PSI-3260 Aplicações de Álgebra Linear
Experiência 11

Decomposição em Valores Singulares 2 – Parte Computacional

1 Melhorando a razão sinal ruído em sinais de voz

Nesta experiência, vamos ver como podemos usar a SVD de uma matriz com amostras de um sinal corrompido por ruído para melhorar a razão sinal-ruído. Por simplicidade, vamos considerar que um sinal de voz "limpo" é corrompido por ruído gaussiano.

A voz é classificada como um sinal *não estacionário*, pois suas características estatísticas variam com o tempo. Considere, por exemplo, $0,45$ s de um sinal de voz mostrado na Figura 1. Os primeiros $0,15$ s correspondem ao final do pronunciamento de uma palavra, no intervalo $0,75 \leq t < 0,93$ s temos um trecho de silêncio em que só se ouve ruído e uma outra palavra começa a ser pronunciada a partir de $t = 0,93$ s. A Tabela 1 contém os valores da média e da variância das amostras desse sinal, considerando esses três trechos. É possível observar que os valores da média e da variância do sinal dependem do trecho considerado.

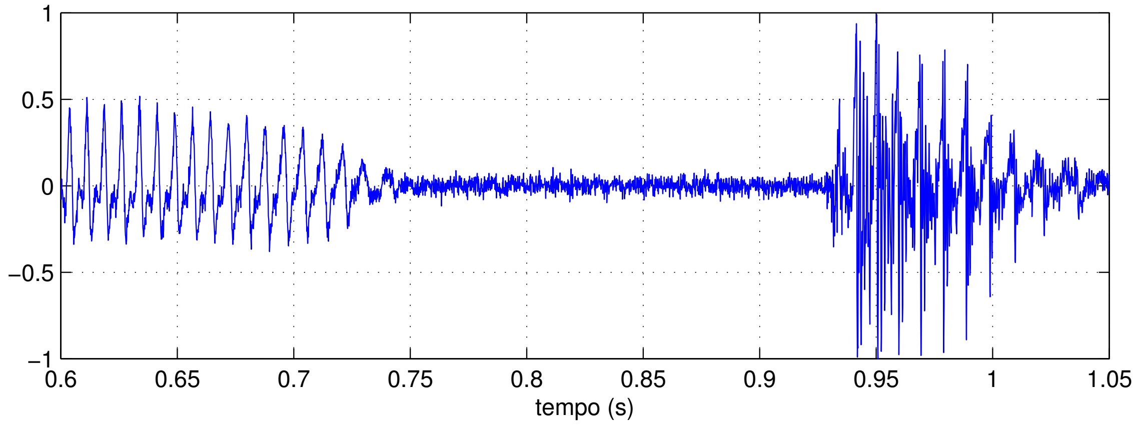

Figura 1: Trecho de um sinal de voz com ruído.

Tabela 1: Média e variância de trechos do sinal da Figura 1.

Intervalo de tempo (s)	Média	Variância
$0,60 \leq t < 0,75$	$0,0014$	$0,0351$
$0,75 \leq t < 0,93$	$-0,0002$	$0,0009$
$0,93 \leq t < 1,05$	$0,0006$	$0,0747$

Para lidar com sinais não estacionários, é comum dividir o sinal em trechos com sobreposição, como ilustrado na Figura 2. No exemplo dessa figura, cada trecho é composto de $K = 100$ amostras e a sobreposição é de $L = 20$ amostras. O sinal de cada trecho pode ser considerado aproximadamente estacionário, ou seja, suas características estatísticas como média e variância são aproximadamente constantes. Além disso, a sobreposição é particularmente útil nas fronteiras entre trechos distintos, como os que aparecem na Figura 1. Estudos mostram que sinais de voz são aproximadamente estacionários em trechos de aproximadamente 20 ms a 40 ms. Portanto, a definição do número de amostras K do trecho do sinal a ser processado é essencial para garantir estacionariedade e depende da frequência de amostragem.

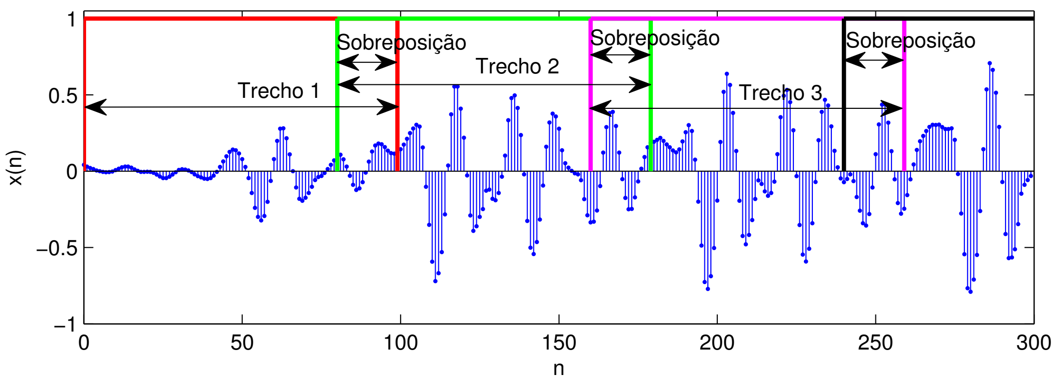

Figura 2: Trechos de um sinal com sobreposição.

Em cada trecho k, as K amostras $x_k(n)$, $n = 0, 1, \cdots, K-1$ de um sinal de voz com ruído podem ser escritas como

$$x_k(n) = s_k(n) + w_k(n),$$

em que $s_k(n)$ são as amostras do sinal de voz "limpo" e $w_k(n)$ as amostras do sinal de ruído. A sequencia temporal $x_k(0), \cdots, x_k(K-1)$ do trecho k (*frame* k) pode ser representada na seguinte matriz[1] $p \times q$

$$\mathbf{H}_{x_k} = \begin{bmatrix} x_k(0) & x_k(1) & \cdots & x_k(q-1) \\ x_k(1) & x_k(2) & \cdots & x_k(q) \\ \vdots & \vdots & & \vdots \\ x_k(p-1) & x_k(p) & \vdots & x_k(K-1) \end{bmatrix},$$

sendo $p + q = K + 1$ e $p \geq q$.

Definindo matrizes semelhantes para o sinal de voz "limpo" e para o ruído em cada trecho k, podemos escrever

$$\mathbf{H}_{x_k} = \mathbf{H}_{s_k} + \mathbf{H}_{w_k}. \tag{1}$$

[1]Essa matriz também é chamada de matriz de Hankel na literatura de processamento de voz. No entanto, uma matriz de Hankel é definida em Álgebra Linear como uma matriz quadrada em que os elementos de cada uma das antidiagonais são iguais.

Cada uma dessas matrizes pode ser decomposta em valores singulares, ou seja,

$$\begin{aligned}
\mathbf{H}_{x_k} &= \mathbf{U}_{x_k}\mathbf{S}_{x_k}\mathbf{V}_{x_k}^T \\
\mathbf{H}_{s_k} &= \mathbf{U}_{s_k}\mathbf{S}_{s_k}\mathbf{V}_{s_k}^T \\
\mathbf{H}_{w_k} &= \mathbf{U}_{w_k}\mathbf{S}_{w_k}\mathbf{V}_{w_k}^T.
\end{aligned}$$

Usando as SVDs dessas matrizes, podemos reescrever (1) como

$$\mathbf{H}_{x_k} = \mathbf{U}_{s_k}\mathbf{S}_{s_k}\mathbf{V}_{s_k}^T + \mathbf{U}_{w_k}\mathbf{S}_{w_k}\mathbf{V}_{w_k}^T.$$

Se fosse possível fazer $\mathbf{S}_{w_k} = \mathbf{0}$, conseguiríamos eliminar completamente o ruído. O problema é que não conhecemos $\mathbf{H}_{s_k}$ e $\mathbf{H}_{w_k}$ na prática. Temos acesso apenas à matriz $\mathbf{H}_{x_k}$.

Para reduzir o efeito do ruído, podemos calcular a SVD da matrix $\mathbf{H}_{x_k}$ e então eliminar ou atenuar os valores singulares menos significativos da matriz contidos em $\mathbf{S}_{x_k}$, o que leva à aproximação $\hat{\mathbf{S}}_{s_k}$. Os valores singulares de menor valor são supostamente pertencentes ao sinal de ruído e a eliminação ou atenuação desses valores singulares possibilita a melhoria na razão sinal-ruído do sinal de voz degradado. Para isso, podemos reconstruir o sinal por meio da seguinte matriz

$$\boxed{\hat{\mathbf{H}}_{s_k} = \mathbf{U}_{x_k}\hat{\mathbf{S}}_{s_k}\mathbf{V}_{x_k}^T.} \tag{2}$$

Para eliminar ou atenuar os valores singulares de menor valor, precisamos definir um limiar para decidir quais valores singulares vamos eliminar. Esse limiar é um parâmetro crucial para o bom funcionamento dessa técnica, já que uma seleção inadequada do limiar pode resultar em uma redução insuficiente ou exagerada do ruído. Neste último caso, a inteligibilidade do sinal de voz pode ser afetada. Na literatura, existem inúmeras técnicas para seleção do limiar. Nosso objetivo não é apresentar ou estudar essas técnicas, apenas mostrar que a SVD pode ser utilizada nessa aplicação.

2 Exercício com o Matlab

Faça os itens seguintes:

1. Considere o sinal de voz do arquivo `voz.wav`, amostrado com 8 kHz (baixe o arquivo do Moodle). Use a função `wavread.m` do Matlab para obter as amostras do sinal. Faça um gráfico do sinal ao longo do tempo em segundos e ouça-o com a função `sound.m`. Não se esqueça de alterar a frequência de amostragem dessa função.

2. Gere um vetor de ruído aleatório com a mesma dimensão do vetor que contém as amostras do sinal de voz do Item 1. O ruído deve ser gaussiano, ter média zero e variância $\sigma^2 = 0{,}01$. Use a função `randn.m` do Matlab para gerar esse ruído e verifique se o ruído gerado ficou com a média e a variância desejadas. Para isso, você pode usar as funções `mean.m` e `var.m` do Matlab. Use também a função `hist.m` para ver se a distribuição do ruído é de fato gaussiana.

3. Some o ruído gerado no Item 2 ao sinal de voz do Item 1. Normalize o sinal resultante de modo que o valor absoluto máximo desse sinal seja igual a um[2]. Faça um novo gráfico do sinal e ouça-o com a função `sound.m`.

4. Considere dois trechos não consecutivos do sinal de voz com ruído gerado no Item 3:

$$\text{Trecho 1:} \quad x(n), \quad n = 1201, 1202, \cdots, 1456$$

e

$$\text{Trecho 2:} \quad x(n), \quad n = 3801, 3802, \cdots, 4056.$$

O Trecho 1 só contém amostras do ruído, enquanto o Trecho 2 contém amostras do sinal de voz somadas a ruído.
Responda:

- Cada trecho tem quantas amostras?
- Dado que a frequência de amostragem é 8 kHz, determine a duração em ms desses trechos.
- Podemos considerar que o sinal é estacionário em cada um desses trechos? Por que?

Construa as matrizes $\mathbf{H}_{x_k}$, $k = 1, 2$ para cada um desses trechos considerando $q = 30$. Apresente em um mesmo gráfico os valores singulares dessas matrizes. Que valor de limiar você usaria para reduzir o efeito do ruído no Trecho 2?

5. No Item 4, foram definidos dois trechos do sinal e para cada trecho você gerou a matriz $\mathbf{H}_{x_k}$. Vamos agora gerar todos os trechos do sinal de voz que devem ser processados. Para isso, você poderá usar a função `analise.m` fornecida no Moodle. Faça um teste com essa função: use o sinal de voz sem ruído do Item 1 e obtenha trechos desse sinal com $K = 256$ amostras (comprimento do *frame*) e sobreposição de $L = 128$ amostras. Depois de processar cada trecho, você precisará reconstruir o sinal. Para isso, use a função `sintese.m` fornecida no Moodle. Teste essa função, colocando na sua entrada a saída da função `analise.m` e verifique se o sinal reconstruído é igual ao sinal original, ou seja, igual à entrada da função `analise.m`.

6. Faça um `for` para processar todos os trechos do sinal de voz com ruído gerado no Item 3. Construa a matriz $\mathbf{H}_{x_k}$ para cada trecho do sinal de voz, considerando $q = 30$, $K = 256$ e $L = 128$. Teste diferentes valores de limiar para zerar os valores singulares menores de $\mathbf{H}_{x_k}$. Considere, por exemplo, limiares no intervalo $[1{,}5; \;\; 5]$. Escolha o limiar que parece mais adequado ao ouvir o sinal de voz reconstruído. O sinal reconstruído é exatamente igual ao sinal original?

7. Para o valor do limiar escolhido no Item 6, apresente em um mesmo gráfico o sinal de voz reconstruído, o sinal de voz com ruído e o sinal de voz "limpo".

[2] Essa normalização é particularmente importante para que não ocorra saturação do sinal com a função `sound.m`.