

# Deep learning

Jorge Centeno





# Conceito de Deep Learning

Está focado em uso de redes neurais com muitas em camadas (deep) usando o algoritmo backpropagation, que aprendem as características diretamente a partir dos dados sem a necessidade de extração manual.

As técnicas mais populares são:

- redes de Perceptron multicamadas.
- **Convolutional neural networks**
- **Autoencoders**
- ...
- Restricted Boltzmann machines
- Long Short-Term Memory LSTM

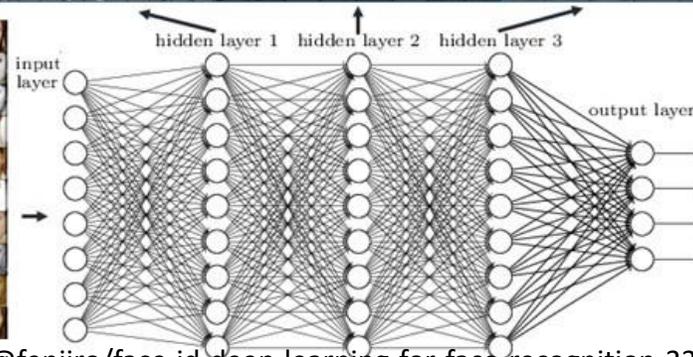
# Mais dados + mais computadores!

*Deep learning* requer grandes quantidades de dados rotulados para garantir o bom aprendizado.

Por exemplo, o reconhecimento de pessoas em uma imagem requer milhões de imagens com exemplos de pessoas em diferentes escalas e poses.

Também requer substancial poder de computação.

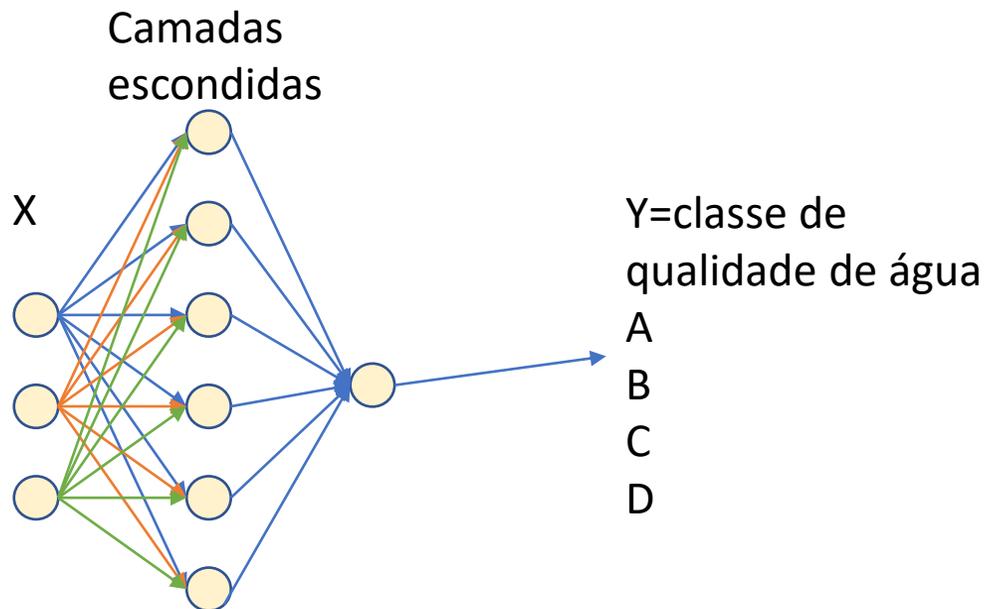
Deep neural networks learn hierarchical feature representations



<https://medium.com/@fenjiro/face-id-deep-learning-for-face-recognition-324b50d916d1>

# Lembrando redes neurais

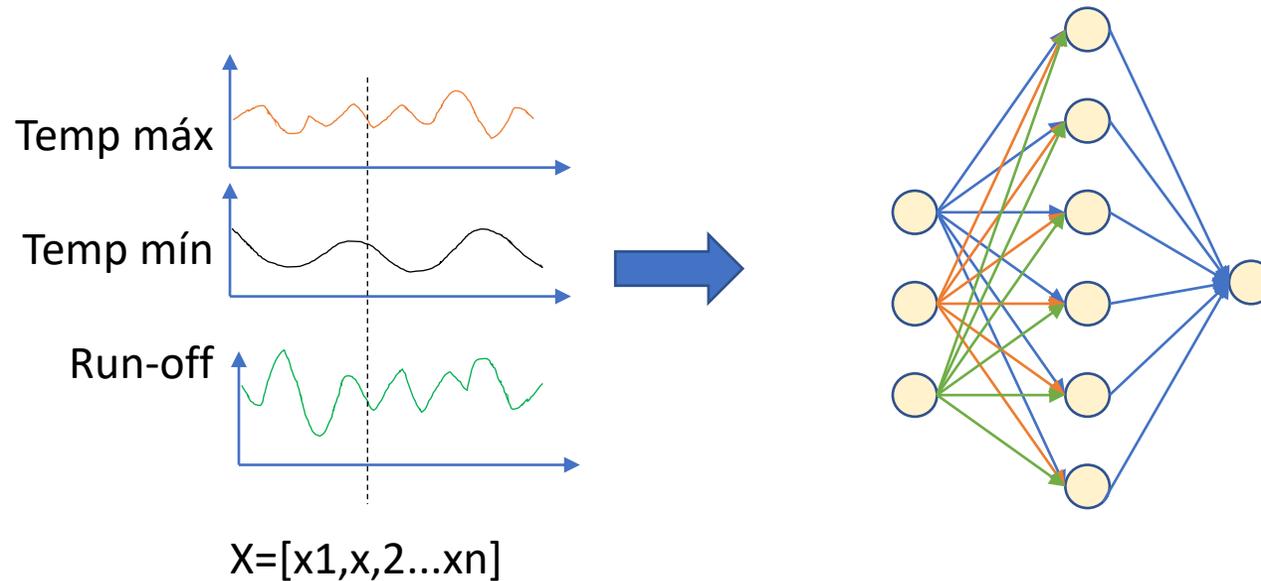
X	Parameters	
1	pH	0.025899
2	Turbidity, NTU	0.044028
3	Total Dissolved Solids	0.00044
4	Total Alkalinity	0.001101
5	Total Hardness	0.000734
6	Nitrate	0.004892
7	Chloride	0.000881
8	Calcium	0.002935
9	Magnesium	0.007338
10	Iron	0.733805
11	Fluoride	0.220142



Cada elemento é apresentado separadamente.  
Podem ser vetores longos com muitas variáveis.

# Em um série de dados

Se apresentamos cada ponto isoladamente, os elementos são ponderados separadamente. Perde-se o contexto, os conceitos de estrutura espacial/temporal.



# Em um série de dados

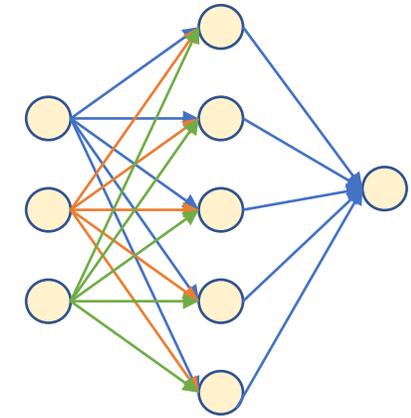
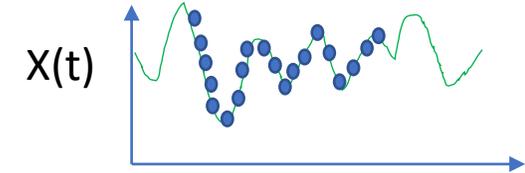
Mas e se ... usamos uma arquitetura que considera a estrutura espacial/temporal?

Ou seja, a interrelação entre as amostras, seja esta espacial ou temporal.

Neste caso, a rede pode também descobrir padrões de variação. Por exemplo, formas em fotografias ou oscilações em séries temporais.



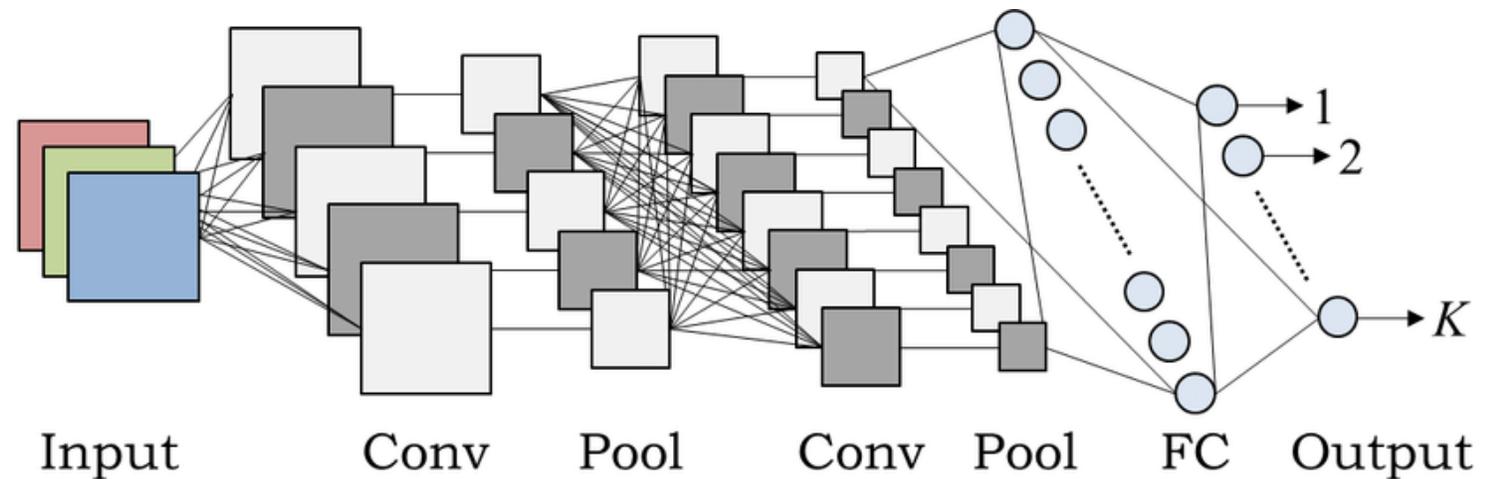
$$X=[x_1,x_2...x_t]$$



# Redes neurais convolucional

usam três ideias básicas:

- campos receptivos locais,
- pesos compartilhados e
- Combinação de saídas (pooling).

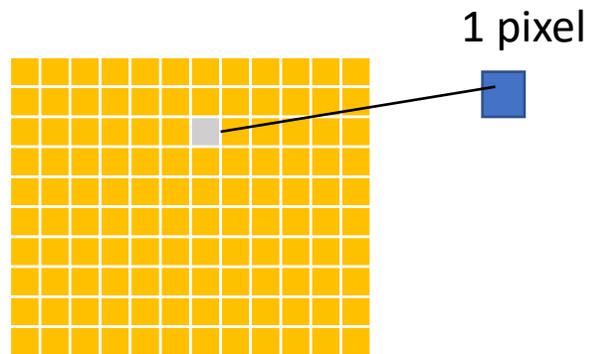


# Campos receptivos locais

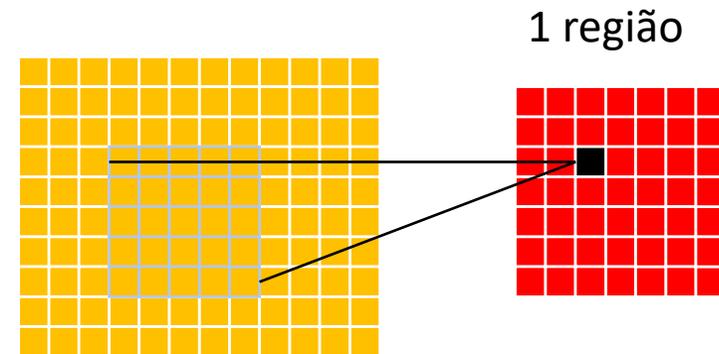
IMAGEM:

Em uma rede convolucional, a entrada é “reformatada” sob forma de uma janela móvel, um quadrado (filtro) de neurônios que captam a variação local.

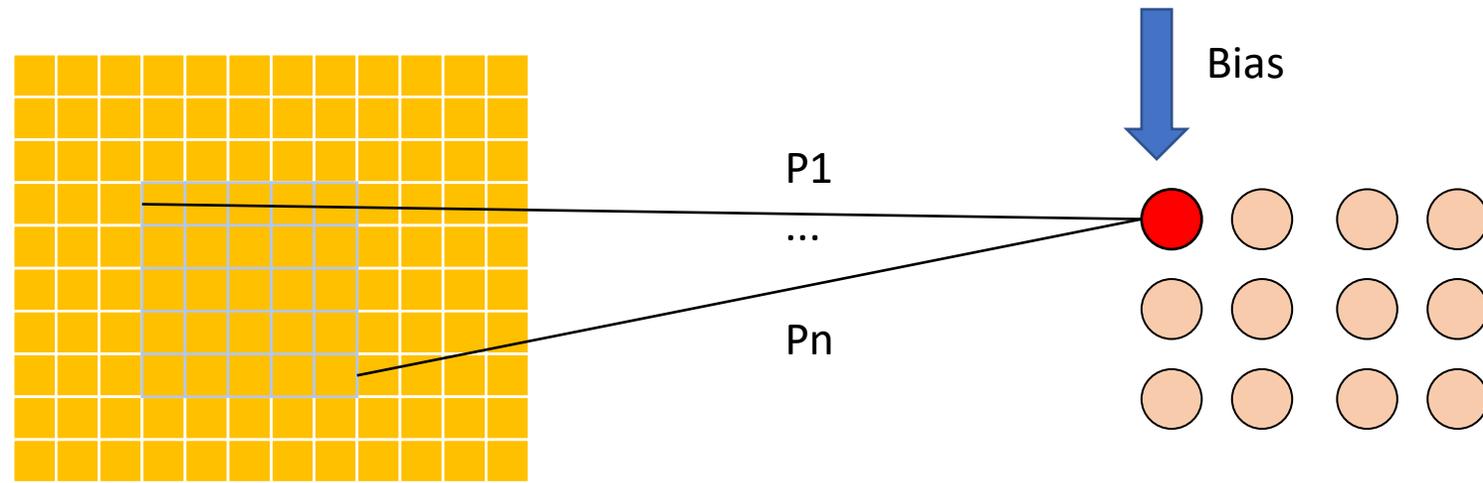
Não apenas o valor de um pixel.



Não



SIM



Cada elemento da matriz (filtro) aprende um peso

O neurônio oculto adiciona o bias.

O neurônio oculto aprende a analisar o seu campo receptivo local particular.

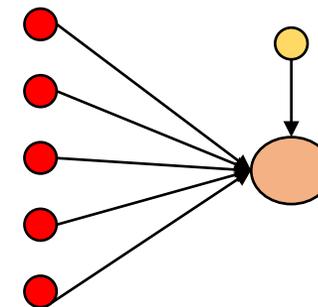
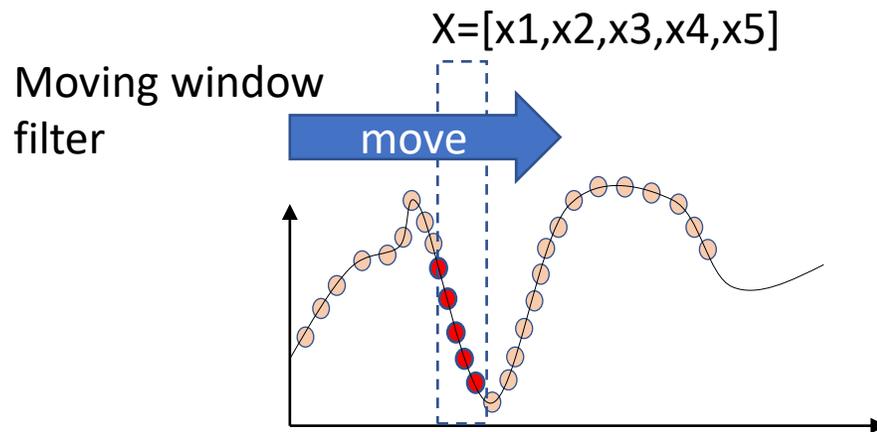
Para cada campo receptivo local, há um neurônio oculto diferente na primeira camada oculta.

# Campos receptivos locais

Série de dados:

Em uma rede convolucional, a entrada é “reformatada” sob forma de uma janela móvel, um linha (filtro) de neurônios que captam a variação local. Não apenas o valor de um evento.

O neurônio adiciona o bias e analisa o campo receptivo local particular.





# Convolução

É a operação de duas funções: a primeira é a série e a segunda (um filtro) que se desloca ao longo da série calculando um resultado para cada posição possível.

Convolução: A partir de duas funções dadas, resulta numa terceira que mede a soma do produto dessas funções ao longo da região de superposição delas em função do deslocamento da segunda ao longo da primeira.

$$(f * g)(k) = \sum_{j=0}^k f(j) \cdot g(k - j)$$

# Convolução

Considere a seguinte série e opere com (convolução) com os filtros:

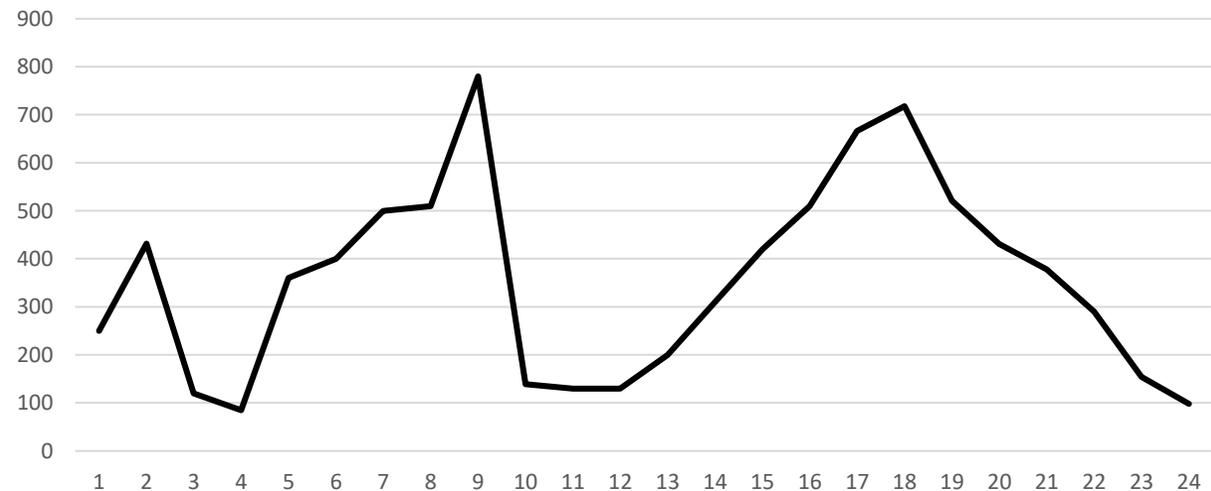
$F1 = [-1 \ 0 \ 1]$  : anterior - posterior

$F2 = [1 \ 0 \ -1]$  : posterior - anterior

pto	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
v	250	432	120	85	360	400	500	510	780	139	130	130	200	310	420	510	666	718	521	431	378	290	154	98

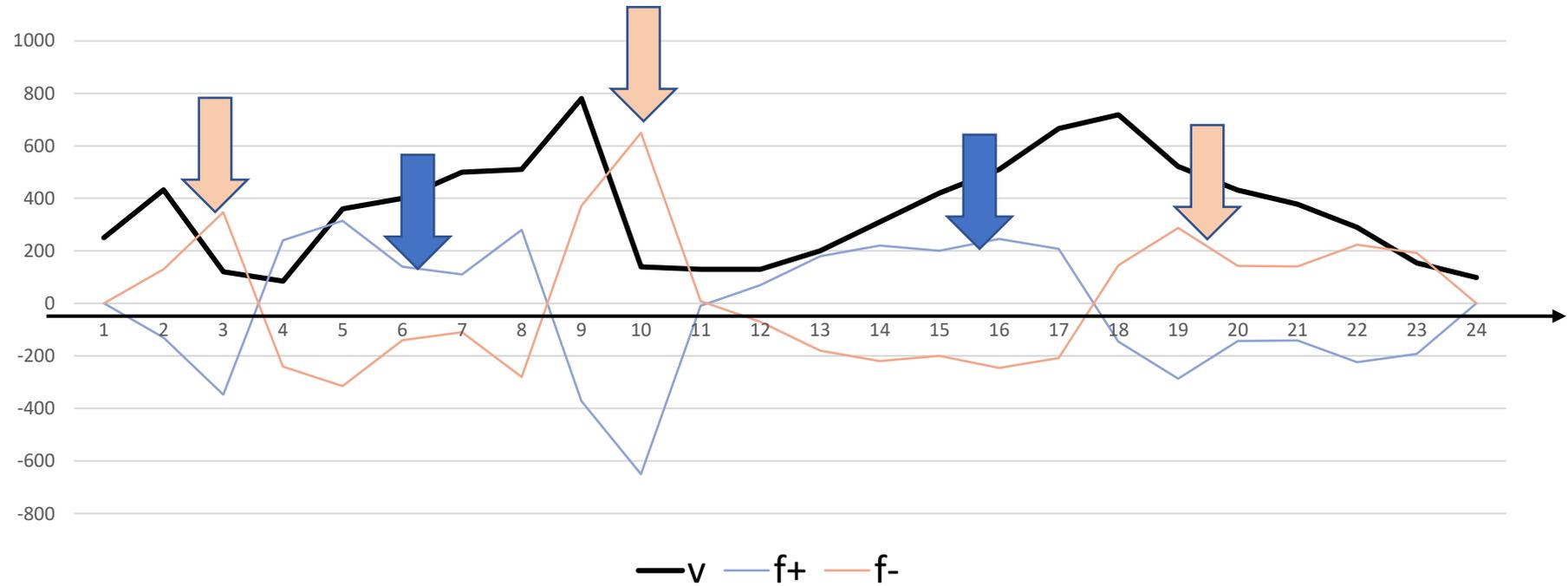
Exemplo: para a segunda posição:

$$R1 = (-1) * 250 + (1) * 120 \\ = 250 - 120 = -130$$

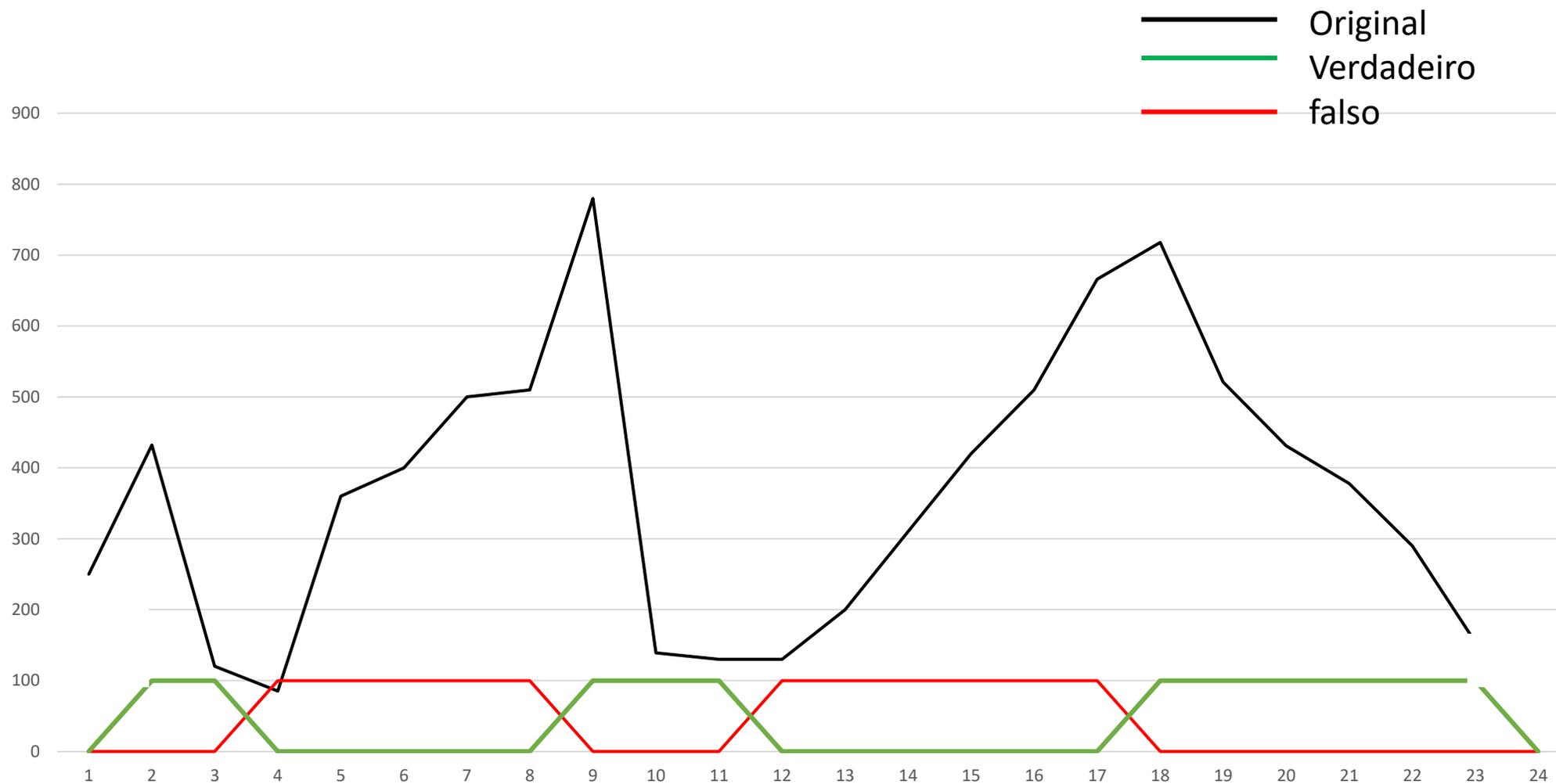




v	250	432	120	85	360	400	500	510	780	139	130	130	200	310	420	510	666	718	521	431	378	290	154	98
f+	XXX	-130	-347	240	315	140	110	280	-371	-650	-9	70	180	220	200	246	208	-145	-287	-143	-141	-224	-192	X
f-	XXX	130	347	-240	-315	-140	-110	-280	371	650	9	-70	-180	-220	-200	-246	-208	145	287	143	141	224	192	X



# TRUE

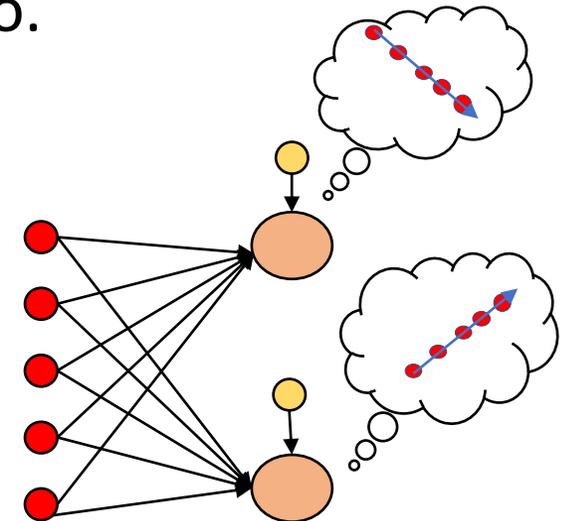
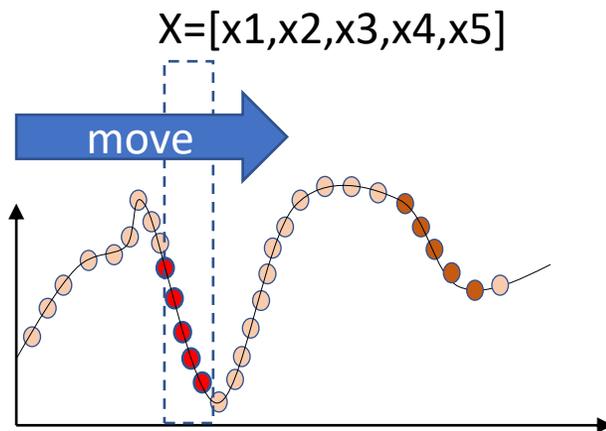


# Propriedades

Todos os neurônios na primeira **camada oculta** detectam exatamente a mesma feição, apenas em locais diferentes na imagem de entrada.

Os pesos permitem que o neurônio oculto possa identificar, digamos, um trecho ascendente na série (1D), ou uma borda vertical (2D).

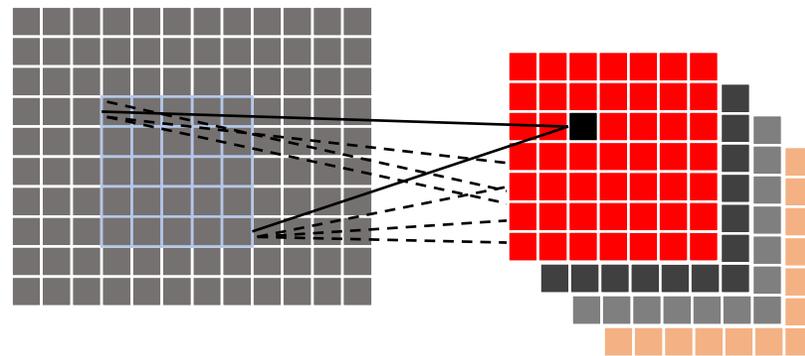
redes convolucionais são invariantes à translação.



# Mapas de feições

No reconhecimento de padrões em imagens são necessários mais de um mapa de feições. Por isso, uma camada convolucional completa consiste de vários diferentes mapas de feições.

No reconhecimento de padrões em séries os mapas de feições são 1D.



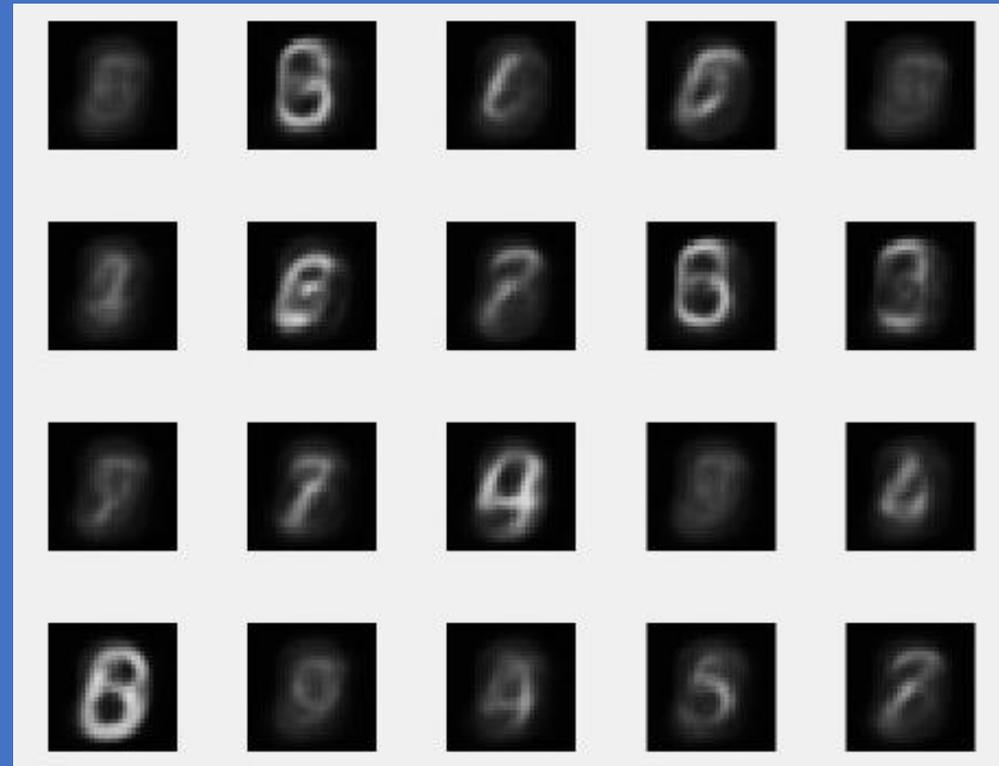
convolutional layer

Uma região é analisada com cada filtro e o resultado é armazenado sob forma de uma nova grade (imagem). A camada convolucional. Cada filtro gera uma camada convolucional.

# Exemplo

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9

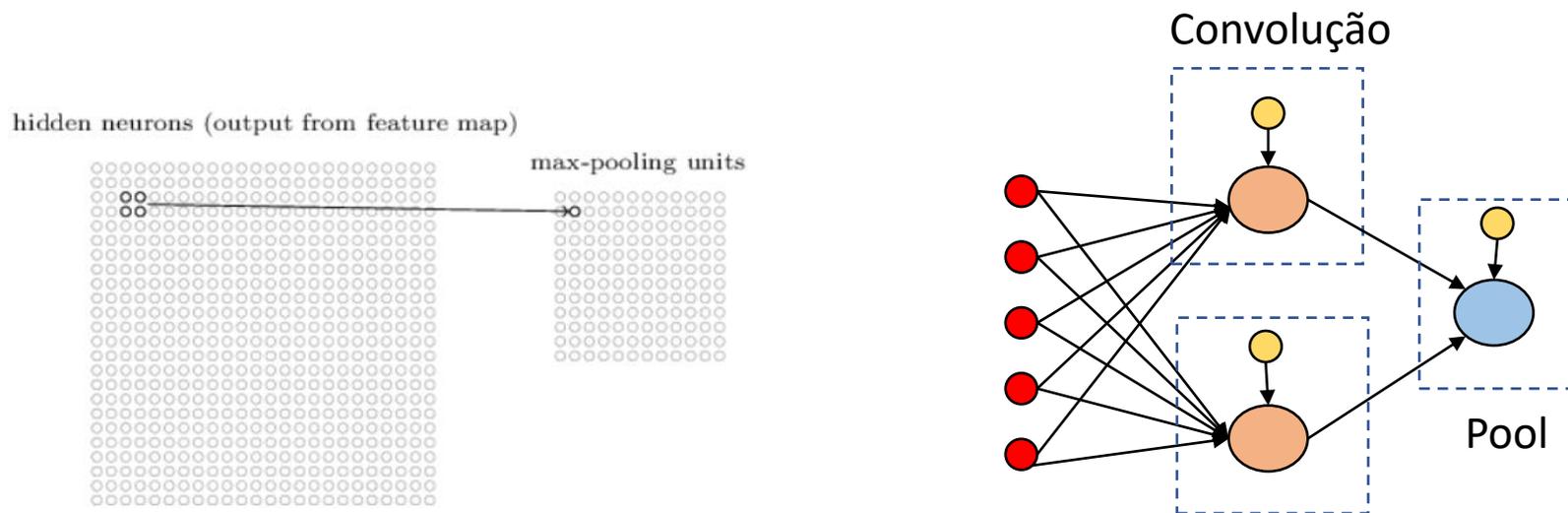
## REDE: FILTROS



# Pooling layer

Após as camadas convolucionais, as redes convolucionais também contêm camadas de agregação (pooling).

Uma camada de agregação recebe cada saída dos mapas de feições da camada convolucional e prepara um mapa de feições resumido.

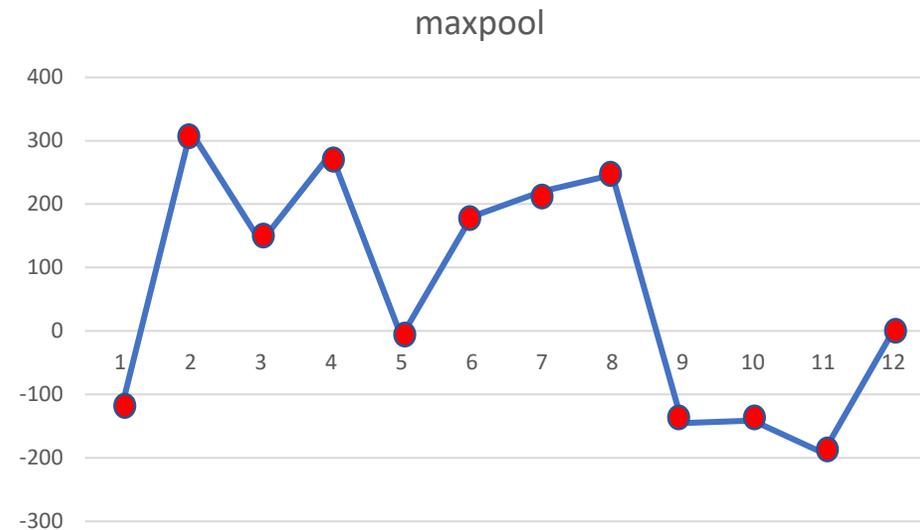
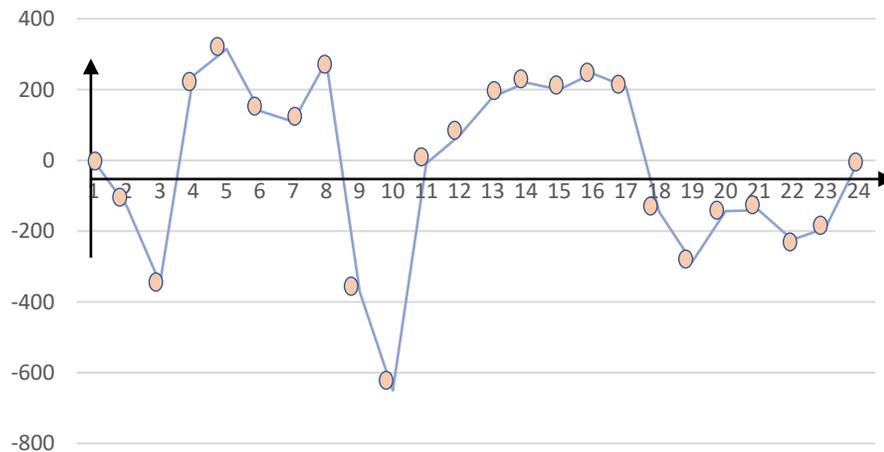


# Pooling layer: camada de agregação

O resultado é uma função mais simples (resumo)

Exemplo: escolher o maior a cada dois elementos

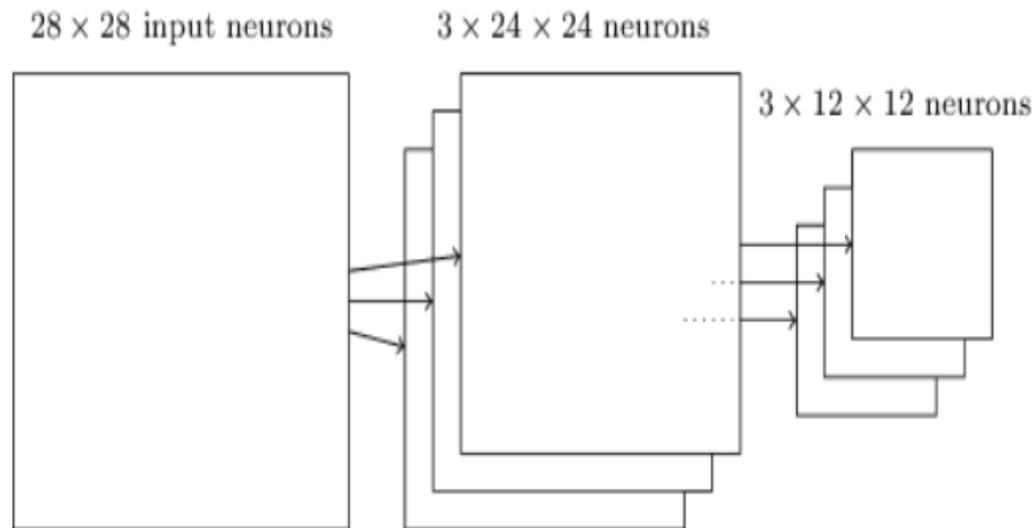
MAXPOOL



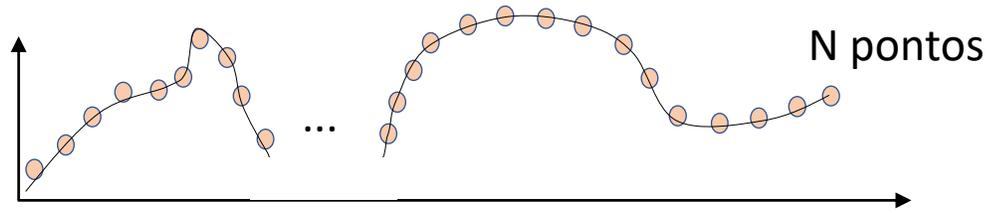
Resume o formato da saída...

# Pooling

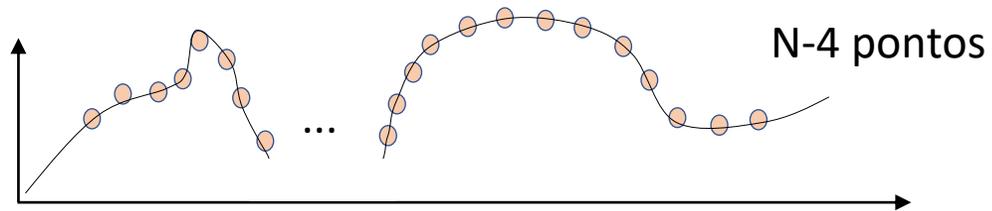
A camada convolucional geralmente envolve mais do que um único mapa de feições. Por isso a regra do máximo (max pool) é aplicada a cada mapa de feições separadamente.



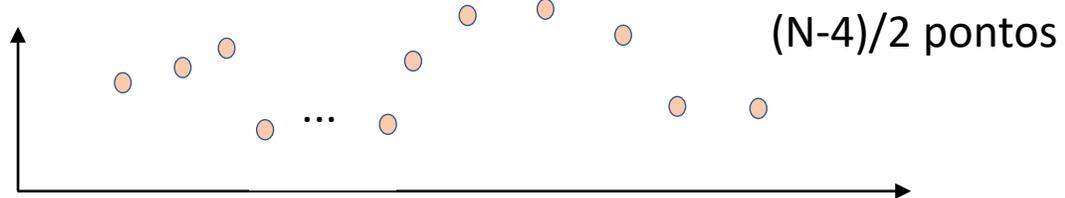
# Pooling



Filtro 5x1 



Pooling (ex: Máximo de 2)



A convolução reduz um pouco a série de dados, visto que os elementos nos extremos não podem ser processados.

Pooling reduz severamente a quantidade de pontos.



max pool serve para avaliar: “com que intensidade um determinado padrão é encontrado em qualquer lugar em uma região da imagem.

O sistema descarta as informações posicionais exatas.

Considera que se uma feição foi encontrada, sua localização exata não é tão importante quanto a sua localização grosseira em comparação com outras feições.

Uma grande vantagem é que existem muitos menos recursos resumidos.



## *L2 pooling*

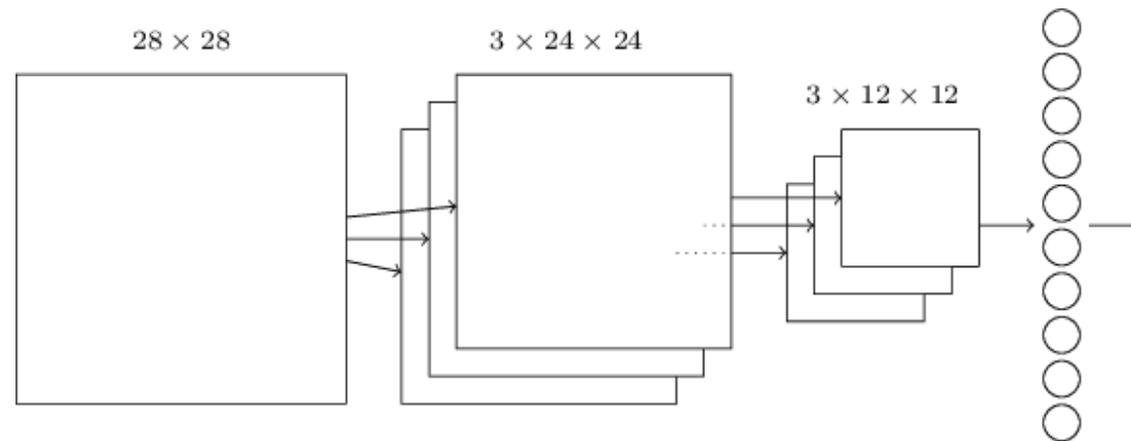
Em lugar de Max-pooling pode-se usar o método L2.

Este adota a raiz quadrada da soma dos quadrados dos valores de ativação da região de neuronios.

# Juntando Tudo

Uma última camada agrupa as possíveis saídas (ex: 10 neurônios de saída );

A rede neural convolucional completa inclui as camadas de convolução, as camadas de pooling e a de saída. Uma série de camadas empilhadas (rede profunda = deep) .





# Water-Quality Classification of Inland Lakes Using Landsat8 Images by Convolutional Neural Networks

Fangling Pu \* , Chujiang Ding, Zeyi Chao, Yue Yu and Xin Xu  
remote sensing: Published: 14 July 2019

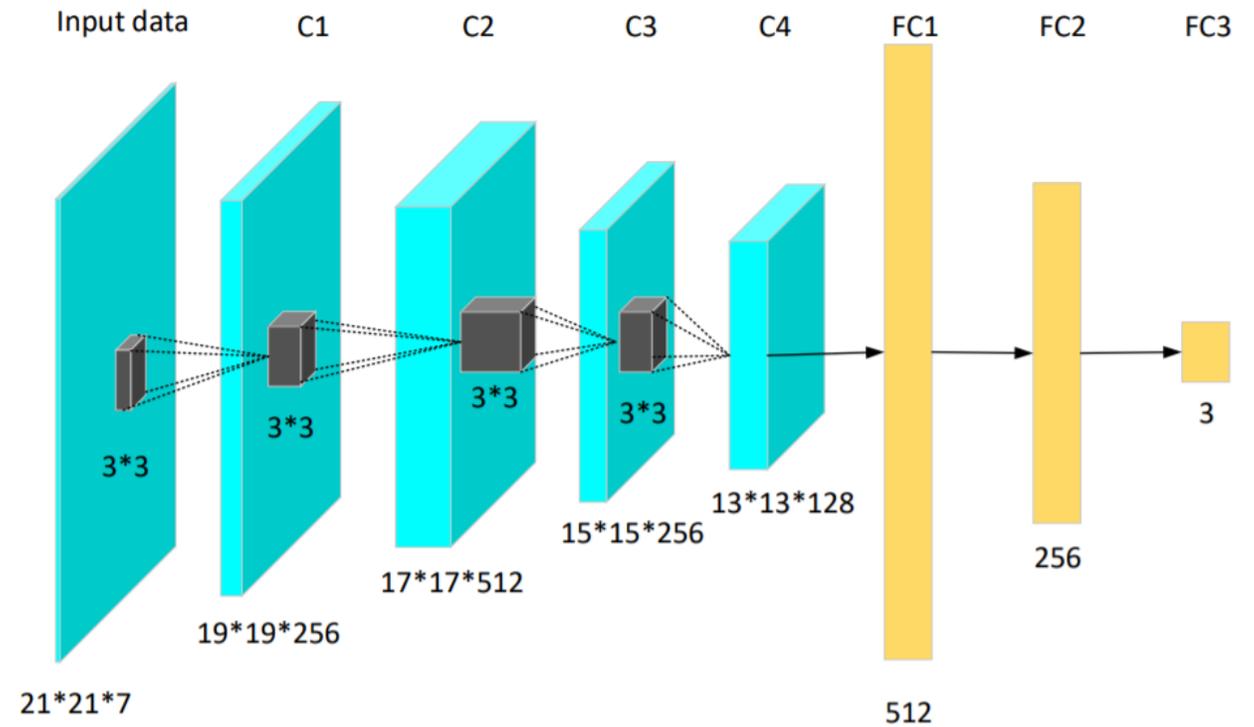
**Abstract:** A convolutional neural network (CNN) with hierarchical structure was designed to represent the relationship between Landsat8 images and in situ water-quality levels. A transfer-learning strategy in the CNN model was introduced to deal with the lack of in situ measurement data. After the CNN model was trained by spatially and temporally matched Landsat8 images and in situ water-quality data that were collected from official websites, the surface quality of the whole water body could be classified. The results demonstrate that the CNN model outperformed widely used machine-learning methods. The trained model at Erhai Lake can be used for the water-quality classification of Chaohu Lake. The introduced CNN model and the water-quality classification method could cover the whole lake with low costs. The proposed method has potential in inland-lake monitoring.

# Rede proposta

C=camada convolucional  
FC= camada Full connected

Entrada: imagem de satélite

Saída: três classes de qualidade de água.





# A Compound Approach for Monthly Runoff Forecasting Based on Multiscale Analysis and Deep Network with Sequential Structure

Shi Chen, Shuning Dong

WATER: <https://www.mdpi.com/2073-4441/12/8/2274>

## **Abstract:**

A novel compound approach combining time-varying filtering-based empirical mode decomposition (TVFEMD), sample entropy (SE)-based subseries recombination, and the **newly developed deep sequential structure incorporating convolutional neural network (CNN)** into a gated recurrent unit network (GRU) is proposed for monthly runoff forecasting.

The runoff series is disintegrated into a collection of subseries adopting TVFEMD, considering the volatility of runoff series caused by complex environmental and human factors. The subseries recombination strategy based on SE and recombination criterion is employed to reconstruct the subseries possessing the approximate complexity.

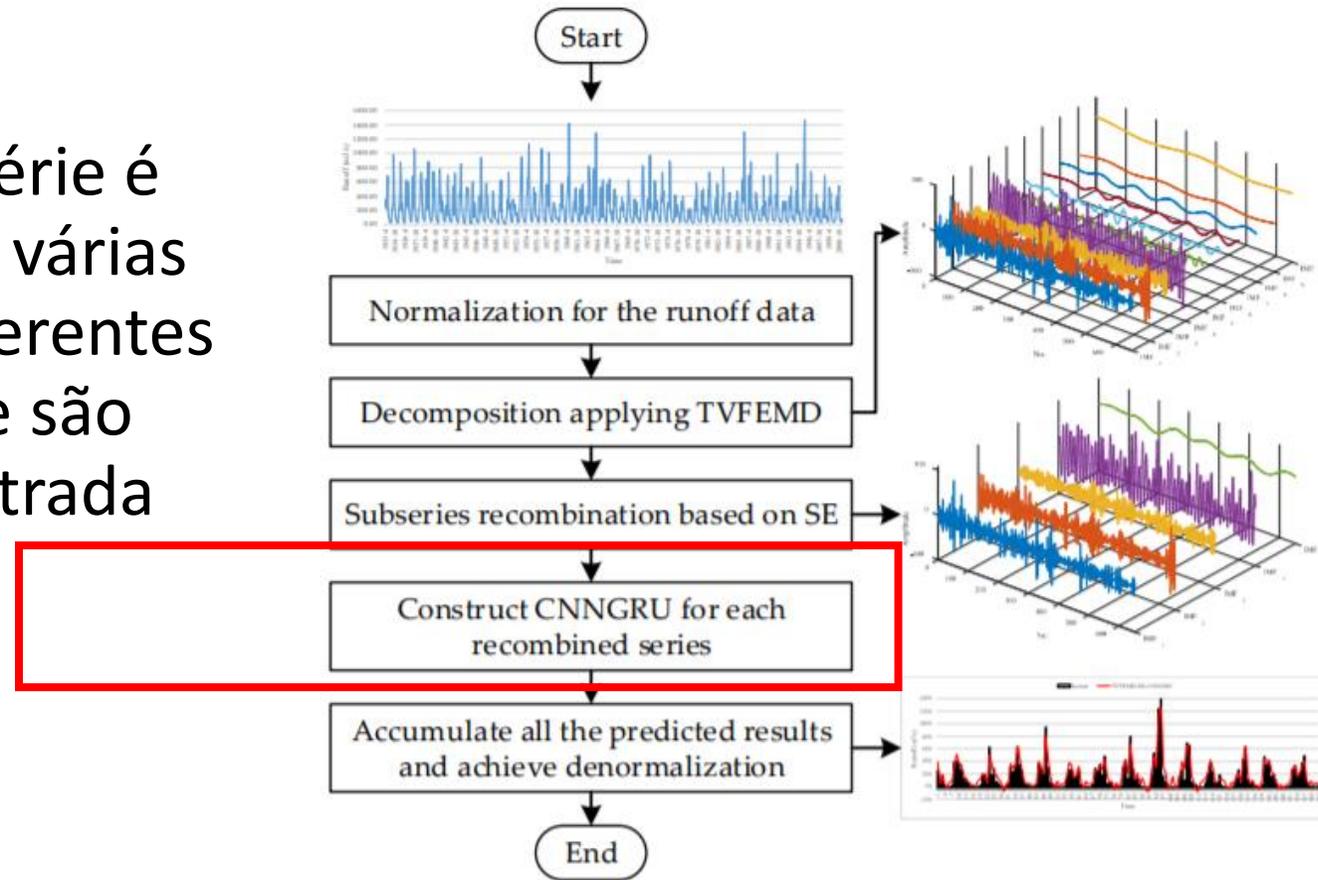
**Subsequently, the newly developed deep sequential structure based on CNN and GRU (CNNGRU) is applied to predict all the preprocessed subseries. Eventually, the predicted values obtained above are aggregated to deduce the ultimate prediction results.**

Eight relevant contrastive models were applied to the monthly runoff series collected from Baishan reservoir, where the experimental results demonstrated that the evaluation metrics obtained by the proposed model achieved an average index decrease of 44.35% compared with all the contrast models.

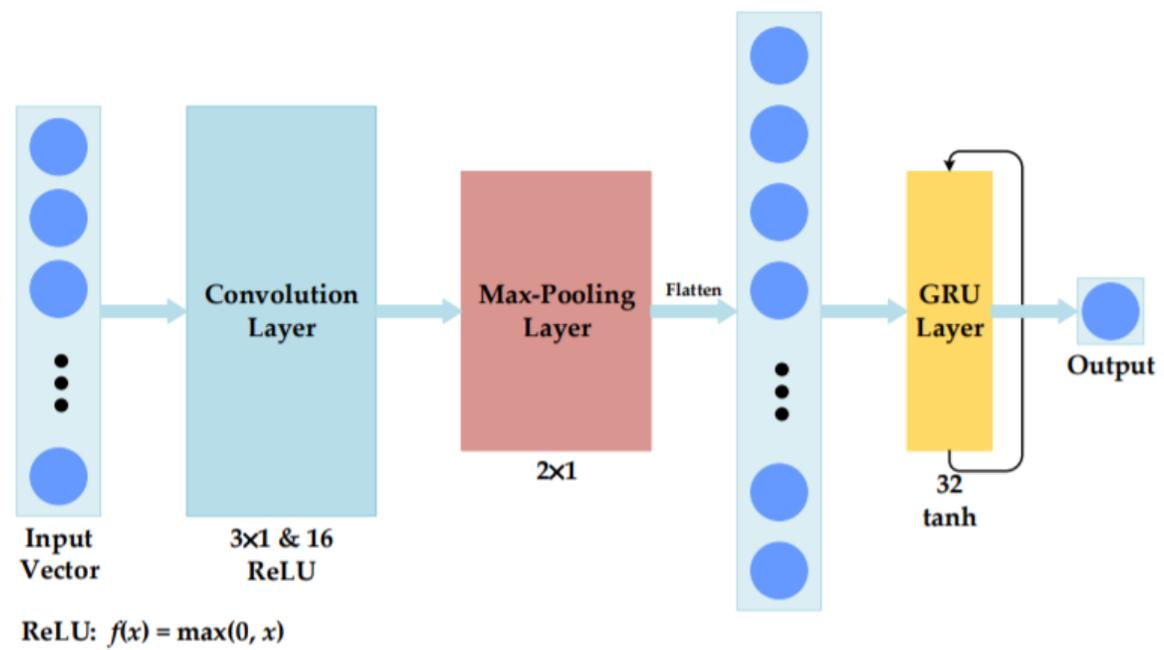
# Método proposto

## Passos

Observe que a série é decomposta em várias funções com diferentes frequências, que são usadas como entrada na rede.



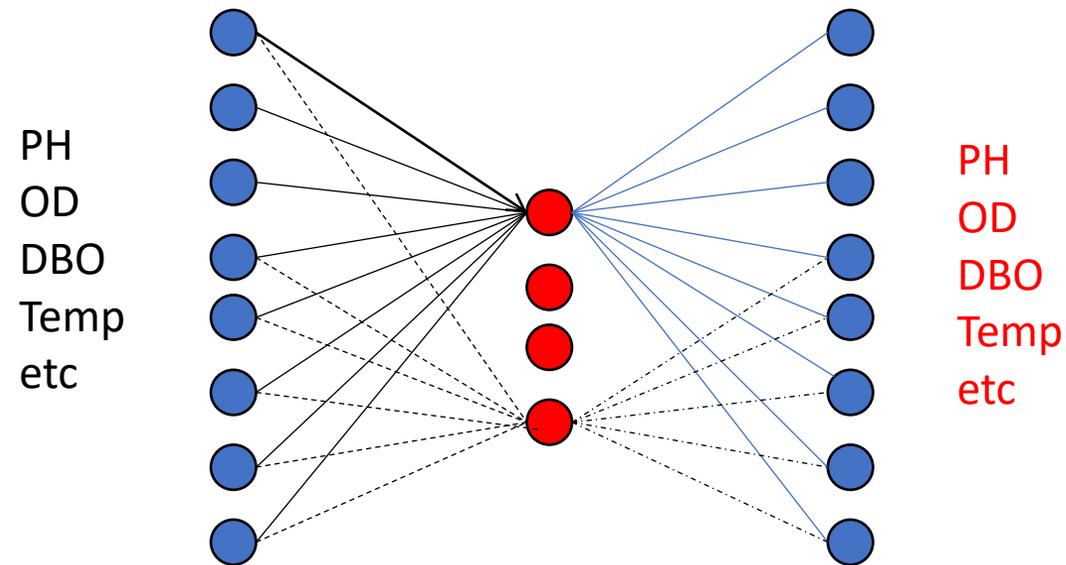
rede





# Autoencoders

Um autoencoder é um algoritmo de aprendizado não supervisionado que aplica back-propagation, definindo os valores de saída como sendo idênticos aos valores de entrada.

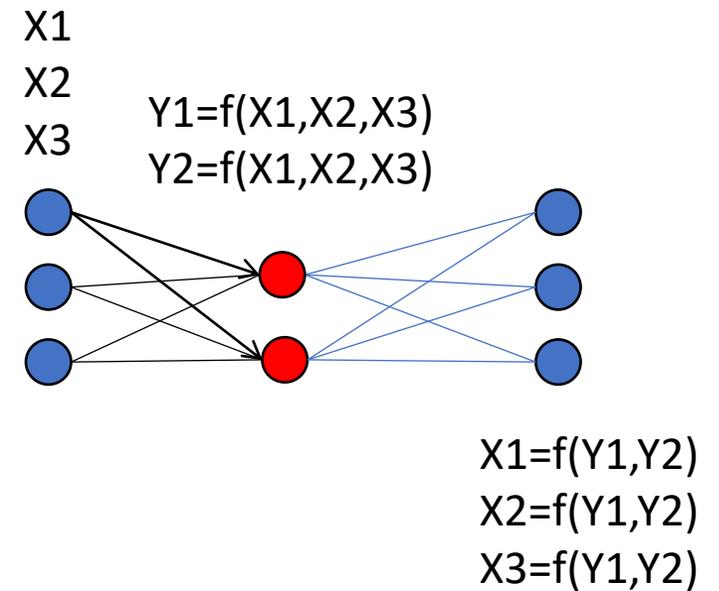


# princípio

O neurônio calcula uma combinação linear dos dados de entrada, na camada escondida.

Se, na camada escondida, é usado um número menor de neurônios, então, todas as variáveis são representadas por uma quantidade menor de variáveis na camada escondida.

Se, é possível recuperar as variáveis originais a partir das variáveis na camada escondida, então, existe redundância nos dados de entrada.





# Autoencoder

- é uma rede neural
- é um algoritmo de aprendizado não supervisionado
- Usa o algoritmo de backpropagation

Autoencoders podem ser usados para encontrar uma representação simplificada da informação de entrada em uma menor dimensão.

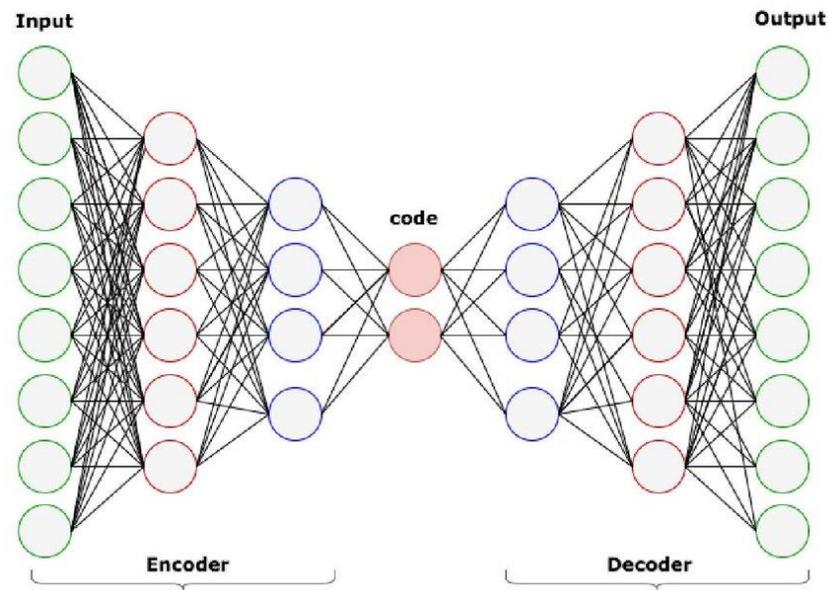
Se é possível restituir os dados originais passando por uma camada estreita (bottle neck) então, existe redundância nos dados de entrada e os dados podem ser representados com menos variáveis menos correlacionadas (similar a transformação das componentes principais).

# Stacked autoencoders

Ainda é possível empilhar várias camadas de *autoencoders* para formar uma rede mais profunda.

Fase Encoder: Uma rede é usada para codificar os dados

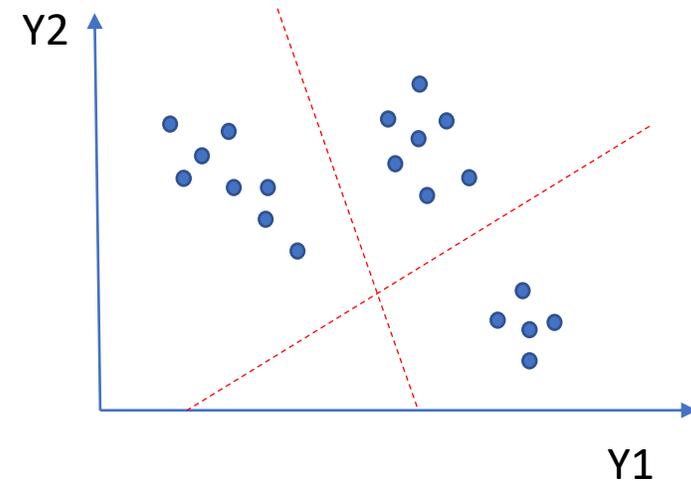
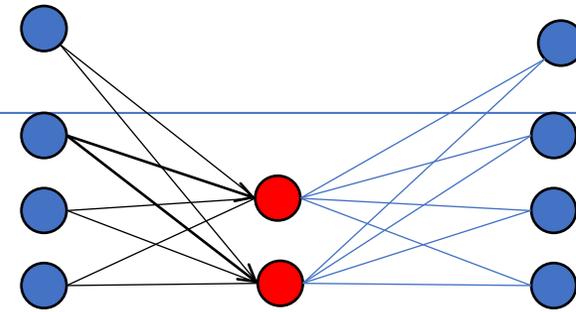
Fase Decoder: Outra para decodificá-los (restituição)



# clustering

Um autoencoder também serve para agrupar amostras com variáveis similares (clustering).

Neste caso, grupos com elementos similares são formados, podendo depois ser interpretados em função das variáveis que caracterizam cada grupo.





# A Semi-supervised Approach for Water Quality Detection Based on IoT Network

Journal of Information Hiding and Multimedia Signal Processing

Ubiquitous International Volume 7, Number 4, July 2016

Ye Yuan & Ke-Bin Jia

Neste exemplo, 14 parâmetros de qualidade de água são usados como entrada: chemical oxygen demand (COD), dissolved oxygen (DO), total phosphorus (TP), 5-day biochemical oxygen demand (BOD5), NH<sub>3</sub>-N, NO<sub>3</sub>-N, oil, chlorophyll, PH, electrical conductivity (EC), turbidity, CL, total coliform (TColi) & temperature (T).

O objetivo é fazer o treinamento não supervisionado para agrupar os dados em “clusters” com propriedades similares. Em uma fase posterior, este resultado é usado em uma nova fase de classificação.