JASA EXPRESS LETTERS ARTICLE asa.scitation.org/journal/jel

An automated method for detecting F0 measurement jumps based on sample-to-sample differences

Jeremy Steffman[a] and Jennifer Cole
Northwestern University, Evanston, Illinois 60208, USA
jeremy.steffman@northwestern.edu, jennifer.cole1@northwestern.edu

Abstract: An algorithm for detecting sudden jumps in measured F0, which are likely to be inaccurate measures, is introduced. The method computes sample-to-sample differences in F0 and, based on a user-defined threshold, determines whether a difference is larger than naturally produced F0 velocities, thus, flagging it as an error. Various parameter settings are evaluated on a corpus of 30 American English speakers producing different intonational patterns, for which F0 tracking errors were manually checked. The paper concludes in recommending settings for the algorithm and ways in which it can be used to facilitate analyses of F0 in speech research.

[Editor: Douglas D. O'Shaughnessy] https://doi.org/10.1121/10.0015045

Received: 14 July 2022 **Accepted:** 17 October 2022 **Published Online:** 1 November 2022

1. Introduction

Though it might be tempting to take accurate F0 measurement for granted in speech research, this is by no means a trivial issue. Speech science and language researchers who deal with F0 measurement know that F0 estimation is prone to error, whether due to noise in the signal, junctures between consonants and vowels, or variation in voice quality, all of which pose problems in the measurement of F0 [see, e.g., Kawahara *et al.* (2005) and Xu (1999) for discussion of errors in F0 estimation due to irregular voicing and consonant junctures]. In this paper, we introduce a simple but effective metric for identifying F0 measurement errors, and we provide an assessment of its effectiveness under various parameter settings, using F0 trajectories from a large corpus of American English intonation. Two developments in speech and phonetics research have increased the need for such a tool. First, the size of data sets under consideration has grown in the past several decades. For researchers using large data sets and speech corpora, manual inspection of all F0 measures in a data set may be prohibitively time-consuming and resource-intensive. Second, speech researchers across disciplines have developed a growing interest in the study of dynamic patterns that unfold over time. Though single-point or interval-mean measures are still informative in the study of F0 in speech, time-series analyses have gained in popularity. Yet a single measurement error in a time-series of F0 measures can pose serious problems for analysis.

The idea behind the tool presented here is that errors in F0 measurement often involve sudden jumps in F0. This method accordingly will not address F0 measurement errors that do *not* result in sudden F0 changes, a caveat that should be kept in mind by users, though in our experience most errors in F0 measurements involve sudden sample-to-sample changes. An open access R script for implementing the algorithm can be retrieved from https://github.com/jsteffman/f0-jumps.

1.1 Benefits of an estimator-independent error detector

There are many existing approaches to mitigating errors in F0 estimation. These approaches generally perform error correction/elimination in the process of F0 estimation, such that the user is not aware of when and where errors have been identified by the F0 estimator. For example, in Praat (Boersma and Weenink, 2020), octave jump errors can be eliminated by adjusting the F0 floor and ceiling and other parameters, such as the octave jump cost (though we do not compare our method directly to F0 estimated with Praat, we do compare it to an octave jump detector in Sec. 3.1, which provides an assessment of how eliminating octave jumps compares to the algorithm we introduce here). Other estimators, such as Harvest (Morise, 2017), eliminate errors in F0 estimation by checking sample-to-sample differences (as does our method) and removing samples that exceed a certain percentage threshold of preceding samples.

The tool we describe here is intended to be complementary to these F0 estimator-dependent error corrections in that it (1) can be applied to F0 from any estimator that outputs non-zero F0 values[1] and (2) is intended to provide both a

[a]Author to whom correspondence should be addressed.

way of removing errors that persist after attempted corrections from an estimator and a readout of files that contain errors, how many errors are within a file, and for which specific samples they occur. An additional use of the algorithm, described in Sec. 4, is to identify files with flagged errors that may contain a property of interest (e.g., non-modal phonation). The present algorithm can thus be used in a more hands-on fashion as compared to estimator-internal error correction and serves a complementary purpose: the assessment and analysis of errors that persist after F0 has been estimated or simply the exclusion of files that contain persistent errors. Another proposed advantage of the algorithm is that, unlike previous approaches, it is grounded in physiological thresholds from speech production, which provides an informed metric for identifying errors.

1.2 Rate of F0 change in speech production

How quickly is too quickly for F0 to change across successive samples? To answer this question, we draw on speech production research that quantifies the maximum rate of change in F0 during speech production (Ohala and Elwan, 1973; Sundberg, 1973; Xu and Sun, 2002). The method of eliciting rapid F0 changes in these studies is to ask speakers to produce an oscillating glissando between high and low F0. These productions are imitative in the sense that the speakers are prompted to produce speech from an auditory model that may be musical (Sundberg, 1973) or based on resynthesized F0 in human speech (Xu and Sun, 2002). From these reproduced F0 oscillations, one can calculate the time it takes for the speaker to go from the minimum F0 value to the maximum for a given F0 movement. Another metric reported in Sundberg (1973) is "reaction time," which analyzes the central portion of an F0 movement in which 6/8 of the movement occurs, starting from the minimum and ending at the maximum value. This region of the F0 movement generally contains a steeper slope (higher velocity) than the beginning and end of the movement, as velocity increases at the start of the F0 movement and then decreases as the F0 target is approached. The rate of change in this higher velocity interval is then calculated. As our purpose is to identify reasonable rate of change maxima to exclude changes that exceed them, we rely on the reaction time measure, using data from Sundberg (1973) as an input to the algorithm (though, importantly, these thresholds can be adjusted in the script that implements the algorithm). Sundberg (1973) and Xu and Sun (2002) show that various factors influence the rate of change of F0. Overall, F0 movements are made more quickly for female speakers than for males and for trained singers than for those without voice training. In addition, falling F0 movements occur faster than F0 rises.

1.3 The algorithm

The process for detecting errors described here is implemented as follows. Take an ordered time-series of F0 measurements at times 1 through n: $F0_{t1 \ldots} F0_{tn}$.

(1) $F0_{t+1}$ is subtracted from $F0_t$, yielding the sample-to-sample difference d.
(2) d is compared to a threshold T; if $d > T$, an "error" is flagged. This is coded as 1. If $d < T$, a 0 is coded, for no error.
    - Specific thresholds for sample-to-sample falling F0 ($F0_{t+1} < F0_t$) or rising F0 ($F0_{t+1} > F0_t$) may be used, capturing the different rates of change evident in the speech production literature referenced above.
(3) The procedure is repeated for all samples in the time-series, yielding a string of 0s and 1s corresponding to all pairs of successive samples in the time-series.

From this string, the script produces a time-series record of errors for all samples, as well as a summary for each unique F0 trajectory identifying (1) the number of sample-to-sample differences that exceeded the threshold, (2) whether *any* sample-to-sample differences exceeded the threshold (coded as a binary 0 or 1), and (3) the proportion of sample-to-sample differences in the trajectory that exceeded the threshold.[2]

The script is designed to print and save this information for each trajectory that had any sample-to-sample differences exceeding the threshold. It can also optionally summarize this information by speaker if the data set contains multiple speakers.

The parameters that need to be set by the user are the spacing between F0 measurements in time (defaulting to 10 ms intervals, which is the recommended setting) and the threshold(s) for determining how fast is too fast. Below, we additionally review and assess other parameter settings that can be modified, and we evaluate the output of the algorithm on a corpus of intonational melodies. Note that in what follows, a less-strict threshold setting below corresponds to a larger tolerated sample-to-sample change in F0. The tested parameters are listed here.

(1) *Rise/fall thresholds*: Should rising F0 movements and falling F0 movements be subject to different thresholds? If no, the less-strict falling threshold is used.
(2) *Male/female thresholds*: Should speaker gender be considered in setting thresholds? If no, the less-strict female threshold is used.
(3) *Tolerate samples*: Should a small number of sample-to-sample differences exceeding the threshold in the same F0 trajectory be tolerated? Values tested are 0 (none tolerated), 1, or 2.

We compare combinations of each of these settings and how they balance hit and false alarm rates in Sec. 2.

2. Assessing the automated method

Our strategy in assessing the effectiveness of the algorithm was to run it with the strictest parameter settings we considered (described in Sec. 2.2). We then performed a manual inspection of all files flagged as containing errors by these settings. As will be described, the settings proved "too strict" in the sense that some files in the manual audit were re-included by way of manual checking. In Sec. 3, we present possible adjustments to the algorithm's parameters that balance hit rate (the number of correctly identified files) with false alarms (files that should not be flagged, but were by the algorithm).

2.1 Corpus for assessment

The data we use to assess the algorithm come from an imitative speech production study that examined intonation in American English [see Cole *et al.* (2022) for details on this experiment; see also Chodroff and Cole (2019)]. In the experimental paradigm, speakers listen to an auditory model representing a given intonational pattern and reproduce the intonational melody in the model when producing a new sentence.[3] The auditory model sentences were designed to have eight different intonational patterns[4] instantiated on the final word of the model sentences, which was always a trisyllabic, stress-initial name. The intonational models (i.e., the stimuli for the imitation) were created by re-synthesizing F0 using the PSOLA method in Praat (Boersma and Weenink, 2020; Moulines and Charpentier, 1990). See Chodroff and Cole (2019) and Cole *et al.* (2022) for more details on the intonational melodies in question and the design of the experiment. Measurements were extracted from the phrase-final trisyllabic stress-initial words. Figure 1(A) shows the mean scaled ERB (Equivalent Rectangular Bandwidth) of eight tune shapes (with F0-tracking errors removed) to give a sense of the variety of F0 trajectory shapes that were elicited. Thirty self-reported native American English speakers each completed 144 trials in the experiment (a combination of tune, target sentence, and model sentence). These data are particularly useful as a

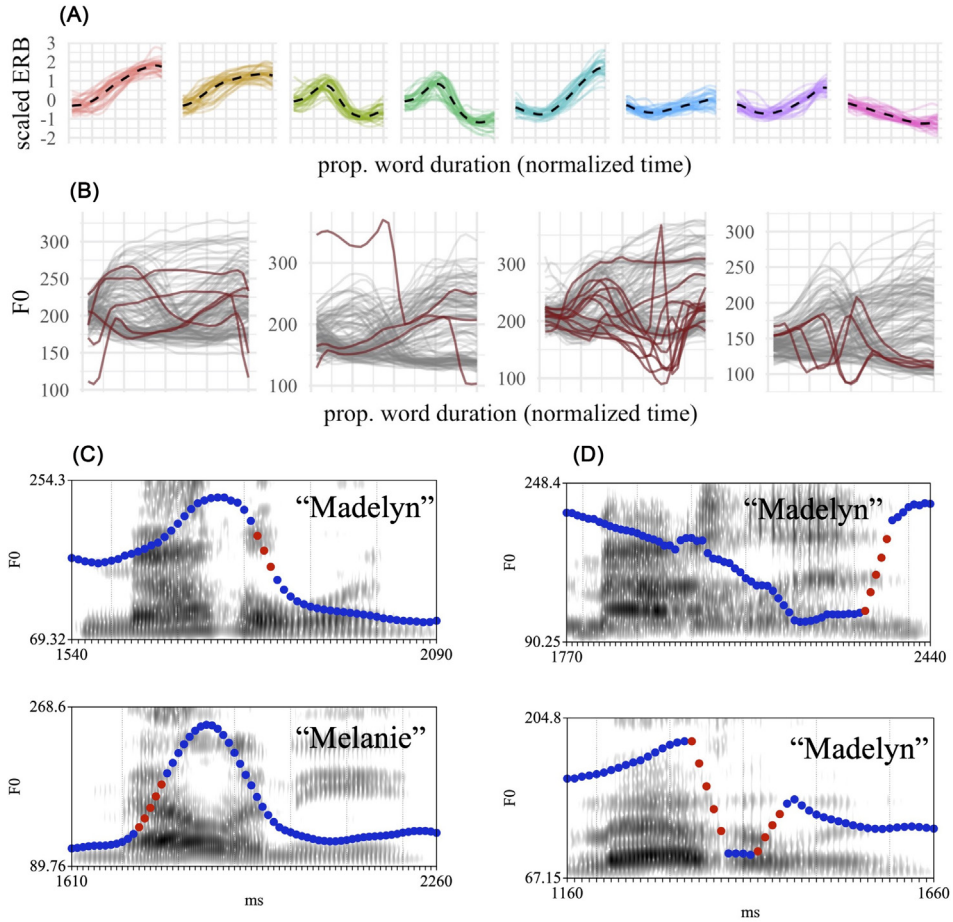

Fig. 1. (A) The eight F0 shapes in the corpus. Black dashed lines are the grand mean, and colorful lines are speaker means. (B) Four example speakers showing trajectories flagged as containing errors (brown lines) and those that were not (gray lines). The average duration for each panel is 557, 535, 678, and 468 ms, respectively (left to right). (C) F0 overlaid on spectrograms for two files that were flagged as containing an error with the strictest settings but later identified as a fast F0 movement. Samples whose differences exceed the threshold are in red. (D) Two files that were authentic errors.

corpus to assess F0-tracking errors as they contain speech from 30 speakers and a variety of shapes of F0 movements, with F0 measures taken from utterance-final position, where non-modal phonation can make F0 tracking difficult. Trials with disfluencies and speech errors were excluded. The total number of tokens submitted to the error-detection algorithm was 4254. F0 was computed in VoiceSauce (Shue *et al.*, 2009), using the STRAIGHT algorithm (Kawahara *et al.*, 2005). F0 was measured at a 10 ms sampling rate.

2.2 Parameters and thresholds

Here, we evaluate the efficiency of several parameter settings in the algorithm. These are listed in Sec. 1.3. Recall that a less-strict threshold setting corresponds to a larger tolerated change in F0.

Sundberg (1973) examined F0 rate of change at different F0 interval sizes, (i.e., imitating smaller vs larger changes in F0). We selected the rate of change that was fastest from these different intervals, which happened to be the largest interval. From this interval, we took the mean rates of change (from the reaction time measure) for male and female speakers who were *not* trained singers.[5]

The first run of the algorithm employed the strictest settings that we assumed would provide a conservative threshold for F0 rate of change. These were (1) rise- and fall-specific thresholds, (2) speaker gender-specific thresholds, and (3) a tolerance of zero sample-to-sample errors. Figure 1(B) shows the results of the error flagging procedure with these settings in the F0 trajectories from four example speakers. Brown lines are those that were flagged as containing an error. As can be seen, the automated method has successfully flagged trajectories with sudden discontinuities that are unlikely to be a reflection of actual F0. In general, we found the algorithm to be effective along these lines.

We subsequently carried out a manual audit of the 507 files that were flagged as containing an error by these strictest settings. The goal here was to assess whether an error was correctly identified by the algorithm. Manual assessment of the data was carried out by visual and auditory inspection by three trained auditors. Visualization of the F0 measured in STRAIGHT with the offending samples highlighted in red [as in Figs. 1(C) and 1(D)] was inspected as was the corresponding sound file. An error was confirmed if the measured F0 did not accurately reflect the perceived pitch at that location in the audio file. An error was disconfirmed when perceived pitch comported with the measured trajectory and when that trajectory showed a smooth change in F0 without sudden discontinuities [Fig. 1(C)]. Of the 507 files audited manually, 51 were determined not to be actual errors in F0 measurement [of the sort in Fig. 1(C)]. Thus, in total, about 10% of the flagged files (51/507) were incorrectly labeled by the algorithm, which, if left unchecked, could lead to 1% of the total corpus being incorrectly marked as an error (51/4254). In this sense, the strictest threshold is potentially "too strict": it flags files as containing errors that are assessed by the trained auditors to involve fast (and accurately measured) F0 changes.

3. Comparison of settings

In this section, we compare the algorithm across various parameter settings, adjusting the three parameters given in Sec. 2.2. For each run of the algorithm under each setting combination, we computed the number of hits [correctly identified errors, as in Fig. 1(D)] and false alarms [errors that were flagged but were in fact just fast F0 movements as in Fig. 1(C)], taking the manually audited files as the standard of comparison. From these, we then computed a d′ score for each individual speaker, allowing us to examine the spread of d′ scores across parameter settings.[6]

The originally used strictest setting, shown as the leftmost red bar in Fig. 2(A), has both the highest number of hits and the highest number of false alarms (see corresponding plots of raw hits and false alarms in the online supplementary image at https://osf.io/4hqdn/). Changing a setting that reduces false alarms also reduces hits. The d′ metric shown in Fig. 2(A) allows us to balance these considerations, and the distribution of values shows some settings are clearly better than others. In particular, allowing any sample-to-sample differences to exceed the threshold is detrimental, and not encoding a difference between F0 rises and falls in the threshold settings greatly reduces d′ to be at or below chance (0), as shown in Fig. 2(A). The effect of including speaker gender in the thresholds is comparatively minimal. The mean highest d′ is, in fact, for parameter settings that use only the more lax female speaker thresholds (and otherwise are as strict as possible).

We complemented the d′ analysis of settings by computing the F-score[7] for each parameter setting combination, shown in Fig. 2(B). The F-score assessment is largely in line with the d′ analysis in suggesting the stricter parameter settings are best, the two highest F-scores being the strictest setting (0.947) or the same parameter settings but without gender-specific thresholds (0.923).

In summary, it seems that the strictest parameter settings are fairly optimal for our corpus in balancing hits and false alarms, with the exception of the inclusion of gender-specific thresholds. Adding gender-specific thresholds provides only marginal improvement in the F-score and no improvement in d′. In this regard, we suggest that gender-specific thresholds may not be worth including, unless the researcher is specifically interested in this question. In the version of the script linked above, we accordingly do not include gender-specific threshold settings (though the script can be modified to include them).

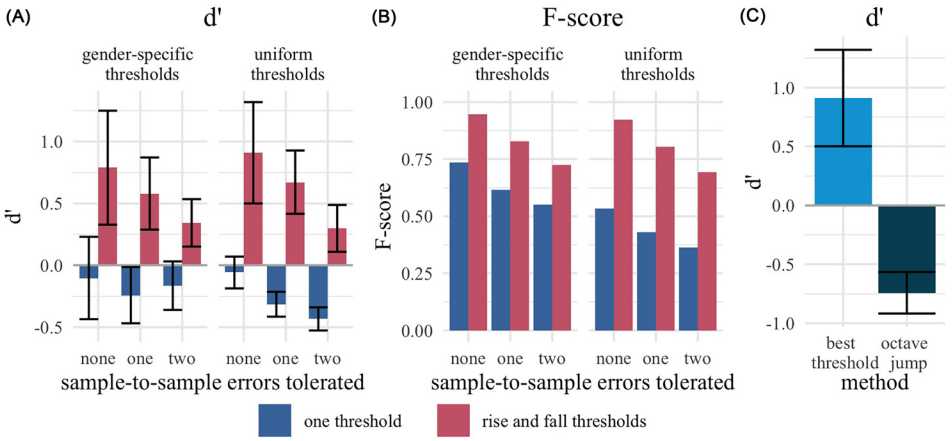

Fig. 2. (A) d′ scores for each parameter setting with bars showing 95% confidence intervals computed over individual speaker d′ values for each parameter setting. (B) F-scores by parameter setting. (C) d′ for the best combination of parameter settings, as compared to an octave jump detector.

3.1 Comparison to an octave jump detector

As noted in Sec. 1.1, one way to minimize errors in F0 tracking is to remove octave jumps in F0 estimation. In Praat (Boersma and Weenink, 2020), this can be accomplished by adjusting the parameters in the Advanced Pitch settings. In this section, we briefly compare performance of our algorithm to one that detects only octave jumps. We computed octave jumps on the basis of a change in F0 that is equivalent to pitch halving or doubling or was larger than this.

Because our goal in this paper is not to compare F0 estimation algorithms, we computed octave jumps from the same F0 measurements in the corpus that we used for the assessment of the algorithm above. We then computed d-prime and F-scores to compare this octave jump detector to the best settings of our algorithm (the strictest settings without gender-specific thresholds, as described above). An octave jump is, in a sense, a very extreme sample-to-sample change. As such, we expected that the octave jump detector should agree with the threshold method in terms of the errors that it flags but that it might miss smaller errors that exceed the threshold but do not rise to the level of an octave jump. The results confirm this. Though the false alarm rate for the octave jump detector was among the lowest of all parameter combinations (2%), the hit rate was correspondingly low (15%). In other words, 15% of files that were manually assessed as containing errors contained octave jumps, and the remaining 85% contained manually identified errors smaller than an octave jump. The F-score for the octave jump detector was thus low (0.36), owing to its low count of true positives. The d′ comparison, shown in Fig. 2(C), agrees with this in indicating that there is a substantial improvement in d′ using the threshold method. This allows us to conclude that, in our corpus at least, detecting/removing octave jumps will catch the largest F0 jumps, but the physiologically informed thresholds offer an improvement over the octave jump criterion in their ability to flag subtler errors.

4. Conclusions

In this paper, we presented a simple but effective tool for identifying F0 tracking errors in time-series data, and we assessed how various parameter settings for the algorithm perform. In conclusion, we offer some views of how this tool can be used in larger-scale F0 research. In our own research, the purpose of the algorithm has been to identify files that will be excluded from our data set, for which our analysis has focused on time-series F0 measures. For this purpose, the algorithm can be paired with a manual audit to re-enter any false alarms into the corpus for analysis. In our experience across several large corpora of the sort presented here (all examining utterance-final F0 patterns), approximately 10% of the files in the corpus are flagged as containing an error (though this rate may be lower when F0 measures are not coming from the end of a phrase). Further, approximately 10% of the flagged files (1% of the total data) are false alarms. Thus, if the researcher is willing to potentially lose 1% of usable data in their corpus, manual intervention is not needed. On the other hand, if the researcher has the time and resources, a manual audit of flagged files allows them to ensure that all files that can be analyzed are included. This is also much less time-intensive than auditing the entire corpus.

Another use for this tool may be to flag files that contain properties of interest (non-modal voicing, jumps to falsetto voice). For example, voice quality features such as creaky voice are often characterized by irregular F0 and can lead to F0 measurement errors (Kawahara *et al.*, 2005; Keating *et al.*, 2015). The algorithm may thus be useful for identifying files containing creak for further inspection, especially if the researcher suspects there is creaky voicing in their data. It has also been noted that changes to falsetto voice engender sudden F0 jumps, though they tend to occur when the speaker is

in a specific modal F0 range. Future developments of the algorithm could incorporate this in considering the value from which a jump occurs, which may be a useful way to identify jumps to falsetto voice (based on empirical data in this vein).

In sum, we believe that the present tool can be fruitfully applied to large-scale F0-based speech research both as a tool for ensuring accuracy in F0 measures and potentially for addressing other research questions.

Acknowledgments

We thank Chun Chan and Lisa Cox for help with the experiment creation. This project was supported by National Science Foundation Grant No. BCS-1944773.

References and links

[1]Some algorithms output zeros for samples where an F0 error is detected. The present algorithms will flag these values as errors, though this is in a sense redundant because they are already labeled as such by virtue of being 0.

[2]The script additionally computes what we call "carryover" errors. These are F0 values that follow a flagged error directly in time and do not necessarily exceed the threshold but are *within* a threshold of the error's value. In other words, these are F0 values that may be wrong because they follow a sudden jump up or down in F0, with values that are close to the errorful F0. See the supplementary image at https://osf.io/4hqdn/ for an example of this. This measure is computed by starting at the start a of a flagged error in F0 measurement and looping through the following vector of F0 samples. Any samples that are within a threshold of that error's flagged F0 value and follow the original error or another carryover error (from a previous loop) linearly in time are flagged as potential carryover errors, though we note these samples may not necessarily be inaccurate (e.g., if the actual error was just a localized perturbation in the midst of a rising F0 movement).

[3]The auditory model sentences in the study were "She quoted Helena," "He answered Jeremy," and "Her name is Marilyn." The sentences that speakers produced were "She quoted Helena," "They honored Melanie," and "She remained with Madelyn." The final word in each case is a tri-syllabic name with initial stress.

[4]These eight tunes represent the combination of two pitch accents (H^*, L^*), phrase accents, and boundary tones described in the prevalent model of American English intonation (Beckman and Pierrehumbert, 1986; Pierrehumbert, 1980).

[5]We also considered if thresholds from trained singers in Sundberg (1973) were appropriate for correctly identifying errors. These thresholds proved to be far too lax, missing many errors. As such, we do not report on them here.

[6]The formula for d′ is

$$d' = z(H) - z(F),$$

where H and F are the hit and false alarm rates (for the algorithm on an individual speaker), which are z transformed.

[7]The F-score is computed as

$$TP/[TP + 0.5(FP + FN)],$$

where TP is the number of true positives (correctly identified F0 errors), FP is the number of false positives (files identified as an error that should not have been), and FN is the number of false negatives (F0 errors that were not flagged). It can also be described as $2 \times [(precision \times recall)/(precision + recall)]$, where precision is the number of true positives divided by the number of all positives (here, flagged errors), and the recall is the number of accurately identified positives divided by all files that should have been flagged as an error.

Beckman, M. E., and Pierrehumbert, J. B. (**1986**). "Intonational structure in Japanese and English," Phonol. Yearb. **3**, 255–309.

Boersma, P., and Weenink, D. (**2020**). "Praat: Doing phonetics by computer (version 6.1.09) [computer program]," http://www.praat.org (Last viewed October 25, 2022).

Chodroff, E., and Cole, J. (**2019**). "Testing the distinctiveness of intonational tunes: Evidence from imitative productions in American English," in *Proceedings of INTERSPEECH 2019*, September 15–19, Graz, Austria, pp. 1966–1970.

Cole, J., Steffman, J., and Tilsen, S. (**2022**). "Shape matters: Machine classification and listeners' perceptual discrimination of American English intonational tunes," in *Proceedings of Speech Prosody 2022*, May 23–26, Lisbon, Portugal, pp. 297–301.

Kawahara, H., Cheveigné, A. d., Banno, H., Takahashi, T., and Irino, T. (**2005**). "Nearly defect-free F0 trajectory extraction for expressive speech modifications based on STRAIGHT," in *Proceedings of INTERSPEECH 2005*, September 4–8, Lisbon, Portugal.

Keating, P. A., Garellek, M., and Kreiman, J. (**2015**). "Acoustic properties of different kinds of creaky voice," in *Proceedings of the 18th International Congress of Phonetic Sciences*, August 10–14, Glasgow, Scotland.

Morise, M. (**2017**). "Harvest: A high-performance fundamental frequency estimator from speech signals," in *Proceedings of INTERSPEECH 2017*, August 20–24, Stockholm, Sweden, pp. 2321–2325.

Moulines, E., and Charpentier, F. (**1990**). "Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones," Speech Commun. **9**(5-6), 453–467.

Ohala, J., and Elwan, W. (**1973**). "Speed of pitch change," J. Acoust. Soc. Am. **53**, 345.

Pierrehumbert, J. B. (**1980**). "The phonology and phonetics of English intonation," Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA.

Shue, Y.-L., Keating, P., Vicenik, C., and Yu, K. (**2009**). "Voicesauce [computer program]," http://www.seas.ucla.edu/spapl/voicesauce/ (Last viewed October 25, 2022).

Sundberg, J. (**1973**). "Data on maximum speed of pitch changes," Speech Transm. Lab. Q. Prog. Status Rep. **4**, 39–47.

Xu, Y. (**1999**). "Effects of tone and focus on the formation and alignment of f_0 contours," J. Phon. **27**(1), 55–105.

Xu, Y., and Sun, X. (**2002**). "Maximum speed of pitch change and how it may relate to speech," J. Acoust. Soc. Am. **111**(3), 1399–1413.