```
> restart;
  with( linalg ):
Warning, new definition for norm
Warning, new definition for trace
```

Homework Solutions for Math 524 -- Chapter 4

Prepared by Douglas Meade (meade@math.sc.edu)
2 December 1995

WWW URL: http://www.math.sc.edu/~meade/math524/hw/ch4hw.ms

Updated for Maple V, Release 4, by Douglas B. Meade (5 November 1996)

All problems are taken from Chapter 4 (pp. 152-155) of "The Mathematics of Nonlinear Programming", by Peressini, Sullivan, Uhl.

All plots have been deleted from this worksheet. To see the plots, download this worksheet and execute the commands in sequence.

#1. An overdetermined system of linear equations is given; the least squares solution is sought. Solving the normal equation will lead to this solution.

Begin by writing the system of equations in the form Ax=b with

```
> A := matrix( 6, 3, [ [1,1,1], [0,0,1], [1,0,1], [2,0,5], [-7,8,0],
  [1,2,-1] ] );
```

$$A := \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \\ 2 & 0 & 5 \\ -7 & 8 & 0 \\ 1 & 2 & -1 \end{bmatrix}$$

```
> b := vector( 6, [3,1,2,8,0,1] );
```

$$b := [3, 1, 2, 8, 0, 1]$$

The least squares solution to this overdetermined system is found by solving the normal equation:

transpose(A) A x = transpose(A) b

```
> M := evalm( transpose(A) &* A );
```

$$M := \begin{bmatrix} 56 & -53 & 11 \\ -53 & 69 & -1 \\ 11 & -1 & 29 \end{bmatrix}$$

```
> c := evalm( transpose(A) &* b );
```

$$c := [22, 5, 45]$$

The solution is found to be

```
> x := evalm( inverse(M) &* c );
```

$$x := \left[\frac{4965}{5839}, \frac{4343}{5839}, \frac{7327}{5839}\right]$$

```
> evalf(evalm(x));
```

$$[.8503168351, .7437917452, 1.254838157]$$

#3

Let H[1], H[2], H[3] denote the heights of the three hills. Each measurement yields an equation relating one or more of these heights. There are a total of six equations, which can be written in matrix form as A H = b with

```
> A := matrix( 6, 3, [ [1,0,0], [0,1,0], [0,0,1], [-1,1,0],
  [-1,0,1], [0,-1,1] ] );
```

$$A := \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & -1 & 1 \end{bmatrix}$$

```
> b := vector( 6, [ 1236, 1941, 2417, 711, 1177, 475 ] );
```

$$b := [1236, 1941, 2417, 711, 1177, 475]$$

The "best" solution to this overdetermined system is the least squares solution. Which is found to

```
> H := evalm( inverse( transpose(A) &* A ) &* transpose(A) &* b );
```

$$H := \left[\frac{2471}{2}, \frac{7771}{4}, \frac{9663}{4}\right]$$

```
> evalf( evalm(H) );
```

$$[1235.500000, 1942.750000, 2415.750000]$$

Note that these heights do a fairly good job of satisfying all six equations. Here is the residual vector, A H - b :

```
> evalm( A &* H - b );
```

$$\left[\frac{-1}{2}, \frac{7}{4}, \frac{-5}{4}, \frac{-15}{4}, \frac{13}{4}, -2\right]$$

```
>
```

#7

The explicit formulae for linear regression could be used, but I prefer to work with the general concept. Thus, let’s first write the overdetermined system, then find the least squares solution.

Assuming the data fits a linear equation of the form y = a + b*x the data points lead to a system of six equations which, when written as Ax=b, appear as:

```
> A := matrix( 6, 2, [ [1,-2], [1,-1], [1,0], [1,1], [1,2], [1,3] ]
  );
```

$$A := \begin{bmatrix} 1 & -2 \\ 1 & -1 \\ 1 & 0 \\ 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{bmatrix}$$

```
> b := vector( 6, [ 12, 11, 8, 5, 2, -3 ] );
```

$$b := [12, 11, 8, 5, 2, -3]$$

The normal equations are (check that the entries are as we expect them to be)

```
> M := evalm( transpose(A) &* A );
```

$$M := \begin{bmatrix} 6 & 3 \\ 3 & 19 \end{bmatrix}$$

```
> c := evalm( transpose(A) &* b );
```

$$c := [35, -35]$$

Since a 2x2 matrix is not too difficult to invert, the solution to Mx=c is not difficult to obtain. It is

```
> x := evalm( inverse( M ) &* c );
```

$$x := \left[\frac{22}{3}, -3\right]$$

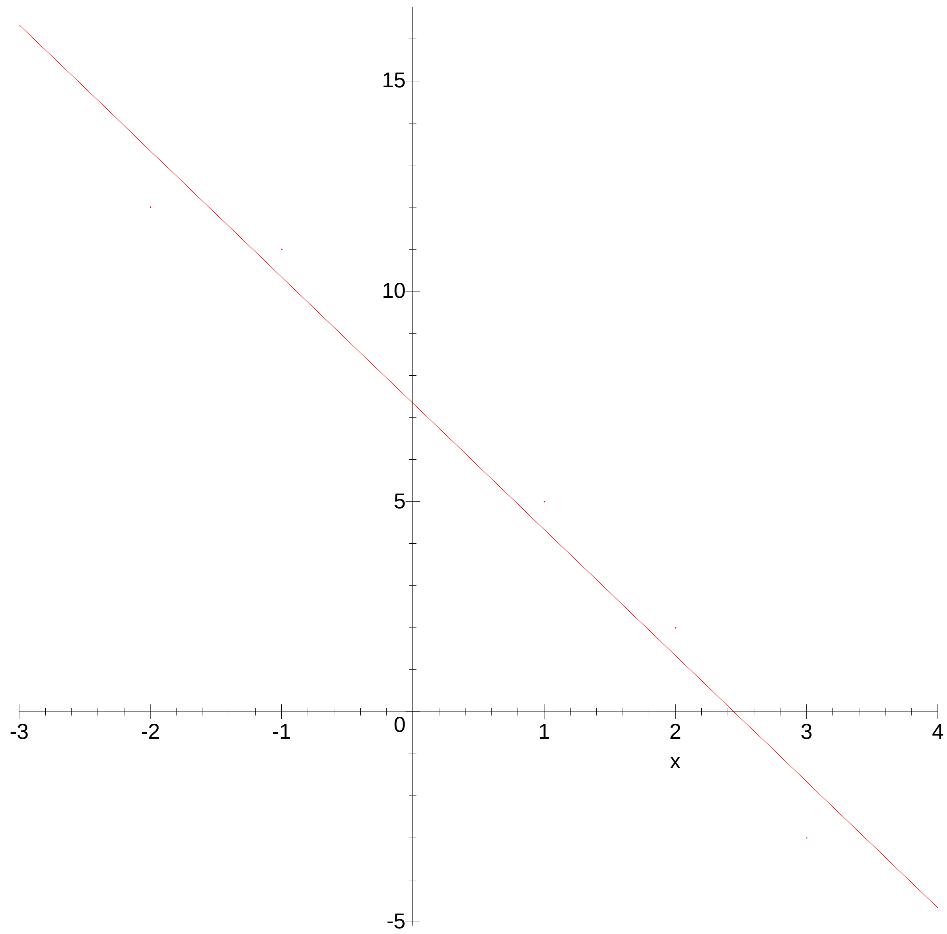
and the best linear fit to this data is y = 22/3 - 3*x

```
> points := augment( submatrix( A, 1..6, [2] ), b );
```

$$points := \begin{bmatrix} -2 & 12 \\ -1 & 11 \\ 0 & 8 \\ 1 & 5 \\ 2 & 2 \\ 3 & -3 \end{bmatrix}$$

```
> Pdata := plot( convert(points,listlist), style=point, color=red ):
  Pfit := plot( 22/3 - 3*x, x=-3..4 ):
  plots[display]( [Pdata, Pfit], title=`Problem #7` );
```

Problem #7



>

#10

a.

These three points all lie on the straight line: y = 1 - x. It really would be a waste of brain power to compute and solve the normal equations.

b.

The extra data point does not lie on the same line. The best linear fit will have to be found from the normal equations (which are easily seen to be) M x = c with

```
> M := matrix( 2, 2, [ [4, 7], [7, 51] ] );
```

$$M := \begin{bmatrix} 4 & 7 \\ 7 & 51 \end{bmatrix}$$

```
> c := vector( 2, [ 11, 54 ] );
```

$$c := [11, 54]$$

The solution to these equations is

```
> x := evalm( inverse( M ) &* c );
```

$$x := \left[\frac{183}{155}, \frac{139}{155}\right]$$

The plot of the data and the two lines should show exactly what has happened here (you do have to look carefully to see the first three data points)

```
> points1 := matrix( 3, 2, [ [-1,2], [0,1], [1,0] ] );
```

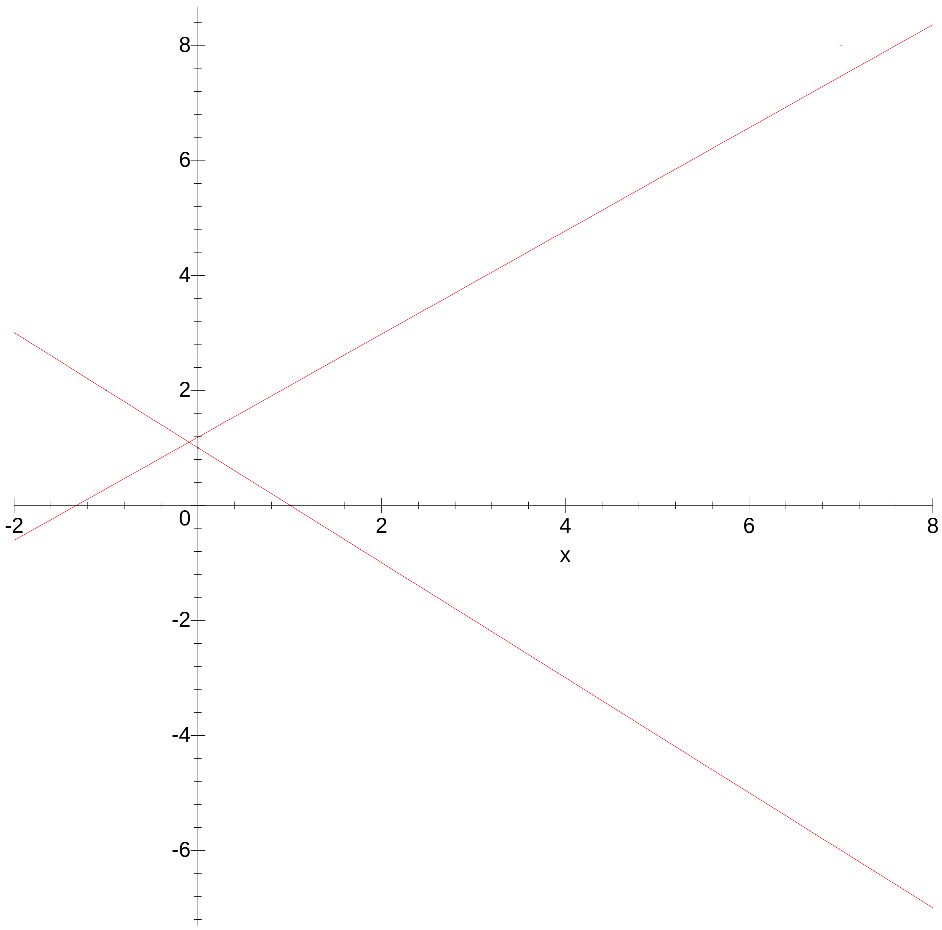
$$points1 := \begin{bmatrix} -1 & 2 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$$

```
> points2 := matrix( 4,2, [ [-1,2], [0,1], [1,0], [7,8] ] );
```

$$points2 := \begin{bmatrix} -1 & 2 \\ 0 & 1 \\ 1 & 0 \\ 7 & 8 \end{bmatrix}$$

```
> P1 := plot( convert(points1,listlist), style=point,
  symbol=diamond, color=blue ):
  P2 := plot( convert(points2,listlist), style=point, color=green ):
  P3 := plot( 1-x, x=-2..8 ):
  P4 := plot( 183/155 + 139/155*x, x=-2..8 ):
  plots[display]( [P1,P2,P3,P4], title=‘Problem #10‘ );
```

Problem #10




```
>
```

#11

This problem is almost embarassingly simple, after the others that we've done. But, here it is (without the intermediate steps):

```
> A := matrix( 3, 2, [ [0,1], [1,2], [2,3] ] );
```

$$A := \begin{bmatrix} 0 & 1 \\ 1 & 2 \\ 2 & 3 \end{bmatrix}$$

The generalized inverse is

```
> Adag := evalm( inverse( transpose(A) &* A ) &* transpose(A) );
```

$$Adag := \begin{bmatrix} \frac{-4}{3} & \frac{-1}{3} & \frac{2}{3} \\ \frac{5}{6} & \frac{1}{3} & \frac{-1}{6} \end{bmatrix}$$

To check this answer, and to see the role of the generalized inverse:

```
> evalm( Adag &* A );
```

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

```
> evalm( A &* Adag );
```

$$\begin{bmatrix} \frac{5}{6} & \frac{1}{3} & \frac{-1}{6} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{-1}{6} & \frac{1}{3} & \frac{5}{6} \end{bmatrix}$$

#4 (orthogonal projection onto a subspace)

The subspace in which we seek our solution is M = range(A) where the columns of A are the two vectors given in the problem (why is this the case?)

```
> A := matrix( 3, 2, [ [1,2], [1,-1], [2,1] ]);
```

$$A := \begin{bmatrix} 1 & 2 \\ 1 & -1 \\ 2 & 1 \end{bmatrix}$$

The (unconstrained) problem is to find the two-dimensional vector w = (alpha, beta) that minimizes the length of x - A w where

```
> x := vector( 3, [1,1,1] );
```

$$x := [1, 1, 1]$$

The solution to this problem is found as in all the previous problems:

```
> w := evalm( inverse( transpose(A) &* A ) &* transpose(A) &* x );
```

$$w := \left[\frac{2}{3}, 0\right]$$

This is the best choice of alpha and beta. The corresponding vector v is

```
> v := evalm( A &* w );
```

$$v := \left[\frac{2}{3}, \frac{2}{3}, \frac{4}{3}\right]$$

```
>
```

#8 (quadratic best fit)

a.

We begin exactly as we began our derivation of the linear regression formulae. Fitting m data points (x1,y1), ..., (xm,ym) to a quadratic function y = c + b*x + a*x^2 leads to a system of m equations for the three unknowns w = (c,b,a). This system will look like

```
[ 1  x1  x1^2   ]            [ y1 ]
[ 1  x2  x2^2   ] [ c ]      [ y2 ]
[ 1  x3  x3^2  ] [ b ] = [ y3 ]
[               ] [ a ]     [     ]
[ 1  xm  xm^2 ]             [ ym ]
```

It is not difficult to compute the normal equation for this system. There is a nice strucure to this 3x3 system

```
> M := matrix( 3, 3, [ [ m, sum(x[i],i=1..m), sum(x[i]^2,i=1..m ) ],
                      [ sum(x[i],i=1..m), sum(x[i]^2,i=1..m ), sum(x[i]^3,i=1..m ) ],
                      [  sum(x[i]^2,i=1..m), sum(x[i]^3,i=1..m ), sum(x[i]^4,i=1..m ) ] ] );
```

$$M := \begin{bmatrix} m & \sum_{i=1}^{m} x_i & \sum_{i=1}^{m} x_i^2 \\ \sum_{i=1}^{m} x_i & \sum_{i=1}^{m} x_i^2 & \sum_{i=1}^{m} x_i^3 \\ \sum_{i=1}^{m} x_i^2 & \sum_{i=1}^{m} x_i^3 & \sum_{i=1}^{m} x_i^4 \end{bmatrix}$$

```
> d := vector( 3, [ sum( y[i], i=1..m ), sum( x[i]*y[i], i=1..m ), sum( x[i]^2*y[i], i=1..m ) ] );
```

$$d := \left[\sum_{i=1}^{m} y_i, \sum_{i=1}^{m} x_i\, y_i, \sum_{i=1}^{m} x_i^2\, y_i \right]$$

While it is possible to explicitly solve this system, I do not see that it has any real value.

```
> w := evalm( inverse(M) &* d ):
  c := w[1];
  b := w[2];
  a := w[3];
```

$$c := \frac{\left(-\left(\sum_{i=1}^{m} x_i^2\right)\left(\sum_{i=1}^{m} x_i^4\right) + \left(\sum_{i=1}^{m} x_i^3\right)^2\right)\left(\sum_{i=1}^{m} y_i\right)}{\%1}$$

$$-\frac{\left(-\left(\sum_{i=1}^{m} x_i\right)\left(\sum_{i=1}^{m} x_i^{4}\right)+\left(\sum_{i=1}^{m} x_i^{2}\right)\left(\sum_{i=1}^{m} x_i^{3}\right)\right)\left(\sum_{i=1}^{m} x_i\, y_i\right)}{\%1}$$

$$-\frac{\left(\left(\sum_{i=1}^{m} x_i\right)\left(\sum_{i=1}^{m} x_i^{3}\right)-\left(\sum_{i=1}^{m} x_i^{2}\right)^{2}\right)\left(\sum_{i=1}^{m} x_i^{2}\, y_i\right)}{\%1}$$

$$\%1 := -m\left(\sum_{i=1}^{m} x_i^{2}\right)\left(\sum_{i=1}^{m} x_i^{4}\right)+m\left(\sum_{i=1}^{m} x_i^{3}\right)^{2}+\left(\sum_{i=1}^{m} x_i\right)^{2}\left(\sum_{i=1}^{m} x_i^{4}\right)-2\left(\sum_{i=1}^{m} x_i\right)\left(\sum_{i=1}^{m} x_i^{2}\right)\left(\sum_{i=1}^{m} x_i^{3}\right)$$
$$+\left(\sum_{i=1}^{m} x_i^{2}\right)^{3}$$

$$b := -\frac{\left(-\left(\sum_{i=1}^{m} x_i\right)\left(\sum_{i=1}^{m} x_i^{4}\right)+\left(\sum_{i=1}^{m} x_i^{2}\right)\left(\sum_{i=1}^{m} x_i^{3}\right)\right)\left(\sum_{i=1}^{m} y_i\right)}{\%1}-\frac{\left(m\left(\sum_{i=1}^{m} x_i^{4}\right)-\left(\sum_{i=1}^{m} x_i^{2}\right)^{2}\right)\left(\sum_{i=1}^{m} x_i\, y_i\right)}{\%1}$$
$$+\frac{\left(m\left(\sum_{i=1}^{m} x_i^{3}\right)-\left(\sum_{i=1}^{m} x_i\right)\left(\sum_{i=1}^{m} x_i^{2}\right)\right)\left(\sum_{i=1}^{m} x_i^{2}\, y_i\right)}{\%1}$$

$$\%1 := -m\left(\sum_{i=1}^{m} x_i^{2}\right)\left(\sum_{i=1}^{m} x_i^{4}\right)+m\left(\sum_{i=1}^{m} x_i^{3}\right)^{2}+\left(\sum_{i=1}^{m} x_i\right)^{2}\left(\sum_{i=1}^{m} x_i^{4}\right)-2\left(\sum_{i=1}^{m} x_i\right)\left(\sum_{i=1}^{m} x_i^{2}\right)\left(\sum_{i=1}^{m} x_i^{3}\right)$$
$$+\left(\sum_{i=1}^{m} x_i^{2}\right)^{3}$$

$$a := -\frac{\left(\left(\sum_{i=1}^{m} x_i\right)\left(\sum_{i=1}^{m} x_i^{3}\right)-\left(\sum_{i=1}^{m} x_i^{2}\right)^{2}\right)\left(\sum_{i=1}^{m} y_i\right)}{\%1}+\frac{\left(m\left(\sum_{i=1}^{m} x_i^{3}\right)-\left(\sum_{i=1}^{m} x_i\right)\left(\sum_{i=1}^{m} x_i^{2}\right)\right)\left(\sum_{i=1}^{m} x_i\, y_i\right)}{\%1}$$
$$-\frac{\left(m\left(\sum_{i=1}^{m} x_i^{2}\right)-\left(\sum_{i=1}^{m} x_i\right)^{2}\right)\left(\sum_{i=1}^{m} x_i^{2}\, y_i\right)}{\%1}$$

$$\%1 := -m\left(\sum_{i=1}^{m} x_i^{2}\right)\left(\sum_{i=1}^{m} x_i^{4}\right)+m\left(\sum_{i=1}^{m} x_i^{3}\right)^{2}+\left(\sum_{i=1}^{m} x_i\right)^{2}\left(\sum_{i=1}^{m} x_i^{4}\right)-2\left(\sum_{i=1}^{m} x_i\right)\left(\sum_{i=1}^{m} x_i^{2}\right)\left(\sum_{i=1}^{m} x_i^{3}\right)$$

$$+\left(\sum_{i=1}^{m} x_i^{\,2}\right)^3$$

b.

Now that we have specific data, let's see what each step of the process looks like. We begin with the linear system of seven equations for the three coefficients: A w = f

```
> A := matrix( 7, 3, [ [1, -2, 4], [1,-1,1], [1,0,0], [1,1,1],
  [1,2,4], [1,3,9], [1,4,16] ] );
```

$$A := \begin{bmatrix} 1 & -2 & 4 \\ 1 & -1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \\ 1 & 4 & 16 \end{bmatrix}$$

```
> f := vector( 7, [ -5, -1, 4, 7, 6, 5, -1 ] );
```

$$f := [-5, -1, 4, 7, 6, 5, -1]$$

The normal equation is M w = d where (check that these matrices agree with those from a)

```
> M := evalm( transpose(A) &* A );
```

$$M := \begin{bmatrix} 7 & 7 & 35 \\ 7 & 35 & 91 \\ 35 & 91 & 371 \end{bmatrix}$$

```
> d := evalm( transpose(A) &* f );
```

$$d := [15, 41, 39]$$

The solution to this system is

```
> w := evalm( inverse(M) &* d );
```

$$w := \left[\frac{61}{14}, \frac{127}{42}, \frac{-22}{21}\right]$$

so the best quadratic fit to this data is y = 61/14 + 127/42*x - 22/21*x^2. Let's see how this looks graphically.

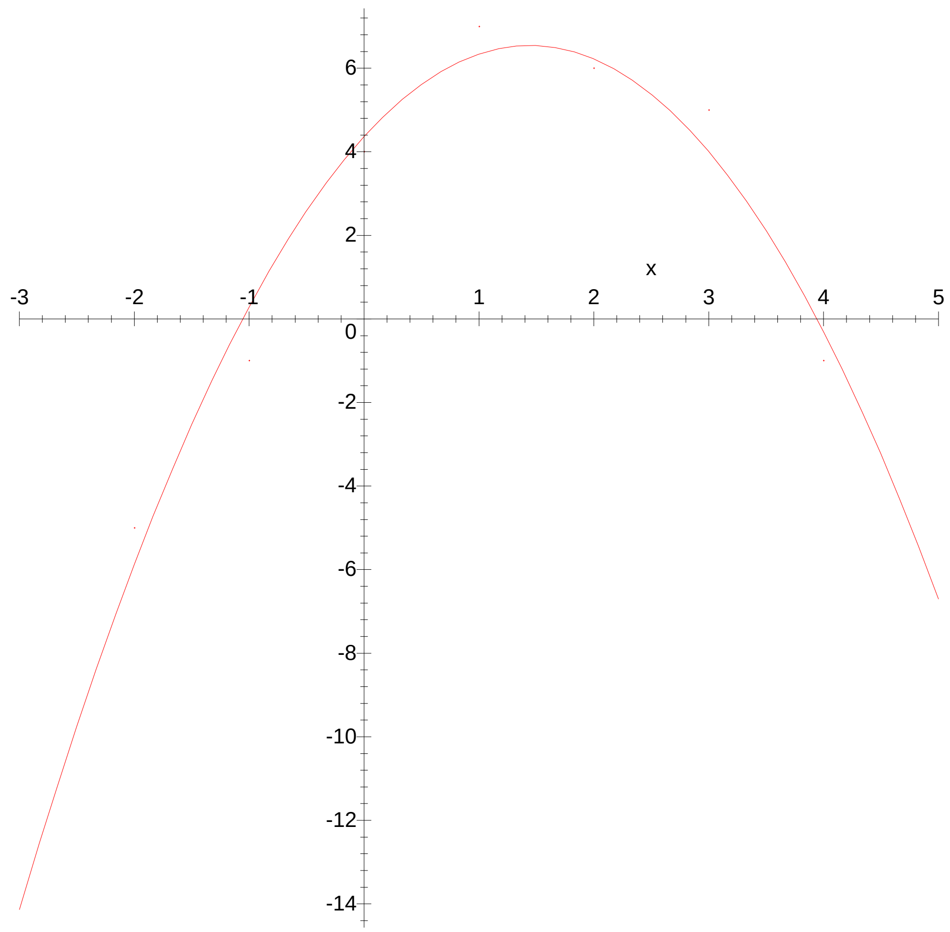
```
> points := augment( submatrix(A,1..7,[2]), f );
```

$$points := \begin{bmatrix} -2 & -5 \\ -1 & -1 \\ 0 & 4 \\ 1 & 7 \\ 2 & 6 \\ 3 & 5 \\ 4 & -1 \end{bmatrix}$$

```
> P1 := plot( convert(points,listlist), style=point, color=red ):
```

```
P2 := plot( 61/14 + 127/42*x - 22/21*x^2, x=-3..5 ):
plots[display]( [P1,P2], title=‘Problem 8b‘ );
```

Problem 8b



\>

End of Worksheet

\>