

Machine Learning Based Model Selection with pyDarwin in Pirana

Mark Sale, Keith Nieforth, James Craig
Certara, USA Inc.



Machine Learning Based Model Selection with *pyDarwin* in *Pirana*: Automated NONMEM PK/PD Model Selection



Background & Objective

pyDarwin[1] is a recently released open-source Python package for automated NONMEM Pharmacokinetic/Pharmacodynamic (PK/PD) model selection developed under a grant from, and in collaboration with FDA[2]. It offers a general solution for searching a wide range of PK and PD models within a user defined model search space and permits customization of model fitness criteria to allow prioritization of desired model performance characteristics in the model selection process.

Pirana is a PMX analysis workbench that integrates separate software tools under a common user interface, lending organization and efficiency to the PMX workflow while preserving the flexible nature of the underlying toolsets[3]. Originally developed for use with NONMEM, Perl speaks NONMEM (PsN), and R/xpose package, Pirana has been enhanced to support command line modeling in R with Certara's NLME-Engine (RsNLME) [4], and now supports machine learning based model selection through a graphical user interface to pyDarwin.

Methods

A basic introduction to machine learning model selection will be provided covering general theory and an overview of available search algorithms in pyDarwin. This will be followed by a demonstration of the Pirana software covering the following topics:

- Overview of Pirana user interface demonstrating how to create, execute, post-process and generate report objects for a single model run
- Comparison of machine learning algorithms for model selection, which include: Genetic Algorithm, Bayesian Optimization, Random Forest, Gradient Boosted Random Tree, Particle Swarm Optimization and full Exhaustive Search
- Setup of the model search space, including specification of structural and omega model parameters, and covariate-parameter relationships to include in the search
- Creation and execution of the automated model selection with the pyDarwin interface in Pirana, including methods to visualize model fitness across iterations of the search
- Exploration of model diagnostics using the final best model selected from pyDarwin's automated model selection

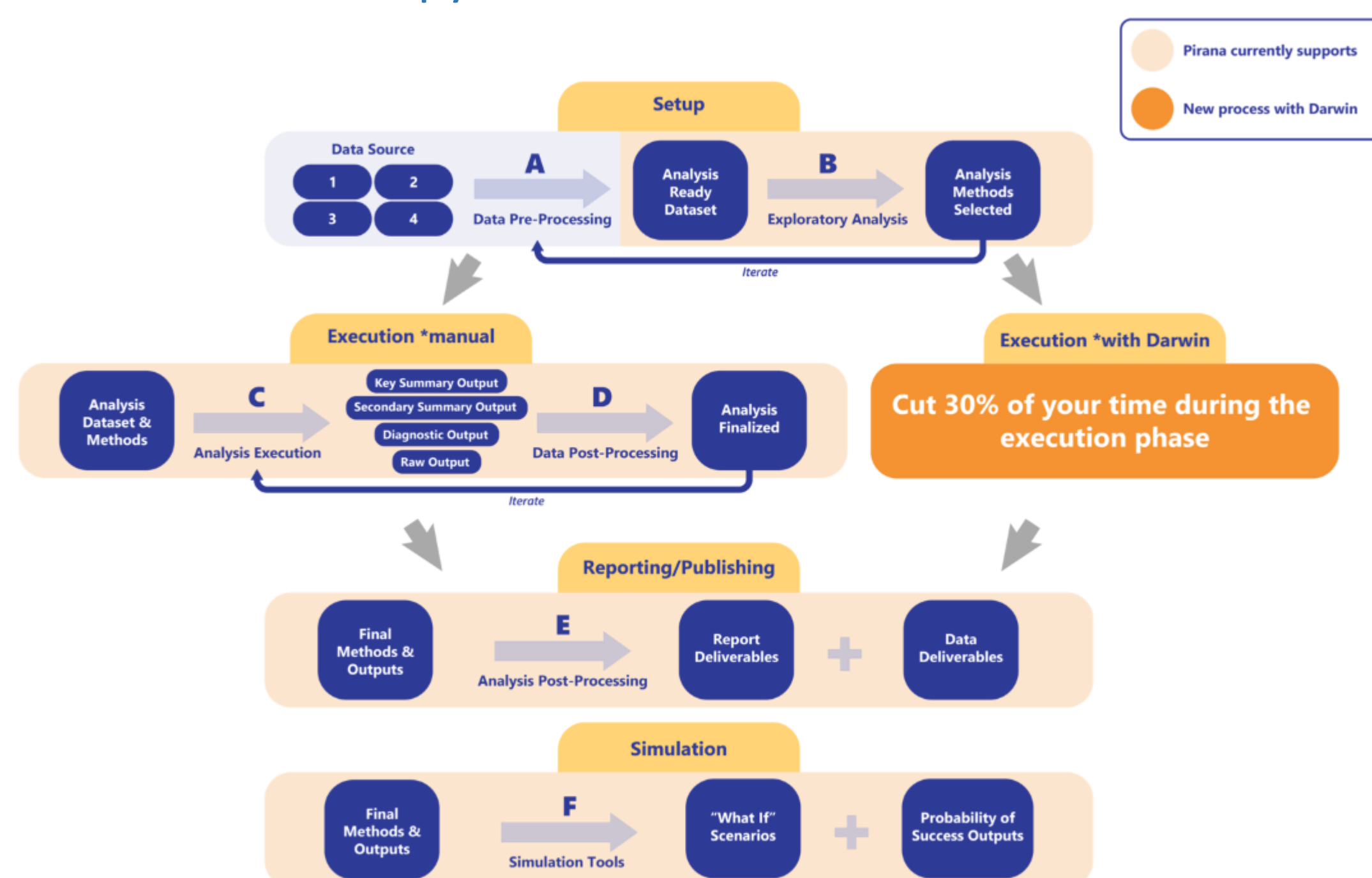
Results

The demonstration provides a comprehensive overview of the theory and execution of a machine learning based PMX model search.

The python command line interface to pyDarwin is logical and well organized but complex, requiring the editing of three text files; one to define the basic NONMEM control file structure, one to define the search space and one to define the search options and parameters. In addition, some knowledge of the underlying Python language.

The availability of pyDarwin in the Pirana workbench provides a robust and efficient method to perform PK/PD analysis within a graphical user interface, integrating with existing Pirana features to view estimation results and generate diagnostics across both the individual candidate models and the final best model selected from the automated search.

Pirana and pyDarwin PMX Workflow



Discussion

Machine learning based pharmacometric (PMX) model development may offer advantages over manual approaches by providing an efficient, objective and more robust method for model selection.

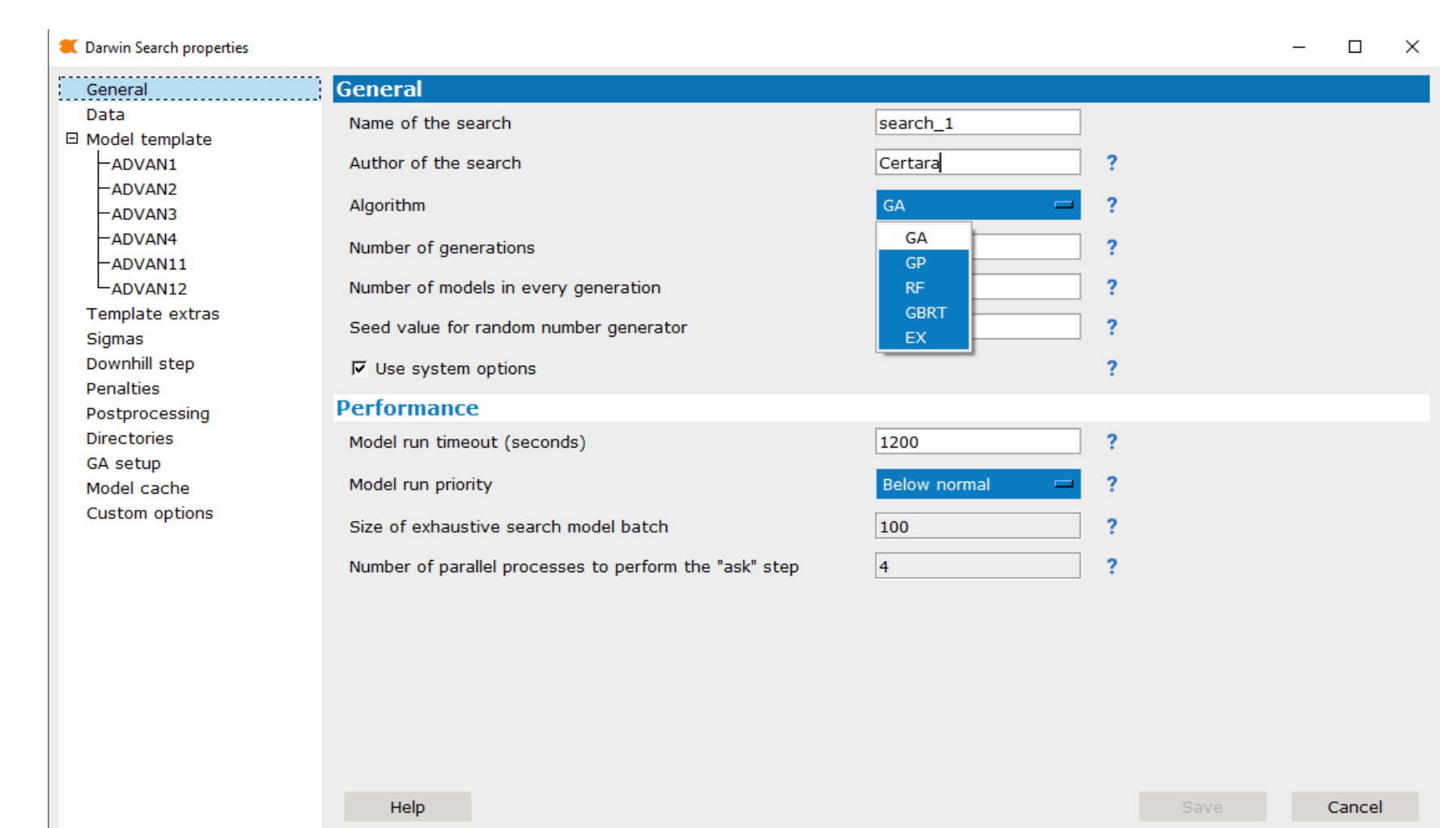
The graphical user interface in Pirana greatly simplifies execution of pyDarwin model searches and enables the usability of machine learning based model selection for PMX workflows[3].

References

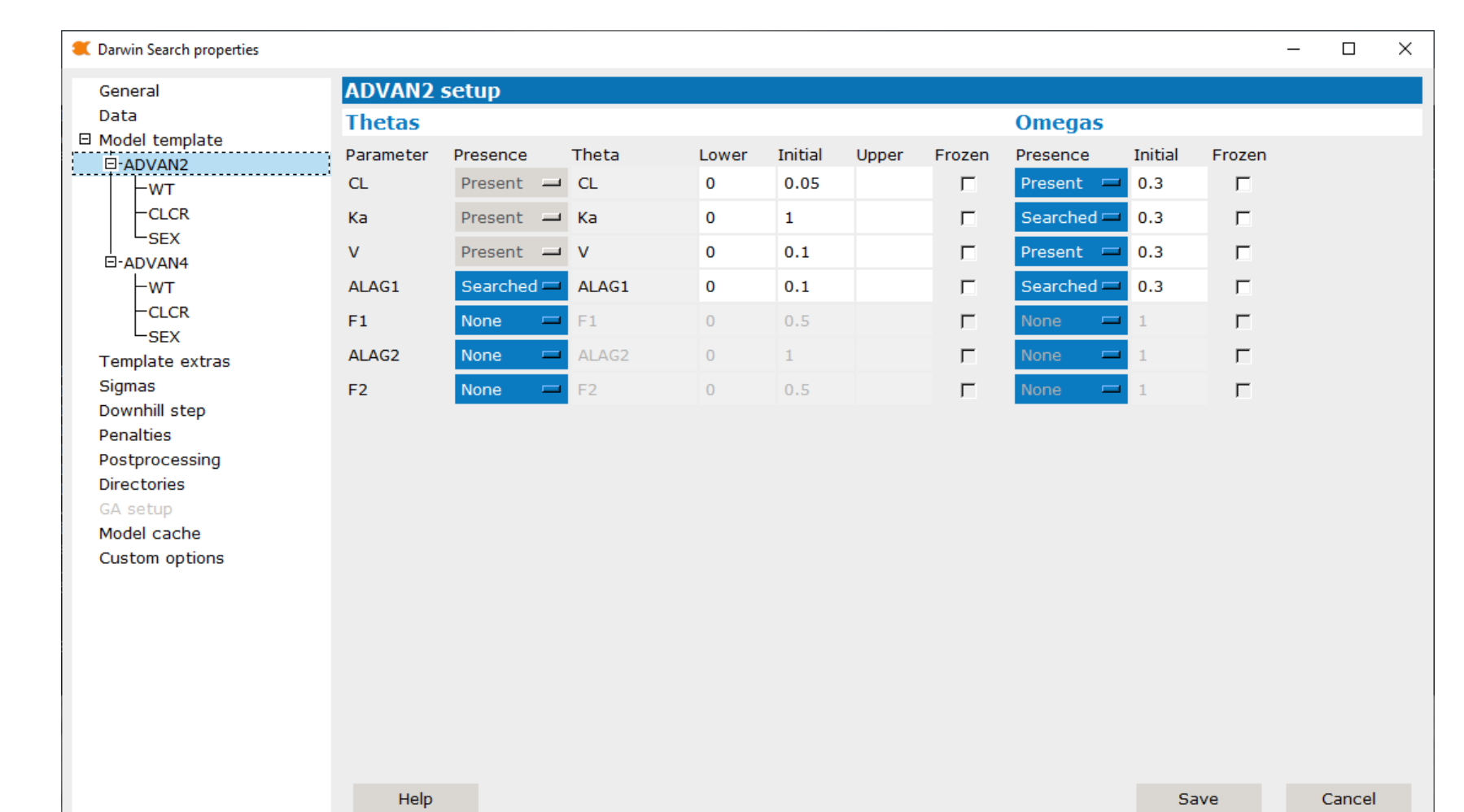
- [1] <https://certara.github.io/pyDarwin/html/index.html>
- [2] <https://grants.nih.gov/grants/guide/rfa-files/rfa-fd-21-027.html>
- [3] Keizer RJ, Karlsson MO and Hooker A. Modeling and Simulation Workbench for NONMEM: Tutorial on Pirana, PsN, and Xpose. CPT: Pharmacometrics & Systems Pharmacology (2013) 2, e50; doi:10.1038/psp.2013.24.
- [4] <https://certara.github.io/R-Certara/>

Additional Figures

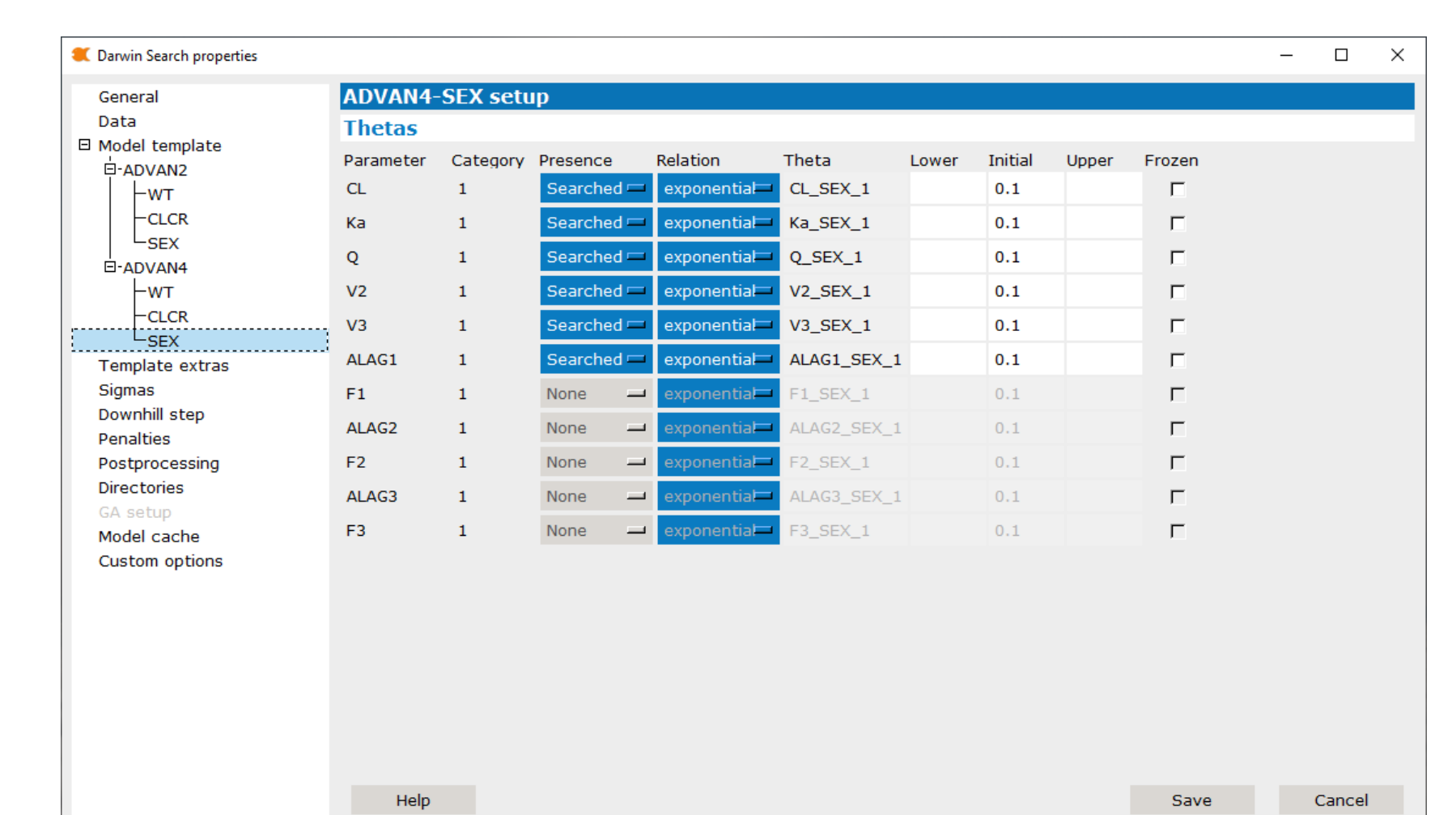
General Setup and Algorithm Selection



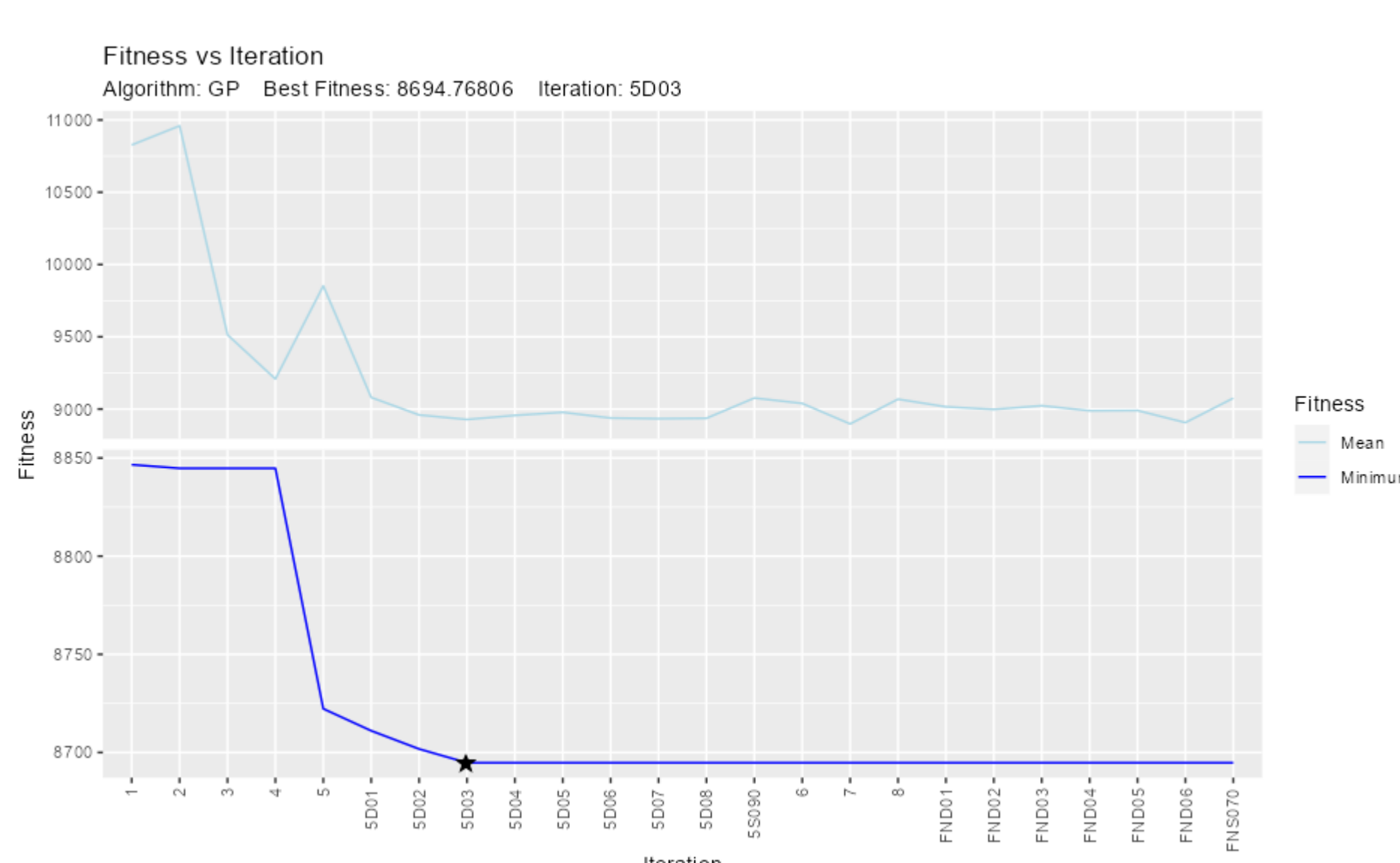
Structural and Omega Parameters Search



Covariate Parameters Search



Fitness vs Iteration Plot



Want to learn more?
<< Scan Here