Spotiflop Algorithm

Introduction to Machine Learning (TI508M)

By Max BORTOLOTTI, Hugo LEYX-VALADE and Roland FONTANES

Table of Contents

[Introduction & Objectives 3](#_Toc184135280)

[The Algorithms 3](#_Toc184135281)

[K-nearest Neighbors 3](#_Toc184135282)

[Random Forest Classifier 6](#_Toc184135283)

[Support Vector Machines 7](#_Toc184135284)

[Comparisons 8](#_Toc184135285)

[Precision 8](#_Toc184135286)

[Sensitivity / Recall 8](#_Toc184135287)

[Specificity 8](#_Toc184135288)

[F1-Score 8](#_Toc184135289)

[Conclusion 8](#_Toc184135290)

# Introduction & Objectives

We are working in a famous music-streaming company and our goal is to determine the type of music the customer is listening to. In the future, this could help to create algorithms of recommendations based on the likes of the auditor.  
  
So based on the multiple features such as the popularity, acousticness or tempo, we must be able to classify any other music in one of the genres.

This is why this project is about classification, because we have several genres and it cannot be clustering because we already know the genres, and we do not want to discover new under-genres. Moreover, this is not regression because genres are not numbers, they are categories.

After having the Data Loading and Pre-processing, we now invite you to discover our work and our results through these following pages.

# The Algorithms

## K-nearest Neighbors

The K-nearest Neighbors (KNN) algorithm is based on the proximity of each value to each other, this is why it was important to choose the right distance algorithm. And among all those existing we decided to concentrate on Euclidean distance and Manhattan distance. Here are the respectively calculated errors for each one:

Une image contenant texte, capture d’écran, Tracé, diagramme

Description générée automatiquementUne image contenant texte, capture d’écran, Tracé, ligne

Description générée automatiquement

You can see here how both algorithms are close in error rate, but the Manhattan technique has less error. This is why we chose it.

Now, let’s go deep into the KNN algorithm by plotting its confusion matrix:

Une image contenant texte, capture d’écran, Parallèle, nombre

Description générée automatiquement

We can clearly see here the True Positive diagonal, which is pretty good for Blues, Bollywood, HipHop, Instrumental, Pop and Indie. But we can also observe a big problem which is that the Indie class is dragging a lot of other genres in its predictions, for example, the Alt class has 97 good predictions and 235 predictions in the Indie class which is a pretty bad performance.

So, we decided to remove the Indie genre temporarily to see how the dataset would respond:

Une image contenant texte, capture d’écran, carré, Parallèle

Description générée automatiquement

Now the Alt genre has way better results. Furthermore, Rock, which was predicted as Indie beforehand, is now considered as Alt. The main reason for this is that the Indie class has way more values in the dataset and so the training is influenced by this. Moreover Rock, Alt and Indie are very close genres in music.

## Random Forest Classifier

A Random Forest Classifier (RFC) is an ensemble machine learning algorithm that creates multiple decision trees using random subsets of training data and features, with each tree "voting" on the final classification. Here is our confusion matrix with this algorithm:

Une image contenant texte, capture d’écran, Parallèle, carré

Description générée automatiquement

## Support Vector Machines

The Support Vector Machine (SVM) is an algorithm that finds the best boundary to separate different groups of data points. It works by creating the largest possible gap between groups, helping to clearly classify new data into the correct category. Here is our confusion matrix with this algorithm:

Une image contenant texte, capture d’écran, Parallèle, nombre

Description générée automatiquement

# Comparisons

## Precision

Une image contenant texte, capture d’écran, Tracé, diagramme

Description générée automatiquement

So in this precision plot we can see how each algorithm has correctly guess the right answer (True Positive Rate). The two most precise algorithm are RFC and SVM in our case and KNN seems to be always behind except for the Bollywood class.

## Sensitivity / Recall

Une image contenant texte, capture d’écran, Tracé, diagramme

Description générée automatiquement

The Recall is here to evaluate, out of all the truly positive cases that exist, how many did we successfully catch. Here, the predominant one is mostly RFC but SVM is really close to it in general, whereas KNN seems to be always behind again.

## Specificity

Une image contenant texte, capture d’écran, Tracé, diagramme

Description générée automatiquement

The specificity tells us how well we identified the negative cases, and here again we can observe almost the same pattern as the recall.

## F1-Score

Une image contenant texte, capture d’écran, Tracé, diagramme

Description générée automatiquement

Finally and maybe the most important one : the F1-score. By combining precision and recall we can identify the overall best algorithm, and here again SVM and RFC are fighting but this last one seems to be still better than the other overall.

# Conclusion

All in all, the RFC seems to give the best performances and results for our dataset, especially with the F1-score. But we have to keep in mind that the dataset we chose was not perfect, as the Indie class was overrepresented.

This project has helped us to discover new algorithms and deepen our knowledge of KNN. The group work was pretty effective, as everyone was invested and wanted to learn more about Machine Learning.

Finally, we hope you enjoyed reading our report while discovering more about the Spotiflop machine learning algorithms.