Angie K. Reyes

PyCon 2018

# Identification of Colombian Bird species using Python

## Introduction

- About me
- Why Python?
- Workshop goals

## Background

- LifeClef challenge
- Motivation
- The important things

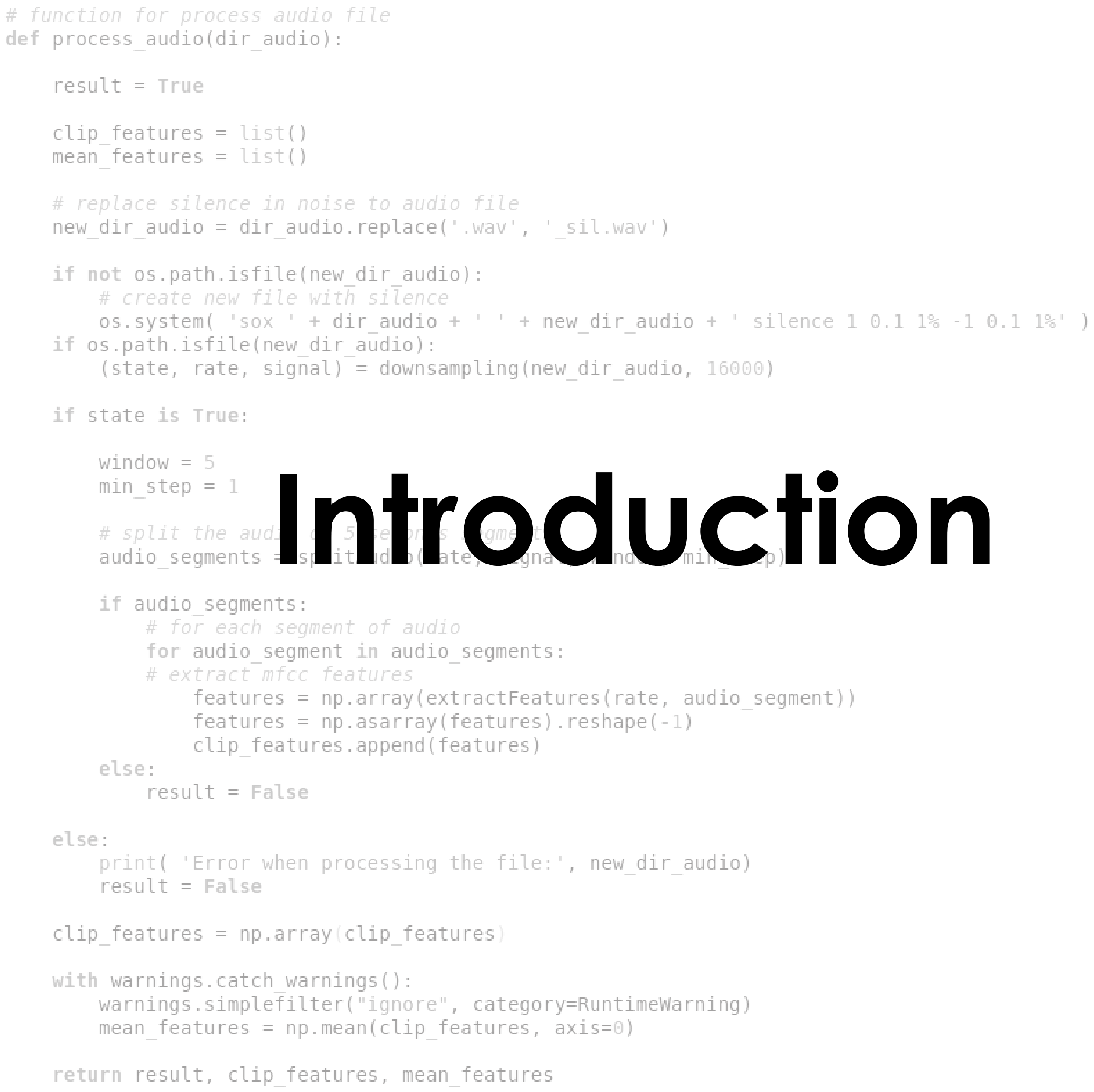
## Content

- Dataset
- Processing data
- Extract of features
- Classification
- Results

## Workshop

- Python & Notebook
- Practical exercise & showing

# Introduction



```
# function for process audio file
def process_audio(dir_audio):

    result = True

    clip_features = list()
    mean_features = list()

    # replace silence in noise to audio file
    new_dir_audio = dir_audio.replace('.wav', '_sil.wav')

    if not os.path.isfile(new_dir_audio):
        # create new file with silence
        os.system( 'sox ' + dir_audio + ' ' + new_dir_audio + ' silence 1 0.1 1% -1 0.1 1%' )
    if os.path.isfile(new_dir_audio):
        (state, rate, signal) = downsampling(new_dir_audio, 16000)

    if state is True:

        window = 5
        min_step = 1

        # split the audio in 5 seconds segments
        audio_segments = splitAudio(rate, signal, window, min_step)

        if audio_segments:
            # for each segment of audio
            for audio_segment in audio_segments:
            # extract mfcc features
                features = np.array(extractFeatures(rate, audio_segment))
                features = np.asarray(features).reshape(-1)
                clip_features.append(features)
        else:
            result = False

    else:
        print( 'Error when processing the file:', new_dir_audio)
        result = False

    clip_features = np.array(clip_features)

    with warnings.catch_warnings():
        warnings.simplefilter("ignore", category=RuntimeWarning)
        mean_features = np.mean(clip_features, axis=0)

    return result, clip_features, mean_features
```

# About me

Introduction

I'm a Systems and Computing Engineer

I'm 25 years old

TICS Girls 2016

PhD student  in the Doctorate in Applied Science program at the Antonio Nariño University.

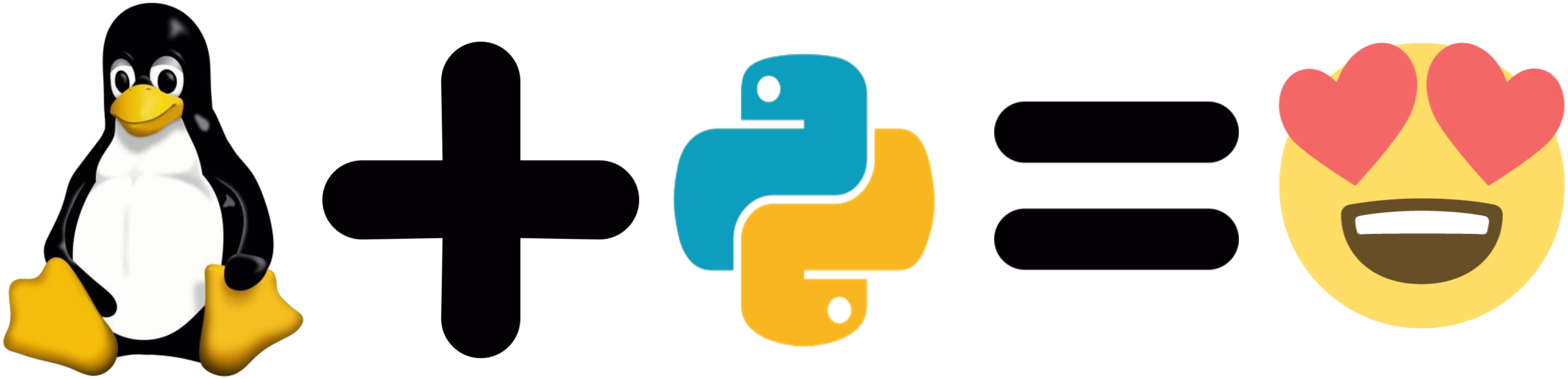
Back-End Development

# Topics

Introduction

- Support Vector Machine (SVM) for Magnetic Resonance Image classification.
- Development of a mobile app and web tool to support non-pharmacological therapies in Alzheimer's patients.
- Identification of bird species using audio feature extraction and SVM.
- Deep Learning for Plant Identification.
- Development and management of big data and machine learning projects (junior developer).
- Power grid modeling using Graph theory.
- Creation of a Smart Grid.

# Why Python?



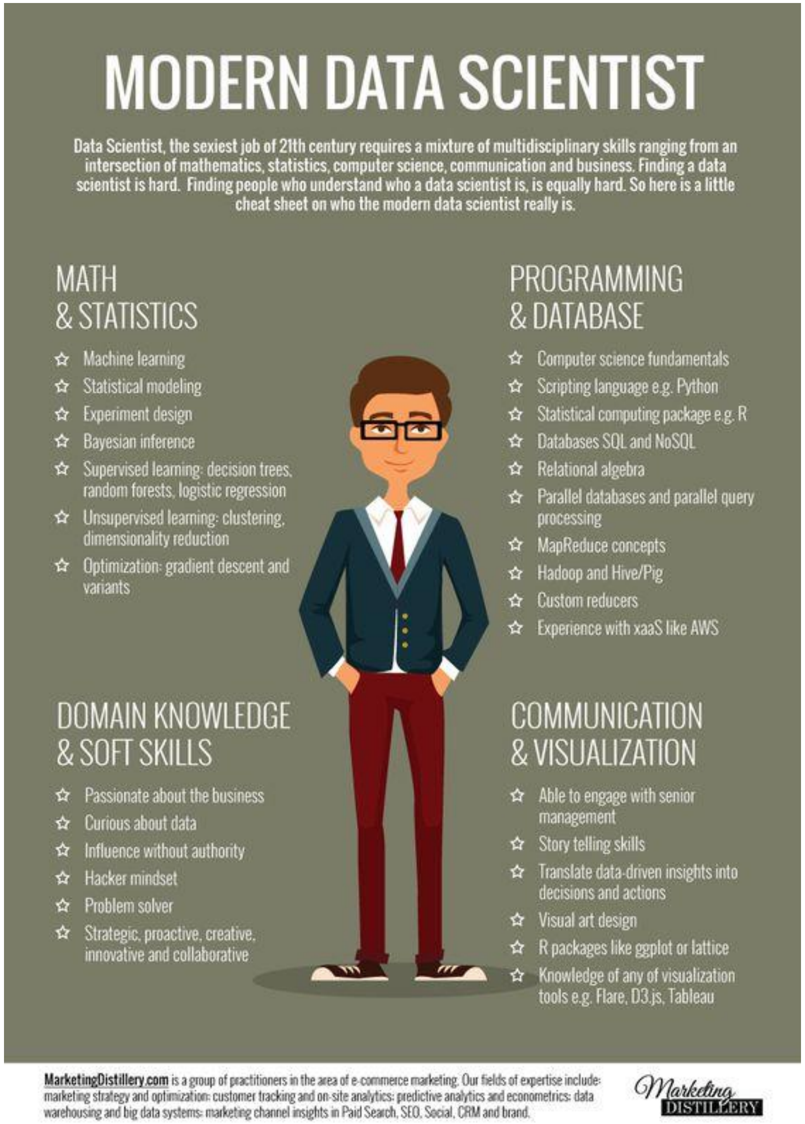
# Workshop goals

**“Desarrolle una pasión por el aprendizaje. Si lo hace, usted nunca dejará de crecer.”**

-Anthony J. D’Angelo.

# Workshop goals

Introduction



## MODERN DATA SCIENTIST

Data Scientist, the sexiest job of 21th century requires a mixture of multidisciplinary skills ranging from an intersection of mathematics, statistics, computer science, communication and business. Finding a data scientist is hard. Finding people who understand who a data scientist is, is equally hard. So here is a little cheat sheet on who the modern data scientist really is.

### MATH & STATISTICS

- Machine learning
- Statistical modeling
- Experiment design
- Bayesian inference
- Supervised learning: decision trees, random forests, logistic regression
- Unsupervised learning: clustering, dimensionality reduction
- Optimization: gradient descent and variants

### PROGRAMMING & DATABASE

- Computer science fundamentals
- Scripting language e.g. Python
- Statistical computing package e.g. R
- Databases SQL and NoSQL
- Relational algebra
- Parallel databases and parallel query processing
- MapReduce concepts
- Hadoop and Hive/Pig
- Custom reducers
- Experience with xaaS like AWS

### DOMAIN KNOWLEDGE & SOFT SKILLS

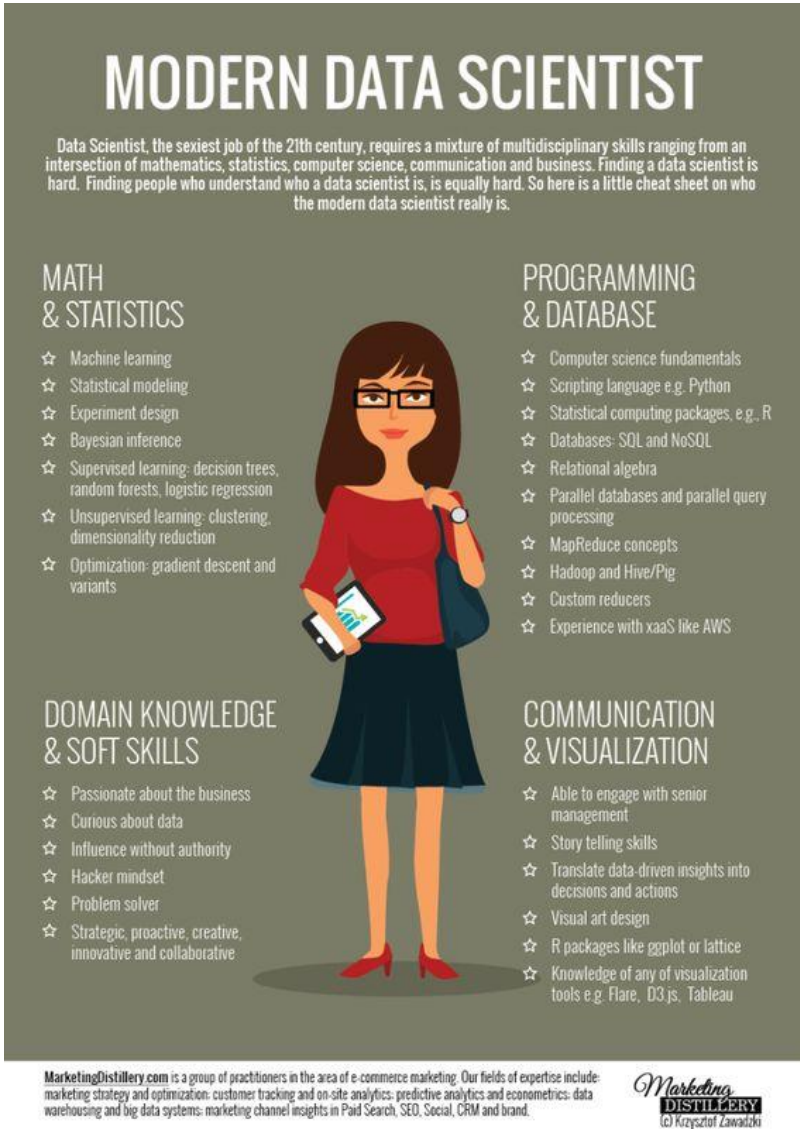
- Passionate about the business
- Curious about data
- Influence without authority
- Hacker mindset
- Problem solver
- Strategic, proactive, creative, innovative and collaborative

### COMMUNICATION & VISUALIZATION

- Able to engage with senior management
- Story telling skills
- Translate data-driven insights into decisions and actions
- Visual art design
- R packages like ggplot or lattice
- Knowledge of any of visualization tools e.g. Flare, D3.js, Tableau

MarketingDistillery.com is a group of practitioners in the area of e-commerce marketing. Our fields of expertise include: marketing strategy and optimization: customer tracking and on-site analytics: predictive analytics and econometrics: data warehousing and big data systems: marketing channel insights in Paid Search, SEO, Social, CRM and brand.

Marketing DISTILLERY



## MODERN DATA SCIENTIST

Data Scientist, the sexiest job of the 21th century, requires a mixture of multidisciplinary skills ranging from an intersection of mathematics, statistics, computer science, communication and business. Finding a data scientist is hard. Finding people who understand who a data scientist is, is equally hard. So here is a little cheat sheet on who the modern data scientist really is.

### MATH & STATISTICS

- Machine learning
- Statistical modeling
- Experiment design
- Bayesian inference
- Supervised learning: decision trees, random forests, logistic regression
- Unsupervised learning: clustering, dimensionality reduction
- Optimization: gradient descent and variants

### PROGRAMMING & DATABASE

- Computer science fundamentals
- Scripting language e.g. Python
- Statistical computing packages, e.g., R
- Databases: SQL and NoSQL
- Relational algebra
- Parallel databases and parallel query processing
- MapReduce concepts
- Hadoop and Hive/Pig
- Custom reducers
- Experience with xaaS like AWS

### DOMAIN KNOWLEDGE & SOFT SKILLS

- Passionate about the business
- Curious about data
- Influence without authority
- Hacker mindset
- Problem solver
- Strategic, proactive, creative, innovative and collaborative

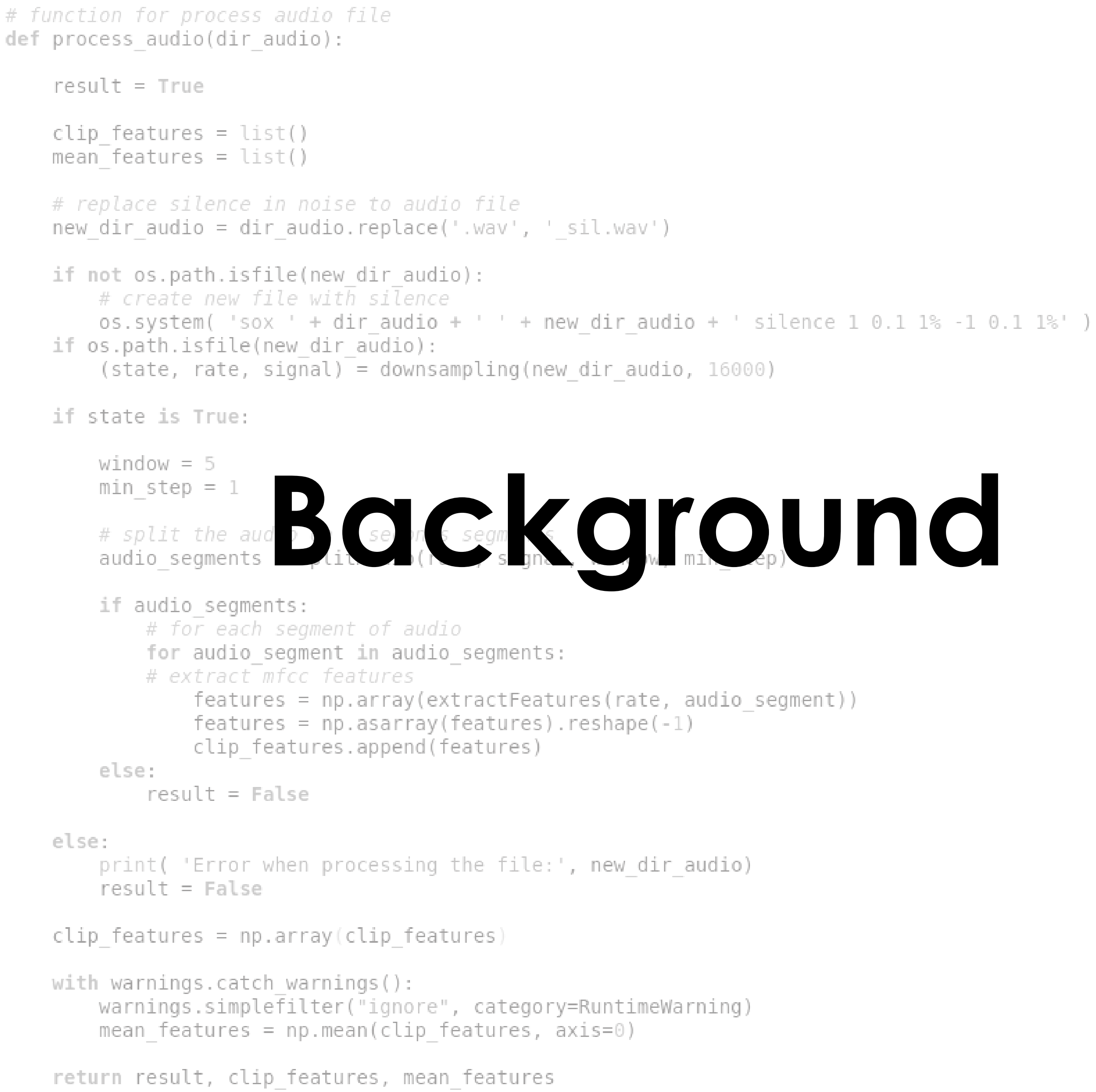
### COMMUNICATION & VISUALIZATION

- Able to engage with senior management
- Story telling skills
- Translate data-driven insights into decisions and actions
- Visual art design
- R packages like ggplot or lattice
- Knowledge of any of visualization tools e.g. Flare, D3.js, Tableau

MarketingDistillery.com is a group of practitioners in the area of e-commerce marketing. Our fields of expertise include: marketing strategy and optimization: customer tracking and on-site analytics: predictive analytics and econometrics: data warehousing and big data systems: marketing channel insights in Paid Search, SEO, Social, CRM and brand.

Marketing DISTILLERY
(c) Krzysztof Zawadzki

# Background



```
# function for process audio file
def process_audio(dir_audio):

    result = True

    clip_features = list()
    mean_features = list()

    # replace silence in noise to audio file
    new_dir_audio = dir_audio.replace('.wav', '_sil.wav')

    if not os.path.isfile(new_dir_audio):
        # create new file with silence
        os.system( 'sox ' + dir_audio + ' ' + new_dir_audio + ' silence 1 0.1 1% -1 0.1 1%' )
    if os.path.isfile(new_dir_audio):
        (state, rate, signal) = downsampling(new_dir_audio, 16000)

    if state is True:

        window = 5
        min_step = 1

        # split the audio in seconds segments
        audio_segments = split_audio(rate, signal, window, min_step)

        if audio_segments:
            # for each segment of audio
            for audio_segment in audio_segments:
            # extract mfcc features
                features = np.array(extractFeatures(rate, audio_segment))
                features = np.asarray(features).reshape(-1)
                clip_features.append(features)
        else:
            result = False

    else:
        print( 'Error when processing the file:', new_dir_audio)
        result = False

    clip_features = np.array(clip_features)

    with warnings.catch_warnings():
        warnings.simplefilter("ignore", category=RuntimeWarning)
        mean_features = np.mean(clip_features, axis=0)

    return result, clip_features, mean_features
```

# Motivation

Background

- Ornithology experts
- Difficult task of recognition
- The birds have regional accents
- Bird migration
- Unusual and endangered birds
- Colombia, second most biodiverse country in the world
- 1,903 bird species recorded in Colombia (2013)

Source: Revision of the status of bird species occurring or reported in Colombia 2013

# LifeClef challenge

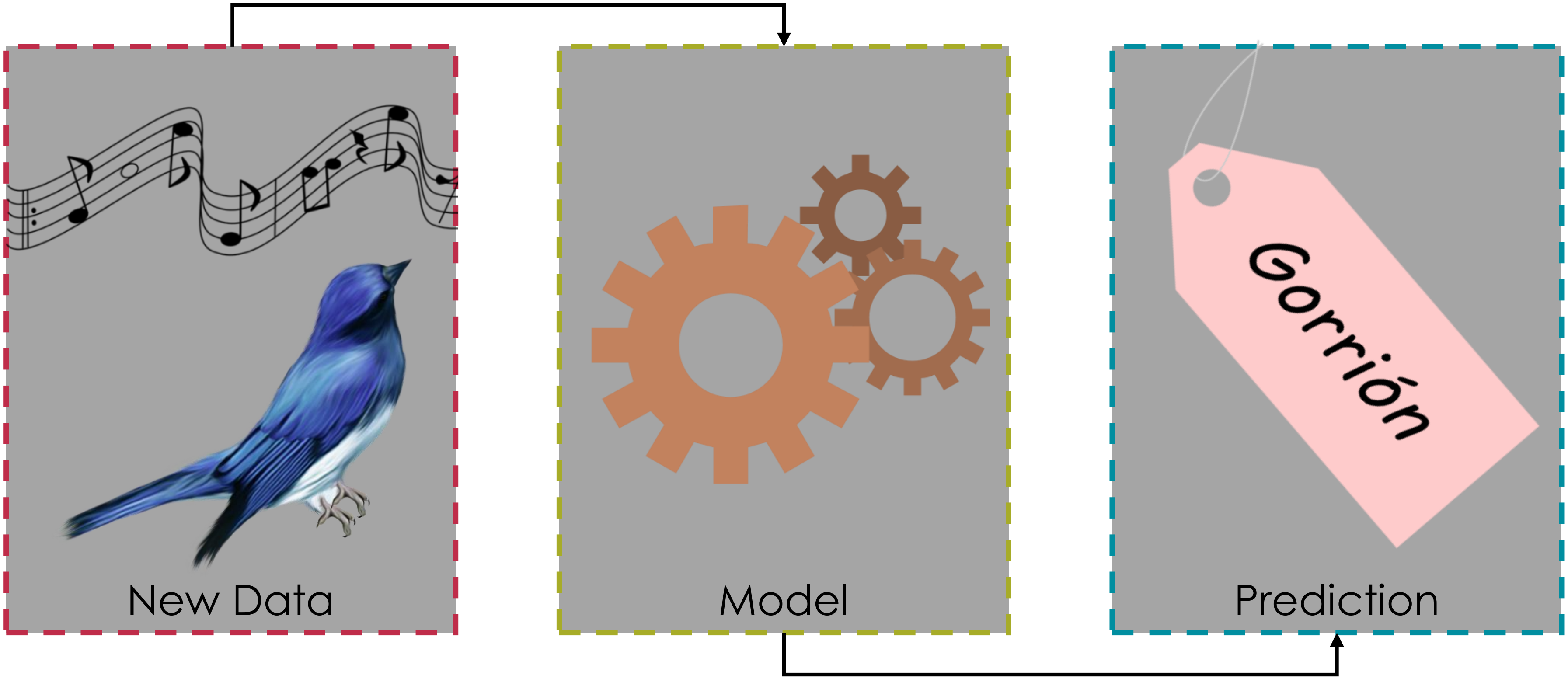
Background

Source: http://www.pbase.com/rsscanlon/image/110872861

# LifeClef challenge

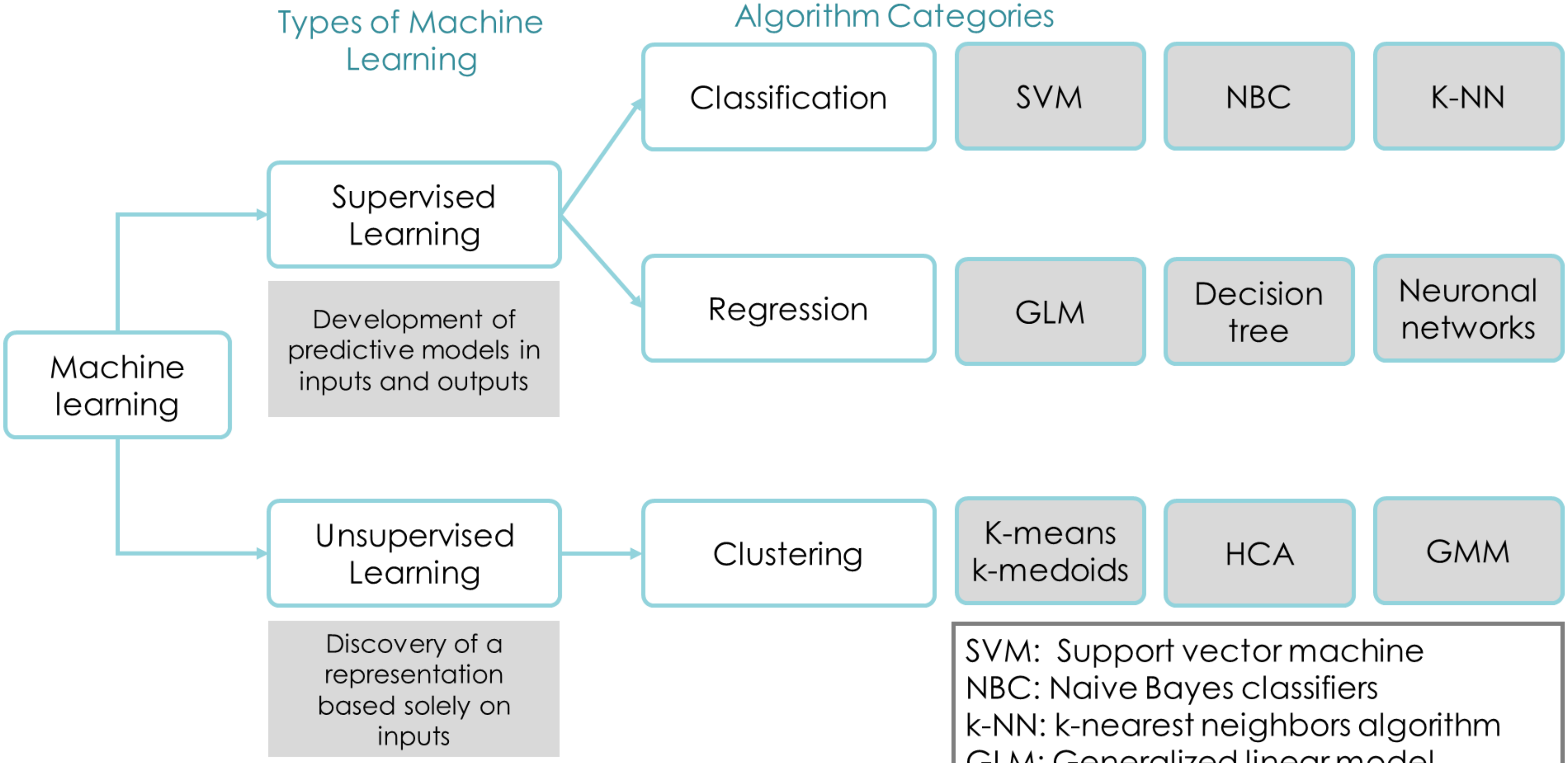
Background

The goal of the task is to identify all audio of birds from test recordings.



# The important things

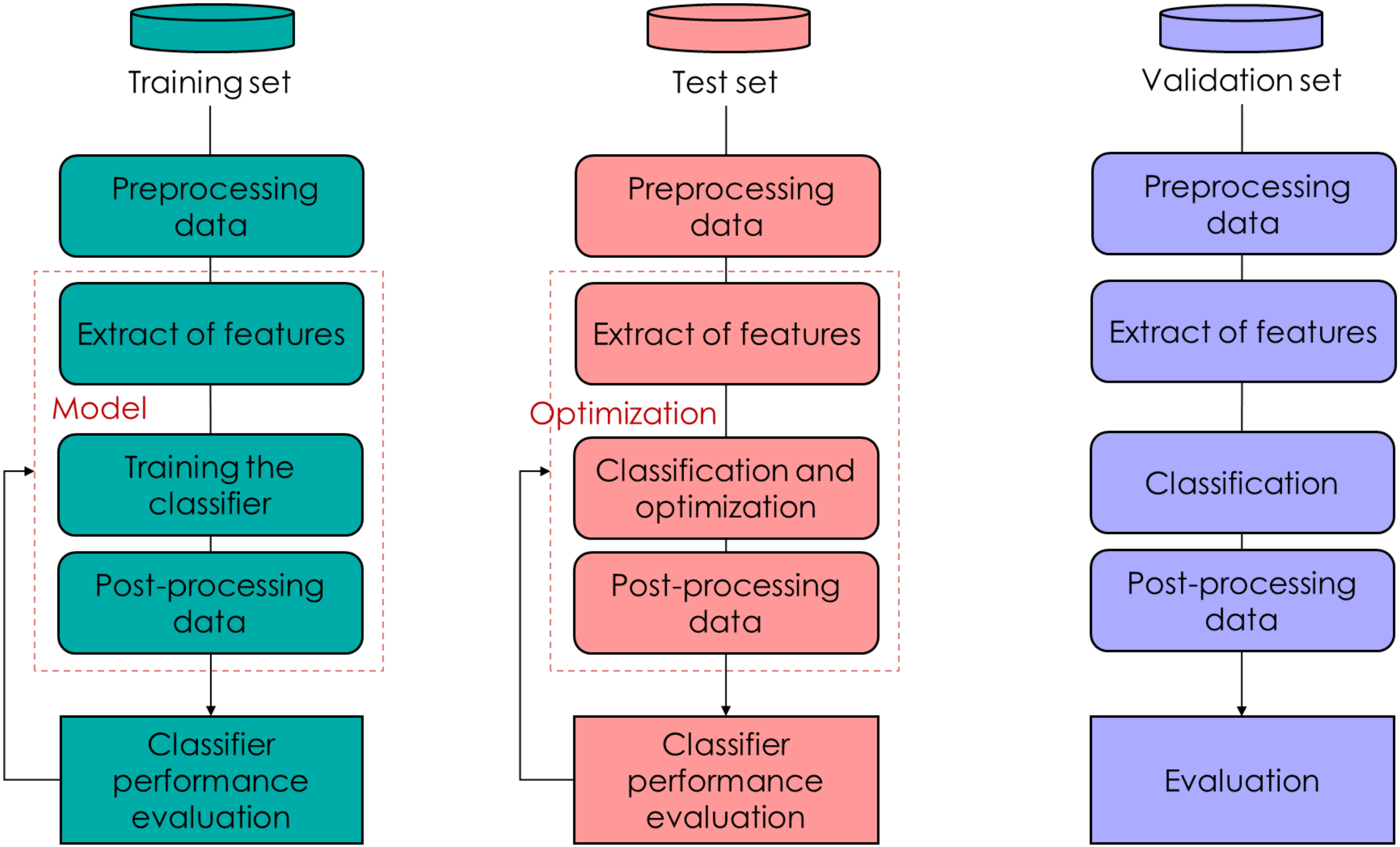
Background



SVM: Support vector machine
NBC: Naive Bayes classifiers
k-NN: k-nearest neighbors algorithm
GLM: Generalized linear model
HCA: Hierarchical cluster analysis
GMM: Modelo de mezcla Gaussiana

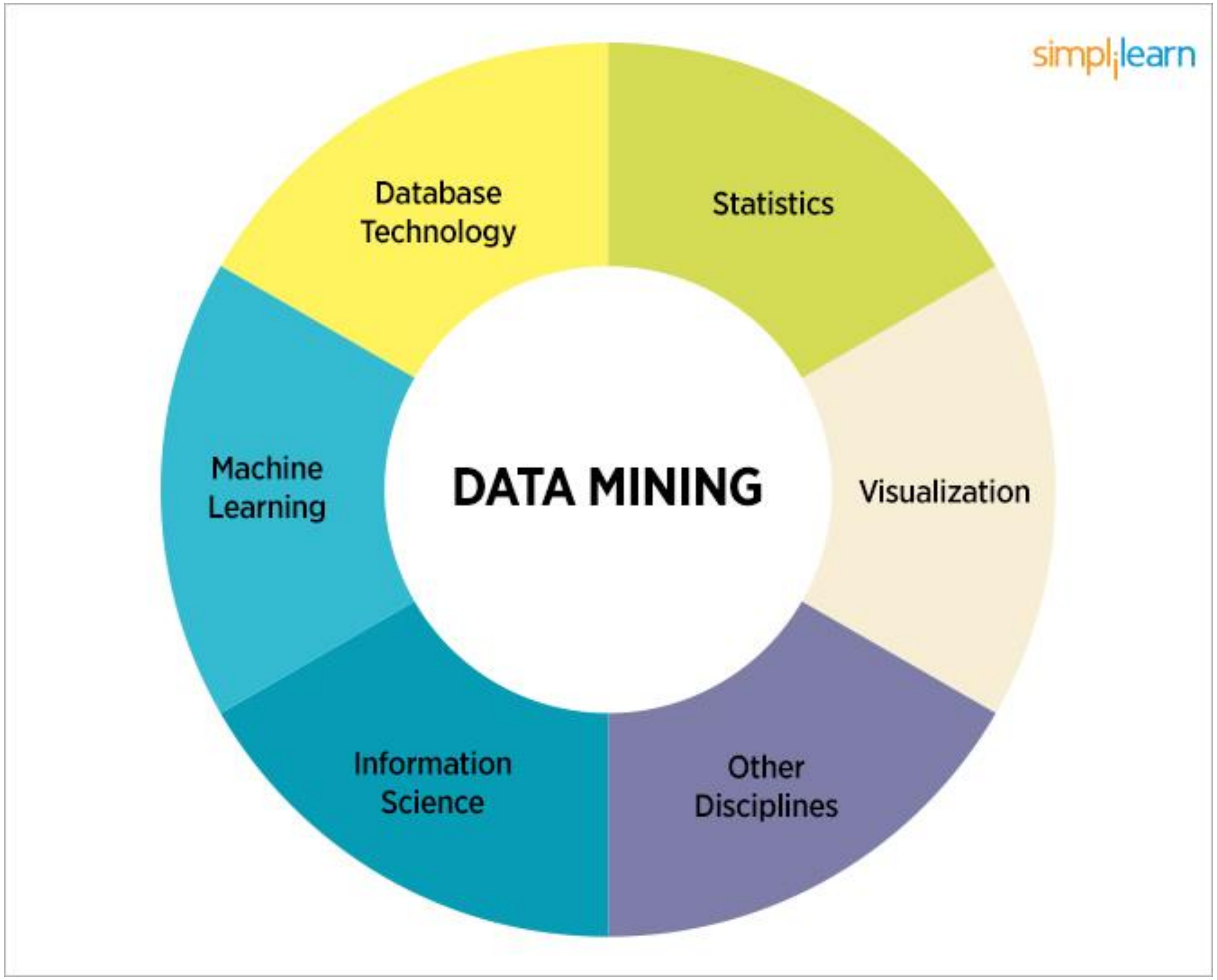
# Machine Learning

Background



# Data Mining

Background



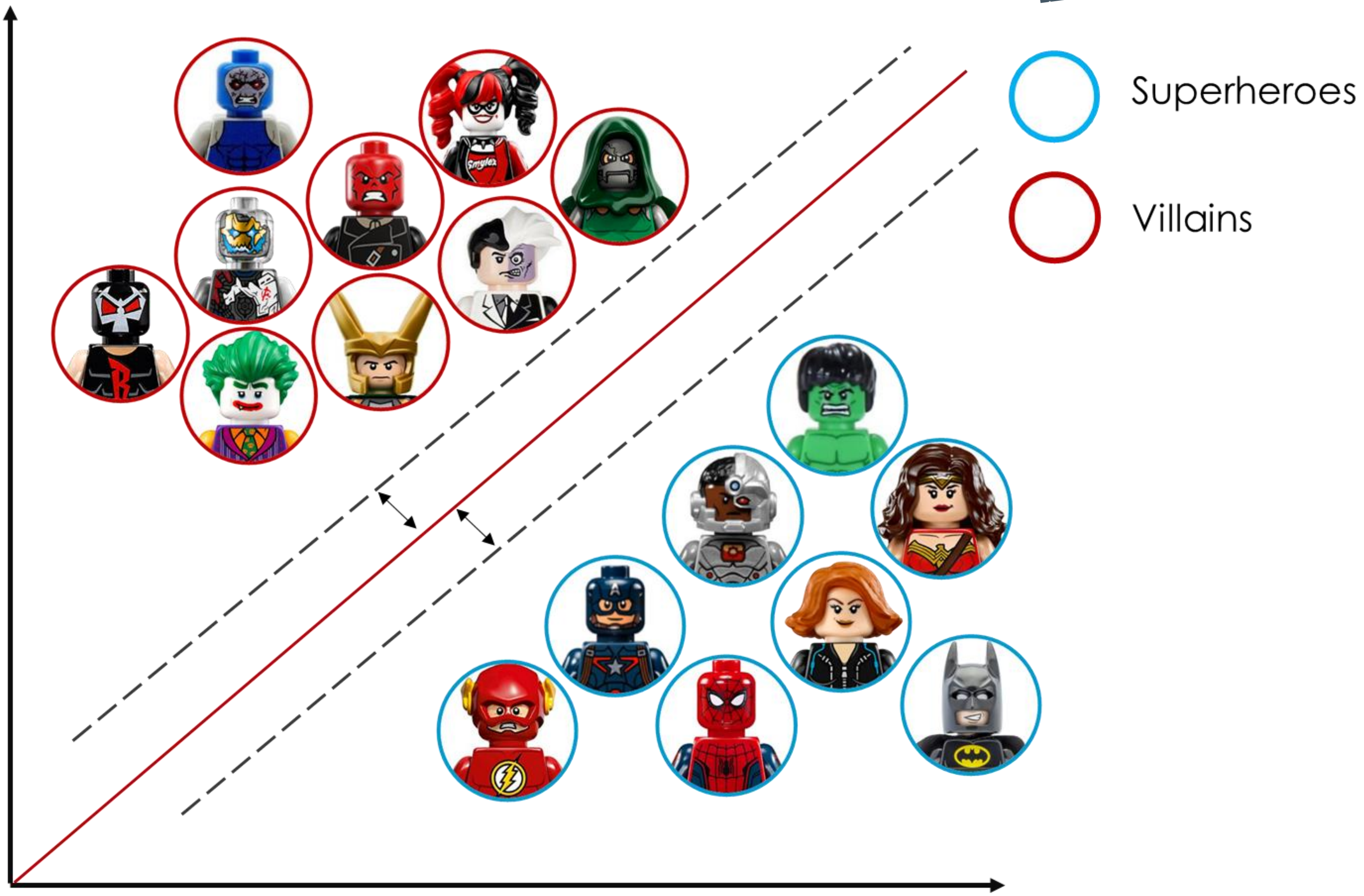
Source: https://www.simplilearn.com/data-mining-vs-statistics-article

# SVM (Support Vector Machines)

Background

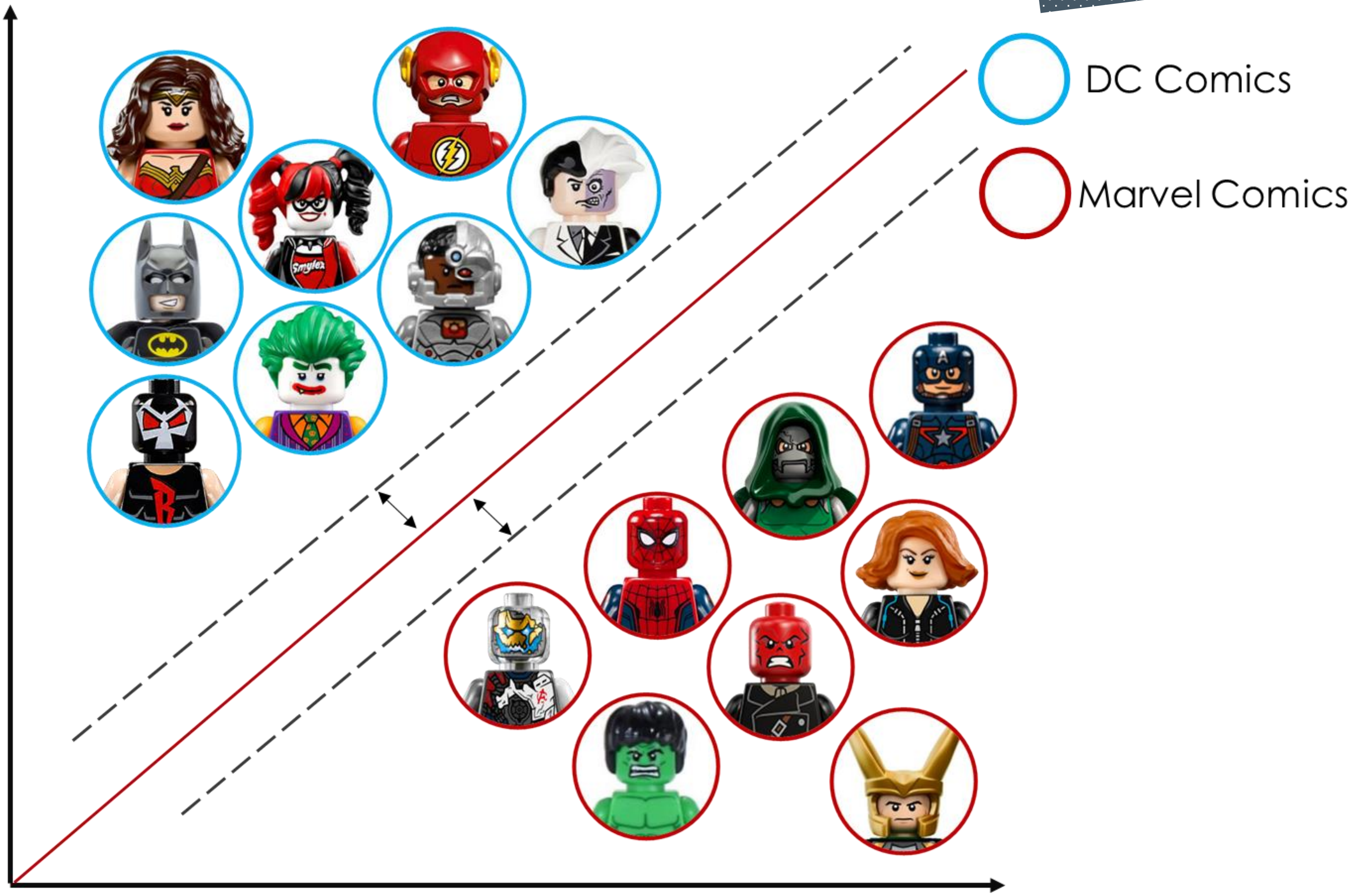
# SVM (Support Vector Machines)

Background



# SVM (Support Vector Machines)

Background

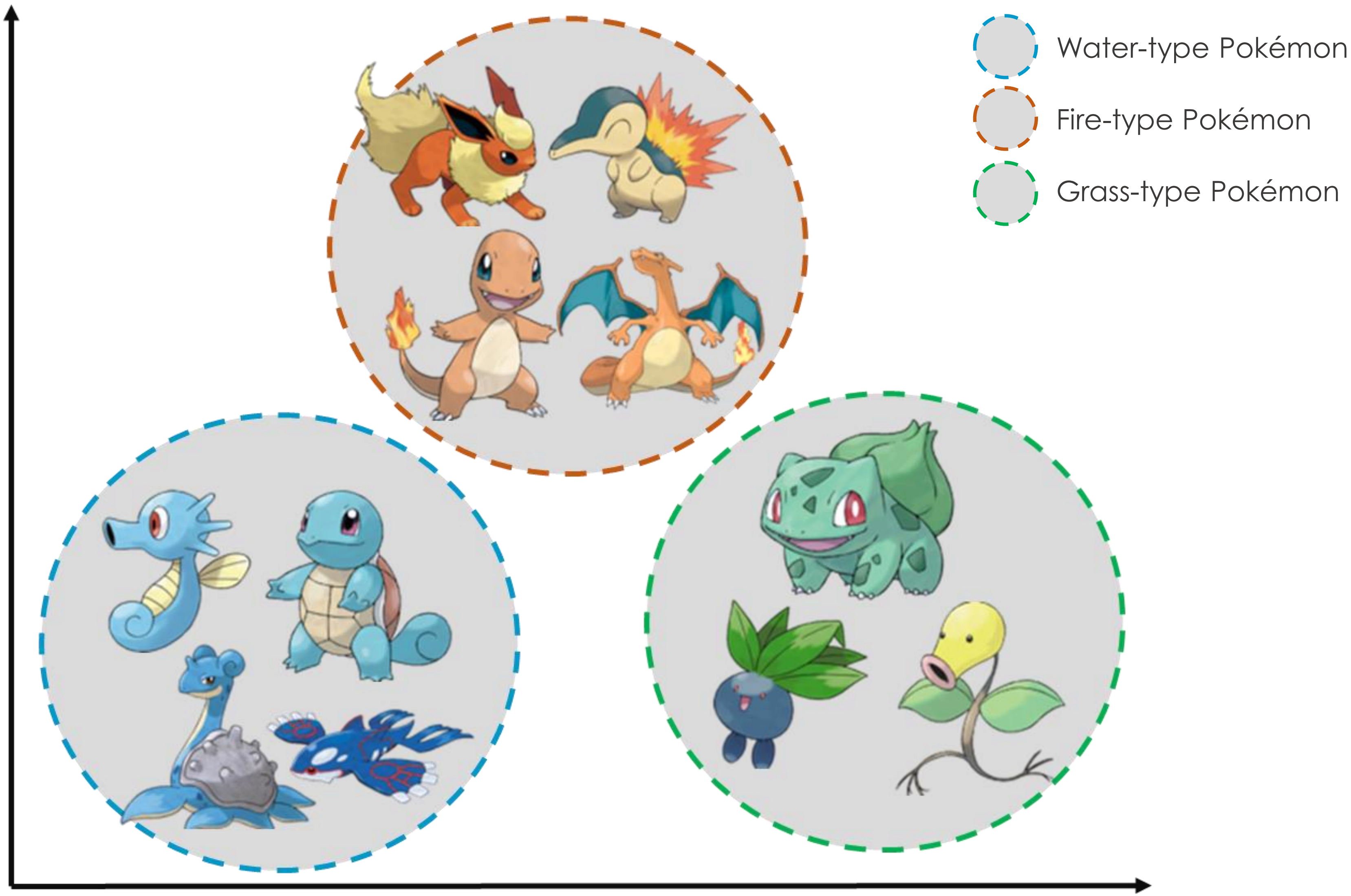


# Clustering (K-means)

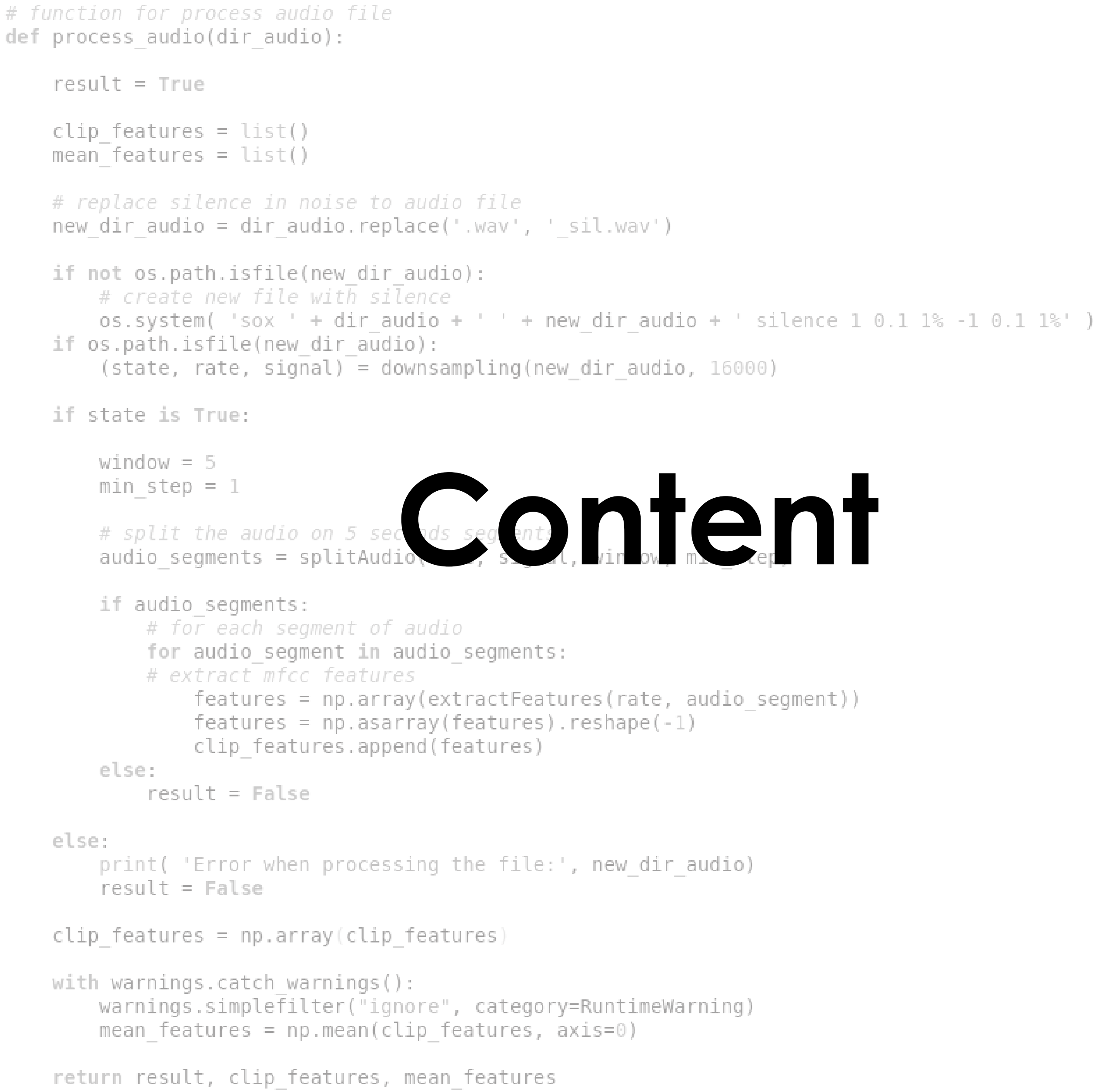
Background

# Clustering (K-means)

Background



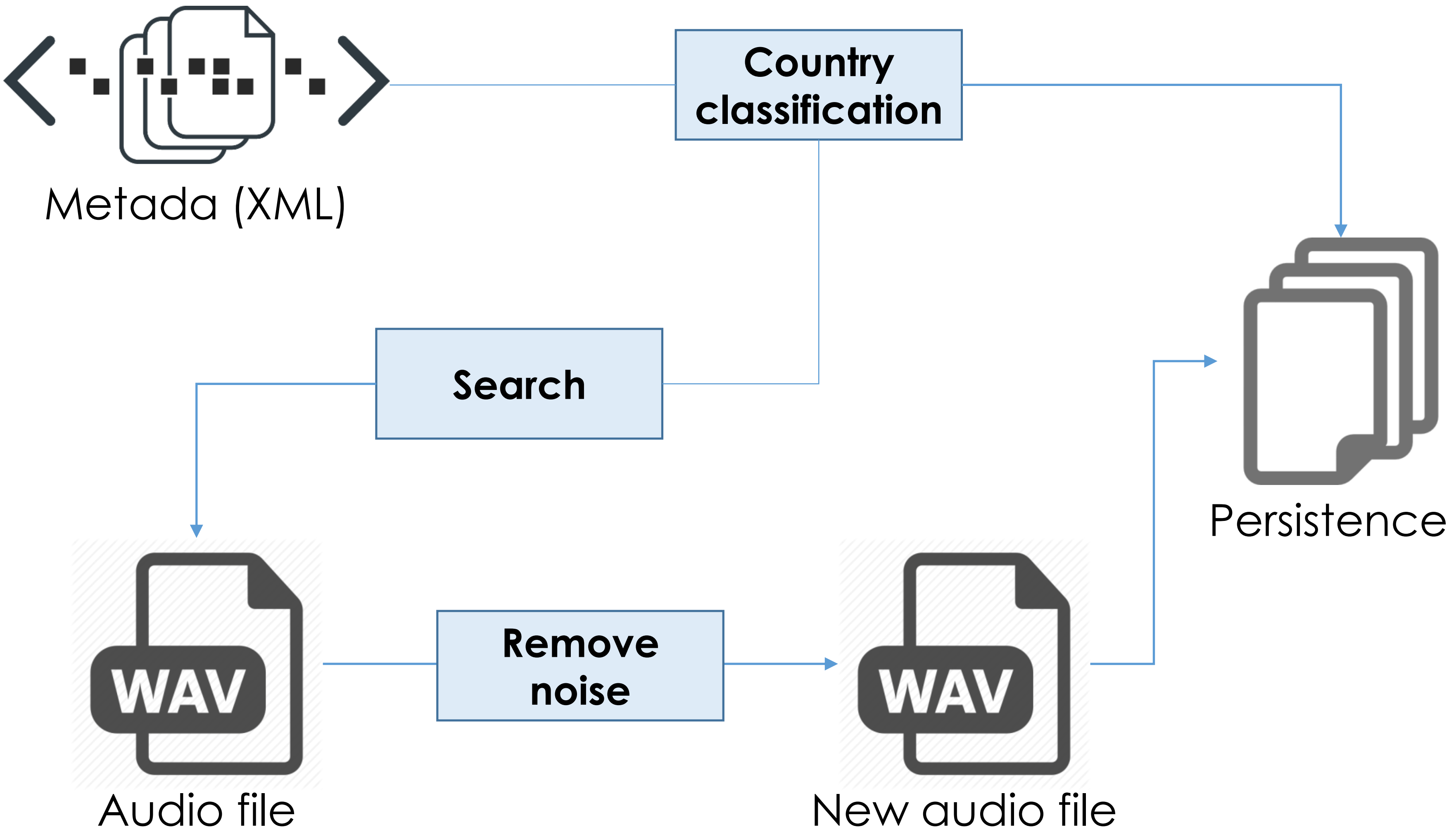
# Content



```
# function for process audio file
def process_audio(dir_audio):

    result = True

    clip_features = list()
    mean_features = list()

    # replace silence in noise to audio file
    new_dir_audio = dir_audio.replace('.wav', '_sil.wav')

    if not os.path.isfile(new_dir_audio):
        # create new file with silence
        os.system( 'sox ' + dir_audio + ' ' + new_dir_audio + ' silence 1 0.1 1% -1 0.1 1%' )
    if os.path.isfile(new_dir_audio):
        (state, rate, signal) = downsampling(new_dir_audio, 16000)

    if state is True:

        window = 5
        min_step = 1

        # split the audio on 5 seconds segments
        audio_segments = splitAudio(rate, signal, window, min_step)

        if audio_segments:
            # for each segment of audio
            for audio_segment in audio_segments:
            # extract mfcc features
                features = np.array(extractFeatures(rate, audio_segment))
                features = np.asarray(features).reshape(-1)
                clip_features.append(features)
        else:
            result = False

    else:
        print( 'Error when processing the file:', new_dir_audio)
        result = False

    clip_features = np.array(clip_features)

    with warnings.catch_warnings():
        warnings.simplefilter("ignore", category=RuntimeWarning)
        mean_features = np.mean(clip_features, axis=0)

    return result, clip_features, mean_features
```

# Dataset

Content



Metada (XML)

Country classification

Search

Persistence

Audio file

Remove noise

New audio file

# Dataset

Content

xeno-canto

**36,496 audio recordings**

**1,500 types of species**

|  |  |
|---|---|
| Otros | 3.638 |
| Ecuador | 3.908 |
| Peru | 2.853 |
| Brasil | 14.248 |
| Colombia | 7.860 |
| Suriname | 337 |
| Venezuela | 2.029 |
| Bolivia | 722 |
| Paraguay | 32 |
| French Guiana | 712 |
| Uruguay | 40 |
| Guyana | 116 |
| Argentina | 1 |

**7,860 audio recordings**

**789 types of species**

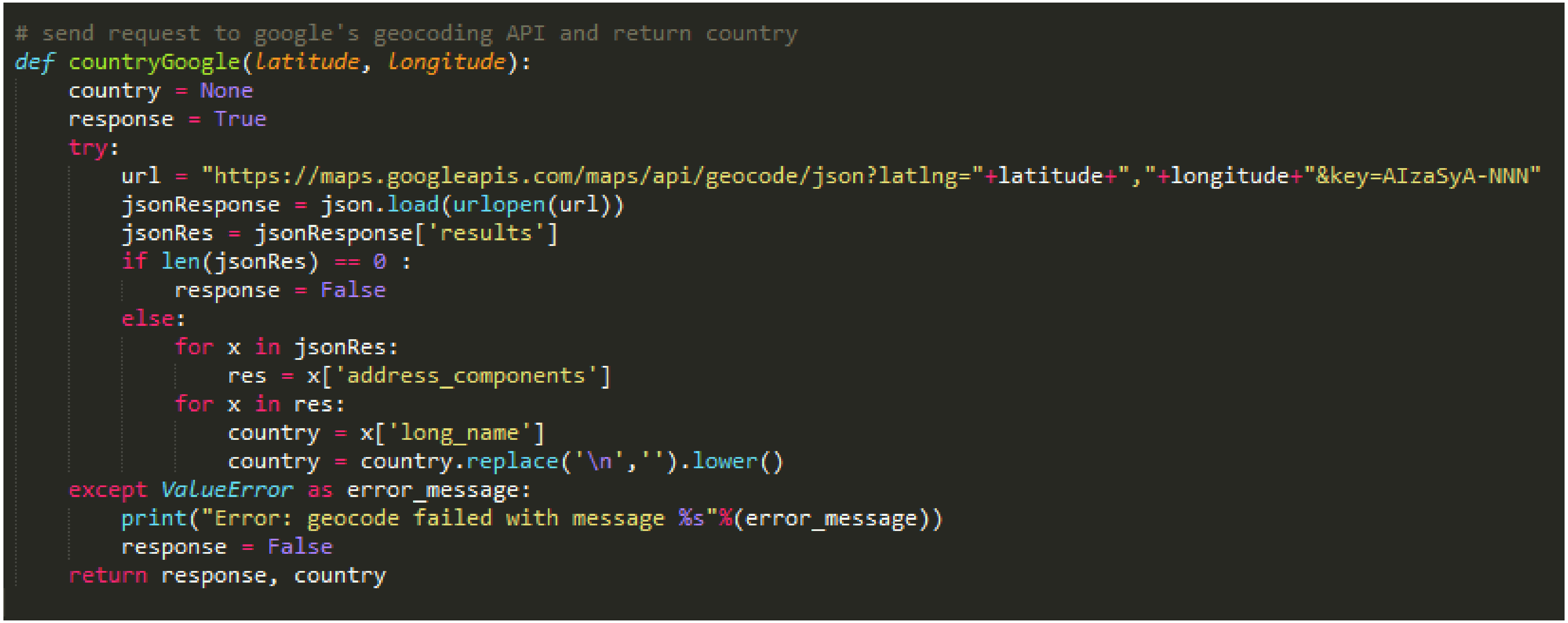
**3,440 audio recordings**

**100 types of specie**

Source https://www.xeno-canto.org/

# Processing data

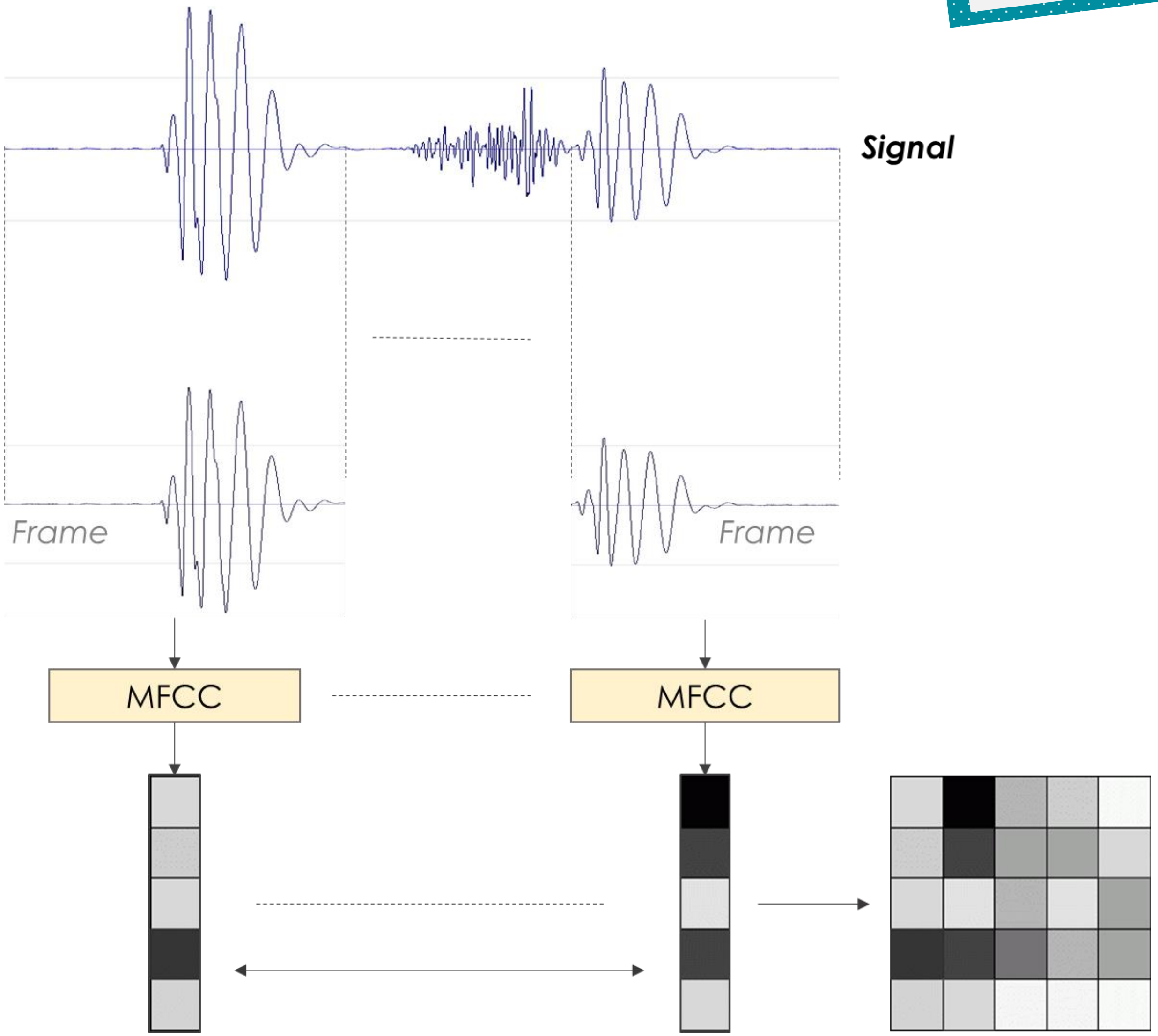
Content



```
# send request to google's geocoding API and return country
def countryGoogle(latitude, longitude):
    country = None
    response = True
    try:
        url = "https://maps.googleapis.com/maps/api/geocode/json?latlng="+latitude+","+longitude+"&key=AIzaSyA-NNN"
        jsonResponse = json.load(urlopen(url))
        jsonRes = jsonResponse['results']
        if len(jsonRes) == 0 :
            response = False
        else:
            for x in jsonRes:
                res = x['address_components']
            for x in res:
                country = x['long_name']
                country = country.replace('\n','').lower()
    except ValueError as error_message:
        print("Error: geocode failed with message %s"%(error_message))
        response = False
    return response, country
```

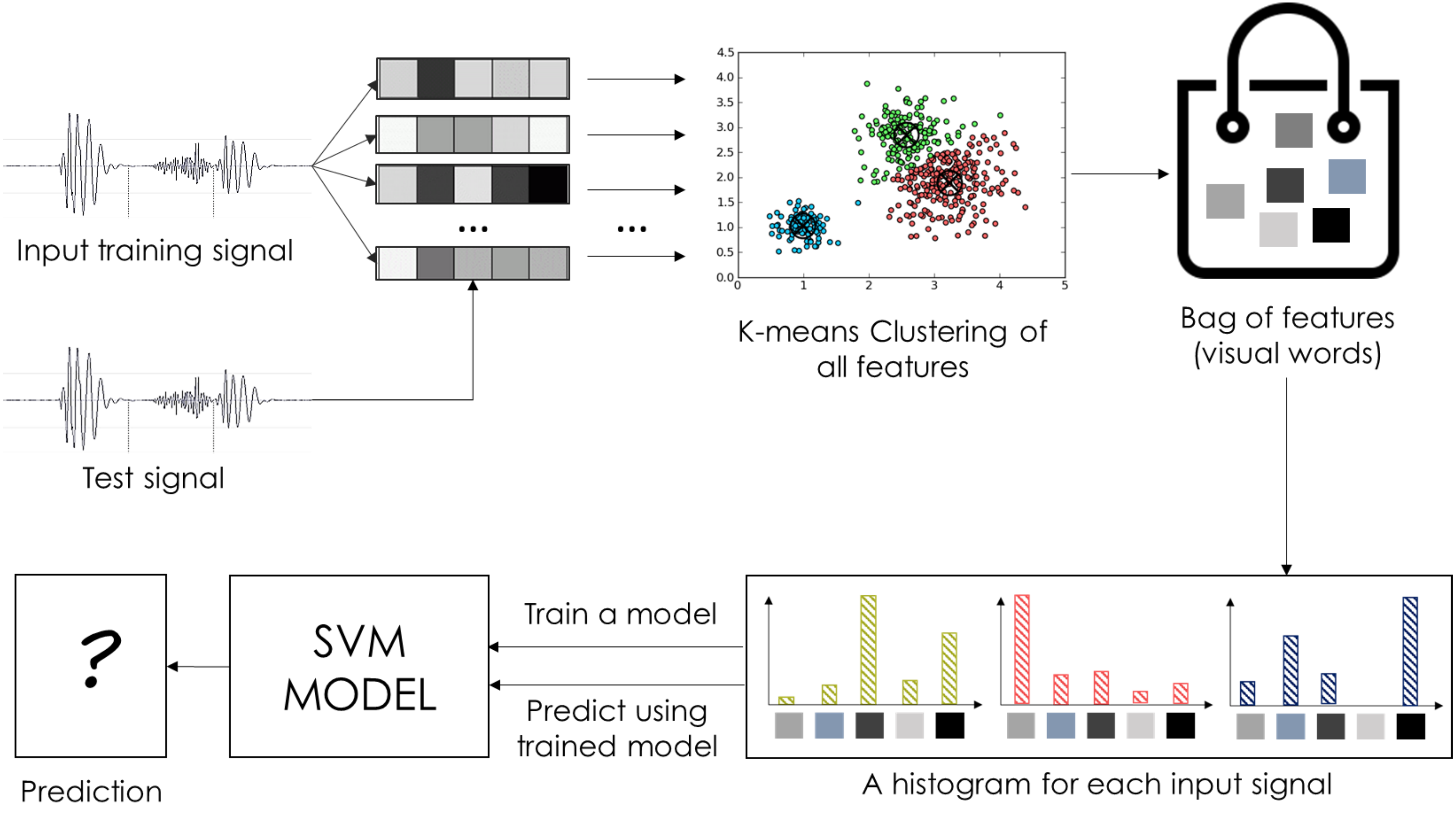
# Extract features

Content



# Classification

Content



Input training signal

Test signal

K-means Clustering of all features

Bag of features (visual words)

A histogram for each input signal

Train a model

Predict using trained model

SVM MODEL

?

Prediction

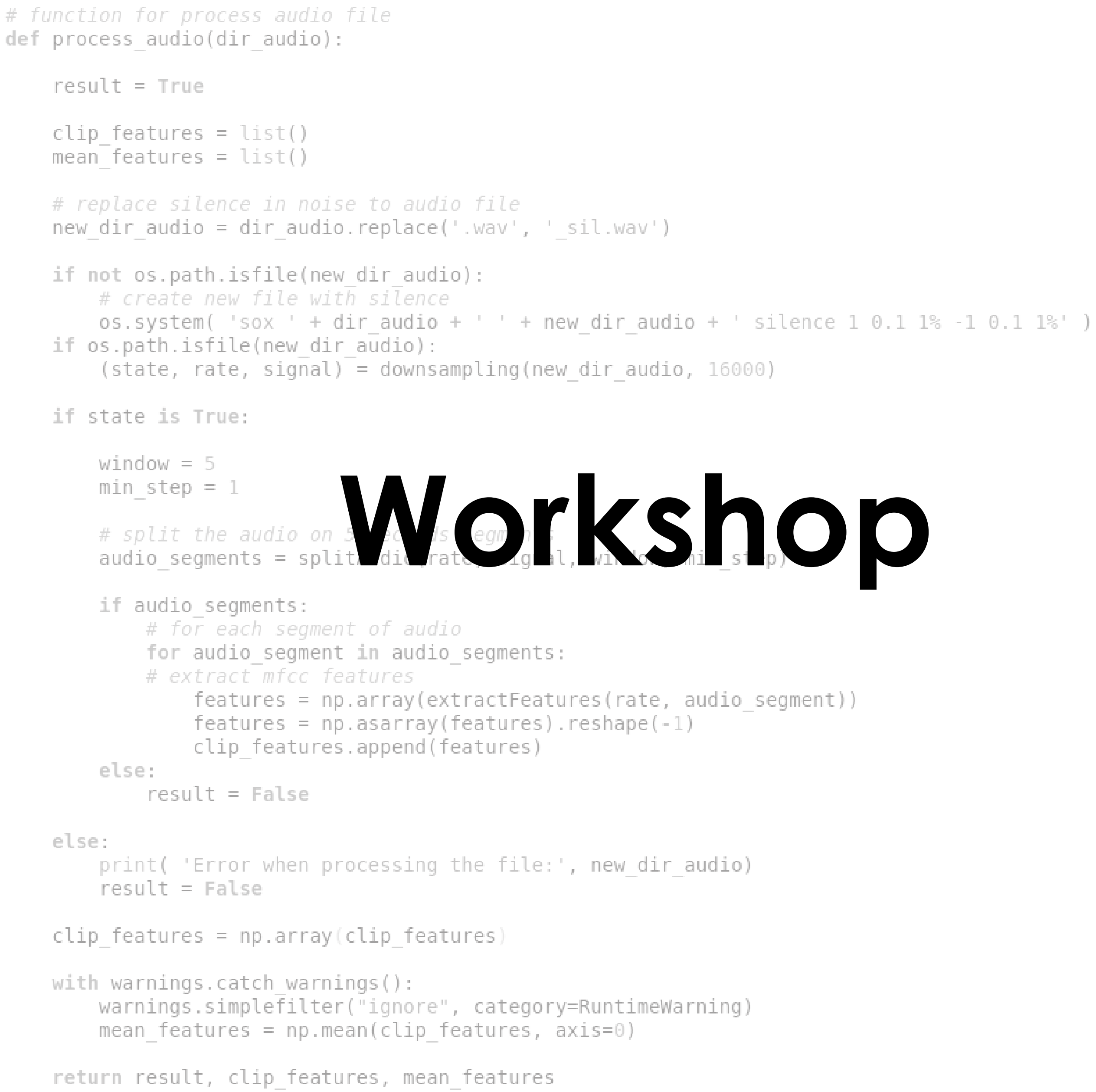
# Results

**Global features**

**vs**

**Bag of features**

# Workshop



```
# function for process audio file
def process_audio(dir_audio):

    result = True

    clip_features = list()
    mean_features = list()

    # replace silence in noise to audio file
    new_dir_audio = dir_audio.replace('.wav', '_sil.wav')

    if not os.path.isfile(new_dir_audio):
        # create new file with silence
        os.system( 'sox ' + dir_audio + ' ' + new_dir_audio + ' silence 1 0.1 1% -1 0.1 1%' )
    if os.path.isfile(new_dir_audio):
        (state, rate, signal) = downsampling(new_dir_audio, 16000)

    if state is True:

        window = 5
        min_step = 1

        # split the audio on 5 seconds segments
        audio_segments = splitAudio(rate, signal, window, min_step)

        if audio_segments:
            # for each segment of audio
            for audio_segment in audio_segments:
            # extract mfcc features
                features = np.array(extractFeatures(rate, audio_segment))
                features = np.asarray(features).reshape(-1)
                clip_features.append(features)
        else:
            result = False

    else:
        print( 'Error when processing the file:', new_dir_audio)
        result = False

    clip_features = np.array(clip_features)

    with warnings.catch_warnings():
        warnings.simplefilter("ignore", category=RuntimeWarning)
        mean_features = np.mean(clip_features, axis=0)

    return result, clip_features, mean_features
```

# Python & Notebook

Workshop

# Practical exercise…

Workshop

## Repository

https://github.com/angiereyesbet/birdPycon2018

## Dataset

13.58.110.45/data/data.tar.gz

(Temporary URL)

Thank You!

angreyes@outlook.com
angreyes@uan.edu.co
angiereyes.bet@gmail.com