Signals #3: discrete orthogonal transforms

Vector and matrix orthogonality

One vector of N signal samples $\mathbf{x}^T = [x_1, x_2, ..., x_N]$ is **orthogonal** to another vector of Nsignal samples $\mathbf{y}^T = [y_1, y_2, ..., y_N]$ if the inner product $< . >$ of these two vectors is equal to zero $< \mathbf{x}, \mathbf{y} >= 0$ ("$*$" - complex conjugation):

$$< \mathbf{x}, \mathbf{y} >= \sum_{n=1}^{N} x_n y_n^* = x_1 y_1^* + x_2 y_2^* + ... + x_N y_N^* = \begin{bmatrix} y_1^* & y_2^* & \cdots & y_N^* \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}.$$

Check orthgonality of three vectors `v1, v2, v3`. Define your own longer othogonal vectors with without 0s.

```
% Orthogonal vectors (like in 3D x-y-z space)
v1 = [ 1 0 0 ]';      % Vector no. 1
v2 = [ 0 1 0 ]';      % Vector no. 2
v3 = [ 0 0 1 ]';      % Vector no. 1
ortho12 = v2' * v1,  % check - alternative v2'*v2, sum( v1.*conj(v2) ), dot( v1, v2 )
```

```
ortho12 = 0
```

Orthogonal matrix A consists of orthogonal columns (or rows):

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{NN} \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{N1} \end{bmatrix} & \begin{bmatrix} a_{12} \\ a_{22} \\ \vdots \\ a_{N2} \end{bmatrix} & \begin{bmatrix} \vdots \\ \vdots \\ \vdots \\ \vdots \end{bmatrix} & \begin{bmatrix} a_{1N} \\ a_{2N} \\ \vdots \\ a_{NN} \end{bmatrix} \end{bmatrix}$$

A matrix is **orthonormal** if inner product of its each column/row vector with itself is equal to 1:

$$< \mathbf{v}, \mathbf{v} >= \sum_{n=1}^{N} v_n v_n^* = v_1 v_1^* + v_2 v_2^* + ... + v_N v_N^* = 1$$

Inverse of an **orthonormal matrix** is equal the matrix complex-conjugation and transposition (columns → rows). Therefore:

$$\mathbf{A}^{-1}\mathbf{A} = (\mathbf{A}^*)^T \mathbf{A} = \begin{bmatrix} a_{11}^* & a_{21}^* & \cdots & a_{N1}^* \\ a_{12}^* & a_{22}^* & \cdots & a_{N1}^* \\ \vdots & \vdots & \ddots & \vdots \\ a_{1N}^* & a_{2N}^* & \cdots & a_{N1}^* \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{NN} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} = \mathbf{I}.$$

$$\begin{bmatrix} \begin{bmatrix} a_{11}^* & a_{21}^* & \cdots & a_{N1}^* \end{bmatrix} \\ \begin{bmatrix} a_{12}^* & a_{22}^* & \cdots & a_{N2}^* \end{bmatrix} \\ \begin{bmatrix} \cdots & \cdots & \cdots & \cdots \end{bmatrix} \\ \begin{bmatrix} a_{1N}^* & a_{2N}^* & \cdots & a_{NN}^* \end{bmatrix} \end{bmatrix} * \begin{bmatrix} \begin{bmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{N1} \end{bmatrix} & \begin{bmatrix} a_{12} \\ a_{22} \\ \vdots \\ a_{N2} \end{bmatrix} & \begin{bmatrix} \vdots \\ \vdots \\ \vdots \\ \vdots \end{bmatrix} & \begin{bmatrix} a_{1N} \\ a_{2N} \\ \vdots \\ a_{NN} \end{bmatrix} \end{bmatrix}$$

Run the section code. Define different bigger orthogonal matrices.

```
% Orthogonal matrices (like in 3D x-y-z space)
A = [ v1 v2 v3 ];     % Vector no. 1
ortho2 = A' * A,      % check whether result, you should obtain diagonl matrix
```

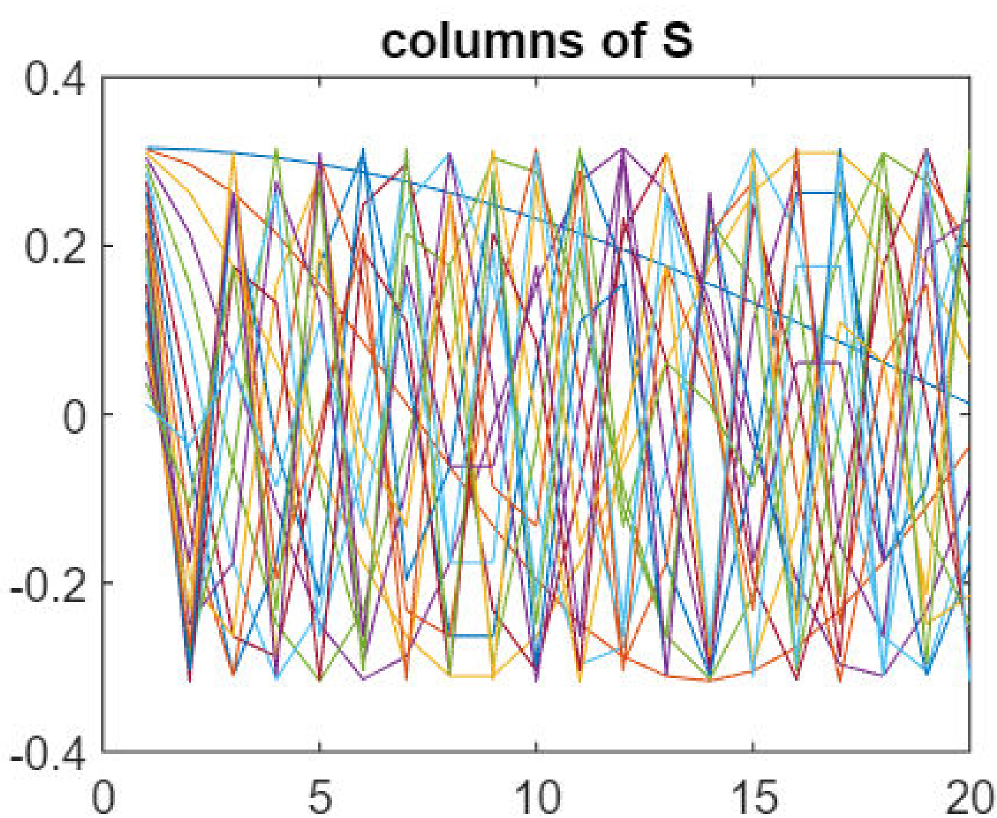
```
ortho2 = 3×3
     1     0     0
     0     1     0
     0     0     1
```

Typically matrices of discrete orthogonal transformations have **sines/cosines** in their columns, with different, increasing frequency of oscillation, e.g. in DCT-IV transform (in columns k - different oscillatory reference signals, in rows n - consecutive samples of the reference signals):

$$S[n,k] = \sqrt{\frac{2}{N}} \cos\left(\frac{\pi}{N}(k+0.5)(n+0.5)\right), \qquad k,n = 0,1,2,...,N-1.$$

Check matrix of the DCT-IV transform. Add matrics of some other orthogonal transformation.

```
% Orthogonal matrix of the DCT-IV orthogonal transform
N = 20;                                       % transformation order (square matrix size)
k = (0:N-1); n=(0:N-1);                        % columns=functions, rows=samples
S = sqrt(2/N)*cos(pi/N*(n'+1/2)*(k+1/2));      % orthogonal matrix of DCT-IV
plot(S); title('columns of S');                % shape of all column vectors of A
```



```
S'*S,                                                   % orthogonamal?
```

```
ans = 20×20
    1.0000    0.0000   -0.0000    0.0000   -0.0000    0.0000    0.0000   -0.0000 ⋯
    0.0000    1.0000   -0.0000   -0.0000    0.0000    0.0000   -0.0000    0.0000
   -0.0000   -0.0000    1.0000    0.0000   -0.0000   -0.0000    0.0000   -0.0000
    0.0000   -0.0000    0.0000    1.0000   -0.0000   -0.0000    0.0000    0.0000
   -0.0000    0.0000   -0.0000   -0.0000    1.0000   -0.0000    0.0000   -0.0000
    0.0000    0.0000   -0.0000   -0.0000   -0.0000    1.0000    0.0000   -0.0000
    0.0000   -0.0000    0.0000    0.0000    0.0000    0.0000    1.0000    0.0000
   -0.0000    0.0000   -0.0000    0.0000   -0.0000   -0.0000    0.0000    1.0000
    0.0000    0.0000   -0.0000    0.0000   -0.0000    0.0000    0.0000   -0.0000
   -0.0000   -0.0000    0.0000    0.0000   -0.0000    0.0000    0.0000    0.0000
      ⋮
```

Discrete orthogonal transform concept

A pair of discrete orthogonal transformations using pair of matrcies , direct and inverse, is defined as:

DIRECT (analysis): $\mathbf{y} = \mathbf{A}\mathbf{x} = \mathbf{S}^{*T}\mathbf{x}$,

INVERSE (synthesis): $\hat{\mathbf{x}} = \mathbf{S}\mathbf{y} = \mathbf{S}(\mathbf{A}\mathbf{x}) = \mathbf{S}\mathbf{S}^{*T}\mathbf{x} = \mathbf{x}$

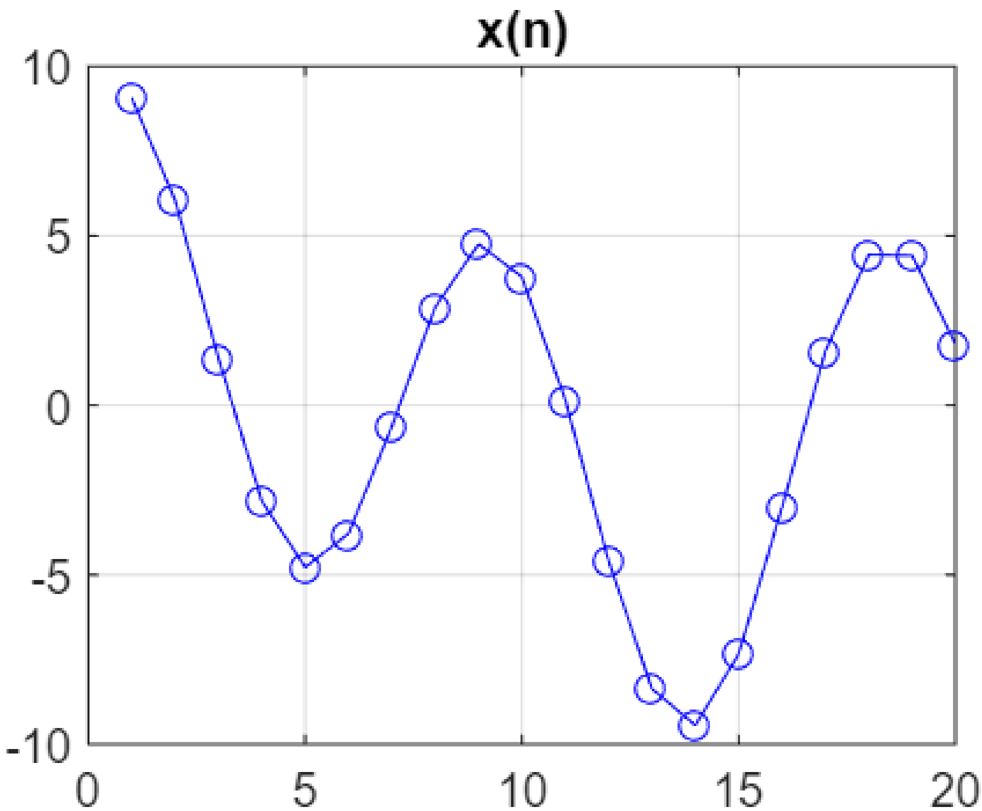
1) Elements of $\mathbf{y}$ are calculated as inner products of $\mathbf{x}$ with all rows of $\mathbf{A} = \mathbf{S}^{*T}$. These inner product measures similarity existing between them.

2) In the INVERSE transformation, vector $\hat{\mathbf{x}}$ is represented as a summations of columns of $\mathbf{S}$ scaled by elements of $\mathbf{y}$. Thefore coeffs of $\mathbf{y}$ tells us how much each column of $\mathbf{S}$ is "present" in $\hat{\mathbf{x}}$.

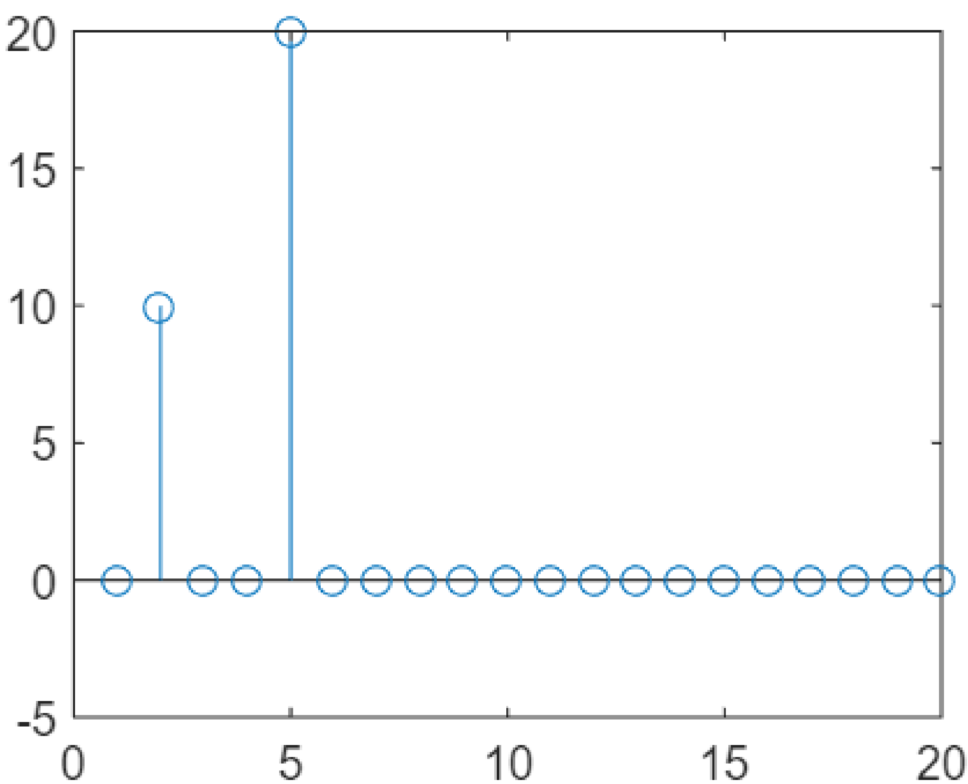
Signal analysis using direct DCT-IV

Perform DCT-IV transformation for different signals: observe their "spectra", i.e. similarity coeffs to rows of `A'`.

```
A = S';                                                 % analysis matrix
x1 = 10*S(:, 2);                                        % signal #1
x2 = 20*S(:, 5);                                        % signal #2
x3 = 30*sqrt(2/N)*cos(pi/N*(n'    +1/2)*(10.5+1/2) );  % signal #3
x4 = 30*sqrt(2/N)*cos(pi/N*(n'+N/4+1/2)*(10  +1/2) );  % signal #4
x5 = randn(1,N);                                        % signal #5
x = x1+x2;                                              % x1, x2, x3, x4, x1+x2, x1+x3, x1+x4
figure; plot(x,'bo-'); title('x(n)'); grid;             % input
```




```
y = A*x;                                    % signal analysis: finding similarity coeffs
figure; stem(y);                            % displaying similarity coefficients
```



Signal filtering using spectrum modification and inverse DCT-IV

$\mathbf{y}$ tells us about "content" of $\mathbf{x}$ as a summation of scaled columns of $\mathbf{S}$. Some component of $\mathbf{x}$ are desirable (they are parts of our signal), some others do not (they are parts of disturbances). By zeroing elements of $\mathbf{y}$ associated of unwanted components of $\mathbf{x}$ and performin the inverse transformation, we are obtaining $\hat{\mathbf{x}}$ with unwated terms removed (denoised signal or separated sigal part). To do this efficiently, all signal components have to be compactly mapped to $\mathbf{y}$ coefficients.

```
%y(5) = 0;                  % removing or not signal component x2
xhat = S*y;                 % signal synthesis: summation of weighted elementary vectors
figure; plot(xhat,'bo-'); title('xhat(n)'); grid;  % output
error = max(abs(x-xhat)),    % signal reconstruction error
```

Record your own speech. Calculate its DCT. Modify it, e.g. put zeros to some y(k). Perform inverse DCT. Listean to the synthesized speech.