VSB Power Line Fault Detection

L.khellouf[1] Mentor: B.Beaufils[2]

[12]OpenClassRooms

Projet 8, juillet 26

Sommaire

Problèmatique

Le Center recherche et developpe des ressources energétiques renouvelables (ENET) cherche a d'étecter les decharges partielles dans les signaux acquis des lignes avec un nouveau compteur. L'objectif de ce travail est de créer un système en appliquant des techniques de machine learning capable d'identifier automatiquement différents états d'une ligne sur la base d'enregistrements audio.

Processus du projet

Les étapes de ce projet sont :

- Etudié les caractéristiques audio les plus pertinentes comme statistical features, fft, ...
- Développement des algorithmes de machine learning pour classifier les enregistrements fournis par ENET .

Un signal audio est une représentation du son, généralement sous la forme d'une tension électrique. Les signaux audio ont des fréquences dans la gamme de fréquences audio d'environ 20 à 20 000 Hz (les limites de l'audition humaine). Les types de domaine de signal sont:

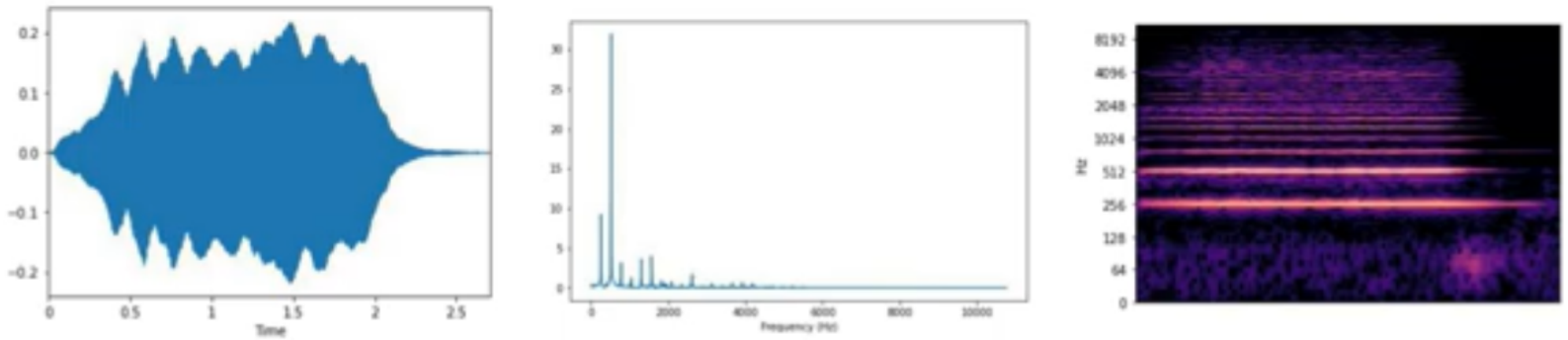


Figure: Signal domain

Fourier Transform

La transformée de Fourier est un outil pour transformer une fonction d'onde ou un signal d'un domaine temporel en domaine fréquentiel.

Spectrogram

Un spectrogramme est une représentation visuelle du spectre de fréquences d'un signal tel qu'il varie dans le temps.

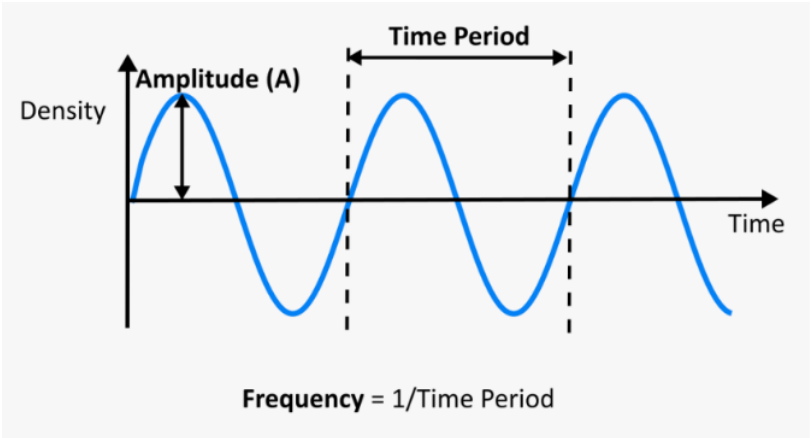


Figure: Amplitude, Time Perdiod and Frequency of Sound Wave

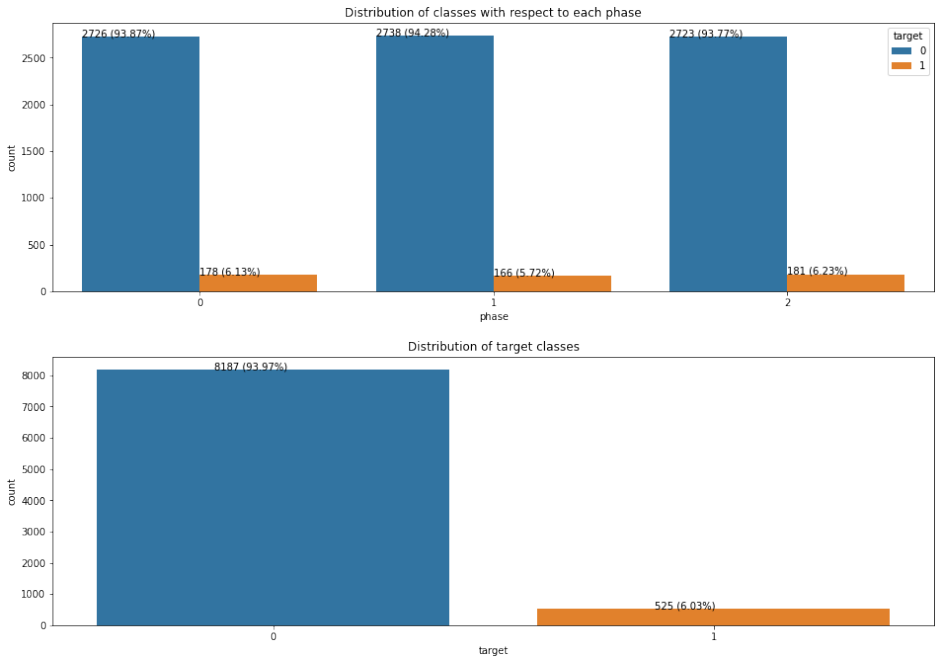
Exploration des données

Le jeu de données [1] utilisé dans ce projet a été développé dans le cadre de la compétition kaggle qui se concentre sur la reconnaissance automatique des sons des lignes éléctriques où se posait le problème de la classification des segments sonores en deux classes : décharge-partiel et no-décharge-partiel.

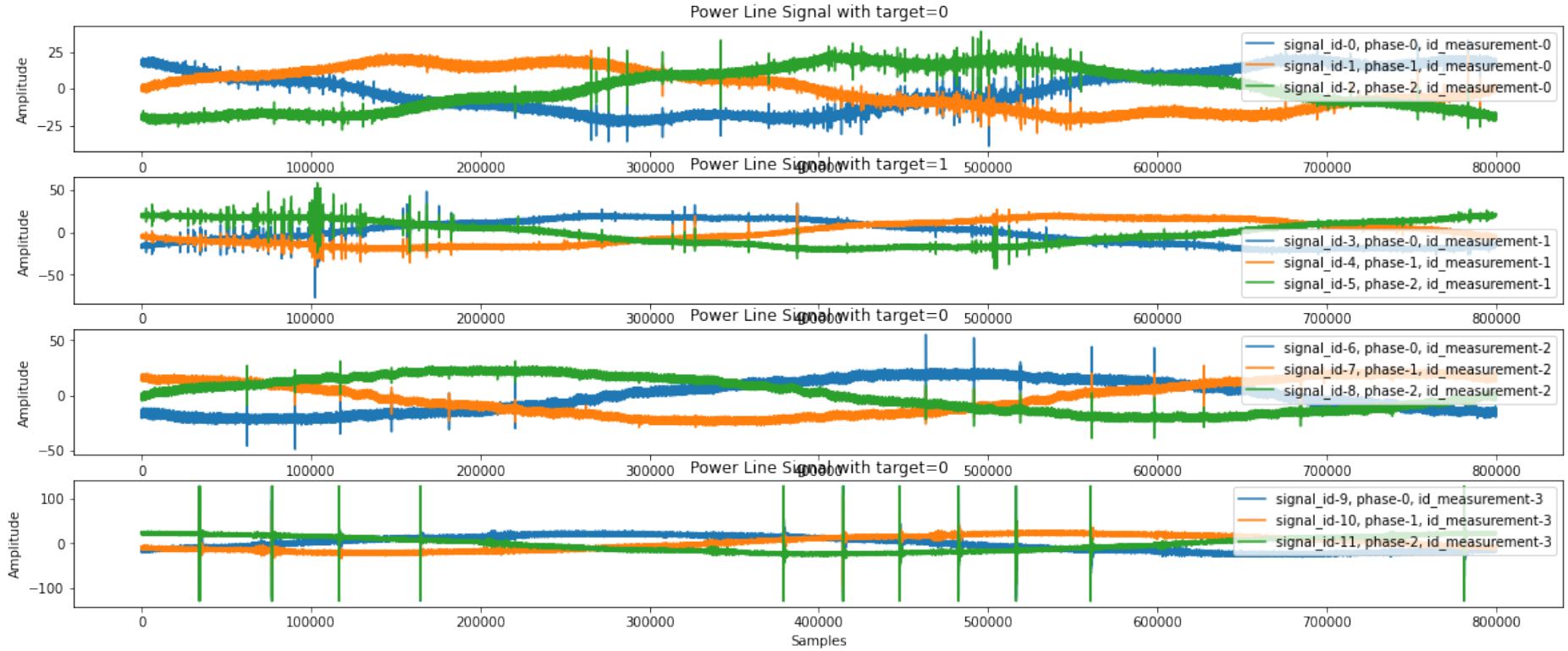
[1] 1https://www.kaggle.com/c/vsb-power-line-fault-detection/ overview

Exploration des données

Comme indiqué dans la compétition kaggle, chaque signal contient 800 000 mesures de la tension d'une ligne électrique, prises en 20 millisecondes. Comme le réseau électrique sous-jacent fonctionne à 50 Hz, cela signifie que chaque signal couvre un seul cycle de réseau complet. Le réseau lui-même fonctionne sur un schéma d'alimentation triphasé et les trois phases sont mesurées simultanément.



Exploration des données



Feature Extraction

Qu'est-ce que feature extraction ?

L'extraction de features est le processus de calcul d'une représentation numérique compacte qui peut être utilisée pour caractériser un segment audio.

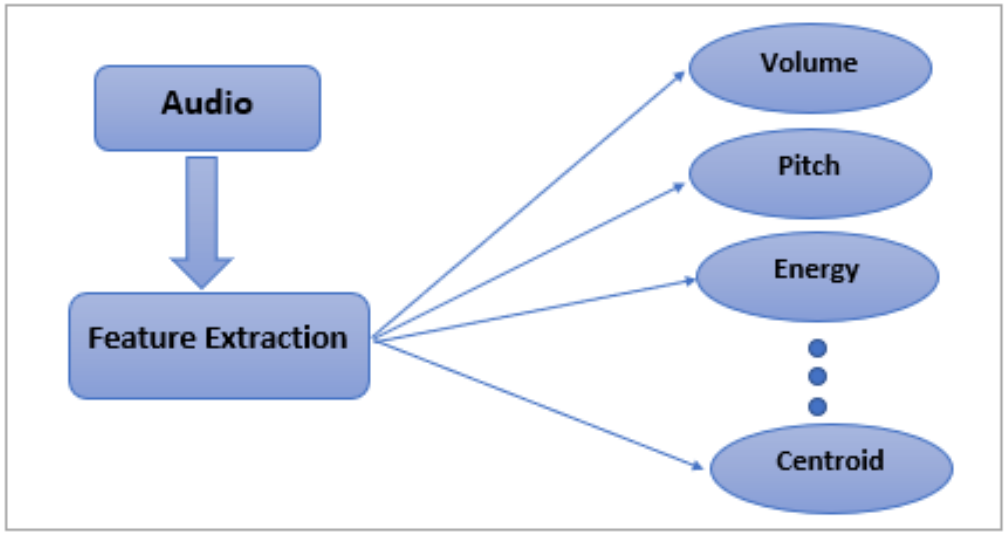
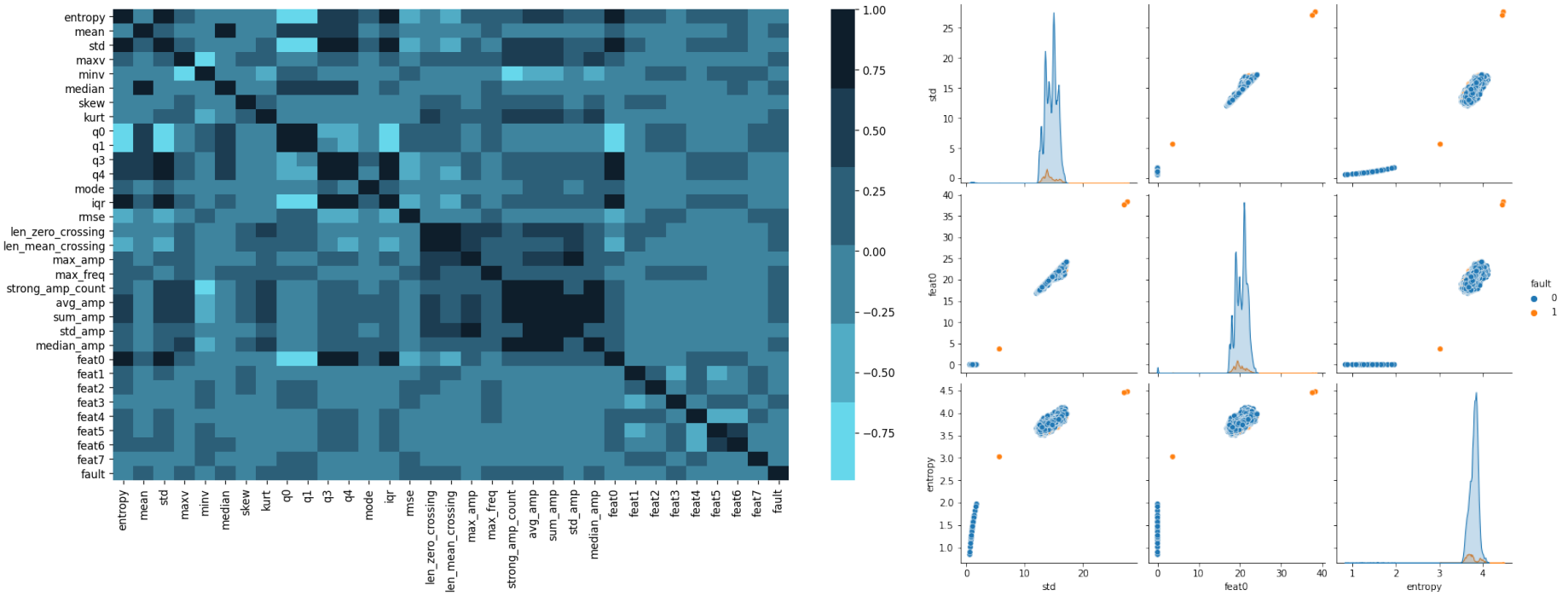


Figure: Process of the feature selection

Feature Extraction



Les points du nuage appartenant à la même catégorie sont distinctement regroupés et liés à une région.

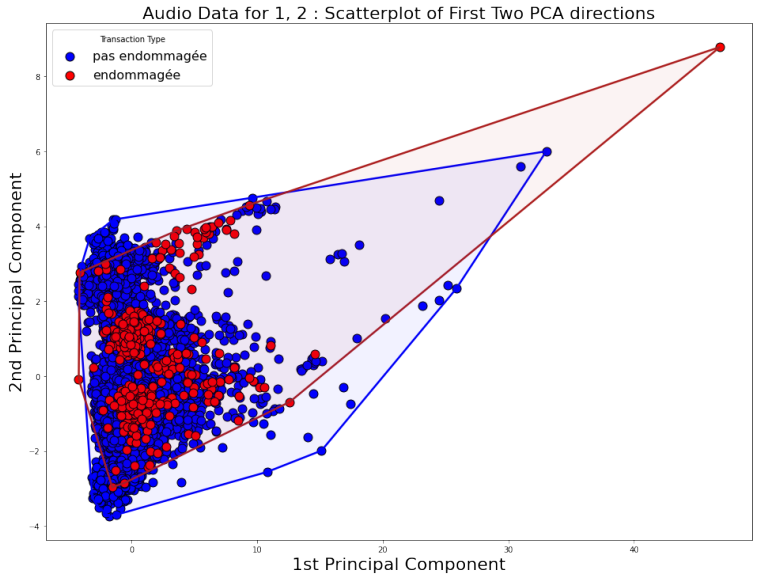


Figure: Audio data for Missing fault and fault : Scatterplot of First Two PCA directions.

Classification de l'audio

Pour classer l'audio en décharge partiel ou pas décharge partiel. Nous avons étudié les méthodes de classification supervisée suivantes :

- Logistic Regression
- Random Forest
- LightGBM
- XGBoost

Indicateur de performances

La mesure de performance clé utilisée par Kaggle pour évaluer les performances du modèle est le coefficient de corrélation de Matthews (MCC). La formule de calcul du MCC est illustrée à la figure suivante.

$$MCC = \frac{(TP * TN) - (FP * FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

Avec TN= True Negative, FP= False Positive, FN= False Negative et TP= True Positive.

Résultats

Nous avons 8712 enregistrements, tel que 525 sont étiquetés comme endommagée et 8187 comme pas endommagée

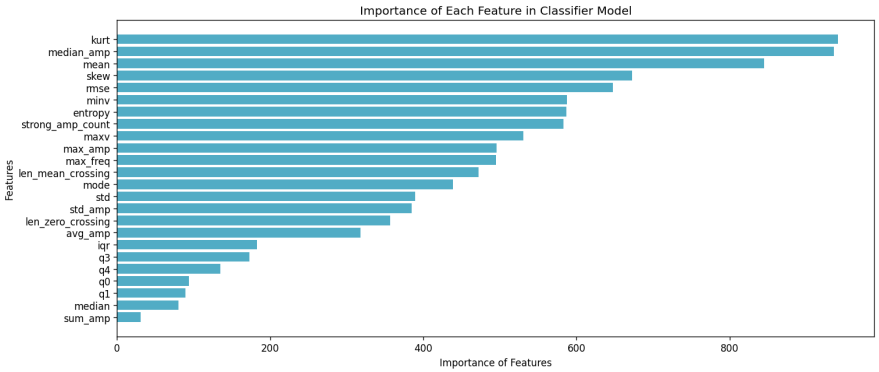
```
1. The F-1 score of the model 0.8212307639954011

2. The recall score of the model 0.7634396737496771

3. The Matthews Correlation Coefficient: 0.6647435081999267

4. Classification reporct
              precision    recall  f1-score   support

           0       0.97      0.99      0.98      5493
           1       0.87      0.53      0.66       344

    accuracy                           0.97      5837
   macro avg       0.92      0.76      0.82      5837
weighted avg       0.97      0.97      0.96      5837


5. Confusion matrix
 [[5465   28]
 [ 161  183]]
```

	CV	Model	f1	mcc	recall
0	0.0	Ub_LogisticRegression	0.484782	0.000000	0.500000
1	1.0	Ub_LogisticRegression	0.484782	0.000000	0.500000
2	2.0	Ub_LogisticRegression	0.484996	0.000000	0.500000
3	3.0	Ub_LogisticRegression	0.484768	0.000000	0.500000
4	4.0	Ub_LogisticRegression	0.484768	0.000000	0.500000
5	0.0	Ub_RandomForest	0.484782	0.000000	0.500000
6	1.0	Ub_RandomForest	0.484782	0.000000	0.500000
7	2.0	Ub_RandomForest	0.484996	0.000000	0.500000
8	3.0	Ub_RandomForest	0.484768	0.000000	0.500000
9	4.0	Ub_RandomForest	0.484768	0.000000	0.500000
10	0.0	Ub_LightGBM	0.695558	0.499300	0.630435
11	1.0	Ub_LightGBM	0.785790	0.581545	0.747254
12	2.0	Ub_LightGBM	0.756803	0.553391	0.695800
13	3.0	Ub_LightGBM	0.733967	0.540035	0.665756
14	4.0	Ub_LightGBM	0.784456	0.572640	0.759464
15	0.0	Ub_XGBoost	0.821655	0.658725	0.771723
16	1.0	Ub_XGBoost	0.827339	0.664596	0.785305
17	2.0	Ub_XGBoost	0.806330	0.626377	0.760156
18	3.0	Ub_XGBoost	0.824775	0.666808	0.772175
19	4.0	Ub_XGBoost	0.773211	0.558467	0.732755
20	0.0	B_LogisticRegression	0.056734	0.007335	0.500455
21	1.0	B_LogisticRegression	0.060535	0.016429	0.502275
22	2.0	B_LogisticRegression	0.061091	-0.056249	0.488934
23	3.0	B_LogisticRegression	0.092986	0.011473	0.504633
24	4.0	B_LogisticRegression	0.079356	0.014413	0.504593
25	0.0	B_RandomForest	0.640748	0.390568	0.832061
26	1.0	B_RandomForest	0.625060	0.366108	0.817990
27	2.0	B_RandomForest	0.630605	0.385085	0.838209
28	3.0	B_RandomForest	0.652853	0.414646	0.849667
29	4.0	B_RandomForest	0.654051	0.428559	0.867764
30	0.0	B_LightGBM	0.821491	0.654000	0.778059
31	1.0	B_LightGBM	0.837629	0.678247	0.812471
32	2.0	B_LightGBM	0.834206	0.668988	0.822693
33	3.0	B_LightGBM	0.877436	0.759616	0.844183
34	4.0	B_LightGBM	0.802069	0.604463	0.793874
35	0.0	B_XGBoost	0.821655	0.658725	0.771723
36	1.0	B_XGBoost	0.827339	0.664596	0.785305
37	2.0	B_XGBoost	0.806330	0.626377	0.760156
38	3.0	B_XGBoost	0.824775	0.666808	0.772175
39	4.0	B_XGBoost	0.773211	0.558467	0.732755

Meilleurs modéles

```
Model- 0 and CV- 0 recall: 0.7522583112447416, f1_score: 0.8188031177804891, mcc: 0.6704522
Model- 0 and CV- 1 recall: 0.7880352362490275, f1_score: 0.8465598987257115, mcc: 0.7133283
Model- 0 and CV- 2 recall: 0.7762270513300862, f1_score: 0.8279536045710096, mcc: 0.6721206
Model- 0 and CV- 3 recall: 0.7812425754335947, f1_score: 0.8439289491921069, mcc: 0.7122555
Model- 0 and CV- 4 recall: 0.7481587075314802, f1_score: 0.7913544354964087, mcc: 0.5954253
```

```
Performance Metrics for LGBMClassifier Cross Validation
1. The F-1 score of the model 0.8255607470629065

2. The recall score of the model 0.7691626022972154

3. The Matthews Correlation Coefficient: 0.6715758442326459

4. Classification report
              precision    recall  f1-score   support

           0       0.97      0.99      0.98      5493
           1       0.87      0.54      0.67       344

    accuracy                           0.97      5837
   macro avg       0.92      0.77      0.83      5837
weighted avg       0.97      0.97      0.96      5837


5. Confusion matrix
 [[5464   29]
 [ 157  187]]
```




```
Performance Metrics for LGBMClassifier Cross Validation
1. The F-1 score of the model 0.8071398920713989

2. The recall score of the model 0.7390323083422543

3. The Matthews Correlation Coefficient: 0.6529204376418938

4. Classification report
              precision    recall  f1-score   support

           0       0.97      1.00      0.98      2694
           1       0.93      0.48      0.63       181

    accuracy                           0.96      2875
   macro avg       0.95      0.74      0.81      2875
weighted avg       0.96      0.96      0.96      2875


5. Confusion matrix
 [[2687    7]
 [  94   87]]
```

Conclusion

Sur la base de la discussion dans les forums Kaggle, les approches suivantes pourraient conduire à un meilleur score MCC:

- La suppression du bruits avec des méthodes comme wavelet, ...
- Utiliser d'autres méthodes d'extraction de features comme les peaks
- Utiliser un meilleur réglage des hyper-paramètres des approches existantes
- Utiliser de meilleures fonctionnalités spécifiques au domaine
- Utiliser une architecture deep learning comme CNN

Le lien vers la compétition kaggle est [2]

[2] 1https://www.kaggle.com/leilakhell/vsb-power-line-fault-detection

Merci pour votre attention