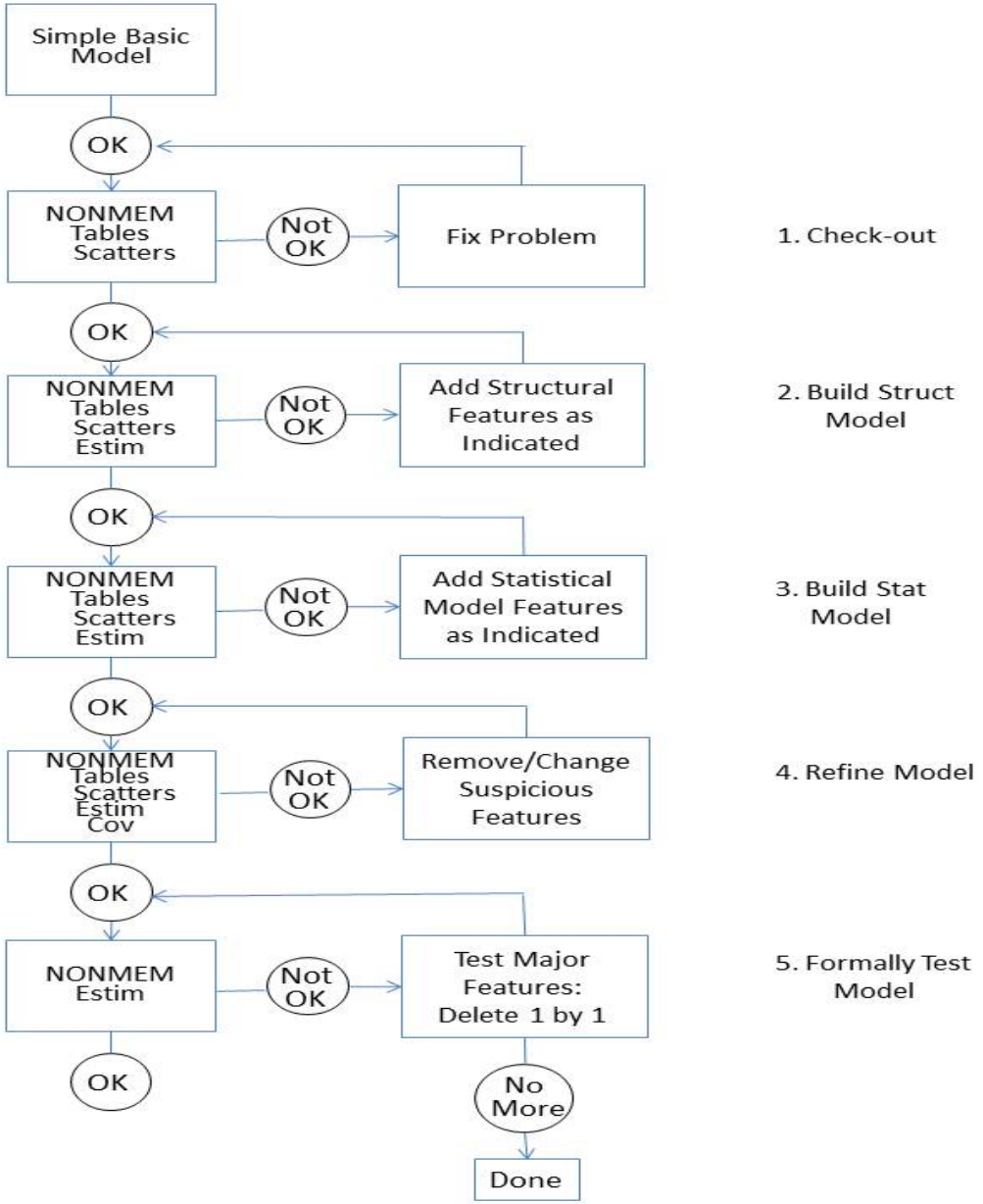
A hybrid genetic algorithm for NONMEM structural model optimization

Mohamed Ismail, Robert R. Bies, Mark Sale
University at Buffalo
Nuventra



Br J Clin Pharmacol 2013

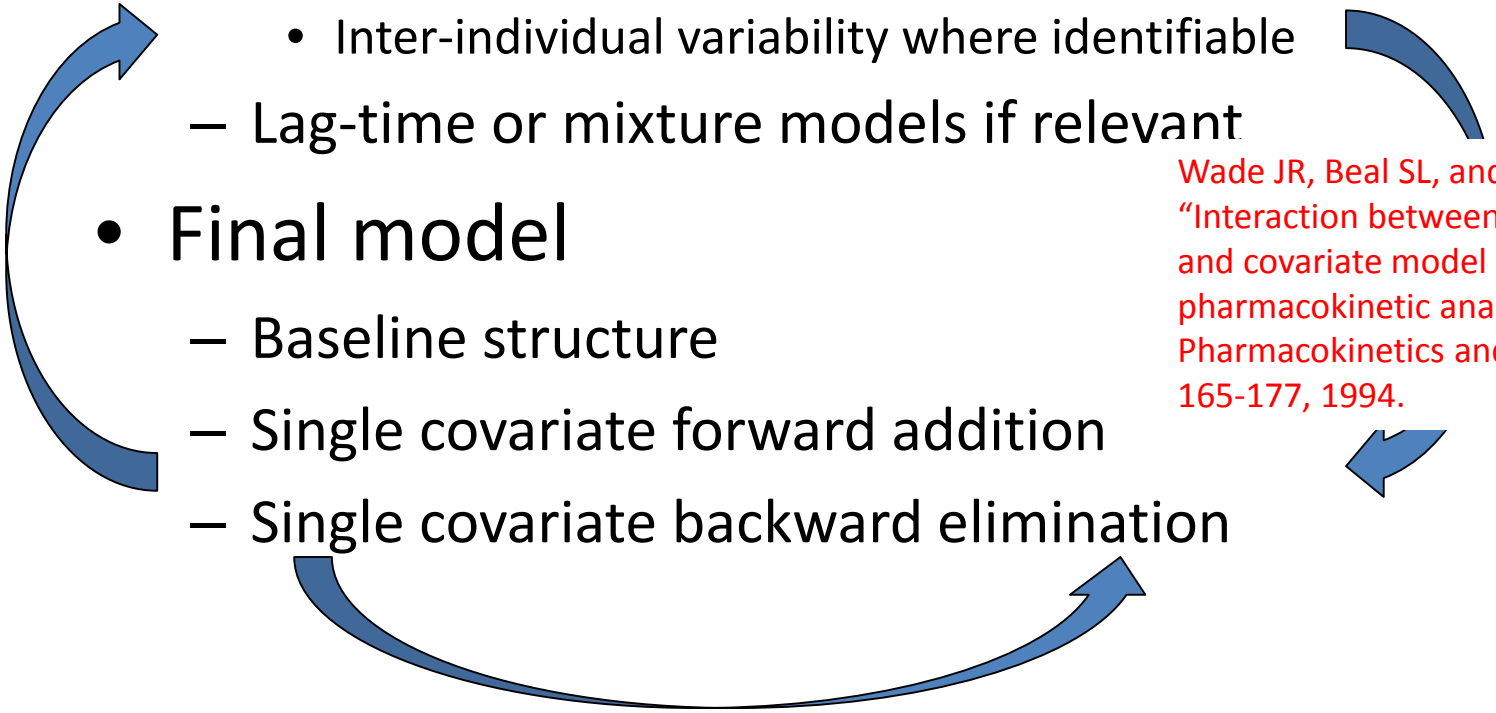
Figure 1. Diagram of model building algorithm from Volume 5 NONMEM manuals. Reproduced with permission from Icon PLC. In the original description of the algorithm, statistical features (variance terms) were added after the structure was final for practical reasons.

Local search: "step-wise" regression

- Base (covariate free) model
  - Keep known physiology in mind
  - Compare compartment structures
    - Residual error structure to minimize systematic errors
    - Inter-individual variability where identifiable
  - Lag-time or mixture models if relevant
- Final model
  - Baseline structure
  - Single covariate forward addition
  - Single covariate backward elimination

Local search: "step-wise" regression

- Base (covariate free) model
  - Keep known physiology in mind
  - Compare compartment structures
    - Residual error structure to minimize systematic errors
    - Inter-individual variability where identifiable
  - Lag-time or mixture models if relevant
- Final model
  - Baseline structure
  - Single covariate forward addition
  - Single covariate backward elimination



Wade JR, Beal SL, and Sambol NC. "Interaction between structural, statistical, and covariate model in population pharmacokinetic analysis", J of Pharmacokinetics and Biopharmaceutics, 22: 165-177, 1994.

Genetic Algorithms

- What are they?
  - A means of evaluating factors in a model where more than one factor can be changed at a single step.
  - Partially automated to allow a more “complete” evaluation of the full grid search space for a particular candidate model.

Genetic Algorithms

- Approach:
  - Replicate “survival of the fittest”
  - Evolutionary process is imposed on the selection and “survival” of the “best” model descriptions
  - Calculate an indicator of how “healthy” a particular individual model in the population is
  - Utilized in multiple fields e.g. placing cell phone towers, predicting stock performance etc.

Genetic Algorithms

- “good” characteristics become more likely
- Efficient at finding “good” regions of solution space
- Slow to converge local “best”
- Adaptations
  - Elitism
    - Retain best candidate to next generation
  - Local search hybrid
    - Compare candidate with each model differing by 1 bit
    - Every 5 generations

Genetic Algorithms

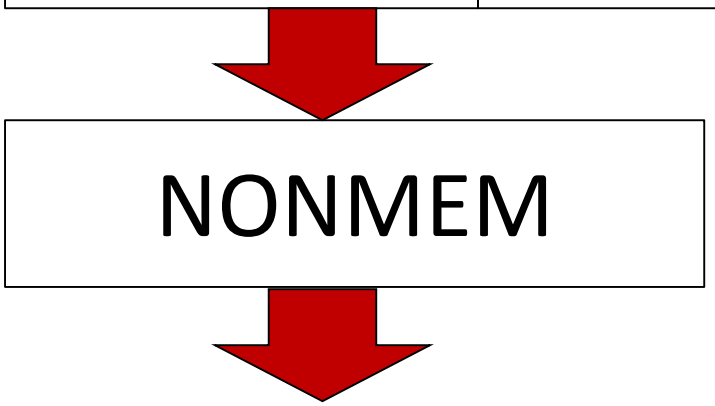
- Implementation in the context of population PK modeling (Bies and Sale 2006, JPP August, Sherer Sale and Bies 2012 JPP)
- Potential models are reduced to a bit-string (base-2 number assembly) that reflects the model “genetic” code
- Each model feature is coded as a base 2 number
  - If there are 2 options the values are 0 or 1 [(0) (1)], if more than two options then one has multiple bits eg. [(0 0), (0 1), (1 0), (1 1)]
- Features are strung together to produce aforementioned bit string
- Model can be reproduced based on the bit string that results

Global optimization: genetic algorithm

- Single-objective
  - Default composite fitness measure (initial implementation)
    - -2 x log-likelihood
    - Penalty per model variable (**10 points**)
    - Penalties for failure to converge (400), covariance (400), and correlation (300)

Model Selection

Compartment structure	Residual error	IIV on CL	Weight on CL	Weight on V
1 compartment	Additive	No relationship	**No relationship**	No relationship
1 compartment w/ lag	Proportional	Additive	Additive	Additive
2 compartments	**Combined**	Proportional	Proportional	**Proportional**
2 compartments w/ lag		**Exponential**	Exponential	Exponential
			Power-law	Power-law



- Model evaluation criteria
  - -2 x log-likelihood
  - Number of parameters
  - Diagnostic plots

Basic genetic algorithm

Candidate models (N = 300 – 500)

Candidate 1.	
Compartment structure 1 compartment **1 compartment lag** 2 compartments 2 compartments lag	Residual error Additive Proportional **Combined**

Candidate 2.	
Compartment structure 1 compartment **1 compartment lag** 2 compartments 2 compartments lag	Residual error **Additive** Proportional Combined

Candidate 3.	
Compartment structure 1 compartment 1 compartment lag **2 compartments** 2 compartments lag	Residual error Additive **Proportional** Combined

Basic genetic algorithm

Candidate models (N = 300 – 500)

Candidate 1. **Fitness = 1,000**	
Compartment structure 1 compartment **1 compartment lag** 2 compartments 2 compartments lag	Residual error Additive Proportional **Combined**

Candidate 2. **Fitness = 1,200**	
Compartment structure 1 compartment **1 compartment lag** 2 compartments 2 compartments lag	Residual error **Additive** Proportional Combined

Candidate 3. **Fitness = 1,050**	
Compartment structure 1 compartment 1 compartment lag **2 compartments** 2 compartments lag	Residual error Additive **Proportional** Combined

Evaluate fitness using NONMEM

Basic genetic algorithm

Candidate models (N = 300 – 500)

Candidate 1. **Fitness = 1,000**	
Compartment structure 1 compartment **1 compartment lag** 2 compartments 2 compartments lag	Residual error Additive Proportional **Combined**

Candidate 2. **Fitness = 1,200**	
Compartment structure 1 compartment **1 compartment lag** 2 compartments 2 compartments lag	Residual error **Additive** Proportional Combined

Candidate 3. **Fitness = 1,050**	
Compartment structure 1 compartment 1 compartment lag **2 compartments** 2 compartments lag	Residual error Additive **Proportional** Combined

Evaluate fitness using NONMEM

1 0 **0 1** Binary representation of model decisions

Basic genetic algorithm

Candidate 1. Fitness = 1,000	
Compartment structure 1 compartment **1 compartment lag** 2 compartments 2 compartments lag	Residual error Additive Proportional **Combined**

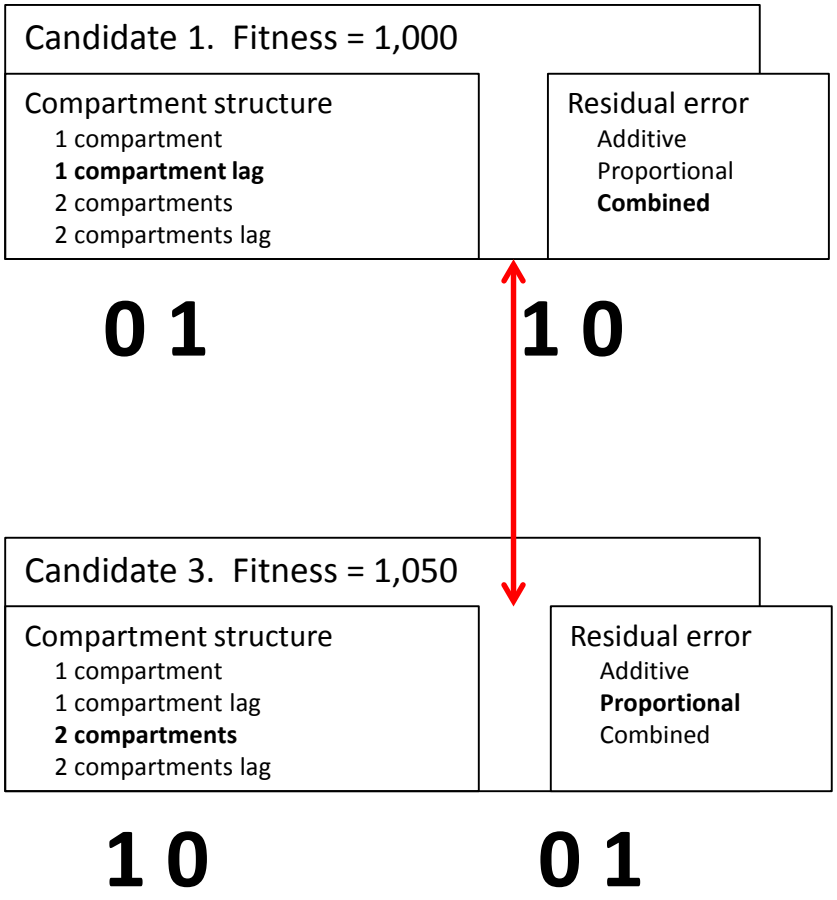
0 1 **1 0**

Candidate 3. Fitness = 1,050	
Compartment structure 1 compartment 1 compartment lag **2 compartments** 2 compartments lag	Residual error Additive **Proportional** Combined

1 0 **0 1**

Reproduction:
Randomly select two models from the candidate pool based on normalized fitness

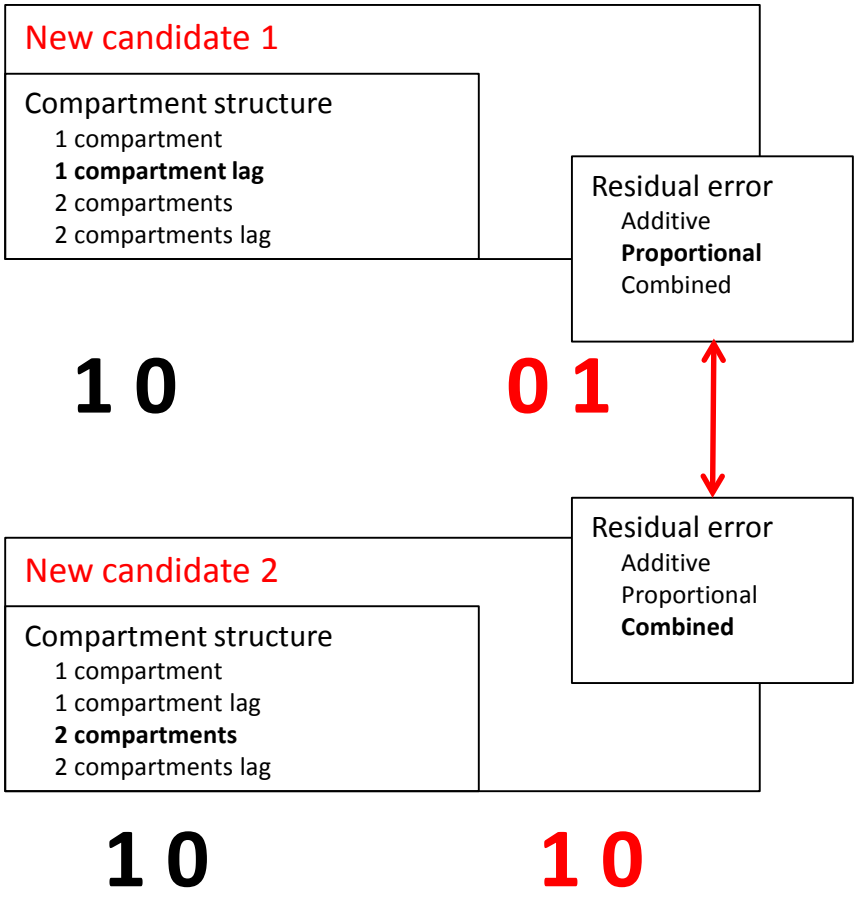
Basic genetic algorithm



Crossover:

Randomly select a model location

Basic genetic algorithm



Crossover:

Randomly select a model location

Swap model information with probability $P_{crossover}$

Basic genetic algorithm

New candidate 1	
Compartment structure 1 compartment **1 compartment lag** 2 compartments 2 compartments lag	Residual error **Additive** Proportional Combined
1 0	**0 0**

Mutation:

Randomly select a model location

New candidate 2	
Compartment structure 1 compartment 1 compartment lag **2 compartments** 2 compartments lag	Residual error Additive Proportional **Combined**
1 0	**1 0**

Change model information with probability $P_{mutation}$

Basic genetic algorithm

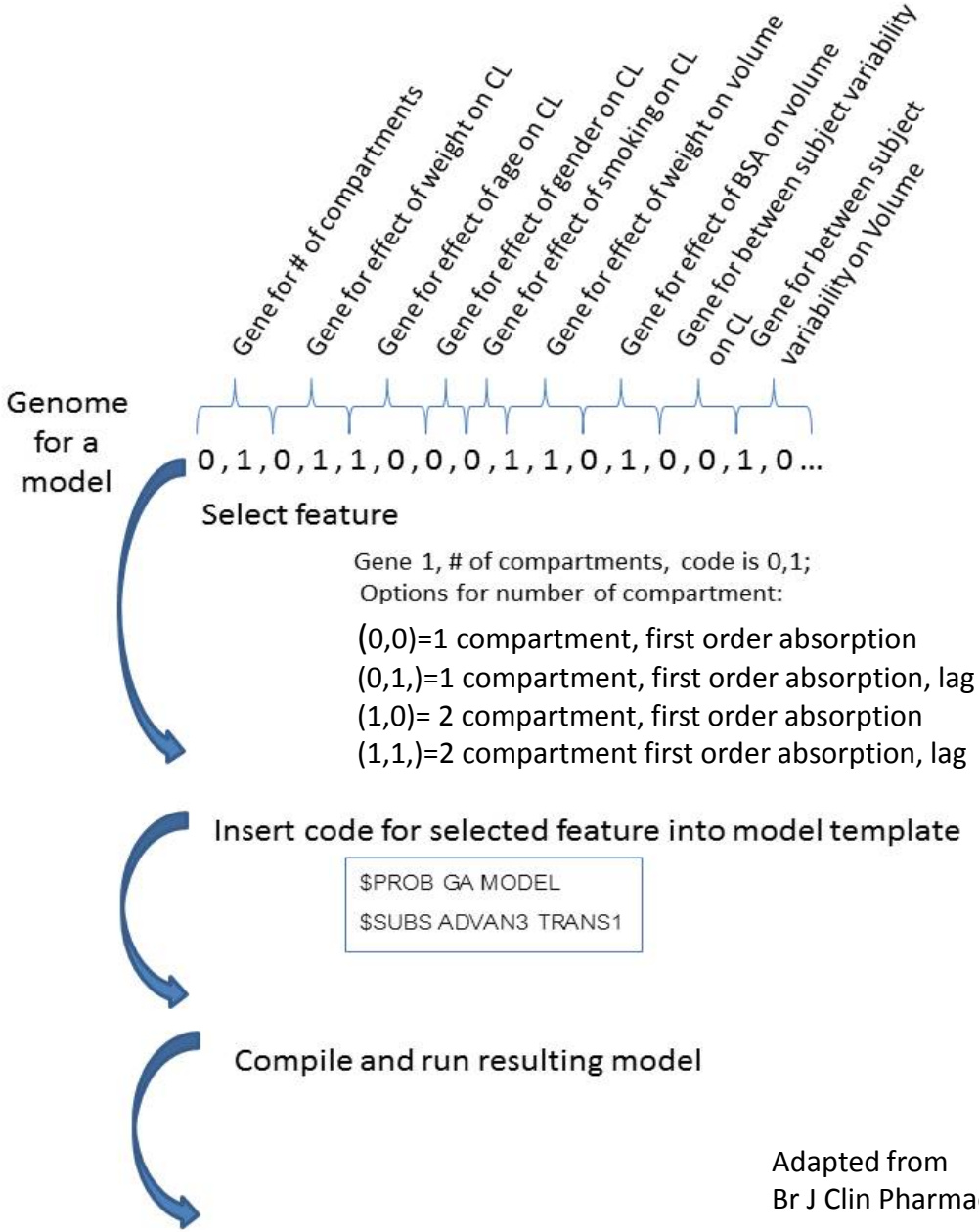
New candidate models

New candidate 1.	
Compartment structure 1 compartment **1 compartment lag** 2 compartments 2 compartments lag	Residual error **Additive** Proportional Combined

New candidate 2.	
Compartment structure 1 compartment 1 compartment lag **2 compartments** 2 compartments lag	Residual error Additive Proportional **Combined**

Repeat reproduction, crossover, and mutation operations until a new candidate pool is created

Repeat process for desired number of 30-50 generations



Adapted from
Br J Clin Pharmacol 2013 Jun 17 Epub ahead of print

Figure 3. Coding of model features and translation into a model. If only two options are examined for a feature (e.g., the effect of Gender on Clearance) only 1 bit will be needed for that gene. If more than two options are examined (e.g., 4 for the basic structure, number of compartments) more than 1 bit is required for that gene. The final genome for each model is constructed by concatenating all the genes together into a bit string.

Covariate Search Comparison

- Evaluation of performance of multiple methods
  - True model simulated with relatively dense sampling
  - Exponential relationship with BMI and CrCL on clearance
  - Exponential relationship BSA and Sex on volume
  - Compared:
    - Stepwise Covariate Modeling
    - LASSO (least absolute shrinkage and selection operator)
    - Single Objective Hybrid Genetic Algorithm

Covariate Search Comparison

Table 5 True and spurious covariate relationships identified in the simulated data by the automated stepwise covariate modeling, Lasso, and SOHGA approaches and the models fit characteristics

Method	"True" covariates		Spurious covariates		Objective function value
	Clearance	Volume of distribution	Clearance	Volume of distribution	
Original model	BMI, CRCL	BSA, Sex	–	–	6101.2
Stepwise covariate modeling (SCM): *p* value for inclusion, *p* value for elimination					
0.05, 0.05	BMI, CRCL	Sex	WT	HT, CV1	6085.9
0.05, 0.01	BMI, CRCL	Sex	–	HT, CV1	6091.1
0.10, 0.01	BMI, CRCL	Sex	–	HT, CV1	6091.1
Lasso model	BMI, CRCL	–	–	–	6254.2
Single-objective, hybrid genetic algorithm					
3.84 point penalty per parameter	BMI, CRCL	Sex	BSA	HT, CV1	6086.7
10 point penalty per parameter	BMI, CRCL	Sex	–	HT	6097.9

BMI body mass index, *BSA* body surface area, *CRCL* creatinine clearance, *CV1* unrelated covariate 1, *HT* height, *WT* weight

Sherer et al 2012, JPKPD

Single-objective, hybrid genetic algorithm (SOHGA) vs. step-wise approach

- Pharmacokinetic data for Risperidone
  - Identical model options / decisions
- Compare information criteria of final models
  - Compare model structures

Compound	Administration method	Number of patients	Number of concentration measurements
CATIE			
Risperidone	Oral	490	1,236

Compound	Administration method	Number of patients	Number of concentration measurements
CATIE			
Risperidone	Oral	490	1,236

	NONMEM model structures tested	First-order (FO) or first-order conditional (FOCE) estimation	Number of covariates collected
Risperidone, oral	ADVAN2, TRANS2 ADVAN4, TRANS4 (with 1, 2, or 3 clearance subpopulations)	FO	9

Compound	Administration method	Number of patients	Number of concentration measurements
CATIE			
Risperidone	Oral	490	1,236

	NONMEM model structures tested	First-order (FO) or first-order conditional (FOCE) estimation	Number of covariates collected
Risperidone, oral	ADVAN2, TRANS2 ADVAN4, TRANS4 (with 1, 2, or 3 clearance subpopulations)	FO	9

	Convergence			Covariance step (condition number)	
	Final step-wise model	**Best SOHGA candidate**		**Final step-wise model**	**Best SOHGA candidate**
Risperidone, oral	Required fixing K_a early in model building process	Successful		Successful (60)	Successful (1.17×10^6)

Compound	Administration method	Number of patients	Number of concentration measurements
CATIE			
Risperidone	Oral	490	1,236

	NONMEM model structures tested	First-order (FO) or first-order conditional (FOCE) estimation	Number of covariates collected
Risperidone, oral	ADVAN2, TRANS2 ADVAN4, TRANS4 (with 1, 2, or 3 clearance subpopulations)	FO	9

	Convergence			Covariance step (condition number)	
	Final step-wise model	**Best SOHGA candidate**		**Final step-wise model**	**Best SOHGA candidate**
Risperidone, oral	Required fixing K_a early in model building process	Successful		Successful (60)	Successful (1.17×10^6)

Compound	Final stepwise model	Best SOHGA candidate model	$AIC_{SOHGA} - AIC_{stepwise}$
Risperidone, oral	AIC = 5,131.1	AIC = 4,853.0	-278.1

Model structure: SOHGA vs. step-wise

Compound	Final step-wise model	Best SOHGA candidate
Risperidone, oral	1 with 3 component mixture on CL	**2 with 2 component mixture on CL**

- Extra degree of freedom
  - Fix k_a based on literature due to instability
    - Risperidone (ΔAIC = -278.1)
      - 1 covariate in final stepwise model
      - 5 covariates in best SOHGA candidate

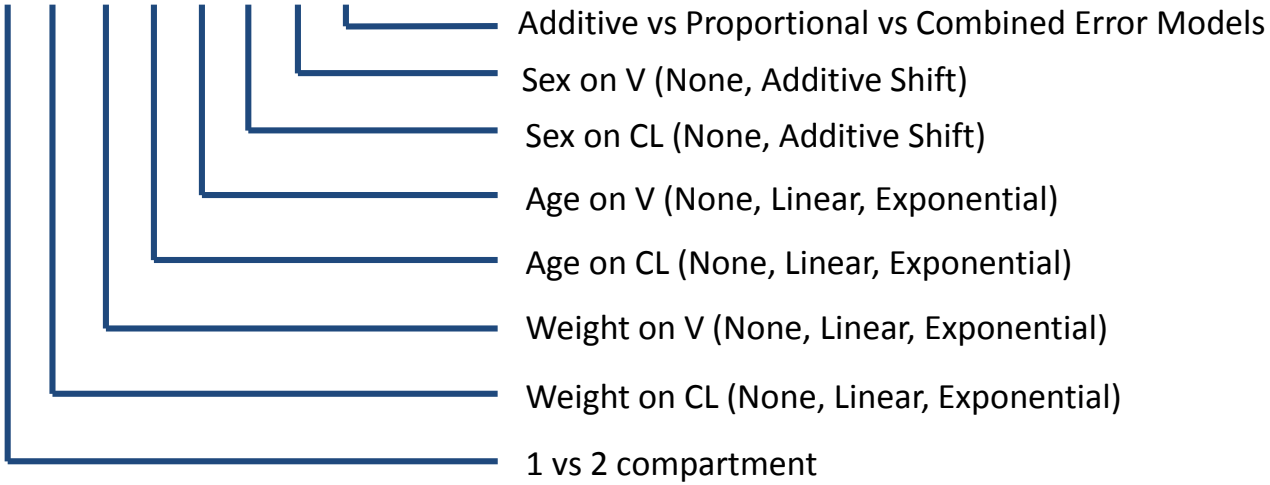
Example Model Search Space

An example:

- Structure: 1, 2 compartment distribution model
- Covariates: Weight on CL, V | Age on CL, V | Sex on CL, V
  - Linear: $TV_{Param} = THETA_A + \left((Cov_i - \widehat{\text{Cov}}) * THETA_B\right)$
  - Exponential: $TV_{Param} = THETA_A * e^{\left(Cov_i - \widehat{\text{Cov}}\right) * THETA_B}$
- Statistical: Additive, Proportional, Combined

Example Model Search Space

- Total number of models:
  - 2*3*3*3*3*2*2*3 = 1944 possible combinations

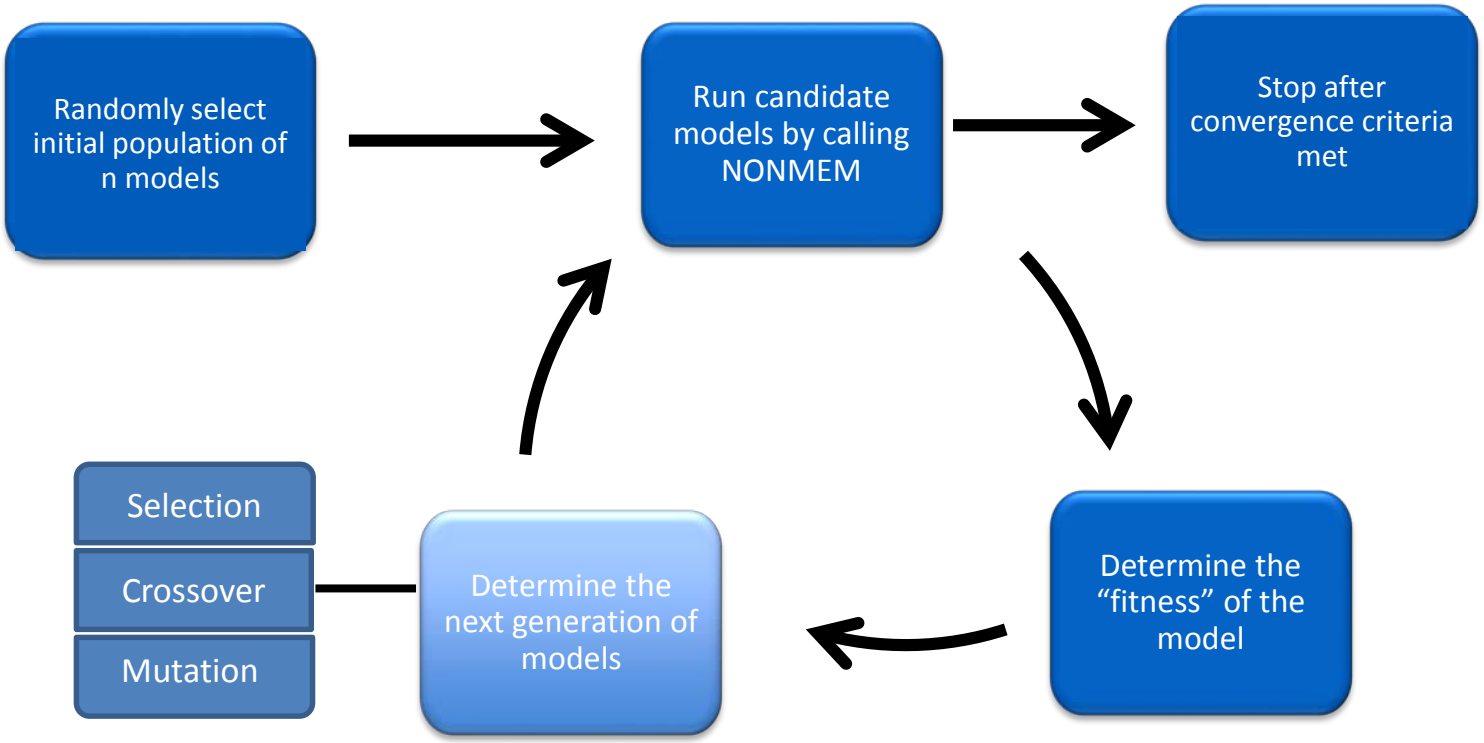


Example Model Search Space

- Total number of models:
  - 2*3*3*3*3*2*2*3 = 1944 possible combinations

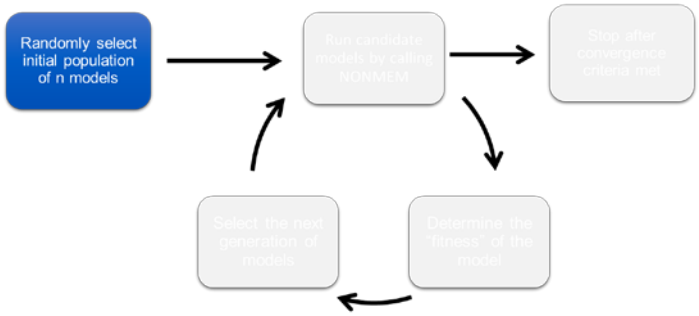
Model	N_{CMT}	Weight on CL	Weight on V	Age on CL	Age on V	Sex on CL	Sex on V	Error Model
1	1	None	None	None	None	None	None	Additive
2	1	Linear	None	None	None	None	None	Additive
3	1	Exponential	None	None	None	None	None	Additive
4	1	None	Linear	None	None	None	None	Additive
5	1	None	Exponential	None	None	None	None	Additive
…	…	…	…	…	…	…	…	…
1944	2	Exponential	Exponential	Exponential	Exponential	Additive	Additive	Combined

Outline of GA



Initial Population

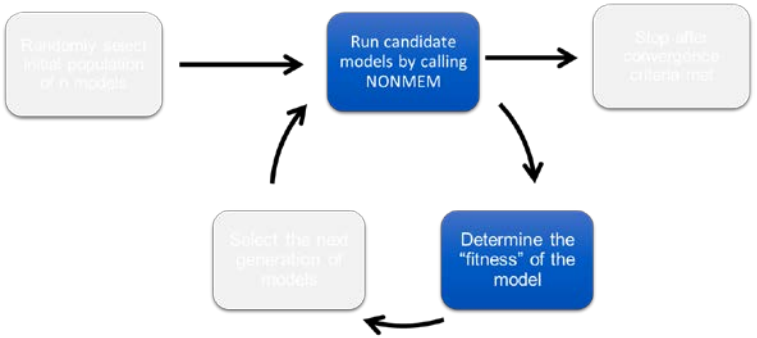
- n models, or “individuals”, are randomly selected from the pool of all combinations
- Models are run simultaneously



Model	N_{CMT}	Weight on CL	Weight on V	Age on CL	Age on V	Sex on CL	Sex on V	Error Model
83	1	Linear	None	Linear	Exponential	None	Exponential	Additive
225	1	Linear	Exponential	Exponential	Linear	None	None	Proportional
343	1	Exponential	None	None	Linear	None	Linear	Proportional
800	2	None	Linear	Exponential	None	Exponential	None	Combined
1284	2	Exponential	Exponential	Linear	Exponential	None	None	Additive
1491	2	Exponential	None	None	Linear	None	Linear	Additive

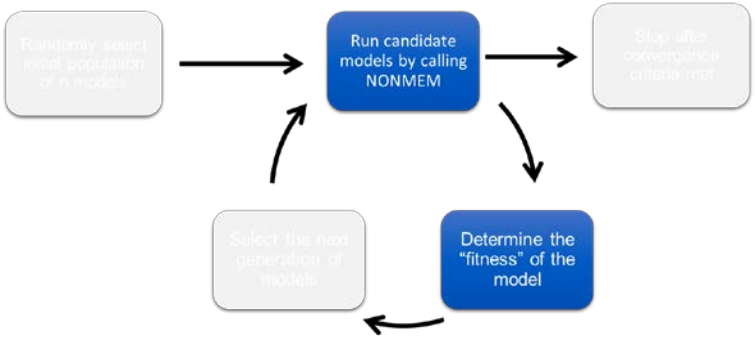
Fitness

- How to determine how “fit” a model is?



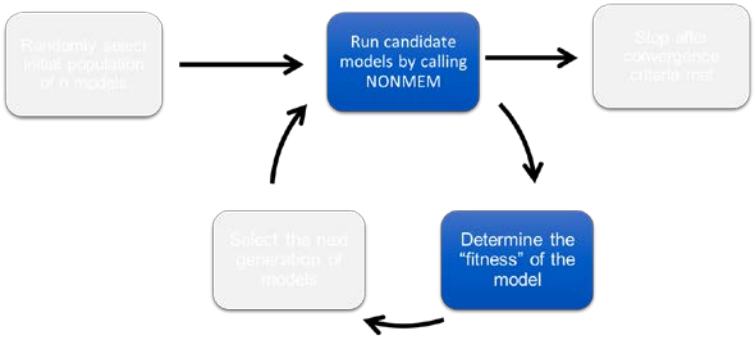
Fitness

- How to determine how “fit” a model is?
- NONMEM objective function?



Fitness

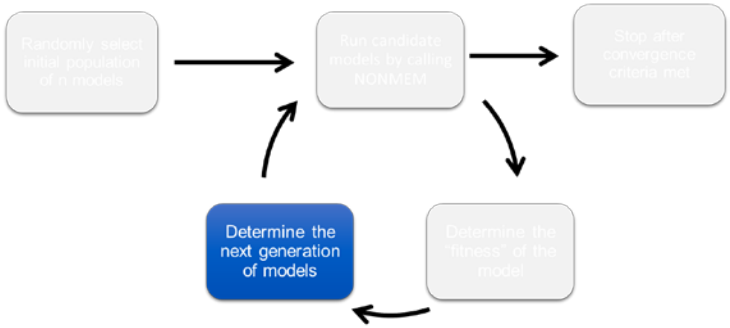
- How to determine how “fit” a model is?
- NONMEM objective function?
- Objective function + Penalty terms



$$Fitness = \underbrace{-2LL + 2 * N_{Par}}_{\text{AIC}} + 20 * Penalty_{Converge} + 10 * Penalty_{Covar}$$

Selection

- Tournament style selection
- Ranked selection method
  - Ideal when fitness values are close in magnitude



for each model i
choose a random opponent model j (excluding i)
the more fit model wins the tournament
winner proceeds to the cross-over pool

Initial Population

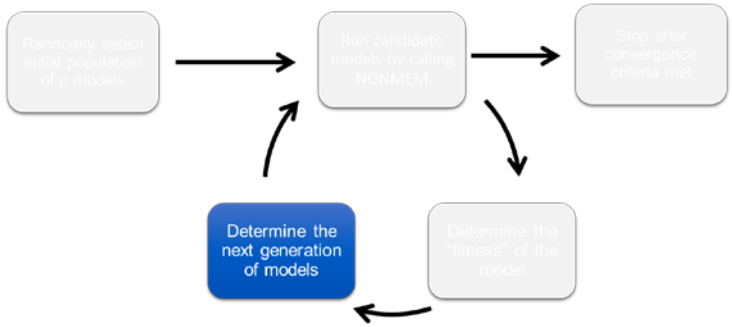
Model	Fitness
83	100
225	102
343	98
800	94
1284	103
1491	109

Crossover Pool

Model	Fitness

Selection

- Tournament style selection
- Ranked selection method
  - Ideal when fitness values are close in magnitude



for each model i
choose a random opponent model j (excluding i)
the more fit model wins the tournament
winner proceeds to the cross-over pool

Initial Population

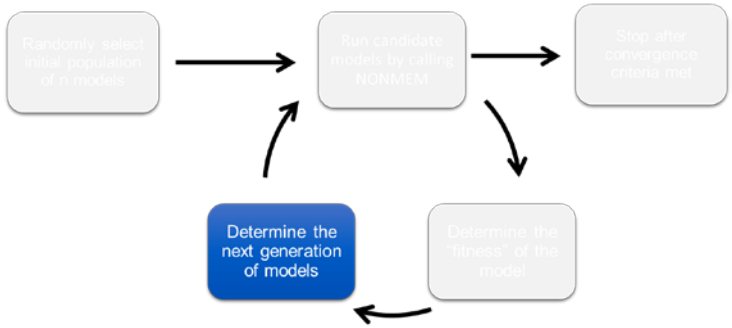
Model	Fitness
83	100
225	102
343	98
800	94
1284	103
1491	109

Crossover Pool

Model	Fitness
800	94

Selection

- Tournament style selection
- Ranked selection method
  - Ideal when fitness values are close in magnitude



for each model i
choose a random opponent model j (excluding i)
the more fit model wins the tournament
winner proceeds to the cross-over pool

Initial Population

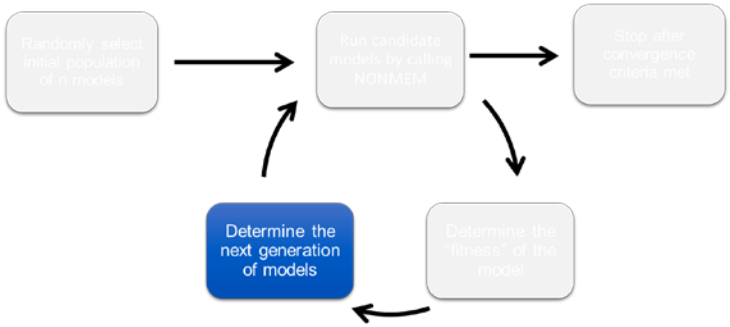
Model	Fitness
83	100
225	102
343	98
800	94
1284	103
1491	109

Crossover Pool

Model	Fitness
800	94
225	102

Selection

- Tournament style selection
- Ranked selection method
  - Ideal when fitness values are close in magnitude



for each model i
choose a random opponent model j (excluding i)
the more fit model wins the tournament
winner proceeds to the cross-over pool

Initial Population

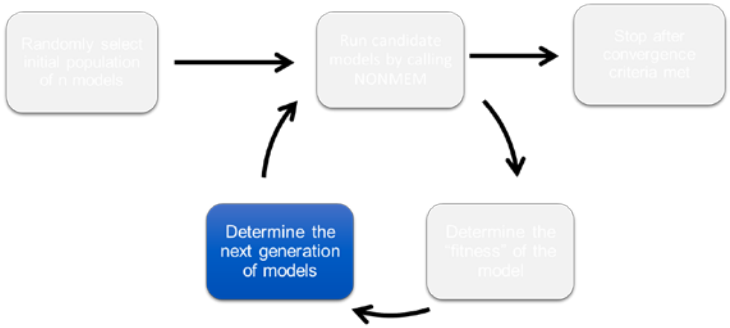
Model	Fitness
83	100
225	102
343	98
800	94
1284	103
1491	109

Crossover Pool

Model	Fitness
800	94
225	102
343	98
800	94

Selection

- Tournament style selection
- Ranked selection method
  - Ideal when fitness values are close in magnitude



for each model i
choose a random opponent model j (excluding i)
the more fit model wins the tournament
winner proceeds to the cross-over pool

Initial Population

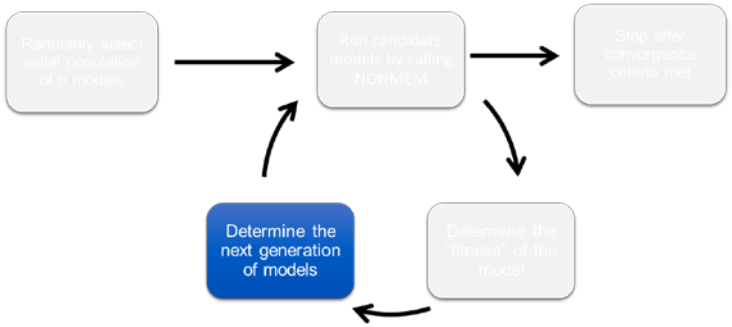
Model	Fitness
83	100
225	102
343	98
800	94
1284	103
1491	109

Crossover Pool

Model	Fitness
800	94
225	102
343	98
800	94
83	100

Selection

- Tournament style selection
- Ranked selection method
  - Ideal when fitness values are close in magnitude



for each model i
choose a random opponent model j (excluding i)
the more fit model wins the tournament
winner proceeds to the cross-over pool

Initial Population

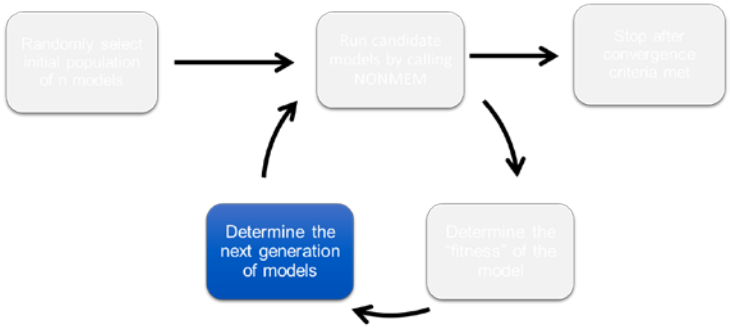
Model	Fitness
83	100
225	102
343	98
800	94
1284	103
1491	109

Crossover Pool

Model	Fitness
800	94
225	102
343	98
800	94
83	100
343	98

Crossover

- Mimics biological reproduction
- Combines elements of well performing models to produce potentially more fit models
- Two-point crossover



Crossover

Parent Chromosomes

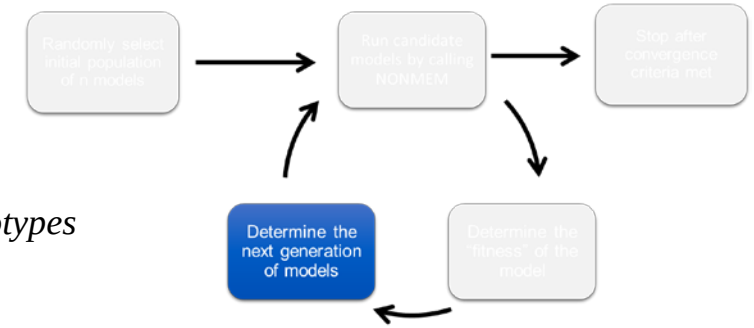
Model	Fitness	N_{CMT}	Weight on CL	Weight on V	Age on CL	Age on V	Sex on CL	Sex on V	Error Model
800	94	2	None	Linear	Exponential	None	Exponential	None	Combined
343	98	1	Exponential	None	None	Linear	None	Linear	Proportional

Progeny

Model	Fitness	N_{CMT}	Weight on CL	Weight on V	Age on CL	Age on V	Sex on CL	Sex on V	Error Model
	---	2	None	None	None	Linear	None	None	Combined
	---	1	Exponential	Linear	Exponential	None	Exponential	Linear	Proportional

Mutation

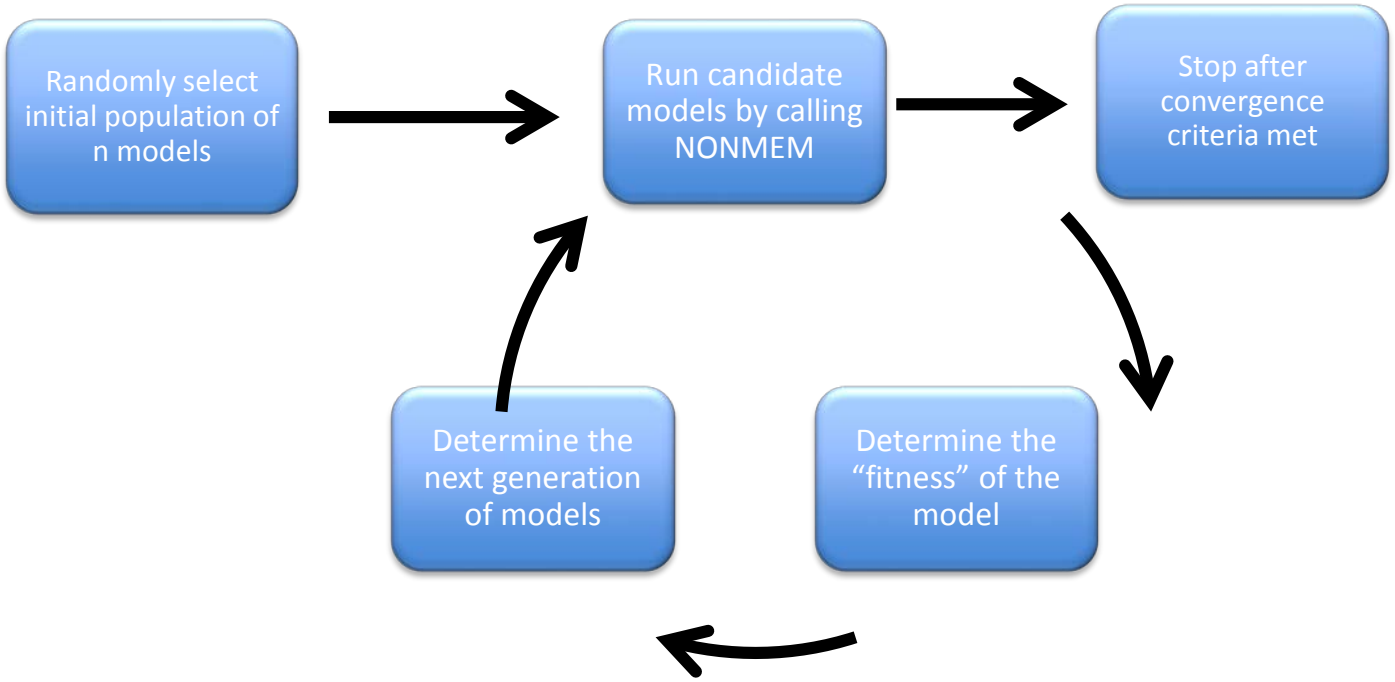
for each model i
for each gene j
mutate gene (T/F) with probability 0.05
if (mutate gene = T)
newPhenotypeIndex = sample integer from 1 to length of phenotypes
phenotype = phenotypes[newPhenotypeIndex]
gene[j] = phenotype



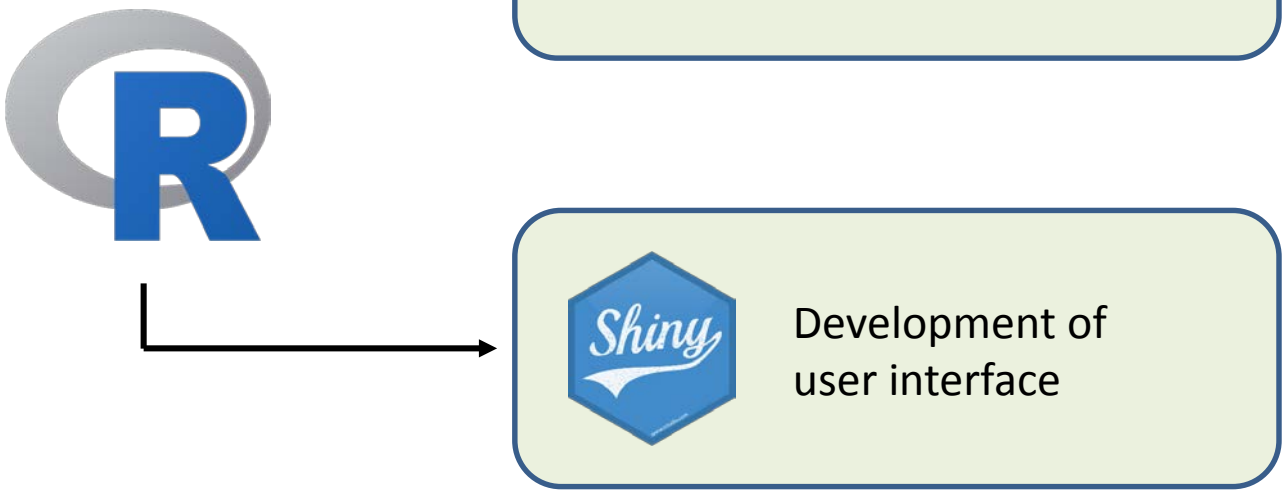
Mutate: F F F F T F F

Weight on CL	Weight on V	Age on CL	Age on V	Sex on CL	Sex on V	Error Model
None	None	None	Linear	Additive	None	Combined

Outline of GA



Software

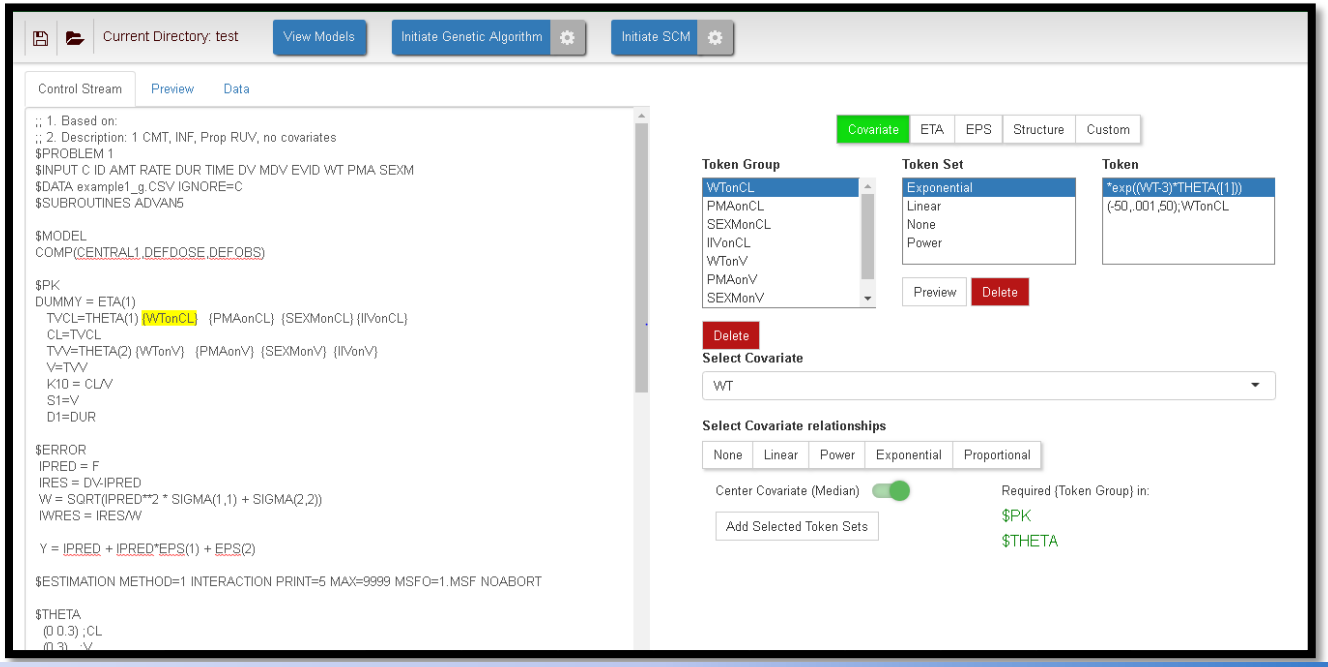


PsN 3
Perl speaks NONMEM

XPOSE

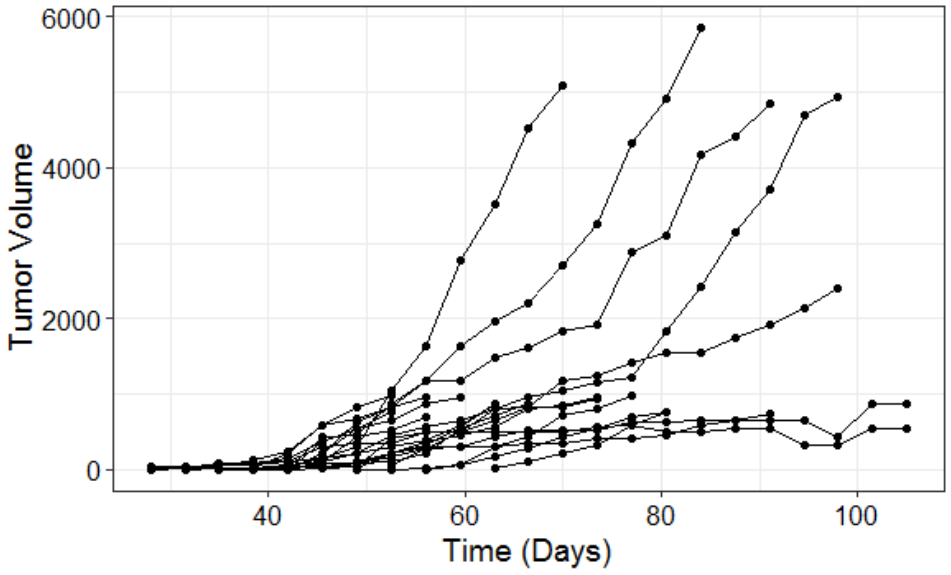
University at Buffalo
The State University of New York

Development of NONMEM Workbench to Implement Genetic Algorithm



Case Study: Tumor Progression Modeling

- Unperturbed tumor growth trajectories of 22 LNCAP xenograft tumors were selected as test dataset



Case Study: Tumor Progression Modeling

- 1584 unique models were created by the GA app with the combinations listed to the right:

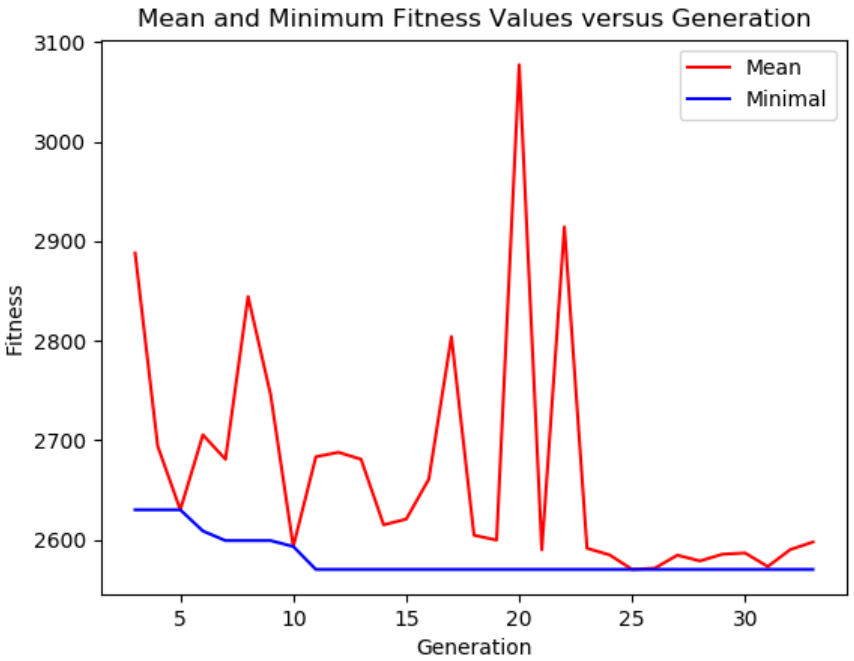
Tumor Growth Model	Equation	# of θ	# of IIV per θ^*	# of RUV^{**}	Number of Unique models
Exponential	$\frac{dV}{dt} = \lambda_0 \times V$	2	4	3	$4^2 \times 3 = 48$
Power	$\frac{dV}{dt} = \lambda_0 \times V^{\gamma}$	3	4	3	$4^3 \times 3 = 192$
Logistic	$\frac{dV}{dt} = \lambda_0 \times V \times (1 - \frac{V}{T_{max}})$	3	4	3	$4^3 \times 3 = 192$
Gompertz	$\frac{dV}{dt} = \lambda_0 \times V \times \log\left(\frac{TUM_{max}}{V}\right)$	3	4	3	$4^3 \times 3 = 192$
Simeoni	$\frac{dV}{dt} = \frac{\lambda_0 \times V}{(1 + (\frac{\lambda_0}{\lambda_1} \times V))^{\psi})^{\frac{1}{\psi}}}$	4	4	3	$4^4 \times 3 = 768$
Koch [1]	$\frac{dV}{dt} = \frac{\lambda_0 \times V \times 2 \times \lambda_1}{(\lambda_1 + 2 \times \lambda_0 \times V)}$	3	4	3	$4^3 \times 3 = 192$
					Sum: 1584

* The four IIV structures are: none, additive, proportional, and exponential.
** The three RUV structures are additive, proportional, and additive plus proportional.

[1] Koch G1, Walz A, Lahu G, Schropp J. Modeling of tumor growth and anticancer effects of combination therapy. J Pharmacokinet Pharmacodyn. 2009 Apr;36(2):179-97.

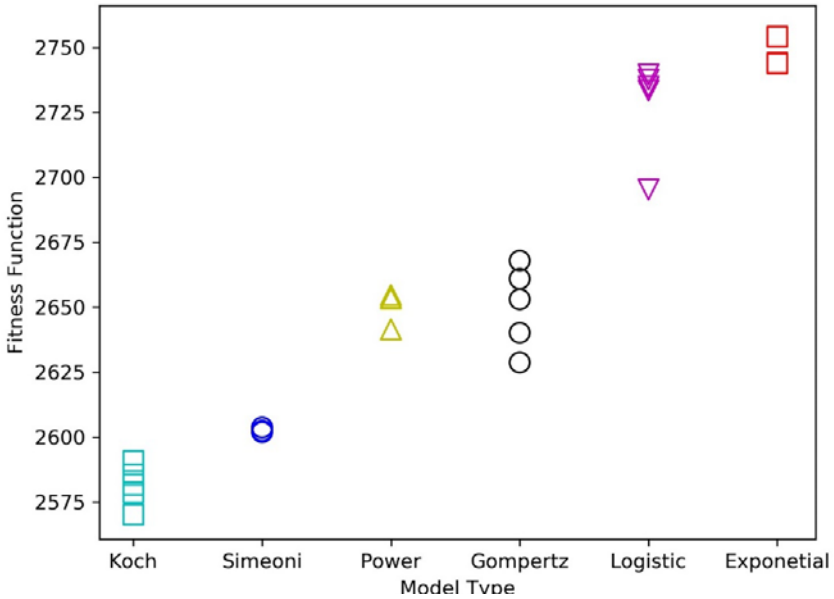
Case Study: Tumor Progression Modeling

- Based on the available computation power (40 available cores), run 38 models simultaneously.
- It took on average 4 minutes to run a generation.
- The algorithm found the best model by the 15th generation
- To confirm model convergence, the system was allowed to continue for a total of 30 generations.
- 250 out of 1584 unique models were run by the 30th generation.



Case Study: Tumor Progression Modeling

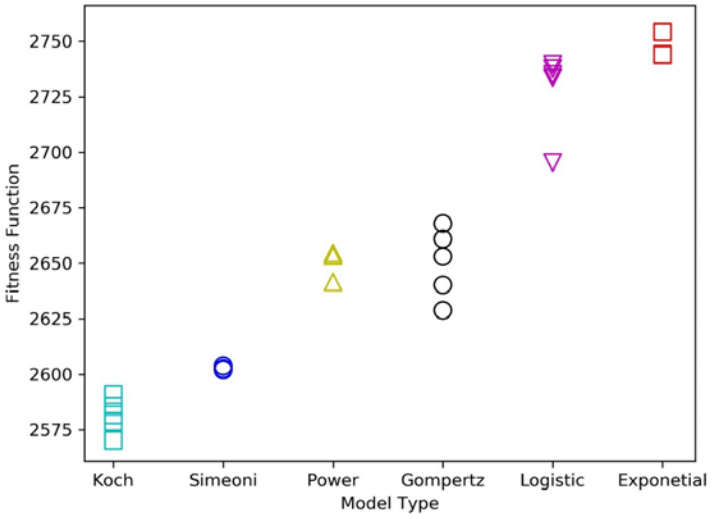
- The Koch growth model performed best for the xenograft tumor dataset.
  - Fitness value of **2572**
- The model with the best fitness had the following IIV characteristics:
  - An exponential IIV model on λ0
  - An exponential IIV model on λ1
  - An exponential IIV model on baseline.
  - The residual error model selected was additive plus proportional.
- Standard step-wise approach conducted by blinded colleague resulted in fitness value of **2748** (Simeoni structure)



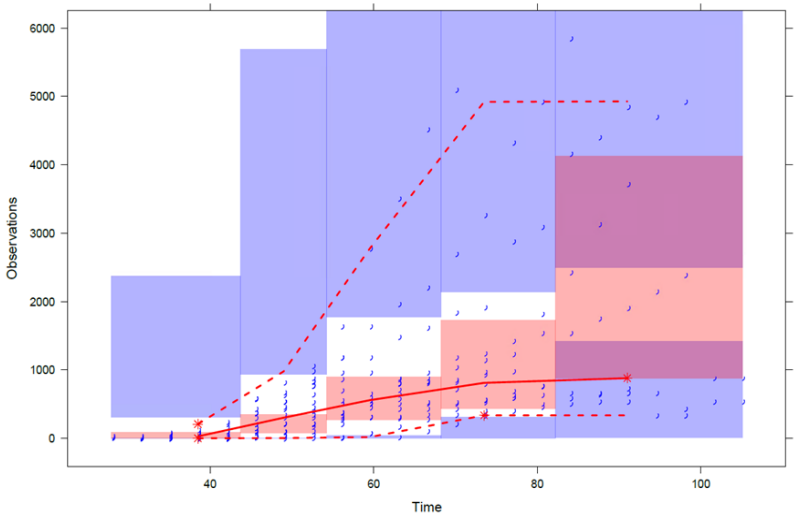
Top five fitness values for the six commonly used growth model categories

Model Selection Results

The Koch growth model performed best for the test dataset. The model with the best fitness had the following IIV characteristics: a exponential IIV model on λ0; exponential IIV model on λ1; and exponential IIV model on baseline;. The residual error model selected was additive plus proportional.



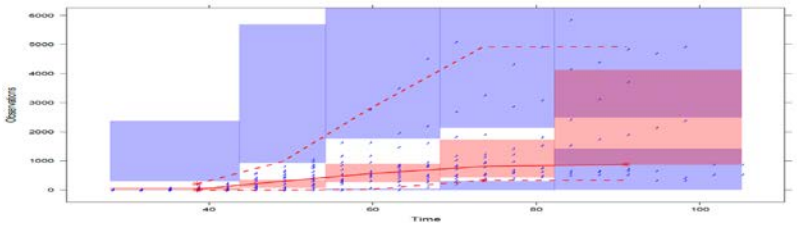
The plot of the top five fitness function for the six commonly used growth model categories (Koch, Simeoni, power, Gompertz, Logistic, and exponential).



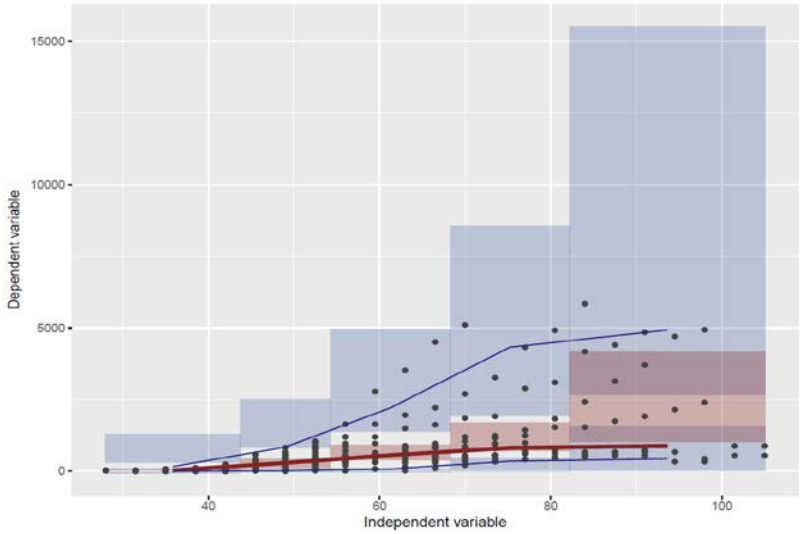
The VPC plot for the Koch model with the best fitness value of 2572. The red dashed lines are the predicted 5th and 95th percentiles.

Model Selection Results

The best fitness function of the GA selected model is 2572 for the Koch model, while the typical approach to model building conducted by a “blinded” colleague resulted in a fitness of 2748 for a Simeoni model. In addition, the best Simeoni model found by GA gets a fitness function of 2602.



The VPC plot for the Koch model with the best fitness value of 2572. The red dashed lines are the predicted 5th and 95th percentiles.



The VPC plot for the manual picked Simeoni model with the fitness value of 2748. The blue solid lines are the predicted 5th and 95th percentiles.

Limitations of SOHGA

- Only post-hoc visual predictive checks
- Single-objective
  - Ad hoc (user defined) weighting scheme
    - i.e., 10 points / parameter is $\chi^2 = 0.0016$
- Equally valid yet very different candidate models are possible
- Does not consider feasibility
    - Could modify weighting scheme

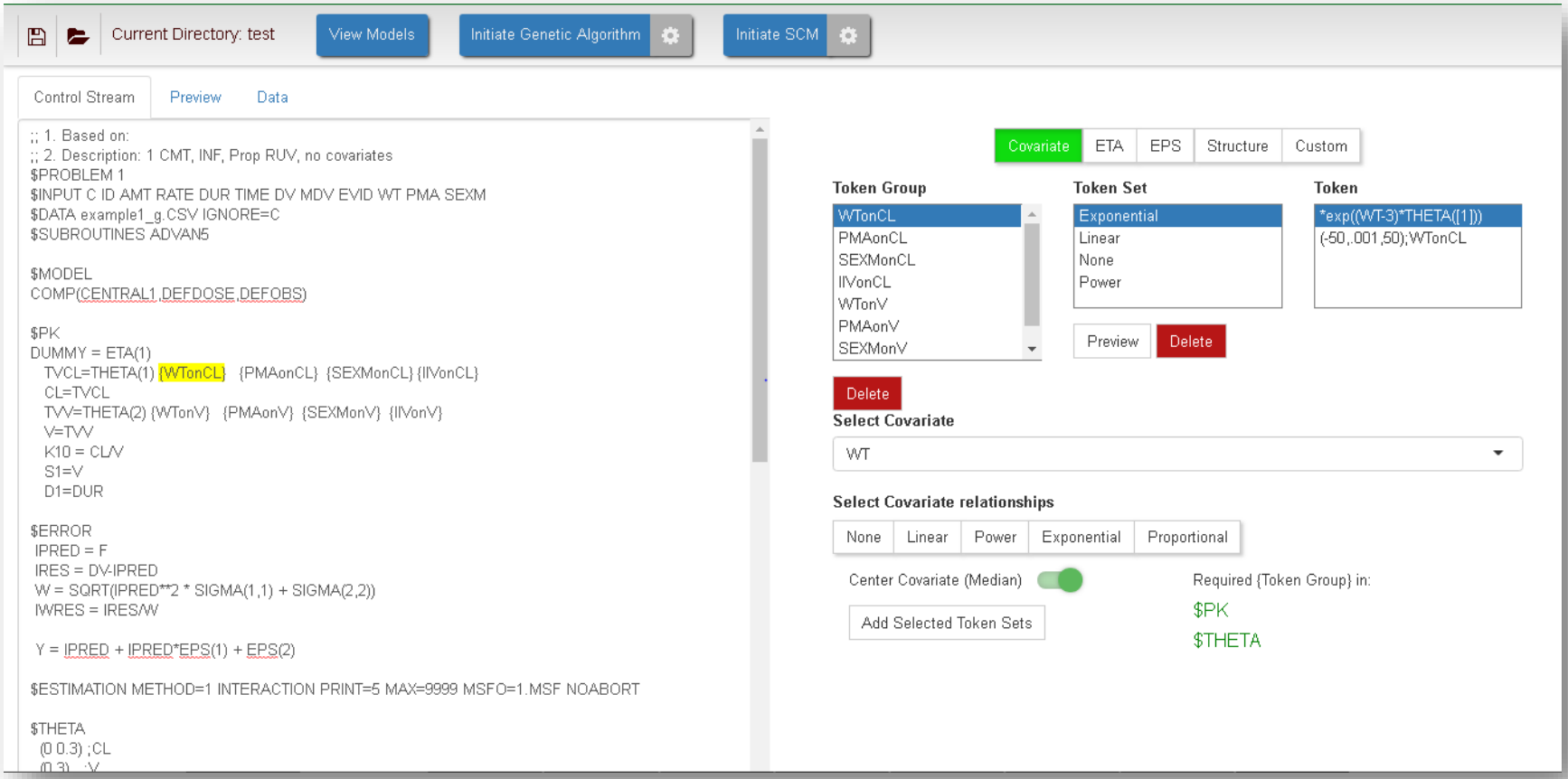
Conclusions

- The genetic algorithm identified a mixed effect model for risperidone PK and tumor trajectories that had substantially better OFV (and converted fitness) compared with the standard model search strategy.

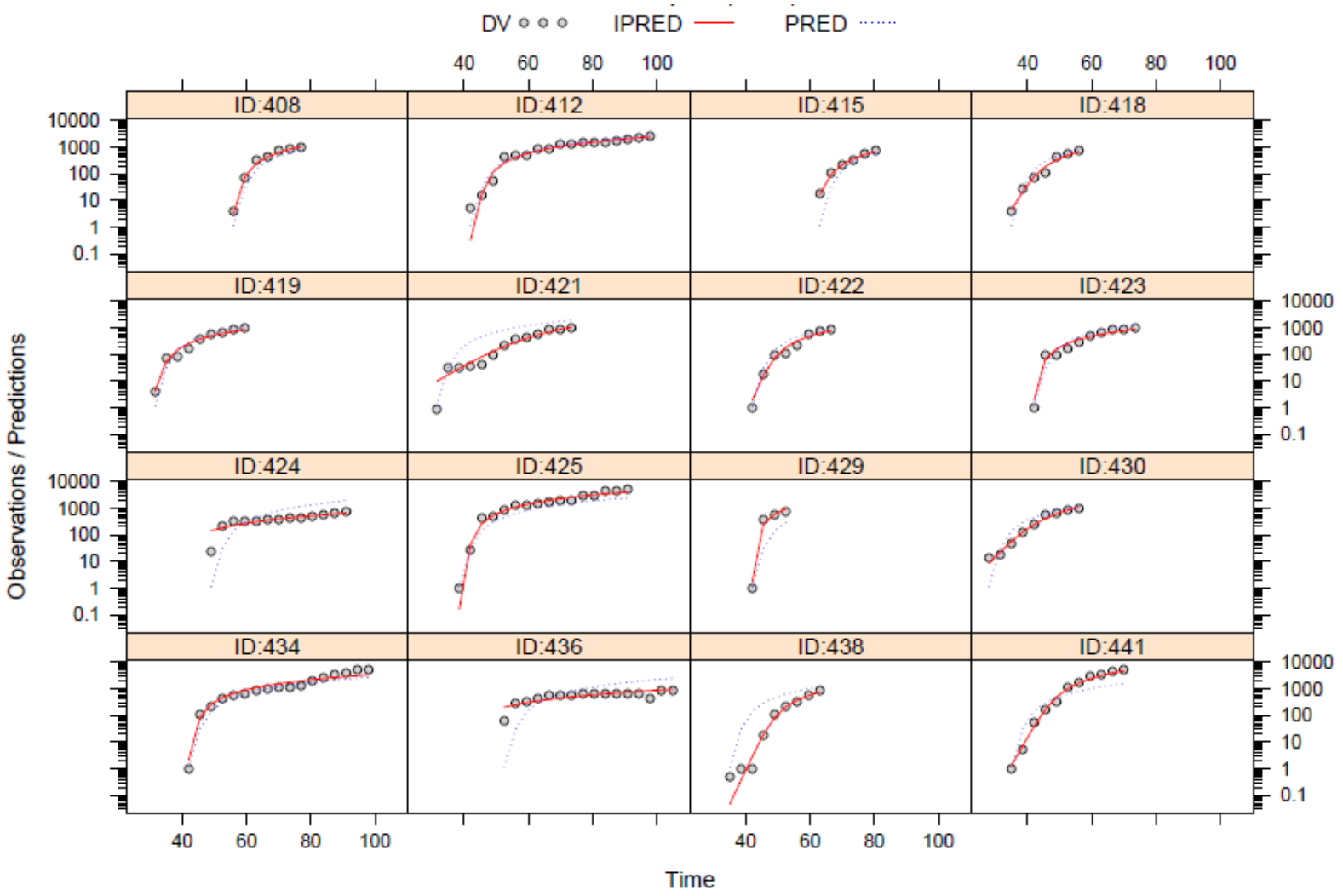
- The current app can improve the accuracy and efficiency of model development. An automated solution for population PK/PD modeling will allow modelers to focus on hypothesis generation and model evaluation rather than text processing and model execution.

Acknowledgements

- Mark Sale, Nuventra
- Mohamed Ismail, AbbVie and University at Buffalo
- Sihang Liu, University at Buffalo
- Nikhil Pillai, University at Buffalo
- Eric Sherer, Louisiana Technological University
- Risperidone
  - Bruce G. Pollock, University of Toronto
  - Jeffrey A. Lieberman, Columbia University
  - Stephen R. Marder, UCLA
- Tumor Trajectories
  - Beth Pflug, Roswell Park Cancer Center, Buffalo


Current Directory: test
View Models
Initiate Genetic Algorithm
Initiate SCM
Control Stream
Preview
Data
;; 1. Based on:
;; 2. Description: 1 CMT, INF, Prop RUV, no covariates
$PROBLEM 1
$INPUT C ID AMT RATE DUR TIME DV MDV EVID WT PMA SEXM
$DATA example1_g.CSV IGNORE=C
$SUBROUTINES ADVAN5
$MODEL
COMP(CENTRAL1,DEFDOSE,DEFOBS)
$PK
DUMMY = ETA(1)
TVCL=THETA(1) {WTonCL} {PMAonCL} {SEXMonCL} {IIVonCL}
CL=TVCL
TVV=THETA(2) {WTonV} {PMAonV} {SEXMonV} {IIVonV}
V=TVV
K10 = CL/V
S1=V
D1=DUR
$ERROR
IPRED = F
IRES = DV-IPRED
W = SQRT(IPRED**2 * SIGMA(1,1) + SIGMA(2,2))
IWRES = IRES/W
Y = IPRED + IPRED*EPS(1) + EPS(2)
$ESTIMATION METHOD=1 INTERACTION PRINT=5 MAX=9999 MSFO=1.MSF NOABORT
$THETA
(0 0.3) ;CL
Covariate
ETA
EPS
Structure
Custom
Token Group
WTonCL
PMAonCL
SEXMonCL
IIVonCL
WTonV
PMAonV
SEXMonV
Token Set
Exponential
Linear
None
Power
Token
*exp((WT-3)*THETA([1]))
(-50,.001,50);WTonCL
Preview
Delete
Delete
Select Covariate
WT
Select Covariate relationships
None
Linear
Power
Exponential
Proportional
Center Covariate (Median)
Add Selected Token Sets
Required {Token Group} in:
$PK
$THETA

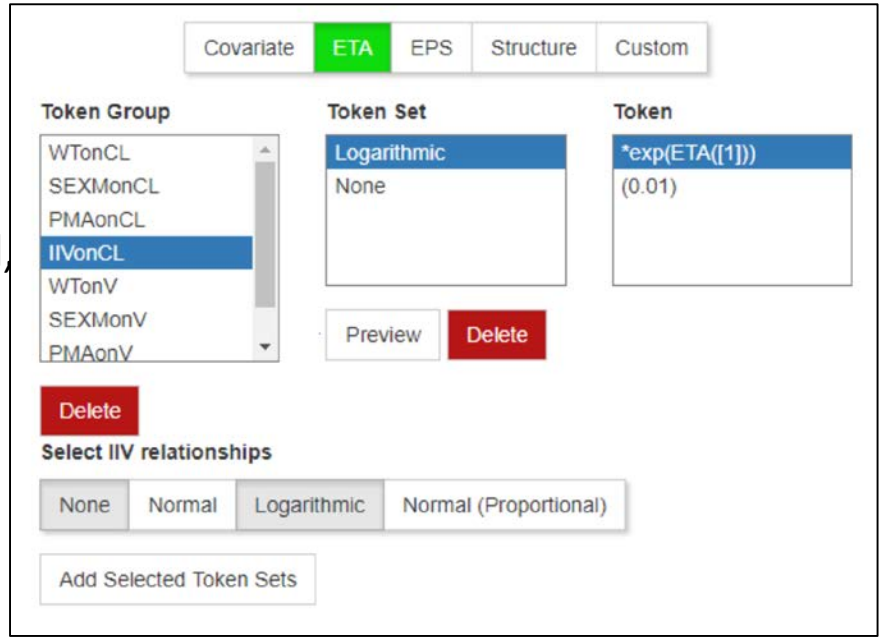
Case Study: Tumor Progression Modeling



NONMEM Workbench

Built in helpers for testing common model features

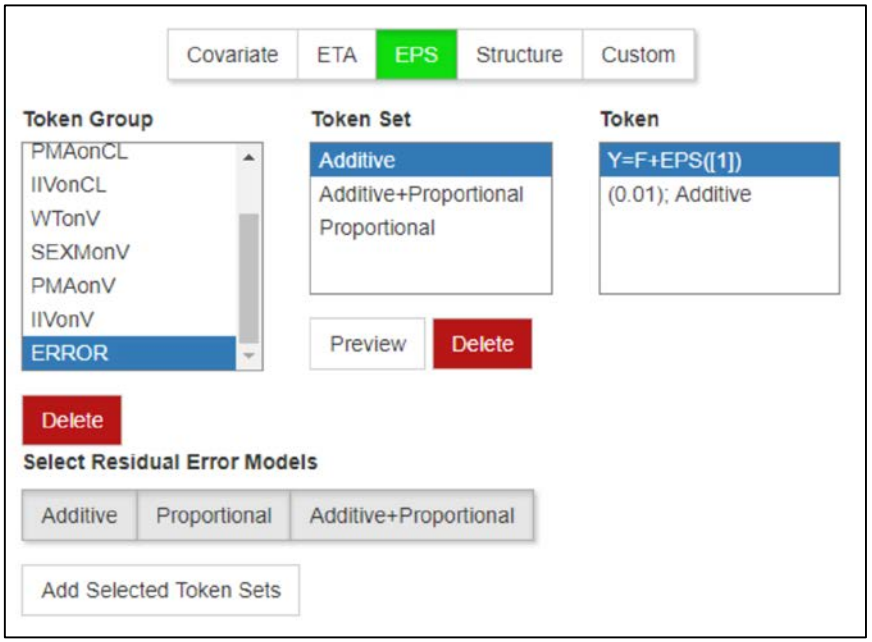
- Covariate effects
  - None, Linear, Power, Exponential, Proportional
- Interindividual Variability
  - None, Normal, Log-normal



NONMEM Workbench

Built in helpers for testing common model features

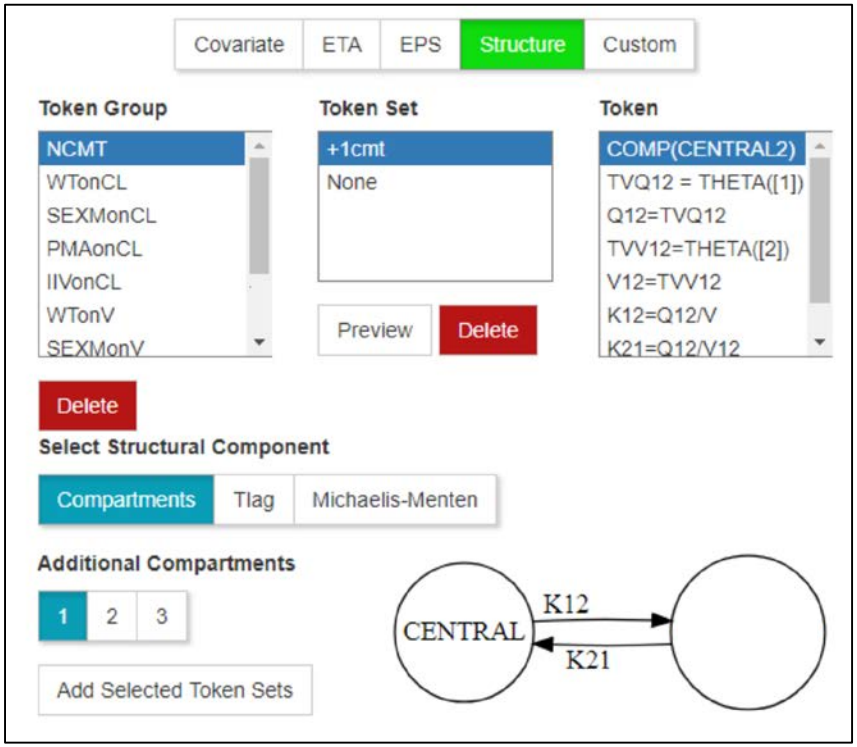
- Covariate effects
  - None, Linear, Power, Exponential, Proportional
- Interindividual Variability
  - None, Normal, Log-normal
- Residual Error
  - Additive, Proportional, Combined



NONMEM Workbench

Built in helpers for testing common model features

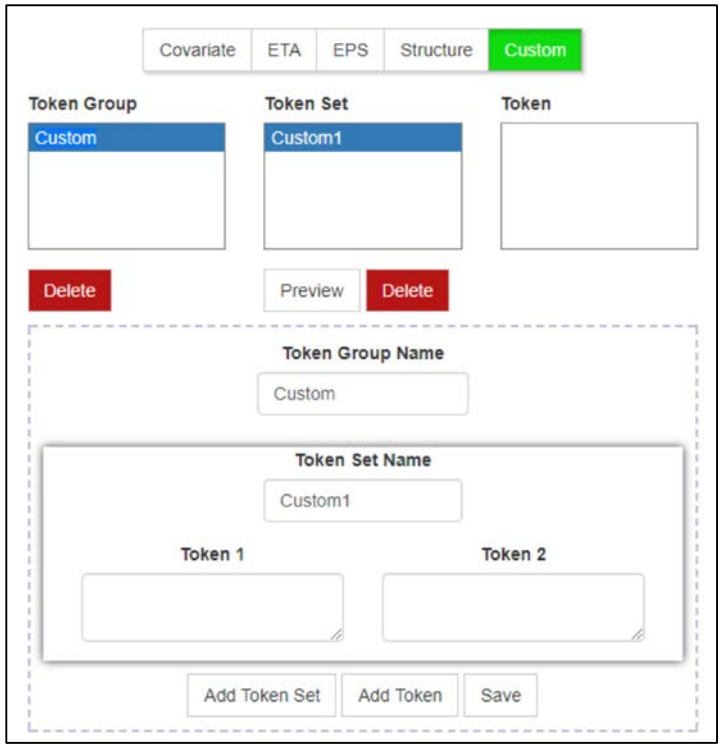
- Covariate effects
  - None, Linear, Power, Exponential, Proportional
- Interindividual Variability
  - None, Normal, Log-normal
- Residual Error
  - Additive, Proportional, Combined
- Structure
  - Number of distribution compartments
  - +/- lag time
  - +/- saturable clearance



NONMEM Workbench

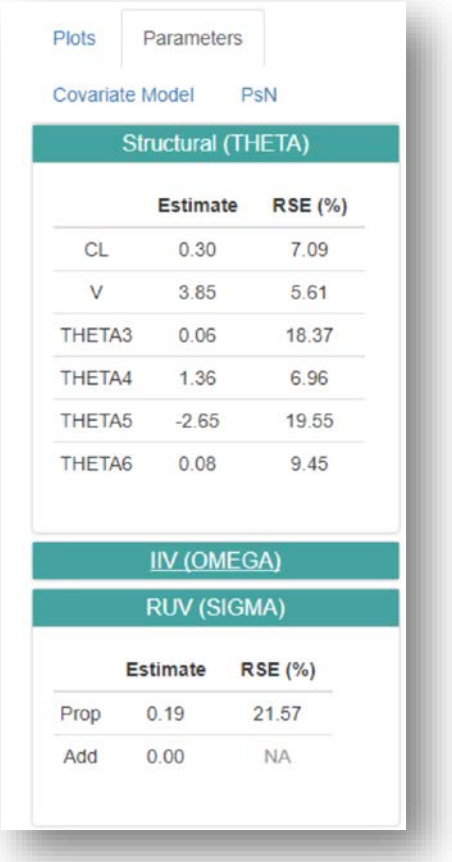
Built in helpers for testing common model features

- Covariate effects
  - None, Linear, Power, Exponential, Proportional
- Interindividual Variability
  - None, Normal, Log-normal
- Residual Error
  - Additive, Proportional, Combined
- Structure
  - Number of distribution compartments
  - +/- lag time
  - +/- saturable clearance
- Initial Estimates
  - Specify a range of initial estimates
- Custom token sets



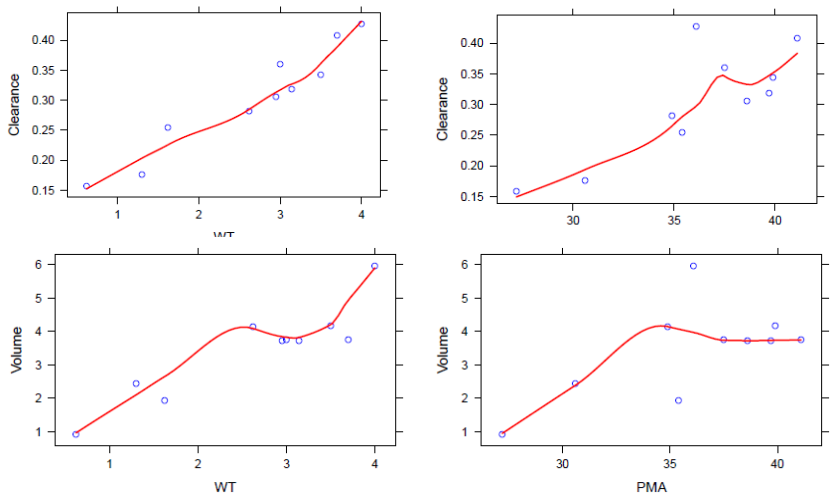
Post-processing

- View final parameter estimates and precision
- Integrated with Xpose
- Automatically generate diagnostics plots
  - Individual plots
  - DV vs PRED
  - DV vs IPRED
  - RES vs TIME, DV
- Run PsN modules
  - VPC
  - Log-likelihood profiling



Post-Processing

- Linear regression between:
  - Covariates and post-hoc parameters
  - Covariates and random effects (ETAs)



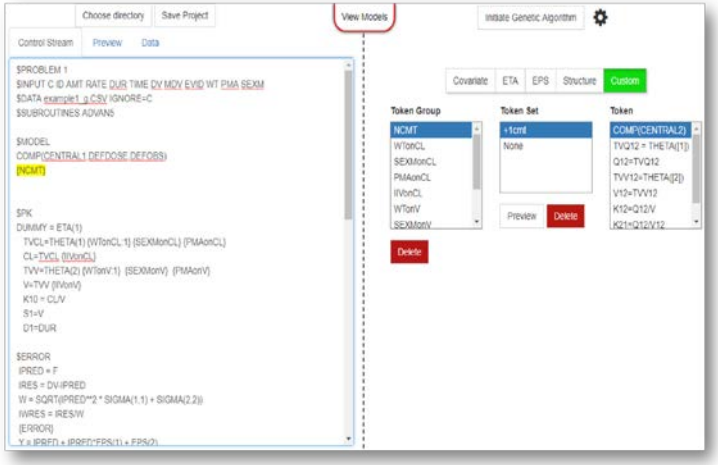
Plots Parameters

Covariate Model PsN

R	Model	Estimate	Pval.x
Linear	WTonCL	0.051	0.0014
Exp	WTonCL	0.188	0.00107
Power	WTonCL	0.353	0.00118
Linear	PMAonCL	0.013	0.00338
Exp	PMAonCL	0.046	0.00203
Power	PMAonCL	1.58	0.00178
Linear	WTonV	0.668	0.00821
Exp	WTonV	0.258	0.00665
Power	WTonV	0.501	0.00433
Linear	PMAonV	0.12	0.099
Exp	PMAonV	0.055	0.0388
Power	PMAonV	1.903	0.0315
Linear	WTonETA1	0	NaN
Linear	PMAonETA1	0	NaN
Linear	WTonETA2	0.188	0.00107
Linear	PMAonETA2	0.046	0.00203

Software Application

- Free and open source
- Cross platform
- Extendable
- Features:
  - Implements robust search for globally optimal model solution
  - Organizes and displays models in tabular interface, allowing user to sort, filter, edit, create, and delete models seamlessly
  - Displays run results, parameter estimates, and precision
  - Integrated with Xpose and PsN
  - Linear regression models between covariates and post-hocs

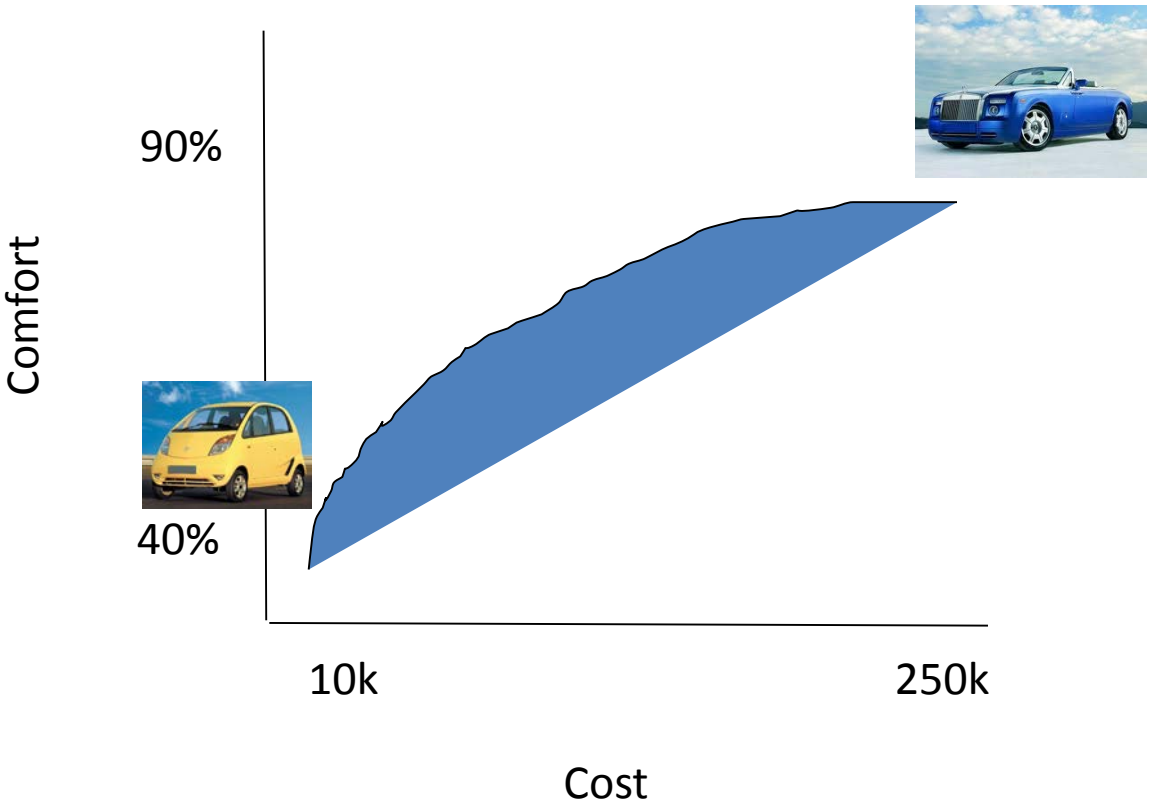


Multi-objective optimization

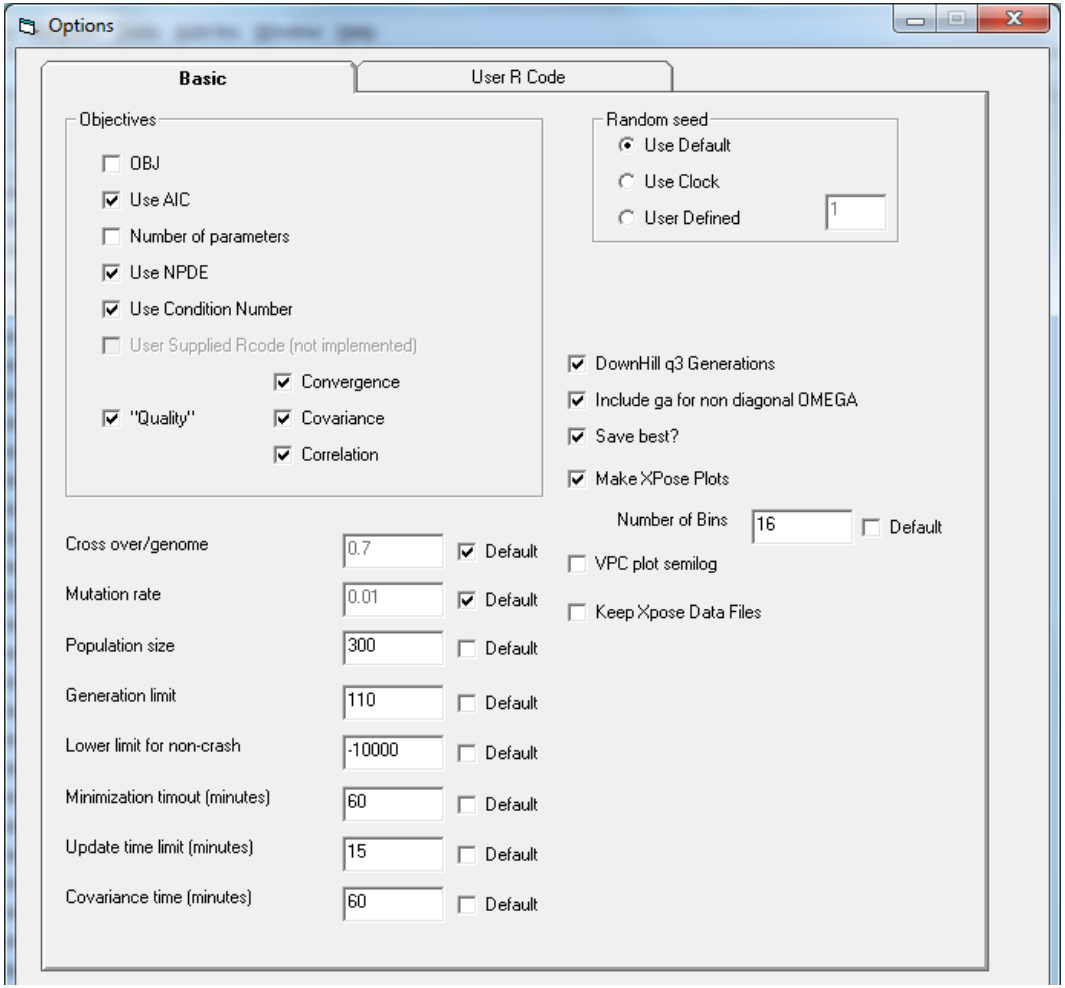
- Single objective GA – composite function.
- Arbitrary (how much is each component of a model worth in terms of a penalty on fitness)
- Decisionmakers
  - preferred not to be presented with a single “best” option
  - wanted subjective elements captured
- NSGA-II (non-dominated sorted genetic algorithm)
  - Possible dimensions
    - Number of parameters
    - -2 Log Likelihood
    - NPDE P-value
    - “Quality” score
    - AIC
  - Uses elitism, diversity and mutation operators
  - Sorts on “non-dominated” solutions
    - Solutions that are at least as good as all others but better in one dimension

Multi-objective optimization

- Optimize over many criteria



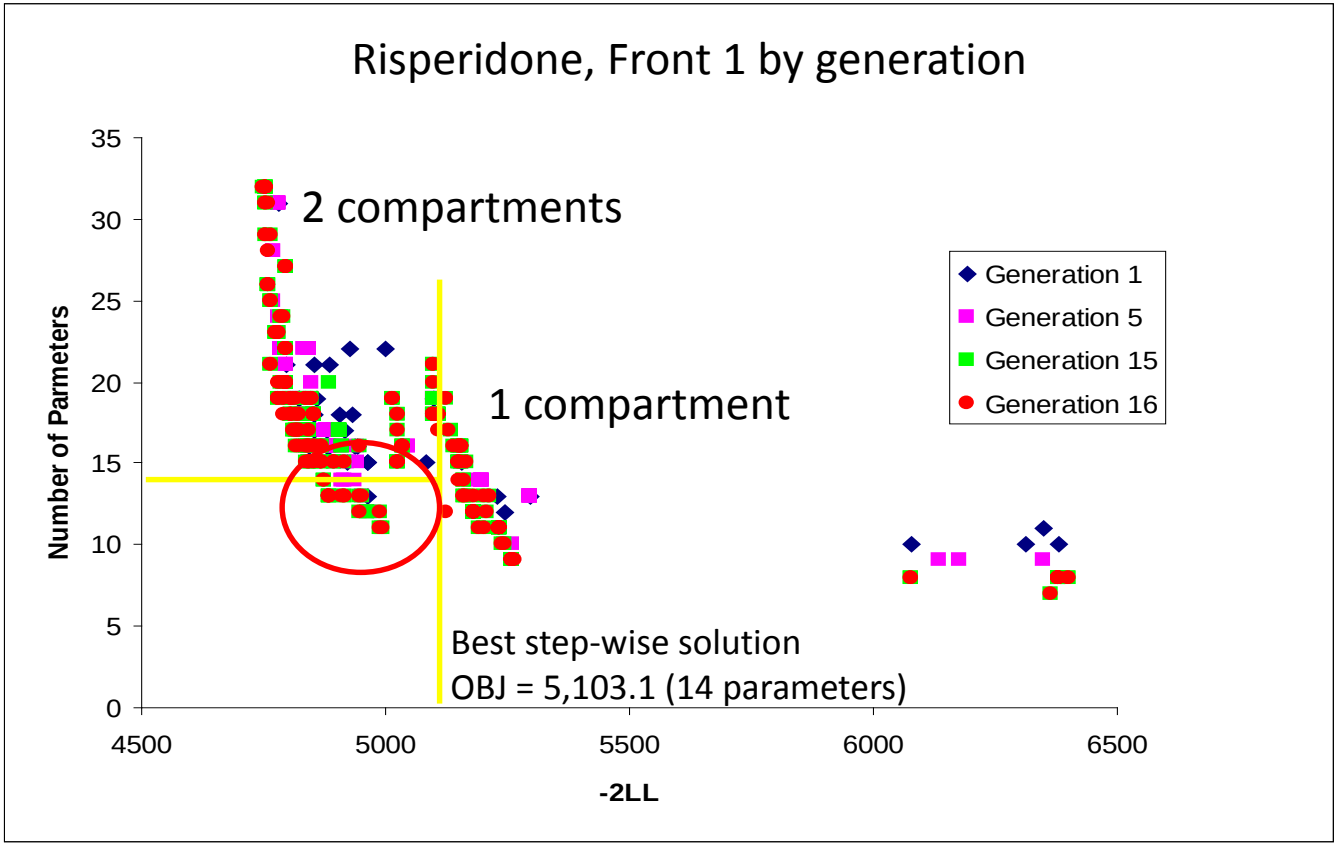
MOGA Options



- There are fewer MOGA options, since penalties do not need to be specified.

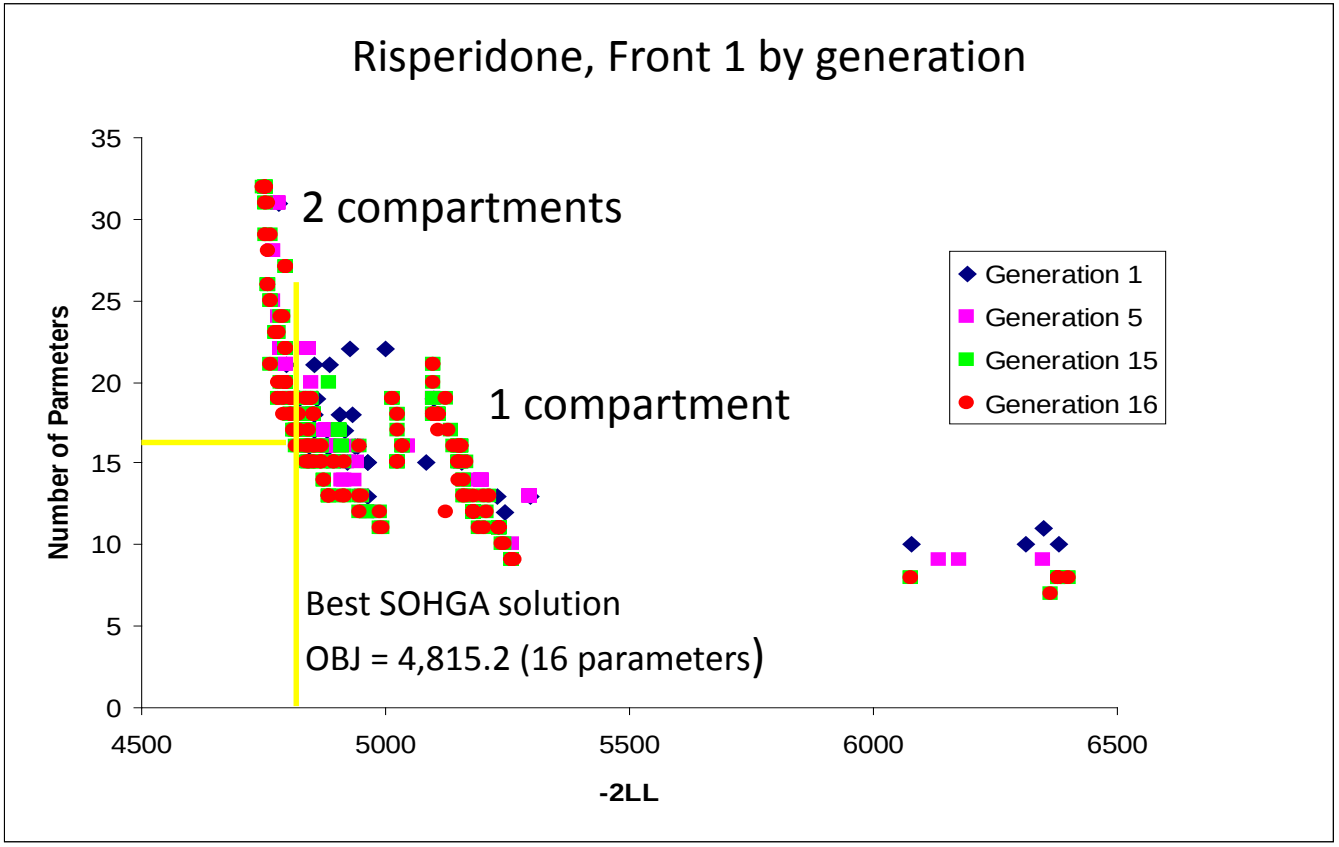
Multi-objective genetic algorithm

- Front in -2xLL vs. # parameters space

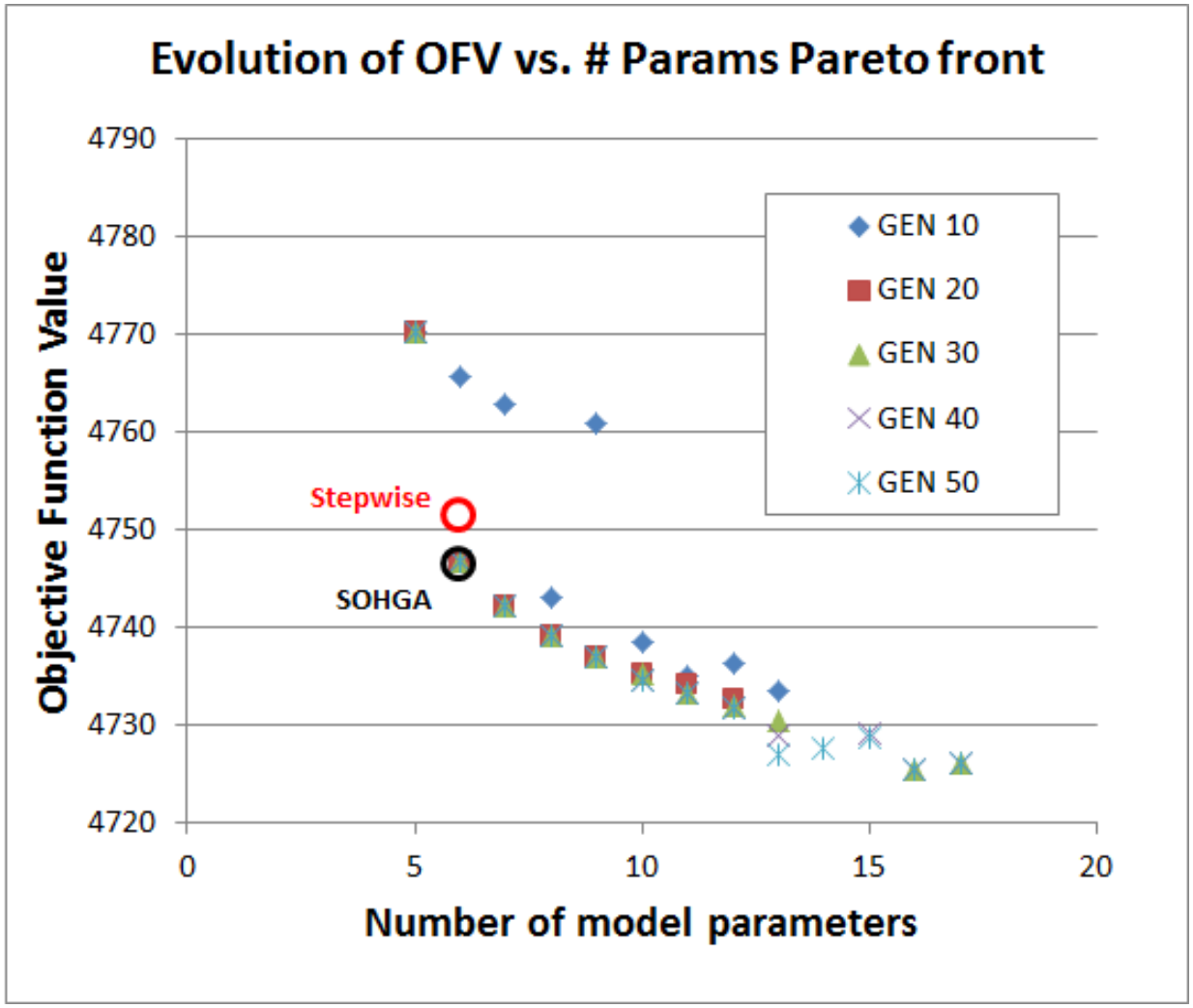


Multi-objective genetic algorithm

- Front in -2xLL vs. # parameters space



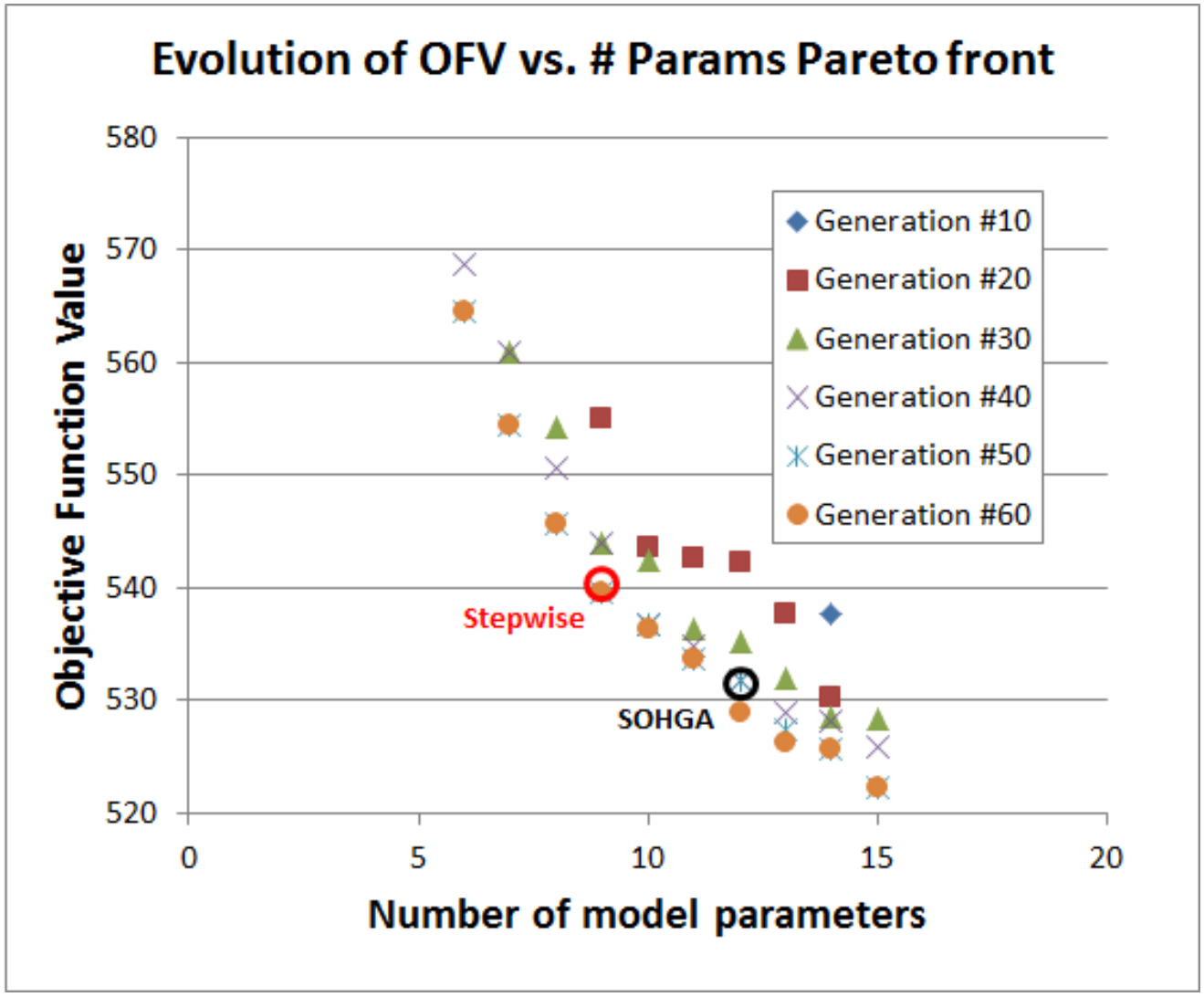
Ziprasidone MOGA



Ziprasidone MOGA

Number of Parameters	5	6	7	8	9	10	11	12	13	14	15	16	17
Objective Function	4770.3	4746.7	4742.2	4739.1	4736.9	4734.7	4733.4	4731.8	4727	4727.6	4728.8	4725.5	4726.1
		stepwise - 4751.2											
		SOHGA - 4746.7											
IIV - CL	X	X	X	X	X	X	X	X	X	X	X	X	X
IIV - V		X	X	X	X	X	X	X	X	X	X	X	X
IIV - Ka									X	X		X	
ADDITIVE													
PROPORTIONAL	X	X	X	X	X	X	X	X	X	X	X	X	X
COMBINED													
CL - AII			X					X	X	X	X	X	X
CL - DI				X	X	X	X	X	X	X	X	X	X
CL - CI				X	X	X	X	X	X	X	X	X	X
CL - AGE					X	X	X			X	X		XX
CL - WT						X	X	X	X	X	X	X	XX
CL - SM2							X	X	X	X	X	X	X
CL - CII										X	X		X
CL - SEX											X	X	
CL - AI											X		
V - WT												X	
V - SEX												X	X

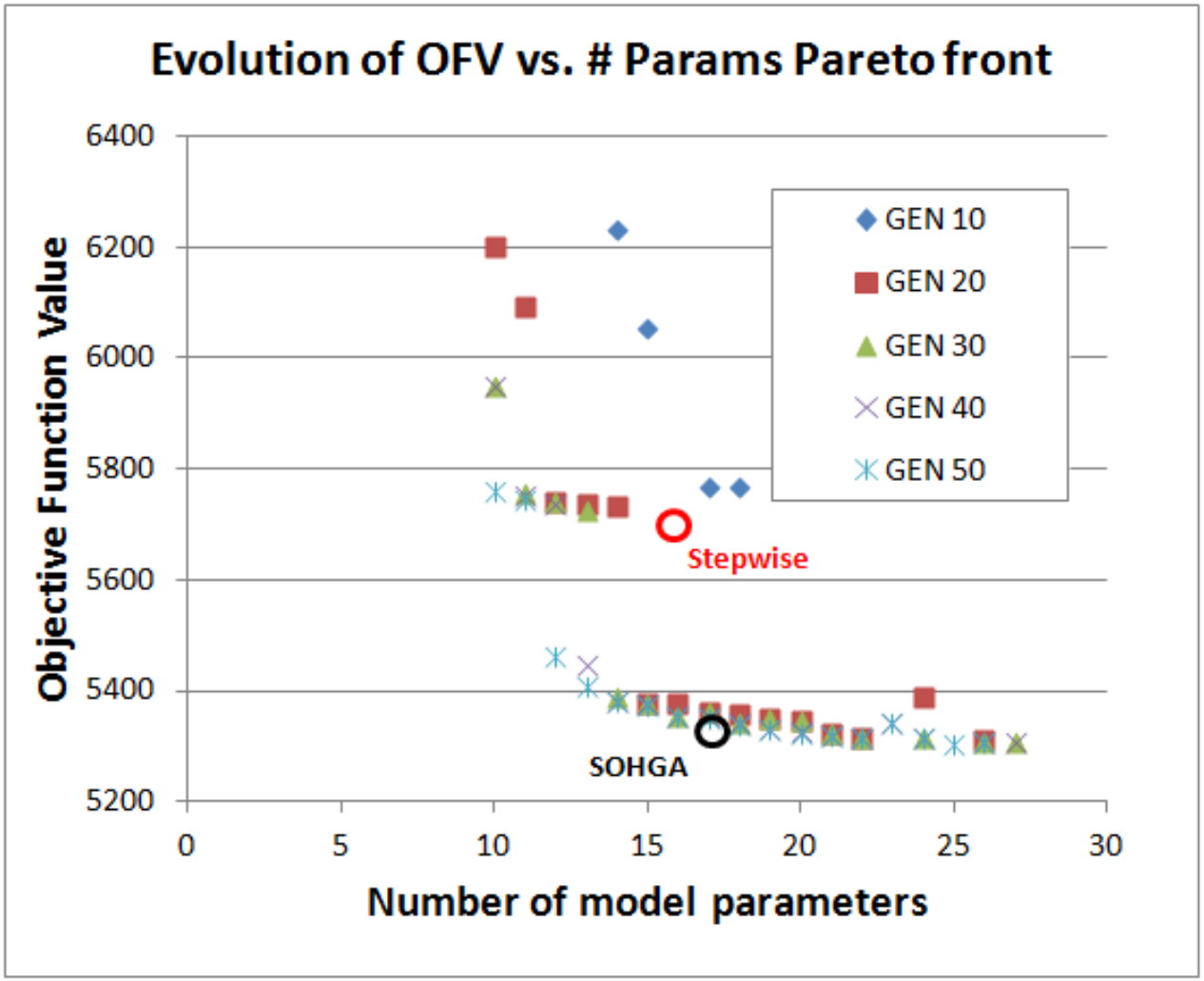
Perphenazine MOGA



Perphenazine MOGA

Number of Parameters	6	7	8	9	10	11	12	13	14	15
Objective Function	564.6	554.4	545.6	539.6	536.3	533.7	529.0	526.3	525.7	522.3
				stepwise - 540.7			**SOHGA - 531.9**			
IIV - CL	X	X	X	X	X	X	X	X	X	X
IIV - V										
IIV - Ka										
ADDITIVE										
PROPORTIONAL	X	X	X	X	X	X	X	X	X	X
COMBINED										
SMK - CL	X	X		X	X	X	X	X	X	X
CIG - CL		X	X	X	X	X	X	X	X	X
RACE - CL			X			X		X	X	X
FLUX - CL				X		X		X	X	X
SEX - CL										
PARX - CL					X	X	X	X		X
WGT - CL									X	X
SMK - V			X	X	X	X	X	X	X	X
AGE - V					X		X		X	X
FLUX - V										
CIG - V							X	X	X	X
RACE - V							X	X	X	X
SEX - V										

Citalopram (IV) MOGA



Citalopram (IV) MOGA

Number of Parameters	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
Objective Function	5759.6	5745.7	5463.8	5409.7	5380.4	5374.5	5353.1	5350.6	5339.7	5329.9	5323.5	5318	5314.5	5343.5	5314.1	5303.3	5308.9
							stepwise - 5695.5	**SOHGA - 5335.6**									
IIV - CL															X		
IIV - V1			X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
IIV - Q	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
IIV - V2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
COVARIANCE				X	X	X	X	X	X	X	X	X	X	X	X	X	X
ADDITIVE																	
PROPORTIONAL	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
COMBINED																	
CL - BMI	X	X	X	X	X		X	X			X	X		X	X	X	X
CL - WT		X	X			X			X			X	X	X	X	X	X
CL - SEX							X	X						X			X
CL - FAT										X	X	X	X	X	X		X
CL - AGE												X			X		
V1 - BMI						X		X	X	X	X	X	X	X	X	X	X
V1 - SEX							X	X	X	X	X	X	X		X	X	X
V1 - WT									X				X			X	X
V1 - FAT																	X
Q - SEX					X	X			X	X			X	X	X	X	X
Q - WT									X	X	X	X	X				X
Q - FAT														X	X	X	X
Q - AGE														X		X	
Q - FFM															X	X	X
V2 - FFM	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	
V2 - SEX							X	X		X	X	X	X	X	X	X	X
V2 - WT										X	X	X	X		X	X	X
V2 - FAT																X	X
V2 - AGE											X		X	X		X	
V2 - BMI														X	X	X	X

The Control Stream Template

- Resembles NONMEM control stream
- With additional {placeholder} text strings
  - Termed token groups

```
$PK
TVCL = THETA(1) {WTonCL}
CL = TVCL * EXP(ETA(1))


$THETA
(0, 10) ; CL
{WTonCL}
```

The Control Stream Template

```
$PK
TVCL = THETA(1) {WTonCL}
CL = TVCL * EXP(ETA(1))

$THETA
(0, 10) ; CL
{WTonCL}
```

Placeholders are replaced with text strings to produce syntactically correct control streams

Token Group	Token Set	Tokens
WTonCL	Linear	+THETA(n)*WT (0.001); WTonCL
WTonCL	Exponential	*exp(THETA(n)*WT) (0.001); WTonCL

Summary

- Implemented genetic algorithm method to more completely search the model space
- Single-objective, hybrid genetic algorithm
  - Arbitrary fitness function
  - Post-hoc assessment of predictive performance

- Multi-objective views additional dimensions
  - Generate non-dominated pareto front to evaluate trade-offs across models for given characteristics