BMI-203: Biocomputing Algorithms
Lecture 6: Optimization II Machine Learning

Ajay N. Jain, PhD

Associate Professor, Cancer Research Institute and Dept. of Laboratory Medicine

University of California, San Francisco

ajain@cc.ucsf.edu

http://jainlab.ucsf.edu

Outline

- Machine learning
  - Machine learning is essentially optimization
  - There is a twist though
- Machine learning methods and examples
  - Symbolic versus numeric approaches
  - K nearest neighbors: Example with gene expression
  - Neural Networks: Examples with molecules
  - Genetic Algorithms: Very brief description

  - Good reference: Machine Learning, Tom M. Mitchell, 1997 (McGraw-Hill)

Machine learning as optimization

- A machine learning task
  - Given a set of training examples and a set of desired output values for each example
  - Induce a function that correctly maps the training examples to the desired output values
- The function induction generally involves some form of optimization: estimation of the parameters of the function

- What's the catch?
  - We don't really care how well we can do on the training set
  - That's because if we have a consistent set of examples and outputs (no identical examples mapped to different outputs), we can always find a perfect mapping function
  - We really only care about the performance of the induced function on new examples
  - We must consider the expressive power and inductive bias of the learning system

Machine Learning: Optimization, but we've got to choose the function and optimization method

- Expressive power
  - If we choose a functional form that can express a large space of complex functions, we may be able to fit the training data without any **generalization**
  - Typically, a function that has many parameters to estimate will be more expressive than one with fewer parameters

- Inductive bias
  - What types of solutions are your function and optimization scheme going to learn?
  - By making different choices of function or optimization method, we can make a significant impact on whether the bias of the learning system is well suited to producing good generalization

Representation of input data can have a big impact on your function and on inductive bias

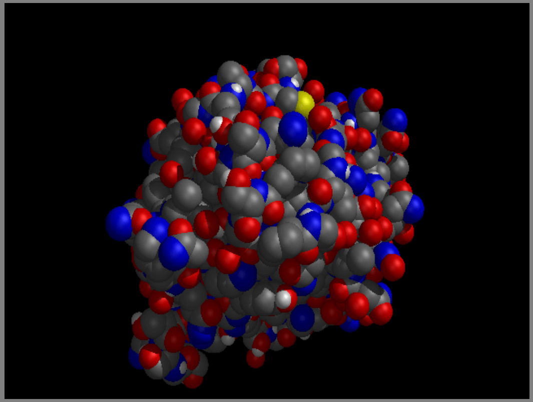
- Does the representation encode the object completely?
  - Enough for the function you care about?
  - Can it accommodate transformations and noise?
- Does the representation encode the object compactly?
  - Is there extraneous information?
  - Is there a well-defined measure of distance between representations that is correlated with outcome?
- If so, you're probably in good shape

```
ACCACCATGA ATCCACTCCT GATCCTTACC TTTGTGGCAG CTGCTCTTGC TGCCCCCTTT
GATGATGATG ACAAGATCGT TGGGGGCTAC AACTGTGAGG AGAATTCTGT CCCCTACCAG
GTGTCCCTGA ATTCTGGCTA CCACTTCTGT GGTGGCTCCC TCATCAACGA ACAGTGGGTG
GTATCAGCAG GCCACTGCTA CAAGTCCCGC ATCCAGGTGA GACTGGGAGA GCACAACATC
GAAGTCCTGG AGGGGAATGA GCAGTTCATC AATGCAGCCA AGATCATCCG CCACCCCCAA
TACGACAGGA AGACTCTGAA CAATGACATC ATGTTAATCA AGCTCTCCTC ACGTGCAGTA
ATCAACGCCC GCGTGTCCAC CATCTCTCTG CCCACCGCCC CTCCAGCCAC TGGCACGAAG
TGCCTCATCT CTGGCTGGGG CAACACTGCG AGCTCTGGCG CCGACTACCC AGACGAGCTG
CAGTGCCTGG ATGCTCCTGT GCTGAGCCAG GCTAAGTGTG AAGCCTCCTA CCCTGGAAAG
ATTACCAGCA ACATGTTCTG TGTGGGCTTC CTTGAGGGAG GCAAGGATTC ATGTCAGGGT
GATTCTGGTG GCCCTGTGGT CTGCAATGGA CAGCTCCAAG GAGTTGTCTC CTGGGGTGAT
GGCTGTGCCC AGAAGAACAA GCCTGGAGTC TACACCAAGG TCTACAACTA CGTGAAATGG
ATTAAGAACA CCATAGCTGC CAATAGCTAA AGCCCCCAGT ATCTCTTCAG TCTCTATACC
AATAAAGTGA CCCTGTTCTC
```

DNA sequence of Human Trypsin

```
MNPLLILTFV AAALAAPFDD DDKIVGGYNC EENSVPYQVS LNSGYHFCGG SLINEQWVVS
AGHCYKSRIQ VRLGEHNIEV LEGNEQFINA AKIIRHPQYD RKTLNNDIML IKLSSRAVIN
ARVSTISLPT APPATGTKCL ISGWGNTASS GADYPDELQC LDAPVLSQAK CEASYPGKIT
SNMFCVGFLE GGKDSCQGDS GGPVVCNGQL QGVVSWGDGC AQKNKPGVYT KVYNYVKWIK
NTIAANS
```

AA sequence of Human Trypsin



3D structure of Human Trypsin

Four things to do for a machine-learning task

- Choose a representation of your input data
- Choose a functional form that will map input examples to outputs
- Choose a method of optimization
- Train your system and evaluate performance
  - Ideal method: blind testing of a trained classifier on new data
  - Other method: cross-validation
    - Train your system on all but a subset of your data
    - Test on the held out subset
    - Repeat to get an estimate of predictive performance

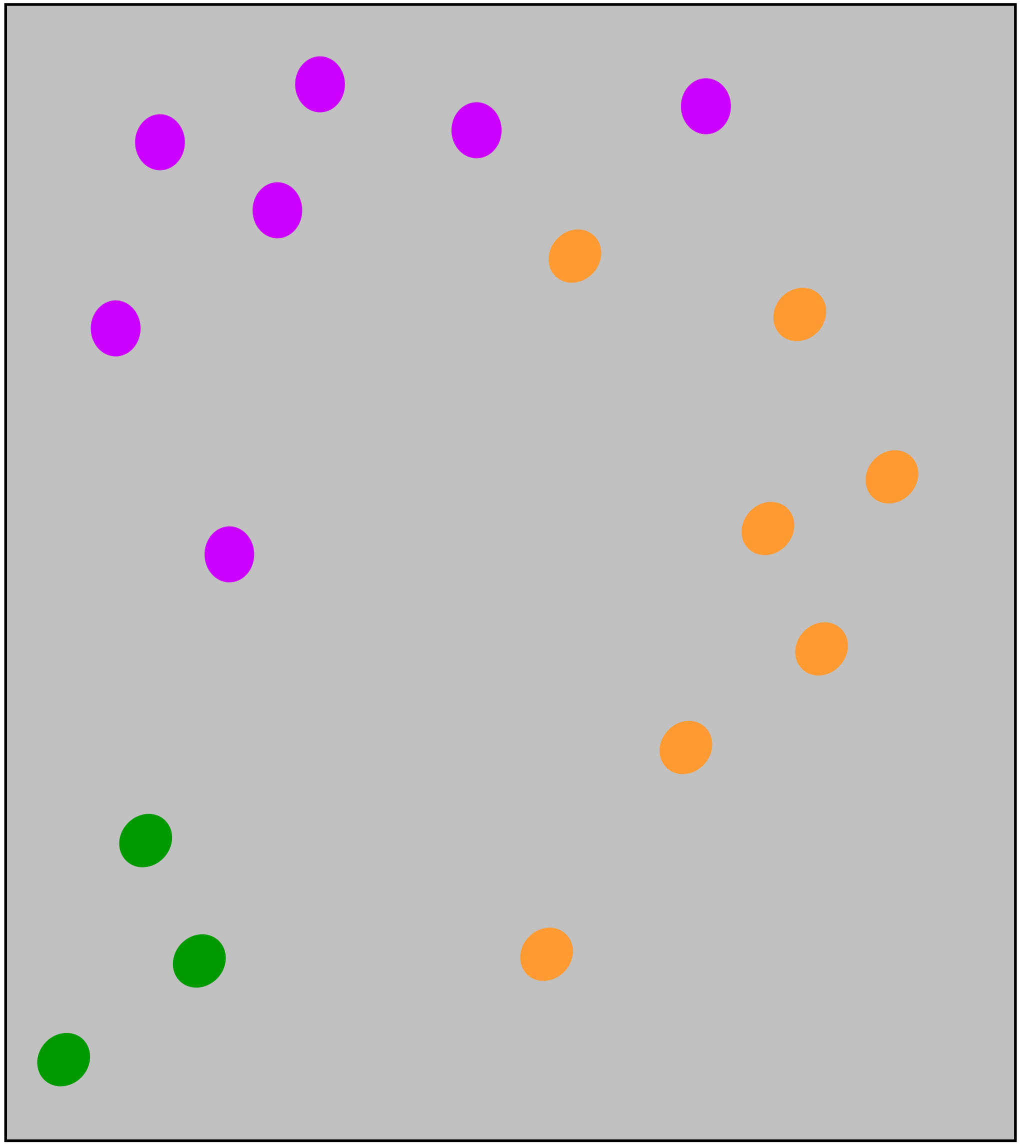
Symbolic versus numeric approaches

- Variety of “symbolic” approaches
  - Decision trees: ID3 etc…
  - [NOTE: Some decision-tree methods have numeric aspect.]
  - Concept learning
- Have the benefit of yielding “explainable” answers
- Tend to work well in areas where you already know the answer
- Interesting for cognitive science types and philosophers
- Not generally the most useful for biocomputing tasks

- Numeric approaches
  - Nearest neighbor classifiers
  - Artificial neural networks
  - Genetic algorithms
  - Kernel-based methods (support vector machines)
- Nearly every real-world application of machine learning to a problem (e.g. speech recognition) is based on a numeric approach

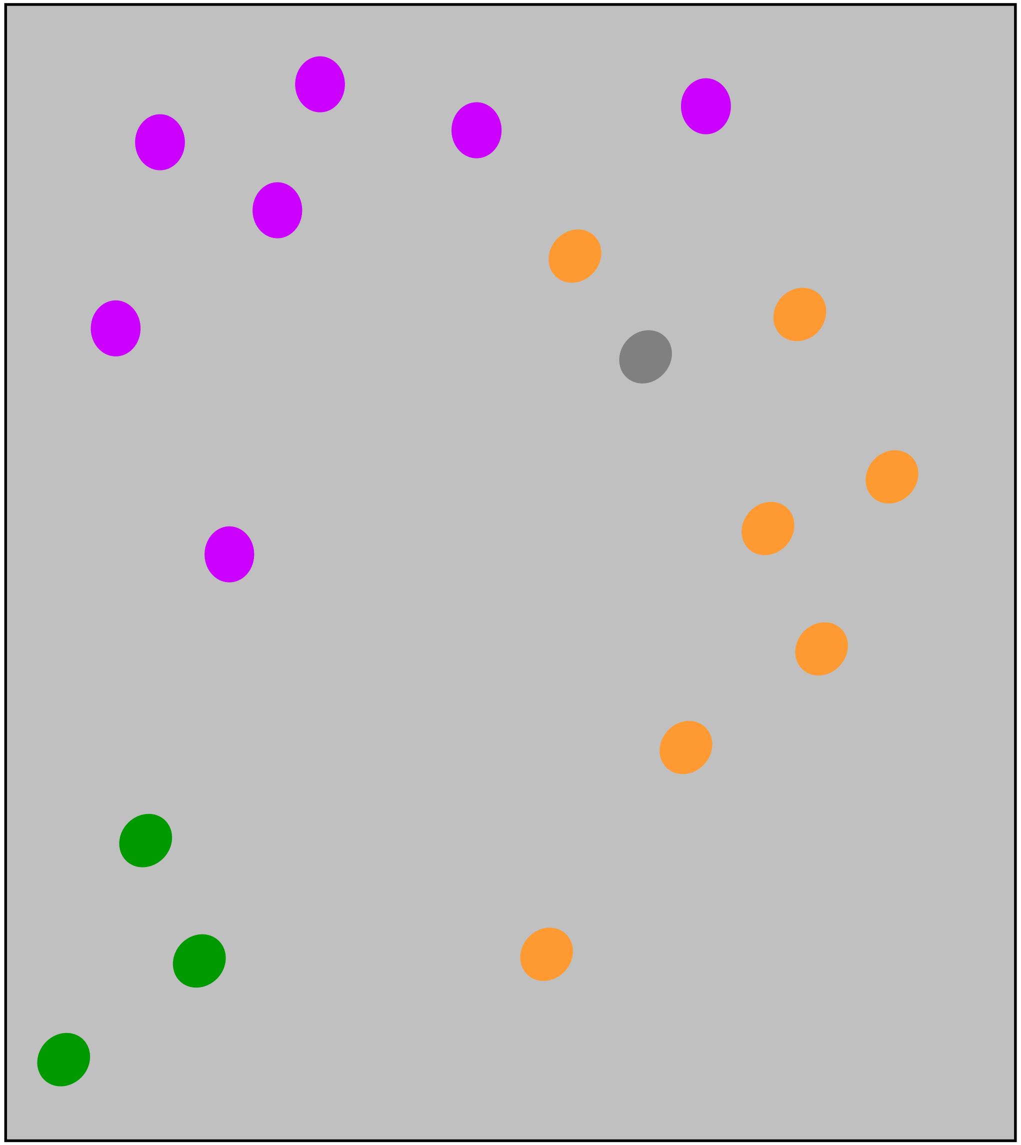
K nearest neighbors

- Data are represented as high-dimensional vectors
- KNN requires
  - Distance metric
  - Choice of K
  - Potentially a choice of element weighting in the vectors
- Given a new example
  - Compute distances to each known example
  - Choose class of most popular



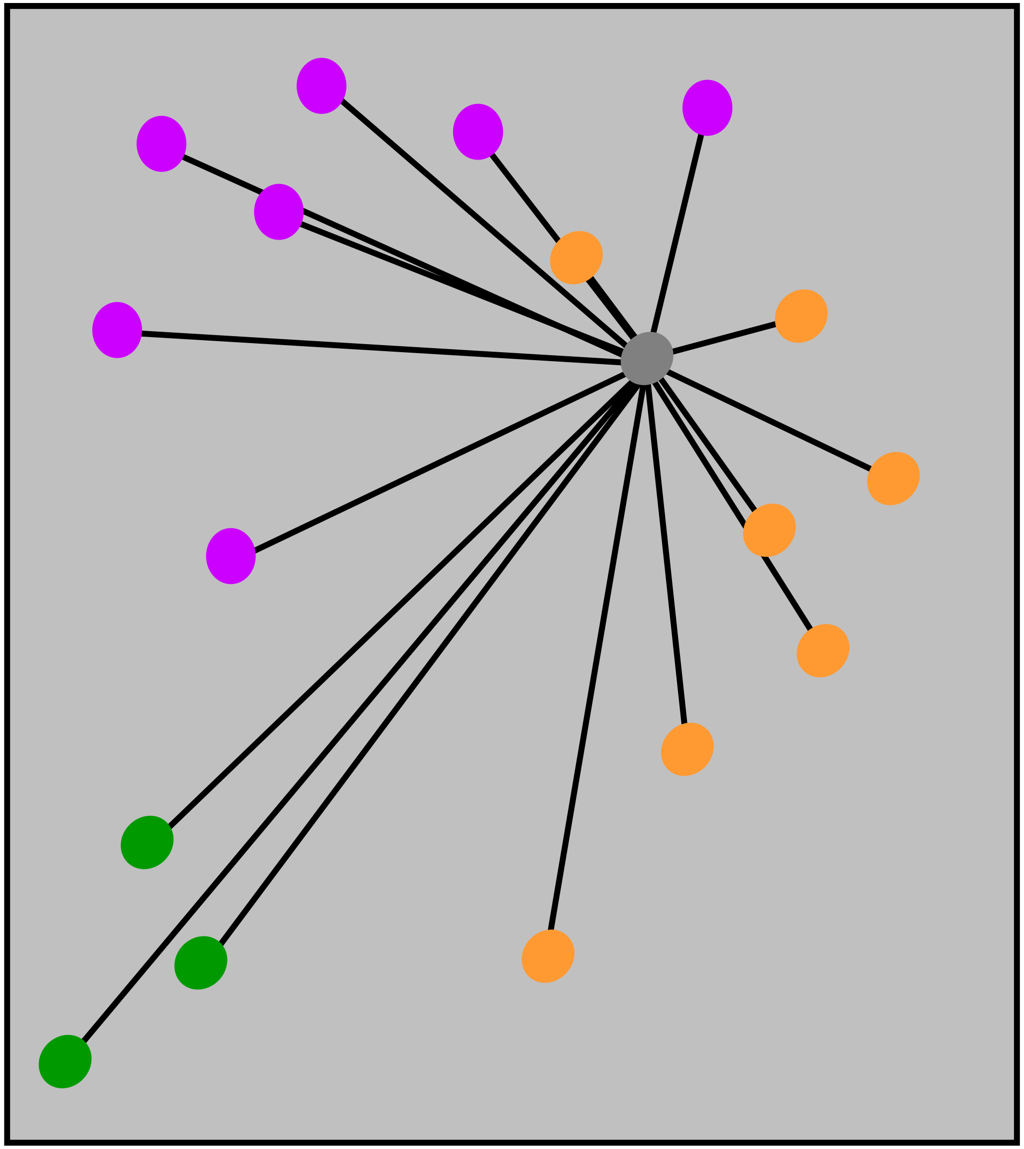
K nearest neighbors

- New item



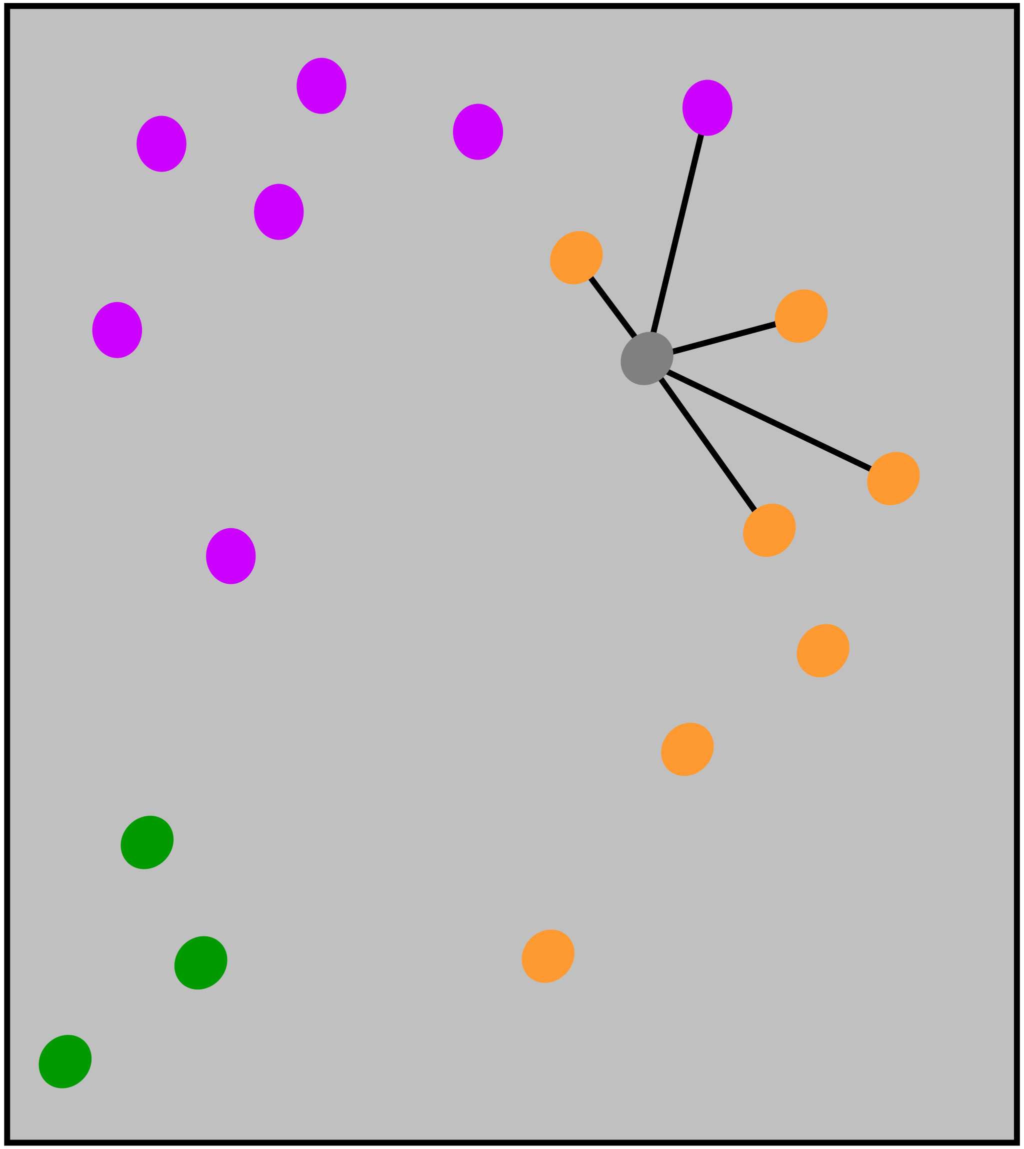
K nearest neighbors

- New item
  - Compute distances



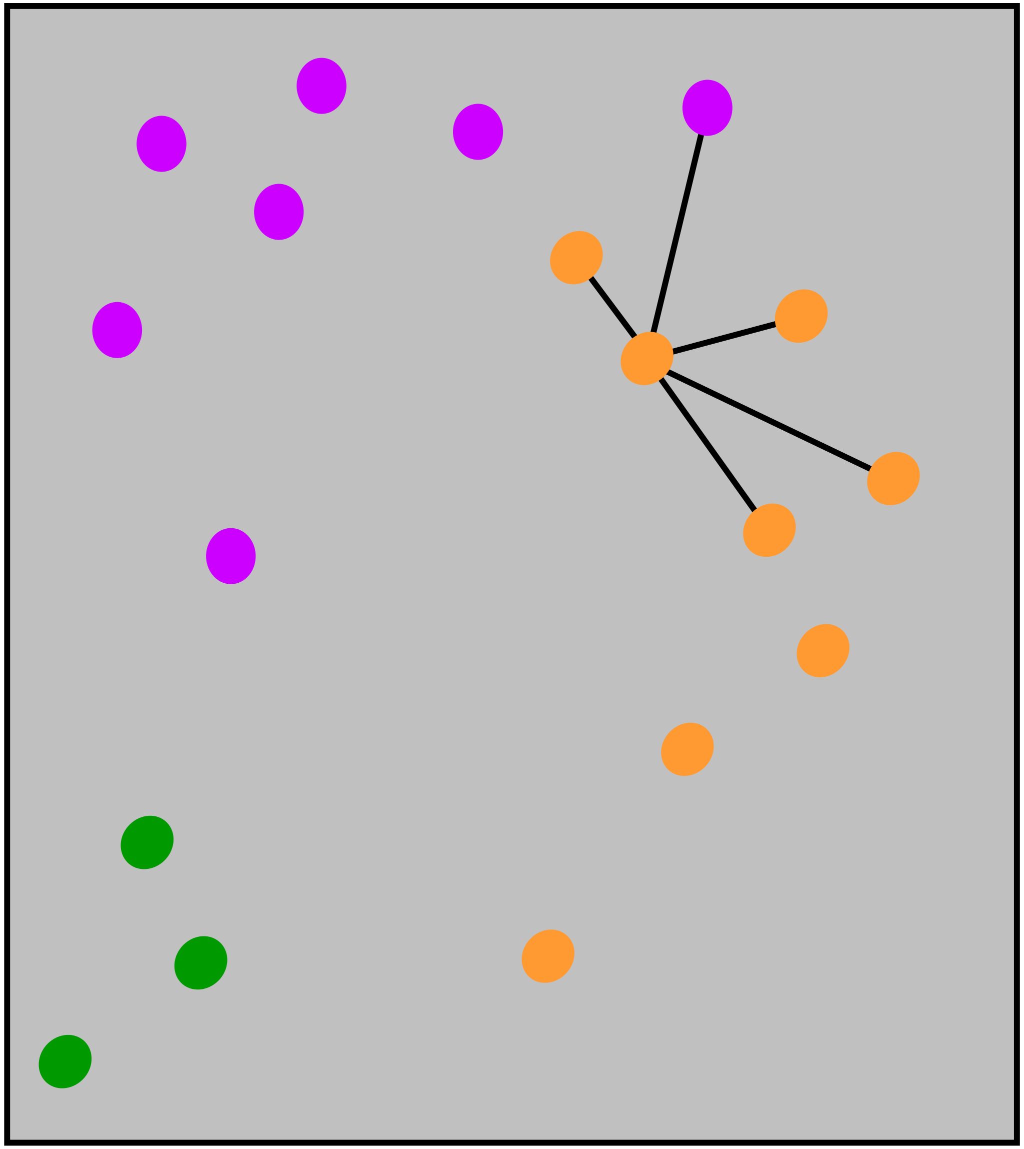
K nearest neighbors

- New item
  - Compute distances
  - Pick K best distances



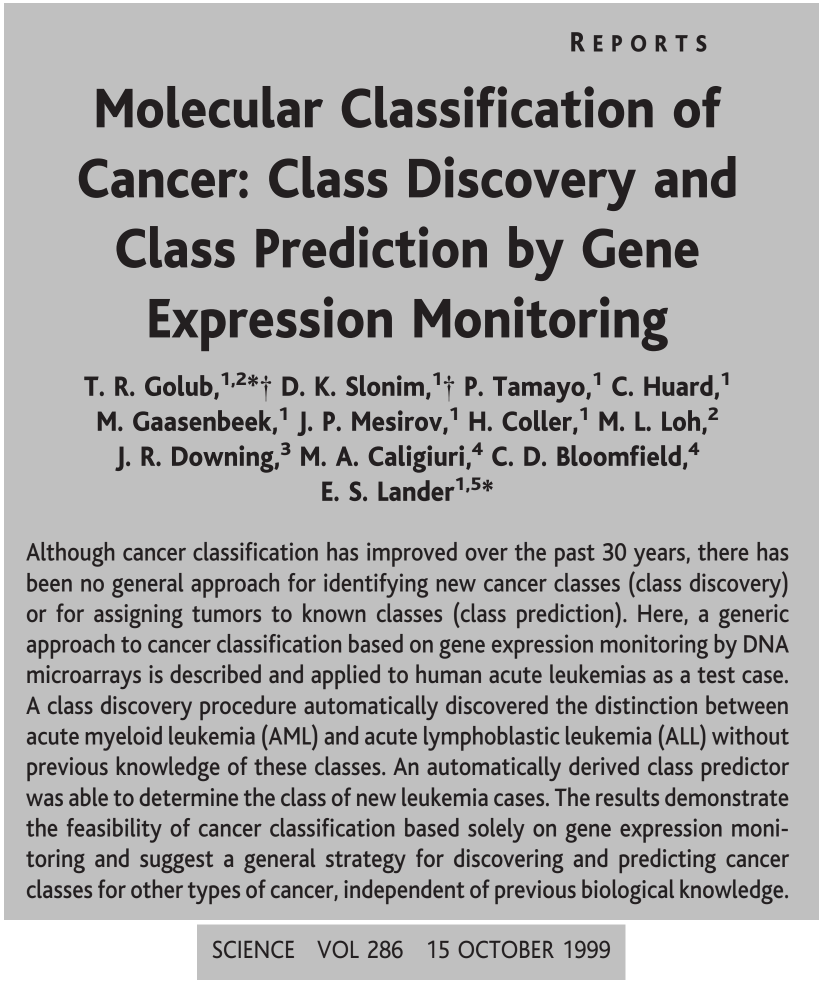
K nearest neighbors

- New item
  - Compute distances
  - Pick K best distances
  - Assign class to new example



Expression arrays: Lymphoblastic versus myeloid leukemia

- Lander data
  - 6817 unique genes
  - Acute Lymphoblastic Leukemia and Acute Myeloid Leukemia (ALL and AML) samples
  - RNA quantified by Affymax oligo-technology
  - 38 training cases (27 ALL, 11 AML)
  - 34 testing cases (20/14)
- Can we classify ALL/AML?
  - Use clustering to see if distance metric and variable selection seems to work
  - Apply formal blind test based on model from the 38 training cases



REPORTS

Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring

T. R. Golub,[1,2*†] D. K. Slonim,[1†] P. Tamayo,[1] C. Huard,[1] M. Gaasenbeek,[1] J. P. Mesirov,[1] H. Coller,[1] M. L. Loh,[2] J. R. Downing,[3] M. A. Caligiuri,[4] C. D. Bloomfield,[4] E. S. Lander[1,5*]

Although cancer classification has improved over the past 30 years, there has been no general approach for identifying new cancer classes (class discovery) or for assigning tumors to known classes (class prediction). Here, a generic approach to cancer classification based on gene expression monitoring by DNA microarrays is described and applied to human acute leukemias as a test case. A class discovery procedure automatically discovered the distinction between acute myeloid leukemia (AML) and acute lymphoblastic leukemia (ALL) without previous knowledge of these classes. An automatically derived class predictor was able to determine the class of new leukemia cases. The results demonstrate the feasibility of cancer classification based solely on gene expression monitoring and suggest a general strategy for discovering and predicting cancer classes for other types of cancer, independent of previous biological knowledge.

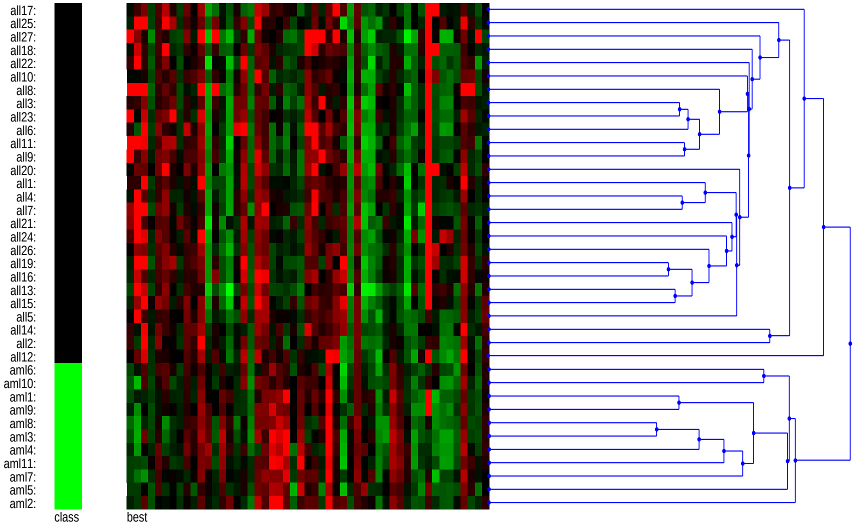
SCIENCE VOL 286 15 OCTOBER 1999

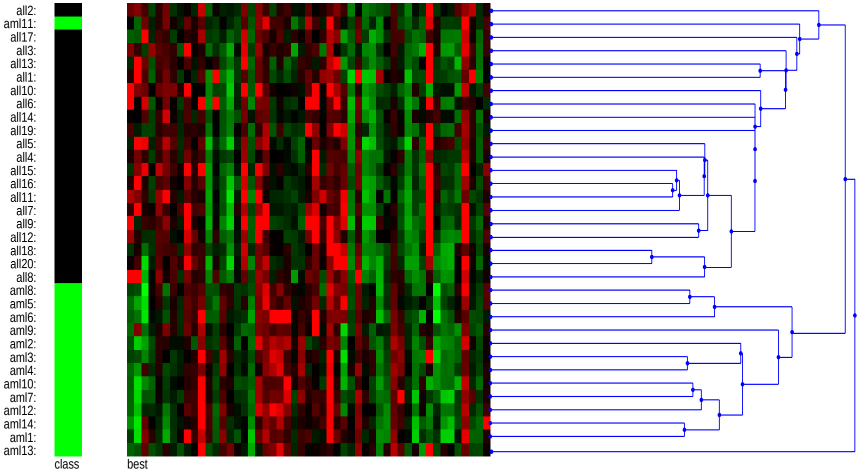
Four things to do for a machine-learning task

- Choose a representation of your input data
  - We’re using vectors of expression data
- Choose a functional form that will map input examples to outputs
  - We have a winner-take-all function that returns a class label given an input vector along with training vectors and classes
- Choose a method of optimization
  - We will use binary variable selection based on a T-test
  - We will try various distance metrics, sizes of K, and number of variables
- Train your system and evaluate performance
  - In this case, we have a training set of 38 cases and a blind test set of 34
  - We will evaluate performance on the 34 test cases

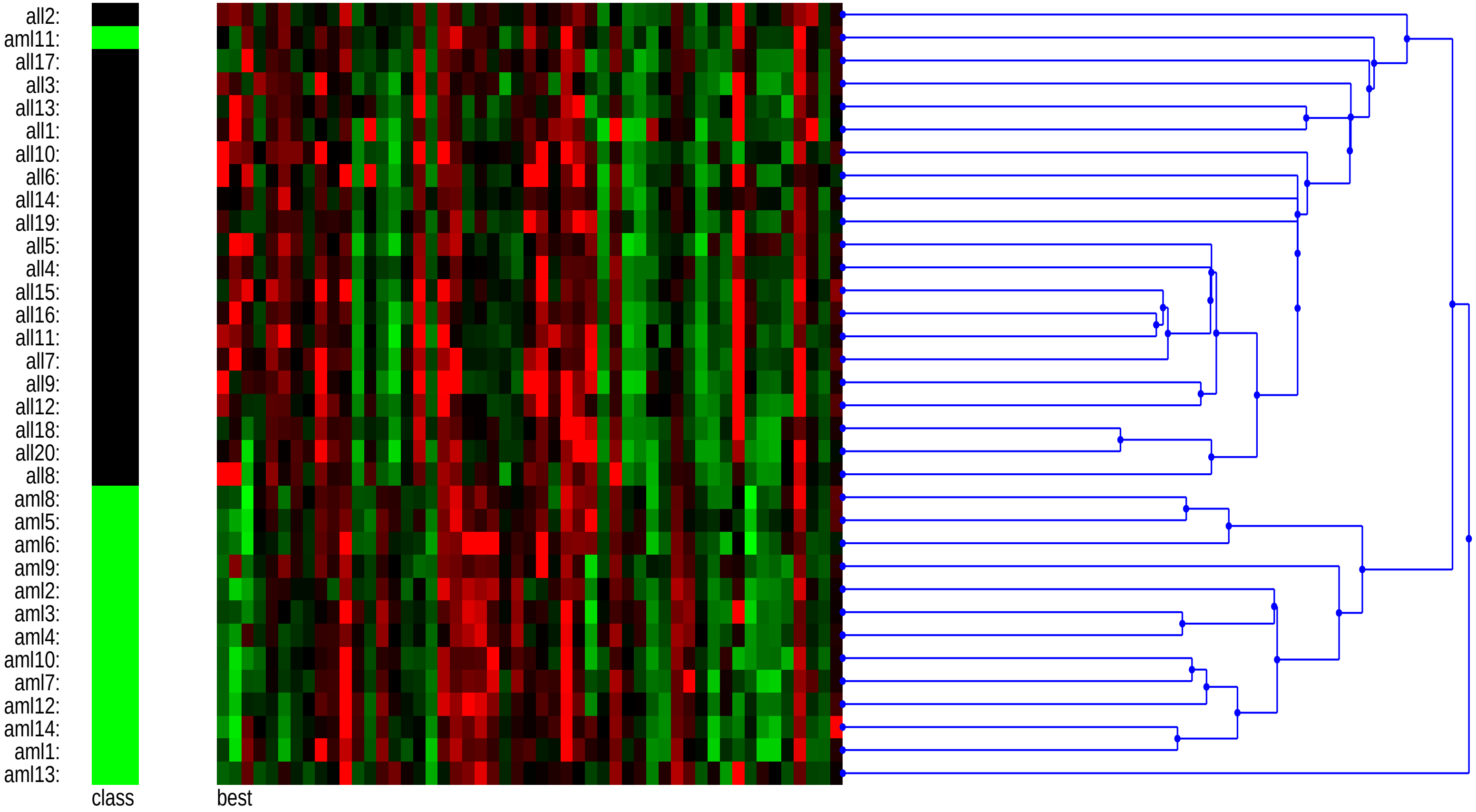
We can use clustering to examine the interaction of variable selection with distance metric

- We know that most of the 6817 genes are not informative as to ALL/AML
- We choose the top N based on a T-test
- We decide that Euclidean distance is a reasonable thing to try
- Taking the top 50 genes by T-test from the training set, we get nearly perfect clustering of both the training and test sets



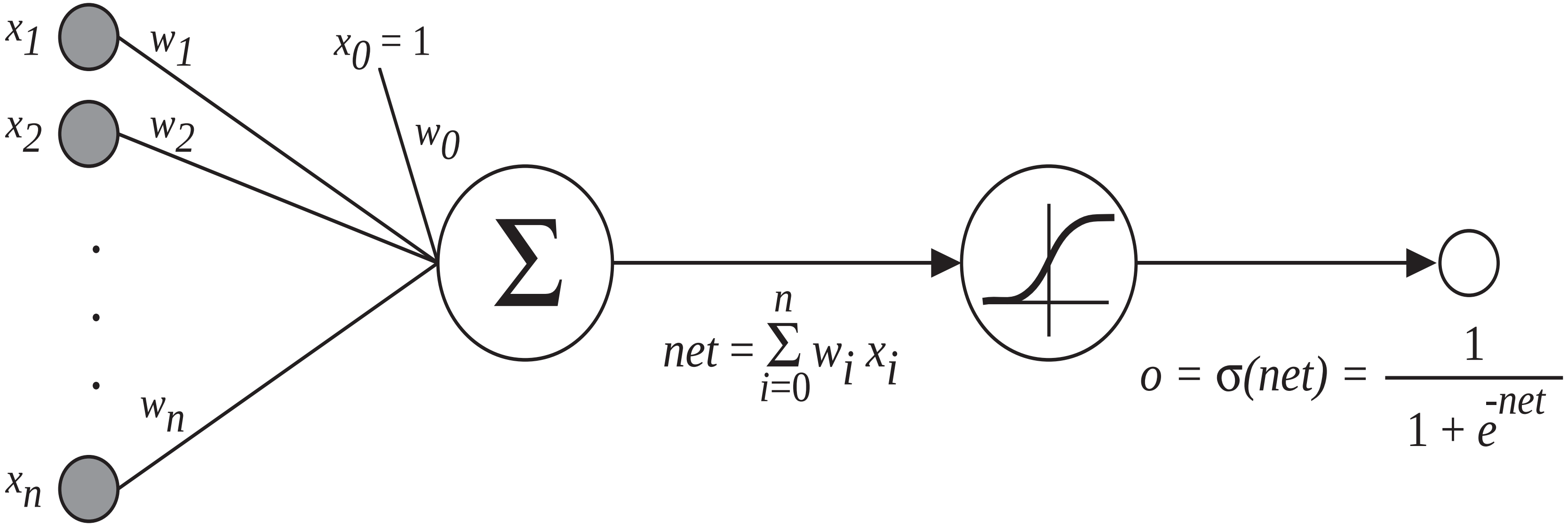


KNN works very well on this data set



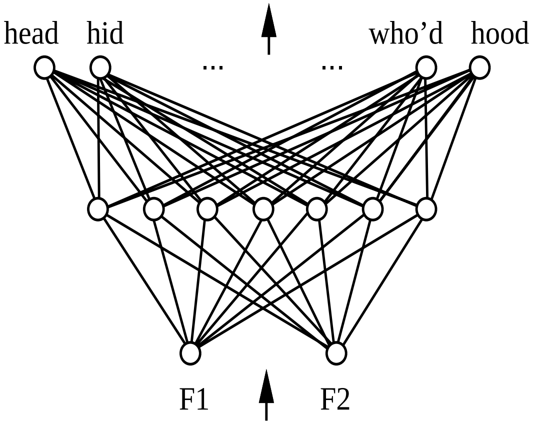
Under many choices of K, gene selection sizes of 20 to 2000, with different distance metrics, we see that the only consistently misclassified sample is AML11. The range of errors is 1-4 under all conditions.

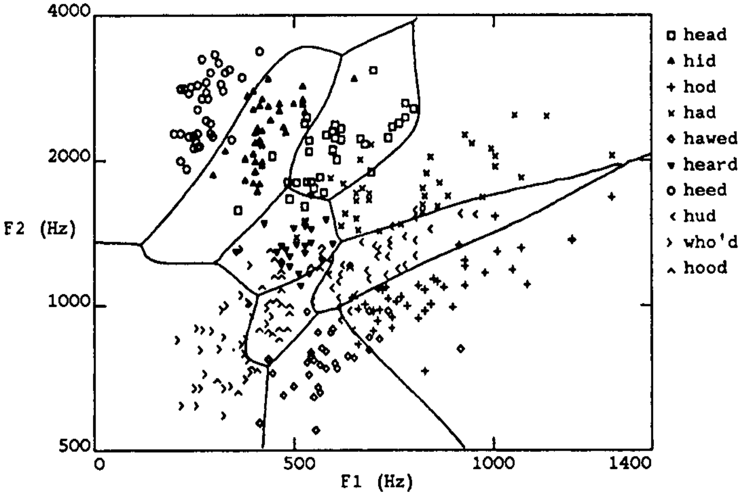
Non-linear learning systems: The sigmoid unit is not the only game in town



$$net = \sum_{i=0}^{n} w_i x_i$$

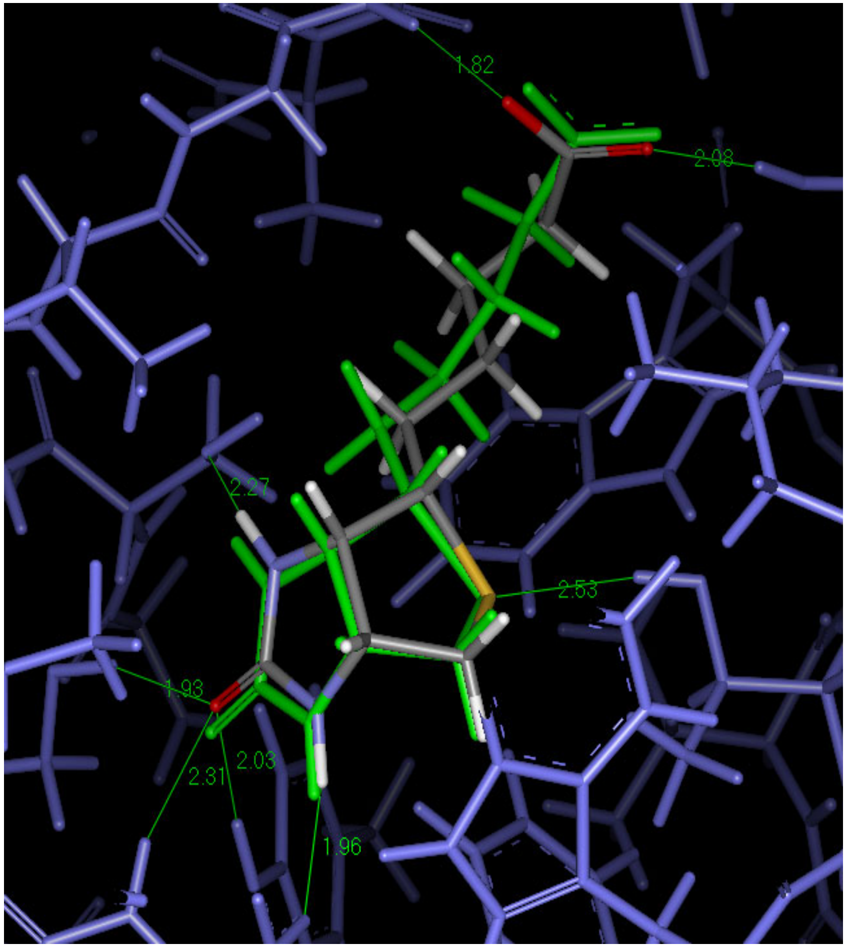
$$o = \sigma(net) = \frac{1}{1 + e^{-net}}$$





Flexible Molecular Docking

- Problem definition
  - Given a protein crystal structure
  - Given a small molecule or many
  - Extremize the value of a **scoring function** by varying molecular pose
- Requirements
  - Predict correct rank-order of binding affinity within 1 protein
  - Reject false positives
  - Generate correct molecular poses at extrema of function
  - Be very fast (86,400s in 1 day)

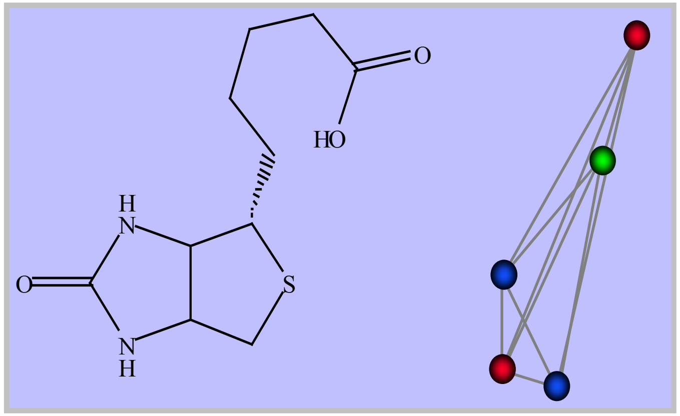


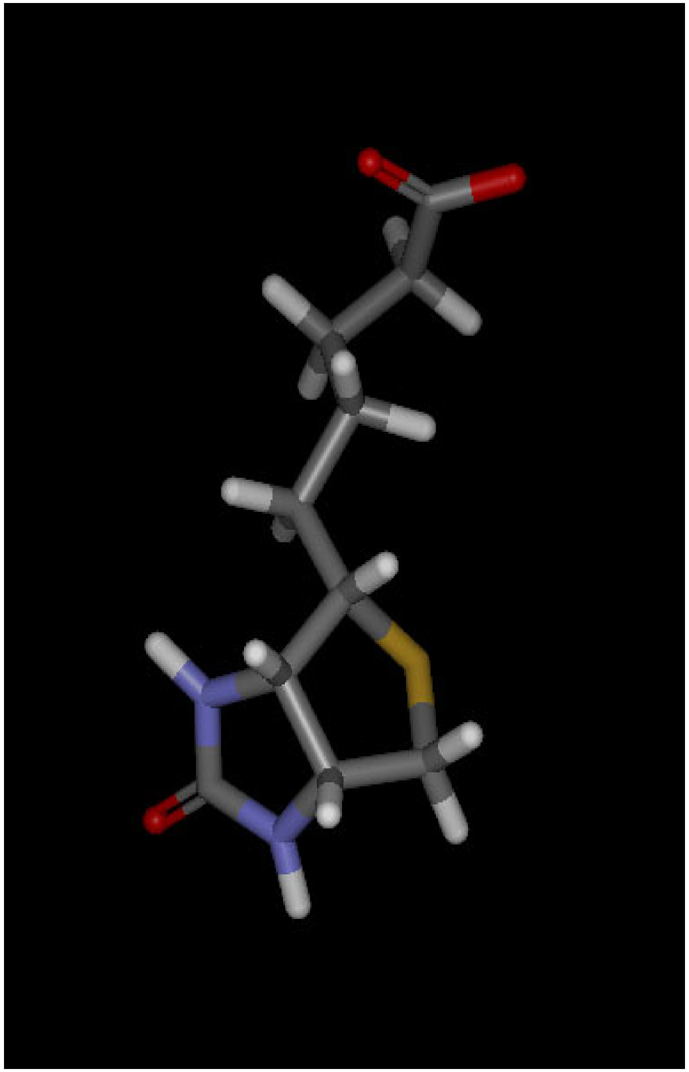
Four things to do for a machine-learning task

- Choose a representation of your input data
  - We need to pick an input representation that includes information relevant to protein ligand binding energies
- Choose a functional form that will map input examples to outputs
  - We're going to predict a binding energy in –log(K_d) units
- Choose a method of optimization
  - We will use gradient-descent to optimize the parameters of the function
  - We will try various schemes to improve performance
- Train your system and evaluate performance
  - In this case, we have a training set of 34 co-crystal structures
  - We will evaluate performance using cross-validation

Clearly the problem must be at least 3D

- The 2D structure of a ligand gives little useful information
  - Molecular comparisons can be made, but only within-scaffold
  - Molecular interactions are not usefully represented
  - Sparse 3D representations are of limited utility as well

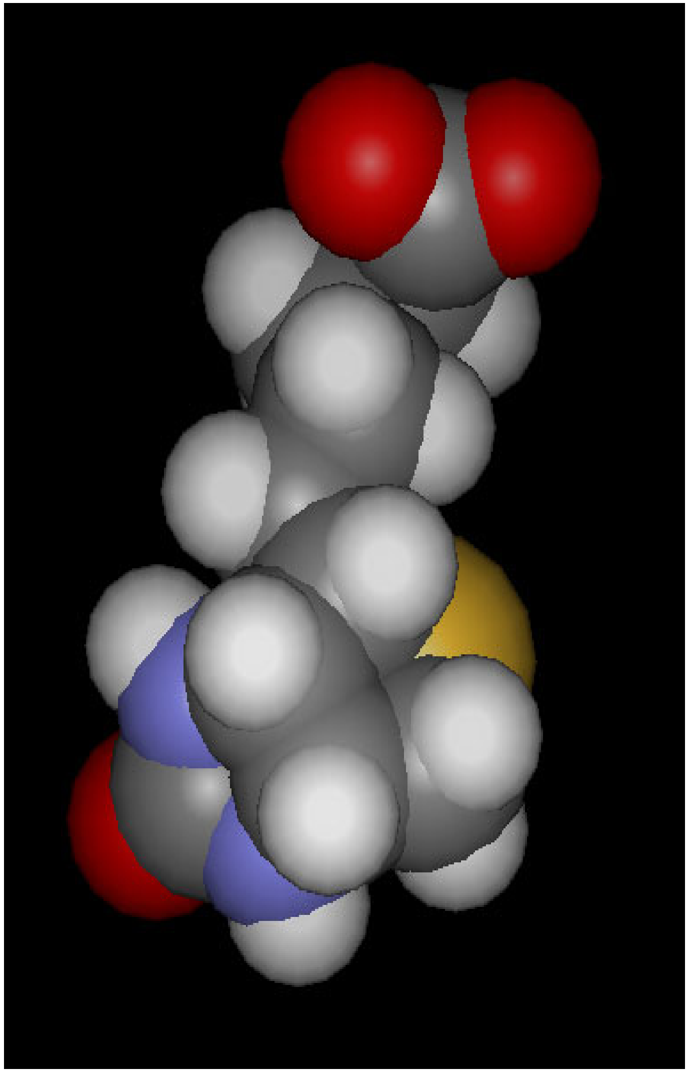




We want to approximate a 3D surface

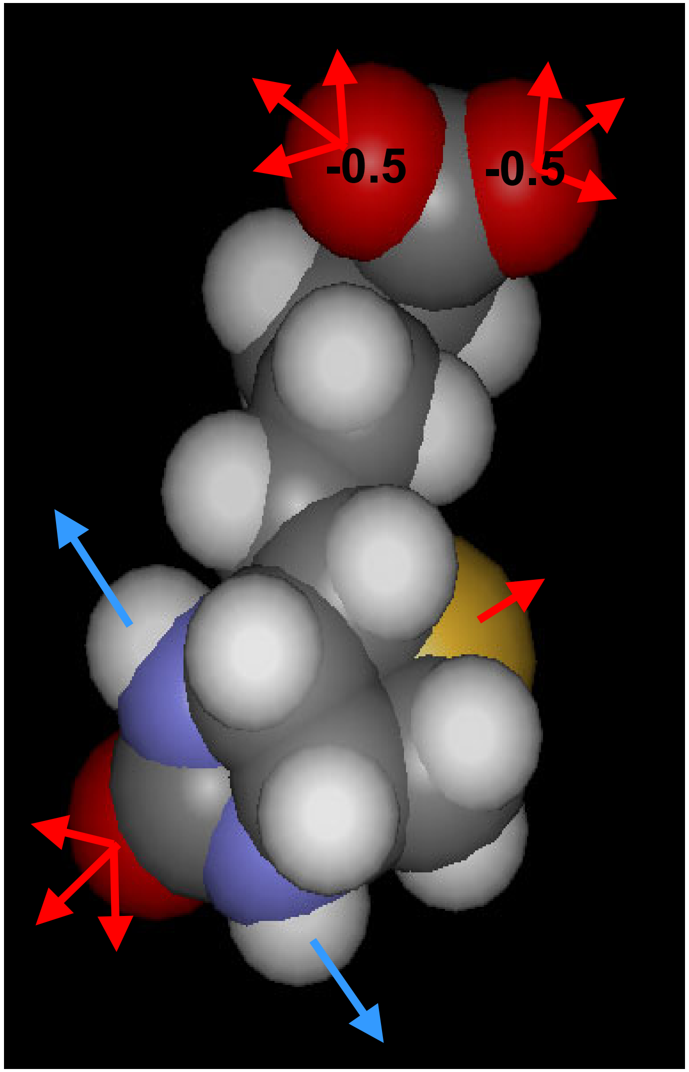
- We will approximate molecules as collections of spheres with fixed radii
  - H = 1.2 C = 1.6 N = 1.5
  - O = 1.4 S = 1.95 P = 1.9
  - F = 1.35 Cl = 1.8 Br = 1.95
  - I = 2.15

- Hard plastic balls is not necessarily a reasonable representation



Molecules have polar contact preferences

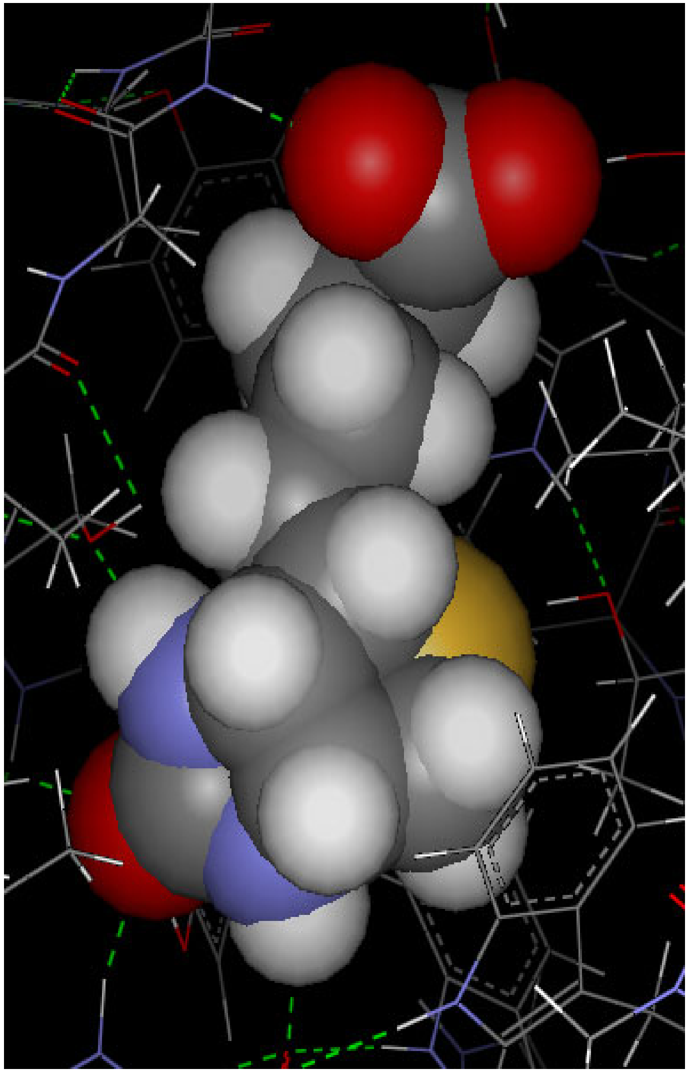
- We will mark atoms as follows
  - Polar positive:
    - H-bond donors
    - Formally positively charged atoms
  - Polar negative
    - H-bond acceptors
    - Formally negatively charged atoms
  - Polar atoms have directional preferences
    - Defined on a local coordinate system
    - Up to three preference vectors



That’s it. We can now compute things about molecular interactions.

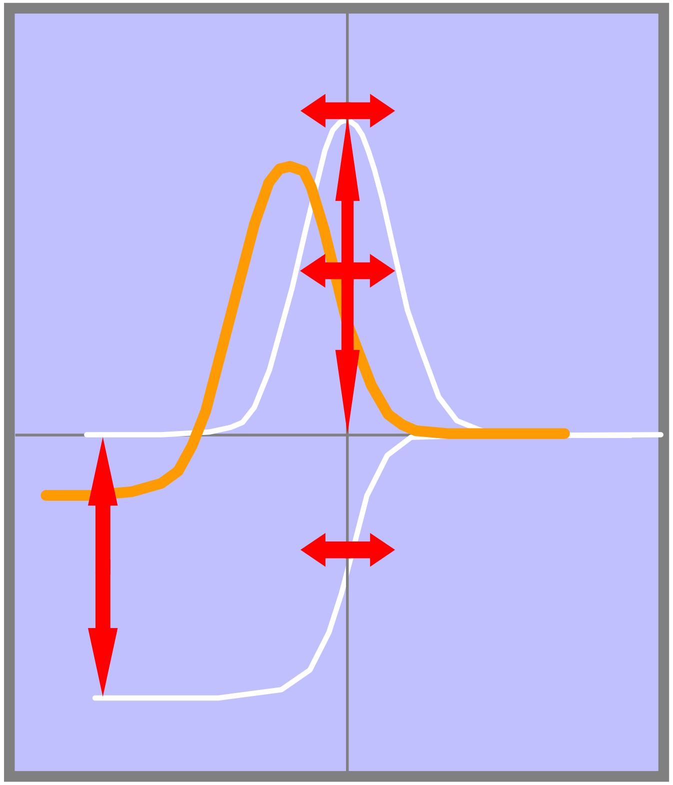
- Biotin/Streptavidin
  - $K_d = 10^{-13.4}$
  - Complete set of hydrogen-bonding or salt-bridging interactions
  - Extensive hydrophobic packing

- We will construct a function, which, given these representations, yields good enough estimates of binding affinities that it is useful for docking.



We can induce a soft function that predicts binding affinities using our "hard plastic" representation

- Co-crystal data used to tune the function
  - 34 structures, ranging from 10-3 to 10-14 in Kd
  - 16 different proteins (heavy on enzymes)
- Linear combination of non-linear functions of protein-ligand atomic surface distances
  - Steric term: Gaussian + sigmoidal
  - Polar term: Gaussian + sigmoidal
  - Polar term is influenced by directionality and formal charge
  - Entropic term: # rot bonds, log(MW)
  - Solvation term: "missed h-bonds"



This looks *nothing* like a molecular mechanics non-bonded force-field.

Attempt #1: Static conformations

- Take each crystal structure exactly as it was solved
- Train the function to optimally fit all 34 examples, optimizing the alignment of each example on each iteration

	Mean error	**RMSD**
Fit to data:	1.1 log(K_d)	1.4 Å
If we now optimize the conformations		
	1.5	2.1

- This is a problem since we don’t know the correct conformation when we dock a new molecule!

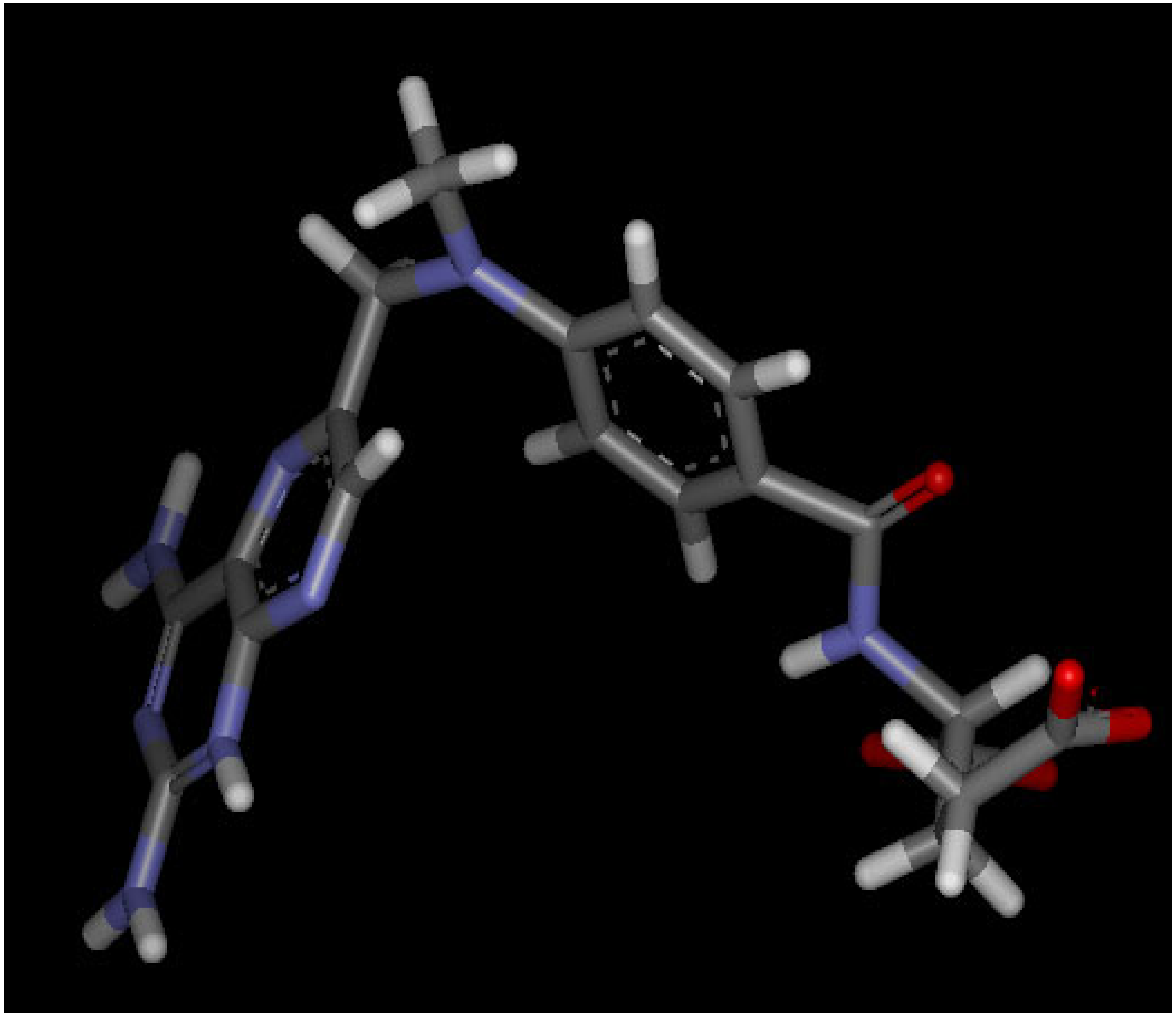
Attempt #2: Dynamic conformations

- Take each crystal structure exactly as it was solved
- Train the function to optimally fit all 34 examples, optimizing the alignment of each example on each iteration **and optimizing the conformation**

	Mean error	RMSD
Fit to data:	0.97 $\log(K_d)$	1.2 Å

- We are in better shape
- We've embedded an aspect of the application of the function into its training: the maximum score under our optimization strategy must be the correct

Frequently, molecules embed strain into their xtal conformations



Attempt #3: Dynamic conformations, beginning from minimized conformations

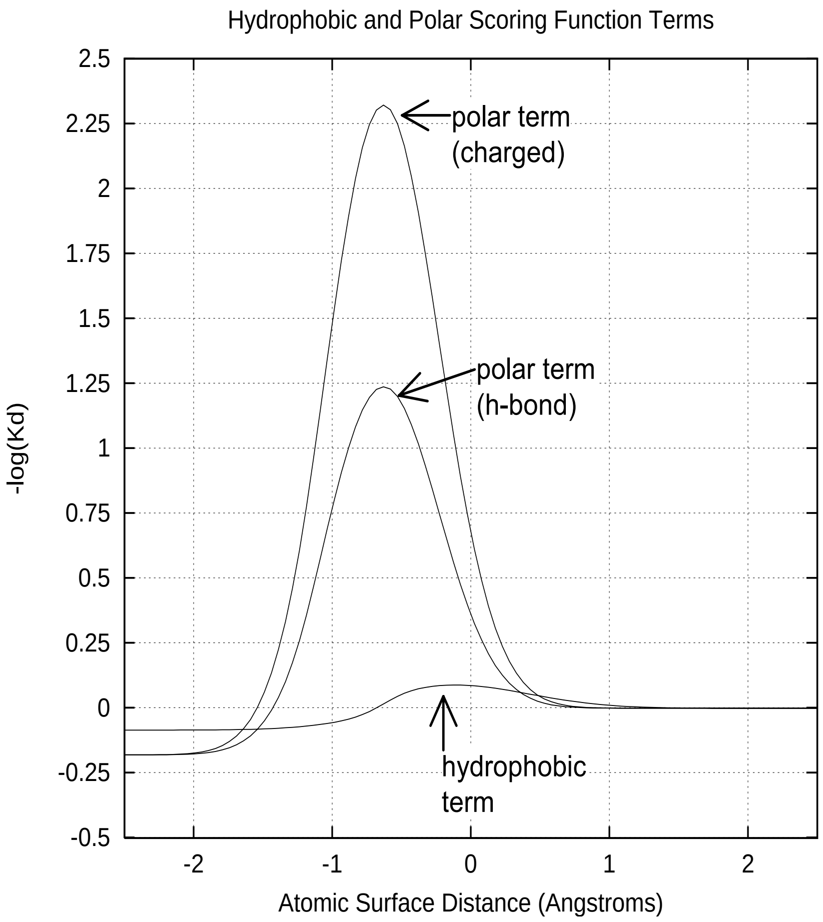
- Take each crystal structure exactly as it was solved
- Minimize (in a vacuum) the ligand
  - Many co-crystals strain ligands by bending things that don’t bend
  - We’ll always be starting from low-energy molecule
- Train the function to optimally fit all 34 examples, optimizing the alignment of each example on each iteration **and optimizing the conformation**

	Mean error	RMSD
Fit to data:	0.72 log(K_d)	0.85Å

- We are in very good shape now
- We’ve embedded another aspect of the application of the function into its training

Empirically derived scoring function learns appropriate magnitudes and geometries

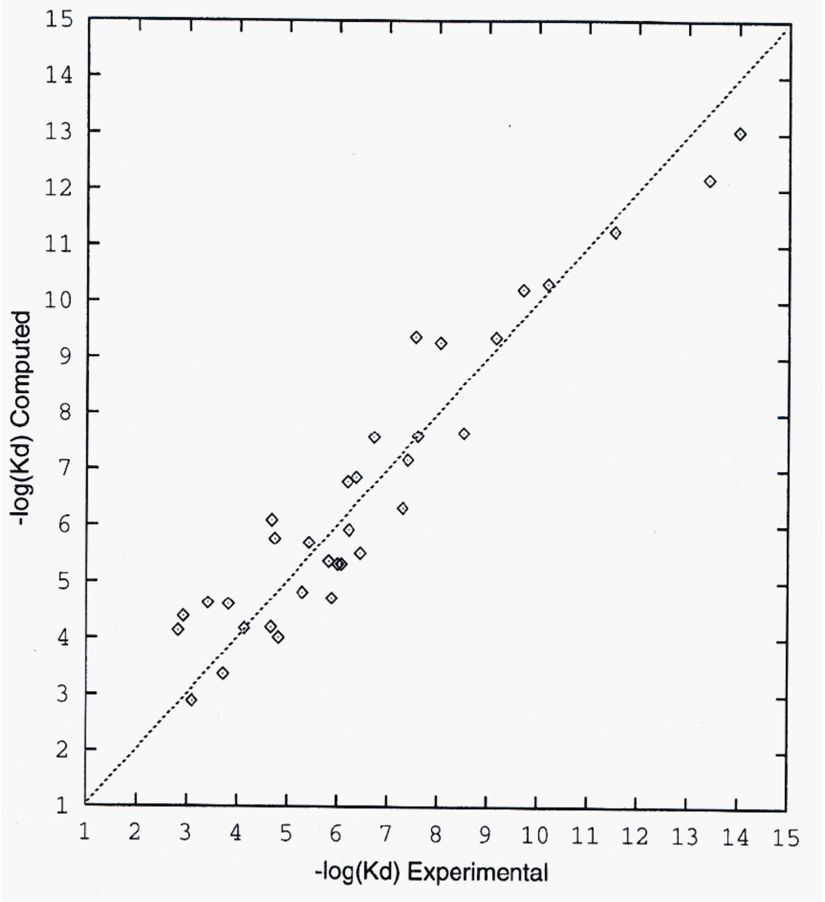
- Steric and polar terms dominate
- Steric term
  - Peaks at about 0.1 log units per ideal contact
  - Ends up dominating the energy function because there are so many such contacts
- Polar term
  - H-bond: peaks at 1.25 log units (H-O distance of 2.0Å)
  - Formal charge scales the polar term: 2.25 units for a tert-amine proton to a carboxylate oxygen



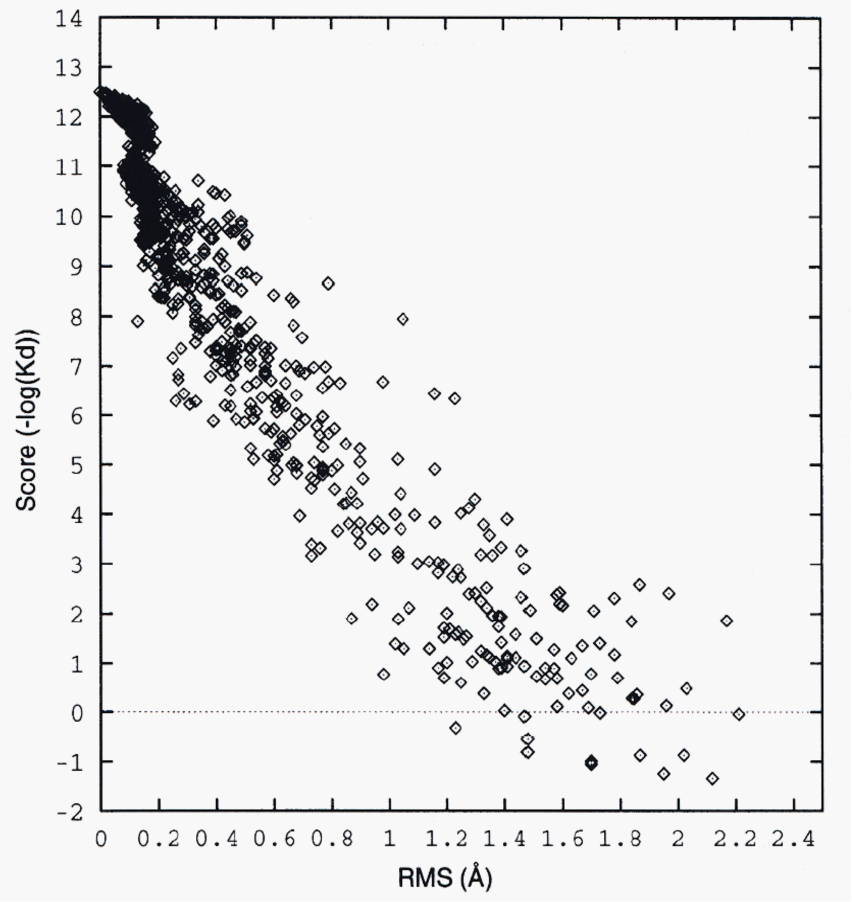
Scoring function wrinkle summary

- Version 1: Estimate parameters from native crystal structures
  - We are just treating the crystal structures as being static
  - We ignore the fact that when we dock, we seek to extremize the scoring function
  - We do apparently well (mean error: 1.15 log units)
- Version 1: We optimize the poses based on the scoring function
  - Our mean error increases to 1.51!
  - We should embed the optimization constraint into the parameter estimation
- Version 2: Now we optimize poses online during parameter estimation
  - Mean error (with pose optimization) drops to 0.97
  - We observe that some ligands that are overpredicted have strained configurations
- Final function F: We minimize structures prior to parameter estimation
  - Mean error: 0.72 log units
  - Cross-validated mean error: 1.0 log units
  - Now we are done.

Scoring function predicts affinities well
Also has nice sharp maximum at correct pose



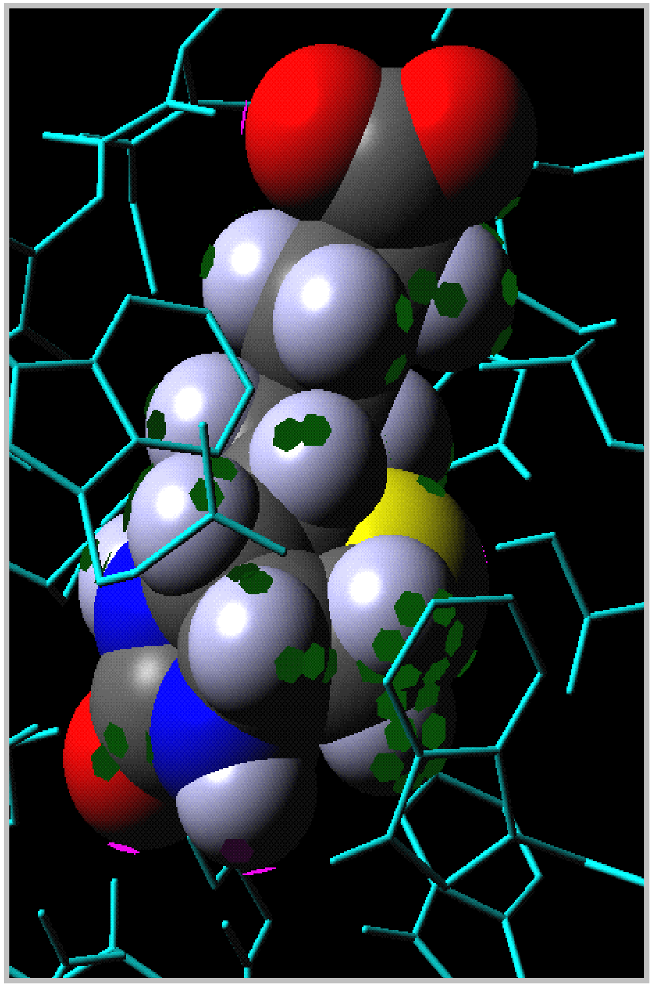
Plot of computed versus experimental pK_d.

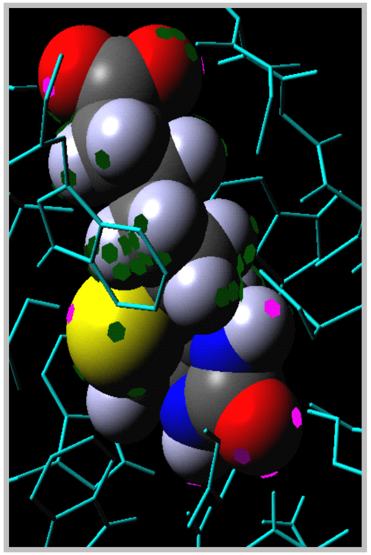


Scores of biotin perturbed from its optimal pose within streptavidin

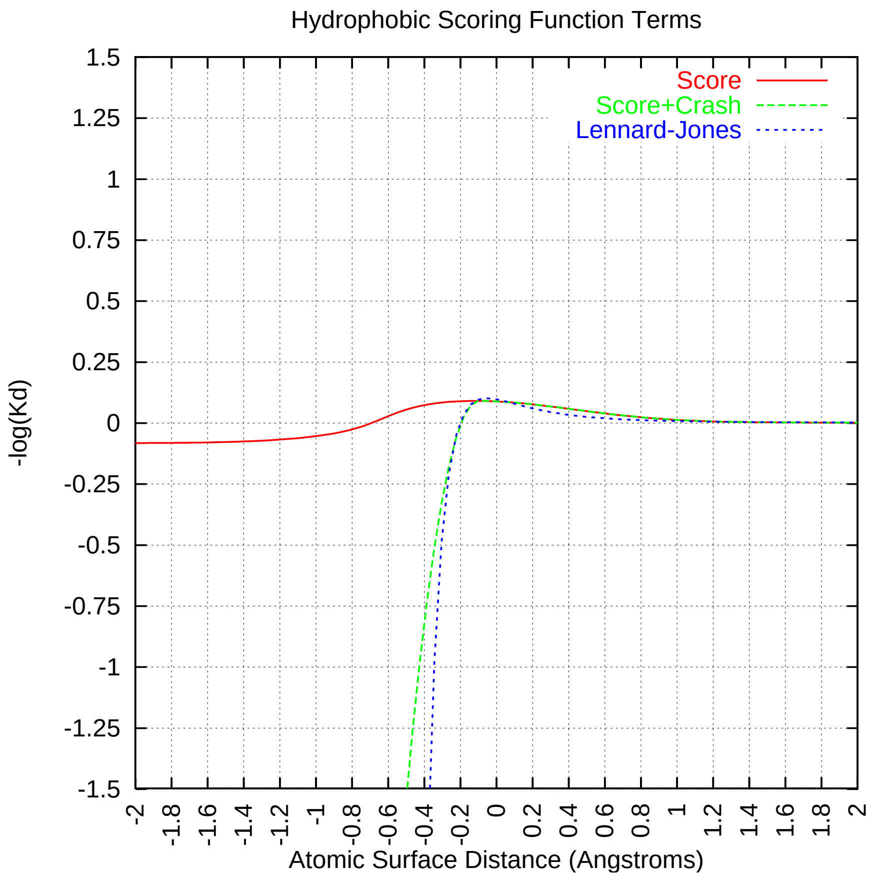
Steric and polar terms dominate

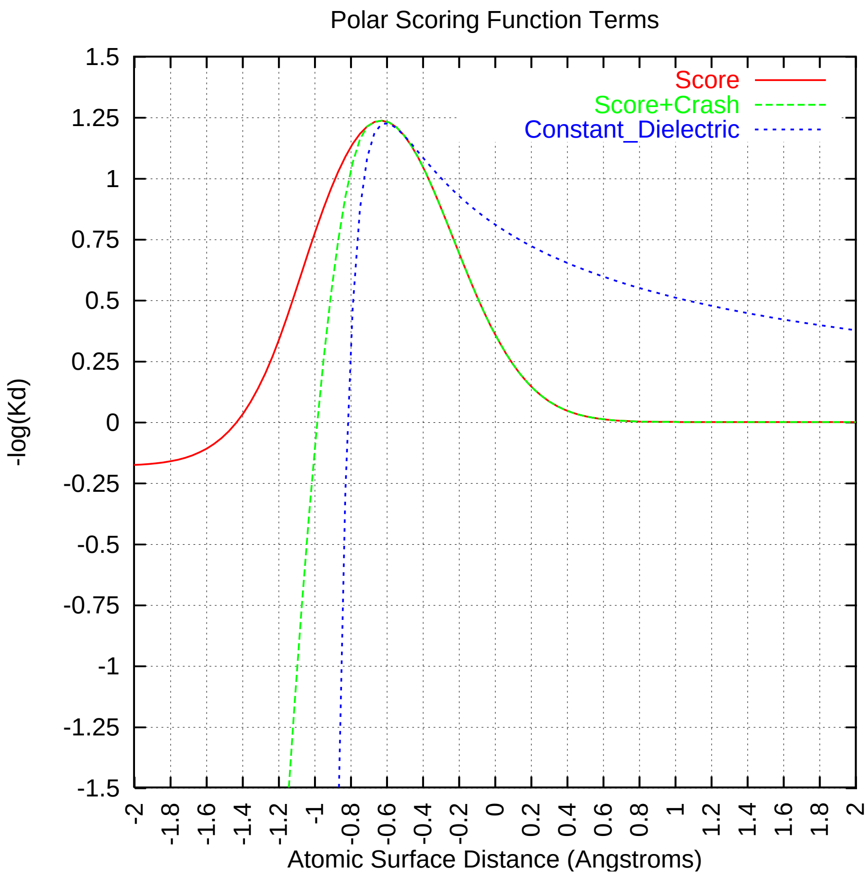
- Breakdown of the scoring function over all complexes
  - Steric: 44%
  - Polar: 26%
  - Solvation: 5%
  - Entropy: 25%
- Streptavidin/biotin breakdown
  - Computed affinity: 12.5
  - Actual: 13.4
  - Steric: 7.6
  - Polar: 9.0
  - Entropy: -1.1 + -2.5
  - Solvation: -0.5





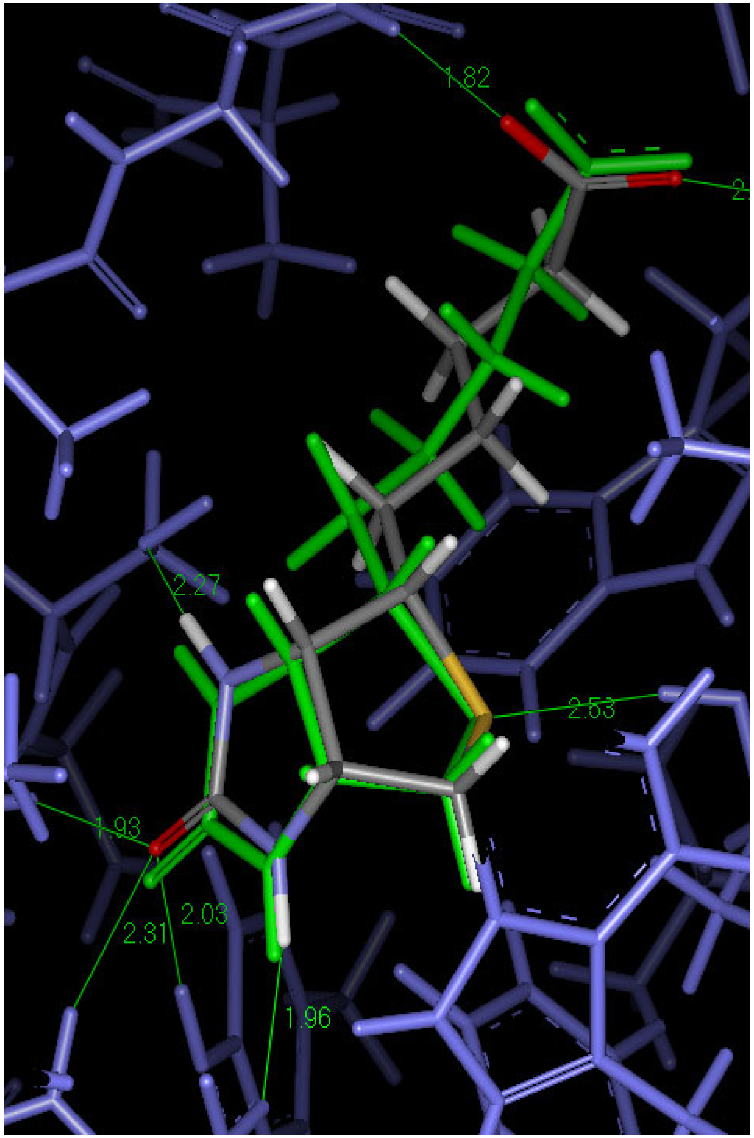
How is the learned scoring function different from physics-based derivations?





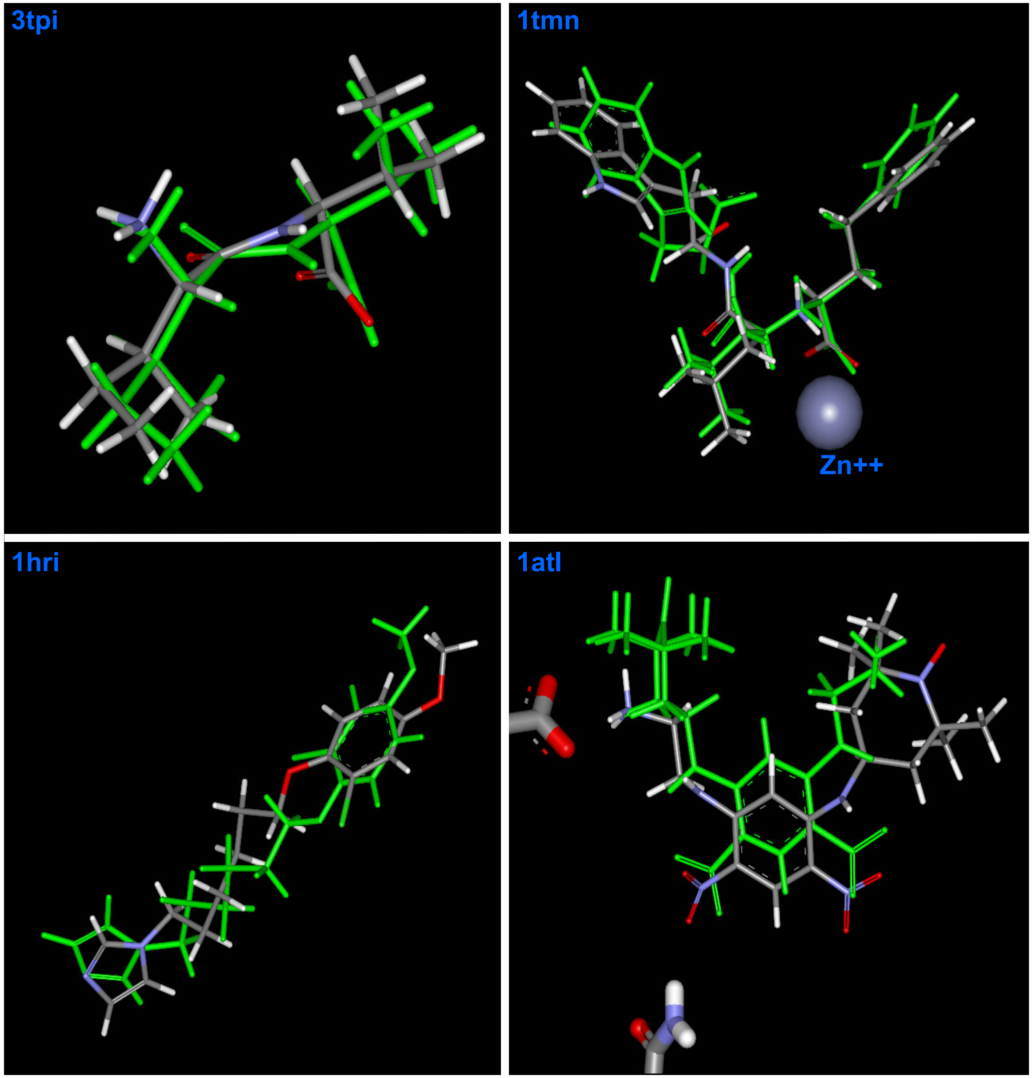
So now we have a scoring function. We can dock, given a search engine.

- We talked about search in the complexity lecture
  - Divide and conquer to address the conformational issue
  - Use molecular similarity trick to generate putative alignments
- Docking accuracy
  - GOLD validation set
  - 81 complexes
    - 15 or fewer rotatable bonds
    - No covalent attachments
  - Considered docking accuracy
    - Best docked pose by rms
    - RMS of top scoring pose



Docking is largely accurate
Scoring function recognizes the correct poses

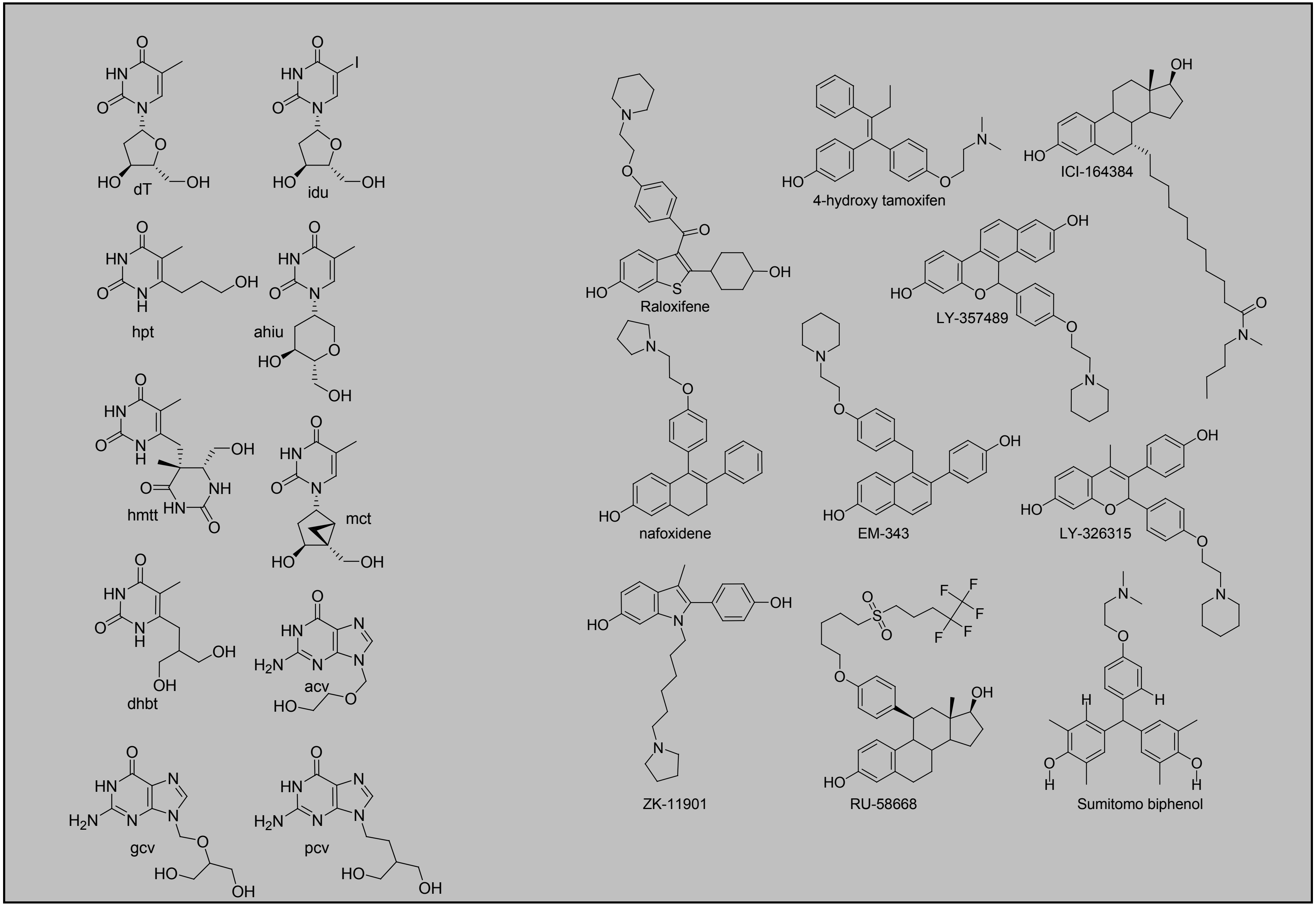
- 81 complex data set
  - 94% of cases, rmsd < 2.5Å for most accurate pose
  - If a good pose was generated, it was the top scoring 86% of the time
- Performance is comparable with GOLD
- But how will this work in screening large databases of molecules?



Screening utility depends on a very low false positive rate against a large background

- Docking targets: Thymidine kinase and Estrogen receptor
  - For each, take 10 known ligands and 990 random ligands
  - Dock all of them with the same parameters
  - Assess the number of false positives to achieve true positive rates of 80, 90, and 100%
  - Comparative data:
    - Bissantz, C.; Folkers, G.; Rognan, D. Protein-based virtual screening of chemical databases. 1. Evaluation of different docking/scoring combinations. J Med Chem 2000, 43, 4759-4767.
    - Compared GOLD, Dock 4.0, FlexX
    - Considered combinations of dockers and scoring functions applied post-docking as well

Diverse ligands
TK hydrophilic, ER hydrophobic

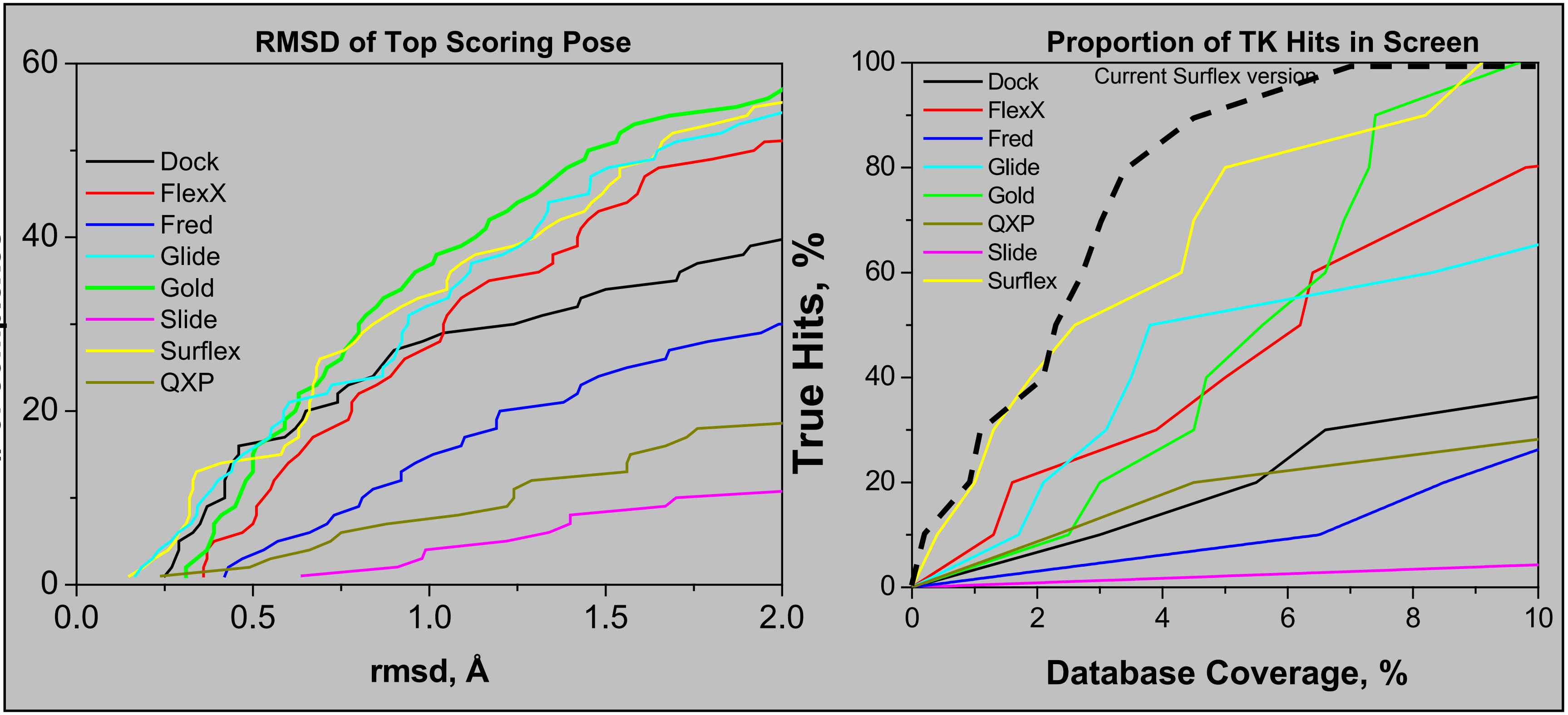


Results: Empirical scoring function yields high specificity for these two cases

- Thymidine kinase
  - Surflex FP rate for 80% TP is almost 10-fold than GOLD, which does best among the other methods
  - FP rate for all TP rates is lowest
- Estrogen receptor
  - With a threshold on protein penetration allowing for a 90% TP rate, Surflex yields more than 10-fold reduction in FP rate over any *single* method
  - Moving to 100% TP, Surflex outperforms the other methods as well

	Thymidine Kinase				Estrogen Receptor					
TP %	Surflex (pen < 3)	DOCK	FlexX	GOLD	Surflex (pen < 6)	Surflex (pen < 12)	DOCK	FlexX	GOLD	GOLD/DOCK
80%	0.9%	23.4%	8.8%	8.3%	0.2%	1.3%	13.3%	57.8%	5.3%	1.2%
90%	2.8%	25.5%	13.3%	9.1%	0.7%	1.6%	17.4%	70.9%	8.3%	1.5%
100%	3.2%	27.0%	19.4%	9.3%		2.9%	18.9%		23.4%	12.1%

Direct comparison of 8 methods by Rognan's group at CNRS confirms our results



- 100 protein/ligand complexes
- 8 docking programs
- Tested on same platform under similar time pressure

- Thymidine kinase example
- 10 true ligands, 990 random

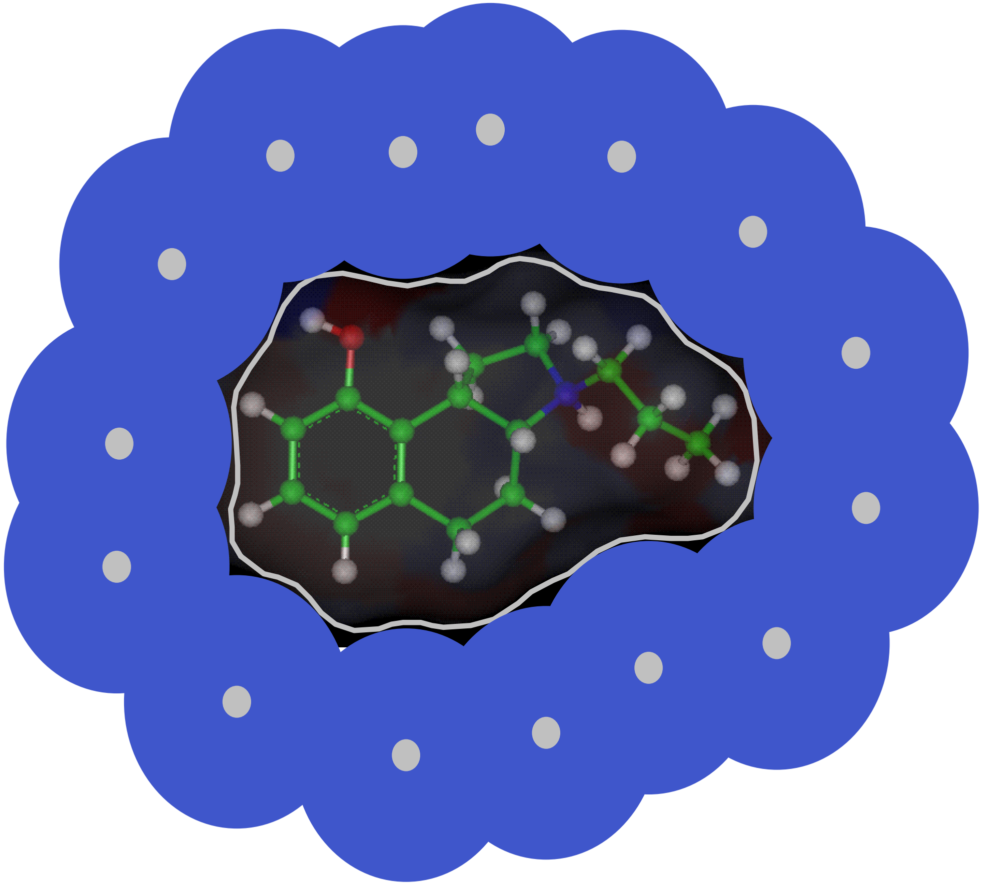
Flexible Molecular Docking

- Papers to read
  - Jain, A.N. (1996). Scoring noncovalent protein-ligand interactions: A continuous differentiable function tuned to compute binding affinities. J Comput Aided Mol Des 10, 427-40.
  - Welch, W., Ruppert, J. & Jain, A.N. (1996). Hammerhead: Fast, fully automated docking of flexible ligands to protein binding sites. Chem Biol 3, 449-62.
  - Ruppert, J., Welch, W. & Jain, A.N. (1997). Automatic identification and representation of protein binding sites for molecular docking. Protein Sci 6, 524-33.
  - Jain, A.N. (2000). Morphological similarity: A 3D molecular similarity method correlated with protein-ligand recognition. J Comput Aided Mol Des 14, 199-213.
  - Jain, A. N. (2003) Surflex: Fully Automatic Flexible Molecular Docking using a Molecular Similarity-Based Search Engine. J Med Chem 46: 499-511.

- Conclusions
  - Possible to induce an effective scoring function empirically
  - Search engine is effective
  - Docking accuracy (rmsd) is competitive with the best available methods
  - Specificity in terms of screening false positive rates is substantially better than competing methods

Suppose you don’t have a protein crystal structure

- **Problem**: induce the protein binding pocket given molecules + potencies
- Characterize ligands based on what the protein “sees”
  - Pure steric shape
  - Polar interactions and direction
- Construct a geometrically meaningful binding site model
  - Basis functions with tunable shape
  - Physical assumptions retained
- Dock new molecules into the model
  - Generates interpretable cross-chemotype relationships
  - Prioritizes chemical synthesis



A collection of well-placed spheres accurately represents the molecular surface of a putative binding pocket

A. N. Jain, N. L. Harris, and J. Y. Park. Quantitative Binding Site Model Generation: Compass Applied to Multiple Chemotypes Targeting the $5HT_{IA}$ Receptor. *Journal of Medicinal Chemistry* 38: 1295-1307, 1995.

Four things to do for a machine-learning task

- Choose a representation of your input data
  - We need to pick an input representation that includes information relevant to protein ligand binding energies: we'll use distance to surface again
  - Since we don't have a protein, we'll use the distance from points on a sphere outside the molecules of interest
- Choose a functional form that will map input examples to outputs
  - We're going to predict a binding energy in $-\log(K_d)$ units
- Choose a method of optimization
  - We will use gradient-descent to optimize the parameters of the function
  - We will embed pose optimization in the learning task
- Train your system and evaluate performance
  - In this case, we have a training set of 20 5-HT1a ligands
  - We have an independent test set of 35 novel ligands

But we have a problem: hidden variables of alignment and conformation

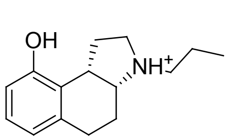
- If we knew the correct poses of the input ligands, this would be very much like training the Surflex scoring function
- We don’t
- So, we need a method for guessing the mutual alignment and conformation initially
- We can then vary our guess

So how do we get an initial alignment guess?

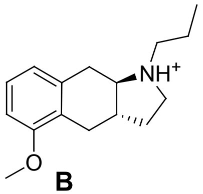
- We need to construct a function of joint molecular pose
- It should have a maximum where molecules are mutually aligned in a **predictively** useful manner
- We can measure utility with model systems

Many cases require a solution to the mutual alignment problem

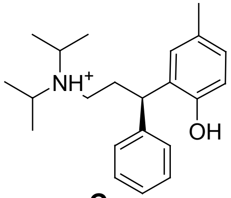
- Many classes of targets are not currently tractable by crystallography
- GPCR and ligand-gated ion channel ligands
  - Molecules A–B: 5-HT1a ligands
  - Molecules C–E: Muscarinic antagonists
  - Molecules F–H: Histamine receptor antagonists
  - Molecules I–K: $GABA_A$ receptor agonists
- We can begin to induce binding site models by making use of molecular similarity methods



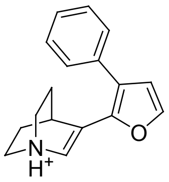
A



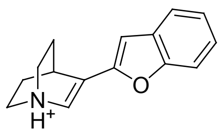
B



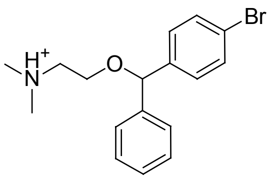
C



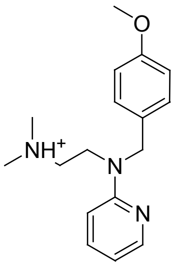
D



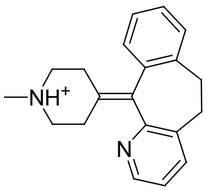
E



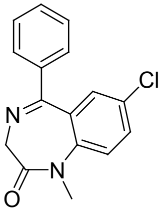
F



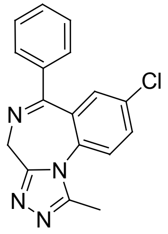
G



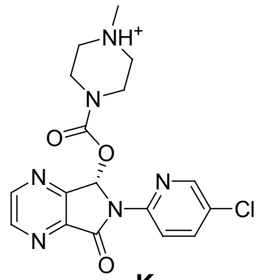
H



I



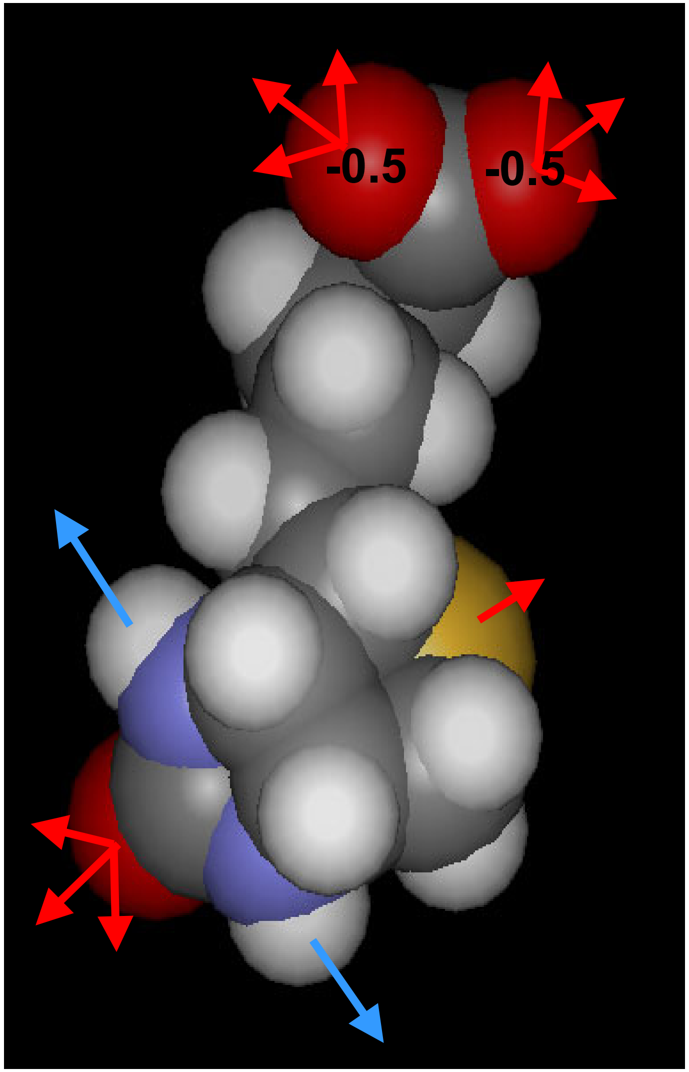
J



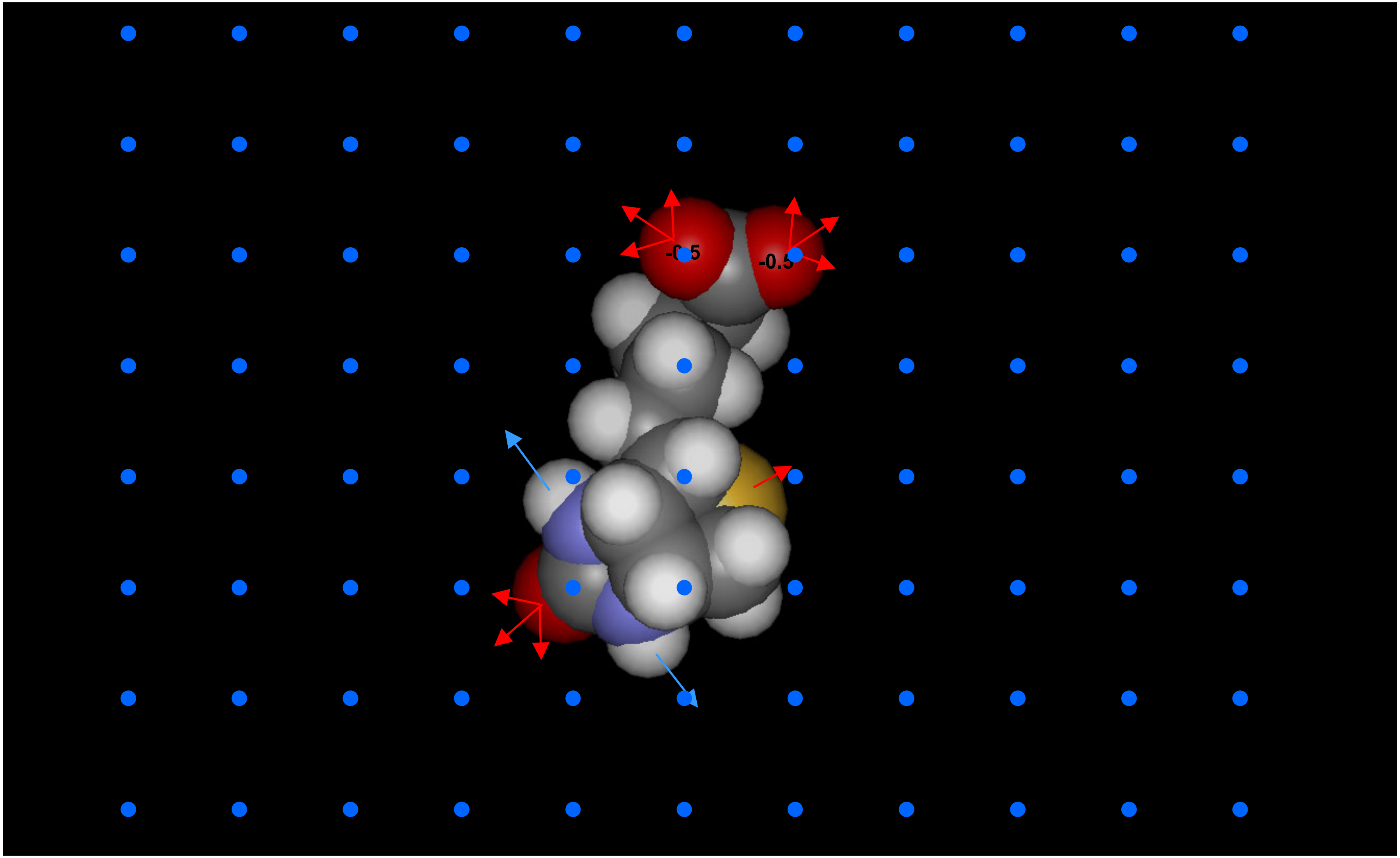
K

We will use precisely the same representation of molecules as for docking

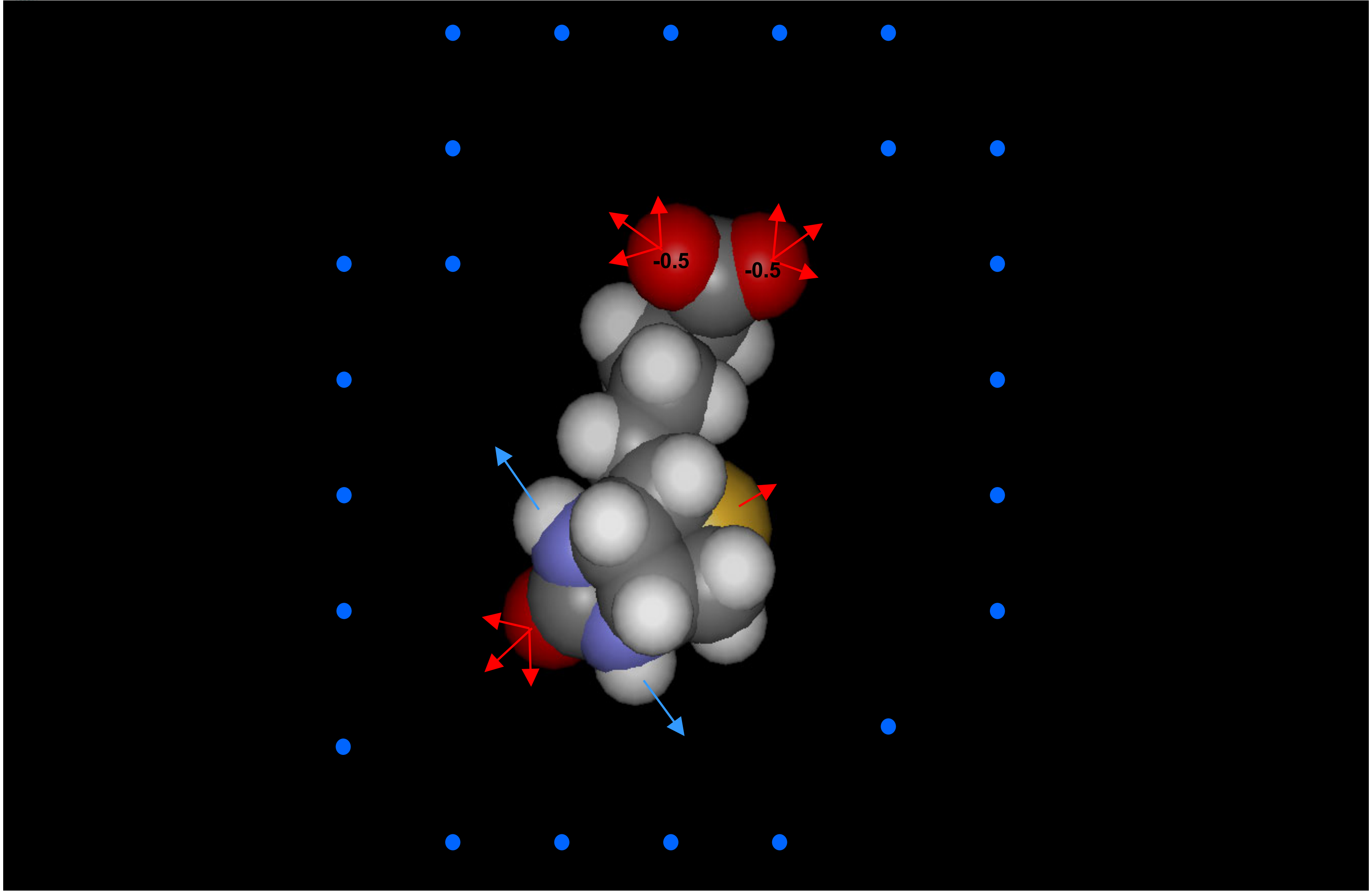
- Our molecular similarity function is a function of distances and vector coincidences
- We will define a set of points as surrogates for protein atoms
- From these points, we will compute distances and vector coincidences



We don’t know where to put the points: We put them everywhere (spacing λ)



We can't have an infinite number of points: We will light up points at a specific distance γ



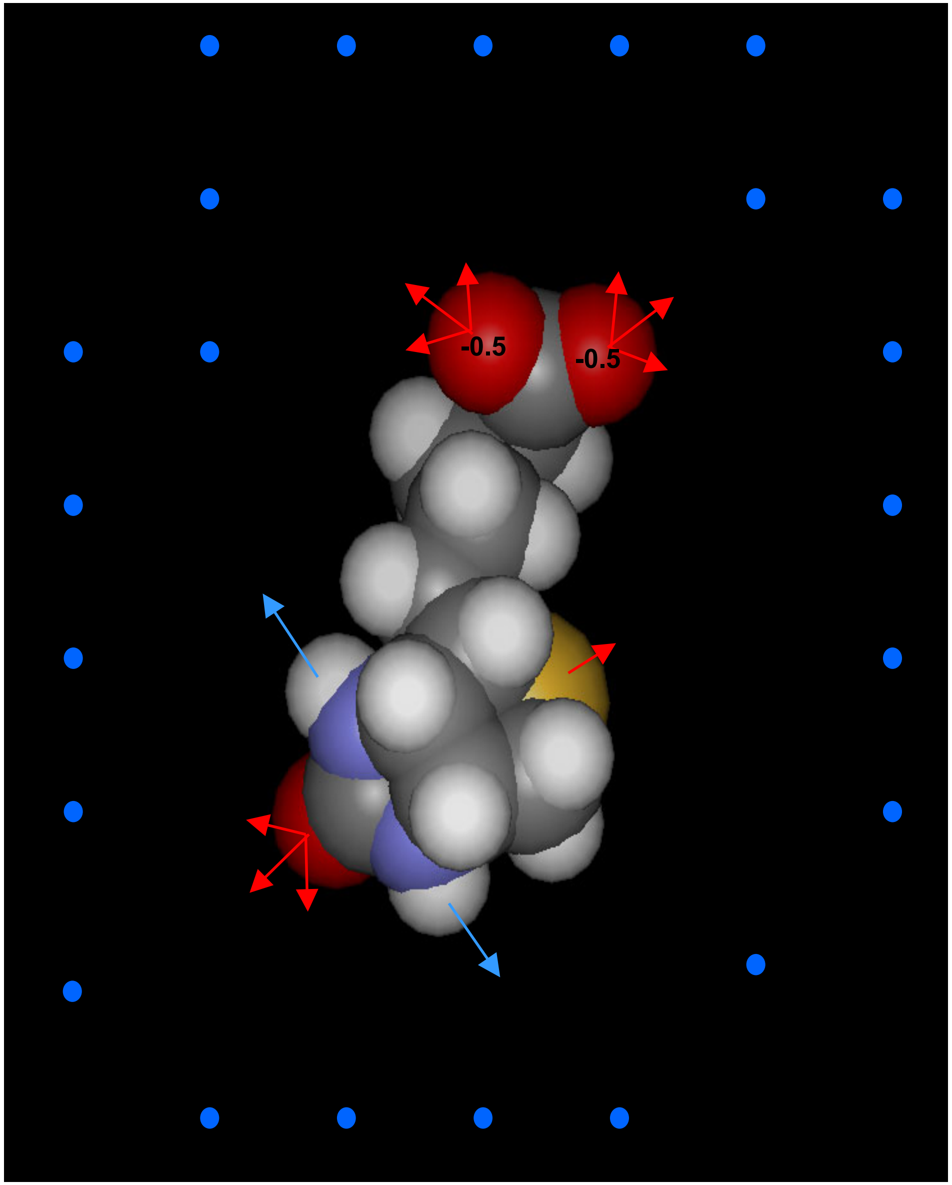
Pseudo-Protein Pocket

We define a Gaussian function of distance to weight the pseudo-protein points

$$w = e^{-(d-\gamma)^2/o}$$

- We cut off points with weight < 0.1
- This gives us points spaced at about γ from the molecule in question
- We can control the sloppiness of the weight using o

$\lambda = 2.0$, $\gamma = 4.0$, $o = 0.2$

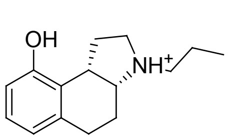


Measure the molecules from the perspective that a protein has, but use a soft function

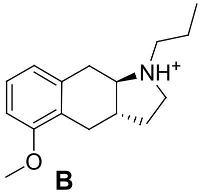
$$f(a,b) = \frac{\sum_i (w_i^a + w_i^b) \left[\begin{array}{c} \sigma(s_i^a - s_i^b, \lambda_1) \\ + \\ \max(S^{a^+}, S^{b^+})\sigma(s_i^{a^+} - s_i^{b^+}, \lambda_1)\sigma(S_i^{a^+} - S_i^{b^+}, \lambda_2) \\ \max(S^{a^-}, S_i^{b^-})\sigma(s_i^{a^-} - s_i^{b^-}, \lambda_1)\sigma(S_i^{a^-} - S_i^{b^-}, \lambda_2) \end{array} \right]}{\sum_i (w_i^a + w_i^b)[1 + \max(S_i^{a^+}, S_i^{b^-}) + \max(S_i^{a^-}, S_i^{b^-})]}$$

We can induce models of what a binding site must look like by using molecular similarity

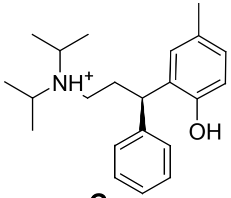
- Four cases
  - Therapeutically important
  - No structures of human proteins known
  - Wealth of chemical scaffolds known for each receptor
- Can we use these molecule sets to construct a structural hypothesis for the way they bind their receptors?
- Will it be useful to identify other ligands of the same receptor from a large library?



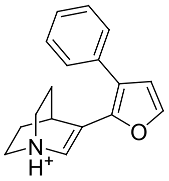
A



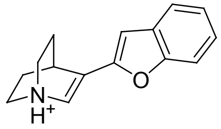
B



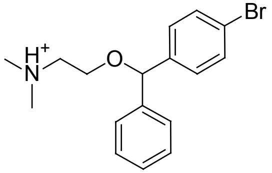
C



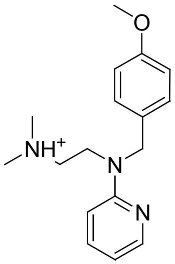
D



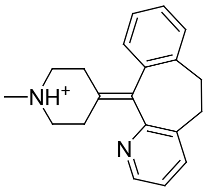
E



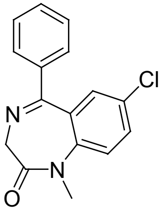
F



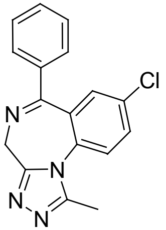
G



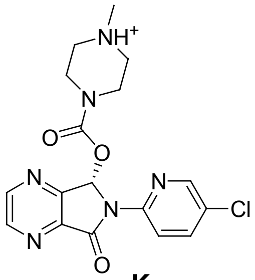
H



I

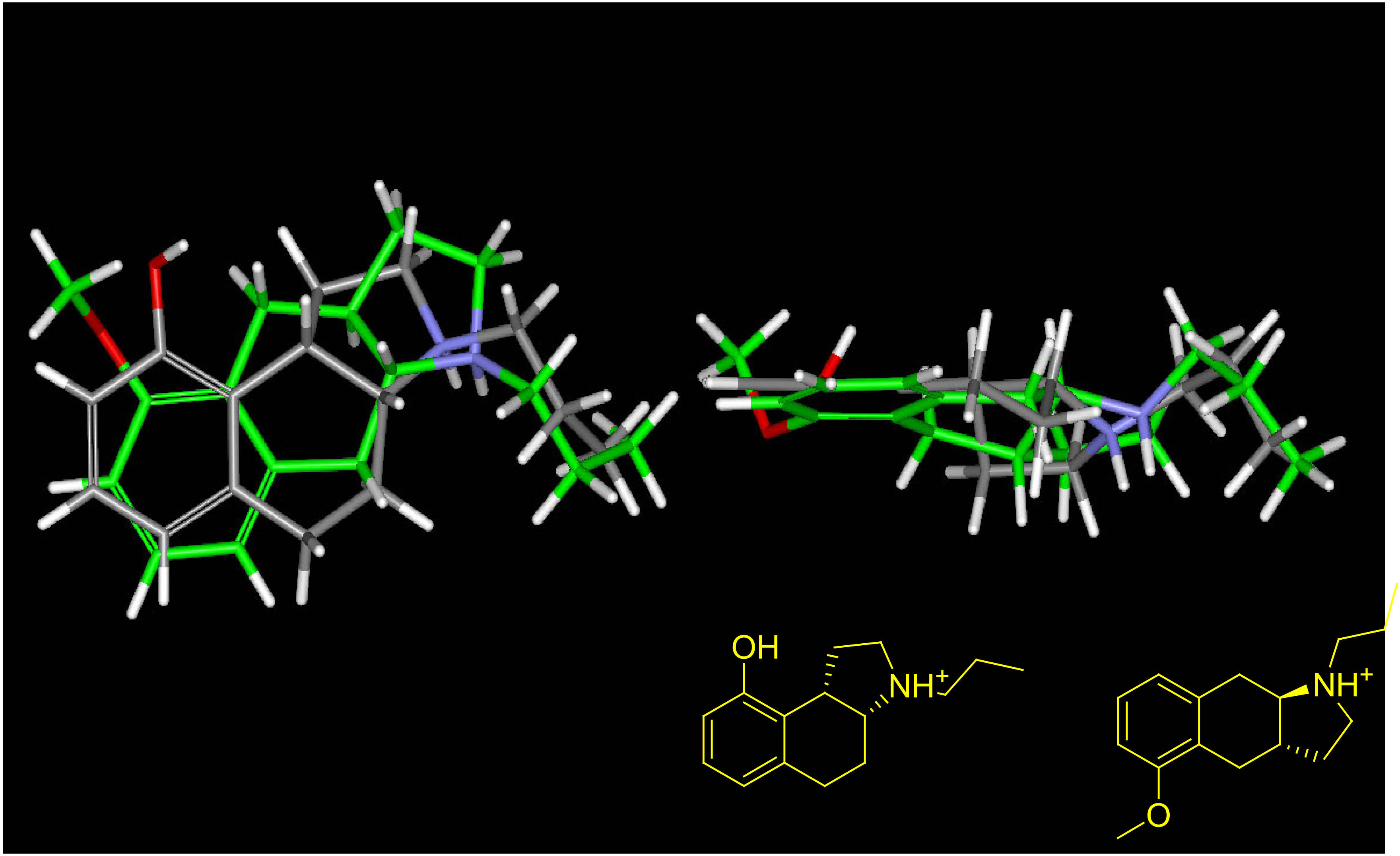


J

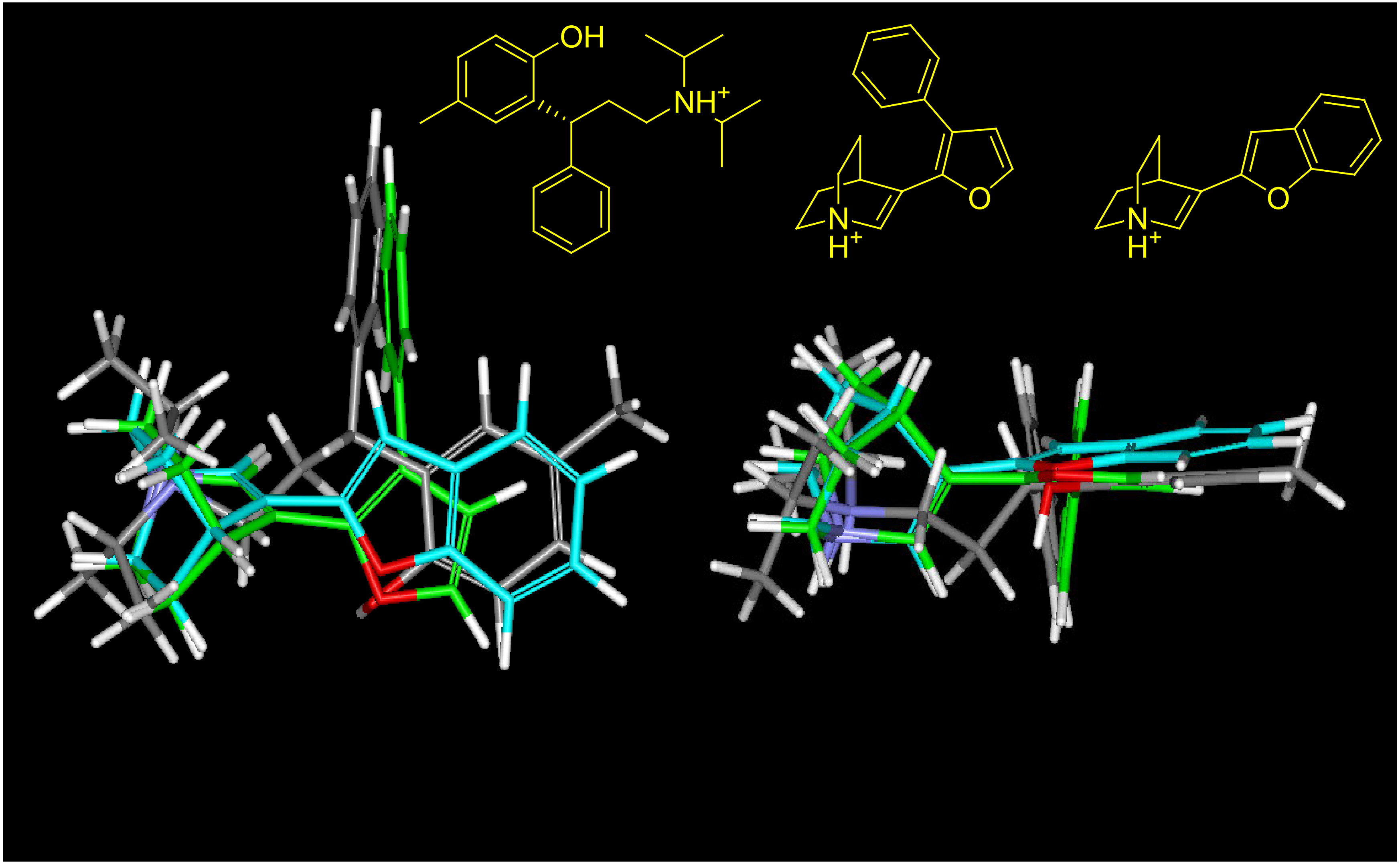


K

5-HT1a overlay that maximizes a joint similarity function looks convincing



Muscarinic overlay that maximizes a joint similarity function looks convincing

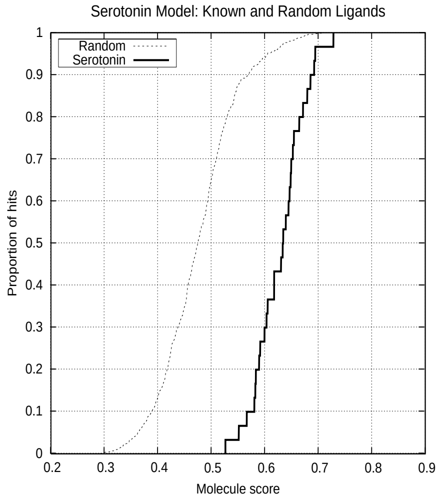


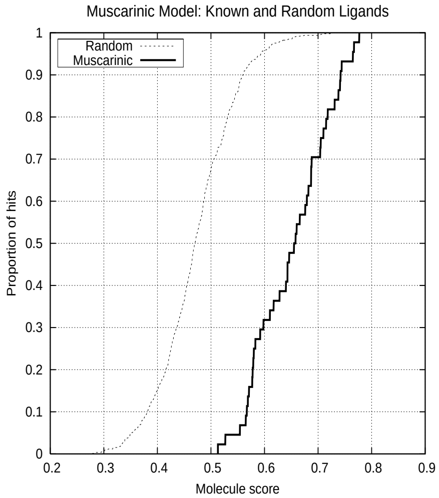
How can we validate these hypotheses in a quantitative way?

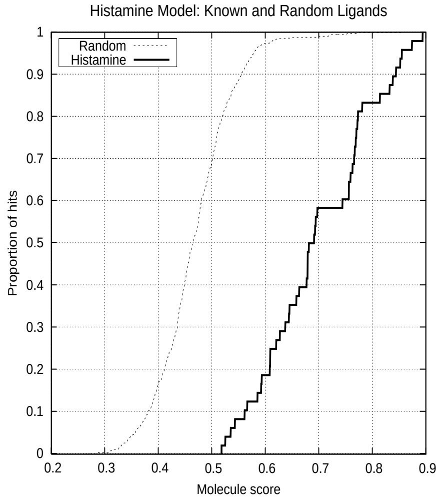
- Consider two sets of molecules
  - Known ligands of any of the four receptors (100 total, from GPCRDB and the Merck Index)
  - Random ligands (990 from previous experiment)
- Compute similarity of the molecule sets to the binding site hypotheses
  - Optimize alignment to each molecule of the hypothesis
  - Compute mean of similarity score to all molecules in hypo
  - Report maximum score as the score of the test ligand
- We hope to see separation between known ligands and the random or non-cognate ligands

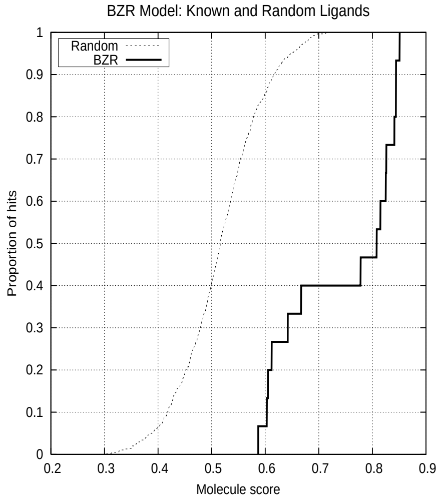
We observe separation between true ligands and non-ligands

- Cumulative distributions of random molecules are shifted far left of true ligands
- True positive rates of 60% are possible with a FP rate of 2-3%
- Not as good as the best docking results, but competitive with many docking methods
- Theoretical enrichment rates of >150-fold


Serotonin Model: Known and Random Ligands


Muscarinic Model: Known and Random Ligands

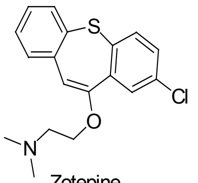

Histamine Model: Known and Random Ligands


BZR Model: Known and Random Ligands

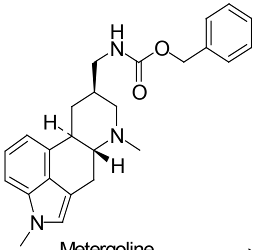
This is not simply an artifact of inductive bias from the molecules used for model construction

- Very different chemical structures are retrievable based on these very simple computational structures
- Not possible to reasonably argue that the results are trivially related to inductive bias

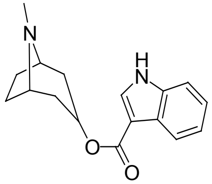
Serotonin



Zotepine
99.8

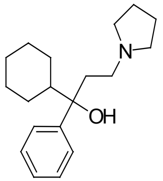


Metergoline
98.9

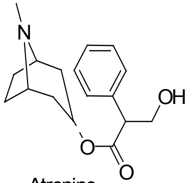


Tropisetron
98.8

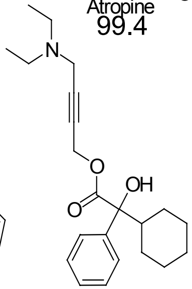
Muscarinic



Procyclidine
100.0

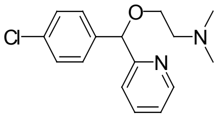


Atropine
99.4

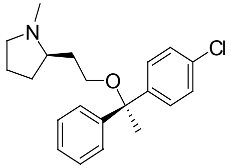


Oxybutynin
98.3

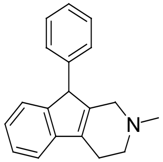
Histamine



Carbinoxamine
100.0

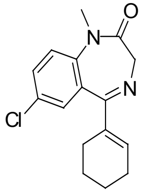


Clemastine
99.9

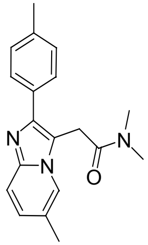


Phenindamine
98.6

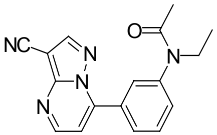
$GABA_A$



Tetrazepam
100.0



Zolpidem
96.8



Zaleplon
87.1

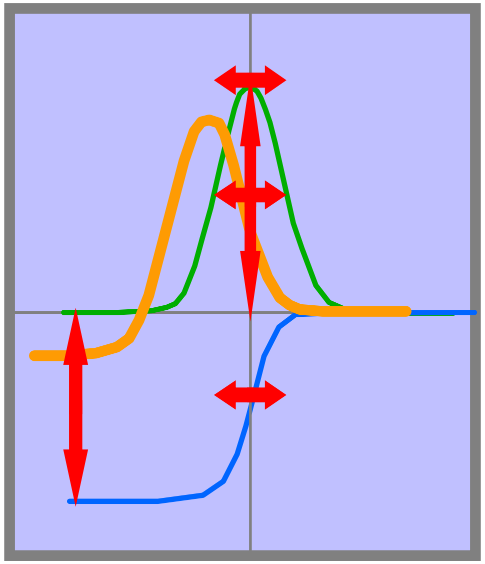
Percentile scores of the ligands: 10/12 are within the top 2% of random scores (6/12 within 1%).

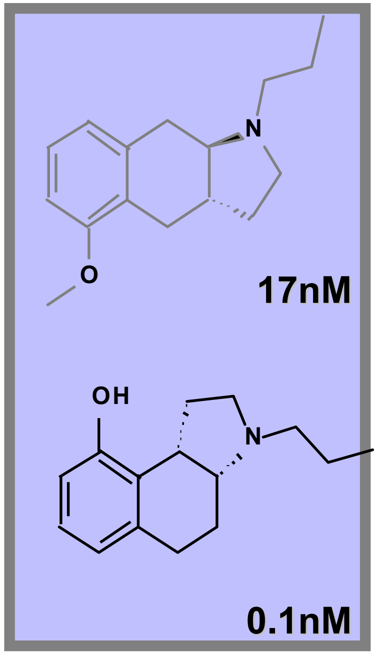
So now we have an alignment method

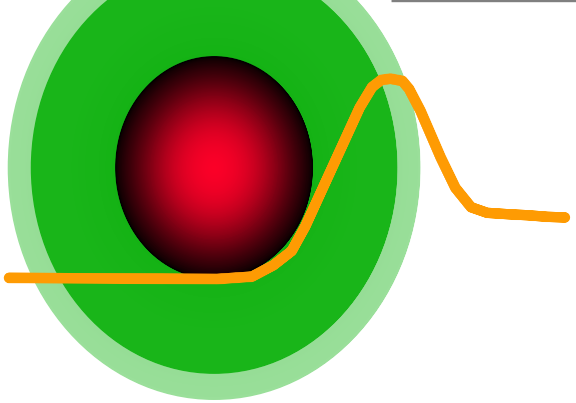
- We can now build quantitatively predictive models of molecular activity
- Very similar procedure and notion to Surflex scoring function

The gaussian + sigmoidal functions of molecular distance make “soft spheres”

- Gaussian + sigmoidal functions of distance-based molecular features
    - Form soft-shelled balls, positive outside and negative inside
    - Very similar to Surflex scoring function
- Alignment and conformation
    - Molecules are docked into the binding site for maximal score
- Model induction
    - Find the site that very active molecules fit that inactive molecules don’t
    - Assume single dominant pose for all molecules



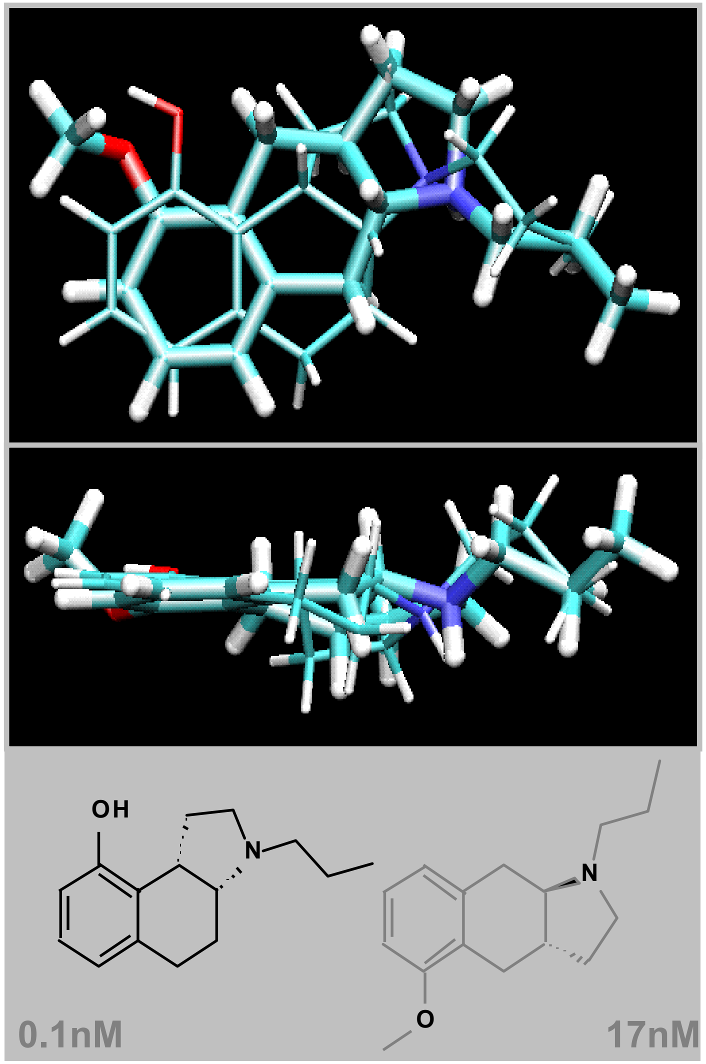




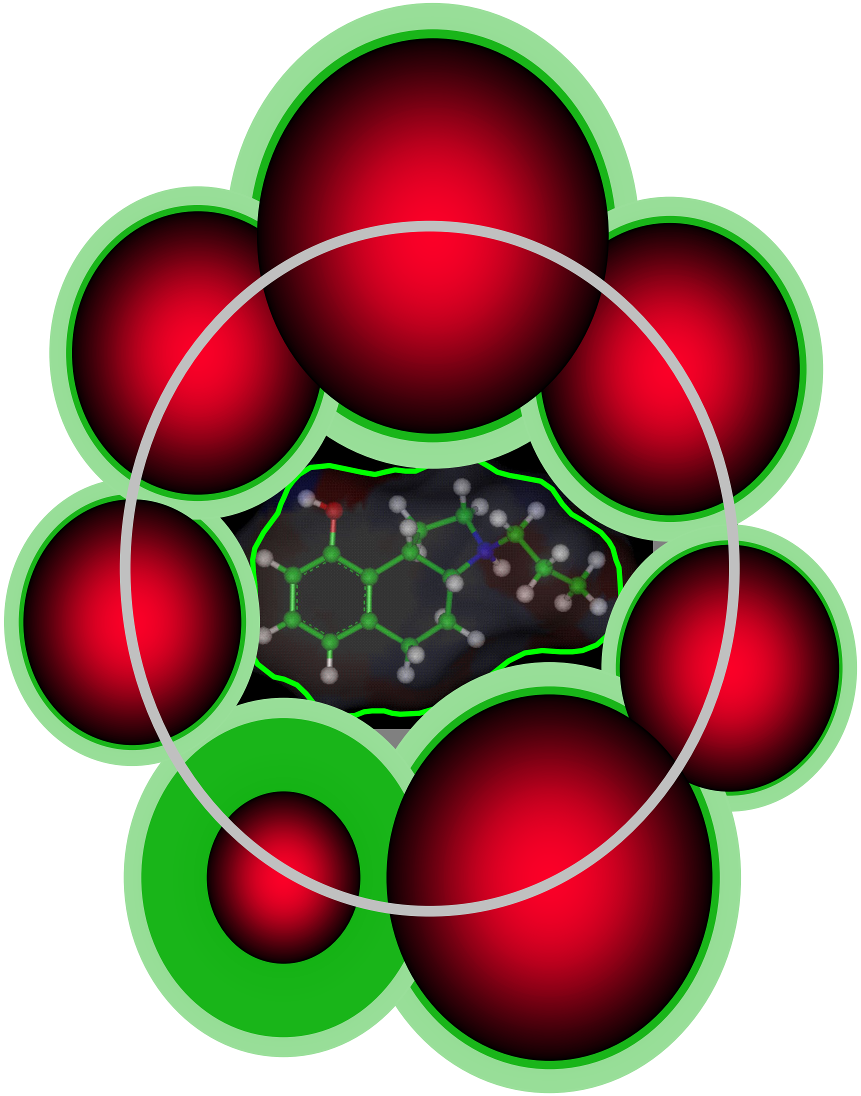
Compass 5-HT1a model predicts accurately across chemotypes

- Small training set: 20 molecules
  - 9 pairs of enantiomers from two chemotypes, 1 pair of diastereomers
  - Cross-validation: 0.5 log units error, 0.90 PRCC

- Blind test on 35 new molecules
  - 0.5 log units error, 0.84 PRCC
  - Novel ring fusion stereochemistry
  - Novel ring structures (cyclobutyl variant and cyclic urea heterocycle)



Model yields weighted geometric models of binding sites



- Technique has been applied to several targets
  - 5-HT1a
  - Steroid binding globulins
  - Muscarinic antagonists
  - Enzyme inhibition

- Results
  - Accurate predictions of potency
  - Relationship of different chemotypes
  - Prospective identification of active compounds of novel structural types

Molecular similarity and quantitative binding site model generation

- Papers to read
  - Jain, A.N., Harris, N.L. & Park, J.Y. (1995). Quantitative binding site model generation: compass applied to multiple chemotypes targeting the 5-HT1A receptor. J Med Chem 38, 1295-308.
  - Ghuloum, A.M., Sage, C.R. & Jain, A.N. (1999). Molecular hashkeys: A novel method for molecular characterization and its application for predicting important pharmaceutical properties of molecules. J Med Chem 42, 1739-48.
  - Jain, A.N. (2000). Morphological similarity: A 3D molecular similarity method correlated with protein-ligand recognition. J Comput Aided Mol Des 14, 199-213.
  - Jain, A.N. (2004). Ligand-Based Structural Hypotheses for Virtual Screening. J Med Chem.

- Conclusions
  - Molecular similarity based on 3D surface shapes is quantitatively related to protein binding specificity of small molecules
  - It is possible to induce plausible models of binding sites based on active small molecules of different chemotypes
  - These models can yield specificity rates in screening that are competitive with molecular docking methods