



# WIKISTONE

**SOUTENANCE DE PROJET - UE INTEROPÉRABILITÉ**

BENGUEZZOU - BOUABID - BOUGASSAA - MEZIANE

GROUPE 4

# WIKISTONE

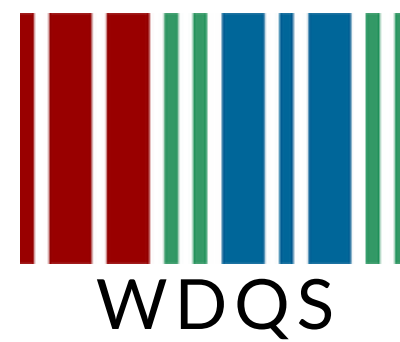
C'EST QUOI ?

UN SYSTÈME PERMETTANT DE RÉPÉRTORIER DES  
DONNÉES PATRIMONIALES



# WIKISTONE

UNE BASE DE CONNAISSANCES INTEROPÉRABLE



## ENDPOINT

<http://localhost:8834/proxy/wdqs/bigdata/namespace/wdq/sparql>

<http://localhost/w/api.php>



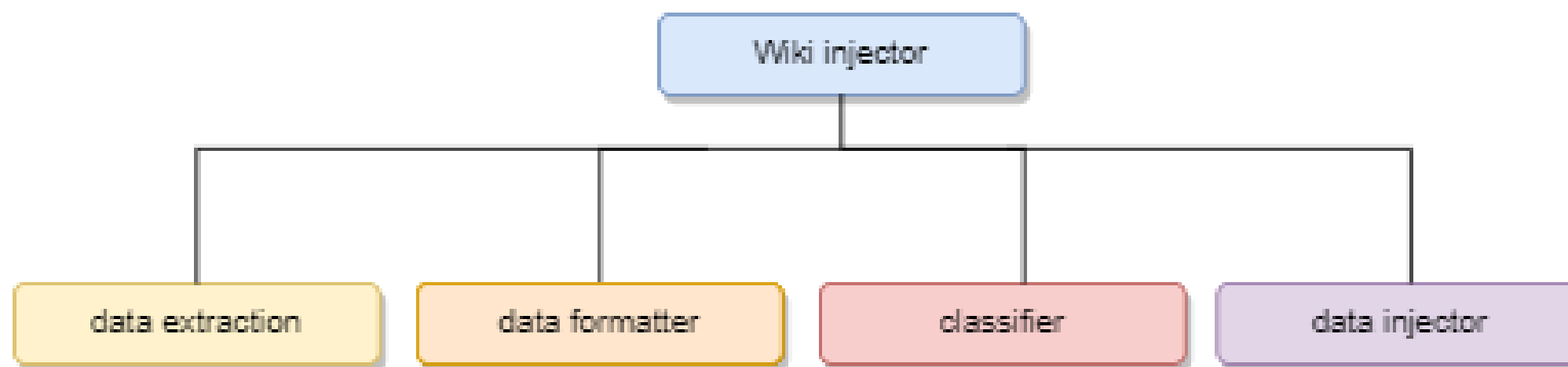
- WIKI-INTERFACE
- WIKI-INJECTOR

# WIKI INJECTOR

UNE APPLICATION POUR L'INGESTION DE DONNÉES



<https://github.com/samuelseuli/python-wikibase/>



# WIKISTONE

UNE APPLICATION POUR L'INGESTION DE DONNÉES

## DATA EXTRACTION



<https://data.culture.gouv.fr/>



REQUÊTE - GET



<https://www.loire.fr/>



Web Scraping

CONNEXION ET ANALYSE  
DE LA PAGE WEB



<https://datacliv.fr/>



TÉLÉCHARGEMENT ET  
LECTURE DU FICHIER

# WIKISTONE

UNE APPLICATION POUR L'INGESTION DE DONNÉES

## DATA FORMATTER

Creation d'un dictionnaire de propriétés

```
PROP_REGION = {
    LABEL: "region",
    TYPE: TYPE_STRING,
    DESCRIPTION: "La Région est la plus récente des collectivités territoriales, ...",
}
```

```
properties = {
    PROP_NAME[LABEL]: PROP_NAME,
    PROP_ADDRESS[LABEL]: PROP_ADDRESS,
    PROP_CITY[LABEL]: PROP_CITY,
    PROP_POSTCODE[LABEL]: PROP_POSTCODE,
    PROP_DEPARTEMENT[LABEL]: PROP_DEPARTEMENT,
    ...
    PROP_CATEGORY[LABEL]: PROP_CATEGORY
}
```

Creation d'un dictionnaire de liaison

```
# A mapping dictionary
mapping = {
    PROP_REGION[LABEL]: PROP_REGION[LABEL],
    PROP_CITY[LABEL]: "commune_1",
    ...
    PROP_DEPARTEMENT[LABEL]: PROP_DEPARTEMENT[LABEL],
    PROP_NAME[LABEL]: "appellation_courante",
}
```

Extraction des données

```
monuments = []

for d in data:
    monument = {}
    # Retrieve the data using the mapping dictionary
    for key, value in mapping.items():
        monument[key] = d[value]

    monuments.append(monument)
```

# WIKISTONE

UNE APPLICATION POUR L'INGESTION DE DONNÉES

DATA CLASSIFICATION



**! Absence des catégories de monuments !**



PREDICTION DES CATÉGORIES À L'AIDE DU  
MACHINE LEARNING

**Labélisation des données**

**Mise en place d'un modèle de prédiction**

# WIKISTONE

UNE APPLICATION POUR L'INGESTION DE DONNÉES

## Labelisation des données - Clustering

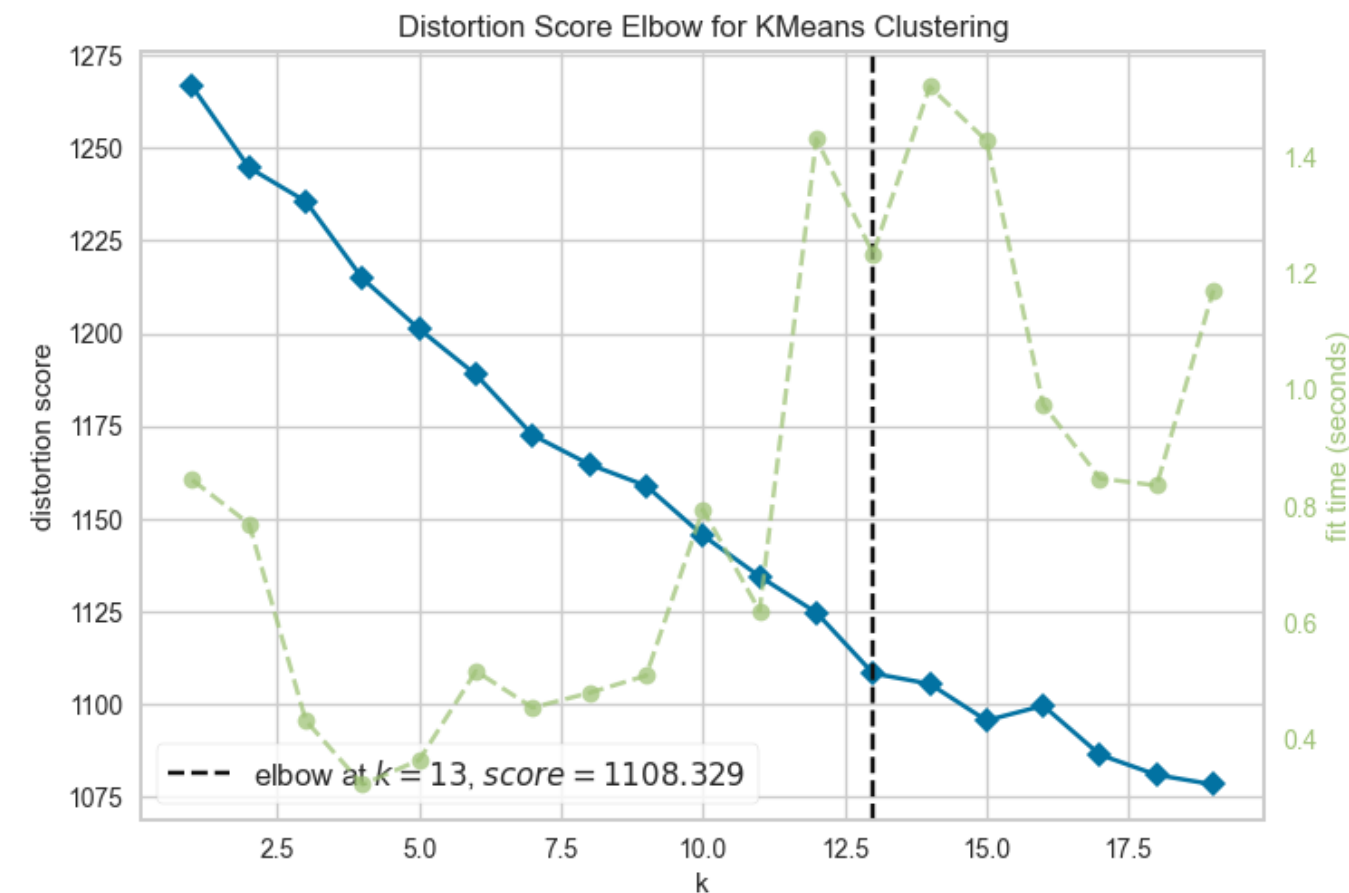
IMPORTATION D'UN GRAND DATASET



**+5000 MONUMENTS**



MÉTHODE DU COUDE : LE NOMBRE DE CLUSTERS OPTIMAL

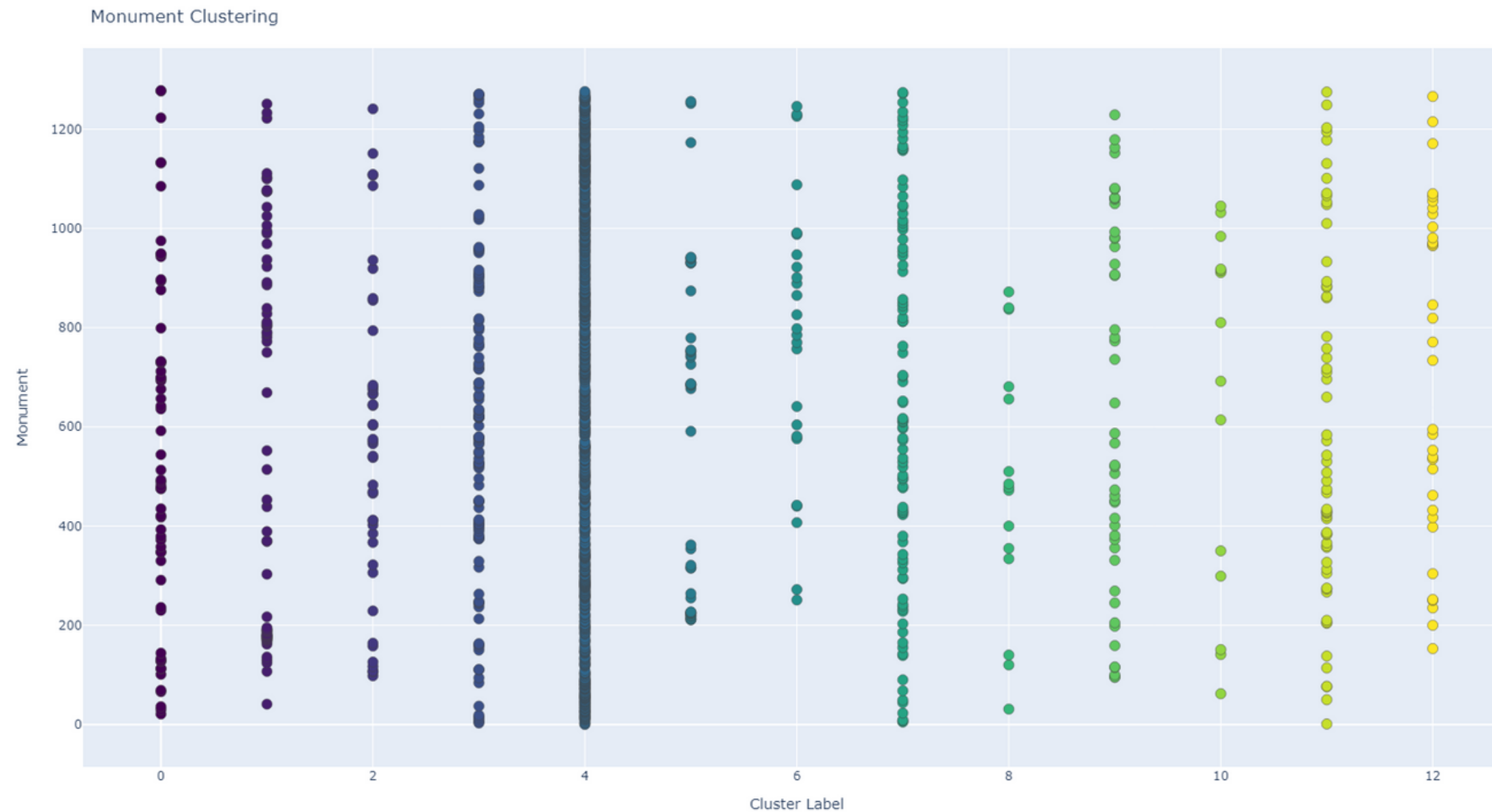




# WIKISTONE

UNE APPLICATION POUR L'INGESTION DE DONNÉES

## DONNÉES CLUSTERISÉES



**RELANCEMENT DU PROCESSUS SUR LES DONNÉES MAL  
LABÉLISÉES**

## RÉSULTATS

CHÂTEAU, EGLISE, MAISON, CROIX, CHAPELLE, IMMEUBLE, MONUMENT AUX MORTS  
HÔTEL, FONTAINE, PONT, MUSÉE, FERME, AUTRE

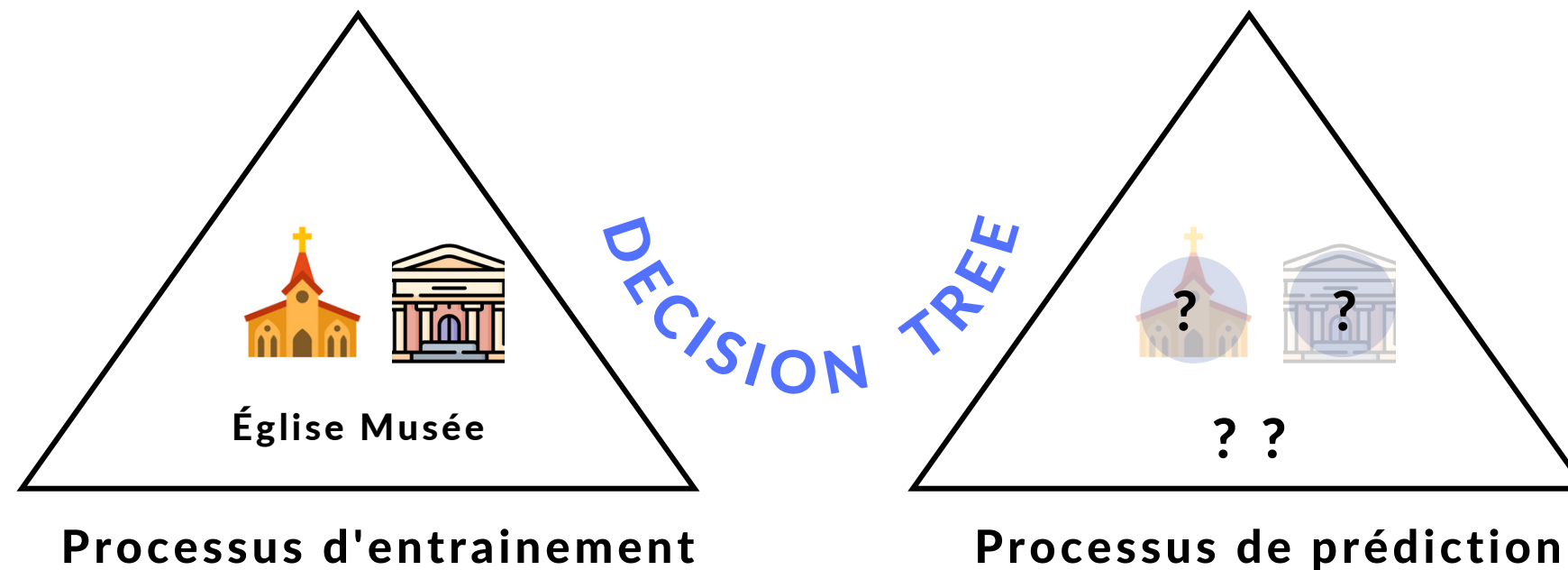
# WIKISTONE

UNE APPLICATION POUR L'INGESTION DE DONNÉES

## Recherche du modèle de prédiction optimal

K-NEAREST NEIGHBORS, DECISION TREE, MULTINOMIAL NB,  
GAUSSIAN SVM, POLYNOMIAL SVM

## Application du modèle de prédiction

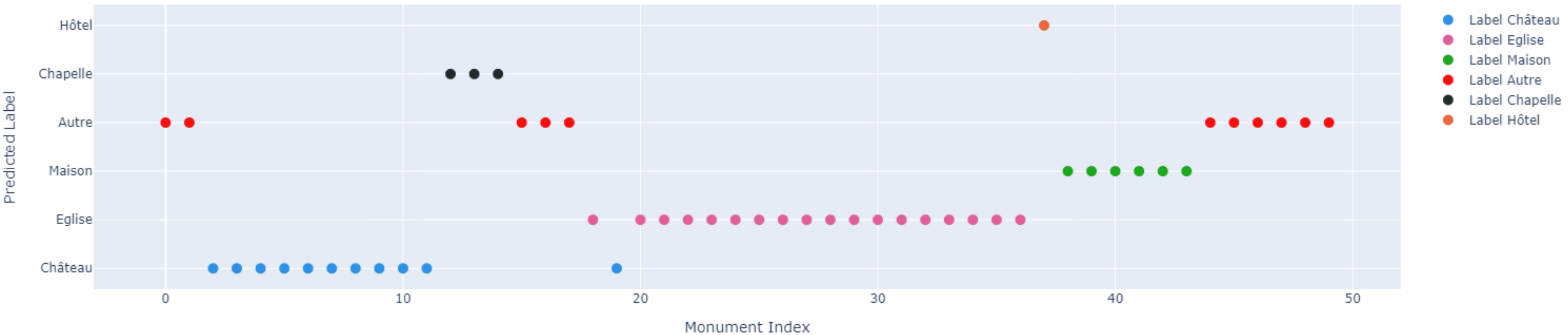


# WIKISTONE

UNE APPLICATION POUR L'INGESTION DE DONNÉES

## EXEMPLE DE RÉSULTATS ISSUS DU WEB SCRAPPING

Monument Labels



### WIKI-INJECTOR EN RÉSUMÉ :

1. EXTRACTION
2. MISE EN FORME
3. CLASSIFICATION
4. INJECTION

# WIKISTONE

WIKI ADMIN 

UNE APPLICATION POUR L'ADMINISTRATION, POUR FACILITER  
L'INGESTION DES DONNÉES



## Utilisation de EEL

```
● ● ●  
  
@eel.expose  
def process_csv_data():  
    process_csv_data_classify(py_wb)
```

PYTHON

```
● ● ●  
  
async function getDataFromPython(buttonId) {  
    if (buttonId.includes('json')) {  
        await eel.process_api_data()();  
    }  
    else if (buttonId.includes('csv')) {  
        await eel.process_csv_data()();  
    }  
    else if (buttonId.includes('web')) {  
        await eel.process_web_data()();  
    }  
}
```

JAVASCRIPT

# WIKISTONE

UNE APPLICATION POUR LA RECHERCHE ET LA  
CONSULTATION DE DONNÉES

## WIKI INTERFACE



- INTERROGATION DE L'UTILISATEUR.
- RÉCUPÉRATION, MISE EN FORME ET AFFICHAGE DES RÉSULTATS.



AFFICHAGE ET GESTION D'UNE CARTE



mapquest

CONVERSION D'ENTRÉES TEXTUELLES EN DONNÉES  
GÉOGRAPHIQUES



# DÉMONSTRATION

---

