Prof. Heitor Silvério Lopes
Prof. Thiago H. Silva

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

CPGEI

Data Mining & Knowledge Discovery

Class 8 – Time-series & Signals Mining

2025

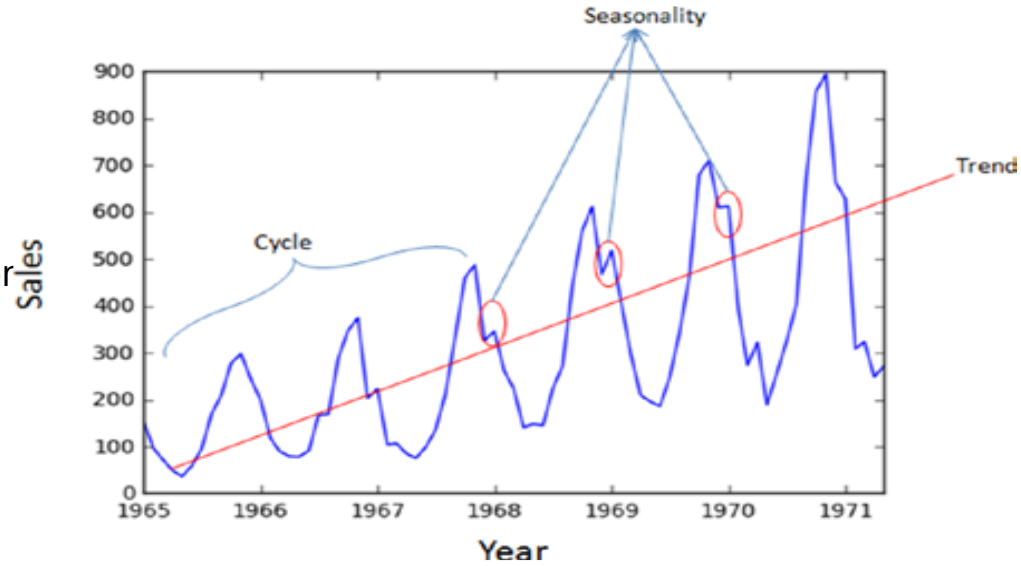
Tasks in time-series & signals mining

- Classification
  - To classify distinct time-series into pre-defined classes according to their temporal behavior
- Clustering / comparison
  - Estimate the similarity of time-series and group them according such similarity

- It is **outside** the scope of this course: time-series prediction
  - Based on the past behavior of a time-series, estimate its future behavior

Main components of time-series

- Trend:
  - Long-term increment or decrement
- Sazonality:
  - Short-term periodic effects
- Cycles:
  - Long oscillations of unknown duration (cyclycal ups and downs), usually caused by external factors
- Residuals:
  - The remainder of the signal after the other three componentes have been eliminated
  - They are not always clearly detectable



Steps for processing time-series

- **Pre-processing**: to eliminate spurious noise (seasonality). It can be done by sampling signal averages (low-pass filter)
- **Normalization**: to adjust time-series to the same amplitude scale
- **Windowing and segmentation**: to divide the time-series into fixed-size (or overlapping) segments to be further processed
- **Feature extraction**: to apply mathematical methods to transform a time-series into a low-dimensional numeric vector, in which each element of the vector ha a specific meaning
- **Feature selection**: to reduce the dimensionality of the data
- **Analysis**: using the reduced vector, to apply the traditional classification/clustering methods

Feature extraction from time-series

- Classes of features:
  - Statistics domain: mean, variance, skewness, kurtosis, standard deviation, etc
  - Time domain: autocorrelation, periodicity, trend
  - Frequency domain: Fourier transform coefficients, spectral entropy, spectral power, dominant Frequency
  - Wavelet domain: wavelet coefficients, energy in specific frequency bands, wavelet entropy
- Other approaches:
  - Neural networks: regular or convolutional NNs
  - Autoencoders

Important python libraries for time-series processing

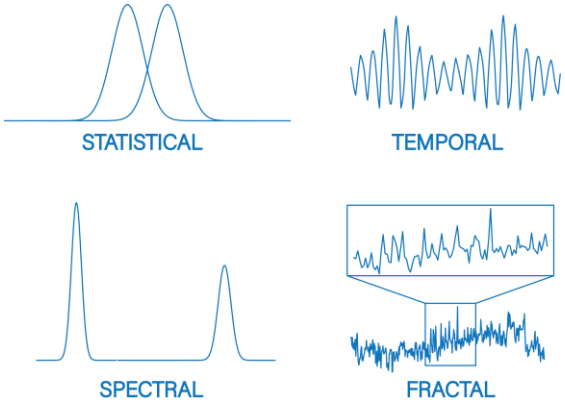
- TSFEL
- Librosa
- pyAudio Analysis
- Tsfresh
- Pyts

What is the problem ?

- How to choose the best atributes of a given dataset for constructing a predictive or descriptive model ?
- If the number of attributes is small:
  - If the user has previous knowledge about the data domain, just select the more representative atributes
  - If the computational power is large, perform a factorial experimente, that is, **test all possible combinations** of attributes and compare results.
- If the number of attributes is (very) large:
  - It is imperative to use **a computationally efficient method** to select the most adequate attributes and discard the remaining

TSFEL library

- Extracts 65 features from a time-series, grouped as:
  - Time domain
  - Statistics domain
  - Spectral domain
  - Fractal domain
- TSFEL requires very low programming effort
- Documents : https://tsfel.readthedocs.io/en/latest/
- GITHUB: https://github.com/fraunhoferportugal/tsfel
- Original paper:
  - Barandas, Marília and Folgado, Duarte, et al., TSFEL: Time Series Feature Extraction Library. *SoftwareX* 11 (2020).
- :



Statistical domain

Features	Computational Cost
ECDF	1
ECDF Percentile	1
ECDF Percentile Count	1
Histogram	1
Interquartile range	1
Kurtosis	1
Max	1
Mean	1
Mean absolute deviation	1
Median	1
Median absolute deviation	1
Min	1
Root mean square	1
Skewness	1
Standard deviation	1
Variance	1

Temporal domain

Features	Computational Cost
Absolute energy	1
Area under the curve	1
Autocorrelation	1
Centroid	1
Entropy	1
Mean absolute diff	1
Mean diff	1
Median absolute diff	1
Median diff	1
Negative turning points	1
Peak to peak distance	1
Positive turning points	1
Signal distance	1
Slope	1
Sum absolute diff	1
Total energy	1
Zero crossing rate	1
Neighbourhood peaks	1

Spectral domain

Features	Computational Cost
FFT mean coefficient	1
Fundamental frequency	1
Human range energy	2
LPCC	1
MFCC	1
Max power spectrum	1
Maximum frequency	1
Median frequency	1
Power bandwidth	1
Spectral centroid	2
Spectral decrease	1
Spectral distance	1
Spectral entropy	1
Spectral kurtosis	2
Spectral positive turning points	1
Spectral roll-off	1
Spectral roll-on	1
Spectral skewness	2
Spectral slope	1
Spectral spread	2
Spectral variation	1
Wavelet absolute mean	2
Wavelet energy	2
Wavelet standard deviation	2
Wavelet entropy	2
Wavelet variance	2

Fractal domain

Features	Computational Cost
Detrended fluctuation analysis (DFA)	3
Higuchi fractal dimension	3
Hurst exponent	3
Maximum fractal length	3
Multiscale entropy (MSE)	1
Petrosian fractal dimension	1

Librosa library

- Specially created for **music** and **audio** analysis
- Documents: https://librosa.org/doc/latest/index.html
- GITHUB: https://github.com/librosa/librosa
- Reference:
  - McFee, B., Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. "librosa: Audio and music signal analysis in python." In *Proceedings of the 14th Python in Science Conference*, pp. 18-25. 2015

Spectral features

- Compute a chromagram from a waveform or power spectrogram.
- Constant-Q chromagram
- Computes the chroma variant "Chroma Energy Normalized" (CENS)
- Variable-Q chromagram
- Compute a mel-scaled spectrogram.
- Mel-frequency cepstral coefficients (MFCCs)
- Compute root-mean-square (RMS) value for each frame, either from the audio samples `y` or from a spectrogram `S`.
- Compute the spectral centroid.
- Compute p'th-order spectral bandwidth.
- Compute spectral contrast
- Compute spectral flatness
- Compute roll-off frequency.
- Get coefficients of fitting an nth-order polynomial to the columns of a spectrogram.
- Computes the tonal centroid features (tonnetz)
- Compute the zero-crossing rate of an audio time series.

Rhythm features

- Estimate the tempo (beats per minute)
- Compute the tempogram: local autocorrelation of the onset strength envelope.
- Compute the Fourier tempogram: the short-time Fourier transform of the onset strength e
- Tempogram ratio features, also known as spectral rhythm patterns.

Feature manipulation

- Compute delta features: local estimate of the derivative of the input data along the selected a
- Short-term history embedding: vertically concatenate a data vector or matrix with delayed cop

Feature inversion

- Approximate STFT magnitude from a Mel power spectrogram.
- Invert a mel power spectrogram to audio using Griffin-Lim.
- Invert Mel-frequency cepstral coefficients to approximate a Mel power spectrogram.
- Convert Mel-frequency cepstral coefficients to a time-domain audio signal

pyAudioAnalysis library

- Download: https://pypi.org/project/pyAudioAnalysis/
  - Extracts MFCCs, spectrogram, chromagram, etc
  - Classify unknown sounds
  - Train and evaluate classifiers for audio segments
  - Train regression models to classification
  - Performs dimensionality reduction
- GITHUB: https://github.com/tyiannak/pyAudioAnalysis/wiki
- Reference:
  - https://hackernoon.com/audio-handling-basics-how-to-process-audio-files-using-python-cli-jo283u3y

Feature ID	Feature Name	Description
1	Zero Crossing Rate	The rate of sign-changes of the signal during the duration of a particular frame.
2	Energy	The sum of squares of the signal values, normalized by the respective frame length.
3	Entropy of Energy	The entropy of sub-frames' normalized energies. It can be interpreted as a measure of abrupt changes.
4	Spectral Centroid	The center of gravity of the spectrum.
5	Spectral Spread	The second central moment of the spectrum.
6	Spectral Entropy	Entropy of the normalized spectral energies for a set of sub-frames.
7	Spectral Flux	The squared difference between the normalized magnitudes of the spectra of the two successive frames.
8	Spectral Rolloff	The frequency below which 90% of the magnitude distribution of the spectrum is concentrated.
9-21	MFCCs	Mel Frequency Cepstral Coefficients form a cepstral representation where the frequency bands are not linear but distributed according to the mel-scale.
22-33	Chroma Vector	A 12-element representation of the spectral energy where the bins represent the 12 equal-tempered pitch classes of western-type music (semitone spacing).
34	Chroma Deviation	The standard deviation of the 12 chroma coefficients.

pyAudio Analysis

Case study #1: Human Activity Recognition (HAR)

- It is a multivariate time-series that aims at identifying human actions:
  - Walking, climbing stairs, descending stairs, sitting, standing, lying down
- Sensors used are 3D- accelerometers and gyroscopes (cell phone)
- Data: https://archive.ics.uci.edu/ml/datasets/human+activity+recognition+using+smartphones
- Google Colab notebook: https://github.com/fraunhoferportugal/tsfel/blob/master/notebooks/TSFEL_HAR_Example.ipynb

Case study #1: Human Activity Recognition (HAR)

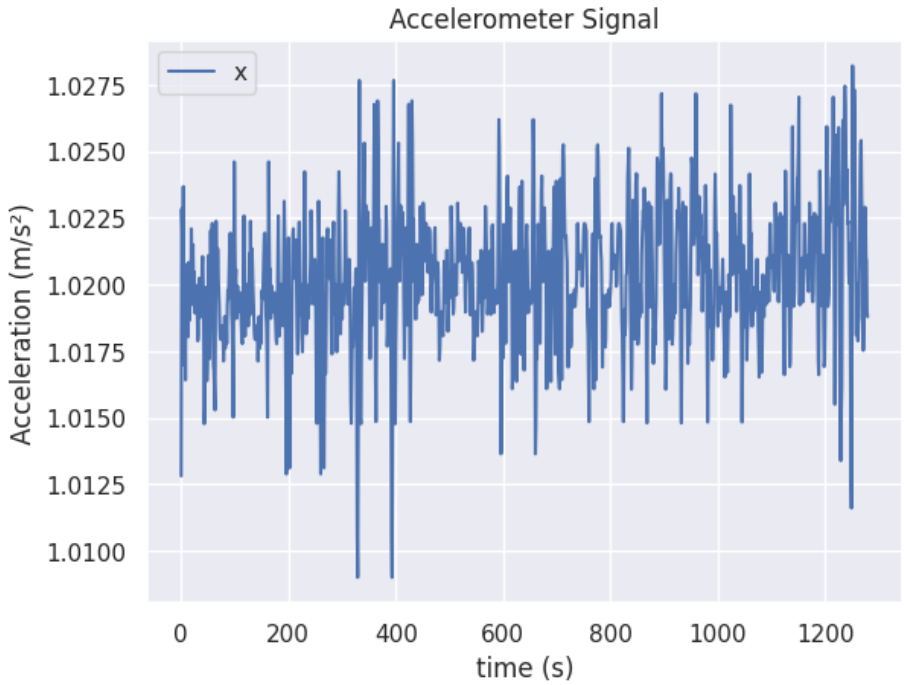
- TSFEL extracts 561 features:

```
import tsfel
import pandas as pd

# load dataset
df = pd.read_csv('Dataset.txt')

# Retrieves a pre-defined feature configuration file to extrac
cfg = tsfel.get_features_by_domain()

# Extract features
X = tsfel.time_series_features_extractor(cfg, df)
```

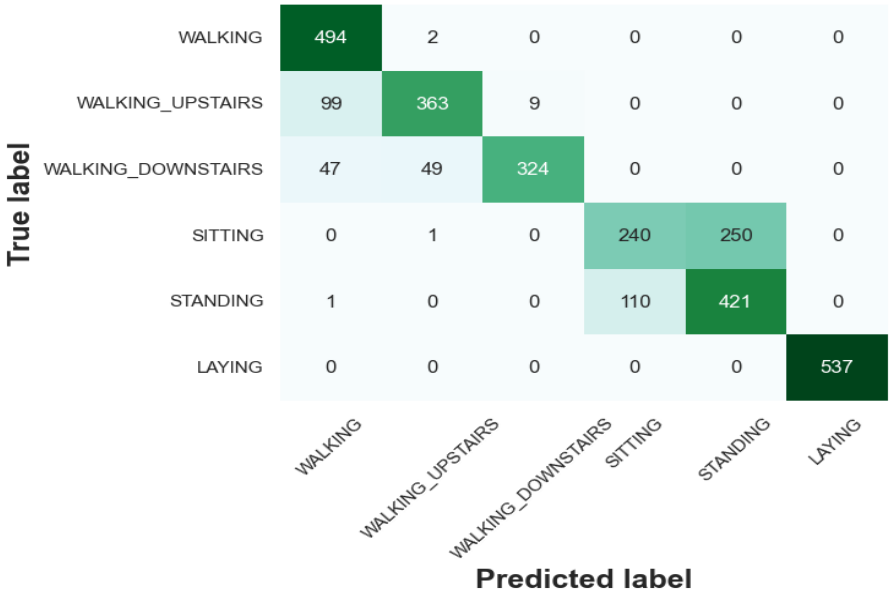


Case study #1: Human Activity Recognition (HAR)

- Classification results using Random Forest

	precision	recall	f1-score	support
WALKING	0.77	1.00	0.87	496
WALKING_UPSTAIRS	0.87	0.77	0.82	471
WALKING_DOWNSTAIRS	0.97	0.77	0.86	420
SITTING	0.69	0.49	0.57	491
STANDING	0.63	0.79	0.70	532
LAYING	1.00	1.00	1.00	537
accuracy			0.81	2947
macro avg	0.82	0.80	0.80	2947
weighted avg	0.82	0.81	0.80	2947

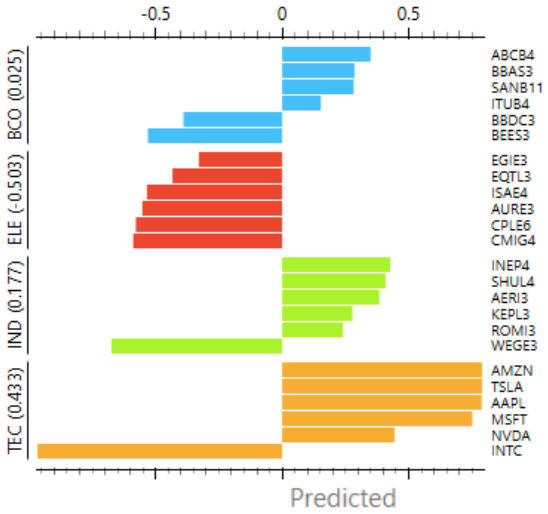
Accuracy: 80.72616219884628%

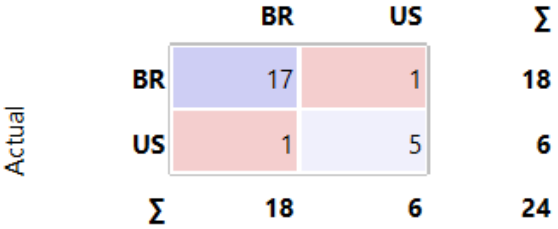


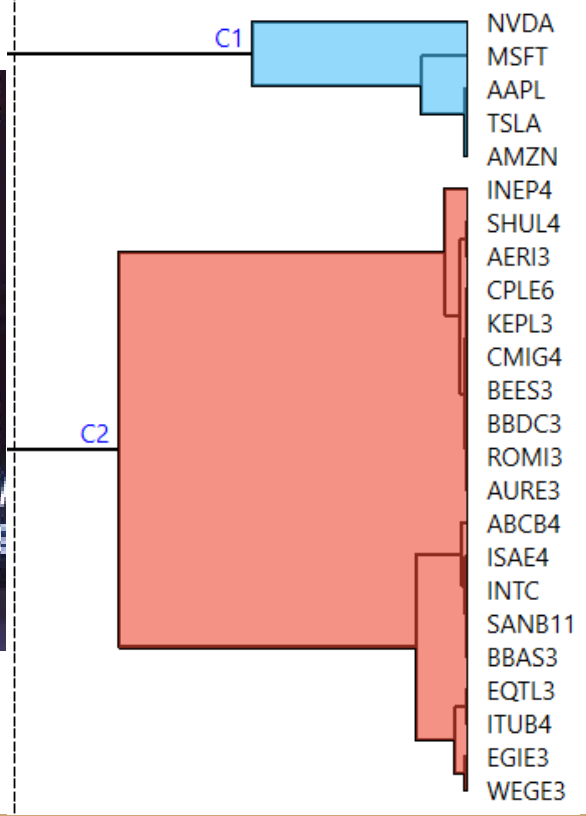
Case study #2: Comparison of Stocks in B3

- 18 stocks negotiated at B3 (Brazil) and 6 from NYSE (USA) were selected
- Closing prices between 2023/dec/4th and 2024/dec/3rd
- Stocks were divided into the following subclasses:
  - Bank sector: BBAS3, BBDC3, ITUB4, SANB11, ABCB4, BEES3
  - Electrical sector: CMIG4, CPLE6, EGIE3, AURE3, EQTL3, ISAE4
  - Industrial sector: SHL4, WEGE3, INEP4, AERI3, KEPL3, ROMI3
  - Technology sector (USA): INTC, MSFT, AAPL, NVDA, TSLA, AMZN
- TSFEL was used for extracting features
- A deep analysis of the features was done, including the effect of feature selection

Case study #2: Comparison of Stocks in B3







Case study#3: Music analysis and classification

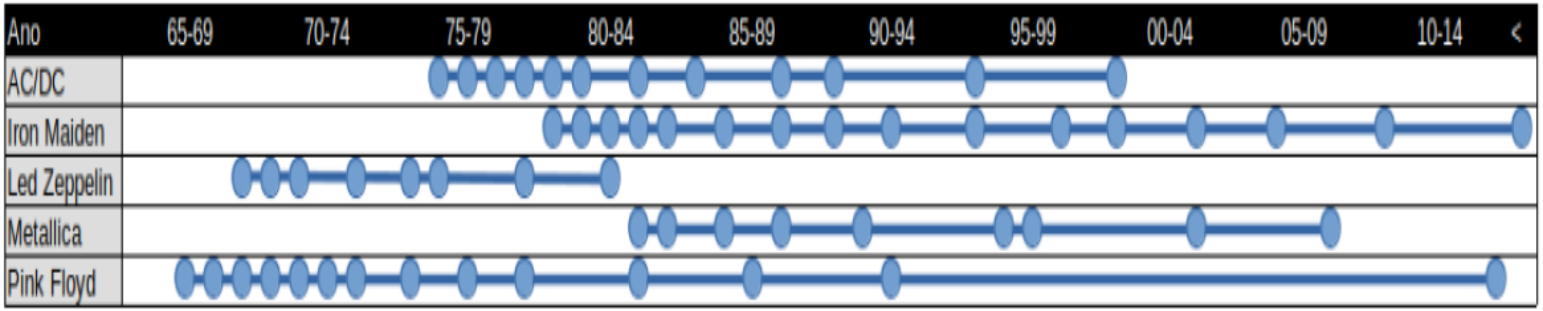
- Based on the paper:
  - Jorge, O.C., Lopes, H.S., Métodos de Aprendizado de Máquina Aplicados ao Reconhecimento Autoral e Temporal de Bandas de Rock and Roll. In: Proc. of XIV Brazilian Conference on Computational Intelligence, 2019.

- Objectives:
  - **Authorial** Classification: Check whether the musical Productions of the same band have their ow “signature” (style coherence)
  - **Temporal** Classification: Check how the musical productions of diferente bans change style over time

Case study#3: Music analysis and classification

- 4 Rock'n roll bands: AC/DC, Iron Maiden, Metallica, Pink Floyd
- Production range: 1967..2014

Banda	Período de Produção	Num. álbuns	Num. músicas
AC/DC	1976-2000	13	130
Iron Maiden	1980-2015	16	158
Led Zeppelin	1969-1982	9	81
Metallica	1983-2008	9	95
Pink Floyd	1967-2014	15	174



Case study#3: Music analysis and classification

- Preprocessing:
  - Input: áudio files in the WAV format
  - Conversion to monoaural and resampling at 16kHz
- Feature extraction:
  - pyAudioAnalysis extracted features regarding:
    - Spectral analysis
    - Chromatic vector
    - Chromatic variation
    - Spectral energy
    - Spectral entropy
    - etc

Case study#3: Music analysis and classification

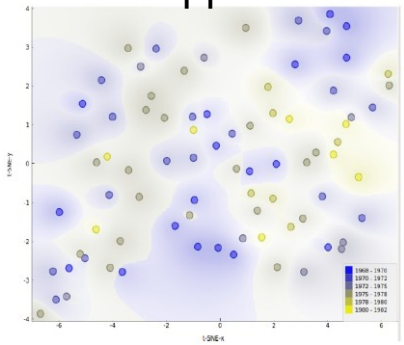
- Methods:
  - Descriptive analysis: classification using C4.5, Ridor, Ripper
  - Predictive analysis: classification using Neural Network (MLP)
  - Visual analysis: T-SNE, silhouette coefficient

- Applicability
  - Authorial classification
  - Temporal classification (5- and 10-years time range)

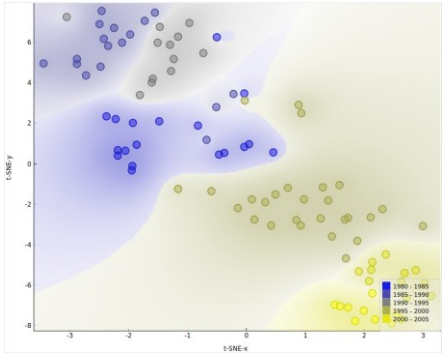
Case study#3: Music analysis and classification

- Results regarding style coherence by band using T-SNEE

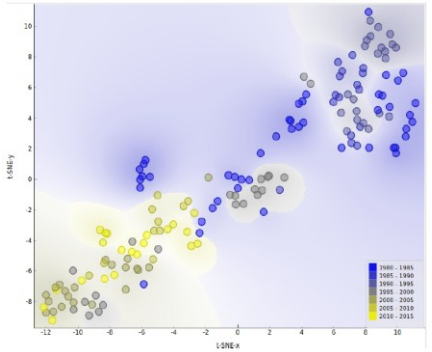
Led Zeppelin



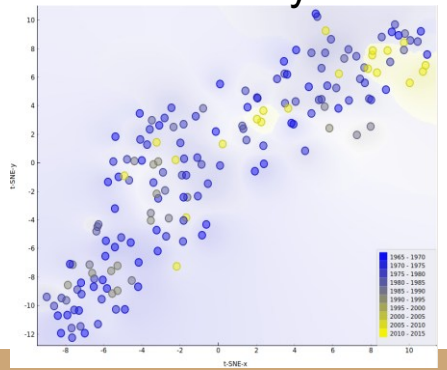
Metallica



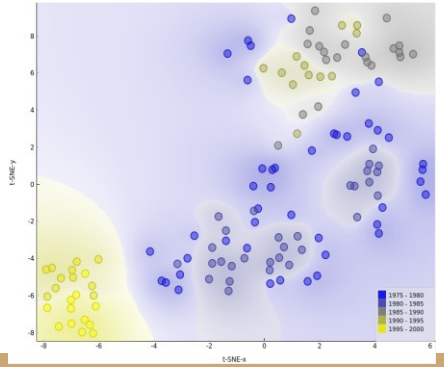
Iron Maiden



Pink Floyd



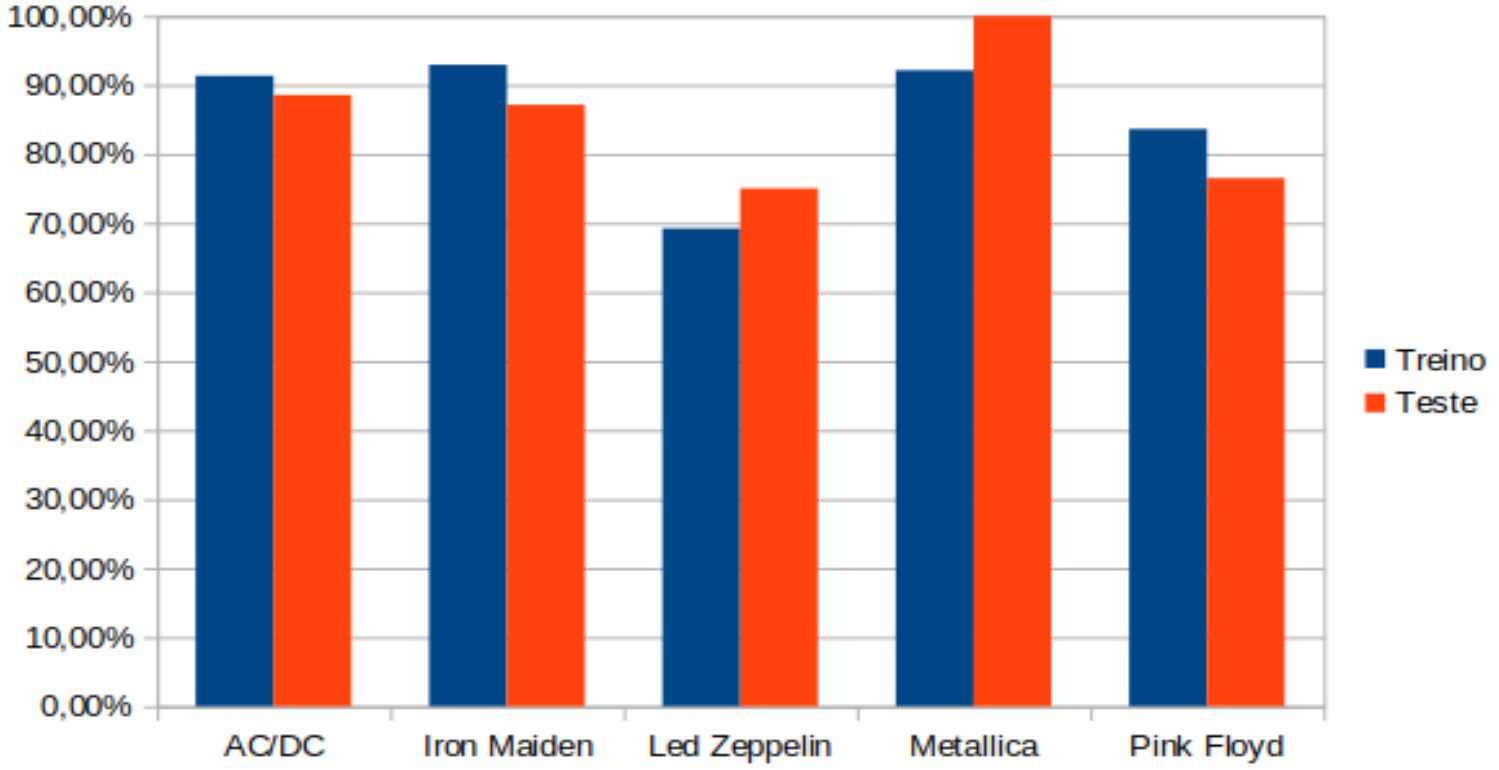
AC / DC



Blue – older Productions
Yellow – recent Productions

Case study#3: Music analysis and classification

- Results for the authorial classification using Neural Networks



Case study#3: Music analysis and classification

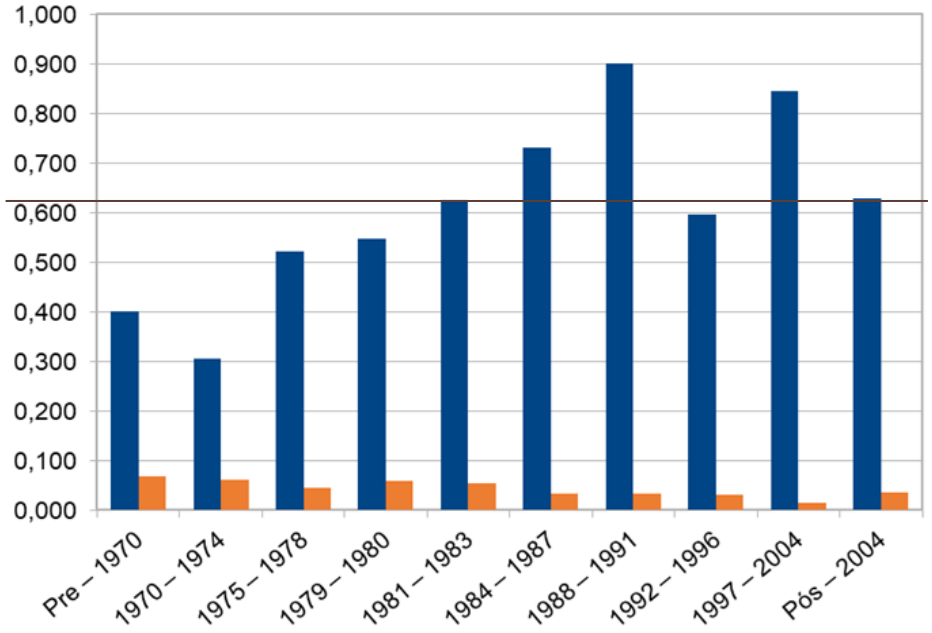
- Temporal classification using Neural Networks for variable periods:

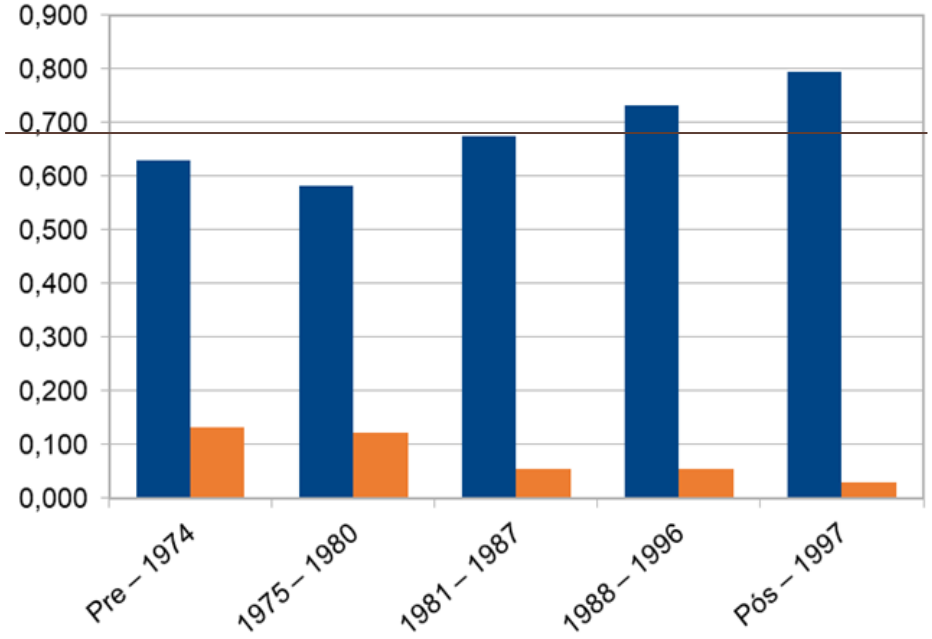
Período	Verdadeiro Positivo	Falso Positivo
Pré - 1970	0,400	0,069
1970 - 1974	0,306	0,061
1975 - 1978	0,522	0,045
1979 - 1980	0,548	0,058
1981 - 1983	0,625	0,055
1984 - 1987	0,731	0,033
1988 - 1991	0,92	0,033
1992 - 1996	0,597	0,031
1997 - 2004	0,846	0,014
Pós - 2004	0,629	0,035
Média	0,611	0,043

Período	Verdadeiro Positivo	Falso Positivo
Pré - 1974	0,630	0,133
1975 - 1980	0,581	0,122
1981 - 1987	0,674	0,055
1988 - 1996	0,732	0,056
Pós - 1997	0,795	0,031
Média	0,682	0,040

Case study#3: Music analysis and classification

- Temporal classification using Neural Networks for short and long periods.
- Before 1980 musical productions are almost exclusive of PF and LZ





Case study#3: Music analysis and classification

- T-SNE: qualitative evaluation of style coherence