

# Automatic Maxima Detection: A Graphical User Interface and a Tutorial

**Frédéric Dandurand**  
*Université de Montréal*

**Thomas R. Shultz**  
*McGill University*

Automatic Maxima Detection is a statistical tool to detect and measure local maxima in functional data (Dandurand & Shultz, 2010). A typical use of AMD is the identification of growth spurts in time-varying data. The current paper provides a complete hands-on tutorial for how to use AMD, a new graphical user interface (GUI) and advanced scripts.

Functional data, typically measured over time, are ubiquitous. For instance, physical height changes over time, and so do temperatures and amounts of precipitation, stock market indices, average national incomes, physiological or brain responses as a function of a time-varying stimulus, etc. Functional data are characterized by the fact that they change over time, and researchers are often interested in characterizing the dynamics of that change. Two types of events in the measure of interest are particularly important: local minima and maxima and spurts. Spurts are characterized by a local increase in the rate of change, that is, a maxima in the velocity (first derivative) of the measure of interest with respect to time.

The task of identifying maxima, minima and spurts in functional data is made difficult by the fact that the data measured are often noisy. The noise can be caused by measurement error, or could be due to external factors unrelated to the variable of interest. For instance, a researcher may be primarily interested in assessing the size

of the mental lexicon of a child over time by measuring the number of words uttered by a child; but other factors could also play role such as the environment in which the measure is taken, the mood and health status of the child when the measure is taken, etc. Because of these random fluctuations, apparent maxima and minima are not necessarily significant, and it is important to identify which ones are robust and reliable.

We previously introduced a technique called Automatic Maxima Detection (AMD) to detect and measure local maxima in functional data (Dandurand & Shultz, 2010). AMD was applied to the identification of significant growth spurts in physical height data (Berkeley growth study : Tuddenham & Snyder, 1954) and in children's vocabulary size: simulated data (McMurray, 2007; Mitchell & McMurray, 2008) and experimental data (Corrigan, 1978). In follow-up work done on more densely sampled children's vocabulary size (Ganger, 2004; Ganger & Brent, 2004), AMD found multiple vocabulary spurts of varying intensity and location in almost all children (Dandurand & Shultz, 2011). In physical height data, AMD successfully identified and measured three important and well-known phenomena of physical growth of children: (1) An adolescent growth spurt in virtually all children; (2) an earlier age of onset for girls' adolescent growth spurts than for boys'; and (3) a smaller, pre-adolescent growth spurt in some children. Such spurts can be obscure and difficult to detect without techniques like AMD (Dandurand & Shultz, 2010).

A complete technical description of AMD along with sample analyses can be found in Dandurand and Shultz

---

Frédéric Dandurand (frederic.dandurand@gmail.com), Department of Psychology, Université de Montréal, 90 avenue Vincent-d'Indy; Montréal, QC H2V 2S9 Canada. Thomas R. Shultz (thomas.shultz@mcgill.ca), Department of Psychology and School of Computer Science, McGill University, 1205 Penfield Avenue, Montreal, QC H3A 1B1 Canada. This work is supported by a grant to TRS and a post-doctoral fellowship to FD, both from the Natural Sciences and Engineering Research Council of Canada.

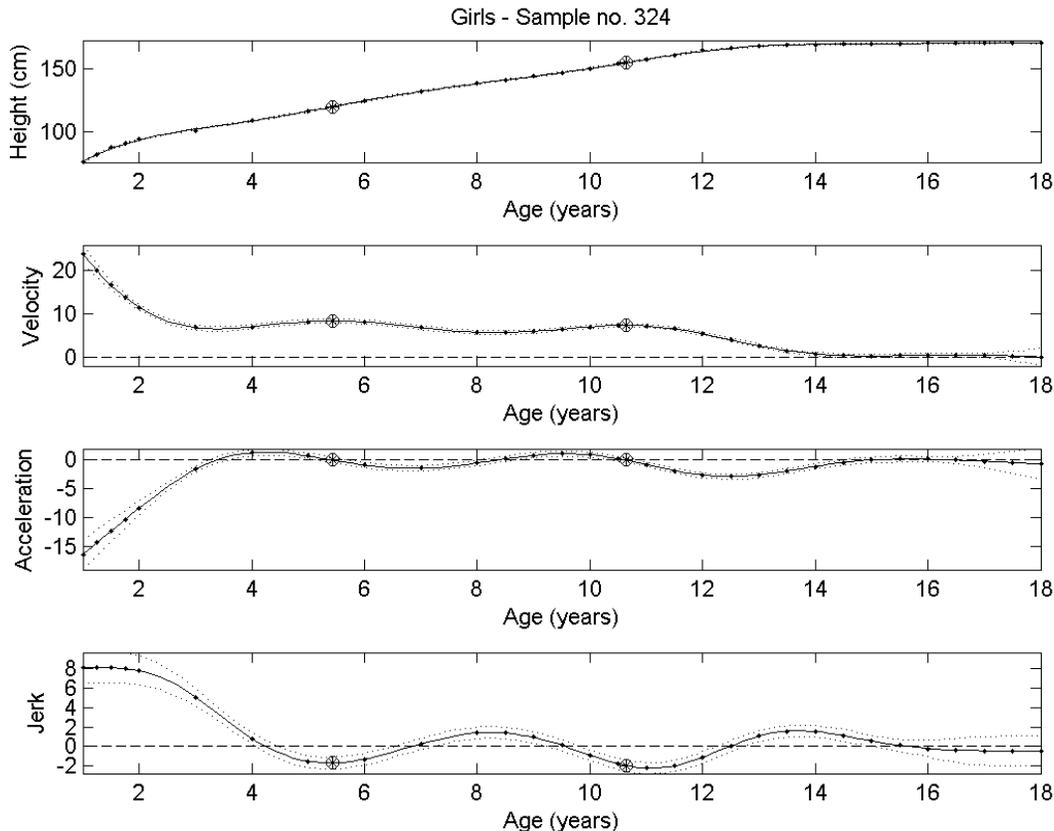


Figure 1 - Example of a figure generated by AMD for individual case data. The figure shows four plots, from top to bottom: the fitted curve and its first three derivatives (velocity, acceleration and jerk). AMD found two growth spurts (i.e., points of maximum velocity). The spurts are marked as large circles at the location where they occur, on each plot.

(2010). In order to make AMD easier for novice users, the current paper introduces a new graphical user interface, and provides a hands-on tutorial and instructions to run AMD simulations.

As most other software, AMD is going through ongoing improvements. Effort is made to preserve backward compatibility, but using later versions may require some modifications of the scripts. Likewise, the graphical user interface may change in future versions. AMD comes bundled with Functional Data Analysis (FDA: Ramsay & Silverman, 2005). AMD 1.1 has been tested with Matlab R2009b (32 bits) and R2010b (64 bits). Compatibility with other versions of Matlab cannot be guaranteed. The current paper describes AMD version 1.1; available for download at: <http://lnslab.org/lib/AMD/>

#### AMD overview

AMD takes as input a sample of data pairs  $(y_i, t_i)$  where  $y_i$  is a measure of interest (e.g., vocabulary size) and  $t_i$  is corresponding sampling time. AMD then fits a smooth and

continuous spline-based function to these data. The level of smoothing is specified by the user. AMD finds and quantifies significant spurts or velocity maxima in individual records, outputs these measures to a file, and also generates figures for individual cases and a summary plot of the data set. Figure 1 shows examples of plots for an individual in the Berkeley growth study.

There are two ways for running analyses in AMD: using a script or the new graphical user interface (GUI).

#### The GUI

The easiest method of interaction with AMD is using its graphical user interface (GUI). To start the GUI, start Matlab, change the current folder in Matlab to the AMD directory (ex.: AMD\_1.1), then type `AMD_GUI`. Figure 2 shows the AMD GUI.

The GUI has 5 main sections: (1) input data, (2) labels, (3) AMD analysis parameters, (4) directories, and (5) status. When data are available for analysis, the "Run AMD..." button becomes enabled. Press it to launch an AMD

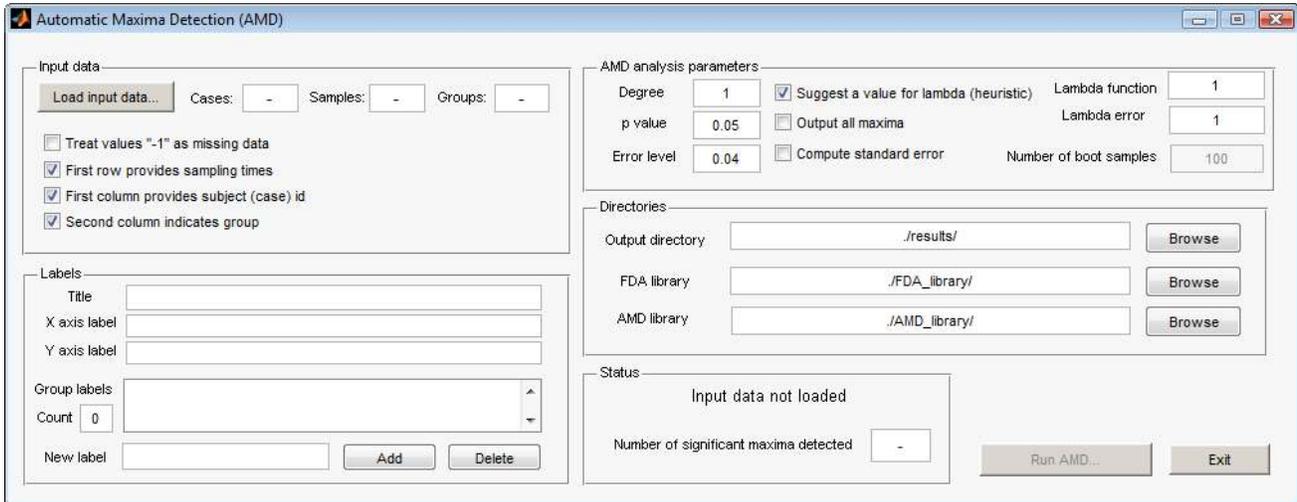


Figure 2 – Screen shot of the AMD GUI.

simulation.

### Input Data

Input data for an AMD analysis are loaded from a text file (see file format below). Before pressing the “Load input data...” button, the user can specify four options:

1. Treat values -1 as missing data. If some data are missing and marked as -1 in the file, AMD can optionally repeat the last known observation to fill them in. If this option is not activated, AMD will treat these -1 as numerical values.
2. First row provides sampling times. If sampling times are not provided in your data file (that is, the first row is the first case), uncheck this option. Samples will then be automatically numbered from 1.
3. First column provides subject (case) ID. If case IDs are not provided, uncheck this option. Case IDs will then be automatically numbered from 1.
4. Second column indicates group. If group IDs are not provided, uncheck this option. All cases will then be put in group 1.

After loading the file, AMD will show, on the right of the “Load input data...” button, how many data entries or cases were found, how many samples per case, and how many groups the cases belong to.

### Input file format

Input data files must be in text format and must provide a rectangular matrix of numbers, in which columns are separated by whitespaces (spaces or tabs). Table 1 is an example of input data for the Corrigan study (Corrigan.dat file) where the colors are to be interpreted as:

- **First row** (optional): indicates sampling times
- **First column** (optional): indicates case (participant) ids
- **Second column** (optional): indicates which group the corresponding participant belongs to
- The actual data are given in **orange**.
- The values in **blue** (intersection of first row and first two columns, if used) are ignored.

### Labels

All labels are optional. They are used on graphs and plots. Users can specify the title of the study, X and Y axes labels, and labels for the groups. Note that, if group labels are provided, the number of labels (shown under “Count”) must match the number of groups in the data file (“Groups” in “Input Data”). To add a group label, type it in the “New label” section, and press the “Add” button. To delete a label, select it in the scroll list showing all labels entered, and press the “Delete” button.

After data are loaded, the GUI shows the following

Table 1. Data from the Corrigan study

0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	0	0	0	0	1	4	5	3	3	10	21	23	18	52	53
2	1	1	1	3	15	22	30	29	22	33	49	41	78	82	97	104
3	1	0	0	1	1	3	6	7	6	6	14	22	26	27	38	33

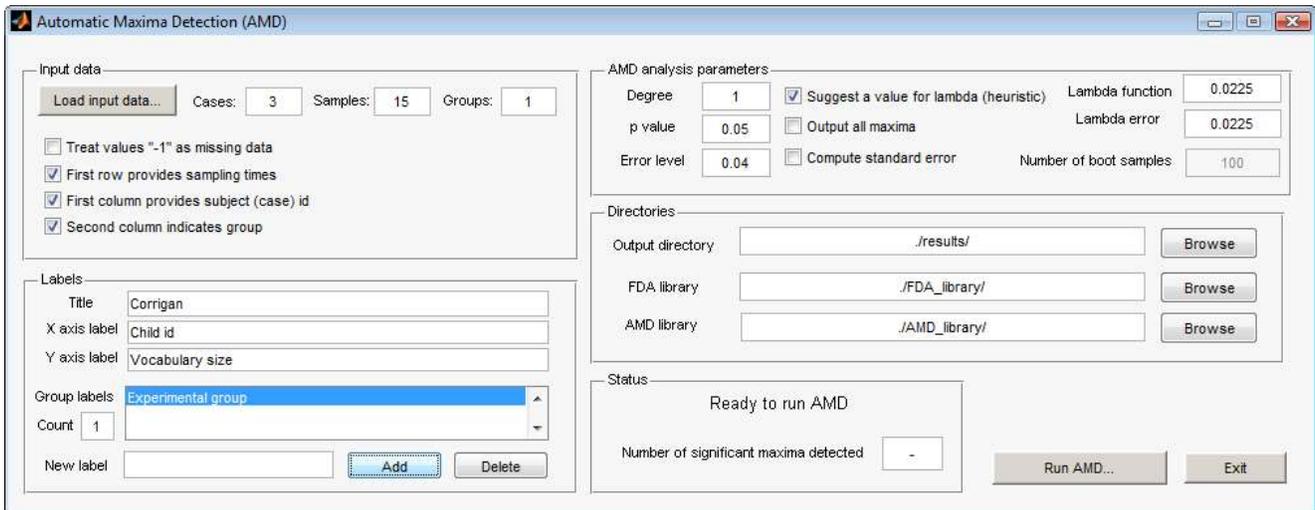


Figure 3 - Screen shot of AMD GUI after data were loaded, and labels provided. The data file loaded contains 3 cases (subjects) belonging to a single group. The number of samples per case is 15.

information loaded from the file: (1) number of cases (e.g., subjects or participants), (2) number of samples (observations) per case, and (3) number of groups.

An example of screen shot of the AMD GUI is given in Figure 3.

#### AMD analysis parameters

In this section, users specify the AMD analysis parameters:

- Degree. Indicates in which derivative to look for maxima. Typical values used are 0 to look for maxima in the function itself (0<sup>th</sup> derivative), and a value of 1 to look for spurts, i.e., maxima in the velocity curve (1<sup>st</sup> derivative). Degree is set to 1 by default.
- P value. Indicates the level of statistical significance to use for the computation of confidence bands used to test statistical significance of maxima (see the section on AMD technical details). The p value is set to .05 by default.
- Error level. Specifies the acceptable tolerance in numerical operations, such as finding the x values corresponding to a zero crossing on the y axis. The default value is 0.04.
- Suggest a value for lambda. Select this option to use a heuristic function to select a value for the smoothing parameter lambda. While a careful manual selection is preferred (Ramsay, Hooker, & Graves, 2009), previous experience with AMD showed that the optimal smoothing value of lambda tends to increase with the number of observations. A plausible hypothesis for this observation goes as follows. Lower sampling rates act as low-pass frequency filters, and result in fewer observations. If the process of interest changes rather

slowly and is thus already captured accurately by the lower end of the frequency spectrum, the higher frequencies obtained using a higher sampling rate will be essentially noise. Selecting a larger smoothing value in FDA then reduces this noise. However, without much knowledge of the process that generates the measured data of interest, it is advisable to err on the side of a larger sampling rate to make sure the full frequency spectrum of the process of interest is covered in the data sample obtained. For the current version of the AMD GUI, the following function was programmed, which provided an empirically satisfactory match to our previous lambda selections (ranging from 0.01 to  $1 \times 10^6$  for number of observations between 11 and 4000):

$$\lambda = \frac{(\text{number of observations})^2}{1 \times 10^5}$$

Further research will be needed to rigorously analyze of the relationship between number of samples and smoothing, and to provide an optimal heuristic function. At any rate, suggesting a heuristic value for lambda does not represent an endorsement from AMD; expert judgement should always be exercised when deciding the amount of smoothing necessary for a given data set.

- Lambda function. Indicates the value of lambda for smoothing the function fitting the data.
- Lambda error. Indicates the value of lambda for the standard error function (confidence bands). James Ramsay (personal communication) mentioned that there are different approaches to setting the value of lambda for the standard error curve. One approach is to use the same value as the one specified for the fitted function, whereas other techniques suggest smoothing the standard error curve more than this function. In the

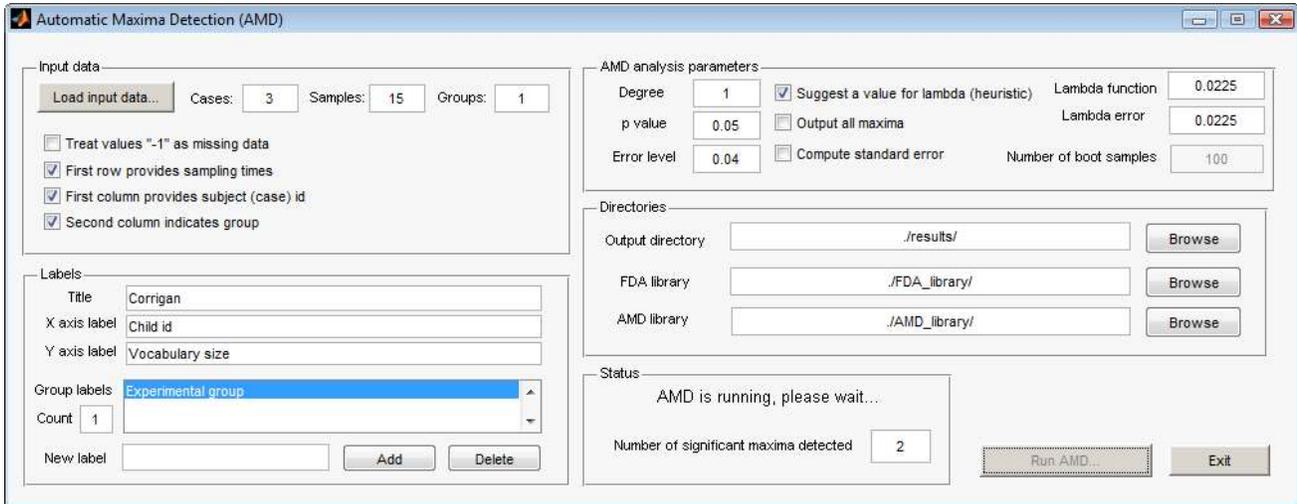


Figure 4 - Screen shot of AMD while analysis is in progress.

AMD GUI, heuristic values of lambda for the fitted function and the standard error curve are the same.

- Output all maxima. Check this option to output all maxima found, irrespective of significance. By default, this option is not checked so that only significant spurts are reported in the output file.
- Compute standard error. Check this option to compute standard error values for these measures of the identified maxima: (1) point where the maximum is most intense, (2) when the maximum starts, (3) its amplitude and (4) its duration.
- Number of boot samples. This field activates when the “Compute standard error” option is activated (checked). AMD uses the specified number of boot samples in its bootstrap technique to estimate the standard error of the measures of the identified maxima.

### Directories

In this section, users can specify where, in their local file directory, to store output results (output directory), and where the FDA and AMD libraries’ code is installed. Although users may wish to change the default output directory (./results, which corresponds to .../AMD\_1.1/results where AMD\_1.1 is the AMD installation directory), changing the FDA and AMD generally won’t be necessary unless users wish to specify different versions than the ones included in the AMD download (not recommended, for compatibility reasons). To change a directory, press the corresponding “Browse” button.

### Running AMD

After a data file is loaded, AMD is ready for performing analysis. Press the “Run AMD...” button. Progress of the

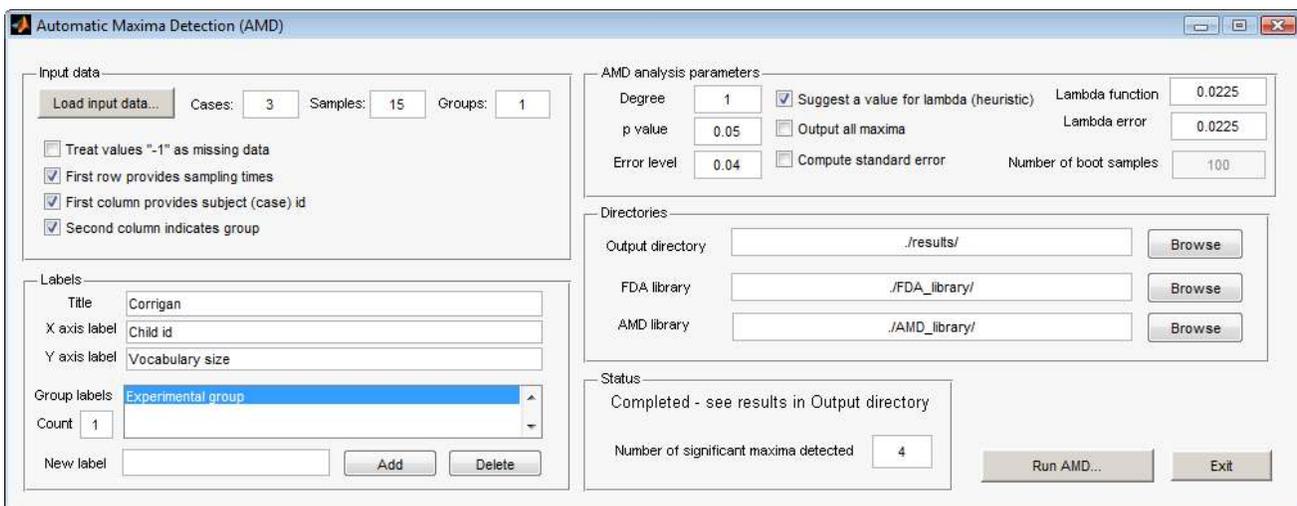


Figure 5- Screen shot of AMD after analysis is completed.

Table 2. Spurts in the Corrigan study without standard deviation.

<i>groupID</i>	<i>caseID</i>	<i>maxID</i>	<i>beginLoc</i>	<i>centerLoc</i>	<i>amplitude</i>	<i>duration</i>
1	1	1	4.480804	5.434781	5.793429	2.099821
1	1	2	8.751569	9.828274	14.12413	2.058305
1	1	3	12.8004	13.75351	61.68768	2.199598
1	2	1	2.702577	4.089185	13.62895	2.846945
1	2	2	10.91551	11.90228	23.18147	1.961466
1	3	1	4.400536	5.338916	4.896737	2.072165

AMD analysis can be monitored in the Status section. The number of spurts detected so far is also indicated in the Status section. Screen shots of the AMD GUI during and after an analysis are presented in Figure 4 and Figure 5 respectively.

After an analysis is completed, results can be found in the output directory specified. The next section explains how to interpret these results.

### AMD Results

As an example, here are the files generated for the Corrigan study.

- AMD\_spurts.txt
- case\_1\_.png
- case\_2\_.png
- case\_3\_.png
- spurts.png

These files can be found in the output directory.

### Output file format

The first file, AMD\_spurts.txt, contains measures of the spurts found. Fields are separated by spaces. Two file formats are used, depending on whether or not standard error information was computed. Table 2 shows an example of spurts in the Corrigan study, without standard error.

Table 3 is an example of spurts data for the Corrigan

study, with standard error (computed with 100 boot samples).

The first two columns identify the case (and group) that the maximum belongs to. The third column is an identifier of the spurt, since a case can have multiple spurts. The following columns report the measures of the maximum (see section “Some AMD technical details” for a description of what these measures refer to Figure 10):

- *beginLoc*. Location where the maximum started on the X axis.
- *stderrorBegin* (if outputted). Standard error of the *beginLoc* measure.
- *centerLoc*. Location on the X axis where the maximum was most intense.
- *stderrCenter* (if outputted). Standard error of the *centerLoc* measure.
- *Amplitude*. Intensity of the maximum.
- *stderrAmp* (if outputted). Standard error of the *Amplitude* measure.
- *Duration*. Duration of the maximum.
- *stderrDur* (if outputted). Standard error of the *Duration* measure.

As reported in Dandurand and Shultz (2010), these measures can be further subjected to statistical analyses such as ANOVAs for identifying differences in spurt measures between groups.

Table 3: Spurts in the Corrigan study, with standard deviations

<i>groupID</i>	<i>caseID</i>	<i>maxID</i>	<i>beginLoc</i>	<i>stderrorBegin</i>	<i>centerLoc</i>	<i>stderrCenter</i>	<i>amplitude</i>	<i>stderrAmp</i>	<i>duration</i>	<i>stderrDur</i>
1	1	1	4.480804	0.137615	5.434781	0.112895	5.793429	0.53182	2.099821	0.07925
1	1	2	8.751569	0.127572	9.828274	0.127902	14.12413	0.449818	2.058305	0.03986
1	1	3	12.8004	0.11189	13.75351	0.111299	61.68768	1.33235	2.199598	0.02010
1	2	1	2.702577	0.112237	4.089185	0.062534	13.62895	1.172657	2.846945	0.09480
1	2	2	10.91551	0.095622	11.90228	0.064398	23.18147	1.066465	1.961466	0.08826
1	3	1	4.400536	0.290167	5.338916	0.347145	4.896737	0.378114	2.072165	0.14575

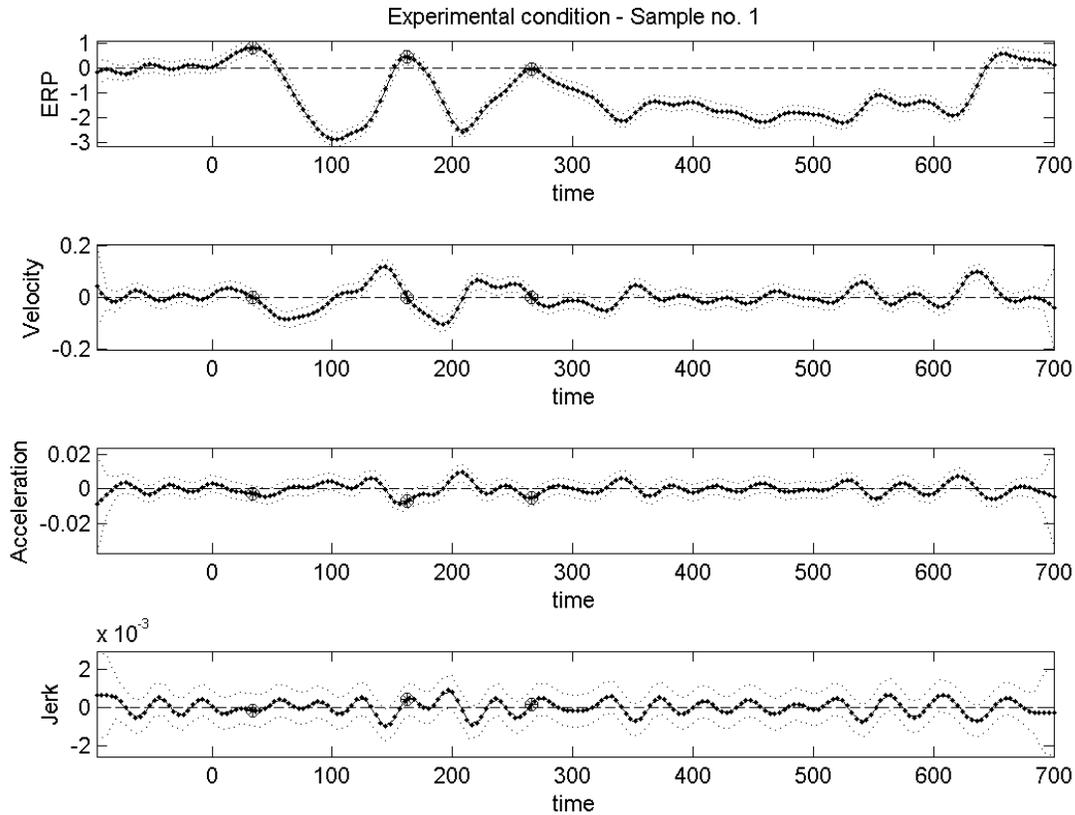


Figure 6 - Example of an AMD graph of an individual case. Here, degree was set to 0 for identifying maxima in ERP data.

### How to interpret figures

AMD outputs two types of plots: for individual curves, and a summary plot of all measures of the maxima. Individual curves are generated for each case.

#### *Graph of individual cases with maxima in the function (Degree = 0)*

When degree is set to 0, AMD seeks maxima in the function itself. As an illustration, we applied AMD to event related potential (ERP) data. Figure 6 shows AMD graphs for an individual electrode. At the top the smooth function can be seen as a thick line (with the original sample data as dots), along with the first three derivatives, aligned along the X axis. The dotted lines below and above each curve show confidence band values for a p value of .05. AMD found three significant maxima\* in this curve (34.0, 162.3 and

\* Finding minima in the ERP function (so-called N components) is also possible with AMD by inverting the signs of all elements in the data matrix loaded prior to analysis. Future versions of AMD will enable users to select for the detection of maxima and/or minima.

265.9 ms, identified by circles). As we can see, these maxima occur in the function itself (curve on the top row).

#### *Graph of individual cases with maxima in the velocity curve (Degree = 1)*

When degree is set to 1, AMD seeks maxima in the velocity curve, that is, it looks for significant spurts. Figure 7 shows AMD graphs for case no. 1 of the Corrigan data. This graph is interpreted exactly as the previous one. The only important difference is that maxima occur in the velocity curve (curve on the second row).

#### *Summary graph*

A summary graph of maxima measures for all cases is also generated by AMD; see Figure 8 for an example.

### Some AMD technical details

This section gives a brief overview of the more technical aspects of AMD. A complete technical description can be found in Dandurand and Shultz (2010).

AMD takes as input a sample of data pairs  $(y_i, t_i)$  where  $y_i$  is a measure of interest (e.g., vocabulary size) and  $t_i$  is corresponding sampling time. AMD first uses Functional

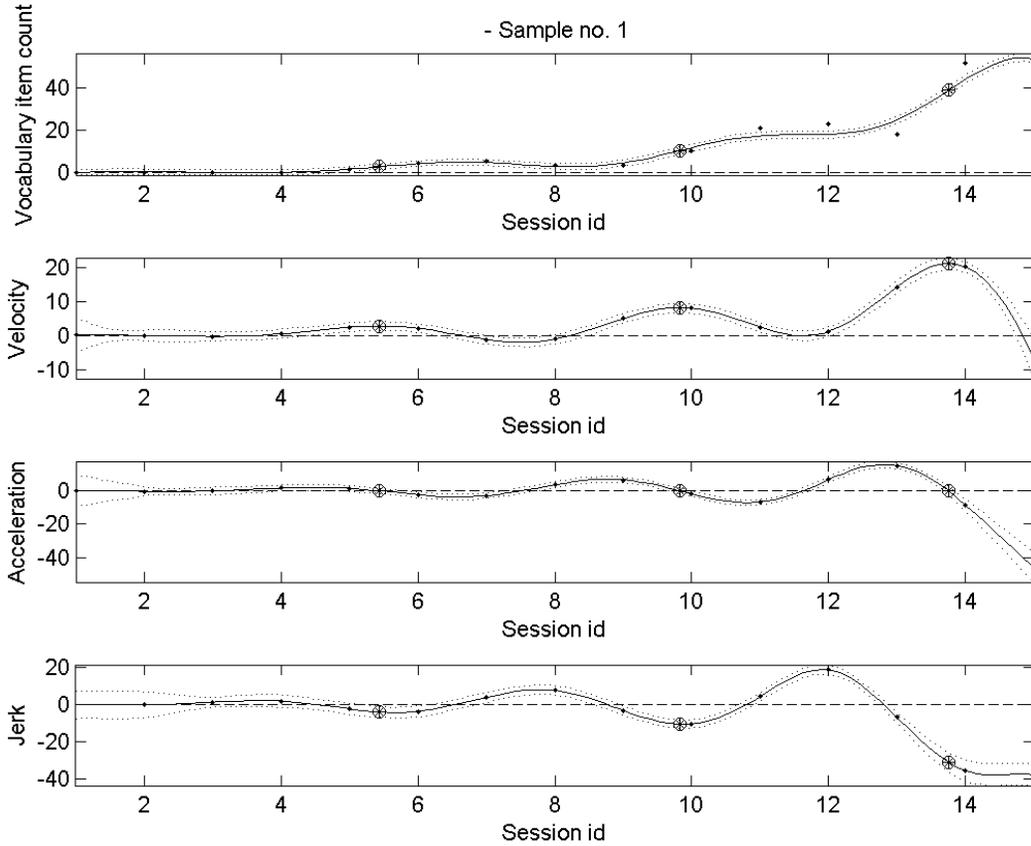


Figure 7 - Example of an AMD graph of an individual case of the Corrigan study. Here, degree was set to 1 for identifying spurts (maxima in velocity) in vocabulary growth data

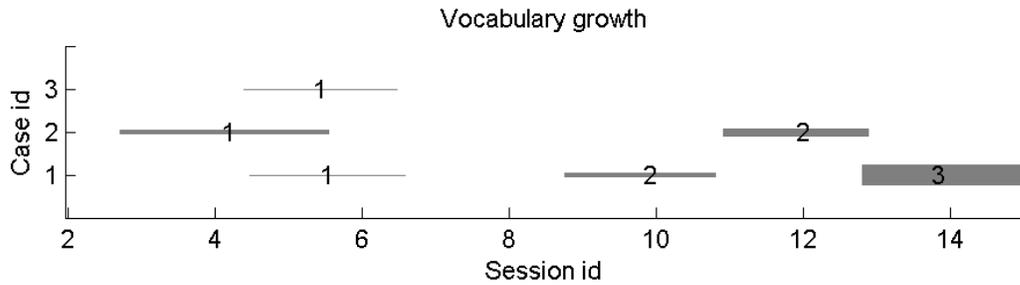


Figure 8 - Example of an AMD summary graph. The Corrigan data contain 3 cases: AMD identified 3 significant spurts in case 1, two spurts in case 3, and one spurt in case 3. Each spurt is identified by a line whose end points show where the spurt began and ended, and whose thickness indicates the intensity of the spurt. The number on the line indicates spurt ID, and is located where the maximum was most intense.

Data Analysis (AMD: Ramsay & Silverman, 2005) to fit a spline-based smooth function to approximate the data sample, as well the first three derivatives of this smoothed, continuous function to identify important function markers (see Figure 6 and Figure 7). The fitted function takes the form of a weighted linear combination of B-spline basis functions  $\phi_k$ ,  $k = 1, \dots, K$ :

$$x(t) = \sum_k^K c_k \phi_k(t)$$

FDA uses a roughness penalty approach to smoothing which limits or penalizes the size of some higher-order derivative of the smoothed function. Coefficients  $c_k$  are selected to minimize a penalized sum of squared errors (SSE) between the estimated function and observed data vector  $y$ :

$$PENSSE(y|c) = (y - \phi_c)'W(y - \phi_c) + \lambda c' Rc$$

where:  $c$  is a vector of coefficients  $c_k$ ;  $W$  is a symmetric

positive definite weight matrix;  $\Phi$  is the matrix of basis function values  $\phi_k(t)$ ; lambda ( $\lambda$ ) is a smoothing parameter, and  $R$  is a roughness penalty matrix, computed as follows:

$$R = \int \left[ \frac{d^2}{dt^2} \phi(t) \right] \left[ \frac{d^2}{dt^2} \phi'(t) \right] dt$$

Note that the fitted curve  $x(t)$  becomes increasingly smooth as lambda ( $\lambda$ ) increases; this smoothing value lambda is the only parameter in AMD that is manually set (now with the help of a suggestion of a heuristically-based value in the AMD GUI). There are techniques to automate selection of lambda (Dandurand & Shultz, 2010). However, as mentioned, existing techniques have important limitations, and so a careful manual selection is preferred (Ramsay et al., 2009).

### Determining which maxima are significant

To determine which maxima are significant, AMD first estimates confidence bands in the function and its derivatives based on the p value provided (see dotted lines surrounding the fitted curve in Figure 6 and Figure 7). Width of confidence intervals (also called point-wise bands) is based on the variance of the fitted function:

$$Var[x] = \Phi' Var[c] \Phi$$

where  $\Phi$  is a matrix of basis function values at the observation points and  $Var[c]$  is the variance of coefficients  $c_k$ , computed as follows:

$$Var[c] = (\Phi' W \Phi)^{-1} \Phi' W \sum_e W \Phi (\Phi' W \Phi)^{-1}$$

where  $W$  is a symmetric positive definite weight matrix; and  $\sum_e$  is the variance-covariance matrix of the residual vector  $\varepsilon$ .

Second, AMD lists all local maxima in the target curve (depending on the degree specified) as maxima candidates.

Finally, for each candidate, a null hypothesis is tested in which a straight line between the two local minima adjacent to the maximum is contained within the confidence band. As can be seen in Figure 9, this null hypothesis thus corresponds to the absence of a maximum. A candidate is a genuine maximum when this null hypothesis has to be rejected.

### Quantifying spurts

As mentioned, AMD also provides rigorous quantification of the important features of significant spurts: (1) when the spurt starts, (2) the point where it is most intense (maximal velocity), (3) the spurt amplitude and (4) the spurt duration. As illustrated in Figure 10 for the case of a spurt, a spurt starts when velocity is at an inflection point, acceleration is at a local maximum, and jerk crosses 0 at a negative slope. A spurt peaks when velocity is at a local maximum, acceleration crosses 0 at a negative slope, and jerk is negative. A spurt ends when velocity is at an inflection point, acceleration is at a local minimum, and jerk crosses 0 on a positive slope. Spurt amplitude is given by the vertical distance from acceleration at the start to acceleration at the end.

### Sample scripts

Finally, for more advanced users, AMD can be run using scripts which are fully customizable. Four sample scripts are provided in the "sample\_scripts" directory: script\_berkeley.m; script\_corrigan.m; script\_mcmurray; and script\_ganger which run AMD simulations for the data sets corresponding to their names. They can be customized for other purposes.

Scripts are typically composed of five sections: (1) setting global parameters, (2) loading input data, (3) setting

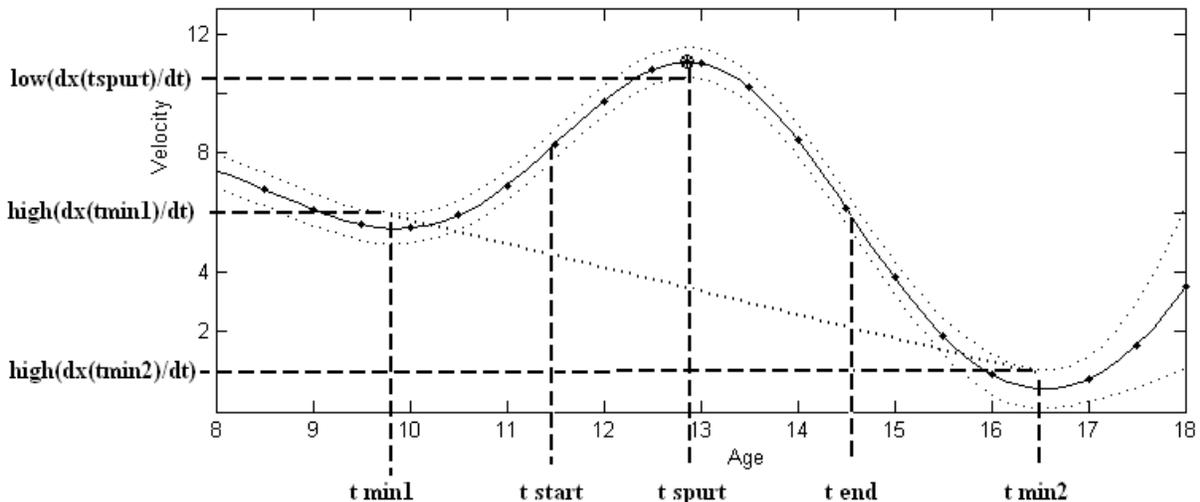


Figure 9 - Example of a significant spurt. Dotted lines above and below the smooth function correspond to 95% confidence bands.

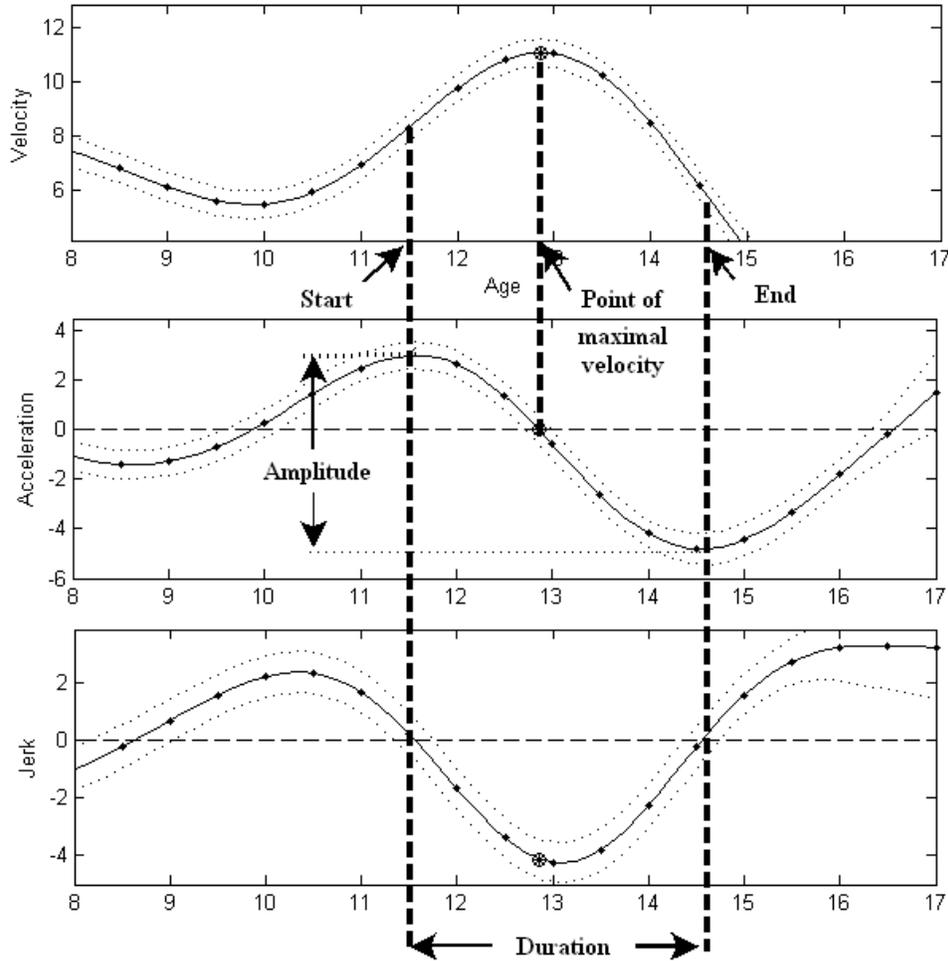


Figure 10 - Example of quantification of a spurt features: where it starts, where it is most intense, its amplitude and duration.

simulation parameters, (4) finding spurts and outputting individual curves, and (5) outputting maxima data and a summary curve.

#### Setting global parameters

The first two lines clean up Matlab memory and graphs:

```
clear all
close all
```

The following commands load the AMD and the FDA libraries' code:

```
addpath('../AMD_library')
addpath('../FDA_library')
```

The next line loads some AMD constants:

```
global_constants;
```

#### Loading input data

The following lines load data from the file, and sets appropriate variables. In this example, the first column indicates subject id, the second column

```
data = load('../data/Berkeley/berkeley.dat');
```

```
sample_times = data(1,3:end);
subject_ids = data(2:end,1);
group_ids = data(2:end,2);
data(:,1) = []; % first column is subject id, so drop
data(:,2) = []; % second column is group id, so drop
data(1,:) = []; % first row is sample times, so drop
```

#### Setting simulation parameters

Setting parameters involves populating a template called **simulation\_parameters\_template**. Details of the fields of this template can be seen in **global\_constants.m**. Fields names are self-explanatory, and the following items are the direct correspondents in the AMD GUI described above: **x\_label**, **y\_label**, **degree**, **pvalue**, **output\_all\_maxima**, **compute\_std\_errors**, **boot\_samples**, **lambda\_fct** and **lambda\_stderr**.

```
lambda = 0.1
```

```
simulation_parameters = simulation_parameters_template;
simulation_parameters(1).title = 'Height of children (Berkeley data)';
simulation_parameters.subject_ids = subject_ids;
```

```

simulation_parameters.sample_ts = sample_times;
simulation_parameters.x_label = 'Age (years)';
simulation_parameters.sample_ys = data;
simulation_parameters.y_label = 'Height (cm)';
simulation_parameters.group_ids = group_ids;
simulation_parameters.group_labels = {'Boys', 'Girls'};
simulation_parameters.ncase = size(subject_ids,1); % number of curves
(cases) in the raw data
simulation_parameters.degree = 1;
simulation_parameters.pvalue = 0.05;
simulation_parameters.error_level = 1e-4;
simulation_parameters.output_all_maxima = false;
simulation_parameters.compute_std_errors = false;
simulation_parameters.boot_samples = 0;
simulation_parameters.lambda_fct = lambda;
simulation_parameters.lambda_stderr = lambda;

```

### *Finding spurts and outputting individual curves*

A vector is defined to hold the maxima data found.

```
significant_spurts_data = [];
```

A loop is used to process all cases (subjects). The find\_spurts function finds maxima for the case id provided. Maxima data found for this case is appended to the vector, and the graph is printed to a file in png format.

```

for id=1:simulation_parameters.ncase
    [spurt_data graph_handle] = find_spurts(simulation_parameters, id);
    significant_spurts_data = [significant_spurts_data spurt_data];
    group = simulation_parameters.group_ids(id); % retrieving group for
    this case
    fileName = [resultsDir, 'case_',
    num2str(simulation_parameters.subject_ids(id)), '_' ...
    char(simulation_parameters.group_labels(group)), '.png'];
    print('-f', '-dpng', fileName);
    close
end

```

### *Outputting maxima data and a summary curve*

Finally, after all individual cases have been processed, the text file containing maxima measures is created using the output\_significant\_data function, the summary graph is generated using plot\_spurts, and this graph is printed to file.

```

data_file_name = [resultsDir, 'AMD_spurts.txt'];
output_significant_data(data_file_name, significant_spurts_data, ...
    simulation_parameters.compute_std_errors);
h = plot_spurts(significant_spurts_data, simulation_parameters,
    SORT_CASE_ID);
print('-f', '-dpng', [resultsDir, 'spurts.png']);
close

```

## Conclusion

AMD is an automated tool to find significant maxima in a function or its derivatives. This document has described how to perform an AMD analysis using its new GUI and using scripts. Future improvements of AMD include the ability to detect both minima and maxima, and a more rigorous study of the heuristic function that suggests a value for the smoothing parameter lambda.

## References

- Corrigan, R. (1978). *Language development as related to stage 6 object permanence development*. Journal of Child Language, 5(02), 173–189.
- Dandurand, F., & Shultz, T. R. (2010). *Automatic detection and quantification of growth spurts*. Behavior Research Methods, 42(3), 809–823. doi:10.3758/BRM.42.3.809
- Dandurand, F., & Shultz, T. R. (2011). *A fresh look at vocabulary spurts*. In Proceedings of the Cognitive Science Society meeting.
- Ganger, J. (2004). Data from Ganger & Brent 2004. Retrieved from <http://www.pitt.edu/~jganger/spurtdownloads.html>.
- Ganger, J., & Brent, M. R. (2004). *Reexamining the vocabulary spurt*. Developmental Psychology, 40(4), 621–632.
- McMurray, B. (2007). *Defusing the childhood vocabulary explosion*. Science, 317(5838), 631.
- Mitchell, C. C., & McMurray, B. (2008). *A stochastic model for the vocabulary explosion*. In B. C. Love, K. McRae, & V. M. Sloutsky (Eds.), Proceedings of the 30th Annual Conference of the Cognitive Science Society (pp. 1919–1924). Austin, TX: Cognitive Science Society.
- Ramsay, J., Hooker, G., & Graves, S. (2009). Functional data analysis with R and MATLAB. New-York: Springer.
- Ramsay, J., & Silverman, B. (2005). Functional data analysis (2nd ed.). New York: Springer.
- Tuddenham, R. D., & Snyder, M. M. (1954). Physical growth of California boys and girls from birth to eighteen years. Berkeley: University of California Press.

*Manuscript received 21 February 2011.*

*Manuscript accepted 22 February 2011.*