

Learning challenge

Last updated: 2018-11-30

For this assignment you will participate in groups of 3 students in a learning challenge. You will need to submit your predictions, as well as a report describing your solutions and the code which you used to generate the predictions.

This assignment is worth 40% of your course grade. The assignment grade will be based on the quality of your work as reflected in the report and code (50%), as well as your ranking in the competition (50%).

Your report should be **2 pages maximum**, and should include the following:

- Description of your computational learning experiments, including:
 - feature engineering
 - learning algorithm(s) tried
 - parameter tuning
 - discussion of the performance of your solution
- Detailed specification of the work done by group members
- The name of the account under which you submitted your results to the competition on Codalab (see below)

You will also need to submit your Python code and the instructions on how to run it. Your code can be either a plain Python script, or an IPython notebook. Include all the code and data necessary to re-run your experiments. Put the report in **PDF format**, the code and the data in a single zip file named with your group number, e.g. **Group7.zip**, and submit it to the BlackBoard assignment.

In addition you will need to submit your solution file the competition server. The competition is hosted on <https://competitions.codalab.org>. One member of the team will need to get a codalab account, and will be responsible for submitting your solution. Indicate the name of this account in your report. See section **Submission to Codalab** for additional details.

IN SUMMARY: submission consists of two parts

1. Zip file with your report and your code (BLACKBOARD)
2. Submission of your solution (CODALAB)

Group work

Your report needs to contain a detailed description of who did what, so make sure to keep track of this information.

Note: it is **not acceptable** to just say *All members worked together and contributed equally*.

If there are any problems with collaboration, such as serious disagreements, a group member not contributing, or a group dissolving, make sure inform the course coordinator as soon as possible via email.

Code reuse rules

Remember this assignment is group work. You are **not allowed** to collaborate or share code with students outside your group. **Submissions will be checked for plagiarism.**

If you are found breaking the above rules you will be reported to the Board of Examiners for fraud.

You are, however, allowed to use code examples provided by the instructor during the course, or as part of the competition.

Dataset

Simple speech recognition

In this challenge the task is to learn to recognize which of several English words is pronounced in an audio recording. This is a multiclass classification task.

Data files The dataset is available for download on BlackBoard. It contains the following files:

- **wav.tgz**: a compressed directory with all the recordings (training and test data) in the form of **wav** files
- **feat.npy**: an array with Mel-frequency cepstral coefficients extracted from each **wav** file. The features at index **i** in this array were extracted from the **wav** file at index **i** of the array in the file **path.npy**.
- **path.npy**: an array with the order of **wav** files in the **feat.npy** array.
- **train.csv**: this file contains two columns: **path** with the filename of the recording and **word** with word which was pronounced in the recording. This is the training portion of the data.
- **test.csv**: This is the testing portion of the data, and it has the same format as the file **train.csv** except that the column **word** is absent.

You can load the files **npz** using the function `numpy.load`, and the CSV files using the `csv` module or the `pandas.read_csv` function.

Evaluation metric The evaluation metric for this task is accuracy (the proportion of correct predictions).

Method There are three important restrictions on the method used:

- the method should be fully automatic, that is, by re-running your code it should be possible to re-create your solution file;
- the method shouldn't use any external dataset which contains the exact same data as the provided data files;
- every software component used should be open-source and possible to install locally. This means that you cannot use proprietary closed-source speech recognition software, or access a web service to carry out any data processing.

Some hints:

- You can use the provided MFCC features for the spoken utterances, or you can extract your own features from the `wav` files.
- Use part of the provided training data as a validation set. Only submit to Codalab after validating your results on this your validation data.

Submission format The submission format is the same as the file `test.csv` with the added column `word` with your predicted word.

The competition is hosted on Codalab at the following URL: <https://competitions.codalab.org/competitions/2067aee4019-ac3a-133b9f2abb88>

You can submit your results in the **Participate** link. After uploading your file, make sure to *submit to leaderboard*.

Over the course of the competition you can make 7 submissions. The results from all the participating teams will be displayed in the **Results** tab.

The submission file should be a `.zip` file with a file named `result.csv` in it. (Make sure there are not additional subdirectories in the zip file.) Your file needs to use a valid CSV format. It is recommended to use the Python `csv` library or the `pandas.to_csv` function to create the file.