Introdução a Transformadas de Fourier

Introdução

Transformada de Fourier

Exemplos

Eng. Henrique Guarneri – PPGERHA/UFPR
henriqueguarneri@gmail.com

Jun 15

Transformada de Fourier

- Metáfora:
  - O que as Transformadas de Fourier fazem ? Dada uma Vitamina, acha os ingredientes.
  - Como? 'Roda' a Vitamina por filtros e extrai a receita.
  - Porque ? Receitas são mais simples de analisar, comparar, e modificar do que Vitaminas.
  - Como conseguir a Vitamina novamente ? Misture os ingredientes.

Fonte:betterexplained

Transformada de Fourier

Em português matemático:

- A Transformada de Fourier 'pega' um padrão com base temporal e mede cada possível ciclo, retornando a 'receita‘ geral dos ciclos (intensidade, deslocamento , velocidade e rotação para cada ciclo que foi encontrado).

Transformada de Fourier

- Vamos verificar como cada padrão pode ser construído com ciclos utilizando algumas ferramentas de visualização.

- Reforçar a intuição;
- Apresentar exemplos simples;
- Equações;
- Exemplo aplicado;

Transformada de Fourier

- Transformada de Fourier é uma transformação matemática.
- A Transformada de Fourier muda nossa perspectiva de 'consumidor' para 'produtor'. 'O que eu vejo ?' para 'Como foi feito ?'.
- Veja a analogia a seguir:

Analogia: Da Vitamina à Receita

Vitamina

Transformação		Ponto de Vista do Produtor
Filtro para Banana	→	Quantidade de Banana (g)
Filtro para Laranja	→	Quantidade de Laranja (g)
Filtro para Leite	→	Quantidade de Leite (g)
Filtro para Água	→	Quantidade de Água (g)

Ponto de Vista do Consumidor

Transformação

Ponto de Vista do Produtor

Fonte:betterexplained

Da Vitamina à Receita:

Podemos então fazer a engenharia reversa da receita filtrando cada ingrediente. Mas:

- **Filtros devem ser independentes.** O filtro de banana precisa capturar bananas, e mais nada. Adicionando mais laranjas não deveria nunca afetar a leitura de bananas.

- **Filtros devem ser completos.** Não vamos conseguir a receita real se deixarmos um filtro de fora. Nossa coleção de filtros deve coletar cada um dos ingredientes.

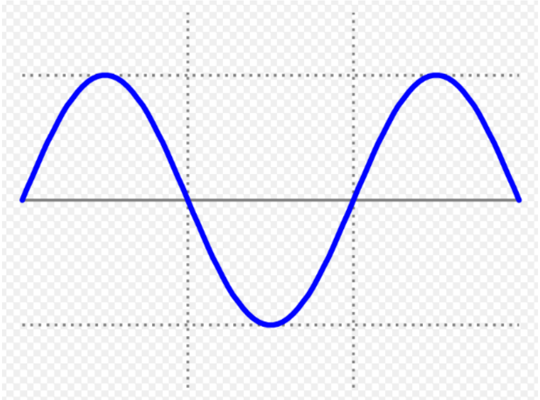
- **Ingredientes devem ser combináveis.** Vitaminas podem ser separáveis e recombinadas sem problemas (Um biscoito ? Nem tanto. Quem quer migalhas ?). Os ingredientes, quando separados e combinados em qualquer ordem, devem produzir o mesmo resultado.

Compreendendo os Ciclos:

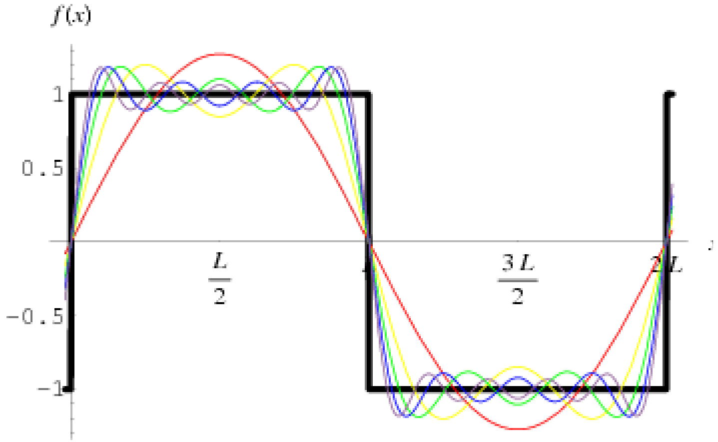
A transformada de Fourier usa um ponto de vista específico:

Qualquer sinal pode ser representado por uma combinação de trajetórias circulares.

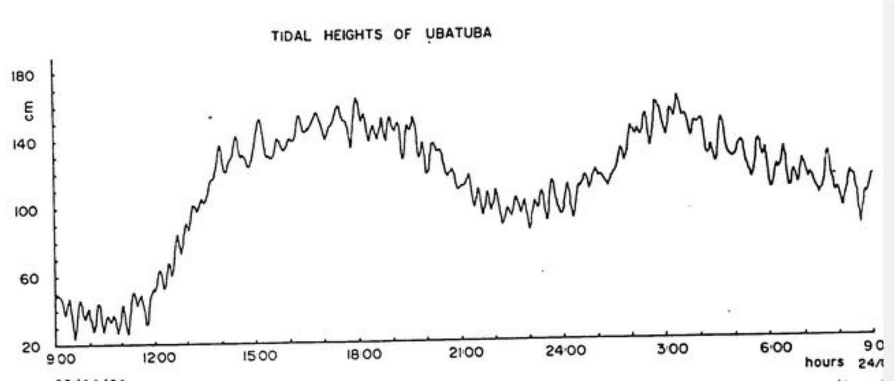
Seno



“Onda” quadrada



Leituras de Marégrafo



Compreendendo os Ciclos:

- A Transformada de Fourier é uma das introspecções mais profundas já feitas. Infelizmente, isso significa equações densas:

FT:

$$\mathrm{F(k)} = \int_{-\infty}^{+\infty} f(t) \times e^{-i2\pi kt/N} dt$$

$$\mathrm{f(t)} = \frac{1}{N} \int_{-\infty}^{+\infty} F(k) \times e^{-i2\pi tk/N} dk$$

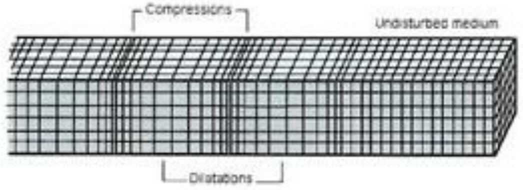
DFT:

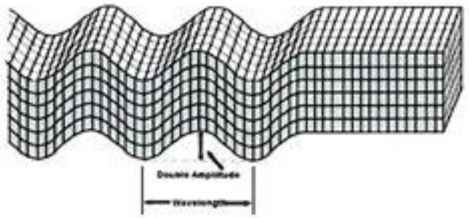
$$X_k = \sum_{n=0}^{N-1} \mathrm{x_n} \times e^{-i2\pi kn/N}$$

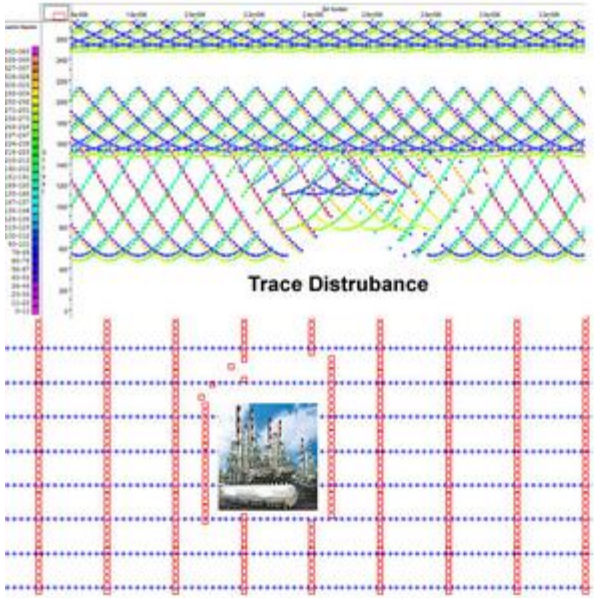
$$\mathrm{x_n} = \frac{1}{N} \times \sum_{k=0}^{N-1} \mathrm{X_k} \times e^{i2\pi tk/N}$$

Compreendendo os Ciclos:

- Se as **vibrações sísmicas** podem ser separadas em "ingredientes" (vibrações de diferentes velocidades, intensidades e direções) , os edifícios podem ser concebidos para evitar que interajam destrutivamente com as mais fortes.



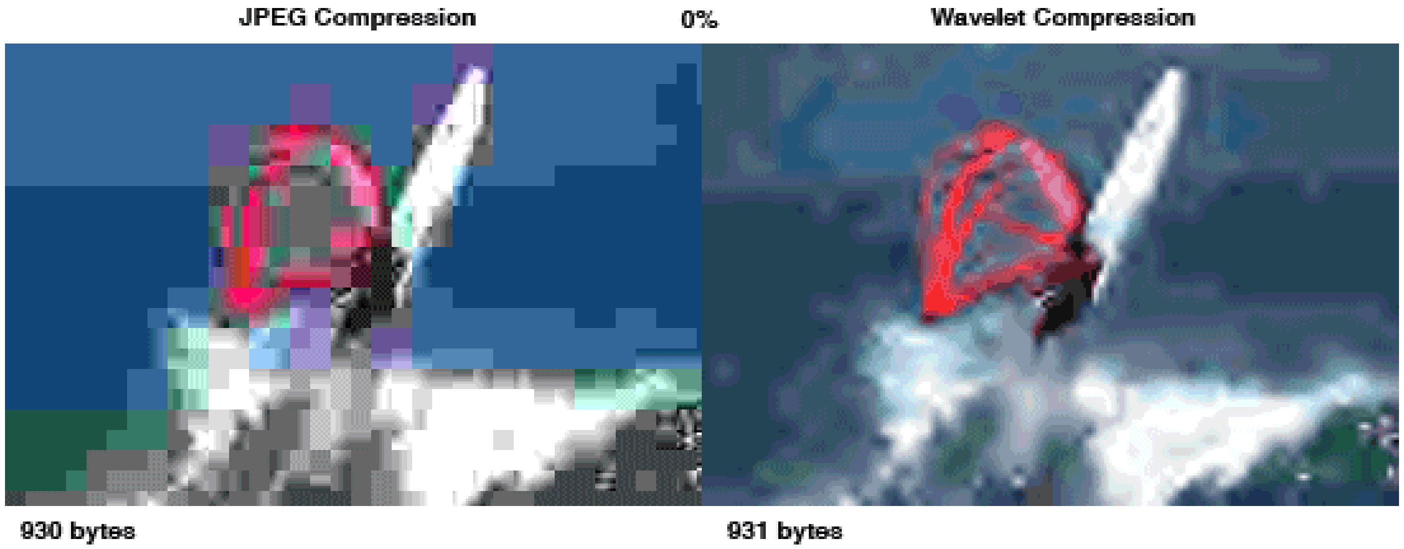




Fonte: GEMCORP

Compreendendo os Ciclos:

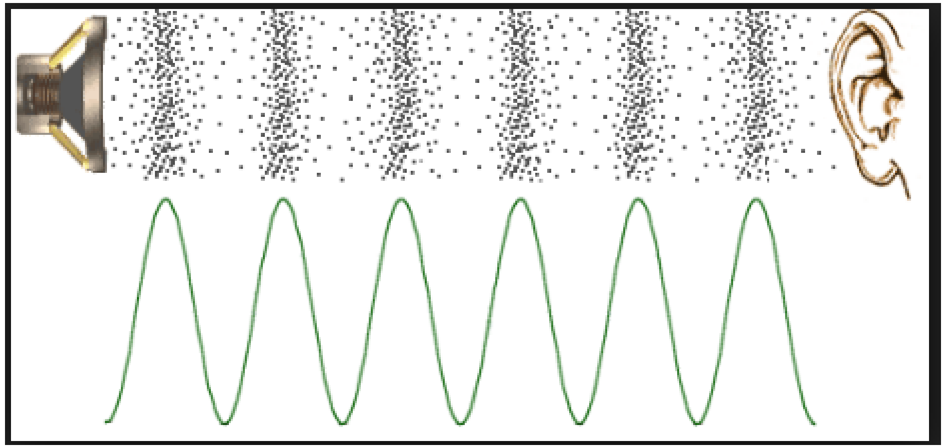
- Se os dados de computador podem ter seus tamanhos reduzidos drasticamente. Identificando e mantendo somente os principais componentes oscilatórios e ignorando o menos importante(" **compressão com perdas “)** (e por este motivo que JPEG e arquivos MP3 são muito menores do que .bmp bruto ou arquivos .wav).



Fonte: National Curve Bank

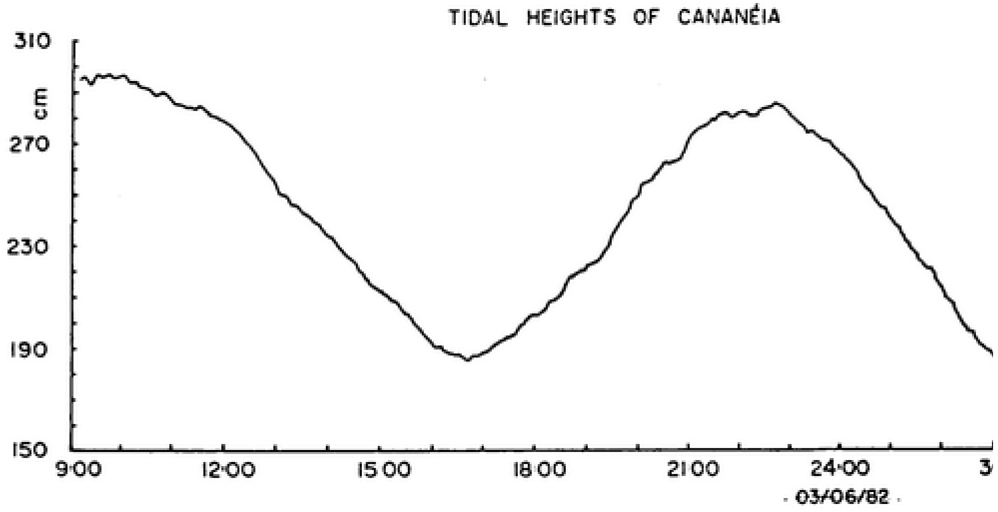
Compreendendo os Ciclos:

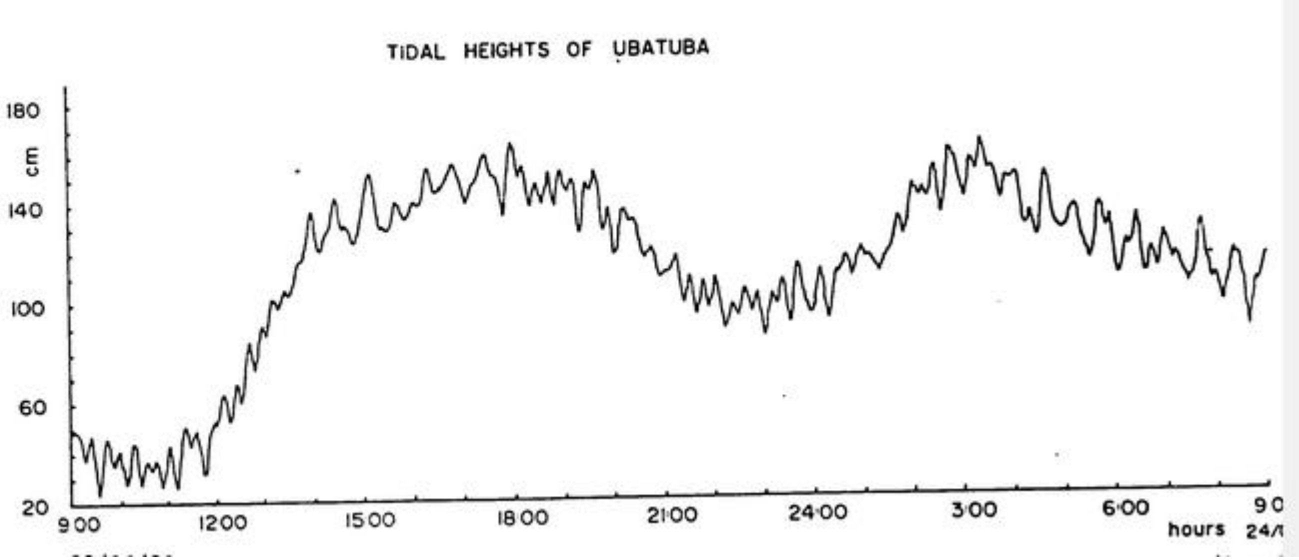
- Se as ondas sonoras podem ser separadas em ingredientes (frequências graves agudas), podemos intensificar as partes que mais gostamos , e ocultar as que não nos agradam. O ruído aleatório pode ser removido. Talvez "receitas " de sons semelhantes possam ser comparadas (SoundHound, Shazam,...)



Compreendendo os Ciclos:

- Se os dados de **nível de água** podem ser representados com padrões oscilantes, talvez possamos separar **os influências da maré lunar**, de **outros planetas, ventos, rios, e ruídos como a passagem de um barco ao lado do equipamento.**

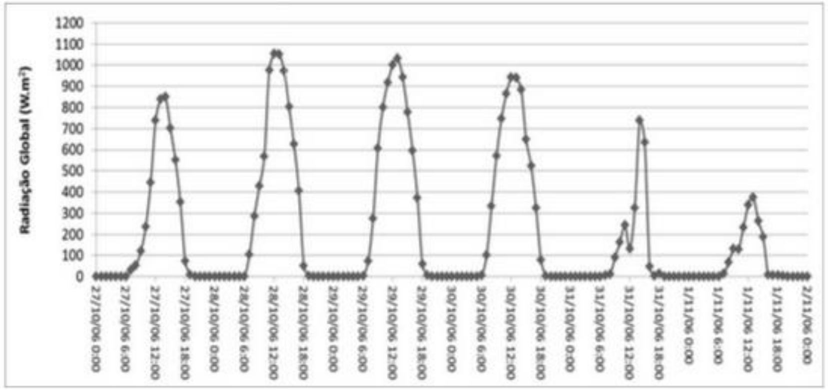


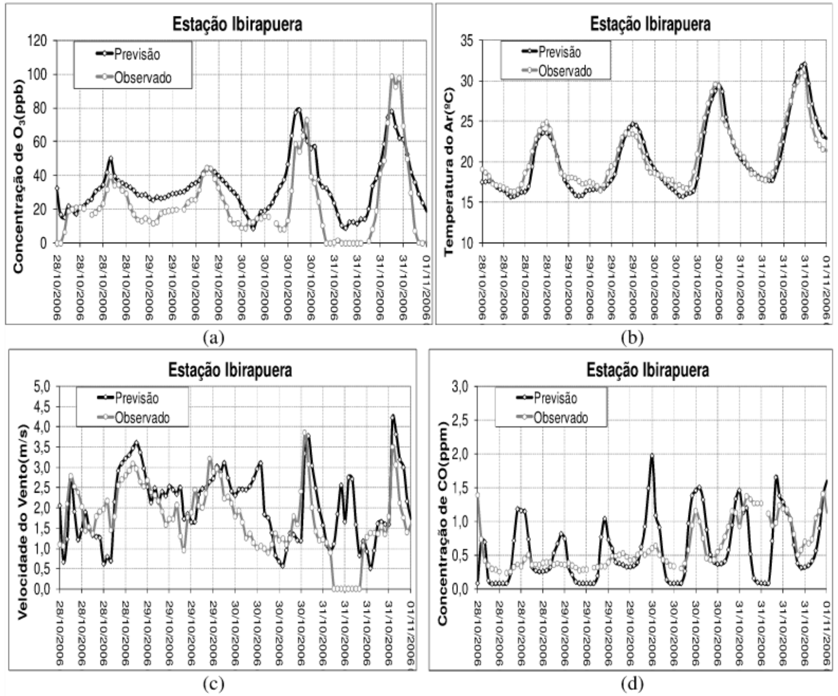


Fonte: http://www.mares.io.usp.br/sudeste/sudeste.html

Compreendendo os Ciclos:

- Se os dados de **poluição atmosférica** no centro de uma cidade apresentam padrões de oscilação, talvez possamos verificar os períodos de ‘**RUSH’**, qual direção de **vento** contribui para o aumento das concentrações, ou a influência da **radiação**. E assim continuar as investigações relacionadas.





Fonte: Silva Junior; Andrade (2013)

Compreendendo os Ciclos:

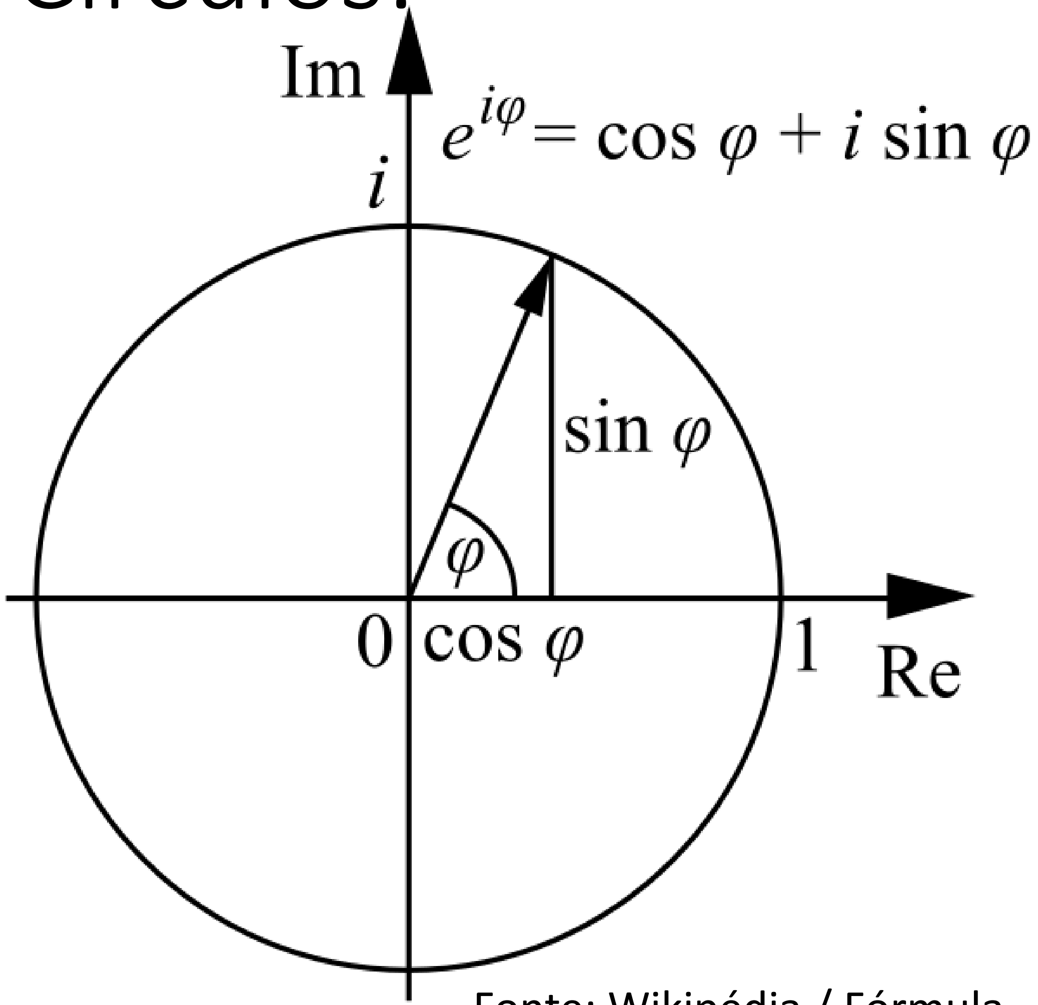
OBJETIVO PRIMÁRIO:

- **Encontrar as causas chave por trás dos efeitos observáveis**.

Distinção entre Senoides e Círculos:

Uma senoide é um padrão oscilatório específico (uma onda seno e/ou cosseno. Em geral refere-se a movimento em uma dimensão.

- Um circulo é um padrão 2D redondo. Podemos chamar uma trajetória circular de uma senoide complexa.
- **A Transformada de Fourier é relacionada a trajetórias circulares e a formula de Euler é uma forma elegante de gerar uma.**

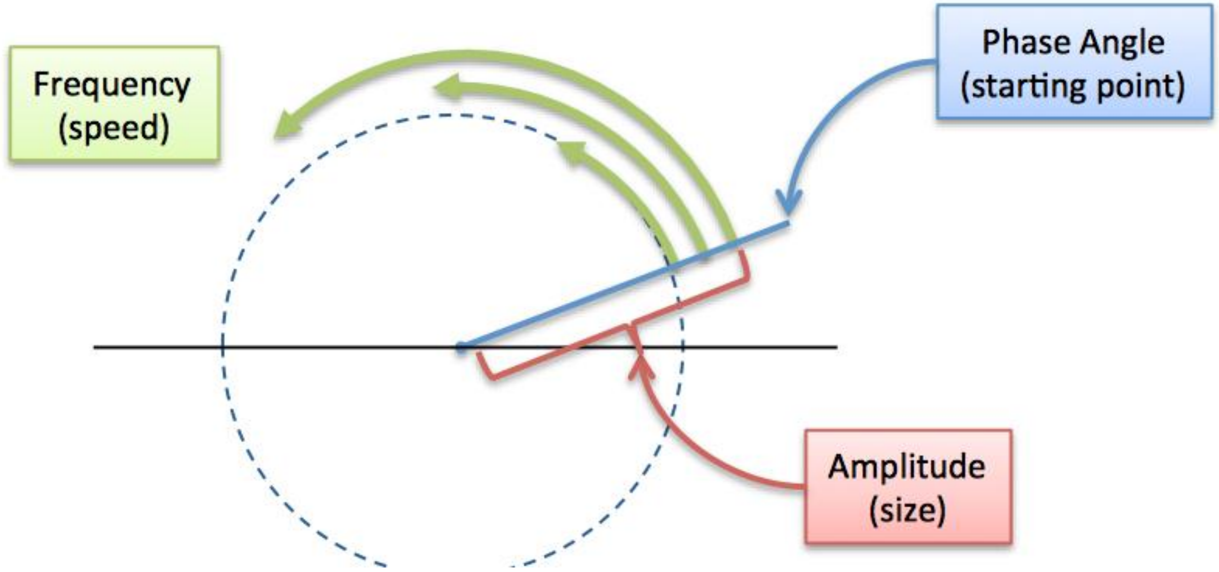


Fonte: Wikipédia / Fórmula de Euler

Características Fundamentais de Trajetórias Circulares:

- O **Tamanho** do Circulo (Amplitude, ex: raio)
- A **Velocidade** de trajetória. (Frequência. 1 circulo/segundo é uma frequência de 1 Hertz (Hz) ou 2*pi radianos/segundo).
- **Ângulo** inicial. (Ângulo de fase, onde 0 graus é o eixo x).

Trajetórias Circulares:



Betterexplained.com (2015)

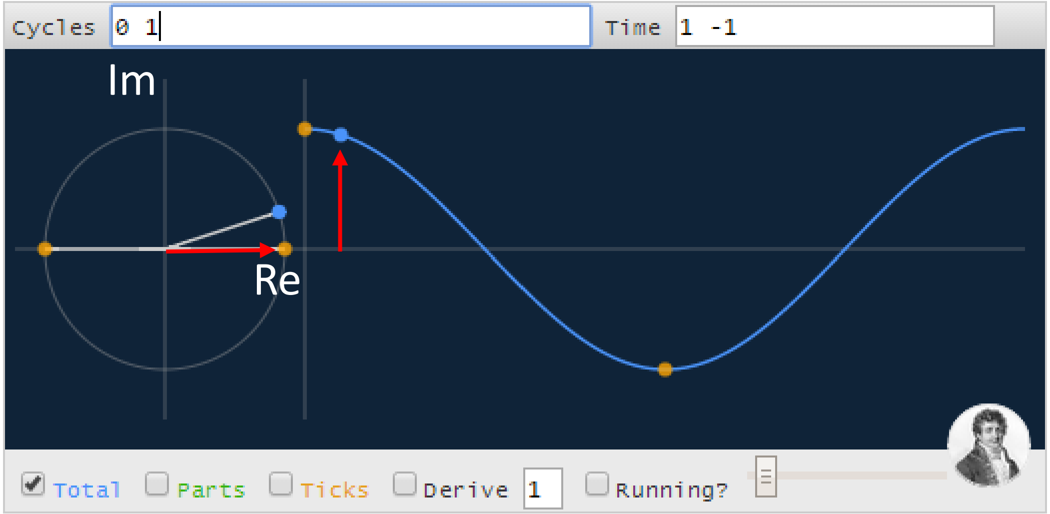
- Toda trajetória circular precisa de **amplitude, frequência e fase**.
- As posições combinadas de cada trajetória no fornecem o sinal.
- Exemplo ilustrativo:

Trajetórias Circulares:

Exemplo:

- http://betterexplained.com/examples/fourier/?cycles=0,1

Magnitude



Trajetórias Circulares:

A magnitude de cada ciclo (período) é listada em ordem, começando em 0Hz.

O Ciclo [0 1] significa:

- Intensidade 0 para o ciclo de 0Hz (0Hz = um ciclo constante, preso no eixo x em zero graus).
- Intensidade 1 para o ciclo 1Hz (completa 1 ciclo por intervalo de tempo).

Trajetórias Circulares:

A parte azul mede a parte real do ciclo:

- Esse conceito pode ser confundido matematicamente: o eixo real do circulo, que geralmente é horizontal, tem sua magnitude indicada no eixo vertical. Pode-se rotacionar mentalmente em 90 graus o circulo para visualizar isso melhor.

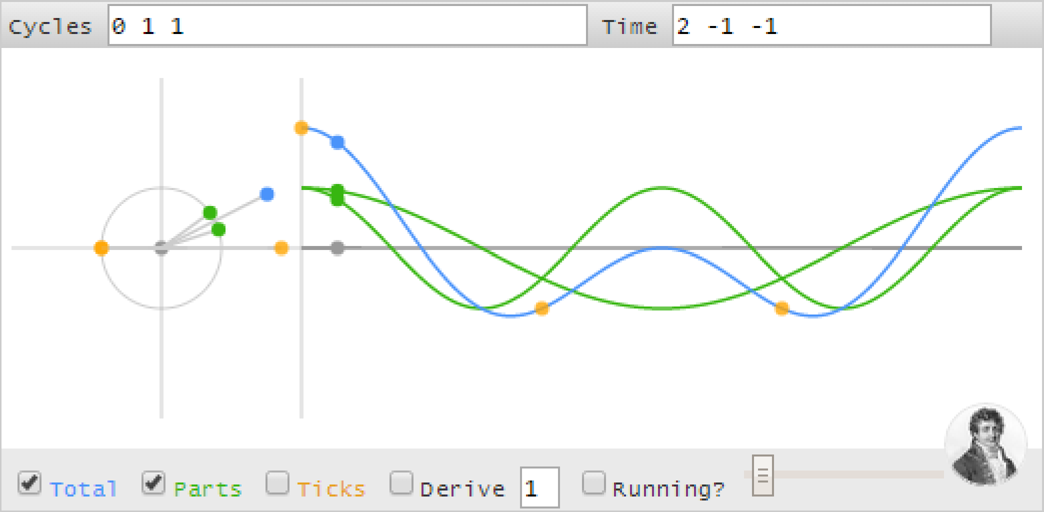
Trajetórias Circulares:

Os pontos de tempo estão espaçados na frequência mais rápida.

- Um sinal de 1Hz precisa de 2 pontos de tempo para um inicio e fim (um único dado de ponto não tem frequência). O valor de tempo [1 -1] mostram a amplitude nesses intervalos igualmente espaçados.
- Estão acompanhando ? [0 1] é um ciclo de 1Hz puro.

Trajetórias Circulares:

Agora vamos adicionar um ciclo de 2Hz na mistura. [0 1 1] significa: 'Nada em 0Hz, 1Hz de intensidade 1, e 2Hz de intensidade 1':



Resultante

Partes

Pontos de Amostragem

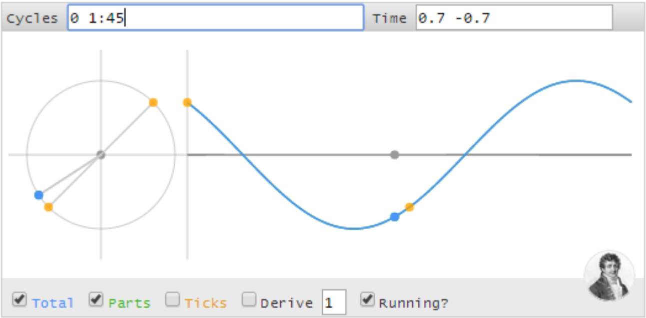
Os pontos verdes são os ciclos e o azul sua combinação.

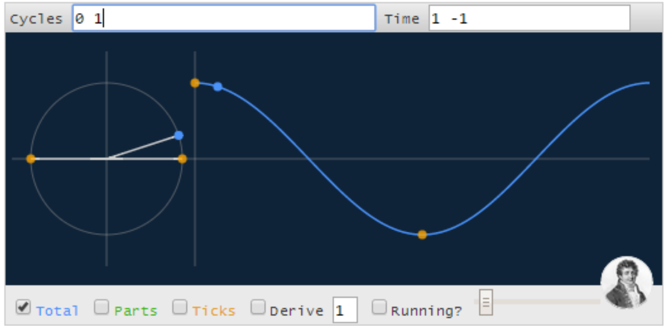
Trajetórias Circulares:

Fases:

Podemos usar magnitude : ângulo para definir a fase.

Betterexplained.com (2015)





- Essa é a versão deslocada de [0 1]. No lado do tempo conseguimos [.7 -.7] no lugar de [1 -1], por que nosso ciclo não está exatamente alinhado com nosso intervalo de medições, que ainda está no ponto central.

Trajetórias Circulares:

A transformada de Fourier encontra o conjunto de velocidades, intensidades e fases para se igualar a qualquer Sinal (função) de tempo.

Trajetórias Circulares:

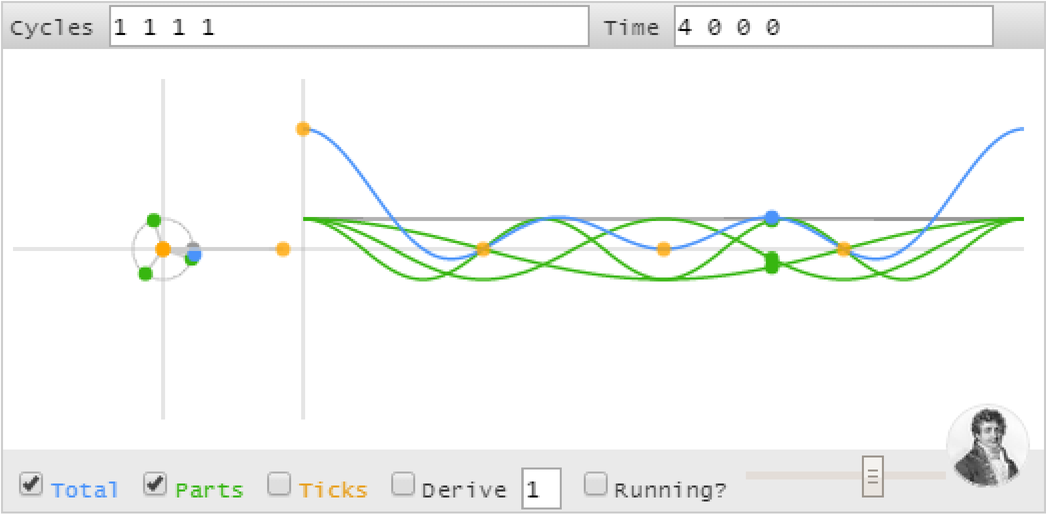
Nosso sinal se torna uma noção abstrata que consideramos como 'observações no domínio temporal' ou 'ingredientes no domínio de frequências'.

Fazendo um pico no tempo:

Fazendo um pico no tempo, tipo (4 0 0 0), usando ciclos !

Notação:

- () para sequência de pontos;
- [] para sequência de ciclos;



Betterexplained.com (2015)

Fazendo um pico no tempo:

Como podemos fazer isso?

- Tempo 0: Todos os ciclos estão no seu máximo (6)
- Tempo 1: 1Hz e 3Hz se cancelam (Posição 1 e 3 são opostas), 0Hz e 2Hz se cancelam também. Total 0.
- Tempo 2: 0Hz e 2Hz se alinham a posição 0, enquanto 1Hz e 3Hz se alinham n posição 2 (lados opostos). O total ainda é zero.
- Tempo 3: 0Hz e 2Hz se cancelam. 1Hz e 3Hz se cancelam.
- Tempo 4 (repete o tempo 0). Todos os ciclos se alinham.

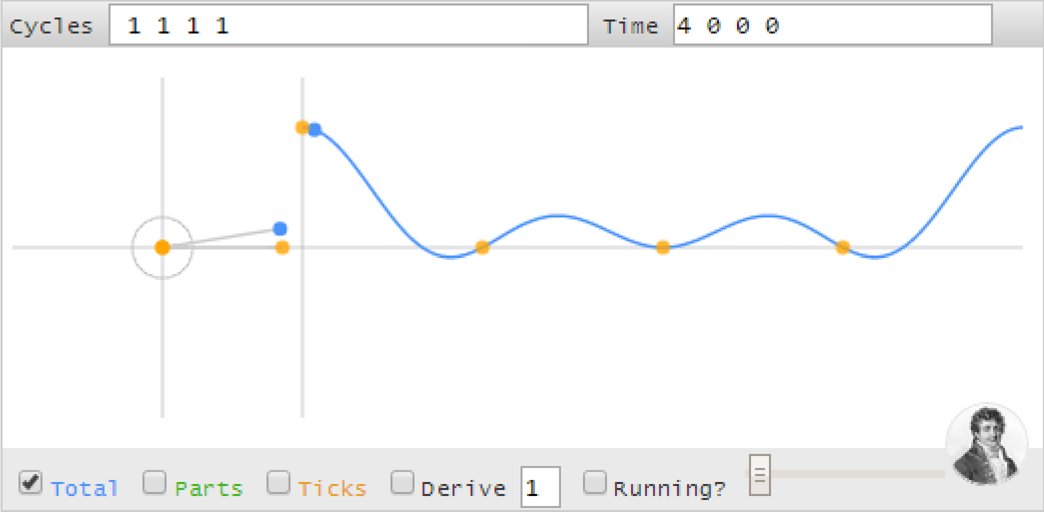
Fazendo um pico no tempo:

- Isso acontece pois:
- Velocidades individuais se cancelam: (0Hz vs 2Hz, 1Hz vs 3Hz);
- Velocidades quando se alinhando se alinhando em zero (0Hz+2Hz vs 1Hz+3Hz).
- Ocorre para ciclos de mesma intensidade e fase nula.
- Quando cada ciclo tem mesma intensidade e fase 0, eles começam alinhados e depois se cancelam. Funciona para [1,1], [1,1,1], [1,1,1,1] e os sinais resultantes são (2 0), (3 0 0), (4 0 0 0).
- Picos temporais, 'atividade amplificada' em um instante de tempo e zero nos outros instantes ('Função' Delta de Dirac).

Fazendo um pico no tempo:

Contribuições:
- Construtivas
- Destrutivas

Pontos de interesse



Betterexplained.com (2015)

Fazendo um pico no tempo:

Para realizarmos um pico no Tempo 2 precisamos usar as fases.

Deslocamentos de fase, os ângulos iniciais, são atrasos no 'mundo' dos ciclos.

Fazendo um pico no tempo:

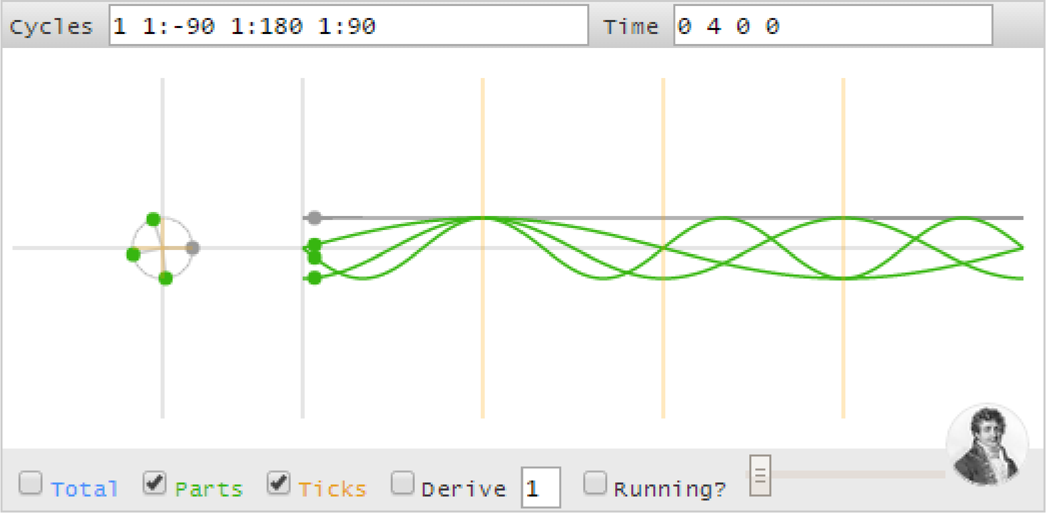
[0 4 0 0]:

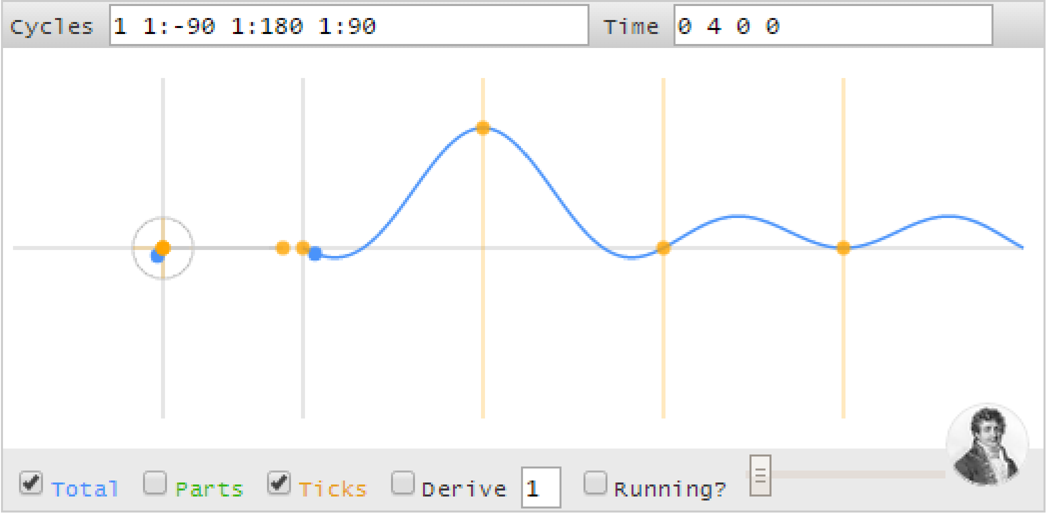
- Um ciclo de 0Hz não se move, então já esta alinhado;
- Um ciclo de 1Hz faz 1 revolução nos 4 segundos, então um atraso de 1 segundo é 1 quarto de volta. Atrasando-o 90 graus para trás (-90 graus) ele chegara com fase=0, seu valor máximo, em t=1;
- Um ciclo de 2Hz é duas vezes mais rápido, logo devemos da a ele o dobro de ângulo de inicio (-180 graus ou deslocamento de fase de 180).
- um ciclo de 3Hz é 3x mais rápido, então vamos dar a ele 3x a distância a percorrer (-270 graus ou um deslocamento de fase de +90).

Fazendo um pico no tempo:

- Se (4 0 0 0) resulta de [1 1 1 1]
- então (0 4 0 0) resulta de [1 1:-90 1:180 1:90]

*Leia Hz como 1 ciclo ao longo de todo o período de tempo.





Betterexplained.com (2015)

Fazendo um pico no tempo:

Sugestão:

- Obtenha as fases que resultam em (0 0 4 0). [1:? 1:? 1:? 1:?]

Fazendo um pico no tempo:

Sugestão:

- Verifique o que produz (0 0 4 0). [1:0 1:180 1:0 1:180]

- Verifique em:

http://betterexplained.com/examples/fourier/?cycles=0,1

Transformada Completa

A Transformada de Fourier constrói a receita frequência-por-frequência:

- Separa o sinal completo (a b c d) em : (a 0 0 0) (0 b 0 0) (0 0 c 0) (0 0 0 d);
- Para qualquer frequência (como 2Hz), a receita 'tentativa' é 'a/4 + b/4 + c/4 + d/4' (a potência (intensidade) de cada pico é dividida ao longo de todas a frequências)
- Espere ! Precisamos deslocar cada fase com uma compensação de fase (o ângulo para um 'atraso de 1 segundo' depende da frequência).
- A receita verdadeira então é = a/4 (sem compensação) +b/4 (1 segundo de compensação) + c/4 (2 segundos de compensação)+ d/4 (3 segundos de compensação).
- Podemos então iterar por cada frequência para obtermos a transformada completa.

Transformada Completa

- Nosso sinal é simplesmente vários picos de tempo !

Exemplo: Para conseguirmos a resposta (4 2 6 1).

- Precisamos identificar as frequências que geram (4 0 0 0), (0 2 0 0), (0 0 6 0), (0 0 0 1).
- Vamos ao programa e obter as frequências que a geram esses resultados:
- (4 0 0 0)=[1 1 1 1]
- (0 2 0 0)=[0.5 0.5:-90 0.5:180 0.5:90]
- (0 0 6 0)=[1.5 1.5:180 1.5 1.5:180]
- (0 0 0 1)=[0.25 0.25:90 0.25:180 0.25:-90]

Somando termo a termo obtemos:

- (4 2 6 1) = [3.25 0.56:-153.4 1.75 0.56:153.4]

Transformada Completa

Detalhamento: Soma vetorial (Módulo: Ângulo)

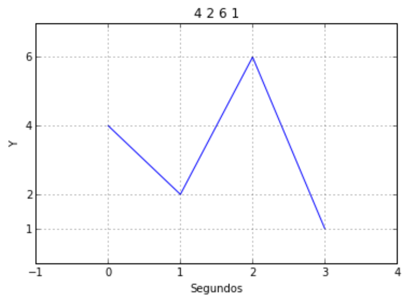
Primeiro termo (0Hz): {1,0}+ {0.5,0} + {1.5,0} + {0.25,0} = {3.25,0} (Não tem diferença de fase).

Segundo termo(1Hz): 1 + 0.5:-90 + 1.5:180 + 0.25:90 = {1,0}+{0,-0.5}+{-1.5,0}+{0,0.25} = {-0.5,-0.25} = 0.56:-153.4 (Módulo:Ângulo)

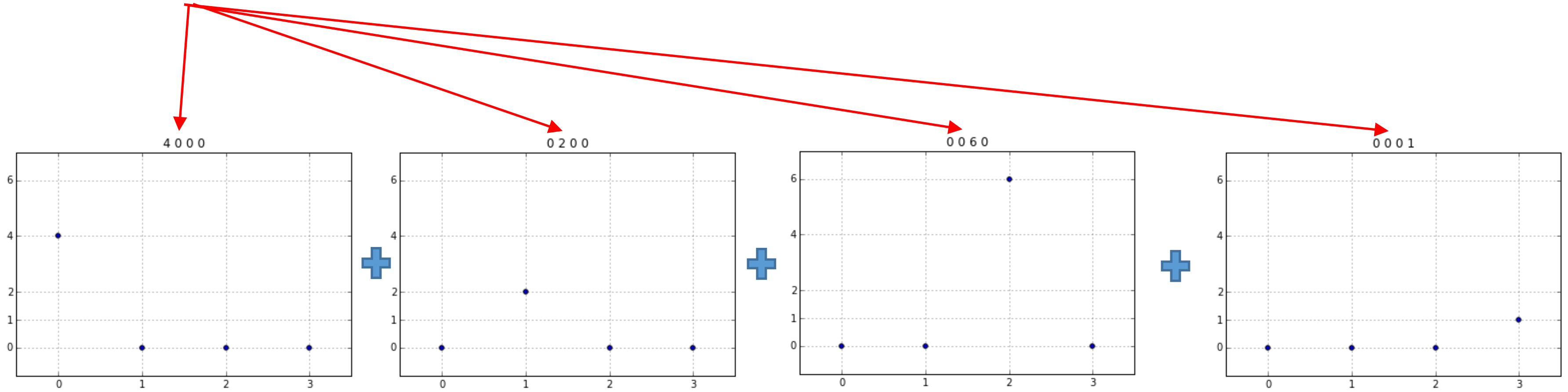
Terceiro termo(2Hz): 1+ 0.5:180+1.5+0.25:180 = 1 – 0.5 + 1.5 – 0.25 = 1.75

Quarto termo(3Hz): 1 + 0.5:90 + 1.5:180 + 0.25:-90 = {1,0} + {0,0.5} + {-1.5,0} + {0,-0.25} = {-0.5,0.25} = 0.56:-153.4

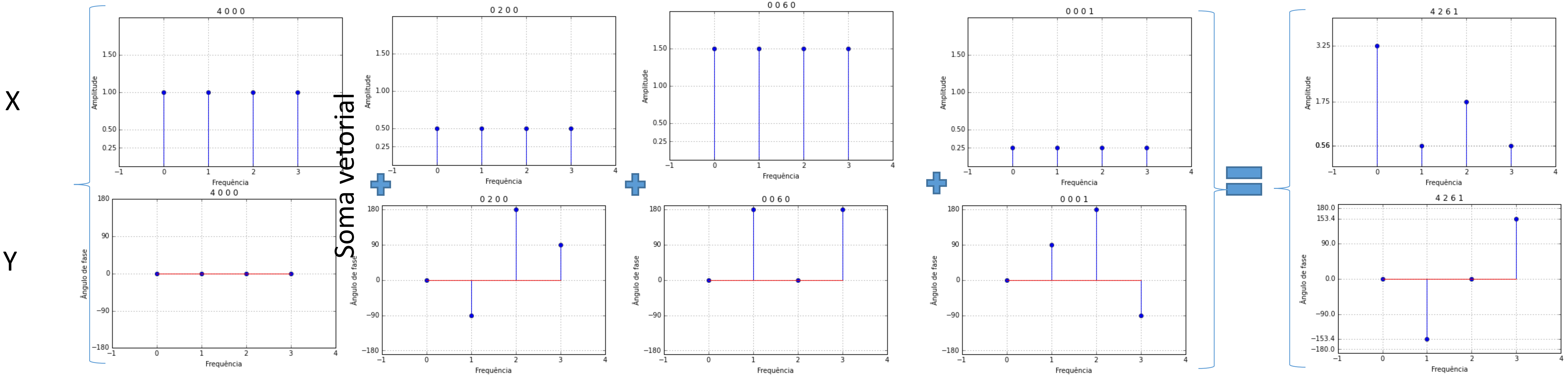
Transformada Completa



- Domínio Temporal;



- frequências, ângulo de fase.



Transformada Completa

Vamos então converter do 'português matemático' para a matemática pura:

Transformada Completa

- Receita de Frequências = Somatório das contribuições para essa frequência (k), para cada pico de tempo(n).
- $X_k = \sum_{n=0}^{N-1} \mathrm{x_n} \times e^{-i2\pi kn/N}$

- $\mathrm{x_n} = \frac{1}{N} \times \sum_{k=0}^{N-1} \mathrm{X_k} \times e^{i2\pi kn/N}$
- Ponto temporal = Somatório das contribuições para esse tempo(n), para cada frequência(k).

Notas:

- N = Número de observações temporais que temos;
- n = amostra 'atual' que estamos considerando (0...N-1);
- x_n = valor do sinal no tempo n;
- k = frequência 'atual' que estamos considerando (0Hz até N-1 Hz);
- X_k = quantidades de frequências k no sinal (amplitude e fase, um número complexo)

Notas:

- O fator **1/N é usualmente passado para a transformada inversa** (de frequências para tempo). É preferido, no entanto alguns autores preferem **1/N na Transformada uma vez que resulta em tamanhos verdadeiros para os picos temporais**. Se preferir pode usar 1/sqrt(N) em ambas as transformadas (Transformando e Invertendo ainda resulta no fator 1/N).

Notas:

- n / N é a porcentagem do tempo que passou .
- 2 * pi * k é a nossa velocidade em radianos / seg.
- e ^ -ix é o nosso movimento para trás do trajeto circular.
- A combinação é o quão longe nós nos movemos , por isso a velocidade e o tempo .

Notas:

- As equações puras para a transformada de Fourier apenas dizem " adicionar os números complexos " . Muitas linguagens de programação não pode lidar com números complexos diretamente , assim convertem tudo para coordenadas retangulares e as adicionam.

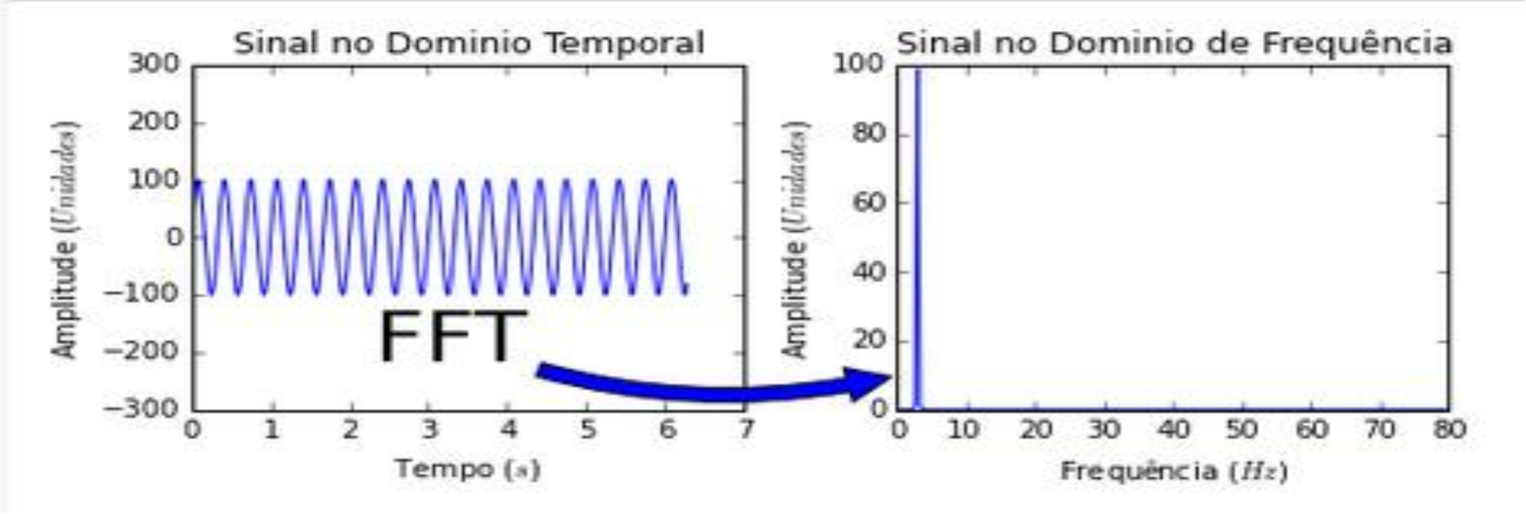
Exemplo 1 - Transformada

Função: F(t) = A * sin(2*pi*f*t)

A = 100 Amplitude em unidades

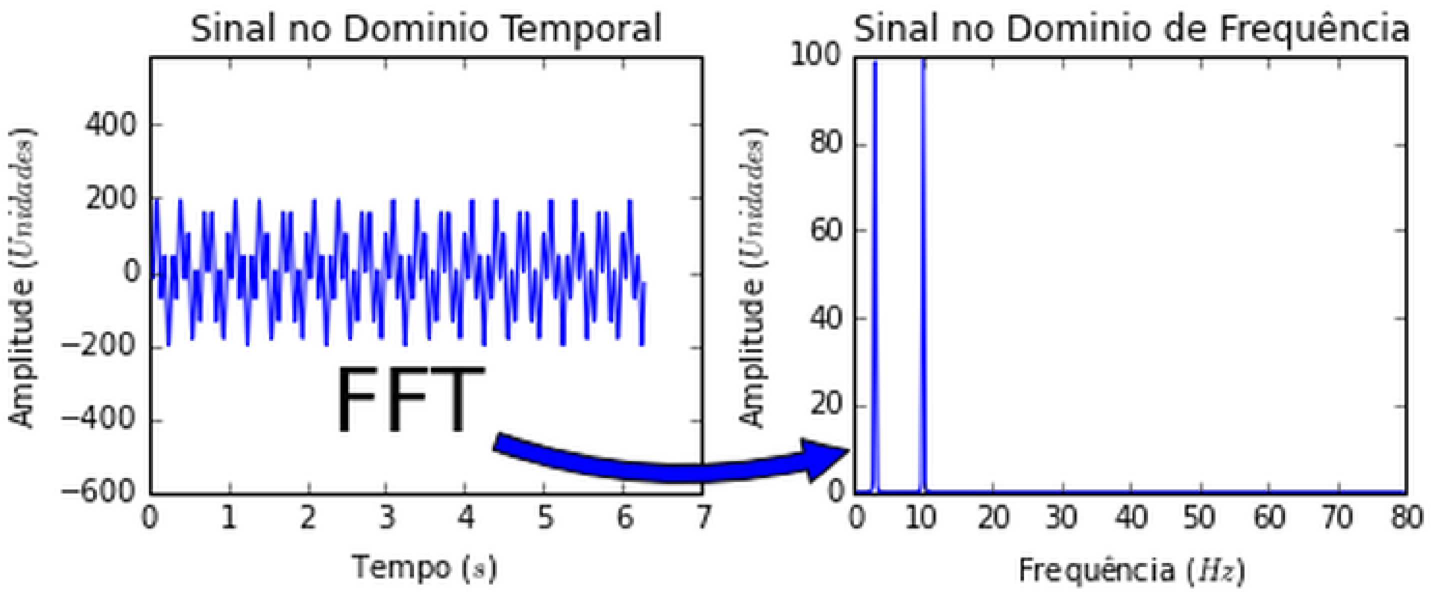
f = 3 Hz

t = de 0 a 2*Pi dividido em 1000 intervalos.



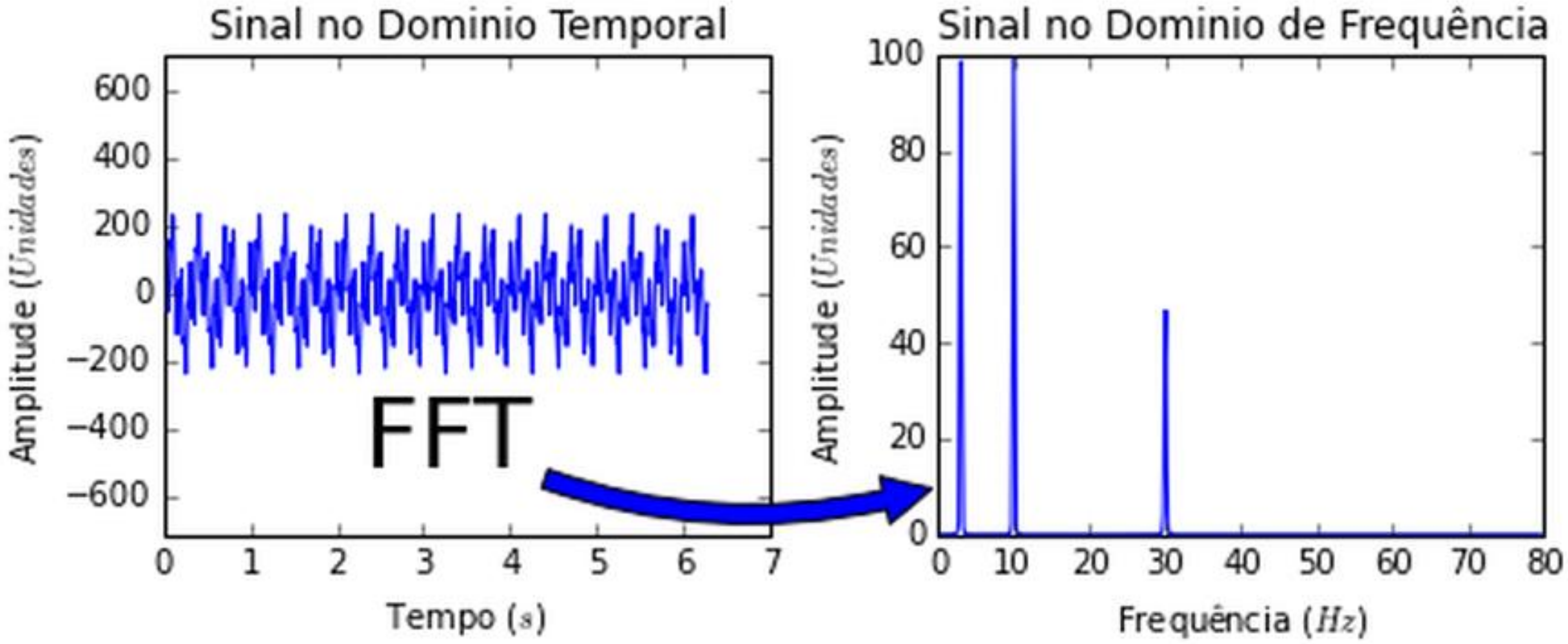
Exemplo 2

- F(t) = A * sin(2*pi*f*t) + B * cos(2*pi*f2*t)
- f = 3.0 Hz
- f2 = 10.0 Hz
- A = 100.0 Amplitude em unidades
- B = 100.0



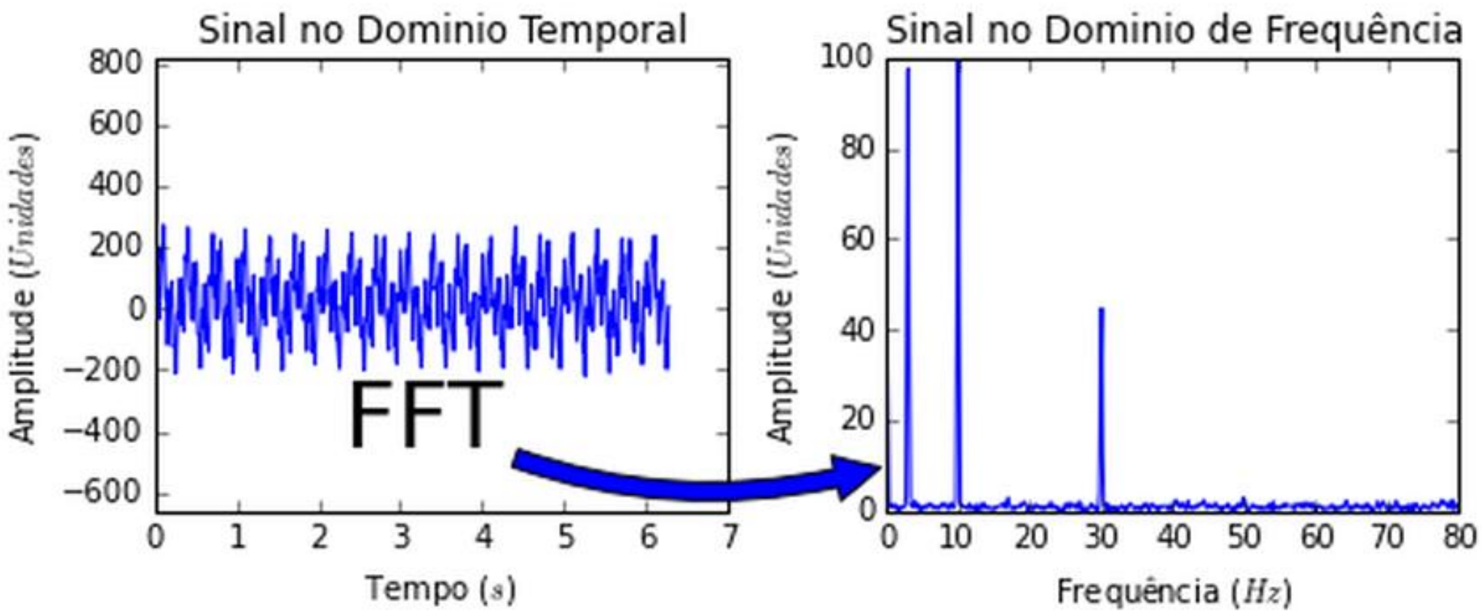
Exemplo 3

- F(t) = A * sin(2*pi*f*t) + B * cos(2*pi*f2*t) + C * sin(2*pi*f3*t)
- f = 3.0 # Frequência em Hz
- f2 = 10.0
- f3 = 30.0
- A = 100.0 Un
- B = 100.0
- C = 50.0



Exemplo 4 – Adicionando Ruído

- F(t) = A * sin(2*pi*f*t) + B * cos(2*pi*f2*t) + C * sin(2*pi*f3*t) + np.random.rand(len(t))*50
- f = 3.0 # Frequência em Hz
- f2 = 10.0
- f3 = 30.0
- A = 100.0 # Amplitude
- B = 100.0
- C = 50.0



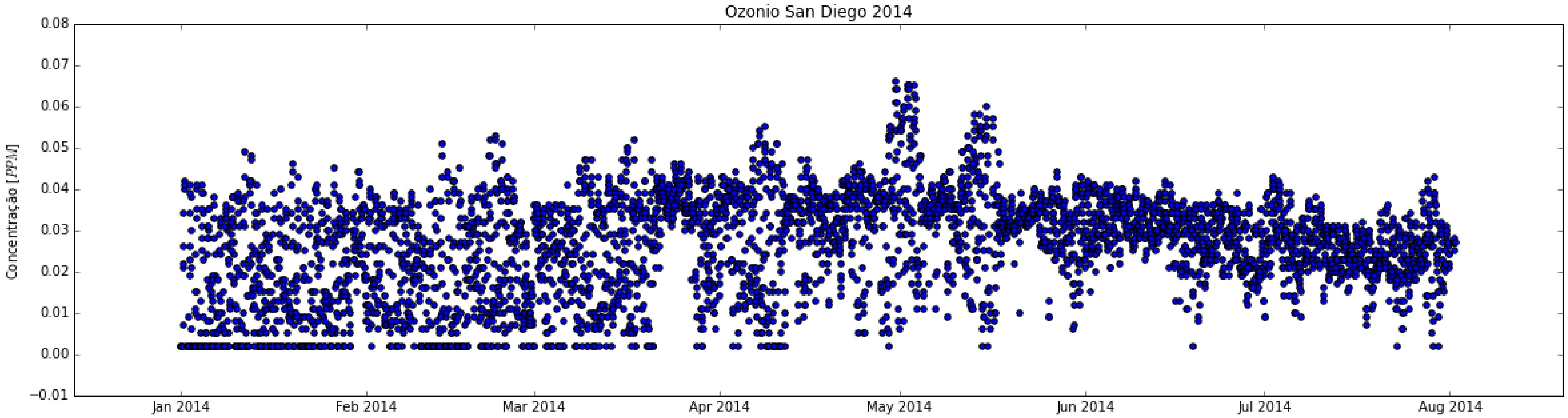
Exemplo 5

- Ozônio San Diego, CA, USA. Jan-Ago 2014. Estação: Chula Vista.
- Dados: http://www.epa.gov/airquality/airdata/ad_data.html
- 32.631231 N -117.059075 W
- Dados Horários:

06:00:00	06:00:00												million		
2014-01-01 07:00:00	2014-01-01 07:00:00	6	73	1	44201	1	32.631231	-117.059075	WGS84	Ozone	...	0.002	Parts per million	0.005	N
2014-01-01 08:00:00	2014-01-01 08:00:00	6	73	1	44201	1	32.631231	-117.059075	WGS84	Ozone	...	0.007	Parts per million	0.005	N
2014-01-01 09:00:00	2014-01-01 09:00:00	6	73	1	44201	1	32.631231	-117.059075	WGS84	Ozone	...	0.021	Parts per million	0.005	N
2014-01-01 10:00:00	2014-01-01 10:00:00	6	73	1	44201	1	32.631231	-117.059075	WGS84	Ozone	...	0.022	Parts per million	0.005	N
2014-01-01 11:00:00	2014-01-01 11:00:00	6	73	1	44201	1	32.631231	-117.059075	WGS84	Ozone	...	0.026	Parts per million	0.005	N
2014-01-01 12:00:00	2014-01-01 12:00:00	6	73	1	44201	1	32.631231	-117.059075	WGS84	Ozone	...	0.034	Parts per million	0.005	N
2014-01-01 13:00:00	2014-01-01 13:00:00	6	73	1	44201	1	32.631231	-117.059075	WGS84	Ozone	...	0.042	Parts per million	0.005	N
2014-01-01	2014-01-01												Parts per		

Exemplo 5 - Entrada

- Intervalos tempo entre amostras devem ser iguais;
- N/A aparentes, são concentrações Zero (0,002) ou erros de registro;
- Para não termos que interpolar vamos mantê-las e supor com licença de aprendizado que sejam realmente o que indicam;



Exemplo 5 – Passos Importantes

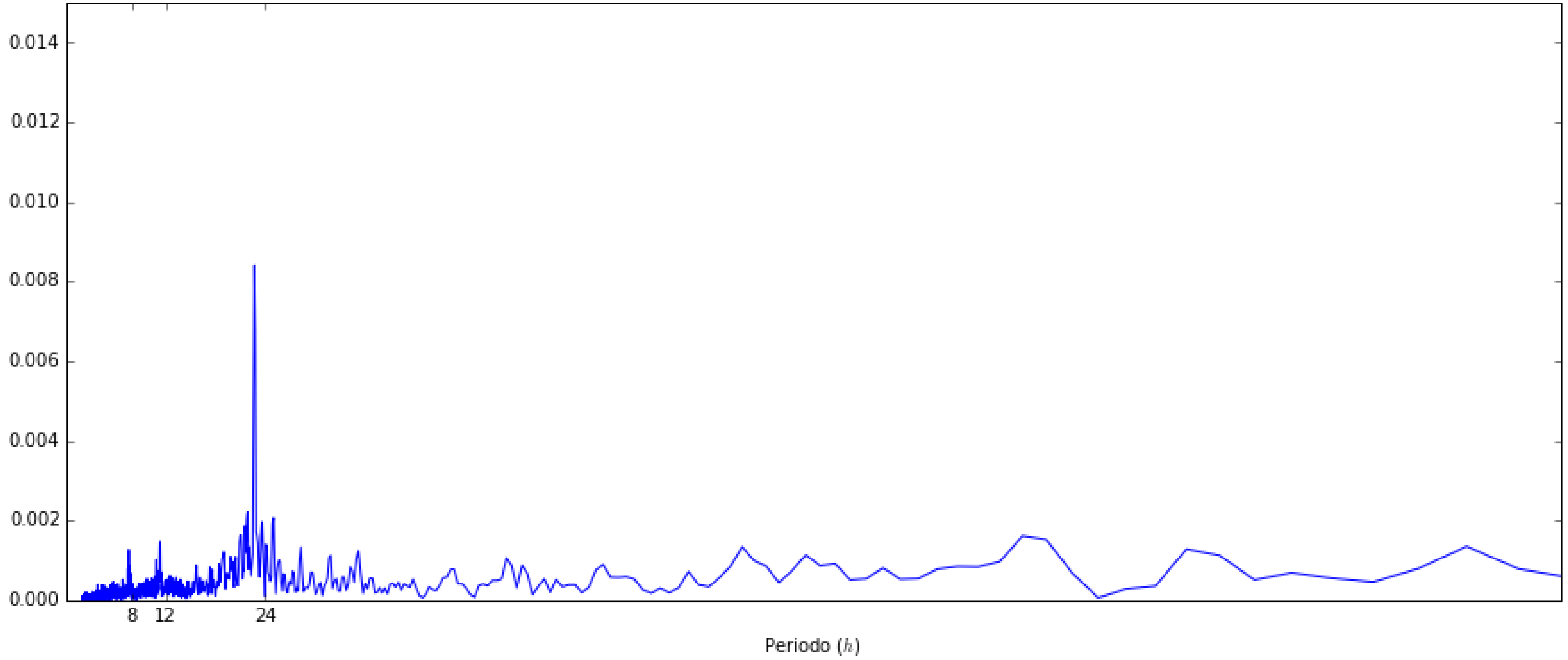
- Arrumar eixo x com frequência de Nyquist;
- Arrumar eixo y com funções janela;
- Verificar como em: balzer82/FFT-Python

Exemplo 5 - Resultados

- Neste caso analisar frequências é estranho;
- Transformamos o eixo x de ‘Frequência’ para ‘Período’ plotando 1/f no lugar de f.

Frequências Importantes:

- O ozônio no local de estudo tem claramente uma variação diária de concentrações como indicado pelo pico de 24hrs.
- Isso pode ser explicado teoricamente pelo ciclo de formação do ozônio troposférico. Com maior geração nos horário de maior incidência de radiação e emissão de COVs e NOxs.
- Os pequenos picos em 8hrs e 12hrs podem ser investigados.



Comentários

Transformadas de Fourier tem várias faces:

- Discreta;
- Continua;
- Uniforme;
- Não Uniforme;
- Permeia muitas partes da matemática avançada;
- Em alguns currículos é conteúdo para um trimestre ou 2;
- PPGMA;

Vessyl – Smart Cup. Obtém a composição, o tipo e as calorias da ‘Vitamina’ que está tomando.

Alguém se arrisca em chutar como funciona ?

Myvessyl.com (2015)

Referências

- Betterexplained.com,. An Interactive Guide To The Fourier Transform | BetterExplained. Disponível em: <http://betterexplained.com/articles/an-interactive-guide-to-the-fourier-transform/>. Acesso em: 1/ 6/ 2015.
- Wolframalpha.com,. fourier transform - Wolfram|Alpha. Disponível em: <http://www.wolframalpha.com/input/?i=fourier+transform>. Acesso em: 1/ 6/ 2015.
- Silva Junior, R.; Andrade, M. Validação de poluentes fotoquímicos e inclusão do inventário de emissões no modelo de qualidade do ar WRF/CHEM, para a região metropolitana de São Paulo. Revista Brasileira de Meteorologia, v. 28, n. 1, p. 105-121, 2013.
- Mares.io.usp.br,.Disponível em: <http://www.mares.io.usp.br/sudeste/sudeste.html>. Acesso em: 1/ 6/ 2015.
- Gemcorpltd.com,. Seismic Vibrations & Offset. Disponível em: <http://gemcorpltd.com/ver1/index.php?option=com_content&view=article&id=62&Itemid=72>. Acesso em: 16/ 6/ 2015.
- Briggs, W.; Henson, V. The DFT. Philadelphia, Pa.: SIAM, 1995.

Referências

- GitHub,. balzer82/FFT-Python. Disponível em: <https://github.com/balzer82/FFT-Python>. Acesso em: 1/ 6/ 2015.