

NAMA KLOMPOK 8

1. AHMAD RUDIYANTO (21157201114)
2. DIAN RAHMAWATI (21157201116)

Persamaan Perbedaan dan Pemfilteran.....

MATLAB memiliki fungsi untuk bekerja dengan persamaan dan filter perbedaan. Ini fungsi beroperasi terutama pada vektor. Vektor digunakan untuk menyimpan sinyal data sampel, atau urutan, untuk pemrosesan sinyal dan analisis data. Untuk sistem multi-input, setiap baris matriks sesuai dengan a titik sampel dengan setiap input muncul sebagai kolom matriks.

function y = filter(b,a,x)

memproses data dalam vektor x dengan filter yang dijelaskan oleh vektor a dan b, membuat data yang difilter y.

Perintah filter dapat dianggap sebagai implementasi yang efisien dari persamaan perbedaan. Struktur filter adalah filter delay-line umum yang disadap dijelaskan oleh persamaan perbedaan di bawah ini, di mana n adalah indeks arus sampel, na adalah urutan polinomial yang dijelaskan oleh vektor a dan nb adalah urutan dari polinomial yang dijelaskan oleh vektor b. output y(n), adalah kombinasi linier dari input saat ini dan sebelumnya,, x(n) x(n-1) ..., dan output sebelumnya,y(n-1) y(n-2) ...

$$a(1)y(n) = b(1)x(n) + b(2)x(n-1) + \dots + b(nb)x(n-nb+1) \\ - a(2)y(n-1) - \dots - a(na)y(n-na+1)$$

Misalkan, kita ingin memuluskan data jumlah lalu lintas kita dengan bergerak filter rata-rata untuk melihat arus lalu lintas rata-rata selama jendela 4 jam. Proses ini diwakili oleh persamaan perbedaan

$$y(n) = \frac{1}{4}x(n) + \frac{1}{4}x(n-1) + \frac{1}{4}x(n-2) + \frac{1}{4}x(n-3)$$

Vektor yang sesuai adalah

$$a = 1; b = [1/4 \ 1/4 \ 1/4 \ 1/4];$$

CATATAN : Masukkan perintah format, format rat, untuk menampilkan dan memasukkan data menggunakan format rasional.

Menjalankan jumlah beban perintah.dat membuat jumlah matriks di Workspace.

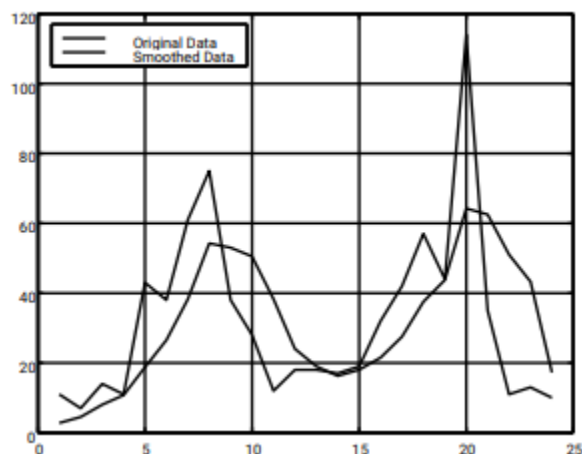
Untuk contoh ini, ekstrak kolom pertama jumlah lalu lintas dan tetapkan ke vektor x.

```
x = count(:,1);
```

Rata-rata bergerak 4 jam dari data dihitung secara efisien dengan `y = filter(b,a,x);`

Bandingkan data asli dan data yang dihaluskan dengan plot yang dilapis dari dua kurva.

```
t = 1:panjang(x); plot(t,x,'-.',t,y,'-'), grid on legend('Data Asli','Data yang Dihaluskan',2)
```



Data yang difilter yang diwakili oleh garis padat adalah rata-rata bergerak 4 jam dari jumlah lalu lintas yang diamati data yang diwakili oleh garis putus-putus.

Untuk aplikasi pemfilteran praktis, Kotak Alat Pemrosesan Sinyal mencakup banyak fungsi untuk merancang dan menganalisis filter.

Analisis Fourier dan Transformasi Fourier Cepat (FFT).....

Analisis Fourier sangat berguna untuk analisis data, karena memecah sinyal menjadi sinusoid penyusun dari frekuensi yang berbeda. Untuk data vektor sampel, Fourier analisis dilakukan menggunakan transformasi Fourier (DFT) diskrit.

Transformasi Fourier (FFT) yang cepat adalah algoritma yang efisien untuk menghitung DFT dari a urutan; itu bukan transformasi yang terpisah. Ini sangat berguna di bidang-bidang seperti pemrosesan sinyal dan gambar, di mana penggunaannya berkisar dari penyaringan, konvolusi, dan analisis frekuensi untuk memperkirakan spektrum daya.

Bagian ini:

- Meringkas fungsi transformasi Fourier

- Memperkenalkan analisis transformasi Fourier dengan contoh tentang aktivitas bintang matahari
- Menghitung besar dan fase data yang ditransformasikan
- Membahas ketergantungan waktu eksekusi pada panjang transformasi

Ringkasan Fungsi.....

MATLAB menyediakan kumpulan fungsi untuk komputasi dan bekerja dengan Fourier Mengubah.

FFT Function Summary

Function	Description
fft	Discrete Fourier transform.
fft2	Two-dimensional discrete Fourier transform.
fftn	N-dimensional discrete Fourier transform.
ifft	Inverse discrete Fourier transform.
ifft2	Two-dimensional inverse discrete Fourier transform.
ifftn	N-dimensional inverse discrete Fourier transform.
abs	Magnitude.

FFT Function Summary (Continued)

Function	Description
angle	Phase angle.
unwrap	Unwrap phase angle in radians.
fftshift	Move zeroth lag to center of spectrum.
cplxpair	Sort numbers into complex conjugate pairs.
nextpow2	Next higher power of two.

Perkenalan.....

Untuk panjang N urutan input x, DFT adalah vektor N panjang, X. fft dan ifft menerapkan hubungan

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi(k-1)\left(\frac{n}{N}\right)} \quad 1 \leq k \leq N$$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j2\pi(k-1)\left(\frac{n}{N}\right)} \quad 1 \leq n \leq N$$

Note : Karena elemen pertama vektor MATLAB memiliki indeks 1, maka penjumlahan dalam persamaan di atas adalah dari 1 hingga N. Ini menghasilkan identic hasil sebagai persamaan Fourier tradisional dengan penjumlahan dari 0 hingga N-1.

Jika $x(n)$ itu nyata, kita dapat menulis ulang persamaan di atas dalam hal penjumlahan fungsi sinus dan kosinus dengan koefisien nyata

$$x(n) = \frac{1}{N} \sum_{k=1}^N a(k) \cos\left(\frac{2\pi(k-1)(n-1)}{N}\right) + b(k) \sin\left(\frac{2\pi(k-1)(n-1)}{N}\right)$$

$$\text{where } a(k) = \text{real}(X(k)), \quad b(k) = -\text{imag}(X(k)), \quad 1 \leq k \leq N$$

Menemukan FFT.....

FFT vektor kolom x x

= [4 3 7 -9 1 0 0 0]'; ditemukan dengan

y = fft(x) yang menghasilkan

y = 6.0000

11.4853 - 2.7574i -2.0000 -

12.0000i

-5.4853 +11.2426i 18.0000

-5.4853 -11.2426i

-2.0000 +12.0000i

11.4853 + 2.7574i

Perhatikan bahwa meskipun urutan x itu nyata, y itu kompleks. Komponen pertama dari data yang ditransformasikan adalah kontribusi konstan dan elemen kelima sesuai dengan frekuensi Nyquist. Tiga nilai terakhir y sesuai dengan frekuensi negatif dan, untuk urutan nyata x, mereka adalah konjugat kompleks dari tiga komponen di paruh pertama y.

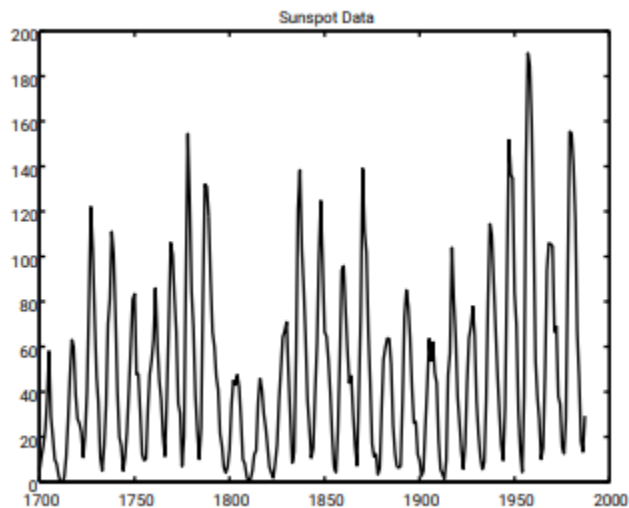
Contoh: Menggunakan FFT untuk Menghitung Periodisitas Bintik Matahari Misalkan, kita ingin menganalisis variasi aktivitas bintik matahari selama 300 terakhir Tahun. Anda mungkin menyadari bahwa aktivitas bintik matahari adalah siklus, mencapai maksimal setiap 11 tahun sekali. Mari kita konfirmasi itu.

Para astronom telah mentabulasi kuantitas yang disebut nomor Wolfer untuk hampir 300 Tahun. Kuantitas ini mengukur jumlah dan ukuran bintik matahari.

```

Memuat dan memplot data bintik matahari
memuat bintik matahari.dat tahun =
bintik matahari(:,1); wolfer =
bintik matahari(:,2);
plot (tahun, wolfer)
title('Sunspot Data')

```



Sekarang ambil FFT dari data bintik matahari.

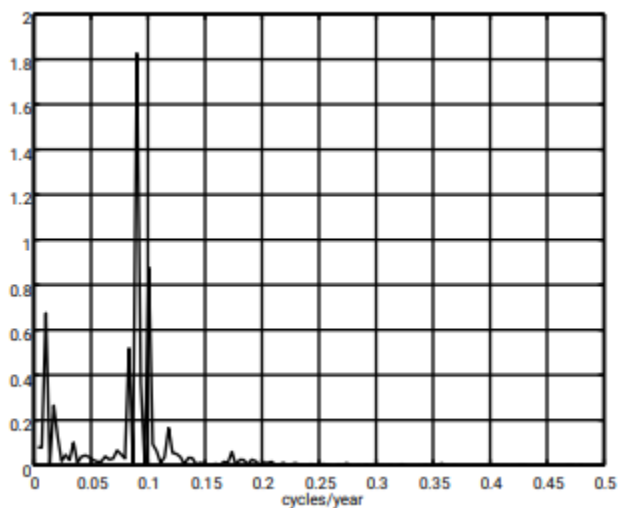
```
Y = fft(wolfer);
```

Hasil dari transformasi ini adalah vektor kompleks, Y. Besarnya Y kuadrat disebut kekuatan dan plot kekuatan versus frekuensi adalah "periodogram." Hapus komponen pertama Y, yang hanyalah jumlah dari data, dan plot hasilnya.

```

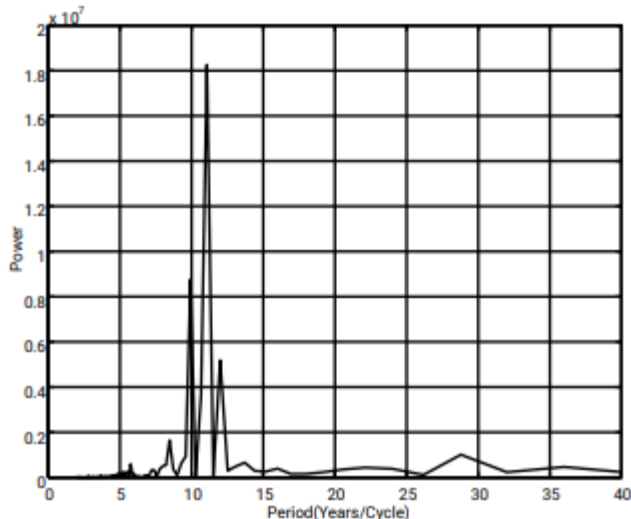
N = length(Y); Y(1) = []; power =
abs(Y(1:N/2)).^2; nyquist = 1/2; freq =
(1:N/2)/(N/2)*nyquist; plot(freq,power), grid on
xlabel('cycles/year') title('Periodogram')
x 10 7

```



Skala dalam siklus / tahun agak tidak nyaman. Mari kita plot dalam tahun / siklus dan perkirakan apa itu satu siklus. Untuk kenyamanan, plot kekuatan versus periode (di mana periode = $1./\text{freq}$) dari 0 hingga 40 tahun/siklus.

periode = $1./\text{freq}$; plot (periode, daya), sumbu ([0 40 0 2e7]), kisi pada ylabel('Power') xlabel('Periode(Tahun/Siklus)')



Untuk menentukan siklus dengan lebih tepat,
`[mp,index] = max(power); periode(indeks)`
`ans = 11.0769`

Besarnya dan Fase Data yang Diubah.....

Informasi penting tentang urutan yang diubah mencakup besarnya dan fase. Fungsi MATLAB `abs` dan `angle` menghitung informasi ini. Untuk mencoba ini, buat vektor waktu `t`, dan gunakan vektor ini untuk membuat urutan `x` terdiri dari dua sinusoid pada frekuensi yang berbeda.

```
t = 0:1/100:10-1/100; x = sin(2*pi*15*t) +  
sin(2*pi*40*t);
```

Sekarang gunakan fungsi `fft` untuk menghitung DFT urutan. Kode di bawah ini menghitung besarnya dan fase dari urutan yang ditransformasikan. Ini menggunakan fungsi `abs` untuk mendapatkan besarnya data, fungsi `angle` untuk mendapatkan informasi fase, dan `unwrap` untuk menghapus lompatan fase lebih besar dari π ke mereka

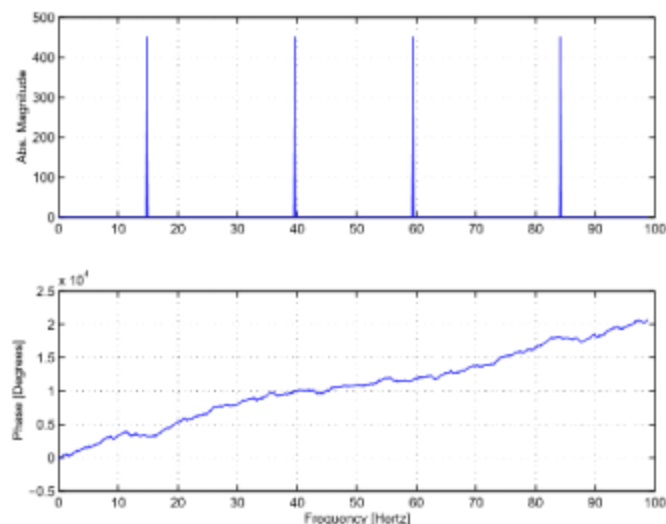
```
2 * pi pelengkap.  
y = fft(x); m = abs(y);  
p = unwrap(angle(y));
```

Sekarang buat vektor frekuensi untuk sumbu x dan plot besarnya dan fase.

```
f = (0:length(y)-1)*100/length(y); subplot(2,1,1), plot(f,m), ylabel('Abs. Magnitude'), kisi-kisi  
pada  
subplot(2,1,1), plot(), ('Abs. Magnitude'), kisi-kisi pada  
subplot(2,1,2), plot(f,p/pi), ylabel('Fase')
```

subplot(2,1,2), plot(*180/pi) y('Fase
[Derajat]'), kisi pada xlabel('Frekuensi [Hertz]')

Plot besarnya simetris sempurna tentang frekuensi Nyquist 50 hertz. Informasi yang berguna dalam sinyal ditemukan di kisaran 0 hingga 50 hertz.



Panjang FFT Versus Kecepatan.....

Anda dapat menambahkan argumen kedua ke fft untuk menentukan sejumlah poin n untuk ubah $y = \text{fft}(x, n)$

Dengan sintaks ini, fft pad x dengan nol jika lebih pendek dari n, atau memotongnya jika lebih panjang dari n. Jika Anda tidak menentukan n, fft default ke panjang input Urutan.

Waktu eksekusi untuk fft tergantung pada panjang transformasi. Ini tercepat untuk kekuatan dua. Ini hampir secepat untuk panjang yang hanya memiliki prima kecil Faktor. Biasanya beberapa kali lebih lambat untuk panjang yang prima atau yang memiliki faktor prima yang besar. Fungsi FFT terbalik ifft juga menerima argumen panjang transformasi. Untuk aplikasi praktis FFT, Kotak Alat Pemrosesan Sinyal mencakup banyak fungsi untuk analisis spektral