Illustration of 3D PCA

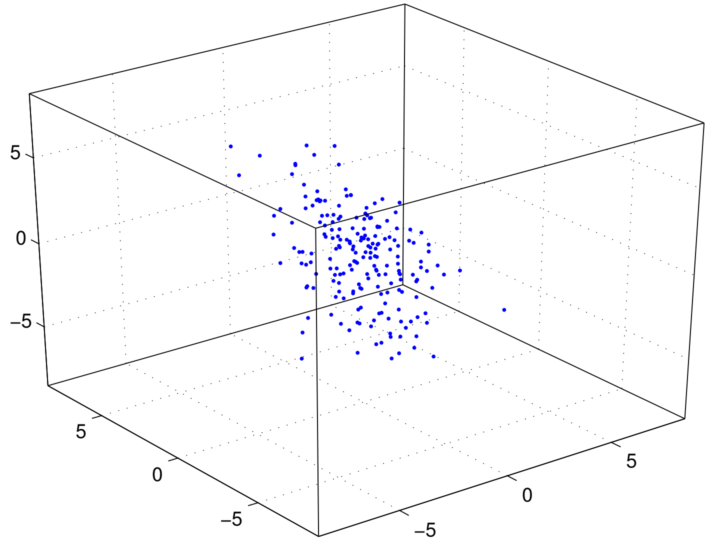
This code illustrates PCA for a 'toy' example. PCA gives a way to approximate an N-point vector by an M-point vector with M < N.

Contents

Make data for example

```
L = 200;  % number of data points

% generate data for example
C = [1 2 -1; 2 2 1; -2 0 2];
X = C * randn(3,L);

% display data
plot3(X(1,:),X(2,:),X(3,:),'b.')
grid
box on
m = max(abs(X(:)))
ax = [-1 1 -1 1 -1 1]*m;
axis(ax);
set(gca,'Projection','perspective');

% data is zero mean - no need to subtract mean first.
```

```
m =

    8.6426
```



Compute PCA matrix

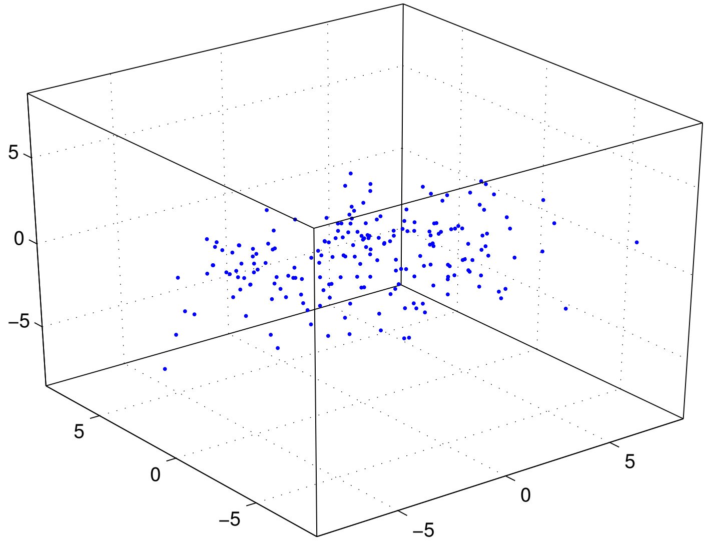
```
A = X * X';      % data covariance matrix - 3 x 3 matrix

% compute eigenvectors and eigenvalues
[E,D] = eig(A);

% check that A = E D E'
err = A - E * D * E';
max(abs(err(:)));

% eigevalues in descending order
d = diag(D);
[tmp, k] = sort(-d);

% resort
d = d(k)
D = diag(d);
E = E(:,k);

% check that A = E D E'
err = A - E * D * E';
max(abs(err(:)));

% Set P matrix
P = E';
```

```
d =

   1.0e+03 *

    2.7497
    1.2326
    0.3223
```

Transform data to new coordinates

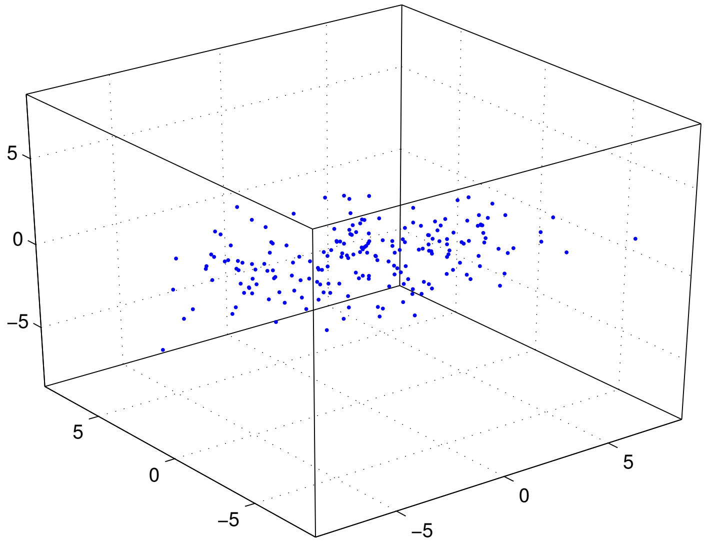
Decorrelate data

```
% transform data
Y = P * X;

% display new data
plot3(Y(1,:),Y(2,:),Y(3,:),'b.')
grid
% axis equal
box on
set(gca,'Projection','perspective');
axis(ax);
```



Approximate each 3-point data vector using two values

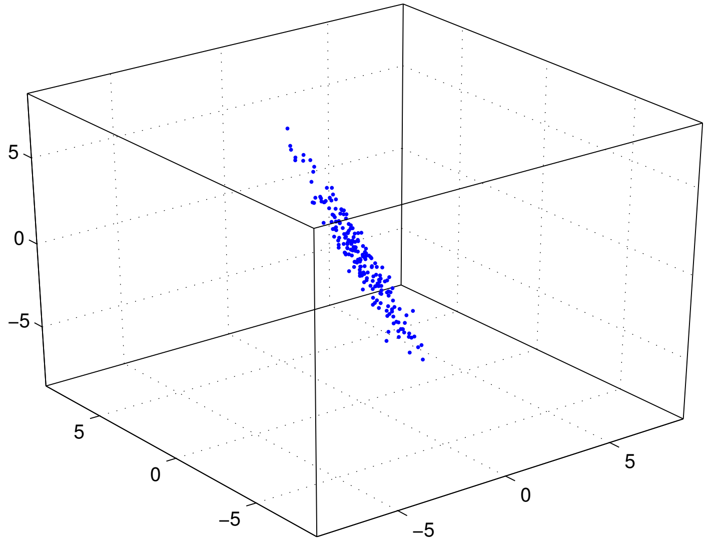
```
Y(3,:) = 0;

% display approximate data
plot3(Y(1,:), Y(2,:), Y(3,:),'b.')
grid
% axis equal
box on
set(gca,'Projection','perspective');
axis(ax);
```



Transform back to original coordinates

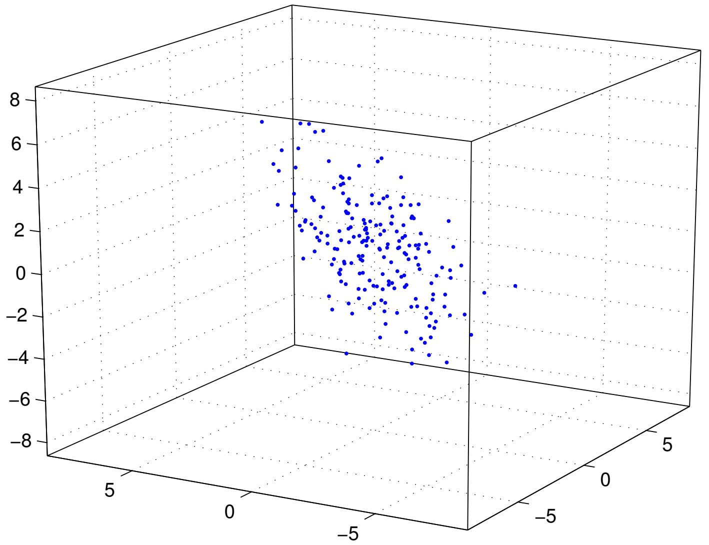
The approximate data lies in a 2-dimensional plane

```
X_approx = P'*Y;

% display approximate data
plot3(X_approx(1,:),X_approx(2,:),X_approx(3,:),'b.')
grid
% axis equal
box on
set(gca,'Projection','perspective');
axis(ax);
```



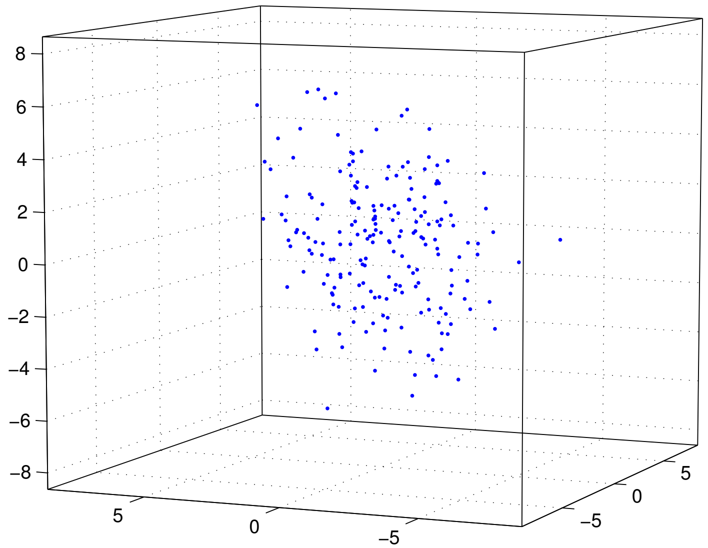
change view

```
view(-60, 18)
```



change view

```
view(-66.7, 7.8)
```


8
6
4
2
0
−2
−4
−6
−8
5
0
−5
5
0
−5