

PsPM Release Notes

Version 6.1.0

21st August 2023

Contents

1 PsPM Version 3.0	3
2 PsPM Version 3.0.1	4
3 PsPM Version 3.0.2	6
4 PsPM Version 3.1	6
5 PsPM Version 4.0	7
6 PsPM Version 4.0.1	9
7 PsPM Version 4.0.2	9
8 PsPM Version 4.1.0	9
9 PsPM Version 4.1.1	11
10 PsPM Version 4.2.0	12
11 PsPM Version 4.2.1	13
12 PsPM Version 4.3.0	14
13 PsPM Version 5.0.0	15
14 PsPM Version 5.1.0	16
15 PsPM Version 5.1.1	17
16 PsPM Version 6.0.0	18
17 PsPM Version 6.1.0	21

1 PsPM Version 3.0

Import

New data types were implemented

- Noldus Observer compatible
- Eyelink

Untested data types

CNT data import has not been not tested – please contact the developers with sample data files.

Filtering for SCR models

Previous versions of PsPM have used a bi-directional high pass filter of 0.0159 for all SCR analyses. We have recently shown a better predictive validity for GLM with a unidirectional filter of 0.05 Hz [?]. This also implies that different filters are used for different models. These are now set as defaults, and the way the default settings are implemented has changed. It is now possible to alter the filter settings in the model definition, although we discourage this.

New SF method

A matching pursuit algorithm is now implemented to approximate the number of spontaneous fluctuations (SF) in skin conductance [?].

General linear modelling (GLM)

Parametric modulators

Parametric modulators (pmods) are z-normalised before being entered into the design matrix. This is to account for possibly very large or very small numbers – a badly scaled design matrix can cause induced instability in the inversion. The parameter estimates of the pmods were not transformed back in previous versions, i. e. the parameter estimates of the pmods were independent of the scaling of the pmods. This is appropriate as long as they are the same for all datasets, or if analysis is done strictly on a within-subject level. Some researchers have reported designs in which inference was to be drawn on parameter estimates of pmods on a between-group level, and where the pmods systematically differed between these groups. To account for this possibility, the parameter estimates are now transformed back, such that they refer to the pmods in their original scaling/units.

Parametric confounds

Previous versions of PsPM contained an option to include a parametric modulator across all event types, to account for confounds across all conditions. For example, in an experiment with 5 conditions, one could have included 5 regressors, plus one reaction time confound across all events, without including an associated regressor that contains the event onsets for all these events. This option was removed.

Design matrix filtering

Previous versions of PsPM filtered the design matrix after orthogonalisation of basis sets. This can introduce unwanted dependencies between regressors. PsPM 3.0 filters the regressors first, then orthogonalises the basis sets.

File format

Some minor changes have been made to the data format. In particular, marker channels from previous versions can not be read with the current version - such data files have to be re-imported. Model files have changed drastically, and model files from previous versions can not be read with the current version of the software.

VB inversion

The VBA toolbox (<http://mbb-team.github.io/VBA-toolbox/>) was updated in October 2014 [?]. This update incorporates bugfixes in this toolbox and slightly changed the model estimates in our test models. In terms of predictive validity, we noted that there was no consistent benefit of the old or new version of this code (Figure 1.1).

2 PsPM Version 3.0.1

Import**New data types were implemented**

- DATAQ Windaq (no ActiveX needed)
- European Data Format (EDF)

Tools

'Preprocess respiration traces' replaces 'Convert Respiration to Respiration Period'. It supports conversions into the following datatypes:

- Respiration period (old functionality)
- Respiration amplitude

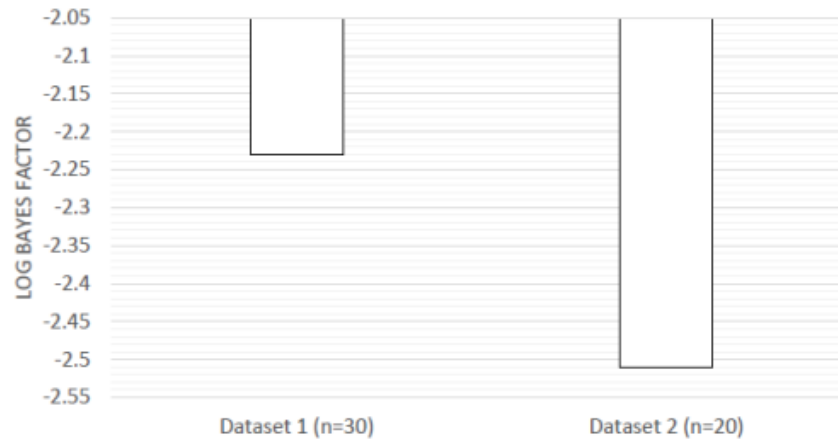


Figure 1.1: Model comparison between old and new versions of the VBA toolbox, based on two delay fear conditioning datasets. The log Bayes factor quantifies the difference between negative log likelihood (nLL) of parameter estimates obtained from model inversion using the old and new version of VBA. A difference in nLL above 3 indicates significant differences in model evidence which is not exceeded for either data set. Analysis and figure contributed by Matthias Staib.

- Respiration line length
- Respiration time stamp

First level contrast

Z-score parameter estimates

The function `pspm_con1` now supports z-scoring trial-by-trial parameter estimates. If selected, all parameter estimates for the same event type, across all trials, will be z-scored before computing the contrast. This option is currently only available for non-linear models.

Review first level model

Next to the regressors on the x- and y-axes, orthogonality plots newly display regressor names along the y-axes.

3 PsPM Version 3.0.2

GLM

Problem with multiple sessions in Design matrix fixed.

4 PsPM Version 3.1

General linear modelling (GLM)

New modalities

For the first time in PsPM, we introduce models for data types other than SCR:

- GLM for evoked HPR
- GLM for fear-conditioned HPR
- GLM for evoked RA
- GLM for fear-conditioned RA
- GLM for evoked RFR
- GLM for evoked RP
- GLM for fear-conditioned PS

Tools & Data preprocessing

Tools are split up into Tools & Data preprocessing. The content of each section is listed below. New functions are written in **green**.

Tools

- Display data
- Rename file
- Split sessions
- Artefact removal
- Downsample data
- Interpolate missing data
- **Extract segments**
- **Segment mean**

Figure 5.1: Model comparison between old and new versions of the VBA toolbox, based on one delay fear conditioning dataset (Dataset 2, see earlier comparison). The log Bayes factor quantifies the difference between negative log likelihood (nLL) of parameter estimates obtained from model inversion using the old and new version of VBA. A difference in nLL above 3 indicates significant differences in model evidence which is not exceeded for the used dataset.

Data preprocessing

- Preprocess heart data
- Preprocess respiration data
- Prepare illuminance GLM
- Find valid fixations
- Convert pupil data
- Find startle sound onsets

Data preparation

Import

Support for Philips Scanphyslog files and for bioread-converted AcqKnowledge files has been added to the import function.

5 PsPM Version 4.0

Change from scr_ to pspm_

The prefix scr_ has been exchanged with pspm_. This applies to all functions contained in PsPM.

Updated toolboxes

Matlabbatch (<https://git.code.sf.net/p/matlabbatch/code-git>) and the VBA toolbox (<https://github.com/MBB-team/VBA-toolbox>) were updated to the latest available versions (08.11.2016). The updated VBA version fully supports newer Matlab versions and warning messages in Matlab 2014 and newer do not appear anymore (during PsPM startup). In terms of predictive validity, we noted that there was no consistent benefit of the old or new version of this code (Figure 5.1).

Subsessions in DCM models

Until now, long periods of NaN values (which are just disregarded) could cause problems for inversion of DCMs. These periods are now split (according to threshold value) into subsessions. The subsessions will be evaluated independently, while at the end the results will be stacked together. Therefore the output-format will not change and should be compatible with earlier releases.

Import

New import function for Labchart files

The new import function allows direct import of raw labchart files without exporting them to a suitable format (as was required to date). The function is only available on Windows systems.

General linear modeling (GLM)

New modalities

- Startle eyeblink for EMG (SEB)

Pupil Models

- Blink channels were removed. Periods during blinks and saccades are set to NaN already in the import function. This applies to all channels of the respective eye. Accordingly, blink validation functionality was removed in the function `pspm_find_valid_fixations`. If needed, blinks and saccades can be imported as custom channels using channel ids 7-10 (two eyes) or 4-5 (one eye), while blinks and left eye come first.
- Importing pupil channels now allows direct specification of a transfer function to convert arbitrary units to mm.
- Importing of eyes which have not been recorded creates channels containing NaN values, so that the same import batches can be used for groups of subjects, even if one eye is missing in some of them.
- Sessions in eyelink files (caused by interruption of recording) will be concatenated according to the timing variable.
- In `pspm_find_valid_fixations`
 - the use of software aspect ratio was replaced by resolution. This allows to correctly map the gaze coordinates to the shapes mapped by the stimulus software.
 - gaze coordinates can be plotted in order to verify the validation settings.
 - interpolation option was removed and should now be used independently.

Minor changes & Bugfixes

- Version-Check bug during startup is now fixed.
- Markerinfos will now according to marker channels be converted to an event-based format.
- The data editor (pspm_data_editor) now allows to specify an output file using the command line.
- pspm_split_sessions additionally allows to specify marker ids at which the file should split.
- Artefact correction was extended with a further function (Simple SCR quality correction).
- Convert pupil data becomes convert data and currently only allows area to diameter conversion (arbitrary units to mm is integrated in the import function)

6 PsPM Version 4.0.1

Bugfixes

- Fix not working options in Matlabbatch module 'Preprocess heart data'

7 PsPM Version 4.0.2

Bugfixes

- Fix error when calling 'Convert data' from GUI
- Fix taking square root if pupil data is recorded in AREA

8 PsPM Version 4.1.0

New Functions

- pspm_convert_pixel2unit: Allows to transfer gaze data from pixel to units. This facilitates the use of pspm_find_valid_fixations which needs data in unit values.
- pspm_compute_visual_angle: Computes visual angle from gaze data.
- pspm_convert_visangle2sps: Convert visual angles from Eyelink to Scanpath speed.
- pspm_bf_spsrf_box: SPSRF basis function with box car.
- pspm_bf_spsrf_gamma: SPSRF basis function with gamma function.

New Features

- `pspm_extract_segments` supports DCM files/structures.
- GLM for SPS.
- `pspm_find_valid_fixations` can now take a bitmap as input.
- A new way to trim data: Start and end times can be chosen according to marker name or value.
- A new function to import SMI iView X EyeTracker files into PsPM.
- A new function to import ViewPoint EyeTracker files into PsPM.

Changes

- `pspm_find_valid_fixations` now computes a circle around the fixation points when run in fixations mode.

Bugfixes

- Heart period response function (`bf_hprf`) coefficients are fixed according to the manual
- `pspm_extract_segments` bugfixes:
 - Fix bugs related to conditions appearing in different sessions.
- `pspm_get_eyelink` now imports markers between two recording sessions (previously these markers were “lost”)
- `pspm_compute_visual_angle`: Fix bug in range computation.
- `pspm_ecg2hb`: Fix out of bounds error occurring for highly noisy data with many outliers.
- `pspm_get_acq`: Fix incorrect linear transformation during ACQ import.
- `pspm_convert_unit`: Fix incorrect transformations between metric units and inches.
- `pspm_resp_pp`: Fix out of bounds error during convolution in cushion mode.
- `pspm_pp` in ‘simple_qa’ mode now uses the default values specified in `pspm_simple_qa`.
- Various further bugfixes.

Improvements

- Blink and saccade channels can be imported with PsPM Eyelink import.
- GLM structure holds the missing data percentage of each condition after segment extraction. Further, the decision to exclude conditions can be made depending on the percentage of missing data.
- `pspm_extract_segments` now works both with GLM files and already loaded structures
- PsPM now checks if SPM is already in MATLAB path. If so, user is asked to remove it from path before initializing PsPM.
- `pspm_load1` now returns statistics about conditions with high NaN ratios.
- `pspm_extract_segments` now is able to use data stored in GLM/DCM structures instead of relying the existence of original data files.
- Multiple new tests to validate the correctness of the functions.

9 PsPM Version 4.1.1

This is a hotfix release that fixes a few issues with 4.1.0 release.

Changes

- `pspm_get_eyelink` now uses the scaling factor from [?] for area based arbitrary units to milimeters conversion.
- `pspm_get_smi` does not perform pixel to milimeters conversion for pupil data anymore. Pupil values are returned as is in pixel units.
- `pspm_get_smi` performs pixel to milimeters conversion for gaze data only if the stimulus resolution in pixels are given as an extra option.

Improvements

- `pspm_get_viewpoint` is now able to import blink and saccade channels from sample files as well. In order to enable this feature, event files must be stored asynchronously in the datafile. See ViewPoint EyeTracker section in this manual for further information.

10 PsPM Version 4.2.0

New Functions

- A new pupil data preprocessing function
- A new pupil foreshortening error correction function
- A new QRS detection algorithm for fMRI environments

New Features

- Previously, Eyelink import used to discard 50 ms worth of samples from each side of every blink or saccade period. This behaviour is preserved; however, sample discard duration can now be set by the user.
- Channel processing functions now store extensive historical data in PsPM files. This data can be printed in table format using the new utility function `pspm_format_history_from_file`
- Pupil channel is now loaded according to a new precedence order. Refer to GLM documentation in this manual to learn more.

Changes

- Eyelink import now returns times and dates in a slightly different format.
- Newest version of PsPM is now obtained from the version string in `pspm.sourceforge.net`, not from the download link zip file name
- GLM now uses the **LAST** channel of a specified modality in a PsPM file.
- Nonlinear SCR model (DCM) now does not use the inter-trial interval duration value (ITI) of the last trial when computing session specific minimum ITI values. Previously, using these last ITI values would lead to abnormally small overall min ITI values in some input files, thereby causing the inference and PCA sections to use less data for all trials. Now, we prevent this from happening.
- Eyelink import parameter blink/saccade edge discard factor default value has been set to 0. This means no extra samples are discarded around blinks/saccades.

Bugfixes

- Fix a bug in `pspm_extract_segments` where trial onsets were not assigned to conditions correctly

- Fix a bug in Viewpoint import where files in DOS format were being parsed incorrectly
- Fix a bug in Eyelink import where blink/saccade periods were misaligned, causing the function to discard useful data and return noise instead

Improvements

- New utility functions to make PsPM more compatible across different operating systems
- An optimized Eyelink import function that is significantly faster and more memory efficient than previous versions
- An optimized SMI event import function that is significantly faster than the previous version
- Multiple new tests to validate the correctness of the functions
- API unification: Now all preprocessing functions use `channel_action` variable to choose what to do with the new channel.

11 PsPM Version 4.2.1

New Functions

- Three new tests (`pspm_hb2hp_test.m`, `pspm_filtfilt_test.m`, `pspm_butter_test.m`)

Changes

- `pspm_display` and `pspm_ecg_editor` do not perform filtering anymore.
- Treatment of missing data in DCM is now the same regardless of whether they are specified as NaN or via `model.missing`
- Eyelink import parameter `blink/saccade edge discard factor` default value has been set to 0. This means no extra samples are discarded around blinks/saccades.

Bugfixes

- Fix a bug in `pspm_hb2hp` where the function crashed when there is no heartbeat left after lower and upper bound filtering of the heartbeat periods
- Fix a bug in `import_eyelink` which imported more markers than the number of markers in the actual `.asc` file
- Fix a bug in `pspm_display` which crashed when trying to plot ECG signal that contains NaN

- Fix a bug in `pspm_prepdata` which returned only NaN when input signal contained some NaN values
- Fix a bug in `pspm_version` which crashed when MATLAB was invoked with `-nojvm`
- Fix a bug in `pspm_get_viewpoint` which returned '+,=,+' lines in the marker list and which crashed when given a viewpoint data file containing multiple sessions separated with '+,=' type of markers
- Fix a bug in `import_viewpoint` which created a new session for every marker containing a '+' somewhere, e.g.: 'CS+'
- Fix a bug in `pspm_get_events` which was not able to locate a marker if it occurred on the first or last sample in a given datafile
- Fix a bug in `pspm_filtfilt` which crashed when the filter parameters were of dimension one

12 PsPM Version 4.3.0

New Features

- In `pspm_get_events`, `import.flank` can be now set to 'all' what would import all markers and data related to them.
- `pspm_display` allows now to display pupil size units and the gaze x & y coordinate units on the y-axis.
- `pspm_extract_segments` can be used now with raw data and thus be called easily within another function.

Changes

- `pspm_version` has a new url and thus do not send any warning about version checks anymore.
- `import_eyelink` do not import markers which are outside the session end time (END marker).
- `import_eyelink` sends a warning whenever two markers have the same timestamp.
- In `pspm_get_eyelink`, `import.flank` set to 'all'.
- The test of `pspm_extract_segments` was adapted to the new feature.
- External functions and libraries were regrouped in one folder called `ext`.

Bugfixes

- Fix a bug in `pspm_bf_lcrf_gm` and `pspm_bf_ldrf_gm` where the offset was wrongly implemented.
- Fix a bug in `pspm_compute_visual_angle` where there was an error in the conversion factor of pixels wrt. to mm.

13 PsPM Version 5.0.0

New Features

- Allow `pspm_data_editor` to load an epoch file.
- Allow `pspm_simple_qa` to suppress classification of discretisation oscillations as artefacts, to expand artefact windows, and to automatically remove small data islands embedded in artefacts.
- Allow `pspm_simple_qa` to store the epochs of data that are filtered out into an output `.mat` file. The accompanying GUI editor is available under 'Artefact removal' in the tools section.
- Allow `pspm_convert_gaze_distance` to convert from distance units to degrees or scanpath speed. The accompanying GUI editor is available under 'Gaze Convert' in the data preprocessing section.
- Add the possibility to select the flank in the Import module of the GUI of PsPM.
- Add the possibility to import DSV (delimiter separated values) as well as CSV (comma separated values) data files.

Changes

- Split the data convert functionality in tools into the 'Gaze Convert' and 'Pupil Size convert' in the data preprocessing section.
- Factor out blink/saccade edge filtering logic out of `pspm_get_eyelink` to `pspm_blink_saccade_filt`.
- Deprecate edge filtering functionality in `pspm_get_eyelink`.
- Make `pspm_pupil_correct_eyelink` use the last gaze channel when there are multiple gaze x (similarly y) channels in the file.
- Make `pspm_extract_segments` return NaN percentages and not ratios.

Bugfixes

- Scale DCM plot XTick by sample rate.
- Correct index offset when dealing with the descending flank for continuous markers.
- Allow `pspm_display` to plot any type of marker channels.
- Fix `pspm_display` behaviour when user tries to load data with less channels than the data he/she is currently displaying.
- Fix `pspm_split_sessions` behaviour when the intertrial interval in the data is random. Add an option `randomITI` (0 or 1) which in the latter case it forces the function to evaluate the mean marker distance using all the markers and not per session. Default value: 0.
- Remove `startsWith` and `endsWith` from all functions for a better backward compatibility.
- Fix bug in `pspm_trim` which was wrongly defining the starting trimming point index.

14 PsPM Version 5.1.0

New Features

- PsPM now has an improved GUI with a more eye-friendly colour and typeface.
- PsPM now has an alternative GUI built by using `.mlapp`, which shows consistent a more modern appearance across different platforms.

New Functions

- `pspm_scr_pp`
 - `pspm_scr_pp` replaces the classic `pspm_simple_qa`.
 - `pspm_scr_pp` now allows users to detect clipping, where the results can be exported together with the original outcome.
 - `pspm_scr_pp` now allows users to just write artefact epochs and leave data unchanged.

Changes

- General
 - Dialogs have been rewritten for avoiding endless loops.

- *Pre-processing* menu has been reconstructed, where *pupil size convert* and *gaze convert* are now under *pupil and eye tracking*.
- PsPM now always generates test data inside the PsPM folder.
- `pspm_interpolate`
 - `pspm_interpolate` now process data ending with NaN well without throwing warnings.
- `pspm_find_sounds`
 - `pspm_find_sounds` now throws warning when some data was excluded due to the strict parameter setting.
- `pspm_split_sessions`
 - `pspm_split_sessions` is now using `pspm_trim` for processing data.

Bugfixes

- `pspm_glm`
 - A bug that leads to the failure of selecting left/right eye has been fixed.
- `pspm_load1` and `pspm_dcm`
 - A bug that leads to the failure of exporting statistics has been fixed.
- `pspm_resp_pp`
 - A bug that leads to the failure of opening batch for respiration data processing has been fixed.
- `pspm_scr_pp`
 - A bug that leads to incorrect matrix construction has been fixed.

15 PsPM Version 5.1.1

New Functions

- `pspm_bf_data`
 - New function template to allow users creating their own basis functions (bf) from a data vector.

UI Improvements

- PsPM now has an improved GUI for the main screen and `pspm_display`.
- PsPM now uses system specific design for Windows, macOS and Linux.
- A bug that affects GUI display has been fixed.

Bugfixes

- `pspm_dcm`
 - A bug of channel and data referring that leads to incorrect missing epoch processing has been fixed.
 - `pspm_dcm` now handles empty epochs correctly.
- `pspm_scr_pp`
 - A bug that throws warning when using GUI has been fixed.
- `pspm_split_sessions`
 - A bug which leads to GUI crash has been fixed.
 - `pspm_split_sessions` now accepts correct number of data files and missing epoch files in the GUI.
 - `pspm_split_sessions` now processes missing epochs correctly.

16 PsPM Version 6.0.0

This is a major release and may have incompatibility with previous versions.

New features

- PsPM now support a develop mode, which allows
 - Minimised terminal output
 - Automagical parameter selection for testing
 - Always overwrite files.
- `pspm_extract_segments`
 - now allows an optional z-score for normalisation.
- `pspm_find_valid_fixations`
 - now allows processing preprocessed left/right/combined pupil data.
- `pspm_gaze_pp`

- now allows preprocessing gaze signals.
- `pspm_overwrite`
 - now handles the behaviour of overwrite if not defined by the user, and is used where applicable.
- `pspm_time2index`
 - now processes adjustable conversion from time to index globally.
- `pspm_update_chantype`
 - now allows generalised behaviour of updating channel types.

Bug Fixes

- General
 - A bug which may lead to failure of pspm path searching has been fixed.
 - A bug which may display figures in the GUI has been fixed.
- `pspm_bf_sprsf_box`
 - now uses correct parameter settings.
- `pspm_convert_ppu2hb`
 - terminates if only one pulse is detected.
- `pspm_data_editor`
 - now able to show figures when import a datafile correctly.
- `pspm_dcm`
 - now handles missing epochs that start at 0 correctly.
 - now processes variables properly to make length consistent.
- `pspm_display`
 - now displays plots with correct x-axis ranges.
- `pspm_glm`
 - now applies correct variable settings.
 - now produces results correctly when facing many short missing epochs.
- `pspm_interpolate`
 - now returns results correctly when applying forced extrapolation.
- `pspm_resp_pp`

- now allows replace as the channel action properly.
- `pspm_scr_pp`
 - now uses inconsistent variables when delivering missing epochs.
 - now writes data out properly when epochs are removed.
 - now properly handles data if the first channel is not `scr`.
- `pspm_sf`
 - now refer to model file correctly.
- `pspm_split_sessions`
 - now processes missing files with an epoch starting at 0 s properly.
 - now properly catches edge cases.
- `ValidSamplesModel`
 - now properly throws warnings if histogram counts are zeros.

Improvements

- General
 - now supports loading ppg data.
 - now displays improved terminal output text.
- `pspm_con1`
 - now assesses the statistics in the arguments for throwing warnings when detecting invalid values.
- `pspm_data_editor`
 - now displays the x and y axis label and text according to the input data.
 - now sorts the epoch list according to the start data point.
- `pspm_get_marker`
 - now detects and update the field `flank` when applicable.
- `pspm_import`
 - now detects and update the field `flank` when applicable.
- `pspm_load_data`
 - now checks whether the input has an empty channel.
 - now autofills some variables.

- `pspm_pupil_pp`
 - now calls `pspm_load_data` to load single channels.
- `pspm_rev_glm`
 - now displays normalised data for visualisation purpose only.
- `pspm_write_channel`
 - now checks whether the input has an empty channel.

GUI

- "Edit defaults" in the GUI is now working properly.
- GUI typeface is now generalised across the software.

For Developers

- General
 - PsPM now uses `.editorconfig` to control the indent.
 - PsPM now uses UTF-8, LF, and MATLAB's soft tab indent as the standard code formatting style.
 - PsPM now uses test data from the repository PsPM-data.
- `pspm_get_text_test`
 - A bug, which leaves residual test files after code testing, has been fixed.

17 PsPM Version 6.1.0

- General
 - Minimum version of MATLAB
 - * The minimum version of MATLAB required for running PsPM is MATLAB 7.1 or R14 Service Pack 3.
 - Entering PsPM
 - * Entering PsPM is now hybrid based on the version of MATLAB, namely through AppDesigner or Guide. PsPM will be entered through AppDesigner when supported and through Guide for older versions of MATLAB. To call PsPM, users should still use the command `pspm` in the Command Window of MATLAB, where the appropriate way of entering PsPM will be recognised automatically.

- * Users can still use their preferred way to enter PsPM. To enter PsPM through AppDesigner, please type `pspm_appdesigner`. To enter PsPM through Guide, which is the classic entrance, please type `pspm_guide`.
- * The AppDesigner version of PsPM UI is newly introduced by MATLAB and the encouraged way for launching PsPM, and it has no functionality difference than the Guide version of PsPM UI.
- * Both AppDesigner and Guide are designed and tested for Windows, macOS (either Intel or Apple Silicon) and Linux. Issues of opening PsPM through either of the two UI systems are encouraged to be reported to PsPM developing group at GitHub.
- Introducing `pspm_options`
 - * A new function `pspm_options` is introduced to PsPM for controlling the default and acceptable values of the struct `options` used by most PsPM functions. The default values of the fields in the struct `options` for various functions can be directly checked by searching in `pspm_options`.
 - * The default values in `pspm_options` have been checked and tested for PsPM. If preferred values are different from defaults, users can specify them when calling the corresponding PsPM functions, and the corresponding functions and `pspm_options` will always respect the users' specification with the highest priority. However, the user's specification needs to satisfy the condition set in `pspm_options`, and invalid values may be reported as errors.
- UI Improvements
 - * The UI for the following features has been improved and tested
 - Launchpad (AppDesigner)
 - Launchpad (Guide)
 - Batch editor
 - Display data
 - Contrast manager
- Help text
 - * The help text has been updated for all PsPM functions. Help text can be checked by right clicking the functions. An example of checking the help text of a PsPM function is shown below.
- Bug Fixes
 - UI
 - * A bug that used to make PsPM crash when *Pupil size convert* and *Gaze convert* in *Data Preprocessing* were selected has been fixed.
 - * A bug that leads to error when PsPM is started offline has been fixed.

- * A bug that leads to incorrect x labels in `pspm_rev_dcm` has been fixed.
- `pspm_convert_...`
 - * A bug that leads PsPM to crash, which is caused by outdated UI calling for `pspm_convert_...` functions, has been fixed.
- `pspm_get_eyelink`
 - * A bug that causes issues when importing eye link data that is scanned at both left and right sides has been fixed.
- `pspm_glm`
 - * A bug that incorrectly identifies the NaNs in the input data has been fixed.
- `pspm_sf`
 - * A bug that has lead to incorrect input datafile assigning has been fixed.
 - * A bug that leads PsPM crash when `pspm_sf` is analysing data with missing epochs has been fixed.
 - * A bug that leads to error when `pspm_sf` analyses data where time unit is defined as marker has been fixed.
- `pspm_split_sessions`
 - * `pspm_split_sessions` now considers `marker_chan_num` when calling `pspm_trim`.
- Improvements
 - `channel`
 - * We have generalised `channel` related variables throughout PsPM, which are given as
 - `chan/channel` to `channel`
 - `chans/channels` to `channels`
 - `channel_units/channels_units/chan_units/chans_units` to `channel_units`
 - `chan_combine` to `channel_combine`
 - `chan_action` to `channel_action`
 - `channels_header` to `channel_header`
 - `chantype` to `channeltype`
 - `import_eyelink`
 - * Improved `import_eyelink` for adding some support for importing eyelink data converted by higher version of .EDF files.
 - `pspm_con2`
 - * `pspm_con2` is now using `pspm_overwrite`.

- `pspm_dcm` and `pspm_dcm_inv`
 - * `.flexevents` and `.fixevents` are now fields of `model` instead of `options`.
 - * `pspm_dcm` now uses `pspm_get_timing` to handle missing epochs.
 - * Improved missing epoch support
 - The field `.missing` is now allocated from `options` to `model`.
 - `.missing_data` is used to load missing epoch data that was loaded from `dcm`, as an optional field.
 - * The index is changed so that the first event will not be excluded when at time 0 in session.
- `pspm_extract_segments`
 - * The first channel is always referred by `marker_chan` as default.
- `pspm_glm`
 - * The fallback for `options.exclude_missing` has been set for `pspm_glm`, which is not to exclude any missing epochs.
 - * A bug that leads to incorrect missing epoch checking has been fixed.
 - * Help text was updated.
 - * `marker_chan_num` now refers to the first marker channel as default.
 - * `pspm_glm` now uses `pspm_get_timing` to handle missing epochs.
- `pspm_interp1`
 - * `pspm_interp1` now considers the data where no valid values are detected and interpolation is not possible, and warnings will be reported in this case.
- `pspm_merge`
 - * The default value is set to be `[1,2]`, meaning the first channel will be merged as the default option.
- `pspm_prepdata`
 - * `pspm_prepdata` now uses interpolation to handle data with NaN.
 - Data that begins and/or ends with NaN will be filled with the first/last non-NaN values in those positions.
- `pspm_sf`
 - * `pspm_sf` now supports missing epochs.
 - Now allow input with NaN that indicates missing epochs.
 - Missing epochs can be specified with `options.missing` or the automatic NaN detection.
 - Missing epochs will be verified with `options.missingthresh`.
 - When the missing epochs are longer than 2s, the result out will be converted to `[]`.
 - The UI of SF analysis now allows using missing epochs from files or not to define any missing epochs.

- * `pspm_sf` now uses `pspm_get_timing` to handle missing epochs.
- * `marker_chan_num` now refers to the first marker channel as default.
- `pspm_sf_dcm`
 - * `pspm_sf_dcm` now uses `pspm_interp1` for interpolating data.
- `pspm_text`
 - * The Matlab file that stores the information of help text `pspm_text.mat` will be stored inside the source folder of PsPM and will be deleted when PsPM is quit.
- `pspm_trim`
 - * `marker_chan_num` now refers to the first marker channel as default.
- Minor Adjustments
 - `pspm_sf`, `pspm_glm` and `pspm_dcm`.
 - * The option `marker_chan_num_event` is removed.
 - * In default, the first marker channel / event channel is always selected and no users' customisation is allowed.
 - * The last data channel / wave channel is always selected.
- Reference document
 - New document *PsPM Reference* for checking the default values and restrictions of options fields.

References

- [1] Bach, D. R, Friston, K. J, & Dolan, R. J. (2013) An improved algorithm for model-based analysis of evoked skin conductance responses. *Biol. Psychol.* **94**, 490–497.
- [2] Bach, D. R & Staib, M. (2015) A matching pursuit algorithm for inferring tonic sympathetic arousal from spontaneous skin conductance fluctuations. *Psychophysiology* **52**, 1106–1112.
- [3] Daunizeau, J, Adam, V, & Rigoux, L. (2014) VBA: a probabilistic treatment of nonlinear models for neurobiological and behavioural data. *PLoS Comput. Biol.* **10**, e1003441.
- [4] Hayes, T. R & Petrov, A. A. (2015) Mapping and correcting the influence of gaze position on pupil size measurements. *Behav. Res. Methods* **48**, 510–527.