

ŞARKILARDAKİ ENSTRÜMANLARIN TESPİTİ

IDENTIFICATION OF INSTRUMENTS IN SONGS

BUSE YENER, NİSA AKSOY

BİLİŞİM SİSTEMLERİ MÜHENDİSLİĞİ

KOCAELİ ÜNİVERSİTESİ

busee.yener@gmail.com

BİLİŞİM SİSTEMLERİ MÜHENDİSLİĞİ

KOCAELİ ÜNİVERSİTESİ

nisaaksoy08@gmail.com

Özet

Bu dokümanda şarkılardaki müzik aletlerini ayırt eden makine öğrenmesi projesine ait detaylar yer almaktadır.

Abstract

This document details a machine learning project that distinguishes musical instruments in songs.

1.Giriş

Projemizi yaparken Makine Öğrenmesi dalında en iyi verim alabildiğimiz Python programlama dilini ve Python dilinde yazılan kodları çalıştırabilmek için Jupyter Notebook geliştirme ortamını kullandık. Öncelikle Python yazılım dili nedir, ne işe yarar?

Python: Web uygulamaları, yazılım geliştirme, veri bilimi ve makine öğreniminde (ML) yaygın olarak kullanılan bir programlama dilidir. Geliştiriciler, etkili ve öğrenmesi kolay olduğu ve birçok farklı platformda çalıştırılabildiği için bu dili kullanır. Ücretsiz olarak indirilebilir, her türlü sistemle iyi bir entegrasyon sağlar ve geliştirme hızını artırır.

Avantajları:

- Geliştiriciler, basit ve İngilizceye benzer bir söz dizimine sahip olduğundan Python dilinde yazılmış programı kolayca okuyabilir ve anlayabilir.
- Geliştiriciler, diğer birçok dile kıyasla çok daha az kod satırıyla bir Python programı yazabildiğinden, geliştiricilerin daha üretken olmasını sağlar.
- Python, neredeyse her görev için yeniden kullanılabilir kodları içeren geniş bir standart kitaplığa sahiptir. Geliştiricilerin kodu sıfırdan yazmasına gerek kalmaz.

- Geliştiriciler Python'ı Java, C ve C++ gibi diğer popüler programlama dilleriyle kolayca kullanabilir.
- Bir sorunla karşılaşırsanız Python topluluğundaki ekiplerden hızlıca destek alabilirsiniz.
- Python'ı öğrenmek için internette birçok yararlı kaynak bulunmaktadır. Kolayca videolar, dersler, belgeler ve geliştirici kılavuzları bulabilirsiniz.
- Windows, macOS, Linux ve Unix gibi farklı bilgisayar işletim sistemleri arasında taşınabilir.

Veri Bilimi ve Makine Öğrenimi: Veri bilimi, verilerdeki değerli bilgileri ayıklamaktır ve makine öğrenimi (ML), bilgisayarlara verilerden otomatik olarak öğrenmesini ve doğru tahminlerde bulunmasını öğretir. Veri bilimciler, aşağıdakiler gibi görevler için Python'ı kullanır:

- Veri temizleme, yanlış verileri düzeltme ve kaldırma
- Verileri ayıklama, verilerin çeşitli özelliklerini seçme
- Veri etiketleme, veriler için anlamlı adlar ekleme
- Verilerden farklı öngörüler çıkarma
- Çizgi grafikler, çubuk grafikler, sütun grafikler ve dilim grafikler gibi grafikler yardımıyla verileri görselleştirme

Veri bilimciler, Makine Öğrenmesi modellerini eğitmek ve verileri doğru şekilde sınıflandıran sınıflandırıcılar derlemek için Python dilinin Makine Öğrenmesi kitaplıklarını kullanır. Görüntü, metin ve ağ trafiği sınıflandırması, konuşma tanıma ve yüz tanıma gibi sınıflandırma görevlerini yapması için çalışan kişiler Python tabanlı sınıflandırıcılar kullanır.

Jupyter Notebook: Veri bilimi, makine öğrenimi, veri analizi ve bilimsel hesaplamalar gibi alanlarda yaygın olarak kullanılan, etkileşimli bir not defteri uygulamasıdır. Bu araç, programcılarının ve veri bilimcilerinin verileri keşfetmelerine, analiz etmelerine, sonuçlarını görselleştirmelerine olanak tanır. Hem kod hem de metin içeriğini bir arada sunabilmesi, Jupyter Notebook'u diğer geliştirme araçlarından ayıran en önemli özelliklerden biridir.

2. Şarkılardaki Enstrümanların Tespiti

Bu proje, piyano, keman, gitar ve kanun gibi çeşitli müzik enstrümanlarına ait ses verilerinin toplanması, işlenmesi ve sınıflandırmaya uygun hale getirilmesinden oluşmaktadır. Projede, YouTube'dan indirilen ses dosyalarının sessizlik ve gürültüden arındırılması, ardından MFCC özelliklerinin çıkarılması ile veriler etiketlenerek makine öğrenmesi modelleri için uygun bir veri seti oluşturulmaktadır.

2.1 Veri Toplama

Proje için her enstrümana özel olarak YouTube çalma listeleri kullanılarak veri toplanmıştır. Ses dosyalarının düzenli bir şekilde saklanması ve işlenmesi için her enstrüman için ayrı bir klasör oluşturulmuştur. Ses verilerinin toplandığı bu işlemde:

Araçlar: Verileri toplamak için yt_dlp kütüphanesi kullanıldı.

İşlem: YouTube playlist'leri kullanılarak, her bir çalma listesindeki tüm videolar indirildi ve ses dosyası olarak kaydedildi. Kaydedilen dosyalar MP3 formatındadır ve her bir enstrüman için ilgili klasörde saklanmıştır.

2.2 Veri Ön işleme

Toplanan ses verilerinin analiz edilebilir hale getirilmesi için ön işleme aşaması gerçekleştirilmiştir. Bu aşama, ses dosyalarındaki gürültü ve sessizlikleri azaltmak, verileri segmentlere ayırmak ve her segment için MFCC çıkarımı yapmak gibi işlemleri içerir.

Sessizlik Temizleme: Ses dosyalarındaki gereksiz sessiz bölümler, analizde istenmeyen gürültüye ve modelin doğruluğunu olumsuz etkileyebilecek unsurlara yol açabilir. Bu nedenle, pydub kütüphanesi kullanılarak her ses dosyasındaki sessizlik bölgeleri temizlenmiştir. Sessizliği giderilen ses dosyaları, işlenebilir duruma gelmiştir.

Gürültü Temizleme: Toplanan verilerde arka plan gürültüsü olabileceğinden, sesin netliği ve ayırt edici özellikleri korumak amacıyla gürültü temizleme işlemi yapılmıştır. noisereduce kütüphanesi kullanılarak her ses kaydının arka plan gürültüsü azaltılmıştır.

Normalizasyon : Gürültü temizleme işleminin ardından, ses sinyali, genlik dalgalanmalarını gidermek amacıyla normalize edilmiştir. Bu adım, ses dosyasının sabit bir ses seviyesi sunmasını sağlamakta ve özellik çıkarımı adımında istikrarlı sonuçlar elde edilmesine katkıda bulunmaktadır.

2.3 Özellik Çıkarımı

Ön işleme adımlarından geçen ses verileri, sınıflandırma modeline uygun öznitelikler çıkarmak için hazırlanmıştır. Bu projede, her bir ses segmenti için MFCC (Mel Frekans Kepstral Katsayıları) çıkarımı yapılmıştır. MFCC, ses verisinin frekans özelliklerini temsil eden ve özellikle ses tanıma işlemlerinde yaygın olarak kullanılan bir tekniktir. Bu proje kapsamında, her ses parçası 3 saniyelik segmentlere ayrılmış ve her segment için 13 MFCC değeri hesaplanarak kaydedilmiştir.

2.4 Veri Segmentasyonu ve Etiketleme

Veri Segmentasyonu: Her bir ses kaydı, analiz sürecine uygun olacak şekilde 3'er saniyelik parçalara bölünmüştür. Bu işlemde, toplam süre boyunca her bir segment ayrı bir dosya olarak kaydedilmiştir. Her segment, gürültü temizleme ve MFCC çıkarımı işlemlerine tabi tutularak dosya sistemi içinde saklanmıştır.

Etiketleme: Veriler, ilerideki model eğitim aşamalarında kullanılmak üzere etiketlenmiştir. Her enstrümana ait segmentler, piyano, keman, gitar, kanun gibi etiketler ile ilişkilendirilmiştir. Bu sayede, model eğitimi sırasında her veriye hangi enstrümana ait olduğu bilgisiyle ulaşılabilecektir.

Veri toplama ve ön işleme süreçlerinin sonucunda, müzik enstrümanlarının ayırt edici özelliklerini yansıtan yüksek kaliteli bir veri seti oluşturulmuştur. Bu aşamada, ses verilerinden gürültü ve sessizlik temizlenmiş, ardından Mel-Frekans Kepstrum Katsayıları (MFCC) çıkarılarak veriler etiketlenmiştir. Bu şekilde hazırlanmış veri seti, makine öğrenmesi modellerinin eğitimi için uygun hale getirilmiş olup, modelin doğruluğunu artırmaya ve enstrümanları doğru bir şekilde sınıflandırmaya yönelik kritik bir temel oluşturmıştır.

```
Toplam veri sayısı: 14647
Örnek bir MFCC verisi: [[-528.546    -511.15637   -474.06735    ... -282.1321    -301.36893
  -323.31006 ]
 [  5.6669626   24.289175   50.58533    ...  110.64725   107.038704
  101.80883 ]
 [ -1.170081    5.1891623   11.636992    ... -50.12626   -49.600876
  -46.741325 ]
 ...
 [ -3.3576822   -3.465158   -1.8829875    ... -11.877163   -5.7145214
   5.3668895 ]
 [ -6.817239    -12.677327   -9.990316    ... -4.5545955  -10.044651
  -8.943605 ]
 [ -5.220399    -10.639755   -12.067743    ... -13.589108   -8.248816
  -5.7354994]]
Örnek bir etiket: gitar
```

Yukarıdaki resimde, proje kapsamında elde edilen toplam veri sayısının 14,647 olduğu görülmektedir. Her bir veri örneği, ses sinyalinin frekans özelliklerini temsil eden MFCC değerlerini içermektedir. MFCC özellikleri, her enstrümanın kendine özgü ses karakteristiklerini yansıtarak makine öğrenmesi algoritmaları için öznitelik setini oluşturmaktadır. Görselde yer alan örnek MFCC verisi, bir gitar sesine ait olup, ses sinyalinin farklı zaman dilimlerindeki frekans bileşenlerini göstermektedir. Bu tür MFCC verileri, enstrüman sınıflandırma modelinin eğitimi için temel veri setini oluşturmaktadır.

3. Makine Öğrenmesi Modelleri

Sonraki adım olarak, toplanan ve işlenen verilerle çeşitli makine öğrenmesi modelleri eğitilmiştir. Bu modeller, müzik enstrümanlarının ses verilerini doğru bir şekilde sınıflandırmak için kullanılmış ve her bir modelin performansı çeşitli metriklerle değerlendirilmiştir.

3.1 Linear Regression (Doğrusal Regresyon)

Doğrusal regresyon, makine öğrenmesinin temel modellerinden biridir ve genellikle doğrusal ilişki kurma amacıyla kullanılır. Bu model, bağımsız değişkenlerin doğrusal bir kombinasyonunu kullanarak, hedef değişkeni tahmin etmeye çalışır. Hızlı eğitim süresi ve anlaşılır yapısı ile dikkat çeker. Ancak, doğrusal olmayan ilişkilerde düşük performans gösterebilir.

Model Eğitimi: Doğrusal regresyon, modelin hızını artırmak ve daha istikrarlı bir eğitim süreci sağlamak amacıyla standart ayarlarda eğitilmiştir. Parametre olarak `fit_intercept=True` kullanılmıştır. Model, 10 epoch boyunca eğitilmiş ve eğitim süresi 0.0105 saniye olarak kaydedilmiştir.

Sonuçlar:

- MSE (Ortalama Kare Hatası): 0.3309
- MAE (Ortalama Mutlak Hata): 0.4209
- RMSE (Kare Kök Hata): 0.5752
- MAPE (Ortalama Mutlak Yüzde Hata): 1.407363e+11
- R2 Skoru: 0.7683
- Eğitim Zamanı: 0.0105 saniye
- Çıkarım Zamanı: 0.0000 saniye

3.2 Ridge Regression (Ridge Regresyonu)

Ridge regresyonu, doğrudan doğrusal regresyon modelinin bir varyasyonudur ve modelin aşırı uyum yapmasını engellemek amacıyla ceza terimi (regularization) ekler. Bu model, doğrusal ilişkilerde oldukça başarılıdır ve genellikle veri setinde multikollinearlik (bağımsız değişkenler arasındaki yüksek korelasyon) bulunduğu tercih edilir.

Model Eğitimi: Ridge regresyonu, parametre olarak `alpha=1` kullanılarak eğitilmiştir. Bu, ceza teriminin gücünü ayarlamak için kullanılan bir parametredir. Model, 10 epoch boyunca eğitim almış ve eğitim süresi 0.0303 saniye olarak kaydedilmiştir.

Sonuçlar:

- MSE (Ortalama Kare Hatası): 0.3309
- MAE (Ortalama Mutlak Hata): 0.4209
- RMSE (Kare Kök Hata): 0.5752
- MAPE (Ortalama Mutlak Yüzde Hata): 1.407364e+11
- R2 Skoru: 0.7683
- Eğitim Zamanı: 0.0303 saniye
- Çıkarım Zamanı: 0.0010 saniye

3.3 Lasso Regression (Lasso Regresyonu)

Lasso regresyonu, Ridge regresyonuna benzer şekilde, ancak farklı olarak daha güçlü bir ceza terimi kullanarak modelin daha basitleştirilmesini sağlar. Bu model, özellikle değişken seçimi yapmak ve bazı öznitelikleri sıfırlamak amacıyla kullanılır. Ancak, doğrusal olmayan ilişkilerde performansı sınırlı olabilir.

Model Eğitimi: Lasso regresyonu, parametre olarak `alpha=0.1` ile eğitilmiştir. Model, 10 epoch boyunca eğitilmiş ve eğitim süresi 0.3440 saniye olarak kaydedilmiştir.

Sonuçlar:

- MSE (Ortalama Kare Hatası): 0.4903
- MAE (Ortalama Mutlak Hata): 0.5572
- RMSE (Kare Kök Hata): 0.7002
- MAPE (Ortalama Mutlak Yüzde Hata): 1.717027e+11
- R2 Skoru: 0.6566
- Eğitim Zamanı: 0.3440 saniye
- Çıkarım Zamanı: 0.0000 saniye

3.4 Support Vector Regressor (Destek Vektör Regresyonu)

Destek Vektör Regresyonu, doğrusal olmayan veri setlerine uygulanabilen güçlü bir modeldir. Bu model, yüksek doğruluk sağlayabilen ve veri üzerinde doğru sınıflandırma yapabilen karmaşık bir yapıya sahiptir. Ancak, eğitim süresi genellikle daha uzun olabilir.

Model Eğitimi: Destek Vektör Regressor, `C=1.0` parametresi ile eğitilmiştir. Bu parametre, modelin genel doğruluğunu etkileyen düzenleme terimini temsil eder. Model, eğitim süresi olarak 13.8702 saniye ve çıkarım süresi olarak 1.8957 saniye harcamıştır.

Sonuçlar:

- MSE (Ortalama Kare Hatası): 0.2564
- MAE (Ortalama Mutlak Hata): 0.3428
- RMSE (Kare Kök Hata): 0.5063
- MAPE (Ortalama Mutlak Yüzde Hata): 1.292479e+11
- R2 Skoru: 0.8204
- Eğitim Zamanı: 13.8702 saniye
- Çıkarım Zamanı: 1.8957 saniye

3.5 Random Forest Regressor (Rastgele Orman Regresyonu)

Random Forest, bir ansamble model olup, birçok karar ağacının birleşimiyle daha doğru tahminler yapar. Her bir ağaç, veriye farklı bir bakış açısı getirir ve nihai karar, ağaçların çoğunluk kararıyla belirlenir. Bu model,

genellikle doğruluk oranının yüksekliği ve düşük aşırı uyum riski ile tercih edilir.

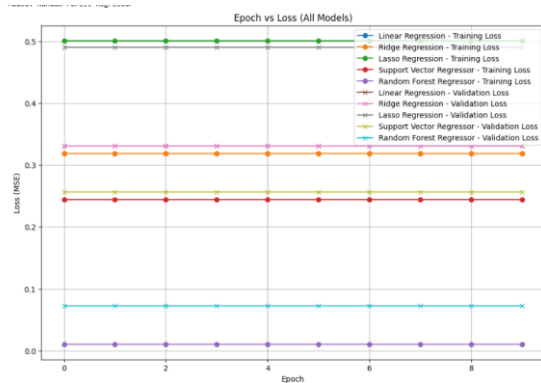
Model Eğitimi: Random Forest modeli, $n_estimators=100$ parametresiyle eğitilmiştir. Bu, modelin 100 karar ağacı ile eğitilmesini sağlar. Eğitim süresi 13.9364 saniye olarak kaydedilmiştir ve çıkarım süresi oldukça hızlıdır (0.0157 saniye).

Sonuçlar:

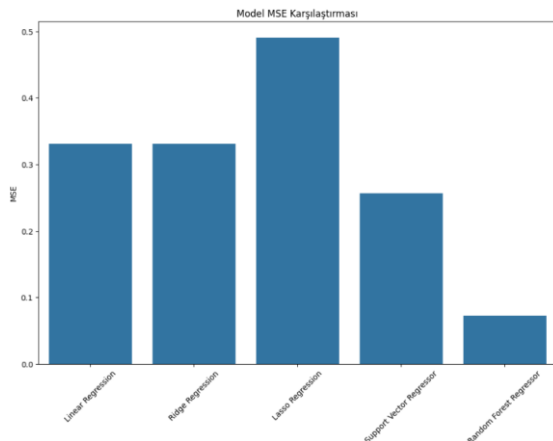
- MSE (Ortalama Kare Hatası): 0.0726
- MAE (Ortalama Mutlak Hata): 0.0900
- RMSE (Kare Kök Hata): 0.2695
- MAPE (Ortalama Mutlak Yüzde Hata): $2.051195e+10$
- R2 Skoru: 0.9491
- Eğitim Zamanı: 13.9364 saniye
- Çıkarım Zamanı: 0.0157 saniye

Eğitilen modellerin performansı, çeşitli metriklerle değerlendirilmiş ve görsel olarak sunulmuştur. Aşağıdaki grafikler, modellerin eğitim süreçleri ve performans metriklerine dair önemli verileri içermektedir:

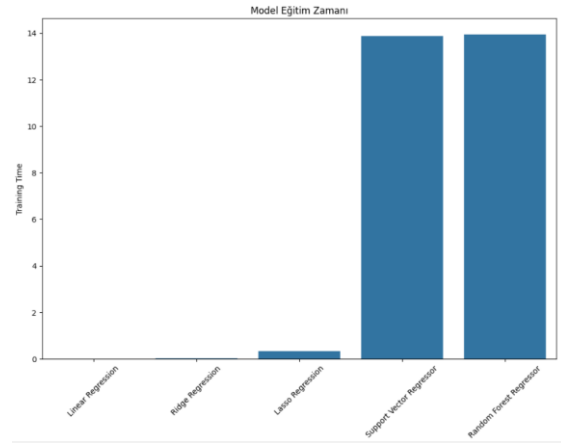
Epoch vs Loss Grafiği: Tüm modellerin eğitim sırasında kayıp (loss) değerlerinin değişimi.



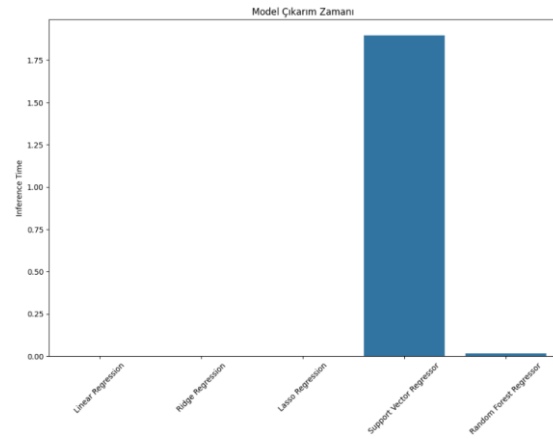
Model MSE Karşılaştırması: Her modelin MSE değeri ile performans karşılaştırması.



Model Eğitim Zamanı: Eğitim süresi karşılaştırması.



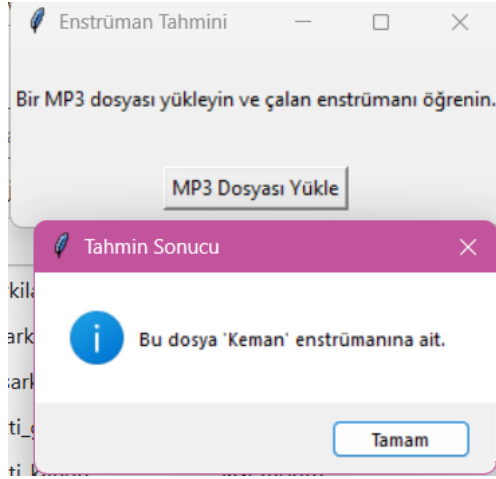
Model Çıkarım Zamanı: Çıkarım (inference) süresi karşılaştırması.



Beş farklı modelin test edilmesinin ardından, Random Forest Regressor en iyi performansı sergileyen model olmuştur. Model, düşük hata oranları ve yüksek doğruluk ile öne çıkmış, özellikle MSE (0.0726) ve R2 skoru (0.9491) ile diğer modellere göre çok daha başarılı sonuçlar elde edilmiştir. Eğitim ve çıkarım süreleri de dikkate alındığında, Random Forest hızlı ve güvenilir sonuçlar sunan bir model olarak öne çıkmaktadır.

4.Arayüz

Makine öğrenmesi modelleri eğitildikten sonra Random Forest Regressor modeli, en iyi performansı gösteren model olarak seçilmiştir. Modelin eğitim süreci tamamlandıktan sonra, modelin kaydedilmesi işlemi gerçekleştirilmiştir. Bu işlem, modelin gelecekteki kullanımlar için tekrar eğitilmeden doğrudan yüklenip tahmin yapabilmesini sağlar.



Modelin kaydedilmesinin ardından, kullanıcı etkileşimini kolaylaştırmak amacıyla Tkinter tabanlı bir grafiksel kullanıcı arayüzü (GUI) geliştirilmiştir. Bu arayüz, kullanıcının ses dosyalarını yükleyip, bu dosyalar üzerinde enstrüman tahmini yapabilmesini sağlar. Kullanıcı, ses dosyasını yükledikten sonra, program otomatik olarak dosyayı işler, gerekli öznitelikleri çıkarır ve kaydedilen Random Forest modelini kullanarak tahmin yapar. Sonuç olarak, kullanıcıya hangi enstrümanın çaldığına dair bir bilgi verilir.

5. Proje Linkleri

5.1 Github

211307048 Buse Yener –

<https://github.com/buseyener/Enstruman-Tanimlayici>

211307089 Nisa Aksoy –

<https://github.com/nisaaksoy/Enstruman-Tanimlayici>

5.2 Drive

<https://drive.google.com/file/d/1ph0Vx8MA63VMAK07Abw2as5MTBuyRRDv/view?usp=sharing>

<https://drive.google.com/file/d/14XNoqaynBfZZyN-oXxedAJLriFF9n60i/view?usp=sharing>

5.3 Google Colab

https://drive.google.com/drive/folders/1wIYjC_7hH1-W8DxMyQrlqTr0CJ_tDp_S

https://drive.google.com/drive/folders/1WhNSINogmJaaNLOU6CzXdg0eWzq_hef7

Kaynakça

1. <https://aws.amazon.com/tr/what-is/python/#:~:text=Python%3B%20web%20uygulamalar%C4%B1%2C%20yaz%C4%B1%C4%B1m%20geli%C5%9Firme,%C3%A7al%C4%B1%C5%9F%C4%B1r%C4%B1labildi%C4%9Fi%20i%C3%A7in%20Python%C4%B1%20kullan%C4%B1r>
2. <https://bilginc.com.tr/blog/python-nedir-python-hakkinda-hersey-158/>
3. <https://www.komtas.com/post/python-programlama-dili-ve-kullanim-alanlari>
4. <https://www.python.org/>
5. <https://jupyter.org/>
6. <https://medium.com/datarunner/jupyter-notebook-nedir-502c024c0076>
7. <https://www.gencbeyinler.net/jupyter-notebook-nedir-detayli-bir-rehber/>
8. <https://www.ozztech.net/yazilim-gelistirme/jupyter-notebook-nedir/>
9. <https://github.com/yt-dlp/yt-dlp>
10. <https://librosa.org/>
11. <https://github.com/timsainb/noisereducer>
12. <https://numpy.org/>
13. <https://efecanxrd.medium.com/mfcc-mel-frequency-cepstral-coefficients-ve-gmm-gaussian-mixture-model-teknikleri-f81a2cc632a5>