Mikko Haikonen*, Hannu Toivonen, and Jerker Björkqvist*

Stress test evaluation of classifiers for audio based bearing diagnostics in HVAC machines

Abstract: Condition based monitoring aims to measure the situation of industrial equipment or processes. This is important for detecting equipment changes or damages. Servicing mechanical devices has traditionally been relying heavily on aural observations. Accurate audio diagnostics require profound understanding and experience. Such skills are not always available on site, especially at remote locations. Therefore automated or assisted diagnostics are very interesting topics in predictive maintenance. Automating the diagnostics may involve using machine learning classifiers. Classifiers are trained using finite training sets. Trained classifiers are then expected to diagnose situations in varying operating conditions. If the classifier is being overfit, it would not perform well in production use. In this work we investigated ways to stress test the audio classifiers by using compression techniques to degrade audio quality for evaluating robustness of trained audio classifiers. We found a practical procedure to provide an additional measure to prevent selecting an overfitted classifier.

Keywords: Diagnostics, Machine learning, Bearing monitoring, Audio classification, HVAC maintenance, compression

***Corresponding Author: Mikko Haikonen:** Solita Oy, E-mail: mikko.haikonen@solita.fi
Hannu Toivonen: Åbo Akademi University, E-mail: hannu.toivonen@abo.fi
***Corresponding Author: Jerker Björkqvist:** Åbo Akademi University, E-mail: jerker.bjorkqvist@abo.fi

1 Background

Condition based monitoring consists essentially of measurements and diagnostics of industrial processes. It is appealing to be able to monitor an industrial process continuously for 24 hours a day seven days a week. Occasionally it would be feasible to store audio measurement data for later processing and analysis. Data storage, however, has costs involved. These costs increase with the amount of data stored. Therefore it is desirable to be able to reduce the storage need by compressing the measured audio data.

At the same time we need to maintain the quality of diagnostics. That means keeping the classification accuracy as high as possible.

We can compress the data in order to reduce its size. If we use a lossy compression algorithm we will loose information. If the lossy data are then applied to perform classification we would like to know how the compression affects the classification results.

When applying machine learning classifiers in condition based monitoring there are two practical problems. The first one is the overfitting of classifiers. That may cause problems whenever the situation at the monitored process changes after training data have been collected and the classifier has been trained. The second problem is the cost of audio data storage. Data compression is one of the ways to decrease data storage costs. However, it is not always clear how much information we can afford to loose in order to maintain an acceptable classification performance.

2 Objectives

The objective of our study was to evaluate classifiers in their capacity to detect the relationship between the selected audio features and bearing changes. Audio samples from HVAC (heating, ventilation, and air conditioning) system were used for classification experiments. We refer to individual audio recordings as samples. The main focus in the evaluation was to prevent overfitting. The ability to correctly classify audio signals measured before and after overhaul was our primary objective. Attention was also paid to the computational cost of the classifiers. Our main goal was to provide additional sanity check mechanisms to avoid selecting a classifier which is overfitted. For this we examined how lossy compression of input data affects the classification performance. We also used additional test samples recorded at a number of rotational speeds of the system under test. This is how we estimated the reliability of k-fold cross validation results.

To summarize, we examine the following questions in this study:

– How does the audio quality affect the classification?

- What would be the appropriate level of audio compression in classification application?
- Which combinations of features and classifiers should be to selected?

3 Methods

We used audio samples collected from an HVAC system before and after overhaul. In the overhaul one of the two drive and fan units was replaced with a new one. Audio signals from air- and contact microphones were recorded. To avoid excessive computations the amount of data was reduced by random sampling of training and testing audio files. Our classification task was to determine for each individual audio sample whether it was recorded before or after the overhaul.

In our example we assigned each testing audio sample a label "before" or "after" based on whether it was recorded before or after the drive unit change. Those labels can be referred to as ground truth. In other words they describe the situation from which the data is being collected. That situation often reflects a property of a system under test. That property may be e.g. whether the system being healthy or faulty. Those kinds of labels are subjective, and therefore we wanted to use a label that is objective, such as whether the sample originates from before or after the overhaul.

In our study we did make use of the information loss that lossy speech encoding causes. We stress tested the trained classifiers by classifying test data which were weakened by applying a lossy data compression at different quality levels. The data were being compressed before computing features for that test sample.

In this way we learned how the lossy compression affected the classification results for different classifiers and feature sets.

3.1 Audio features

A feature can be thought as a statistic or key performance indicator (KPI) that represents some essential property of an audio sample. Classifiers were trained using those features for training. There was one trained classifier per each classifier type and feature set.

In audio related classification tasks it is fairly typical to resort to well known sets of audio features that have been reported in the other studies. At high level, the features can be grouped under e.g. temporal, spectral, cepstral and chroma feature spaces. Those features may include indicators like energy, zero-crossing rate, (spectral) centroid, entropy, crest factor, etc. [1]. The benefit of using those well known features is that they are known to capture some important properties of audio samples.

While those well-established features provide rich and interesting information about the audio and its source, they also have some downsides. If the model uses a large mixture of those statistics weighted and selected by possibly complicated rules, it would be a challenge to find a link between the classification results and physical realities of the system under test. In addition, it may also be necessary to repeat some of the feature-engineering if and when the system under test changes.

The audio classification features we selected attempt to reduce the need for additional feature engineering if the detection task changes. We used a feature space that allows linking the classification results back to the physical process. Therefore we used features like spectrum (DFT), autocorrelation (ACF) and partial autocorrelation (PACF). A particular benefit of ACF and PACF features is their close link to data generating process characteristics [2]. Autocorrelation feature can also bring insights into summative- and modulation effects [3]. Both autocorrelation and partial autocorrelation are also closely linked to linear predictive coding used in human speech processing [4].

The only hyper parameter of our feature sets is the number N of features being used. In autocorrelation and partial autocorrelation case, it is the maximumum lag for which the coefficients are being used. In FFT case it stands for the number of frequency bins that is used. We used the empirically selected value N=64. It provided enough reach to the low frequencies for the autocorrelation type of features, while providing reasonable frequency resolution for FFT features. Our attempt was to keep N as low as reasonably possible, because unnecessarily high values of N would be a form of overfit which we wanted to avoid. We kept the value N constant to keep managing and analysing results relatively simple.

One trade-off that N involves is the handling of the first (lag 0) autocorrelation coefficient. It always equals 1 for lag value zero. We kept that unnecessary value in feature sets, despite the fact that it caused zero variance for that feature. The reason for including this (unnecessary) value is that in otherwise, omitting lag 0 and taking including an additional lag value from higher lags, would have given different frequency information for ACF and PACF feature sets. That difference could in some unlikely cases cause differences in performance of

classifiers if the value N is relatively low and the system dynamics have not attenuated sufficiently from PACF responses at the lag N+1.

By this feature selection we also aimed to avoid having two-dimensional feature sets for each audio sample. Keeping the features in one dimension reduces significantly the feature-engineering work and the need of selecting additional hyper parameters for the audio processing purposes. Restricting ourselves to one dimensional feature vectors eliminated the need to split the input audio into time slices or frames, eliminating the selection of frame length parameter and possible processing window types.

We could say that the feature selection we have done is positioned somewhere between system identification and machine learning [5].

Since analysis of audio samples and speech analysis have many similarities it is very natural to use speech compression as a vehicle for stress testing the classifiers and consider audio samples as "machine speech".

3.2 Experiment setup

In this study we used an affordable and compact Zoom H6 audio recorder. As microphones we used the recorder's air microphones and an external Schertler Unidyn P48 active contact microphone. The audio was collected at 44100 samples per second and stored with 16 bit resolution. The sampling rate and resolution were kept moderate in order to avoid excessive storage and computations.

3.3 System under test

The HVAC machine fans and mechanics are enclosed inside steel chassis with a removable top lid. There are two blower units inside the chassis. The HVAC machine was Enervent LTR-6. It has a 1210 mm wide, 670 mm high and 658 mm deep sheet metal chassis. The removable lid of the chassis was considered most suitable surface to mount the contact microphones on. The lid is 1210 by 658 mm in size. For measurement purposes, adhesive tape was attached, on which it was easy to mark the positions. Positions were arranged into rectangular grid of 66 locations. The grid consisted of six parallel rows, 100 mm apart, each containing 11 marks 100 mm apart. Markings of measurement positions were used in order to record corresponding locations for before and after scenario. For mounting the microphone to this lid, we used the slightly adhesive putty that comes with the microphone. In retrospect, it would have been better to use standard length strips of two-sided adhesive tape, since the putty gradually wears out and the way microphone is attached to surface changes within the measurement batch. The adhesive putty was replaced for the separate batches.

According to human judgement most of the excess noise disappeared after the blower unit replacement. The unit has not been dismantled for mechanic inspection, but is kept in case there would be need to take any mechanical control measurements. We attempted not to make judgement of the blower condition before the overhaul. The only fact we know is that it was replaced.

We can not state anything about the relative condition of the second blower unit. That unit was not changed. Whatever our subjective judgement of the condition of any of those two units, we are not able to quantify them with measurements.

3.4 Classifier selection

We selected models that have different properties in order to understand how sensitive different type of models are to audio quality changes. Recursive partition (RPART) model was selected since it is very easy to interpret. Vector quantization is used in speech applications as mapping of audio features into code words [4] and as similarity metric in song identification [6] and therefore we used learning vector quantizer (LVQ) as model candidate.

For classification we used random forest, support vector machines (SVM), learning vector quantization and decision tree. We used relatively simple models to maintain the link between results and a physical process. That link is useful when the classification results are taken into operational use. Support vector machine has good generalization performance and is tested to perform well against the most popular classifiers [7] and is being reported being used in vibration analysis applications [8].

3.5 Model validation

The process of training and cross-validating classifiers was used as tool to understand how well the selected feature space reflects the mechanics and the class of samples. Our stress test complemented the results from traditional cross validation results.

Samples are divided into three sets: training set, test set and validation set. The training set is used for model training and the validation set is used to guide the model training procedure. We used 60% of the training set for model training and 40% for validation. Models were trained using the R language package caret [9]. Th caret package offers a common interface for model training and testing.

Traditional cross validation methods including K-fold [10] cross validation were used [11]. We used 5 folds that were repeated 15 times. Our key idea was to extend e.g. K-fold cross validation by stress testing the classifiers. In stress testing the trained classifiers are stressed by testing them with samples of varying audio quality. For each classifier and validation sample, we used several levels of audio quality to examine how the prediction accuracy reacts to this stress. The purpose of the stress test was to examine how sensitive the classifiers were to changes in input data. From this experiment, we obtained a stress profile for each classifier and each type of audio feature. We used these stress profiles to find the best combination of classifier and feature set for the task at hand.

Since analysis of audio samples and speech analysis share many similarities it is natural to use speech compression as a vehicle for stress testing the classifiers and consider audio samples as "machine speech".

4 Classifier stress testing

Various strategies are used in testing classifier performance. In addition to basic cross validation there are several noise injection strategies to emulate real life situations, where the operating conditions or the process under examination changes. In speech recognition tasks various ambient noises are added to input signals [12].

Another metric we use in analyzing the classifier performance is a compression metric. Those compression metrics can be considered either features or metrics.

4.1 Classification model performance metrics

In order to compare the relative performances of classifiers we need evaluation metrics. We decided to use balanced accuracy for cross validation tests for two reasons. It is a fairly simple metrics that can be calculated from confusion matrix in different test scenarios. Another reason is that we had some class imbalance in our testing and training sets. Balanced accuracy is the arithmetic mean of sensitivity and specificity metrics. Sensitivity is the proportion of positive cases being classified as positive. Specificity is the proportion of negative subjects being classified as being negative [13]. This means that if 95 percents of subjects would actually be positive, would it be possible to reach high accuracy by classifying all the cases as positive. Using balanced accuracy would give a penalty of doing that. In our case the subject corresponds to individual audio recording sample.

The imbalance in our case stems from the fact that in the first measurements we only had two sets of measurements at the same rotational speed (RPM), giving us good balance in class representation. However, we later decided to record additional samples to see how changing the rotational speed or fan unit affects the classification performance. Therefore we have many times more samples from the "after" scenario than we have from the "before" scenario. Therefore we needed to select a metric that addresses that class imbalance problem we have in our data.

4.2 Stress test procedure and metrics

Ideas of using compression metrics are used in song identification. There Normalized Compression Distance (NCD) is used to measure similarity between two audio samples [6]. We applied this idea in a simplified form by only compressing the samples and using the final file size relative to the speex [14] encoded (`speexenc`) file size to estimate the information that is available in the speex-encoded files. This measure was then used as a proxy for the information. We used this measure to interpret our classifier stress test results. It is worth noting that this procedure was applied to the encoded audio which was used as input in the classifier testing but not training phase. It means that we bypassed the feature generating in classifier training and therefore the training feature engineering does not affect the measure. In other words, that does not measure how much information the features extract from input data. We wanted the final classification results to show that.

The classifiers were trained only by using the original audio samples (WAV). Classifier training and cross validation used only these original audio data. Then the idea was to simulate the unwanted but evident changes in system under test, the operating conditions,

changes in audio recordings or ambient interferences. Those changes were simulated by gradually decreasing the quality of audio. We thought that this decreasing would in some way be application relevant and not just adding some noise to the input data. Therefore we used "speex" audio encoding to perform the task. The audio encoder's purpose is to reduce the amount of data while keeping the audio quality sufficiently good. It compresses the audio according to the target audio quality level. That level is naturally also reflected in the resulting file size. Since using the encoded audio in the classification system is unnecessarily complicated, we decoded (`speexdec`) the encoded audio back to WAV format keeping track of the quality levels for each file. The resulting WAV files are of equal size regardless of encoding quality, but the actual audio quality is different.

To measure the proxy for information content in the test files we used the standard gzip algorithm [6] so that every encoding quality and every sample file can be estimated in terms of its information content. That gives us a way to estimate how consistent the classification results are in this stress test. The classifier clearly can not create information that does not exist in the input. Therefore we can expect that with less information the performance of the classifier will degrade.

We run the training procedure of classifiers using the original audio. Also the cross validation of the classifiers is done using original audio samples. Those results are stored. Then we precomputed the speex encodings of all test audio samples at every quality level from 0, corresponding to the worst quality, to 10, corresponding to best quality. All the files were then decoded back (using lossy process) to WAV files which were used for generating the features for stress testing. Then for all the encoded files the gzip was run. Finally we calculated the original sizes of encoded audio files and the gzip compressed the files. By simple division operation we computed compressing ratio per file. In figure 4 we can see how the audio information decreased when speex encoding quality decreases.

The trajectories seen in figure 4 show the compression ratio per each subclass of the input audio. Subclasses that begin with "ARPM" stand for "after" case, and are at different RPMs, with "1" corresponding to the lowest and "4" to the highest rotation speed. The subclass "FOLDER01" is the initial measurement of he "before" case and "FOLDER03" is the initial measurement of the "after" case. The additional subclass "SAW" refers to the audio that was recorded from a slightly bearing faulted industrial table saw. According to the indicator, "SAW" has the highest information content. Human perception of audio recordings is in agreement with this indicator.

We conducted the stress tests in two severity levels. The first level follows the procedure described above and used audio samples collected at the same RPM. These results are seen in the figure 2. We call that stress test 1. Then we conducted a more demanding test where the test samples were selected randomly from "before" and "after" samples from original recordings and different RPMs that correspond to subclasses ARPM1, ARPM2, ARPM3 and ARPM4. We see the stress test results for RPM3 in figure 3.

To be sure that compression ratios are proportional to Shannon entropy of the speex encoded files a short c++ program [15] was slightly modified so that it takes the filename as argument and the entropies were generated for all the encoded files. In figure 5 we see that the Shannon entropies are quite well in proportion to the compression ratios. Therefore the compression ratios can be used as an information content metric. Using gzip is more practical than the custom C++ program which lacks installation procedures and documentation. In addition the gzip would be easier accessible to anybody wishing to use this method.

5 Discussion

We need to compare the image 2 and 3, and keep in mind that the information available to the classifier is at most relative to figure 4. First observation is that the plain ACF features are not withstanding the stress test 2 very well. The accuracies drop consistently from about 90% of balanced accuracy to less than 70%. It is also notable that the ACF results do not seem to reflect the available information very well. Where the information content decreases, the ACF feature performance stays almost constant across quality levels. We do not consider that a good indication. That is why we would avoid using the ACF features altogether. From figure1 we see different results. Classical K-fold cross validation gives quite optimistic results.

With FFT features random forest performs well in stress test 1 but in stress test 2 the accuracy improves when available information decreases. That would lead us to reject random forest at least in FFT case. As a feature the partial autocorrelation performs most consistently. In comparison to the ACF it is logical, since the PACF theoretically reduces the dependencies with

lag value L and the smaller lags, thus making the feature space less correlated.

Simple decision tree RPART would be eliminated, since its performance improves some times at very poorest audio qualities while the available information decreases. We consider that as a warning sign. Candidates that remain are LVQ, and both SVM classifiers. According to these results it appears that linear SVM performs better than radial one. The choice between lvq and linear SVM is not trivial. The lvq produces stress test results that are more consistent with the audio information content seen in figure 4.

Selection of appropriate compression level can be done with help of Figure 3. Once we have a candidate feature set and classifier selected, we can determine the balanced accuracy level which is considered sufficient for the application. This is up to the requirements of the application. If our selection of classifier would be e.g. lvq using PACF feature set, it would appear that compression qualities between 5 and 10 (inclusive) would be possible.

In this case we would select the lvq classifier with PACF feature set and apply compression qualities from 5 to 10 depending on the exact data storage costs and how critical the classification accuracy would be in the application.

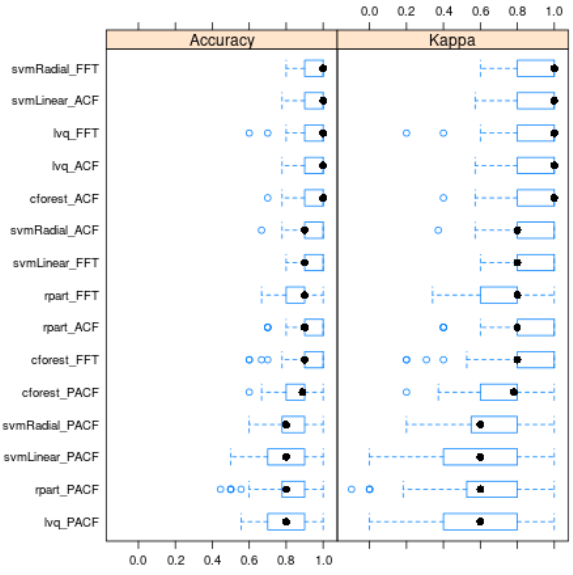

Fig. 1. Cross validation results

conclude the classifier evaluation by selecting LVQ as our primary and linear SVM classifier as our secondary choices. We would use PACF feature set to perform bearing change detection in production conditions.

6 Conclusion

We have developed and experimented with two stress test methods for extending the classical classifier cross validation tests. The stress test 1 uses gradually degrading input audio as test material. In the results seen in figure 2 those profiles are useful in estimating how consistently given classifier works.

Stress test 2 seen in figure 3 is more difficult for classifiers since it uses test data from several rotation speeds and it is degraded in quality. It probably gives more realistic estimate regarding the performance we can expect from given classifiers and feature sets in real world application.

We consider the gradual and quantified scale of test information contents as a very useful tool. That tool allows us to make common measure comparisons between the classifiers and feature sets.

Ability to quantify and compare information content against classifier performance is not something a human has. That is where we consider this method outperforming or helping human perception. We would

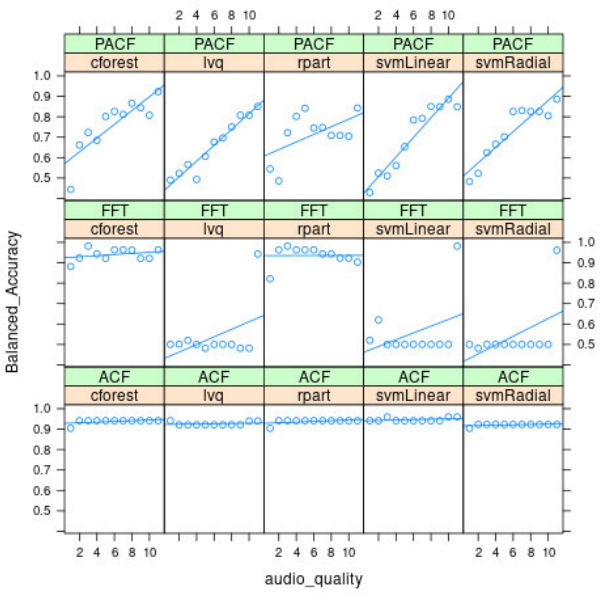

Fig. 2. Stress test 1: Impact of changing only audio quality to balanced accuracy. Every subplot represents combination of classifier and feature set. Within subplots the horizontal axis is audio quality indicator where 0 stands for poorest and 11 is original audio recording. Vertical axis is balanced accuracy.

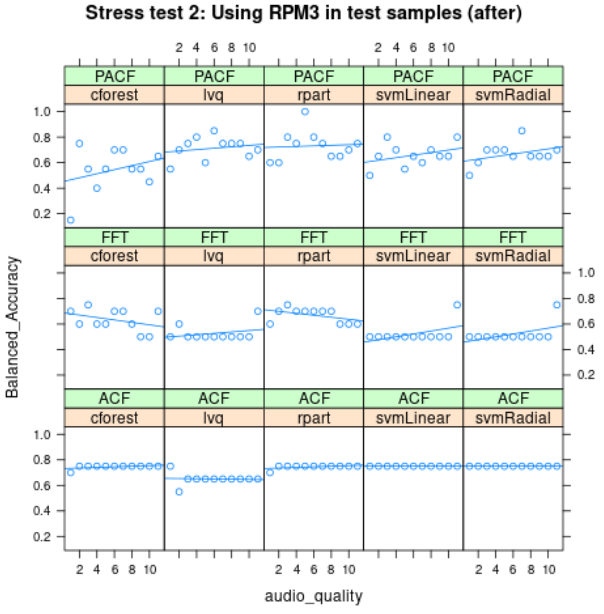

Fig. 3. Stress test 2: This test is more challenging than stress test 1. In this case the input data contained audio samples from different rotation speeds (RPM) and the audio quality is gradually reduced as in stress test 1. The ACF feature results are negatively affected for all classifiers.

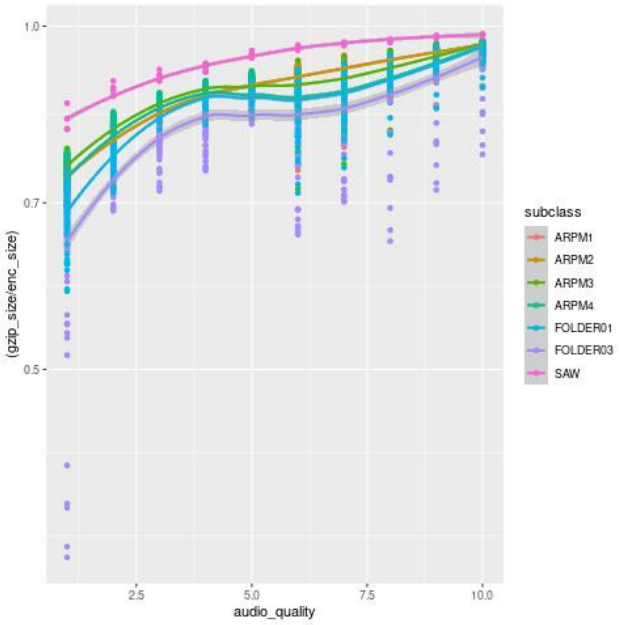

Fig. 4. Estimate of information in speex compressed audio samples. Vertical axis shows gzip compressed speex encoded file size divided by speex encoded file. The ratio shows approximation of information still available in speex encoded files. Value 1 would signify speex encoded file that does not compress at all using gzip compression, whereas low numbers close to 0 indicate files that are compressed a lot and do not initially carry much information

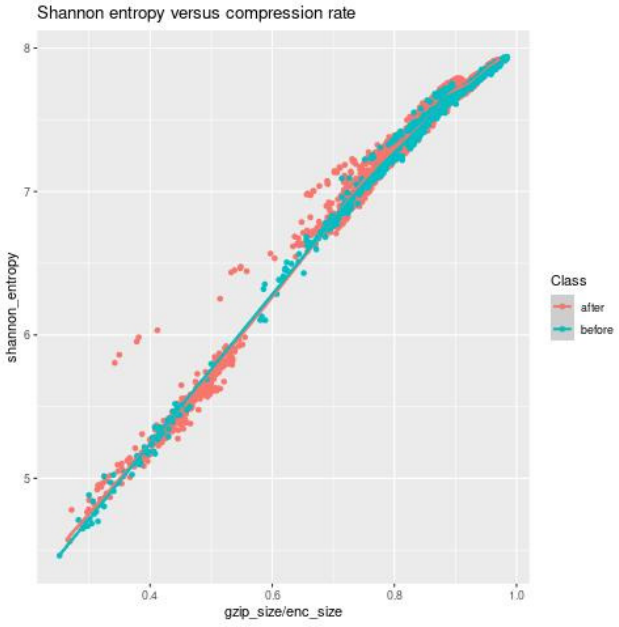

Fig. 5. Shannon entropy versus compression ratio of speex encoded files

References

[1] Ugarte Juan P, Arias-Arias Jose. Unveiling relevant acoustic features for bird species automatic classification. *Expert Systems With Applications*. 2024;.

[2] Box G E, Jenkins G M. *Time Series Analysis, forecasting and control*. Holden-Day Inc. 1976.

[3] Lange H F. *Correlation Techniques*. Iliffe Books Ltd. 1967.

[4] Rabiner Lawrence R, Schafer Ronald W. *Introduction to Digital Speech Processing*. now Publishers Inc. 2007.

[5] Ljung Lennart. *System Identification*. Prentice Hall PTR. 1999.

[6] Ahonen Teppo. Cover song identification using compression based distance measures. Ph.D. thesis, University of Helsinki. 2015.

[7] Theodoridis S. Machine Learning. In: *Machine Learning (Third Edition)*, edited by Theodoridis S, pp. 1–17. Academic Press, third edition ed. 2026;.
URL https://www.sciencedirect.com/science/article/pii/B978044329238500007X

[8] Hosameldin Ahmed, Asoke Nandi. *Condition Monitoring With Vibration Signals*. Wiley. 2020.

[9] Kuhn Max. Building Predictive Models in R Using the caret Package. *Journal of Statistical Software November 2008, Volume 28, Issue 5 http://wwwjstatsoftorg/*. 2008;.

[10] T Hastie, R Tibshirani, J Friedman. *The Elements of Statistical learning*. Springer Science Business Media. 2009.

[11] Kuhn M, Johnson K. *Applied Predictive Modeling*. Springer Science+Business Media. 2013.

[12] Chen J, Wang Y, Wang D. A Feature Study for Classification-Based Speech separation at Low Signal-to-Noise Ratios. *IEEE/ACM Transactions on audio, speech and language processing*. 2014;22, number 12:1993–2002.

[13] Poduri S R S Rao. *Statistical Methods with Medical Applications*. John Wiley & Sons, Incorporated. 2016.

[14] speex project. Speex: A Free Codec For Free Speech. 2022.
URL https://speex.org

[15] TFE TIMES blog. C++ entropy. 2017.
URL https://tfetimes.com/c-entropy/