## Instructions for use of spectral software package SpecOptim

Matt Aitkenhead 4th November 2019, James Hutton Institute

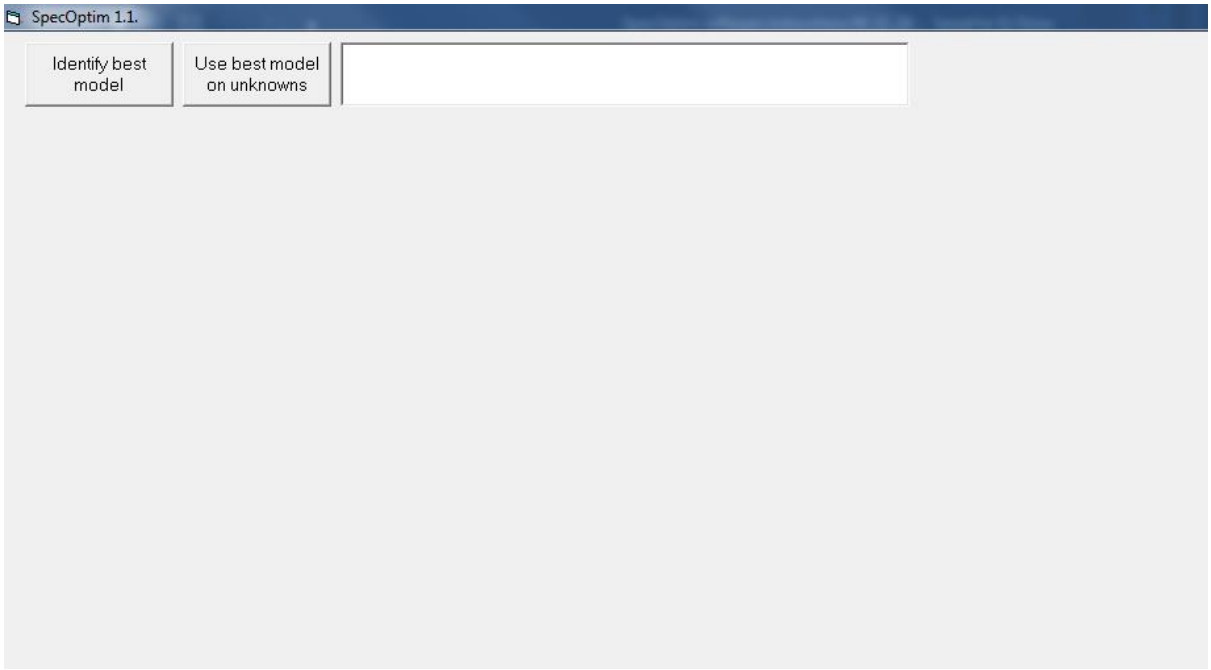
### Introduction

The SpecOptim software package is a small executable developed through the NERC-funded RESIST project. One of the aims of the project is to evaluate the effectiveness of different methodologies in capturing and interpreting spectral information from soils, either in the lab (dried and milled) or the field *in situ*. The SpecOptim package allows the user to explore the effects of different preprocessing and calibration model approaches on spectral datasets.

This enables identification of the optimal processing approach for specific spectral datasets, regardless of the material being analysed, the instrument being used, or the wavelength range being explored. Identification of the 'best approach' for spectral datasets has resulted in a significant body of literature, with different methods being found to work best under different conditions.

The software works well but can be 'broken' if the user does something unexpected to the processing chain. I have not incorporated sophisticated error messaging, so in the case of a fault please get in touch with me at matt.aitkenhead@hutton.ac.uk. Future updates of the software will make it more robust and user-friendly, and less easy to crash through unexpected user actions, so feedback is appreciated.

### What does each button do?




- Identify best model: this identifies the processing chain that provides the most accurate spectral calibration model, from a large number of options.
- Use best model on unknowns: the 'optimal' model can be used here to make estimates of properties from new samples, using spectra captured in the same way as those used to develop the model.

## Identifying the best model

This component of the software will identify the calibration with the best accuracy, from a number of preprocessing and calibration model options. It can take several hours to run, and while running will go 'blank'. So (frustratingly, I know!) it will be impossible to determine how long each run will take. Because of this, I recommend leaving this step to run overnight.

The preprocessing options explored include the following:

- Spectral reduction (taking the mean value of each ten values), which reduces the amount of information in the spectra but also makes model development faster. The options with this 'on' and 'off' are explored.
- Moving window subtraction, in which a moving window with diameter 10% of the number of spectral values is passed across the spectra. The average value of the moving window at each point is subtracted from the actual values. This option is explored 'on' and 'off'.
- Normalisation: the options explored here are Euclidean Normalisation, Standard Normal Variate, Multiplicative Scatter Correction and 'no normalisation'.
- Savitsky-Golay first-order smoothing.
- Zero derivative (i.e. no change), first-order derivative and second-order derivative.

These preprocessing options are carried out in series, and all possible combinations are explored giving a total of 96 combinations of preprocessing options.

The model calibration options explored are backpropagation neural networks (BPNN) and Partial Least Squares (PLS). The additional option of Multiple Linear Regression was also included but has been disabled as I am not confident that I have implemented it correctly. Additional options will be added in later versions of the software, depending on feedback and user interest (and my ability to implement the mathematics).

This approach can also be used to incorporate additional information, such as environmental characterisation of sample sites. I have found that incorporating this information increases the accuracy of the calibration model, sometimes by a large amount. The environmental information must be provided as a set of numerical parameters, either continuous variables or as categorical class memberships. Examples I have used include topography (e.g. elevation, slope, flow accumulation), climate (e.g. monthly mean temperature and rainfall), soil type (by class) and land cover (by class).

To run this step, the 'training' data must be formatted correctly. In a comma-delimited CSV file, there should be a header with the following values:

- The number of spectra
- The number of values in each spectrum
- A flag (0 or 1) indicating whether environmental or other information is given in addition to the spectral information.
- A value indicating the number of parameters of environmental information (if none, a zero is required here).
- The number of output values which the model will have (i.e. the number of parameters that you want the model to be able to estimate).

Following this, the spectral and environmental data must be given as rows of numbers (with no additional text), and with the spectral information first followed by the environmental descriptors. Values do not have to be normalised to fit within any particular range, or log/power normalised (unless you wish to do so). Negative values are also allowed. The file must be stored in an otherwise empty directory – this will avoid accidental overwriting of existing files, as the software does not check to see if a file already exists before creating one with the same name. Folder names can be used to identify which project or dataset you are working with.

Several files are produced by this step. Each is named 'X…..txt' (where X is the name of your original CSV file) with the ellipses having a different text designation according to the function of the file:

- Xbest_model.txt: this is the file where the optimal model is stored upon successful completion of the process. It is structured in a manner enabling the model to be read back in and used, so editing the file is not recommended.
- Xbest_results.csv: each model is developed using the 10-fold cross-validation approach, with a final subsample (one-third of the data) being withheld for testing. This test dataset contains columns of values, with the 'actual' and 'modelled' values of each output parameter being given in successive columns. This enables the user to verify the estimate of model accuracy and carry out further statistical evaluation of their own, if desired. After the columns of values is a brief list of text giving information about the optimal preprocessing model that was identified.
- Xcombination_datasets.txt: part of the process involves developing a separate training/testing/validation dataset for each processing option. These are generated at the start of the process and saved in this large file. Once the model is complete, feel free to delete this file as it can take up significant space on your hard drive.
- Xcombination_results.csv: this file contains information about each preprocessing / modelling combination and the calculated accuracy of the model generated. There are six columns of digits describing each model, followed by statistical evaluation values. The first of these is the mean RSQ of each model, taken across all of the output parameters. This is used as the evaluation of the model accuracy, and so the model which gives the highest value in this column is the one identified as the 'best'. There is some debate about whether this 'mean RSQ' approach is the statistically best and most appropriate way to evaluate models. The second value is the RMSE value taken across all outputs, with each output normalised within the range [0, 1]. I have found that there is usually a strong (negative) correlation between the mean RSQ values and the RMSE values, across the 196 model evaluated for each run.

### Use the best model on new spectra

Assuming that you have new spectra from the same instrument and sample type (e.g. soil), you can use the optimal model to make an estimate of the same properties used in the model development. To do this, create a new CSV file 'test1.CSV' with the same header style and formatting as the training file used to create the model. NOTE the header of this file will have only four values instead of five, as you do not need to provide the number of output parameters to be estimated. The spectral data should be provided in the same style and with the same preparation as the data used to create the model.

The user will be prompted to identify firstly the file containing the optimised model, and then the file containing this new data. If everything works correctly, the model will run relatively quickly (a few seconds) and will produce a new file ‘test1outs.csv’ which contains several columns of values. Each column corresponds to one of the model outputs, and there will be the same number of rows as in the test dataset.