

Wykład VI

Aproksymacja i nadokreślone układy równań

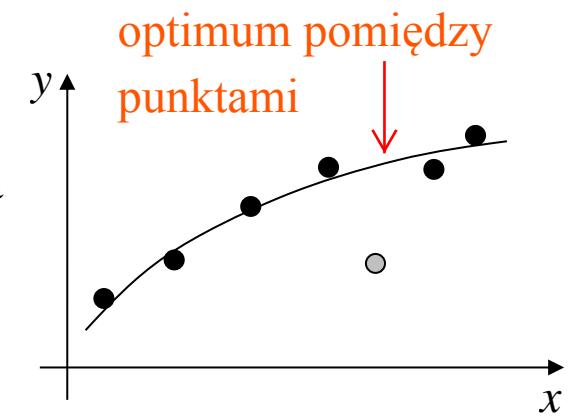
- Zadania dopasowania krzywej i przybliżenia przebiegu funkcji
- Aproksymacja średniokwadratowa i wielomiany ortogonalne
- Aproksymacja jednostajna (min-max) i wielomiany Czebyszewa
- Projektowanie filtrów cyfrowych metodami aproksymacji
- Aproksymacja Padégo funkcjami wymiernymi

Zadanie dopasowania funkcji ciągłej do danych dyskretnych

Powszechnie w badaniach eksperymentalnych. Najczęściej dane wejściowe są znane dokładnie a dane wyjściowe są zaburzone z powodu natury statystycznej (parametr próbki losowej) lub zakłócenia pomiarowego (np. szумy cieplne wzmacniacza).

Celem jest określenie spараметryzowanej zależności między danymi wejściowymi i wyjściowymi. Rozwiążanie polega na wyborze klasy zależności (np. liniowa czy kwadratowa) i parametrów tej zależności (np. współczynniki prostej).

Kryterium dopasowania (*least-squares, least-moduli, minimax*) ma wpływ na rozwiązanie i jego wrażliwość na „odstające” dane (*outliers*).



Zadanie przybliżenia przebiegu funkcji

Sformułowanie bliskie interpolacji – funkcja aproksymująca równa aproksymowanej w zadanych węzłach, poza węzłami aproksymacja ma być dobrym przybliżeniem wg wybranego kryterium.

Celem takiego postępowania jest zastąpienie czasochłonnej obliczeniowo funkcji jej dobrym i szybko obliczanym przybliżeniem (np. wielomianem lub ilorazem wielomianów). Powszechna praktyka w kalkulatorach (np. funkcja *sin*) oraz w bibliotekach i środowiskach numerycznych (np. *erf* w Matlabie).

Pożądany jest równomiernie rozłożony błąd przybliżenia – stosowane kryterium min-max.

Zastosowania aproksymacji

- **Redukcja wymiarowości opisu danych**, np. tabela wartości jest zastępowana kilkoma współczynnikami aproksymacji.
- **Odszumianie sygnałów** – konieczne założenie o klasie sygnału, dopasowanie parametrów (np. dopasowanie amplitudy, fazy i częstotliwości sygnału sinusoidalnego do spróbkowanego napięcia sieciowego)
- **Regresja liniowa i nieliniowa** – wyznaczanie zależności wiążących dane (np. wejściowe i wyjściowe). Przykładowo, określanie zależności między średnią temperaturą otoczenia a średnim spalaniem silnika samochodu.
- **Identyfikacja obiektów dynamicznych metodą dopasowania odpowiedzi czasowych lub częstotliwościowych (*nonlinear parameter estimation*)**. Np. określanie parametrów mechanicznych i elektrycznych silnika prądu stałego na podstawie zarejestrowanych sygnałów napięcia, prądu i prędkości obrotowej podczas rozruchu.
- **Szybkie obliczenia trudnych funkcji**, jak np. całki nieanalityczne, funkcje trygonometryczne.

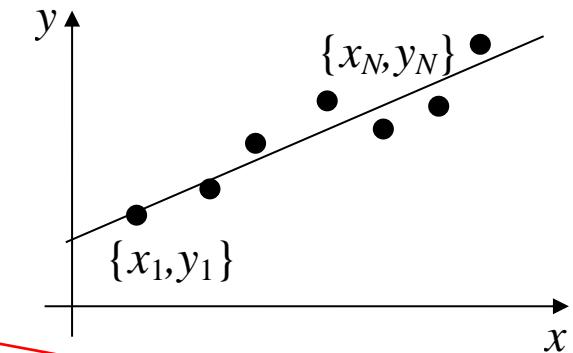
Prosty przypadek na początek

Regresja liniowa – dopasowanie prostej do danych wg kryterium najmniejszej sumy kwadratów odchyłek.

Dla zadanych par wartości $\{x_i, y_i\}, i = 1, \dots, N$ poszukujemy modelowej zależności pomiędzy nimi w postaci $y_i = ax_i + b + e_i$, gdzie e jest odchyłką, a i b parametrami modelu. Parametry mają minimalizować sumę kwadratów odchyłek pomiędzy zadanym y_i i wyliczonym z modelu $\hat{y}_i = ax_i + b$, czyli kryterium:

minimaliz.
kryterium

$$J_1 = \sum_{i=1}^N e_i^2 = \sum_{i=1}^N (y_i - \hat{y}_i)^2 = \sum_{i=1}^N (y_i - ax_i - b)^2 = J_1(a, b)$$



Minimum kryterium znajdziemy obliczając pochodne cząstkowe i przyrównując je do zera ($\frac{\partial J_1}{\partial a} = 0, \frac{\partial J_1}{\partial b} = 0$). $a, b = ?$

układ
równ.

$$\begin{cases} \bar{y} = a\bar{x} + b \\ \bar{xy} = b\bar{x} + a\bar{x^2} \end{cases}$$

gdzie: $\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i, \bar{y} = \frac{1}{N} \sum_{i=1}^N y_i, \bar{xy} = \frac{1}{N} \sum_{i=1}^N x_i y_i, \bar{x^2} = \frac{1}{N} \sum_{i=1}^N x_i^2$

Rozwiązuje powyższy układ równań ze względu na poszukiwane (estymowane) parametry otrzymujemy:

$$a = \frac{\bar{xy} - \bar{x}\bar{y}}{\bar{x^2} - \bar{x}^2} \quad b = \bar{y} - a\bar{x} \quad \text{rozwiązanie}$$

Warunek na minimum (dodatnia określoność macierzy drugich pochodnych) jest spełniony (sprawdzić).

Jakiego typu funkcją (powierzchnią) w przestrzeni (a, b) jest J_1 ? Czy jej minimum jest unikalne?

Przykłady Trend zmian

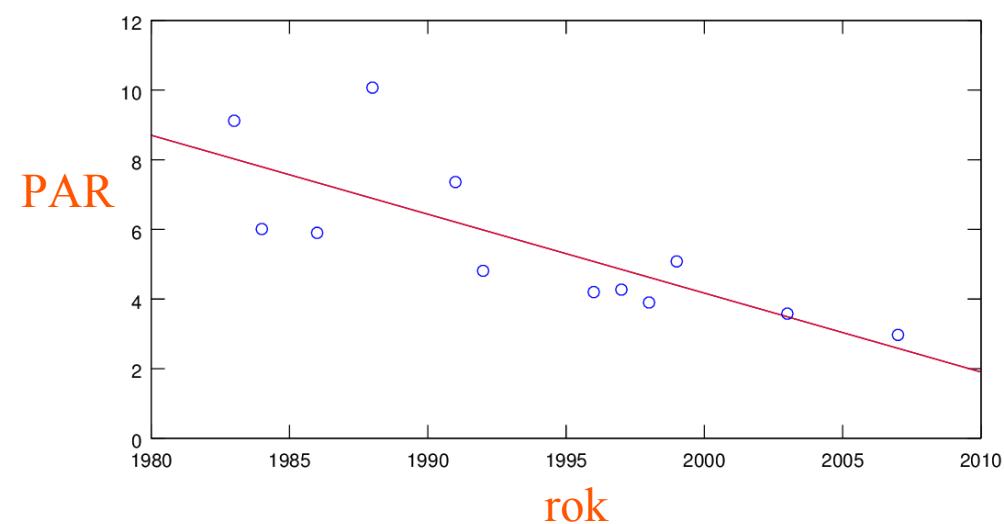
Regresja liniowa jest często używana do ilustrowania trendów zmian w zjawiskach. Kluczem do czytelnego przedstawienia trendu jest wybór takiej cechy zjawiska, która uwypukla zmiany.

Przykład Głośność muzyki (*loudness war*)

Wraz z przejściem na cyfrowy zapis muzyki (lata 80-te XX wieku) studia nagraniowe zaczęły stosować narzędzia DSP do produkcji muzyki i powszechnie stało się poprawianie oryginalnego brzmienia. Słyszalna jakościowa zmiana na przestrzeni ostatnich lat to zmniejszanie zakresu dynamiki dźwięku (zróżnicowanie na ciche i głośne dźwięki) poprzez nieliniowe „upakowanie” dźwięków w górnych granicach zakresu (nazywane kompresją dynamiki, nie mylić z kompresją audio typu MP3). Zobaczmy jak to zjawisko, nazywane *loudness war*, przebiegało w kolejnych produkcjach zespołu Metallica. Cechą prezentowaną jest PAR (peak to average ratio, wartość maksymalna do średniej wyprostowanej) dla drugich utworów z kolejnych płyt.

Jeszcze wyraźniej widać to zjawisko dla tych samych utworów pochodzących z kolejnych edycji po tzw. *remasteringu*. Przykłady można łatwo odnaleźć wśród utworów z tzw. głównego nurtu.

Inne przykłady trendów:
zobacz [prawo Moore'a](#) i pokrewne.



Sformułowanie ogólniejsze – liniowa regresja wielowymiarowa

Poszukujemy zależności liniowej między wieloma zmiennymi wejściowymi i pojedynczą zmienną wyjściową. Przyjmując oznaczenia macierzowe:

$$\mathbf{X} = \begin{bmatrix} 1 & x_1^{(1)} & \dots & x_1^{(n)} \\ 1 & x_2^{(1)} & \dots & x_2^{(n)} \\ \vdots & \vdots & & \vdots \\ 1 & x_N^{(1)} & \dots & x_N^{(n)} \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}, \quad \mathbf{a} = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix}, \quad \hat{\mathbf{y}} = \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_N \end{bmatrix}, \quad \mathbf{e} = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix} = \mathbf{y} - \hat{\mathbf{y}}$$

zależność między danymi przybierze postać macierzowego układu równań: $\mathbf{y} = \mathbf{Xa} + \mathbf{e}$

Nie można go rozwiązać w zwykłym sensie, ponieważ $\mathbf{y} = \mathbf{Xa}$ to sprzeczny i nadokreślony układ równań. Możemy podać rozwiązanie spełniające wszystkie równania z małym łącznym błędem.

Minimalizowana suma kwadratów odchyłek przyjmuje w zapisie macierzowym postać:

$$J_1 = \mathbf{e}^T \mathbf{e} = (\mathbf{y} - \mathbf{Xa})^T (\mathbf{y} - \mathbf{Xa}) = \mathbf{y}^T \mathbf{y} - 2\mathbf{a}^T \mathbf{X}^T \mathbf{y} + \mathbf{a}^T \mathbf{X}^T \mathbf{X} \mathbf{a} = J_1(\mathbf{a})$$

Różniczkując $J_1(\mathbf{a})$ względem wektora \mathbf{a} i przyrównując do zera:

$$\frac{\partial J_1}{\partial \mathbf{a}} = -2\mathbf{X}^T \mathbf{y} + 2\mathbf{X}^T \mathbf{X} \mathbf{a} = \mathbf{0} \quad \text{skąd} \quad \mathbf{X}^T \mathbf{X} \mathbf{a} = \mathbf{X}^T \mathbf{y} \quad (\text{równanie normalne})$$

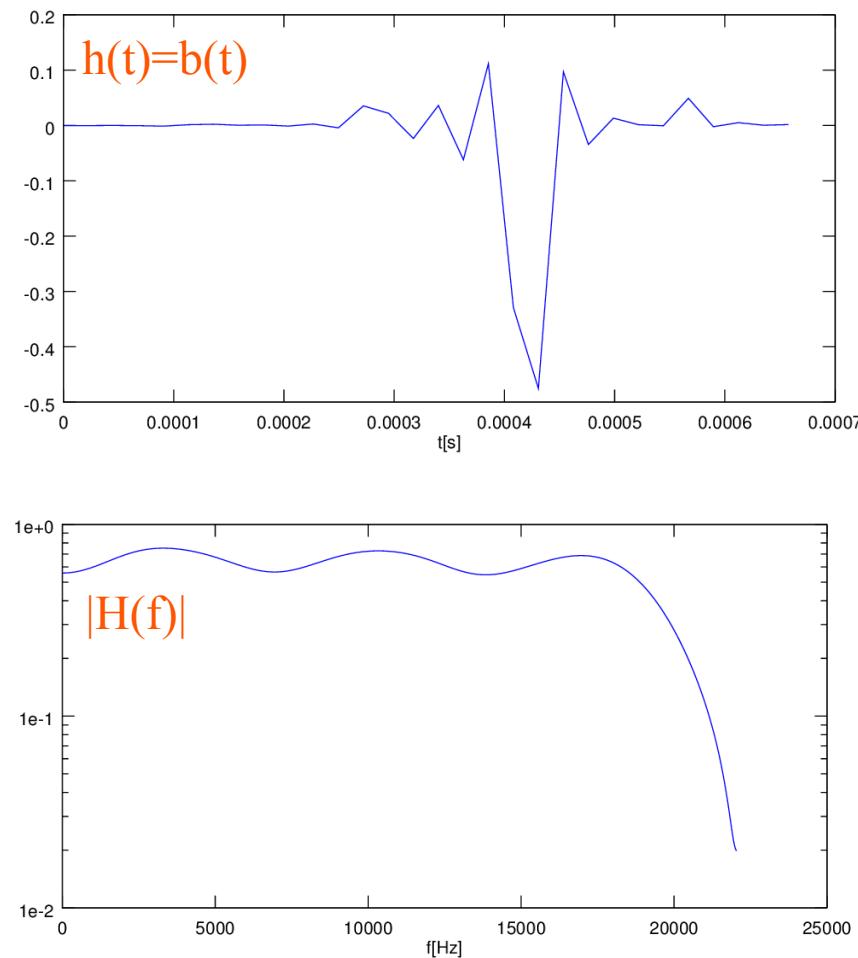
otrzymujemy: $\mathbf{a} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$ - klasyczny wzór na estymator LS (*Least Squares*).

Jaką krzywą jest $J_1(\mathbf{a})$? Czy jej minimum jest unikalne?

Przykład Identyfikacja odpowiedzi impulsowej kanału audio

Powróćmy do tematu identyfikacji odpowiedzi impulsowej z użyciem sygnału PRBS. Tym razem będziemy identyfikować odpowiedź toru przykładowej karty dźwiękowej SoundBlaster z analogowym połączeniem zwrotnym kablem jack-jack. Poniżej program i wyniki do rozwinięcia na zajęciach w laboratorium.

```
load sndb_prbs;
M=30; % dlugosc licznika
L=length(u);
% model FIR o dlugosci M
U=[];
for i=1:M
    U=[U, u(M-i+1:L-i+1)];
end
Y=y(M:L);
b=U\Y;
Hls=freqz(b(1:M), [1], L/2);
ids=1:L/2;
Freqs=(ids-1)*fs/L;
semilogy(Freqs, abs(Hls), 'b-')
```

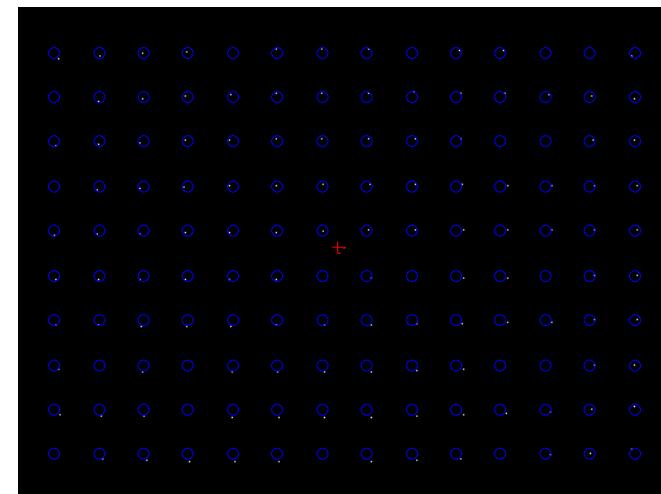
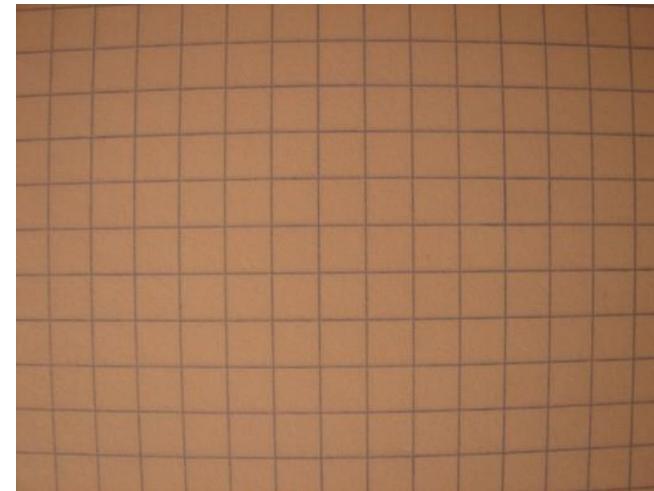


Przykład: Korekcja zniekształceń kamery cyfrowej

Przeanalizujmy program korygujący zniekształcenia beczkowe zdjęć z aparatu zrobionych w trybie makro. Opieramy się na pokazanym obok testowym obrazie kratowym. Problem rozwinie my na ćwiczeniach.

```
i1=imread('Krata1.jpg');
i1g=double(rgb2gray(i1));
[sx, sy]=size(i1g);
cx=sx/2;
cy=sy/2;

% wyszukanie przecięć linii
wzorzec=[
0 0 0 1 1 1 0 0 0
0 0 0 1 1 1 0 0 0
0 0 0 1 1 1 0 0 0
1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1
0 0 0 1 1 1 0 0 0
0 0 0 1 1 1 0 0 0
0 0 0 1 1 1 0 0 0
];
ic=normxcorr2(wzorzec,i1g);
ig=ic<-0.7;
ig=bwmorph(ig,'shrink',Inf);
[ix,iy]=find(ig>0);
r=sqrt((ix-cx).^2+(iy-cy).^2);
[rs,i]=sort(r);
```



```
% ustalenie punktów odniesienia (siatki regularnej)
x0=ix(i(1)); y0=iy(i(1));
dx=abs(ix(i(1))-ix(i(4)))-1;
dy=abs(iy(i(1))-iy(i(4)))-3;
jx=x0+dx*[-4:1:5];
jy=y0+dy*[-6:1:7];
[jX, jY]=meshgrid(jx,jy);
jX=jX'; jx=jX(:);
jY=jY'; jy=jY(:);
rm=sqrt((jx-cx).^2+(jy-cy).^2);
[rms,j]=sort(rm);
```

```
imshow(ig), hold on
plot(cy,cx,'r+') % punkt centralny
plot(jY,jX,'bo') % punkty odniesienia
```

```
% identyfikacja modelu algorytmem LS
% uzupełnij wyliczanie modelu stopnia 3 zależności rs od rms
% wartości wyliczone z modelu zapisz do zmiennej R
```

???

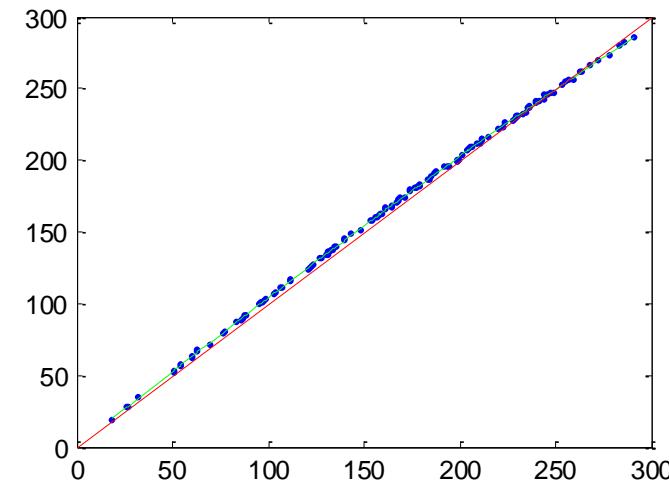
??? A(1), A(2), A(3) =?

???

```
figure, plot(rms,rs,'b.', rms,R,'g-', [0 3e2], [0 3e2], 'r-')
```

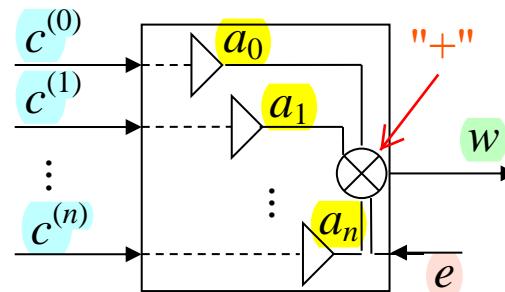
```
% prostowanie obrazka
```

```
xx=1:sx;
yy=1:sy;
[XX,YY]=meshgrid(xx,yy);
RR=sqrt((XX-cx).^2+(YY-cy).^2);
RRr=RR*A(1)+RR.^2*A(2)+RR.^3*A(3);
RRu=RR./RR;
ilr=interp2(ilg,(YY-cy).*RRu+cy,(XX-cx).*RRu+cx);
figure, subplot(2,1,1), imagesc(ilg), subplot(2,1,2), imagesc(ilr), colormap gray
```



Od estymatora LS do aproksymacji średniokwadratowej

Zastosowanie estymatora LS wykracza poza regresję liniową. Poszczególne „wejścia” w macierzy \mathbf{U} mogą być widziane jako składowe (elementy bazy), które skalowane poprzez współczynniki i sumowane dają najlepsze przybliżenie (aproksymację) wielkości „wyjściowej” (wzorca) w sensie sumy kwadratów odchyłek. Możemy to poglądotwo przedstawić w postaci blokowej:



$c^{(0)}, \dots, c^{(n)}$ – elementy składowe bazy
 a_0, \dots, a_n – “amplitudy” składowych
 w – aproksymowany „wzorzec”
 e – błąd aproksymacji

Dla przykładu zapiszmy problem aproksymacji wielomianem w postaci jednomianów do stopnia n na zbiorze N danych $\{x_i, y_i\}$, $i = 1, \dots, N$, tj. $c^{(k)} = \{x_i^k\}$, $w = \{y_i\}$, $J_1 = \sum_{i=1}^N (y_i - \sum_{k=0}^n a_k x_i^k)^2$

$$\begin{bmatrix} 1 & x_1 & \dots & x_1^n \\ 1 & x_2 & \dots & x_2^n \\ \vdots & \vdots & & \vdots \\ 1 & x_N & \dots & x_N^n \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix}$$

$c^{(1)}$

```
% Przykład aproksymacji wielomianowej z bazą jednomianów
% {x,y} – dane, N - ilość danych, n – stopień wielomianu
U=vander(x);
U=U(:,N-n:N);
a=inv(U'*U)*U'*y; % współczynniki od najwyższej potęgi
% Inaczej: a=U\y; albo a=polyfit(x,y,n);
plot(y-U*a) % wyrysuj wektor odchyłek
polyval(a,1) % wartość wielomianu w wybranym punkcie
```

Baza jednomianów jest niekorzystna (jak już wiemy z tematu interpolacji) z dwóch powodów:

- macierz układu nadokreślonego (o strukturze Vandermonde'a) jest źle uwarunkowana już dla niskiego stopnia wielomianu aproksymacyjnego
- nakład obliczeniowy na wyznaczenie współczynników jest duży i powtarzalny przy zwiększaniu stopnia wielomianu

Wybór bazy jest problemem doboru postaci składowych rozwinięcia dla osiągnięcia małego błędu przy małym nakładzie obliczeniowym. Ten cel osiągniemy wtedy gdy składowe bazy (kolumny macierzy \mathbf{X}) będą ortonormalne, tj.:

$$\sum_{i=1}^N c_i^{(k)} c_i^{(l)} = \begin{cases} 1, & k = l \\ 0, & k \neq l \end{cases} \quad \text{ortogonalność kolumn macierzy}$$

Wtedy: $\mathbf{a} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} = \mathbf{I} \mathbf{X}^T \mathbf{y}$, czyli parametry są obliczane jako transformacja macierzowa $\mathbf{a} = \mathbf{X}^T \mathbf{y}$, lub w zapisie skalarnym: $a_k = \sum_{i=1}^N c_i^{(k)} y_i, \quad k = 0, \dots, n$ iloczyn skalarny: baza*dane

Popularne bazy ortonormalne to (normalizowane):

- wielomiany trygonometryczne (baza Fouriera) – dobre przybliżenie dla funkcji okresowych
- wielomiany Czebyszewa – najmniejsze oscylacje w przedziale $[-1,1]$
- wielomiany Legendre'a (pojawiły się przy kwadraturach Gaussa)

Problem doboru bazy do aproksymowanego kształtu – szybkość zbieżności szeregu.

Przykład Aproksymacja średniokwadratowa wielomianem trygonometrycznym

Projektowanie okresowego sygnału pobudzającego do identyfikacji obiektu dynamicznego.

Wymagane pasmo sygnału ograniczone do n prążków i najwierniejsze oddanie zadanego kształtu na N próbkach do zaprogramowania generatora ARB.

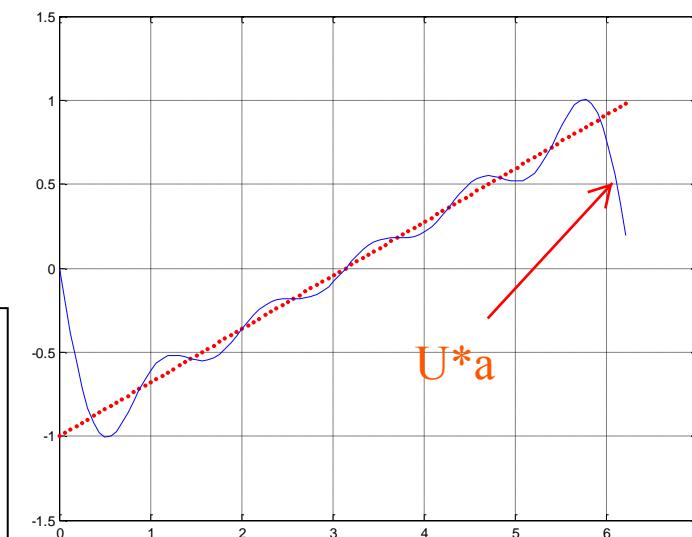
Spróbujmy przybliżyć sygnał piłokształtny szeregiem sinusów, tzn. zminimalizować kryterium o postaci:

$$J_1 = \sum_{i=1}^N \left(y_i - \sum_{k=0}^n a_k \sin(k\omega_0 t_i) \right)^2$$

```

N=100; % Liczba definiowanych punktów czasowych
n=5; % Liczba sinusoid składowych do aproks.
t=[0:2*pi/N:2*pi*(N-1)/N];
y=sawtooth(t);% Zadany kształt piłokształtny
U=sin(t*[1:n])*sqrt(2/N); % U - macierz z sin() w kolumnach
a=U*y;
plot(t,y,'r.',t,U*a)

```



Próba powyższej aproksymacji dla przebiegu zadanego $\text{sawtooth}(t-\pi/4)$ zakończy się niepowodzeniem, bo ten przebieg nie jest dobrze przybliżany bazą sinusów. Musimy użyć łączonej bazy sinusów i kosinusów. W wersji zespolonej jest to baza DFT.

% Równoważne sposoby tworzenia pełnej rzeczywistej bazy trygonometrycznej

U1=[sin(t*[1:n]) cos(t*[1:n])] *sqrt(2/N);
U2=dftmtx(N)*sqrt(2/N); U2=[imag(U2(:,2:n+1)) real(U2(:,2:n+1))]; najpierw sin(), potem cos() w kolumnach

Uwagi o aproksymacji zbioru danych funkcją nieliniową względem parametrów

Powszechnie występujące w naukach eksperymentalnych zadanie dopasowania krzywej do zbioru danych pomiarowych (*curve fitting*) jest klasycznym zadaniem aproksymacji. Krzywa dopasowywana stanowi zależność modelową wiążącą dane, która została opracowana na podstawie analizy zjawiska generowania danych (np. zależność opisująca rozpad promieniotwórczy pierwiastków) lub została wywnioskowana z przebiegu danych.

Zastosowanie dopasowania krzywej znajdujemy w identyfikacji obiektów dynamicznych metodą dopasowania modelowej odpowiedzi na znane pobudzenie do odpowiedzi zmierzonej na obiekcie. Takie odpowiedzi dynamiczne są najczęściej nieliniowe ze względu na parametry (np. odpowiedź impulsowa $h(t)$ obiektu oscylacyjnego z parametrami K, ξ, ω_0).

Minimalizowane kryterium sumy kwadratów nie ma wtedy postaci minimalizowanej analitycznie:

$$J_1 = \sum_{i=1}^N (y_i - f(\mathbf{a}, t_i))^2$$

minimalizacja

gdzie:

y_i - zmierzona odpowiedź dynamiczna, $f(\mathbf{a}, t_i)$ - modelowa odpowiedź dynamiczna.

Wyznaczenie rozwiązania wymaga zastosowania numerycznego algorytmu iteracyjnego poszukiwania minimum kryterium (czym zajmiemy się na zajęciach z optymalizacji). Kryterium J_1 nie jest już kwadratowe względem parametrów, bo funkcja $f(\mathbf{a}, t_i)$ nie jest liniowa. Zatem kryterium może mieć, poza rozwiązaniem optymalnym, minima lokalne odpowiadające dopasowaniom suboptimalnym.

Aproksymacja wielomianami Czebyszewa w ujęciu interpolacyjnym

Na wykładzie dotyczącym interpolacji obserwowałyśmy niepożądane zachowanie wielomianu interpolującego wysokiego rzędu w postaci silnych oscylacji na końcach przedziału. Pozornie jest to sprzeczność z twierdzeniem Weierstrassa o istnieniu dla dowolnej funkcji pewnego wielomianu przybliżającego ją z dowolnie małym błędem, jednak to twierdzenie mówi o pewnym wielomianie, a nie wielomianie na węzłach równoodległych.

Dla wielomianu interpolacyjnego $p_N(x)$ na węzłach $\{x_i, f(x_i)\}, i = 0, \dots, N$ błąd między funkcją i jej przybliżeniem wielomianowym wynosi (czego nie wyprowadzamy, zob. np. Buchanan):

$$R_N(x) = f(x) - p_N(x) = (x - x_0)(x - x_1) \dots (x - x_N) \frac{f^{(N+1)}(\xi)}{(N+1)!} = L_N(x) \frac{f^{(N+1)}(\xi)}{(N+1)!}$$

Pewne wybory węzłów (np. równoodległe) mogą powodować duże wartości czynnika $L_N(x)$.

Przez odpowiedni dobór położenia węzłów możemy zminimalizować jego maksymalną wartość. Optymalne położenia węzłów na przedziale $[-1, 1]$ to zera wielomianu Czebyszewa stopnia $N+1$ równe (Mathews, str. 237):

$$x_i = \cos\left(\frac{2i+1}{2(N+1)}\pi\right), \quad i = 0, 1, \dots, N$$

Wielomiany Czebyszewa kolejnych stopni można zdefiniować rekurencyjnie:

$$C_0(x) = 1 \quad C_1(x) = x \quad C_{n+1}(x) = 2xC_n(x) - C_{n-1}(x)$$

Przykład Trudna funkcja $(1 + x^2)^{-1}$ raz jeszcze

Porównajmy jakość aproksymacji funkcji, która sprawiała problemy przy węzłach równoodległych (a która była bardzo dobrze przybliżana funkcją sklejaną za cenę mniej zwartego opisu).

% 11 węzłów Czebyszewa w przedziale [-5:5]

N=10;

xk=5*cos((2*[0:N]+1)/(2*(N+1))*pi)'

4.9491

4.5482

3.7787

2.7032

1.4087

0.0000

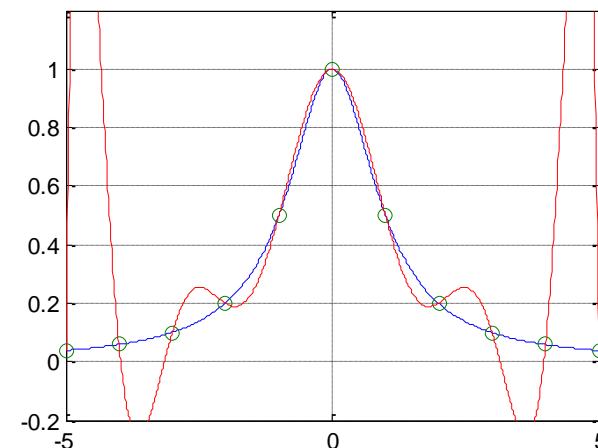
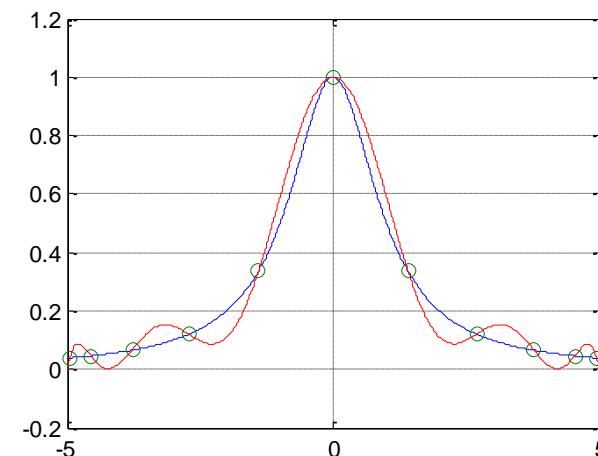
-1.4087

-2.7032

-3.7787

-4.5482

-4.9491



Podstawiając węzły Czebyszewa do postaci Newtona uzyskujemy wynik znacznie lepszy niż przy węzłach równoodległych (spójrz na rysunki obok).

Projektowanie cyfrowych filtrów FIR metodami aproksymacji

Alternatywą do projektowania filtrów FIR metodą okien są metody aproksymacji wg wybranej normy. Ogólnie, wielkością minimalizowaną jest:

$$J = \int_{\omega} \left[W(\omega) (|H(\omega)| - D(\omega)) \right]^p d\omega$$

gdzie:

$H(\omega)$ - transmitancja projektowanego filtra,

$D(\omega)$ - wzorzec charakterystyki amplitudowej (charakterystyka fazowa z założenia liniowa),

$W(\omega)$ - funkcja wagowa.

Dla $p=2$ jest to aproksymacja średniokwadratowa (*least squares*), dla $p=\infty$ aproksymacja minimax (*equiripple design*).

Funkcje Matlaba projektujące FIR na zasadzie aproksymacji:

- `remez` - algorytm Remeza aproksymacji minimax, implementacja Parks-McClellan (*exchange algorithm*, zob. Buchanan, str. 273 albo Evans, str. 168)
- `firls` - algorytm *weighted LS*
- `firlpnorm` – definiowany stopień normy (domyślnie $p=128$), algorytm *quasi-Newton search*

Aproksymacja Padé funkcjami wymiernymi

Dotychczas przy okazji interpolacji i aproksymacji cały czas **operowaliśmy węzłami** (lepiej lub gorzej rozłożonymi), które były podstawą do opisu pełniejszego (interpolacja pomiędzy węzłami) lub przyblizonego ale bardziej zwarteego (aproksymacja). **Aproksymacja Padégo funkcjami wymiernymi (ilorazem wielomianów)** postaci:

$$R_{N,M}(x) = \frac{A(x)}{B(x)} = \frac{a_0 + a_1 x + \dots + a_N x^N}{1 + b_1 x + \dots + b_M x^M}$$

przyjmuje inną filozofię przybliżenia funkcji. Spełnia ona **warunek równości jej wartości i kolejnych pochodnych z wartościami funkcji i jej kolejnych pochodnych w punkcie zerowym (lub dowolnym innym po podstawieniu zmiennych)**.

Rozwijając funkcję aproksymowaną w szereg MacLaurina wokół punktu 0:

$$f(x) = \sum_{k=0}^{\infty} \frac{1}{k!} f^{(k)}(0) x^k$$

stawiamy warunki: $R_{N,M}(0) = f(0)$ i $\left. \frac{d^k}{dx^k} R_{N,M}(x) \right|_{x=0} = \left. \frac{d^k}{dx^k} f(x) \right|_{x=0}, k = 1, 2, \dots, N + M$

co jest tożsame z porównywaniem współczynników przy kolejnych potęgach w wyrażeniu:

$$A(x) = f(x)B(x)$$

Uzyskujemy **układ $N+M+1$ równań** i wyznaczamy współczynniki wielomianów licznika i mianownika wymiernej funkcji aproksymującej (zob. np. *Numerical Recipes* albo *Mathews*).

Przykład Transformacja biliniowa pomiędzy dziedziną operatorów s i z

Zależność pomiędzy dziedziną operatora $z = e^{j2\pi \frac{f}{f_{pr}}}$ i operatora $s = j2\pi f$ powszechnie używaną w cyfrowym przetwarzaniu sygnałów (np. do transformacji filtra analogowego do postaci cyfrowej) nazywa się transformacją biliniową z powodu jej wymiernej dwuliniowej postaci:

$$z = \frac{1 + \frac{1}{2f_{pr}}s}{1 - \frac{1}{2f_{pr}}s} \quad s = 2f_{pr} \frac{z-1}{z+1}$$

przybliżającej zależność dokładną: $z = f(s) = e^{sT_p}, \quad T_p = \frac{1}{f_{pr}}$

Jest to zależność przybliżona i można ją uzyskać na zasadzie aproksymacji Padégo.

$$f(s)|_{s=0} = 1, \quad f'(s)|_{s=0} = T_p, \quad f''(s)|_{s=0} = T_p^2$$

$$\downarrow \quad a_0 + a_1 s = (1 + b_1 s)(1 + T_p s + \frac{1}{2} T_p^2 s^2), \quad a_0 = 1, \quad a_1 = \frac{1}{2f_{pr}}, \quad b_1 = -\frac{1}{2f_{pr}} \quad (\text{c.b.d.p.})$$

Przykład Przybliżenie funkcji Matlaba $\text{erf}()$.

Przykład Trudna funkcja $(1+x^2)^{-1}$ po raz ostatni

Czy jest sens stosować w tym przypadku aproksymację Padégo ?

Aproksymacja Padé daje z reguły dużo lepsze efekty niż rozwinięcie wielomianowe MacLaurina i często lepsze niż aproksymacja wielomianowa Czebyszewa. Jednak nie opracowano dotąd metody analizy błędu tego przybliżenia (zob. *Numerical Recipes*).

Podsumowanie, kluczowe elementy tego wykładu:

- aproksymujemy, żeby:
 - zapisać modelową relację między danymi wejściowymi i wyjściowymi (zakłóconymi)
 - w zwarty sposób przedstawić obszerny zbiór powiązanych danych
 - szybko lub z małym zużyciem pamięci liczyć wartości funkcji trudnych obliczeniowo
- aproksymacja liniowa według zasady najmniejszej sumy kwadratów odchyłek daje rozwiązanie zamknięte wyrażone równaniem macierzowym $\mathbf{a} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$
- aproksymacja wielomianowa Czebyszewa używa węzłów rozmiieszczonych nieregularnie dla zmniejszenia oscylacji wielomianu
- aproksymację można interpretować w kategoriach reprezentacji sygnału poprzez kombinację liniową elementów z bazy (np. wielomianowej, sinusowej/kosinusowej, DFT)
- aproksymacja Padégo ilorazem wielomianów daje często bardzo dobre efekty
- aproksymacja nieliniowa względem parametrów wymaga użycia algorytmów numerycznych minimalizacji