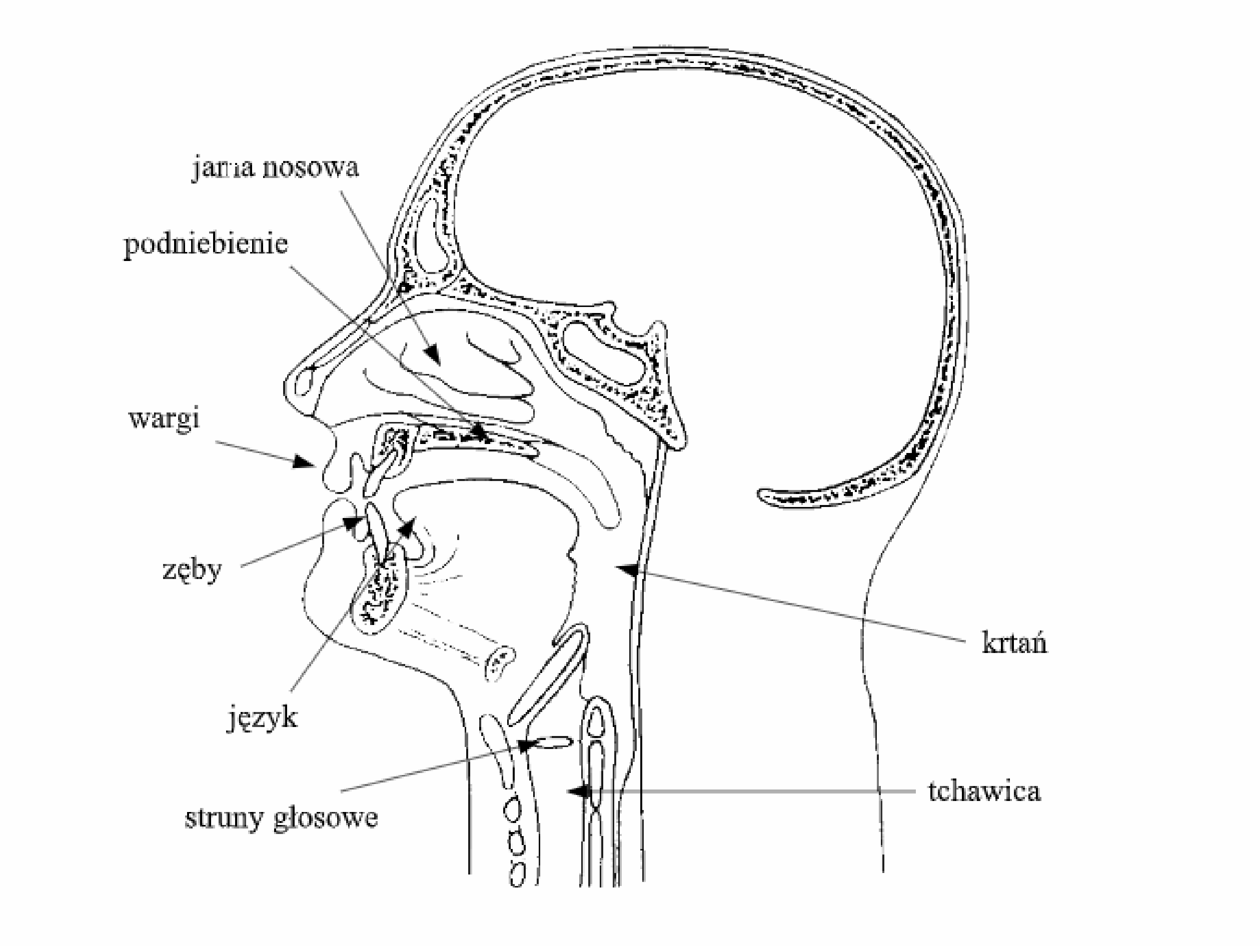
Laboratorium 8

1. Liniowe Kodowanie Predykcyjne (LPC) i zastosowania

Tor oddechowy kształtuje widmo sygnału krtaniowego powstającego przy przechodzeniu fali dźwiękowej przez struny głosowe (Rys. 1). Dokonując estymacji parametrów opisujących trakt głosowy jako układ filtrujący sygnał pobudzenia, można wyznaczyć jego charakterystykę amplitudowo–częstotliwościową.



Rys. 1. Schemat traktu głosowego

Liniowe kodowanie predykcyjne (LPC, *ang. Linear Predictive Coding*), zastosowali po raz pierwszy Saito i Itakura w 1966 roku. Model ten, oparty jest na analizie budowy narządu mowy człowieka. Transmitancja filtru predykcji liniowej jest wyrażona następującym wzorem:

$$H(z) = \frac{1}{1 - \sum_{k=1}^{p} a_k z^{-k}} \tag{1}$$

gdzie p jest liczbą współczynników, a_k jest współczynnikiem filtra predykcji liniowej o indeksie $k = 1, \ldots, p$.

Otrzymane moduły współczynników filtra predykcji liniowej $|a_1|, \ldots, |a_p|$ są **cechami** sygnału dla metody LPC. Liniowe kodowanie predykcyjne oparte jest na kombinacji liniowej p próbek. Na podstawie p poprzednich wartości sygnału akustycznego $s(n)$ przewiduje się wartości kolejne:

$$s'(n) = -\sum_{k=1}^{p} a_k \cdot s(n-k) \tag{2}$$

gdzie $s'(n)$ jest prognozą wartości sygnału w chwili *n*-tej.

Błąd między bieżącą próbką i przewidywaną można wyrazić jako:

$$err(n) = s(n) - s'(n) \tag{3}$$

Przyjmując, że N jest liczbą próbek w oknie, należy znaleźć optymalne wartości współczynników $a_1, \ldots, a_p$ takie, aby średni błąd prognozy był jak najmniejszy. Najczęściej jako kryterium jakości predykcji wybiera się błąd średniokwadratowy, zdefiniowany następująco:

$$J = \sigma^2 = \frac{1}{N-p}\sum_{n=p}^{N-1} err^2(n) = \frac{1}{N-p}\sum_{n=p}^{N-1}\left[s(n) + \sum_{j=1}^{p} a_j s(n-j)\right]^2 \tag{4}$$

Aby wyznaczyć optymalne współczynniki $a_1, \ldots, a_p$, należy obliczyć pochodną cząstkową J względem każdego z tych współczynników i przyrównać ją do zera. Otrzymujemy w ten sposób układ p równań z p niewiadomymi, który posiada następujące rozwiązanie:

$$\mathbf{a} = -\mathbf{R}^{-1}\mathbf{r} \tag{5}$$

gdzie

$$\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix}, \mathbf{R} = \begin{bmatrix} r(0) & r(1) & \cdots & r(p-1) \\ r(1) & r(0) & \cdots & r(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ r(p-1) & r(p-2) & \cdots & r(0) \end{bmatrix}, \mathbf{r} = \begin{bmatrix} r(1) \\ r(2) \\ \vdots \\ r(p) \end{bmatrix} \tag{6}$$

$$r(k) = \frac{1}{N-p}\sum_{n=p}^{N-1} s(n)s(n+k) \tag{7}$$

Znając współczynniki predykcji, można z równania (4) obliczyć jej błąd $J_{min} = \sigma_{\min}^2$:

$$J_{\min} = \sigma_{\min}{}^{2} = r(0) + a^{T} r = r(0) + \sum_{j=1}^{p} a_j r(j) \tag{8}$$

Obliczenie współczynników filtra predykcji liniowej ze wzoru (5) wymaga wyznaczenia macierzy odwrotnej $\mathbf{R}^{-1}$ dla każdej „ramki” sygnału akustycznego. Rozwiązaniem tego zadania jest zastosowaniem algorytmu Durbina-Levinsona opisanego w literaturze. Algorytm ten jest skuteczny, ponieważ potrzebuje tylko M^2 mnożeń, aby wyliczyć współczynniki filtra predykcji liniowej $a_1, \ldots, a_p$ (złożoność obliczeniowa wynosi $O(N^2)$). Współczynniki te tworzą **wektor cech** $[|a_1|, |a_2|, ..., |a_p|]$, który będzie używany w etapie klasyfikacji (zobacz wzór 1). W literaturze autorzy stosują liczby współczynników filtra predykcji liniowej od 8 do 20.

Metody z rodziny liniowego kodowania predykcyjnego – LPC. Polegają na podziale cyfrowego sygnału mowy na krótkie segmenty, które są parametryzowane. Kompresja metodą LPC polega na reprezentacji i transmisji sygnału oryginalnego w postaci współczynników filtru analizy $|a_1|, \ldots, |a_p|$. Rekonstrukcja sygnału wykonywana jest przy pomocy filtru syntezy mowy o charakterystyce odwrotnej do filtru analizy. W literaturze metoda LPC często określa się mianem parametrycznej reprezentacji mowy, gdzie filtr modeluje właściwości narządu mowy, który jest pobudzany prostym sygnałem syntetycznym.

Metoda LPC jest zwykle używana do analizy mowy i resyntezy. Jest stosowana jako forma kompresji głosu przez firmy telekomunikacyjne, na przykład w standardzie GSM. Jest również używana do bezpiecznego połączenia bezprzewodowego, w którym głos musi być zaszyfrowany i wysłany przez wąski kanał. Metody LPC można również użyć dla ekstrakcji cech.

Większość źródłowych kodeków mowy opartych jest na liniowym kodowaniu predykcyjnym (LPC – Linear Predictive Coding). Algorytmy kodeków źródłowych wykorzystują model wytwarzania sygnału mowy przez człowieka. Sygnał mowy powstaje przez przefiltrowanie pobudzenia przez trakt głosowy. W dostatecznie krótkich przedziałach czasu sygnał mowy może być traktowany jako sygnał stacjonarny, zatem możliwe jest wyznaczenie parametrów filtru cyfrowego, odpowiadającego transmitancji traktu głosowego. Wtedy zamiast przesyłać próbki sygnału mowy, wystarczy przesłać współczynniki filtru, wraz z dodatkowymi informacjami niezbędnymi do odtworzenia sygnału w dekoderze. Operację taką należy wykonać dla każdej z ramek czasowych sygnału. Algorytm LPC wykorzystywany jest do takiego doboru współczynników filtru, aby uzyskać transmitancję filtru najlepiej dopasowaną do transmitancji traktu głosowego. Aby zminimalizować błąd predykcji, najczęściej stosowany jest algorytm Levinsona-Durbina.

– Analiza w dziedzinie czasu, odwzorowująca rezonansową strukturę traktu głosowego
– Ramkowanie, okienkowanie i autokorelacja między ramkami sygnału wejściowego
– Aproksymacja każdej kolejnej próbki jako liniowa kombinacja N poprzednich próbek

$$\hat{s}[n] = \sum_{k=1}^{P} a_k s(n-k)$$

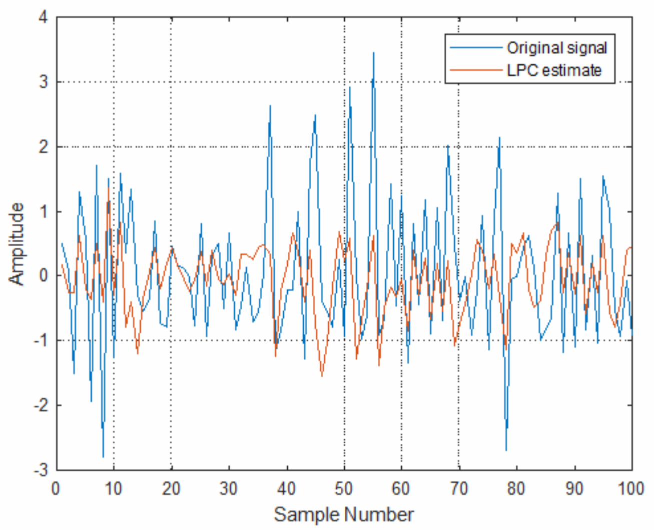
Przykład 1

Przeanalizować działanie funkcji. Czerwonym kolorem zaznaczono przybliżenie LPC.

```
noise = randn(50000,1);
x = filter(1,[1 1/2 1/3
1/4],noise);
x = x(end-4096+1:end);

a = lpc(x,3);
est_x = filter([0 -
a(2:end)],1,x);

plot(1:100,x(end-
100+1:end),1:100,est_x(end-
100+1:end))
grid
xlabel('Sample Number')
ylabel('Amplitude')
legend('Original signal','LPC
estimate')
```



2. Line Spectral Frequencies (LSF) – Częstotliwości widma liniowego

W latach 80-tych częstotliwości widma liniowego (ang. *Line Spectral Frequencies – LSF*) (inna nazwa to ang. *Line Spectrum Pair - LSP*) zostały wprowadzone jako alternatywna reprezentacja współczynników predykcji. Metoda ta była intensywnie rozwijana szczególnie przez japoński przemysł telefoniczny.

Współczynniki LSF są zerami dwóch wielomianów utworzonych z użyciem filtru inwersyjnego. Definiując je jako:

$$P(z) = A(z) + z^{-(M+1)} A(z^{-1}) \tag{9}$$

oraz

$$Q(z) = A(z) - z^{-(M+1)} A(z^{-1}) \tag{10}$$

można zapisać, że

$$A(z) = \frac{P(z) - Q(z)}{2} \tag{11}$$

Wielomiany (9) i (10) odpowiadają sztucznemu wydłużeniu tuby o przekrojach. Dodatkowy przekrój jest albo całkowicie zamknięty, wtedy jego powierzchnia wynosi 0, bądź też całkowicie otwarty. W tym ostatnim przypadku jego powierzchnia jest nieskończona. W związku z tym wszystkie zera wielomianów i leżą na kole jednostkowym na płaszczyźnie .

W rzeczywistości wielomian ma zero dla z = -1, a wielomian dla z = 1. Pozostałe zera obu wielomianów są zespolone i przeplatają się wzajemnie.

Przykład 2

Przeanalizować działanie funkcji `lpc()` dla dwóch współczynników.

```
X1=[1, 2, 3];
wynik1=lpc(X1,2)

X2=[2, 2, 2];
wynik2=lpc(X2,2)

X3=[3, 3, 3];
wynik3=lpc(X3,2)

X4=[0, 0, 0];
wynik4=lpc(X4,2)
```

Przykład 3

Zastosować funkcję `lpc()` dla sygnałów.

```
zdrowy1=[1, 2, 1, 3, 4, 1, 5]
zdrowy2=[1, 3, 1, 4, 1, 2, 2]
zdrowy3=[1, 3, 1, 4, 1, 3, 2]
uszkodzony1=[101, 100, 77, 88, 78, 99]
uszkodzony2=[89, 90, 95, 97, 99, 96]
uszkodzony3=[87, 97, 102, 88, 86, 96]
```

Zastosować 2 współczynniki.

Rozwiązanie przykładu 3

```
zdrowy1=[1, 2, 1, 3, 4, 1, 5]
lpc_zdrowy1=lpc(zdrowy1,2)    % lpc_zdrowy1 =1.0000 -0.2613 -0.4682

zdrowy2=[1, 3, 1, 4, 1, 2, 2]
lpc_zdrowy2=lpc(zdrowy2,2)  % lpc_zdrowy2 = 1.0000 -0.2679  -0.5179

zdrowy3=[1, 3, 1, 4, 1, 3, 2]
lpc_zdrowy3=lpc(zdrowy3,2)  % lpc_zdrowy3 =1.0000  -0.2595  -0.5373


uszkodzony1=[101, 100, 77, 88, 78, 99]
lpc_uszk1=lpc(uszkodzony1,2)        % 1.0000   -0.7674   -0.0250

uszkodzony2=[89, 90, 95, 97, 99, 96]
lpc_uszk2=lpc(uszkodzony2,2)        %  1.0000   -0.9334    0.1120
```

```
uszkodzony3=[87, 97, 102, 88, 86, 96]
lpc_uszk3=lpc(uszkodzony3,2)        % 1.0000   -0.9039    0.0842
```

Przykład 4

Zastosować obliczone w przykładzie 3 współczynniki LPC do rozpoznawania dwóch sygnałów: "zdrowy", "uszkodzony". Do każdego sygnałów mamy 2 próbki uczące: zdrowy1, zdrowy2, uszkodzony1, uszkodzony2 oraz po jednym sygnale testowym: zdrowy3, uszkodzony3. W programie użyć klasyfikatora najbliższego sąsiada z metryką Manhattanna.

Sygnały

```
zdrowy1=[1, 2, 1, 3, 4, 1, 5]
zdrowy2=[1, 3, 1, 4, 1, 2, 2]
uszkodzony1=[101, 100, 77, 88, 78, 99]
uszkodzony2=[89, 90, 95, 97, 99, 96]
```

zastosować do tworzenia wzorców.

Natomiast sygnały

```
zdrowy3=[1, 3, 1, 4, 1, 3, 2]
uszkodzony3=[87, 97, 102, 88, 86, 96]
```

zastosować do testowania.

Rozwiązanie przykładu 4

```
D=sum(abs(lpc_zdrowy3- lpc_zdrowy1))
D=sum(abs(lpc_zdrowy3- lpc_zdrowy2))
D=sum(abs(lpc_zdrowy3- lpc_uszk1))
D=sum(abs(lpc_zdrowy3- lpc_uszk2))

D =   0.0709
D =    0.0278  %  - OK rozpoznana klasa zdrowy
D =    1.0202
D =    1.3232

D=sum(abs(lpc_uszk3- lpc_zdrowy1))
D=sum(abs(lpc_uszk3- lpc_zdrowy2))
D=sum(abs(lpc_uszk3- lpc_uszk1))
D=sum(abs(lpc_uszk3- lpc_uszk2))

D =    1.1950
D =    1.2381
D =    0.2458
D =    0.0573  % rozpoznana klasa "uszkodzony",
             % ponieważ odległość D była najmniejsza do "lpc_uszk2"
```

Wyniki były dobre, ale proszę zwrócić uwagę, że współczynniki ujemne LPC powodują szereg błędów w rozpoznawaniu.

Przykładowo mamy 4 wektory

```
Klasa1_probka1=[1.0000 -0.2613 -0.4682]  - ucząca próbka klasy 1
Klasa1_probka2=[1.0000 -0.2613  0.4682]  - ucząca próbka klasy 1

Klasa2_probka1=[1.0000 -0.2613  0.3682] - ucząca próbka klasy 2
```

%%%

```
Klasa2_probka2=[1.0000 -0.2613 -0.3782] - testowa próbka klasy 2
```

Jak widać `Klasa2_probka2=[1.0000 -0.2613 -0.3782]` zostanie sklasyfikowana jako klasa 1, ponieważ liczby ujemne spowodowały błąd. W takim przypadku można zrobić wartość bezwzględną `abs(Klasa2_probka2)` lub wyzerować liczby ujemne `Klasa2_probka2(Klasa2_probka2<0)=0`

Bezpieczniej zatem obliczenia wykonać następująco:

```
D=sum(abs(abs(lpc_zdrowy3)- abs(lpc_zdrowy1)))
...
...
...
```

Zadania do wykonania

W sprawozdaniu powinny znaleźć się:
1) Informacje na temat co to jest LPC?
2) Jakie są zastosowania LPC?
3) Wykonane zadania - skrypty w m.plikach oraz otrzymane wykresy.
4) Wnioski z przeprowadzonych zadań.

Zad. 1
Zastosować funkcję `lpc()` dla sygnałów A, B, C.

```
A=[5, 6, 7, 8, 9];
B=[10, 8, 6, 8, 10];
C=[5, 6, 5, 6, 5];
```

Przeanalizować działanie `lpc()` dla 1, 2, 3 współczynników

Zad. 2
Zastosować funkcję `lpc()` dla sygnałów F i G.

```
F=[20, 30, 25, 15];
```

```
G=[1, 1, 1, 2, 2];
```
Przeanalizować działanie `lpc()` dla 1, 2, 3 współczynników.

Zad. 3
Zastosować funkcję `lpc()` dla sygnałów
wiatrak_20.wav
przekladnia20.wav
Przeanalizować działanie `lpc()` dla 1, 2, 10 współczynników.

Zad 4
Zastosować funkcję `lpc()` dla sygnałów.
```
samochod1=[1, 50, 1, 50, 1, 50, 1]
samochod2=[2, 49, 2, 49, 2, 49, 2]
samochod3=[1, 48, 2, 49, 3, 50, 4]

ciezarowka1=[10, 20, 10, 20, 10, 20]
ciezarowka2=[11, 21, 11, 21, 11, 21]
ciezarowka3=[12, 22, 12, 22, 12, 22]
```
Do obliczeń zastosować 2 współczynniki.

Zad 5
Zastosować obliczone w zadaniu 4 współczynniki LPC do rozpoznawania. W programie użyć klasyfikatora najbliższego sąsiada z metryką Manhattanna.
Sygnały
```
samochod1=[1, 50, 1, 50, 1, 50, 1]
samochod2=[2, 49, 2, 49, 2, 49, 2]
ciezarowka1=[10, 20, 10, 20, 10, 20]
ciezarowka2=[11, 21, 11, 21, 11, 21]
```
zastosować do tworzenia wzorców.

Natomiast sygnały
```
samochod3=[1, 48, 2, 49, 3, 50, 4]
ciezarowka3=[12, 22, 12, 22, 12, 22]
```
zastosować do testowania.

Zad. 6
Użyć funkcji `lpc()` – *Linear Predictive Coding* (tworzy współczynniki filtru na podstawie sygnału). Korzystając z funkcji `lpc()`, przeprowadzić rozpoznawanie na próbkach WAV, Użyć 10 współczynników. Dane *.wav wczytać przez komendę `import`.

Program można rozpocząć od komendy
```
a=lpc(data, 10);
```

Ze względu na to, że wartości mogą być ujemne warto zastosować wartość bezwzględną funkcja `abs()`. Klasyfikator najbliższego sąsiada (metryka Manhattan) opiera się na odejmowaniu zatem liczby ze znakiem ujemnym powodują błędy.

```
a=abs(lpc(data, 10));
```

Zad 7

Zastosować funkcję `poly2lsf()` dla współczynników LPC następujących sygnałów.

```
samochod1=[1, 50, 1, 50, 1, 50, 1]
samochod2=[2, 49, 2, 49, 2, 49, 2]
samochod3=[1, 48, 2, 49, 3, 50, 4]

ciezarowka1=[10, 20, 10, 20, 10, 20]
ciezarowka2=[11, 21, 11, 21, 11, 21]
ciezarowka3=[12, 22, 12, 22, 12, 22]
```

Zastosować 2 współczynniki.

Zad 8

Zastosować obliczone w zadaniu współczynniki LSF do rozpoznawania.
Sygnały

```
samochod1=[1, 50, 1, 50, 1, 50, 1]
samochod2=[2, 49, 2, 49, 2, 49, 2]
ciezarowka1=[10, 20, 10, 20, 10, 20]
ciezarowka2=[11, 21, 11, 21, 11, 21]
```

zastosować do tworzenia wzorców.

Natomiast sygnały

```
samochod3=[1, 48, 2, 49, 3, 50, 4]
ciezarowka3=[12, 22, 12, 22, 12, 22]
```

zastosować do testowania.

Pytania

1) W jaki sposób zastosować LPC do ekstrakcji cech?
2) Jakie parametry możemy ustawiać w metodzie LPC?
3) Gdzie możemy zastosować LPC?
4) Jak duże okno możemy ustawić w metodzie LPC i w jaki sposób jest wykonywane okienkowanie?