You can use the data in this folder to derive a starting feature set for automatic joke success ratings.

**ground\_truth\_ratings.csv** has the human-labeled verdicts on joke success

* Column E has the human rater’s score when considering audience laughter both during and after the joke
* Column F has the human rater judgement while pretending to be the robot (looking only at the period after the joke for laughter)
* The spreadsheet also tracks which performance and which joke the responses came from

**The zip files** contain data from two real robot performances. (We have 24 total usable performances that we can analyze, but developing a data extraction script/protocol for just a few is a good starting place.)

* The jokes and post-joke responses in each folder are numbered sequentially, and their order (starting with joke 0) matches the order of jokes in that particular routine in the ground truth spreadsheet, so you can match the extracted audio features to the correct human-rated response as you go
* Note that the joke numbering in each of these performance folders do not match the joke IDs in the ground truth spreadsheet. The joke IDs have a different purpose: to identify which jokes are the same over the complete dataset so that we can analyze things like individual joke success, etc.

A good place to start is making a spreadsheet with the human ground truth ratings from column F, performance ID, joke ID, and whatever features you extract using Pratt. Each row will be one observation (i.e., a single post-joke audio recording), and each column will be a class label (i.e., the human ground truth rating) or a feature.

Once you have the data organized in this way, MATLAB’s Classification Learner app or Python’s scikitlearn can be good tools for trying out different machine learning models. (You can also export code from MATLAB’s app to move towards a usable code view of the machine learning approaches that you tried in the app.)

Don’t forget the need to cross-validate; if we use all the data we have to train a model and then test with that same data, the performance of the classifier will be unrealistically good (overfit). By withholding some of the data during training, we can get a better idea of how our classifiers will perform on new observations. If we withhold data from a full performance (the leave-one-”subject”-out cross validation I mentioned), we can get a pretty good idea of how well our models would work on a new/future performance.