



دانشگاه صنعتی شریف

دانشکده مهندسی برق

پروژه پایانی درس یادگیری آماری

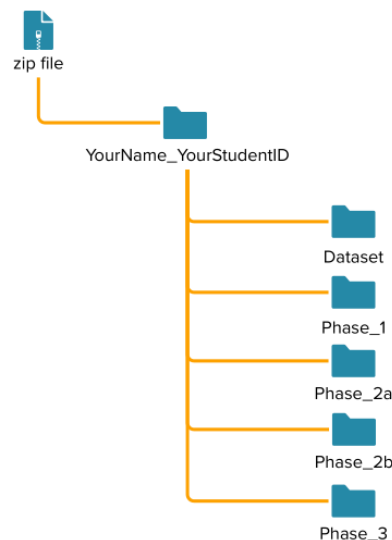
طبقه بندی سیگنال صوت

نیمسال تحصیلی ۱۴۰۰-۱۴۰۱

❖ نکات مهم

لطفاً به هنگام انجام پروژه و آماده سازی نتایج به موارد زیر توجه نمایید.

- گزارش پروژه باید به صورت کامل و با تمام جزئیات نوشته شود. در گزارش خود، بخش‌ها و زیربخش‌های مربوط به هر فاز را به صورت جداگانه بیاورید.
 - کدها خوانا و مرتب نوشته شده و تا حد امکان کامنت‌گذاری شوند.
 - کدها مستقیماً اجرا شده و خروجی‌های مطلوب را تولید نمایند.
 - در انجام پروژه مشورت مجاز است ولی بدیهی است در صورت مشاهده هرگونه تشابه غیر معمول بین کدها و یا نتایج، طرفین نمره صفر از پروژه دریافت خواهند کرد.
 - با توجه به اینکه سوال‌های دانشجویان غالباً مشترک هستند، لطفاً هرگونه ابهام و یا سؤال را در سامانه CW مطرح نمایید تا سایر دانشجویانی که سوالی مشابه دارند نیز به پاسخ‌ها دسترسی داشته باشند.
 - در پایان تمامی مستندات لازم را در یک فایل zip قرار دهید. نام فایل باید به صورت YourName_YourStudentID باشد. در داخل فایل zip باید یک پوشه به همین نام وجود داشته باشد. سپس مستندات لازم برای هر فاز در یک پوشه جداگانه قرار داده شوند. ساختار پوشه بندی مطلوب در شکل زیر نشان داده شده است.
- توجه: لازم نیست تا دیتاست مورد استفاده را به همراه فایل‌های خود آپلود نمایید. بلکه تنها لازم است تا آدرس‌دهی‌ها برای دسترسی به دیتاست مطابق ساختار زیر باشد.



شکل ۱- ساختار مطلوب فایل‌های تحویلی نهایی

❖ مقدمه

در این پروژه قصد داریم تا با موضوع طبقه بندی سری های زمانی با استفاده از روش های یادگیری آماری آشنا شویم. یکی از مهمترین سری های زمانی، سیگنال های صوتی هستند. طبقه بندی سیگنال های صوتی کاربردهای زیادی نظیر تبدیل متن به گفتار و بالعکس، تعیین ژانر موسیقی، دستیار مجازی (نظیر Google Assistant و Siri) و ... دارد.

❖ اهداف پروژه

در این پروژه اهداف زیر دنبال می شوند:

- آشنایی با سری های زمانی و به خصوص سیگنال های صوتی و نحوه آماده سازی آنها برای آموزش مدل های یادگیری
- آشنایی با نحوه استخراج ویژگی و اهمیت آن در آموزش مدل های یادگیری ماشین
- آشنایی با روش های یادگیری با/بدون نظارت در سری های زمانی
- مقایسه عملکرد طبقه بندی های مختلف بر روی سیگنال های صوتی

❖ مراحل انجام کار

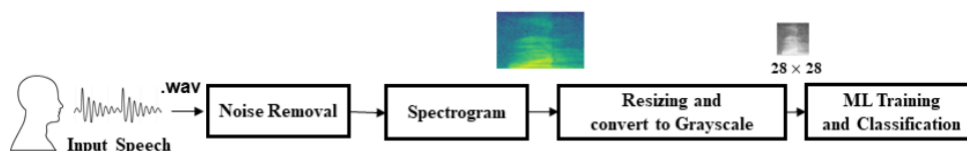
این پروژه در سه فاز آماده سازی داده ها، آموزش طبقه بند با روش یادگیری با نظارت^۱ و یادگیری بدون نظارت به عنوان پیش پردازش^۲ انجام می شود. در بخش یادگیری با نظارت، تاثیر استخراج ویژگی توسط Spectrogram و طبقه بندی های مختلف در دقت نهایی را بررسی می نماییم. سپس در بخش یادگیری بدون نظارت به عنوان پیش پردازش، می خواهیم کاربرد روش های یادگیری بدون نظارت برای تولید برجسب برای داده ها را مورد بررسی قرار دهیم.

¹ Supervised Learning

² Unsupervised Learning for Preprocessing

فاز اول – آماده سازی داده ها

دیتاستی که قرار است در این پروژه با آن کار کنیم شامل ۲۰۰۰ نمونه فایل صوتی با فرمت .wav. که با نرخ 8 kHz نمونه برداری شده‌اند. در هر فایل صوتی یک شخص یک رقم از ۰ تا ۹ را بیان کرده است. هدف آن است تا با آموزش یک طبقه‌بند^۳ سیگنال‌های صوتی را دسته‌بندی کنیم. روند کلی کار در شکل زیر نشان داده شده است.



الف) پوشه مربوط به دیتاست را از سامانه CW دانلود نمایید.

ب) با توجه به ساختار پوشه‌بندی دیتاست که در اختیارتان، برچسب‌های متناظر را تولید نمایید.

ب) صداهاى ضبط شده نویزى هستند. نویزهاى موجود شامل نویزهاى فرکانس بالا، نویز فرکانس پایین(حدود ۵۰ هرتز) و همچنین میزانی نویز سفید می‌باشد. لذا لازم است تا در این بخش نویزهاى موجود تا حد امکان حذف گردند تا یادگیری مدل بهتر صورت گیرد.

ج) زمان فایل‌های صوتی مختلف یکسان نمی‌باشد در نتیجه لازم است در پیش پردازش طول همه نمونه‌ها را به یک مقدار ثابت برسانید. (برای مثال می‌توانید طول همه نمونه‌ها را به طول نمونه‌ای که بیش‌ترین طول را دارد برسانید). توجه کنید که در برخی از فایل‌ها، زمان‌هایی وجود دارد که گوینده ساکت است، می‌توانید با حذف این زمان‌ها دقت یادگیری مدل خود را افزایش دهید.

³ Classifier

فاز دوم – یادگیری با نظارت

همانطور که می‌دانید در یادگیری با نظارت^۴، داده‌ها و برچسب‌های متناظر در دسترس هستند. بنابراین در قدم بعدی لازم است تا با استخراج ویژگی‌های معنادار از داده‌های آموزشی، مدل یادگیری را آموزش دهیم. در این بخش می‌خواهیم تاثیر استخراج ویژگی‌های معنادار در میزان دقت شبکه آموزش داده شده را مشاهده نماییم. به همین دلیل ابتدا یکبار مدل یادگیری را بدون و بار دیگر با استفاده از ویژگی‌های استخراج شده از سیگنال‌های اصلی آموزش می‌دهیم و به مقایسه نتایج بدست آمده می‌پردازیم.

آموزش مدل بدون استخراج ویژگی

الف) داده‌ها و برچسب‌هایی که در فاز اول آماده کرده‌اید را فراخوانی کنید و آنها را به ترتیب در متغیرهای X و Y قرار دهید. حال با استفاده از یکی از روش‌های طبقه‌بندی مطرح شده در درس، مدل نهایی را با هدف کمینه کردن خطا روی داده‌های تست طراحی کنید. همانطور که میدانید ممکن است خطا روی داده‌های آموزش پایین باشد ولی به علت واریانس بالای مدل خطای تست مناسب نباشد. در نتیجه لازم است با صلاح دید خود قسمتی از دیتاست آموزشی را به عنوان داده‌های ارزیابی^۵ جدا کنید تا تخمینی از میزان خطای تست داشته باشید. برای تقسیم داده‌ها می‌توانید از تابع `model_selection.train_test_split` در کتابخانه `ScikitLearn` استفاده نمایید. به منظور تخمین دقیقتر خطای مدل و همچنین محاسبه هاپیرپارامترهای آن از روش `K-fold cross validation` استفاده نمایید. همچنین می‌توانید از روش‌های کاهش بعد نظیر `PCA` (یا هر روش دلخواه دیگر) استفاده نمایید.

ب) مدل‌های مختلفی که برای آموزش استفاده کرده‌اید را به همراه هاپیرپارامترهایشان در یک جدول بیاورید. همچنین دقت بهترین مدل بدست آمده بر روی داده‌های آموزش، ارزیابی و تست را در یک جدول جداگانه بیاورید. ماتریس `Confusion` را بر روی داده‌های تست برای بهترین مدل گزارش دهید. بهترین مدل بدست آمده را با نام `Model_Phase1a.pkl` با استفاده از ماژول `Pickle` ذخیره نمایید.

پ) تابعی با نام `Predict_Phase1` بنویسید که ورودی آن ماتریس داده‌ها و آدرس مدل ذخیره شده در بخش (ب) است. این تابع ابتدا مدل را فراخوانی کرده و با اعمال پیش‌پردازش‌های لازم، طبقه‌بندی را انجام داده و مقادیر خروجی را در فایلی به نام `phase1a_predicted.npy` ذخیره می‌نماید. در نهایت داده‌های تست `phase1_test.npy` را برای دریافت پیش‌بینی انجام شده توسط مدل به تابع موردنظر بدهید.

توجه: به دلیل آنکه داده‌ها تست و ارزیابی، برای بررسی عملکرد شبکه استفاده می‌شوند، لازم است تا دقیقاً همان مراحل که در پیش‌پردازش داده‌های آموزشی انجام شده است، بر روی این داده‌ها نیز اعمال شود. اما توجه کنید که باید از همان پارامترهای نمونه‌های آموزشی برای پیش‌پردازش داده‌های تست و ارزیابی استفاده نمایید.

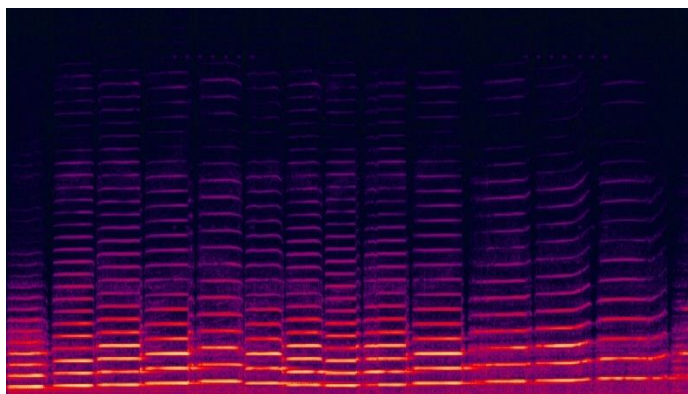
آموزش مدل با استخراج ویژگی

در این بخش می‌خواهیم تا از سیگنال‌های صوت ویژگی‌های معناداری استخراج نماییم و با استفاده از آنها به آموزش مدل یادگیری بپردازیم. یکی از روش‌های استخراج ویژگی از سیگنال‌های صوتی، تهیه `Spectrogram` برای سیگنال مذکور و استخراج اطلاعات

^۴ Supervised Learning

^۵ Validation

فرکانس-زمان از آن است. در شکل زیر نمونه‌ای از spectrogram بدست آمده ناشی از سیگنال صوتی ساز ویالون^۶ نشان داده شده است.

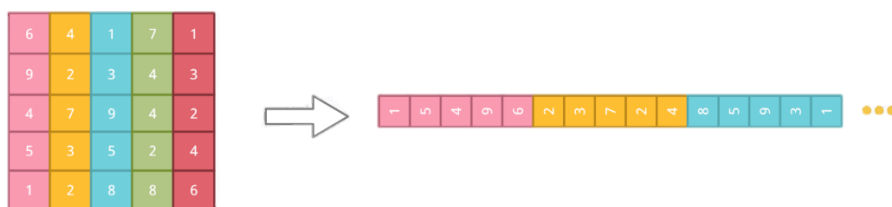


شکل ۲- Spectrogram بدست آمده از صدای ضبط شده ویالون

(الف) در مورد spectrogram و نحوه محاسبه آن از روی سیگنال حوزه زمان با استفاده از تبدیل Gabor توضیح دهید.

(ب) در این بخش می‌خواهیم spectrogram سیگنال‌های صوتی را بدست بیاوریم. برای اینکار می‌توانید از توابع موجود در کتابخانه‌های matplotlib و یا scipy استفاده نمایید. برای کاهش حجم محاسبات به هنگام آموزش مدل، سائز تصاویر خروجی spectrogram را 64×64 در نظر بگیرید. spectrogram مربوط به سیگنال‌های صوتی را بدست آورده و در یک تانسور (آرایه چندبعدی) ذخیره نمایید.

(پ) حال لازم است تا از spectrogram‌های بدست آمده ویژگی‌های معنادار استخراج نماییم. یک راهکار ساده تبدیل ماتریس دو بعدی spectrogram به یک بردار یک بعدی است.^۷ بدین ترتیب در خروجی یک بردار $X \in \mathbb{R}^{1 \times N_t \times N_f}$ خواهیم داشت که در آن N_t تعداد نقطه‌های نمونه‌برداری شده از زمان و N_f تعداد نقطه‌های نمونه برداری شده از فرکانس می‌باشند.



شکل ۳- مثالی از تبدیل ماتریس به بردار

(ت) با استفاده از بردار ویژگی بدست آمده در بخش (پ)، مدل‌های مختلف را برای رسیدن به کمترین خطا بر روی داده‌های تست آموزش دهید. همانند قبل مدل‌های مختلفی را که آموزش می‌دهید به همراه هایپرپارامترهایشان را در یک جدول و دقت بهترین مدل بدست آمده بر روی داده‌های آموزش، ارزیابی و تست را در جدولی دیگر گزارش دهید. ماتریس Confusion را بر روی داده‌های تست نیز گزارش دهید و بهترین مدل را به نام **Model_Phase1b.pkl** با استفاده از ماژول Pickle ذخیره نمایید.

^۶ Violin

^۷ Flatten

ث) تابعی با نام Predict_Phase2 بنویسید که ورودی آن ماتریس داده‌های صوتی و آدرس مدل ذخیره شده در بخش (ت) است. این تابع ابتدا مدل را فراخوانی کرده و با اعمال پیش‌پردازش‌های لازم، طبقه‌بندی را انجام داده و مقادیر خروجی را در فایلی به نام phase1b_predicted.npy ذخیره می‌نماید. در نهایت داده‌های تست phase1_test.npy را برای دریافت پیش‌بینی انجام شده توسط مدل به تابع موردنظر بدهید.

ج) نتایج بدست آمده را با بخش آموزش مدل بدون استخراج ویژگی مقایسه نمایید.

چ) (امتیازی) علاوه بر روش ارائه شده در قسمت (پ)، راه‌حل‌های دیگری که بتواند ویژگی‌های بهتری استخراج کرده و به نتایج بهتر منجر شود، مشمول نمره اضافه خواهند بود. (لازم است تا در گزارش پروژه در مورد معیار و یا الگوریتم استفاده شده، به طور کامل توضیح داده شود)

فاز سوم – یادگیری بدون نظارت به عنوان پیش پردازش

همانطور که میدانیم، استفاده از روش یادگیری با نظارت زمانی امکانپذیر است که داده‌ها به همراه برچسبشان موجود باشند. فرایند برچسب‌گذاری باید به صورت دستی توسط اپراتور انسانی برای تک تک داده‌ها به صورت مجزا انجام شود که کاری بسیار زمان‌بر است. امروزه تعداد داده‌هایی که برچسب‌گذاری شده‌اند در مقایسه با آن‌هایی که برچسبی ندارند، به مراتب کمترند. یکی از کاربردهای روش‌های بدون نظارت، آن است که به سیستم اجازه دهیم تا نمایش‌های^۸ مورد نیاز برای تشخیص یا طبقه‌بندی ویژگی‌ها را از داده‌های خام به طور خودکار کشف کند.^۹ بنابراین می‌توانیم با در اختیار داشتن تعداد محدودی داده برچسب‌دار، داده‌هایی که بدون برچسب هستند را به صورت نیمه خودکار برچسب‌گذاری نماییم.

توجه: در این قسمت می‌توانید از Spectrogramهای بدست آمده از سیگنال‌های صوتی به عنوان داده‌های ورودی استفاده نمایید.

دیتاستی که در فاز اول با آن کار کردید را در نظر بگیرید. می‌خواهیم شبیه‌سازی‌ای برای برچسب‌گذاری داده‌ها انجام دهیم. به همین منظور فرض می‌کنیم که برای ۲۰۰۰ نمونه‌ای که در اختیار داریم، برچسبی موجود نمی‌باشد و می‌خواهیم تنها ۵۰ نمونه را به صورت دستی برچسب‌گذاری نماییم و سپس سایر داده‌ها را با روشی نیمه خودکار برچسب‌گذاری کنیم.

سوالی که مطرح است این است که کدام نمونه‌ها را برچسب‌گذاری نماییم تا بیشترین دقت در خروجی حاصل شود. آیا برچسب‌گذاری تعدادی نمونه از هر کلاس به صورت تصادفی عملکرد بهتری دارد یا آنکه می‌توانیم برچسب‌گذاری را دقیق‌تر انجام دهیم؟

الف) ابتدا از هر کلاس به تعداد برابر (۵ نمونه) را به صورت تصادفی انتخاب کنید و برچسب آن‌ها را از برچسب‌هایی که در اختیارتان هست، قرار دهید و یک دیتاستی با ۵۰ نمونه بسازید. سپس یک طبقه‌بند بر روی داده‌های آماده شده آموزش دهید و نتایج بدست آمده را گزارش نمایید. **روی داده تست**

ب) با استفاده از یک روش Clustering نظیر K-Means داده‌های خام را به ۵۰ خوشه تقسیم کنید.

پ) سپس از هر خوشه یک نمونه که تا حد خوبی نماینده ویژگی‌های سایر نمونه‌ها باشد را انتخاب و ۵۰ نماینده بدست آمده را با استفاده از برچسب‌هایی که در اختیارتان است، برچسب‌گذاری نمایید. (توجه: برای انتخاب نماینده هر خوشه می‌توانید از معیار فاصله تا مرکز خوشه استفاده نمایید).

ت) داده‌هایی که در بخش (پ) به صورت دستی برچسب‌گذاری کرده‌اید را برای آموزش طبقه‌بندی که در بخش (الف) بکار بردید، استفاده نمایید و نتایج را گزارش و تحلیل کنید. **روی داده تست**

ث) تا الان در هر خوشه یک نماینده داریم که دارای برچسب است. برای هر نمونه، برچسب متناظر را برابر با برچسب نماینده خوشه‌ای که متعلق به آن است قرار دهید. بدین ترتیب تمامی داده‌ها را برچسب‌گذاری کرده‌اید. همچنین می‌توانید با استفاده از این ۵۰ داده یک طبقه‌بند آموزش داده و از آن برای برچسب‌گذاری داده‌ها استفاده نمایید.

^۸ Representations

^۹ [Representation Learning](#)

ج) در نهایت برای دیتاست آماده شده در بخش (ث) یک طبقه‌بند آموزش دهید و نتایج بدست آمده را با بخش‌های (الف) و (ت) مقایسه و تحلیل نمایید.