Principal Component Analysis

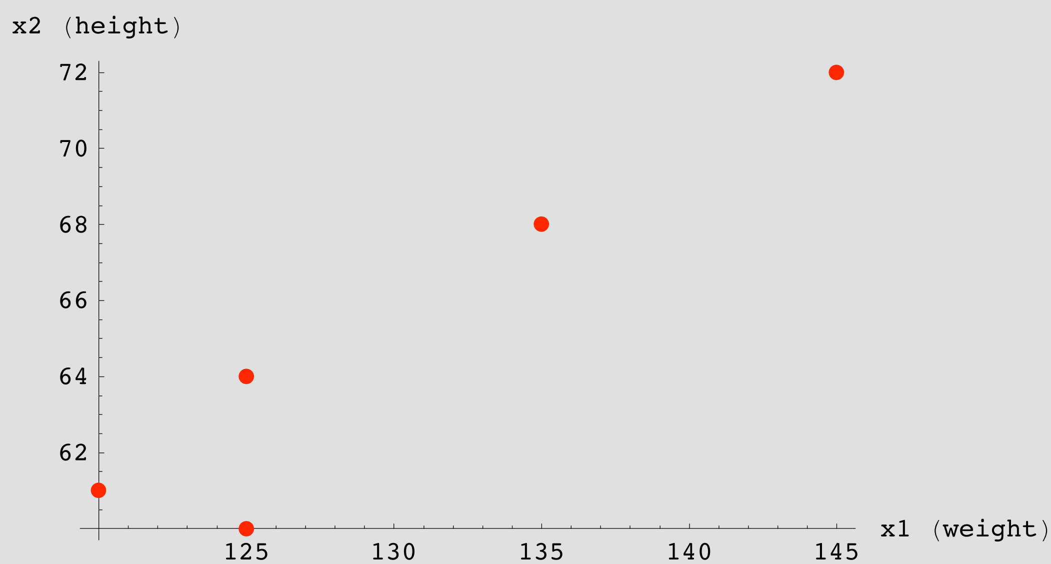
Sample Covariance Matrix

Convert the matrix a to mean deviation form and calculate its sample covariance matrix.
(See Lay 7.5, pp489-491)

```
Clear[a];

a = (120  125  125  135  145
      61   60   64   68   72);

ListPlot[Transpose[a],
  ImageSize → 400,
  AxesLabel → {"x1 (weight)", "x2 (height)"},
  PlotStyle → {PointSize[0.02], Red}];
```



Write a procedure to calculate the **sample mean** of a matrix and use it to calculate the sample mean of a.

```
sampleMean[a_?MatrixQ] :=
 Module[{n = Dimensions[a][[2]]},
        1/n Apply[Plus, Transpose[a]]]
```

```
m = sampleMean[a];
% // MatrixForm
```

$$\begin{pmatrix} 130 \\ 65 \end{pmatrix}$$

Write a procedure to calculate the **mean deviation form** of a matrix and use it to calculate the mean deviation form of a.

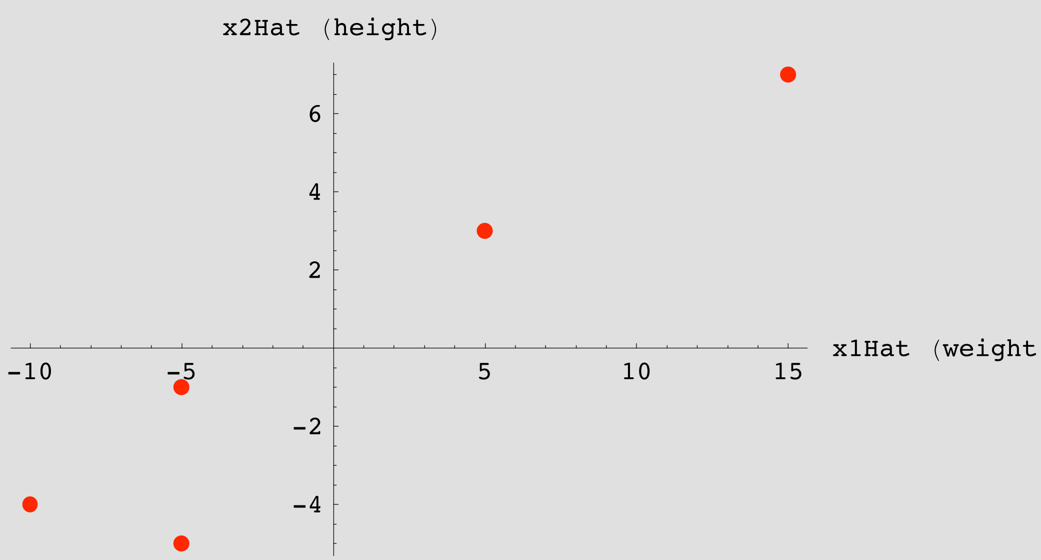
```
<< LinearAlgebra`MatrixManipulation`
```

```
meanDeviationForm[a_?MatrixQ] :=
  Module[{m = sampleMean[a], n = Dimensions[a][[2]]},
          Transpose[Table[Transpose[a][[k]] - m,
                                  {k, 1, n}]]];
```

```
b = meanDeviationForm[a];
% // MatrixForm

ListPlot[Transpose[b],
  ImageSize → 400,
  AxesLabel → {"x1Hat (weight)", "x2Hat (height)"},
  PlotStyle → {PointSize[0.02], Red}];
```

$$\begin{pmatrix} -10 & -5 & -5 & 5 & 15 \\ -4 & -5 & -1 & 3 & 7 \end{pmatrix}$$



Check: b should have zero sample mean.

```
sampleMean[b];
% // MatrixForm
```

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Write a procedure to calculate the **sample covariance matrix** of a given matrix and use it to calculate the sample covariance matrix of a.

```
sampleCovarianceMatrix[a_?MatrixQ] :=
 Module[{n = Dimensions[a][[2]]},
         1/(n - 1) a.Transpose[a]]
```

```
s = sampleCovarianceMatrix[b];
% // MatrixForm
```

$$\begin{pmatrix} 100 & \frac{95}{2} \\ \frac{95}{2} & 25 \end{pmatrix}$$

Principal Components

Bring forward the sample covariance matrix from the previous example.

```
s // MatrixForm
```

$$\begin{pmatrix} 100 & \frac{95}{2} \\ \frac{95}{2} & 25 \end{pmatrix}$$

We seek to diagonalize the matrix s.
Calculate the eigendata for s.

```
{evals, evecs} = Eigensystem[s]
```

$$\left\{\left\{\frac{1}{2}\left(125+5\sqrt{586}\right), \frac{1}{2}\left(125-5\sqrt{586}\right)\right\}, \left\{\left\{-\frac{10}{19}+\frac{1}{95}\left(125+5\sqrt{586}\right), 1\right\}, \left\{-\frac{10}{19}+\frac{1}{95}\left(125-5\sqrt{586}\right), 1\right\}\right\}\right\}$$

Construct d, the diagonal matrix of eigenvalues in decreasing order.

```
d = DiagonalMatrix[evals] // N;
% // MatrixForm
```

$$\begin{pmatrix} 123.019 & 0. \\ 0. & 1.98141 \end{pmatrix}$$

Construct the orthonormal matrix p, consisting of normalized eigenvectors of s.

```
p = Transpose[Table[evecs[[k]] / Norm[evecs[[k]] // N],
    {k, 2}]];
% // MatrixForm

( 0.899901  -0.436094 )
( 0.436094   0.899901 )
```

Check the diagonalization.

```
s == p.d.Transpose[p]

True
```

The columns of p are the principal components of the data.

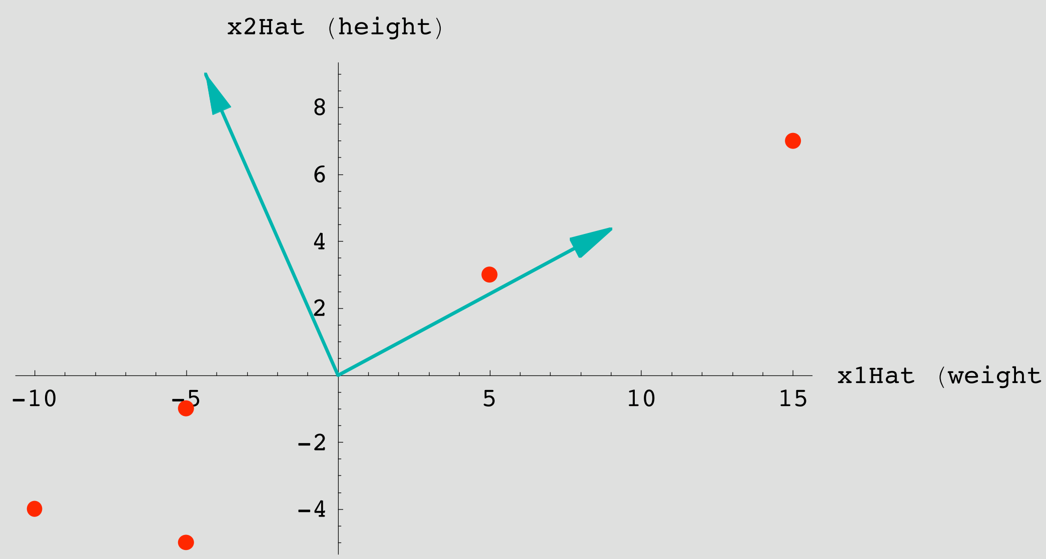
```
<< Graphics`Graphics`
<< Graphics`Arrow`
```

```
showColorfulVectors[vecs_, color_, opts___] :=
 Show[Graphics[Flatten[{Thickness[.004], color, Table[Arrow[
       vecs[[i, 1]], vecs[[i, 1]] + vecs[[i, 2]]], {i, Length[vecs]}]}]],
  opts, AspectRatio → Automatic, Axes → True]
```

```
{u1, u2} = Transpose[p]
o = {0, 0};

DisplayTogether[
  ListPlot[Transpose[b],
   ImageSize → 400,
   AxesLabel → {"x1Hat (weight)", "x2Hat (height)"},
   PlotStyle → {PointSize[0.02], Red}],
  Show[showColorfulVectors[{{o, 10 u1}, {o, 10 u2}}, ManganeseBlue]]];

{{0.899901, 0.436094}, {-0.436094, 0.899901}}
```



Use the principal components to define and relate the variables X and Y.
X represents the original data. Y represents the transformed data.
The change of basis matrix p is orthonormal.

```
x = {x1, x2};
y = {y1, y2};

y = Transpose[p].x;
% // MatrixForm
```

$$\begin{pmatrix} 0.899901\,x1 + 0.436094\,x2 \\ -0.436094\,x1 + 0.899901\,x2 \end{pmatrix}$$

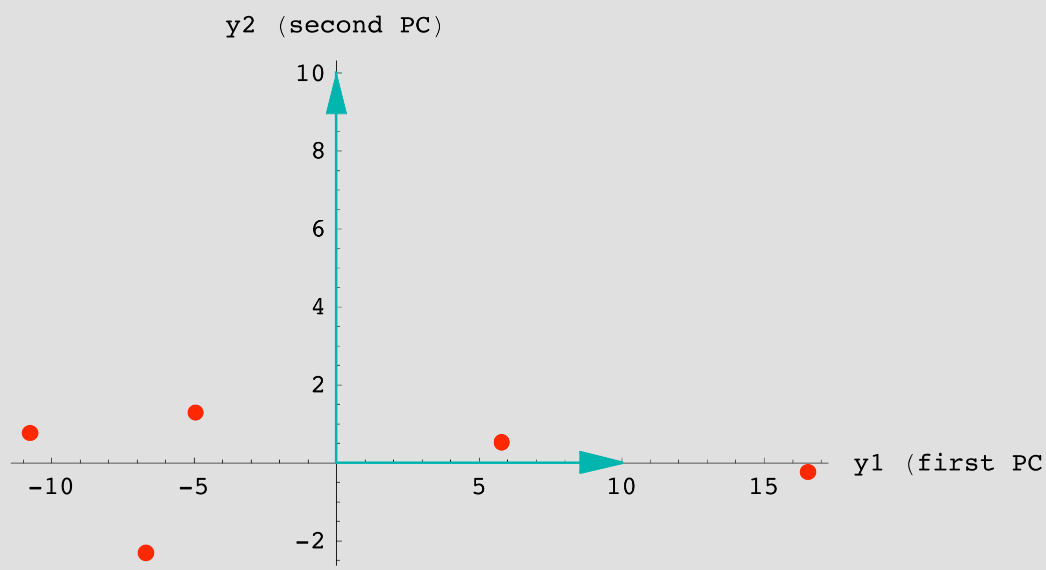
The matrix d is the covariance matrix for the transformed data.
y1 and y2 are independent.

```
d // MatrixForm
```

$$\begin{pmatrix} 123.019 & 0. \\ 0. & 1.98141 \end{pmatrix}$$

```
v1 = {1, 0};
v2 = {0, 1};
o = {0, 0};

DisplayTogether[
  ListPlot[Transpose[Transpose[p].b],
   ImageSize → 400,
   AxesLabel → {"y1  (first  PC)", "y2  (second  PC)"},
   PlotStyle → {PointSize[0.02], Red}],
  Show[showColorfulVectors[{{o, 10 v1}, {o, 10 v2}}, ManganeseBlue]]];
```



Calculate the percentage of the total variance contained in the first principal component.
Tr[d] is the trace of the matrix d.

```
firstPC = u1

percentVarianceFirstPC = d[[1, 1]] / Tr[d]

{0.899901, 0.436094}
```

```
0.984149
```

The first principal component accounts for 98.4% of the variance in this data.