Chapter 1

Optimizing an acoustical representation against perceptual judgments

1.1 Introduction

Mathieu : AGAIN, USE PRESENT TIME !!

In this second part of the study, 78 samples are selected from the SOL database and labeled using a perceptual study performed with 32 subjects. Each subject gave different labels based on her/his opinion on which two samples are similar or not.

1. **Feature extraction**

   In this part, the same feature extraction method will be applied to the samples. The MFCC will be extracted and then the standardization will be applied. For the scattering, the features will be extracted and then the std and median method of preprocessing will be applied.

2. **The metric ranking**

   To start the study of this method, the two ranking metrics used in the first part should be generalized. The new MAP and P@5 that are used are obtained in the following way :

   - Compute the metric ranking for each label provided by each subject.
   - Compute the average of the ranking metrics over the 32 set of labels corresponding to each subject.

3. **Results.**

features	map	pat5
mfcc	50.74	55.66
scattering	40.17	44.01

Table 1.1: Results of applying the MFCC and the scattering to the ground truth labeling problem

The table shows the results of the ranking metrics applied to the features extracted using the MFCC and the scattering. The MFCC outperforms the scattering even with the preprocessing applied.

1.2 Metric learning

In the first part of the study, the problem was treated without the need of methods of learning. In this part however, the problem is more complicated. This complication is due to the fact that there are factors that affects the decision of each subject. Those factors are based on the cabling of the neurons inside the brain. Those cabling are established for each subject differently based on how he experienced those sounds. The problem is to find a space that is not just representative to the physical aspect of sound but that takes into account the opinion of different subjects. This problem would be very complicated to solve without considering techniques of learning.

1.2.1 K-NN vs search query

For each query, the space of interest is closest five samples. This evaluation indicates that beyond those 5 samples, there is no change of precision based on the accuracy of those samples. Thus the evaluation of a search query is highly related to the k nearest neighbors classification.

In a k nearest neighbors classification, the euclidean distance is computed between each new sample and the entire samples of the space. For each new sample, a decision is made based on the classes of the the five nearest neighbors. If the majority of the nearest neighbors correspond to a certain class, the new sample will be assigned this class.

In both cases (K-NN and search query), an ideal case is that the k nearest neighbor of each sample corresponds to the real class of that sample. However in the K-NN classifier, it is enough to have the majority of the samples assigned with the exact class

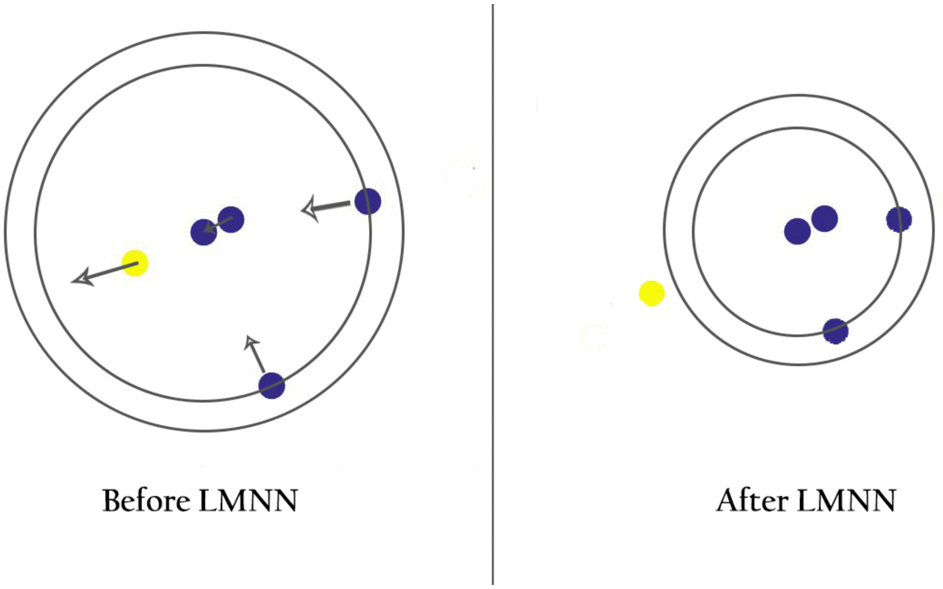

Figure 1.1: An example showing the motivation behind the use of the LMNN

of the sample where in the search query it is needed to have the k first results from the same class of the search query.

The LMNN short for "Large Margin Nearest Neighbor" is a learning method designed to solve the K-NN classification problem. Its concept however is well suited for the problem of search query addressed in this paper.

1.2.2 Large Margin Nearest Neighbors

The learning technique that is considered in this paper is the large margin nearest neighbors [8]. In the LMNN the ideal space would be one where each observation has the same class of its K nearest neighbors. This is done in the following way :

- Compute the pair wise Euclidean distance of the space.
- Attract the K nearest neighbors corresponding to the same class(Shrink the distance between the observation and the k nearest corresponding to the same class).
- Set a marge based on the farthest nearest neighbor from the same class.
- Push all observations from different classes outside this marge(expand the distance between the observation and other samples that violate the margin.)

To better understand this idea, let us have a look at the following example in figure 1.1. Consider a space of five samples that are divided in two classes. The blue class corresponds to samples played by a violin and the yellow class corresponds to samples played by a guitar. Let us focus on the effect of applying LMNN on the blue sample at the center. The settings of the LMNN will be tuned to a problem of three nearest neighbors classification. As showed in the first part of the figure 1.1 that the three nearest neighbors to the blue sample in the middle are as follows : [Violin, Guitar, Violin]. The LMNN algorithm will thus aims to attract the two other samples that correspond to the violin class. To insure that the guitar will not be one of the nearest neighbors of that sample a margin will be set and the yellow sample should be pushed outside the margin. If the LMNN is applied with success the result will be similar to the second part of the figure 1.1. The same process will be applied to each sample of the space.

Learning a Mahalanobis distance

To perform the process explained above, the LMNN uses a projection of the space by computing a Mahalanobis distance. A Mahalanobis distance is defined as follows :

$$D_{\mathbf{M}}(\vec{x_i}, \vec{x_j}) = (\vec{x_i} - \vec{x_j})^T \mathbf{M} (\vec{x_i} - \vec{x_j})$$

By considering M as a positive semidefinite matrix M can be decomposed as follows :

$$M = L^T L$$

. Thus the Mahalanobis distance can be looked at as being a projection from an euclidean space to another euclidean space using the operator L :

$$D_{\mathbf{L}}(\vec{x_i}, \vec{x_j}) = ||\mathbf{L}(\vec{x_i} - \vec{x_j})||$$

. By using this projection, the same process can be applied to obtain the ranking metrics (MAP and P@5).

The Model

The model to solve this distance learning problem is based on three different optimization problems :

1. Mahalanobis metric for clustering(MMC). [9]
2. Pseudometric Online Learning Algorithm(POLA) . [10]

3. Neighborhood Component Analysis(NCA). [11]

The model formulates the parameter estimation as a convex optimization over the space of positive semidefinite matrices similar to the MMC. The margin by which the classifier is accurate for labeled examples will be maximized similar to POLA. And it is build to learn a Mahalanobis distance for optimization of K-NN classifier accuracy similar to the NCA.

The loss function

The end result of applying LMNN would be to efficiently attract the k nearest neighbors and push impostors outside a certain margin. This formulations leads to a two termed loss function. The penalization is defined as follows :

1. Impose penalty on large distances between each sample and the k nearest neighbors with the same class label.
2. Impose penalty on small distances between each sample and samples corresponding to different classes.

Before presenting the equations of the two terms of the loss function lets first give the notation that will be used.

A target neighbor is one of the k nearest neighbors for each sample that correspond to the same class of that sample. The notation for such neighbor is the following : $j \rightsquigarrow i$ indicates that $\vec{x_j}$ is a target neighbor of $\vec{x_i}$.
An impostor is a neighbor that violates the marge set by the distance of each sample. This impostor will have a different label than the observation : $\vec{y_l} \neq \vec{y_i}$.
The equation will be presented in terms of the linear transformation L of the input space.

Now that the terminology is presented, let us start by giving the equation of the first term of the loss function. This term is the one responsible of pulling the K-nearest neighbors corresponding to the same class as the observation and is given by :

$$\epsilon_{pull}(L) = \Sigma_{j \rightsquigarrow i} ||L(\vec{x_i} - \vec{x_j})||^2$$

The first term of the loss function is thus being put on the sum of the distances between the observation and the k nearest neighbors corresponding to the same class in the projected space.

The second term of the loss function will push the impostors outside a defined margin. It is defined as follows :

$$\epsilon_{push}(L) = \sum_{i,j \rightsquigarrow i} \sum_{l} (1 - y_{il})[1 + ||L(\vec{x_i} - \vec{x_j})||^2 - ||L(\vec{x_i} - \vec{x_l})||^2]_+$$

A factor y_{il} is introduced and it is equal to 1 if the observations i and l are from the same class and 0 otherwise. This factor will guarantee that this term will only affect the observations that are from different class than the observation i. In the equation, the margin that the observation should be pushed outside of it is defined as follows : $1 + ||L(\vec{x_i} - \vec{x_j})||^2$. From this margin, the distance between each observation and the one that correspond to different classes is subtracted. Only the positive value of the subtraction is taken to ensure that the penalization is being put on the observations that violates the margin. The sum of the distances is made on the entire space.

Now that the two terms are defined we define the loss function as the pondered sum of those two factors. The equation is given as follows : $\epsilon(L) = (1 - \mu)\epsilon_{pull(L)} + \mu\epsilon_{push}(L)$

Once the optimization problem is solved, The result will be the matrix L which will project the observations into a space that will better represent the data. Before applying the LMNN to the ground truth labeling problem, it will be applied to the true labeling problem to test its effect on the musical space for features extracted using the MFCC and the scattering.

1.2.3 LMNN applied to the true labeling problem

After preprocessing the data in the MFCC and the scattering space, the LMNN will be applied. At first the data will not be divided into a part for training and a part for testing. Rather the application of the LMNN will be on the hole data set and the results will be also on the hole data set.

features	p@5 before LMNN	p@5 after LMNN
16 class of instruments		
mfcc	86.89	87.09
scattering	93.87	99.99
32 class of instruments with variation		
mfcc	85.12	86.16
scattering	90.94	99.92
498 class of playing techniques		
mfcc	45.19	46.38
scattering	57.98	88.08

Table 1.2: Comparison between the P@5 before and after LMNN for feature extracted using the MFCC and the scattering with T=250ms

The first table shows the comparison between the P@5 before and after applying the LMNN. An improvement is noted for all the different type of classes for both features extracted using the MFCC and the scattering. The scattering continue to outperform the MFCC after the LMNN. For the two type instruments the scattering after the LMNN gives almost a perfect result. And for the last type of classes of playing techniques, the scattering after gives a remarkable improvement of 30.1%.

features	MAP before LMNN	MAP after LMNN
16 class of instruments		
mfcc	24.22	24.73
scattering	30.73	60.07
32 class of instruments with variation		
mfcc	22.29	25.27
scattering	28.07	56.88
498 class of playing techniques		
mfcc	8.78	9.61
scattering	20.41	44.42

Table 1.3: Comparison between the MAP before and after LMNN for feature extracted using the MFCC and the scattering with T=250ms

Again for the MAP, The scattering continues to outperform the MFCC. with a big improvement after applying the LMNN. This shows that the scattering contains the necessary variability for the problem of music search query.

To further show the importance of the LMNN, two division of train test is made on the space and the results are shown in the following tables for the 32 class of instrument with variation:

features	MAP before LMNN	MAP after LMNN
Division into 80% for train and 20% for test		
mfcc	23.24	24.54
scattering	31.00	60.70
Division into 50% for train and 50% for test		
mfcc	22.74	24.02
scattering	30.70	62.98

Table 1.4: Results of the MAP for the division of the space into a part for train and a part for test

The first table shows that for both the mfcc and the scattering an improvement was noted even with a split of 50/50 between train and test. However for the scattering the improvement was greater and it provided the same improvement without the division into train test.

features	P@5 before LMNN	P@5 after LMNN
Division into 80% for train and 20% for test		
mfcc	73.90	75.14
scattering	83.05	98.44
Division into 50% for train and 50% for test		
mfcc	81.17	81.86
scattering	90.02	99.31

Table 1.5: Results of the P@5 for the division of the space into a part for train and a part for test

The same can be noted for the P@5. The division was made on the entire space by taking 80% of each class for train and 20% for test(50% for train and 50% for test from each class).

Now that the effect of LMNN have been proven to be very effective on the scattering while it provides slight improvement for the MFCC, one last test should be made. This test is to prove that the effectiveness of the LMNN on the scattering is not due to the space volume rather than the variability this space contains.
The MFCC was extended in the following way :

- add the delta coefficients (a total of 12 coefficients)

- add the delta delta coefficients (a total of 12 coefficients)
- Multiply the pair wise coefficients, the total will 666 coefficients

$$((36 * 35)/2) + 36 = 666$$

Then the same number of coefficients is taken from the scattering. The results are given in the following table :

features	MAP before LMNN	map after LMNN	p@5 before LMNN	p@5 after LMNN
mfcc	10.38	13.35	58.01	77.80
scattering	28.43	51.35	92.85	99.85

Table 1.6: Table showing the comparison between the scattering and the MFCC with the same number of features (666) before and after applying the LMNN

The results shown in the table above were anticipated. The scattering transform outperformed the MFCC after applying the LMNN even when the space of MFCC features was expended. The MFCC precision degraded with the additional features. The LMNN could not rise the precision of the MFCC even with a high space of features.

Now that the LMNN has proven to be efficient for the musical search query problem based on the physical labeling of the observation, it is time to see its effect on the ground truth problem.

1.2.4 LMNN applied to the ground truth problem

To apply the LMNN to a series of multiple labeling opinion some alternation should be made. It should be noted that this problem was not addressed before. Many methods were proposed and tested to take advantage of the LMNN to make it adaptable to the problem :

1. **Summing the distances**

   The first method that was proposed is to sum over the distances. Two ways of summing the distances was used :

   The first one is to sum over the matrices L of projection obtained by the LMNN. To better understand the difference between this approach and summing directly

the distances let us see the equations below :
Summing the projection matrices L can be written as :

$$L = \sum_{k=1}^{nobs} L_k$$

Thus the new distances will be :

$$(\vec{x_i} - \vec{x_j})^T \sum_{k=1}^{nobs} L_k^T \sum_{k=1}^{nobs} L_k(\vec{x_i} - \vec{x_j})$$

$$= (\vec{x_i} - \vec{x_j})^T \sum_{k=1,w=1}^{nobs} L_k^T L_w(\vec{x_i} - \vec{x_j})$$

$$= (\vec{x_i} - \vec{x_j})^T (\sum_{k=1,w=1,k=w}^{nobs} L_k^T L_w + \sum_{k=1,w=1,k\neq w}^{nobs} L_k^T L_w)(\vec{x_i} - \vec{x_j})$$

$$= (\vec{x_i} - \vec{x_j})^T \sum_{k=1,w=1,k=w}^{nobs} L_k^T L_w(\vec{x_i} - \vec{x_j}) + (\vec{x_i} - \vec{x_j})^T \sum_{k=1,w=1,k\neq w}^{nobs} L_k^T L_w(\vec{x_i} - \vec{x_j})$$

$$\sum_{k=1}^{nobs} d_{M_k}(\vec{x_i} - \vec{x_j}) + \sum_{k=1,w=1,k\neq w}^{nobs} (L_k\vec{x_i} - L_k\vec{x_j})^T(L_w\vec{x_i} - L_w\vec{x_j}) \quad eq.1$$

The first part of eq.1 is the same as summing the distances directly, the second part however do not have a physical explanation. A sum over the distance is better understood. However a comparison is provided for further study since it achieved good results.

features	Sum	MAP	MAP after LMNN	p@5	p@5 after LMNN
mfcc	Sum over L	50.74	55.02	55.66	60.99
mfcc	Sum over distances	50.74	54.24	55.66	60.04
scattering	Sum over L	40.17	49.63	44.01	61.85
scattering	Sum over distances	40.17	51.46	44.01	65.57

Table 1.7: Results of summing the distances and summing the projection matrix L

Lets look at all the different cases that might be encountered to better understand the improvement provided by the sum over the distances. In the following study, the focus is only on two different labeling opinions from the 32. The figure 1.2 show an example of applying LMNN on two different labeling opinion for the same samples with three nearest neighbors considered. The pairwise distance matrix before applying the LMNN is given by :

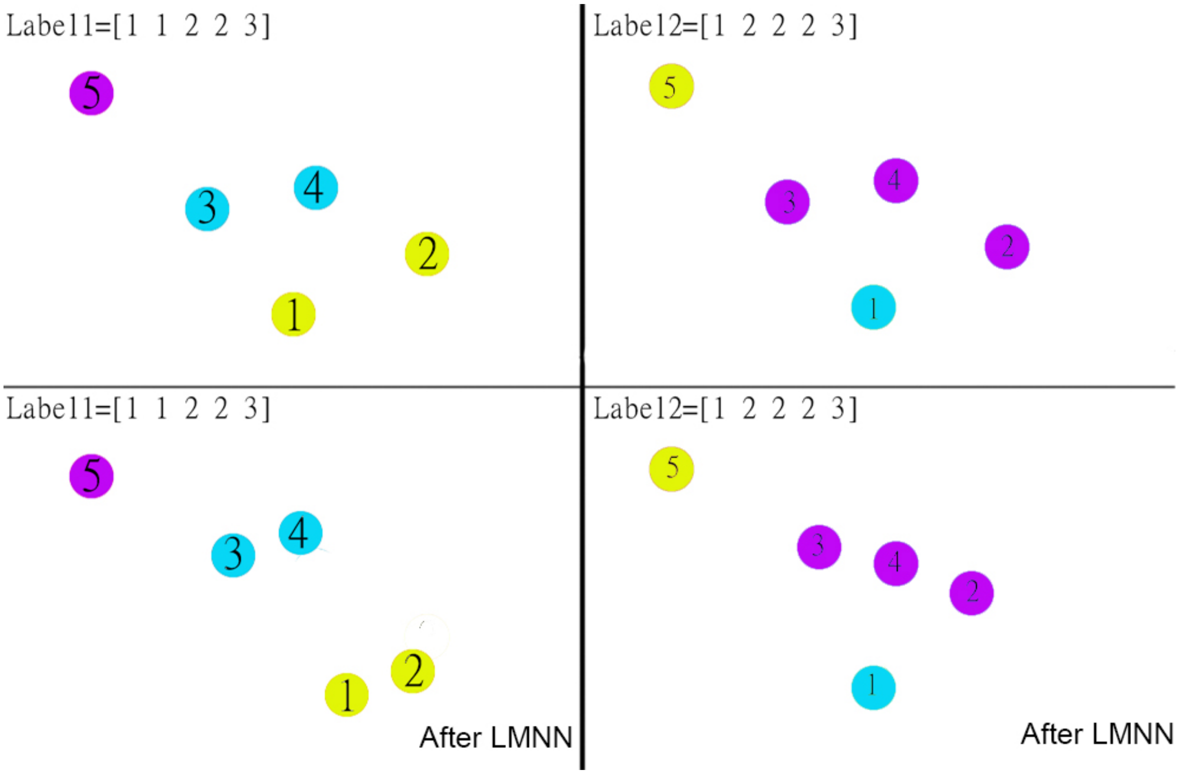

Figure 1.2: LMNN applied on two different labeling opinion

$$\begin{bmatrix} 1 & d_{12} & d_{13} & d_{14} & d_{15} \\ d_{21} & 1 & d_{23} & d_{24} & d_{25} \\ d_{31} & d_{32} & 1 & d_{34} & d_{35} \\ d_{41} & d_{42} & d_{43} & 1 & d_{45} \\ d_{51} & d_{52} & d_{53} & d_{54} & 1 \end{bmatrix}$$

After applying the LMNN with sum over distances Four different cases can be presented :

(a) The two labeling opinion agrees on the similarity.
The application of LMNN will conduct to a smaller distance if the labels are similar. If the observations are target neighbors both distances will be smaller in the new space for both labeling provided by the two different users. Observation 3 and 4 in figure 1 illustrate that idea. In this case the two distances D_{M_1} and D_{M_2} will be smaller than the original distance in the space before the application of LMNN. Let us now go back to update our new pair wise distance matrix:

$$D_{M_1}34 < d_{34}\ and\ D_{M_2}34 < d_{34}$$

the sum of those two terms is thus :

$$D_{M_1}34 + D_{M_2}34 < 2d_{34}$$

.

(b) The two labeling opinion agrees on the non similarity.
The same analogy can be made for the case were both opinion agree that the two observation belongs to different classes. If we take for example the observations 1 and 3 we will have at the end

$$D_{M_1}34 + D_{M_2}34 > 2d_{34}$$

.

(c) The two labeling opinion dos not agrees on the similarity.
If the two labeling opinion disagree on the similarity between two observation, in one space we will have smaller distance while in the second we will have bigger distance. For example if we look at samples 2 and 4 in figure 1, we can see that in the new spaces distance between those two observations is bigger for the new space with LMNN based on labeling opinion 1, and bigger in the space with LMNN based on labeling opinion 2. we will thus have :

$$D_{M_1}24 > d_{24} \ and \ D_{M_2}24 < d_{24}$$

. Let us introduce two factors $\alpha > 1$ and $\beta < 1$ this implies :

$$D_{M_1}24 = \alpha d_{24} \ and \ D_{M_2}24 = \beta d_{24}$$

$$D_{sum}24 = \alpha d_{24} + \beta d_{24}$$

.

(d) An observation is not affected in both spaces by LMNN.
If the distance between two observations that are not target neighbors for the LMNN is bigger than a certain margin the distance will not be affected directly. This distance will slightly change because the observation themselves have been displaced in their own small margin. We can thus make the assumption that the variation in distance in the new space is negligible and the for example the distance between observation 1 and 5 will be

$$D_{sum}15 = 2d_{15}$$

Now that we have formulated all the factors we can construct our new matrix based on the sum of the projection matrix L. To simplify the notation we will take on factor to represent the different factors in case 3, for example let $2A = \alpha + \beta$.

$$\begin{bmatrix} 1 & 2Ad_{12} & > 2d_{13} & > 2d_{14} & 2d_{15} \\ 2Ad_{21} & 1 & 2Cd_{23} & 2Bd_{24} & 2d_{25} \\ > 2d_{31} & 2Cd_{32} & 1 & < 2d_{34} & 2d_{35} \\ > 2d_{41} & 2Bd_{42} & < 2d_{43} & 1 & 2d_{45} \\ 2d_{51} & 2d_{52} & 4d_{53} & 2d_{54} & 1 \end{bmatrix}$$

Let us now examine two cases, one were we have an agreement and one were we have a disagreement :

(a) Case were we have agreement between different opinions.
It should be noted again that we are searching to compare distances and not the exact value of the distances. So let us take two distances were in the first both users agree that two samples belongs to the same class and in the other they agree that they don't. for example we can take d_{13} *and* d_{34}. In the space before the application of LMNN we had : $d_{13} - d_{34}$ in the new space we will have a difference between a value that is twice bigger than the first distance and a value that is twice smaller than the second distance. So we have been able to have a space that make the difference in distances bigger if we have two exact opinions on labeling. And for both labeling opinions the error will be minimized.

(b) Case were we have disagreement between opinions. Let us take again distance $d_1$3 but this time let us compare it with $d_2$1. Let us take F as a value that is bigger than 1 to represent the factor by which in the new space the distance between samples 1 and 3 is bigger. we will thus have in the new space $2Fd_{13} - 2Ad_{d21}$. We now that the first distance is going to be bigger than the original distance between samples 1 and 2. On the other hand the distance between 1 and 2 in the new space will depend on the original distance between 1 and 2. So if the distance in the new space is smaller than in the original space, labeling based on first user will have smaller error than labeling based on second user. But since the first factor is well optimized we will have optimization in both case.
This idea that the "winner" between the two opinion is based on the representation in the original space is very important. And it will be essential in a case were we have more than two opinions, since the probability that two samples will be considered similar by the users is related to the physical aspect of the sound.

2. **Pondering the sum over distances**

In this case, the sum has been preponderated based on a factor of success of each labeling opinion relatively to the other labeling opinions. The success is based on a precision factor obtained by applying the normalized mutual information. The distances will thus be multiplied each one by the precision of the labeling opinion over the entire labeling opinion space. The results are as follows :

features	MAP	MAP after LMNN	P@5	p@5 after LMNN
mfcc	50.74	54.35	55.66	60.51
scattering	40.17	52.61	44.01	66.89

Table 1.8: Results of pondering the sum

Pondering the distance provides inconsistent results. For the MFCC the effect is minimal whether for the scattering it provides improvement of around 1%.

3. **Normalizing the distance**

   In this method, the distances has been normalized before performing the sum. This is to reduce the effect of big distances provided by one user over small distances provided by the rest for the same two observations. The results are presented in the following table :

features	metrics	MAP	MAP after LMNN	P@5	P@5 after LMNN
mfcc	normalized distance sum	50.74	54.31	55.66	60.29
mfcc	normalized pondered distance sum	50.74	54.39	55.66	60.09
scattering	normalized distance sum	40.17	54.03	44.01	67.33
scattering	normalized pondered distance sum	40.17	54.30	44.01	67.63

Table 1.9: Results of normalizing the distances

   Normalizing before effecting the sum proved to be beneficial for the scattering and not for the MFCC. This is due to the high conventionality of the scattering that might have an affect over the distances. This effect will be reduced by applying a normalization over the distances.

4. **Class Matching**

   The last method that was tested is to apply a class matching over the space of different observation. This is done using the NMI(normalized mutual information) technique. After matching the space of opinions, all the classes will be well aligned. Now using a majority vote over the space of observations, only one label will obtained. The LMNN will be thus trained only once on the obtained labeling. This will induce a shorter amount of training time and a faster algorithm. The results are given in the following table :

features	map	mapafter	pat5	pat5after
mfcc	50.74	55.34	55.66	61.77
scattering	40.17	49.41	44.01	60.75

Table 1.10: metrics: lmnnclassmatching, type: instrument16GT

   The class matching proved to be beneficial for the MFCC over the sum of distances. For the scattering summing the distances is better but with dramatically more computation time : For the class matching 21.01 seconds and for the sum of distances 2316.59 seconds.

1.3 Conclusion

This paper provided a study of two problems of music search query. The first one based on the physical aspect of the sound. The results obtained for this part and the preprocessing technique for the scattering are novel work. The precision obtained on the search query for musical purposes proved that the standard deviation of the scattering should be studied in order to achieve state of the art results. This technique of preprocessing should be tested for different problems in the domain of audio and images. In the second part, the problem of studying multiple user opinion in order to find a space that respects all the opinions was addressed. This novelty work has put some new foundation for solving the problem.

Bibliography

[1] Beth Logan *Mel Frequency Cepstral Coefficients for Music Modeling.* 2000

[2] J. Andén and S. Mallat. *Multiscale scattering for audio classification.*. ISMIR 2011

[3] J. Andén *Time and frequency scattering for audio classification.* January 7, 2014

[4] Hermann Ludwig Ferdinand von Helmholtz *On the sensations of tone as a physiological basis.* 1895

[5] Stephan Mallat *Recursive interferometric Representations* 18th European Signal Processing Conference 2010

[6] Perfecto Herrera-Boyer and al. *Automatic Classification of Musical Instrument Sounds.* Journal of New Music Research 2003

[7] Yan Maresz and al. *Ircam solo instruments UltimateSoundBank reference guide*

[8] K. Q. Weinberger, L. K. Saul. *Distance Metric Learning for Large Margin Nearest Neighbor Classification.* Journal of Machine Learning Research (JMLR) 2009

[9] P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell *Distance metric learning, with application to clustering with side-information.* Cambridge, MA, 2002.

[10] Shalev-Shwartz, Y. Singer, and A. Y. Ng. *Online and batch learning of pseudo-metrics* Banff, Canada, 2004.

[11] Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov. *Neighbourhood components analysis* Cambridge, MA, 2005