CRYING REMOVING

# Learning features on crying sections

Before removing the ‘Crying segments’ (CSs), the first step involves differentiating them from the ‘Non-Crying segments’ (NCSs). This is done by supervised learning, using the 37 labeled signals from the current database. The learning phase will enable to determine one or more characteristics differentiating the CSs and NCS, so that an automatically CS detection can be computed. It will be useful when the database enlarges with other recordings.

The labeling of raw signals, the analysis of differences between CSs and NCs, and the determination of a threshold are the three necessary phases before CSs can be removed.

## Labelling the crying sections thanks to annotators

The first step was to label the signals with CSs and NCSs. This was done by 3 annotators on the entire signal basis. Independently, Lindsay, Arrabella and I listened to the different samples and annotated them on Audacity. An example of annotated signal is in Figure 1.

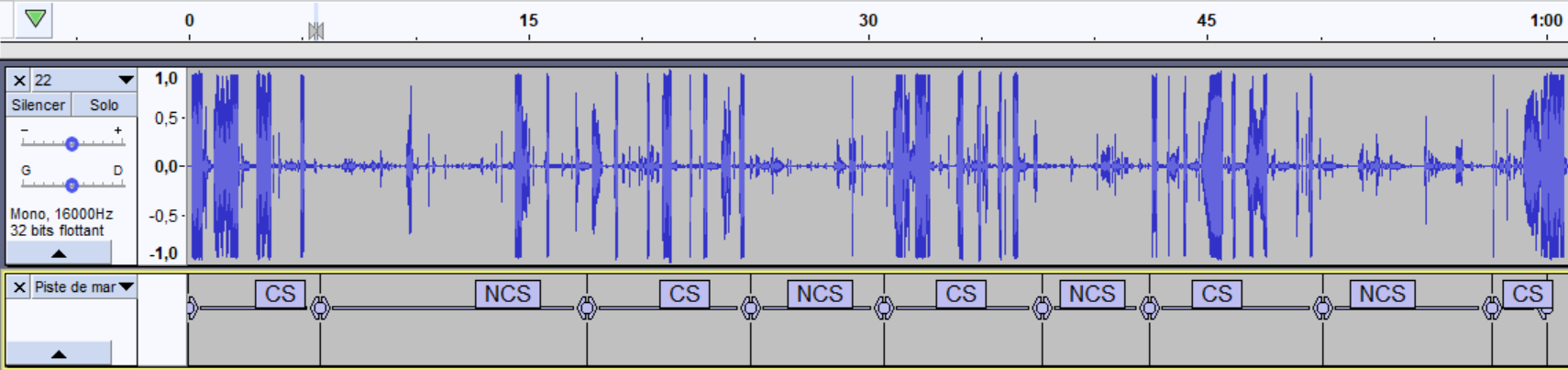


Figure 1: Julie's annotations of signal 22 on Audacity

The labels were extracted as text files. Each text file follows a strict name structure: ObersatorID\_SampleID.txt. Every line corresponds to a CS or NCS, with the beginning and end time of the section (cf figure 2).

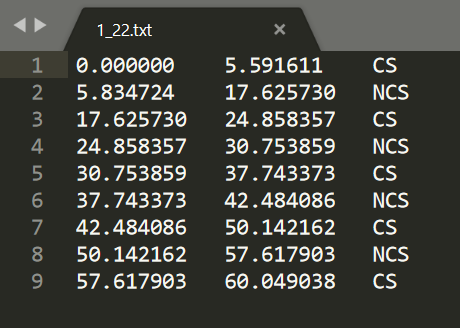


Figure 2: Text file generated by Audacity after Julie's annotations on signal 22

These files are read one by one in MATLAB, with the aim of creating a vector of 0 and 1, respectively corresponding to the NCS and CS labels.

The level of agreement between the annotators was then measured on each signal using Fleiss’ KAPPA. It is a statistical measure which assesses the [reliability of agreement](https://en.wikipedia.org/wiki/Inter-rater_reliability) between a fixed number of raters when assigning [categorical ratings](https://en.wikipedia.org/wiki/Categorical_rating) to a number of items[[1]](#endnote-1). In the project, three raters (Arabella, Lindsay and I), two categorical ratings (CS/NCS) and sixty items (60 sections of 1 second) were used to find a Fleiss’ KAPPA coefficient for each signal. A KAPPA coefficient equal to 0 means no agreement, while 1 means perfect agreement. It was implemented on Matlab using a function in the Matlab File Exchange [[2]](#endnote-2) . The KAPPA coefficient mean of all the signals is 0.6 🡪 à revoir car valeur negative!!.

Finally, each CS with a 2/3 or 3/3 agreement have been retained. The figure 3 illustrates the final annotated labels of signal 22.

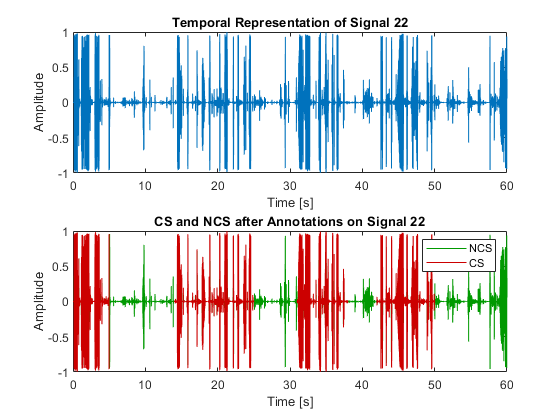


Figure 3: Annotated CS and NCS of Signal 22

## Analysis of differences between CSs and NCSs

Once the learning database has been labeled, an analysis was necessary to determine which feature(s) will enable a correct distinction between CSs and NCs. The spectrogram, the power spectrum and a set of other features (initially implemented for the part ...) were tested.

The sections of signals used to make the comparison between CSs and NCs were defined as "all pure segments of 3 seconds contained in the learning base". The purity of segments, ie the fact that they are entirely composed of CSs or NCs, as well as the duration of these sections have been determined in an empirical way (cf annex ...).

### Spectrogram

The spectrograms have confirmed a first difference. Generated on signal parts with crying and non-crying areas, they were designed with good temporal resolution, so that CSs can easily be distinguished from NCs (window = 1s, overlap = 25%). They highlight a much higher intensity in the CSs, particularly marked around 350Hz but visible throughout the frequency band. The spectrogram of the first 15s of signal 22 is in the figure 4 below.

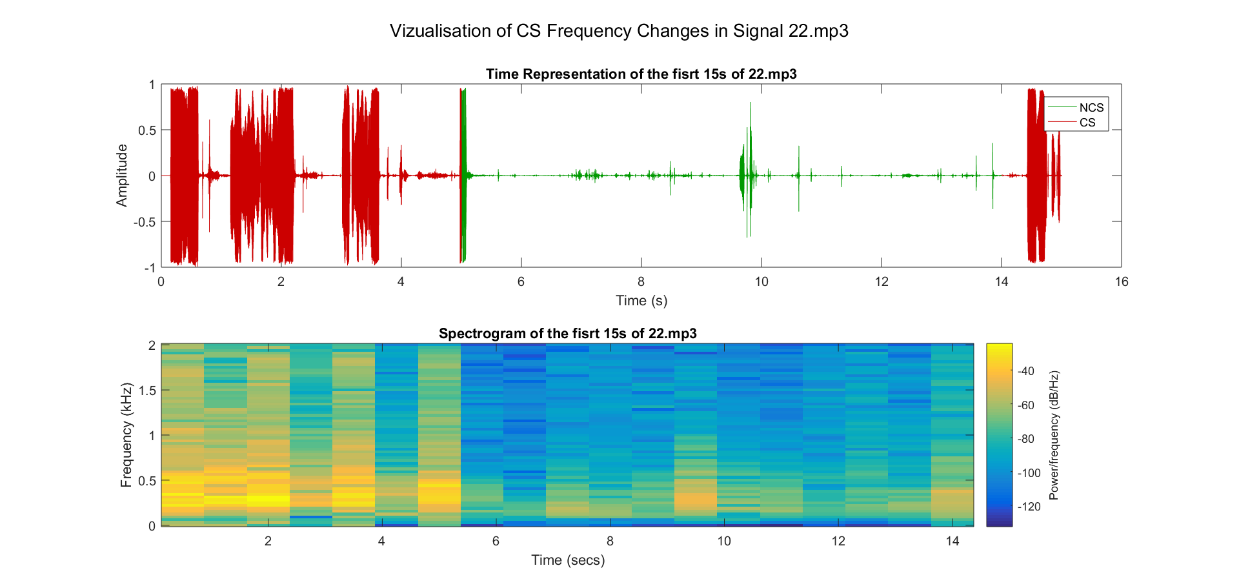


Figure 4: Visualization of CSs/NCSs Frequency Changes in the first 15s of Signal 22

### Power spectrum

The power spectrum was then computed to better visualize the differences for CSs and NCs in frequency composition and intensity of each frequency. The figures ... and ... represents the power spectrums. Taking a logarithmic scale in the figure ... highlights the differences in the high frequencies. The divergence can be seen in dotted lines with the first and third quartiles.

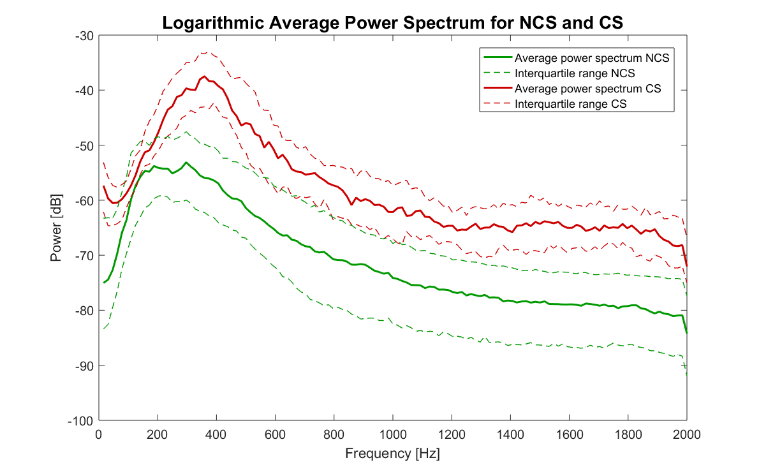
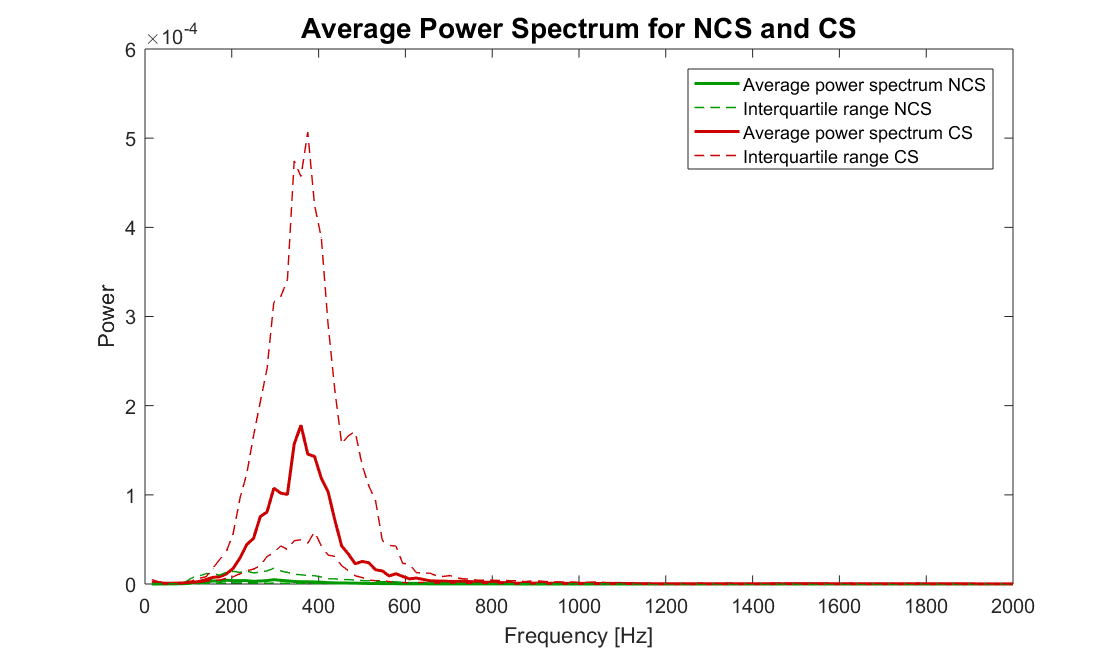


Figure 2: Average Power Spectrum for NCS and CS

Figure 3: Logarithmic Average Power Spectrum for CSs and NCSs

The power differs between the CSs and NCs, with a limited overlap in the following frequency bands: [250-450] and [1400-1900] Hz.

### Overview of other Spectral Features Differences

A panel of other spectral features has been implemented to have a more global vision. For better results, the frequency band used is limited to [296-407] Hz, which corresponds to the first and third frequency quartiles of the CSs.

The characteristics tested are partly those implemented in the part .... They are represented in a boxplot form, allowing a quick vision of differences between CSs and NCSs.

ICIIIIII

METTRE LES BOITES A MOUSTACHES : 1 pour MFCC, 1 pour LPC, 1 pour tous sauf de p25 à P800-1200

Faire commentaires

4 : Choix de ce qu’on garde pour différencier.

Finally, it was decided that the Power Ratio could be enough to differentiate the CS and NCS. Comme pas beaucoup de temps et qu’il y a déjà une grande différence dans PR, c’est ce qu’on utilise When labelling the signals with a window of 1s, a huge difference can be seen in the power spectrum average between 600Hz to 1000Hz (cf figure 5).

A threshold in the power spectrum average must be determined to differentiate a CS from an NCS. The power ratio average in the frequencies between the first and third quartile of the CS power spectrum will be used + dire les autres facteurs. Dire que des tests ont été fait, don’t les résultats se trouvent en annexe.

Mieux des window de 3s car pwelch pas trop performant quand trop petit

## Threshold Establishing

In the interest of time and efficiency, the Receiver Operating Characteristic (ROC) curve was used to find a correct threshold. It is a fundamental tool for diagnostic test evaluation as well as classifier decision. In this case, only one classifier is employed (“>”), making the ROC curve useful for threshold determining only.

This method tells how the model is right or wrong, based on sensitivity and specificity. These two probabilities are computed thanks to the possible outcomes, summarized in the table … below.

|  |  |  |
| --- | --- | --- |
| CRYING | | |
|  | **Present**  Actual CS | **Absent**  Actual NCS |
| **Positive**  Predicted CS | True Positive (TP)  CS/CS | False Positive (FP)  NCS/CS |
| **Negative**  Predicted NCS | False Negative (FN)  CS/NCS | True Negative (TN)  NCS/CS |

*Mettre légende*

The sensitivity is the probability that a test result will be positive (ie CS) when crying is present. It is the True Positive Rate, expressed as a percentage.

The specificity is the probability that a test result will be negative (ie NCS) when crying is not present. It is the True Negative Rate, expressed as a percentage.

In a ROC curve the True Positive Rate (Sensitivity) is plotted in function of the False Positive Rate (100-Specificity). Each point represents a sensitivity/specificity pair corresponding to a particular decision threshold. The point closest to the upper left corner will have the highest sensitivity and specificity and therefore will match the best threshold.

The following figure … represents the ROC curve for 500 thresholds, taken linearly between 0 and the maximum power band ratio. The yellow cross in the ROC curve shows the closest point to the upper left corner, corresponding to a power ratio threshold equals to 0.0018.

Changer valeur et courbe (en mettant dans le titre fentetre, purity and frequency band)

Mettre légende pour seuil sur image.

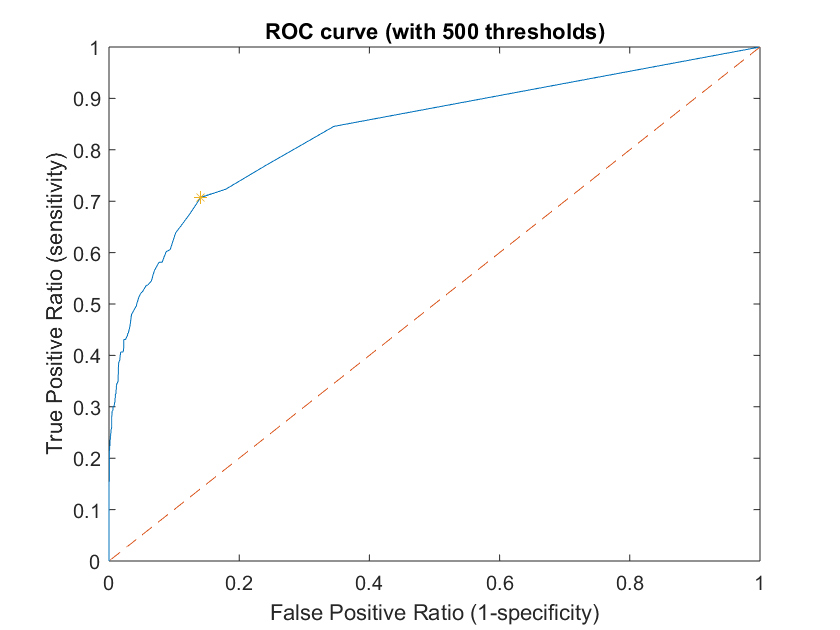


Figure 4: ROC curve used for CS determining

Partie Crying removing: parler de la longeur minimale retenu.

## Crying Removal

# Results Crying Removal

CSs were detected with 70% accuracy, while 86% for NCSs avec accuracy globale de …. . Ce qu’on voulait c’était enlever les CSs (on se fiche un peu des NCSs. The figure below shows the different stages before getting a signal without much crying, on sample 22. The rough signal [a] was first labelling by annotators [b] to be able to learn characteristics specific to CSs. A power ratio threshold was then established, allowing a new labelling [c]. Finally, crying sections were removed and signal shorten [d].

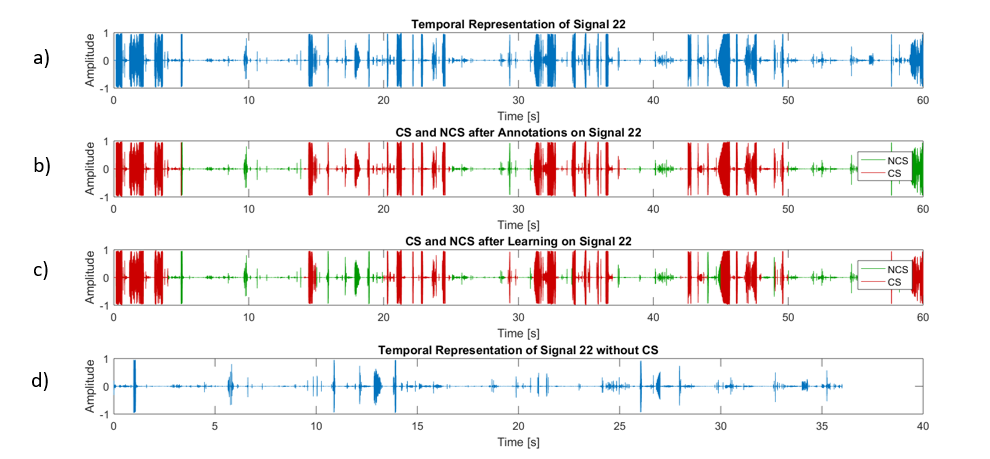


Figure 5: Temporal Representations of the different steps to remove crying.

Remettre bonne image

The area under the ROC curve, AUC, is a measure of how well a parameter can distinguish between two diagnostic groups (CS/NCS).

# Discussion

The preprocessing step allowed to obtain a signal containing much less noises. DIRE TOUS CE QUE CA PREND EN COMPTE. In the allocated time, and because this work is done to write a medical paper only, the in-depth study on preprocessing is appropriate. However, it is only a first version that could then be improved.

Concerning the removal of crying, other more efficient methods could have been used, especially with more advanced machine learning techniques. The paper [[3]](#endnote-3) uses for example Deep Learning for cry automatic detection of babies aged between 0 and 6 months. Even without going that far, taking into account more features, choosing different classifiers and using cross validation is one step to improve results.

Dire que peut etre pas assez de CS et NCS dans training + Utilisation de d’autres claasifiers comme … + cross validation (donner plusieurs méthodes et expliquer en quelques lignes avec des tirets). Demander à Fae d’autres idées

## Analysis of differences between CS and NCS

Once the theoretical labeling of the signals has been done, it was necessary to learn how to detect CSs. In order to know which characteristics will be most appropriate for the detection of CSs, a first study has been done. It is based on box plots illustrating the differences between NCSs and CSs on common signal processing characteristics. Power spectrum features as well as MFCC coefficients and LPC coefficients are analyzed (figures …)

Once the theoretical labeling of the signals has been done, it was necessary to learn how to detect CSs. In order to know which characteristics will be most appropriate for the detection of CSs, a first study has been done. It is based on box plots illustrating the differences between NCSs and CSs on common signal processing characteristics. Power spectrum features as well as MFCC coefficients and LPC coefficients are analyzed (figures …)

Spectrograms of some signal parts where crying is present are also generated to compare the frequency differences between NCSs and CSs. To do so, a small window length was used to have a good time resolution (window=1s, overlap=25%). The spectrogram of the first 15s of signal 22 is in the figure 4 below.

They are then tested thanks to K-fold Cross Validation.

## Cross validation (A mettre dans discussion + Machine Learning ??)

Cross-validation is a statistical method often used to estimate the skill of machine learning models on a limited data sample. In this particular case, it will be helpful to evaluate the better threshold.

A training dataset (required to determine the threshold) and a validation dataset (which allows threshold testing on new samples) are required. Different cross-validation methods exist to differentiate those two datasets. Exhaustive cross-validation methods learn and test on all possible ways to divide the original sample into a training and a validation set, whereas the non-exhaustive ones do not compute all ways of splitting the original sample.

The k-fold cross-validation method was chosen. It is non-exhaustive but remains a method that does not introduce much bias and allows a quick calculation time. The procedure has a single parameter called k, that refers to the number of groups that a given data sample is to be split into. Each CS and NCS is assigned to an individual group and stays in that group for the duration of the procedure. It will be used for the validation 1 time and k-1 times to train the model.

The general method is as follows[[4]](#endnote-4):

1. Shuffle the dataset randomly.

2. Split the dataset into k groups

3. For each unique group:

1. Take the group as a test data set

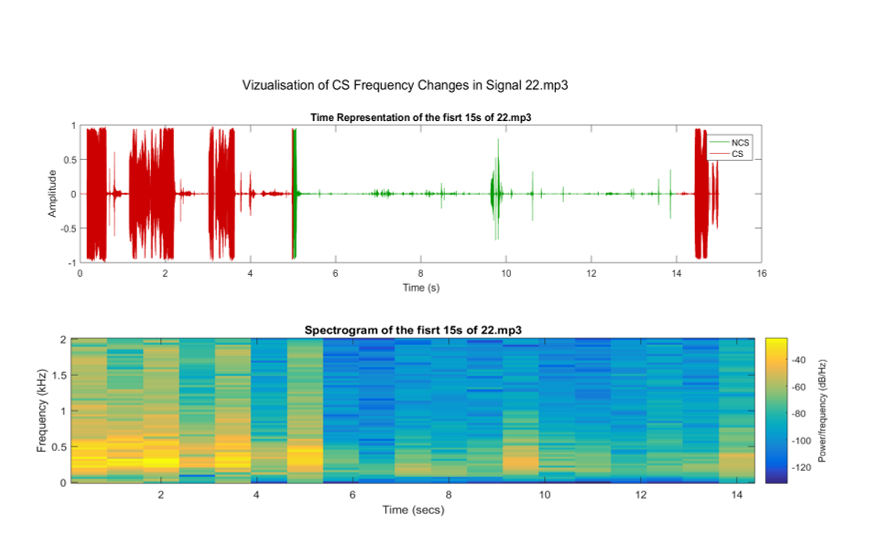
2. Take the remaining groups as a training data set

3. Fit a model on the training set and evaluate it on the test set

4. Retain the evaluation score and discard the model

4. Summarize the skill of the model using the sample of model evaluation scores

The value for k is chosen such that each train and test groups are large enough to be statistically representative of the entire dataset. The k-value 5 or 10 are often used as it was shown empirically that they don’t suffer neither from excessively high bias nor from very high variance.[[5]](#endnote-5) In this case, the dataset is composed of 37 signals with a label window of 1s, which allows 2183 observations. 10 was then chosen as the k value.



1. <https://en.wikipedia.org/wiki/Fleiss%27_kappa> [↑](#endnote-ref-1)
2. <https://github.com/dgolden1/matlab_fleiss_kappa/blob/master/fleiss_kappa.m> [↑](#endnote-ref-2)
3. Yizhar Lavner, Rami Cohen, Dima Ruinskiy, Hans IJzerman. Baby Cry Detection in Domestic Environment using Deep Learning. 2016. [↑](#endnote-ref-3)
4. <https://machinelearningmastery.com/k-fold-cross-validation> [↑](#endnote-ref-4)
5. An Introduction to Statistical Learning, 2013 [↑](#endnote-ref-5)