EIVIA 2025:
Deep Learning for Time Series and Applications to Healthcare

Gonzalo Uribarri
KTH Royal Institute of Technology & SciLifeLab

@gonzauri

Course Plan

1/ INTRODUCTION

- Challenges of ML for healthcare
- Introduction to ML for time series

2/ TIME SERIES: Standard Algorithms

- Basic ML for TS Classification
- Deep Learning for TS Classification

3/ TIMES SERIES: State-of -the-art

- Modern architectures for TS Classification
- ROCKET and InceptionTime

4/ APPLICATION

- Case studies in healthcare: Eye-tracking and EEG for diagnosis
- Proper Evaluation

5/ FUTURE

- TS models for large datasets
- Transfer learning: Foundational models for TS?

Classification Strategies

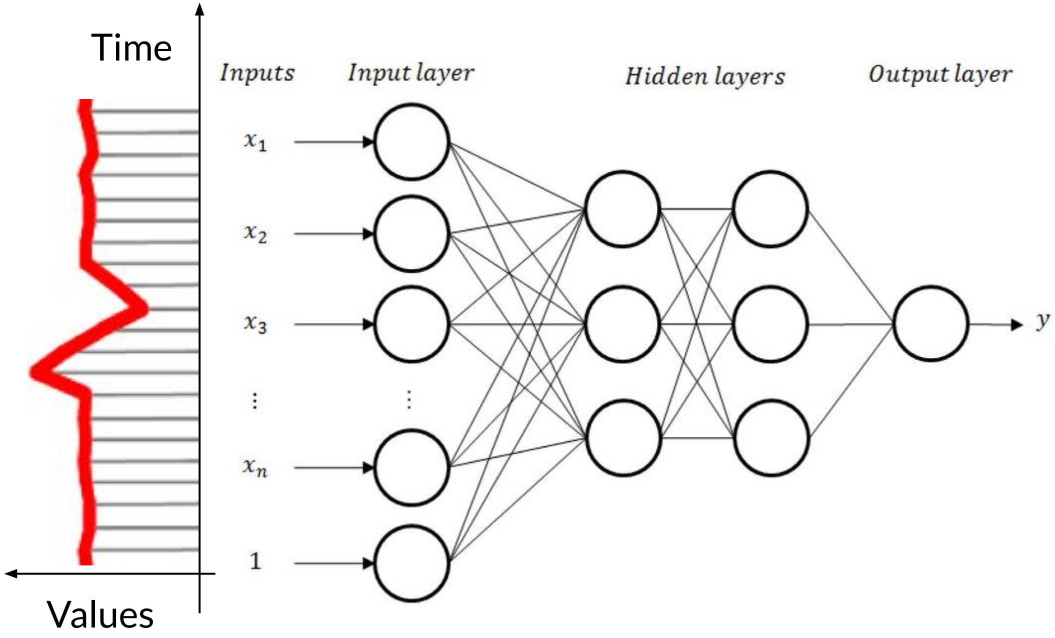
- Time Series Classification
  - Feature Based
    - Predefined Features
    - Spectral Features
    - Intervals / Dictionary
  - Distance Based
    - Elastic distance
    - Representation Learning
  - End-to-end
    - CNNs
    - RNNs
    - Transformers

Focus

Neural Nets:
Naive MLP approach

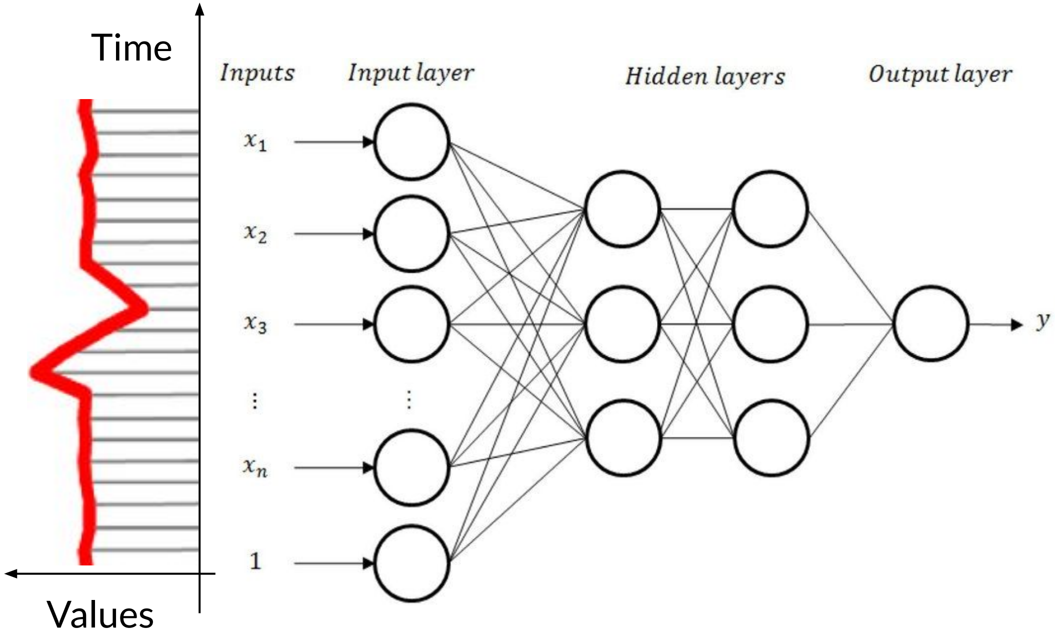
Naive MLP Approach

Why can't we use MLPs to classify?



Naive MLP Approach

Why can't we use MLPs to classify?

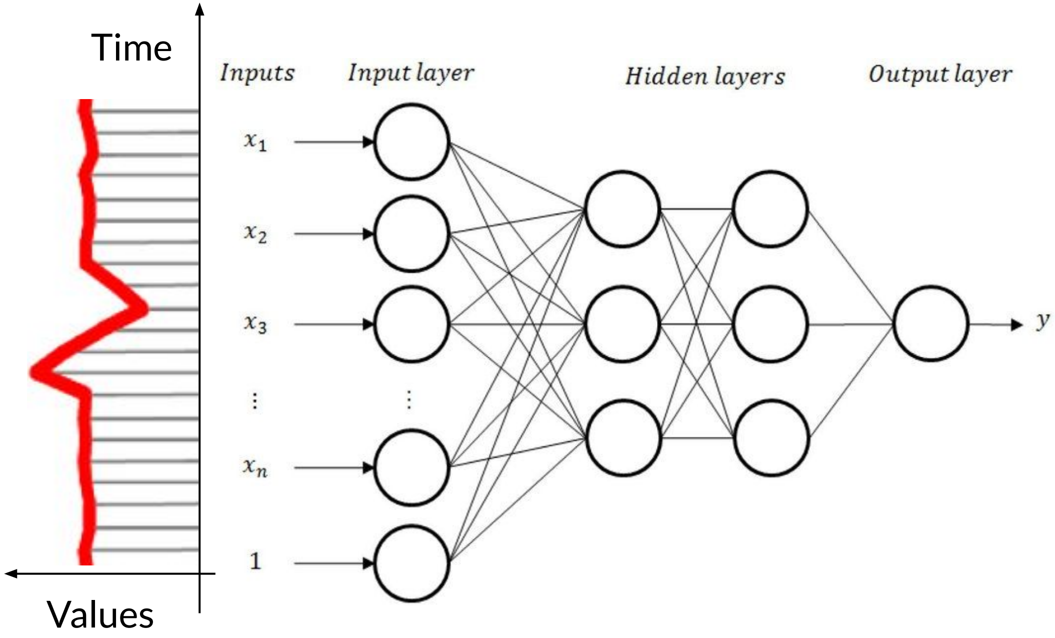


If we use **each timestep as a feature**:

- The number of parameters needed scale poorly with the input length
- Don't exploit the regularities & invariance of the data
- Two "similar" time series present different features.

Naive MLP Approach

Why can't we use MLPs to classify?



If we use **each timestep as a feature**:

- The number of parameters needed scale poorly with the input length
- Don't exploit the regularities & invariance of the data
- Two "similar" time series present different features.

We need proper architectures: Inductive Bias

Neural Nets: Recurrent Neural Networks (RNNs)

Time Series Classification

Feature Based

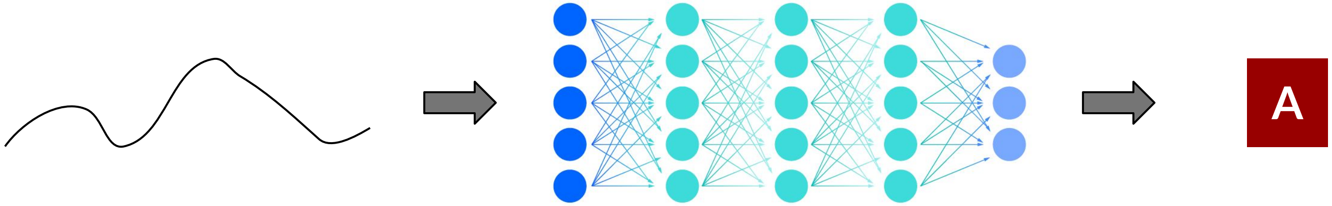
- Predefined Features
- Spectral Features
- Intervals / Dictionary

Distance Based

- Elastic distance
- Representation Learning

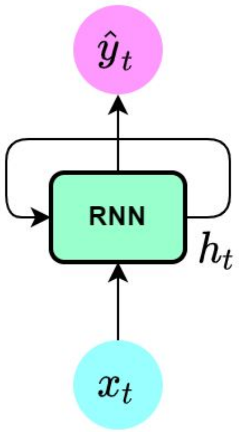
End-to-end

- CNNs
- RNNs
- Transformers



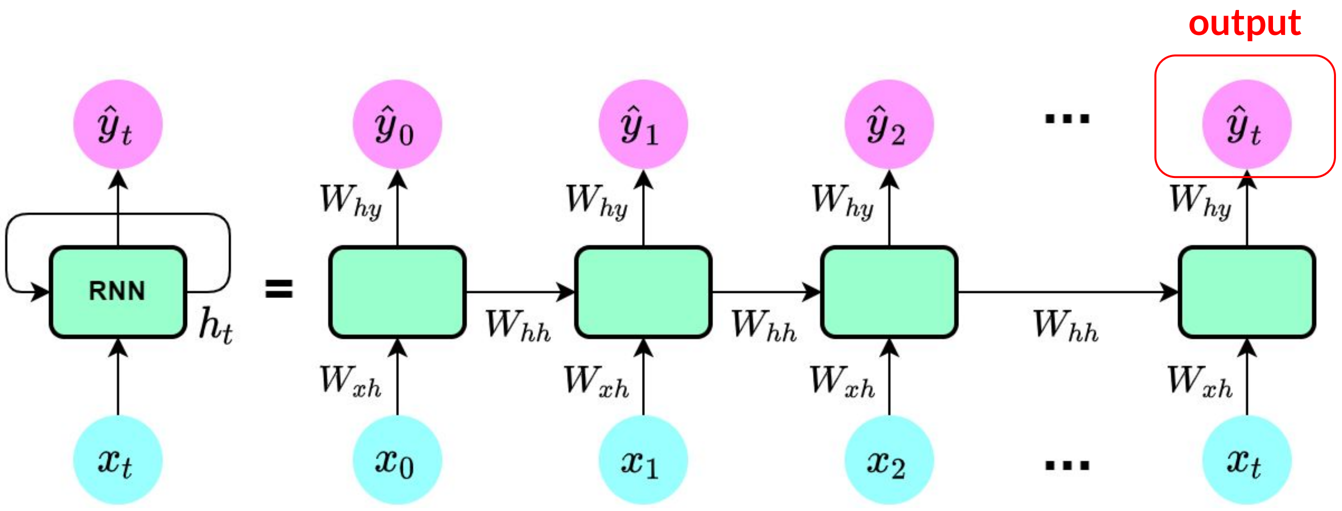
Time Series Data

Another strategy for sequential data is to recursively process each of the steps with the same network.



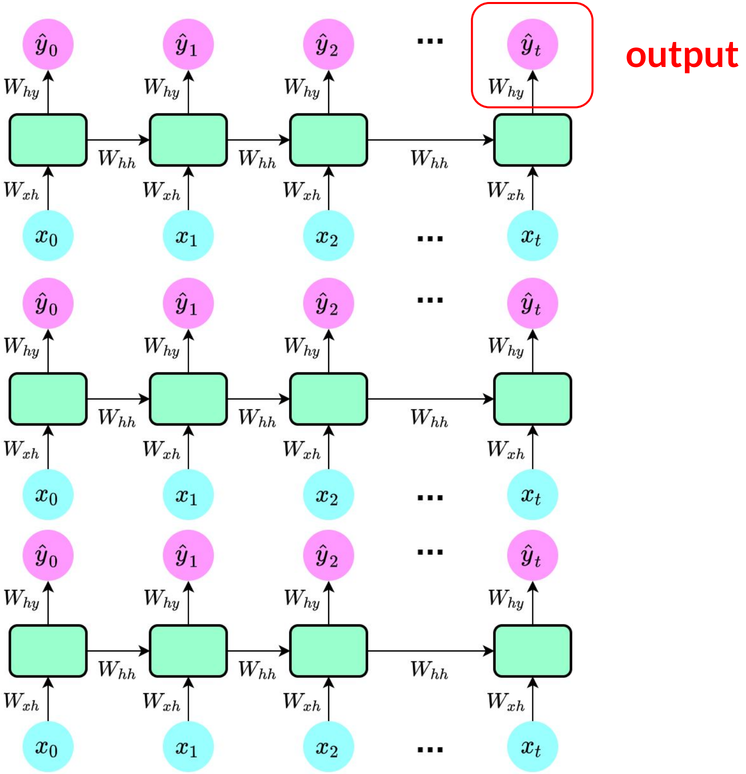
Time Series Data

Another strategy for sequential data is to recursively process each of the steps with the same network.

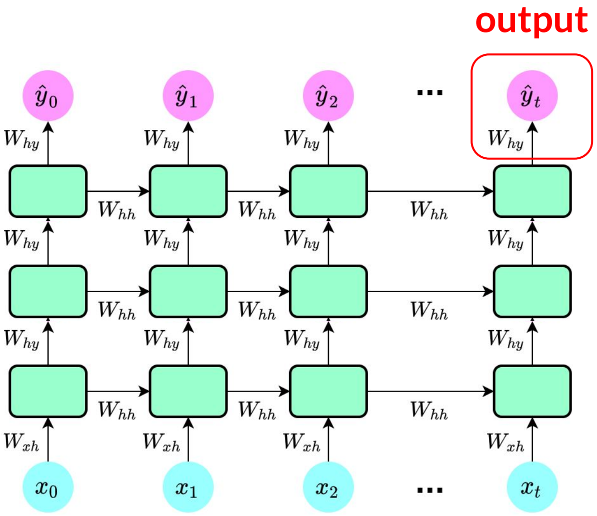


Unfolded RNN: Shared Weights!

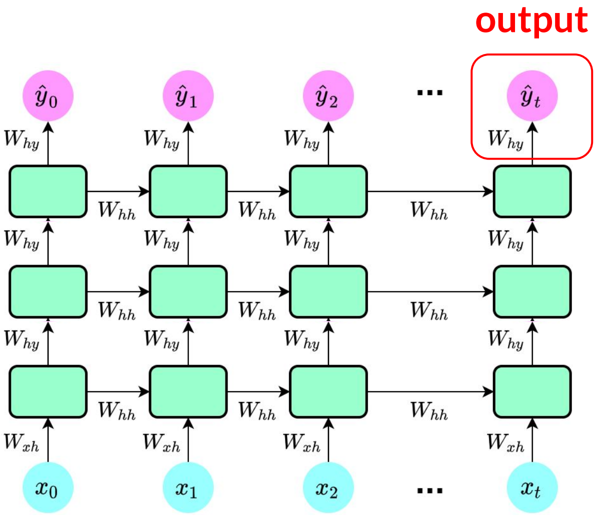
Time Series Data



Time Series Data



Time Series Data

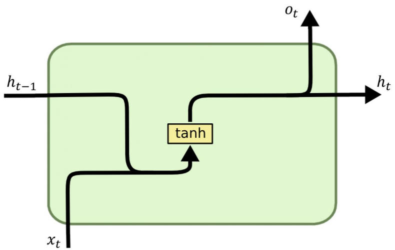


The cells can be stacked to create deeper and more complex models.

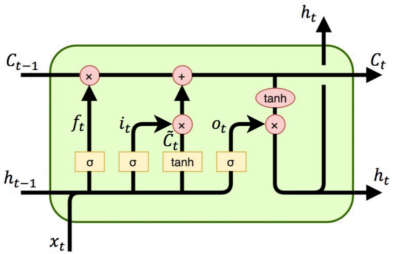
But they are **hard to train** (unstable and not parallelizable).

Time Series Data

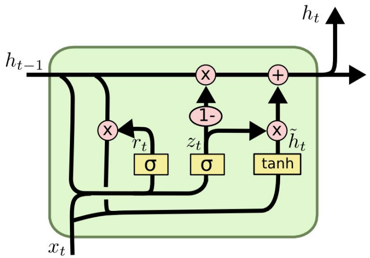
There are different types of RNNs cells:



Classical RNN



LSTM

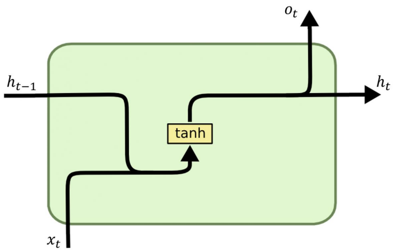


GRU

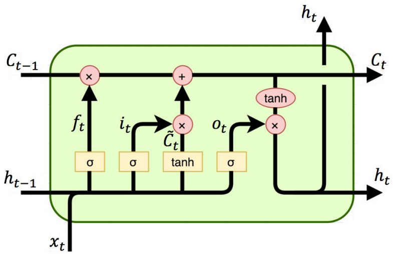
LSTMs 1997 Jürgen Schmidhuber

Time Series Data

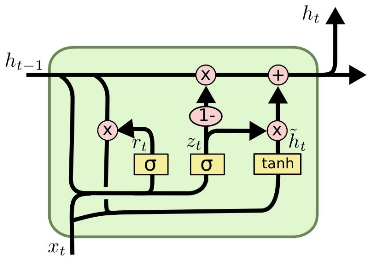
There are different types of RNNs cells:



Classical RNN



LSTM



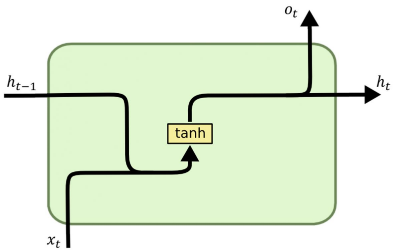
GRU

- Google's use of LSTMs in **Google Voice Search** in 2015 dramatically improved accuracy.
- In 2016, **Google Translate** started using neural networks (stacked LSTMs), having previously used statistical models.

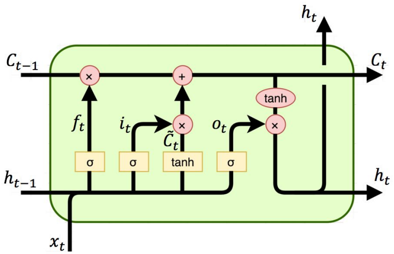
LSTMs 1997 Jürgen Schmidhuber

Time Series Data

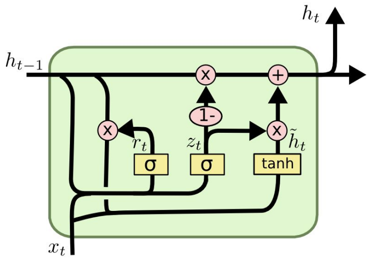
There are different types of RNNs cells:



Classical RNN



LSTM



GRU

Best model for large sequential dataset before Transformers

- Google's use of LSTMs in **Google Voice Search** in 2015 dramatically improved accuracy.
- In 2016, **Google Translate** started using neural networks (stacked LSTMs), having previously used statistical models.

LSTMs 1997 Jürgen Schmidhuber

Neural Nets: Convolutional Neural Networks (CNNs)

Time Series Classification

Feature Based

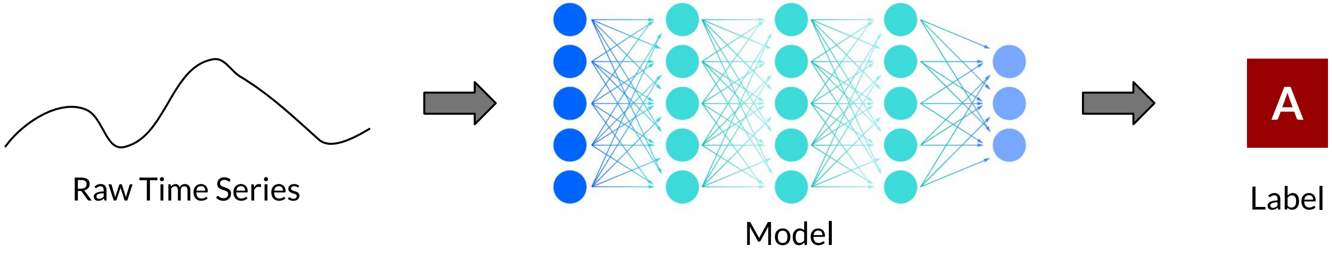
- Predefined Features
- Spectral Features
- Intervals / Dictionary

Distance Based

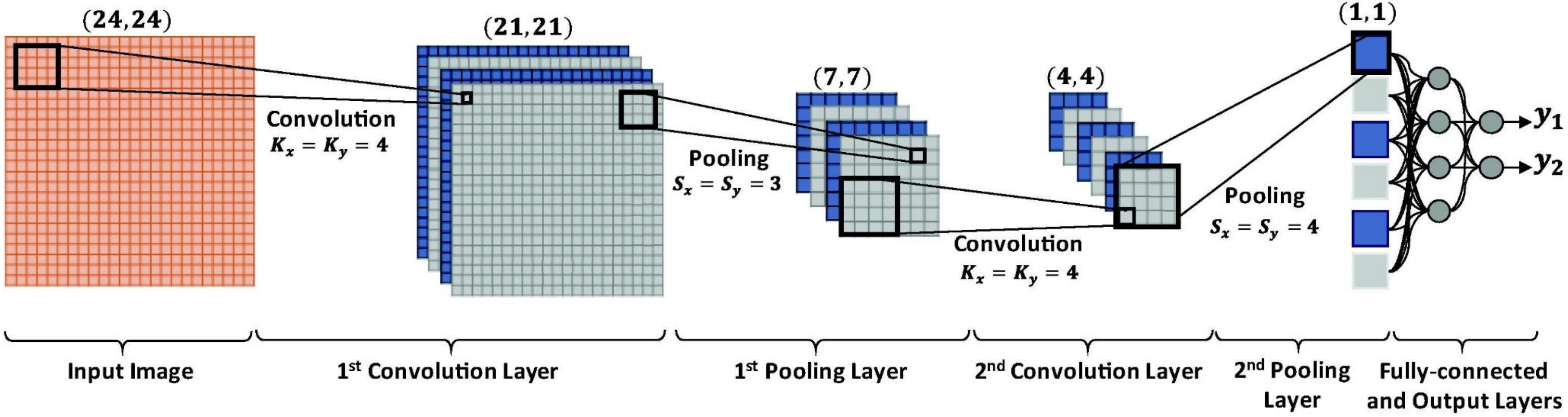
- Elastic distance
- Representation Learning

End-to-end

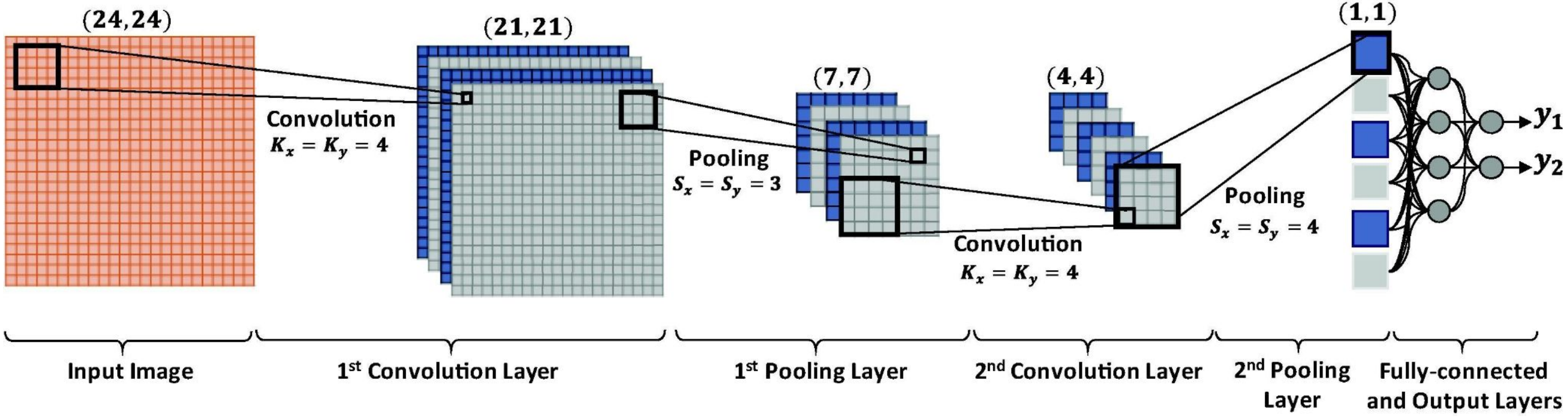
- CNNs
- RNNs
- Transformers



CNNs for Images Classification [Refresh]



CNNs for Images Classification [Refresh]



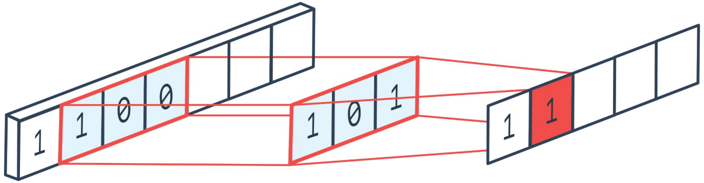
Key Concepts about CNNs

- **Weight Sharing:** Less parameters
- **Translational Invariance:** Exploit data regularities
- **Local Receptive Fields:** Spatial Hierarchies
- **Pooling Operations:** Proper Data Downsampling

CNNs for Time Series Classification

For time series we use 1D-convolutions.

Convolution moves in the time dimension.

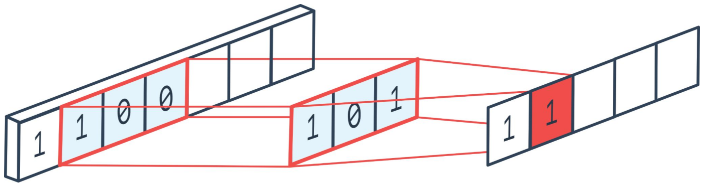


The kernel size is **(c x k)**, where c is the number of channels and k the number of elements in the kernel. In this example is **(1 x 3)**.

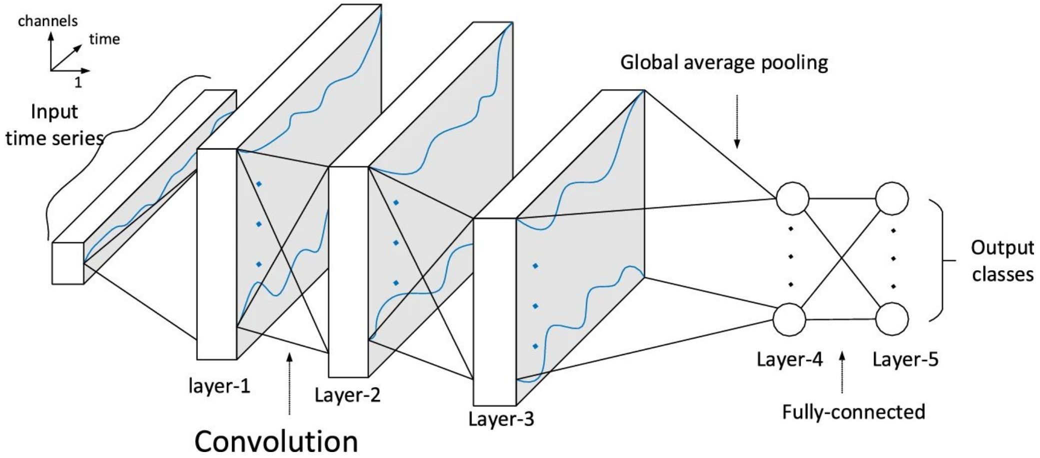
CNNs for Time Series Classification

For time series we use 1D-convolutions.

Convolution moves in the time dimension.



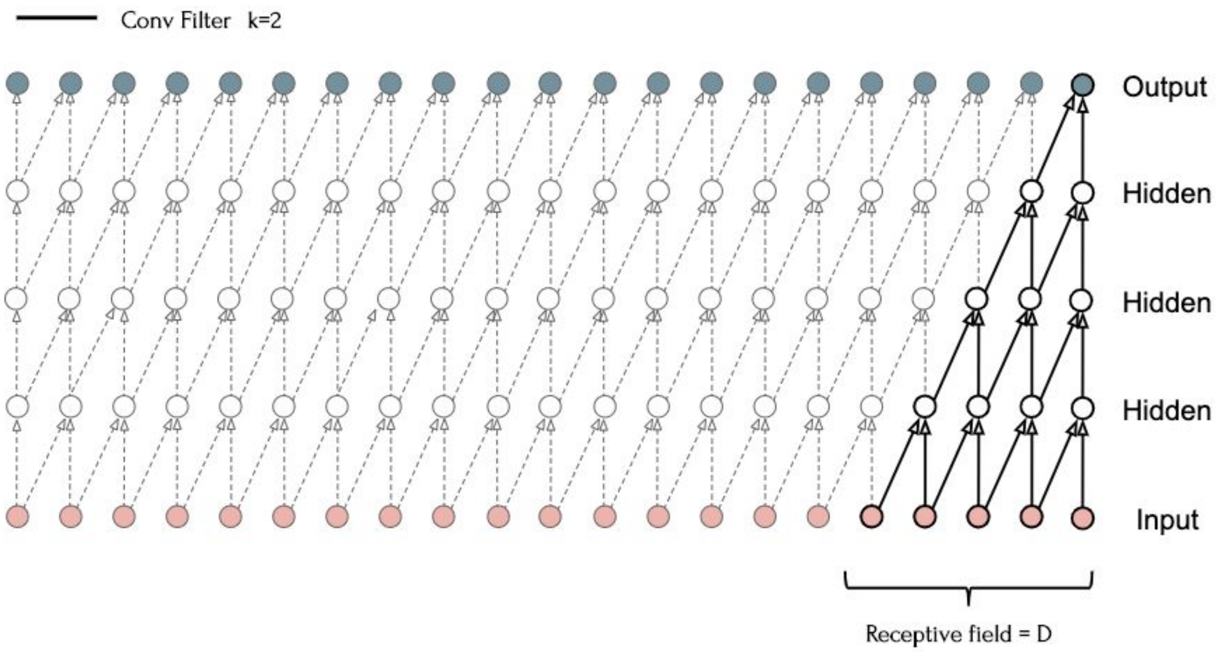
The kernel size is **(c x k)**, where c is the number of channels and k the number of elements in the kernel. In this example is **(1 x 3)**.



A **1-D CNN** with 3 conv layers anw 2 fully connected layers. Notice the time-pooling operation at the last convolutional layer.

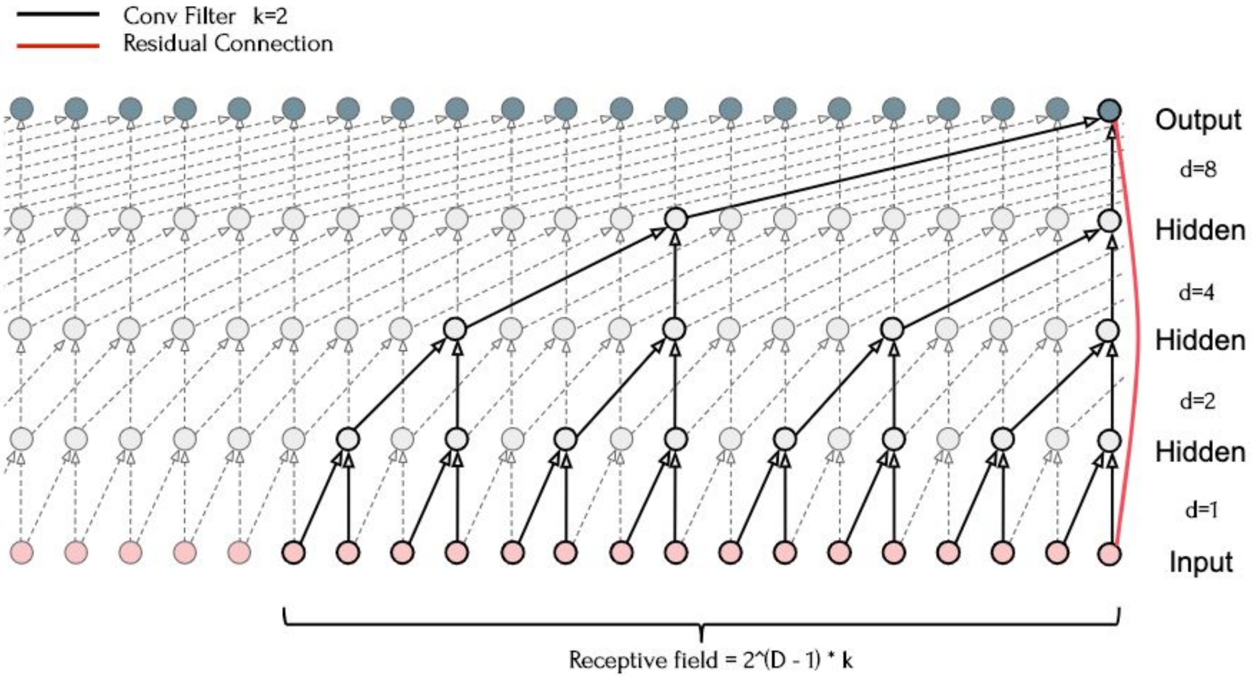
Receptive Field

To have a larger receptive field, we need a deeper network.



Receptive Field

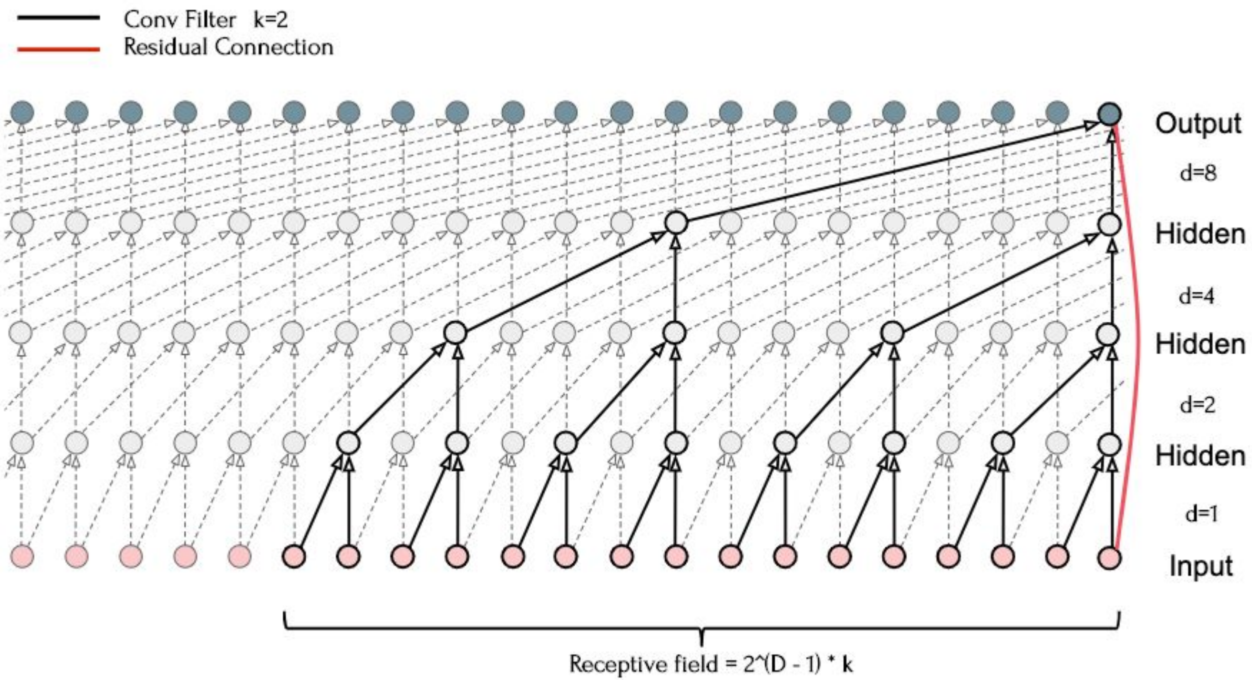
One way to solve this is using Dilated Convolutions.



Receptive Field

One way to solve this is using Dilated Convolutions.

This strategy was proposed by Google Deepmind in 2016. The **WaveNet** architecture was the first capable of effectively dealing with raw audio time series.



State-of-the-art (SOTA)

Models for TSC

Models for time series classification are tested on the **UEA** archive and the **UCR** archive (benchmarks for multivariate and univariate TSC).

Models for TSC

Models for time series classification are tested on the **UEA** archive and the **UCR** archive (benchmarks for multivariate and univariate TSC).

UEA

SPRINGER NATURE Link

Find a journal Publish with us Track your research Search

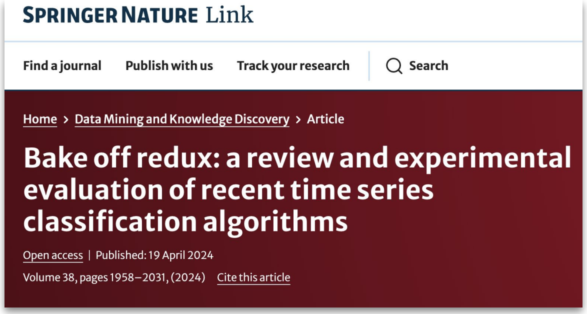
Home › Data Mining and Knowledge Discovery › Article

The great multivariate time series classification bake off: a review and experimental evaluation of recent algorithmic advances

Open access | Published: 18 December 2020

Volume 35, pages 401–449, (2021) Cite this article

UCR



SPRINGER NATURE Link

Find a journal Publish with us Track your research Search

Home › Data Mining and Knowledge Discovery › Article

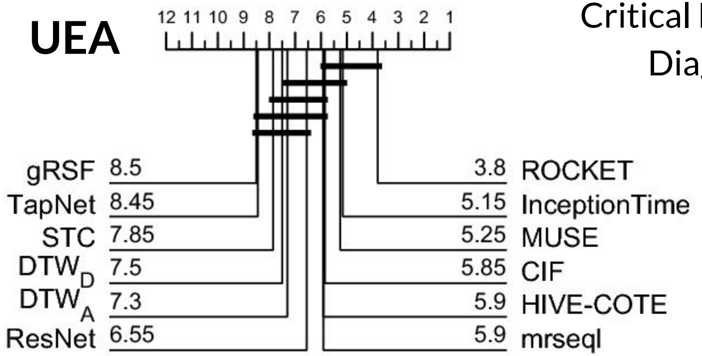
Bake off redux: a review and experimental evaluation of recent time series classification algorithms

Open access | Published: 19 April 2024

Volume 38, pages 1958–2031, (2024) Cite this article

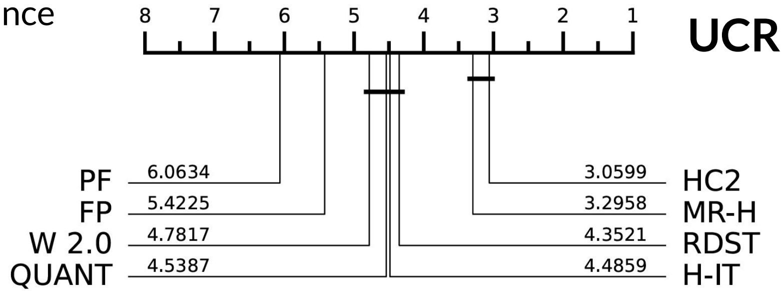
Models for TSC

Models for time series classification are tested on the **UEA** archive and the **UCR** archive (benchmarks for multivariate and univariate TSC).



Alejandro Pasos Ruiz et al. [2021]

Critical Difference Diagrams

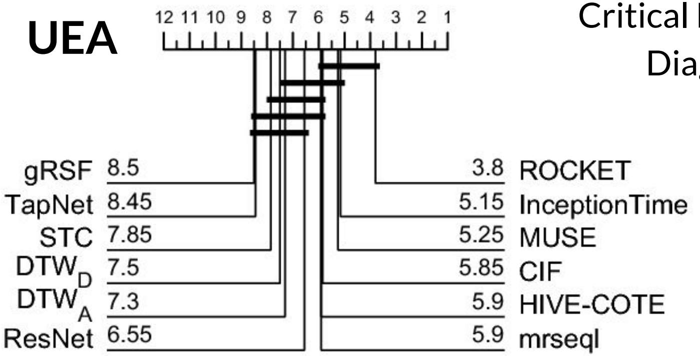


Middlehurst et al. [2024]

Models for TSC

Models for time series classification are tested on the **UEA** archive and the **UCR** archive (benchmarks for multivariate and univariate TSC).

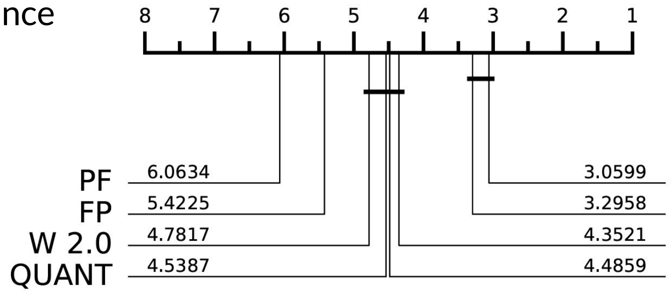
UEA



Alejandro Pasos Ruiz et al. [2021]

Critical Difference Diagrams

UCR



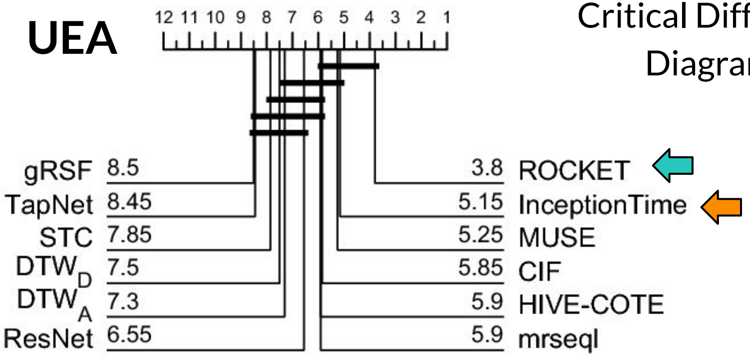
Middlehurst et al. [2024]

The position of each model represents its mean rank (lower is better)

Models for TSC

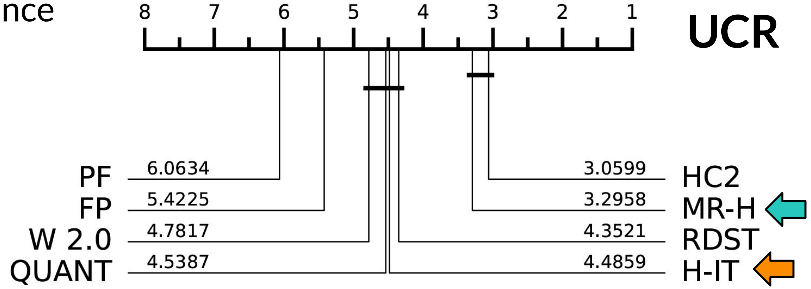
Models for time series classification are tested on the **UEA** archive and the **UCR** archive (benchmarks for multivariate and univariate TSC).

UEA



Alejandro Pasos Ruiz et al. [2021]

Critical Difference Diagrams



UCR

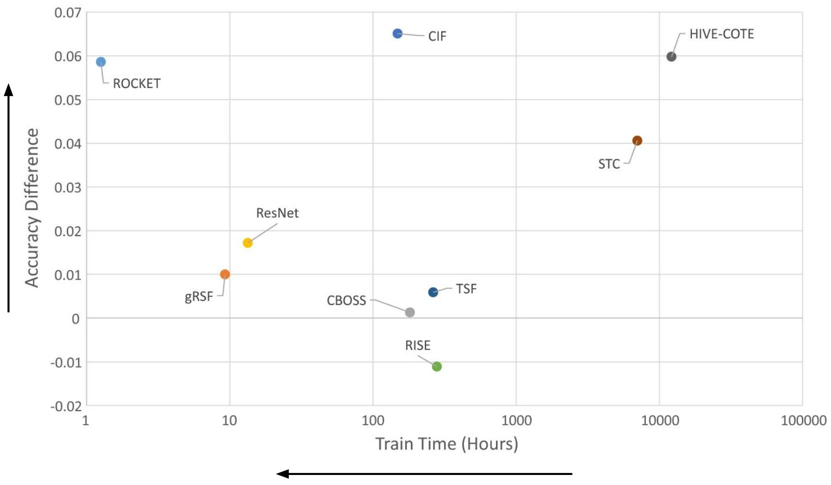
Middlehurst et al. [2024]

InceptionTime and ROCKET are two different models that performs very well.

Models for TSC

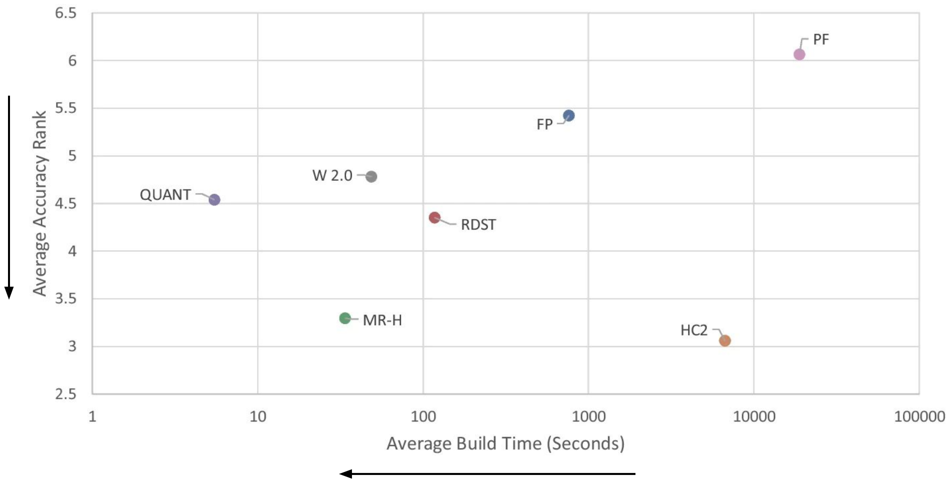
But accuracy is not all, efficiency also matters.

UEA



Alejandro Pasos Ruiz et al. [2021]

UCR

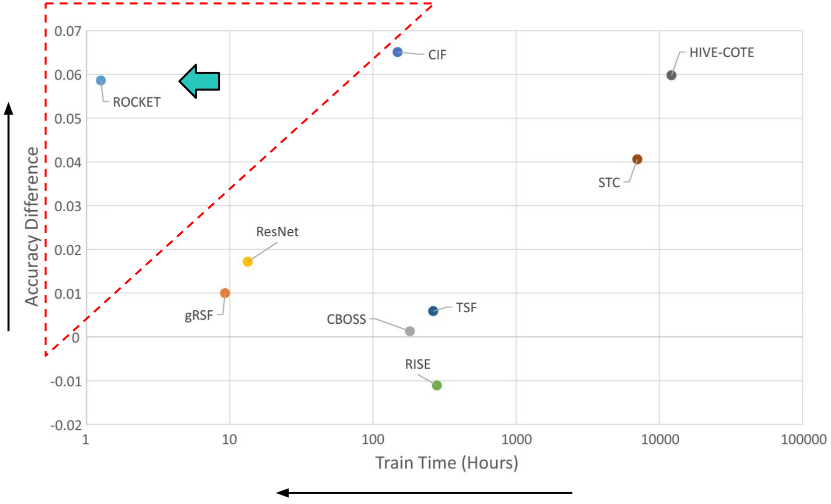


Middlehurst et al. [2024]

Models for TSC

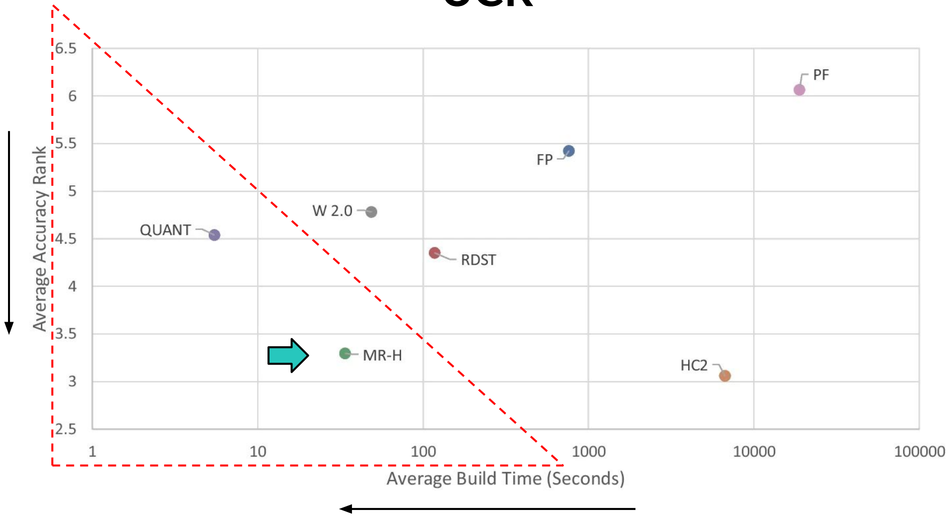
But accuracy is not all, efficiency also matters.

UEA



Alejandro Pasos Ruiz et al. [2021]

UCR



Middlehurst et al. [2024]

InceptionTime

Time Series Classification

Feature Based

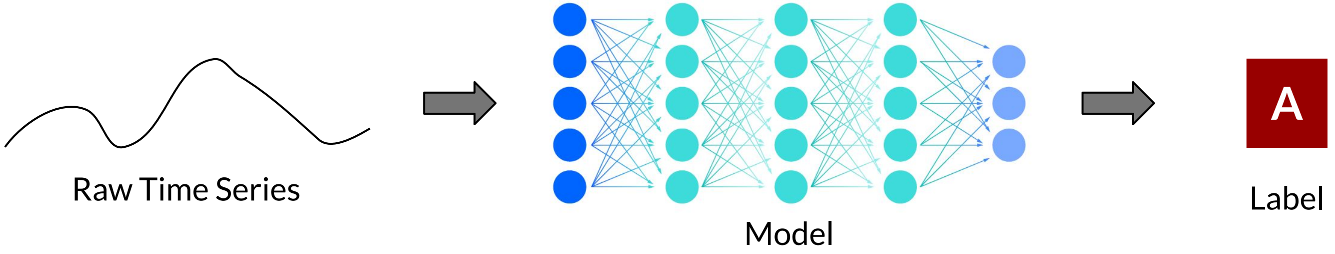
- Predefined Features
- Spectral Features
- Intervals / Dictionary

Distance Based

- Elastic distance
- Representation Learning

End-to-end

- CNNs
- RNNs
- Transformers



InceptionTime

SPRINGER NATURE Link

Find a journal　Publish with us　Track your research　Search

Home > Data Mining and Knowledge Discovery > Article

InceptionTime: Finding AlexNet for time series classification

Published: 07 September 2020

Volume 34, pages 1936–1962, (2020)　Cite this article

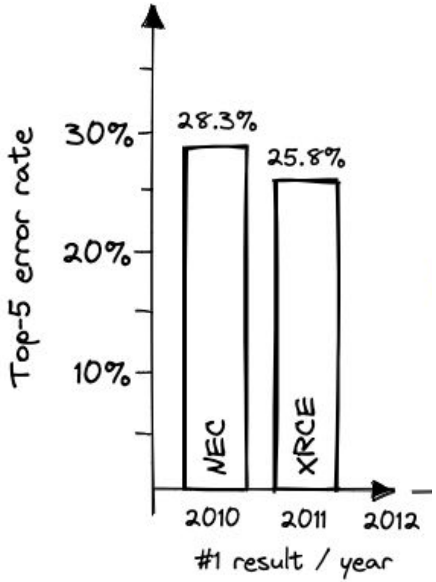
AlexNet - A little bit of history

ImageNet is an **HUGE** image dataset released in 2009, containing 12 million images in 22,000 categories.

(a) ImageNet Synset: One sample image from each category

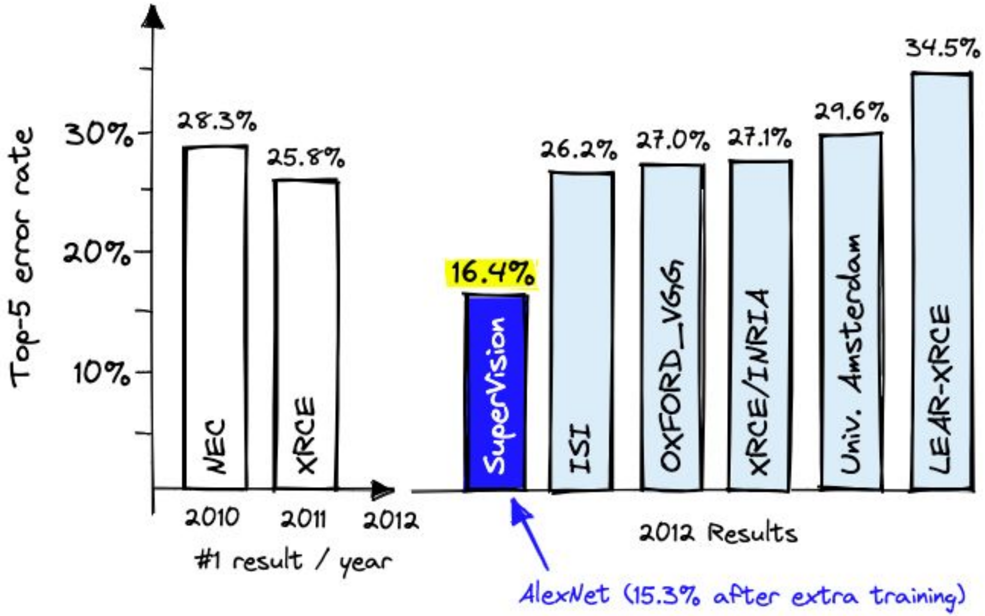
AlexNet - A little bit of history

The **ImageNet Challenge** was a competition to use 1.2 million images to classify 1000 different classes of images.



AlexNet - A little bit of history

The **ImageNet Challenge** was a competition to use 1.2 million images to classify 1000 different classes of images.

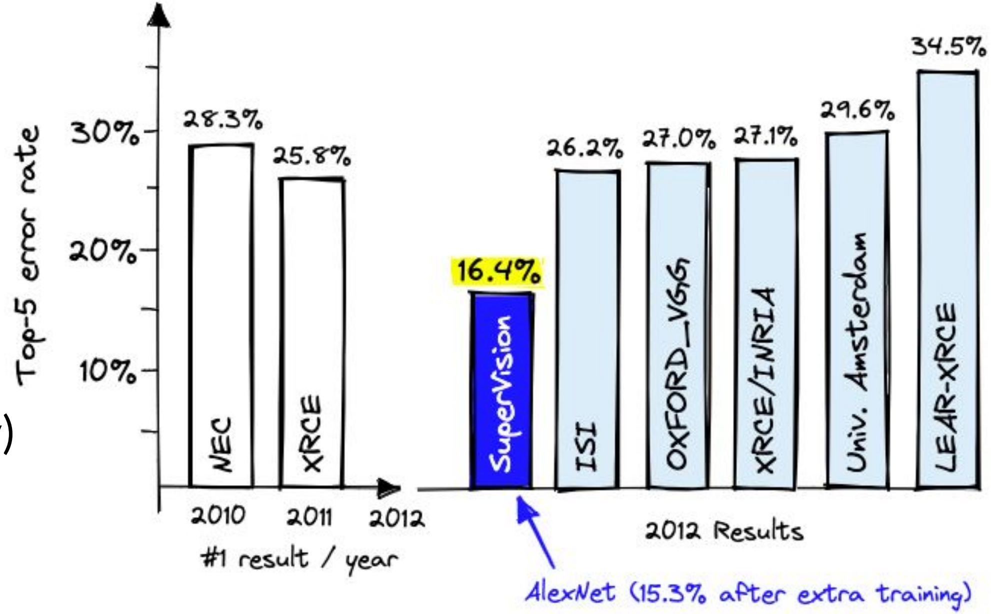


AlexNet - A little bit of history

The **ImageNet Challenge** was a competition to use 1.2 million images to classify 1000 different classes of images.

Key elements:

- Deep (8 layers)
- ReLU
- Dropout
- Augmentation (on-the-fly)
- **GPU !!!**

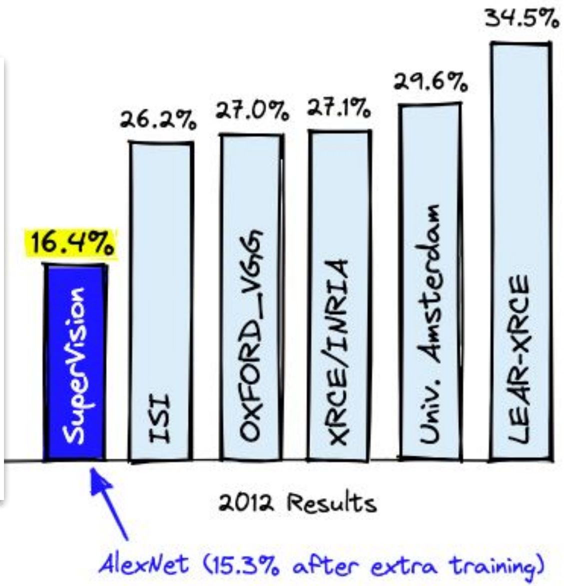


AlexNet - A little bit of history

The **ImageNet Challenge** was a competition to use 1.2 million images to classify 1000 different classes of images.

Imagenet classification with deep convolutional neural networks
A Krizhevsky, I Sutskever… - Advances in neural …, 2012 - proceedings.neurips.cc
… We trained a large, **deep** convolutional neural network to **classify** the 1.2 million high-resolution images in the **ImageNet** LSVRC-2010 contest into the 1000 different classes. On the test …
Save Cite Cited by 138321 Related articles All 98 versions

ImageNet classification with deep convolutional neural networks
A Krizhevsky, I Sutskever, GE Hinton - Communications of the ACM, 2017 - dl.acm.org
… We trained a large, **deep** convolutional neural network to **classify** the 1.2 million high-resolution images in the **ImageNet** LSVRC-2010 contest into the 1000 different classes. On the test …
Save Cite Cited by 36022 Related articles All 7 versions



InceptionTime

SPRINGER NATURE Link

Find a journal | Publish with us | Track your research | Search

Home › Data Mining and Knowledge Discovery › Article

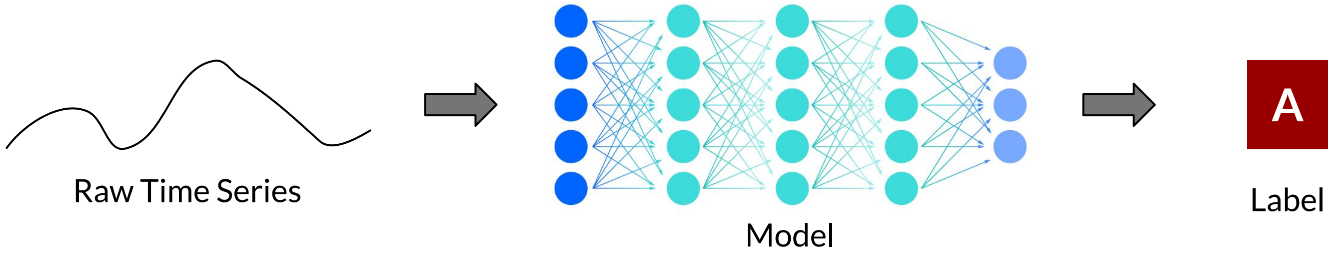
InceptionTime: Finding AlexNet for time series classification

Published: 07 September 2020

Volume 34, pages 1936–1962, (2020) Cite this article

InceptionTime

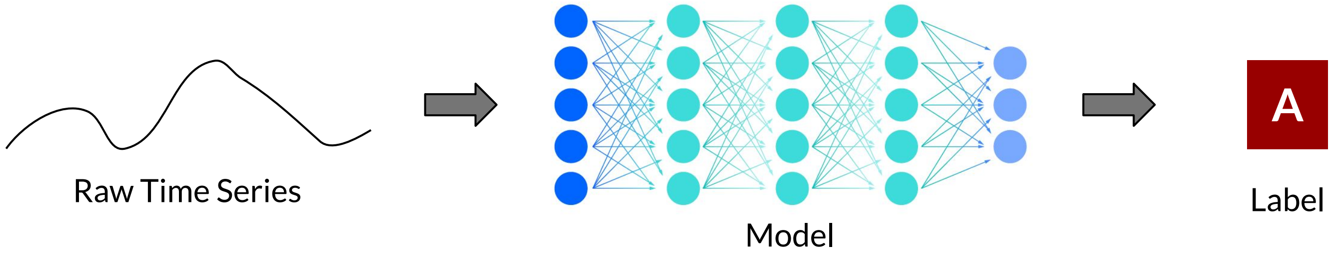
It is a **Deep 1-D Convolutional Neural Network** model inspired by the **Inception-v4** architecture.



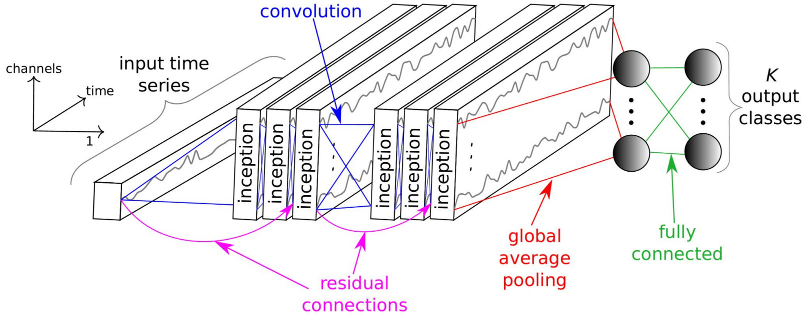
Ismail Fawaz et al. [2020]

InceptionTime

It is a **Deep 1-D Convolutional Neural Network** model inspired by the **Inception-v4** architecture.



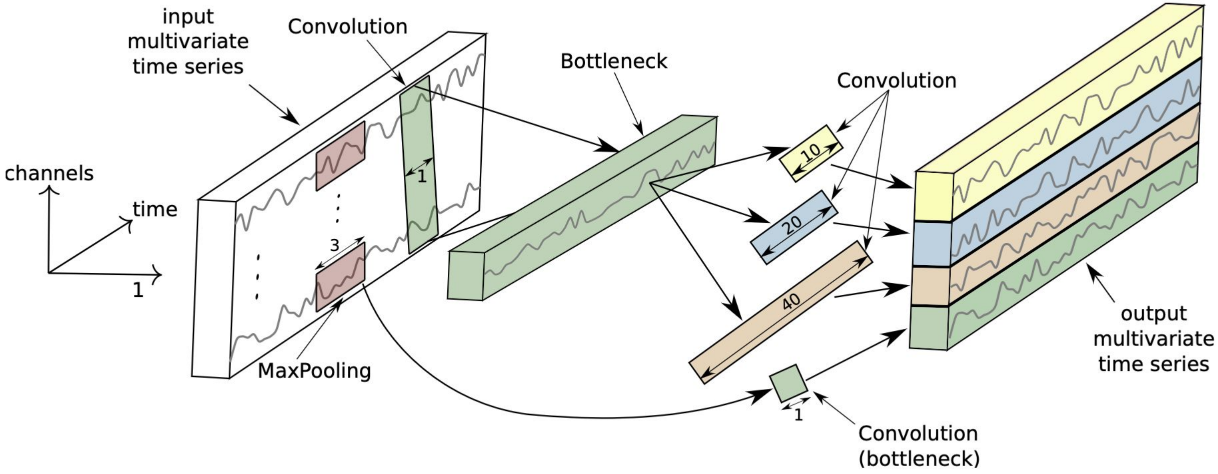
It is composed by **InceptionTime Modules**



Ismail Fawaz et al. [2020]

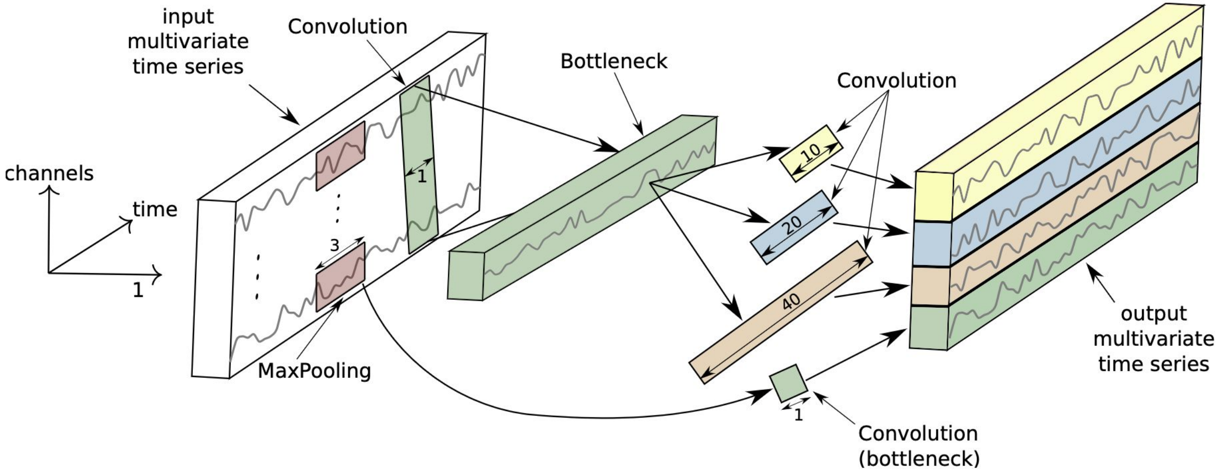
InceptionTime

InceptionTime Module



InceptionTime

InceptionTime Module

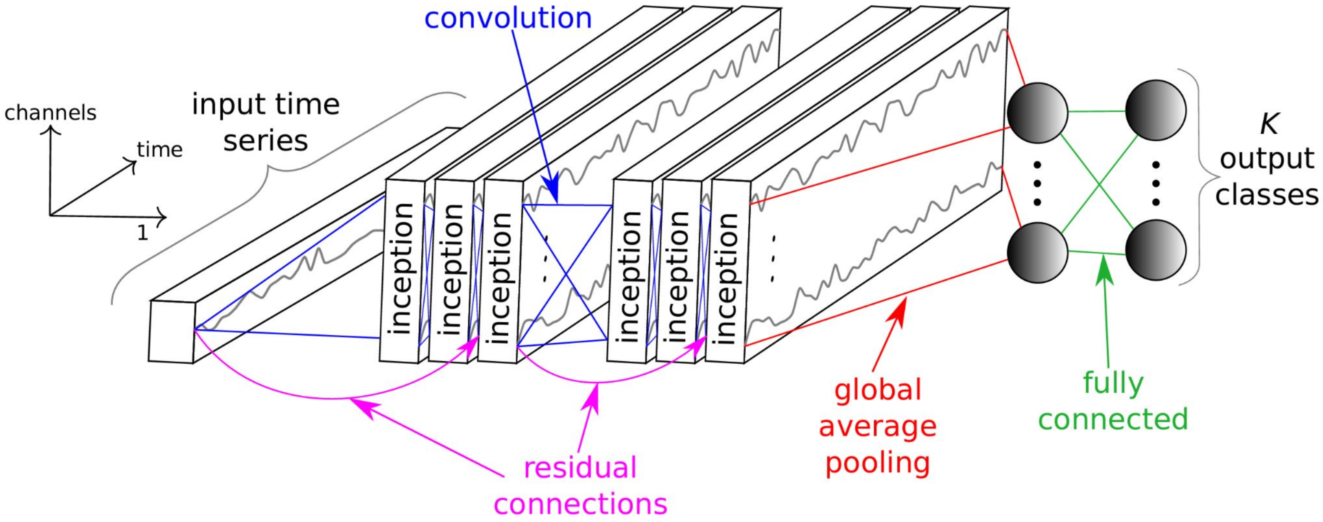


Key elements:

- **Multi-Scale Feature Extraction:** Filter of different sizes.
- **Large Kernels:** Larger than ones used for images
- **Bottlenecks:** 1x1 Convolutions (mixing channels)

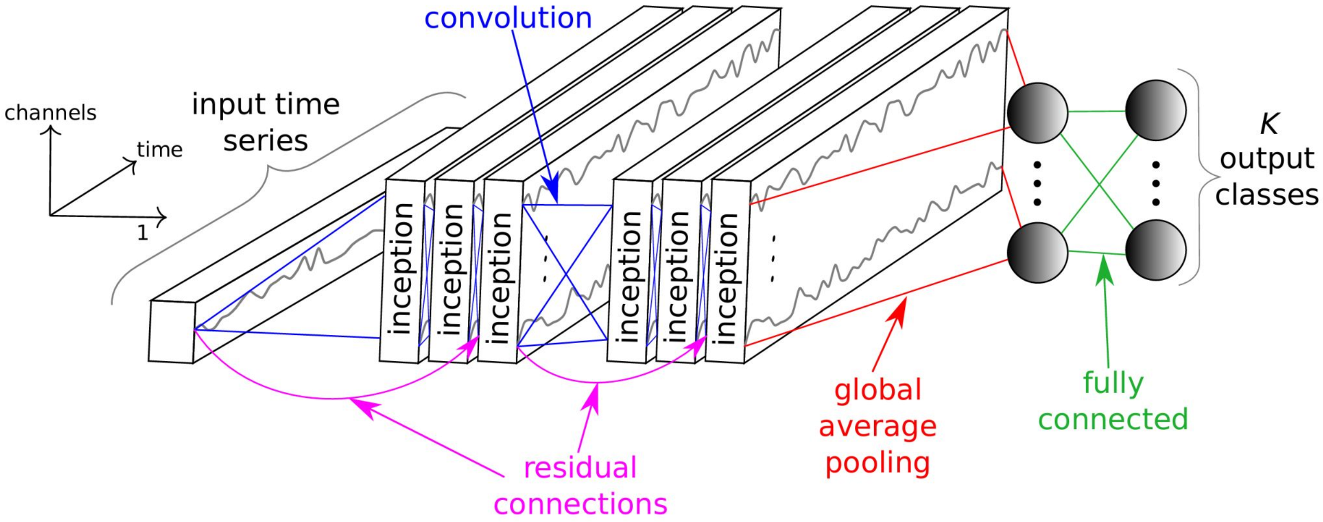
InceptionTime

InceptionTime Architecture



InceptionTime

InceptionTime Architecture



Key elements:

- **Residual Connections:** Improve training stability.
- **Global Average Pooling:** Time pooling operation before the fully connected layer.

InceptionTime

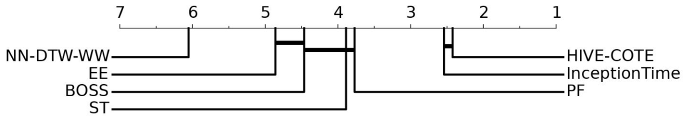


Fig. 5: Critical difference diagram showing the performance of InceptionTime compared to the current state-of-the-art classifiers of time series data.

Performance on UCR

InceptionTime

InceptionTime: Finding AlexNet for Time Series Classification 9

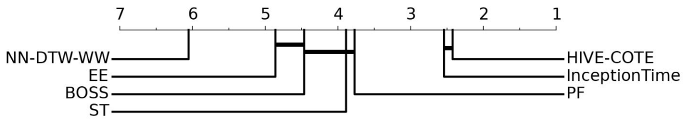


Fig. 5: Critical difference diagram showing the performance of InceptionTime compared to the current state-of-the-art classifiers of time series data.

Performance on UCR

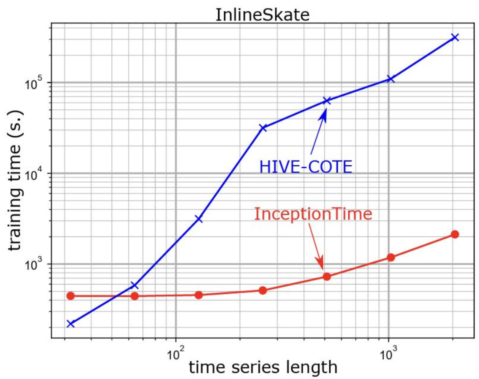


Fig. 7: Training time as a function of the series length for the InlineSkate dataset.

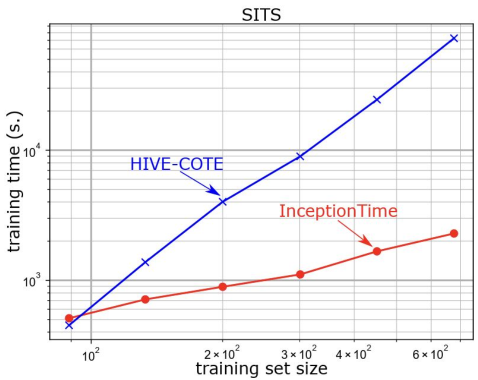


Fig. 8: Training time as a function of the training set size for the SITS dataset.

InceptionTime Strengths and Weaknesses

- State-of-the-art performance for TSC
- Fast inference
- Good Scaling
- **Truly Multivariate**

InceptionTime Strengths and Weaknesses

- State-of-the-art performance for TSC
- Fast inference
- Good Scaling
- **Truly Multivariate**

- Complex Training
- High Variance (Counter using an ensemble)
- Difficult to interpret

ROCKET

Time Series Classification

Feature Based

- Predefined Features
- Spectral Features
- Intervals / Dictionary

Distance Based

- Elastic distance
- Representation Learning

End-to-end

- CNNs
- RNNs
- Transformers

Time Series Classification

Feature Based

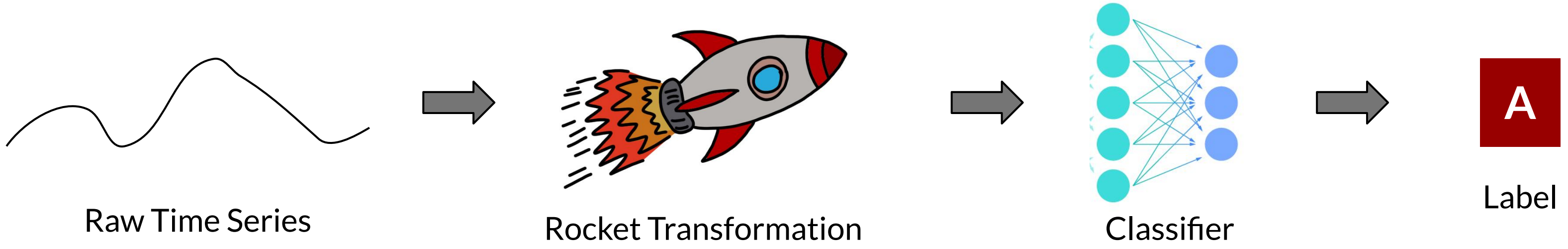
- Predefined Features
- Spectral Features
- Intervals / Dictionary

Distance Based

- Elastic distance
- Representation Learning

End-to-end

- CNNs
- RNNs
- Transformers



SPRINGER NATURE Link

Find a journal Publish with us Track your research Search

Home > Data Mining and Knowledge Discovery > Article

ROCKET: exceptionally fast and accurate time series classification using random convolutional kernels

Published: 13 July 2020

Volume 34, pages 1454–1495, (2020) Cite this article

Download PDF

Access provided by KTH Royal Institute of Technology

Angus Dempster, François Petitjean & Geoffrey I. Webb

15k Accesses 28 Altmetric 2 Mentions Explore all metrics →

SPRINGER NATURE Link

Find a journal Publish with us Track your research Search

Home > Data Mining and Knowledge Discovery > Article

ROCKET: exceptionally fast and accurate time series classification using random convolutional kernels

Published: 13 July 2020

Volume 34, pages 1454–1495, (2020) Cite this article

Download PDF

Access provided by KTH Royal Institute of Technology

Angus Dempster, François Petitjean & Geoffrey I. Webb

15k Accesses 28 Altmetric 2 Mentions Explore all metrics →

RESEARCH-ARTICLE

MiniRocket: A Very Fast (Almost) Deterministic Transform for Time Series Classification

Authors: Angus Dempster, Daniel F. Schmidt, Geoffrey I. Webb Authors Info & Claims

KDD '21: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining • Pages 248 - 257
https://doi.org/10.1145/3447548.3467231

Published: 14 August 2021 Publication History

Check for updates

218 2,560

Get Access

SPRINGER NATURE Link

Find a journal Publish with us Track your research Search

Home > Data Mining and Knowledge Discovery > Article

MultiRocket: multiple pooling operators and transformations for fast and effective time series classification

Open access | Published: 29 June 2022

Volume 36, pages 1623–1646, (2022) Cite this article

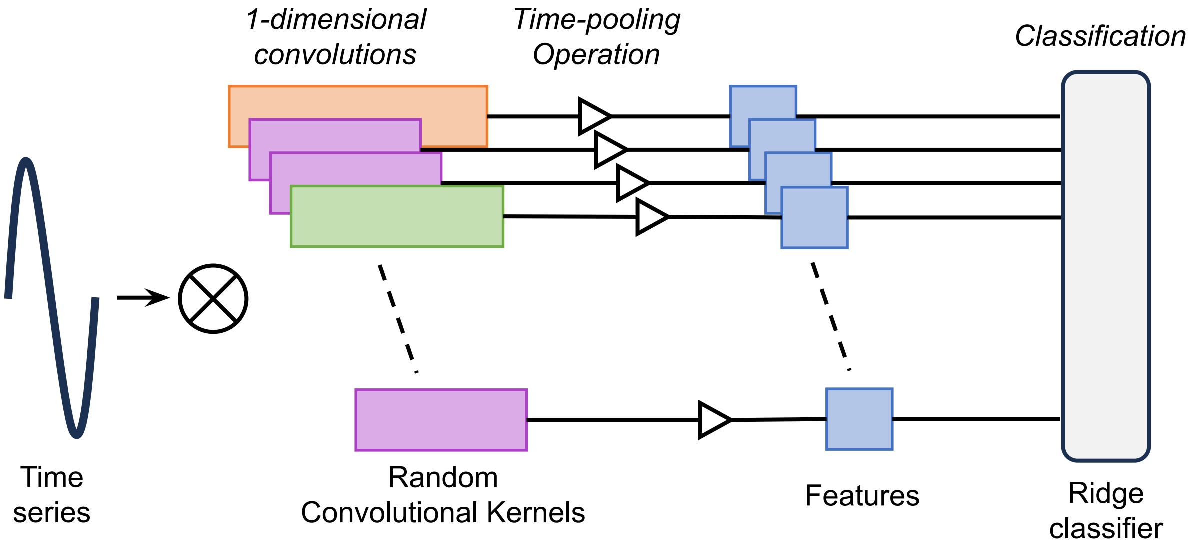
ROCKET models

Random Convolutional Kernel Transform (ROCKET)* is a **transformation stage** which can be applied to **time-series data**.

* Dempster et al. [2020]

ROCKET models

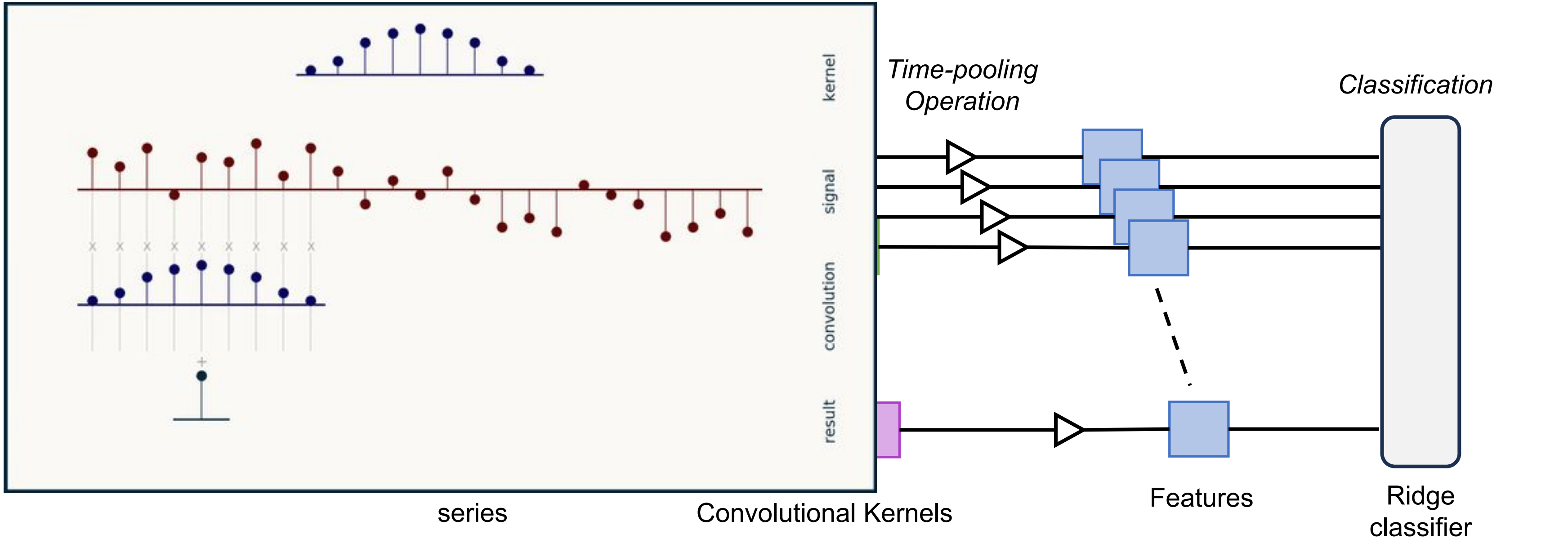
Random Convolutional Kernel Transform (ROCKET)* is a **transformation stage** which can be applied to **time-series data**.



* Dempster et al. [2020]

ROCKET models

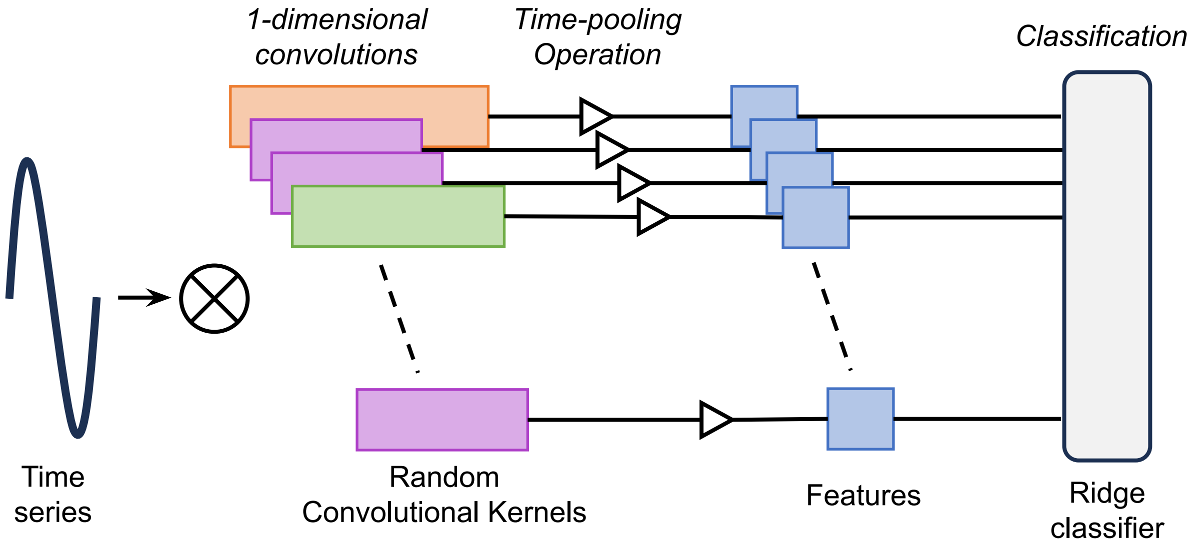
Random Convolutional Kernel Transform (ROCKET)* is a **transformation stage** which can be applied to **time-series data**.



* Dempster et al. [2020]

ROCKET models

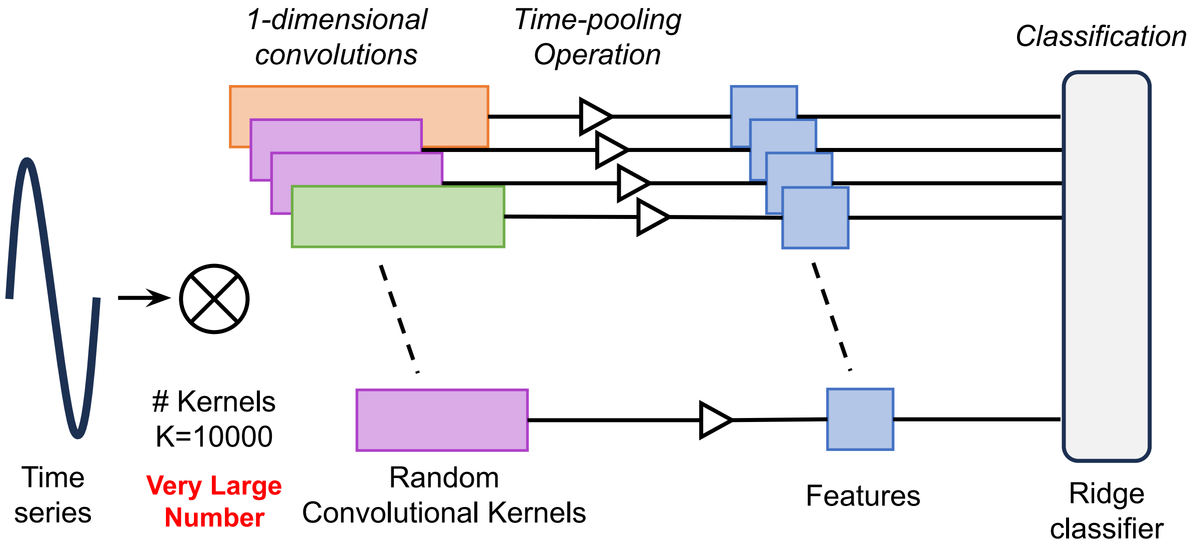
Random Convolutional Kernel Transform (ROCKET)* is a **transformation stage** which can be applied to **time-series data.**



* Dempster et al. [2020]

ROCKET models

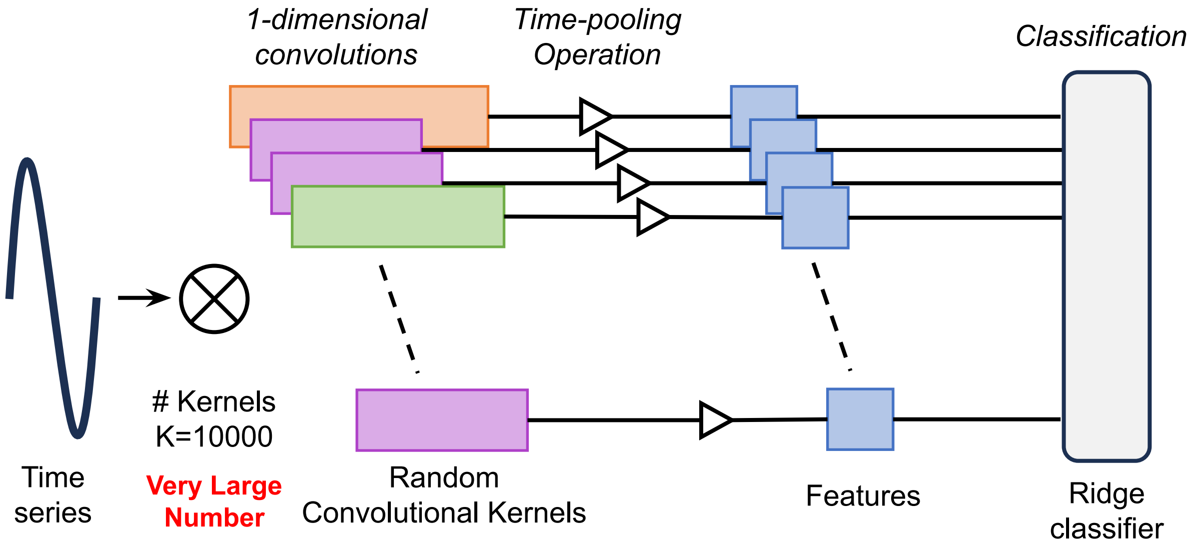
Random Convolutional Kernel Transform (ROCKET)* is a **transformation stage** which can be applied to **time-series data**.



* Dempster et al. [2020]

ROCKET models

Random Convolutional Kernel Transform (ROCKET)* is a **transformation stage** which can be applied to **time-series data.**

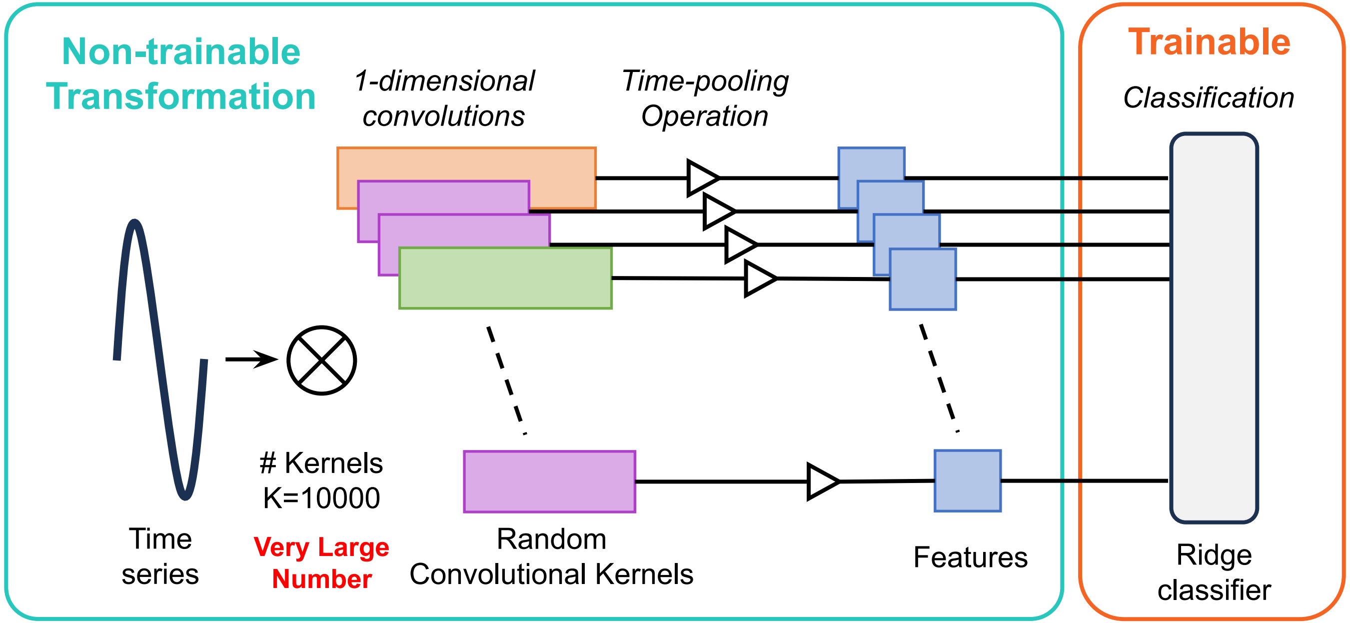


* Dempster et al. [2020]

ROCKET models

Random Convolutional Kernel Transform (ROCKET)* is a **transformation stage** which can be applied to **time-series data.**

This is KEY!



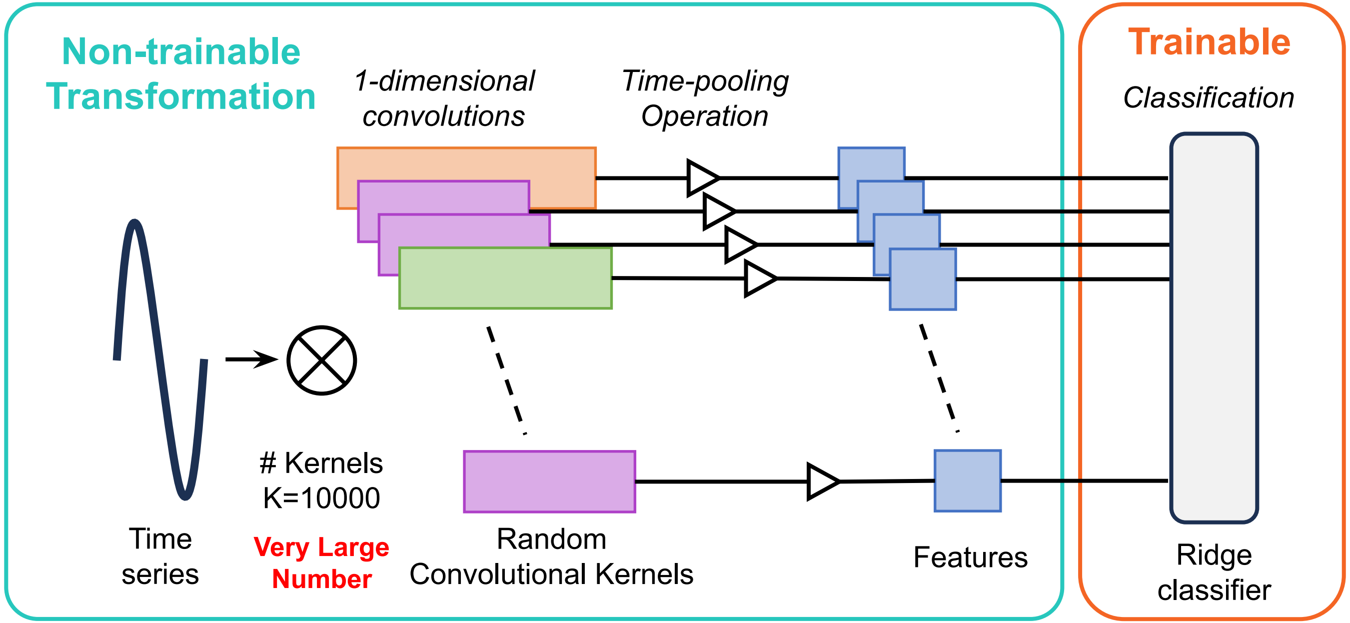
* Dempster et al. [2020]

ROCKET models

Random Convolutional Kernel Transform (ROCKET)* is a **transformation stage** which can be applied to **time-series data.**

This is KEY!

ROCKET
Kernels: Random
Pooling: MAX + PPV
 # Features: 20000



* Dempster et al. [2020]

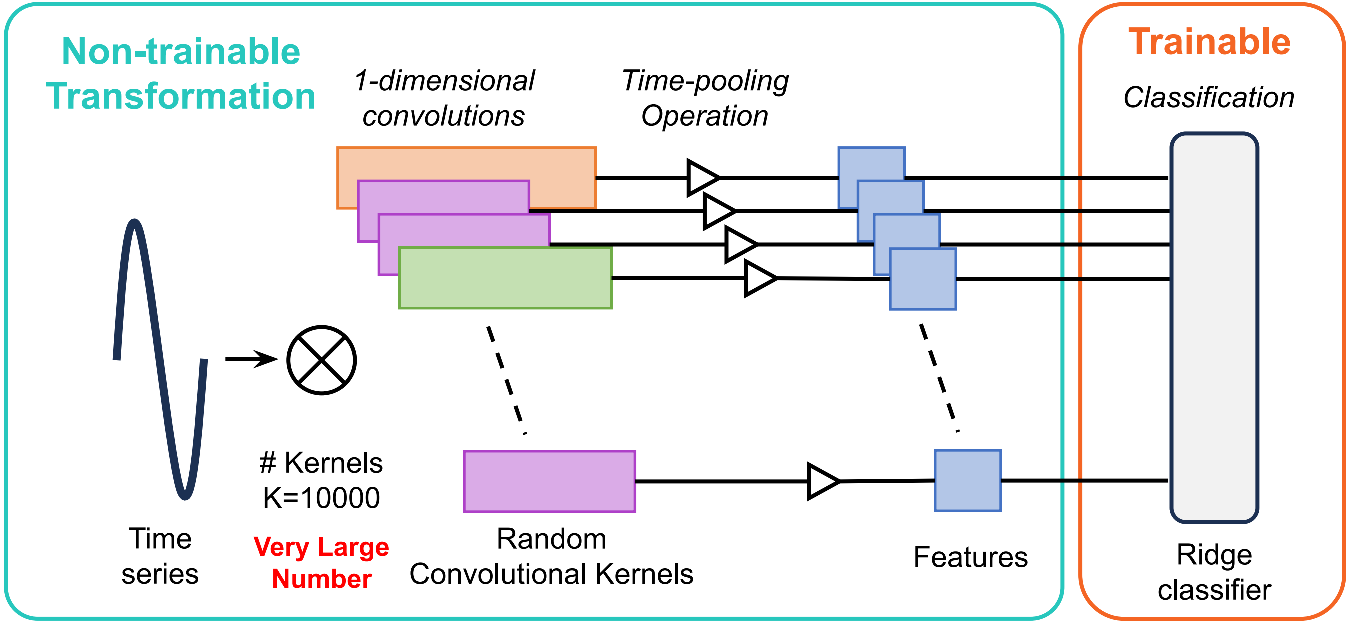
ROCKET models

Random Convolutional Kernel Transform (ROCKET)* is a **transformation stage** which can be applied to **time-series data.**

This is KEY!

ROCKET
Kernels: Random
Pooling: MAX + PPV
 # Features: 20000

MiniRocket
Kernels: Dictionary
Pooling: PPV
 # Features: 10000



ROCKET models

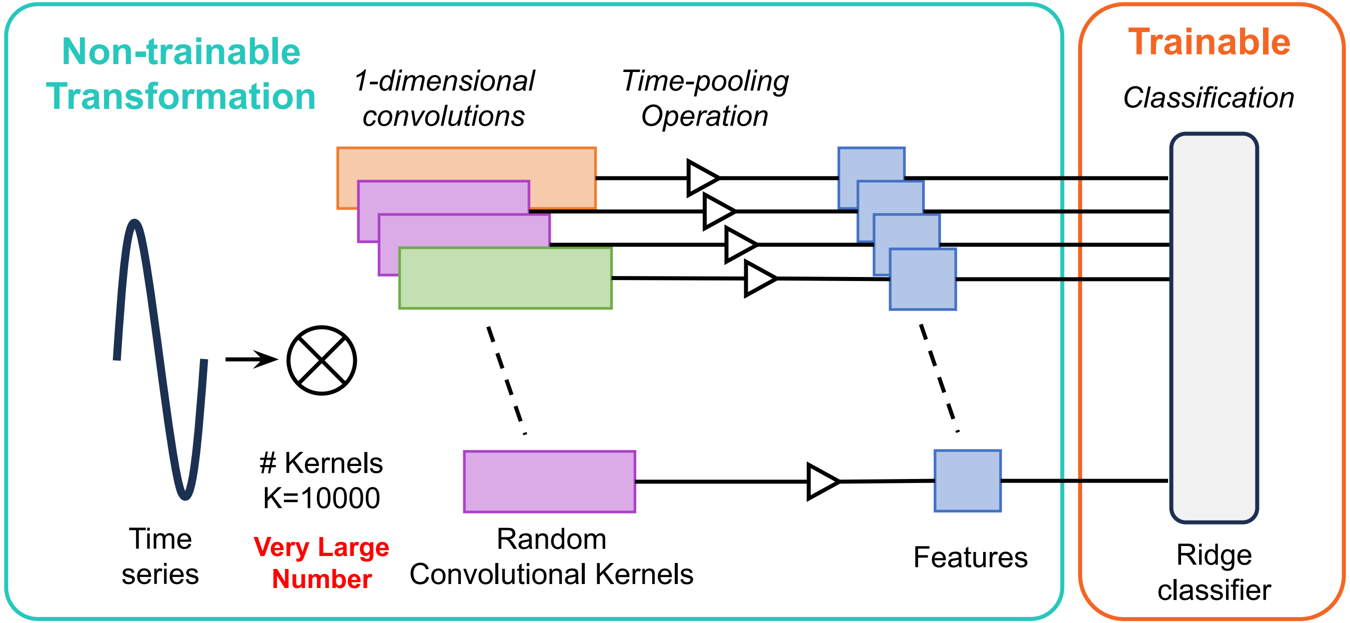
Random Convolutional Kernel Transform (ROCKET)* is a **transformation stage** which can be applied to **time-series data.**

This is KEY!

ROCKET
Kernels: Random
Pooling: MAX + PPV
 # Features: 20000

MiniRocket
Kernels: Dictionary
Pooling: PPV
 # Features: 10000

MultiRocket
Kernels: Dictionary
Pooling:
PPV+MPV+MIPV+LSPV
 # Features: 50000



* Dempster et al. [2020]

ROCKET Strengths and Weaknesses

- State-of-the-art performance for TSC
- Fast and simple training
- Less prone to overfitting (less parameters)

ROCKET Strengths and Weaknesses

- State-of-the-art performance for TSC
- Fast and simple training
- Less prone to overfitting (less parameters)

- It produces many features (many useless)
- Scales poorly with the number of channels
- Difficult to interpret

Hands-on Time:
Notebook 2 Updated