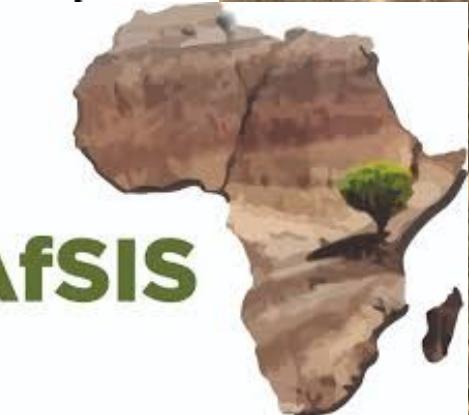
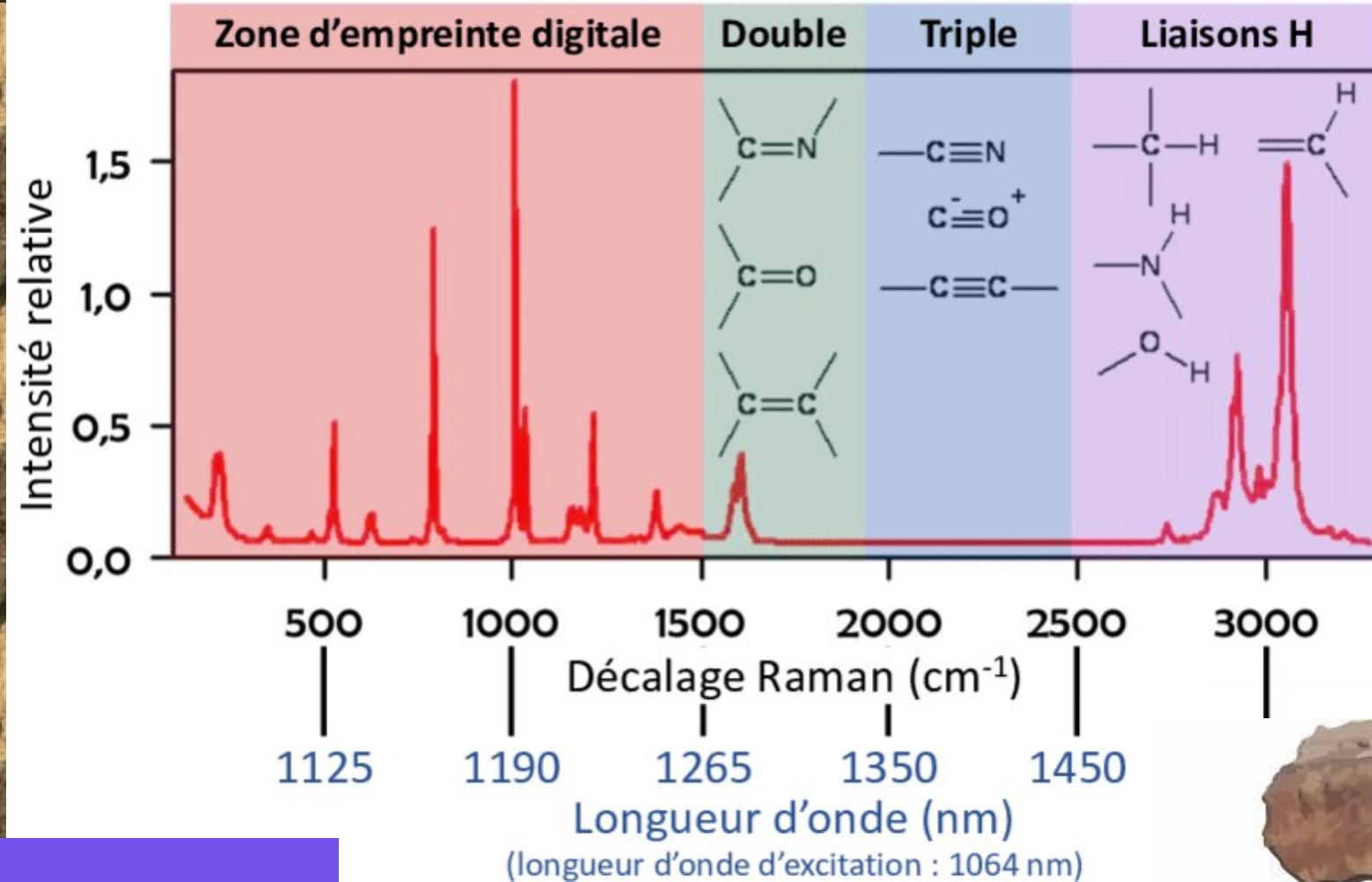




Preuve de concept: Prédiction de la fertilité du sol par machine learning



Outline

- **Introduction**
 - Objectif de l'étude
 - Jeux de données
- **Analyse exploratoire**
 - Les données AFSiS
 - Analyses de composition chimique
 - Sélection des spectres infrarouges
 - Distribution géographique
- **Modélisation**
 - Modèle de départ
 - Pistes de modélisation
 - Corrélation entre mesures et fertilité du sol
 - Corrélation entre spectroscopie et chromatographie
 - Régression multivariée par les moindres carrés partiels (PLS)
- **Conclusions**
 - Résultat et possible amélioration

Modalités de la soutenance

5 min - Présentation de la problématique, de son interprétation et des pistes de recherche envisagées.

5 min - Présentation de l'exploration.

10 min - Présentation des différentes pistes de modélisation effectuées.

5 min - Présentation du modèle final sélectionné et résultats.

5 à 10 minutes de questions-réponses.

Objectif du modèle

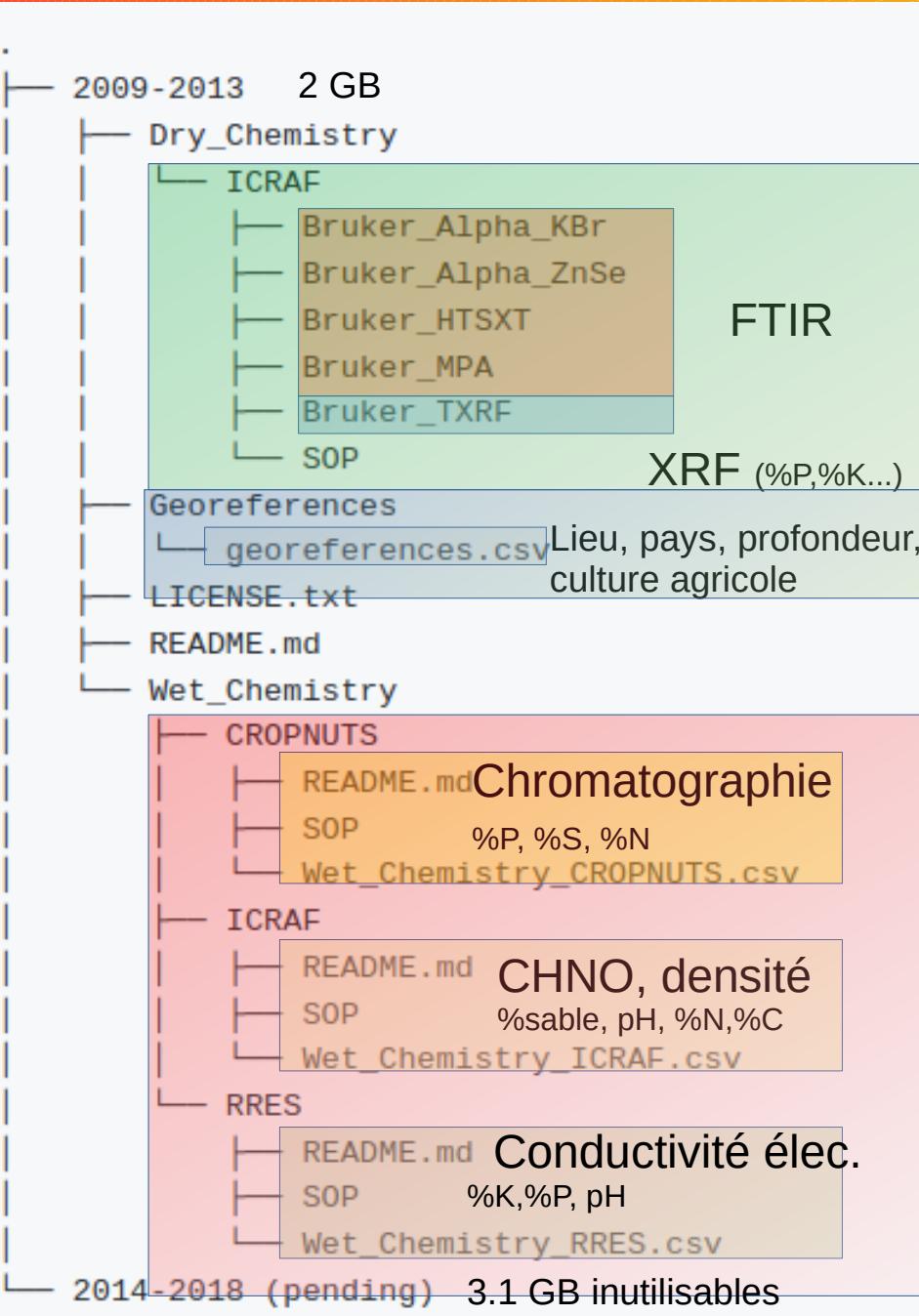
Dans cette étude avec un apprentissage supervisé, on essaye de trouver le meilleur modèle pour

- Prédire la fertilité du sol à partir des mesures de concentration rapides et économiques
- Prédire la composition du sol à partir des données des spectroscopie infrarouge

Exploration des données

- Structure des données
- Nettoyage : imputation celles vides, outliers

Structure des données



Facile, rapide
économique

Plusieurs laboratoires



Spectres infrarouges
mesurés avec
instruments différents
pour chaque
échantillon

Longue, chère



Propriétés chimiques
mesurées que pour
une minorité des
échantillons avec
techniques différentes
dans chaque
laboratoire

Distribution géographique



Procédure de modélisation : composition et fertilité

	SSN	M3 Ca	M3 K	M3 Al	M3 P	M3 S	PH
0	icr006475	207.1	306.30	1095.0	4.495	18.960	4.682
1	icr006586	1665.0	1186.00	1165.0	12.510	13.600	7.062
2	icr007929	2518.0	72.57	727.6	21.090	14.810	7.114
3	icr008008	734.3	274.60	1458.0	109.200	11.400	5.650
4	icr010198	261.8	91.76	2166.0	3.958	5.281	5.501

Geo

Lab 1 → CSV1

Lab 2 → CSV2

Lab 3 → CSV3

Intégration des données composition chimique du sol et géoreferences
(inner joins sur ID échantillon)

Nettoyage des données

Sélection des variables d'entrée

Corrélation avec la présence de cultures agricoles

	SSN	pH	Leco_N_ppm	C % Org	ICP OES K mg/kg	ICP OES P mg/kg
0	icr006454	7.85	800.0	0.94	8517.919223	96.575131
1	icr006455	8.03	600.0	0.70	10859.303780	117.423139
2	icr006474	5.01	500.0	0.57	1343.124117	87.040073
3	icr006475	4.57	500.0	0.47	1487.768795	83.555482
4	icr006492	6.78	900.0	0.98	2999.240760	150.936463

Nettoyage des données

- 1) Suppression des doublons
- 2) Élimination des valeur négatives
- 3) Suppression des colonnes presque vides/variables inutiles
- 4) Imputation des valeurs manquantes 0 ou moyenne des valeurs

```
todrop = ['Soil material','Scientist', 'Site', 'Country', 'Region', 'Gid','RES ID']

df_geoelements_reduced = df_geoelements.drop(todrop, axis = 1)

print(df_geoelements_reduced.shape)
print(df_geoelements_reduced.isna().sum())

(467, 54)
SSN          0
pH          15
%N          1

^def replace_missings(data):
    # this replaces missings with medians
    # NOTE: mixed string num columns it does not do anything with
    for cols in data.get_numeric_data().columns:
        data[cols].fillna(value=data[cols].median(), inplace=True)

replace_missings(df_geoelements_reduced)

print(df_geoelements_reduced['Cultivated'].unique())
df_geoelements_reduced['Cultivated'] = df_geoelements_reduced['Cultivated'].fillna('unknown')
```

Valeurs aberrants

```
def detect_outliers(df, n, features):
    """
    Takes a dataframe df of features and returns a list of the indices
    corresponding to the observations containing more than n outliers according
    to the Tukey method.
    """
    outlier_indices = []

    # iterate over features(columns)
    for col in features:
        # 1st quartile (25%)
        Q1 = np.percentile(df[col], 25)
        # 3rd quartile (75%)
        Q3 = np.percentile(df[col], 75)
        # Interquartile range (IQR)
        IQR = Q3 - Q1

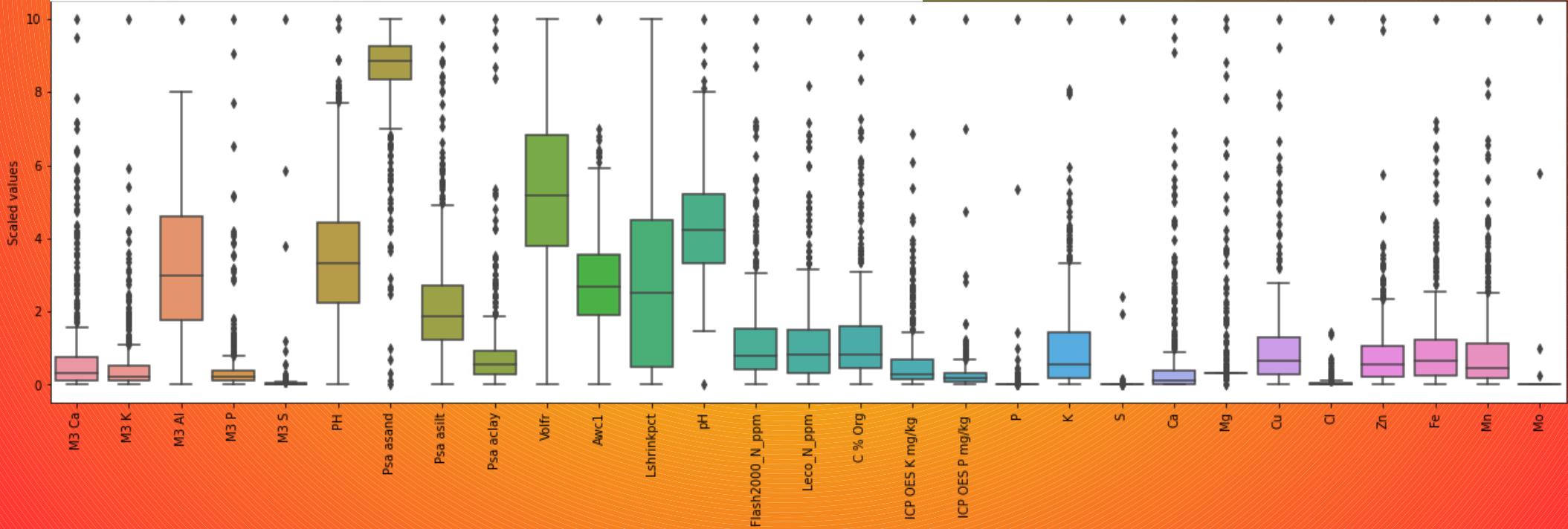
        # outlier step
        outlier_step = 3 * IQR

        # determine a list of indices of outliers for feature col
        outlier_list_col = df[(df[col] <= Q1 - outlier_step) | (df[col] >= Q3 + outlier_step)].index

        # append the found outliers into the list of outlier indices
        outlier_indices.extend(outlier_list_col)

    return outlier_indices
```

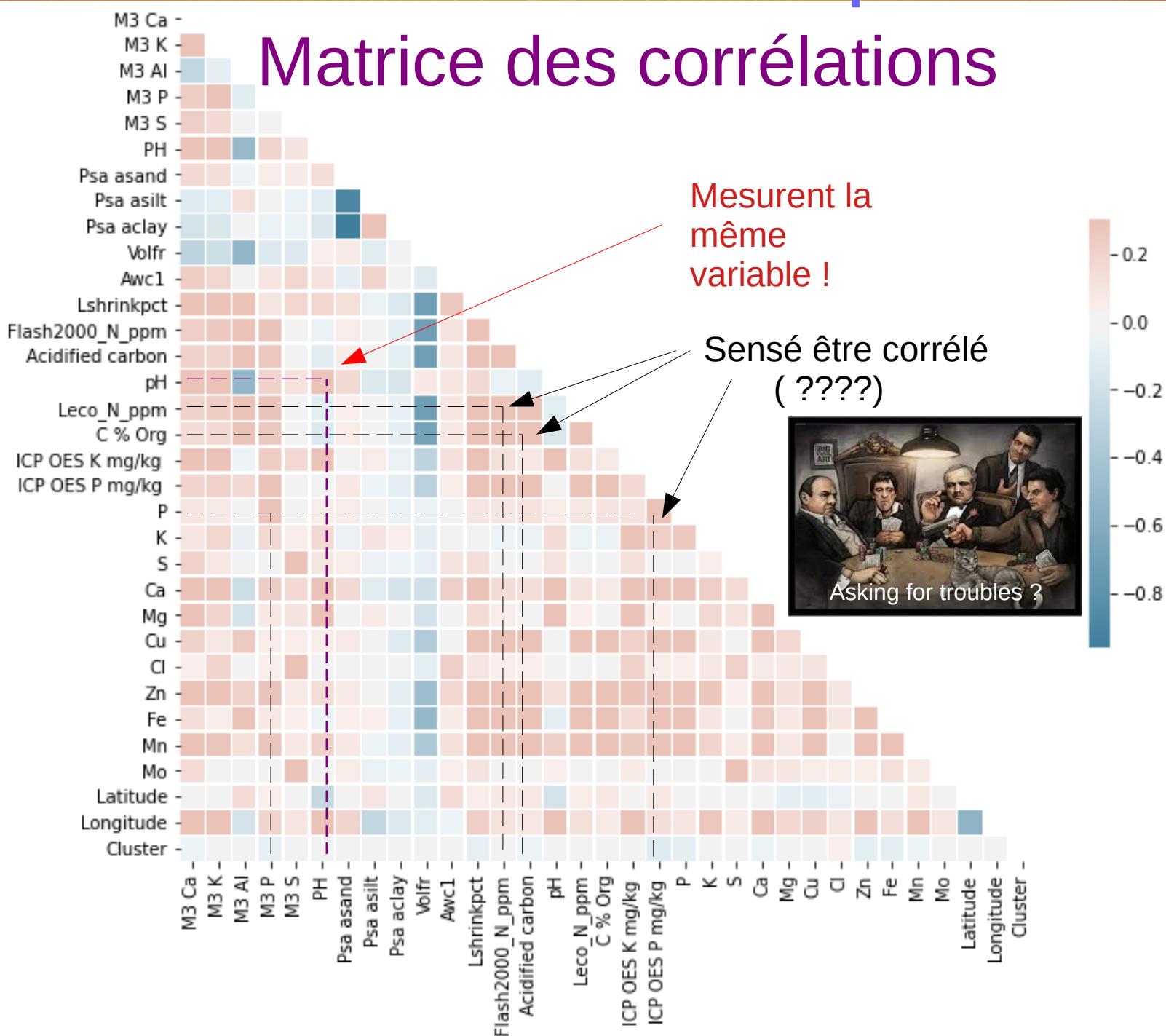
209/476 outliers



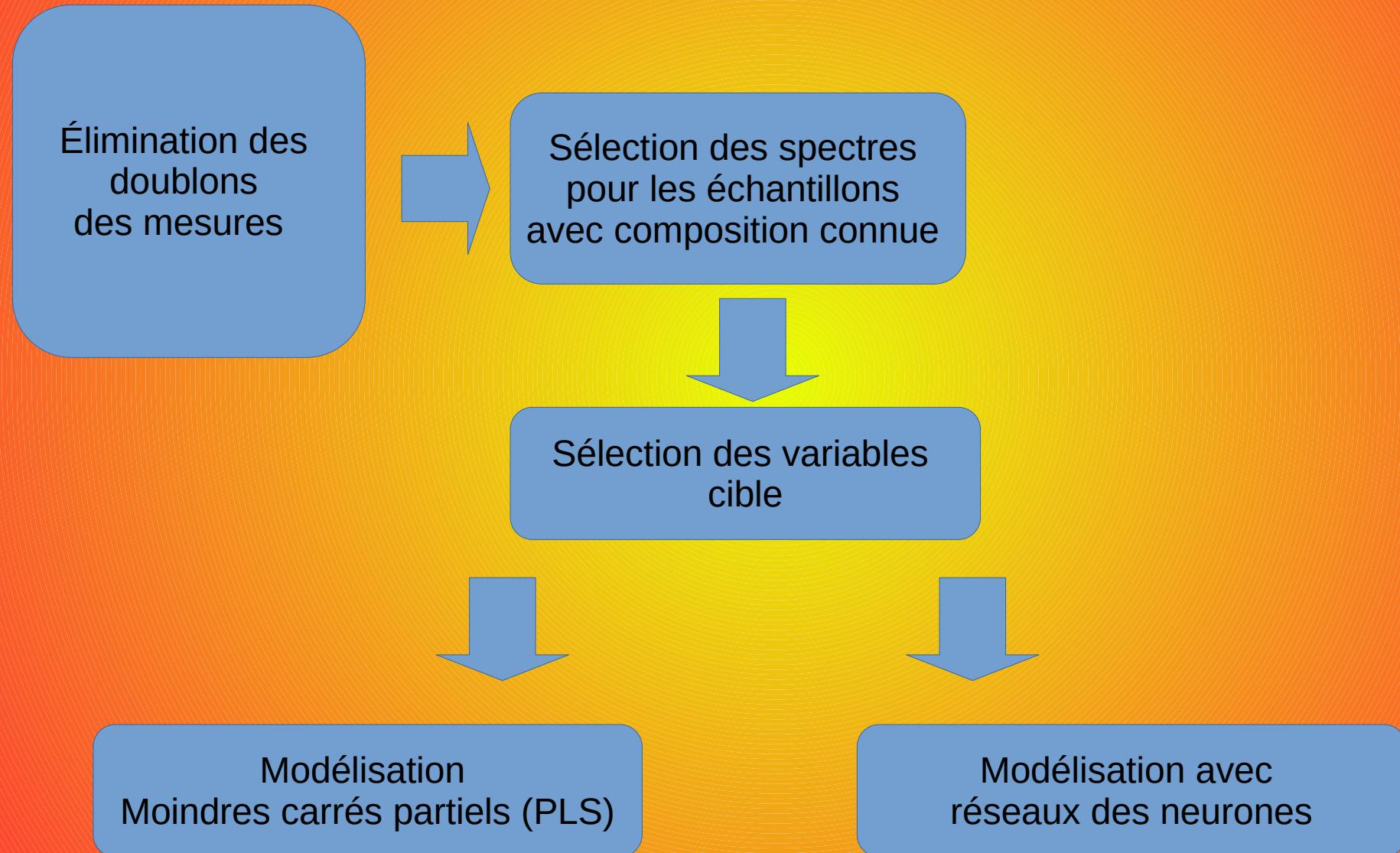
Identification des
valeurs aberrantes
avec la règle $2,0 \times$
écart interquartile

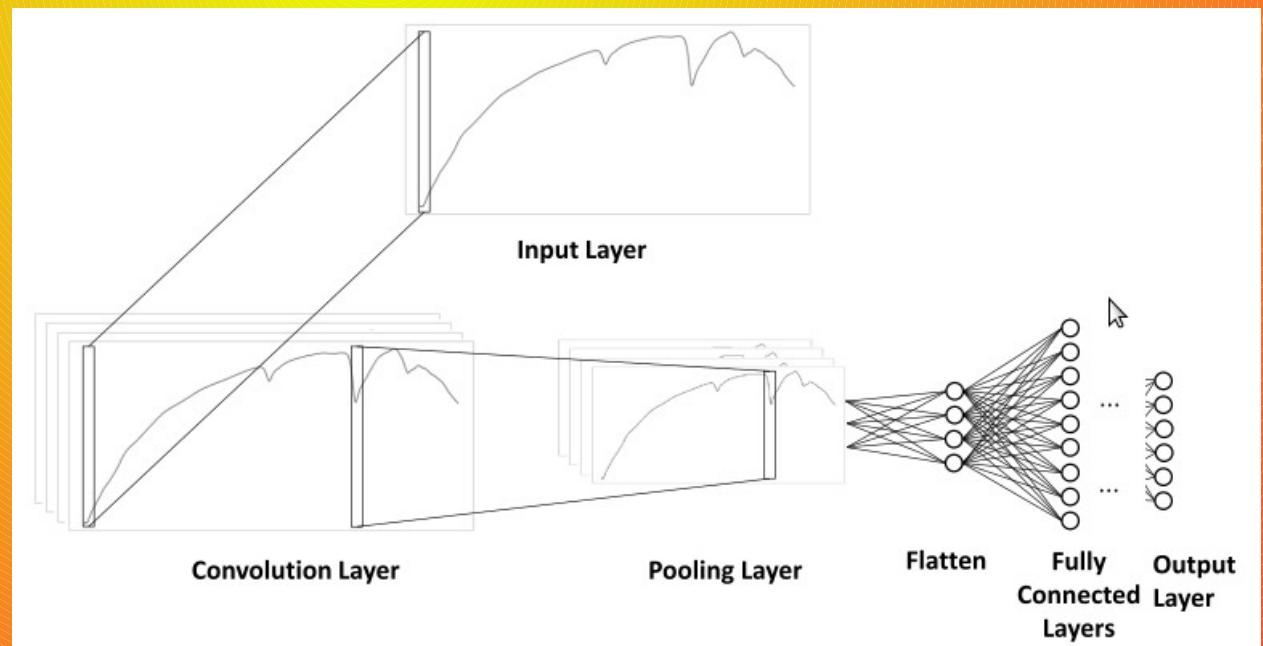
Sélection variables composition

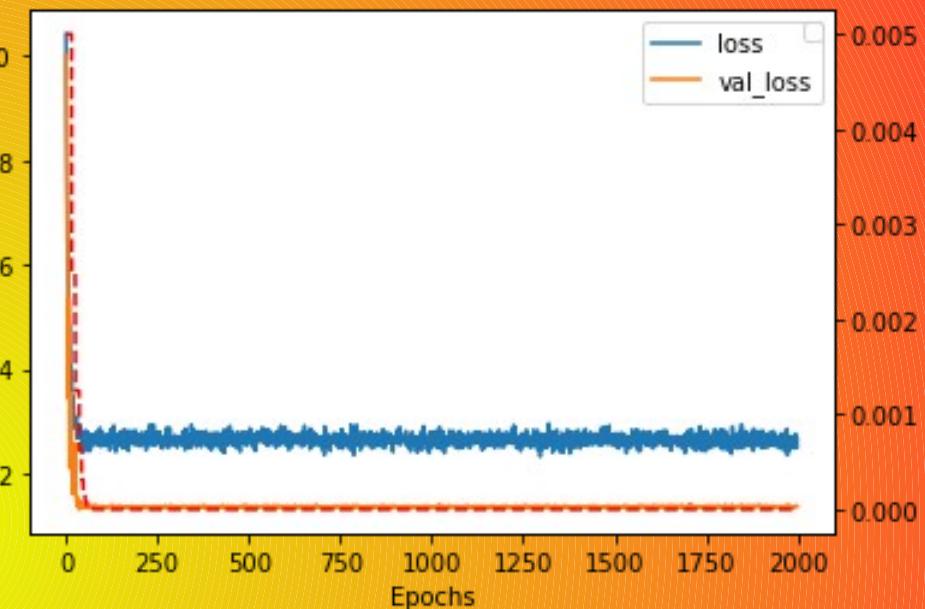
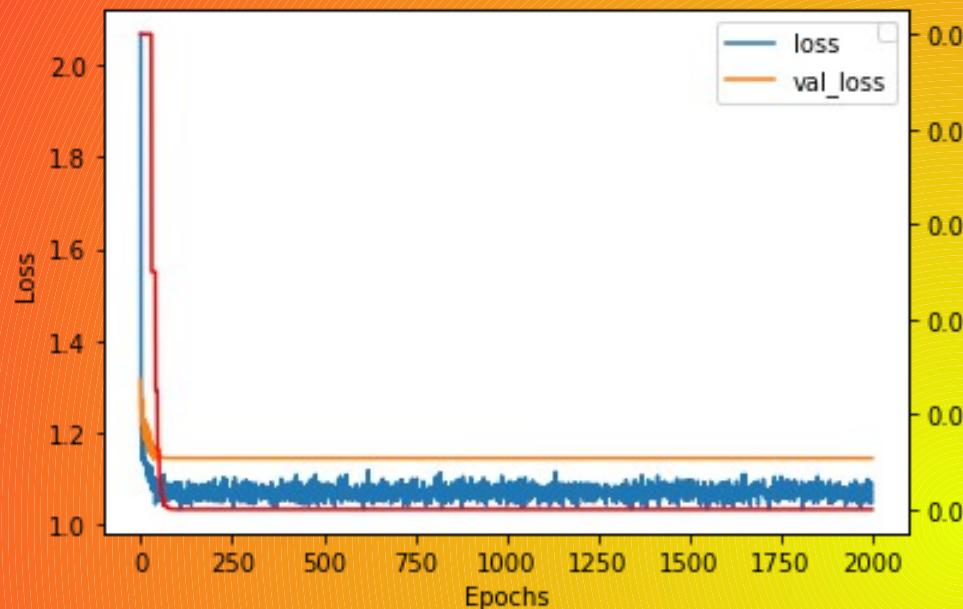
Matrice des corrélations



Procédure de modélisation : spectroscopie et composition







CNN - 3 variables prediction

