Nightingale song clustering

Elisa Ancarani, Irene Gentilini, Jens Hartsuiker

Nightingale song clustering

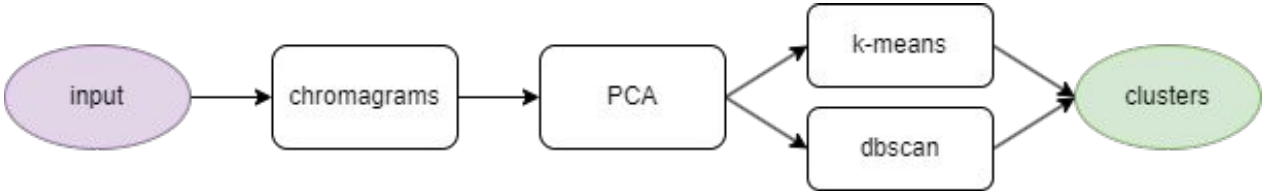
- The dataset is a set of nightingale song tracks
- The problem consists of finding the number of labels of tracks
- To find it we need to cluster the songs
- We tried three methods
  - Baseline
  - Autoencoder
  - Contrastive learning

https://github.com/LIA-UniBo/NSCNet/blob/main/docs/AII%20-%20Project%20Report.pdf

Baseline

NSCNet inspired the baseline method:

- The songs were encoded using **chromagrams**
- The dimension of the input was reduced with **PCA**
- Song clustering with both **K-means and DBSCAN**



- Issues with DBSCAN:
  - really small data points from PCA results in a single cluster
  - tried also DBSCAN directly on the chromagrams but data points are still to small for the model
- Chromagrams are a good option, but Mel-spectrograms are more represented in the literature"

Autoencoder

- We took inspiration from NSCNet who used a VAE
- We used an autoencoder
- As the autoencoder does not constrain to a certain distribution, but the performance is still not ideal

Autoencoder Architecture

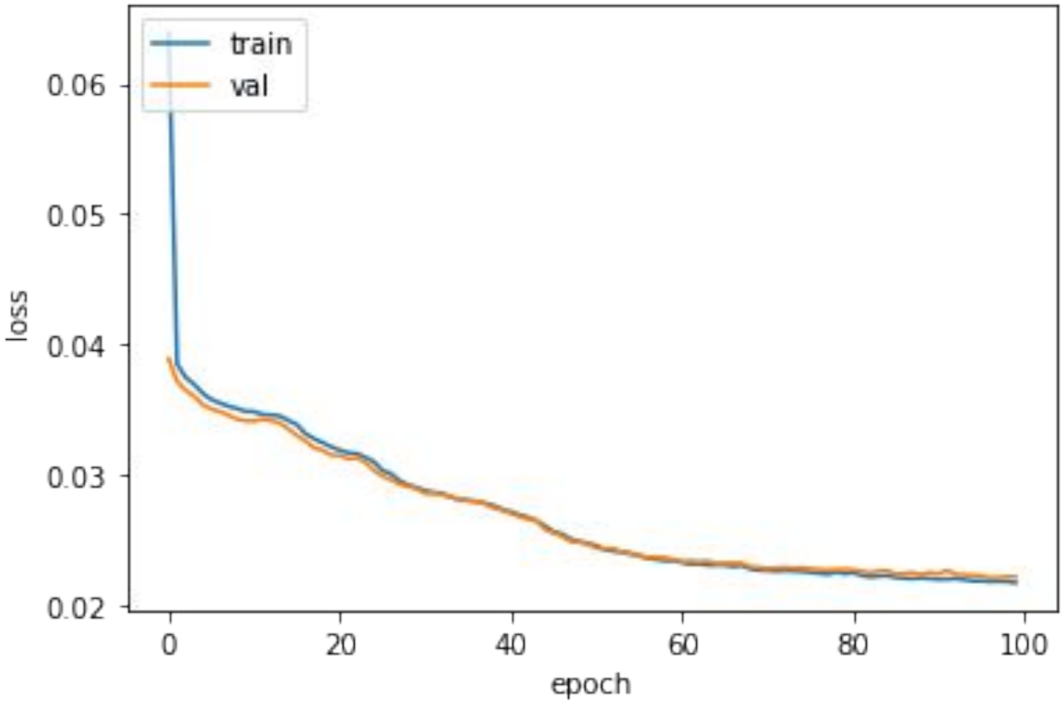
(12,474)

3 conv layers, stride == 2 (3,58) still 100% construction

Dense layers to a vector of length 10, 25, 35, 50, 100 or 600

PCA has a feature space of 12*12 = 144

MSE Training Loss



Calinski-Harabasz Score

Some experimenting in this article:

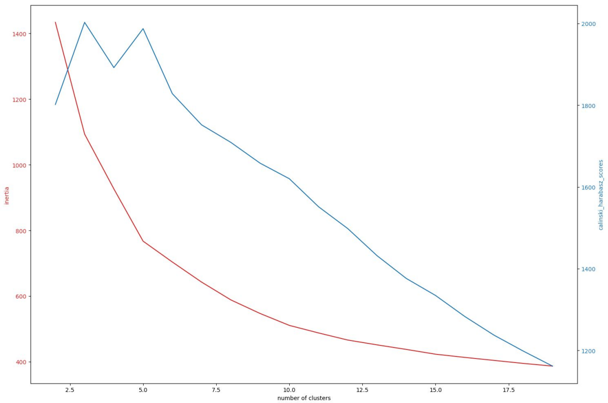
[Are You Still Using the Elbow Method? | by Samuele Mazzanti | Feb, 2023 | Towards Data Science]

Score based on division:

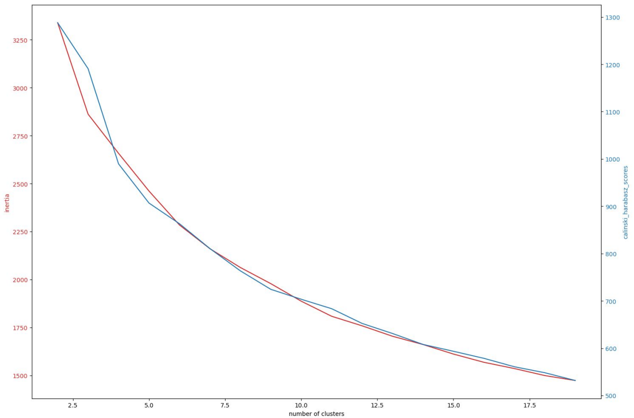
- distance cluster centroids to global centroid (separation) by ->
- distance cluster items to cluster centroids (cohesion)

Evaluating different cluster amounts [1,2,...,20] with different sizes of feature vectors

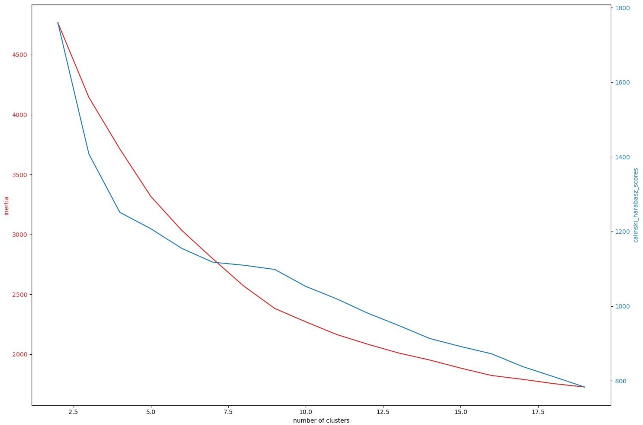
10



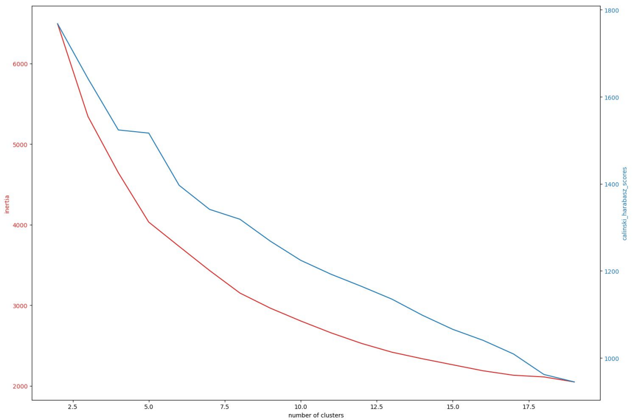
25



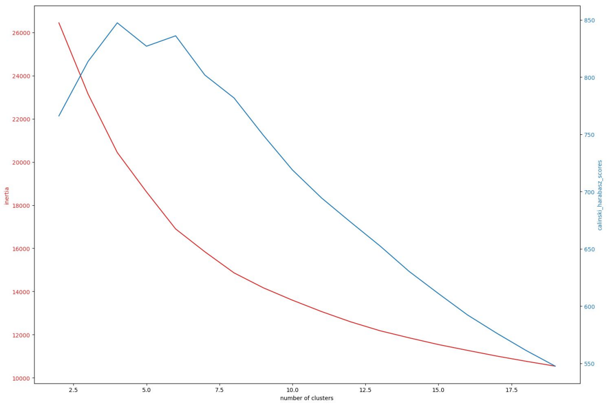
35



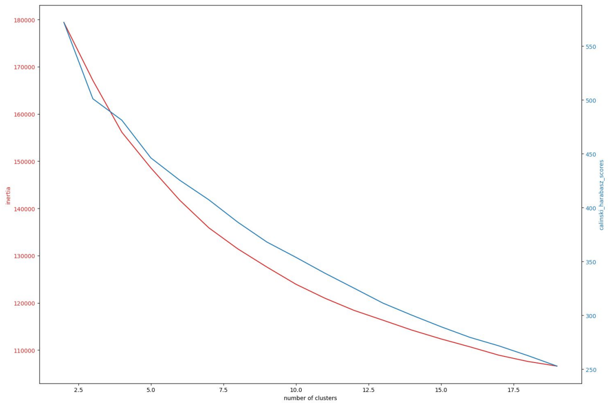
50



100



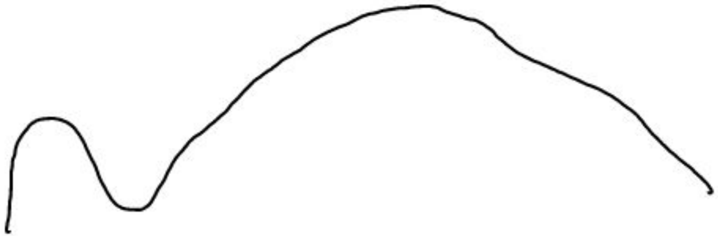
600



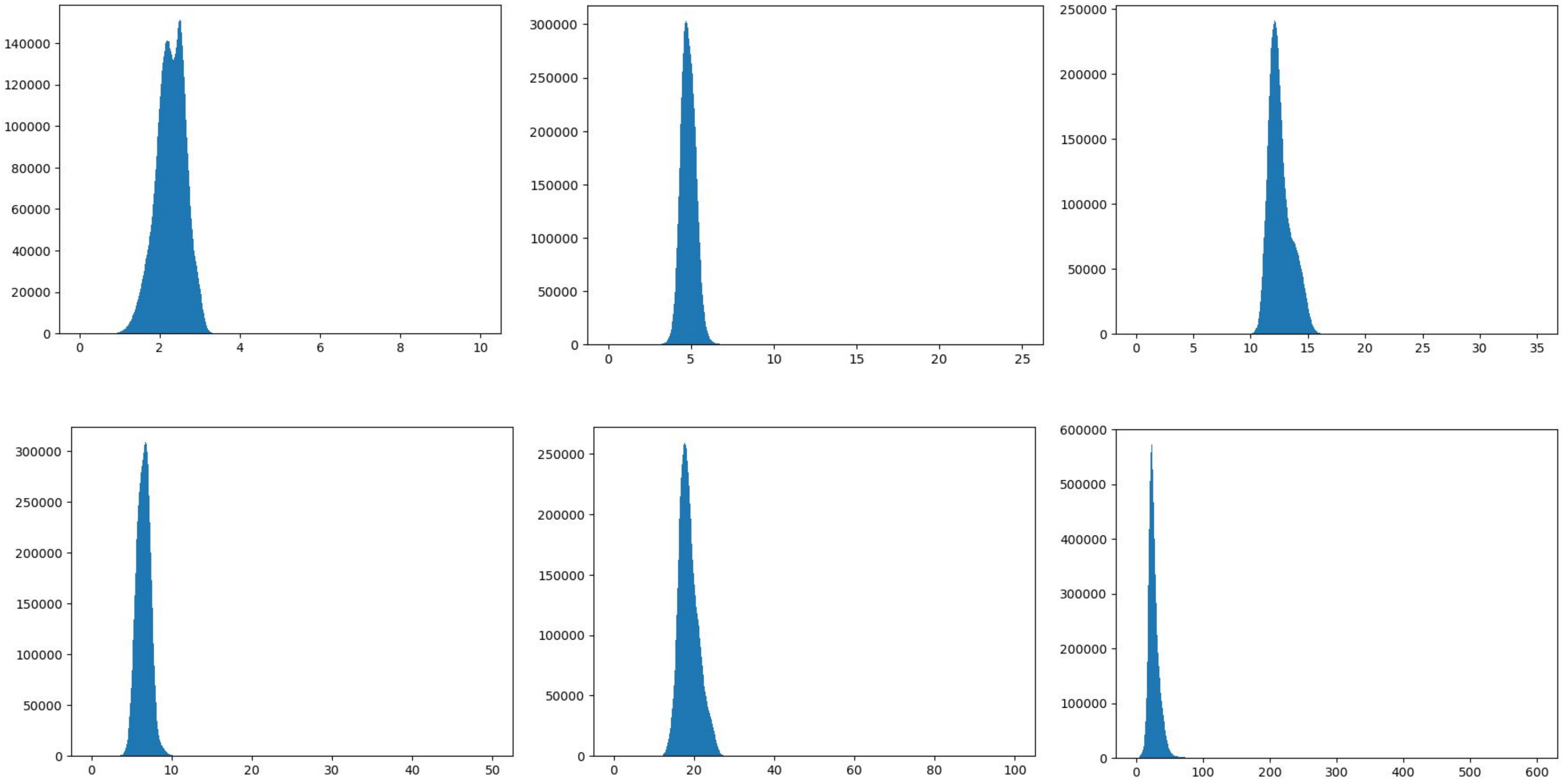
What would an ideal distance distribution look like?

2D Feature space

Distance distribution



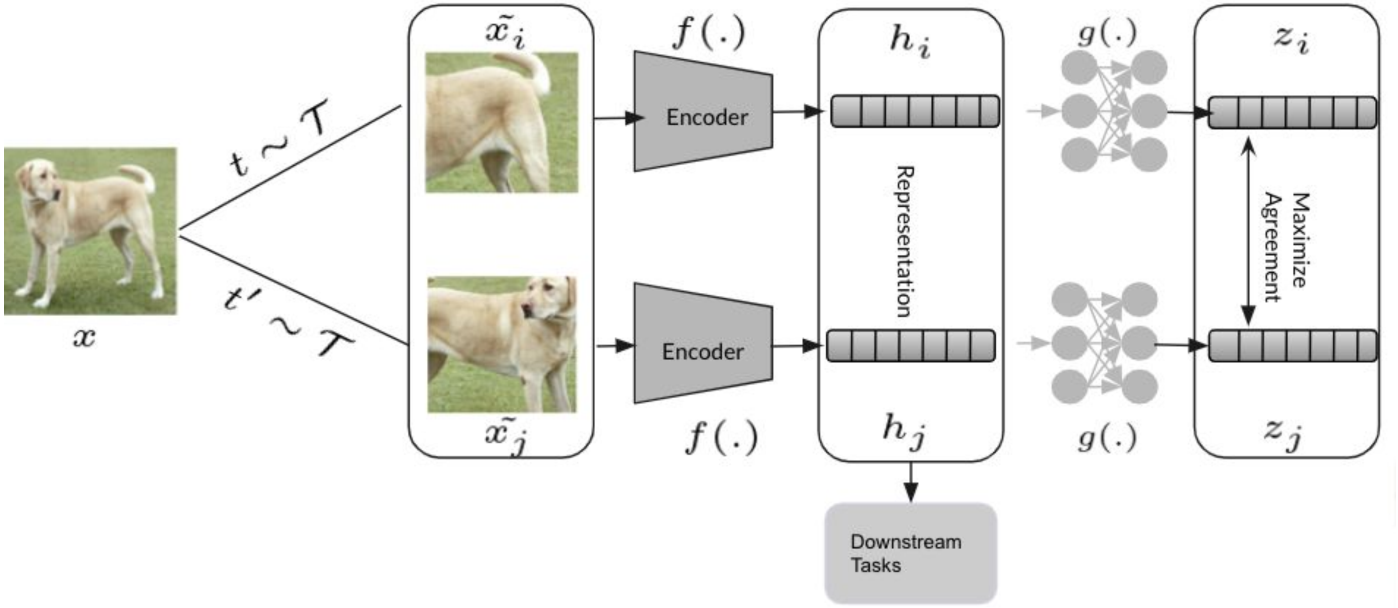
Distributions of distances of encoded-vectors
Distance between pairs on x-axis vs number of pairs on y-axis



Variational Autoencoders force the encoded vectors in a normal distribution, Autoencoders don’t, but it ends up happening anyway https://arxiv.org/pdf/2002.05709.pdf

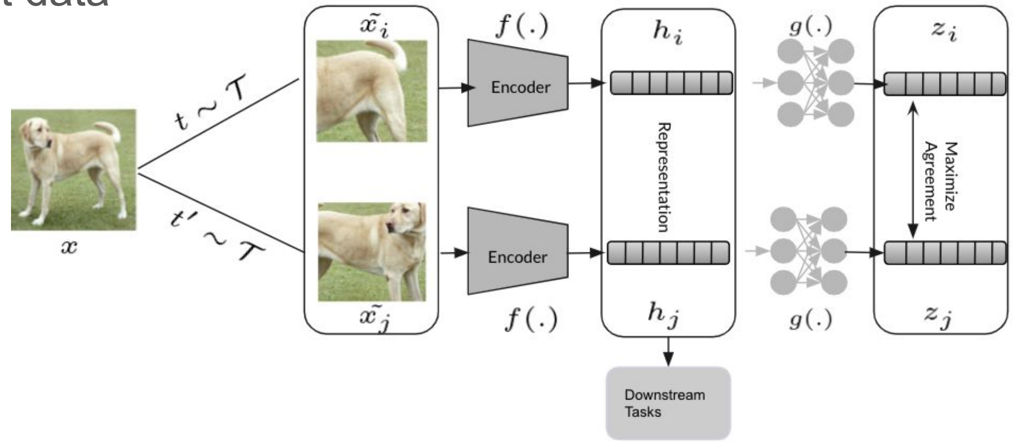
Contrastive learning

- We worked on the **SimCLR** framework

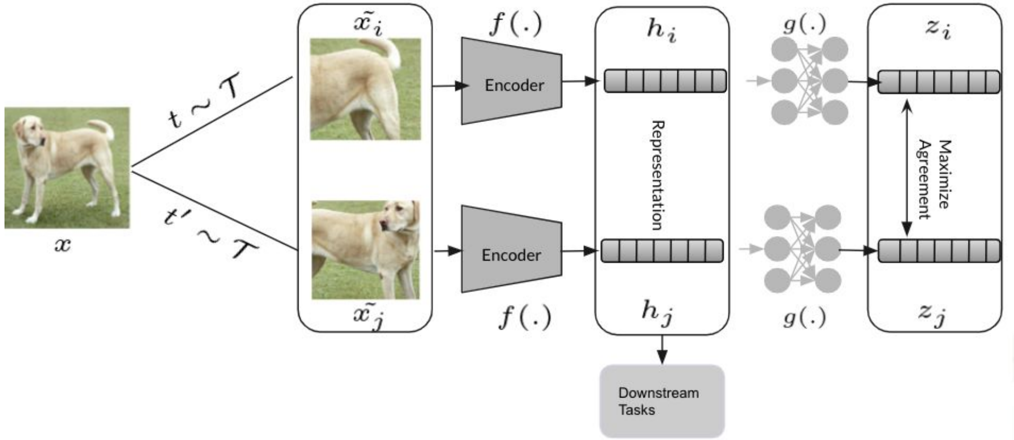


- **Self-supervised learning method**
    - in which a model learns to differentiate between similar and dissimilar pairs of data points.
- **Similar images are mapped together**
    - a siamese neural network is trained to map two different augmentations of the same instance close together in an embedding space
    - We used the NT-XEnt
- **Different images are mapped further apart**
- Model learns useful representations that can be transferred to downstream tasks.
- **K-means** is used to cluster the output data

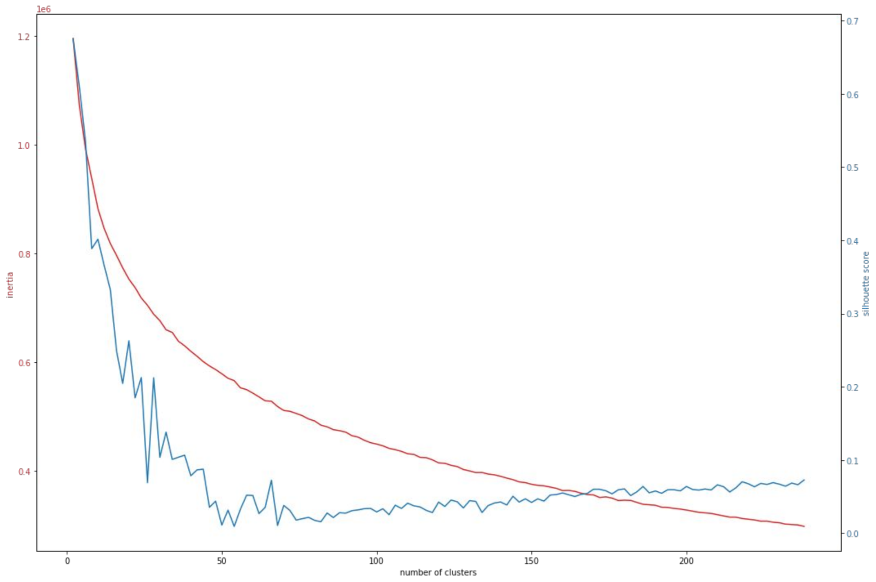
$$l_{i,j} = -\log \frac{\exp(\mathrm{sim}(\mathbf{z_i}, \mathbf{z_j})/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\mathrm{sim}(\mathbf{z_i}, \mathbf{z_k})/\tau)}$$



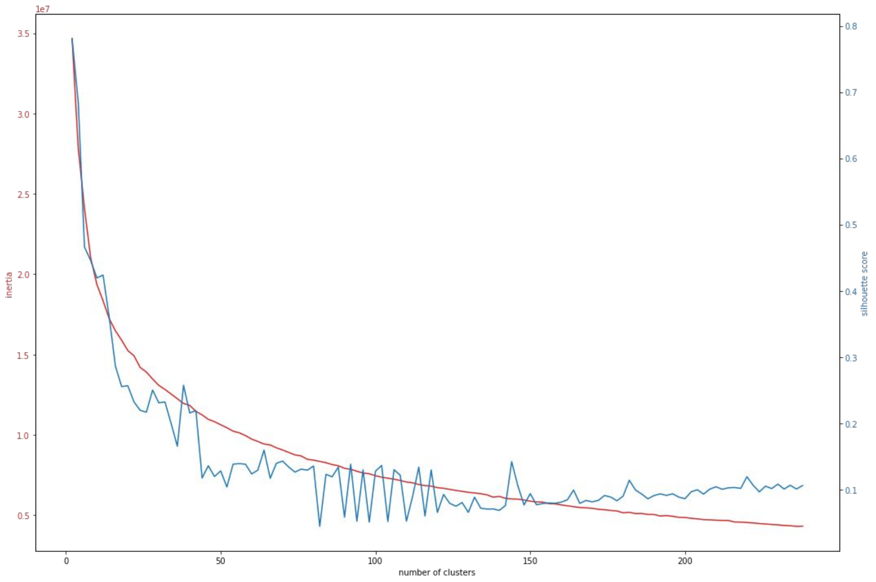
- The data is **unlabelled**
    - so we do not know which images are similar
- To simulate labels, we take an image and we augment it, with:
    - random affine transformation (translation only)
    - random erasing
- The augmentations should be kept together
    - Because they represent two images of the same class.
- **ResNet18** and **ResNet50**
    - were trained to learn visual representation of these images
    - we also tried other versions of Resnet



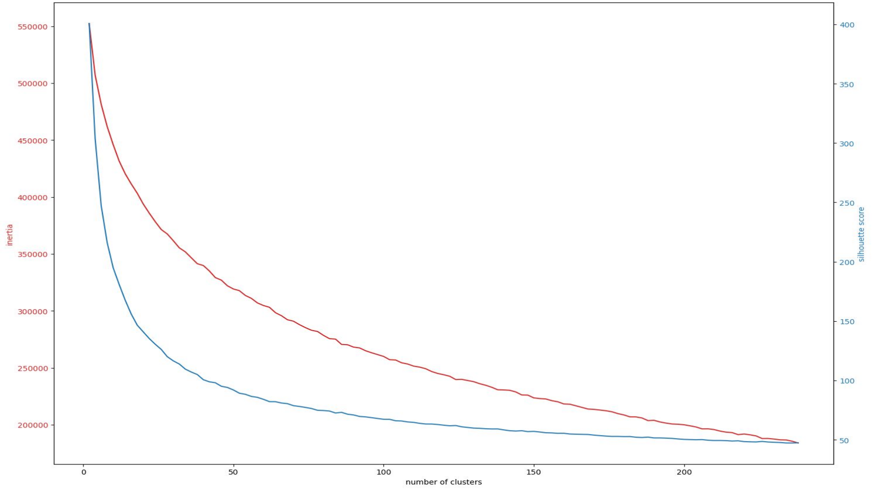
ResNet18



ResNet50

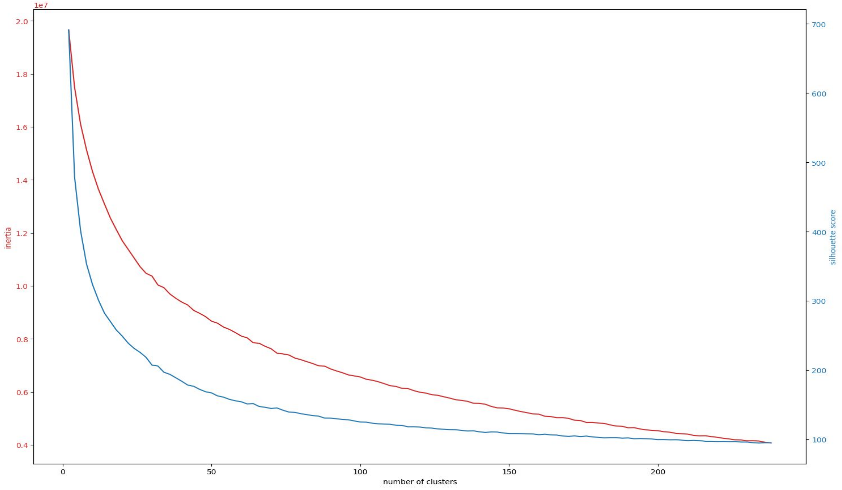


Our Contrastive Learning Method is unable to distinguish between the number of clusters



ResNet18

Here's the results with Calinski-Harabasz score



ResNet50

Contrastive learning - Results

- Contrastive learning does not lead to satisfactory results yet

However:

- Contrastive learning works well when a **larger number of data** is available
  - data should be more than 20,000 instances
  - if it were possible to collect more data this method would be very valuable.
- With **more augmentation** results could still improve
- **Data could be augmented by separating tracks into segments.**
- An end to end approach called **contrastive clustering** could be explored.

https://arxiv.org/pdf/2009.09687.pdf