The Use of a SOM to Generate QSAR Sets

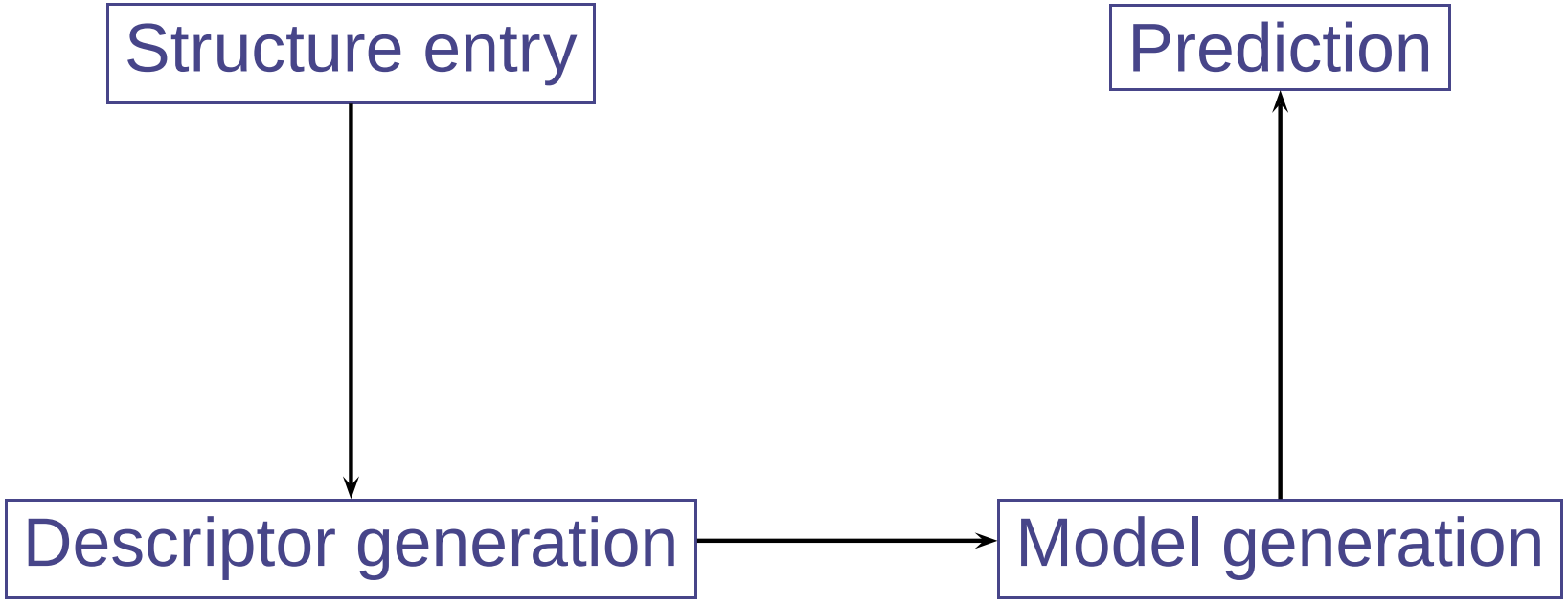
Rajarshi Guha

Pennsylvania State University

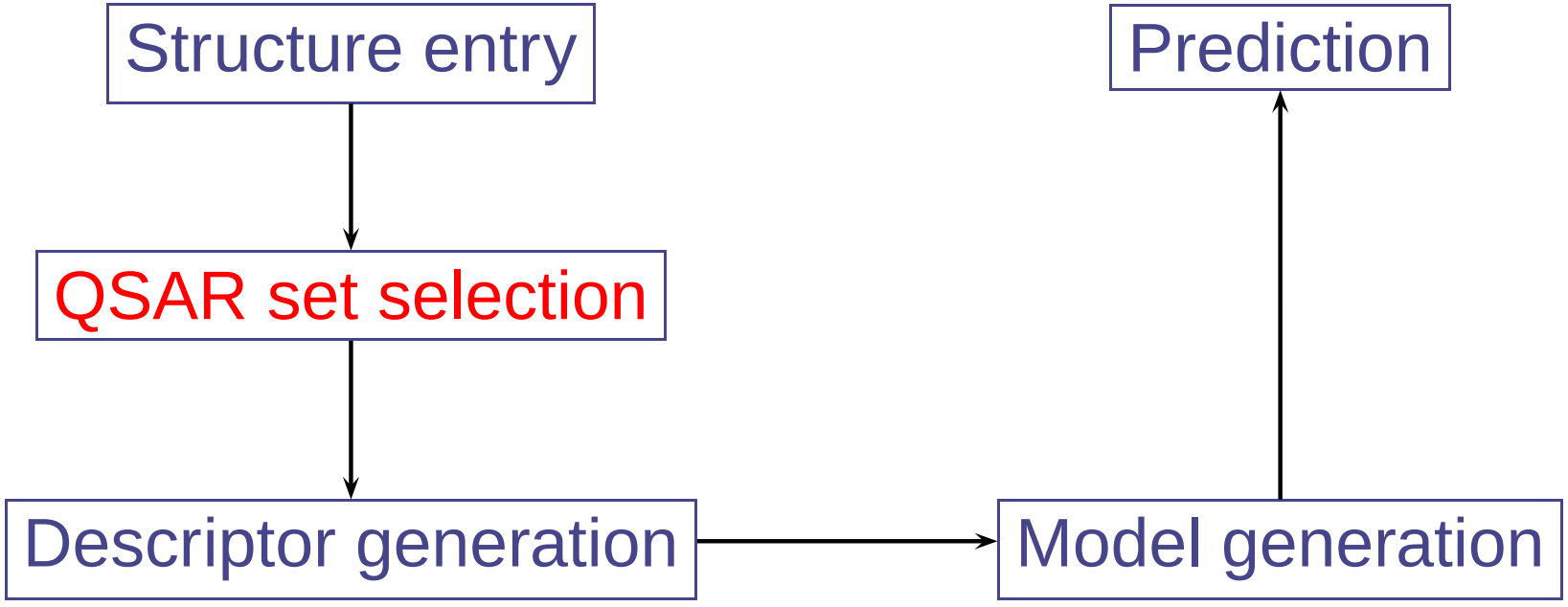
Introduction

- What are QSAR sets and what role do they play?
- Set selection methods
- The construction and working of a SOM
- The overall procedure
- Application and results

The Steps of QSAR Modelling



The Steps of QSAR Modelling



What are QSAR Sets?

- 3 sets of molecules are required during QSAR modelling
  - Training set
    - Used during training of models
    - Best models are based on TSET statistics
  - Cross validation set
    - Used during the training of neural network models to prevent over training
  - Prediction set
    - Never used during training
    - Used to validate the final models

QSAR Set Selection Methods

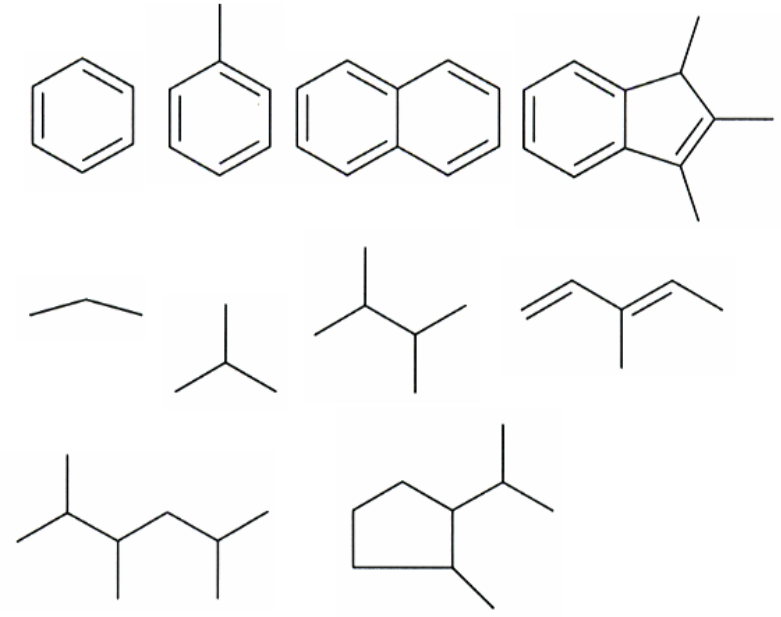
- Random selection
- Activity binning
- Sphere exclusion
- Self Organizing Maps

Why Does Set Selection Matter?

- QSAR models are trained and validated using different sets
- A QSAR model tries to capture features of an entire dataset and uses those features to make predictions
- Therefore, training and validation sets should be representative of the entire dataset

Why Does Set Selection Matter?

- An example - a dataset of 10 compounds:



- For the model to be able to make good predictions, both aromatic and non-aromatic compounds must be represented in the TSET

How Does an SOM Help?

- The goal is to create QSAR sets which are *representative* of the overall dataset
- Similar groups of molecules should be represented in the QSAR sets according to their proportions in the dataset
- Therefore we need to detect *similar groups* of molecules
- Enter the SOM

What is an SOM

- A SOM is an unsupervised neural network
- Some features
  - Transforms non-linear multidimensional datasets to 2D grids
  - Maintains the topology of the dataset
  - Training occurs via competition between the neurons
  - Does not require knowledge of the dependent variable
  - Can be used for detecting similarity and degrees of similarity
- It is assumed that the input patterns fall into sufficiently distinct groupings

How Does it Work?

- The steps involved in using a SOM to make QSAR sets are:
  - Construct the map
  - Train the map
    - Allow neurons to compete
    - Modify winning neurons
  - Use the map to detect classes
  - Use the classes to generate QSAR sets

How Does it Work?

- ***Construct the map***
  - Square grid which wraps around at the edges (toroidal)
  - Each unit on the grid is a neuron and is represented as a vector of weights
  - Length of the vector equals the number of descriptors
  - Neurons are initialized with random weights

How Does it Work?

- ***Train the map***
  - Training examples are presented to all the units
  - The units compete for selection
  - The selected neuron and surrounding neighbors get modified
  - Multiple iterations result in groups of units becoming sensitized to features of the input vector

Some Details of the Training Process

- The winner is the neuron which is closest to the training vector in terms of Euclidean distance

$$d_{sj} = \sqrt{\sum_{i=1}^{m}(s_{si} - w_{ji})}$$

- The winning neuron and neighbors are modified
- Degree of modification reduces with each training iteration
- Once all the training vectors have been presented, repeat the cycle.

Some Details of the Training Process

- ***Modification of Neurons***
  - The winning neuron and *topologically* close neurons are modified
  - Leads to smoothing which leads to global ordering
  - The general form of modification is

$$m_i(t+1) = m_i(t) + h_{ci}(t)[x(t) - m_i(t)]$$

  - $h_{ci}(t)$ is termed as the neighborhood function
  - As $t \to \infty, h_{ci}(t) \to 0$

Some Details of the Training Process

- A Gaussian neighborhood function was used in this study

$$h_{ci}(t) = \alpha(t) \exp\left(-\frac{\|r_c - r_i\|^2}{2\sigma^2(t)}\right)$$

- $\alpha(t)$ is the learning factor and controls *how much* a neuron is modified
  - $\alpha(t)$ decreases with training iterations
  - This study used a constant decrement

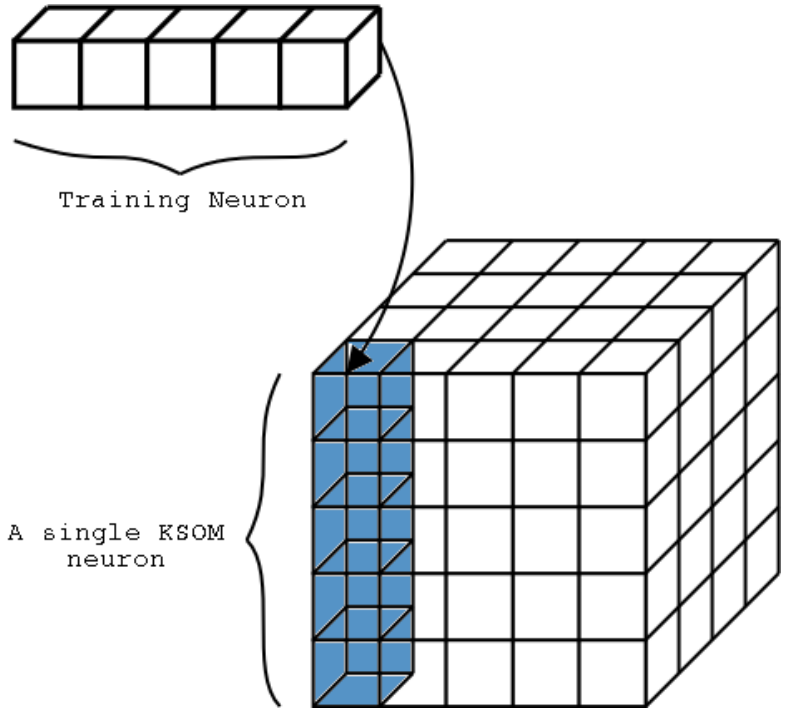
$$\alpha(t+1) = \alpha(t) - 0.01$$

  - When $\alpha(t) = 0$ training stops

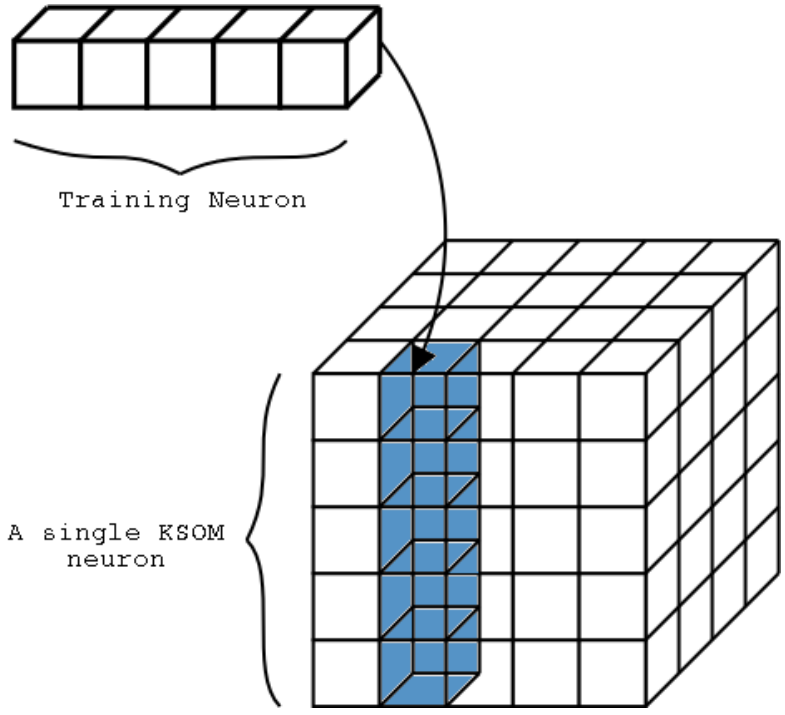
How Does it Work?

- ***Use the map to detect classes***
  - Assign an arbitrary class to the first neuron
  - For each neuron calculate distances to each nearest neighbor
  - If the distance to a neighbor is less than a user-specified threshold, then the neighbor is in the same class as the grid point
  - After class asignments of the grid neurons, use these assignments to divide the dataset into two classes

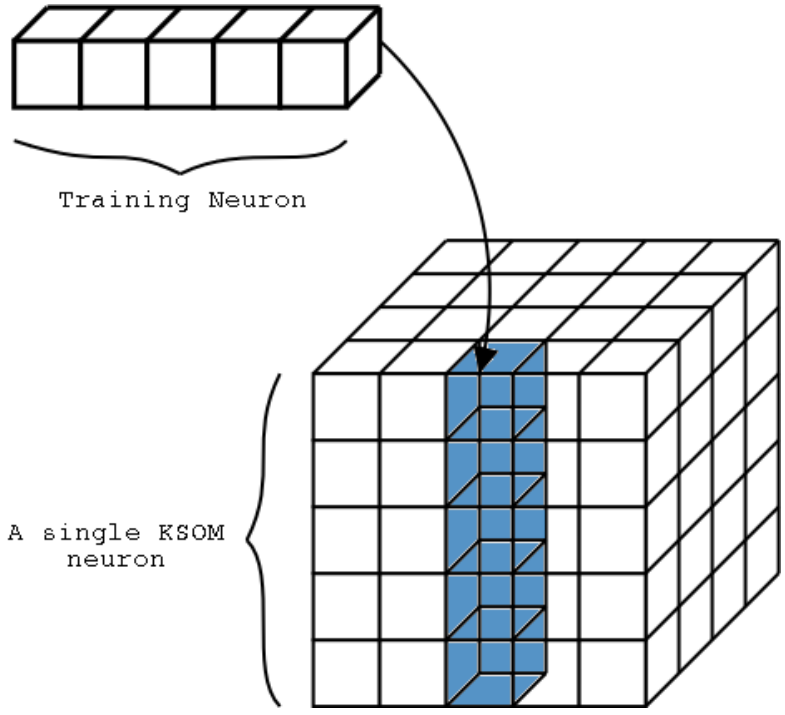
Training



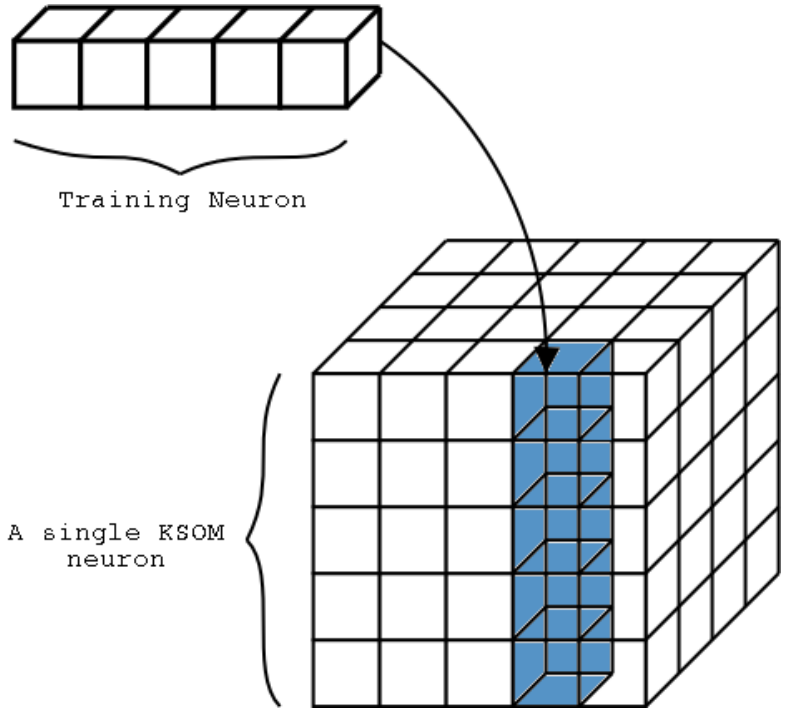
Training



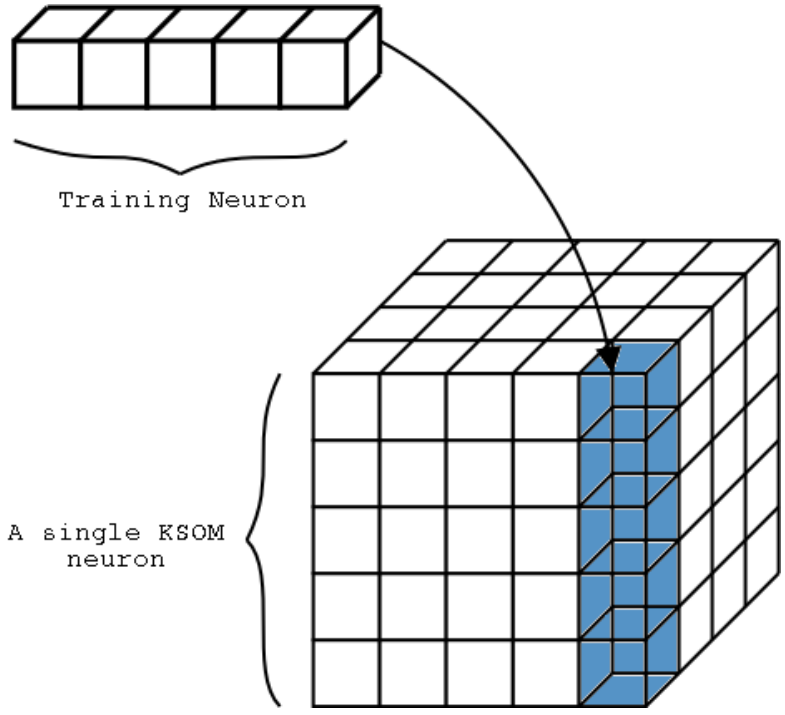
Training



Training



Training

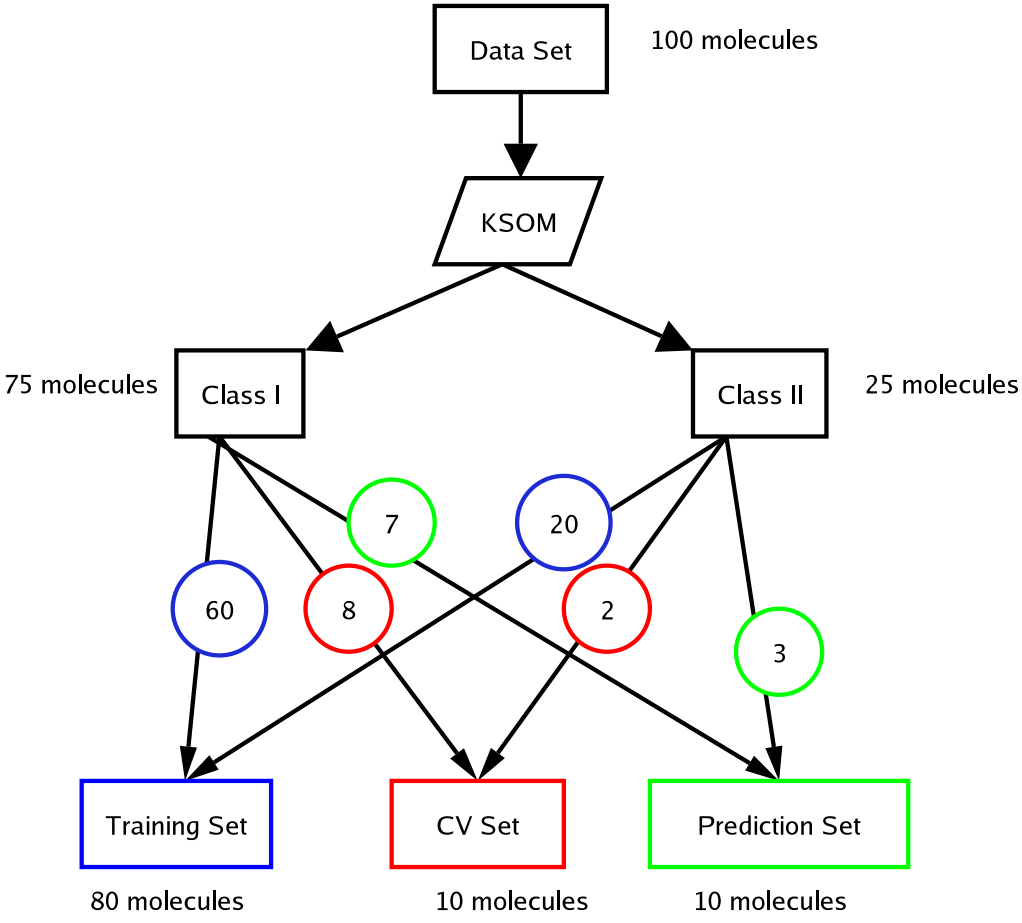


How Does it Work?

- ***Selecting a Threshold Value***
  - Run the SOM once to get an idea about the distances between each neuron
  - Next run the SOM multiple times setting the threshold value at 10% to 90% of the maximum distance observed in the first step
  - Look for class breakups which are approximately 80 - 20

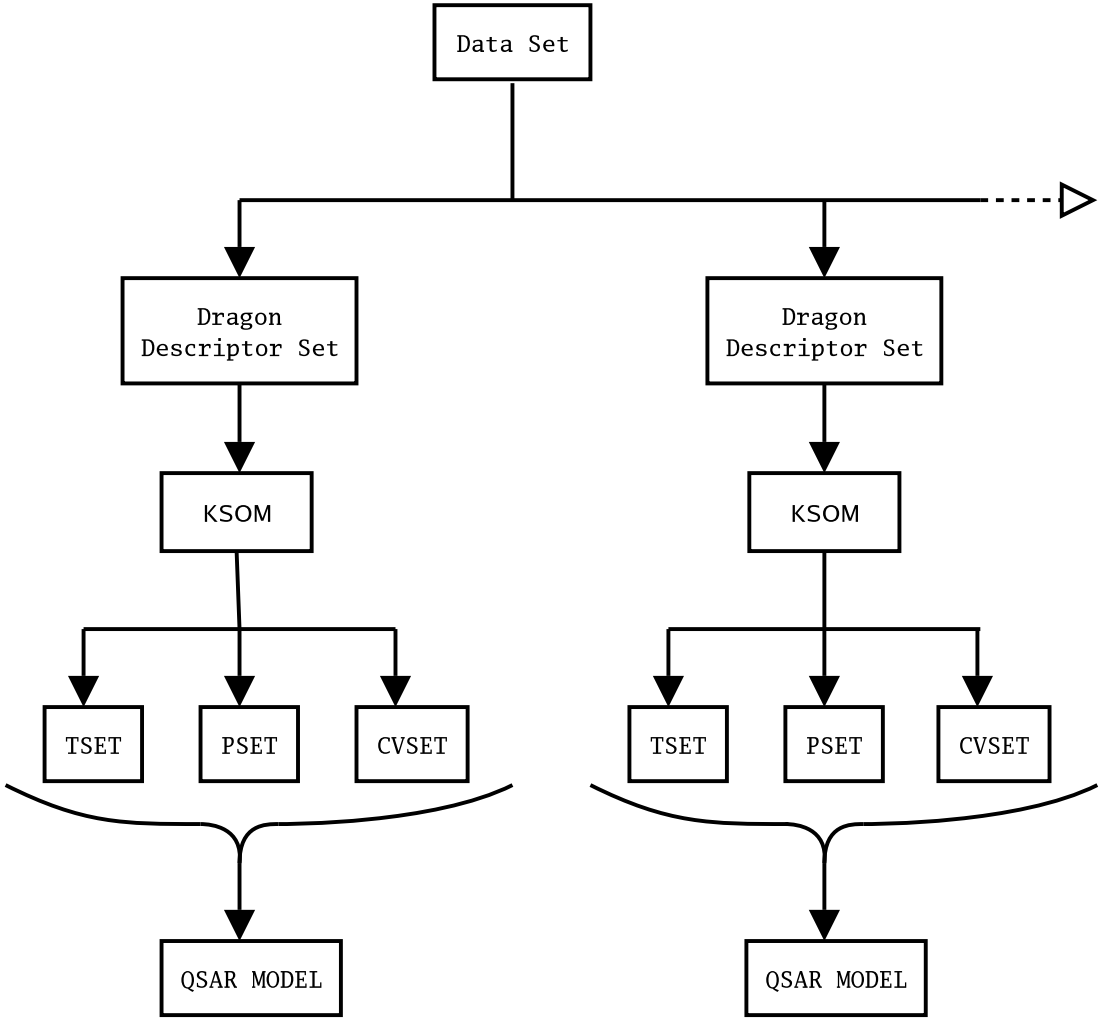
How Does it Work?

- ***Use the SOM classes to create QSAR sets***



How Does it Work?

- ***The Overall Procedeure***



Testing the Technique

- This technique was tested with a dataset DHFR inhibitors previously studied by Mattioni et al
- 333 molecules in the dataset
- 6 sets of Dragon descriptors were used to create 6 QSAR sets
- QSAR models were generated using the ADAPT methodology for each QSAR set generated

J. Mol. Graph. Model., 2003, **21**, 391-419

Results

- Most of the models did not do better than the published model
- However, 2 models did occur which were half the size of the published model
- The 2 best models were more *consistent* than the one published

Results (CNN Models)

Dragon Descriptor Set	CNN Architecture	RMS Error		
		Training	Cross Validation	Prediction
BCUT & 2D Autocorrelation	5-3-1	0.63	0.68	0.79
BCUT & Galvez Indices	5-3-1	0.62	0.62	0.71
GETAWAY	5-2-1	0.73	0.73	0.65
MoRSE & 2D Autocorrelation	5-3-1	0.63	0.63	0.68
MoRSE & GETAWAY	9-5-1	0.49	0.59	0.76
MoRSE & WHIM	6-5-1	0.60	0.61	0.65
Published	10-6-1	0.45	0.49	0.66

Results (R^2 Values)

	Training	Cross Validation	Prediction
MoRSE & 2D Autocorrelation	0.68	0.60	0.64
MoRSE & WHIM	0.75	0.78	0.64
Published	0.83	0.78	0.64

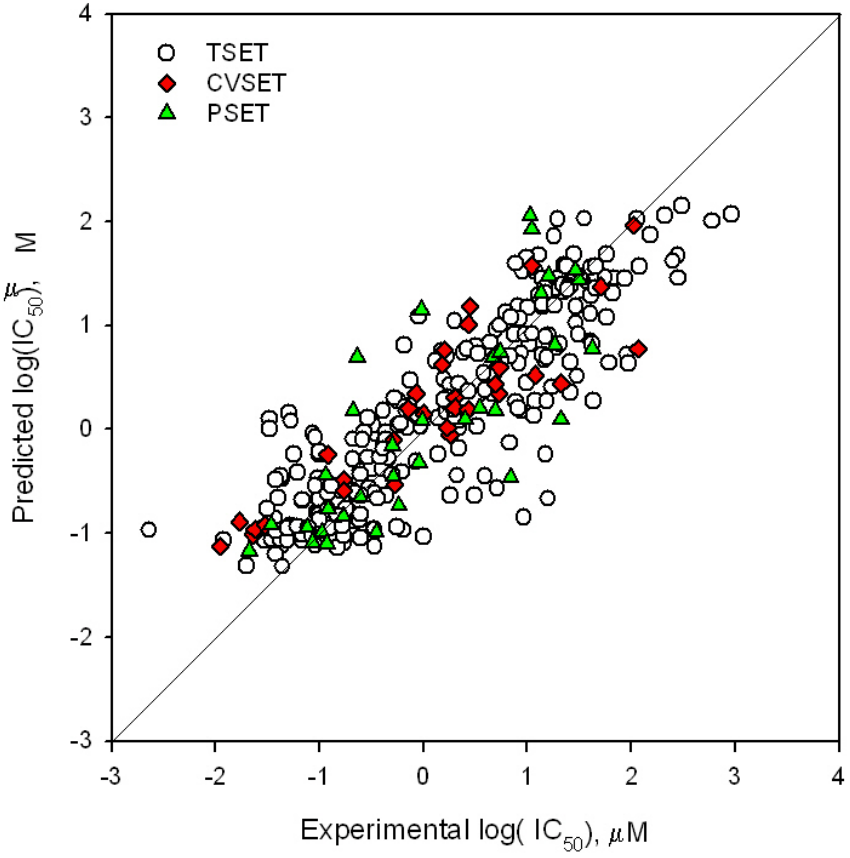
Result (Random Sets vs SOM Sets)

	Random Sets			MoRSE - WHIM Sets		
	Mean RMSE	Std. Dev.	R^2	Mean RMSE	Std. Dev.	R^2
TSET	0.57	0.02	0.75	0.58	0.005	0.74
CVSET	0.59	0.03	0.73	0.57	0.0010	0.76
PSET	0.80	0.13	0.56	0.63	0.020	0.63

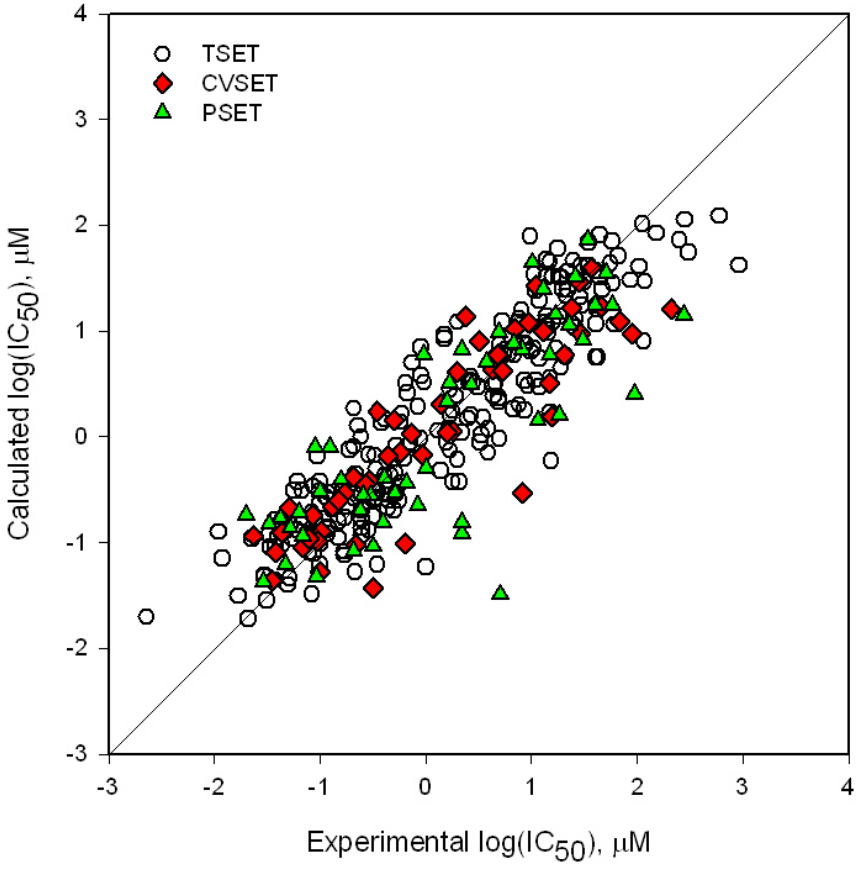
Results (Scrambled Dependant Variable)

	Scrambled		MoRSE - WHIM	
	RMSE	R^2	RMSE	R^2
TSET	0.74	0.62	0.60	0.75
CVSET	0.85	0.48	0.61	0.78
PSET	0.90	0.39	0.65	0.64

Results (Type III CNN Model Plot)



Best SOM Based Model



Original Results

Future Work

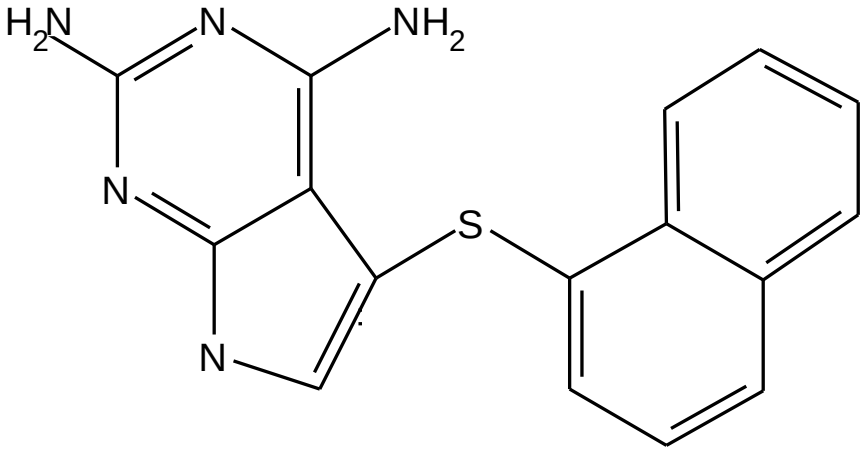
- Try and reduce the arbitrariness at various stages in the SOM
  - One possibility is to use PC's of all Dragon descriptors
  - Use a majority rules technique to decide on class memberships in a SOM
- The SOM can also be used to decide whether an unknown compound can be reliably analyzed using a QSAR model

Conclusions

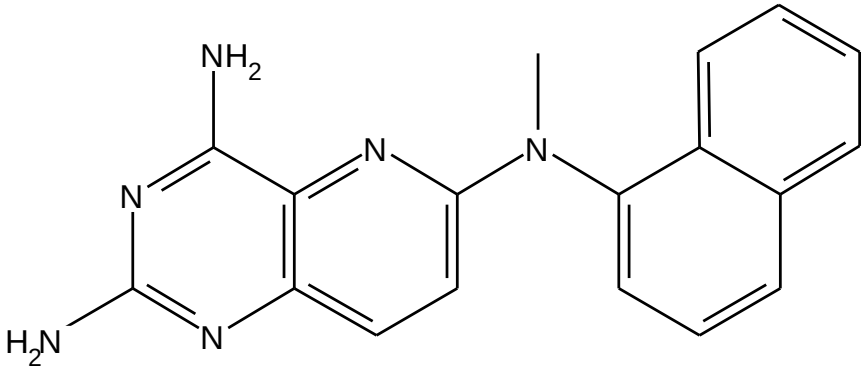
- The SOM appears to be able to generate representative sets
- This is evidenced by
  - more consistent statistics for QSAR sets
  - smaller and simpler QSAR models
- The SOM technique represents a more rational method of designing QSAR sets

Extra Information

Outliers

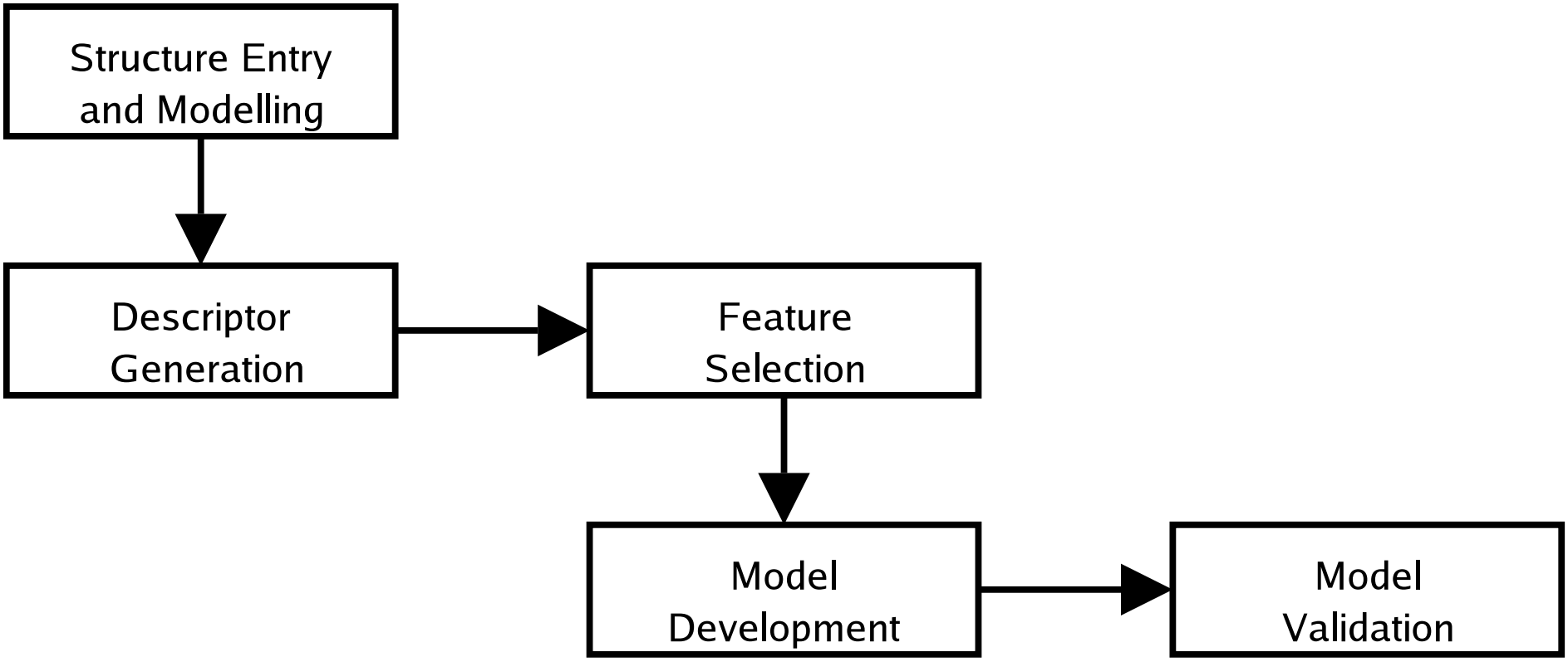


Current Outlier



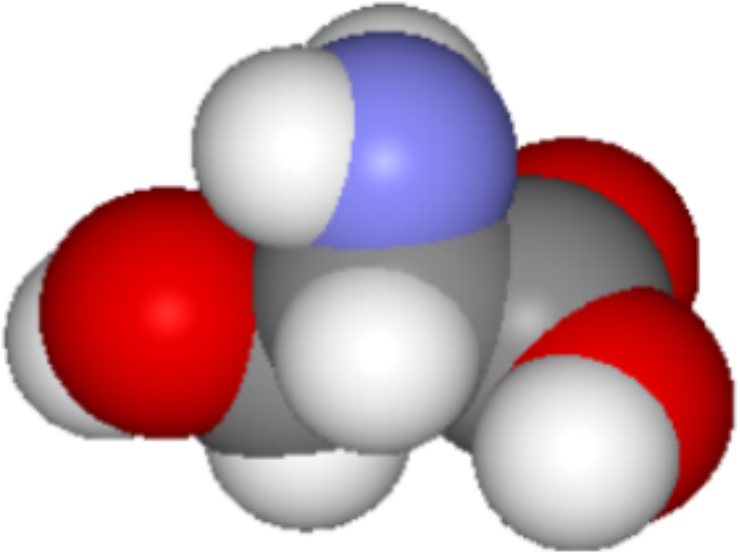
Original Outlier

ADAPT Methodology - Details



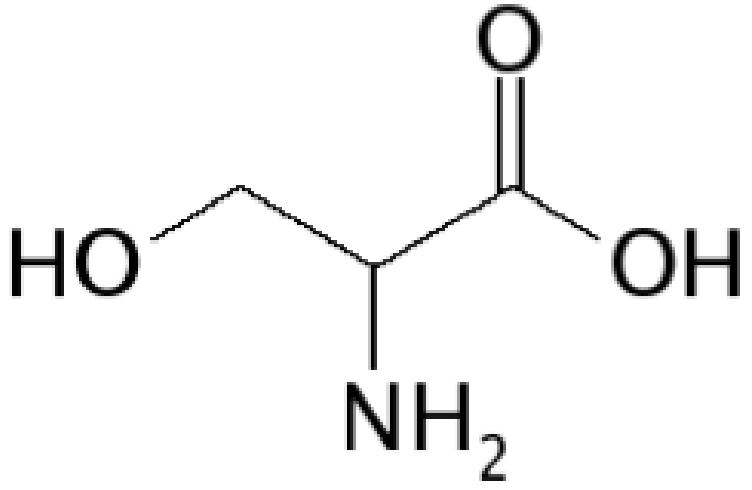
ADAPT Descriptors

- Geometric Descriptors
  - Depends on 3D molecular structures
  - Molecular geometries should be optimized
  - Examples are *moment of inertia* and *surface areas*



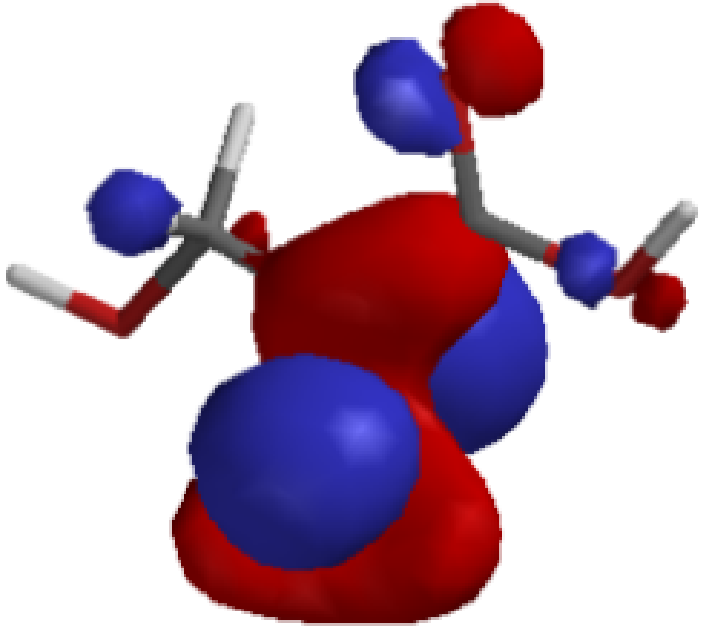
ADAPT Descriptors

- Topological Descriptors
  - Depends only on the 2D molecular skeleton
  - Not always very interpretable
  - Examples are *connectivity indices* and *molecular distance edges*



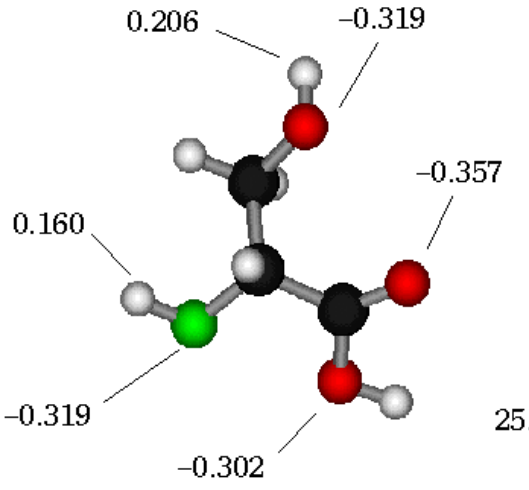
ADAPT Descriptors

- Electronic Descriptors
  - Describe features obtained by QM calculations
  - Examples are *HOMO / LUMO energies* and *partial charges*

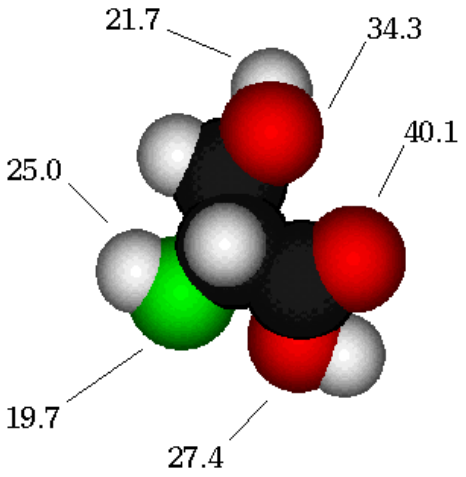


ADAPT Descriptors

- Hybrid Descriptors
  - These are a combination of two or more of the preceding types
  - The CPSA descriptors are well known hybrids describing the propensity of molecules for polar interaction

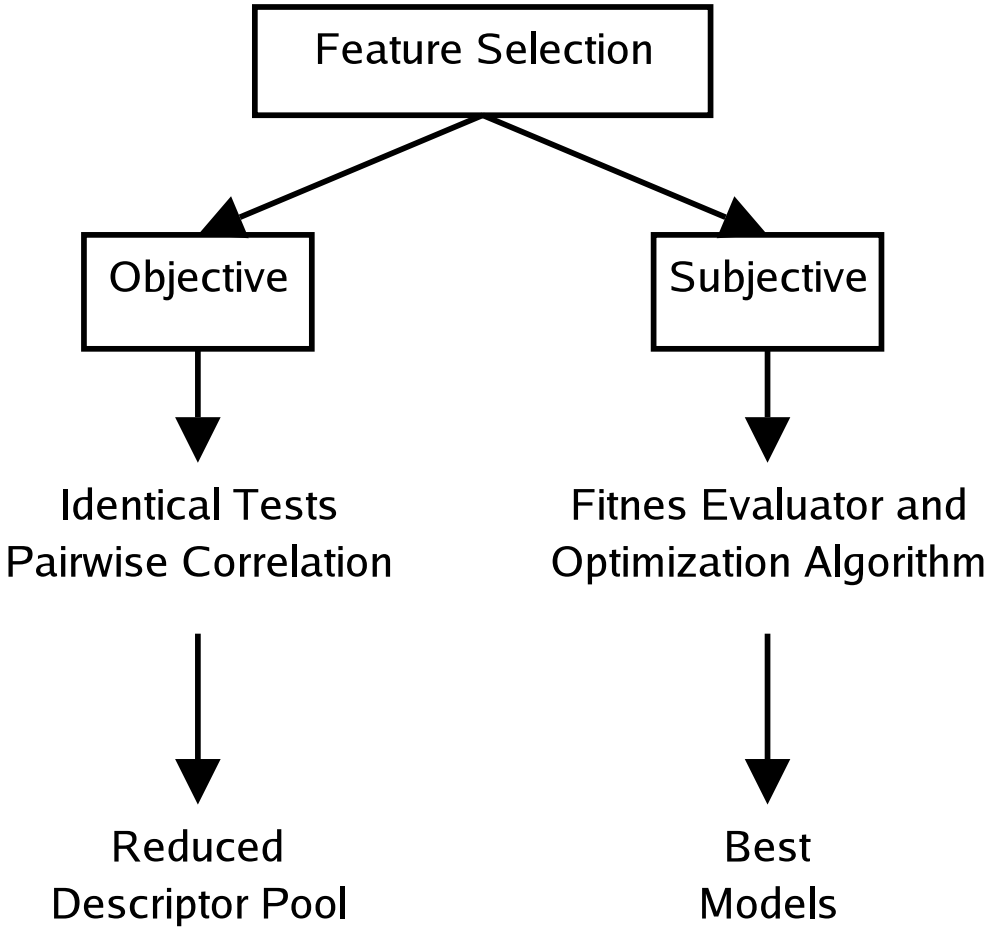


AM1 Partial Charges (e^-)



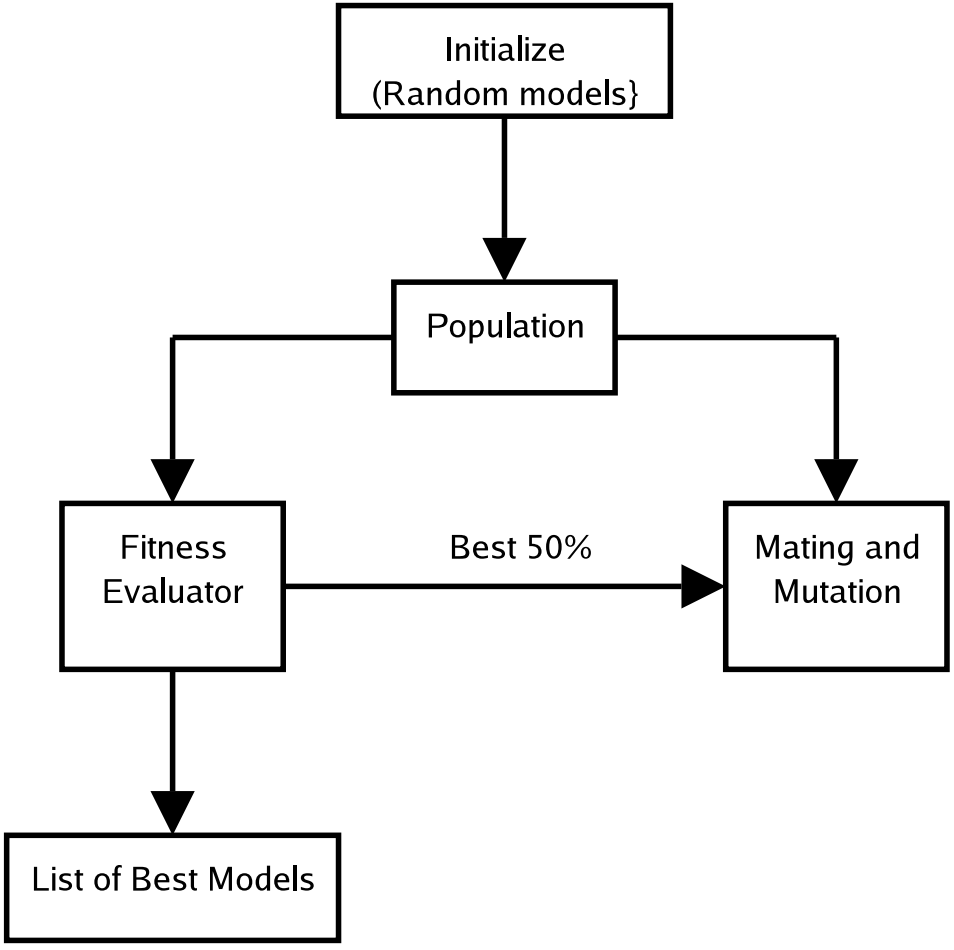
Solvent-Accessible Partial Surface Areas ($Å^2$)

Feature Selection



Optimization Algorithms

- Genetic Algorithm



Multiple Linear Regression

- The biological property is modelled by the equation

$$Property = M_1x_1 + M_2x_2 + \cdots M_nx_n$$

- A GA/SA is used to survey the reduced descriptor pool
- Best model is characterised by fewer descriptors and low RMSE
- Advantages
  - Simple, fast, interpretable
- Disadvantages
  - Biological properties rarely follow linear models

Neural Networks

- The CNN is a fitness evaluator for the GA
- The cost function is defined as

$$\mathrm{COST} = \mathrm{RMS}_{tset} + 0.4|\mathrm{RMS}_{tset} - \mathrm{RMS}_{cvset}|$$

- Number of hidden layer neurons are determined empirically
- Advantages
  - very accurate, models non-linear relationships
- Disadvantages
  - *black box*, low interpretability, computationally expensive

The Anatomy of a Neural Network