# Extending Validation and Providing Interpretability for QSAR Models

Rajarshi Guha

Department of Chemistry
Pennsylvania State University

Jet Propulsion Laboratory, September 8, 2005

# Outline

# Outline

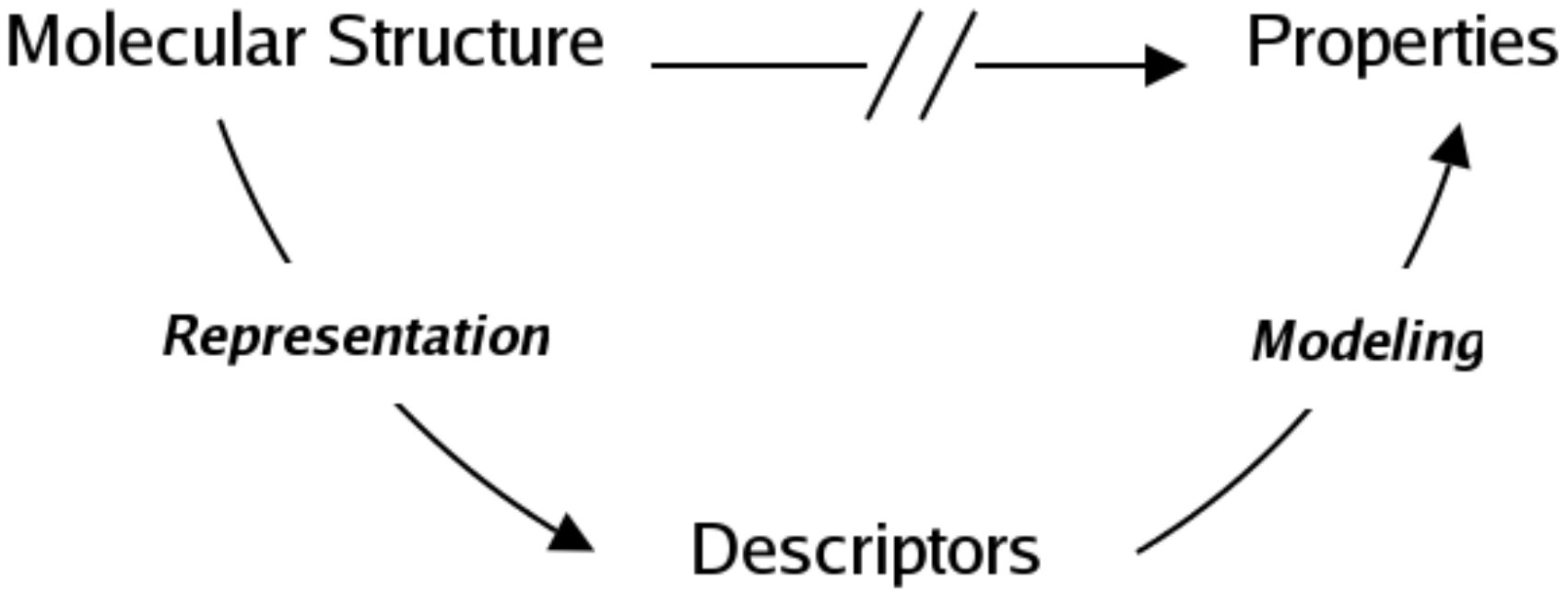
# What is the Aim of a QSAR / QSPR Model?

- Predict properties of molecules or classifiy molecules based on structural features
- Properties can include
  - Physical properties like *boiling point* or *aqueous solubility*
  - Biological activities like *carcinogenicity*, $LD_{50}$ or *drug potency*
- QSAR modeling can be considered to be an application of data mining

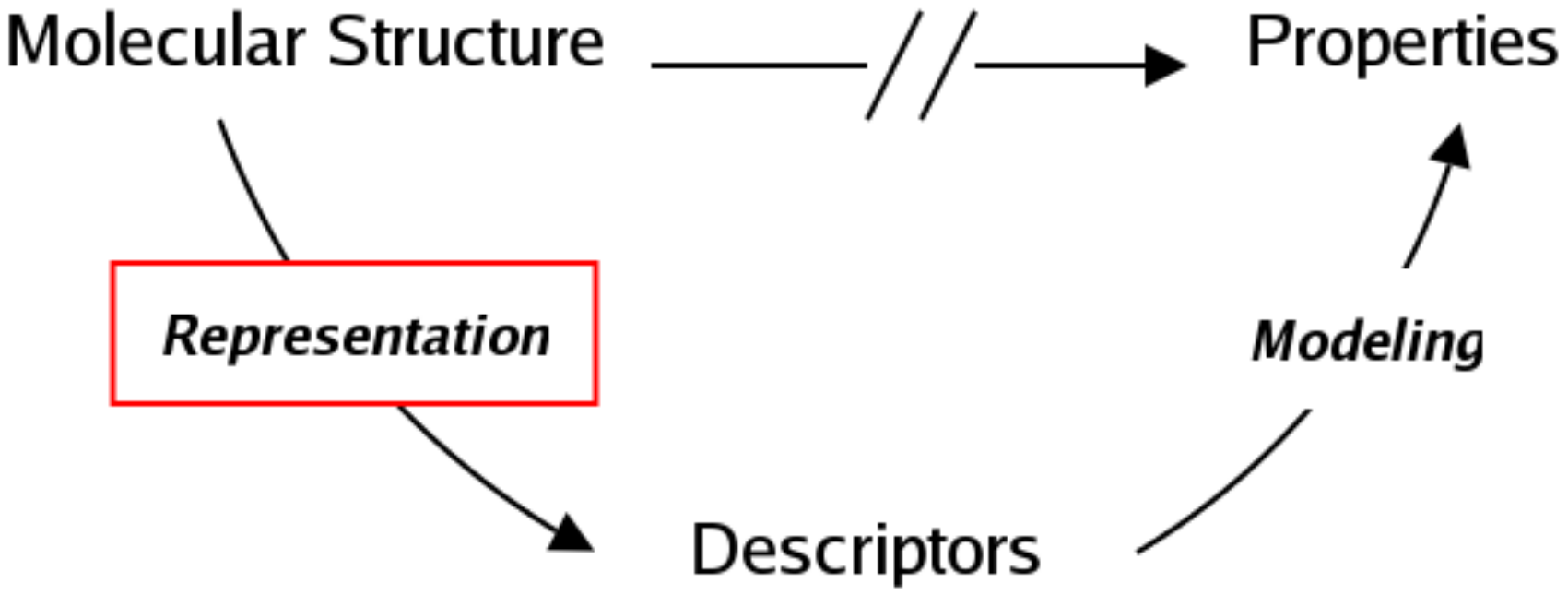
# Why Develop QSAR Models?

- Compound screening, especially for virtual libraries
- ADME/Tox modeling - *fail early, fail cheap* principle
- Can be used to focus on specific compounds
- A model can provide insight into mechanism or mode of action

# The QSAR Pipeline



# The QSAR Pipeline



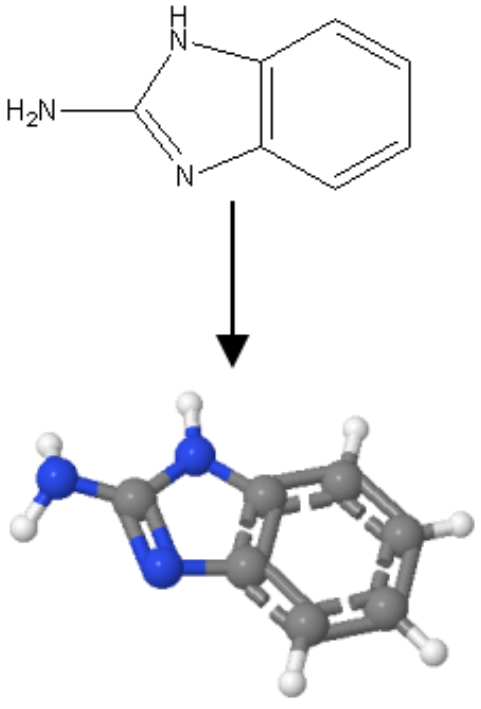
# Structure Representation

## Data Entry

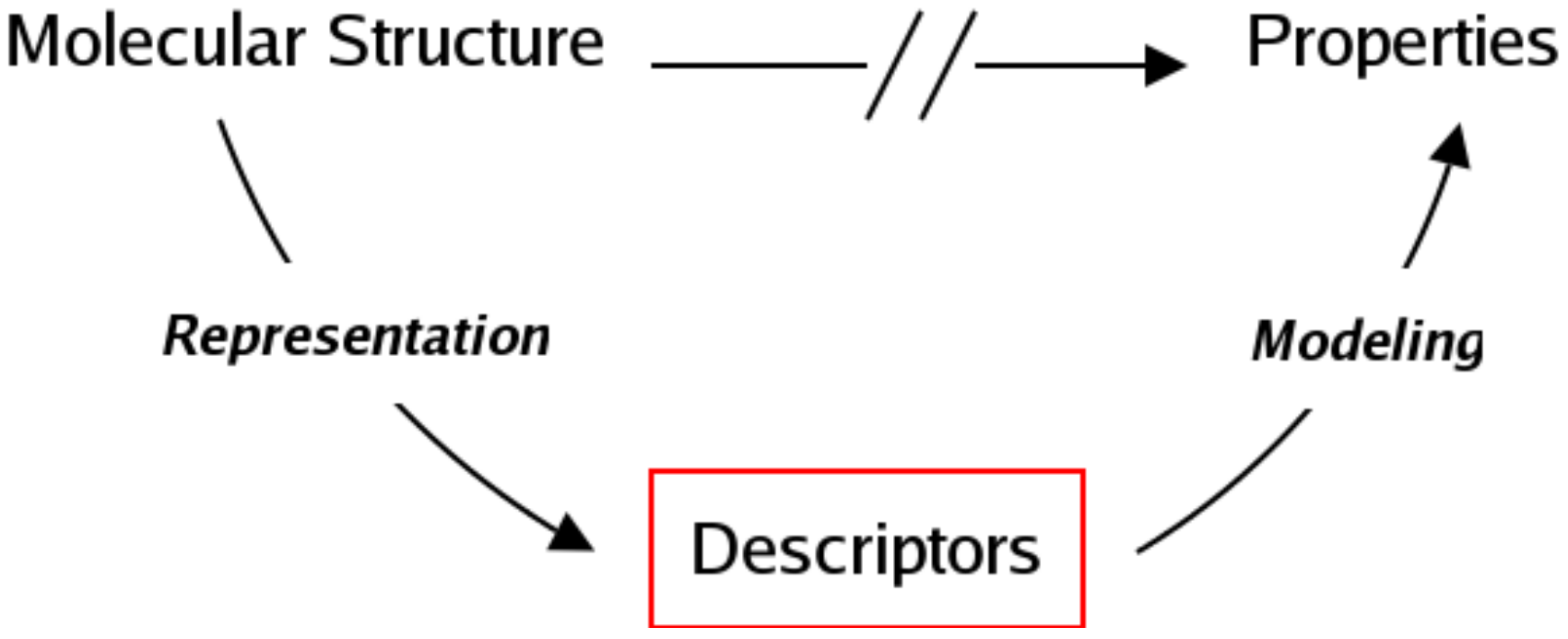
- Directly enter 3D structures (Hyperchem)
- Convert 2D structures (e.g., SMILES) to 3D using Corina or Concord

## Structure Optimization

- Geometry optimization is carried out using MOPAC with the PM3 Hamiltonian
- Electronic optimization uses the AM1 Hamiltonian



# The QSAR Pipeline



# Molecular Descriptors

- Molecular descriptors can be broadly divided into 3 groups
  - Topological
  - Geometric
  - Electronic
- The three classes are combined to generate hybrid descriptors

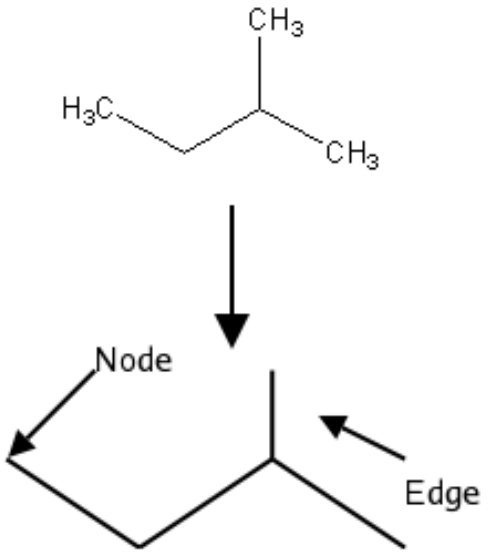
# Molecular Descriptors - Topological

## Characteristics

- Considers a molecule as a graph
- The descriptors are various graph invariants

## Examples

- Connectivity indices
- Substructure counts
- Path length descriptors



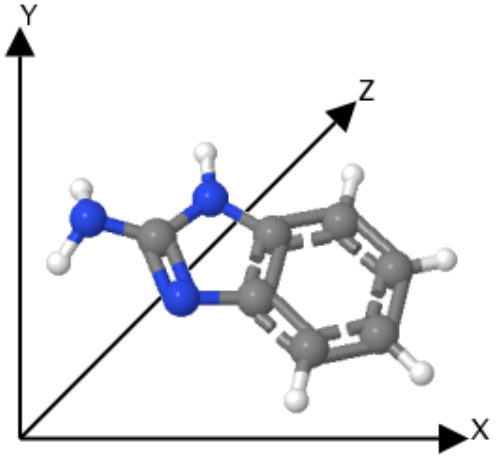
# Molecular Descriptors - Geometric

## Characteristics

- Characterizes the geometry of the molecule
- Dependent on accurate 3D conformations

## Examples

- Moments of inertia
- Molecular surface area and volume
- Length to breadth ratio



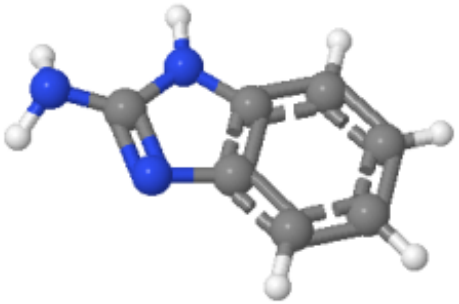
# Molecular Descriptors - Electronic

## Characteristics

- Derived from *ab initio* or semi-empirical calculations
- Characterizes the electronic environment of a molecule

## Examples

- HOMO energies
- Dipole moments
- Partial charges



$$H\psi = E\psi$$

Charges
Dipole moments
HOMO / LUMO Energies
Electronegativity

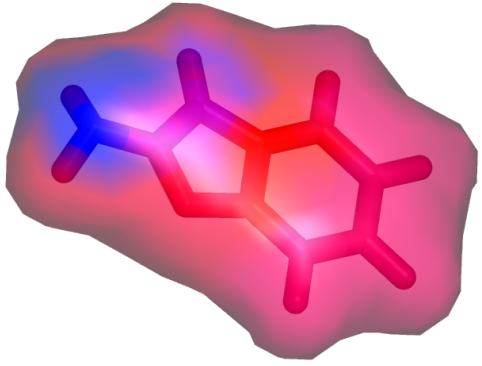
# Molecular Descriptors - Hybrid

## Characteristics

- These descriptors usually combine electronic features and geometric or topological features
- These descriptors are usually information rich

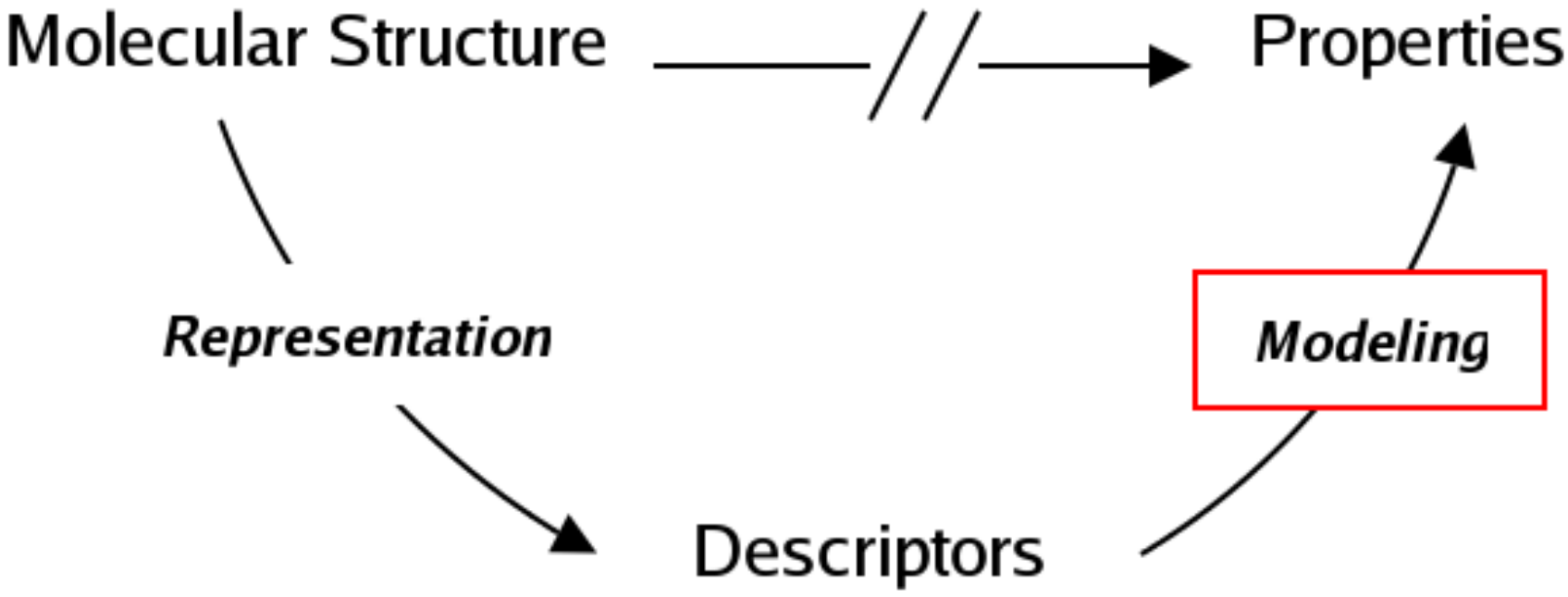
## Examples

- Charged Polar Surface Areas
- Hydrophobic Polar Surface Areas
- H-bond descriptors



Hydrophobic Surface Area

# The QSAR Pipeline



# Building Predictive Models

- At this stage we have a large pool of descriptors for each molecule
- Before we build a predictive model we need to reduce this pool to work with *relevent* and *information rich* descriptors
- Thus modeling can be broken into two steps:
  - Feature selection
  - Model development

# Building Predictive Models - Feature Selection

## Objective

- Uses only independent variables
- Correlation test
- Identical test
- Vector space analysis

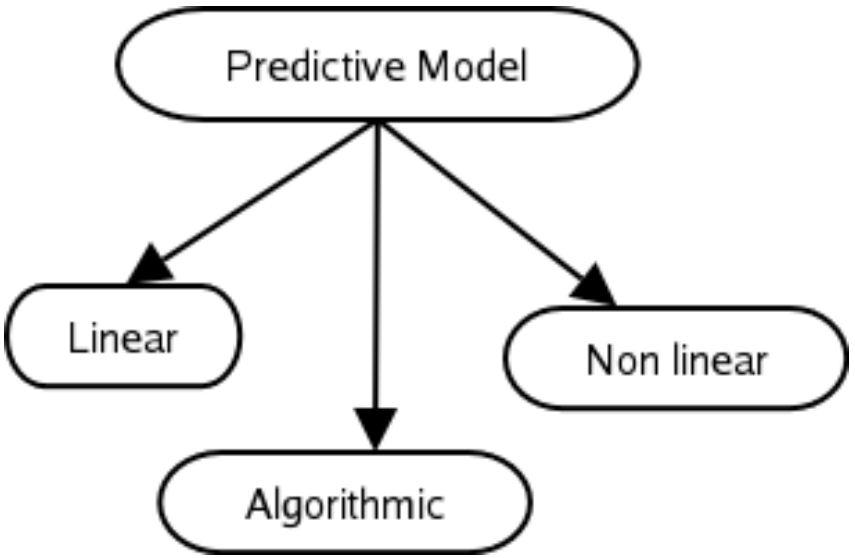
## Subjective

- Uses the dependent variable
- Searches for good descriptor subsets
- Genetic algorithms
- Simulated annealing

# Model Development

## Model Characteristics

- Complexity
- Computational needs
- Flexibility
- Accuracy



# Model Development

## Linear Models

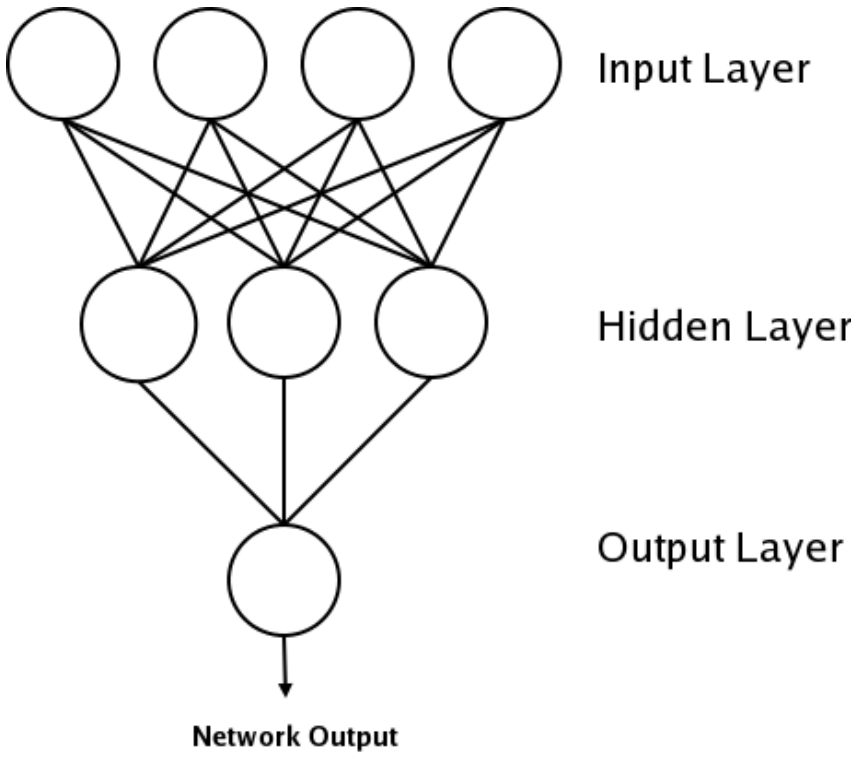
$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots \beta_p x_p$$

- Multiple linear regression, PLS, ...
- Simple and fast to compute
- Not very flexible
- Amenable to interpretation

# Model Development

## Non-linear Models

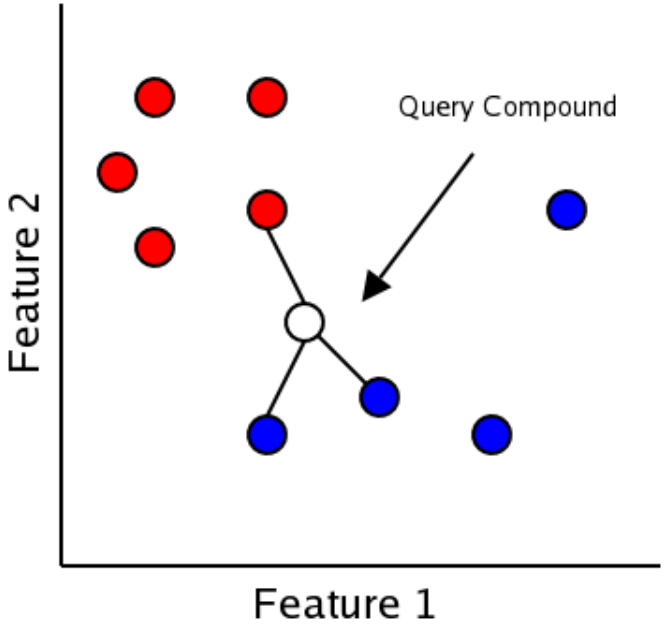
- Neural networks
- Models are complex and computationally intensive to train
- Very flexible
- Black box methodology



# Model Development

## Algorithmic

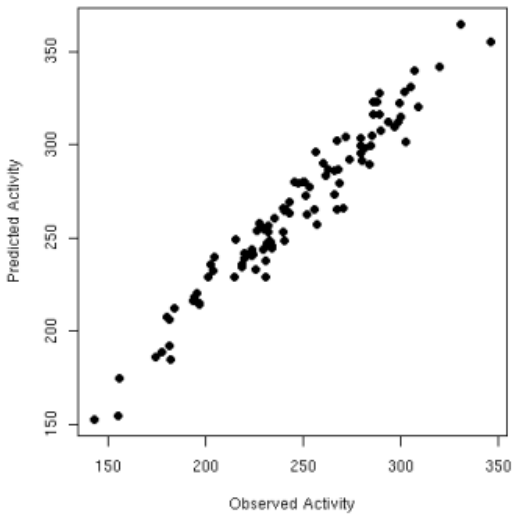
- $k$NN, random forests, ...
- Models are of low complexity and rapid to compute
- Very flexible
- Can be interpreted in some cases

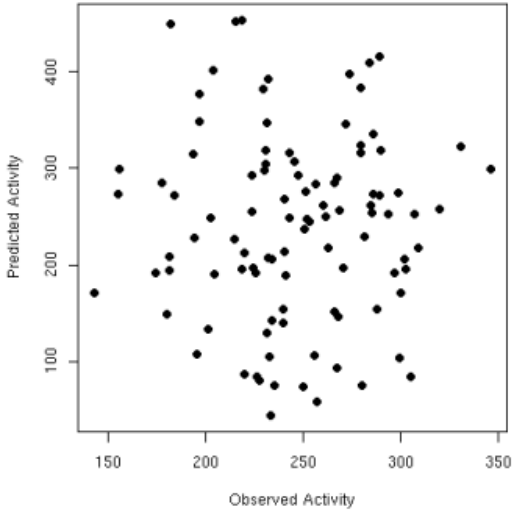


# Model Validation

## Y Scrambling

- This procedure ensures that the model is not due to chance
- Scramble the dependent variable (Y) and make predictions
- A random scatter plot indicates that the model was probably not due to chance

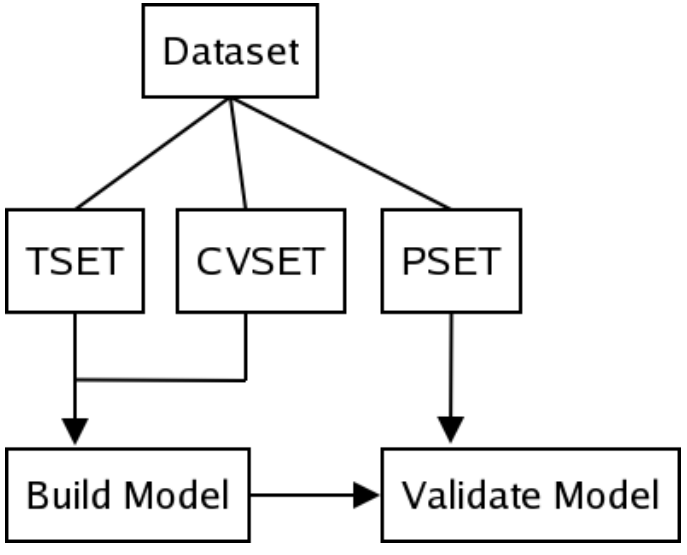


# Model Validation

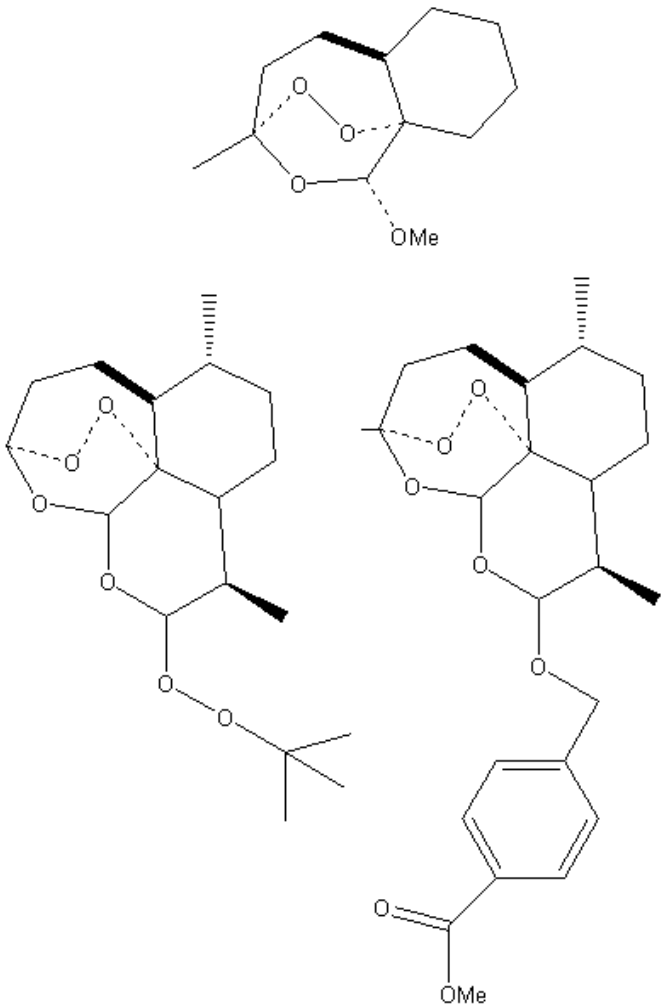
## External Prediction Test

- This procedure tests the model's generalizability
- The PSET is used *only* during this stage
- Characterizes the behavior of the model when faced with new data



# Artemisinin Dataset

- 179 analogs of artemisinin
- Measured property was the logarithm of the relative activity
- A number of molecules had the same value of log RA but diverse structures




Avery, M.A. et al., *J. Med. Chem.*, **2002**, *45*, 292-303

# QSAR Preliminaries

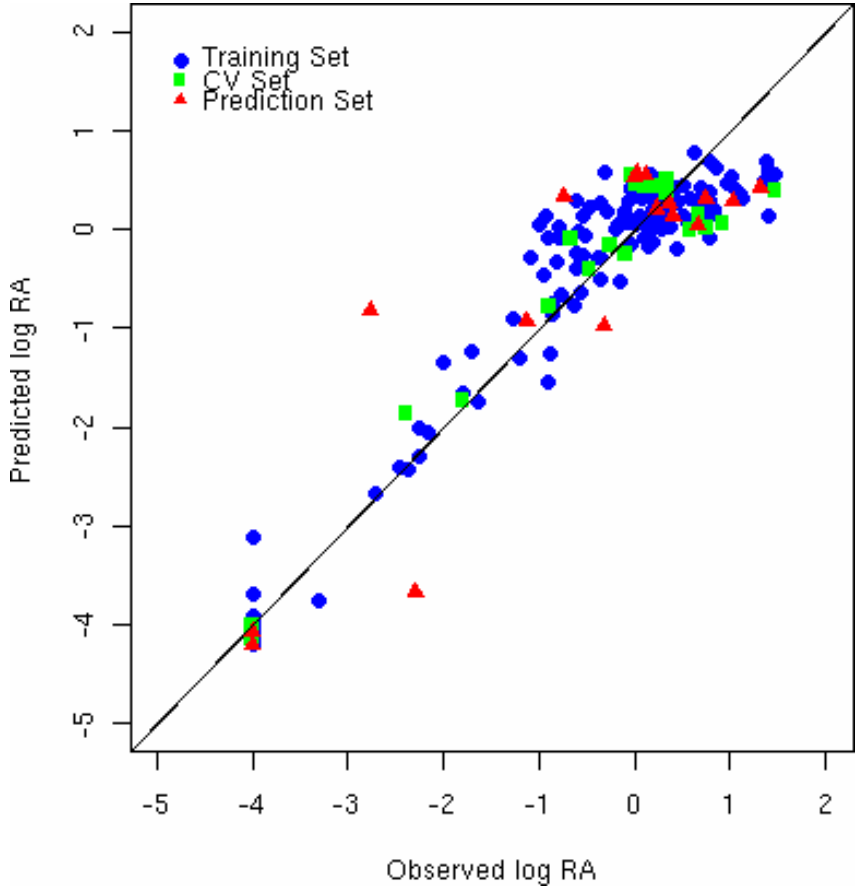
- 179 molecules were divided into 144 (TSET), 17 (CVSET), 18 (PSET)
- The sets were generated using an activity binning method
- 299 descriptors calculated, reduced to 65 descriptors
- Linear and non-linear models were built

Guha, R.; Jurs, P.C; *J. Chem. Inf. Comput. Sci.*, **2004**, *44*, 1440-1449

# Summary of the Best CNN Model

- The model architecture was 10-5-1
- Relatively complex model
- Good statistics

| | $R^2$ | RMSE |
|---|---|---|
| TSET | 0.96 | 0.47 |
| PSET | 0.88 | 0.74 |



# Outline

# Types of Validation

## Model Validation

- Goal is to test the reliability of the model
- Ensures that the model is not due to chance factors
- Based on dataset used to develop the model

# Types of Validation

## Model Validation

- Goal is to test the reliability of the model
- Ensures that the model is not due to chance factors
- Based on dataset used to develop the model

## Model Applicability

- Goal is to test the applicability of the model to new compounds
- Tells us: The model will predict the activity well (or not)
- Similar to confidence measures

# Why Isn't Model Validation Enough?

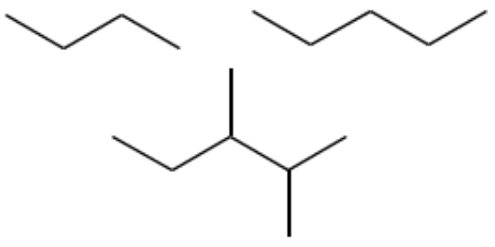
## Training

- Aim is to capture molecular features related to activity
- Features not captured by the model will not be recognized

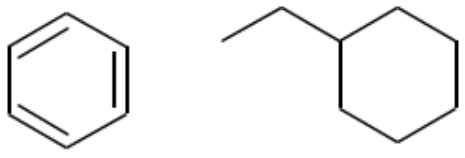
## Prediction

- The PSET is used to see how well the model captured molecular features
- PSET is taken from the same dataset as the TSET
- It will have features in common with the TSET

TSET / PSET Molecules



New Molecules



# Extrapolation Is Not A Good Idea!

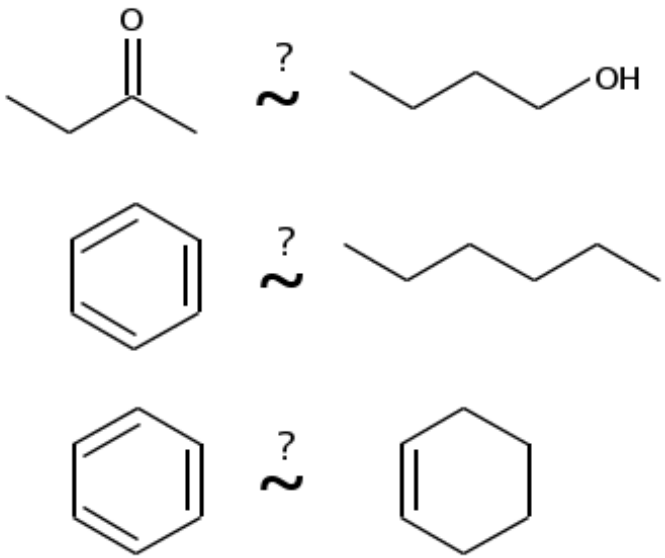
# What Is Model Applicability?

## Question?

How will a model perform when faced with molecules that it has not been trained on or validated with?

## Aspects

- Similarity to the TSET?
- Structural or statistical similarity?
- Quantitative or qualitative?



# How To Assess Model Applicability

## Define Model Performance

Performance is measured by prediction residuals. The model performs well on a new molecule if it predicts its activity with low residual error.

## Correlate *'X'* With Performance

- 'X' could be similarity between a query molecule and the original training set
- 'X' could be derived from a cluster membership approach
- Alternatively, *predict performance* itself

Guha, R.; Jurs, P.C; *J. Chem. Inf. Model.*, **2005**, *45*, 65–73

# Classifying Performance

## Why?

- Our interest is in the model itself
- We can quantify applicability

## How?

1. Consider residuals for TSET
2. Choose a cutoff - residuals above the cutoff are bad and below are good
3. Build a classifier with these class assignments
4. Predict class of residual for query molecules

# Methodology

## Choices Made

- Cutoffs obtained via visual inspection giving 2 classes
- Investigated PLS, LDA, CNN as classifiers
- Pseudo convex data to reduce imbalance classes
- Descriptors from the original models
- Original models were linear regression

## Datasets

- Boiling point (TSET = 235, PSET = 42)
- Activity of artemisinin analogs (TSET = 161, PSET = 18)

Goll, E.S. et al., *J. Chem. Inf. Comput. Sci.*, **1999**, *39*, 974-983

Avery, M.A. et al., *J. Med. Chem.*, **2002**, *45*, 292-303

# Results

## Class Breakup

| | | Class Size | |
|---|---|---|---|
| **Dataset** | **Residual Cutoff** | **Good** | **Bad** |
| Artemisinin | 1.0 | 133 | 46 |
| Boiling Point | 1.0 | 213 | 64 |

# Results

## Weighted Success Rates

| Method | Dataset | TSET | PSET |
|---|---|---|---|
| LDA | Artemisinin | 0.51 | 0.50 |
| | Boiling Point | 0.52 | 0.53 |
| PLS | Artemisnin | 0.51 | 0.46 |
| | Boiling Point | 0.36 | 0.53 |
| CNN | Artemisinin | 0.79 | 0.80 |
| | Boiling Point | 0.98 | 0.93 |

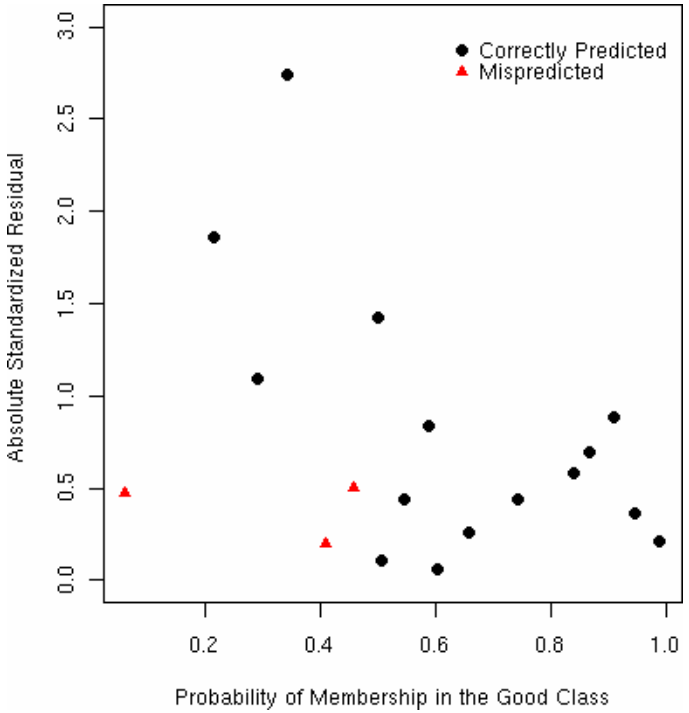
Weston, J. et al., *Bioinformatics*, **2003**, *19*, 764-771

# Results

## Artemisinin / CNN Classifier (4-3-1)

| *TSET* | **Predicted** | |
|---|---|---|
| **Actual** | bad | good |
| bad | 38 | 4 |
| good | 27 | 92 |

| *PSET* | **Predicted** | |
|---|---|---|
| **Actual** | bad | good |
| bad | 4 | 0 |
| good | 3 | 11 |

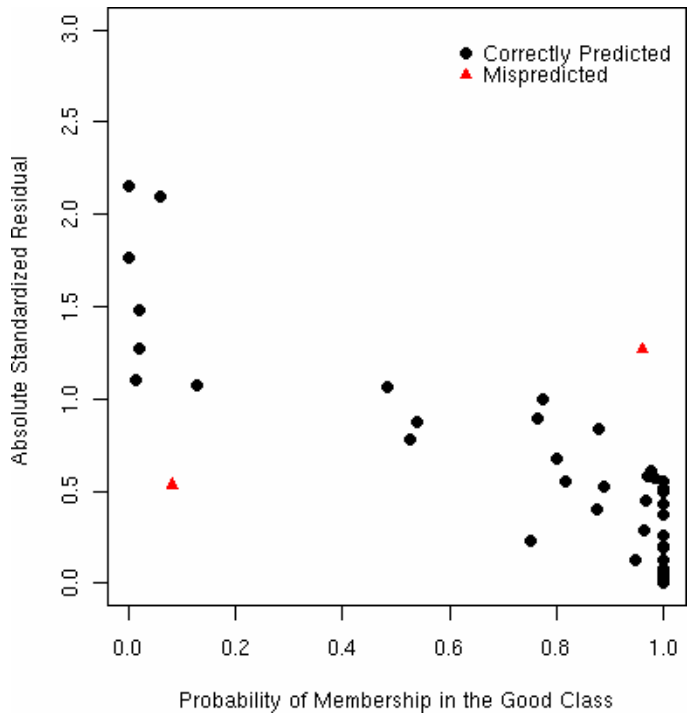


# Results

## Boiling Point / CNN Classifier (4-3-1)

| *TSET* | **Predicted** | |
|---|---|---|
| **Actual** | bad | good |
| bad | 54 | 0 |
| good | 5 | 176 |

| *PSET* | **Predicted** | |
|---|---|---|
| **Actual** | bad | good |
| bad | 9 | 1 |
| good | 1 | 31 |



# Summary

- Model validation is required to ensure model reliability
- Model applicability allows us to decide whether the model will be useful for new compounds
- The classification approach can be applied to *any* quantitative model
- The role of structural similarity needs further investigation

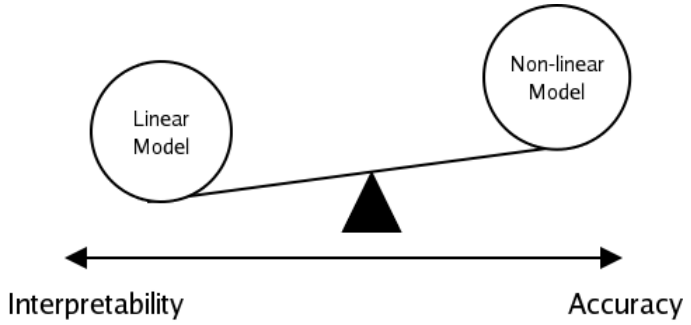
# Outline

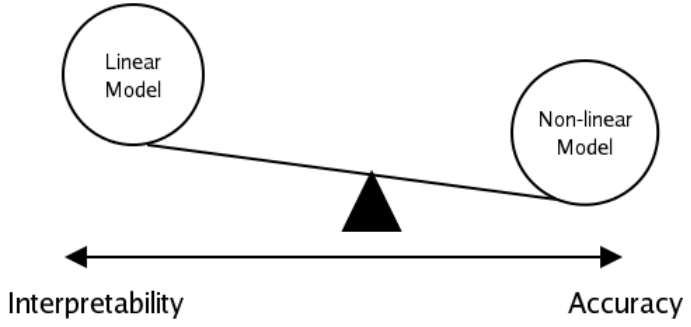
# Isn't a Prediction Enough?

- Predictive models are good for screening purposes
- To understand *why* a compound is active we need an interpretation
- Interpretation is one way to approach the inverse QSAR problem
- Interpretability depends on modeling technique & descriptors involved

# Interpretability and Accuracy

- Interpretability generally involves a trade off with accuracy
- Linear regression models are amenable to interpretation, but not very accurate
- Neural networks are black boxes, but are more accurate
- Some techniques lie in between (random forests)



# Aspects of Interpretability

## Broad Interpretation

- Essentially describes which descriptors are important
- Good for understanding which descriptors to focus on
- Based on randomization

## Detailed Interpretation

- Describes how the property (activity) relates to the descriptor
- Gives us conclusions like:
  **high** value of DESC leads to **low** values of activity
- Allows for a detailed understanding of the SAR in QSAR

# CNN Interpretation in the Literature

- Relative importance of input neurons
- Uses the training set to develop measures of importance
- In many cases the methods depend on the nature of the network

Guha, R., Jurs, P.C., *J. Chem. Inf. Model.*, **2005**, *45*, 800–806

Tickle, A.B. et al., *Intl. Conf. on Neural Networks*, **1997**, *4*, 2530-2534

Yao, S. et al., *Proc. Fifth IEEE Intl. Conf. on Fuzzy Systems*, **1996**, *1*, 361-367

# Goals

## Analogy with PLS Interpretations

The method is analogous to the PLS approach for linear models which considers the linear combination coefficients for each latent variable as indicating the *effect* of a descriptor on the output

# Goals

## Analogy with PLS Interpretations

The method is analogous to the PLS approach for linear models which considers the linear combination coefficients for each latent variable as indicating the *effect* of a descriptor on the output

## Utilizing CNN weights and biases . . .

- Correlate input descriptors to network output through each hidden neuron
- Order the hidden neurons
- Consider hidden neurons as *latent variables*

# Some Preliminaries

## We know . . .

- The transfer function is sigmoidal

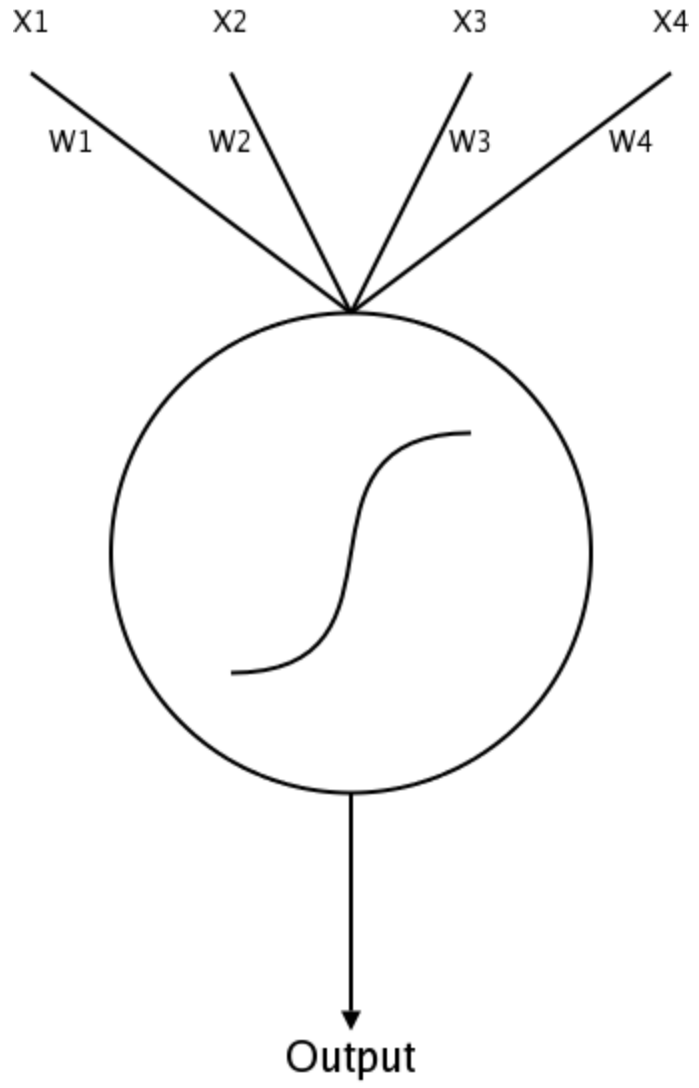
$$O = \frac{1}{1 + \exp(-\sum w_i x_i)}$$

- We can approximate this as

$$O \sim \exp\left(w_1 x_1 + \cdots + w_n x_n\right)$$

## This indicates . . .

- $O$ is an increasing function of its inputs
- Output from a hidden neuron is always positive



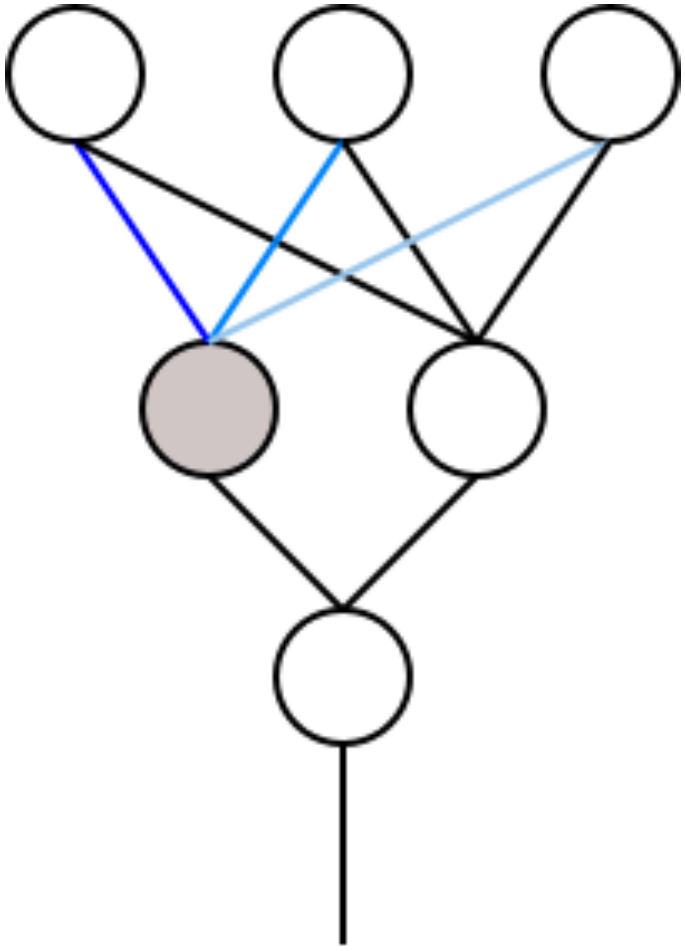
# What Do The Weights Tell Us?

## The absolute values tell us . . .

The weights, $w_i$, determine which input neuron dominates the contribution to a hidden neuron

## The signs tell us . . .

The nature of the correlation between an input to a neuron and the output from the neuron



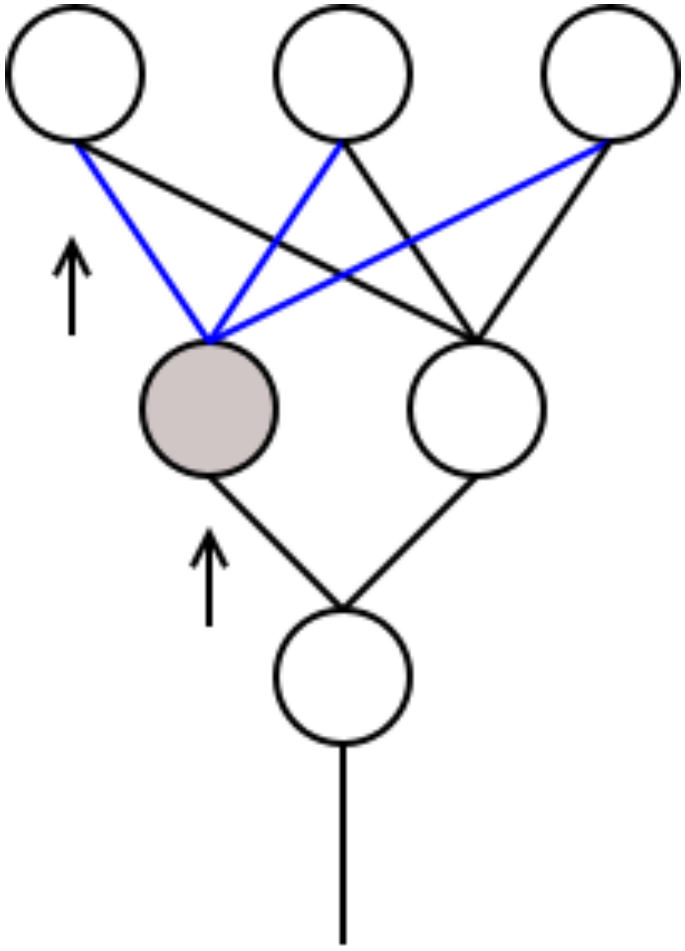
# What Do The Weights Tell Us?

## The absolute values tell us . . .

The weights, $w_i$, determine which input neuron dominates the contribution to a hidden neuron

## The signs tell us . . .

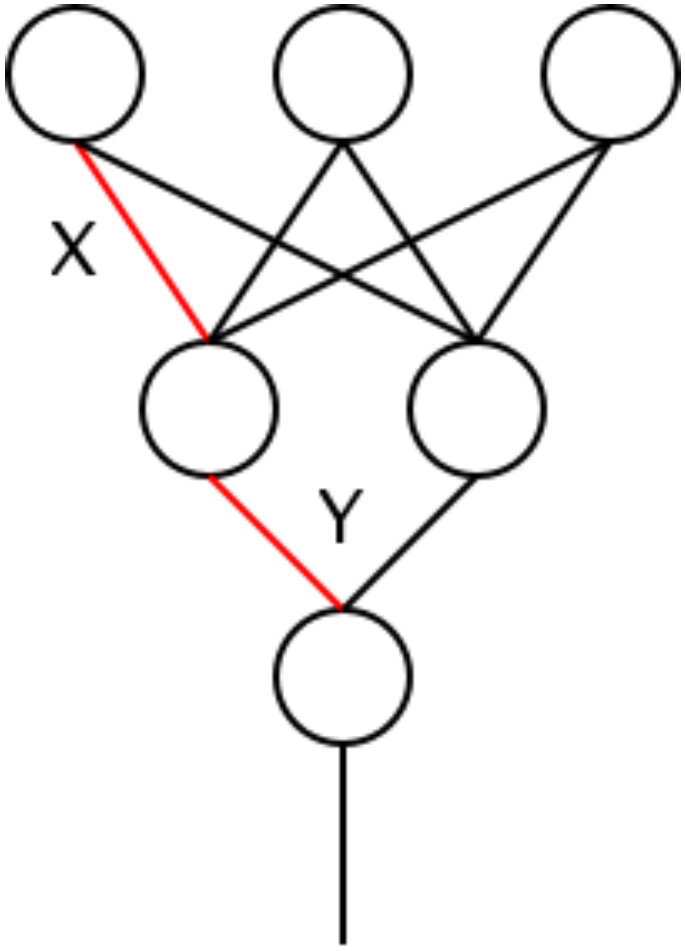
The nature of the correlation between an input to a neuron and the output from the neuron



# Effective Weights

## What are they?

- As input flows from an input neuron to the output neuron it is acted on by two weights
- The effective weight for an input neuron is thus $XY$



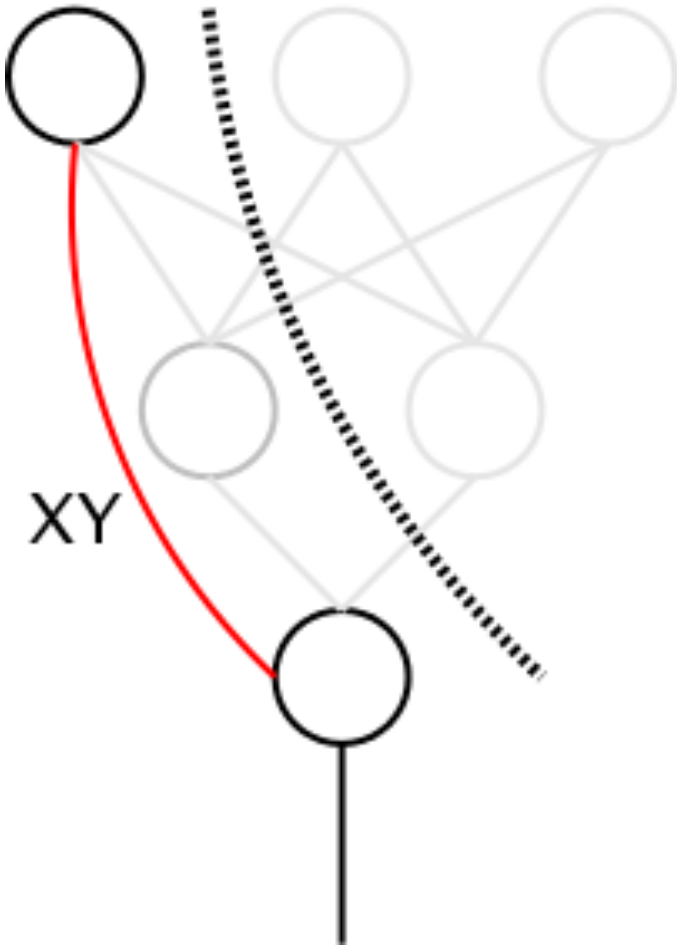
# Effective Weights

## What are they?

The result is that the network looks like a single connection between the input neuron and the output neuron with a weight $XY$

## Effective Weight Matrix

| | Hidden Neuron | |
|---|---|---|
| Descriptor | 1 | 2 |
| Desc 1 | 52.41 | 29.30 |
| Desc 2 | 37.65 | 22.14 |
| Desc 3 | −10.50 | −16.85 |



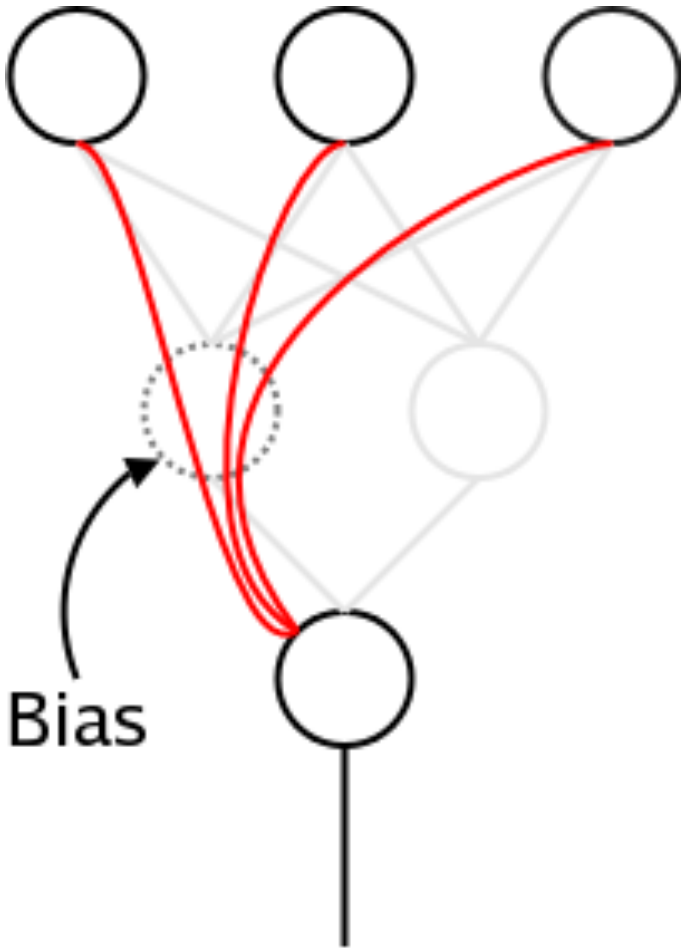
# Ordering Hidden Neurons

## Contribution of a hidden neuron . . .

- Depends on the output of the neuron
- Depends on the inputs to the neuron

## Quantifying Contributions

- Take the column means of the effective weight matrix
- Also include bias terms for each hidden neuron
- Convert to a proportional scale for ease of use (SCV)



# Validation of the Method

- Build a linear model with $N$ descriptors and interpret it
- Build a CNN model with the same descriptors and interpret it

The two interpretations should match since both models should encode similar SPR trends

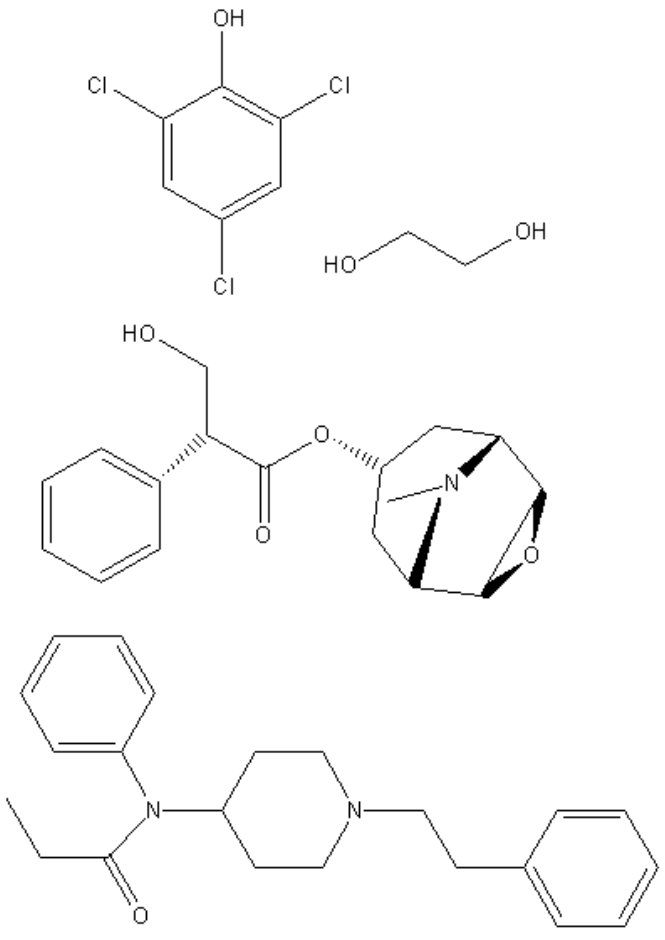
# Skin Permeability

## Dataset

- Original work reported linear models
- Measured activity was the permeability coefficient ($K_p$)
- $-5.03 < log(K_p) < -0.85$

## Model details

- 7 descriptor OLS model
- $R^2 = 0.84$, $RMSE = 0.37$ log units
- CNN model was 7-5-1
- $R^2 = 0.94$, $RMSE = 0.23$ log units




Patel, H. et al., *Chemosphere*, **2002**, *48*, 603-613

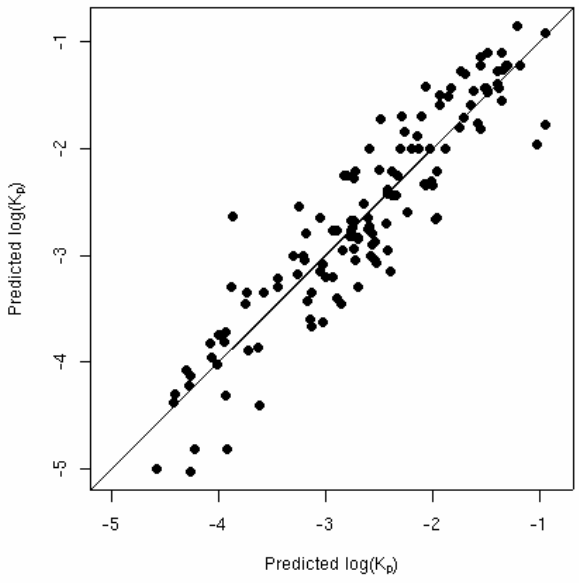
# Linear Interpretation

## Component 1 focuses on . . .

- Smaller size
- Lower polar surface area
- Larger hydrophobic surface area

## Component 2 focuses on . . .

- Larger hydrophobic surface area
- Larger surface area
- Corrections for the overestimation or underestimation of some molecules in component 1




| | Component | |
|---|---|---|
| Descriptor | 1 | 2 |
| SA | −0.08 | **0.52** |
| FPSA-2 | **−0.52** | 0.14 |
| NN | −0.36 | −0.03 |
| MOLC-9 | **0.61** | 0.11 |
| PPHS-1 | 0.03 | **0.69** |
| WPHS-3 | 0.09 | **0.48** |
| RNHS | **0.46** | −0.04 |

# CNN Interpretation - Effective Weight Matrix

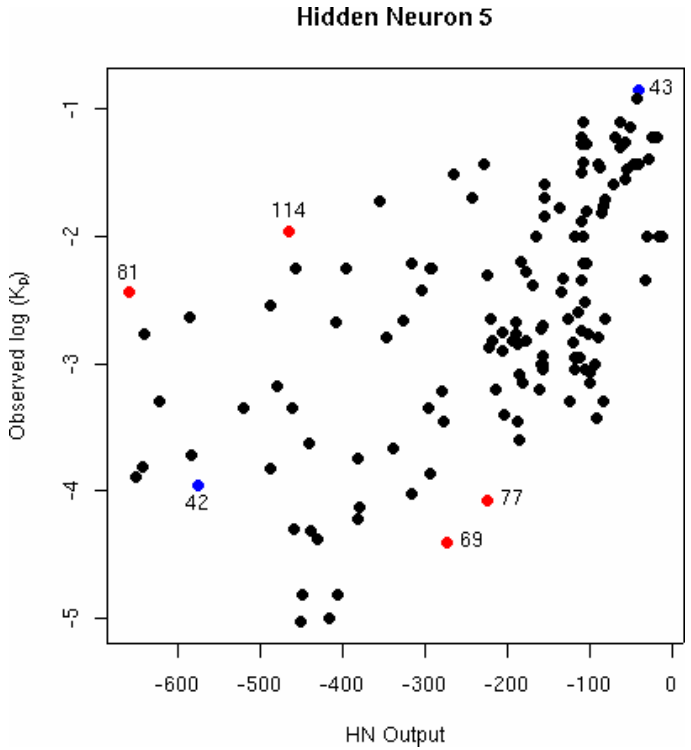
| | Hidden Neuron | | | | |
|---|---|---|---|---|---|
| Descriptor | 5 | 2 | 4 | 3 | 1 |
| SA | -44.17 | **67.34** | 8.33 | 8.18 | 5.96 |
| FPSA-2 | **-156.82** | -10.72 | **20.85** | -13.07 | -92.47 |
| NN | **-97.81** | 2.22 | -6.65 | 1.71 | -12.70 |
| MOLC-9 | -28.85 | **17.79** | 15.40 | -11.36 | -1.20 |
| PPHS-1 | **106.55** | **31.30** | -16.76 | -13.99 | 34.55 |
| WPHS-3 | -11.36 | **-14.31** | -2.31 | -10.01 | 54.16 |
| RNHS | 20.16 | -5.89 | **-49.57** | 23.88 | 27.09 |
| SCV | 0.85 | 0.13 | 0.02 | 0.01 | 0.00 |

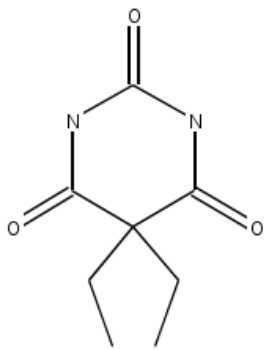
- The most important neuron focuses on hydrophobic & polar effects
- The next most important neuron focuses on size effects

# CNN Interpretation - Score Plot for Hidden Neuron 5

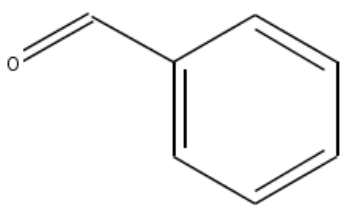
## Observations . . .

- SCV = 0.85
- Active molecules area characterized by low polar surface area and larger hydrophobic surface area
- Does not perform too well on inactive molecules
- 69,77 and 81,114 are mispredicted




42 (-3.95)

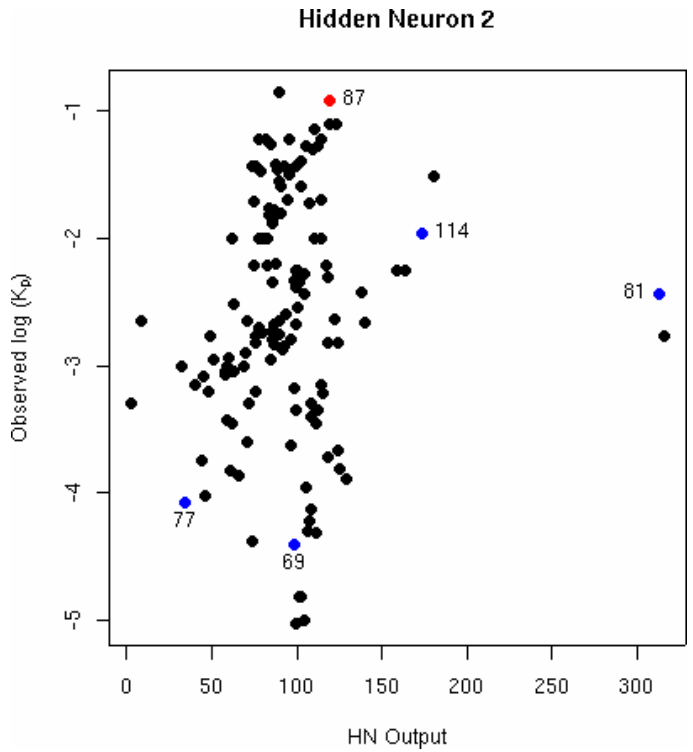


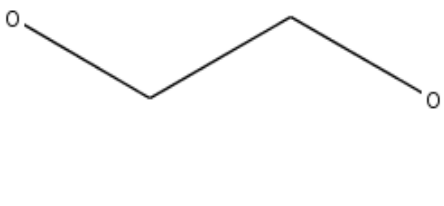

43 (-0.85)

# CNN Interpretation - Score Plot for Hidden Neuron 2

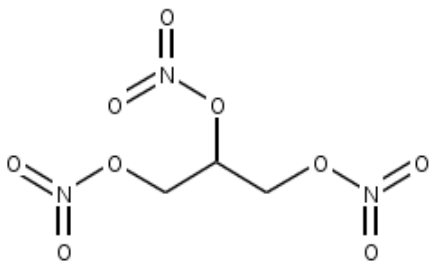
## Observations . . .

- SCV = 0.13
- Corrects for 69,77 and 81,114
- Describes larger molecules with higher hydrophobic surface area
- MOLC-9 balances the effect of MW
- Molecule 87 is underestimated

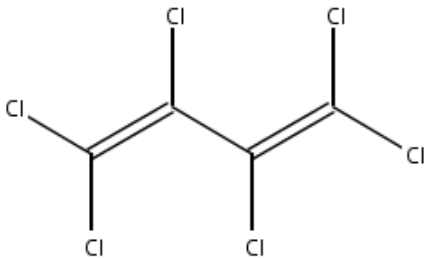


77 (-4.07)



114 (-1.96)

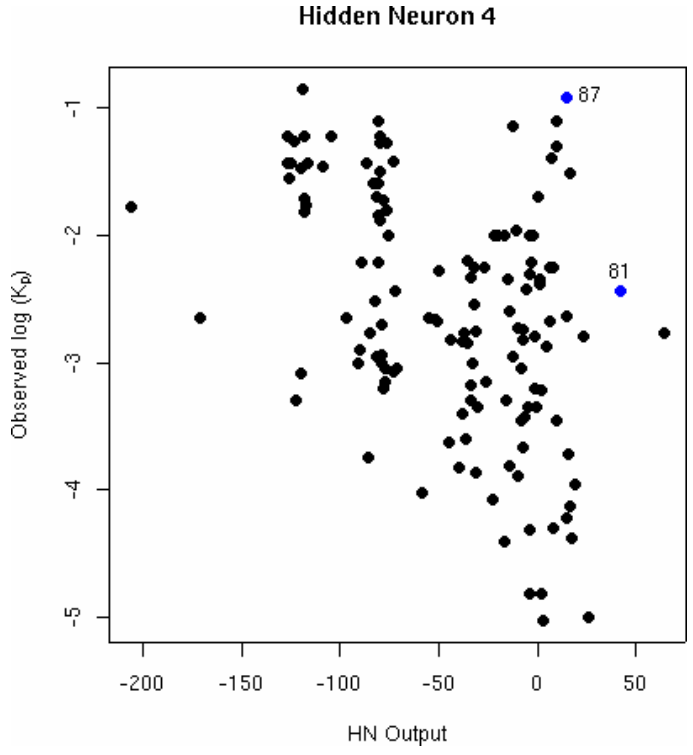


87 (-0.92)

# CNN Interpretation - Score Plot for Hidden Neuron 4

## Observations . . .

- Corrects underestimation of molecule 87 by HN 2
- Further corrects for molecule 81
- Does not perform well for inactive molecules



# Summary

## Caveats

- The method *linearizes* the network
- Clearly, the interpretations will loose some of the details of the encoded SPR's

# Summary

## Caveats

- The method *linearizes* the network
- Clearly, the interpretations will loose some of the details of the encoded SPR's

## Conclusions

- CNN interpretations appear to be valid
- Discrepancies may be present if we do not select optimal descriptor subsets for the CNN model
- The method avoids complexity and uses only the weights and biases and hence does not use the training set explicitly

# Conclusions

- Validation & interpretation are two important aspects of QSAR modeling

- Validation is required to assess reliability & applicability
- A classification approach to validation is quite general in nature and performs well

- Interpretation plays an important role in drug *design*
- The broad and detailed interpretation methods reduce the black box nature of CNN QSAR models

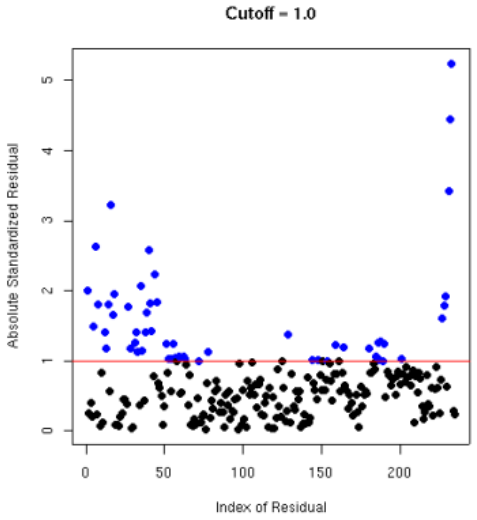
# Acknowledgements

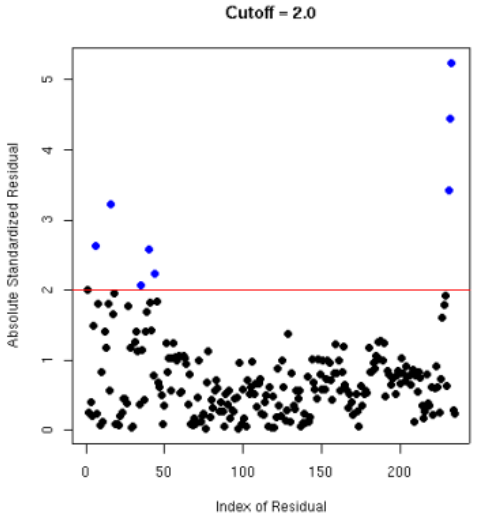
- Prof. P.C. Jurs
- Dr. B.E. Mattioni
- Dr. D.T. Stanton
- NSF

# Classifying Performance

## Choices

- How do we choose a cutoff?
- How many classes do we take?
- What classifier do we use?
- How do we handle unbalanced classes?
- Which descriptors do we use for the classifier?



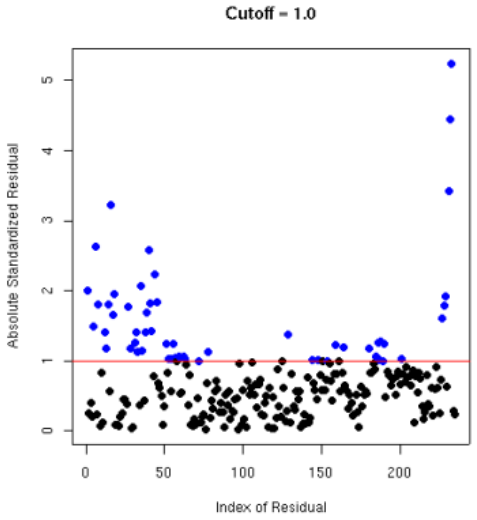
# Classifying Performance

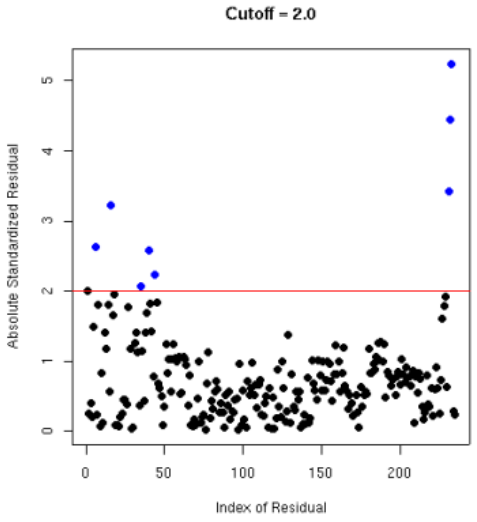
## Choices

- How do we choose a cutoff?
- How many classes do we take?
- What classifier do we use?
- How do we handle unbalanced classes?
- Which descriptors do we use for the classifier?

## Possibilities

- Visual inspection
- Regression diagnostics



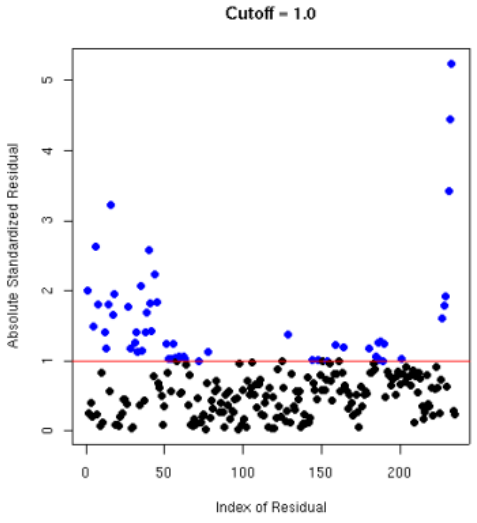
# Classifying Performance

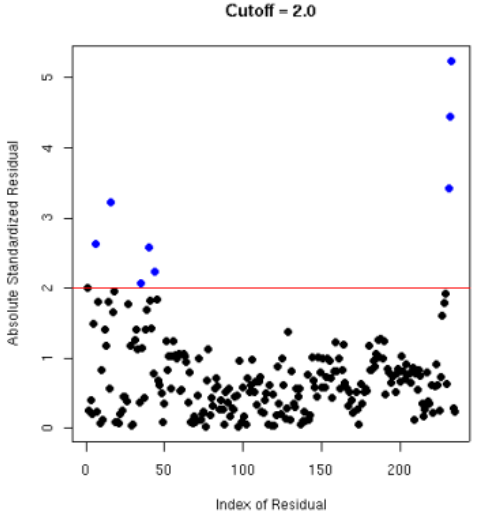
## Choices

- How do we choose a cutoff?
- How many classes do we take?
- What classifier do we use?
- How do we handle unbalanced classes?
- Which descriptors do we use for the classifier?

## Possibilities

- Depends on the size of the dataset
- More classes allow for finer analysis



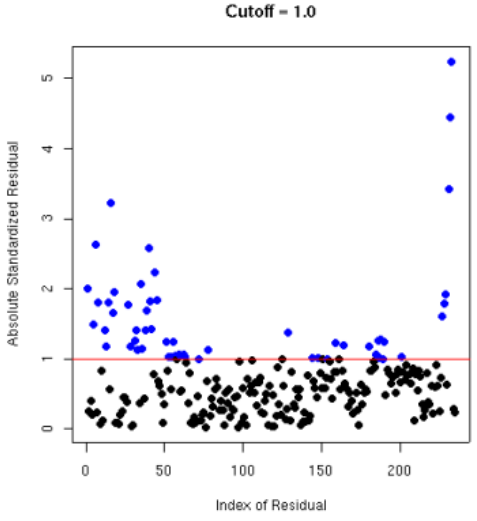
# Classifying Performance

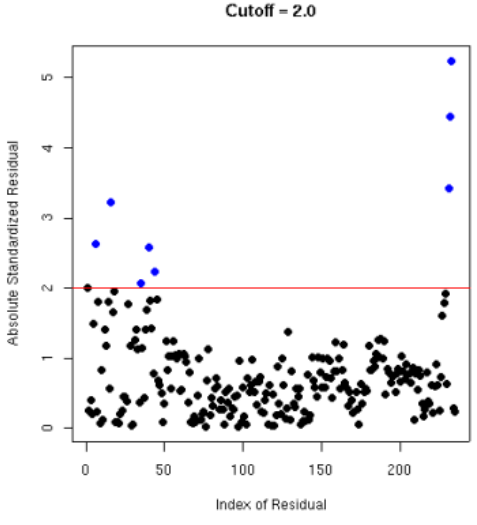
## Choices

- How do we choose a cutoff?
- How many classes do we take?
- What classifier do we use?
- How do we handle unbalanced classes?
- Which descriptors do we use for the classifier?

## Possibilities

- Linear: LDA and PLS
- Non-linear: CNN



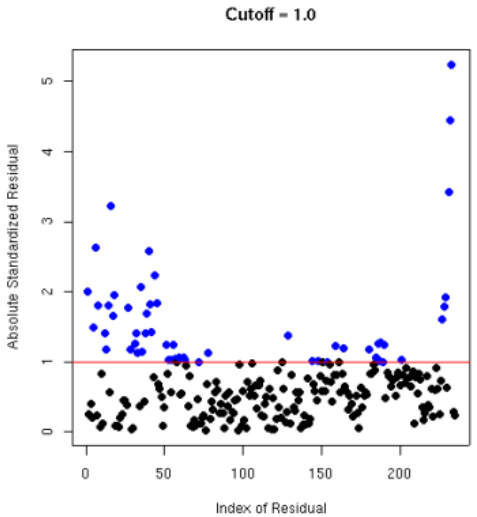
# Classifying Performance

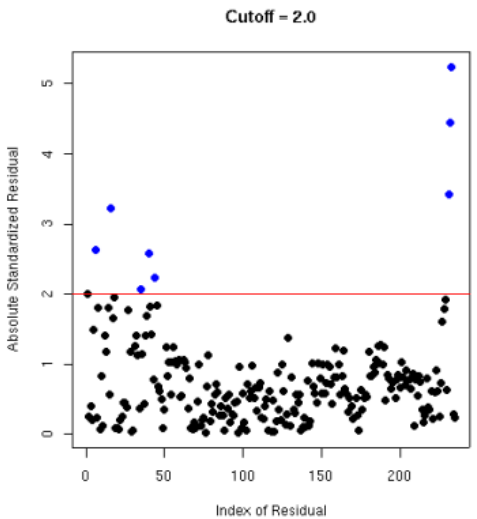
## Choices

- How do we choose a cutoff?
- How many classes do we take?
- What classifier do we use?
- How do we handle unbalanced classes?
- Which descriptors do we use for the classifier?

## Possibilities

- Oversampling or undersampling
- Use pseudo convex data




Breiman, L., *Technical Report 513*, **1998**, Dept. of Statistics, UC Berkeley

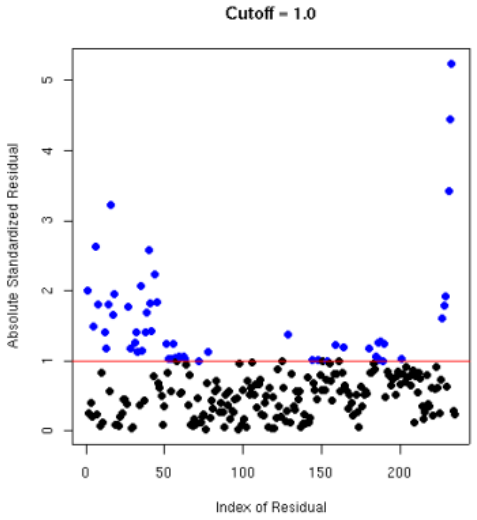
# Classifying Performance

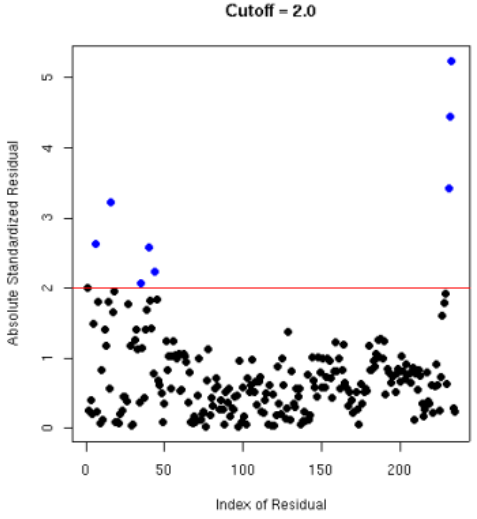
## Choices

- How do we choose a cutoff?
- How many classes do we take?
- What classifier do we use?
- How do we handle unbalanced classes?
- Which descriptors do we use for the classifier?

## Possibilities

- The descriptors used in the original model
- Global descriptors



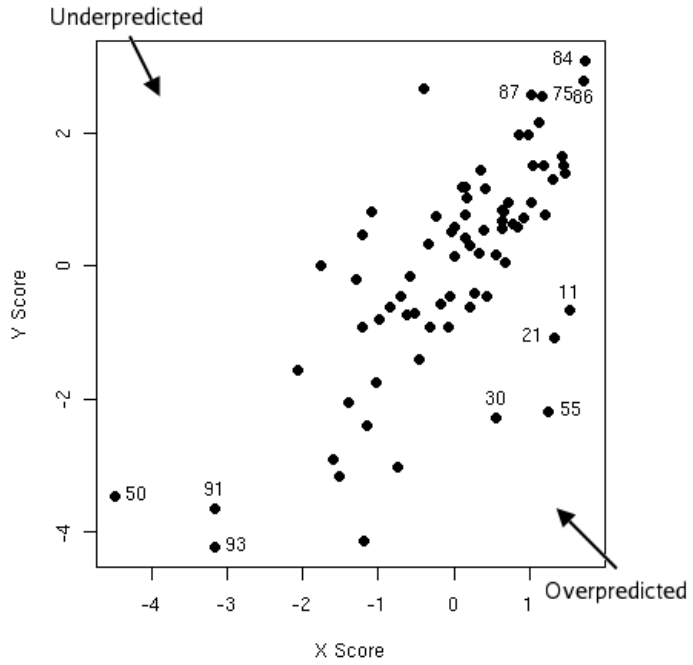
# PLS Interpretation - Artemisinin

## Component 1

- SURR-5 is most weighted
- Low values of SURR-5 $\Rightarrow$ high values of predicted activity

## Interpretation

- Active compounds have high absolute values of SURR-5
- Indicates large hydrophobic surface area
- Consistent with cell based assay which depends on cell membrane transport




| | MDEN-23 | RNHS-3 | SURR-5 |
|---|---|---|---|
| C1 | -0.16 | 0.55 | -0.82 |

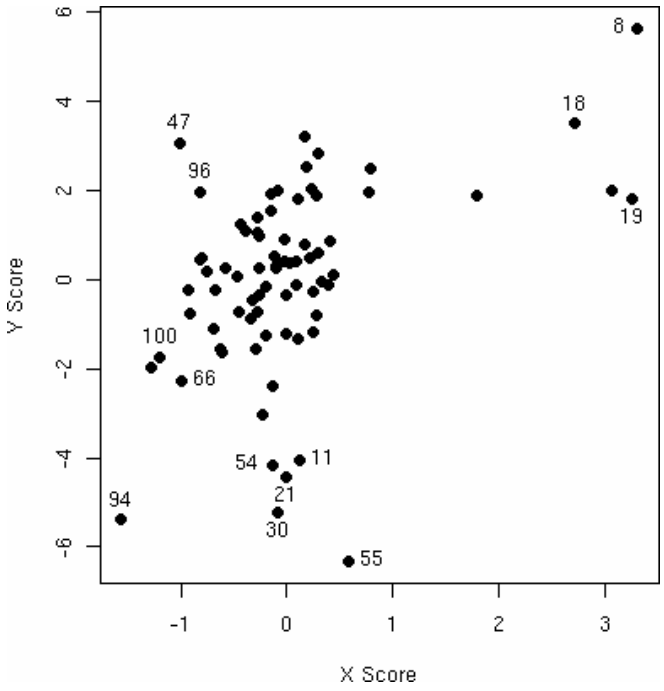
# PLS Interpretation - Artemisinin

## Component 2

- MDEN-23 is the most weighted
- High values of MDEN-23 $\Rightarrow$ high values of predicted activity

## Interpretation

- High values of MDEN-23 imply larger number of paths between secondary and tertiary N
- Experiments indicate removal of basic groups reduce potency

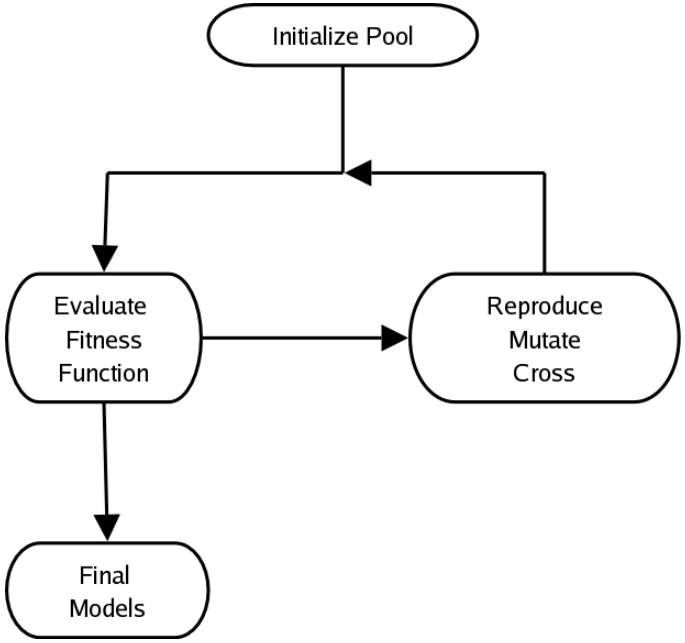



| | MDEN-23 | RNHS-3 | SURR-5 |
|---|---|---|---|
| C1 | 0.93 | -0.17 | -0.29 |

# Genetic Algorithms

## Features

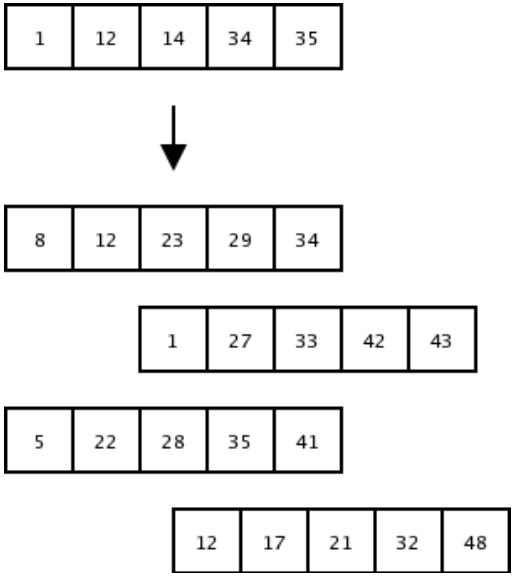
- Stochastic optimization procedure
- Based on evolutionary principles
- Applicable to large search spaces
- Not always guaranteed to find the optimal solution
- Fitness is evaluated by the *objective function*



# Genetic Algorithms

## Initialization

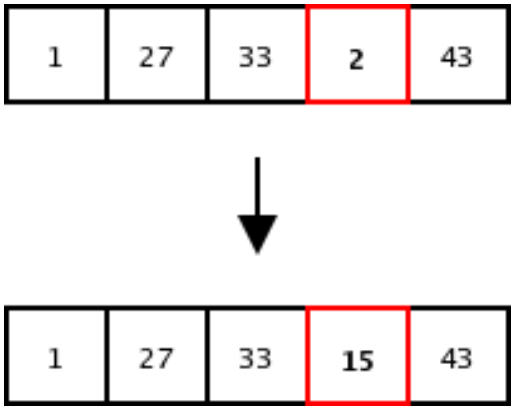
- Initially random chromosomes
- Chromosome length is the size of the descriptor subset
- Genes are the descriptors
- Larger pool sizes allow for a wider search



# Genetic Algorithms

## Mutation

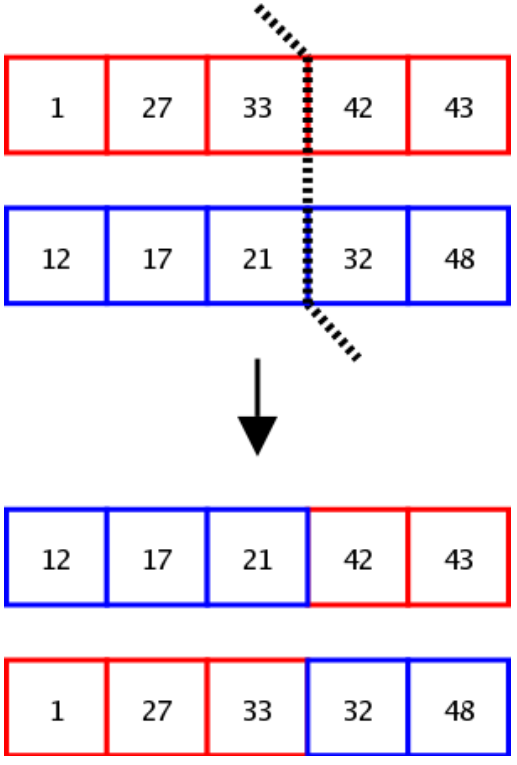
- Mutation frequency is relatively low
- Mutations randomly change a single descriptor
- Helps to get out of local minima



# Genetic Algorithms

## Crossover

- Crossover results in swapping of genes
- Crossover between fit individuals should lead to children with good aspects of both parents
- In single point crossover
  - Choose crosspoint
  - Swap corresponding sections
  - Results in two new children



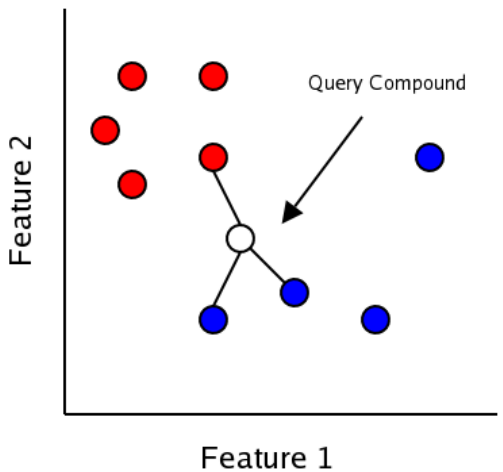
# Simulated Annealing

- Uses the idea of annealing in physical systems
- Based on the Boltzman distribution
- Stochastic in nature
- Good for large search spaces

# $k$ Nearest Neighbor

## Features

- Model free technique
- Very simplistic method
- $k$ is obtained by trial and error or cross-validation
- Used for regression and classification



# Computational Neural Networks

## Pros

- Very flexible modeling technique
- Generally quite accurate
- Large variety of modifications to the basic algorithm

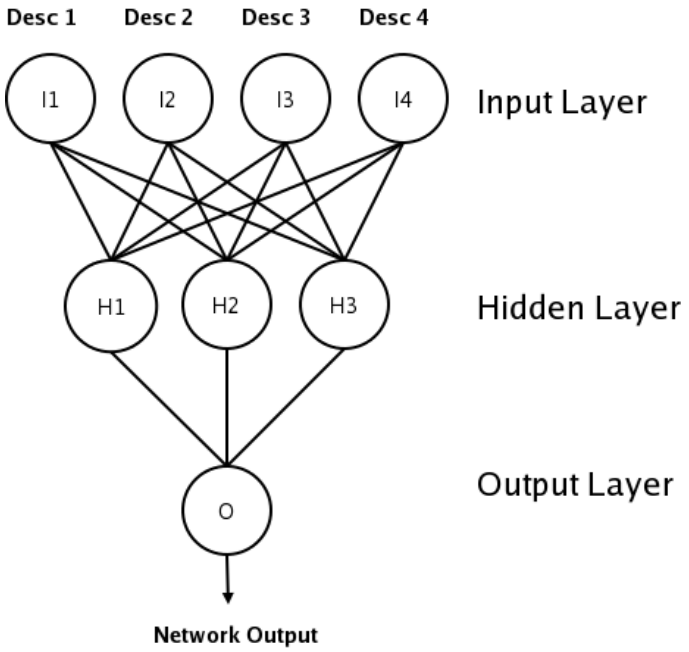
## Cons

- Time consuming to build
- Optimal architectures is diffifcult to decide on
- Very difficult to interpret

# Computational Neural Networks

## Structural Features

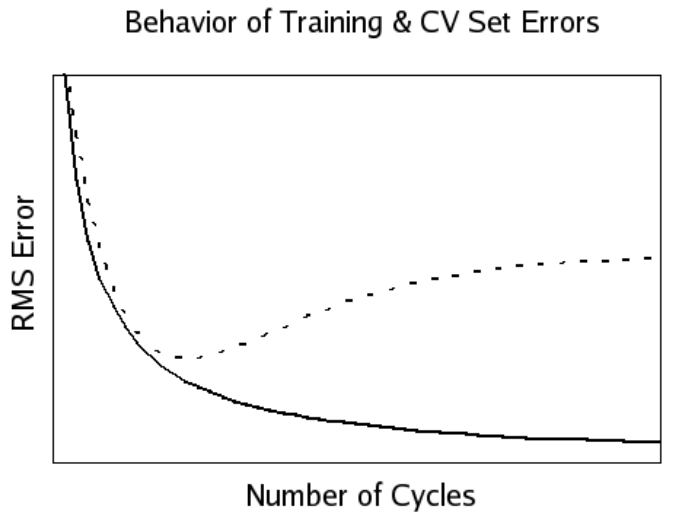
- Input neurons are descriptors
- All neurons in a layer are connected to neurons in the next layer
- Hidden and output neurons utilize a sigmoidal transfer function
- Weights and biases must be optimized

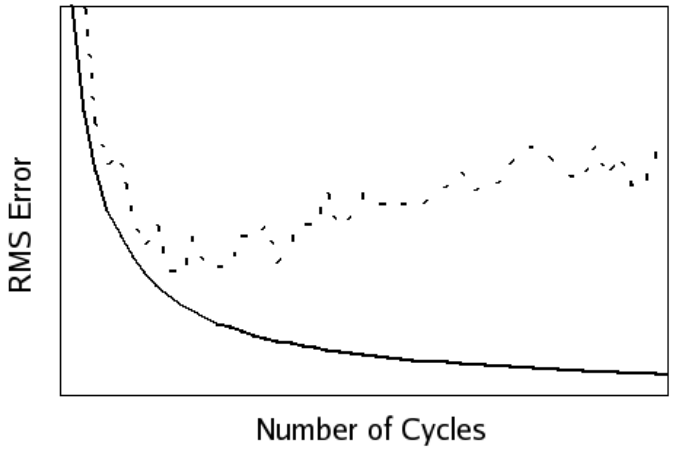


# Computational Neural Networks

## Training

- Training allows the CNN to learn characteristic features of the datasset
- Training $\rightarrow$ Optimization of weights & biases
- Overtraining is prevented by cross validation
- Training and cross validation statistics can provide a cost function



# Random Forest

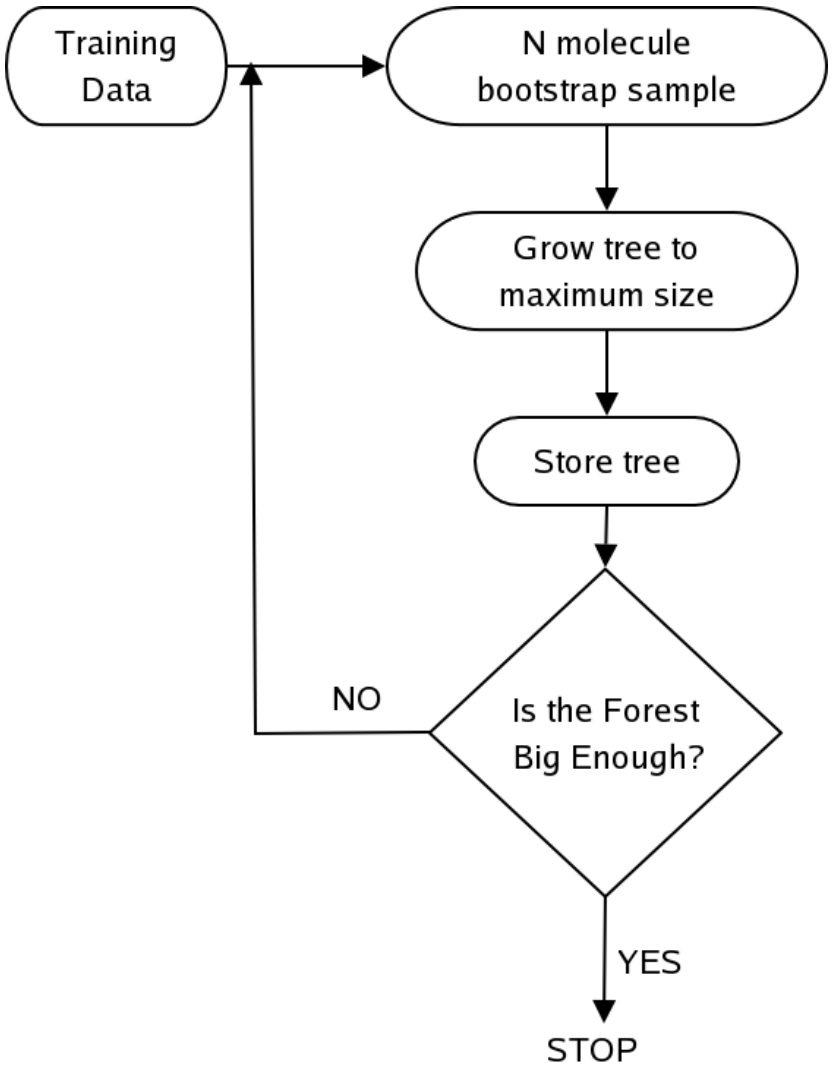
## Features

- Built in estimation of prediction accuracy
- Measure of descriptor importance
- Measure of similarity

## Random Forest vs. Decision Tree

- RF is faster because it searches fewer descriptors at each split
- A decision tree must be pruned via cross validation. RF trees are grown to full depth
- RF is much more accurate than a decision tree

# Random Forest



# OOB Samples

## What is OOB?

- Due to bootstrap sampling, each tree uses $\approx 2/3$ of the TSET
- The remaining 1/3 is the Out Of Bag sample
- The OOB sample can be used to estimate performance during training or measure descriptor importance

## Descriptor Importance

- Decision trees create describe explicit relationships between descriptors and predictions
- These relationships are hidden in a RF model
- A randomization procedure on OOB samples can be used to rank descriptors in importance

# Weight Success Rates

$$WSR = \frac{1}{2}\left(\frac{\text{No. True Positives}}{\text{No. Positive Examples}} + \frac{\text{No. True Negatives}}{\text{No. Negative Examples}}\right)$$

- $0 \leq WSR \leq 1$
- Useful for characterizing unbalanced classification problems

# Why Do We Ignore The Bias Term?

## Equipartitioning View

- When considering effective weights via a given hidden neuron, the bias term must be partitioned.
- The simplest approach is to equipartition the bias term
- The net result is that the same value is added to each effective weight.

## Constant Bias View

- CNN's exhibit the universal function approximation property
- A sufficient condition for this is that the transfer function has a non-zero derivative at the origin
- This implies that the bias can be taken as a constant rather than trainable weight

Hornik, K. *Neural Networks*, **1993**, *6*, 1069-1072

# Broad Interpretation

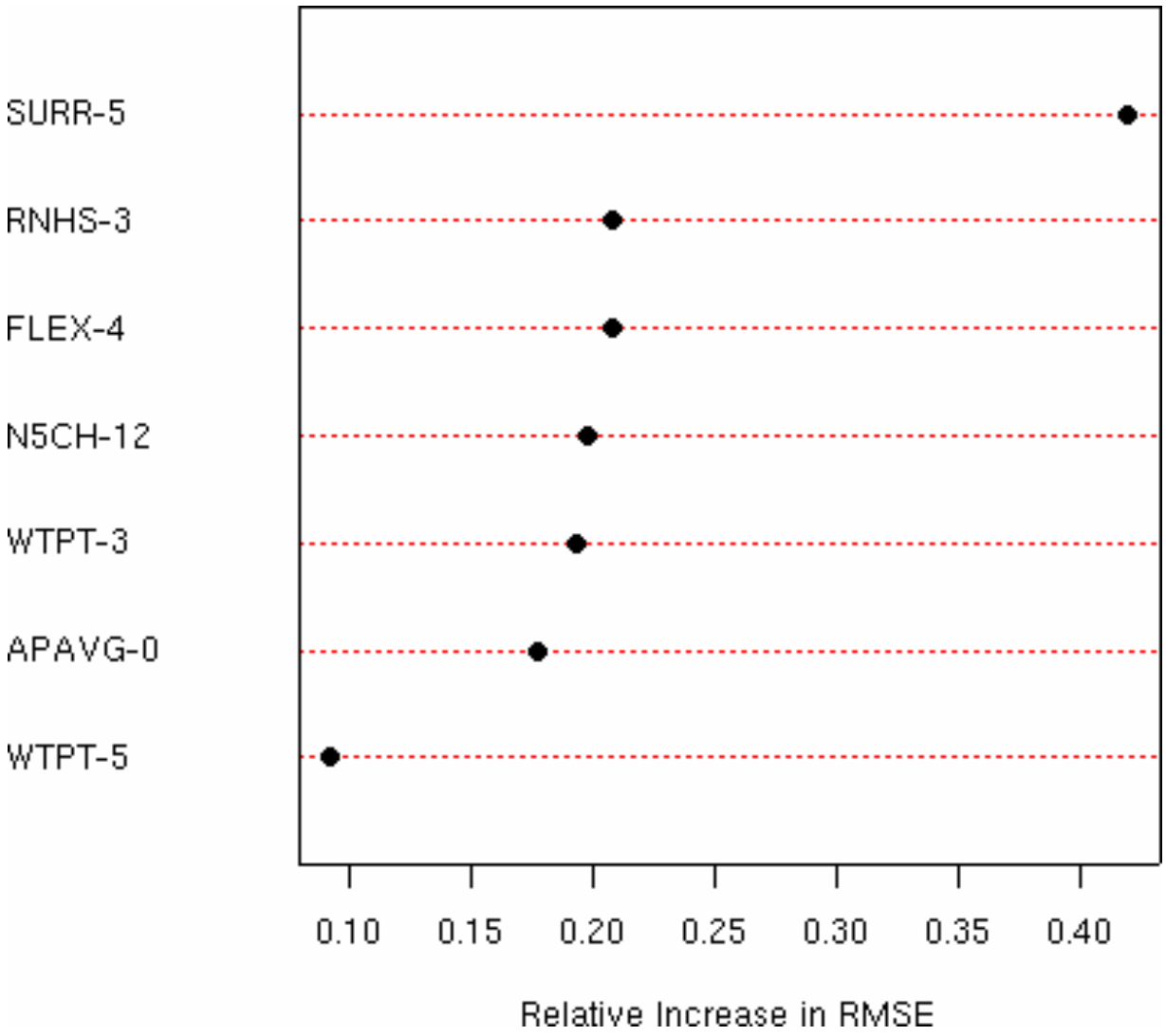
## Background

Similar in concept to the idea of descriptor importance for the random forest technique. Measures the sensitivity of the CNN model to changes in specific descriptors

## Method

- Use the training set to evaluate the RMSE for the model
- Scramble a single descriptor and evaulate the RMSE for the model
- The difference between the *scrambled* RMSE and original RMSE indicates the importance of this descriptor
- Repeat for each descriptor in the model

# Broad Interpretation



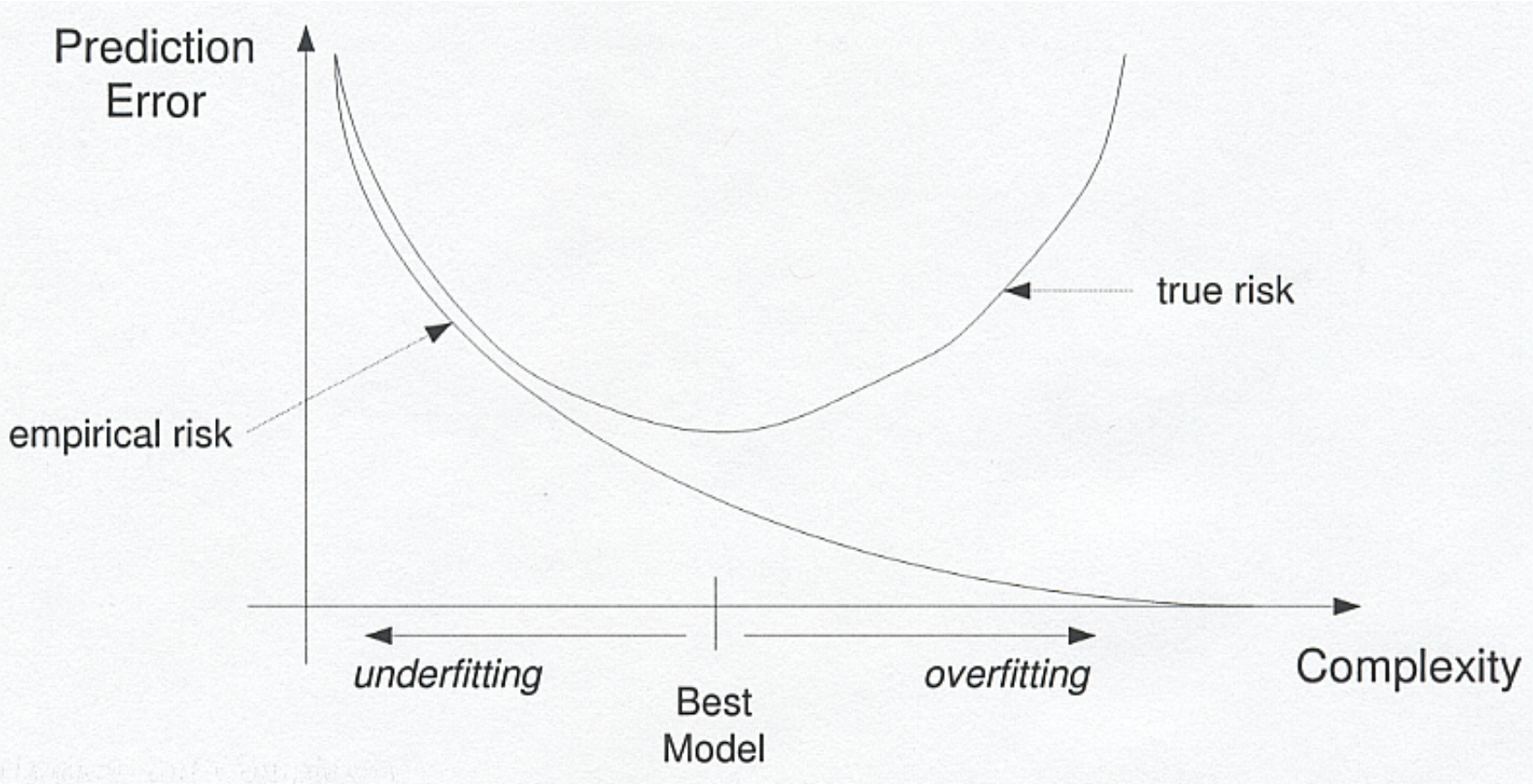
# Model Complexity

- Very simple models or very complex models may perform poorly
- The optimal performance and complexity are obtained from a trade off between *bias* and *variance*

$$Mean\ Squared\ Error = Var(\hat{y}) + Bias^2(\hat{y}), \quad \text{where}$$

$$Var \propto \frac{1}{complexity} \quad \text{and} \quad Bias \propto complexity$$

# Model Complexity




http://www.ece.wisc.edu/~nowak/ece901/lecture3.pdf