4** Given three ***unequally*** spaced points $(x_i, y_i)$, $(x_{i+1}, y_{i+1})$, and $(x_{i+2}, y_{i+2})$, use Taylor series expansion to develop a finite difference formula to evaluate the first derivative $dy/dx$ at the point $x = x_i$. Verify that when the spacing between these points is equal, the three-point forward difference formula is obtained. The answer should involve $y_i$, $y_{i+1}$, and $y_{i+2}$.

**Solution**

First write down the Taylor series expansions for $y_{i+1}$ and $y_{i+2}$ assuming the value $y_i$ is known:

$$y_{i+1} = y_i + (x_{i+1} - x_i)\left.\frac{dy}{dx}\right|_{x = x_i} + \frac{(x_{i+1} - x_i)^2}{2}\left.\frac{d^2y}{dx^2}\right|_{x = x_i} + \frac{(x_{i+1} - x_i)^3}{6}\left.\frac{d^3y}{dx^3}\right|_{x = \xi_i}$$

$$y_{i+2} = y_i + (x_{i+2} - x_i)\left.\frac{dy}{dx}\right|_{x = x_i} + \frac{(x_{i+2} - x_i)^2}{2}\left.\frac{d^2y}{dx^2}\right|_{x = x_i} + \frac{(x_{i+2} - x_i)^3}{6}\left.\frac{d^3y}{dx^3}\right|_{x = \eta_i}$$

The next step is to combine the above two equations in such a way that we retain the first derivative (which we want to solve for) and lose the second derivative in order to obtain a smaller truncation error. This is done by multiplying the first equation by $(x_{i+2} - x_i)^2$ and the second equation by $(x_{i+1} - x_i)^2$ and subtracting the result:

$$(x_{i+2} - x_i)^2 y_{i+1} = (x_{i+2} - x_i)^2 y_i + (x_{i+2} - x_i)^2 (x_{i+1} - x_i)\left.\frac{dy}{dx}\right|_{x = x_i} + \frac{(x_{i+2} - x_i)^2 (x_{i+1} - x_i)^2}{2}\left.\frac{d^2y}{dx^2}\right|_{x = x_i} + \frac{(x_{i+2} - x_i)^2 (x_{i+1} - x_i)^3}{6}\left.\frac{d^3y}{dx^3}\right|_{x = \xi_i}$$

$$(x_{i+1} - x_i)^2 y_{i+2} = (x_{i+1} - x_i)^2 y_i + (x_{i+1} - x_i)^2 (x_{i+2} - x_i)\left.\frac{dy}{dx}\right|_{x = x_i} + \frac{(x_{i+1} - x_i)^2 (x_{i+2} - x_i)^2}{2}\left.\frac{d^2y}{dx^2}\right|_{x = x_i} + \frac{(x_{i+1} - x_i)^2 (x_{i+2} - x_i)^3}{6}\left.\frac{d^3y}{dx^3}\right|_{x = \eta_i}$$

Note that the third term on the right hand sides of these equations is identical. Subtracting,

$$(x_{i+1} - x_i)^2 y_{i+2} - (x_{i+2} - x_i)^2 y_{i+1} = (x_{i+1} - x_i)^2 y_i - (x_{i+2} - x_i)^2 y_i + (x_{i+1} - x_i)^2 (x_{i+2} - x_i)\left.\frac{dy}{dx}\right|_{x = x_i}$$
$$- (x_{i+2} - x_i)^2 (x_{i+1} - x_i)\left.\frac{dy}{dx}\right|_{x = x_i} + \frac{(x_{i+1} - x_i)^2 (x_{i+2} - x_i)^3}{6}\left.\frac{d^3y}{dx^3}\right|_{x = \eta_i} - \frac{(x_{i+2} - x_i)^2 (x_{i+1} - x_i)^3}{6}\left.\frac{d^3y}{dx^3}\right|_{x = \xi_i}$$

Combining the last two terms which constitute the truncation error, and solving for the first derivative yields:

$$\left.\frac{dy}{dx}\right|_{x = x_i} = \frac{(x_{i+1} - x_i)^2 y_{i+2} - (x_{i+2} - x_i)^2 y_{i+1} + [(x_{i+2} - x_i)^2 - (x_{i+1} - x_i)^2] y_i}{(x_{i+1} - x_i)^2 (x_{i+2} - x_i) - (x_{i+2} - x_i)^2 (x_{i+1} - x_i)} + TE$$

where TE is the truncation error. When the spacing between the points is equal, i.e. $x_{i+1} - x_i = x_{i+2} - x_{i+1} = h$ and $x_{i+2} - x_i = 2h$, the above expression reduces to:

$$\left.\frac{dy}{dx}\right|_{x = x_i} = \frac{h^2 y_{i+2} - 4h^2 y_{i+1} + [4h^2 - h^2] y_i}{2h^3 - 4h^3} + O(h^2)$$

which simplifies to:

$$\left.\frac{dy}{dx}\right|_{x = x_i} = \frac{-y_{i+2} + 4y_{i+1} - 3y_i}{2h} + O(h^2)$$

which is identical to the three-point forward difference formula given in Table 6-3.

**6.5** Using a four-term Taylor series expansion, derive a four-point backward difference formula for evaluating the first derivative of a function given by a set of unequally spaced points. The formula should give the derivative at point $x = x_i$, in terms of $x_i$, $x_{i-1}$, $x_{i-2}$, $x_{i-3}$, $f(x_i)$, $f(x_{i-1})$, $f(x_{i-2})$, and $f(x_{i-3})$.

## Solution

A second-order accurate four-point backward difference formulae is obtained by using Taylor series expansion and eliminating the second-derivative terms. The procedure is analogous for both. First express $f(x_{i-1})$, $f(x_{i-2})$, and $f(x_{i-3})$ in Taylor series expansions about $x = x_i$:

$$f(x_{i-1}) = f(x_i) - (x_i - x_{i-1})\left.\frac{df}{dx}\right|_{x=x_i} + \frac{(x_i - x_{i-1})^2}{2}\left.\frac{d^2f}{dx^2}\right|_{x=x_i} - \frac{(x_i - x_{i-1})^3}{6}\left.\frac{d^3f}{dx^3}\right|_{x=x_i} + \frac{(x_i - x_{i-1})^4}{24}\left.\frac{d^4f}{dx^4}\right|_{x=\xi_i} \quad \text{(A)}$$

$$f(x_{i-2}) = f(x_i) - (x_i - x_{i-2})\left.\frac{df}{dx}\right|_{x=x_i} + \frac{(x_i - x_{i-2})^2}{2}\left.\frac{d^2f}{dx^2}\right|_{x=x_i} - \frac{(x_i - x_{i-2})^3}{6}\left.\frac{d^3f}{dx^3}\right|_{x=x_i} + \frac{(x_i - x_{i-2})^4}{24}\left.\frac{d^4f}{dx^4}\right|_{x=\eta_i} \quad \text{(B)}$$

$$f(x_{i-3}) = f(x_i) - (x_i - x_{i-3})\left.\frac{df}{dx}\right|_{x=x_i} + \frac{(x_i - x_{i-3})^2}{2}\left.\frac{d^2f}{dx^2}\right|_{x=x_i} - \frac{(x_i - x_{i-3})^3}{6}\left.\frac{d^3f}{dx^3}\right|_{x=x_i} + \frac{(x_i - x_{i-3})^4}{24}\left.\frac{d^4f}{dx^4}\right|_{x=\zeta_i} \quad \text{(C)}$$

The next step is to combine the above three equations in such a way that we retain the first derivative (which we want to solve for) and lose the second derivative and perhaps the third derivative in order to obtain a smaller truncation error.

First, let us derive the lower order formula. By inspection, we can see that the second derivative is eliminated if the first equation (A) is multiplied by $(x_i - x_{i-2})^2(x_i - x_{i-3})^2$, the second equation (B) by $(x_i - x_{i-1})^2(x_i - x_{i-3})^2$, and the third equation (C) by $2(x_i - x_{i-1})^2(x_i - x_{i-2})^2$ and adding the first two equations and subtracting the third from the result:

$$(x_i - x_{i-2})^2(x_i - x_{i-3})^2 f(x_{i-1}) = (x_i - x_{i-2})^2(x_i - x_{i-3})^2 f(x_i) - (x_i - x_{i-2})^2(x_i - x_{i-3})^2(x_i - x_{i-1})\left.\frac{df}{dx}\right|_{x=x_i}$$

$$+ \frac{(x_i - x_{i-2})^2(x_i - x_{i-3})^2(x_i - x_{i-1})^2}{2}\left.\frac{d^2f}{dx^2}\right|_{x=x_i} - \frac{(x_i - x_{i-2})^2(x_i - x_{i-3})^2(x_i - x_{i-1})^3}{6}\left.\frac{d^3f}{dx^3}\right|_{x=x_i}$$

$$+ \frac{(x_i - x_{i-2})^2(x_i - x_{i-3})^2(x_i - x_{i-1})^4}{24}\left.\frac{d^4f}{dx^4}\right|_{x=\xi_i} \quad \text{(A')}$$

$$(x_i - x_{i-1})^2(x_i - x_{i-3})^2 f(x_{i-2}) = (x_i - x_{i-1})^2(x_i - x_{i-3})^2 f(x_i) - (x_i - x_{i-1})^2(x_i - x_{i-3})^2(x_i - x_{i-2})\left.\frac{df}{dx}\right|_{x=x_i}$$

$$+\frac{(x_i-x_{i-1})^2(x_i-x_{i-3})^2(x_i-x_{i-2})^2}{2}\left.\frac{d^2f}{dx^2}\right|_{x=x_i}-\frac{(x_i-x_{i-1})^2(x_i-x_{i-3})^2(x_i-x_{i-2})^3}{6}\left.\frac{d^3f}{dx^3}\right|_{x=x_i}$$

$$+\frac{(x_i-x_{i-1})^2(x_i-x_{i-3})^2(x_i-x_{i-2})^4}{24}\left.\frac{d^4f}{dx^4}\right|_{x=\eta_i} \qquad \text{(B')}$$

$$2(x_i-x_{i-1})^2(x_i-x_{i-2})^2f(x_{i-3}) = 2(x_i-x_{i-1})^2(x_i-x_{i-2})^2f(x_i)-2(x_i-x_{i-1})^2(x_i-x_{i-2})^2(x_i-x_{i-3})\left.\frac{df}{dx}\right|_{x=x_i}$$

$$+(x_i-x_{i-1})^2(x_i-x_{i-2})^2(x_i-x_{i-3})^2\left.\frac{d^2f}{dx^2}\right|_{x=x_i}-\frac{(x_i-x_{i-1})^2(x_i-x_{i-2})^2(x_i-x_{i-3})^3}{3}\left.\frac{d^3f}{dx^3}\right|_{x=x_i}$$

$$+\frac{(x_i-x_{i-1})^2(x_i-x_{i-2})^2(x_i-x_{i-3})^4}{12}\left.\frac{d^4f}{dx^4}\right|_{x=\zeta_i} \qquad \text{(C')}$$

Combining these equations (A')+(B')-(C') and lumping the truncation error terms together as TE, we have:

$$(x_i-x_{i-2})^2(x_i-x_{i-3})^2f(x_{i-1})+(x_i-x_{i-1})^2(x_i-x_{i-3})^2f(x_{i-2})-2(x_i-x_{i-1})^2(x_i-x_{i-2})^2f(x_{i-3})$$

$$= (x_i-x_{i-2})^2(x_i-x_{i-3})^2f(x_i)+(x_i-x_{i-1})^2(x_i-x_{i-3})^2f(x_i)-2(x_i-x_{i-1})^2(x_i-x_{i-2})^2f(x_i)$$

$$+2(x_i-x_{i-1})^2(x_i-x_{i-2})^2(x_i-x_{i-3})\left.\frac{df}{dx}\right|_{x=x_i}-(x_i-x_{i-2})^2(x_i-x_{i-3})^2(x_i-x_{i-1})\left.\frac{df}{dx}\right|_{x=x_i}$$

$$-(x_i-x_{i-1})^2(x_i-x_{i-3})^2(x_i-x_{i-2})\left.\frac{df}{dx}\right|_{x=x_i}+TE$$

Solving for $\left.\frac{df}{dx}\right|_{x=x_i}$ yields:

$$\left.\frac{df}{dx}\right|_{x=x_i}$$

$$=\frac{(x_i-x_{i-2})^2(x_i-x_{i-3})^2f(x_{i-1})+(x_i-x_{i-1})^2(x_i-x_{i-3})^2f(x_{i-2})-2(x_i-x_{i-1})^2(x_i-x_{i-2})^2f(x_{i-3})}{2(x_i-x_{i-1})^2(x_i-x_{i-2})^2(x_i-x_{i-3})-(x_i-x_{i-2})^2(x_i-x_{i-3})^2(x_i-x_{i-1})-(x_i-x_{i-1})^2(x_i-x_{i-3})^2(x_i-x_{i-2})}$$

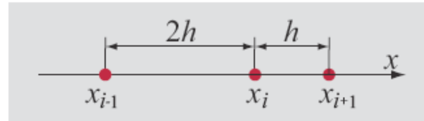
$$+\frac{[2(x_i-x_{i-1})^2(x_i-x_{i-2})^2-(x_i-x_{i-1})^2(x_i-x_{i-3})^2-(x_i-x_{i-2})^2(x_i-x_{i-3})^2]f(x_i)}{2(x_i-x_{i-1})^2(x_i-x_{i-2})^2(x_i-x_{i-3})-(x_i-x_{i-2})^2(x_i-x_{i-3})^2(x_i-x_{i-1})-(x_i-x_{i-1})^2(x_i-x_{i-3})^2(x_i-x_{i-2})}$$

$$+TE$$

For equally spaced points, i.e. $x_i-x_{i-1} = x_{i-1}-x_{i-2} = x_{i-3}-x_{i-2} = h$, this yields the following second order accurate formula:

$$\left.\frac{df}{dx}\right|_{x=x_i} = \frac{37f(x_i)+36f(x_{i-1})-9f(x_{i-2})+8f(x_{i-3})}{30h}$$

**6.6** Derive a finite difference approximation formula for $f''(x_i)$ using three points $x_{i-1}$, $x_i$, and $x_{i+1}$, where the spacing is such that $x_i - x_{i-1} = 2h$ and $x_{i+1} - x_i = h$.




Solution

Expand $f(x_{i+1})$ and $f(x_{i-1})$ in terms of a Taylor series about $x = x_i$:

$$f(x_{i+1}) = f(x_i) + h\left.\frac{df}{dx}\right|_{x=x_i} + \frac{h^2}{2}\left.\frac{d^2f}{dx^2}\right|_{x=x_i} + \frac{h^3}{6}\left.\frac{d^3f}{dx^3}\right|_{x=\xi_i}$$

$$f(x_{i-1}) = f(x_i) - 2h\left.\frac{df}{dx}\right|_{x=x_i} + 2h^2\left.\frac{d^2f}{dx^2}\right|_{x=x_i} - \frac{4h^3}{3}\left.\frac{d^3f}{dx^3}\right|_{x=\eta_i}$$

Since we want a difference formula for the second derivative in terms of the values of the function, we seek to eliminate the first derivative from the above expressions. This can be done by multiplying the first equation by 2 and adding it to the second equation:

$$2f(x_{i+1}) + f(x_{i-1}) = 3f(x_i) + 3h^2\left.\frac{d^2f}{dx^2}\right|_{x=x_i} + O(h^3)$$

Solving for the second derivative yields:

$$\left.\frac{d^2f}{dx^2}\right|_{x=x_i} = \frac{2f(x_{i+1}) + f(x_{i-1}) - 3f(x_i)}{3h^2} + O(h^3)$$

This result can also be obtained by using a Lagrange interpolating polynomial over the given three points and differentiating it twice and evaluating at $x=x_i$.

**6.7** A particular finite difference formula for the first derivative of a function is:

$$f'(x_i) = \frac{-f(x_{i+3}) + 9f(x_{i+1}) - 8f(x_i)}{6h}$$

where the points $x_i$, $x_{i+1}$, $x_{i+2}$, and $x_{i+3}$ are all equally spaced with step size $h$. What is the order of the truncation or discretization error?

**Solution**

The Taylor expansions are:

$$f(x_{i+1}) = f(x_i) + h\frac{df}{dx}\bigg|_{x=x_i} + \frac{h^2}{2}\frac{d^2f}{dx^2}\bigg|_{x=x_i} + \frac{h^3}{6}\frac{d^3f}{dx^3}\bigg|_{x=\xi_i}$$

$$f(x_{i+3}) = f(x_i) + 3h\frac{df}{dx}\bigg|_{x=x_i} + \frac{9h^2}{2}\frac{d^2f}{dx^2}\bigg|_{x=x_i} + \frac{9h^3}{2}\frac{d^3f}{dx^3}\bigg|_{x=\eta_i}$$

Substituting into the given difference formula yields:

$$f'(x_i) = \frac{-f(x_{i+3}) + 9f(x_{i+1}) - 8f(x_i)}{6h} = \frac{df}{dx}\bigg|_{x=x_i} + O(h^2)$$

Thus, the truncation error is of order $h^2$.

**6.8** The following data show the number of female and male physicians in the U.S. for various years (American Medical Association):

| Year | 1970 | 1980 | 1990 | 2000 | 2002 | 2003 | 2006 |
|---|---|---|---|---|---|---|---|
| # males | 308,627 | 413,395 | 511,227 | 618,182 | 638,182 | 646,493 | 665,647 |
| # females | 25, 401 | 54,284 | 104,194 | 195,537 | 215,005 | 225,042 | 256,257 |

(*a*) Calculate the rate of change in the number of male and female physicians in 2003 by using the three-point backward difference formula for the derivative, with unequally spaced points, Eq. (6.37).

(*b*) Use the result from part (*a*) and the three-point central difference formula for the derivative with unequally spaced points, Eq. (6.36), to calculate (predict) the number of male and female physicians in 2006.

**Solution**

(*a*) Eq.(6.37) is:

$$f'(x_{i+2}) = \frac{x_{i+2}-x_{i+1}}{(x_i - x_{i+1})(x_i - x_{i+2})} y_i + \frac{x_{i+2}-x_i}{(x_{i+1}-x_i)(x_{i+1}-x_{i+2})} y_{i+1} + \frac{2x_{i+2}-x_i-x_{i+1}}{(x_{i+2}-x_i)(x_{i+2}-x_{i+1})} y_{i+2}$$

For this problem, $x_{i+2} = 2003$, $x_{i+1} = 2002$, and $x_i = 2000$. Applying the above formula to the number of male physicians yields:

$$f'_{male}(2003) = \frac{1}{(-2)(-3)}(618182) + \frac{3}{(2)(-1)}(638182) + \frac{4}{(3)(1)}(646493)$$

$$= 103030 - 957273 + 861990 = 7747 \text{ male physicians per year}$$

Applying the above formula to the number of female physicians yields:

$$f'_{female}(2003) = \frac{1}{(-2)(-3)}(195537) + \frac{3}{(2)(-1)}(215005) + \frac{4}{(3)(1)}(225042)$$

$$= 32589.5 - 322507.5 + 300056 = 10138 \text{ female physicians per year}$$

(*b*) Equation (6.36) is:

$$f'(x_{i+1}) = \frac{x_{i+1}-x_{i+2}}{(x_i - x_{i+1})(x_i - x_{i+2})} y_i + \frac{2x_{i+1}-x_i-x_{i+2}}{(x_{i+1}-x_i)(x_{i+1}-x_{i+2})} y_{i+1} + \frac{x_{i+1}-x_i}{(x_{i+2}-x_i)(x_{i+2}-x_{i+1})} y_{i+2}$$

Now, $x_{i+1} = 2003$, $x_i = 2002$, and $x_{i+2} = 2006$. Applying this formula to the male physicians and using the result from part (*a*),

$$f'_{male}(2003) = 7747 = \frac{-3}{(-1)(-4)}(638182) + \frac{-2}{(1)(-3)}(646493) + \frac{1}{(4)(3)}(y_{i+2})$$
$$= -478636.5 + 430995.3 + \frac{1}{12}y_{i+2}$$

Solving for $y_{i+2}$ yields $y_{i+2} = 664658$. Comparing with the exact value of 665647, this is in error by 0.15%. Applying the three-point ventral difference formula to the female physicians and using the result from part (*a*),

$$f'_{female}(2003) = 10138 = \frac{-3}{(-1)(-4)}(215005) + \frac{-2}{(1)(-3)}(225042) + \frac{1}{(4)(3)}(y_{i+2})$$
$$= -161254 + 150029 + \frac{1}{12}y_{i+2}$$

Solving for $y_{i+2}$ yields $y_{i+2} = 256356$. Comparing with the exact value of 256,257, this is in error by less than 0.038%.

**6.9** Use the data from Problem 6.8 and the four-point backward difference formula that was derived in Problem 6.5 for evaluating the first derivative of a function specified at unequally spaced points to calculate the following quantities.
(*a*) Evaluate the rate of change in the number of male and female physicians in 2006.
(*b*) Use the data from 2003, 2006, together with the slopes in 2006 from part (*a*) to estimate the year in which the number of female and male physicians will be equal. Use the three-point central difference formula for the derivative (Eq. (6.36)) of a function specified at unequally spaced points.

**Solution**

(a) Using the formula for the first derivative derived in Problem 6.5:

$$\left.\frac{df}{dx}\right|_{x=x_i}$$

$$= \frac{(x_i-x_{i-2})^2(x_i-x_{i-3})^2 f(x_{i-1}) + (x_i-x_{i-1})^2(x_i-x_{i-3})^2 f(x_{i-2}) - 2(x_i-x_{i-1})^2(x_i-x_{i-2})^2 f(x_{i-3})}{2(x_i-x_{i-1})^2(x_i-x_{i-2})^2(x_i-x_{i-3}) - (x_i-x_{i-2})^2(x_i-x_{i-3})^2(x_i-x_{i-1}) - (x_i-x_{i-1})^2(x_i-x_{i-3})^2(x_i-x_{i-2})}$$

$$+ \frac{[2(x_i-x_{i-1})^2(x_i-x_{i-2})^2 - (x_i-x_{i-1})^2(x_i-x_{i-3})^2 - (x_i-x_{i-2})^2(x_i-x_{i-3})^2]f(x_i)}{2(x_i-x_{i-1})^2(x_i-x_{i-2})^2(x_i-x_{i-3}) - (x_i-x_{i-2})^2(x_i-x_{i-3})^2(x_i-x_{i-1}) - (x_i-x_{i-1})^2(x_i-x_{i-3})^2(x_i-x_{i-2})}$$

$$+ TE$$

In this problem, $x_i$ = 2006, $x_{i-1}$ = 2003, $x_{i-2}$ = 2002, and $x_{i-3}$ = 2000. For male physicians, the rate of change is:

$$= \frac{(4)^2(6)^2(646{,}493) + (3)^2(6)^2(638{,}182) - 2(3)^2(4)^2(618{,}182)}{2(3)^2(4)^2(6) - (4)^2(6)^2(3) - (3)^2(6)^2(4)}$$

$$+ \frac{\left[2(3)^2(4)^2 - (3)^2(6)^2 - (4)^2(6)^2\right](665{,}647)}{2(3)^2(4)^2(6) - (4)^2(6)^2(3) - (3)^2(6)^2(4)} = \frac{401114520 - 407375964}{-1296}$$

$$= \frac{-6261444}{-1296}$$

or 4831 male physicians per year. For female physicians, the rate of change is:

$$= \frac{(4)^2(6)^2(225{,}042) + (3)^2(6)^2(215{,}005) - 2(3)^2(4)^2(195{,}537)}{2(3)^2(4)^2(6) - (4)^2(6)^2(3) - (3)^2(6)^2(4)}$$

$$+ \frac{\left[2(3)^2(4)^2 - (3)^2(6)^2 - (4)^2(6)^2\right](256{,}257)}{2(3)^2(4)^2(6) - (4)^2(6)^2(3) - (3)^2(6)^2(4)} = \frac{142971156 - 156829284}{-1296}$$

$$= \frac{-13858128}{-1296}$$

or 10,693 female physicians per year.

(b) Eq.(6.36) is:

$$f'(x_{i+1}) = \frac{x_{i+1}-x_{i+2}}{(x_i-x_{i+1})(x_i-x_{i+2})}\,y_i + \frac{2x_{i+1}-x_i-x_{i+2}}{(x_{i+1}-x_i)(x_{i+1}-x_{i+2})}\,y_{i+1} + \frac{x_{i+1}-x_i}{(x_{i+2}-x_i)(x_{i+2}-x_{i+1})}\,y_{i+2}$$

But now, $x_i$ = 2003, $x_{i+1}$ = 2006, and $x_{i+2}$ is the unknown we seek. The quantity $y_{i+2}$ must be the same for male and female physicians. Therefore, the last two terms will be common for both:

$$f'_{male}(2006) + \frac{(x_{i+2}-2006)}{3(x_{i+2}-2003)}\,y_i^{male} + \frac{(2009-x_{i+2})}{3(x_{i+2}-2006)}\,y_{i+1}^{male}$$

$$= f'_{female}(2006) + \frac{(x_{i+2}-2006)}{3(x_{i+2}-2003)}\,y_i^{female} + \frac{(2009-x_{i+2})}{3(x_{i+2}-2006)}\,y_{i+1}^{female}$$

For the slope based on the formula from Problem 6.5, this equation reduces to:

$$-17586(x_{i+2}-2006)(x_{i+2}-2003) + (x_{i+2}-2006)^2(421{,}451)$$

$$+(2009-x_{i+2})(x_{i+2}-2003)(409{,}390) = 0$$

This is just a quadratic equation for $x_{i+2}$ which can be solved by hand. There are two real roots, $x_{i+2}$ = -1974.96 and $x_{i+2}$ = 2027.49. The first root is clearly non-physical (negative). So the answer is that between the years of 2027 and 2028, the number of male and female physicians will be equal.

**6.10** Use Lagrange interpolation polynomials to find the finite difference formula for the second derivative at the point $x = x_i$ using the unequally spaced points $x_i$, $x_{i+1}$, and $x_{i+2}$. What is the second derivative at $x = x_{i+1}$ and at $x = x_{i+2}$?

**Solution**

The Lagrange interpolation polynomial for these three points is given by Eq.(6.33) in Chapter 6:

$$f(x) = \frac{(x-x_{i+1})(x-x_{i+2})}{(x_i-x_{i+1})(x_i-x_{i+2})}y_i + \frac{(x-x_i)(x-x_{i+2})}{(x_{i+1}-x_i)(x_{i+1}-x_{i+2})}y_{i+1} + \frac{(x-x_i)(x-x_{i+1})}{(x_{i+2}-x_i)(x_{i+2}-x_{i+1})}y_{i+2}$$

Expanding the products:

$$f(x) = \frac{(x^2-(x_{i+1}+x_{i+2})x+x_{i+1}x_{i+2})}{(x_i-x_{i+1})(x_i-x_{i+2})}y_i + \frac{(x^2-(x_i+x_{i+2})x+x_ix_{i+2})}{(x_{i+1}-x_i)(x_{i+1}-x_{i+2})}y_{i+1} + \frac{(x^2-(x_i+x_{i+1})x+x_ix_{i+1})}{(x_{i+2}-x_i)(x_{i+2}-x_{i+1})}y_{i+2}$$

The second derivative can be found by simply differentiating this function twice to yield:

$$\frac{d^2f}{dx^2} = \frac{2y_i}{(x_i-x_{i+1})(x_i-x_{i+2})} + \frac{2y_{i+1}}{(x_{i+1}-x_i)(x_{i+1}-x_{i+2})} + \frac{2y_{i+2}}{(x_{i+2}-x_i)(x_{i+2}-x_{i+1})}$$

Note that the second derivative does not depend on $x$ since $f(x)$ was quadratic to begin with. Thus the second derivative is a constant and has the same value given by the above expression both at $x = x_{i+1}$ and at $x = x_{i+2}$.

**6.11** Given the function $f(x) = \dfrac{2x-1}{(x^4\sin x + x + 1)^{1/4}}$ find the value of the first derivative at $x = 2$.

(*a*) Use analytical differentiation by hand.

(*b*) Use four-point central difference formula with $x_{i-2} = 1.96$, $x_{i-1} = 1.98$, $x_{i+1} = 2.02$, and $x_{i+2} = 2.04$. (Write a MATLAB program in a script file to carry out the calculations.)

**Solution**

(*a*) The derivative is found by using the relations from calculus summarized in Chapter 2:

(a) Apply the rule for differentiation of a quotient:

$$\frac{d}{dx}\left(\frac{u}{v}\right) = \frac{v\dfrac{du}{dx} - u\dfrac{dv}{dx}}{v^2}$$

In this case, $u(x) = x^2 - 1$, and $v(x) = x^3\tan x$. Thus,

$$f'(x) = \frac{\left(x^3\tan x\right)(2x) - (x^2-1)\left(3x^2\tan x + x^3\sec^2 x\right)}{x^6\tan^2 x}$$; evaluating at $x = 2$ (and remembering that the argument of the trigonometric functions is in radians,

$$f'(2) = \frac{(8\tan 2)(4) - (4-1)\left(12\tan 2 + 8\sec^2 2\right)}{64\tan^2 2} = -0.4249$$

(*b*) The four-point central difference formula is given in Table 6-1:

$$f'(x_i) = \frac{f(x_{i-2}) - 8f(x_{i-1}) + 8f(x_{i+1}) - f(x_{i+2})}{12h}$$

The following script file calculates the derivative using this formula with $h = 0.02$:

```
clear, clc
f=@ (x) (x^2-1)/(x^3*tan(x));
xi_minus_2=1.96; xi_minus_1=1.98; xi_plus_1=2.02; xi_plus_2=2.04;
fi_minus_2=f(xi_minus_2); fi_minus_1=f(xi_minus_1);
fi_plus_1=f(xi_plus_1); fi_plus_2=f(xi_plus_2); h=0.02;
first_deriv=(fi_minus_2-(8*fi_minus_1)+(8*fi_plus_1)-fi_plus_2)/12/h
```

When executed in the command window, this script file produces the following output:

```
first_deriv =
   -0.4249
>>
```

**6.12** For the function given in Problem 6.11, find the value of the second derivative at $x = 2$.
(*a*) Use analytical differentiation by hand.
(*b*) Use five-point central difference formula with $x_{i-2} = 1.96$, $x_{i-1} = 1.98$, $x_i = 2$, $x_{i+1} = 2.02$, and $x_{i+2} = 2.04$. (Write a MATLAB program in a script file to carry out the calculations.)

**Solution**

(a) Apply the rule for differentiation of a quotient:

$$\frac{d}{dx}\left(\frac{u}{v}\right) = \frac{v\frac{du}{dx} - u\frac{dv}{dx}}{v^2}$$

In this case, $u(x) = x^2 - 1$, and $v(x) = x^3\tan x$. Thus,

$$f'(x) = \frac{\left(x^3 \tan x\right)(2x) - (x^2 - 1)\left(3x^2 \tan x + x^3 \sec^2 x\right)}{x^6 \tan^2 x}$$; evaluating at $x = 2$ (and remembering that the argument of the trigonometric functions is in radians,

$$f'(2) = \frac{(8\tan 2)(4) - (4-1)\left(12\tan 2 + 8\sec^2 2\right)}{64\tan^2 2} = -0.4249$$

(*b*) The five-point central difference formula for the second derivative is given in Table 6-1:

$$f''(x_i) = \frac{-f(x_{i-2}) + 16f(x_{i-1}) - 30f(x_i) + 16f(x_{i+1}) - f(x_{i+2})}{12h^2}$$

The following script file calculates the derivative using this formula with $h = 0.02$:

```
clear, clc
f=@ (x) (x^2-1)/(x^3*tan(x));
xi_minus_2=1.96; xi_minus_1=1.98; xi=2; xi_plus_1=2.02; xi_plus_2=2.04;
fi_minus_2=f(xi_minus_2); fi_minus_1=f(xi_minus_1); fi=f(xi);
fi_plus_1=f(xi_plus_1); fi_plus_2=f(xi_plus_2); h=0.02;
second_deriv=(-fi_minus_2+(16*fi_minus_1)-(30*fi)+(16*fi_plus_1)-fi_plus_2)/
12/h/h
```

When executed in the command window, this script file produces the following output:

```
second_deriv =
   -0.2067
```

**6.13** The following data for the velocity component in the *x*-direction, *u*, are obtained as a function of the two coordinates *x* and *y*.

| | $x = 0$ | $x = 1$ | $x = 2$ | $x = 3$ | $x = 4$ |
|---|---|---|---|---|---|
| $y = 0$ | 0 | 2 | 8 | 13 | 15 |
| $y = 1$ | 3 | 7 | 10 | 15 | 18 |
| $y = 2$ | 14 | 8 | 14 | 22 | 22 |
| $y = 3$ | 7 | 9 | 12 | 16 | 17 |
| $y = 4$ | 5 | 7 | 10 | 9 | 14 |

Use the four-point central difference formula for $\frac{\partial^2 u}{\partial y \partial x}$ to evaluate this derivative at the point $(2, 3)$.

**Solution**

The four-point central difference formula for the mixed partial derivative is given by Eq.(6.65):

$$\left.\frac{\partial^2 u}{\partial x \partial y}\right|_{\substack{x = x_i \\ y = y_i}} = \frac{[u(x_{i+1}, y_{i+1}) - u(x_{i-1}, y_{i+1})] - [u(x_{i+1}, y_{i-1}) - u(x_{i-1}, y_{i-1})]}{2h_x \cdot 2h_y}$$

In this problem, $h_x = h_y = 1$. For $(x_i, y_i) = (2, 3)$, $u(x_{i+1}, y_{i+1}) = u(3, 4) = 9$, $u(x_{i-1}, y_{i+1}) = u(1, 4) = 7$, $u(x_{i+1}, y_{i-1}) = u(3, 2) = 22$, and $u(x_{i-1}, y_{i-1}) = u(1, 2) = 8$. Substituting these values into the difference formula yields:

$$\left.\frac{\partial^2 u}{\partial x \partial y}\right|_{\substack{x = 2 \\ y = 3}} = \frac{[9 - 7] - [22 - 8]}{2(1) \cdot 2(1)} = \frac{-12}{4} = -3$$

**6.14** Use Lagrange polynomials to develop a difference formula for the second derivative of a function that is specified by a discrete set of data points with unequal spacing. The formula determines the second derivative at point $(x_i, y_i)$ using points $(x_{i-1}, y_{i-1})$, $(x_i, y_i)$, and $(x_{i+1}, y_{i+1})$.

**Solution**

Using Eq. (5.45),

$$f(x) = \frac{(x - x_i)(x - x_{i+1})}{(x_{i-1} - x_i)(x_{i-1} - x_{i+1})} f(x_{i-1}) + \frac{(x - x_{i-1})(x - x_{i+1})}{(x_i - x_{i-1})(x_i - x_{i+1})} f(x_i) + \frac{(x - x_{i-1})(x - x_i)}{(x_{i+1} - x_{i-1})(x_{i+1} - x_i)} f(x_{i+1})$$

Differentiating once with respect to x,

$$f'(x) = \frac{(2x - x_i - x_{i+1})}{(x_{i-1} - x_i)(x_{i-1} - x_{i+1})} f(x_{i-1}) + \frac{(2x - x_{i-1} - x_{i+1})}{(x_i - x_{i-1})(x_i - x_{i+1})} f(x_i) + \frac{(2x - x_i - x_{i-1})}{(x_{i+1} - x_{i-1})(x_{i+1} - x_i)} f(x_{i+1})$$

Differentiating again,

$$f''(x) = \frac{2f(x_{i-1})}{(x_{i-1} - x_i)(x_{i-1} - x_{i+1})} + \frac{2f(x_i)}{(x_i - x_{i-1})(x_i - x_{i+1})} + \frac{2f(x_{i+1})}{(x_{i+1} - x_{i-1})(x_{i+1} - x_i)}$$

Note that the second derivative is just a constant based on the function values at each point and the spacing between the points. This formula gives the same value for the second derivative at the $x_i$ for every i.

**6.15** Using Lagrange polynomials, develop a difference formula for the third derivative of a function that is specified by a discrete set of data points. The formula determines the third derivative at point $(x_i, y_i)$ using points $(x_{i-1}, y_{i-1})$, $(x_i, y_i)$, $(x_{i+1}, y_{i+1})$, and $(x_{i+2}, y_{i+2})$. The points are spaced such that $x_i - x_{i-1} = x_{i+2} - x_{i+1} = h$ and $x_{i+1} - x_i = 2h$.

**Solution**

Using Eq. (5.45),

$$f(x) = \frac{(x-x_i)(x-x_{i+1})(x-x_{i+2})}{(x_{i-1}-x_i)(x_{i-1}-x_{i+1})(x_{i-1}-x_{i+2})}f(x_{i-1}) + \frac{(x-x_{i-1})(x-x_{i+1})(x-x_{i+2})}{(x_i-x_{i-1})(x_i-x_{i+1})(x_i-x_{i+2})}f(x_i) + \frac{(x-x_{i-1})(x-x_i)(x-x_{i+2})}{(x_{i+1}-x_{i-1})(x_{i+1}-x_i)(x_{i+1}-x_{i+2})}f(x_{i+1}) + \frac{(x-x_{i-1})(x-x_i)(x-x_{i+1})}{(x_{i+2}-x_{i-1})(x_{i+2}-x_i)(x_{i+2}-x_{i+1})}f(x_{i+2})$$

Differentiating once with respect to x,

$$f'(x) = \frac{3x^2 - 2(x_{i+1}+x_i+x_{i+2})x + x_ix_{i+1} + x_{i+1}x_{i+2} + x_ix_{i+2}}{(x_{i-1}-x_i)(x_{i-1}-x_{i+1})(x_{i-1}-x_{i+2})}f(x_{i-1}) + \frac{3x^2 - 2(x_{i+1}+x_{i-1}+x_{i+2})x + x_{i-1}x_{i+1} + x_{i+1}x_{i+2} + x_{i-1}x_{i+2}}{(x_i-x_{i-1})(x_i-x_{i+1})(x_i-x_{i+2})}f(x_i) + \frac{3x^2 - 2(x_i+x_{i-1}+x_{i+2})x + x_ix_{i-1} + x_ix_{i+2} + x_{i-1}x_{i+2}}{(x_{i+1}-x_{i-1})(x_{i+1}-x_i)(x_{i+1}-x_{i+2})}f(x_{i+1}) + \frac{3x^2 - 2(x_i+x_{i-1}+x_{i+1})x + x_ix_{i-1} + x_ix_{i+1} + x_{i-1}x_{i+1}}{(x_{i+2}-x_{i-1})(x_{i+2}-x_i)(x_{i+2}-x_{i+1})}f(x_{i+2})$$

Differentiating again,

$$f''(x) = \frac{6x - 2(x_{i+1}+x_i+x_{i+2})}{(x_{i-1}-x_i)(x_{i-1}-x_{i+1})(x_{i-1}-x_{i+2})}f(x_{i-1}) + \frac{6x - 2(x_{i+1}+x_{i-1}+x_{i+2})}{(x_i-x_{i-1})(x_i-x_{i+1})(x_i-x_{i+2})}f(x_i) + \frac{6x - 2(x_i+x_{i-1}+x_{i+2})}{(x_{i+1}-x_{i-1})(x_{i+1}-x_i)(x_{i+1}-x_{i+2})}f(x_{i+1}) + \frac{6x - 2(x_i+x_{i-1}+x_{i+1})}{(x_{i+2}-x_{i-1})(x_{i+2}-x_i)(x_{i+2}-x_{i+1})}f(x_{i+2})$$

Differentiating one more time,

$$f'''(x) = \frac{6}{(x_{i-1}-x_i)(x_{i-1}-x_{i+1})(x_{i-1}-x_{i+2})}f(x_{i-1})$$
$$+\frac{6}{(x_i-x_{i-1})(x_i-x_{i+1})(x_i-x_{i+2})}f(x_i)$$
$$+\frac{6}{(x_{i+1}-x_{i-1})(x_{i+1}-x_i)(x_{i+1}-x_{i+2})}f(x_{i+1})$$
$$+\frac{6}{(x_{i+2}-x_{i-1})(x_{i+2}-x_i)(x_{i+2}-x_{i+1})}f(x_{i+2})$$

Note that the third derivative is just a constant based on the function values at each point and the spacing between the points. This formula gives the same value for the second derivative at the $x_i$ for every i. Substituting $x_i - x_{i-1} = x_{i+2} - x_{i+1} = h$, and $x_{i+1} - x_i = 2h$,

$$f'''(x) = -\frac{1}{2h^3}f(x_{i-1}) + \frac{1}{h^3}f(x_i) - \frac{1}{h^3}f(x_{i+1}) + \frac{1}{2h^3}f(x_{i+2}) \text{ or}$$

$$\boxed{f'''(x) = \frac{f(x_{i+2}) - 2f(x_{i+1}) + 2f(x_i) - f(x_{i-1})}{2h^3}}$$

**6.16** Write a MATLAB user-defined function that determines the first derivative of a function that is given by a set of discrete points with equal spacing. For the function name use `yd = FirstDeriv(x,y)`. The input arguments `x` and `y` are vectors with the coordinates of the points, and the output argument `yd` is a vector with the values of the derivative at each point. At the first and last points, the function should calculate the derivative with the three-point forward and backward difference formulas, respectively. At all the other points `FirstDeriv` should use the two-point central difference formula. Use `FirstDeriv` to calculate the derivative of the function that is given in Problem 6.1.

**Solution**

The following user-defined MATLAB function solves this problem:

```
function yd = FirstDeriv(x,y)
n=length(x); h=x(2)-x(1);
for i=2:n-1
    yd(i)=(y(i+1)-y(i-1))/2/h;
end
yd(1)=(-3*y(1)+4*y(2)-y(3))/2/h;
yd(n)=(y(n-2)-4*y(n-1)+3*y(n))/2/h;
```

The user-defined function `FirstDeriv` is next used in the script file for calculating the derivative of the function that is given in Problem 6.1.

```
x=0.398:0.001:0.402;
fx=[0.063 0.06375 0.0649 0.0654 0.0658];
dfx=FirstDeriv(x,fx)
```

When the script is executed, the following result is displayed in the Command Window:

```
dfx =
    0.5500    0.9500    0.8250    0.4500    0.3500
```

**6.17** Write a MATLAB user-defined function that calculates the second derivative of a function that is given by a set of discrete data points with equal spacing. For the function name and arguments use `ydd=SecDeriv(x,y)`, where the input arguments `x` and `y` are vectors with the coordinates of the points, and `ydd` is a vector with the values of the second derivative at each point. For calculating the derivative, the function `SecDeriv` should use the finite difference formulas that have a truncation error of $O(h^2)$. Use `SecDeriv` for calculating the derivative of the function that is given by the following set of points:

| $x$ | -1 | -0.5 | 0 | 0.5 | 1 | 1.5 | 2 | 2.5 | 3 | 3.5 | 4 | 4.5 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $f(x)$ | -3.632 | -0.3935 | 1 | 0.6487 | -1.282 | -4.518 | -8.611 | -12.82 | -15.91 | -15.88 | -9.402 | 9.017 |

**Solution**

From Table 6-1, the second derivative at the interior points can be calculated using the three-point central difference formula:

$$f''(x_i) = \frac{f(x_{i-1}) - 2f(x_i) + f(x_{i+1})}{h^2}$$

At the first point, to obtain second order accuracy, the second derivative must be evaluated using the four-point forward difference formula:

$$f''(x_i) = \frac{2f(x_i) - 5f(x_{i+1}) + 4f(x_{i+2}) - f(x_{i+3})}{h^2}$$

At the last point, to obtain second order accuracy, the second derivative must be evaluated using the four-point forward difference formula:

$$f''(x_i) = \frac{-f(x_{i-3}) + 4f(x_{i-2}) - 5f(x_{i-1}) + 2f(x_i)}{h^2}$$

The following user-defined MATLAB function accomplishes this task:

```
function ydd=SecDeriv(x,y)
n=length(x); h=x(2)-x(1);
for i=2:n-1
    ydd(i)=(y(i-1)-2*y(i)+y(i+1))/h/h;
end
ydd(1)=(2*y(1)-5*y(2)+4*y(3)-y(4))/h/h;
ydd(n)=(-y(n-3)+4*y(n-2)-5*y(n-1)+2*y(n))/h/h;
```

The user-defined function `SecDeriv` is next used in the Command Window for calculating the derivative of the function that is given.

```
>> format long
>> x=-1:0.5:4.5;
>> y=[-3.632 -0.3935 1 0.6487 -1.282 -4.518 -8.611 -12.82 -15.91 -15.88
-9.402 9.017];
>> ydd=SecDeriv(x,y)
ydd =
  Columns 1 through 3
  -7.78080000000000  -7.38000000000000  -6.97920000000000
  Columns 4 through 6
  -6.31760000000000  -5.22120000000000  -3.42800000000000
  Columns 7 through 9
  -0.46400000000000   4.47600000000000  12.48000000000000
  Columns 10 through 12
  25.79200000000001  47.76399999999999  69.73599999999999
```

**6.18** Write a MATLAB user-defined function that determines the first and second derivatives of a function that is given by a set of discrete points with equal spacing. For the function name use `[yd,ydd] = FrstScndDeriv(x,y)`. The input arguments `x` and `y` are vectors with the coordinates of the points, and the output arguments `yd` and `ydd` are vectors with the values of the first and second derivatives, respectively, at each point. For calculating both derivatives, the function should use the finite difference formulas that have a truncation error of $O(h^2)$.

(*a*) Use the function `FrstScndDeriv` to calculate the derivatives of the function that is given by the data in Problem 6.17.

(*b*) Modify the function (rename it `FrstScndDerivPt`) such that it also creates three plots (one page in a column). The top plot should be of the function, the second plot of the first derivative, and the third of the second derivative. Apply the function `FrstScndDerivPt` to the data in Problem 6.17.

**Solution**

(*a*) The user-defined function `FrstScndDeriv` is listed below. It uses the equations used in Problems 2.14 and 2.15 for calculating the derivatives.

```
function [yd,ydd] = FrstScndDeriv(x,y)
n=length(x); h=x(2)-x(1);
for i=2:n-1
    yd(i)=(y(i+1)-y(i-1))/2/h;
    ydd(i)=(y(i-1)-2*y(i)+y(i+1))/h/h;
end
yd(1)=(-3*y(1)+4*y(2)-y(3))/2/h;
yd(n)=(y(n-2)-4*y(n-1)+3*y(n))/2/h;
ydd(1)=(2*y(1)-5*y(2)+4*y(3)-y(4))/h/h;
ydd(n)=(-y(n-3)+4*y(n-2)-5*y(n-1)+2*y(n))/h/h;
end
```

The user-defined function `FrstScndDeriv` is next used in the Command Window for calculating the derivative of the function that is given in Problem 6.15.

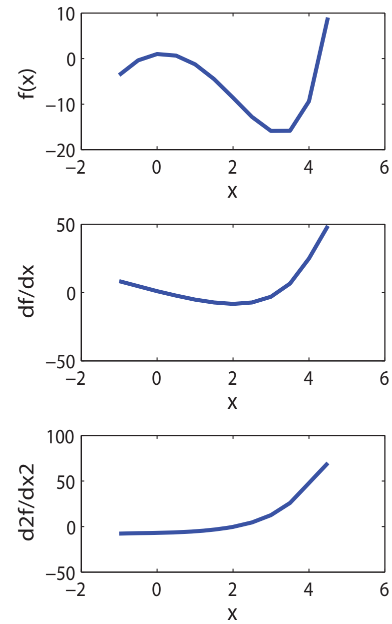
```
>> format long
>> x=-1:0.5:4.5;
>> y=[-3.632 -0.3935 1 0.6487 -1.282 -4.518 -8.611 -12.82 -15.91 -15.88
-9.402 9.017];
```

```
>> [yd,ydd] = FrstScndDeriv(x,y)
yd =
  Columns 1 through 4
   8.32200000000000   4.63200000000000   1.04220000000000  -2.28200000000000
  Columns 5 through 8
  -5.16670000000000  -7.32900000000000  -8.30200000000000  -7.29900000000000
  Columns 9 through 12
  -3.06000000000000   6.50800000000000  24.89700000000000  48.77900000000000
ydd =
  Columns 1 through 4
  -7.78080000000000  -7.38000000000000  -6.97920000000000  -6.31760000000000
  Columns 5 through 8
  -5.22120000000000  -3.42800000000000  -0.46400000000000   4.47600000000000
  Columns 9 through 12
  12.48000000000000  25.79200000000001  47.76399999999999  69.73599999999999
```

(*b*) The function from part (*a*) is modified such that it also creates three plots (on the same page in a column). The top plot should be of the function, the second plot of the first derivative, and the third of the second derivative. The listing of the function is:

```
function [yd,ydd] = FrstScndDerivPt(x,y)
n=length(x); h=x(2)-x(1);
for i=2:n-1
    yd(i)=(y(i+1)-y(i-1))/2/h;
    ydd(i)=(y(i-1)-2*y(i)+y(i+1))/h/h;
end
yd(1)=(-3*y(1)+4*y(2)-y(3))/2/h;
yd(n)=(y(n-2)-4*y(n-1)+3*y(n))/2/h;
ydd(1)=(2*y(1)-5*y(2)+4*y(3)-y(4))/h/h;
ydd(n)=(-y(n-3)+4*y(n-2)-5*y(n-1)+2*y(n))/h/h;
subplot(3,1,1)
plot(x,y); xlabel('x'); ylabel('f(x)');
subplot(3,1,2)
plot(x,yd); xlabel('x'); ylabel('df/dx');
subplot(3,1,3)
plot(x,ydd); xlabel('x'); ylabel('d2f/dx2');
```

When the function is used for calculating the derivative of the function given in Problem 6.15, the display is the same as in Part (*a*). In addition, the figure on the right is displayed in the Figure Window:



**6.19** Write a MATLAB user-defined function that determines the first derivative of a function that is given in an analytical form. For the function name and arguments use `dfx = DiffAnaly(Fun,xi)`. `Fun` is a name for the function that is being differentiated. It is a dummy name for the function that is imported into `DiffAnaly`. The actual function that is differentiated should be written as an anonymous function, or as a user-defined function, that calculates the values of $f(x)$ for given values of $x$. It is entered as a function handle when `DiffAnaly` is used. `xi` is the value of $x$ where the derivative is calculated. The user-defined function should calculate the derivative by using the two-point central difference formula. In the formula, the values of $(x_{i+1})$ and $(x_{i-1})$ should be taken to be 5% higher and 5% lower than the value of $(x_i)$, respectively.
(*a*) Use `DiffAnaly` to calculate the first derivative of $f(x) = e^x \ln x$ at $x = 2$.
(*b*) Use `DiffAnaly` to calculate the first derivative of the function from Problem 6.11 at $x = 2$.

**Solution**

The listing of the user-defined function `DiffAnaly` is:

```
function dfx = DiffAnaly(Fun,xi)
h=0.05*xi;
dfx=(Fun(xi+h)-Fun(xi-h))/(2*h);
```

Parts *a* and *b* are solved in the following script file:

```
% Solution of HW6_19se
% Part a
clear, clc
Fa=@(x) exp(x)*log(x);
dFa = DiffAnaly(Fa,2)
% Part (b)
Fb=@(x) (x^2-1)/(x^3*tan(x));
dFb = DiffAnaly(Fb,2)
```

When the script is executed the following results are sisplayed in the Command Window.

```
dFa =
   8.837095456120050
dFb =
```

```
-0.426737579782771
```

**6.20** Modify the MATLAB user-defined function in Problem 6.19 to include Richardson's extrapolation. The function should calculate a first estimate for the derivative as described in Problem 6.19, and a second estimate by taking the values of $(x_{i+1})$ and $(x_{i-1})$ to be 2.5% higher and 2.5% lower than the value of $(x_i)$, respectively. The two estimates should then be used with Richardson's extrapolation for calculating the derivative. For the function name and arguments use `dfx=DiffRichardson(Fun,xi)`.

($a$) Use the function to calculate the derivative of $f(x) = e^x \ln x$ at $x = 2$.

($b$) Use the function to calculate the first derivative of the function that is given in Problem 6.11 at $x = 2$.

## Solution

Richardson's extrapolation for numerical differentiation is given by Eq.(6.45):

$$D = \frac{1}{3}\left(4D\left(\frac{h}{2}\right) - D(h)\right) + O(h^4)$$

The listing of the user-defined function `DiffRichardson` is:

```
function dfx = DiffRichardson(Fun,xi)
h1=0.05*xi;
dfxh1=(Fun(xi+h1)-Fun(xi-h1))/(2*h1);
h2=0.025*xi;
dfxh2=(Fun(xi+h2)-Fun(xi-h2))/(2*h2);
dfx=(4*dfxh2-dfxh1)/3;
```

Parts *a* and *b* are solved in the following script file:

```
% Solution of HW6_20se
% Part a
clear, clc
Fa=@(x) exp(x)*log(x);
dFa = DiffRichardson(Fa,2)
% Part (b)
Fb=@(x) (x^2-1)/(x^3*tan(x));
dFb = DiffRichardson(Fb,2)
```

When the script is executed the following results are sisplayed in the Command Window.

```
dFa =
   8.816228266468253
dFb =
  -0.424938238076213
```

**6.21** Write a MATLAB user-defined function that calculates the second derivative of a function that is given in an analytical form. For the function name and arguments use `ddfx = DDiffAnaly(Fun,xi)`. `Fun` is a name for the function that is being differentiated. It is a dummy name for the function that is imported into `DiffAnaly`. The actual function that is differentiated should be written as an anonymous function, or as a user-defined function, that calculates the values of $f(x)$ for given values of $x$. It is entered as a function handle when `DiffAnaly` is used. `xi` is the value of $x$ where the second derivative is calculated. The function should calculate the second derivative with the three-point central difference formula. In the formula, the values of $(x_{i+1})$ and $(x_{i-1})$ should be taken to be 5% higher and 5% lower than the value of $(x_i)$, respectively.

(*a*) Use the function to calculate the second derivative of $f(x) = \frac{2^x}{x}$ at $x = 2$.

(*b*) Use the function to calculate the second derivative of the function that is given in Problem 6.11 at $x = 2$.

**Solution**

The listing of the user-defined function `DDiffAnaly` is:

```
function ddfx = DDiffAnaly(Fun,xi)
h=0.05*xi;
ddfx=(Fun(xi-h)-2*Fun(xi)+Fun(xi+h))/(h^2);
```

Parts *a* and *b* are solved in the following script file:

```
% Solution of HW6_21se
% Part a
clear, clc
Fa=@(x) 2^x/x;
ddFa = DDiffAnaly(Fa,2)
% Part (b)
Fb=@(x) (x^2-1)/(x^3*tan(x));
ddFb = DDiffAnaly(Fb,2)
```

When the script is executed the following results are sisplayed in the Command Window.

```
ddFa =
```

```
   0.575324415664413
ddFb =
  -0.210384802742089
```

**6.22** Write a MATLAB user-defined function that evaluates the first derivative of a function that is given by a set of discrete points with unequal spacing. For the function name use `yd = FirstDerivUneq(x,y)`. The input arguments `x` and `y` are vectors with the coordinates of the points, and the output argument `yd` is a vector with the values of the first derivative at each point. The differentiation is done by using second-order Lagrange polynomials (Section 6.5). At the first and last points, the function should calculate the derivative with Eqs. (6.35) and (6.37), respectively. At all the other points the function should use Eq. (6.36). Use `FirstDerivUneq` to calculate the derivative of the function that is given by the following set of points:

| $x$ | -1 | -0.6 | -0.3 | 0 | 0.5 | 0.8 | 1.6 | 2.5 | 2.8 | 3.2 | 3.5 | 4 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $f(x)$ | -3.632 | -0.8912 | 0.3808 | 1.0 | 0.6487 | -0.3345 | -5.287 | -12.82 | -14.92 | -16.43 | -15.88 | -9.402 |

## Solution

The listing of the user-defined function `FirstDerivUneq` is:

```
function yd = FirstDerivUneq(x,y)
n=length(x);
yd(1)=(2*x(1)-x(2)-x(3))*y(1)/(x(1)-x(2))/(x(1)-x(3))...
    +(x(1)-x(3))*y(2)/(x(2)-x(1))/(x(2)-x(3))...
    +(x(1)-x(2))*y(3)/(x(3)-x(1))/(x(3)-x(2));
for j=2:n-1
    yd(j)=(x(j)-x(j+1))*y(j-1)/(x(j-1)-x(j))/(x(j-1)-x(j+1))...
        +(2*x(j)-x(j-1)-x(j+1))*y(j)/(x(j)-x(j-1))/(x(j)-x(j+1))...
        +(x(j)-x(j-1))*y(j+1)/(x(j+1)-x(j-1))/(x(j+1)-x(j));
end
yd(n)=(x(n)-x(n-1))*y(n-2)/(x(n-2)-x(n-1))/(x(n-2)-x(n))...
    +(x(n)-x(n-2))*y(n-1)/(x(n-1)-x(n-2))/(x(n-1)-x(n))...
    +(2*x(n)-x(n-2)-x(n-1))*y(n)/(x(n)-x(n-2))/(x(n)-x(n-1));
```

The user-defined function `FirstDerivUneq` is next used in the script file for calculating the derivative of function given in the problem statement:

```
% HW6_22 Script
```

```
clear, clc
x=[-1 -0.6 -0.3 0 0.5 0.8 1.6 2.5 2.8 3.2 3.5 4];
Fx=[-3.632 -0.8912 0.3808 1.0 0.6487 -0.3345 -5.287 -12.82 -14.92 -16.43 -15.88
-9.402];
dFx=FirstDerivUneq(x,Fx)
```

When the script is executed the following results are sisplayed in the Command Window.

```
dFx =
  Columns 1 through 8
    8.3446    5.3594    3.1520    1.0265   -2.3118   -4.0719   -7.2162
-7.3425
  Columns 9 through 12
   -5.6179   -0.5702    6.0043   19.9077
```

**6.23** Write a MATLAB user-defined function that evaluates the second derivative of a function that is given by a set of discrete data points with unequal spacing. For the function name use `yd = SndDerivUneq(x,y)`. The input arguments `x` and `y` are vectors with the coordinates of the points, and the output argument `yd` is a vector with the values of the second derivative at each point. Use the following scheme for the differentiation. Write a third-order Lagrange polynomial $f(x)$ for four points $(x_{i-1})$, $(x_i)$, $(x_{i+1})$, and $(x_{i+2})$, and derive formulas for the second derivative of the polynomial at each of the four points. `SndDerivUneq` uses the formula for $f''(x_{i-1})$ for calculating the second derivative at the first data point, and the formula for $f''(x_{i+2})$ and $f''(x_{i+1})$ for calculating the second derivative at the last data point and one point before the last, respectively. The formula for $f''(x_i)$ is used in all the points in between. Use the function to calculate the second derivative of the function that is given by the following set of points:

| $x$ | -1 | -0.6 | -0.3 | 0 | 0.5 | 0.8 | 1.6 | 2.5 | 2.8 | 3.2 | 3.5 | 4 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $f(x)$ | -3.632 | -0.8912 | 0.3808 | 1.0 | 0.6487 | -0.3345 | -5.287 | -12.82 | -14.92 | -16.43 | -15.88 | -9.402 |

**Solution**

A third-order Lagrange polynomial $f(x)$ for four points $(x_{i-1})$, $(x_i)$, $(x_{i+1})$, and $(x_{i+2})$ is:

$$f(x) = \frac{(x-x_i)(x-x_{i+1})(x-x_{i+2})}{(x_{i-1}-x_i)(x_{i-1}-x_{i+1})(x_{i-1}-x_{i+2})}y_{i-1} + \frac{(x-x_{i-1})(x-x_{i+1})(x-x_{i+2})}{(x_i-x_{i-1})(x_i-x_{i+1})(x_i-x_{i+2})}y_i +$$
$$\frac{(x-x_{i-1})(x-x_i)(x-x_{i+2})}{(x_{i+1}-x_{i-1})(x_{i+1}-x_i)(x_{i+1}-x_{i+2})}y_{i+1} + \frac{(x-x_{i-1})(x-x_i)(x-x_{i+1})}{(x_{i+2}-x_{i-1})(x_{i+2}-x_i)(x_{i+2}-x_{i+1})}y_{i+2}$$

Differentiation of the equation above gives:

$$f'(x) = \frac{3x^2 - 2x(x_{i+1} + x_i + x_{i+2}) + x_i x_{i+1} + x_{i+1} x_{i+2} + x_i x_{i+2}}{(x_{i-1} - x_i)(x_{i-1} - x_{i+1})(x_{i-1} - x_{i+2})} y_{i-1} +$$
$$\frac{3x^2 - 2x(x_{i+1} + x_{i-1} + x_{i+2}) + x_{i-1} x_{i+1} + x_{i+1} x_{i+2} + x_{i-1} x_{i+2}}{(x_i - x_{i-1})(x_i - x_{i+1})(x_i - x_{i+2})} y_i +$$
$$\frac{3x^2 - 2x(x_i + x_{i-1} + x_{i+2}) + x_{i-1} x_i + x_i x_{i+2} + x_{i-1} x_{i+2}}{(x_{i+1} - x_{i-1})(x_{i+1} - x_i)(x_{i+1} - x_{i+2})} y_{i+1} +$$
$$\frac{3x^2 - 2x(x_i + x_{i-1} + x_{i+1}) + x_{i-1} x_i + x_i x_{i+1} + x_{i-1} x_{i+1}}{(x_{i+2} - x_{i-1})(x_{i+2} - x_i)(x_{i+2} - x_{i+1})} y_{i+2}$$

Second differentiation of the equation above gives:

$$f''(x) = \frac{6x - 2(x_{i+1} + x_i + x_{i+2})}{(x_{i-1} - x_i)(x_{i-1} - x_{i+1})(x_{i-1} - x_{i+2})} y_{i-1} + \frac{6x - 2(x_{i+1} + x_{i-1} + x_{i+2})}{(x_i - x_{i-1})(x_i - x_{i+1})(x_i - x_{i+2})} y_i +$$
$$\frac{6x - 2(x_i + x_{i-1} + x_{i+2})}{(x_{i+1} - x_{i-1})(x_{i+1} - x_i)(x_{i+1} - x_{i+2})} y_{i+1} + \frac{6x - 2(x_i + x_{i-1} + x_{i+1})}{(x_{i+2} - x_{i-1})(x_{i+2} - x_i)(x_{i+2} - x_{i+1})} y_{i+2}$$

The second derivative at the four points $(x_{i-1})$, $(x_i)$, $(x_{i+1})$, and $(x_{i+2})$ is:

$$f''(x_{i-1}) = \frac{6x_{i-1} - 2(x_{i+1} + x_i + x_{i+2})}{(x_{i-1} - x_i)(x_{i-1} - x_{i+1})(x_{i-1} - x_{i+2})} y_{i-1} + \frac{4x_{i-1} - 2(x_{i+1} + x_{i+2})}{(x_i - x_{i-1})(x_i - x_{i+1})(x_i - x_{i+2})} y_i +$$
$$\frac{4x_{i-1} - 2(x_i + x_{i+2})}{(x_{i+1} - x_{i-1})(x_{i+1} - x_i)(x_{i+1} - x_{i+2})} y_{i+1} + \frac{4x_{i-1} - 2(x_i + x_{i+1})}{(x_{i+2} - x_{i-1})(x_{i+2} - x_i)(x_{i+2} - x_{i+1})} y_{i+2}$$

$$f''(x_i) = \frac{4x_i - 2(x_{i+1} + x_{i+2})}{(x_{i-1} - x_i)(x_{i-1} - x_{i+1})(x_{i-1} - x_{i+2})} y_{i-1} + \frac{6x_i - 2(x_{i+1} + x_{i-1} + x_{i+2})}{(x_i - x_{i-1})(x_i - x_{i+1})(x_i - x_{i+2})} y_i +$$
$$\frac{4x_i - 2(x_{i-1} + x_{i+2})}{(x_{i+1} - x_{i-1})(x_{i+1} - x_i)(x_{i+1} - x_{i+2})} y_{i+1} + \frac{4x_i - 2(x_{i-1} + x_{i+1})}{(x_{i+2} - x_{i-1})(x_{i+2} - x_i)(x_{i+2} - x_{i+1})} y_{i+2}$$

$$f''(x_{i+1}) = \frac{4x_{i+1} - 2(x_i + x_{i+2})}{(x_{i-1} - x_i)(x_{i-1} - x_{i+1})(x_{i-1} - x_{i+2})} y_{i-1} + \frac{4x_{i+1} - 2(x_{i-1} + x_{i+2})}{(x_i - x_{i-1})(x_i - x_{i+1})(x_i - x_{i+2})} y_i +$$
$$\frac{6x_{i+1} - 2(x_i + x_{i-1} + x_{i+2})}{(x_{i+1} - x_{i-1})(x_{i+1} - x_i)(x_{i+1} - x_{i+2})} y_{i+1} + \frac{4x_{i+1} - 2(x_i + x_{i-1})}{(x_{i+2} - x_{i-1})(x_{i+2} - x_i)(x_{i+2} - x_{i+1})} y_{i+2}$$

$$f''(x_{i+2}) = \frac{4x_{i+2}-2(x_{i+1}+x_i)}{(x_{i-1}-x_i)(x_{i-1}-x_{i+1})(x_{i-1}-x_{i+2})}y_{i-1} + \frac{4x_{i+2}-2(x_{i+1}+x_{i-1})}{(x_i-x_{i-1})(x_i-x_{i+1})(x_i-x_{i+2})}y_i + \frac{4x_{i+2}-2(x_i+x_{i-1})}{(x_{i+1}-x_{i-1})(x_{i+1}-x_i)(x_{i+1}-x_{i+2})}y_{i+1} + \frac{6x_{i+2}-2(x_i+x_{i-1}+x_{i+1})}{(x_{i+2}-x_{i-1})(x_{i+2}-x_i)(x_{i+2}-x_{i+1})}y_{i+2}$$

The listing of the user-defined function `SndDerivUneq` is:

```
function yd = SndDerivUneq(x,y)
n=length(x);
yd(1)=(6*x(1)-2*(x(3)+x(2)+x(4)))*y(1)/((x(1)-x(2))*(x(1)-x(3))*(x(1)-
x(4)))...
    +(4*x(1)-2*(x(3)+x(4)))*y(2)/((x(2)-x(1))*(x(2)-x(3))*(x(2)-x(4)))...
    +(4*x(1)-2*(x(2)+x(4)))*y(3)/((x(3)-x(1))*(x(3)-x(2))*(x(3)-x(4)))...
    +(4*x(1)-2*(x(2)+x(3)))*y(4)/((x(4)-x(1))*(x(4)-x(2))*(x(4)-x(3)));
for j=2:n-2
    yd(j)=(4*x(j)-2*(x(j+1)+x(j+2)))*y(j-1)/((x(j-1)-x(j))*(x(j-1)-
x(j+1))*(x(j-1)-x(j+2)))...
    +(6*x(j)-2*(x(j+1)+x(j-1)+x(j+2)))*y(j)/((x(j)-x(j-1))*(x(j)-
x(j+1))*(x(j)-x(j+2)))...
    +(4*x(j)-2*(x(j-1)+x(j+2)))*y(j+1)/((x(j+1)-x(j-1))*(x(j+1)-x(j))*(x(j+1)-
x(j+2)))...
    +(4*x(j)-2*(x(j-1)+x(j+1)))*y(j+2)/((x(j+2)-x(j-1))*(x(j+2)-x(j))*(x(j+2)-
x(j+1)));
end
yd(n-1)=(4*x(n-1)-2*(x(n-2)+x(n)))*y(n-3)/((x(n-3)-x(n-2))*(x(n-3)-x(n-
1))*(x(n-3)-x(n)))...
    +(4*x(n-1)-2*(x(n-3)+x(n)))*y(n-2)/((x(n-2)-x(n-3))*(x(n-2)-x(n-1))*(x(n-
2)-x(n)))...
    +(6*x(n-1)-2*(x(n-2)+x(n-3)+x(n)))*y(n-1)/((x(n-1)-x(n-3))*(x(n-1)-x(n-
2))*(x(n-1)-x(n)))...
```

```
    +(4*x(n-1)-2*(x(n-2)+x(n-3)))*y(n)/((x(n)-x(n-3))*(x(n)-x(n-2))*(x(n)-x(n-
1)));
yd(n)=(4*x(n)-2*(x(n-1)+x(n-2)))*y(n-3)/((x(n-3)-x(n-2))*(x(n-3)-x(n-1))*(x(n-
3)-x(n)))...
    +(4*x(n)-2*(x(n-1)+x(n-3)))*y(n-2)/((x(n-2)-x(n-3))*(x(n-2)-x(n-1))*(x(n-
2)-x(n)))...
    +(4*x(n)-2*(x(n-2)+x(n-3)))*y(n-1)/((x(n-1)-x(n-3))*(x(n-1)-x(n-2))*(x(n-
1)-x(n)))...
    +(6*x(n)-2*(x(n-2)+x(n-3)+x(n-1)))*y(n)/((x(n)-x(n-3))*(x(n)-x(n-
2))*(x(n)-x(n-1)));
```

The user-defined function `SndDerivUneq` is next used in the script file for calculating the derivative of function given in the problem statement:

```
% HW6_23se Script
clear, clc
x=[-1 -0.6 -0.3 0 0.5 0.8 1.6 2.5 2.8 3.2 3.5 4];
Fx=[-3.632 -0.8912 0.3808 1.0 0.6487 -0.3345 -5.287 -12.82 -14.92 -16.43 -15.88
-9.402];
ddFx=SndDerivUneq(x,Fx)
```

When the script is executed the following results are sisplayed in the Command Window.

```
ddFx =
   -7.6933    -7.4419    -7.2533    -7.0037    -6.2943    -5.9801    -2.8063
4.8824    8.5333    17.0057    25.8429    40.5714
```

**6.24** Write a MATLAB user-defined function that evaluates the partial first derivatives $\frac{df}{dx}$ and $\frac{df}{dy}$ of a function $f(x, y)$ that is specified by discrete tabulated points with equal spacing. Use two-point central difference formulas at the interior points and one-sided three-point forward and backward difference formulas at the endpoints. For the function name use `[dfdx,dfdy]= ParDer(x,y,f)`. The input arguments `x` and `y` are vectors with the values of the independent variables. `f` is a vector with the value of $f$ at each point. The output arguments `dfdx` and `dfdy` are vectors with the values of the partial derivatives at each point. Use `ParDer` to calculate the partial derivatives with respect to $x$ and $y$ of the function given in Problem 6.13.

## Solution

Since the points are equally spaced, we can use the formulae from Table 6-1 in each coordinate direction. For the interior points, the two-point central difference formulae are:

$$\left.\frac{\partial f}{\partial x}\right|_{(x_i,y_0)} = \frac{f(x_{i+1},y_0)-f(x_{i-1},y_0)}{2(\Delta x)}$$

and $$\left.\frac{\partial f}{\partial y}\right|_{(x_0,y_i)} = \frac{f(x_0,y_{i+1})-f(x_0,y_{i-1})}{2(\Delta y)}$$

For the boundary points, the one-sided three-point difference formulae are:

$$\left.\frac{\partial f}{\partial x}\right|_{(x_1,y_i)} = \frac{-3f(x_1,y_i)+4f(x_2,y_i)-f(x_3,y_i)}{2(\Delta x)}$$

$$\left.\frac{\partial f}{\partial x}\right|_{(x_n,y_i)} = \frac{f(x_{n-2},y_i)-4f(x_{n-1},y_i)+3f(x_n,y_i)}{2(\Delta x)}$$

$$\left.\frac{\partial f}{\partial y}\right|_{(x_i,y_1)} = \frac{-3f(x_i,y_1)+4f(x_i,y_2)-f(x_i,y_3)}{2(\Delta y)}$$

and $$\left.\frac{\partial f}{\partial y}\right|_{(x_i,y_n)} = \frac{f(x_i,y_{n-2})-4f(x_i,y_{n-1})+3f(x_i,y_n)}{2(\Delta y)}$$

The following function script calculates the partial derivatives according to the above formulae:

```
function [dfdx,dfdy]=ParDer(x,y,f)
n=length(x); deltax=x(2)-x(1); deltay=y(2)-y(1);
m=length(y);
% use two-point central differences for interior points:
for i=2:n-1
    for j=1:m
    dfdx(i,j)=(f(i+1,j)-f(i-1,j))/(2*deltax);
    end
end
```

```
for j=2:m-1
    for i=1:n
        dfdy(i,j)=(f(i,j+1)-f(i,j-1))/(2*deltay);
    end
end
% use one-sided three-point forward and bacward differences at the end
% points:
for j=1:m
    dfdx(1,j)=(-3*f(1,j)+4*f(2,j)-f(3,j))/(2*deltax);
    dfdx(n,j)=(3*f(n,j)-4*f(n-1,j)+f(n-2,j))/(2*deltax);
end
for i=1:n
    dfdy(i,1)=(-3*f(i,1)+4*f(i,2)-f(i,3))/(2*deltay);
    dfdy(i,m)=(3*f(i,m)-4*f(i,m-1)+f(i,m-2))/(2*deltay);
end
```

When executed for the function specified in Problem (6.13), the following output is produced:

```
>> x=[0 1 2 3 4];
>> y=[0 1 2 3 4];
>> f=[0 3 14 7 5
8 10 14 12 10
2 7 8 9 7
13 15 22 16 9
15 18 22 17 14];
>>
>> [dfdx,dfdy]=ParDer(x,y,f)
dfdx =
   15.0000   12.0000    3.0000    9.0000    9.0000
    1.0000    2.0000   -3.0000    1.0000    1.0000
    2.5000    2.5000    4.0000    2.0000   -0.5000
    6.5000    5.5000    7.0000    4.0000    3.5000
   -2.5000    0.5000   -7.0000   -2.0000    6.5000
dfdy =
   -1.0000    7.0000    2.0000   -4.5000    0.5000
    1.0000    3.0000    1.0000   -2.0000   -2.0000
    7.0000    3.0000    1.0000   -0.5000   -3.5000
   -0.5000    4.5000    0.5000   -6.5000   -7.5000
    2.5000    3.5000   -0.5000   -4.0000   -2.0000
```

Note that in the MATLAB script, i is the index used for the variable x and j is the index used for the variable y. Consequently, the transpose of what is given for the function values in the table in Problem (6.13) is what has to be input to the program.

**6.25** Write a MATLAB user-defined function that evaluates the partial second derivatives $\frac{d^2f}{dx^2}$ and $\frac{d^2f}{dy^2}$ of a function $f(x, y)$ that is specified by discrete tabulated points with equal spacing. Use three-point central difference formulas for the interior points and one-sided four-point forward and backward difference formulas for the end points. For the function name use `[dfdx2,dfdy2]= ParDerSnd(x,y,f)`. The input arguments `x` and `y` are vectors with the values of the independent variables. `f` is a vector with the value of $f$ at each point. The output arguments `dfdx2` and `dfdy2` are vectors with the values of the partial second derivatives at each point. Use `ParDerSnd` to calculate the partial second derivatives with respect to $x$ and $y$ of the function given in Problem 6.13.

**Solution**

Since the points are equally spaced, we can use the formulae from Table 6-1 in each coordinate direction. For the interior points, the three-point central difference formulae are:

$$\left.\frac{\partial^2 f}{\partial x^2}\right|_{(x_i,y_0)} = \frac{f(x_{i-1},y_0)-2f(x_i,y_0)+f(x_{i+1},y_0)}{(\Delta x)^2}$$

and
$$\left.\frac{\partial^2 f}{\partial y^2}\right|_{(x_0,y_i)} = \frac{f(x_0,y_{i-1})-2f(x_0,y_i)+f(x_0,y_{i+1})}{(\Delta y)^2}$$

For the boundary points, the one-sided four-point difference formulae are:

$$\left.\frac{\partial^2 f}{\partial x^2}\right|_{(x_1,y_i)} = \frac{2f(x_1,y_i)-5f(x_2,y_i)+4f(x_3,y_i)-f(x_4,y_i)}{(\Delta x)^2}$$

$$\left.\frac{\partial^2 f}{\partial x^2}\right|_{(x_n,y_i)} = \frac{-f(x_{n-3},y_i)+4f(x_{n-2},y_i)-5f(x_{n-1},y_i)+2f(x_n,y_i)}{(\Delta x)^2}$$

$$\left.\frac{\partial^2 f}{\partial y^2}\right|_{(x_i,y_1)} = \frac{2f(x_i,y_1)-5f(x_i,y_2)+4f(x_i,y_3)-f(x_i,y_4)}{(\Delta y)^2}$$

and
$$\left.\frac{\partial^2 f}{\partial y^2}\right|_{(x_i,y_n)} = \frac{-f(x_i,y_{n-3})+4f(x_i,y_{n-2})-5f(x_i,y_{n-1})+2f(x_i,y_n)}{(\Delta y)^2}$$

The following function script calculates the partial derivatives according to the above formulae:

```
function [dfdx2,dfdy2]=ParDerSnd(x,y,f)
n=length(x); deltax=x(2)-x(1); deltay=y(2)-y(1);
m=length(y);
% use three-point central differences for interior points:
for i=2:n-1
```

```
    for j=1:m
    dfdx2(i,j)=(f(i+1,j)-2*f(i,j)+f(i-1,j))/(deltax^2);
    end
end
for j=2:m-1
    for i=1:n
        dfdy2(i,j)=(f(i,j+1)-2*f(i,j)+f(i,j-1))/(deltay^2);
    end
end
% use one-sided four-point forward and bacward differences at the end
% points:
for j=1:m
    dfdx2(1,j)=(2*f(1,j)-5*f(2,j)+4*f(3,j)-f(4,j))/(deltax^2);
    dfdx2(n,j)=(2*f(n,j)-5*f(n-1,j)+4*f(n-2,j)-f(n-3,j))/(deltax^2);
end
for i=1:n
    dfdy2(i,1)=(2*f(i,1)-5*f(i,2)+4*f(i,3)-f(i,4))/(deltay^2);
    dfdy2(i,m)=(2*f(i,m)-5*f(i,m-1)+4*f(i,m-2)-f(i,m-3))/(deltay^2);
end
```

When executed for the function specified in Problem (6.13), the following output is produced:

```
>> format compact
>> x=[0 1 2 3 4];
>> y=[0 1 2 3 4];
>> f=[0 3 14 7 5
8 10 14 12 10
2 7 8 9 7
13 15 22 16 9
15 18 22 17 14];
>> [dfdx2,dfdy2]=ParDerSnd(x,y,f)
dfdx2 =
   -45   -31   -32   -26   -21
   -14   -10    -6    -8    -8
    17    11    20    10     5
    -9    -5   -14    -6     3
   -35   -21   -48   -22     1
dfdy2 =
    34     8   -18     5    28
    10     2    -6     0     6
    -8    -4     0    -3    -6
    23     5   -13    -1    11
    11     1    -9     2    13
>>
```

Note that in the MATLAB script, i is the index used for the variable x and j is the index used for the variable y. Consequently, the transpose of what is given for the function values in the table in Problem (6.13) is what has to be input to the program.

**6.26** The following data is obtained for the velocity of a vehicle during a crash test.:

| $t$ (ms) | 0 | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 |
|---|---|---|---|---|---|---|---|---|---|
| $v$ (mph) | 30 | 29 | 27 | 24 | 18 | 12 | 5 | 1 | 0 |

If the vehicle weight is 2,000 lb, determine the instantaneous force $F$ acting on the vehicle during the crash.
The force can be calculated by $F = m\frac{dv}{dt}$, and the mass of the car $m$ is $2000/32.2$ slug.
Note that 1 ms = $10^{-3}$ s and 1 mile = 5,280 ft.
(*a*) Solve by using the user-defined function `FirstDeriv` that was written in Problem 6.16.
(*b*) Solve by the using MATLAB built-in function `diff`.

**Solution**

The two parts are solved in the following script file:

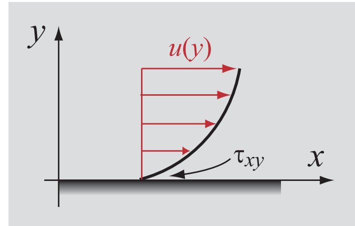
```
% HW 6_26se Script
clear, clc,
m=2000/32.2;
t=[0:10:80]*1E-3;  % Time in s
v=[30 29 27 24 18 12 5 1 0]*5280/3600; % Velocity in ft/s
% Part a
disp('Part (a)')
Fa=m*FirstDeriv(t,v)
% Part b
h=0.01;
disp('Part (b)')
Fb=m*diff(v)/h
```

When the script is executed, the following results are displayed in the Command Window:

```
Part (a)
Fa =
  1.0e+004 *
   -0.4555    -1.3665    -2.2774    -4.0994    -5.4658    -5.9213    -5.0104
-2.2774    0.4555
```

```
Part (b)
Fb =
  1.0e+004 *
   -0.9110   -1.8219   -2.7329   -5.4658   -5.4658   -6.3768   -3.6439
-0.9110
```

**6.27** The distribution of the $x$-component of the velocity $u$ of a fluid near a flat surface is measured as a function of the distance $y$ from the surface:




| $y$ (m) | 0 | 0.002 | 0.004 | 0.006 | 0.008 |
|---|---|---|---|---|---|
| $u$ (m/s) | 0 | 0.005 | 0.008 | 0.017 | 0.022 |

The shear stress $\tau_{yx}$ in the fluid is described by Newton's equation:

$$\tau_{yx} = \mu\frac{\partial u}{\partial y}$$

where $\mu$ is the coefficient of dynamic viscosity. The viscosity can be thought of as a measure of the internal friction within the fluid. Fluids that obey Newton's constitutive equation are called Newtonian fluids. Calculate the shear stress at $y = 0$ using ($i$) the two-point forward, and ($ii$) the three-point forward approximations for the derivative. Take $\mu = 0.00516$ N-s/m$^2$.

**Solution**

($i$) The two-point forward difference formula applied at $y = 0$ yields:

$$\left.\frac{\partial u}{\partial y}\right|_{y=0} = \frac{u(0.002) - u(0)}{0.002} = \frac{0.005 - 0}{0.002} = 2.5\ \text{s}^{-1}$$

The shear stress at $y = 0$ is then given by $\left.\tau_{yx}\right|_{y=0} = (0.002)(2.5) = 0.005\ \text{N}/m^2$ .

($ii$) The three-point forward difference formula applied at $y = 0$ yields:

$$\left.\frac{\partial u}{\partial y}\right|_{y=0} = \frac{-3u(0) + 4u(0.002) - u(0.004)}{2(0.002)} = \frac{-3(0) + 4(0.005) - 0.008}{0.004} = 3.0\ \text{s}^{-1}$$

The shear stress at $y = 0$ is then given by $\left.\tau_{yx}\right|_{y=0} = (0.002)(3.0) = 0.006\ \text{N}/m^2$ . It is to be expected that this answer is more accurate than the value obtained in part ($i$) since it is obtained from a higher order accurate formula.

**6.28** The refractive index $n$ (how much the speed of light is reduced) of fused silica at different wavelengths $\lambda$ is displayed in the table.

| $\lambda$ ($\mu$m) | 0.2 | 0.25 | 0.3 | 0.36 | 0.45 | 0.6 | 1.0 | 1.6 | 2.2 | 3.37 |
|---|---|---|---|---|---|---|---|---|---|---|
| $n$ | 1.551 | 1.507 | 1.488 | 1.475 | 1.466 | 1.458 | 1.450 | 1.443 | 1.435 | 1.410 |

Use the data to calculate the dispersion (spreading of light beam) defined by $\frac{dn}{d\lambda}$ at each wavelength.

(*a*) Use the user-defined function `FirstDerivUneq` written in Problem 6.22.

(*b*) Use MATLAB's built-in function `diff`.

**Solution**

The two parts are solved in the following script file:

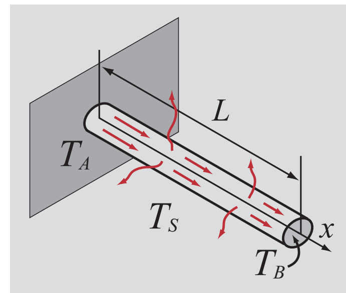
```
clear, clc,
L=[0.2 0.25 0.3 0.36 0.45 0.6 1.0 1.6 2.2 3.37]*1E-6; % Time in s.
n=[1.551 1.507 1.488 1.475 1.466 1.458 1.45 1.443 1.435 1.41];
% Part a
disp('Part (a)')
Disa=FirstDerivUneq(L,n)
% Part b
disp('Part (b)')
Disb=diff(n)./diff(L)
```

When the script is executed, the following results are displayed in the Command Window:

```
Part (a)
Disa =
  1.0e+006 *
   -1.1300   -0.6300   -0.3058   -0.1700   -0.0825   -0.0442   -0.0167
-0.0125   -0.0161   -0.0267
Part (b)
Disb =
  1.0e+005 *
   -8.8000   -3.8000   -2.1667   -1.0000   -0.5333   -0.2000   -0.1167
-0.1333   -0.2137
```

**6.29** A fin is an extended surface used to transfer heat from a base material (at $x = 0$) to an ambient. Heat flows from the base material through the base of the fin, through its outer surface, and through the tip. Measurement of the temperature distribution along a pin fin gives the following data:

| $x$ (cm) | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $T$ (K) | 473 | 446.3 | 422.6 | 401.2 | 382 | 364.3 | 348.0 | 332.7 | 318.1 | 304.0 | 290.1 |




The fin has a length $L = 10$ cm, constant cross-sectional area of $1.6 \times 10^{-5}$ m$^2$, and thermal conductivity $k = 240$ W/m/K. The heat flux (W/m$^2$) is given by $q_x = -k\frac{dT}{dx}$

(*a*) Determine the heat flux at $x = 0$. Use the three-point forward difference formula for calculating the derivative.

(*b*) Determine the heat flux at $x = L$. Use the three-point backward difference formula for calculating the derivative.

(*c*) Determine the amount of heat (in W) lost between $x = 0$ and $x = L$. (The heat flow per unit time in Watts is the heat flux multiplied by the cross-sectional area of the fin.)

**Solution**

(*a*) The three point forward difference formula for the derivative is:

$$f'(x_i) = \frac{-3f(x_i) + 4f(x_{i+1}) - f(x_{i+2})}{2h}$$

In the present problem $h = 0.01$m. Applying the formula at $x = 0$ yields:

$$\left.\frac{dT}{dx}\right|_{x=0} = \frac{-3T(0) + 4T(1) - T(2)}{2(0.01)} = \frac{-3(473) + 4(446.3) - 422.6}{0.02} = -2820 \text{ K/m}$$

Therefore, the heat flux at $x = 0$ is: $\left.q_x\right|_{x=0} = -k\left.\frac{dT}{dx}\right|_{x=0} = -(240)(-2820) = 676.8 \text{ kW/m}^2$ .

(*b*) The three point backward difference formula for the derivative is:

$$f'(x_i) = \frac{f(x_{i-2}) - 4f(x_{i-1}) + 3f(x_i)}{2h}$$

Applying the formula at $x = 0.1$ m yields:

$$\left.\frac{dT}{dx}\right|_{x=10} = \frac{T(8) - 4T(9) + 3T(10)}{2(0.01)} = \frac{318.1 - 4(304) + 3(290.1)}{0.02} = -1380 \text{ K/cm}$$

Therefore, the heat flux at $x = 0.1$ m is: $$q_x\big|_{x=10} = -k\frac{dT}{dx}\bigg|_{x=10} = -(240)(-13.8) = 331.2 \text{ kW/m}^2 .$$

($c$) The amount of heat flow at $x = 0$ is from part ($a$), $676.8 \text{ kW/m}^2 \times 1.6 \times 10^{-5} \text{ m}^2 = 10.8288 \text{ W}$. Similarly, the amount of heat flow at $x = 0.1$ is from part ($b$), $331.2 \text{ kW/m}^2 \times 1.6 \times 10^{-5} \text{ m}^2 = 5.2992 \text{ W}$. The amount of heat lost between $x = 0$ and $x = 10 \text{ cm}$ is $10.8288 \text{ W} - 5.2992 \text{ W} = 5.5296 \text{ W}$.

**6.30** The altitude of the space shuttle during the first two minutes of its ascent is displayed in the following table (www.nasa.gov):

| *t* (s) | 0 | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 | 100 | 110 | 120 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *h* (m) | -8 | 241 | 1,244 | 2,872 | 5,377 | 8,130 | 11,617 | 15,380 | 19,872 | 25,608 | 31,412 | 38,309 | 44,726 |

Assuming the shuttle is moving straight up, determine its velocity and acceleration at each point. Display the results in three plots (*h* versus time, velocity versus time, and acceleration versus time).
(*a*) Solve by using the user-defined function `FrstScndDerivPt` that was written in Problem 6.18 (*b*).
(*b*) Solve by using the MATLAB built-in function `diff`.

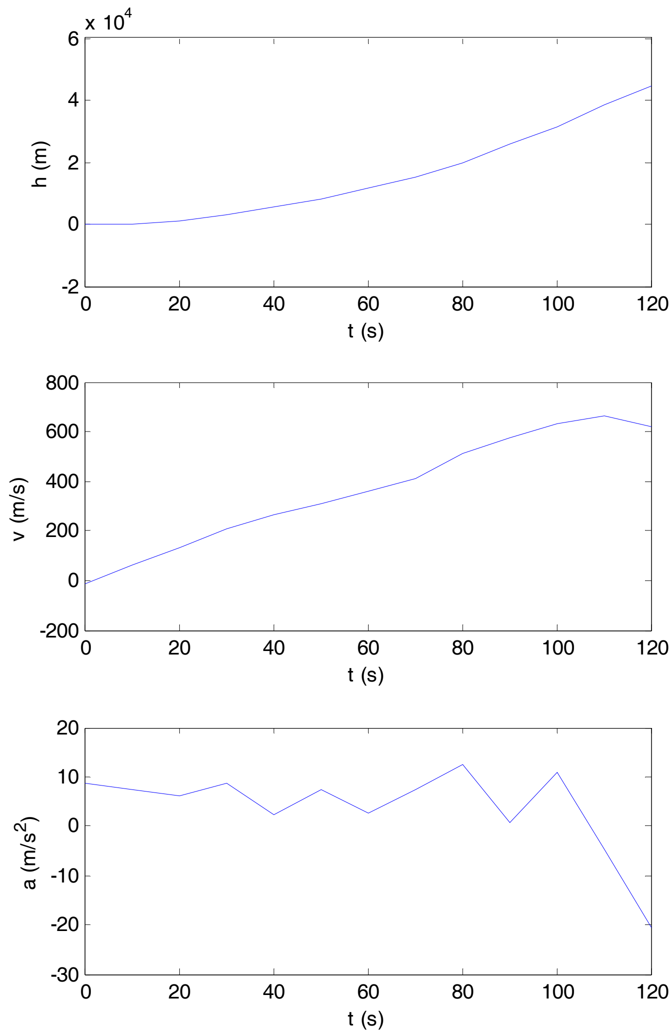
**Solution**

(*a*) The following scripts file uses the user-defined function `FrstScndDerivPt` that was written in Problem 6.18 (*b*) to solve the problem. (The user-defined function `FrstScndDerivPt` was modified such that the axes labels correspond to the variables in the current problem.)

```
clear, clc,
t=0:10:120;
h=[-8 241 1244 2872 5377 8130 11617 15380 19872 25608 31412 38309 44726];
[yd,ydd] = FrstScndDerivPt(t,h)
```

When the script is executed, the following results are displayed in the Command Window, and the following figure is displayed.

```
yd =
  -12.8000   62.6000  131.5500  206.6500  262.9000  312.0000  362.5000
412.7500  511.4000  577.0000  635.0500  665.7000  617.7000
ydd =
    8.8300    7.5400    6.2500    8.7700    2.4800    7.3400    2.7600
7.2900   12.4400    0.6800   10.9300   -4.8000  -20.5300
```

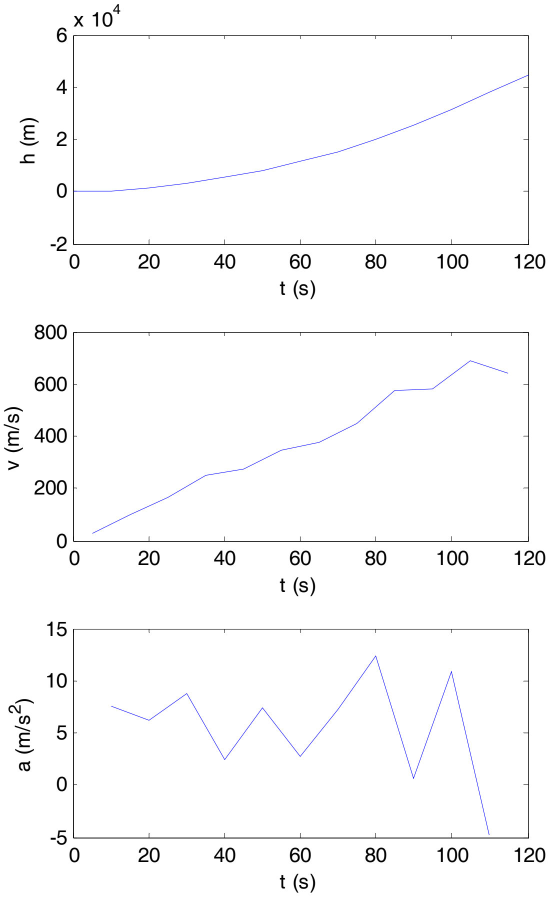



(*b*) The following scripts file uses MATLAB built-in function `diff` to solve the problem.

```
clear, clc,
t=0:10:120;
h=[-8 241 1244 2872 5377 8130 11617 15380 19872 25608 31412 38309 44726];
dt=diff(t);
v=diff(h)./dt;
ndt=length(dt);
for i=1:ndt
    tv(i)=t(i)+dt(i)/2;
end
ddt=diff(tv);
a=diff(v)./ddt;
nddt=length(ddt);
for i=1:nddt
    ta(i)=tv(i)+ddt(i)/2;
end
subplot(3,1,1)
plot(t,h); xlabel('t (s)'); ylabel('h (m)');
subplot(3,1,2)
plot(tv,v); xlabel('t (s)'); ylabel('v (m/s)');
subplot(3,1,3)
plot(ta,a); xlabel('t (s)'); ylabel('a (m/s^2)');
```

When the script is executed, the following results are displayed in the Command Window, and the following figure is displayed.

```
v =
   24.9000   100.3000   162.8000   250.5000   275.3000   348.7000   376.3000
449.2000   573.6000   580.4000   689.7000   641.7000
a =
    7.5400     6.2500     8.7700     2.4800     7.3400     2.7600     7.2900
12.4400     0.6800    10.9300    -4.8000
```


x 10^4
h (m)
t (s)
v (m/s)
t (s)
a (m/s^2)
t (s)

**6.31** A small rocket is fired vertically upward. Its altitude as a function of time is measured and displayed in the following table:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $t$ (s) | 0 | 2 | 4 | 6 | 8 | 10 | 12 | 14 |
| $h$ (ft) | 0 | 80 | 350 | 800 | 1,400 | 2,100 | 3,100 | 4,100 |
| $t$ (s) | 16 | 18 | 20 | 22 | 24 | 26 | 28 | 30 |
| $h$ (ft) | 5,300 | 6,500 | 7,900 | 9,300 | 10,700 | 12,200 | 13,800 | 15,300 |

Write a MATLAB program in a script file that first determines the rocket's velocity ($v = \frac{dh}{dt}$) and then the acceleration ($a = \frac{dv}{dt}$) at each point. Display the results in three plots ($h$ versus time, velocity versus time, and acceleration versus time).
($a$) Solve by using the user-defined function `FirstDeriv` that was written in Problem 6.16.
($b$) Solve by using the MATLAB built-in function `diff`.

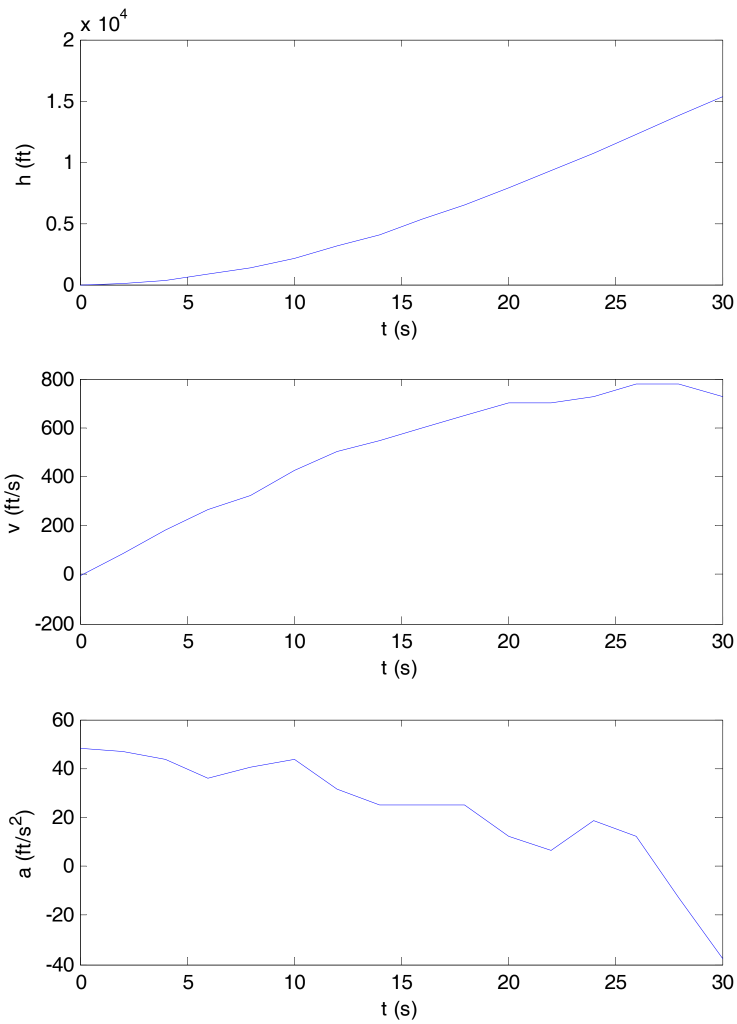
**Solution**

($a$) The following scripts file uses the user-defined function `FirstDeriv` that was written in Problem 6.18 ($b$) to solve the problem.

```
clear, clc,
t=0:2:30;
h=[0 80 350 800 1400 2100 3100 4100 5300 6500 7900 9300 10700 12200 13800
15300];
v=FirstDeriv(t,h)
a=FirstDeriv(t,v)
subplot(3,1,1)
plot(t,h); xlabel('t (s)'); ylabel('h (m)');
subplot(3,1,2)
plot(t,v); xlabel('t (s)'); ylabel('v (m/s)');
subplot(3,1,3)
plot(t,a); xlabel('t (s)'); ylabel('a (m/s^2)');
```

When the script is executed, the following results are displayed in the Command Window, and the figure that follows is displayed.

```
v =
   -7.5000   87.5000  180.0000  262.5000  325.0000  425.0000  500.0000
550.0000   600.0000   650.0000   700.0000   700.0000   725.0000   775.0000
775.0000  725.0000
a =
   48.1250   46.8750   43.7500   36.2500   40.6250   43.7500   31.2500
25.0000   25.0000   25.0000   12.5000    6.2500   18.7500   12.5000  -
12.5000  -37.5000
```



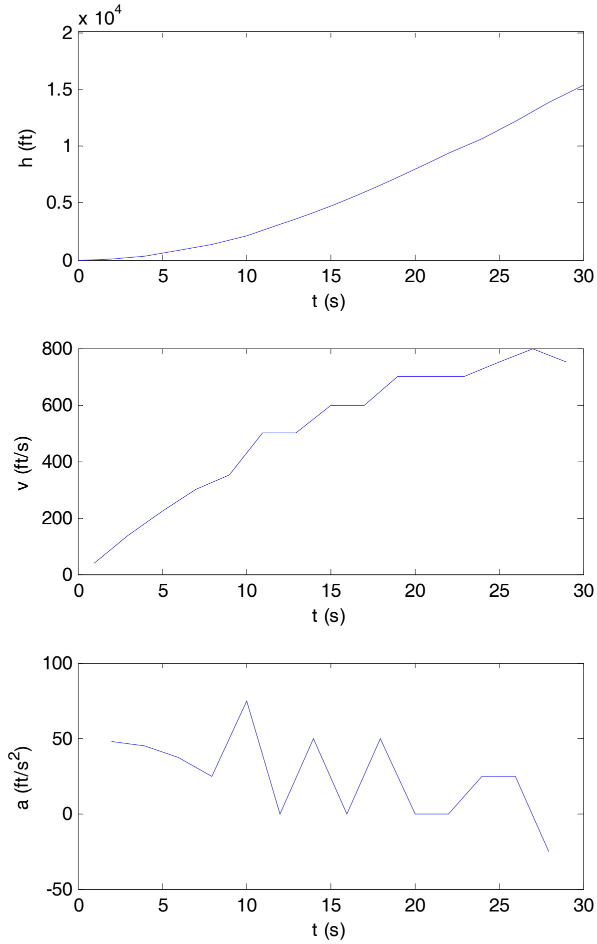

(*b*) The following scripts file uses MATLAB built-in function `diff` to solve the problem.

```
clear, clc,
```

```
t=0:2:30;
h=[0 80 350 800 1400 2100 3100 4100 5300 6500 7900 9300 10700 12200 13800
15300];
dt=diff(t);
v=diff(h)./dt
ndt=length(dt);
for i=1:ndt
    tv(i)=t(i)+dt(i)/2;
end
ddt=diff(tv);
a=diff(v)./ddt
nddt=length(ddt);
for i=1:nddt
    ta(i)=tv(i)+ddt(i)/2;
end
subplot(3,1,1)
plot(t,h); xlabel('t (s)'); ylabel('h (ft)');
subplot(3,1,2)
plot(tv,v); xlabel('t (s)'); ylabel('v (ft/s)');
subplot(3,1,3)
plot(ta,a); xlabel('t (s)'); ylabel('a (ft/s^2)');
```

When the script is executed, the following results are displayed in the Command Window, and the following figure is displayed.

```
v =
    40    135    225    300    350    500    500    600    600    700    700    700
750    800    750
a =
   47.5000    45.0000    37.5000    25.0000    75.0000          0    50.0000
0    50.0000          0          0    25.0000    25.0000   -25.0000
```


x 10^4
h (ft)
t (s)
v (ft/s)
t (s)
a (ft/s^2)
t (s)

**6.32** Use the user-defined function `FrstScndDeriv` written in Problem 6.18 to calculate the velocity and the acceleration of the rocket in Problem 6.30. Display the results in three plots (*h* versus time, velocity versus time, and acceleration versus time).

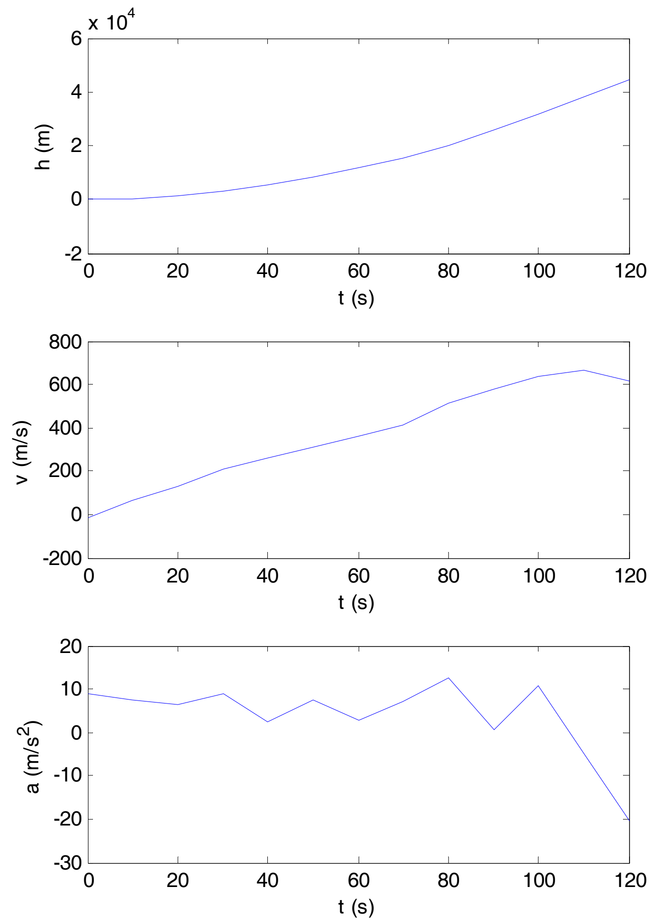
## Solution

The following scripts file uses the user-defined function `FrstScndDeriv` that was written in Problem 6.18 to solve the problem.

```
clear, clc,
t=0:10:120;
h=[-8 241 1244 2872 5377 8130 11617 15380 19872 25608 31412 38309 44726];
[v,a] = FrstScndDeriv(t,h)
subplot(3,1,1)
plot(t,h); xlabel('t (s)'); ylabel('h (m)');
subplot(3,1,2)
plot(t,v); xlabel('t (s)'); ylabel('v (m/s)');
subplot(3,1,3)
plot(t,a); xlabel('t (s)'); ylabel('a (m/s^2)');
```

When the script is executed, the following results are displayed in the Command Window, and the following figure is displayed.

```
v =
  -12.8000   62.6000  131.5500  206.6500  262.9000  312.0000  362.5000
412.7500  511.4000  577.0000  635.0500  665.7000  617.7000
a =
    8.8300    7.5400    6.2500    8.7700    2.4800    7.3400    2.7600
7.2900   12.4400    0.6800   10.9300   -4.8000  -20.5300
```


x 10^4
6
4
2
0
-2
h (m)
0
20
40
60
80
100
120
t (s)
800
600
400
200
0
-200
v (m/s)
0
20
40
60
80
100
120
t (s)
20
10
0
-10
-20
-30
a (m/s^2)
0
20
40
60
80
100
120
t (s)

**6.33** The charge on the capacitor in the RLC circuit shown at various times after the switch is closed at time $t = 0$ is given in the following table. The current, $I$, as a function of time is given by $I(t) = \frac{dQ}{dt}$. Determine the current as a function of time by numerically differentiating the data.
(*a*) Use the user-defined function `FirstDerivUneq` that was written in Problem 6.22.
(*b*) Use the MATLAB built-in function `diff`.
In both parts plot *I* vs. *t*.

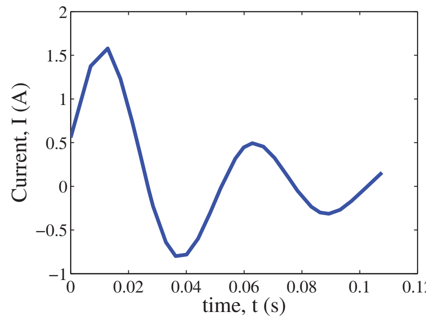
| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $Q \times 10^3$ (C) | 0 | 6.67 | 16.93 | 23.38 | 27.44 | 29.02 | 29.5 | 29.29 | 27.17 | 24.58 | 21.43 | 18.56 | 16.61 | 16.01 |
| $t$ (ms) | 0 | 6.9 | 12.84 | 17.23 | 21.24 | 24.05 | 27.01 | 28.49 | 32.95 | 36.33 | 40.1 | 44.06 | 48.21 | 51.94 |
| $Q \times 10^3$ (C) | 16.87 | 18.06 | 19.51 | 21.42 | 22.99 | 24.05 | 24.12 | 23.35 | 22.5 | 21.53 | 20.33 | 19.44 | 18.98 | 19.36 |
| $t$ (ms) | 56.94 | 59.9 | 62.86 | 66.69 | 70.52 | 75.3 | 78.48 | 83.26 | 86.29 | 89.33 | 93.21 | 97.09 | 101.6 | 107.7 |

**Solution**

(*a*) The user-defined function `FirstDerivLag` is used in the following script file for calculating $I(t) = \frac{dQ}{dt}$, and making the plot.

```
Q=[0 6.67 16.93 23.38 27.44 29.02 29.5 29.29 27.17 24.58 21.43 18.56 16.61 ...
    16.01 16.87 18.06 19.51 21.42 22.99 24.05 24.12 23.35 22.5 21.53 20.33 ...
    19.44 18.98 19.36]*0.001;
t=[0 6.9 12.84 17.23 21.24 24.05 27.01 28.49 32.95 36.33 40.1 44.06 48.21 ...
    51.94 56.94 59.9 62.86 66.69 70.52 75.3 78.48 83.26 86.29 89.33 93.21 ...
    97.09 101.6 107.7]*0.001;
I=FirstDerivLag(t,Q)
plot(t,I)
xlabel('time, t (s)'); ylabel('Current, I (A)');
```

When the program is executed, the following values of $I$ are displayed in the Command Window, and the figure that is shown on the right is displayed in the Figure Window.

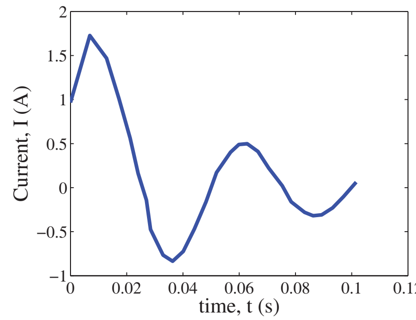



```
I =
  Columns 1 through 7
     0.5579     1.3754     1.5789     1.2305
0.7478    0.3674   -0.0405
  Columns 8 through 14
   -0.2250    -0.6408    -0.7990    -0.7815    -
0.6003   -0.3071    -0.0186
  Columns 15 through 21
    0.3165     0.4459     0.4937     0.4543     0.3262     0.1018    -0.0511
  Columns 22 through 28
   -0.2342    -0.2998    -0.3148    -0.2693    -0.1705    -0.0322     0.1568
```

(*b*) The `diff` command in MATLAB performs the equivalent of a two-point forward difference. Therefore, it can be used for evaluating the derivative $\frac{dQ}{dt}$ at all points except the last point. The following script file uses the MATLAB built-in function `diff`, to calculate $I(t) = \frac{dQ}{dt}$. The plot does not include the last point since no value for I is calculated for the last point.

```
Q=[0 6.67 16.93 23.38 27.44 29.02 29.5 29.29 27.17 24.58 21.43 18.56 16.61 ...
    16.01 16.87 18.06 19.51 21.42 22.99 24.05 24.12 23.35 22.5 21.53 20.33 ...
    19.44 18.98 19.36]*0.001;
t=[0 6.9 12.84 17.23 21.24 24.05 27.01 28.49 32.95 36.33 40.1 44.06 48.21 ...
    51.94 56.94 59.9 62.86 66.69 70.52 75.3 78.48 83.26 86.29 89.33 93.21 ...
    97.09 101.6 107.7]*0.001;
I = diff(Q)./diff(t)
n=length(Q);
plot(t(1:n-1),I)
xlabel('time, t (s)'); ylabel('Current, I (A)');
```

When the program is executed, the following values of *I* are displayed in the Command Window, and the figure that is shown on the right is displayed in the Figure Window.




```
I =
  Columns 1 through 7
     0.9667     1.7273     1.4692     1.0125
0.5623    0.1622   -0.1419
  Columns 8 through 14
   -0.4753   -0.7663   -0.8355   -0.7247   -
0.4699   -0.1609    0.1720
  Columns 15 through 21
    0.4020    0.4899    0.4987    0.4099    0.2218    0.0220   -0.1611
  Columns 22 through 27
   -0.2805   -0.3191   -0.3093   -0.2294   -0.1020    0.0623
```

**\6.34** The following data for mean velocity $\bar{u}$ near the wall in a fully developed turbulent pipe air flow was measured [J. Laufer, "The Structure of Turbulence in Fully Developed Pipe Flow," U.S. National Advisory Committee for Aeronautics (NACA), Technical Report 1174, 1954]:

| $y/R$ | 0.0030 | 0.0034 | 0.0041 | 0.0051 | 0.0055 | 0.0061 | 0.0071 | 0.0075 | 0.0082 |
|---|---|---|---|---|---|---|---|---|---|
| $\bar{u}/U$ | 0.140 | 0.152 | 0.179 | 0.221 | 0.228 | 0.264 | 0.300 | 0.318 | 0.343 |

$y$ is the distance from the wall, $R = 4.86$ in. is the radius of the pipe, and $U = 9.8$ ft/s. Use the data to calculate the shear stress $\tau$ defined by $\tau = \mu\frac{d\bar{u}}{dy} = \mu\frac{U}{R}\frac{d(\bar{u}/U)}{d(y/R)}$. $\mu = 3.8 \times 10^{-7}$ lb-s/ft$^2$ is the dynamic viscosity. Note that $\tau$ will have units of lb/ft$^2$.

(*a*) Use the user-defined function `FirstDerivUneq` written in Problem 6.22.

(*b*) Use MATLAB's built-in function `diff`.

**Solution**

*a*) The following scripts file uses the user-defined function `FirstDerivUneq` that was written in Problem 6.22 to solve the problem.

```
clear, clc
meu=3.8E-7;
yR=[0.003 0.0034 0.0041 0.0051 0.0055 0.0061 0.0071 0.0075 0.0082];
uU=[0.14 0.152 0.179 0.221 0.228 0.264 0.3 0.318 0.343];
Tau=meu*FirstDerivUneq(yR,uU)
```

When the script is executed, the following results are displayed in the Command Window, and the figure that follows is displayed.

```
Tau =
   1.0e-004 *
      0.1022     0.1258     0.1519     0.0931     0.1311     0.1938     0.1612
0.1582     0.1133
```

*b*) The following scripts file uses MATLAB's built-in function `diff` to solve the problem.

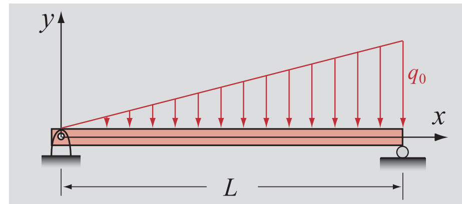
```
clear, clc,
```

```
meu=3.8E-7;
yR=[0.003 0.0034 0.0041 0.0051 0.0055 0.0061 0.0071 0.0075 0.0082];
uU=[0.14 0.152 0.179 0.221 0.228 0.264 0.3 0.318 0.343];
tau=meu*diff(uU)./diff(yR)
```

When the script is executed, the following results are displayed in the Command Window, and the figure that follows is displayed.

```
tau =
  1.0e-004 *
    0.1140    0.1466    0.1596    0.0665    0.2280    0.1368    0.1710
0.1357
```

**6.35** A 10-meter-long uniform beam is simply supported at both ends and is subjected to the triangular load shown. The deflection of the beam is given by the differential equation:




$$\frac{d^2y}{dx^2} = -\frac{M(x)}{EI}$$

where $y$ is the deflection, $x$ is the coordinate measured along the length of the beam, $M(x)$ is the bending moment, and $EI = 1.2 \times 10^7$ N-m$^2$ is the flexural rigidity of the beam. The following data is obtained from measuring the deflection of the beam versus position:

| $x$ (m) | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $y$ (mm) | 0 | -16 | -30.6 | -42.5 | -50.7 | -54.3 | -52.6 | -45.7 | -33.9 | -18.1 | 0 |

Using the data, determine the bending moment $M(x)$ at each location $x$. Solve the problem by using the user-defined function `SecDeriv` written in Problem 6.17.

## Solution

The bending moment $M(x)$ can be solved for from the given ODE:

$$M(x) = -EI\frac{d^2y}{dx^2}$$

The problem is solved in the following script file where the bending moment $M(x)$ at each location $x$ is calculated by using the user-defined function `SecDeriv` written in Problem 6.17

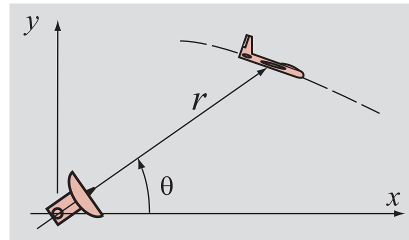
```
EI=1.2E7;
x=0:10;
y=[0 -16 -30.6 -42.5 -50.7 -54.3 -52.6 -45.7 -33.9 -18.1 0]*1E-3;
M=-EI*SecDeriv(x,y)
```

When the script file is executed, the following result is displayed in the Command Window:

```
M =
  1.0e+004 *
   -0.1200   -1.6800   -3.2400   -4.4400   -5.5200   -6.3600   -6.2400
-5.8800   -4.8000   -2.7600   -0.7200
```

Notice that the numerical solution gives small moments at the ends. This is not accurate, since the moments at the ends is zero.

**6.36** A radar station is tracking the motion of an aircraft. The recorded distance to the aircraft, $r$, and the angle $\theta$ during a period of 60 s is given in the following table. The magnitude of the instantaneous velocity and acceleration of the aircraft can be calculated by:




$$v = \sqrt{\left(\frac{dr}{dt}\right)^2 + \left(r\frac{d\theta}{dt}\right)^2} \qquad a = \sqrt{\left[\frac{d^2r}{dt^2} - r\left(\frac{d\theta}{dt}\right)^2\right]^2 + \left[r\frac{d^2\theta}{dt^2} + 2\frac{dr}{dt}\frac{d\theta}{dt}\right]^2}$$

Determine the magnitudes of the velocity and acceleration at the times given in the table. Plot the velocity and acceleration versus time (two separate plots on the same page). Solve the problem by writing a program in a script file. The program evaluates the various derivatives that are required for calculating the velocity and acceleration, and then makes the plots. For calculating the derivatives use

(*a*) the user-defined function `FrstScndDeriv` that was written in Problem 6.18;

(*b*) MATLAB's built-in function `diff`.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| *t* (s) | 0 | 4 | 8 | 12 | 16 | 20 | 24 | 28 |
| *r* (km) | 18.803 | 18.861 | 18.946 | 19.042 | 19.148 | 19.260 | 19.376 | 19.495 |
| θ(rad) | 0.7854 | 0.7792 | 0.7701 | 0.7594 | 0.7477 | 0.7350 | 0.7215 | 0.7073 |
| *t* (s) | 32 | 36 | 40 | 44 | 48 | 52 | 56 | 60 |
| *r* (km) | 19.617 | 19.741 | 19.865 | 19.990 | 20.115 | 20.239 | 20.362 | 20.484 |
| θ(rad) | 0.6925 | 0.6771 | 0.6612 | 0.6448 | 0.6280 | 0.6107 | 0.5931 | 0.5750 |

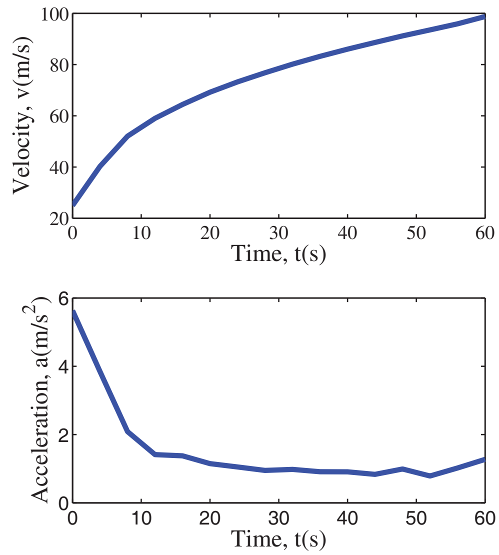
## Solution

(*a*) The following script, in which the user-defined function `FrstScndDeriv` (written in Problem 6.16) is used for calculating the derivatives and solve this problem.

```
t=0:4:60;
r=[18.803 18.861 18.946 19.042 19.148 19.260 19.376 19.495 ...
    19.617 19.741 19.865 19.990 20.115 20.239 20.362 20.484]*1000;
theta=[0.7854 0.7792 0.7701 0.7594 0.7477 0.7350 0.7215 0.7073 ...
    0.6925 0.6771 0.6612 0.6448 0.6280 0.6107 0.5931 0.5750];
[drdt,d2rdt2] = FrstScndDeriv(t,r);
```

```
[d_theta_dt,d2_theta_dt2] = FrstScndDeriv(t,theta);
v=sqrt(drdt.^2+(r.*d_theta_dt).^2)
term1=(d2rdt2-r.*d_theta_dt.^2).^2;
term2=(r.*d2_theta_dt2+2*drdt.*d_theta_dt).^2;
a=sqrt(term1+term2)
subplot(2,1,1)
plot(t,v); xlabel('Time, t(s)'); ylabel('Velocity, v(m/s)');
subplot(2,1,2)
plot(t,a); xlabel('Time, t(s)'); ylabel('Acceleration, a(m/s^2)');
```

When the program is executed, the following results are displayed in the Command Window, and the figure shown on the right in displayed in the Figure Window.

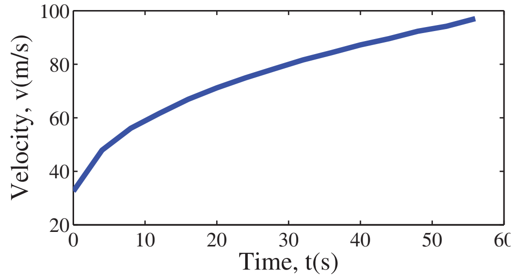


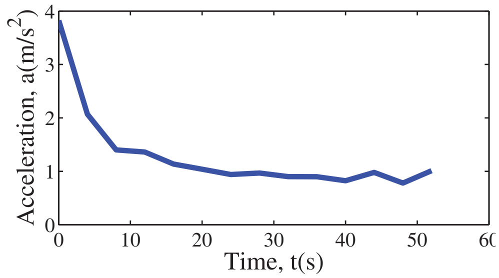

```
v =
  Columns 1 through 8
   24.9465    40.2577    52.0643    58.9943
64.4460    69.2163    73.2385    76.8224
  Columns 9 through 16
   80.1847    83.2256    86.0325    88.6492
91.2148    93.5353    95.8875    98.7576
a =
  Columns 1 through 8
    5.6307     3.8442     2.0864     1.4140
1.3771     1.1505     1.0521     0.9519
  Columns 9 through 16
    0.9801     0.9111     0.9104     0.8335
0.9911     0.7883     1.0230     1.2743
```

(*b*) The following script, in which the MATLAB built-in function `diff` is used for calculating the derivatives solves this problem. Note that when the `diff` function is used for calculating the first derivative by writing `drdt=diff(r)./diff(t)`and `d_theta_dt=diff(theta)./diff(t)` it does not calculate the derivative at the last point. The second derivative is calculated by writing `d2rdt2=diff(r,2)/h/h` and `d2_theta_dt2=diff(theta,2)/h/h`. In this case the second derivative is not calculated in the last two points.

```
clear all
t=0:4:60;
r=[18.803 18.861 18.946 19.042 19.148 19.260 19.376 19.495 ...
    19.617 19.741 19.865 19.990 20.115 20.239 20.362 20.484]*1000;
theta=[0.7854 0.7792 0.7701 0.7594 0.7477 0.7350 0.7215 0.7073 ...
    0.6925 0.6771 0.6612 0.6448 0.6280 0.6107 0.5931 0.5750];
[drdt,d2rdt2] = FrstScndDeriv(t,r);
[d_theta_dt,d2_theta_dt2] = FrstScndDeriv(t,theta);
h=t(2)-t(1);
n=length(t);
drdt=diff(r)./diff(t); d_theta_dt=diff(theta)./diff(t);
d2rdt2=diff(r,2)/h/h; d2_theta_dt2=diff(theta,2)/h/h;
v=sqrt(drdt.^2+(r(1:n-1).*d_theta_dt).^2)
term1=(d2rdt2-r(1:n-2).*d_theta_dt(1:n-2).^2).^2;
term2=(r(1:n-2).*d2_theta_dt2+2*drdt(1:n-2).*d_theta_dt(1:n-2)).^2;
a=sqrt(term1+term2)
subplot(2,1,1)
plot(t(1:n-1),v(1:n-1)); xlabel('Time, t(s)'); ylabel('Velocity, v(m/s)');
subplot(2,1,2)
plot(t(1:n-2),a(1:n-2));  xlabel('Time,  t(s)');  ylabel('Acceleration,  a(m/
s^2)');
```

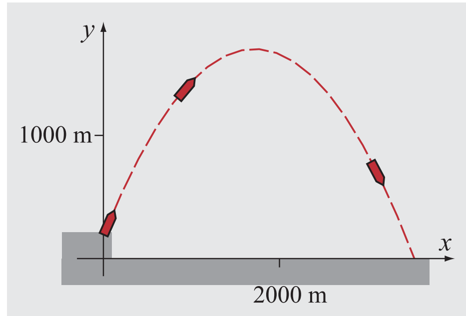
When the program is executed, the following results are displayed in the Command Window, and the figure shown on the right in displayed in the Figure Window.




```
v =
  Columns 1 through 8
   32.5524    47.8824    56.0760    61.6806
66.9330   71.1781   74.9427   78.3148
  Columns 9 through 15
   81.6400    84.3719    87.2359    89.5852
92.3555   94.2112   97.0550
a =
  Columns 1 through 8
    3.8237     2.0687     1.4008     1.3618
1.1366    1.0389    0.9395    0.9672
  Columns 9 through 14
    0.9000     0.8987     0.8236     0.9795
0.7803    1.0109
```

**6.37** A projectile is shot from a 200-m-tall cliff as shown. Its position ($x$ and $y$ coordinates) as a function of time, $t$, is given in the table that follows. The velocity of the projectile, $v$, is given by $v = \sqrt{v_x^2 + v_y^2}$ where the horizontal and vertical components, $v_x$ and $v_y$ are given by $v_x = \frac{dx}{dt}$ and $v_y = \frac{dy}{dt}$. Write a MATLAB program in a script file that




(*a*) determines $v_x$ and $v_y$ by using the user-defined function `FirstDeriv` that was written in Problem 6.16;
(*b*) determines the velocity $v$ of the projectile;
(*c*) displays a figure with plots of $v_x$, $v_y$ and $v$ as a function of time (three plots in one figure).

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| *t* (s) | 0 | 2 | 4 | 6 | 8 | 10 | 12 | 14 | 16 | 18 |
| *x* (m) | 0 | 198 | 395 | 593 | 790 | 988 | 1185 | 1383 | 1580 | 1778 |
| y (m) | 200 | 523 | 806 | 1050 | 1254 | 1420 | 1546 | 1633 | 1681 | 1690 |
| *t* (s) | 20 | 22 | 24 | 26 | 28 | 30 | 32 | 34 | 36 | |
| *x* (m) | 1975 | 2173 | 2370 | 2568 | 2765 | 2963 | 3160 | 3358 | 3555 | |
| y (m) | 1659 | 1589 | 1480 | 1331 | 1144 | 917 | 651 | 345 | 1 | |

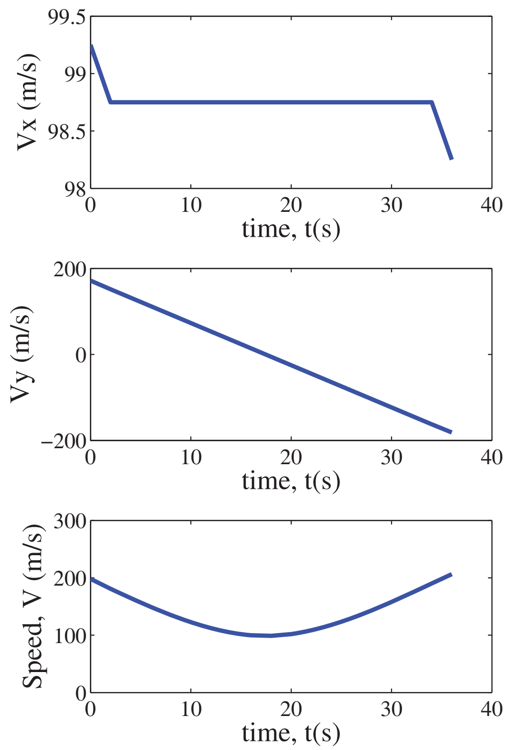
## Solution

The following script, in which the user-defined function `FirstDeriv` (written in Problem 6.14) is used for calculating the derivatives and solve this problem.

```
clear all
t=0:2:36;
x=[0 198 395 593 790 988 1185 1383 1580 1778 1975 2173 2370 ...
    2568 2765 2963 3160 3358 3555];
y=[200 523 806 1050 1254 1420 1546 1633 1681 1690 1659 1589 1480 1331 ...
    1144 917 651 345 1];
vx=FirstDeriv(t,x); vy=FirstDeriv(t,y);
v=sqrt((vx.^2)+(vy.^2));
```

```
subplot(3,1,1)
plot(t,vx); xlabel('time, t(s)'); ylabel('Vx (m/s)');
subplot(3,1,2)
plot(t,vy); xlabel('time, t(s)'); ylabel('Vy (m/s)');
subplot(3,1,3)
plot(t,v); xlabel('time, t(s)'); ylabel('Speed, V (m/s)');
```

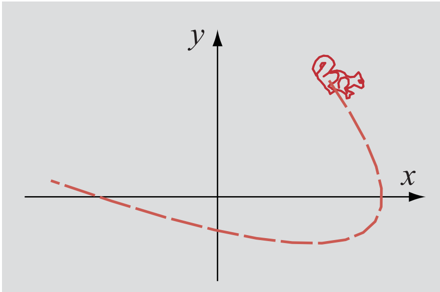
When the program is executed, the following results are displayed in the Command Window, and the figure shown on the right in displayed in the Figure Window.




```
vx =
  Columns 1 through 8
    99.2500    98.7500    98.7500    98.7500
98.7500    98.7500    98.7500    98.7500
  Columns 9 through 16
    98.7500    98.7500    98.7500    98.7500
98.7500    98.7500    98.7500    98.7500
  Columns 17 through 19
   98.7500    98.7500   98.2500
vy =
  Columns 1 through 8
   171.5000   151.5000   131.7500   112.0000
92.5000    73.0000    53.2500    33.7500
  Columns 9 through 16
    14.2500    -5.5000   -25.2500   -44.7500
-64.5000   -84.0000 -103.5000 -123.2500
  Columns 17 through 19
 -143.0000 -162.5000 -181.5000
v =
  Columns 1 through 8
   198.1485   180.8420   164.6500   149.3170   135.3064   122.8029   112.1924
104.3582
  Columns 9 through 16
    99.7729    98.9030   101.9271   108.4164   117.9483   129.6440   143.0518
157.9308
```

```
Columns 17 through 19
173.7831  190.1521  206.3863
```

**6.38** The position of a squirrel ($x$ and $y$ coordinates) running around as a function of time, $t$, is given in the table that follows. The velocity of the squirrel, $v$, is given by $v = \sqrt{v_x^2 + v_y^2}$, where $v_x = \frac{dx}{dt}$ and $v_y = \frac{dy}{dt}$. The acceleration of the squirrel, $a$, is given by $a = \sqrt{a_x^2 + a_y^2}$, where $a_x = \frac{d^2x}{dt^2}$ and $a_y = \frac{d^2y}{\partial t^2}$. Write a MATLAB program in a script file that




(*a*) determines $v$ and $a$ by using the user-defined function `FrstScndDeriv` that was written in Problem 6.18;
(*b*) displays a figure with plots of $v_x$, $v_y$ and $v$ as a function of time (three plots in one figure);
(*c*) displays a second figure with plots of $a_x$, $a_y$ and $a$ as a function of time (three plots in one figure).

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| *t* (s) | 0 | 2 | 4 | 6 | 8 | 10 | 12 | 14 |
| *x* (m) | 61 | 72.8 | 81.9 | 87.9 | 90.9 | 90.8 | 87.3 | 80.5 |
| y (m) | 65 | 46.7 | 30.3 | 15.8 | 3.2 | -7.4 | -15.8 | -22.1 |
| *t* (s) | 16 | 18 | 20 | 22 | 24 | 26 | 28 | 30 |
| *x* (m) | 70.4 | 56.9 | 39.9 | 19.4 | -4.6 | -32.2 | -63.3 | -98 |
| y (m) | -26.2 | -28.1 | -27.9 | -25.3 | -20.5 | -13.4 | -4.1 | 7.6 |

**Solution**

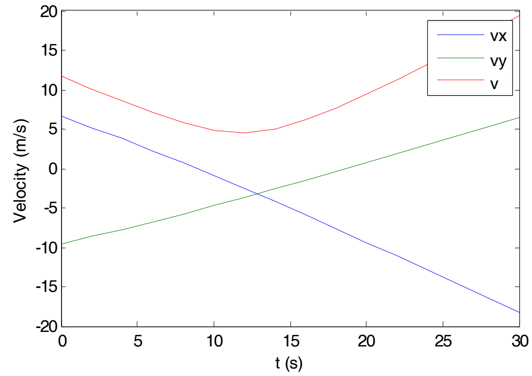
The problem is solved in the following script file:

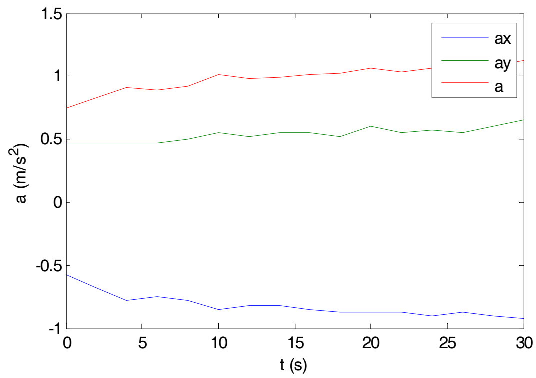
```
clear
clc
t=0:2:30;
x=[61 72.8 81.9 87.9 90.9 90.8 87.3 80.5 70.4 56.9 39.9 19.4 -4.6 -32.2 -63.3 -
98];
y=[65 46.7 30.3 15.8 3.2 -7.4 -15.8 -22.1 -26.2 -28.1 -27.9 -25.3 -20.5 -13.4 -
4.1 7.6];
[vx,ax] = FrstScndDeriv(t,x);
[vy,ay] = FrstScndDeriv(t,y);
```

```
v=sqrt(vx.^2+vy.^2)
a=sqrt(ax.^2+ay.^2)
subplot(2,1,1)
plot(t,vx,t,vy,t,v);
xlabel('t (s)'); ylabel('Velocity (m/s)');
legend('vx','vy','v')
subplot(2,1,2)
plot(t,ax,t,ay,t,a);
xlabel('t (s)'); ylabel('a (m/s^2)');
legend('ax','ay','a')
```

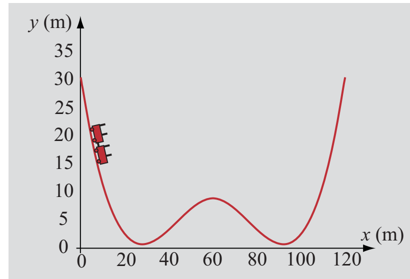
When the script is executed, the following results are displayed in the Command Window, and the figures that follow are displayed in the figure window.

```
v =
   11.6564   10.1270    8.5980    7.1388    5.8451    4.8345    4.4873
4.9609     6.0877     7.6368     9.4011    11.2778    13.2386    15.2370
17.2675   19.3563
a =
    0.7458    0.8254    0.9090    0.8878    0.9223    1.0124    0.9779
0.9915     1.0124     1.0204     1.0610     1.0335     1.0680     1.0335
1.0817    1.1305
```


20
15
10
5
0
-5
-10
-15
-20
Velocity (m/s)
0
5
10
15
20
25
30
t (s)
vx
vy
v


1.5
1
0.5
0
-0.5
-1
a (m/s2)
0
5
10
15
20
25
30
t (s)
ax
ay
a

**6.39** The position of the roller coaster cars ($x$ and $y$ coordinates) as a function of time, $t$, is given in the table that follows. Determine the velocity, $v$, given by $v = \sqrt{v_x^2 + v_y^2}$, where $v_x = \frac{dx}{dt}$ and $v_y = \frac{dy}{dt}$, and the acceleration, $a$, given by $a = \sqrt{a_x^2 + a_y^2}$, where $a_x = \frac{d^2x}{dt^2}$ and $a_y = \frac{d^2y}{\partial t^2}$ of the cars. Write a MATLAB program in a script file that:




(*a*) Determines $v$ and $a$ as a function of time. (The user-defined functions `FirstDerivUneq` and `SndDerivUneq` can be used if Problems 6.22 and 6.23 were solved.)
(*b*) Displays plots of $y$ versus $x$, $v$ versus $x$, and $a$ versus $x$. (Three figures on one page.)

| $t$ (s) | 0 | 1 | 2 | 2.5 | 3 | 3.5 | 4 | 4.5 | 5 | 5.5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $x$ (m) | 0 | 4.1 | 14.9 | 25.4 | 37.5 | 48.4 | 59 | 69.6 | 80.3 | 92.2 | 103.5 | 115.3 | 119.8 |
| $y$ (m) | 31.6 | 22.3 | 7.1 | 2 | 3.54 | 7.6 | 10 | 8.2 | 4.3 | 1.8 | 5.7 | 21.2 | 31.1 |

**Solution**

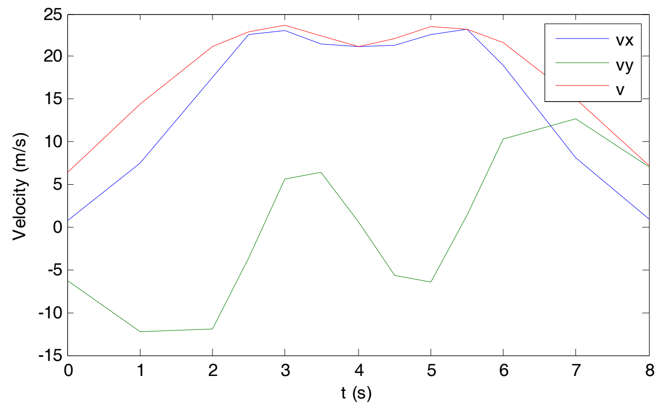
The problem is solved in the following script file:

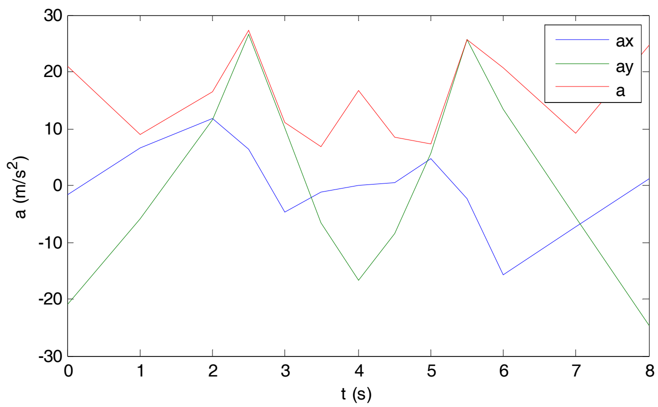
```
clear
clc
t=[0 1 2 2.5 3 3.5 4 4.5 5 5.5 6 7 8 ];
x=[0 4.1 14.9 25.4 37.5 48.4 59 69.6 80.3 92.2 103.5 115.3 119.8];
y=[31.6 22.3 7.1 2 3.54 7.6 10 8.2 4.3 1.8 5.7 21.2 31.1];
vx = FirstDerivUneq(t,x);
vy = FirstDerivUneq(t,y);
ax = SndDerivUneq(t,x);
ay = SndDerivUneq(t,y);
v=sqrt(vx.^2+vy.^2)
a=sqrt(ax.^2+ay.^2)
subplot(2,1,1)
```

```
plot(t,vx,t,vy,t,v);
xlabel('t (s)'); ylabel('Velocity (m/s)');
legend('vx','vy','v')
subplot(2,1,2)
plot(t,ax,t,ay,t,a);
xlabel('t (s)'); ylabel('a (m/s^2)');
legend('ax','ay','a')
```

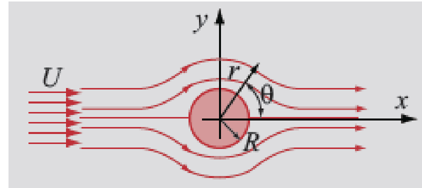
When the script is executed, the following results are displayed in the Command Window, and the figures that follow are displayed in the figure window.

```
v =
    6.3941   14.3375   21.2268   22.8787   23.6719   22.4495   21.2085
22.0495   23.4887   23.2422   21.6441   15.0901    7.1507
a =
   21.0394    8.9275   16.5750   27.3202   11.1645    6.7476   16.8000
8.4095    7.3756   25.7123   20.7583    9.2005   24.6702
```


Velocity (m/s)
t (s)
vx
vy
v


a (m/s2)
t (s)
ax
ay
a

**6.40** The nondimensional stream function $\frac{\Psi}{UR}$ for potential flow over a cylinder of radius $R$ in an incompressible flow of uniform velocity $U$ is given in the following table as a function of the non-dimensional coordinate $r/R$ and the polar angle $\theta$. The non-dimensional radial component $u_r/U$ and the azimuthal component $u_\theta/U$ of the velocity are given by:




$$\frac{u_r}{U} = \frac{1}{(r/R)}\frac{\partial\left(\frac{\Psi}{UR}\right)}{\partial\theta} \quad \text{and} \quad \frac{u_\theta}{U} = -\frac{\partial\left(\frac{\Psi}{UR}\right)}{\partial(r/R)}$$

Calculate $u_r/U$ and $u_\theta/U$ at every point. Write a MATLAB program that uses two-point central difference formulas at the interior points and one-sided three-point forward and backward difference formulas at the endpoints. The user-defined function `ParDer` can be use instead if Problem 6.24 was solved.

| | θ=0° | θ=36° | θ=72° | θ=108° | θ=144° | θ=180° | θ=216° | θ=252° | θ=288° | θ=324° | θ=360° |
|---|---|---|---|---|---|---|---|---|---|---|---|
| r/R=0.2 | 0 | -2.8214 | -4.5651 | -4.5651 | -2.8214 | 0 | 2.8214 | 4.5651 | 4.5651 | 2.8214 | 0 |
| r/R=0.6 | 0 | -0.6270 | -1.0145 | -1.014 | -0.6270 | 0 | 0.6270 | 1.0145 | 1.0145 | 0.6270 | 0 |
| r/R=1.0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| r/R=1.4 | 0 | 0.4031 | 0.6522 | 0.6522 | 0.4031 | 0 | -0.4031 | -0.6522 | -0.6522 | -0.4031 | 0 |
| r/R=1.8 | 0 | 0.7315 | 1.1835 | 1.1835 | 0.7315 | 0 | -0.7315 | -1.1835 | -1.1835 | -0.7315 | 0 |
| r/R=2.2 | 0 | 1.0260 | 1.6600 | 1.6600 | 1.0260 | 0 | -1.0260 | -1.6600 | -1.6600 | -1.0260 | 0 |
| r/R=2.6 | 0 | 1.3022 | 2.1070 | 2.1070 | 1.3022 | 0 | -1.3022 | -2.1070 | -2.1070 | -1.3022 | 0 |
| r/R=3.0 | 0 | 1.5674 | 2.5362 | 2.5362 | 1.5674 | 0 | -1.5674 | -2.5362 | -2.5362 | -1.5674 | 0 |

## Solution

The user-defined function `ParDer` from Problem 6.24 can be used here by recognizing that x=θ and y=r/R. The following script uses the function `ParDer` from Problem 6.24:

```
%   Problem (6.40)
x=[0 36 72 108 144 180 216 252 288 324 360];
y=[0.2 0.6 1.0 1.4 1.8 2.2 2.6 3.0];
f=[0 0 0 0 0 0 0 0
-2.8214 -0.6270 0 0.4031 0.7315 1.0260 1.3022 1.5674
-4.5651 -1.0145 0 0.6522 1.1835 1.66 2.1070 2.5362
-4.5651 -1.014 0 0.6522 1.1835 1.66 2.1070 2.5362
-2.8214 -0.6270 0 0.4031 0.7315 1.0260 1.3022 1.5674
0 0 0 0 0 0 0 0
2.8214 0.6270 0 -0.4031 -0.7315 -1.0260 -1.3022 -1.5674
4.5651 1.0145 0 -0.6522 -1.1835 -1.66 -2.1070 -2.5362
4.5651 1.0145 0 -0.6522 -1.1835 -1.66 -2.1070 -2.5362
2.8214 0.6270 0 -0.4031 -0.7315 -1.0260 -1.3022 -1.5674
0 0 0 0 0 0 0 0];
[dfdx,dfdy]=ParDer(x,y,f);
```

```
n=length(x); m=length(y);
for i=1:n
    for j=1:m
        u_r_over_U(i,j) = (1/y(j))*dfdx(i,j);
        u_theta_over_U(i,j) = -dfdy(i,j);
    end
end
u_r_over_U
u_theta_over_U
```

When executed, the output is:

```
>> Problem_6_40
u_r_over_U =
  -0.4667  -0.0346        0   0.0095   0.0134   0.0154   0.0166   0.0173
  -0.3170  -0.0235        0   0.0065   0.0091   0.0105   0.0113   0.0117
  -0.1211  -0.0090        0   0.0025   0.0035   0.0040   0.0043   0.0045
   0.1211   0.0090        0  -0.0025  -0.0035  -0.0040  -0.0043  -0.0045
   0.3170   0.0235        0  -0.0065  -0.0091  -0.0105  -0.0113  -0.0117
   0.3919   0.0290        0  -0.0080  -0.0113  -0.0130  -0.0139  -0.0145
   0.3170   0.0235        0  -0.0065  -0.0091  -0.0105  -0.0113  -0.0117
   0.1211   0.0090        0  -0.0025  -0.0035  -0.0040  -0.0043  -0.0045
  -0.1211  -0.0090        0   0.0025   0.0035   0.0040   0.0043   0.0045
  -0.3170  -0.0235        0   0.0065   0.0091   0.0105   0.0113   0.0117
  -0.4667  -0.0346        0   0.0095   0.0134   0.0154   0.0166   0.0173
u_theta_over_U =
        0        0        0        0        0        0        0        0
  -7.4453  -3.5268  -1.2876  -0.9144  -0.7786  -0.7134  -0.6767  -0.6492
 -12.0466  -5.7064  -2.0834  -1.4794  -1.2598  -1.1544  -1.0953  -1.0507
 -12.0491  -5.7064  -2.0827  -1.4794  -1.2598  -1.1544  -1.0953  -1.0507
  -7.4453  -3.5268  -1.2876  -0.9144  -0.7786  -0.7134  -0.6767  -0.6492
        0        0        0        0        0        0        0        0
   7.4453   3.5268   1.2876   0.9144   0.7786   0.7134   0.6767   0.6492
  12.0466   5.7064   2.0834   1.4794   1.2598   1.1544   1.0953   1.0507
  12.0466   5.7064   2.0834   1.4794   1.2598   1.1544   1.0953   1.0507
   7.4453   3.5268   1.2876   0.9144   0.7786   0.7134   0.6767   0.6492
        0        0        0        0        0        0        0        0
>>
```