# Acoustic Noise in Gravitational-wave Detectors

**Dr. Guillermo Valdes**
Postdoctoral researcher
Texas A&M University – Laboratory of Space Systems and Optomechanics

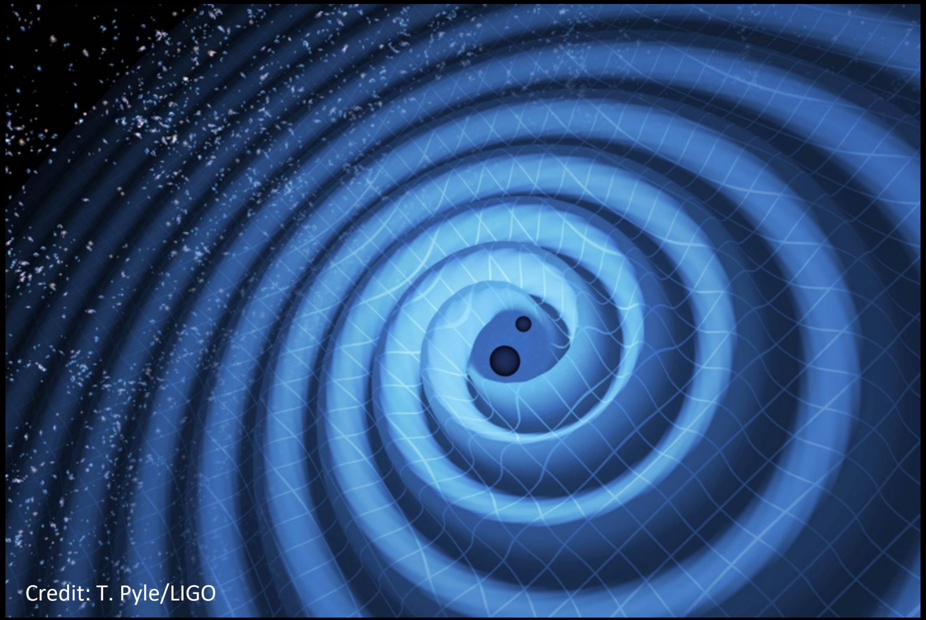
# Introduction

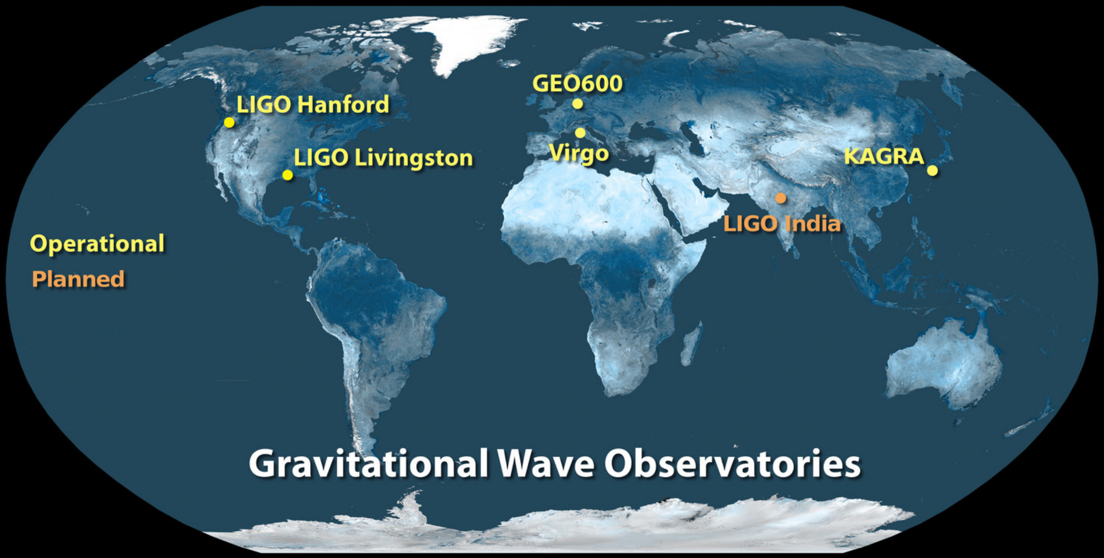
- The sensitivity of **gravitational-wave detectors** can be affected by sporadic short-duration noises (so-called **glitches**) from distinct non-astrophysical sources.
- It is essential to find the glitches' origin to mitigate them.
- We employ machine learning techniques to identify acoustic noise that produces glitches in gravitational-wave detectors.
- We summarize the feature extraction procedures, unsupervised and supervised learning combined with parallel computing techniques to achieve this goal.

**This work is possible thanks to the efforts of the LIGO, Virgo, KAGRA scientific collaboration members.**

# Background

- Gravitational waves are ripples in the space time.
- Gravitational-wave detectors are extremely sensitive instruments, capable of measuring changes in distance thousand of times smaller than the diameter of a proton.

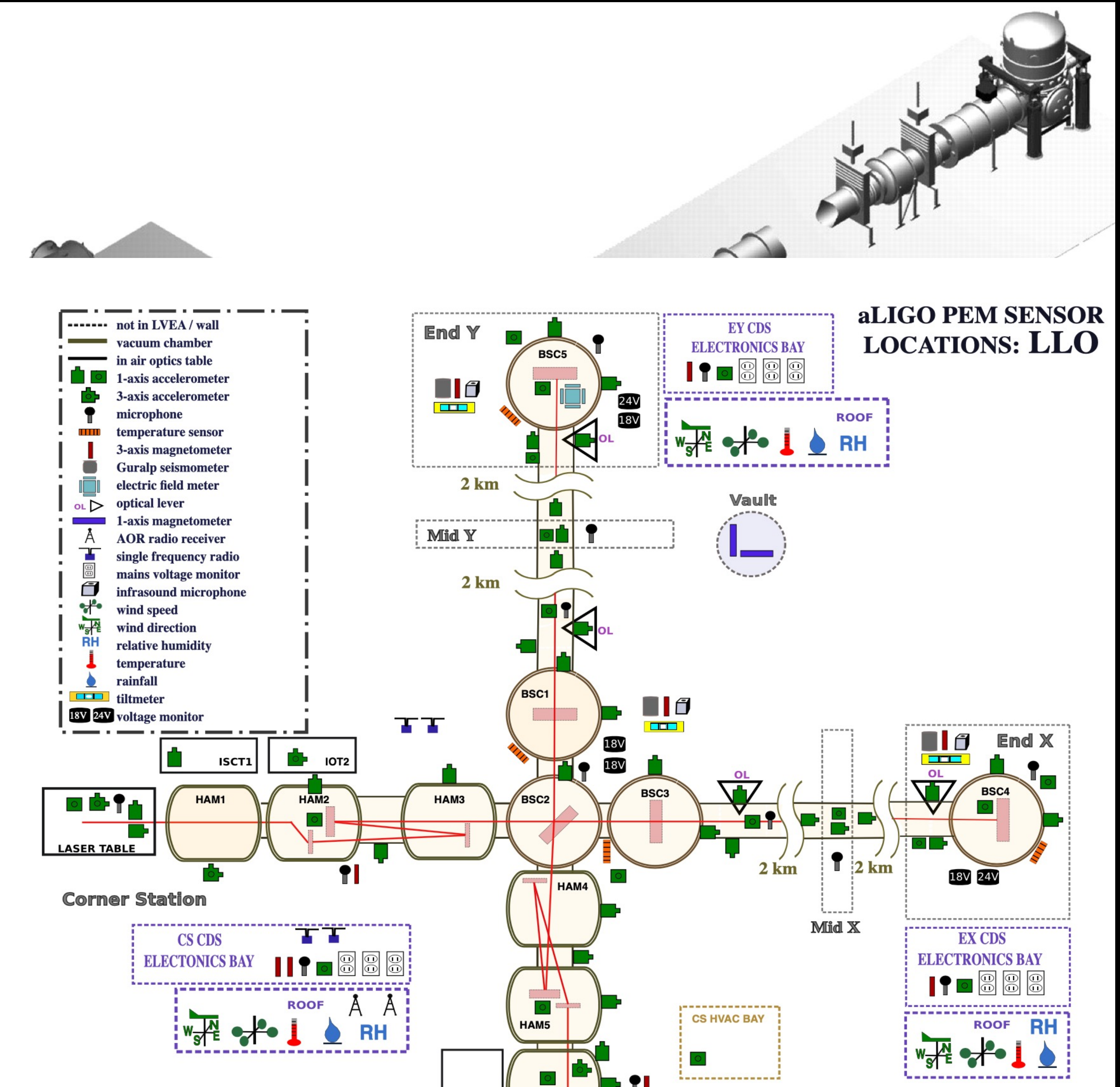



Credit: Caltech/MIT/LIGO Lab

# Background

- Gravitational-wave detectors such as LIGO, consist of an L-shape laser interferometer and several auxiliary systems that make its control possible.
- LIGO uses thousands of sensors monitoring its status and local environment to characterize its behavior and noise.




Credit: Caltech/MIT/LIGO Lab
Created by M. Tse, V. Roma, T. Hardwick

# Motivation

## In summary:

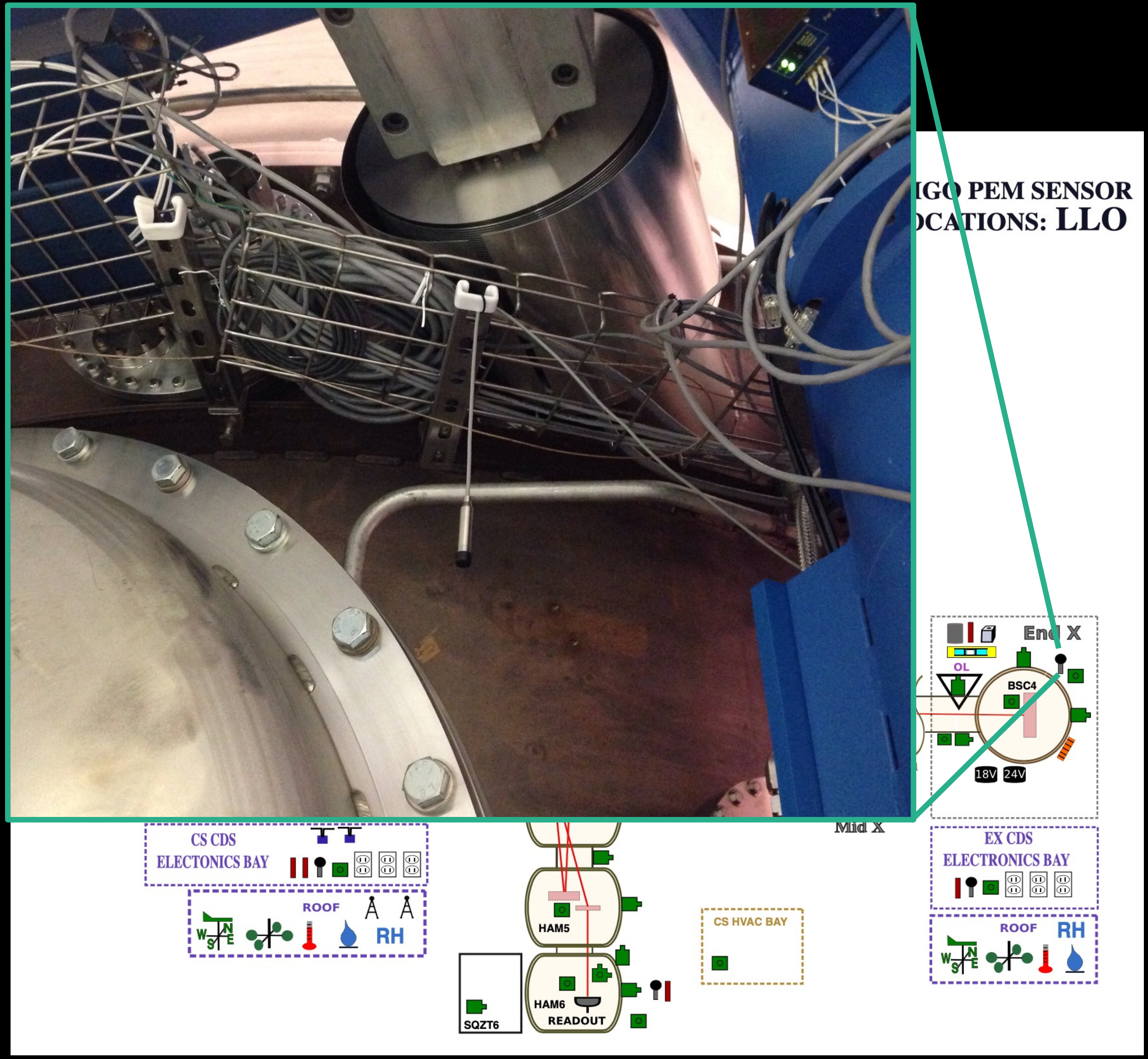
- We have a lot of noise and a lot of data to analyze.
- We need an automated tool to find the noise's origin.

## This work can help to:

- Improve the detector's sensitivity.
- Validate candidate events.

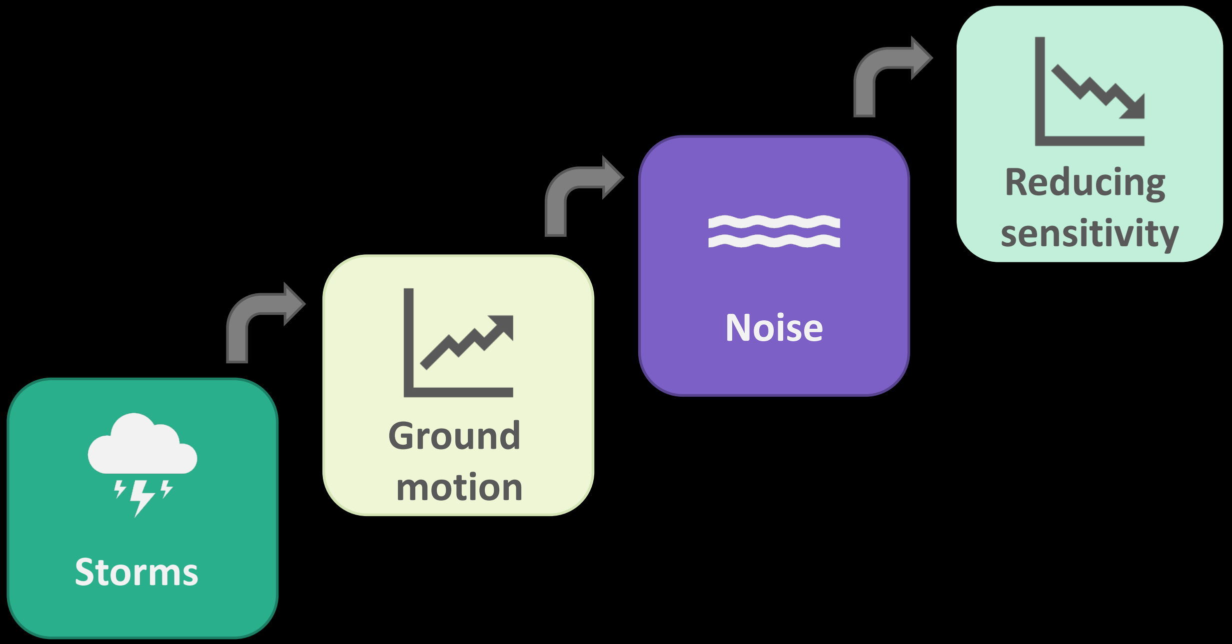
# Acoustic Noise

- We call acoustic noise the one recorded by microphones.
- We choose it because we consider it easier to recognize by humans.
- We use Omicron triggers to collect the data.
- Omicron is a computational tool that finds segments (so-called triggers) of data distinct from the rest.



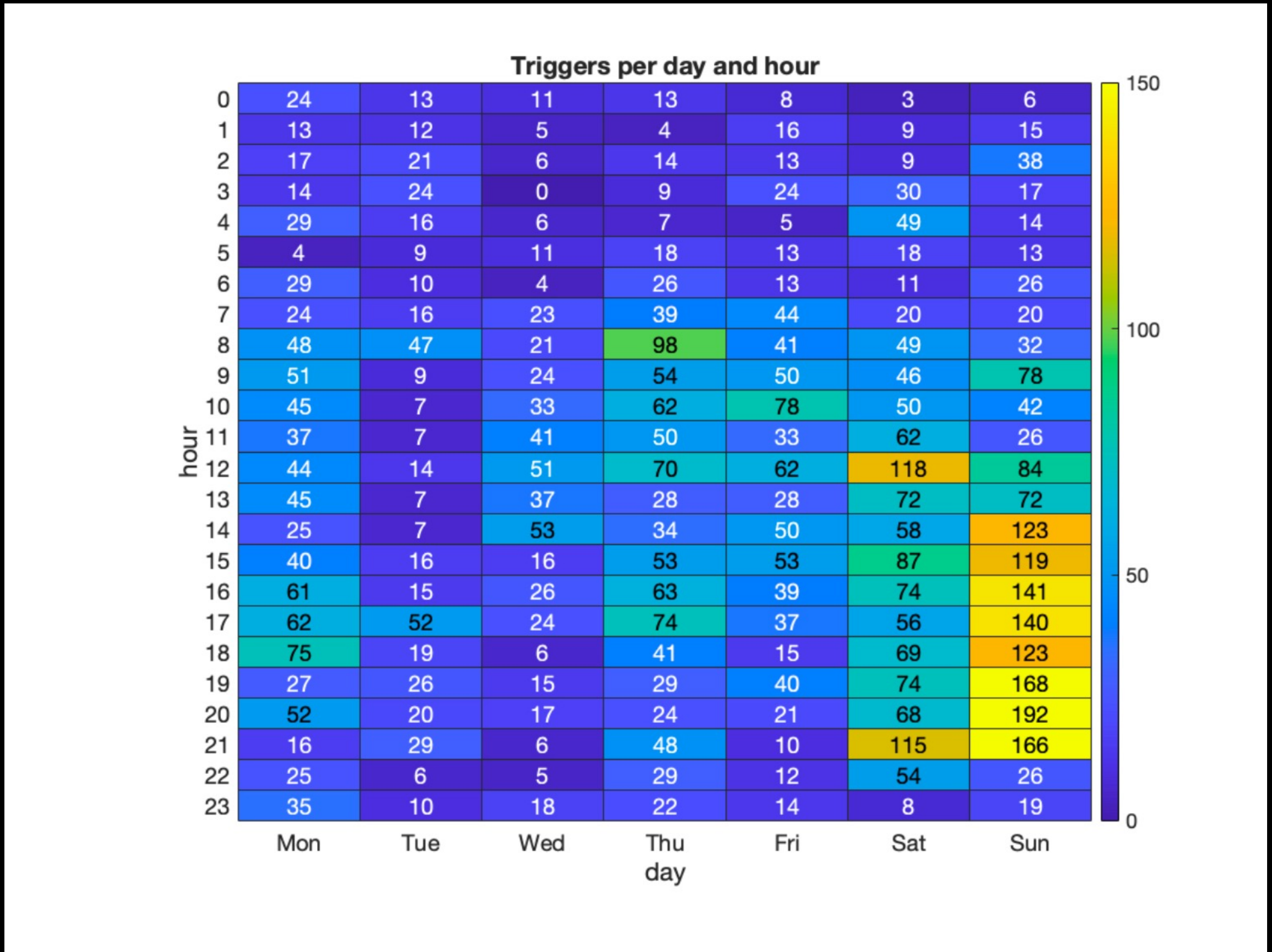
# For example:

- **Thunderstorms** are frequent in the south of Louisiana, USA.
- Rumbling thunder strikes **elevate ground motion**.
- **Light-scattering** noise is generated when light is reflected from moving surfaces and resonates in optical cavities.
- Scattering **reduces** the gravitational-wave detector's **sensitivity**.



# Acoustic noise in LIGO Livingston during O3

- We use Omicron triggers of the LIGO Livingston X-end microphone.
- The high count of triggers on Sundays between 14:00 h and 22:00 h contrasts with the low trigger count on Tuesdays because we studied observing times.
- The detector is more likely to be locked on the quiet Sunday afternoons than on Tuesdays, dedicated for maintenance.



# Acoustic Noise Identification

This work has been developed with the help of graduate students at the Texas A&M University - Laboratory of Space Systems and Optomechanics.

Feature extraction

Dimensionality reduction

Clustering

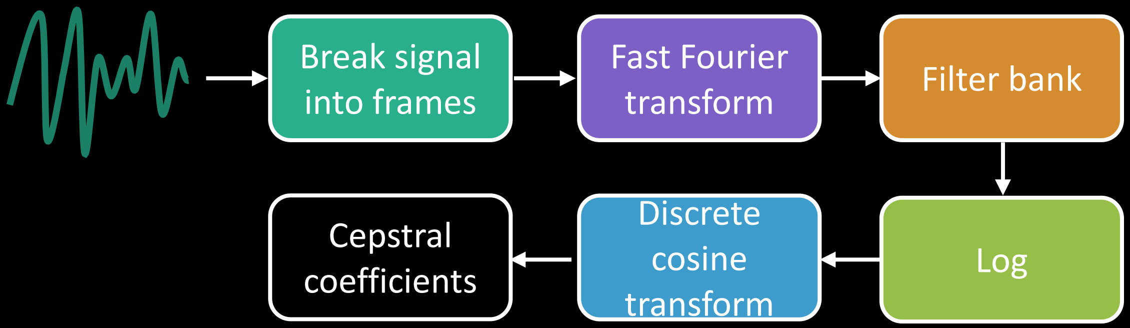
Classification

# Feature Extraction

- The **mel-frequency cepstral coefficients (MFCC)** are features that encode the sound to mimics the function and capabilities of the human ear.

- They were introduced in the 1980s and used in speech recognition and environmental sound classification.

- In MATLAB – Audio Toolbox and in Python – Librosa package.



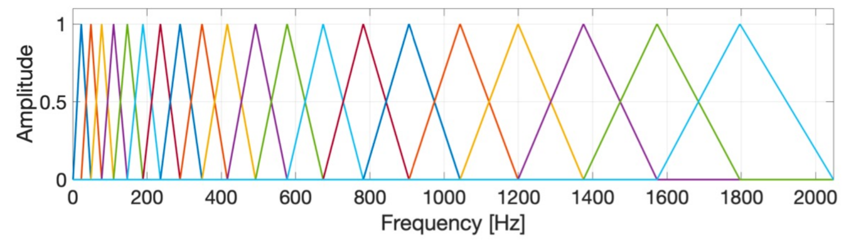




Figure 1: The mel-filterbank is designed as half-overlapped triangular filters equally spaced on the mel-scale.

# High-throughput computing

- **High-throughput computing (HTC)** is the use of distributed computing facilities for applications requiring large computing power over a long period of time [1].

1. Rajkumar Buyya, ... S. Thamarai Selvi, in *Mastering Cloud Computing*, 2013

- HTC is a resource to complete your analyses in a shorter time.
- HTCondor is an open-source HTC software framework.
- The Open Science Grid (OSG) is a nationally-funded consortium of computing resources available for most researchers.

# High-throughput computing

- We need 3 files to complete the feature extraction of several samples.

```
#!/home/guillermo.valdes/.conda/envs/hht-py37/bin/python
#

import argparse
import librosa
from gwpy.timeseries import TimeSeries

# command line options
parser = argparse.ArgumentParser()
parser.add_argument("channel")
parser.add_argument("gps", type=int)
parser.add_argument("duration", type=float)
args = parser.parse_args()
print(args)

# define parameters
chname = args.channel
gps   = args.gps
dur = args.duration

# request data
data = TimeSeries.get(chname, gps, gps+dur)

# audio signal parameters
fs = 1/(data.dt.value)
xa = data.value

# extracting features in segments
Mfcc = librosa.feature.mfcc(y=xa, sr=fs, n_mfcc=13)
```

feature extraction happens here

```
#!/usr/bin/env condor_submit

universe = vanilla
executable = /home/guillermo.valdes/.conda/envs/hht-py37/bin/python
arguments = " /home/ExtractFeatures.py $(chanl) $(StartGPS) $(dur)"

request_cpus = 1
request_disk = 1000
request_memory = 1000

output = /home/logs/out.$(StartGPS)
error = /home/logs/err.$(StartGPS)
```

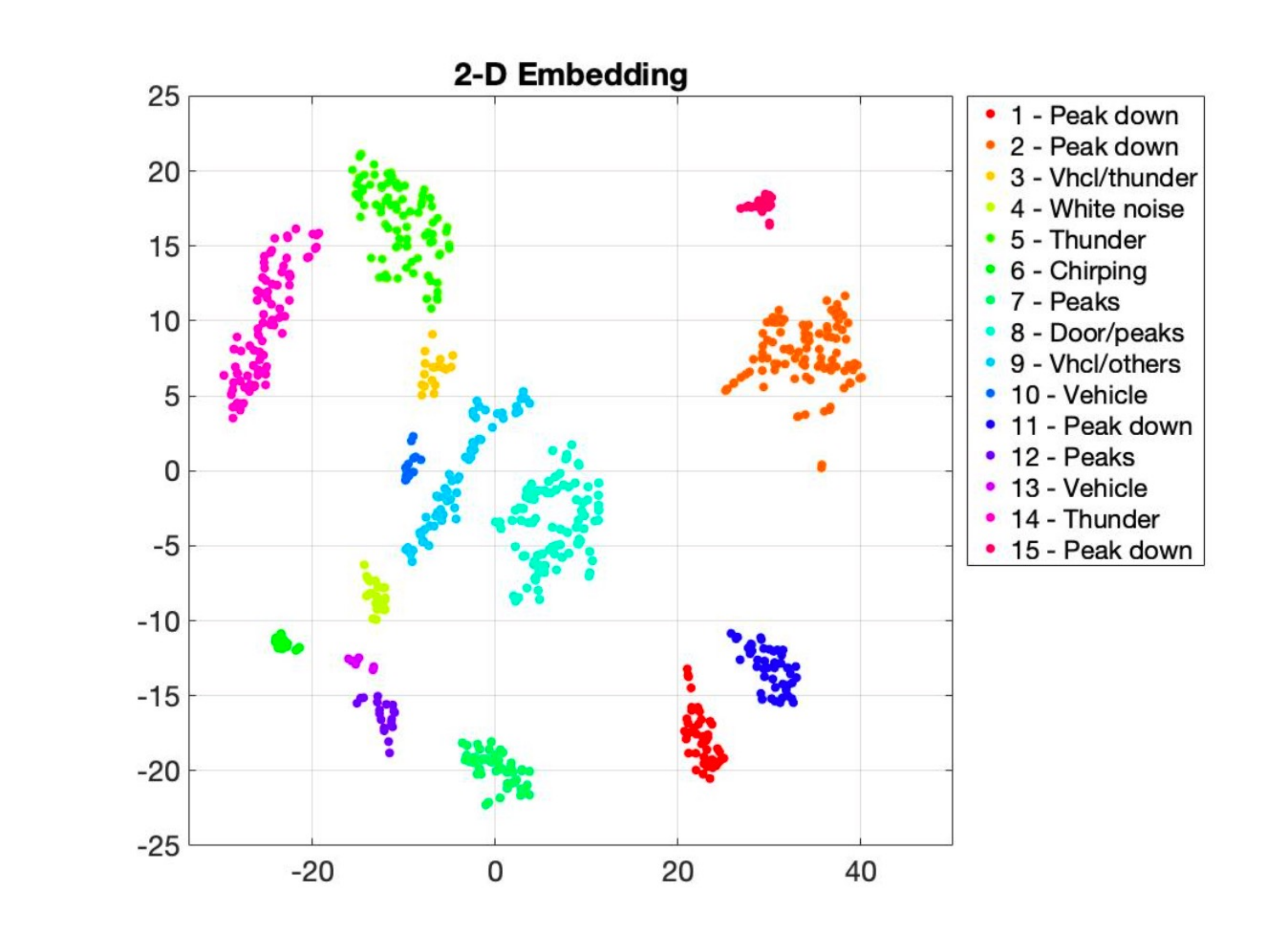
here we call the feature extraction code

```
JOB 1 ExtractFeatures.sub
RETRY 1 1
VARS 1 chanl="L1:PEM-EY_MIC_VEA_PLUSY_DQ" startGPS="1242086421" dur="1"

JOB 2 ExtractFeatures.sub
RETRY 2 1
VARS 2 chanl="L1:PEM-EY_MIC_VEA_PLUSY_DQ" startGPS="1242086422" dur="1"

JOB 3 ExtractFeatures.sub
RETRY 3 1
VARS 3 chanl="L1:PEM-EY_MIC_VEA_PLUSY_DQ" startGPS="1242086425" dur="1"
```

this file runs the green file several times

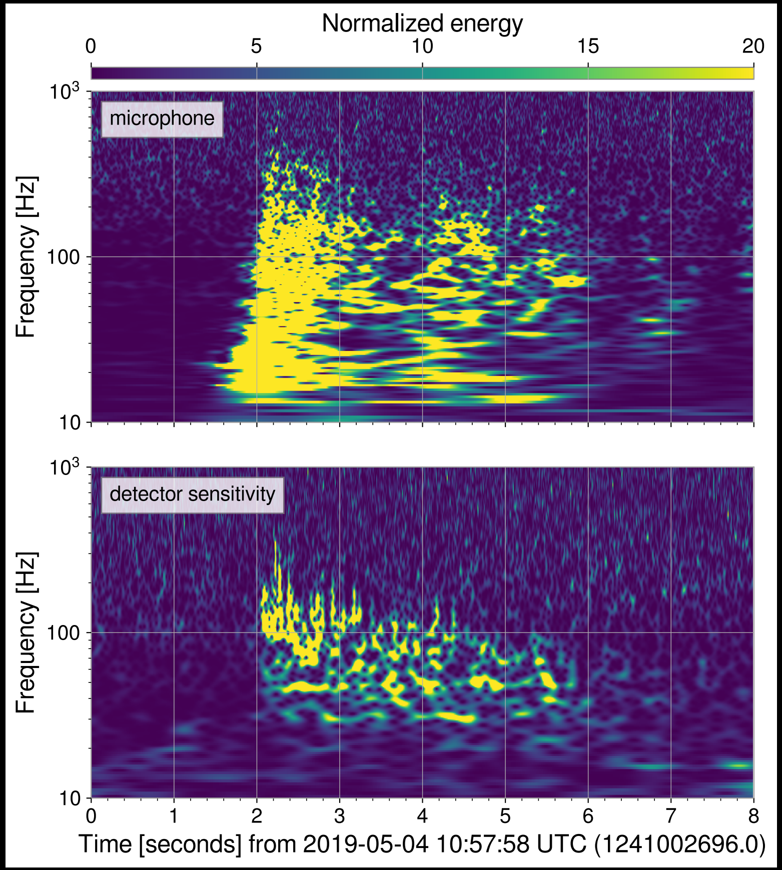
# Dimensionality reduction and clustering

- For dimensionality reduction, we employ t-distributed stochastic neighbor embedding (t-SNE).
- We use spectral clustering, a graph-based algorithm for clustering data points.
- We use triggers with SNR > 100 (665 observations) since we expect these loud triggers to affect the detector's sensitivity if any.

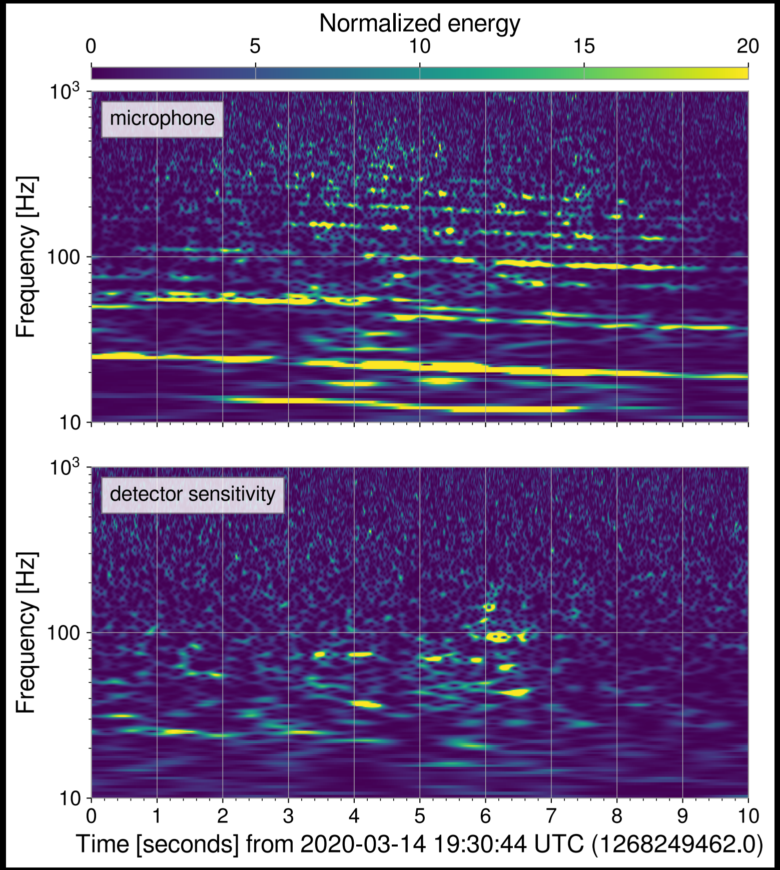


# What is the effect on the detector?

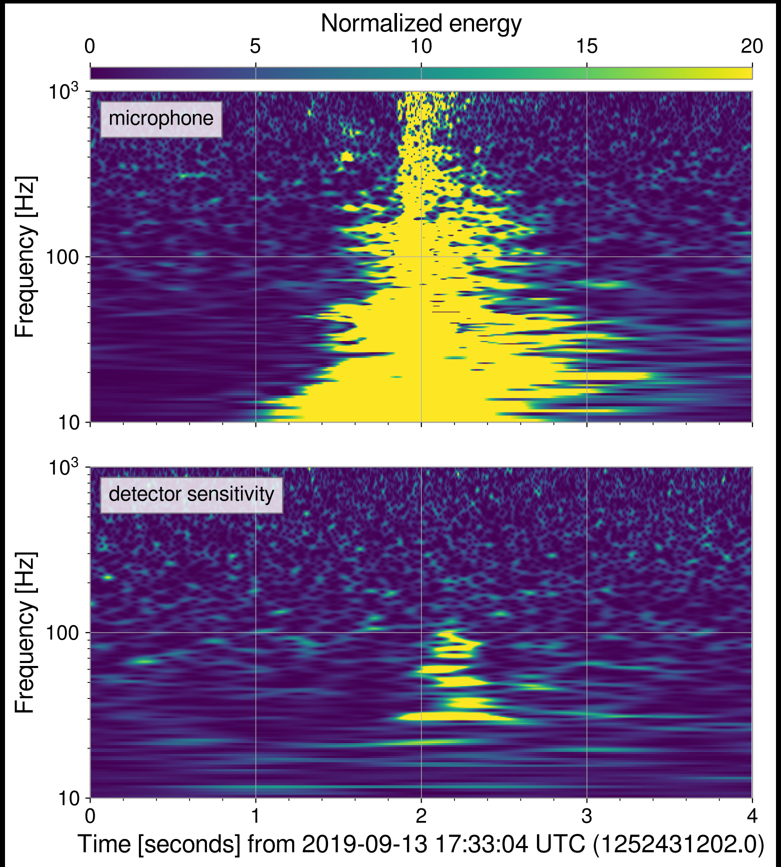
**Thunderstorm**




**Vehicle**

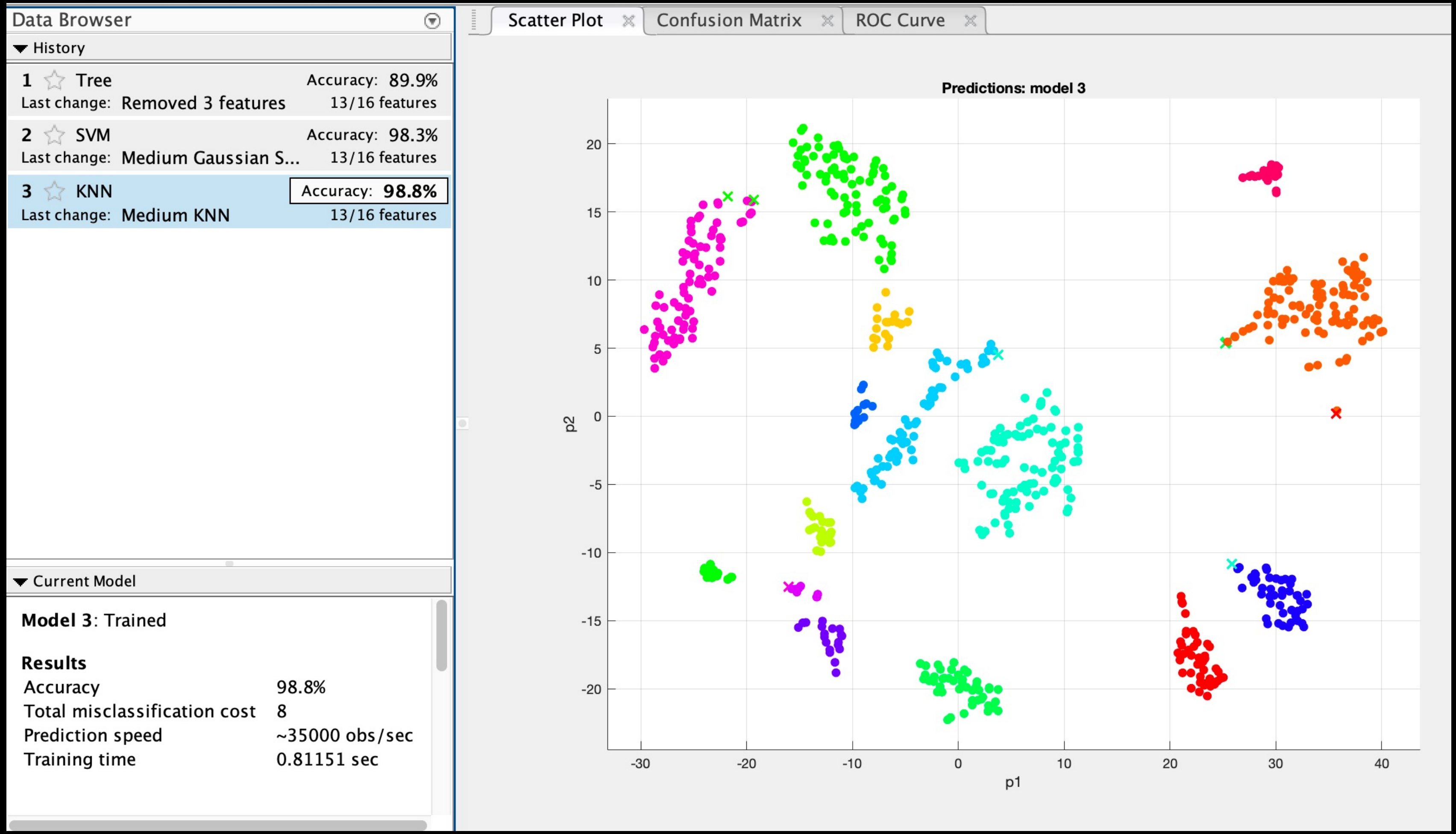



**Closing door**



# Classification

- MATLAB offers a convenient tool to compare the performance of the classifiers.
- We use the same data set to train the classifiers.
- Once we choose a classifier, we can use it for other microphones.



# Summary

- Gravitational-wave detectors are extremely sensitive and are affected by local environmental noise.
- We employ microphones, machine learning techniques, and high-throughput computing to find the origin of the noise.
- We use mel-frequency cepstral coefficients to obtain features from the microphones' recordings.
- Noise produced by events such as thunderstorms and passing-by-vehicles couple into the detector output.
- Our tool can be used to complement the Data Quality Reports for gravitational-wave event candidates.

# Resources

- MFCC introduction
  - https://medium.com/prathena/the-dummys-guide-to-mfcc-aceab2450fd
  - http://practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficients-mfccs/
- Computational options
  - https://librosa.org/doc/latest/index.html
  - https://scikit-learn.org/stable/index.html
  - https://www.mathworks.com/help/audio/ref/mfcc.html
  - https://www.mathworks.com/solutions/machine-learning.html
- High-throughput computing
  - https://support.opensciencegrid.org/support/home