Advanced Time Series

Lecture 5: Classification II

Gleb Ivashkevich

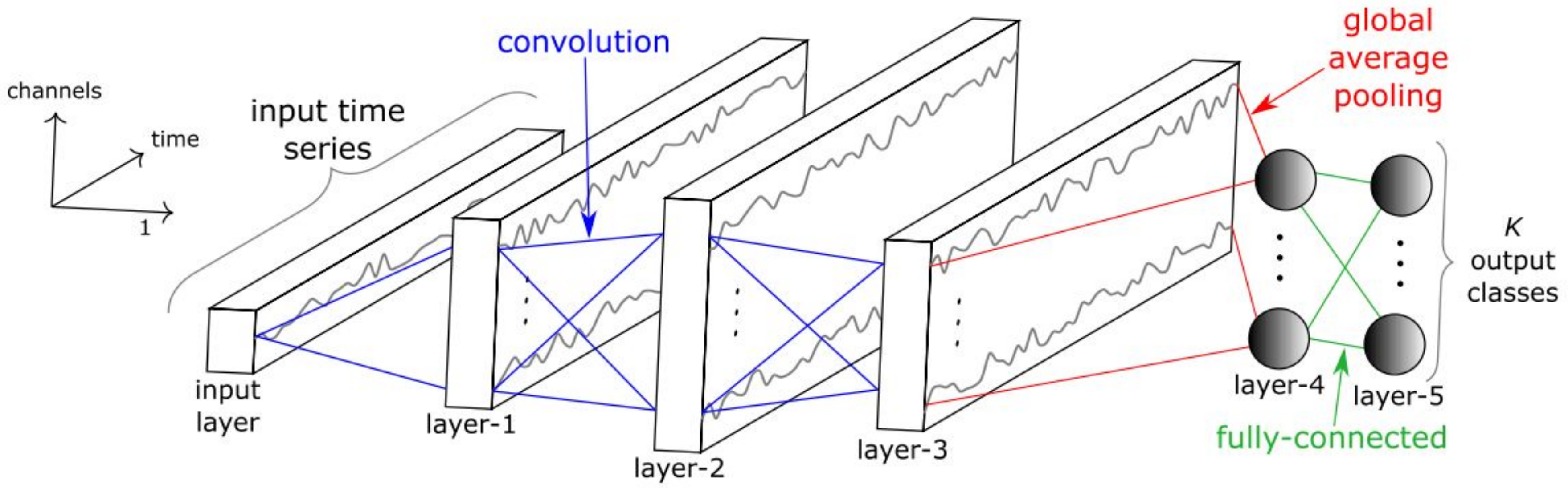
Today

Time series classification → time-to-event and PdM:

- **fully convolutional network** for t. s. classification
- **InceptionTime** for t. s. Classification
- **other architectures** for t. s. classification
- **predictive maintenance**: setup
- **survival analysis** basics, TTE distributions

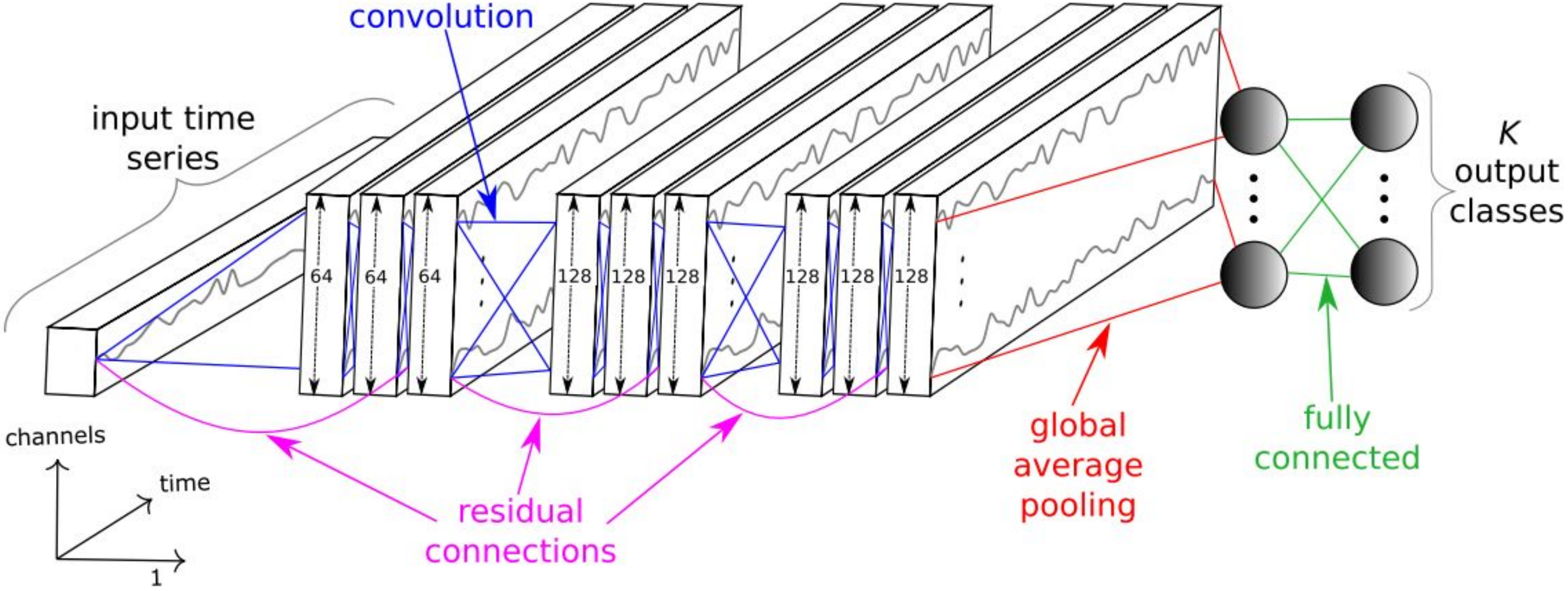
Classification architectures

Fully convolutional



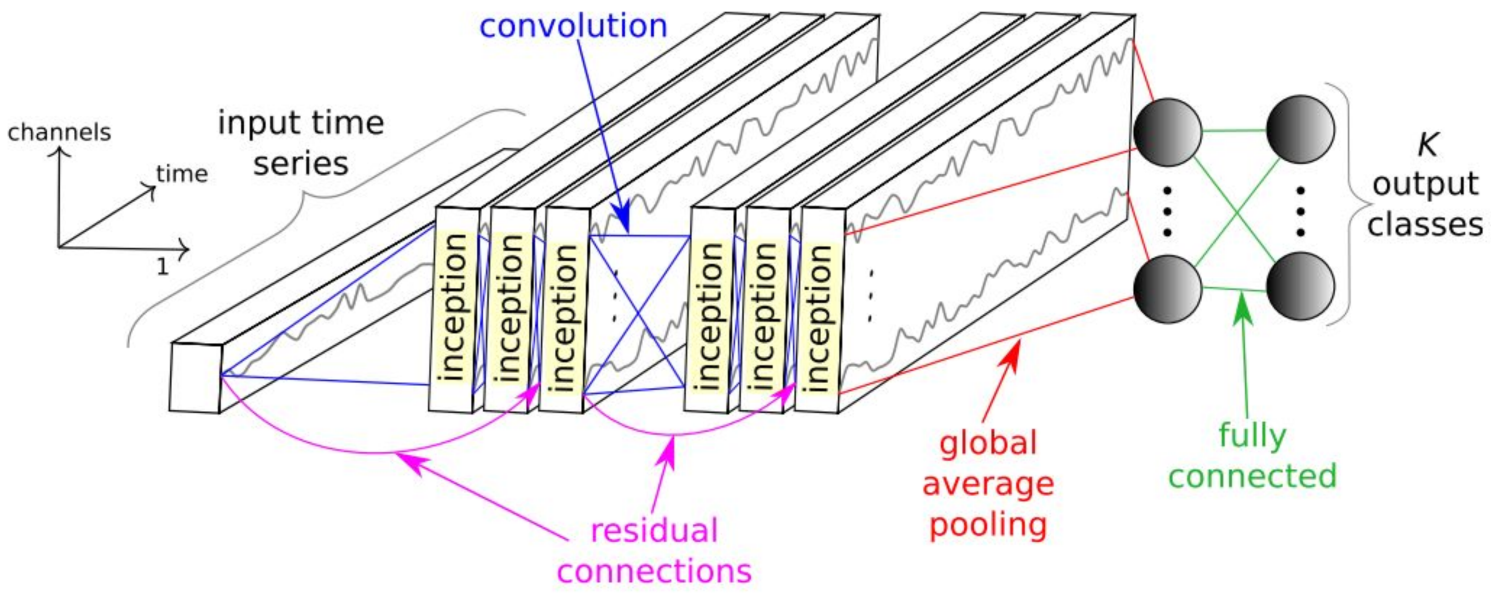
- Pictures: [Deep learning for time series classification: a review]

Residual



- Pictures: [Deep learning for time series classification: a review]

InceptionTime



- Picture: [InceptionTime: Finding AlexNet for Time Series Classification]()

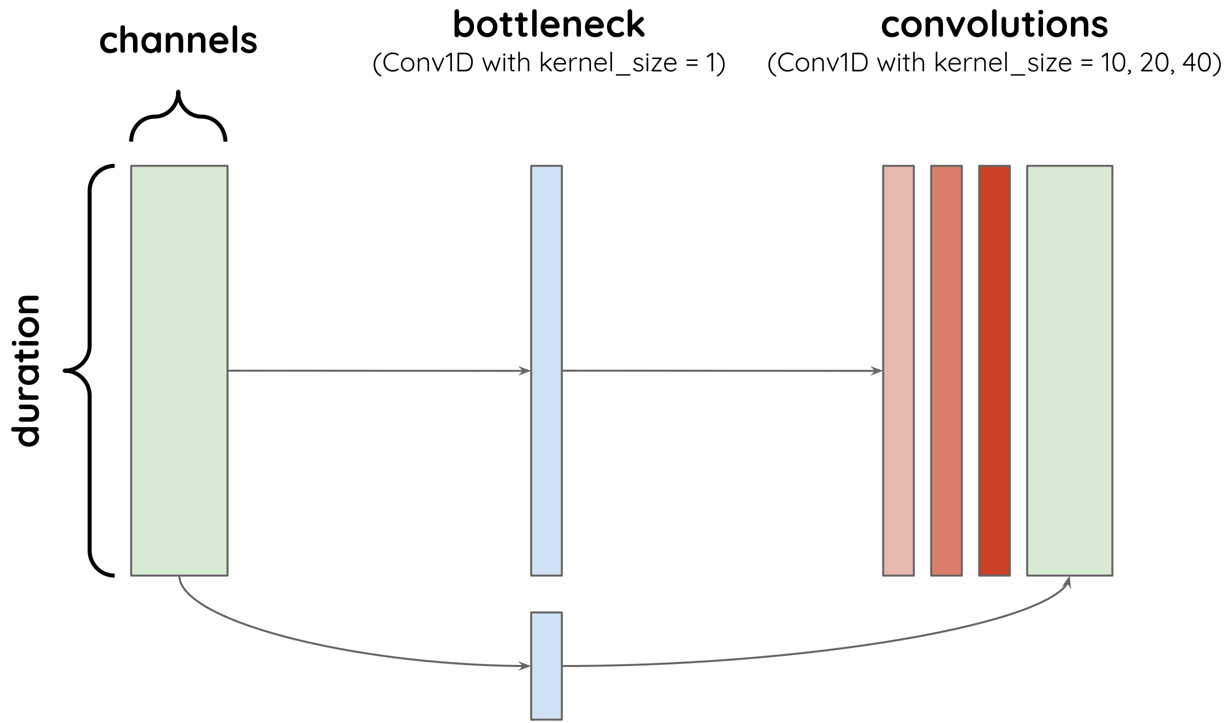
InceptionTime implementation

InceptionTime

Inception blocks:

- introduced in Going Deeper with Convolutions
- more efficient computationally
- nicely captures multiple (although close) spatial/time scales

Inception block for t. s.



Other architectures

ROCKET

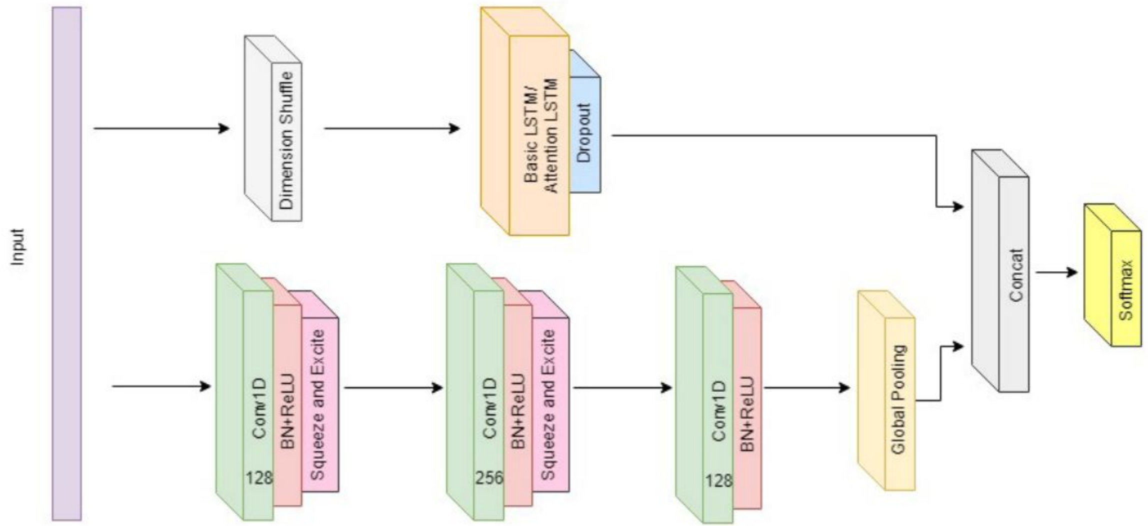
“Single-layer” convolutional network:

- random kernels (size, dilation, etc.)
- linear classifier

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

LSTM-FCN

[Multivariate LSTM-FCNs for Time Series Classification]



TTE and predictive maintenance

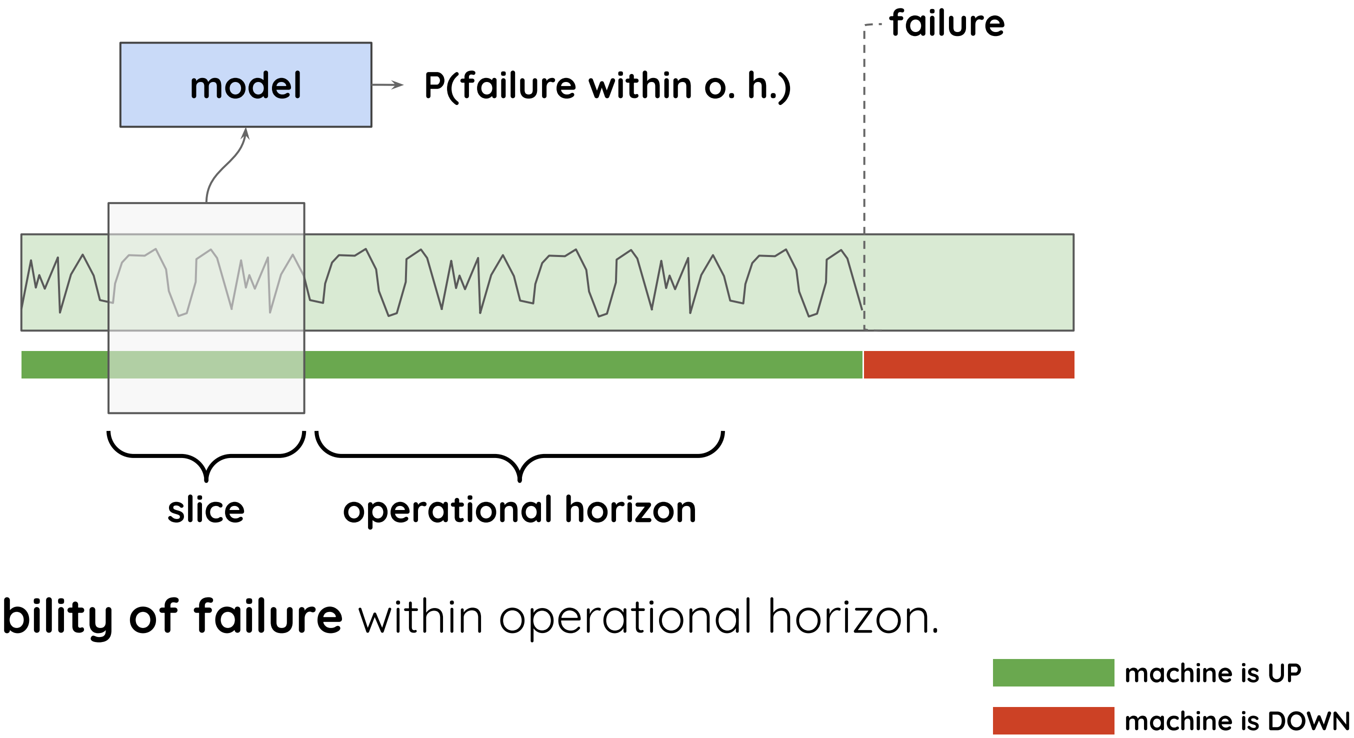
TTE and PdM

Typical scenario:

- **equipment,** vehicles, etc. fails from time to time
- **sensors** provide time series data (often used for other reasons)
- **failures data** is collected as well
- can we **predict failures using sensors data?**

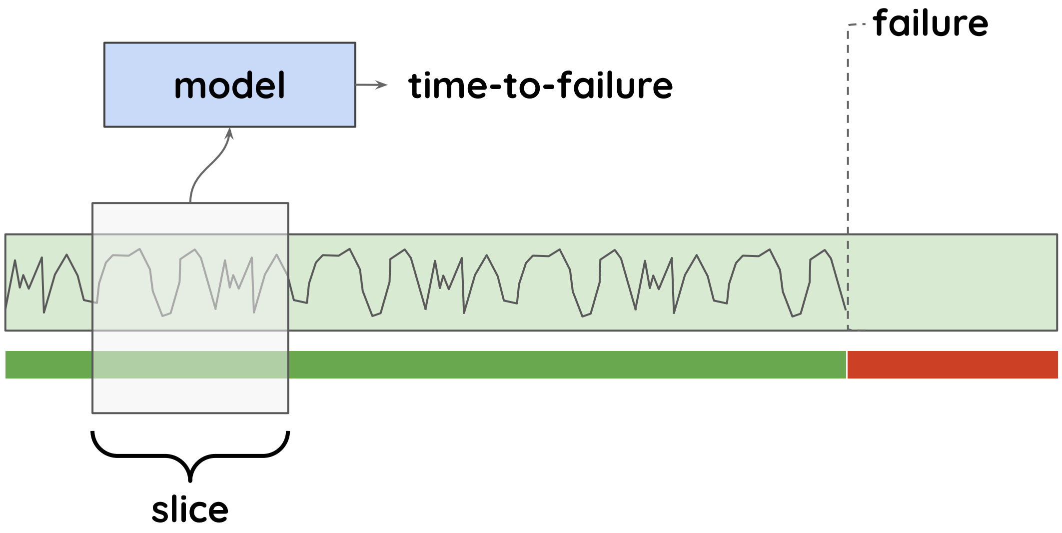
Value: improved **operational efficiency**

Setup: probability



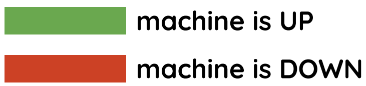
Predict **probability of failure** within operational horizon.

Setup: TTE



Predict **time-to-failure**.

Way more unstable if formulated naively.



TTE and PdM

Naive formulation:

- create some windowed features/use deep learning model
- train a classification model
- rolling predictions

TTE and PdM

When formulated naively:

- failure probability over a **single** o. h. may be **not enough**: no planning beyond o. h.
- **hard to communicate**
- no intrinsic **risk** concept

TTE and PdM

Solution:

- **survival analysis**
- well known in medicine and other domains
- has intrinsic **risk** concept
- can be married with **deep learning**

Survival analysis

Concept 1:

- **survival function:** probability of surviving past t

$$S(t) = P(\underbrace{T}_{\text{time of failure}} > t)$$

Survival analysis

Concept 2:

- **hazard function:** conditioned event rate

$$\lambda(t) = -\frac{S'(t)}{S(t)}$$

Hazard function deciphered

Concept 2:

- **hazard function:** conditioned event rate

$$\lambda(t) = -\frac{P(T>t+dt)-P(T>t)}{S(t)dt}$$

Hazard function deciphered

Concept 3:

- **proportional** hazards model (Cox regression)

$$\lambda(t) = \lambda_0(t)\exp(a_i X_i)$$

covariates

TTE and PdM

Given the model:

- get **entire** survival function
- quantify how covariates influence the risk
- extend **beyond** linear model

Next time

- extension of survival analysis for deep learning
- representation learning

questions?