# Protein-Ligand Scoring with Convolutional Neural Networks

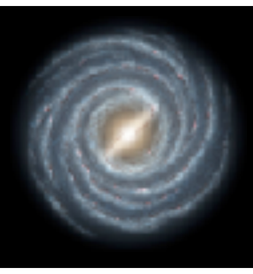
David Koes

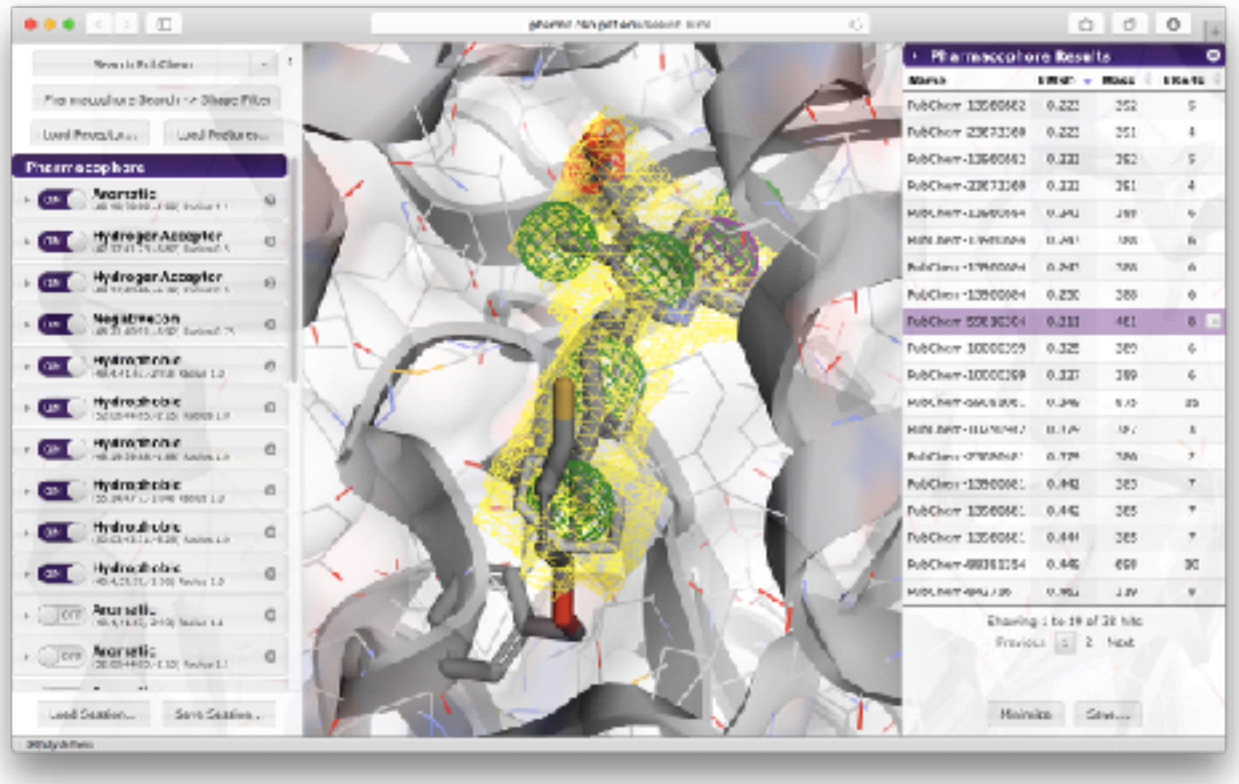
@david_koes

Novartis

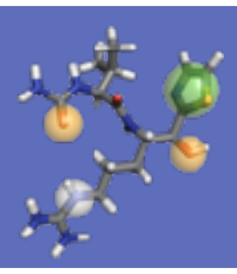
June 23, 2017

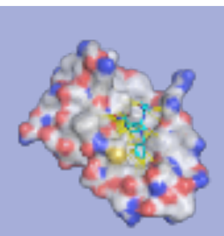
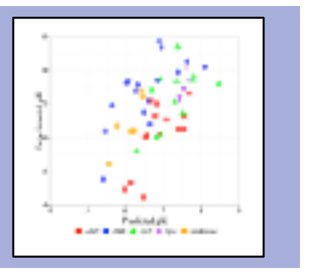
# Removing barriers to computational drug discovery one bit at a time

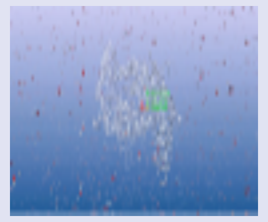
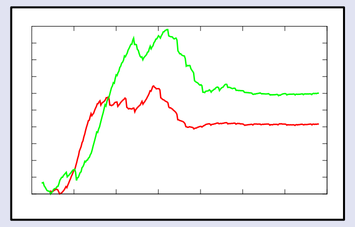
**Purchasable**  **Accessible**

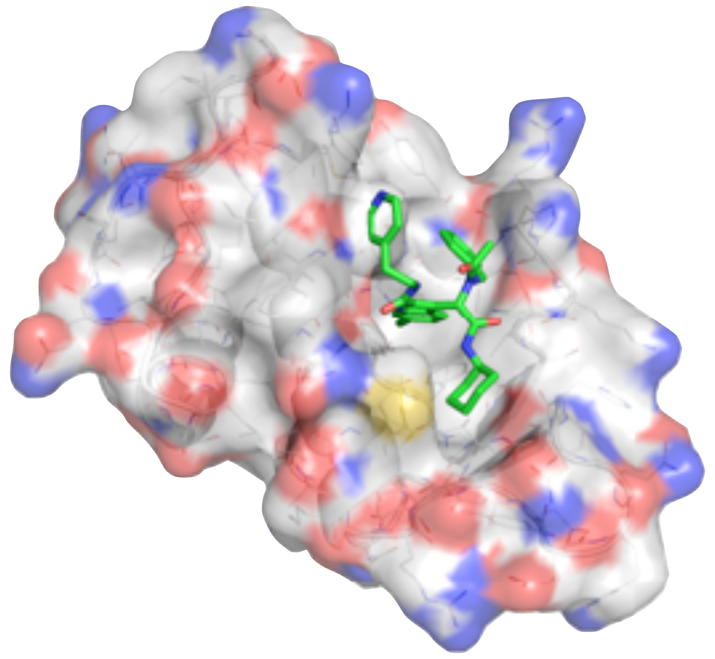


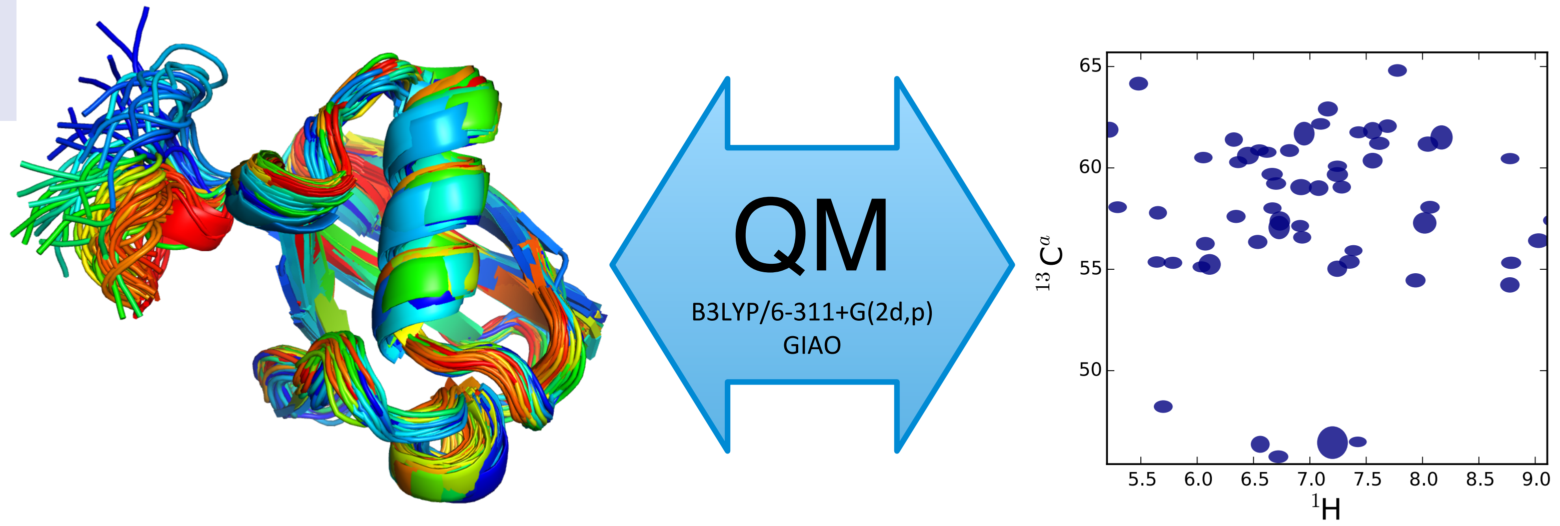
http://pharmit.csb.pitt.edu

 **Matching** 

 **Scoring** 

 **Dynamics** 

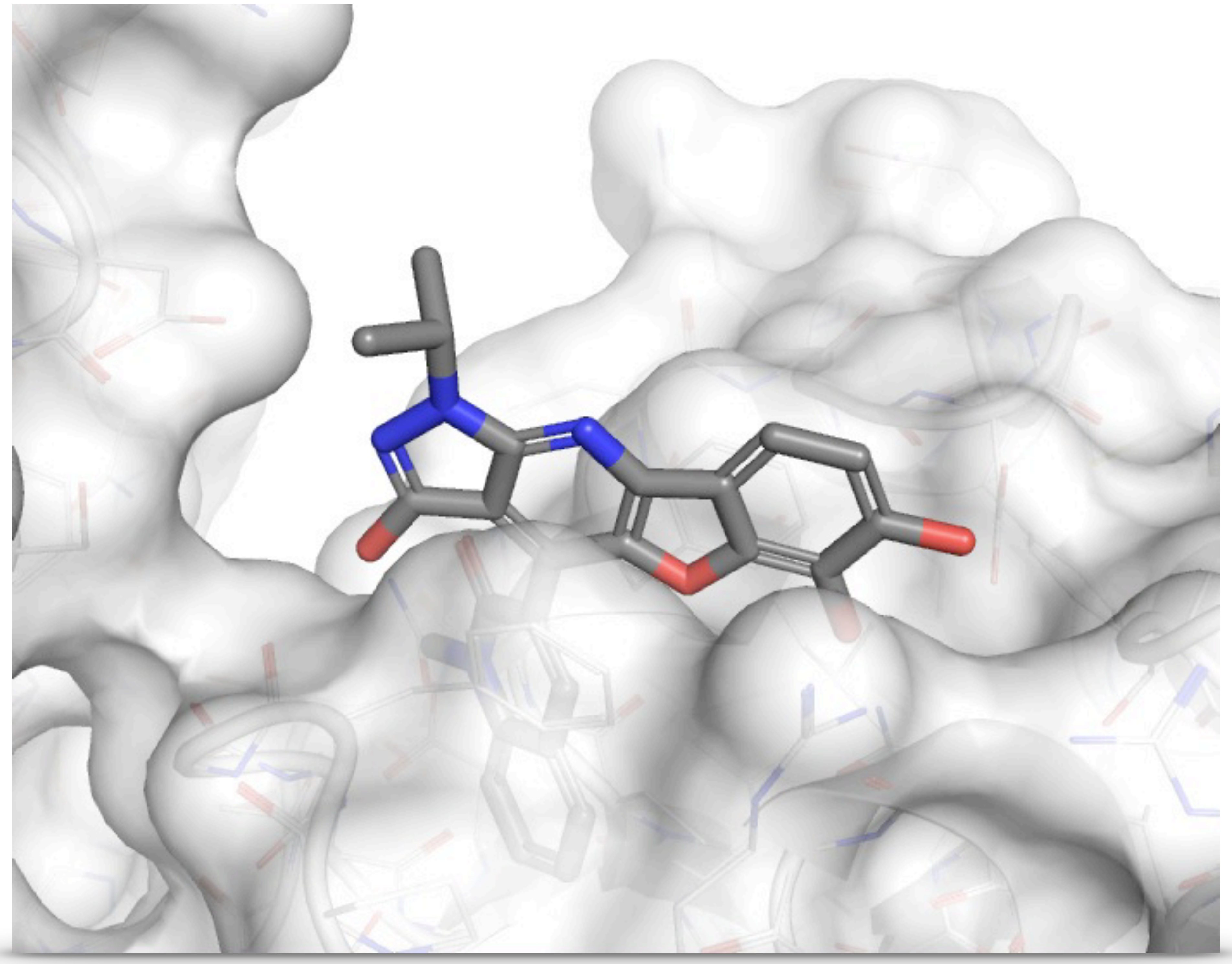




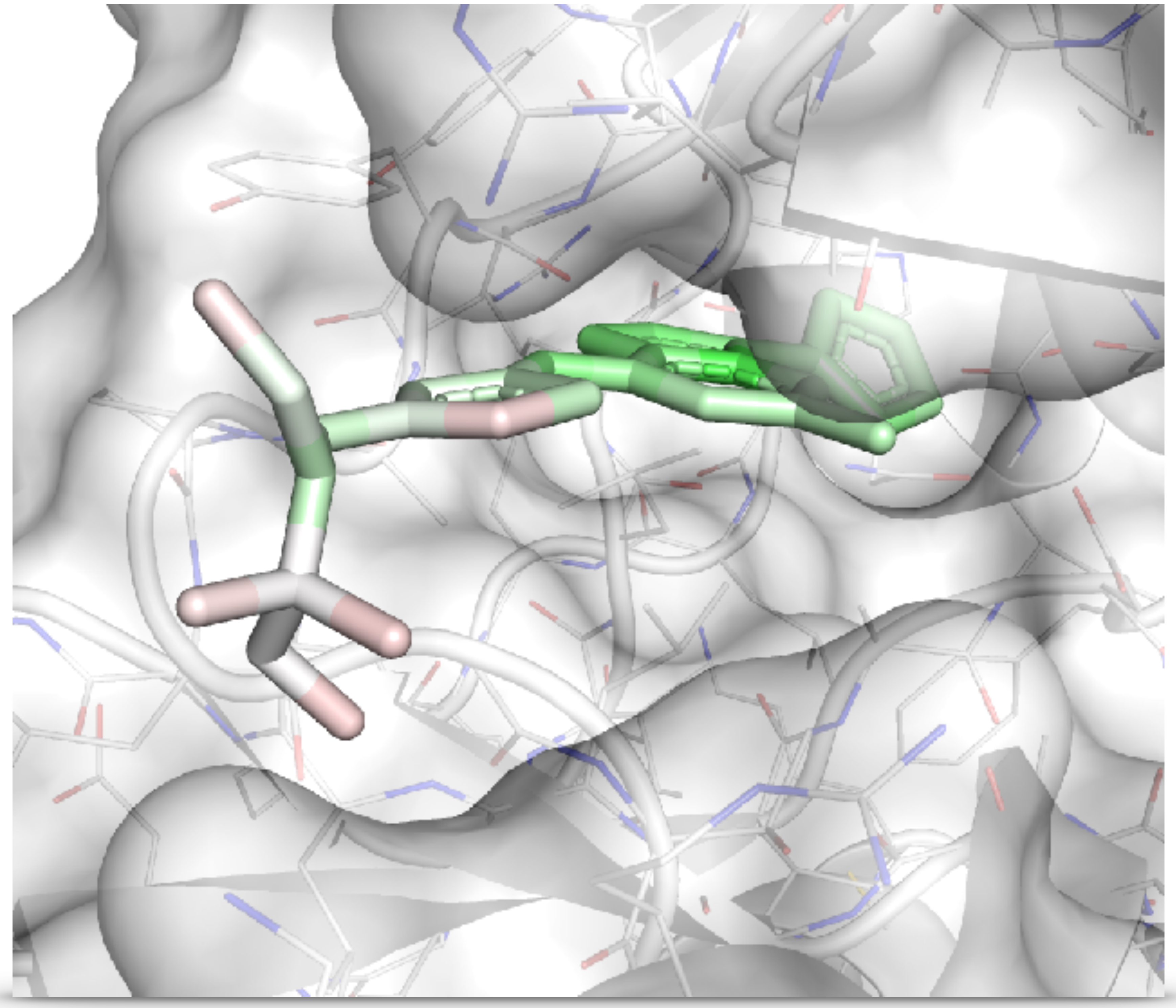

https://github.com/dkoes/MD2NMR

# Structure Based Drug Design

## Virtual Screening



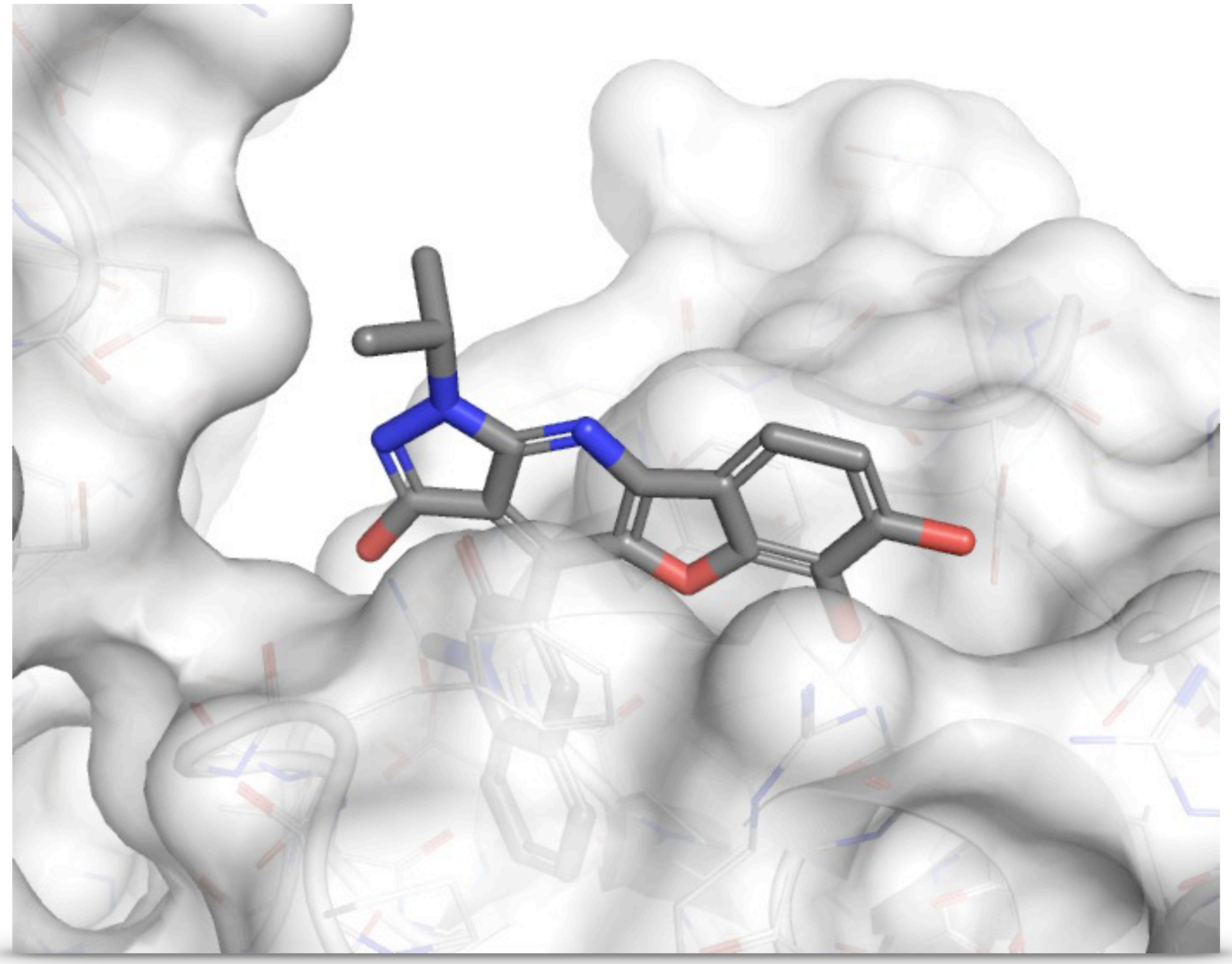
## Lead Optimization



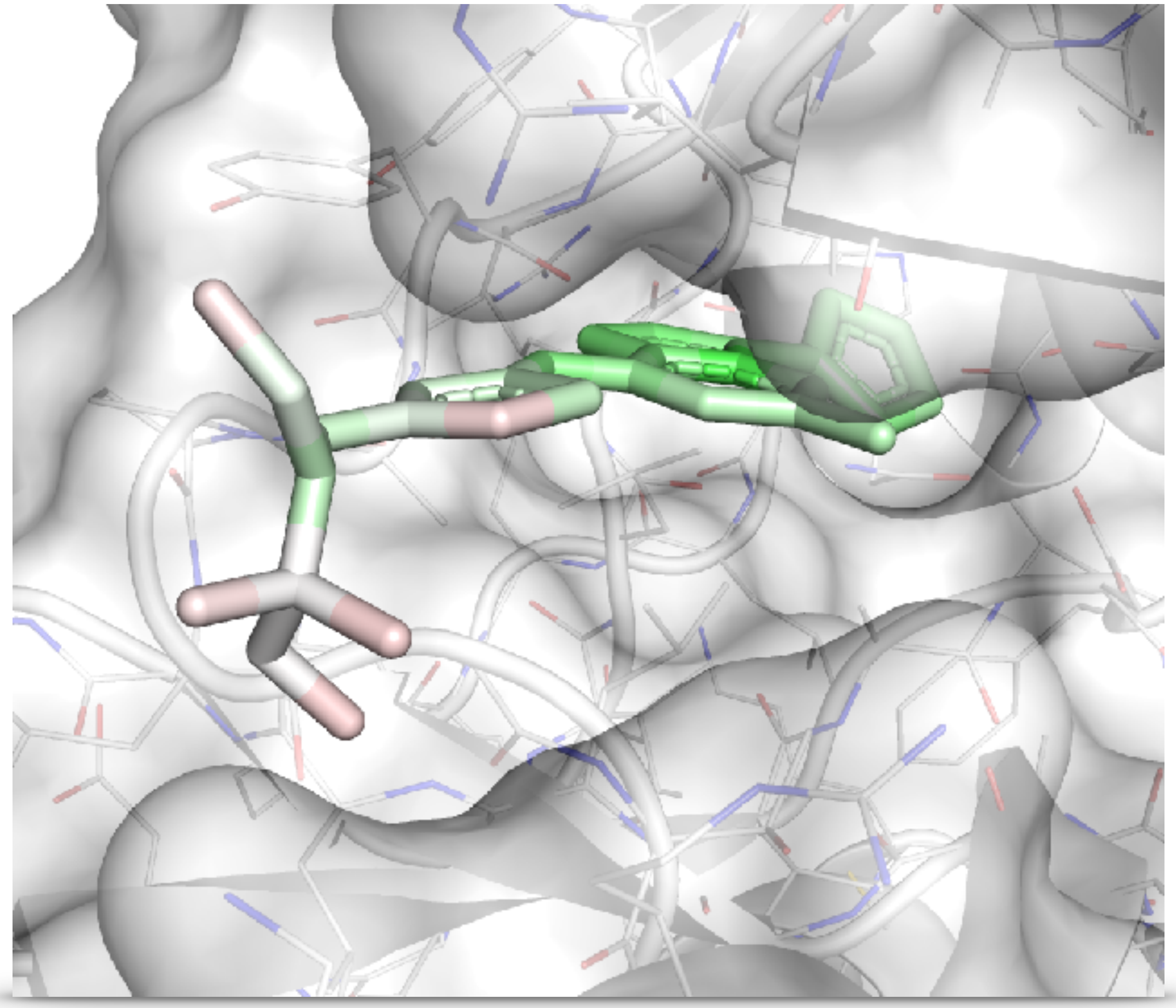
Pose Prediction → Binding Discrimination → Affinity Prediction

# Structure Based Drug Design

## Virtual Screening

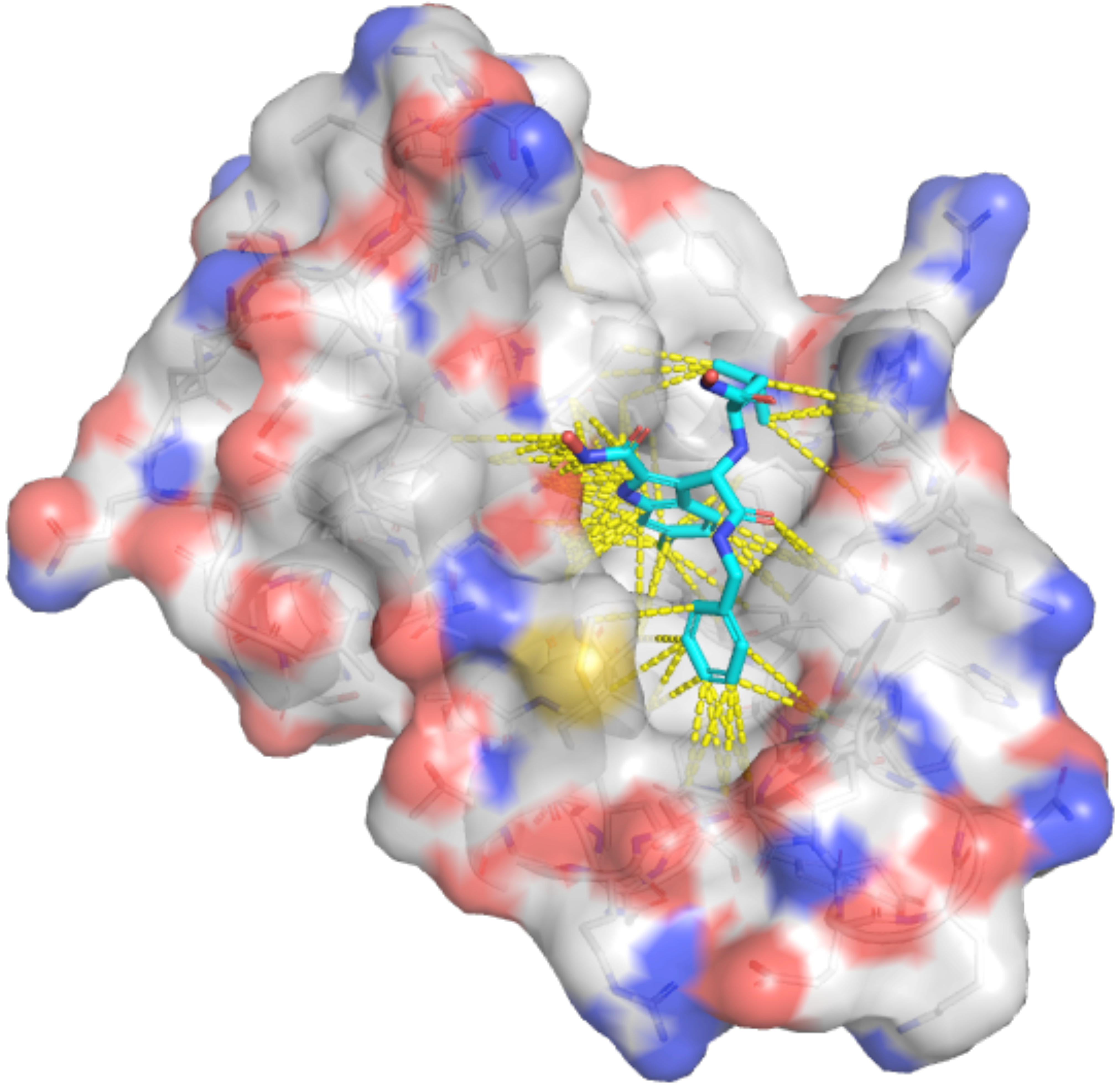


## Lead Optimization

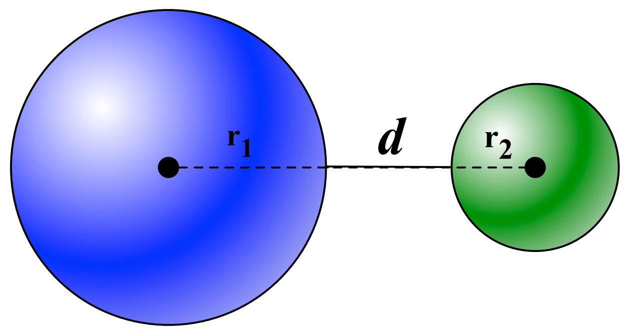


Pose Prediction → Binding Discrimination → Affinity Prediction

# Protein-Ligand Scoring



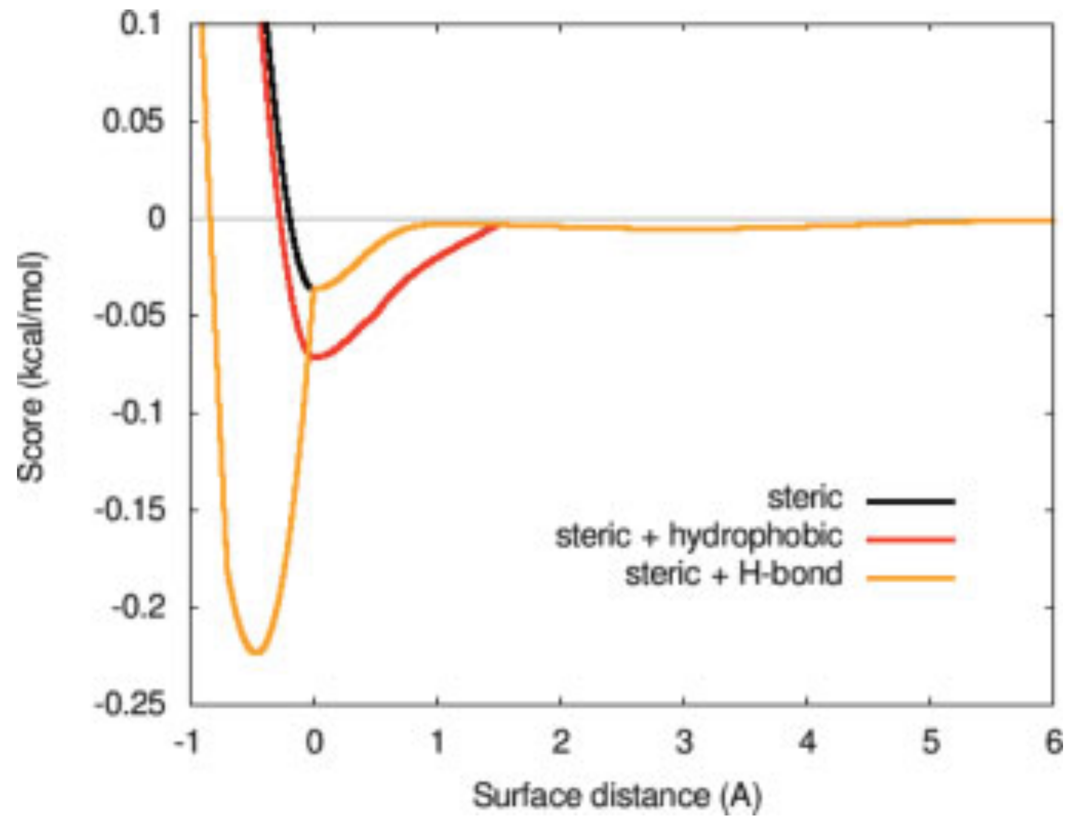
## AutoDock Vina




$$
\begin{aligned}
\mathrm{gauss}_1(d) &= w_{\mathrm{guass}_1} e^{-(d/0.5)^2} \\
\mathrm{gauss}_2(d) &= w_{\mathrm{guass}_2} e^{-((d-3)/2)^2} \\
\mathrm{repulsion}(d) &= \begin{cases} w_{\mathrm{repulsion}} d^2 & d < 0 \\ 0 & d \geq 0 \end{cases}
\end{aligned}
$$

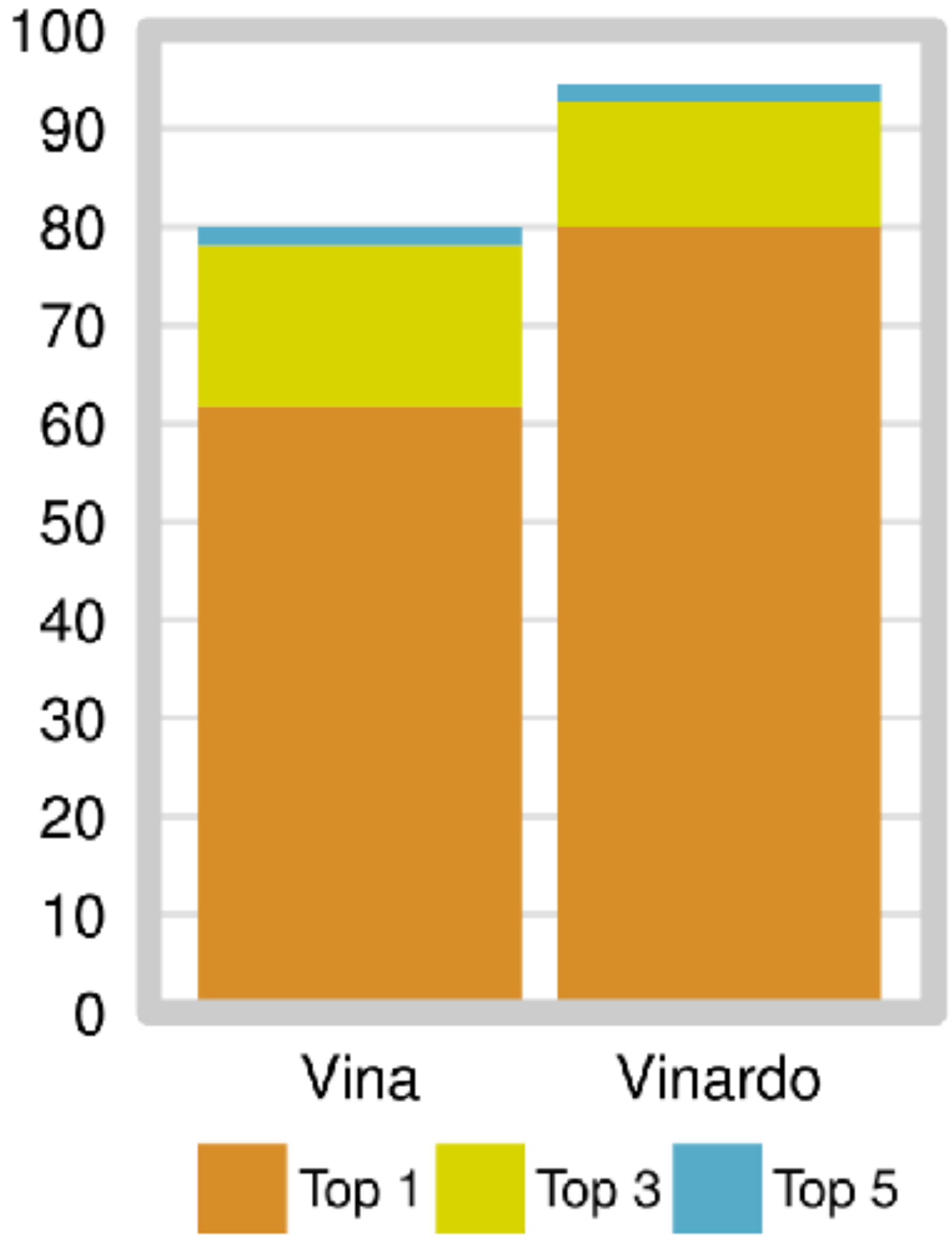
$$
\mathrm{hydrophobic}(d) = \begin{cases} w_{\mathrm{hydrophobic}} & d < 0.5 \\ 0 & d > 1.5 \\ w_{\mathrm{hydrophobic}}(1.5 - d) & \mathit{otherwise} \end{cases}
$$

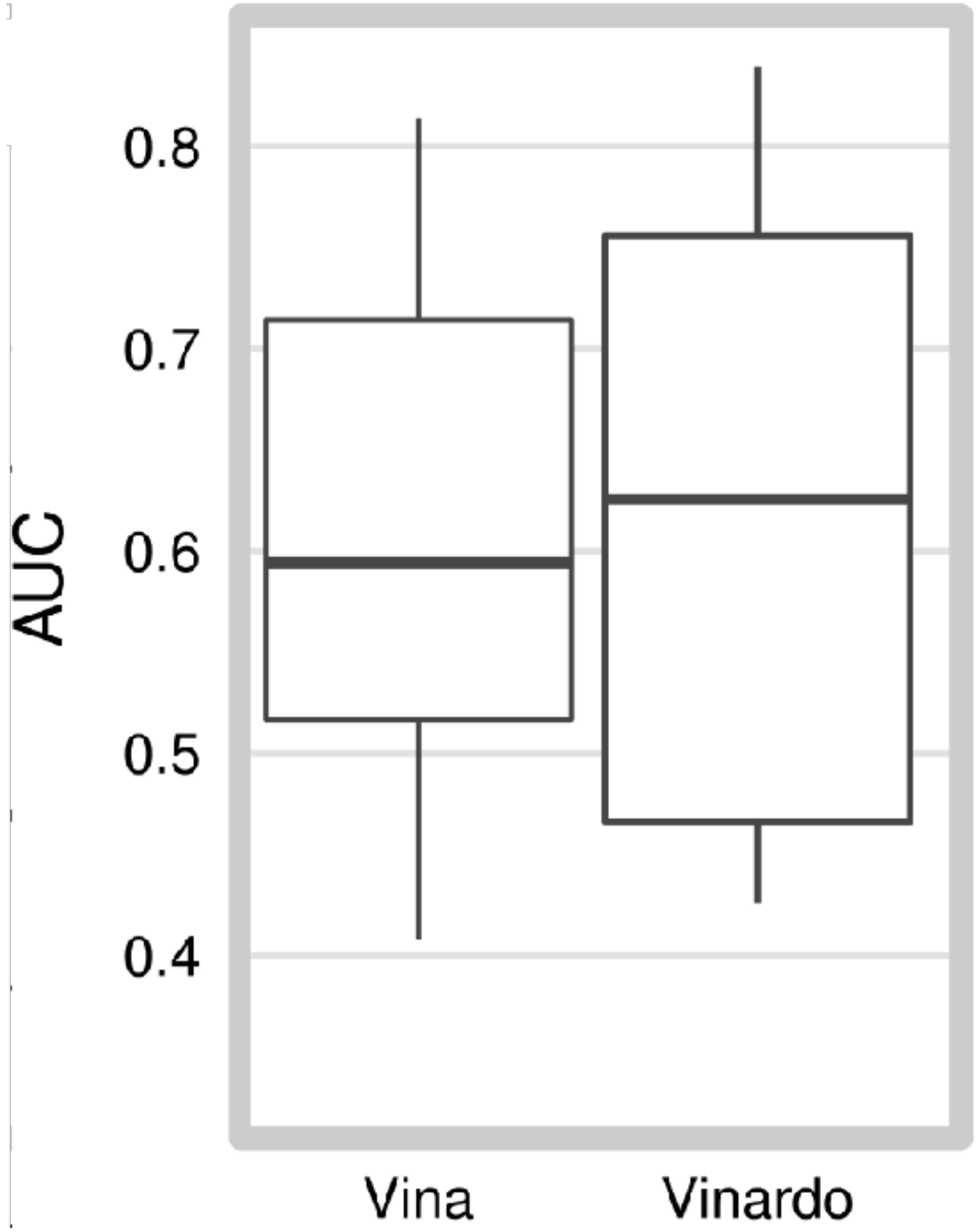
$$
\mathrm{hbond}(d) = \begin{cases} w_{\mathrm{hbond}} & d < -0.7 \\ 0 & d > 0 \\ w_{\mathrm{hbond}}(-\frac{10}{7}d) & \mathit{otherwise} \end{cases}
$$

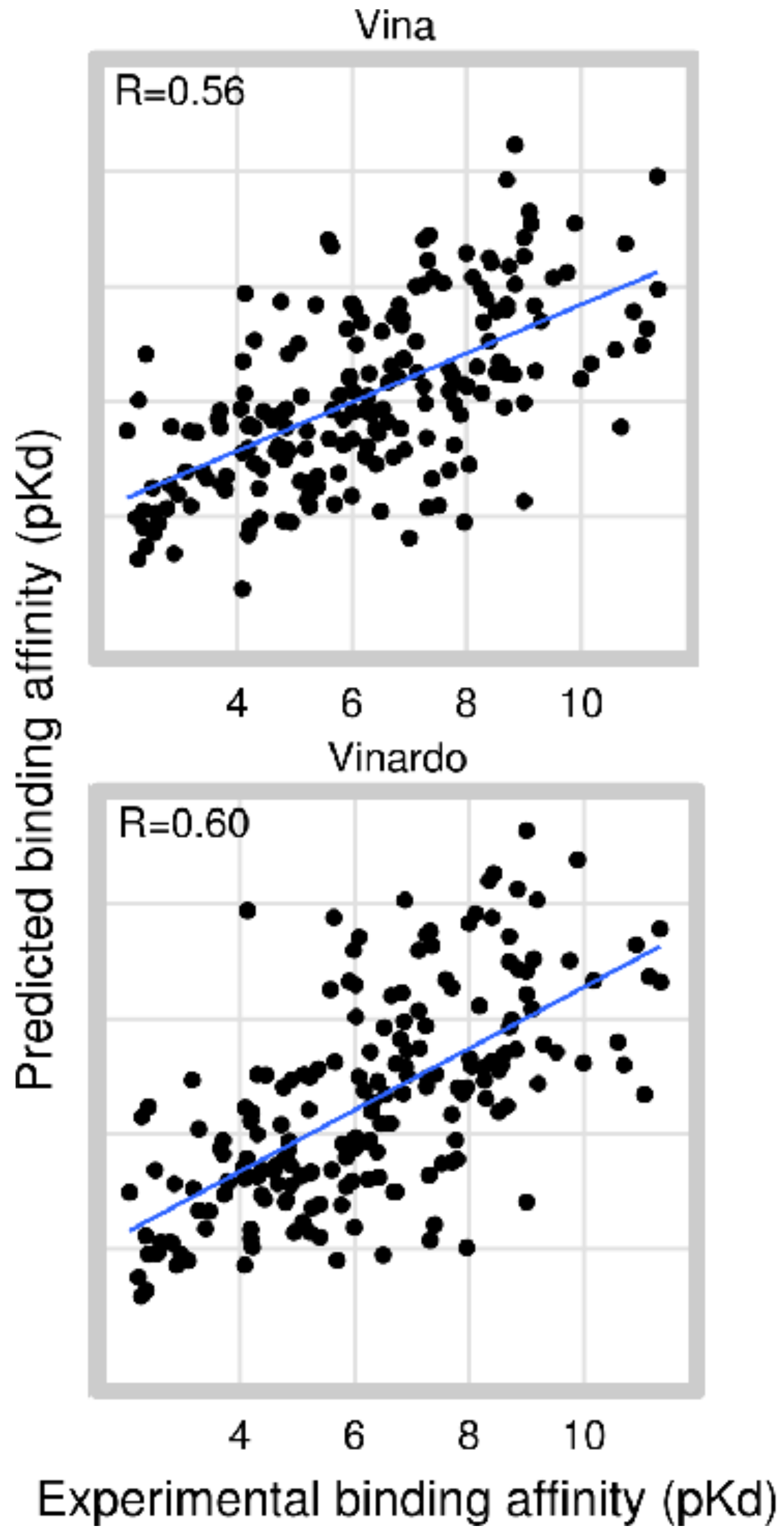



O. Trott, A. J. Olson, AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization and multithreading, *Journal of Computational Chemistry* 31 (2010) 455-461

# State of the Art




Pose Prediction Binding Discrimination Affinity Prediction

Quiroga R, Villarreal MA (2016) Vinardo: A Scoring Function Based on Autodock Vina Improves Scoring, Docking, and Virtual Screening. *PLoS ONE* 11(5): e0155183. doi:10.1371/journal.pone.0155183

# Can we do better?

Accurate pose prediction, binding discrimination, **and** affinity prediction without sacrificing performance?

# Can we do better?

Accurate pose prediction, binding discrimination, **and** affinity prediction without sacrificing performance?

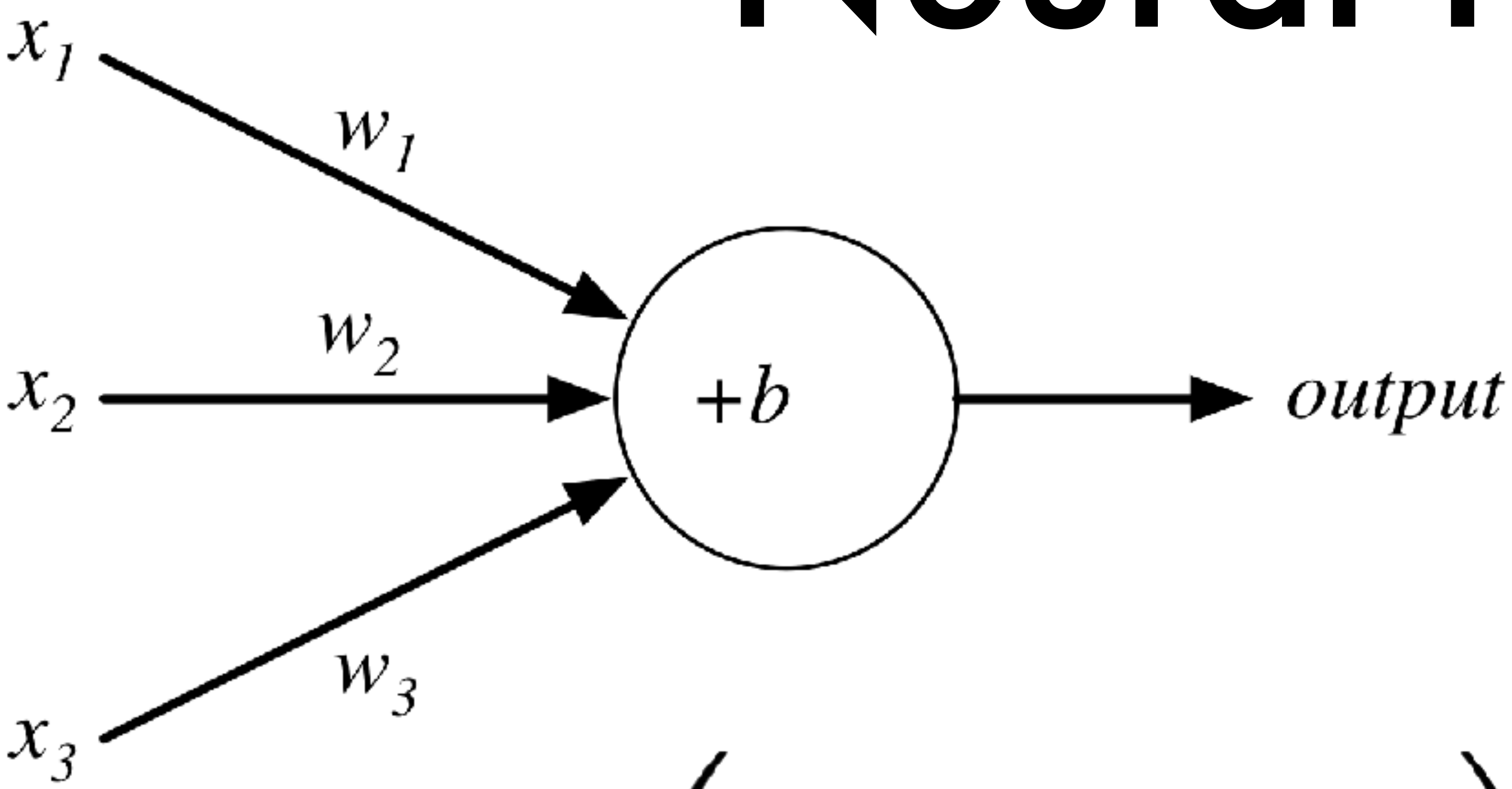
**Key Idea:** Leverage "big data"

- 231,655,275 bioactivities in PubChem
- 125,526 structures in the PDB
- 16,179 annotated complexes in PDBbind

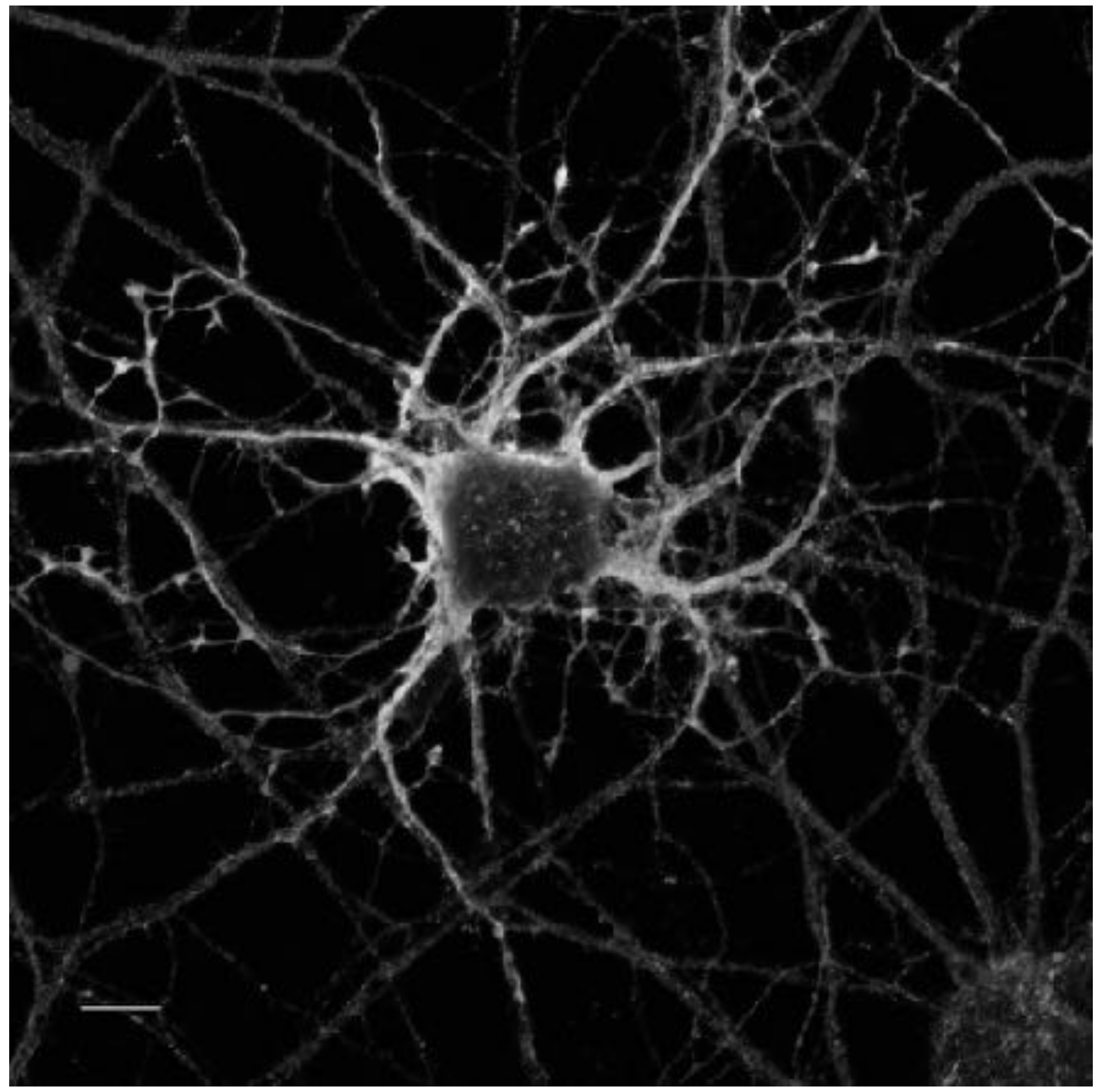
# Machine Learning

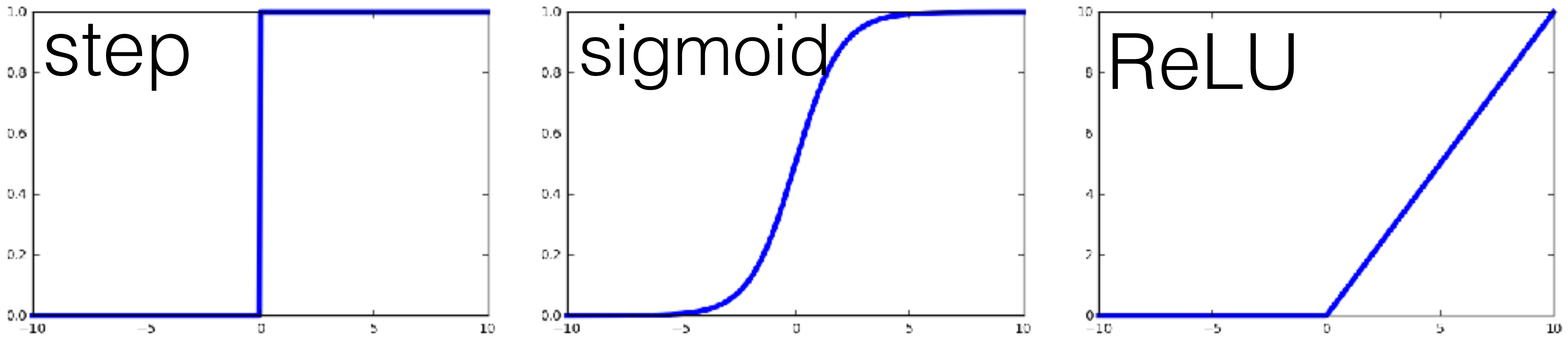
Features $X \rightarrow$ **Model** $\rightarrow y$ Prediction

# Neural Networks

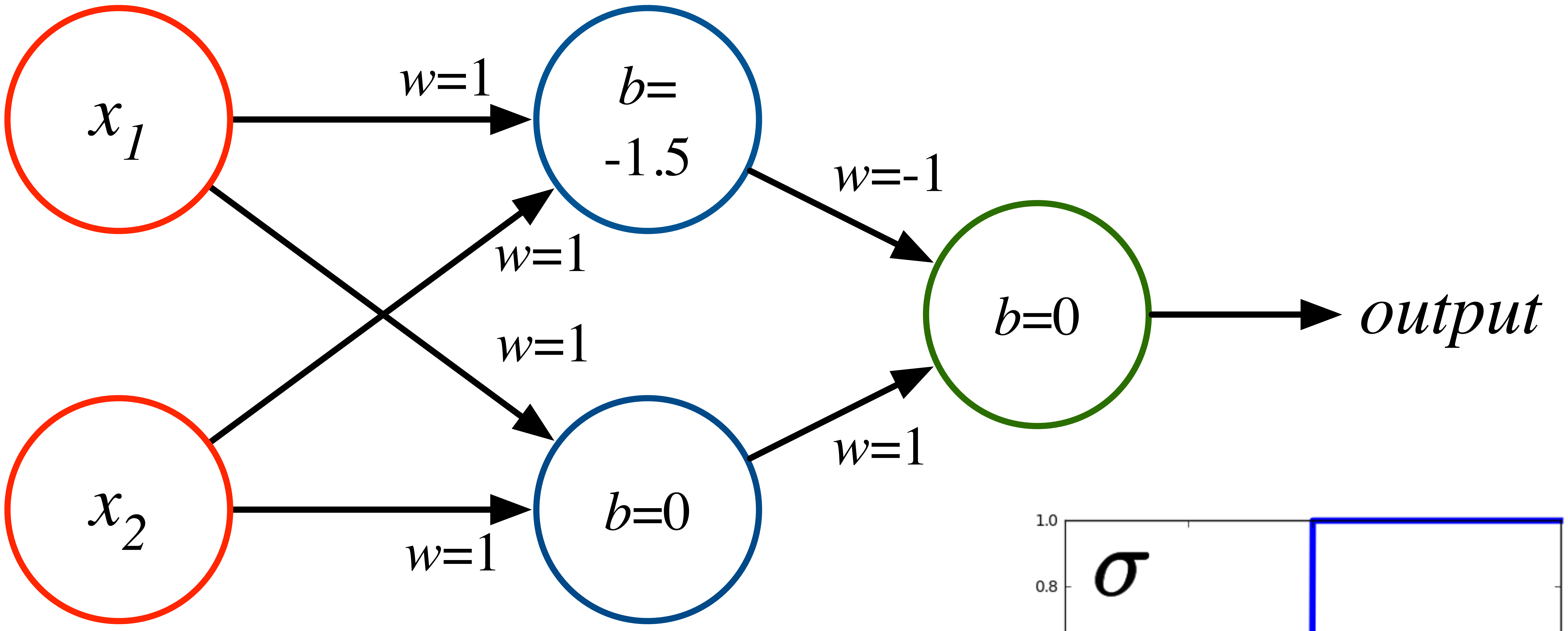


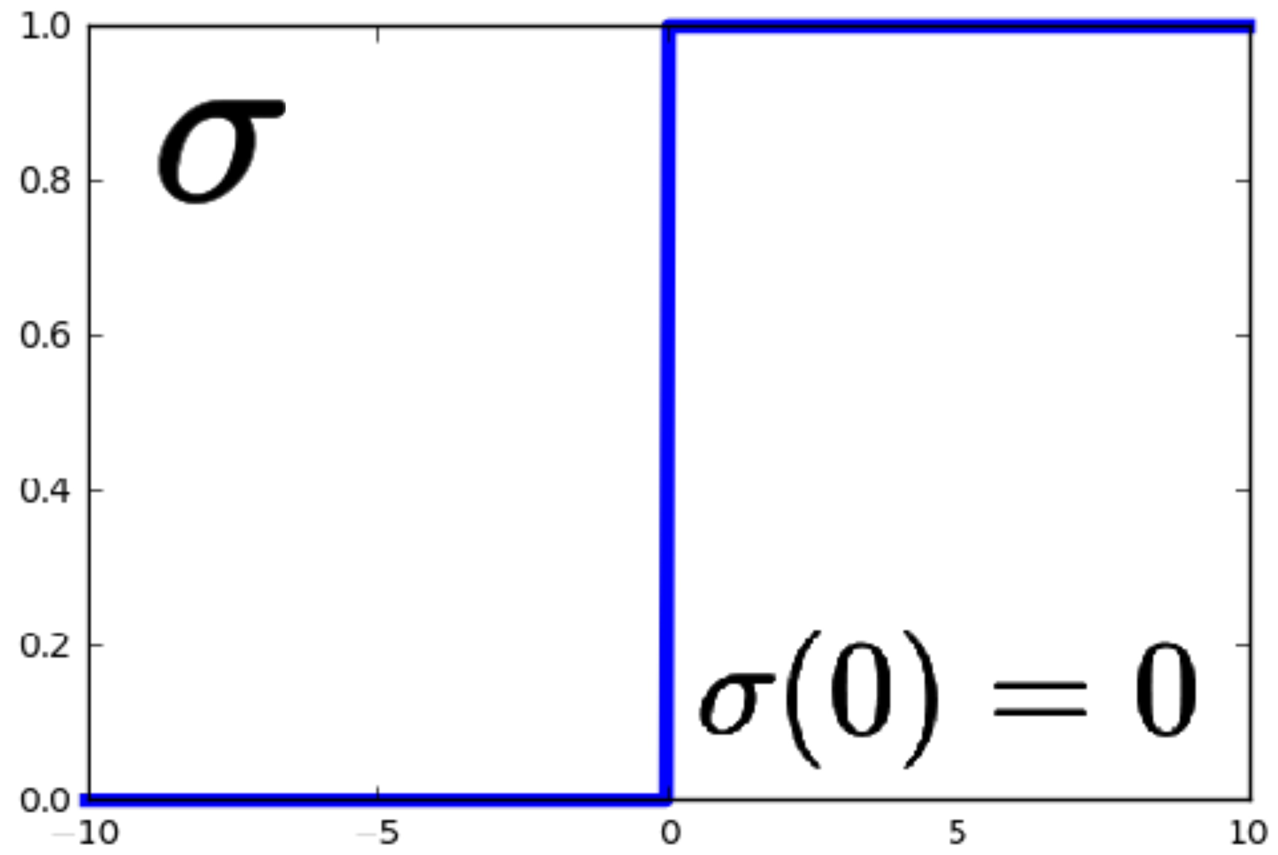

$$output = \sigma\left(\sum_i w_i x_i + b\right)$$



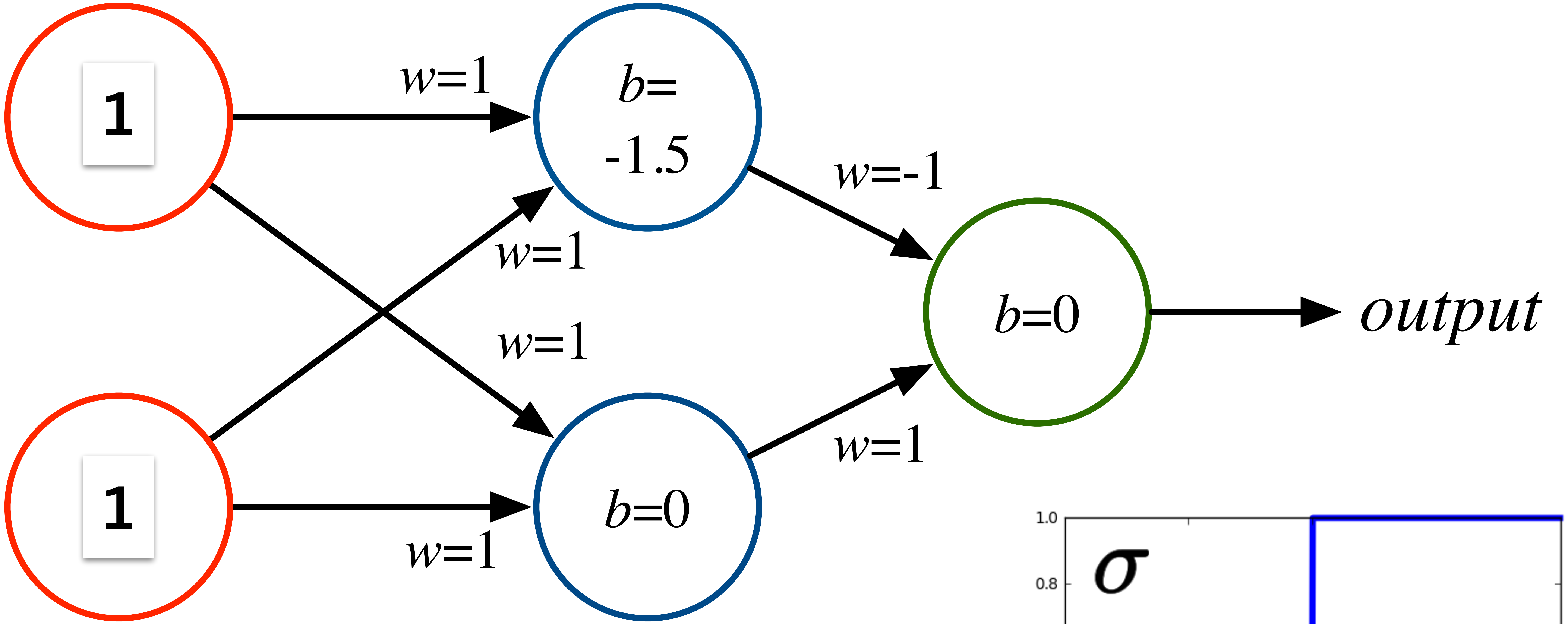


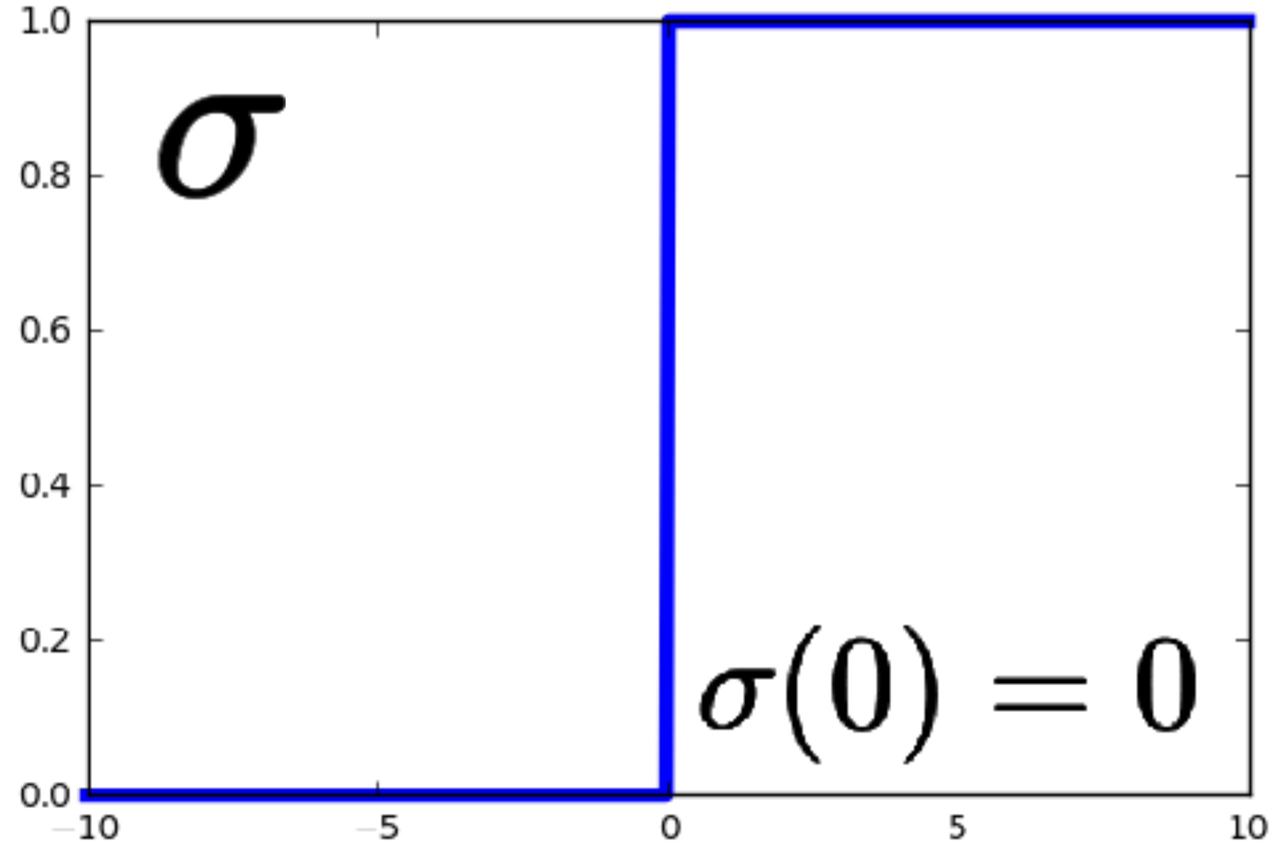
# Neural Network Example



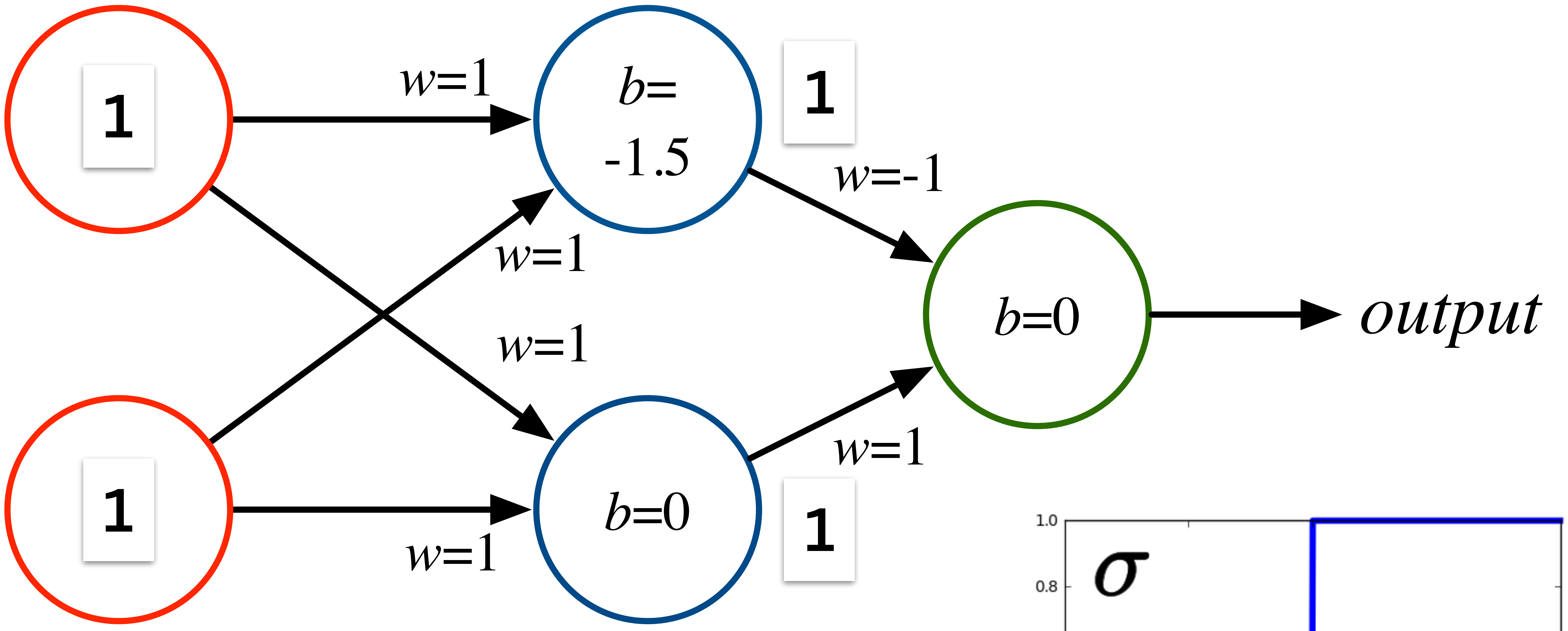



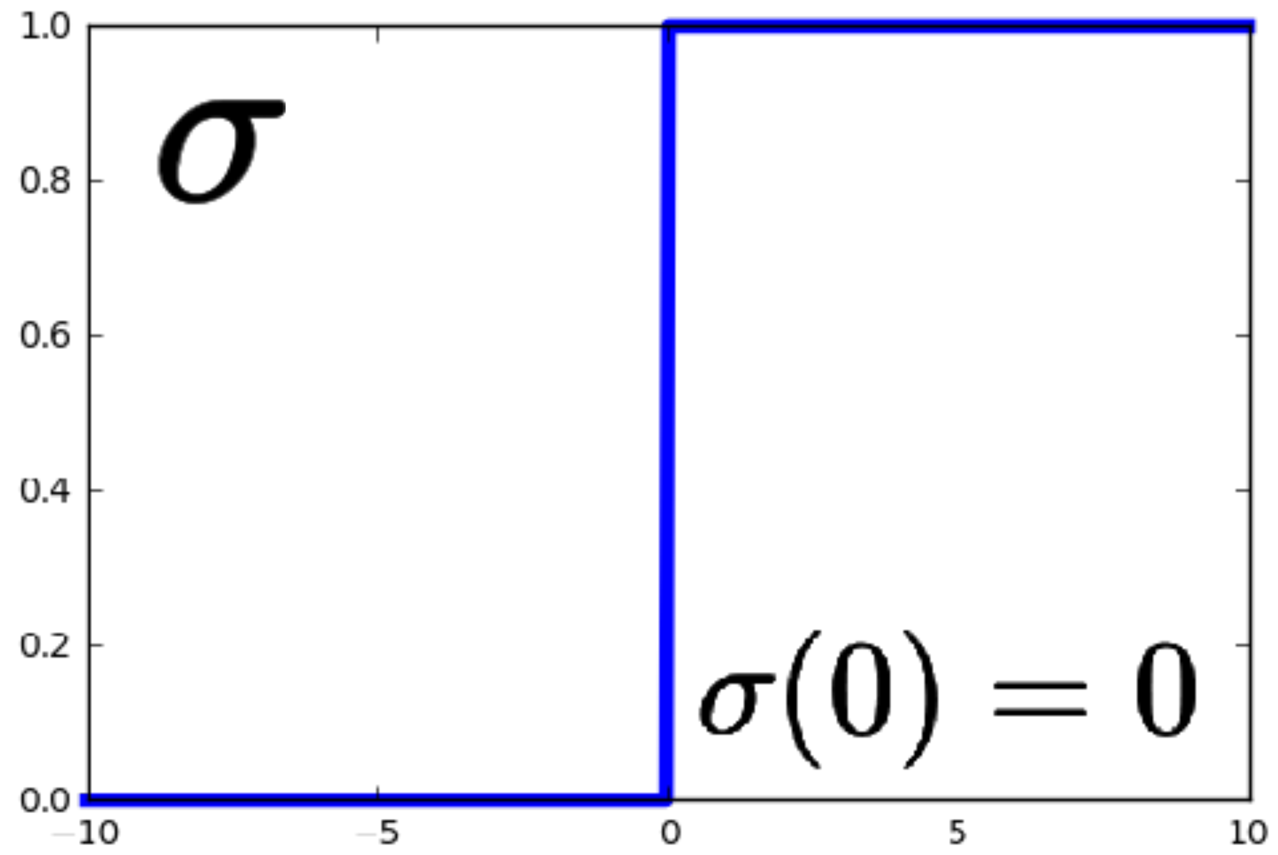
# Neural Network Example



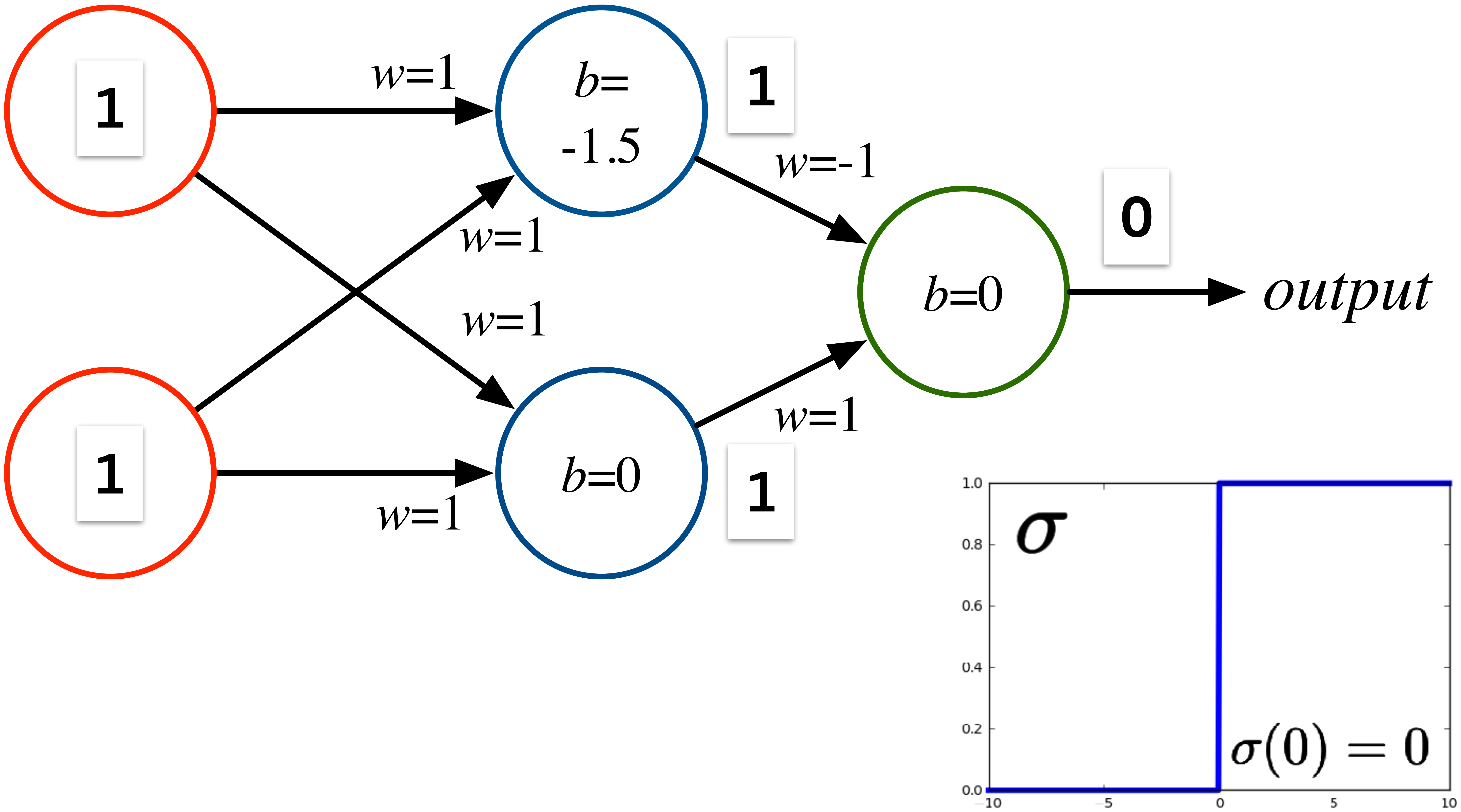



# Neural Network Example

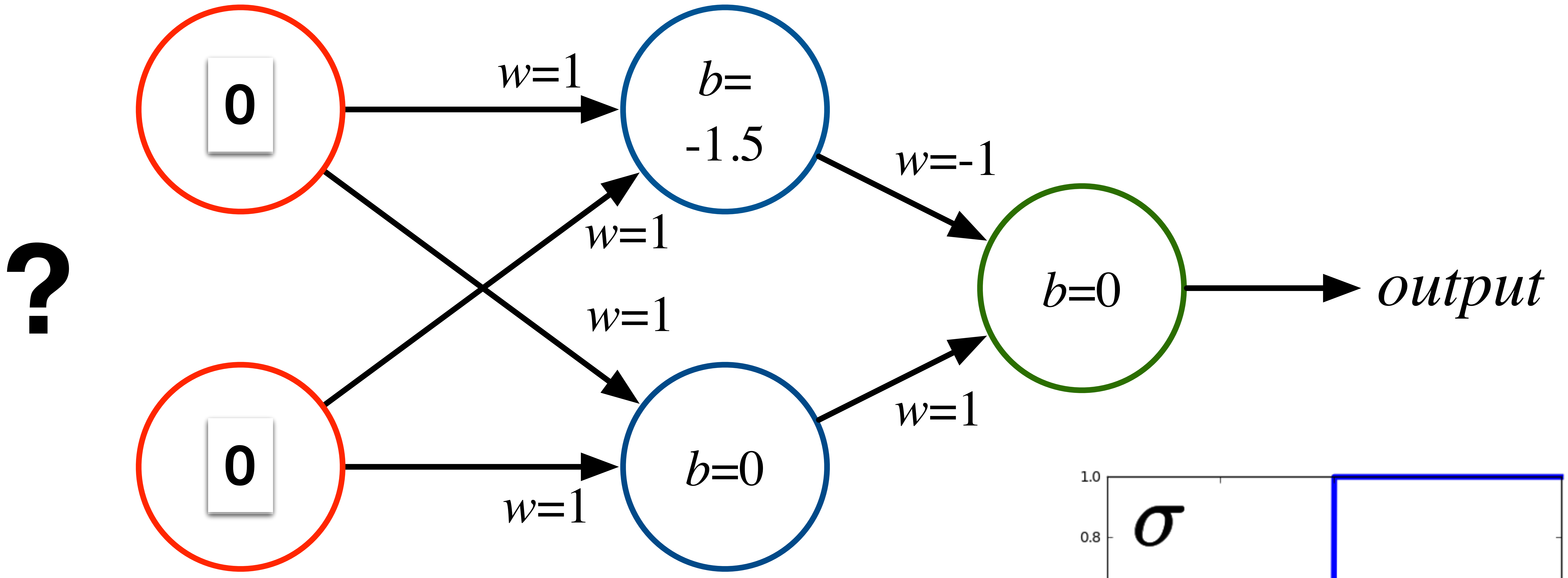


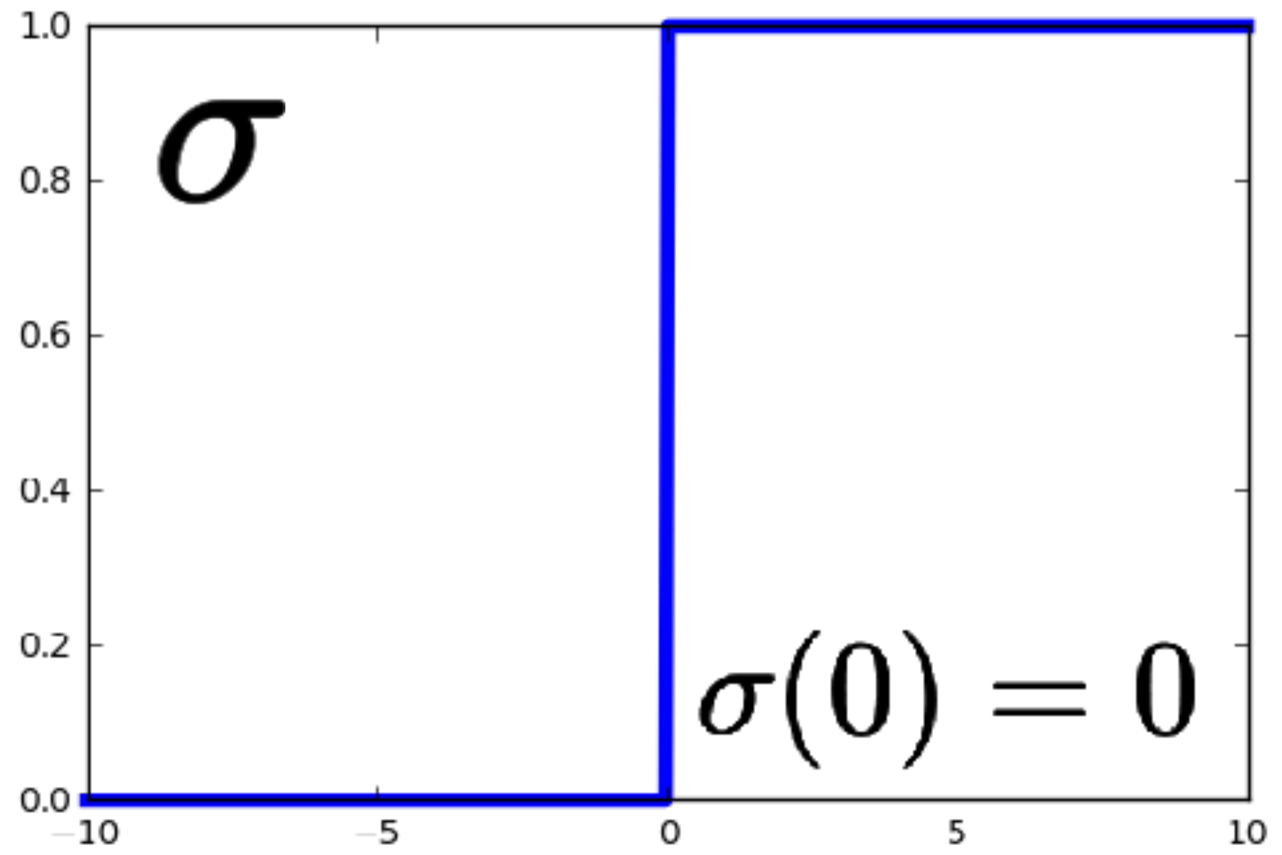



# Neural Network Example

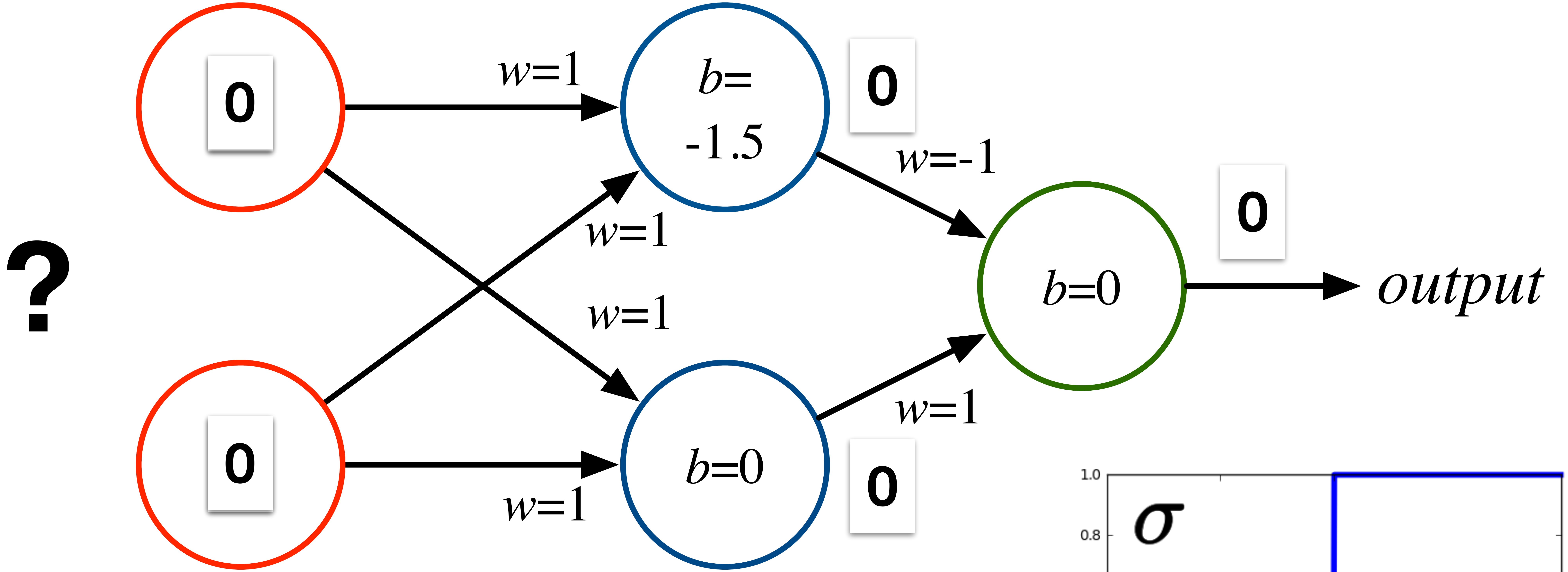


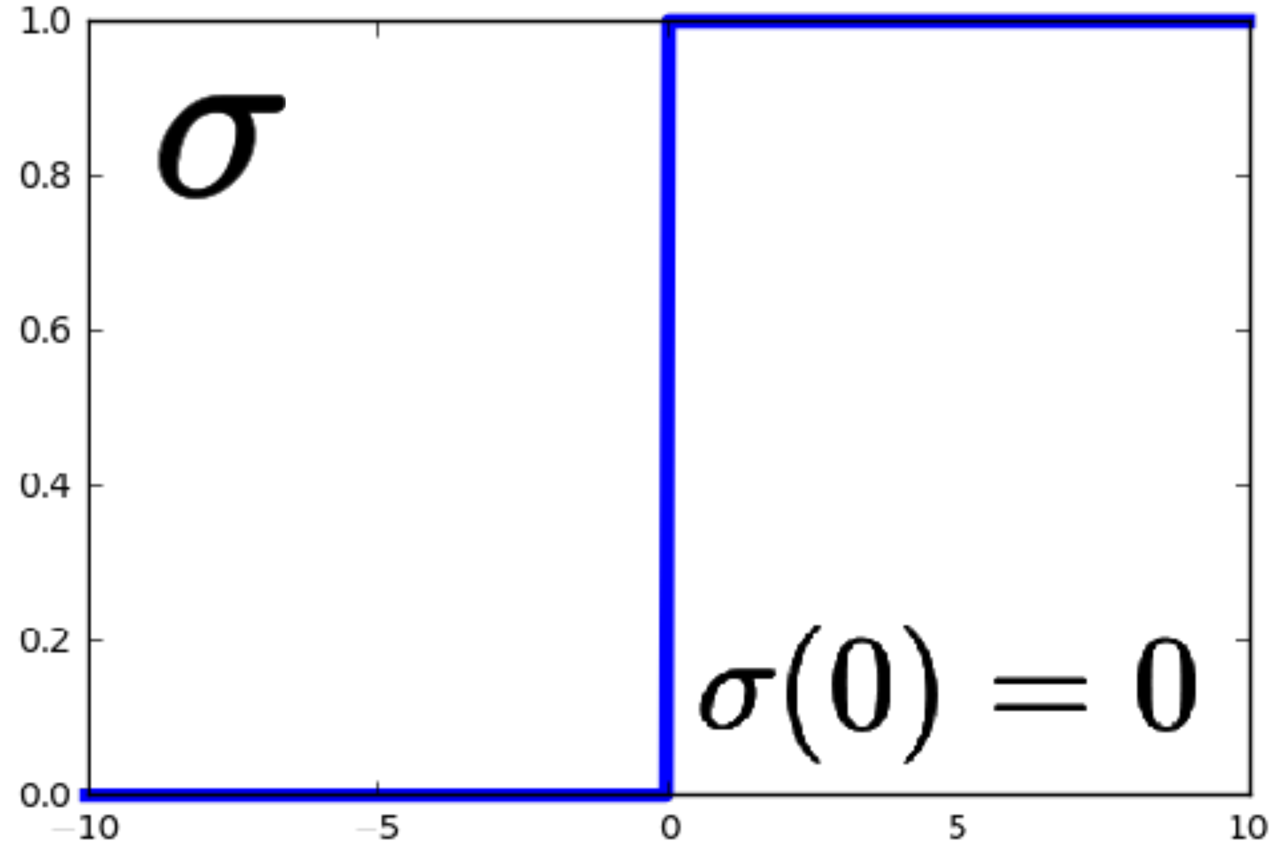
# Neural Network Example



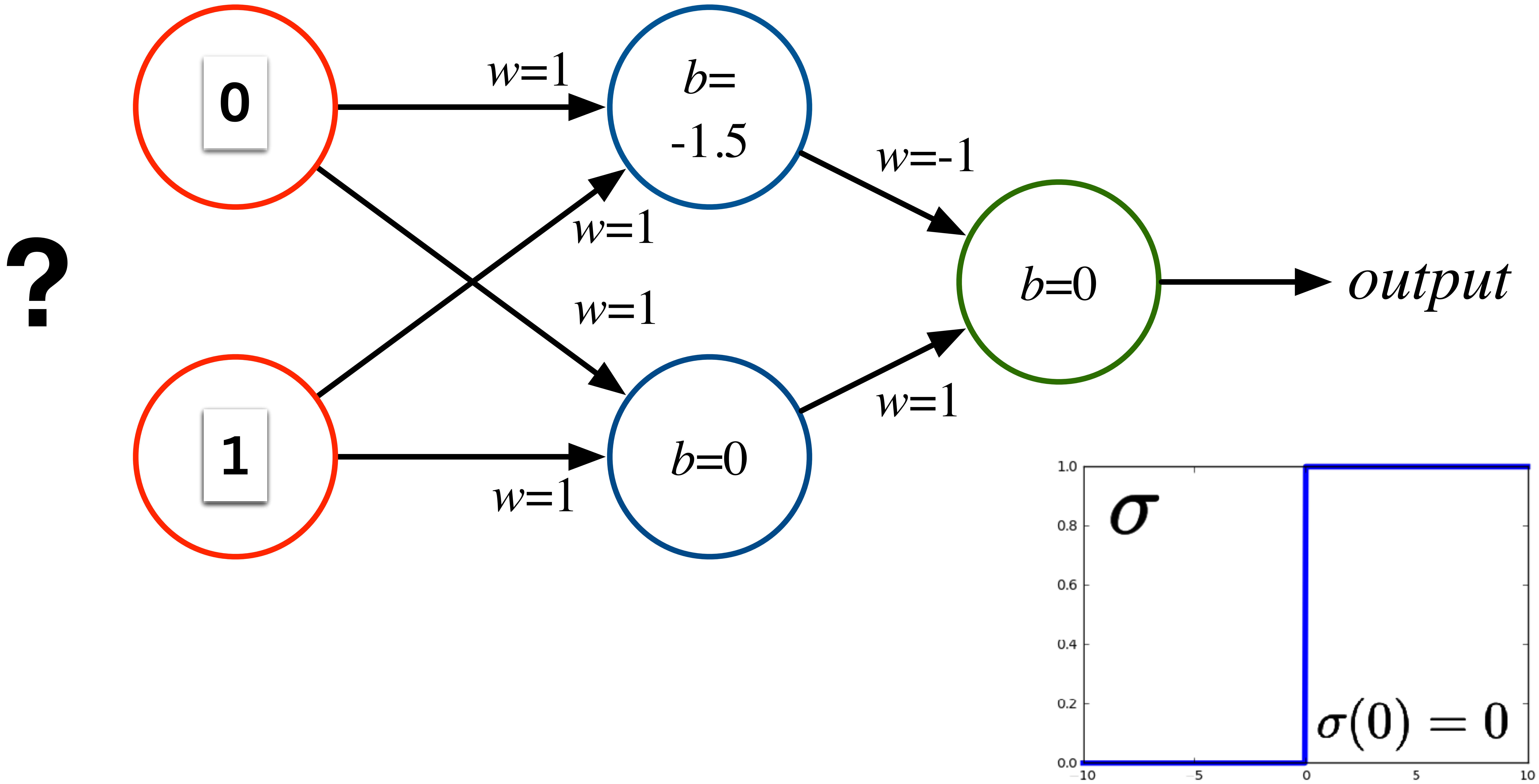



# Neural Network Example

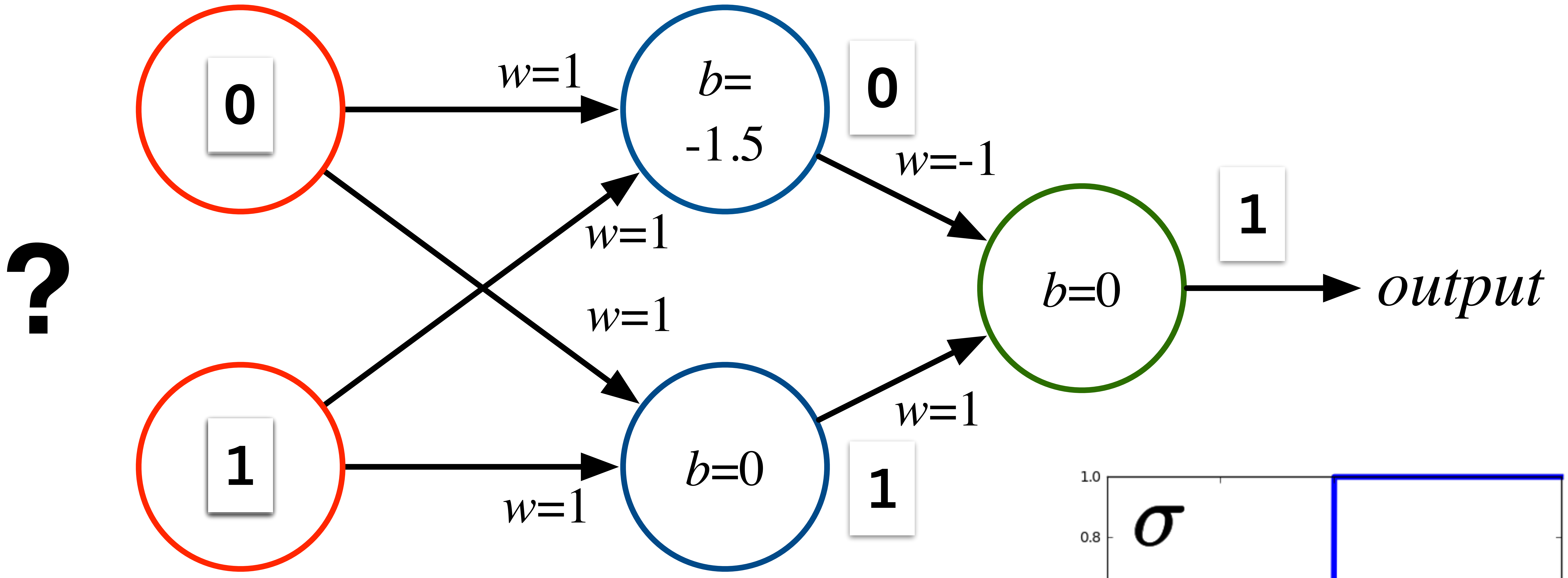


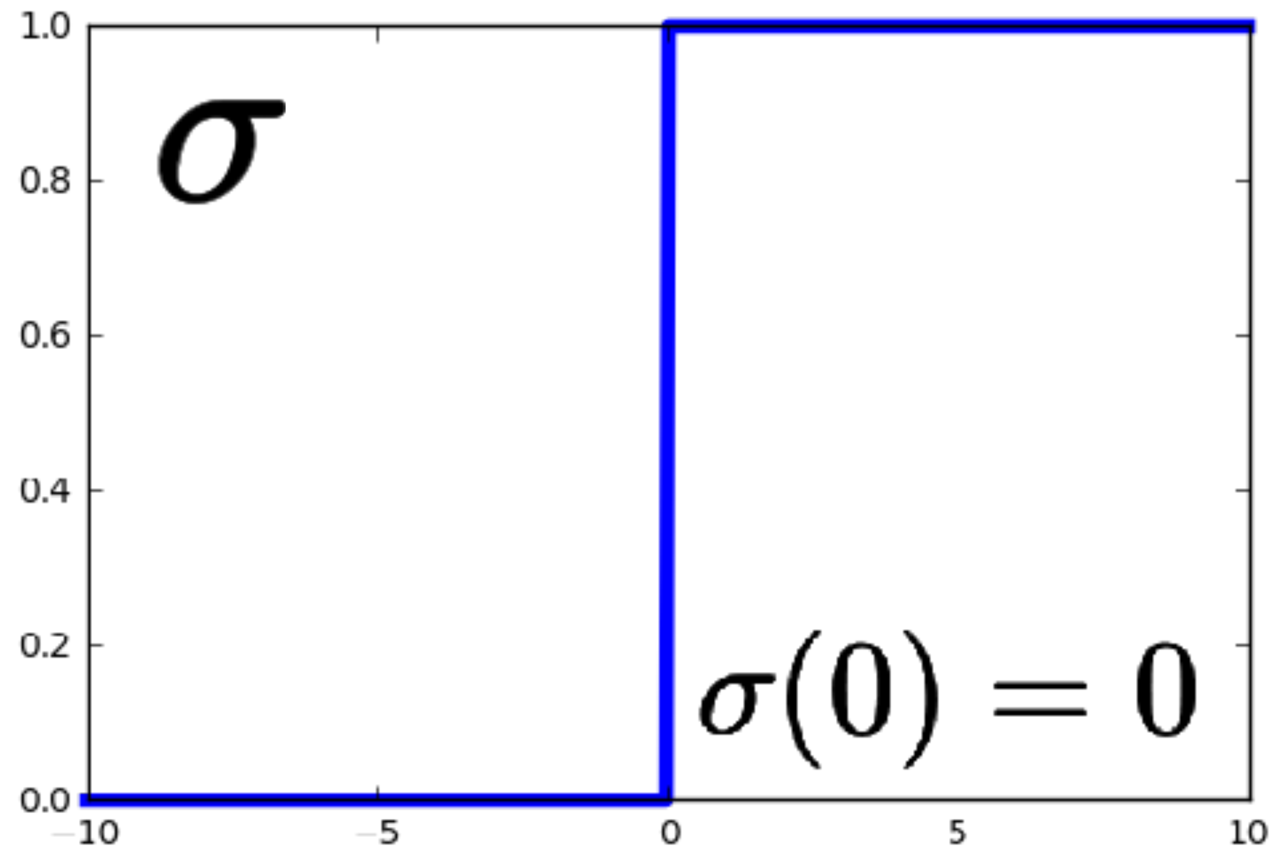



# Neural Network Example

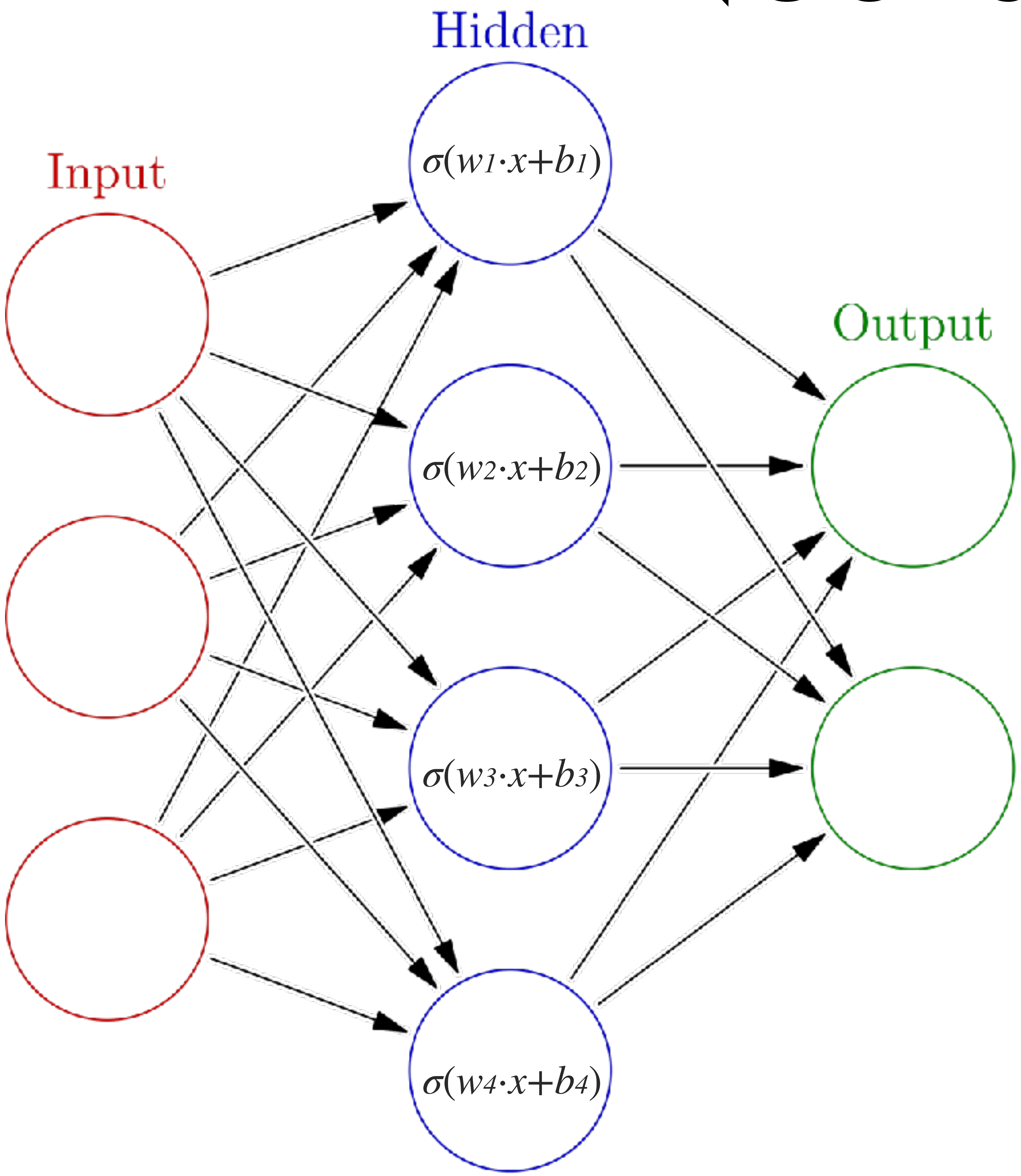


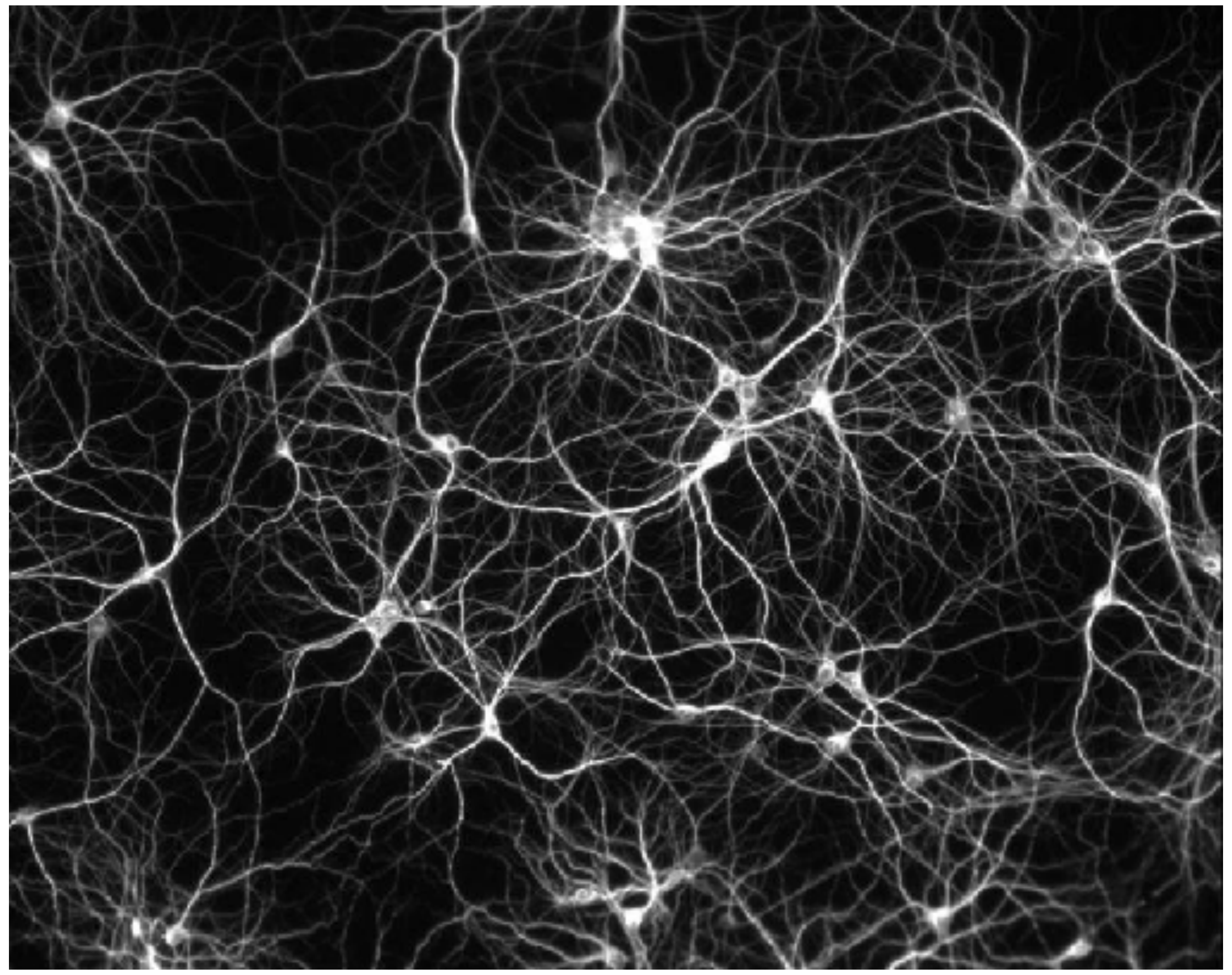
# Neural Network Example



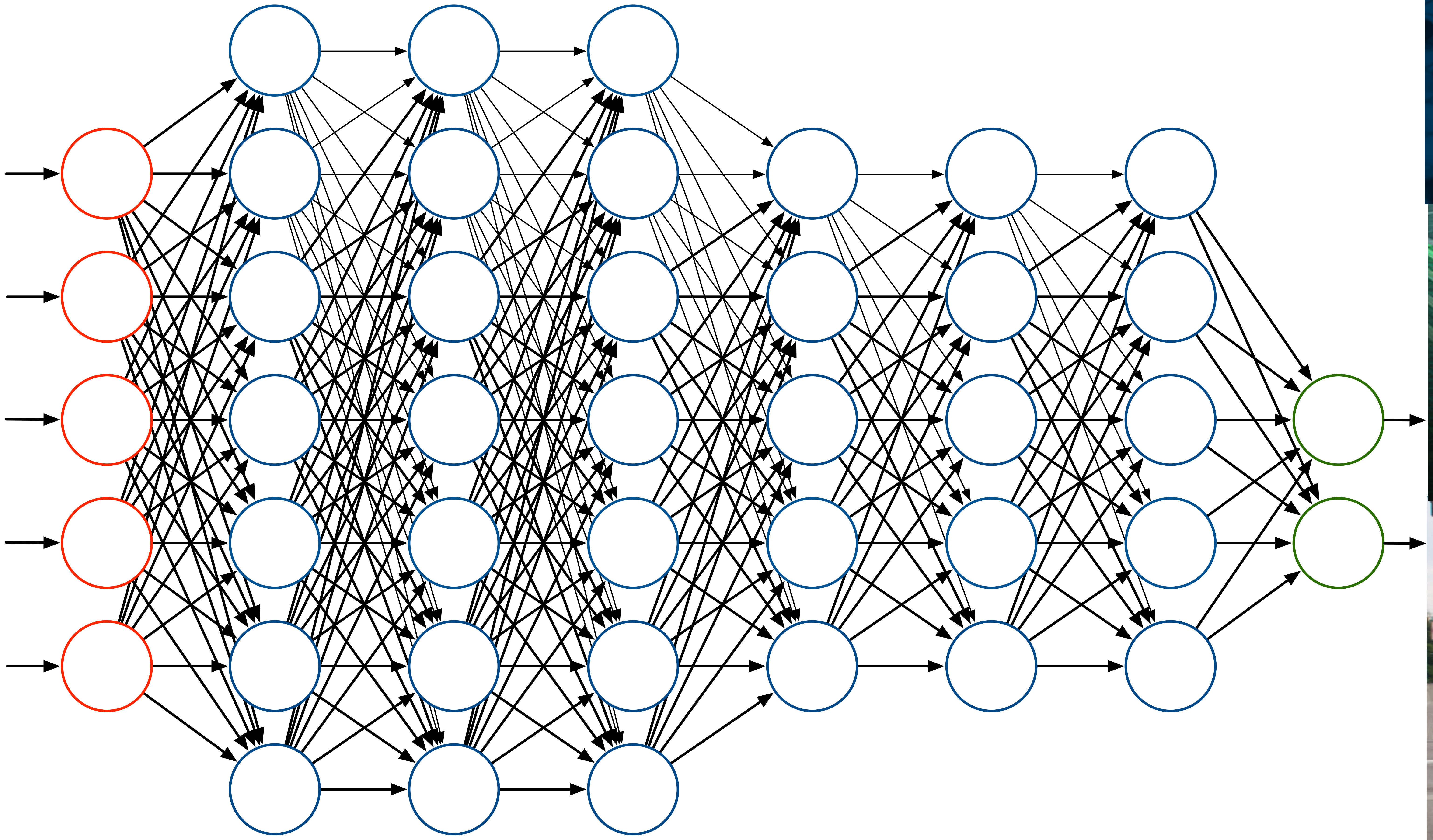
# Neural Networks



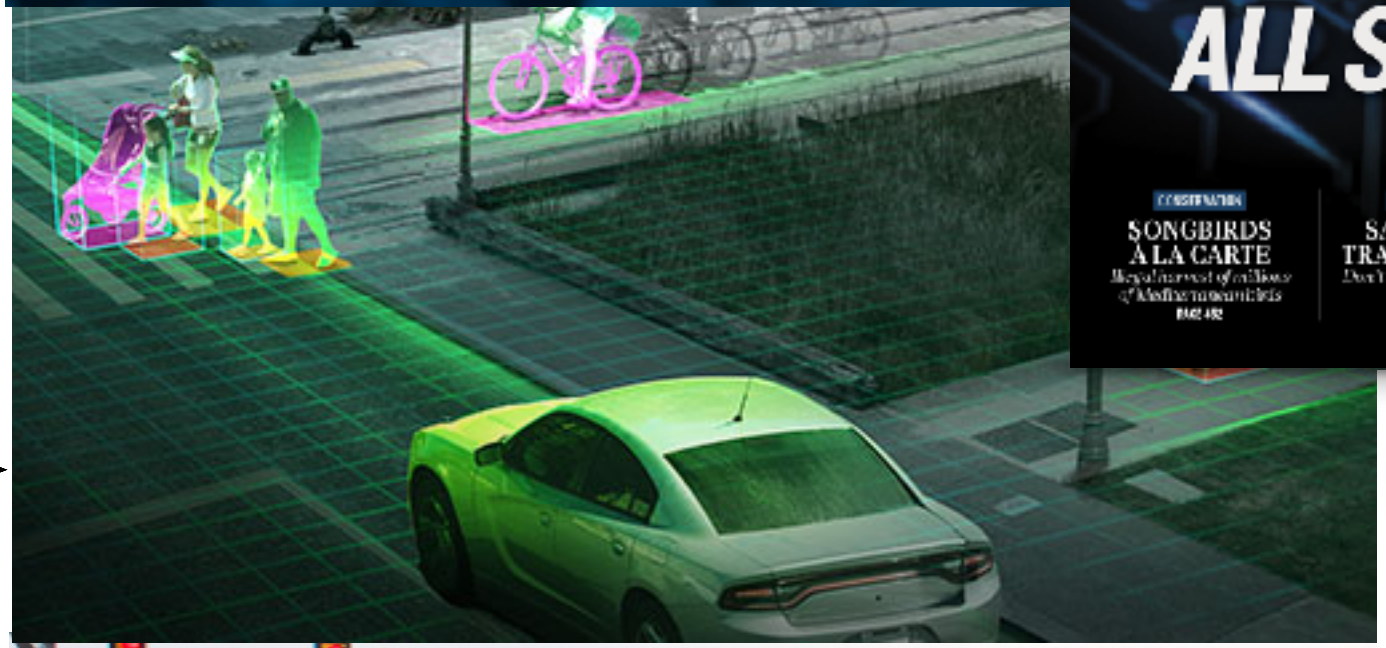



The **universal approximation theorem** states that, under reasonable assumptions, a feedforward **neural network** with a finite number of nodes **can approximate any continuous** function to within a given error over a bounded input domain.

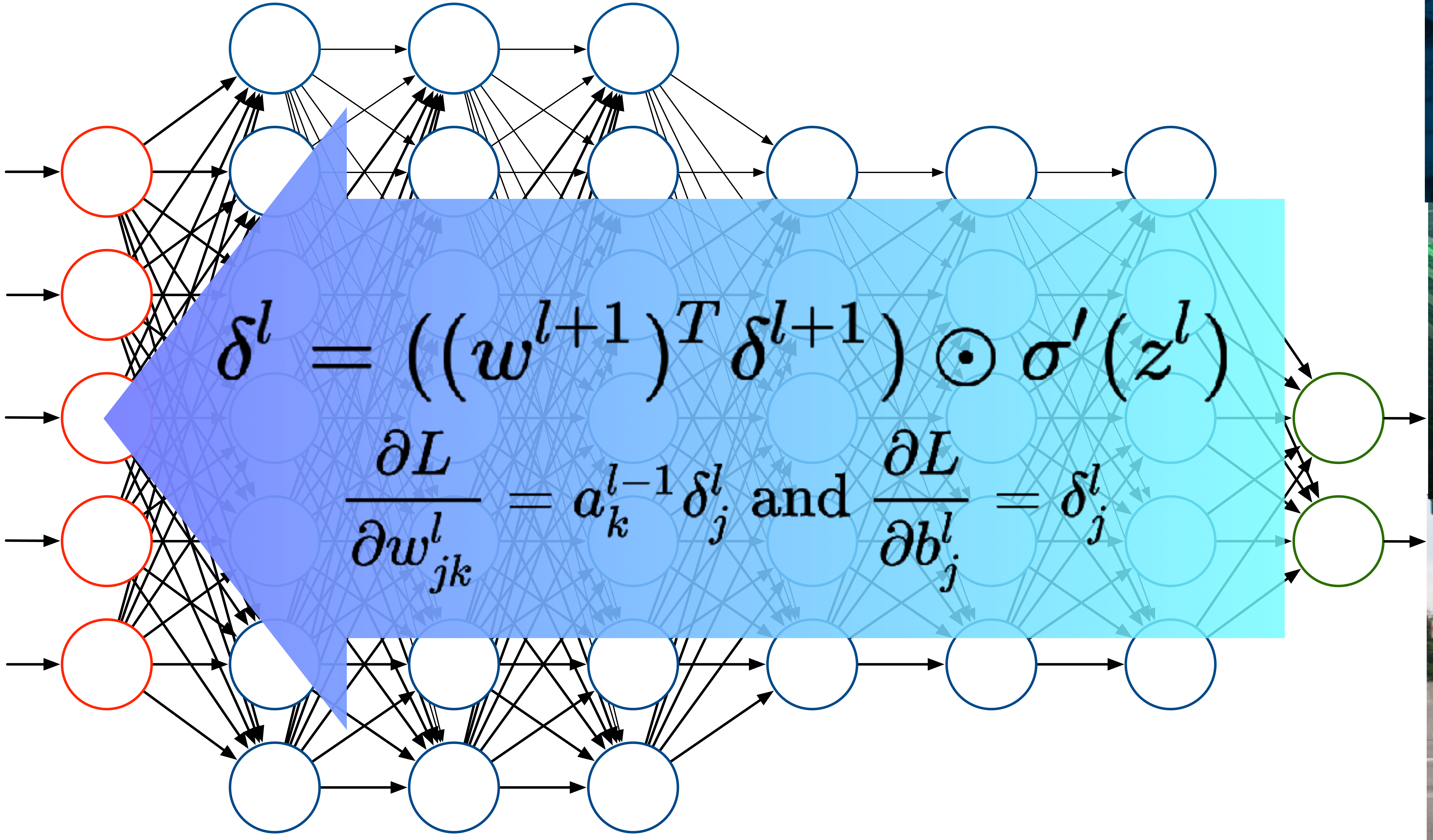
# Deep Learning







# Deep Learning

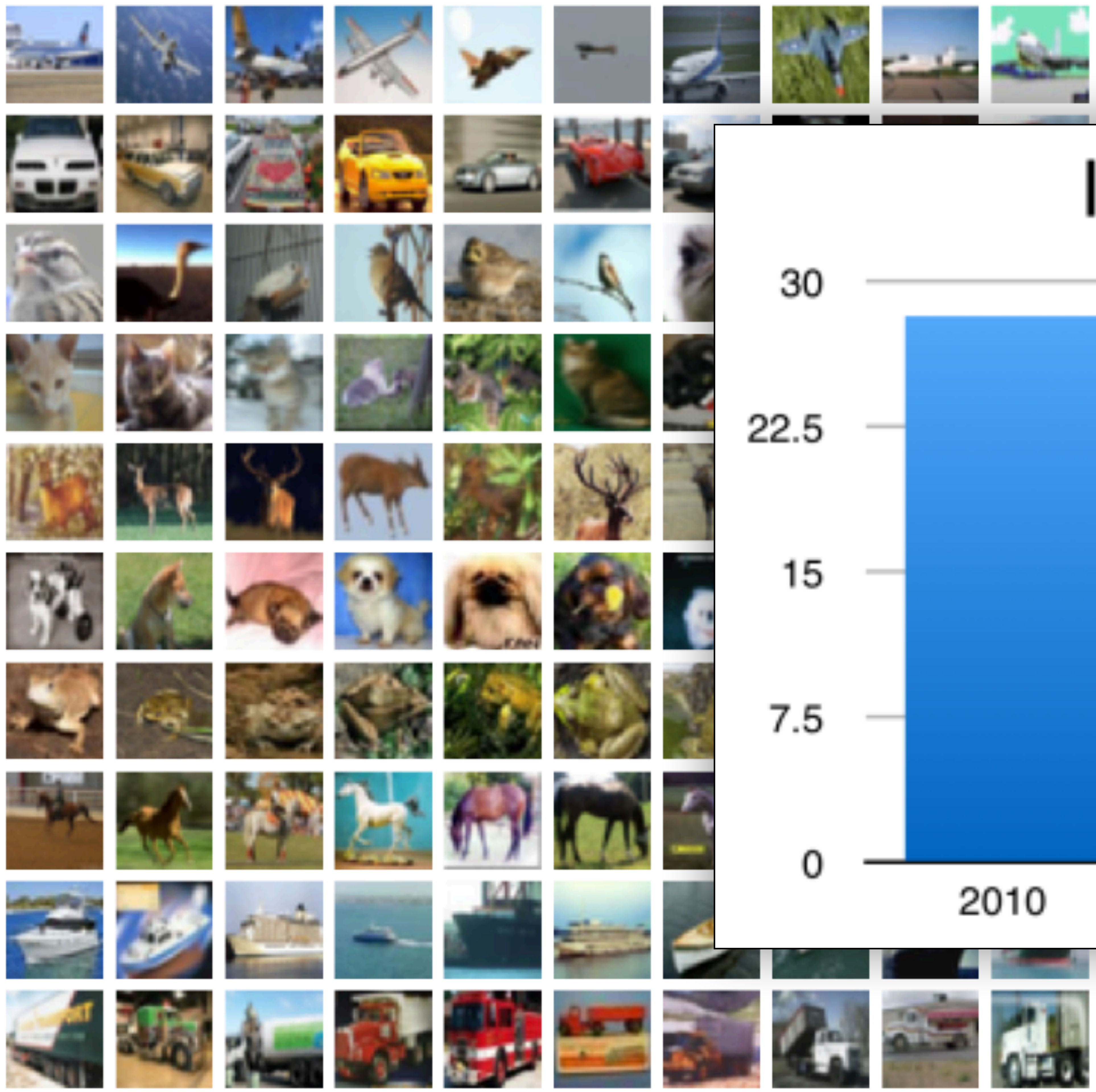


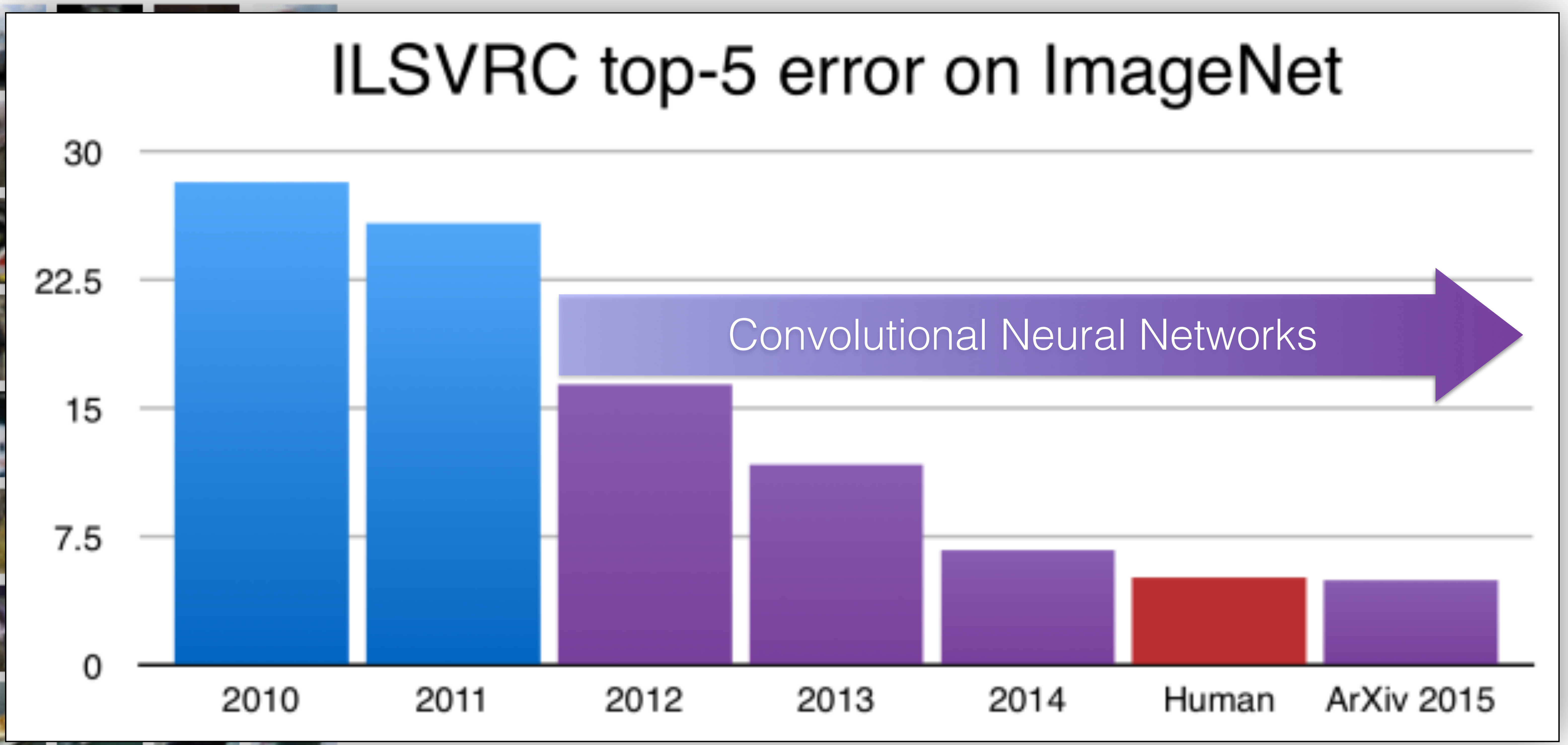
$$\delta^l = ((w^{l+1})^T \delta^{l+1}) \odot \sigma'(z^l)$$

$$\frac{\partial L}{\partial w^l_{jk}} = a^{l-1}_k \delta^l_j \text{ and } \frac{\partial L}{\partial b^l_j} = \delta^l_j$$

# Image Recognition

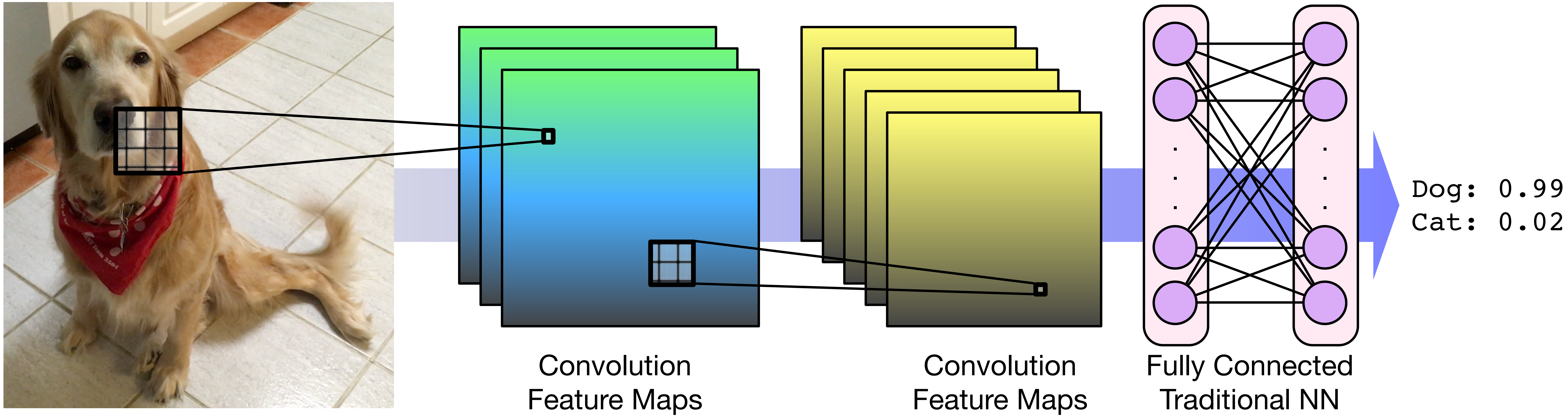
airplane
automobile
bird
cat
deer
dog
frog
horse
ship
truck

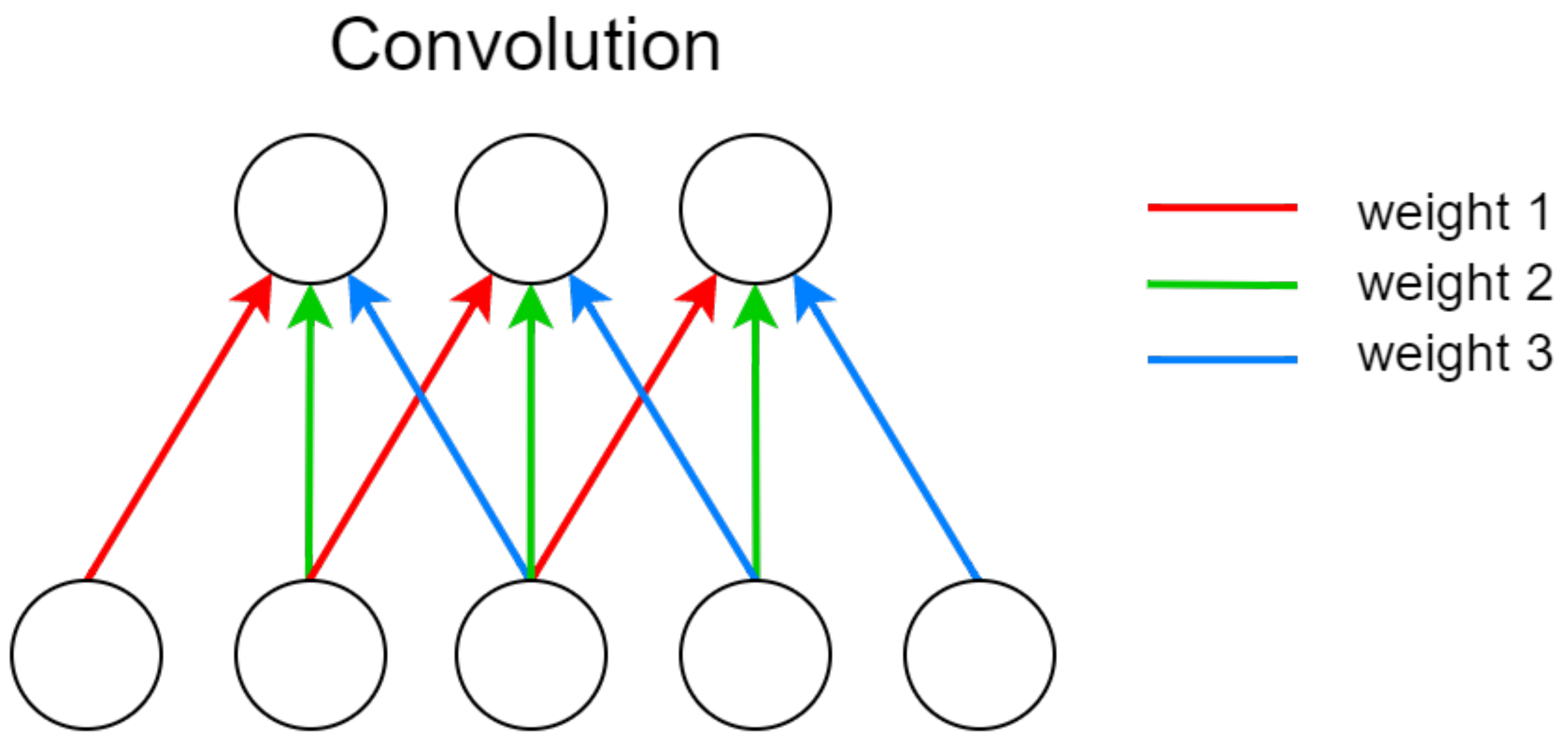


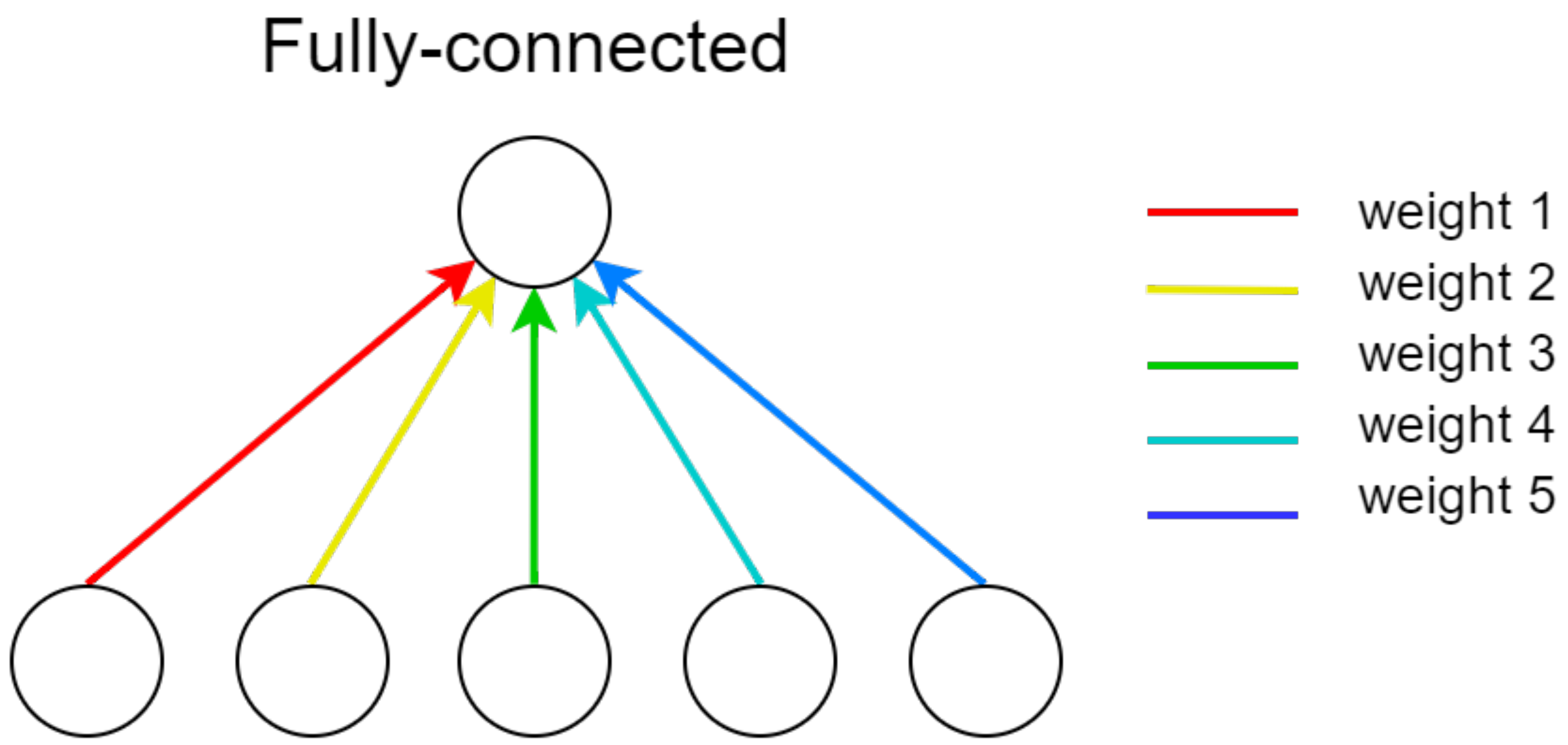



https://devblogs.nvidia.com

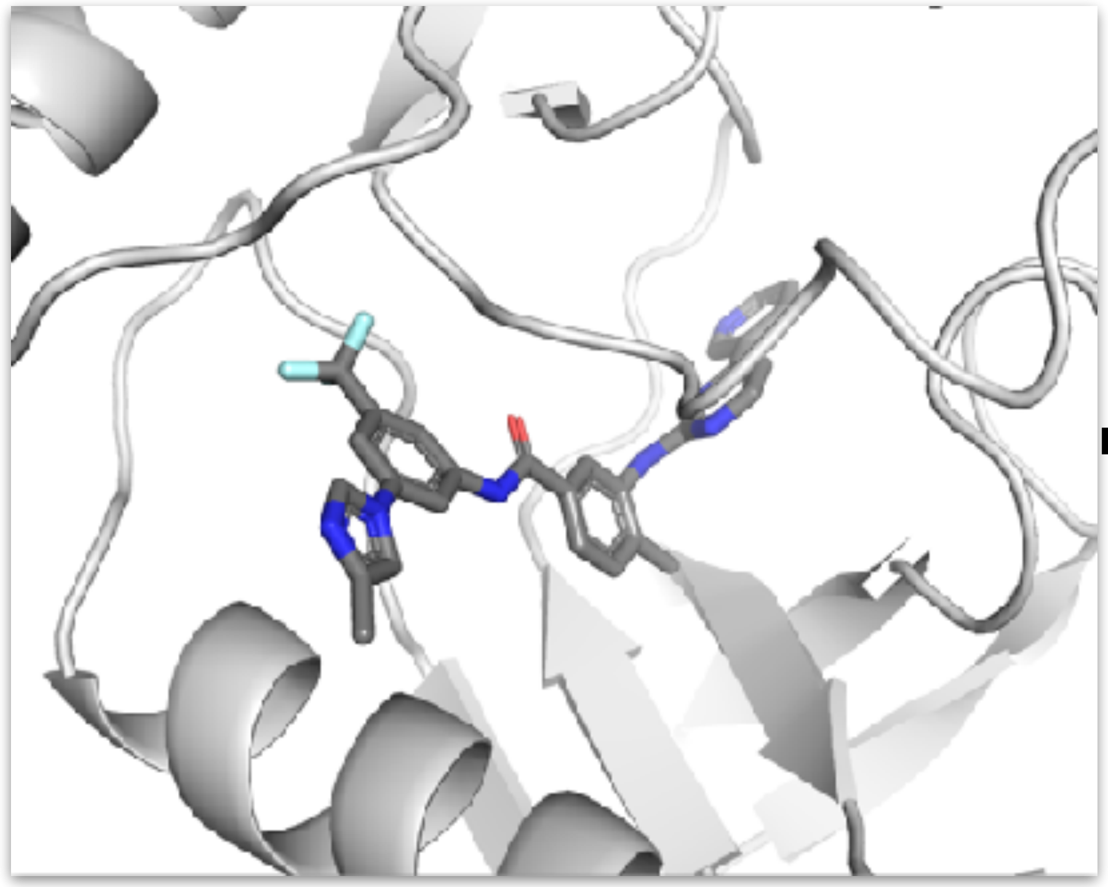
# Convolutional Neural Networks



# CNNs for Protein-Ligand Scoring



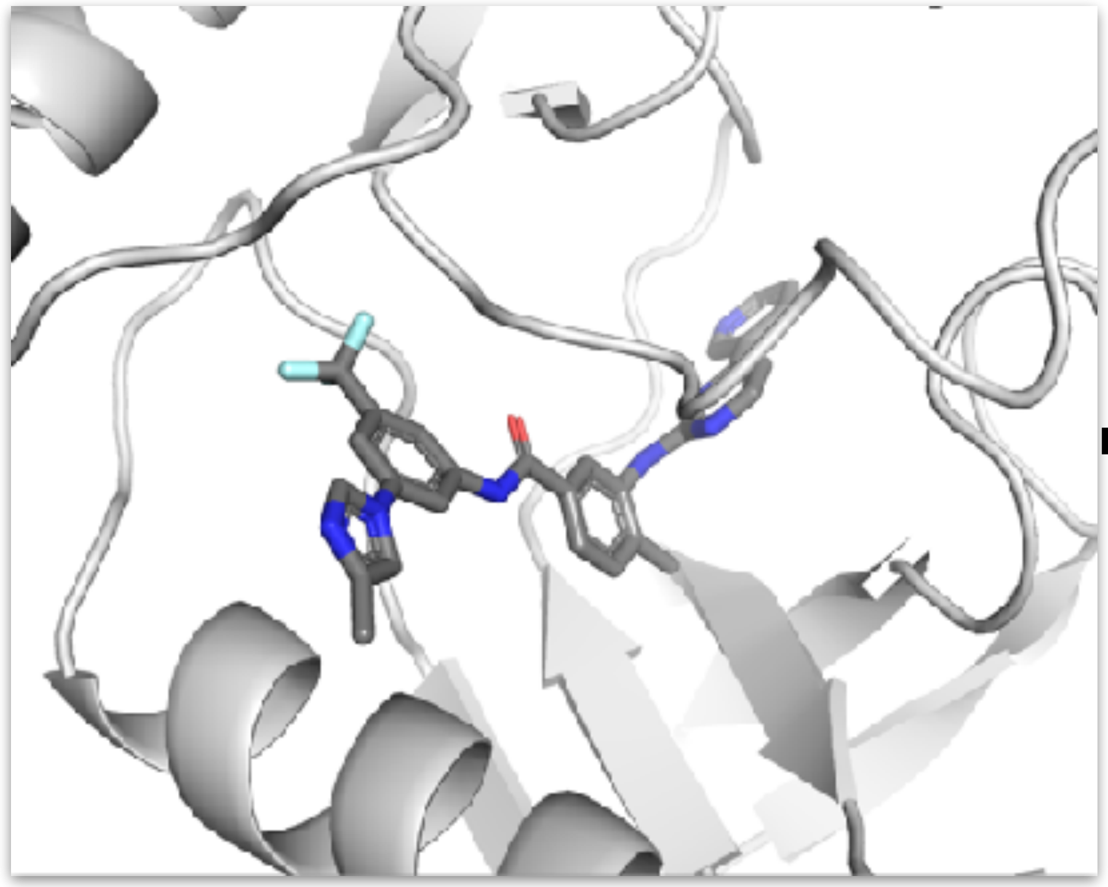
CNN

Pose Prediction

Binding Discrimination

Affinity Prediction

# CNNs for Protein-Ligand Scoring



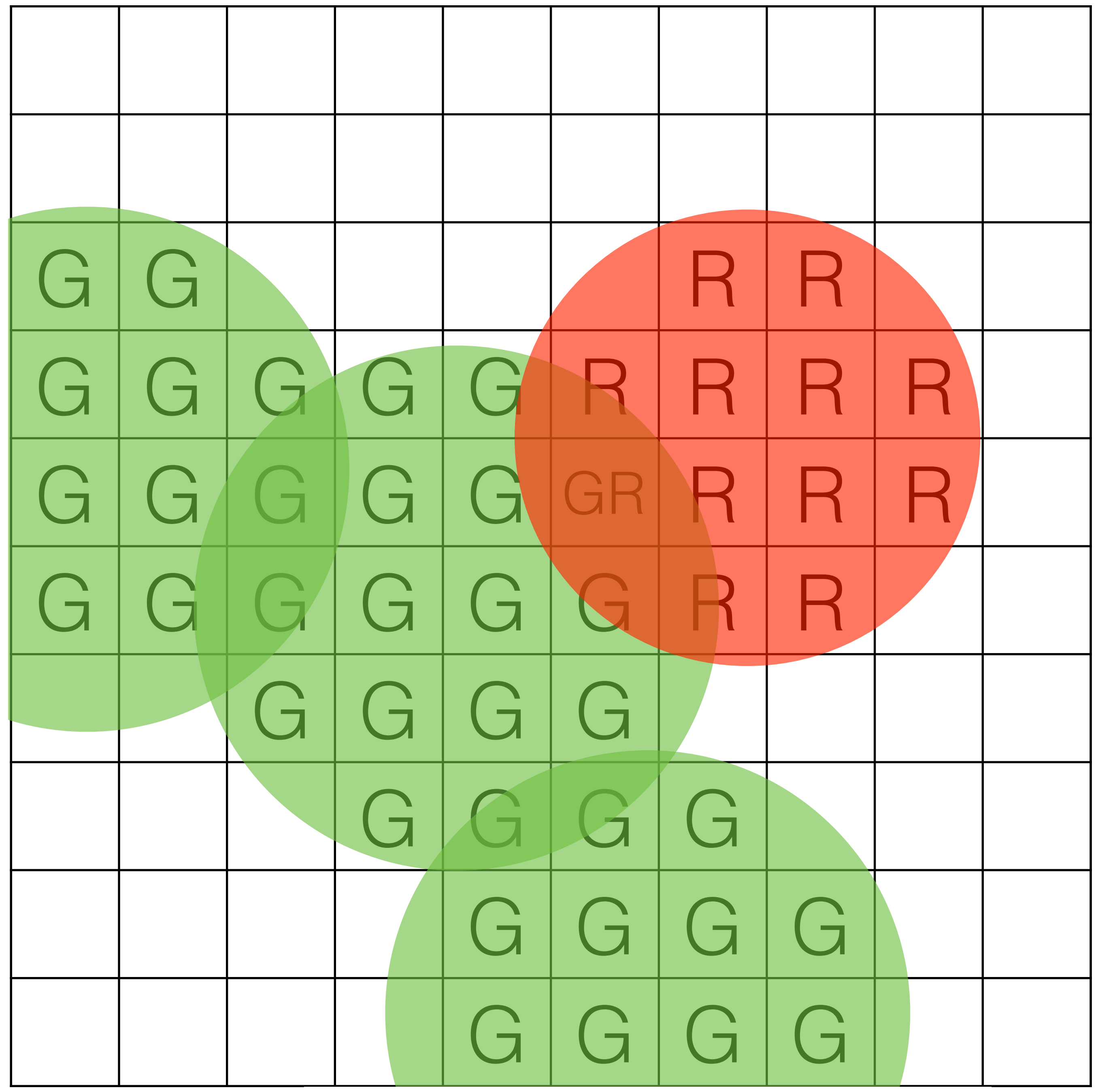
- Input representation
- Training
- Model optimization
- Visualize and Evaluation

Pose Prediction

Binding Discrimination

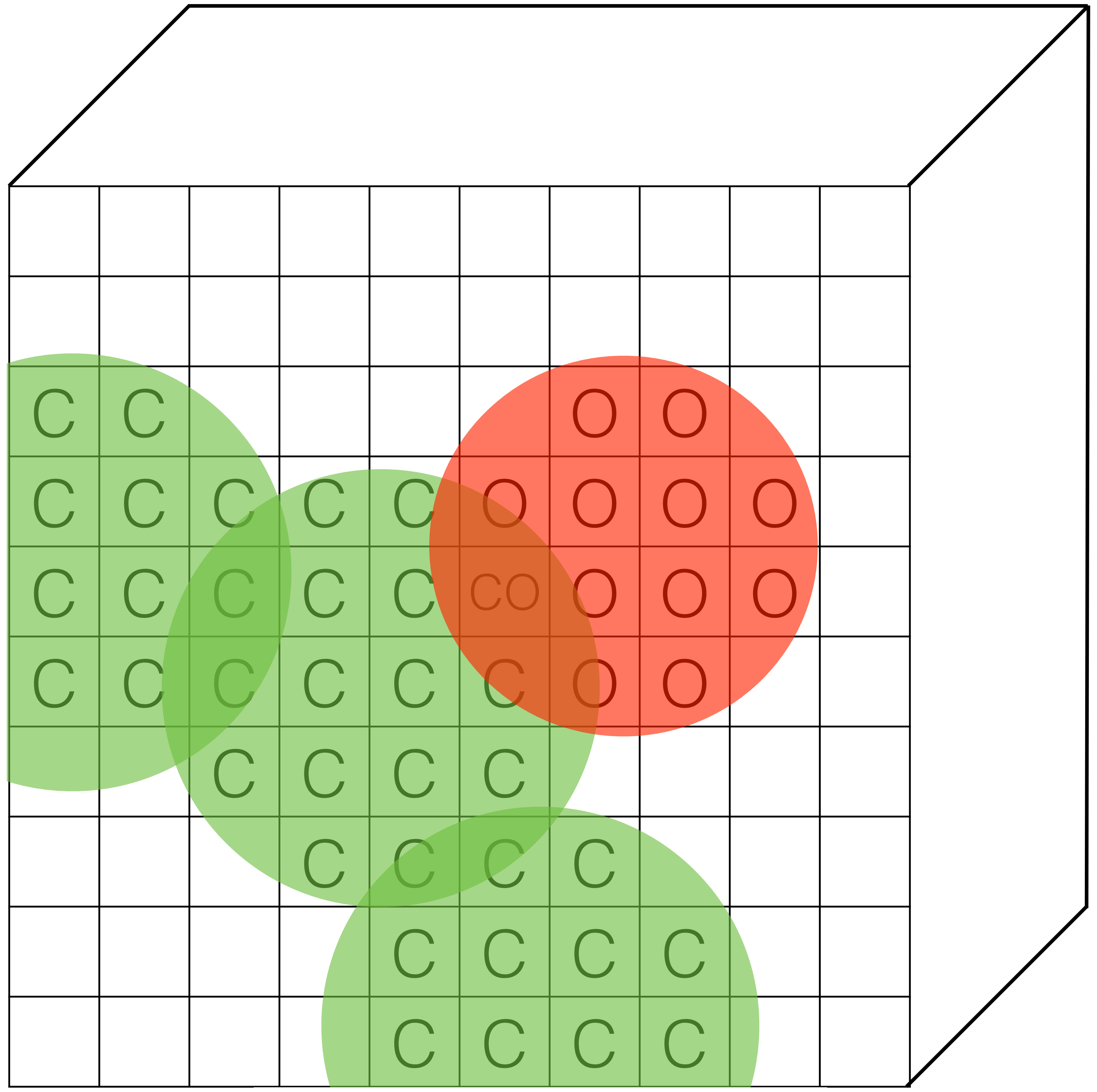
Affinity Prediction

# Protein-Ligand Representation




(R,G,B) pixel

# Protein-Ligand Representation



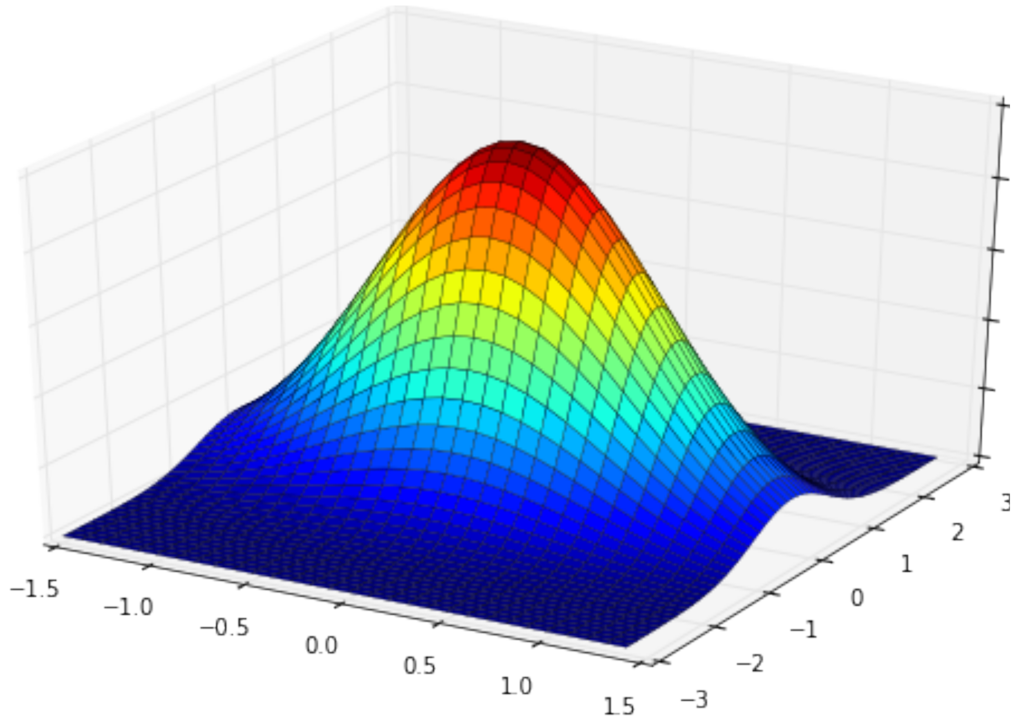

(R,G,B) pixel →

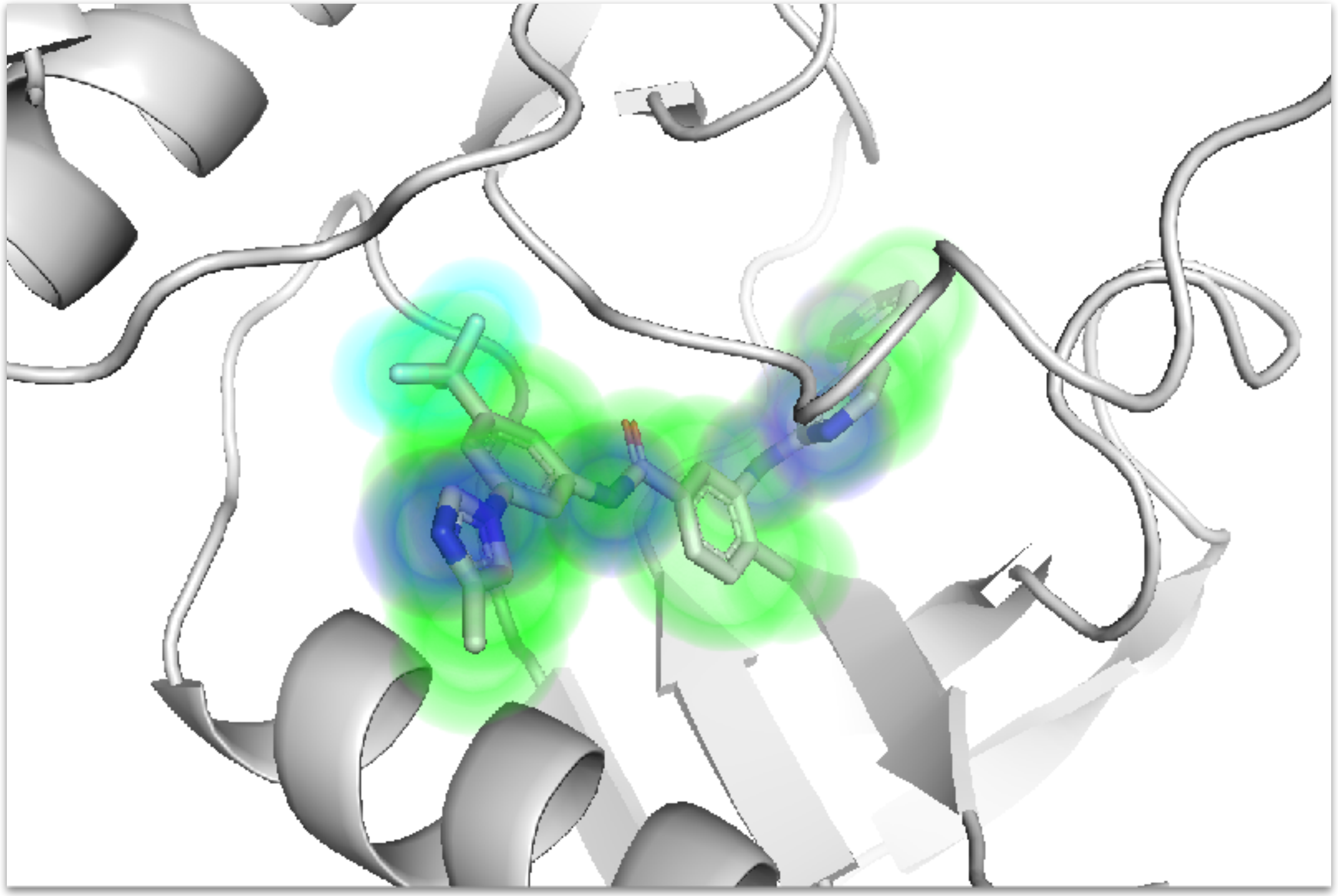
(Carbon, Nitrogen, Oxygen,…) **voxel**

The only parameters for this representation are the choice of **grid resolution**, **atom density**, and **atom types**.

# Atom Density

$$A(d,r) = \begin{cases} e^{-\frac{2d^2}{r^2}} & 0 \leq d < r \\ \frac{4}{e^2 r^2} d^2 - \frac{12}{e^2 r} d + \frac{9}{e^2} & r \leq d < 1.5r \\ 0 & d \geq 1.5r \end{cases}$$



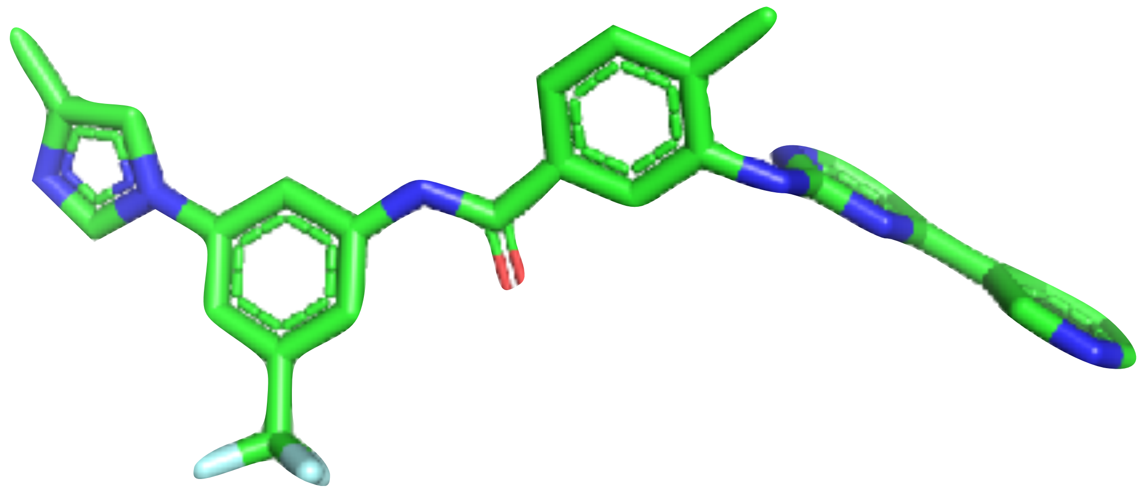



Gaussian

# Atom Types

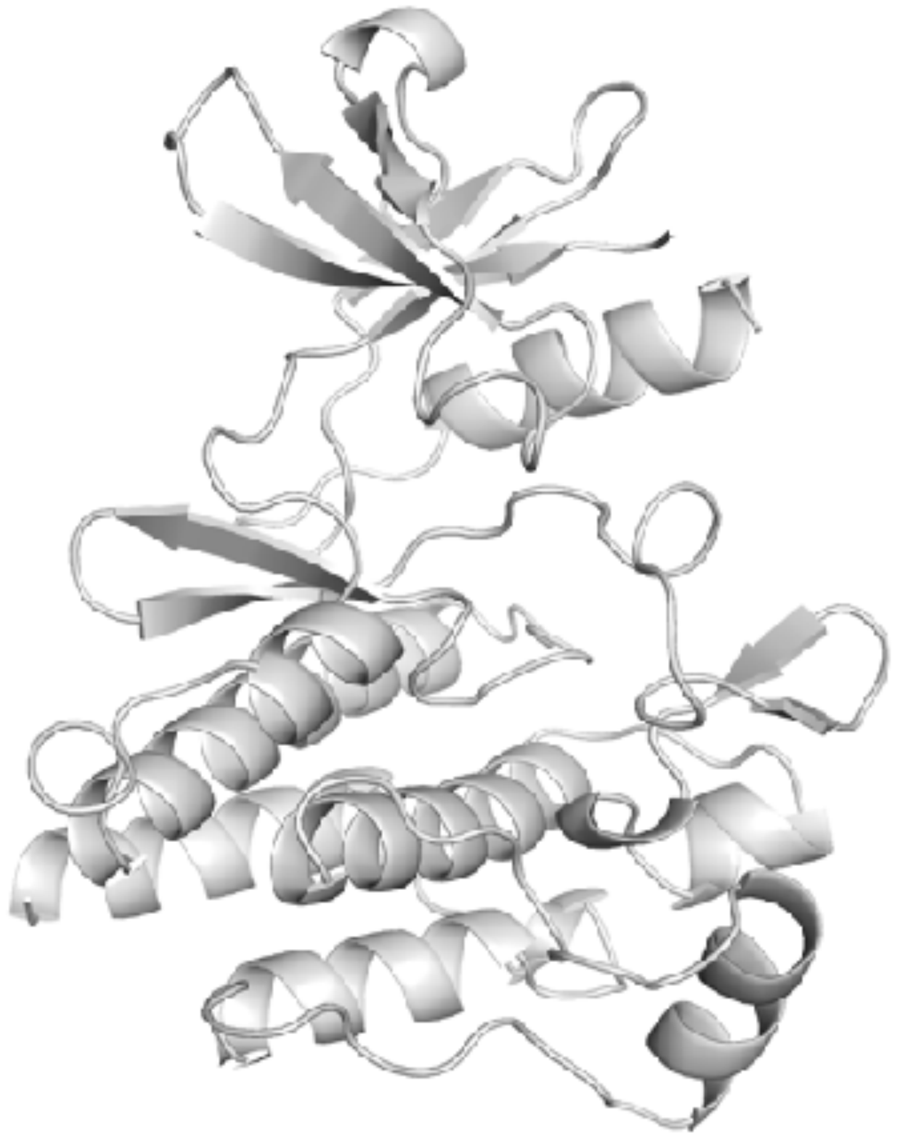
**Ligand**

AliphaticCarbonXSHydrophobe
AliphaticCarbonXSNonHydrophobe
AromaticCarbonXSHydrophobe
AromaticCarbonXSNonHydrophobe
Bromine
Chlorine
Fluorine
Iodine
Nitrogen
NitrogenXSAcceptor
NitrogenXSDonor
NitrogenXSDonorAcceptor
Oxygen
OxygenXSAcceptor
OxygenXSDonorAcceptor
Phosphorus
Sulfur
SulfurAcceptor



**Receptor**

AliphaticCarbonXSHydrophobe
AliphaticCarbonXSNonHydrophobe
AromaticCarbonXSHydrophobe
AromaticCarbonXSNonHydrophobe
Calcium
Iron
Magnesium
Nitrogen
NitrogenXSAcceptor
NitrogenXSDonor
NitrogenXSDonorAcceptor
OxygenXSAcceptor
OxygenXSDonorAcceptor
Phosphorus
Sulfur
Zinc



# Training Data

## Pose Prediction

337 protein-ligand complexes

- curated for electron density
- diverse targets
- <10μM affinity
- **generate poses** with Vina
  - 745 <2Å RMSD (actives)
  - 3251 >4Å RMSD (decoys)

4056 protein-ligand complexes

- diverse targets
- wide range of affinities
- **generate poses** with AutoDock Vina
- include minimized crystal pose
  - 8,688 <2Å RMSD (actives)
  - 76,743 >4Å RMSD (decoys)

# Training Data

## Binding Discrimination

DUD•E

102 targets
- 22,645 actives
- 1,407,145 decoys
- <10μM affinity
- **true poses unknown**
- **trust** docked poses

## Affinity Prediction

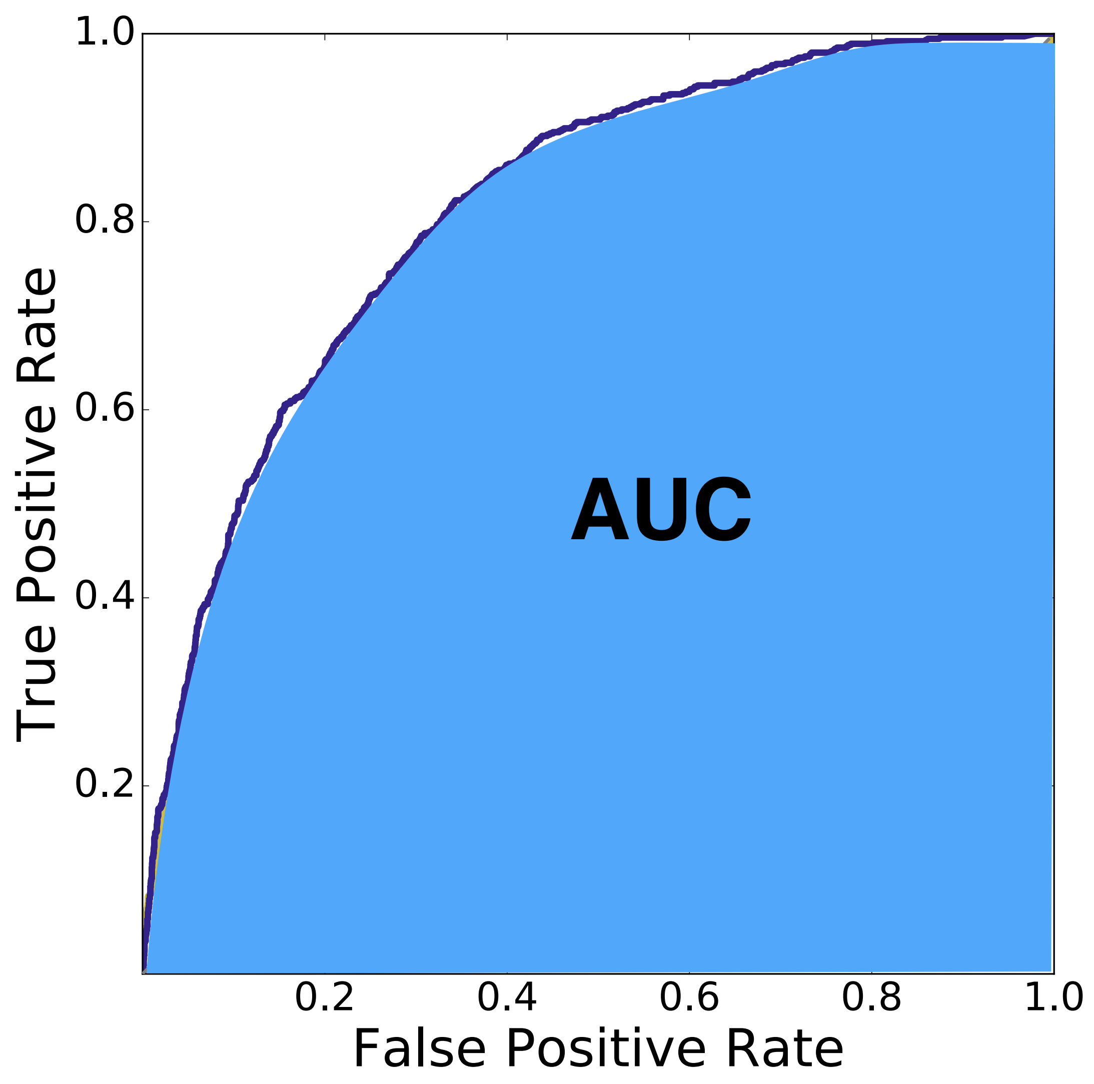
PDBbind

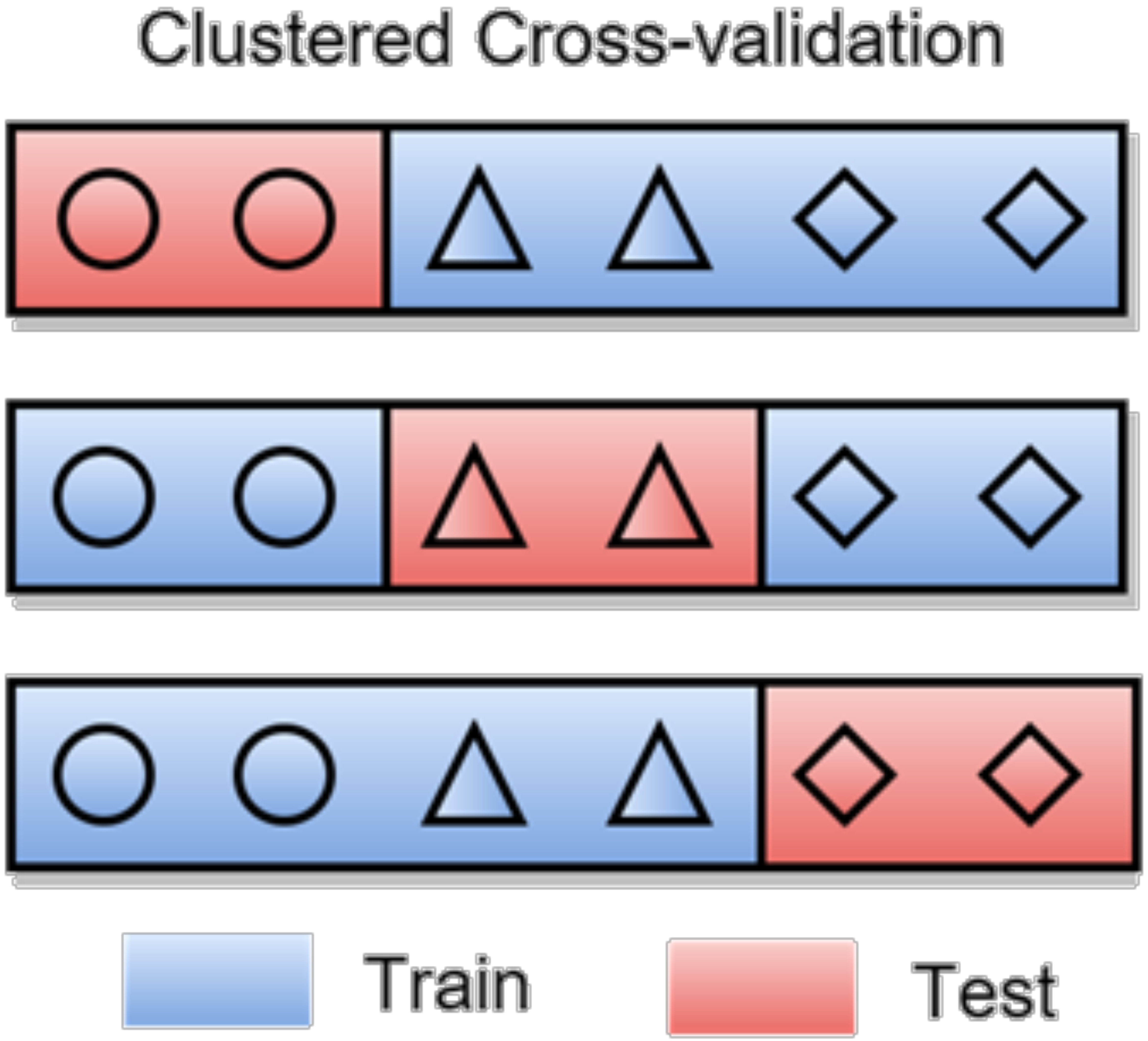
- 8,688 low RMSD poses
- assign known affinity
- **regression problem**

# Model Evaluation

**CSAR**: >90% similar targets kept in same fold

**DUD-E & PDBbind**: >80% similar targets kept in same fold

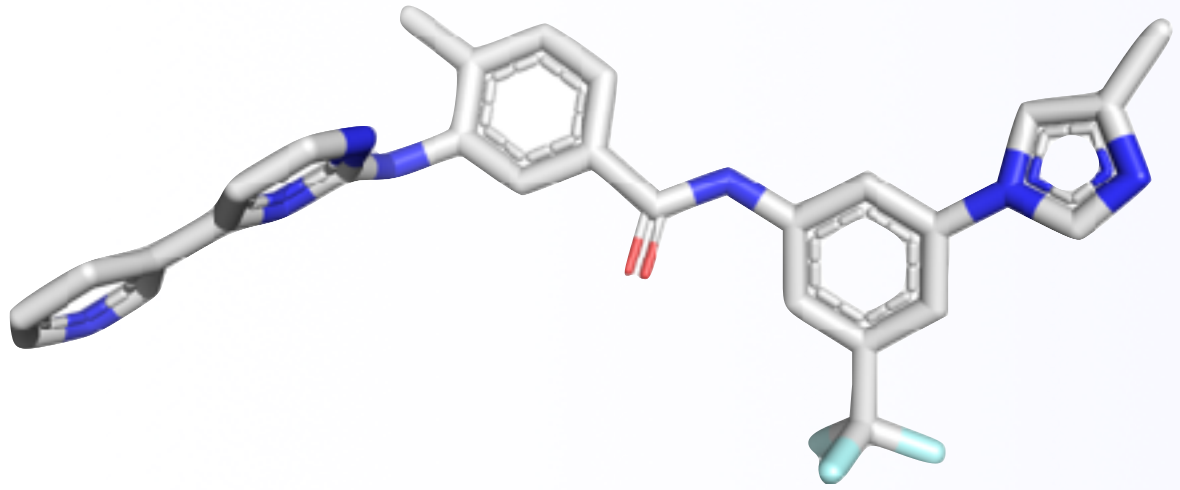


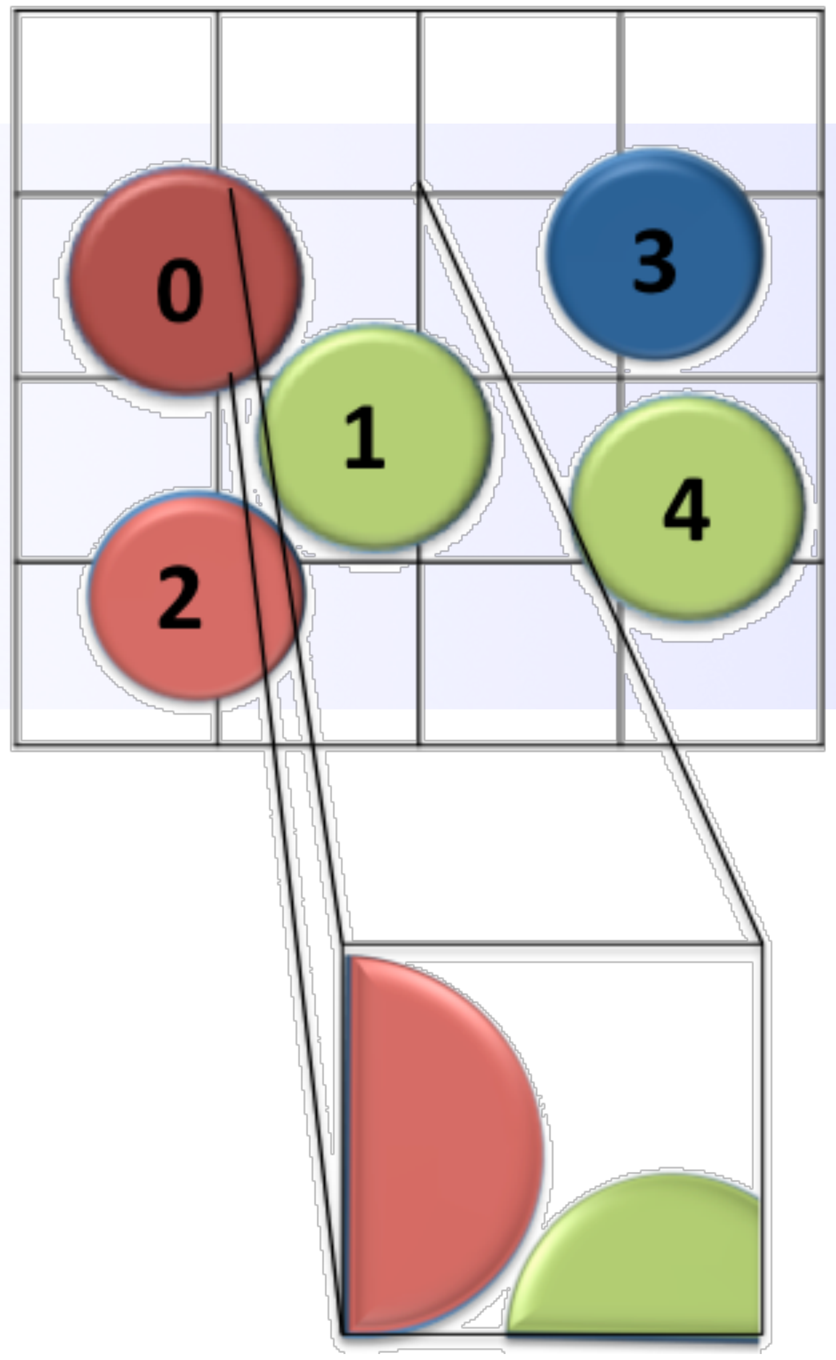



# Model Training

## Custom **`MolGridDataLayer`**

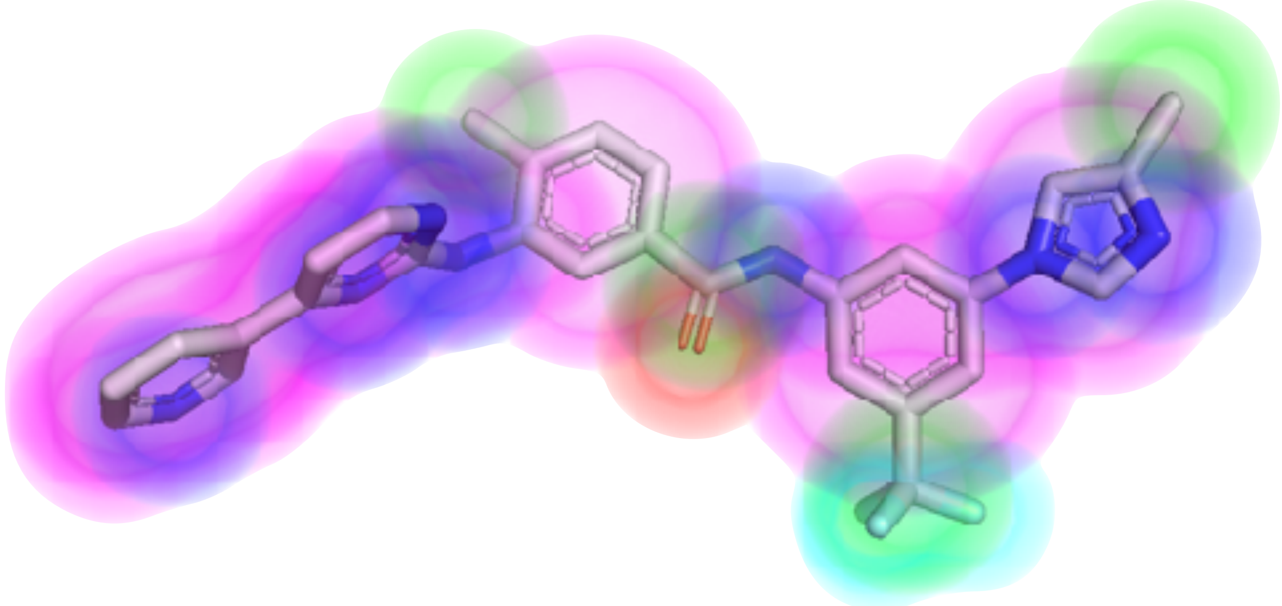
Parallelize over *atoms* to obtain a mask of atoms that overlap each grid region
Use exclusive scan to obtain a list of atom indices from the mask
Parallelize over *grid points*, using reduced atom list to avoid $O(N_{atoms})$ check





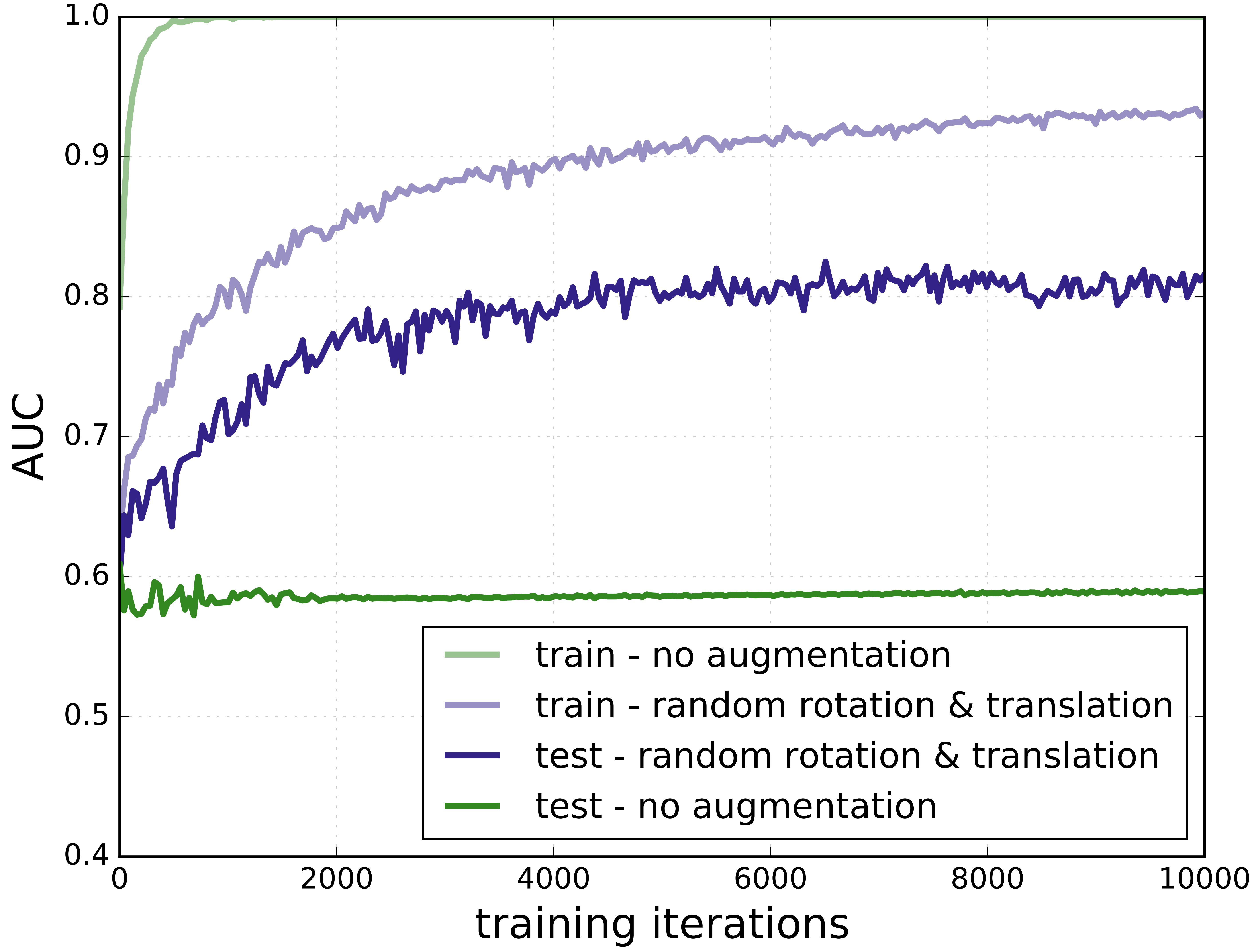

For example, consider subgrid 5:

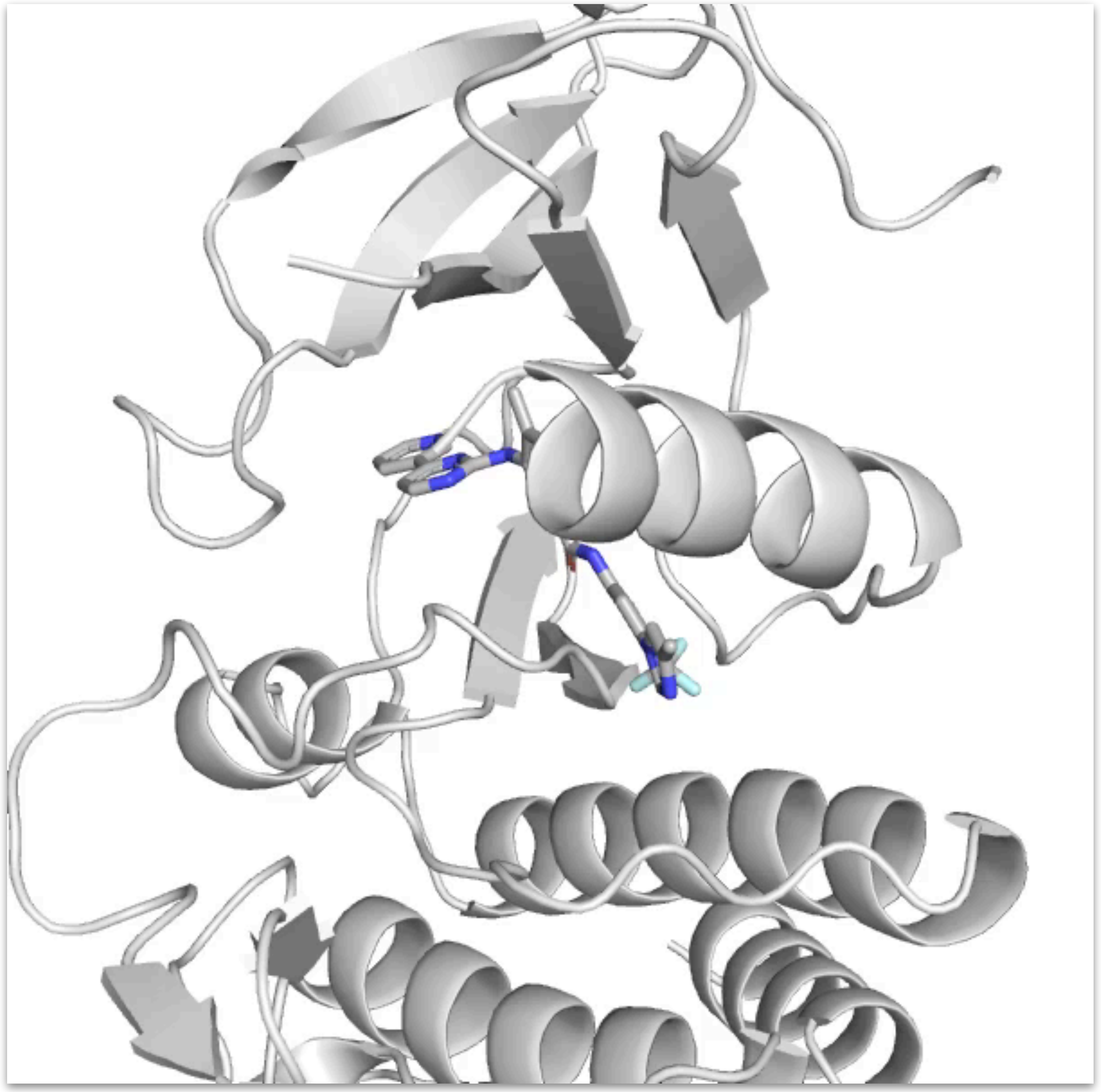
| | | | | | |
|---|---|---|---|---|---|
| Atom mask: | 1 | 1 | 0 | 0 | 0 |
| Exclusive scan: | 0 | 1 | 2 | 2 | 2 |
| Final indices: | 0 | 1 | | | |



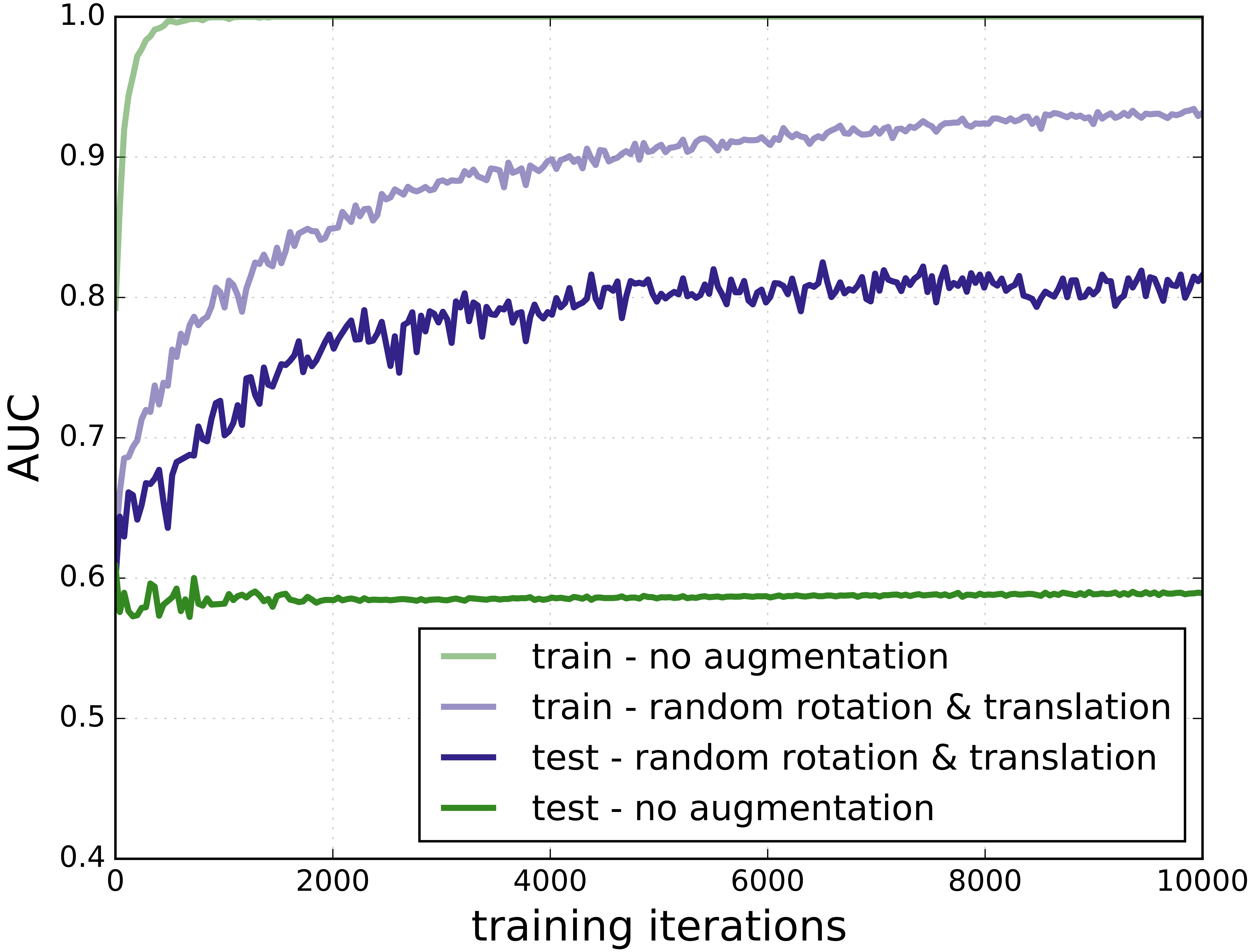
NVIDIA

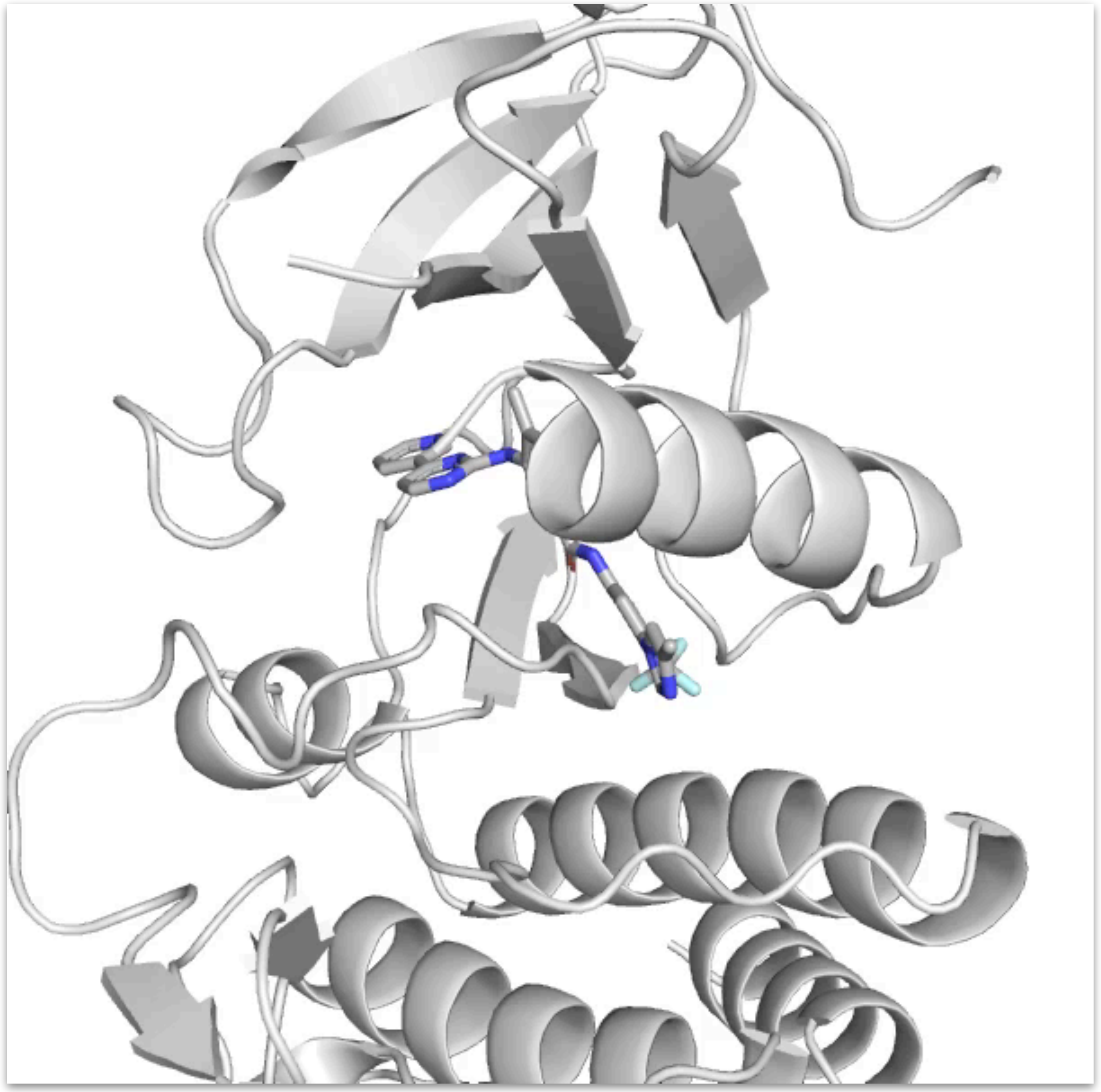
# Data Augmentation

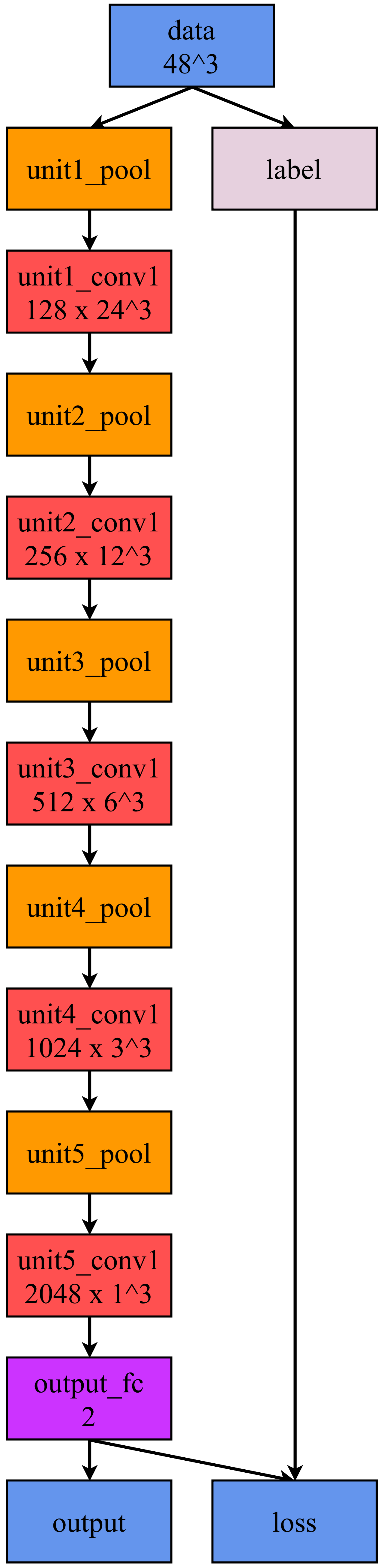


# Data Augmentation






# Model Optimization

Atom Types

- Vina (34)
- element-only (18)
- ligand-protein (2)

Atom Density Type

- Boolean
- Gaussian

Radius Multiple

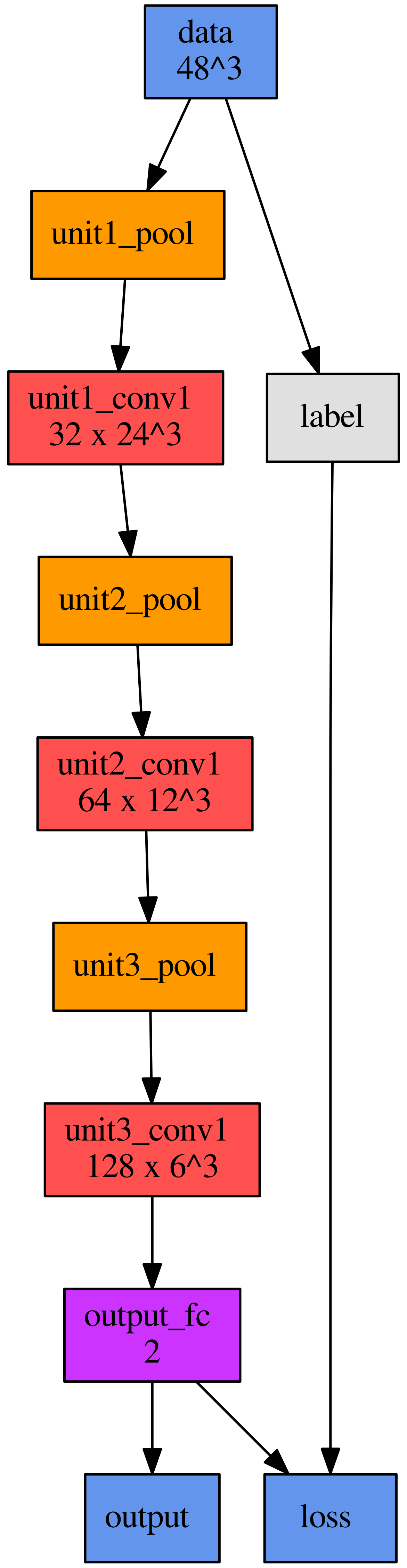
Resolution

Pooling

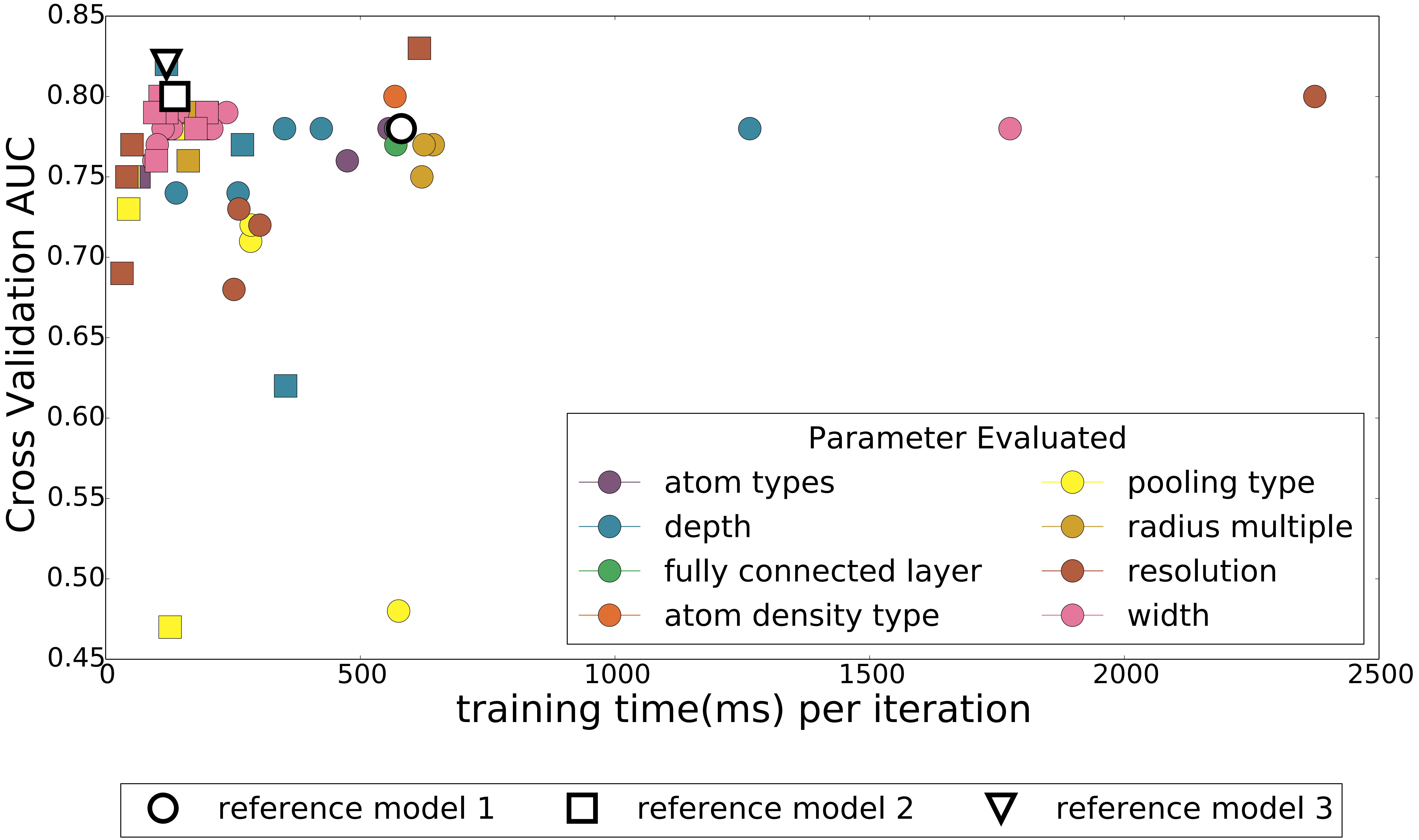
Depth

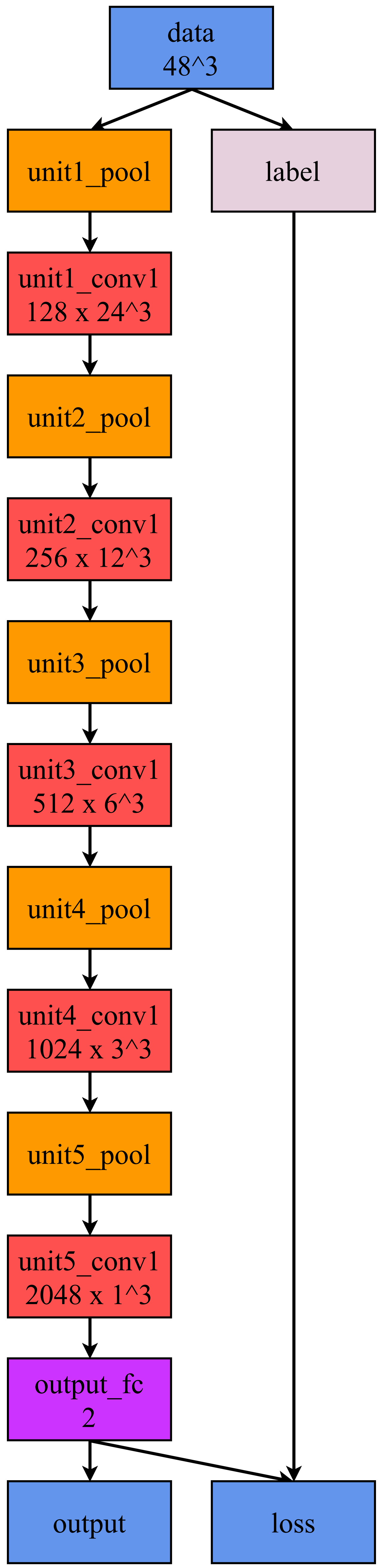
Width

Fully Connected Layers

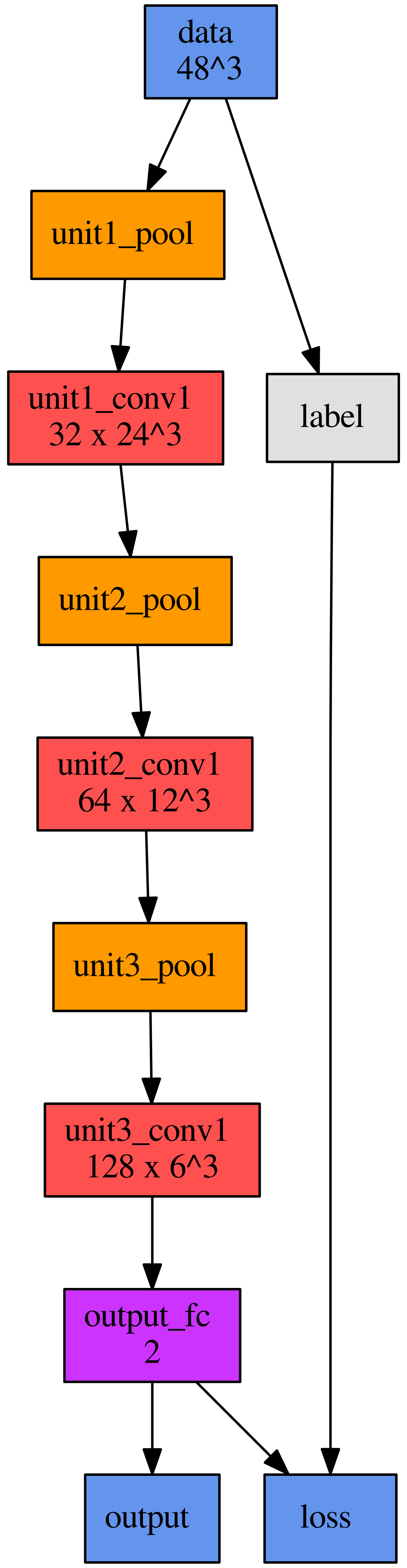


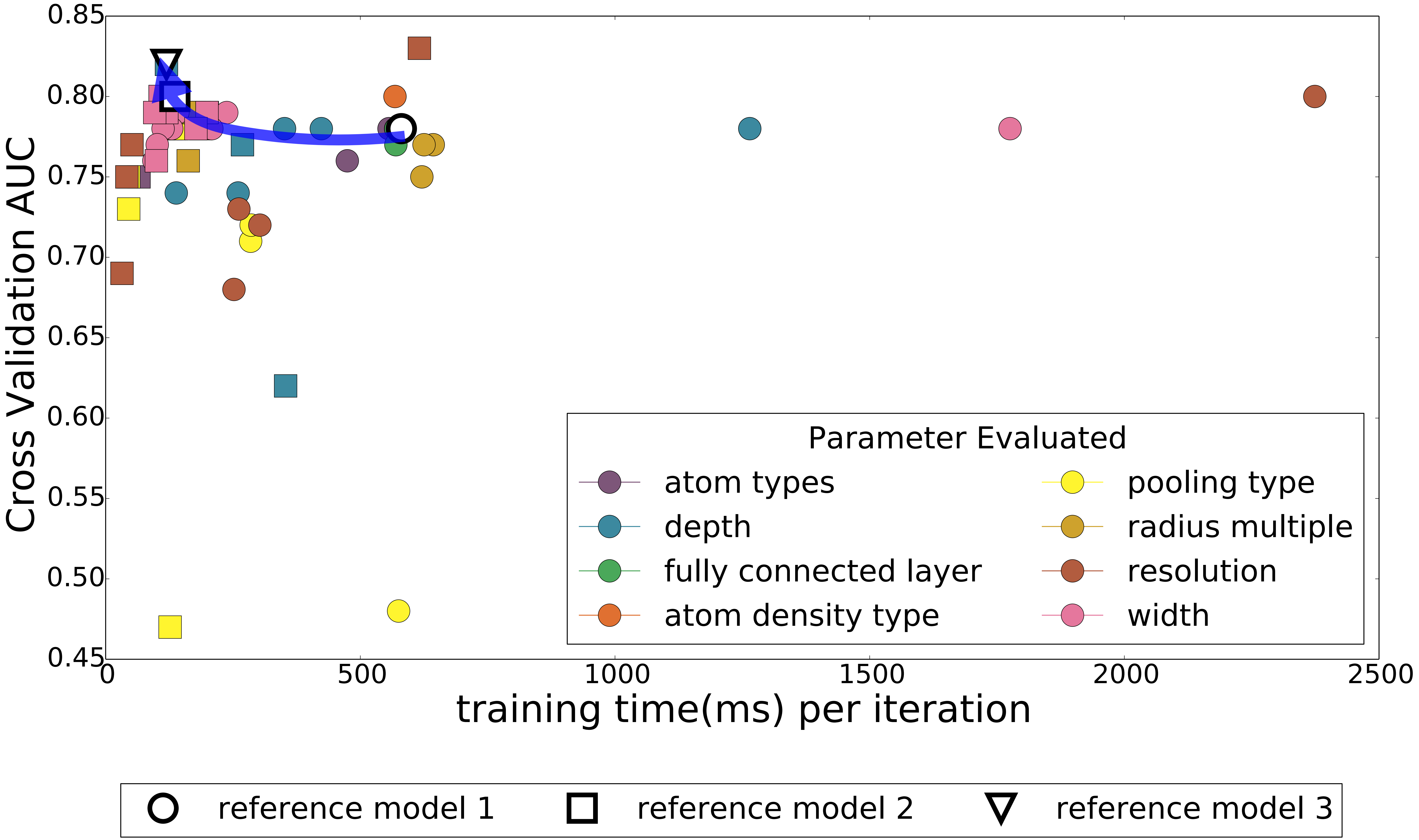
# Model Optimization

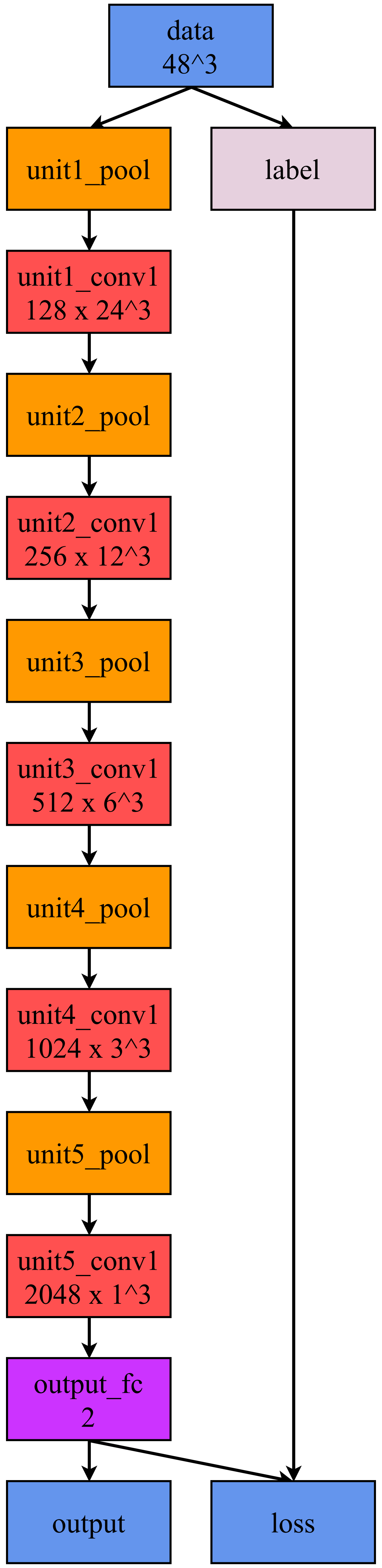


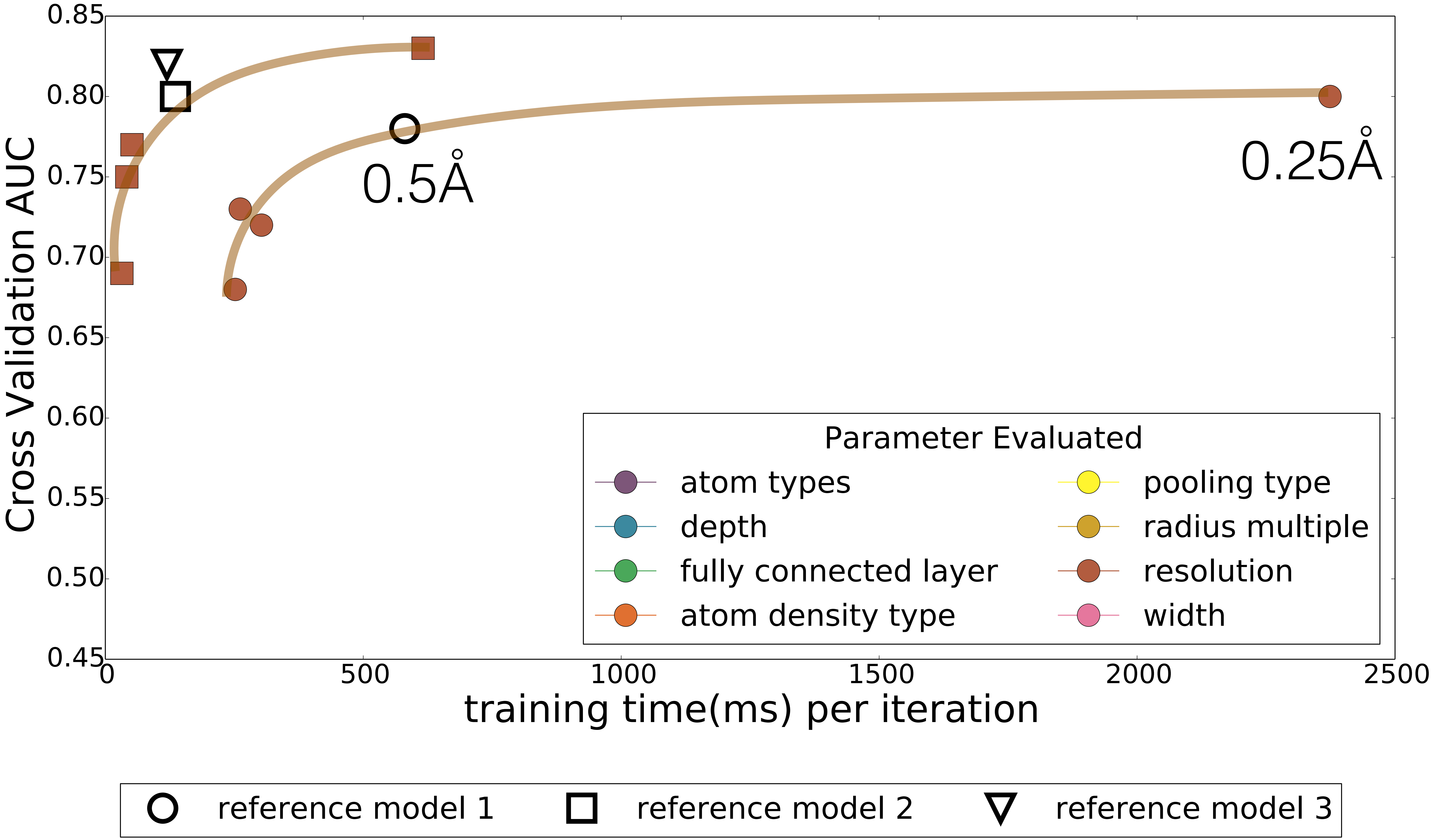
# Model Optimization



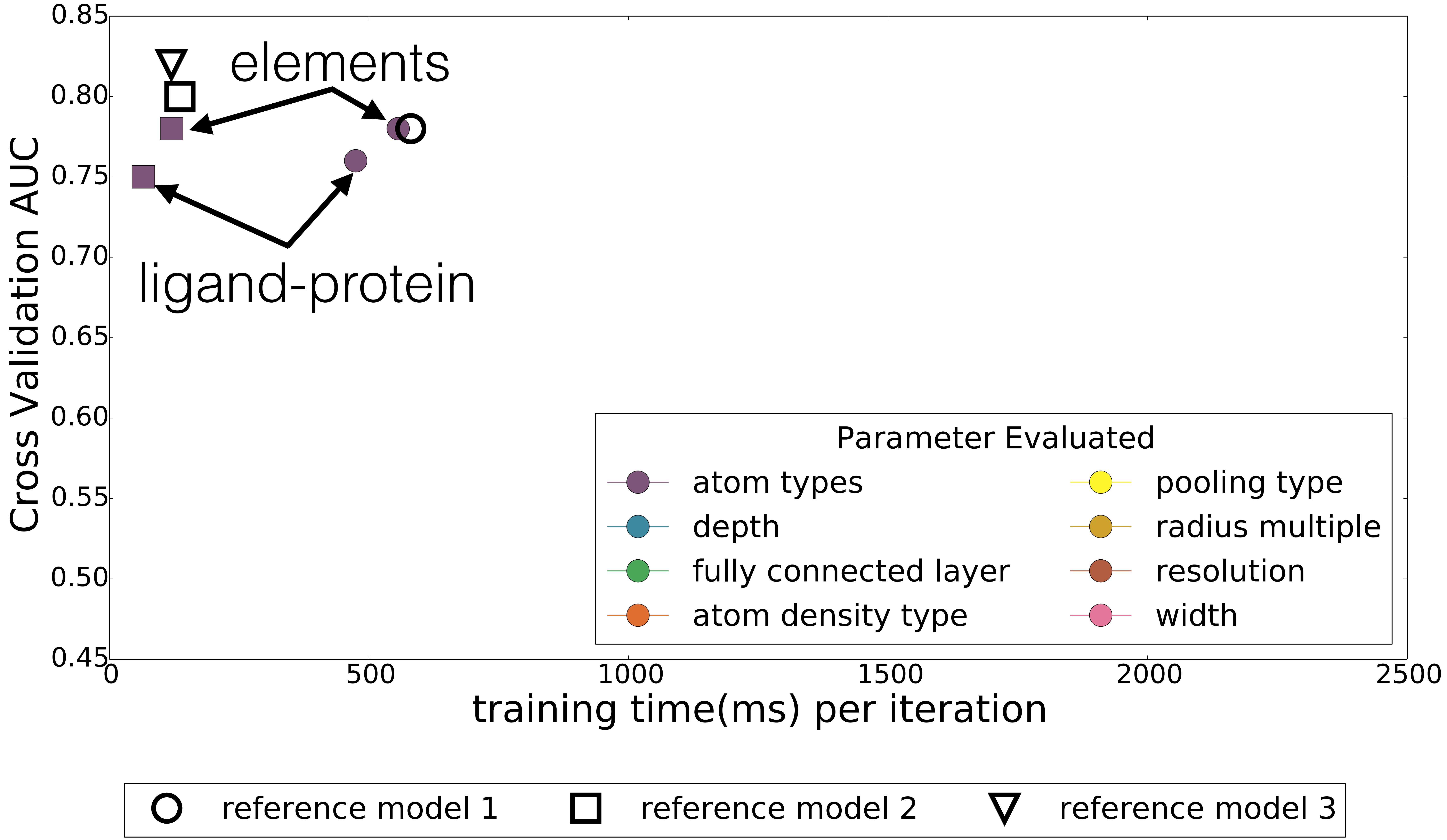



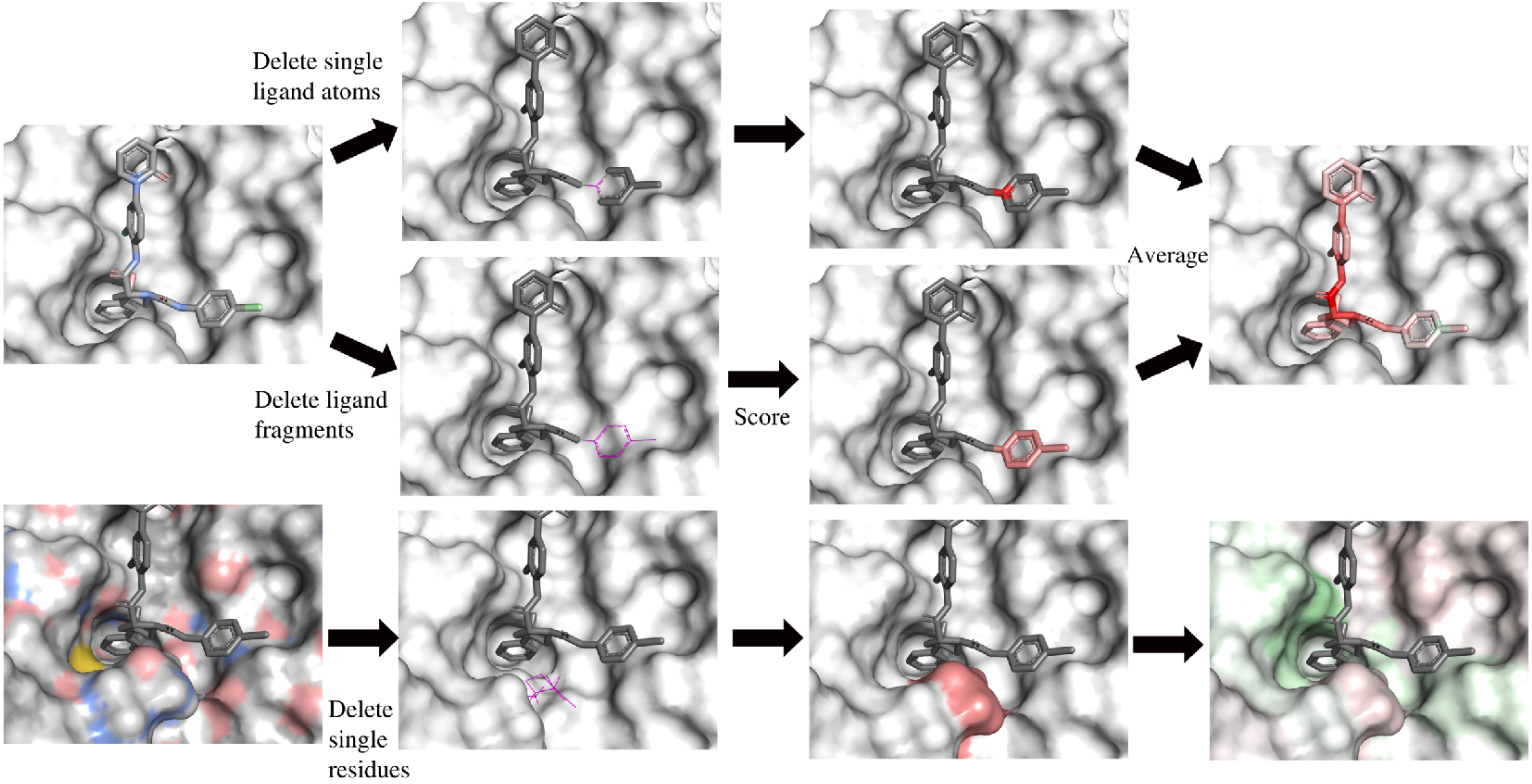
# Grid Resolution



# Atom Types


elements
ligand-protein
Cross Validation AUC
0.85
0.80
0.75
0.70
0.65
0.60
0.55
0.50
0.45
0
500
1000
1500
2000
2500
training time(ms) per iteration
Parameter Evaluated
atom types
depth
fully connected layer
atom density type
pooling type
radius multiple
resolution
width
reference model 1
reference model 2
reference model 3

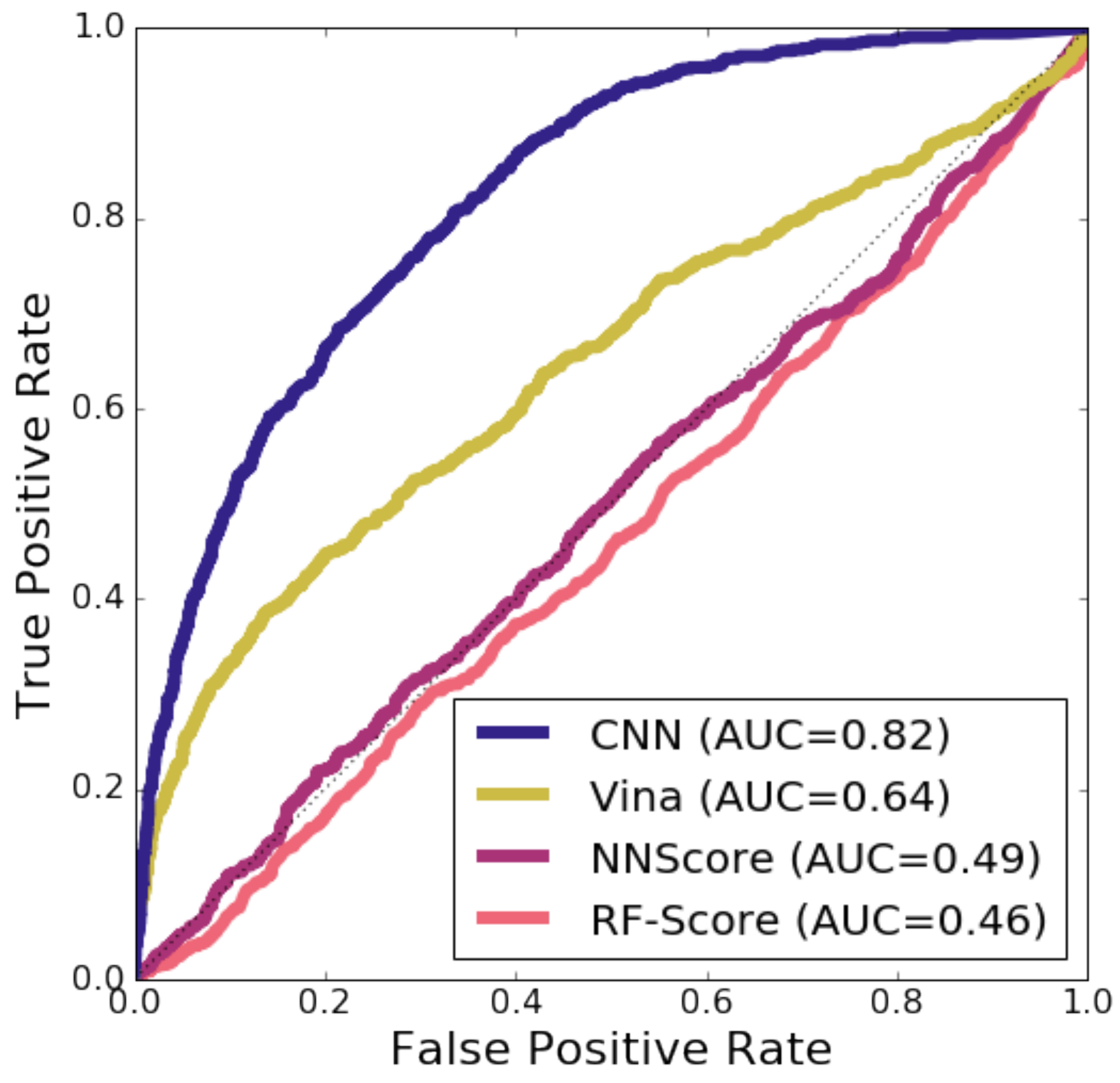
# Visualization



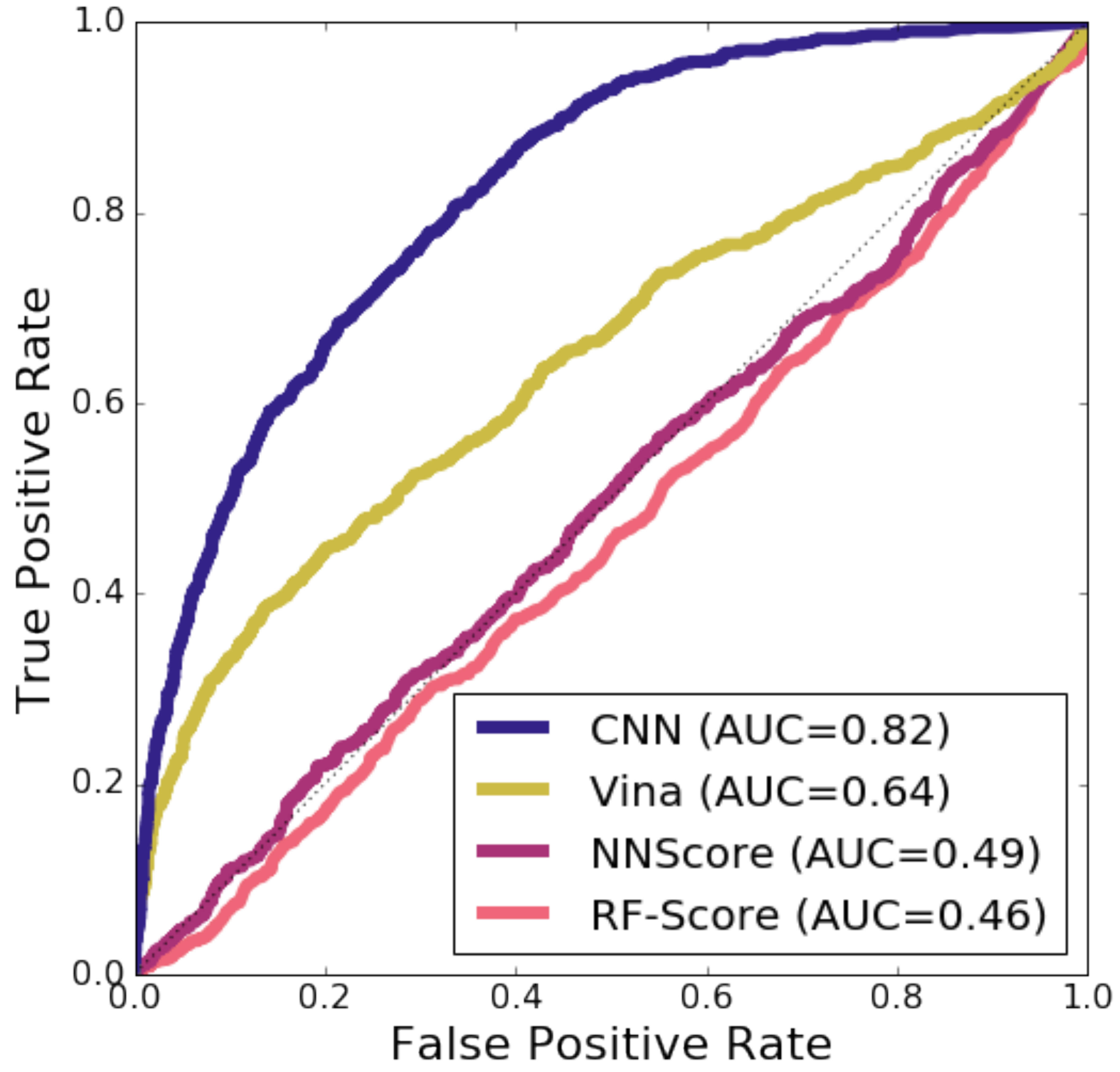
# Visualizing Enzymes

# Cross-Validation Evaluation

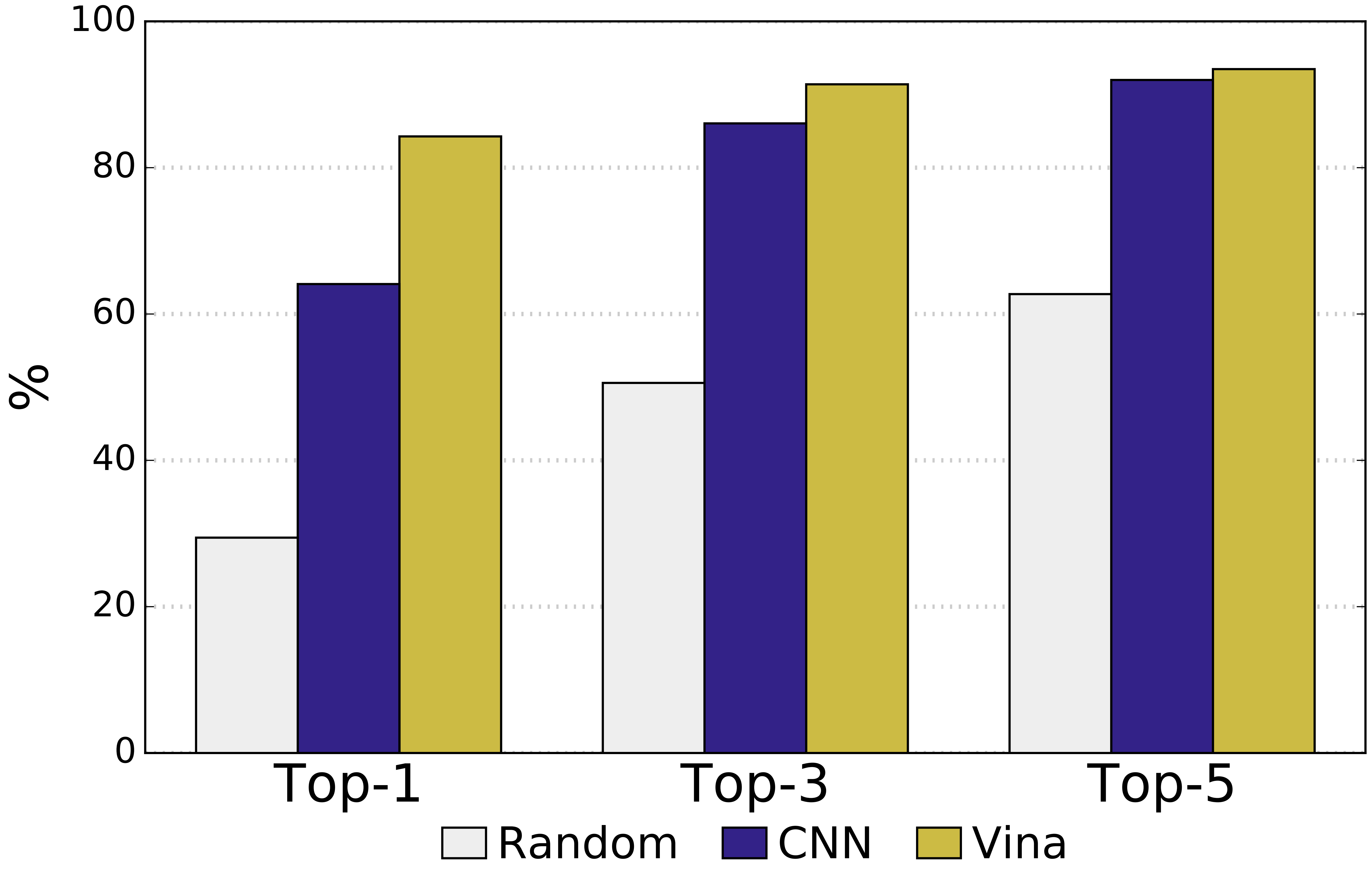
# Pose Prediction (CSAR)



# Pose Prediction (CSAR)

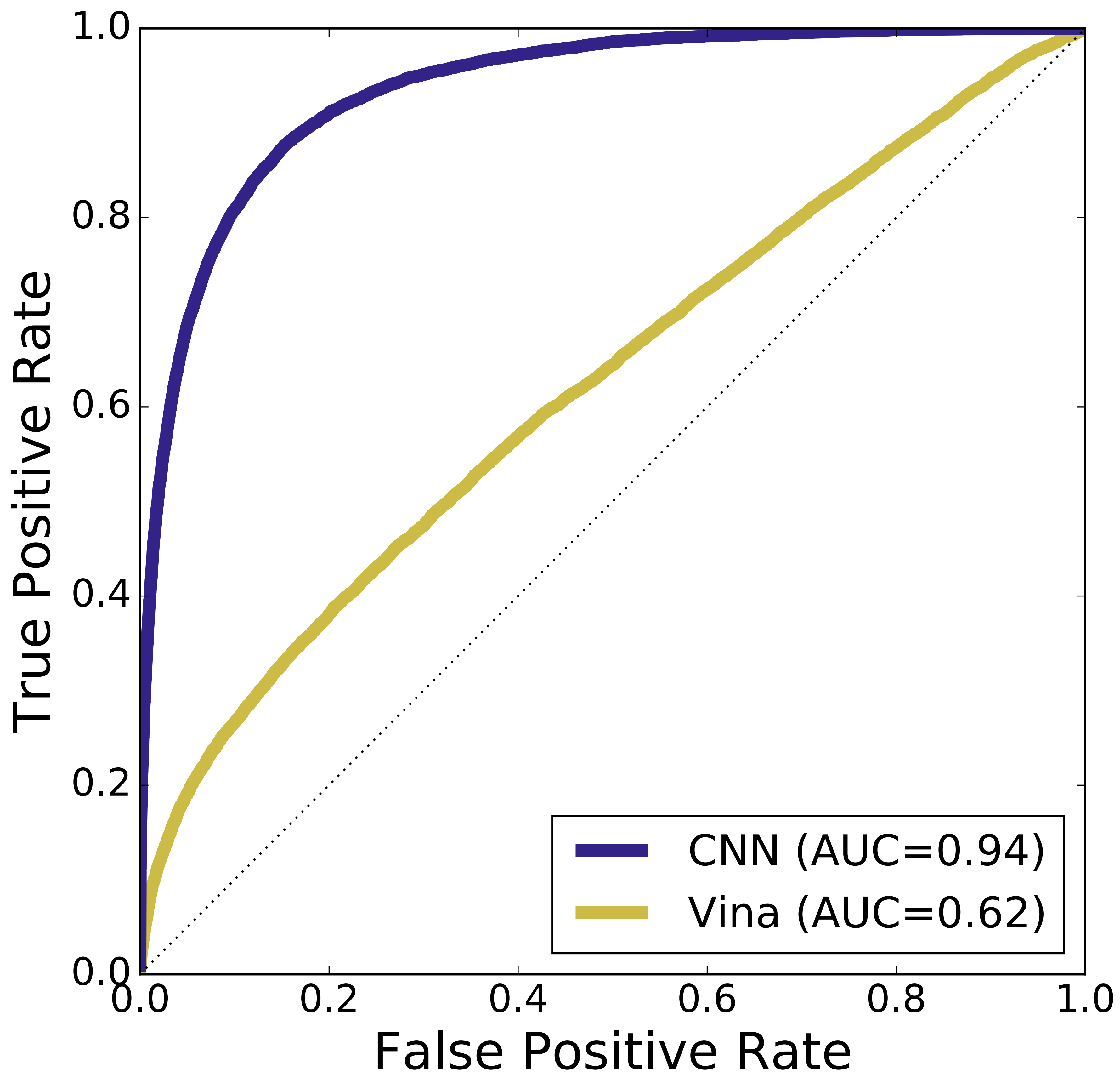



*inter*-target ranking

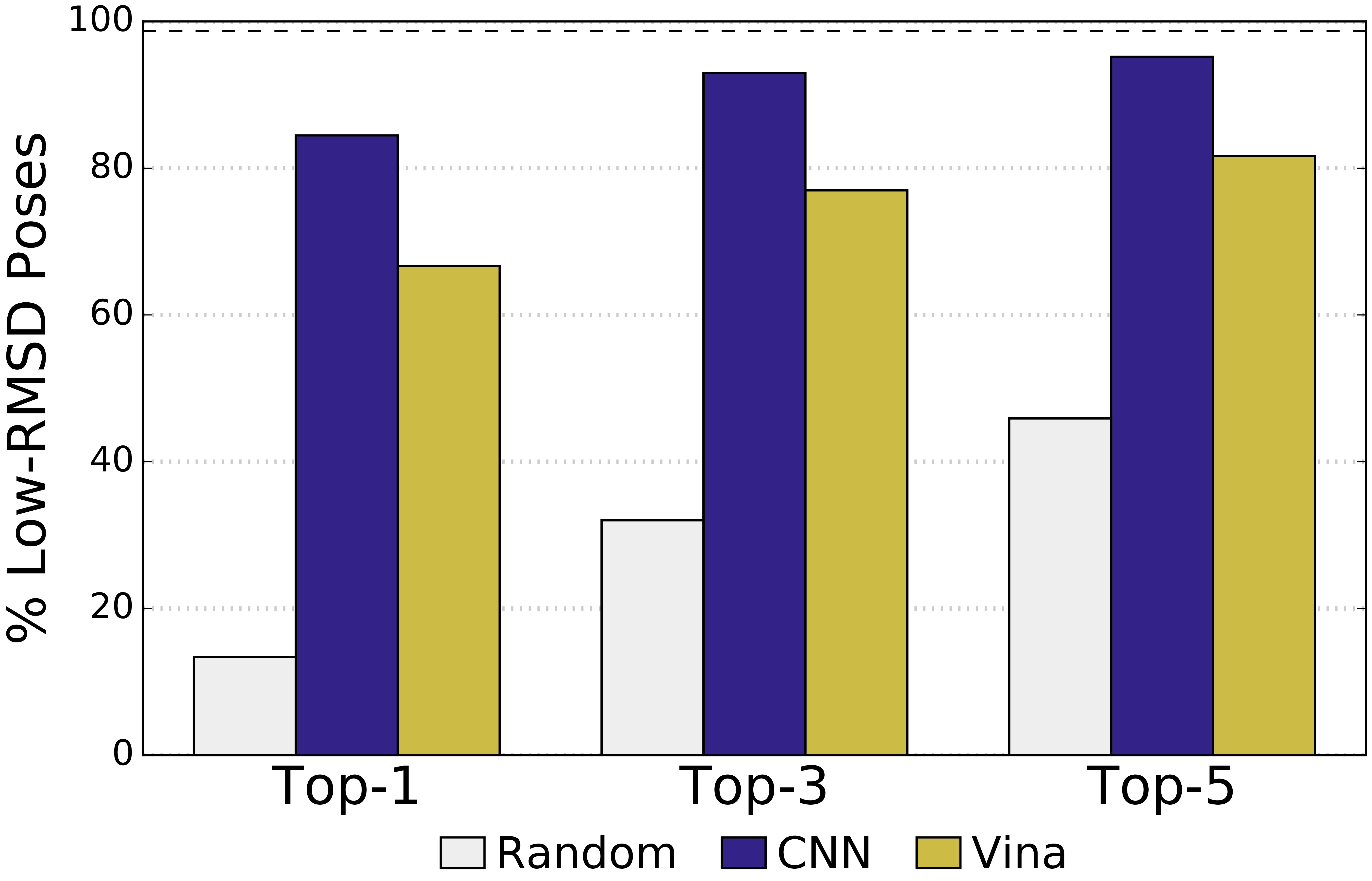



*intra*-target ranking

# Pose Prediction (PDBbind)

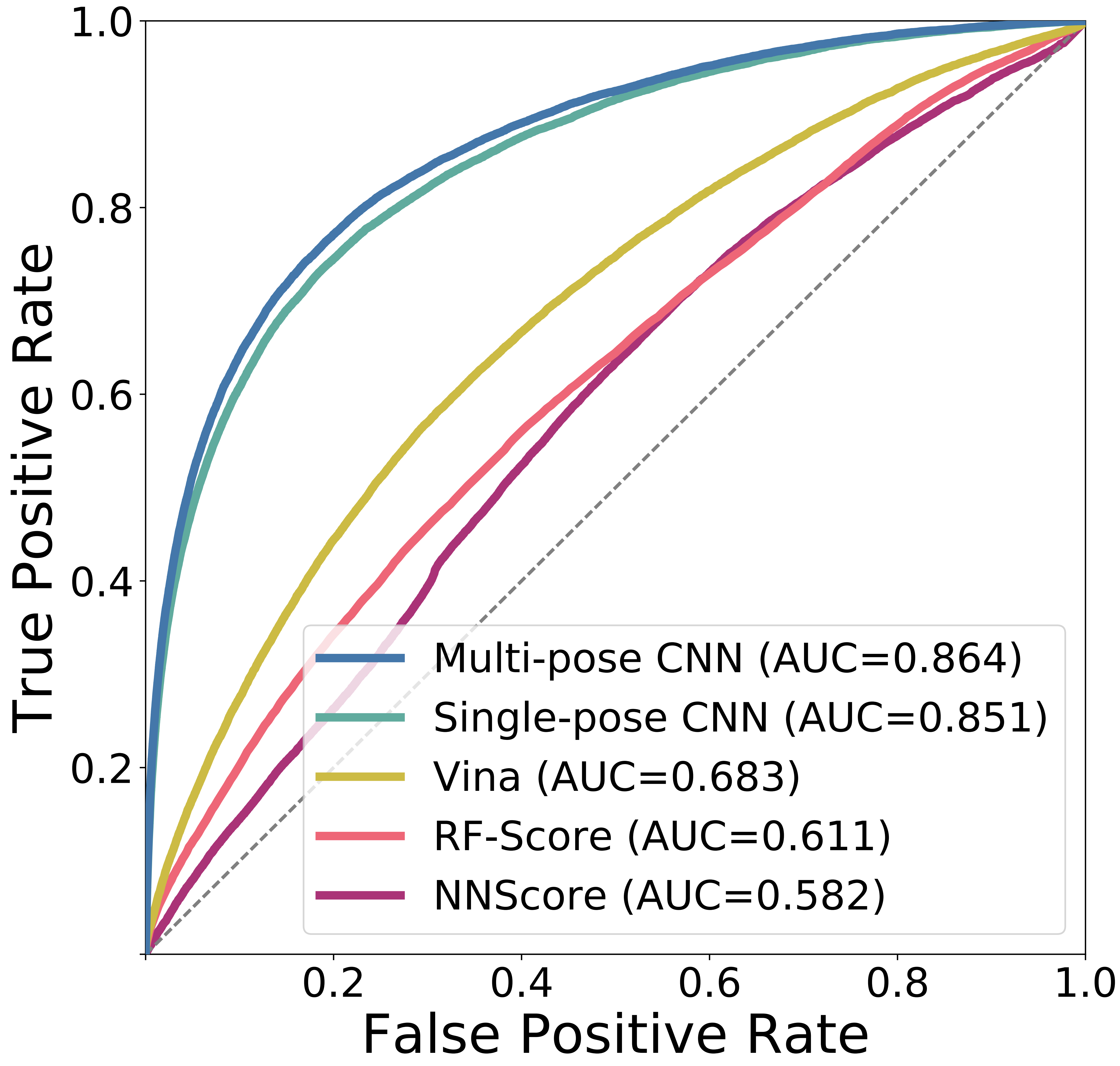



*inter*-target ranking




*intra*-target ranking

# Binding Determination

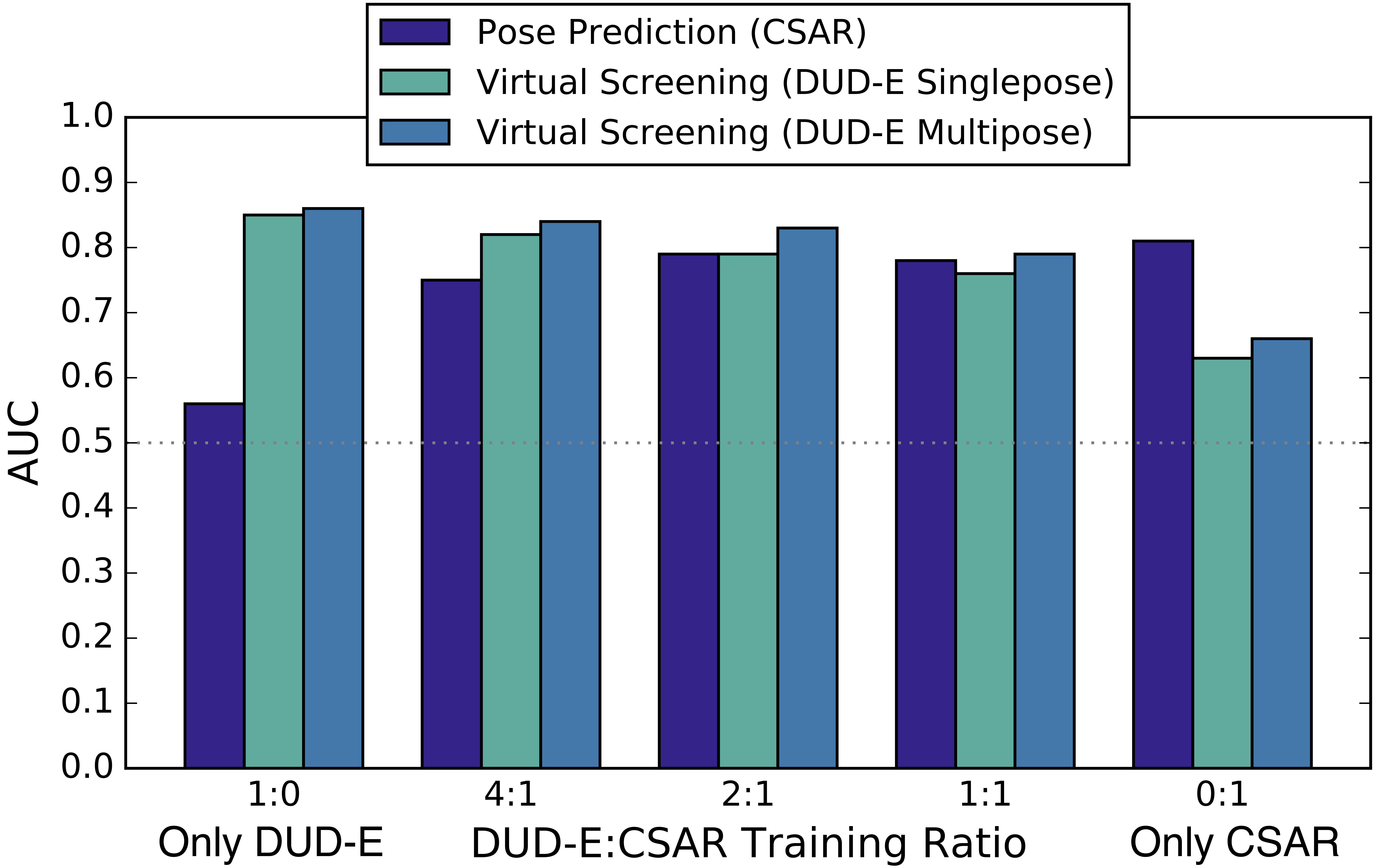



DUD•E

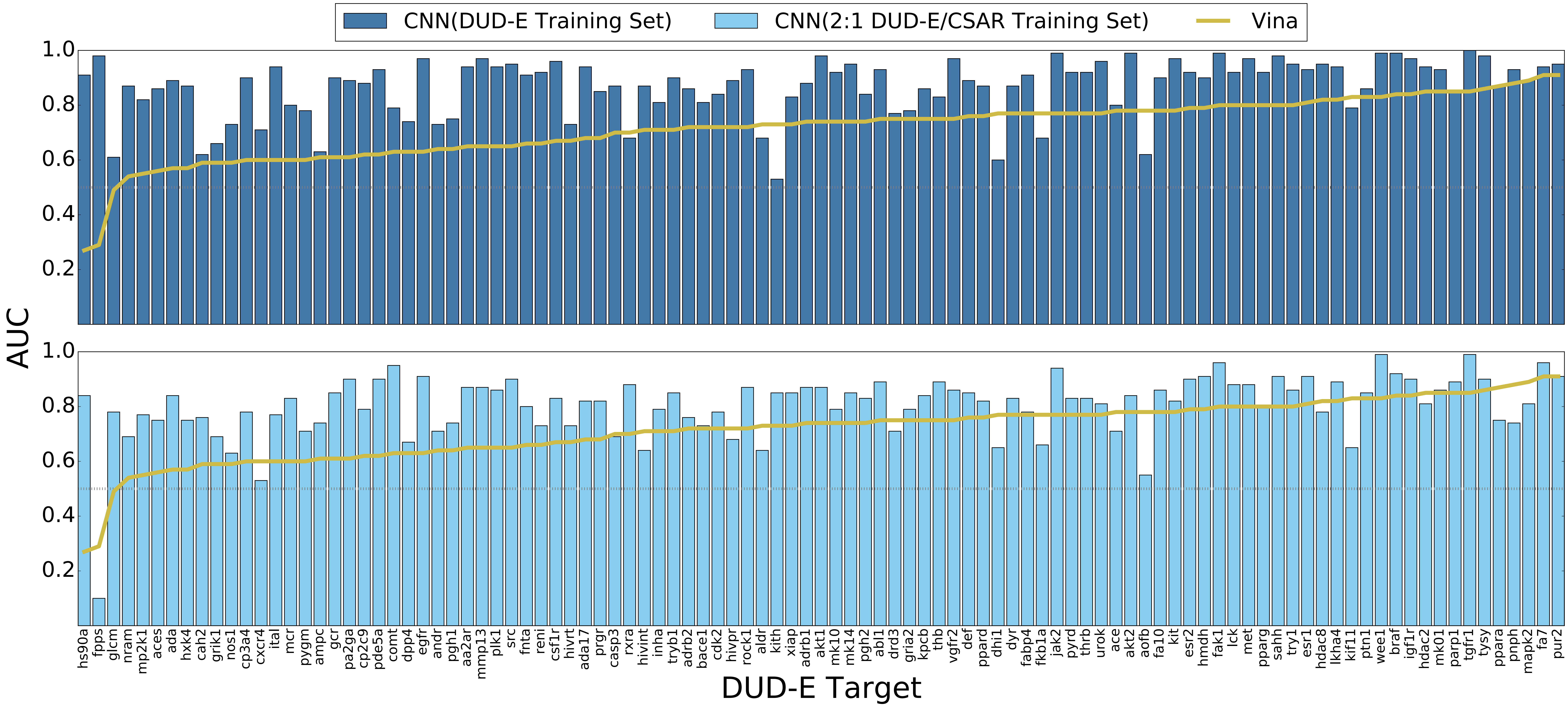
102 targets

- 22,645 actives
- 1,407,145 decoys
- <10μM affinity
- **true poses unknown**
- use top docked pose

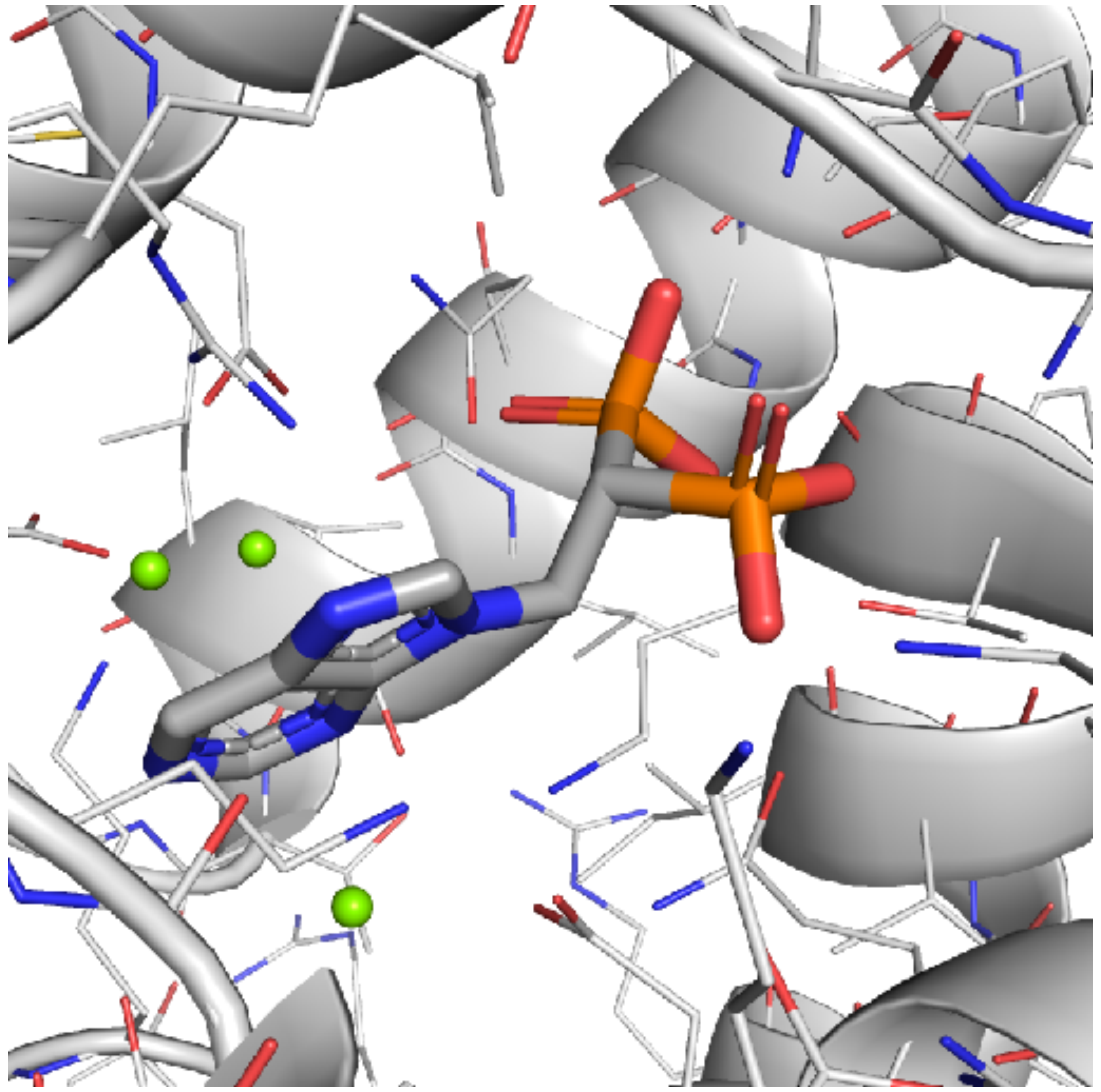
# Combined Training



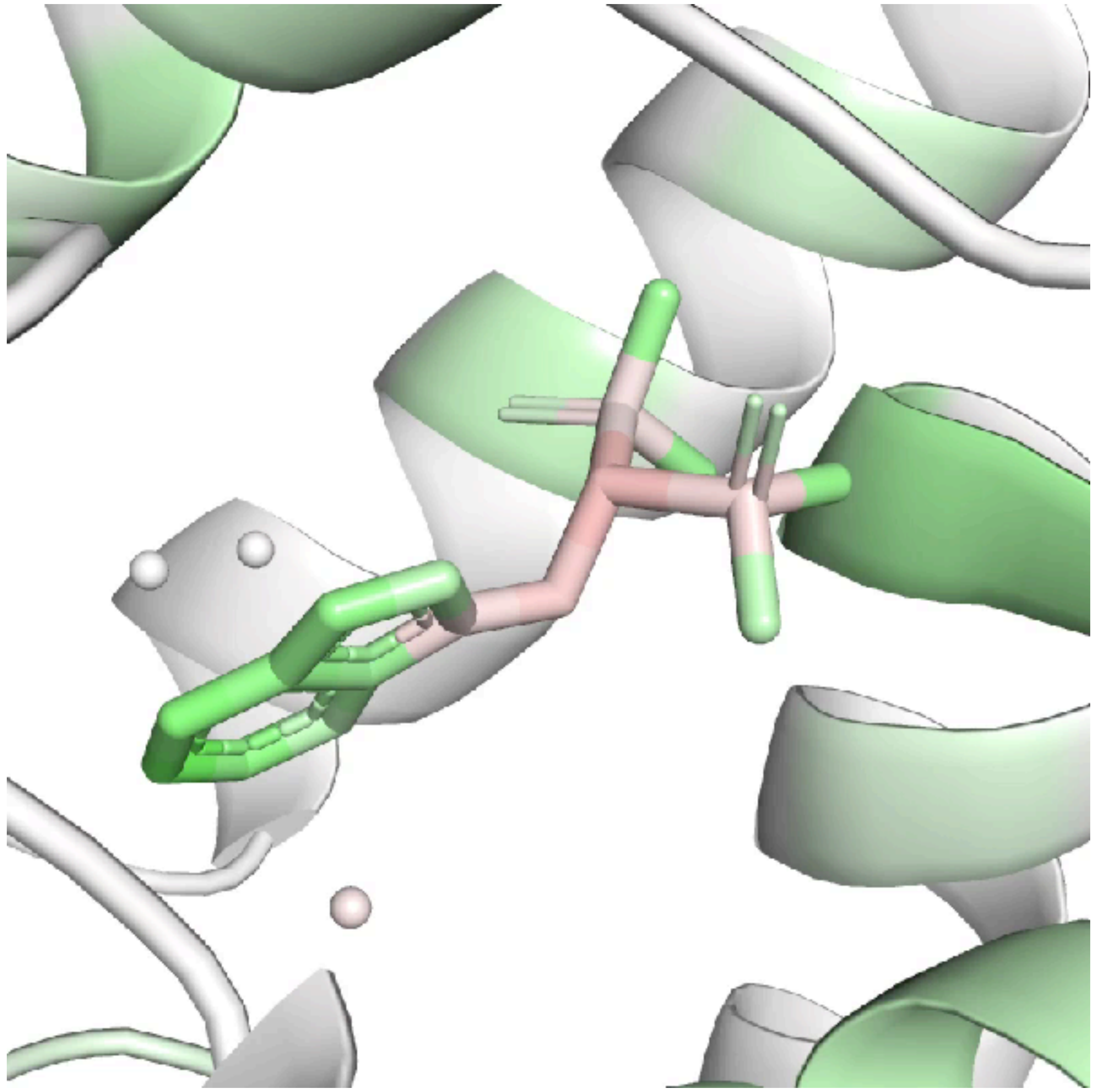
# Binding Determination



# fpps (farnesyl diphosphate synthase)

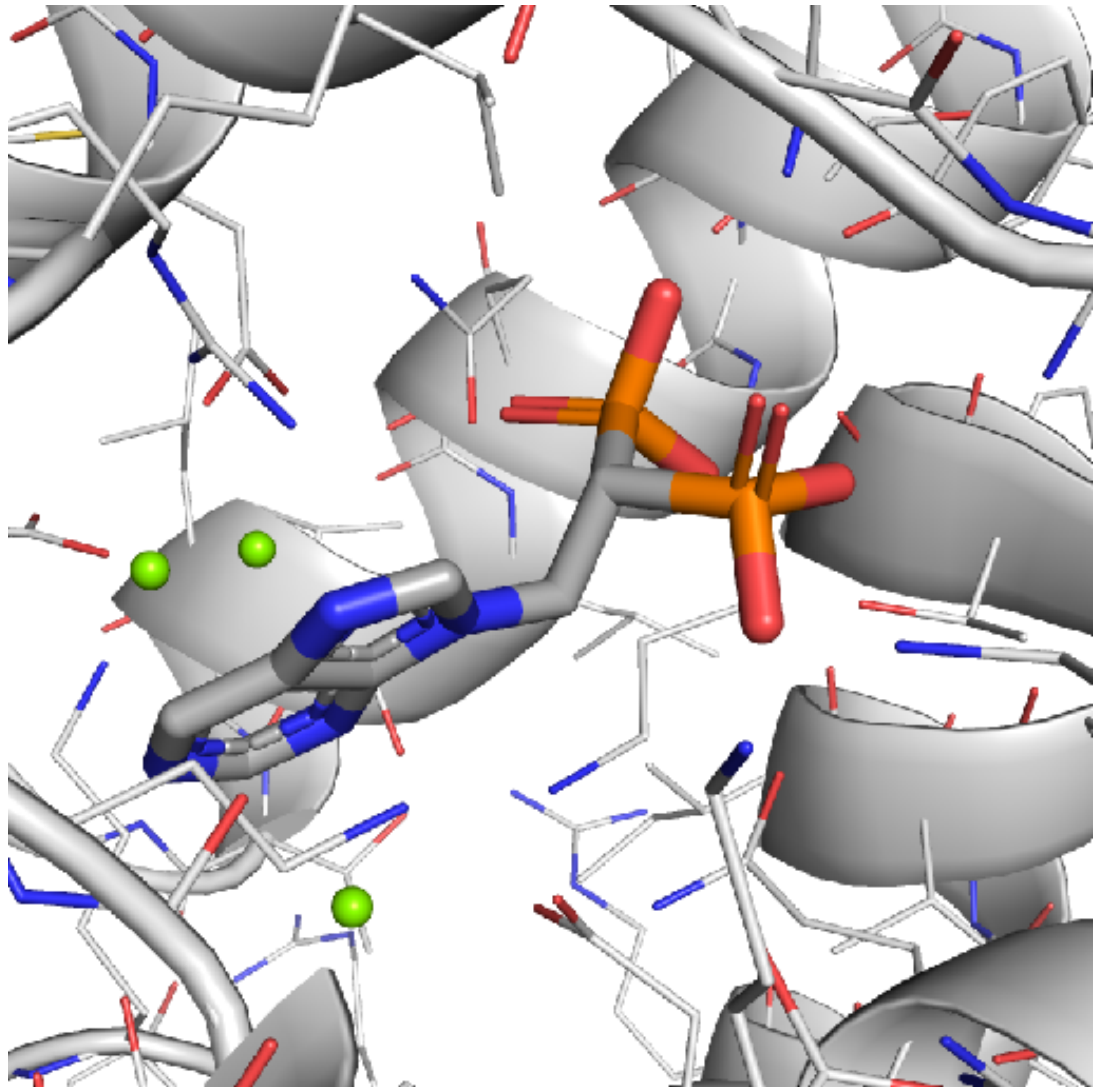


CHEMBL457424
Top Vina Pose (-8.2 kcal/mol)

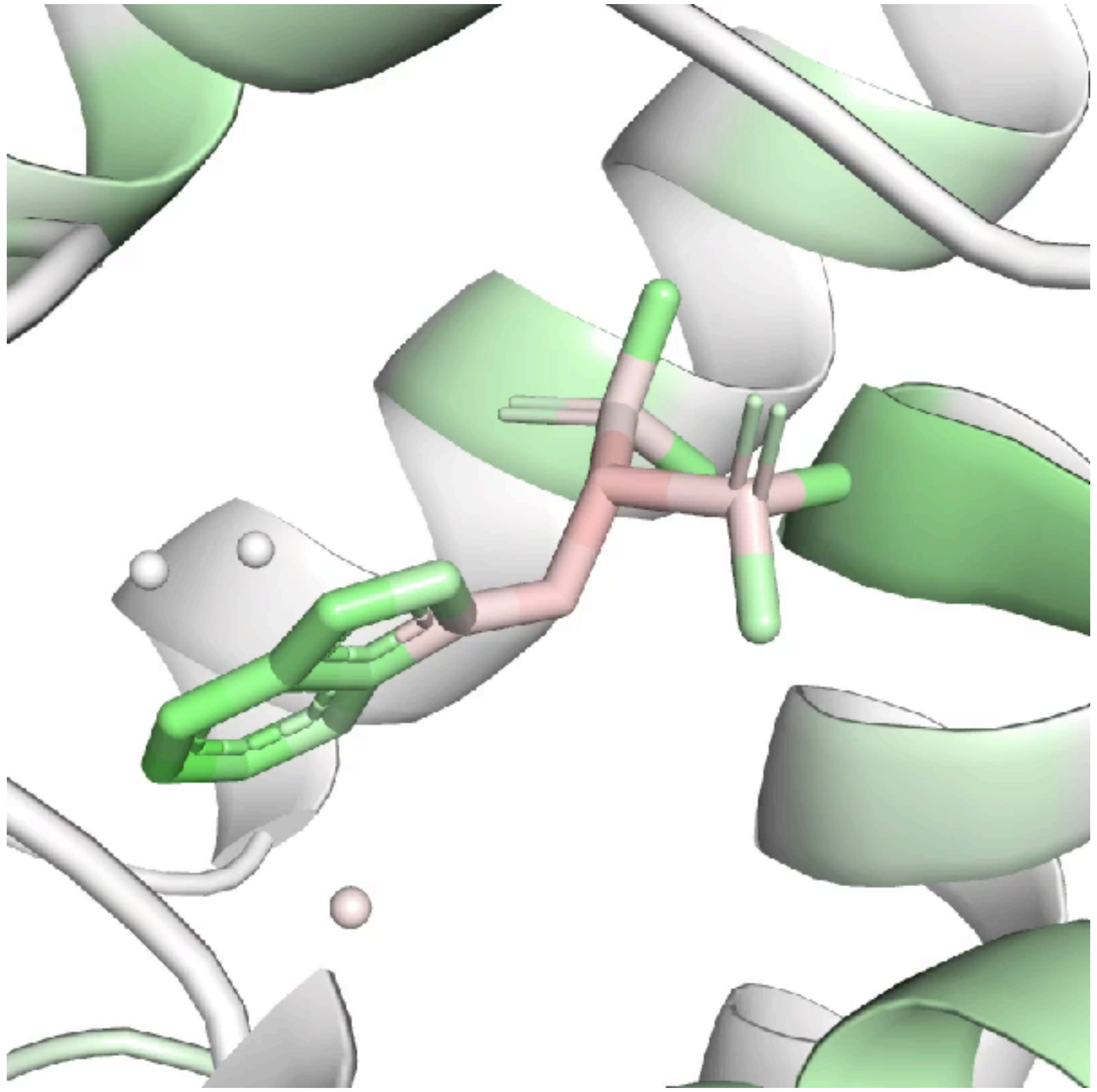


DUD-E Training Set
Score = 0.93 ± 0.03

# fpps (farnesyl diphosphate synthase)

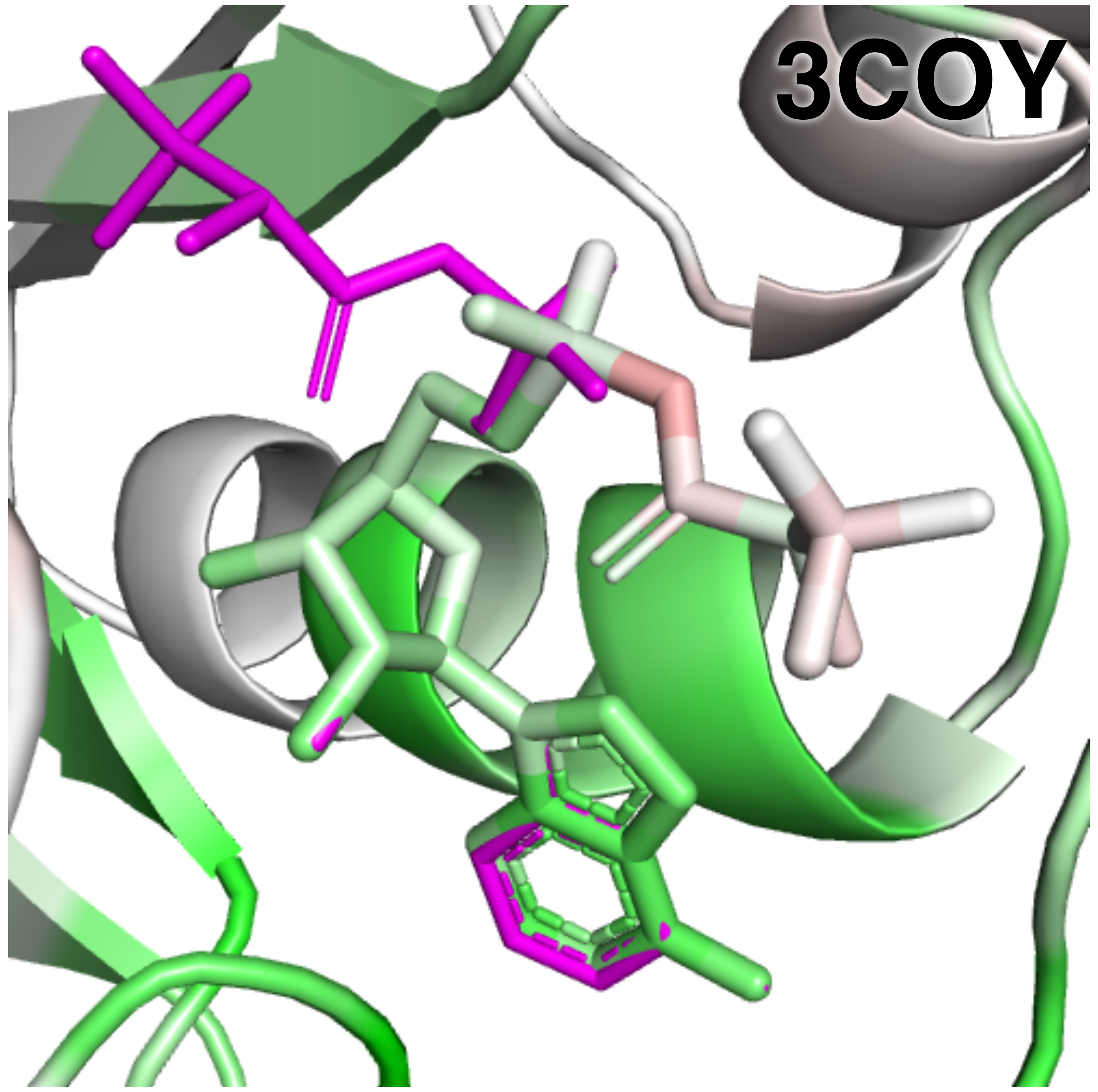


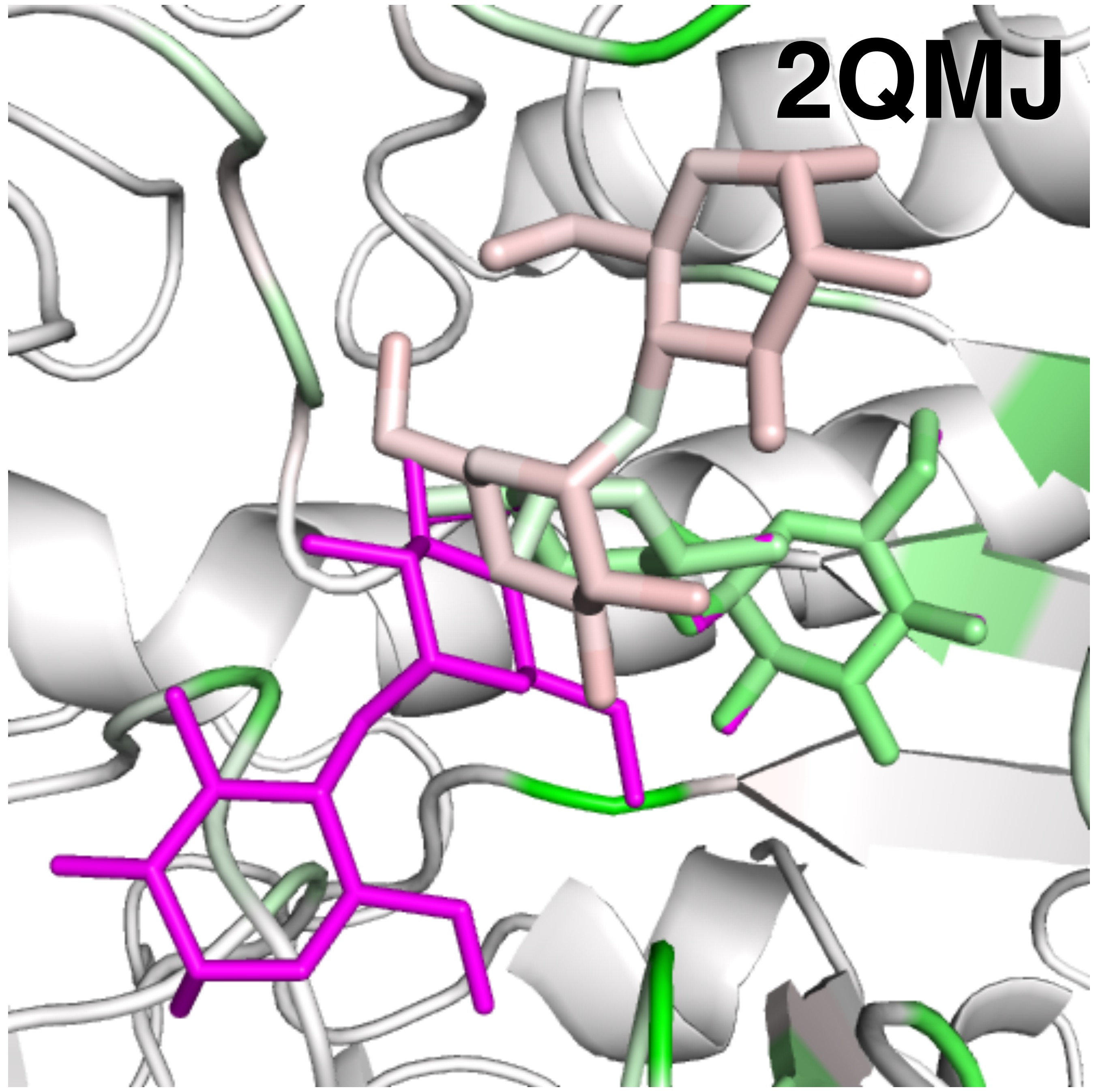
CHEMBL457424
Top Vina Pose (-8.2 kcal/mol)

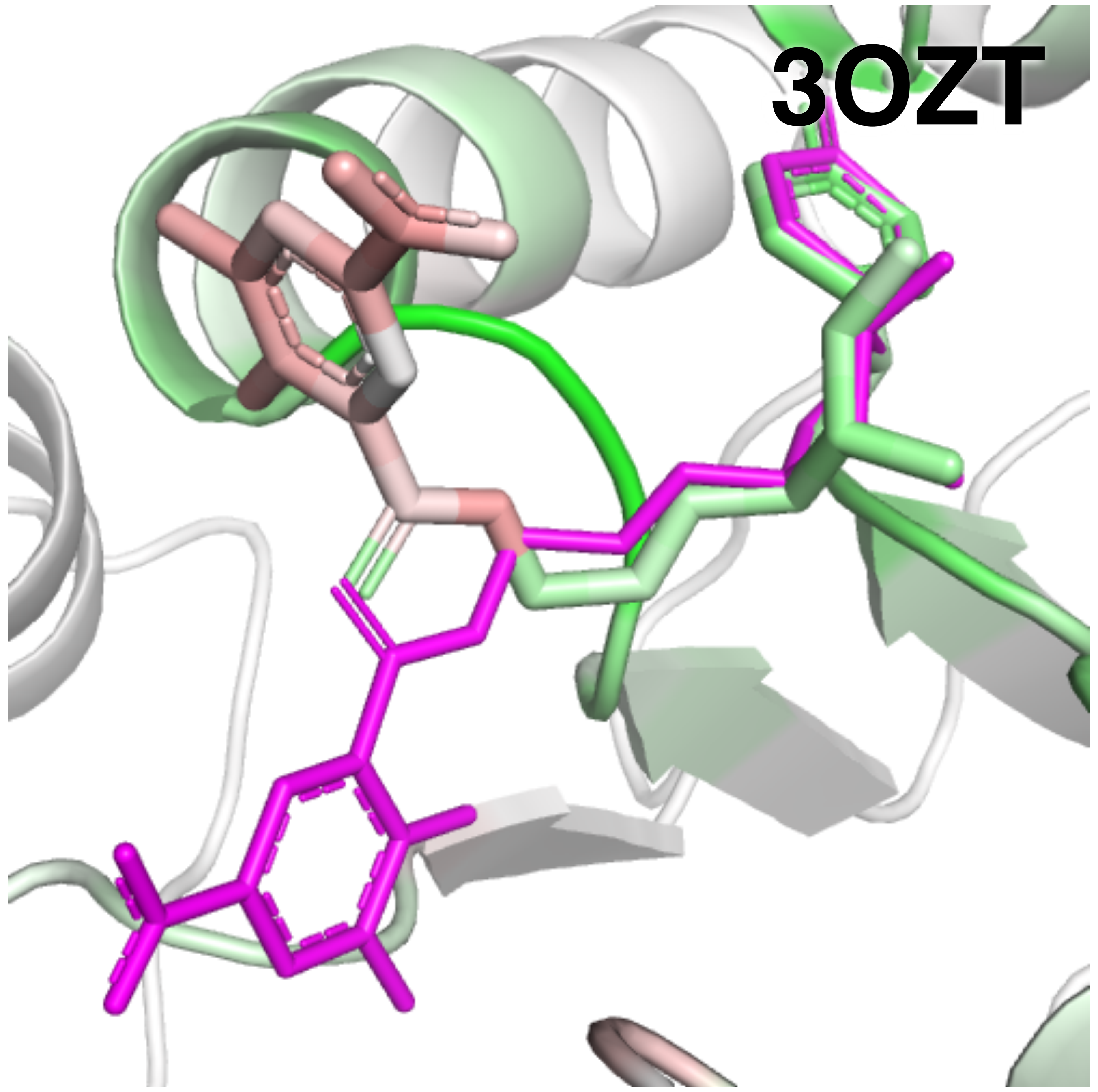


DUD-E Training Set
Score = 0.93 ± 0.03

# Pose Sensitivity

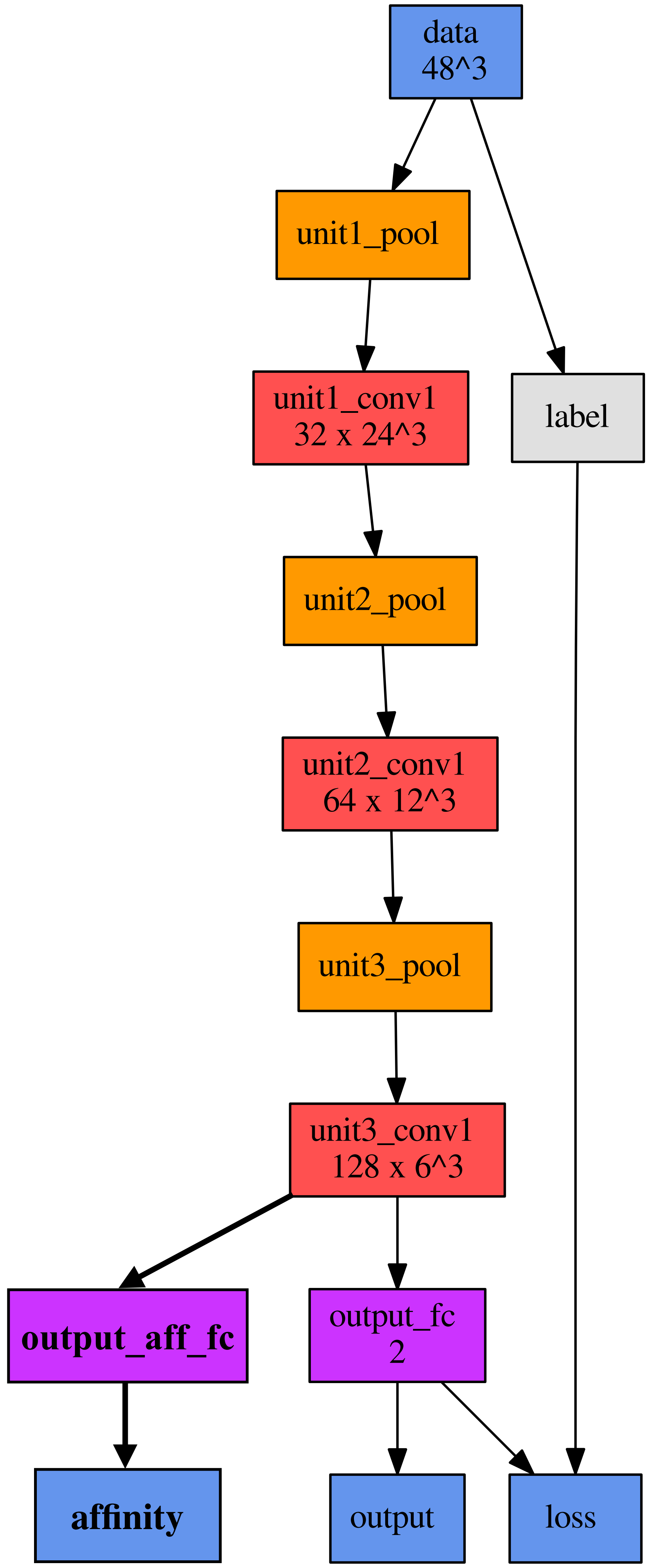


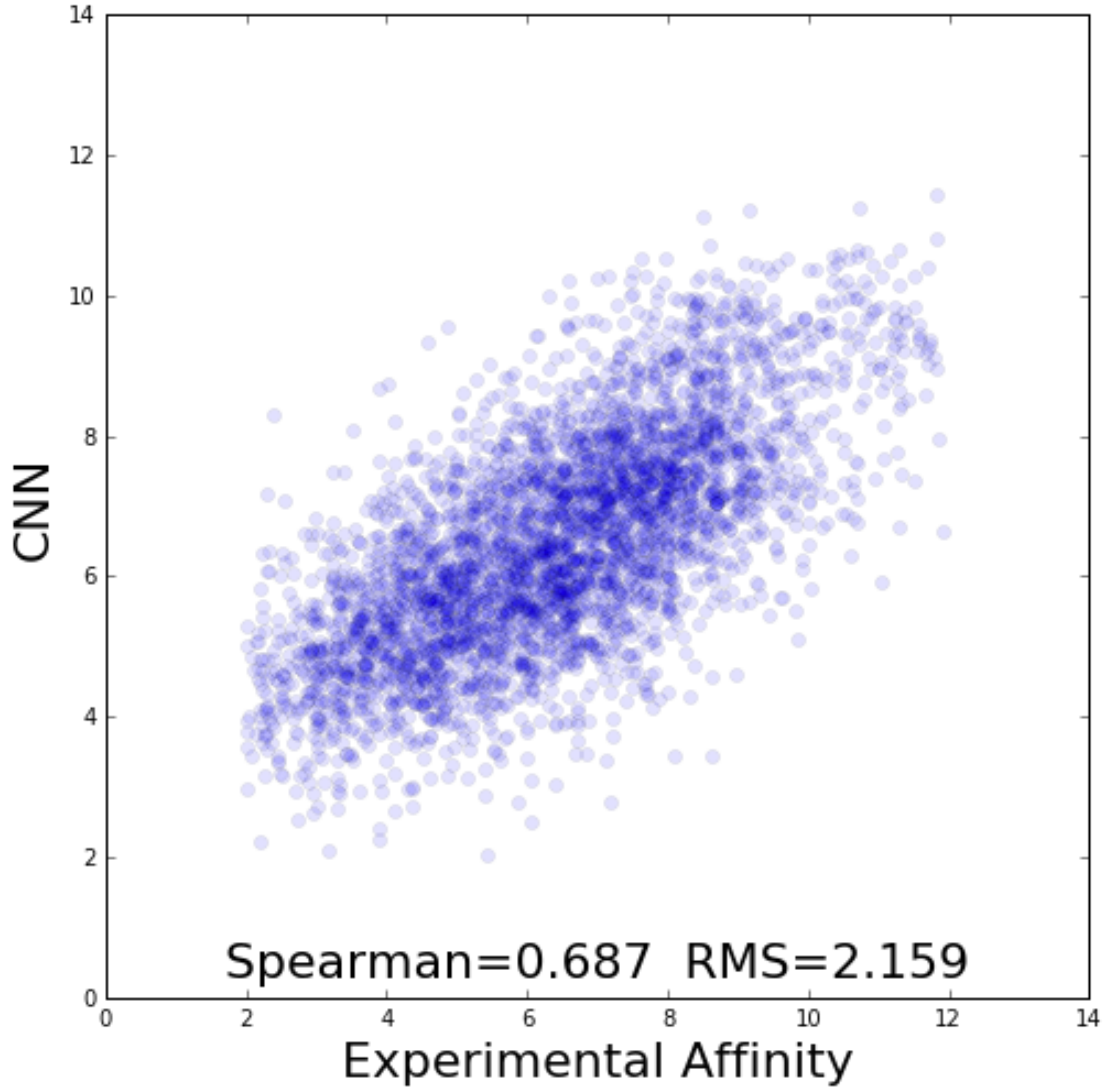



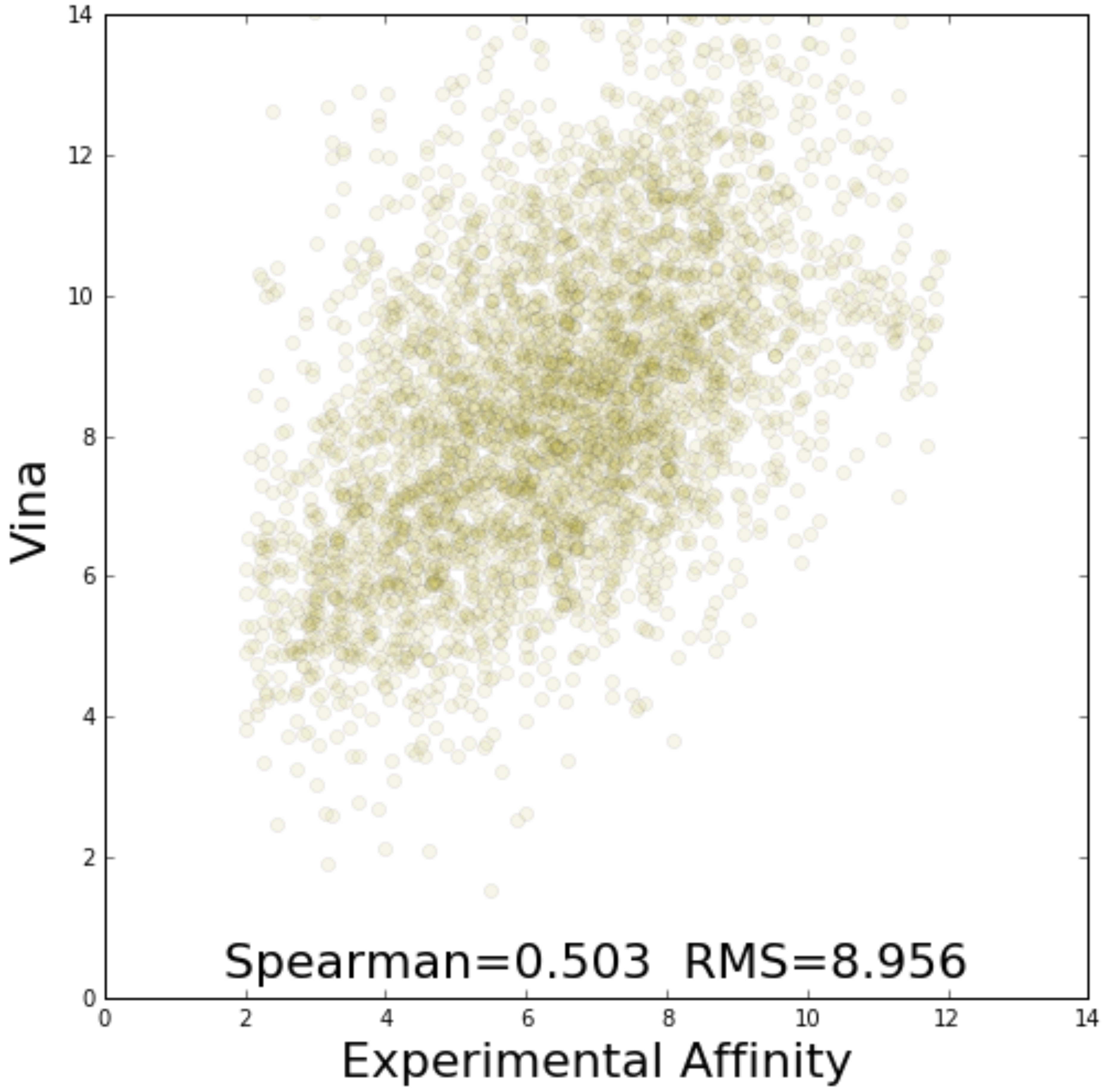




Partially Aligned Poses
Combined 2:1 Training Set

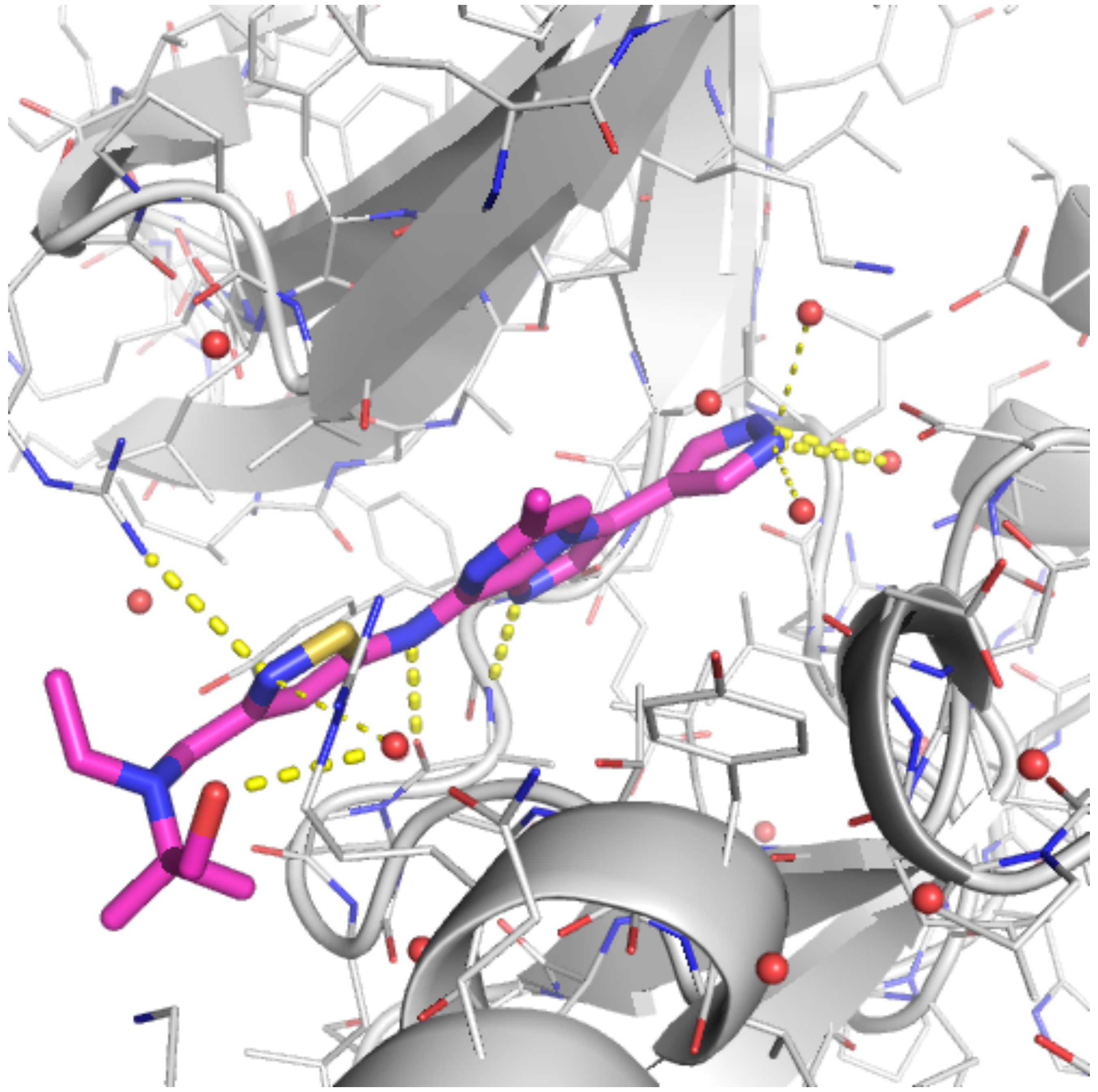
# Affinity Prediction



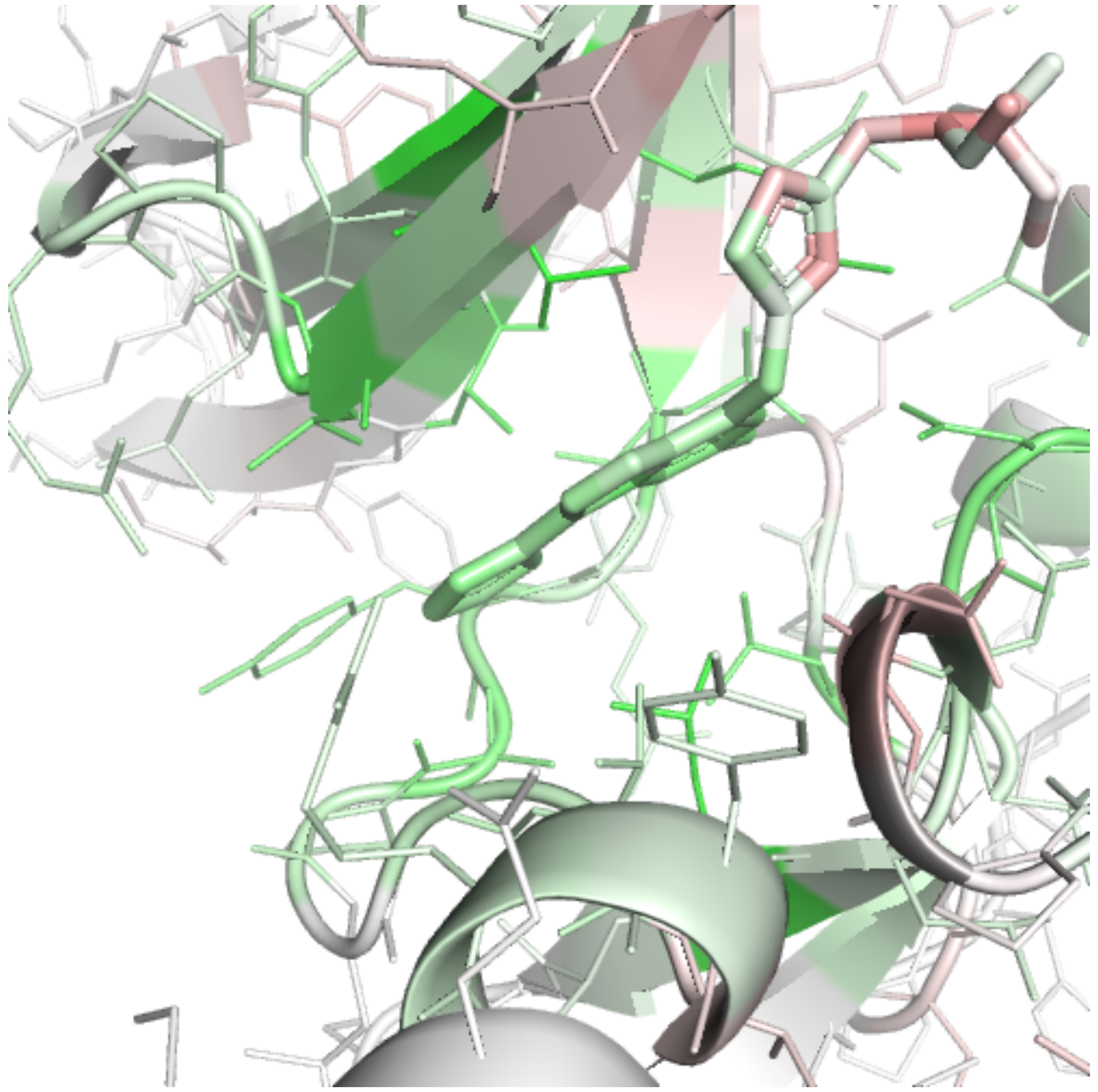



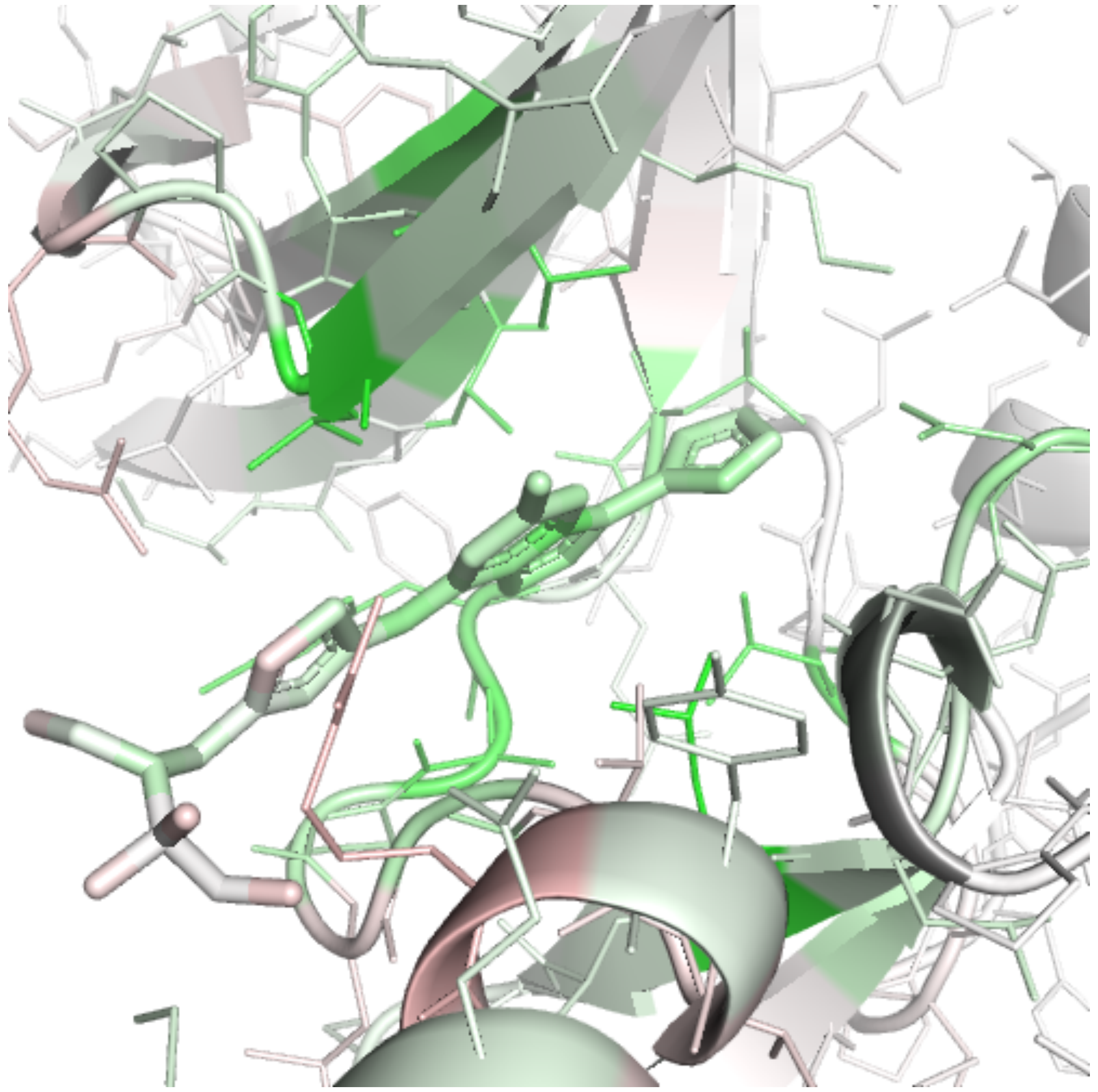
# Examples



3MYG

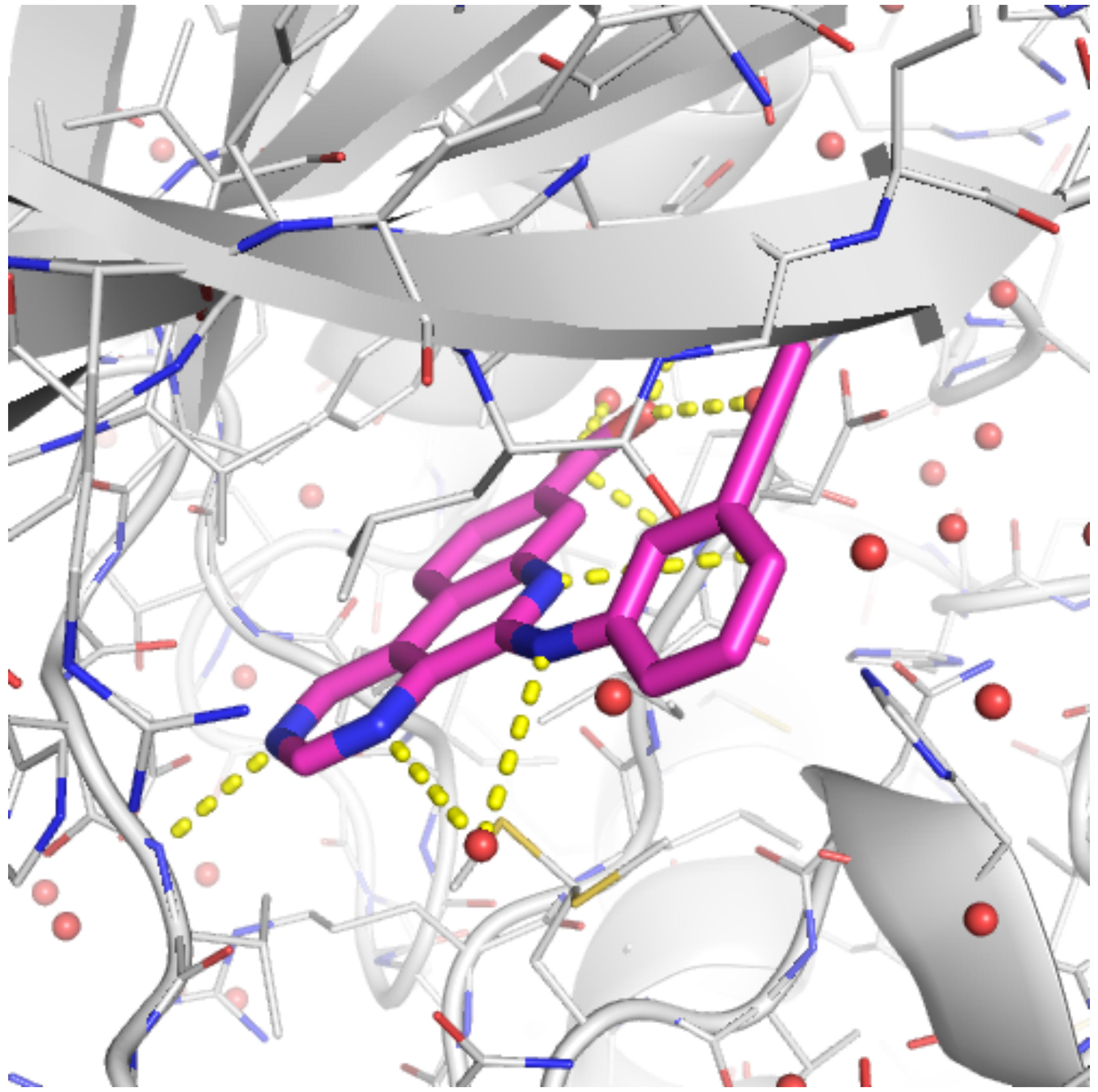


Vina (12.71Å)

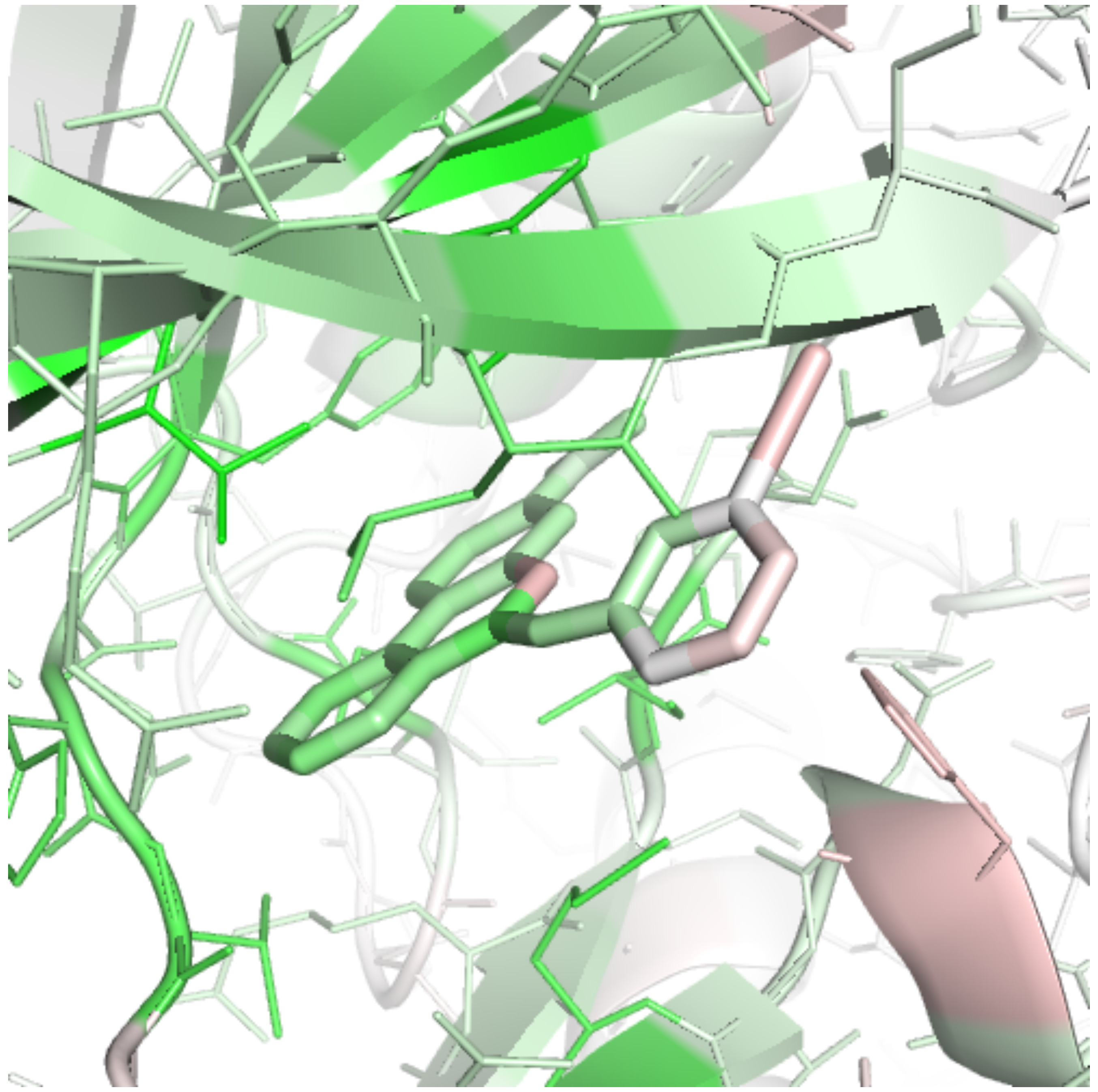


CNN (0.96Å)

# Examples

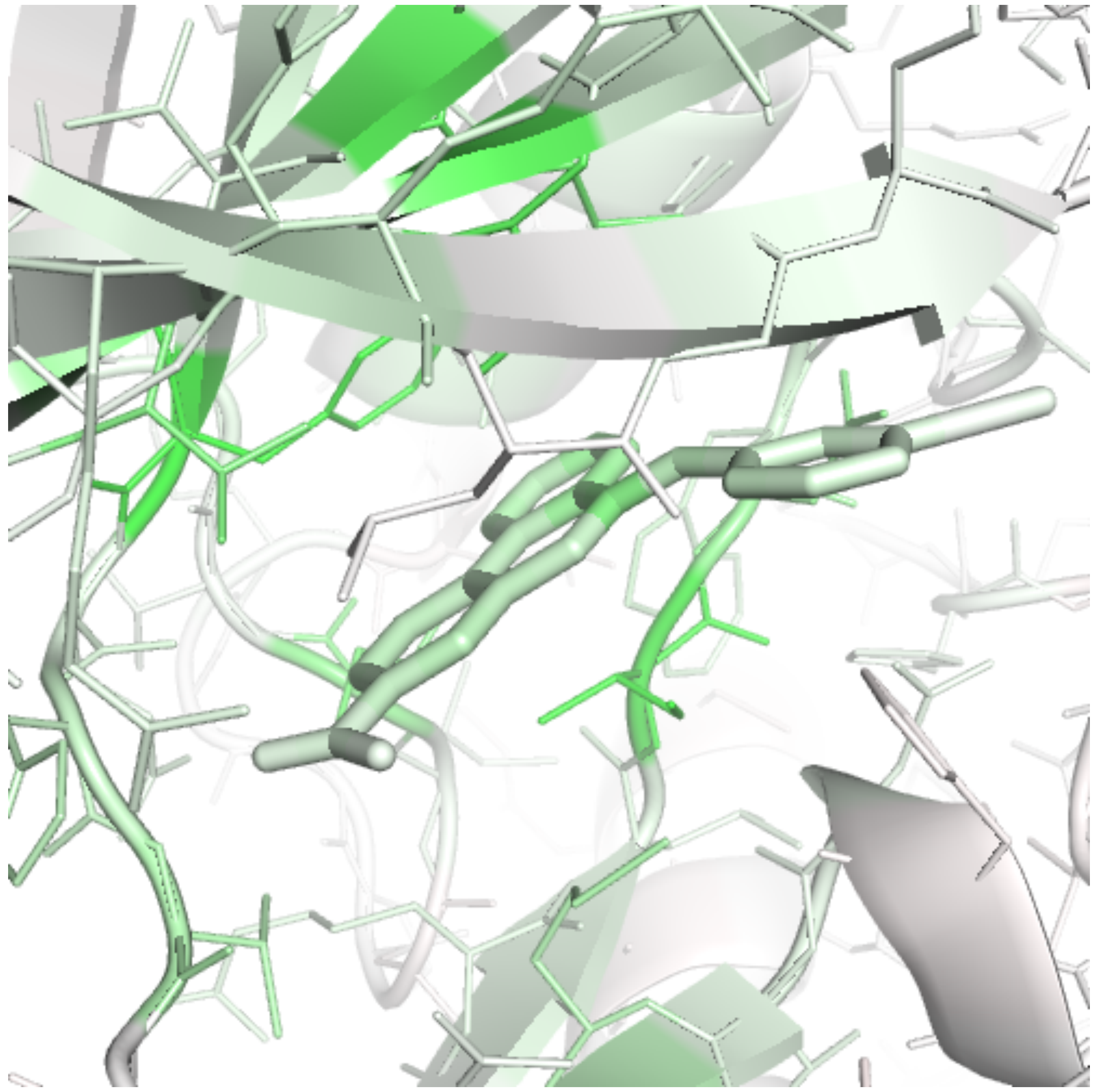


3PE2



Vina (0.25Å)

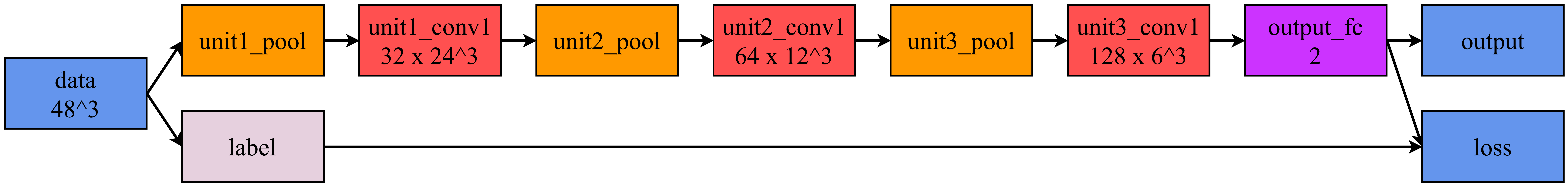
😃



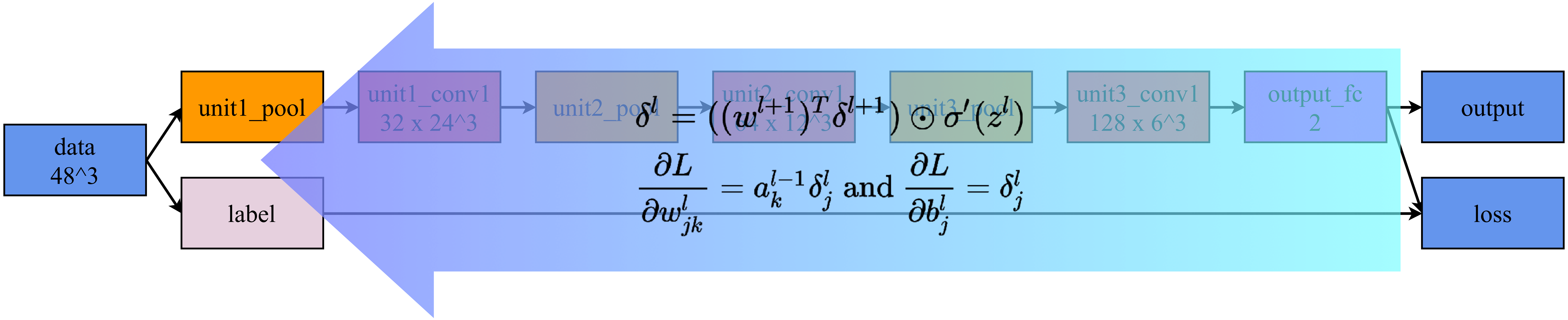
CNN (5.27Å)

🙁

# Beyond Scoring



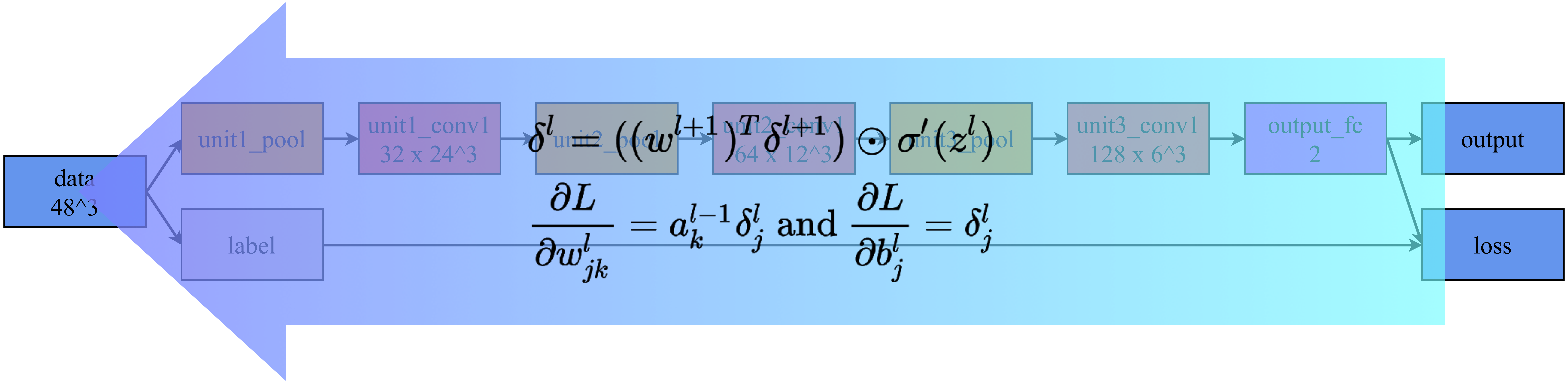
# Beyond Scoring




$$\delta^l = ((w^{l+1})^T \delta^{l+1}) \odot \sigma'(z^l)$$

$$\frac{\partial L}{\partial w^l_{jk}} = a^{l-1}_k \delta^l_j \text{ and } \frac{\partial L}{\partial b^l_j} = \delta^l_j$$

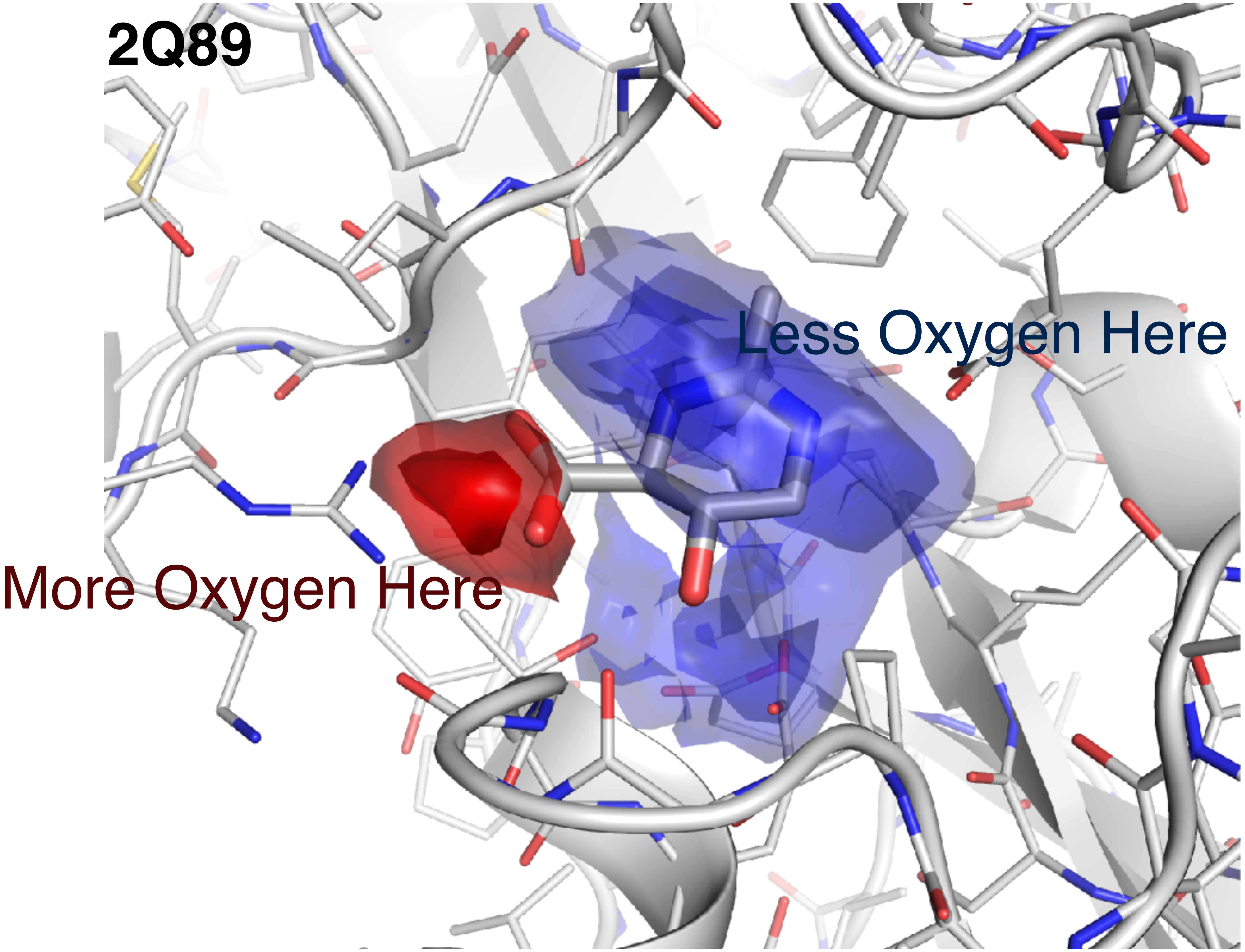
# Beyond Scoring



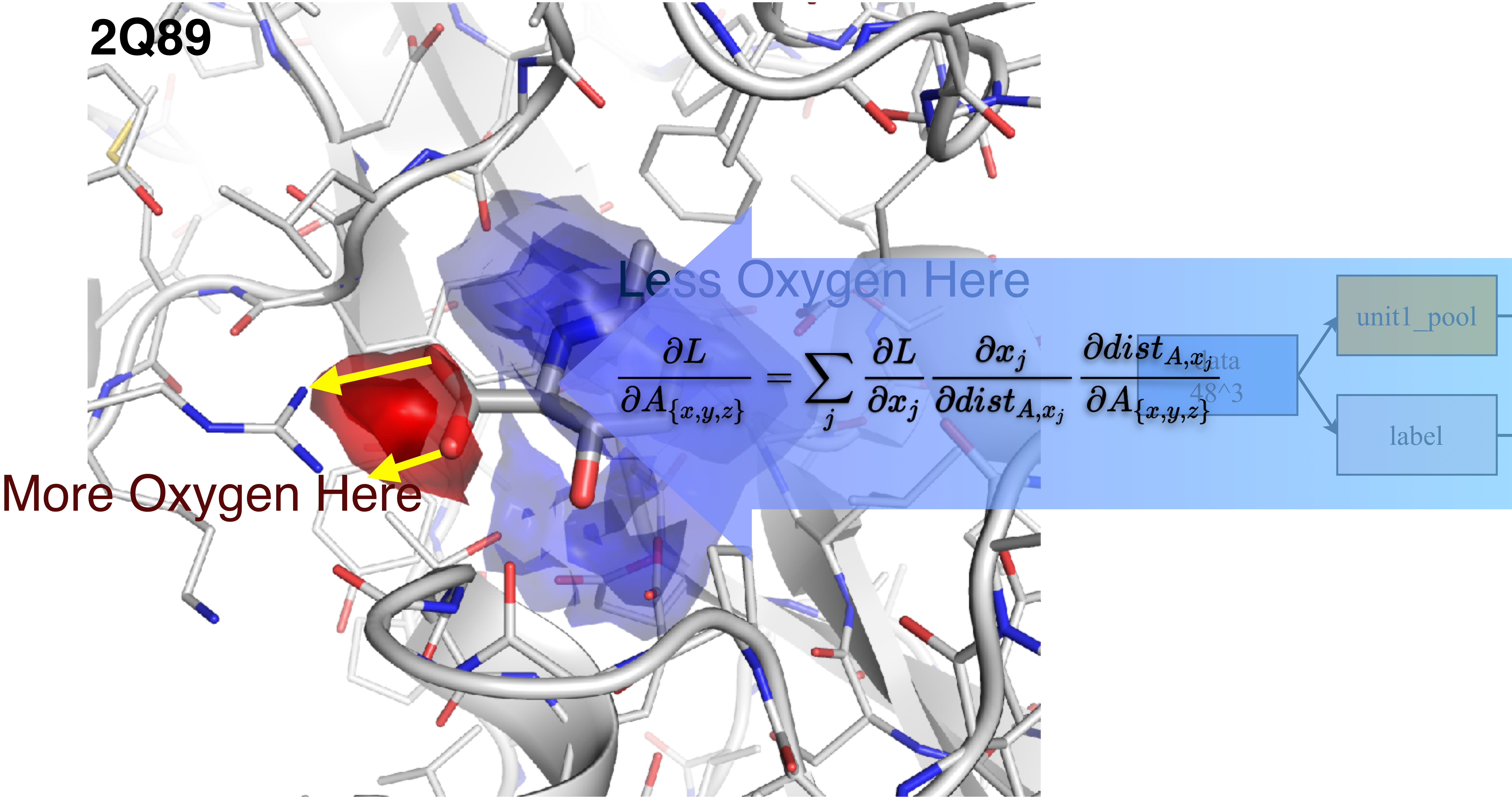

$$\delta^l = ((w^{l+1})^T \delta^{l+1}) \odot \sigma'(z^l)$$

$$\frac{\partial L}{\partial w^l_{jk}} = a^{l-1}_k \delta^l_j \text{ and } \frac{\partial L}{\partial b^l_j} = \delta^l_j$$

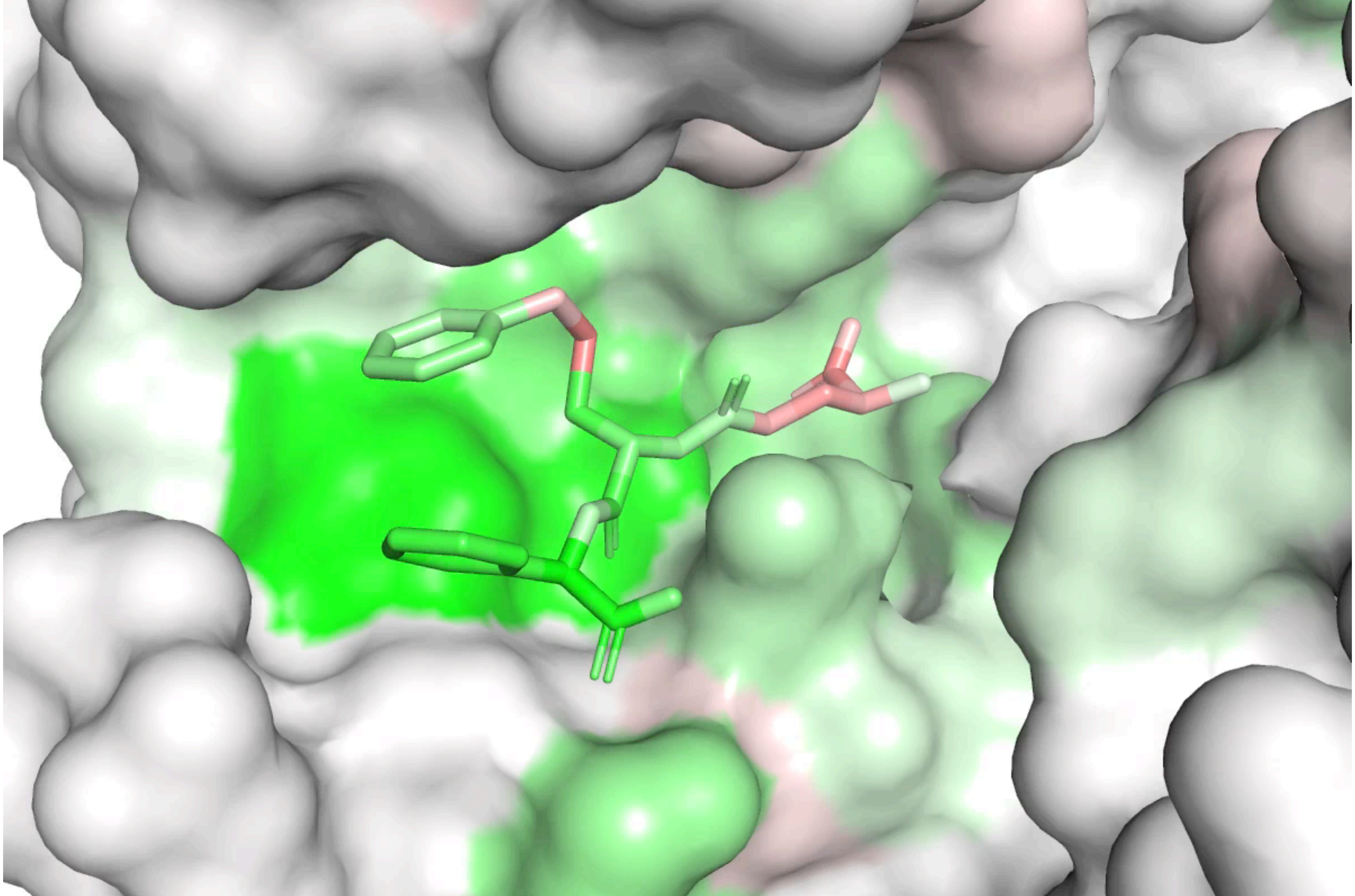
# Beyond Scoring

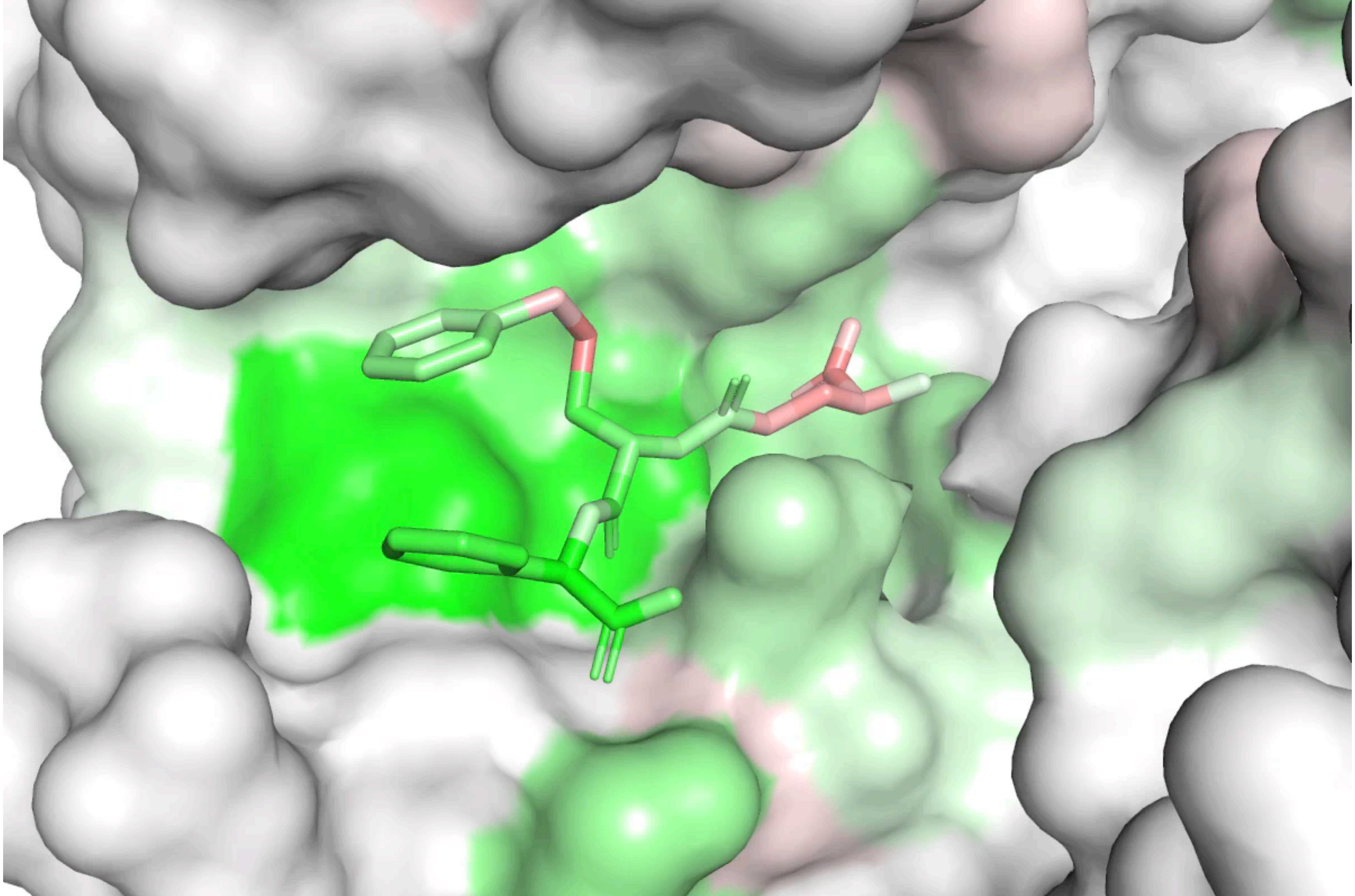


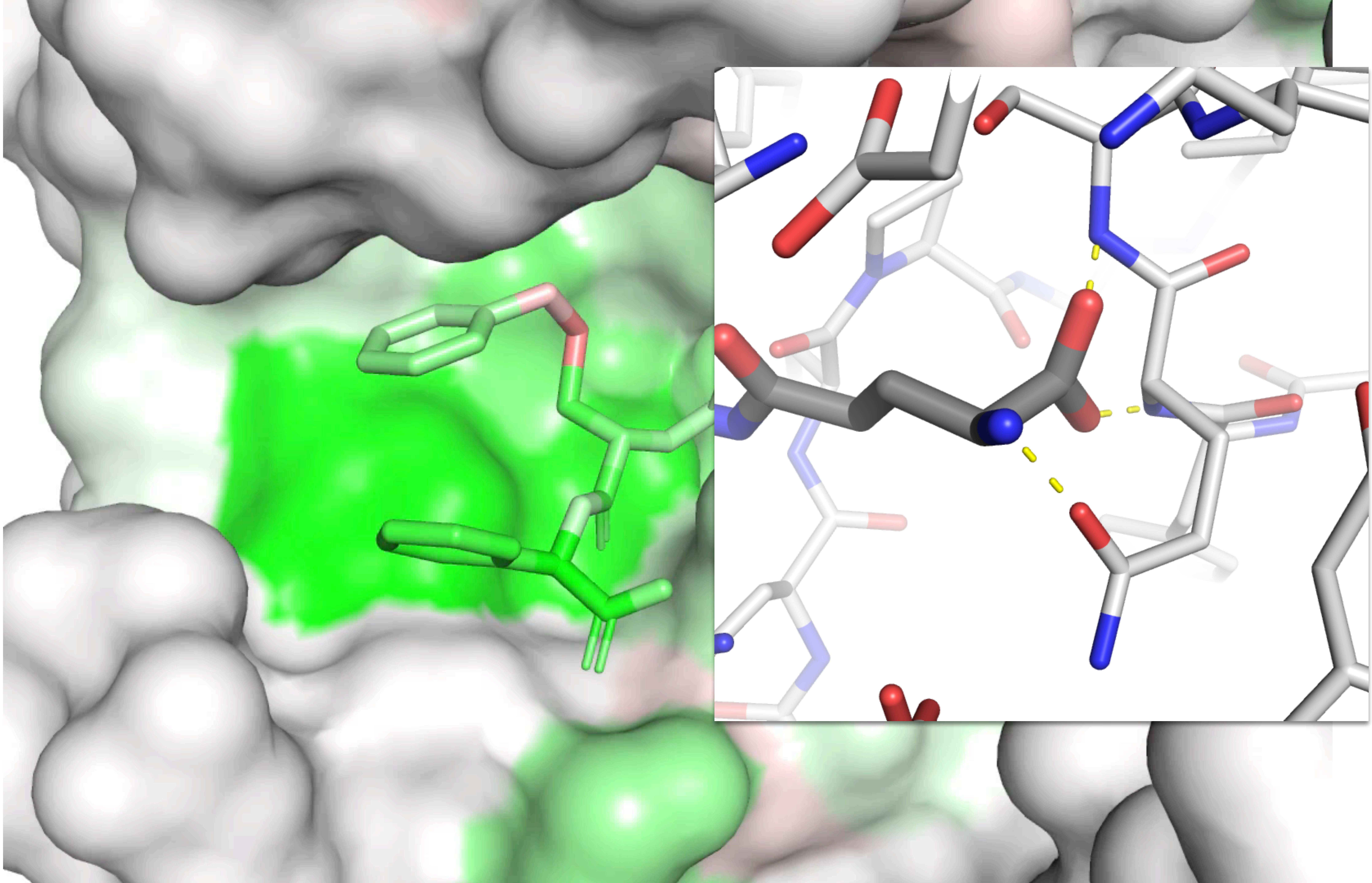
# Beyond Scoring

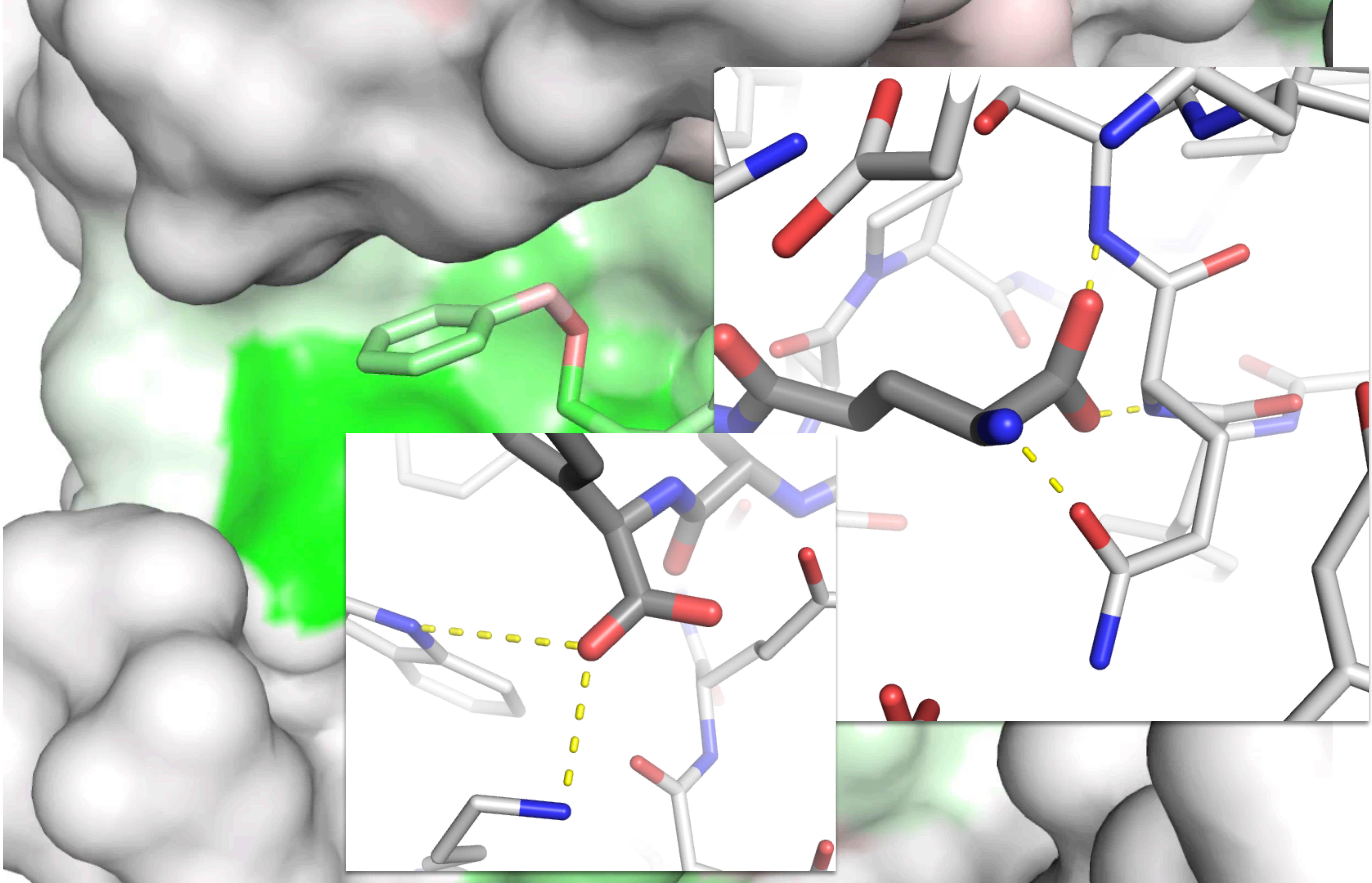


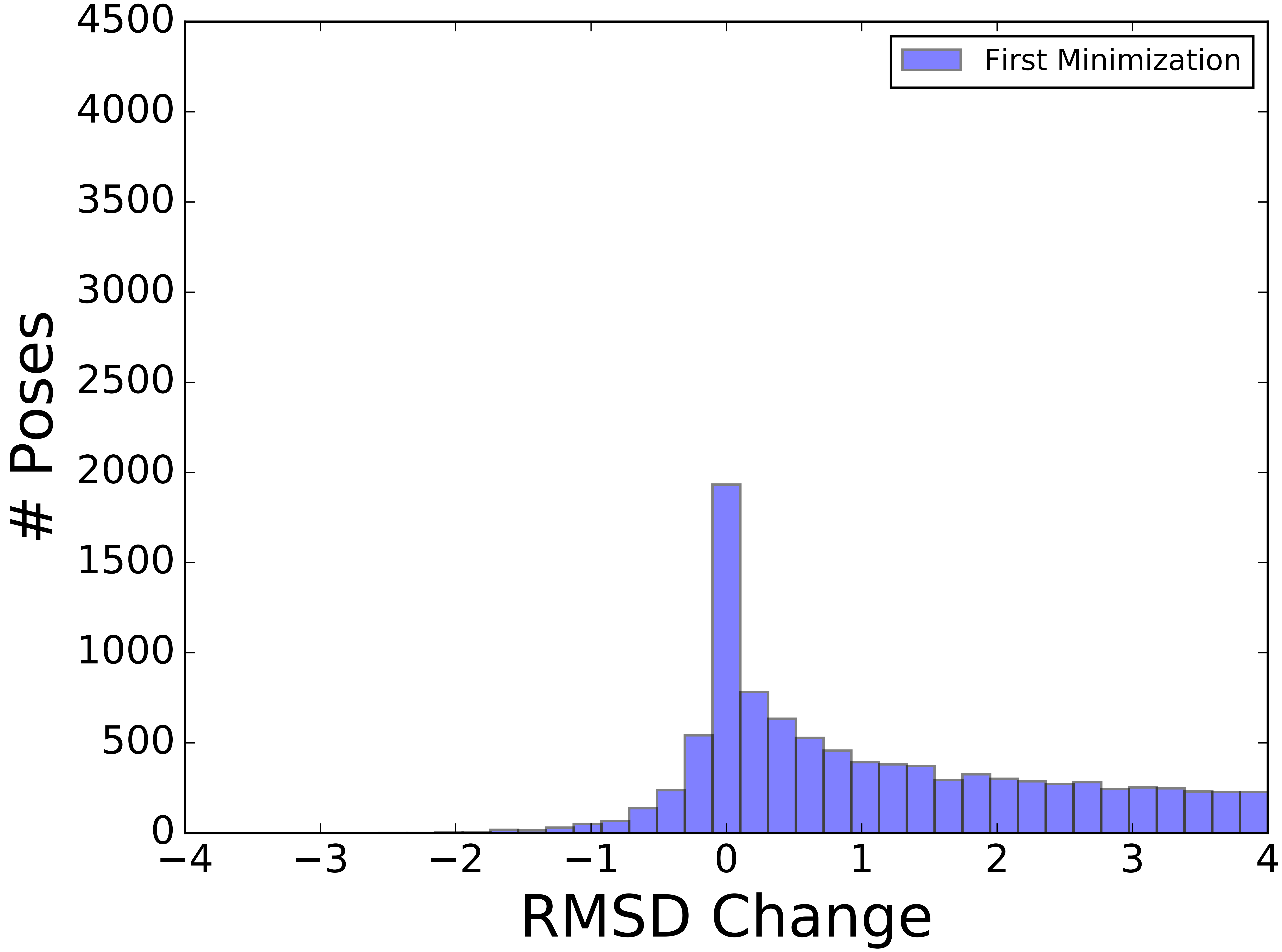

$$\frac{\partial L}{\partial A_{\{x,y,z\}}} = \sum_j \frac{\partial L}{\partial x_j} \frac{\partial x_j}{\partial dist_{A,x_j}} \frac{\partial dist_{A,x_j}}{\partial A_{\{x,y,z\}}}$$

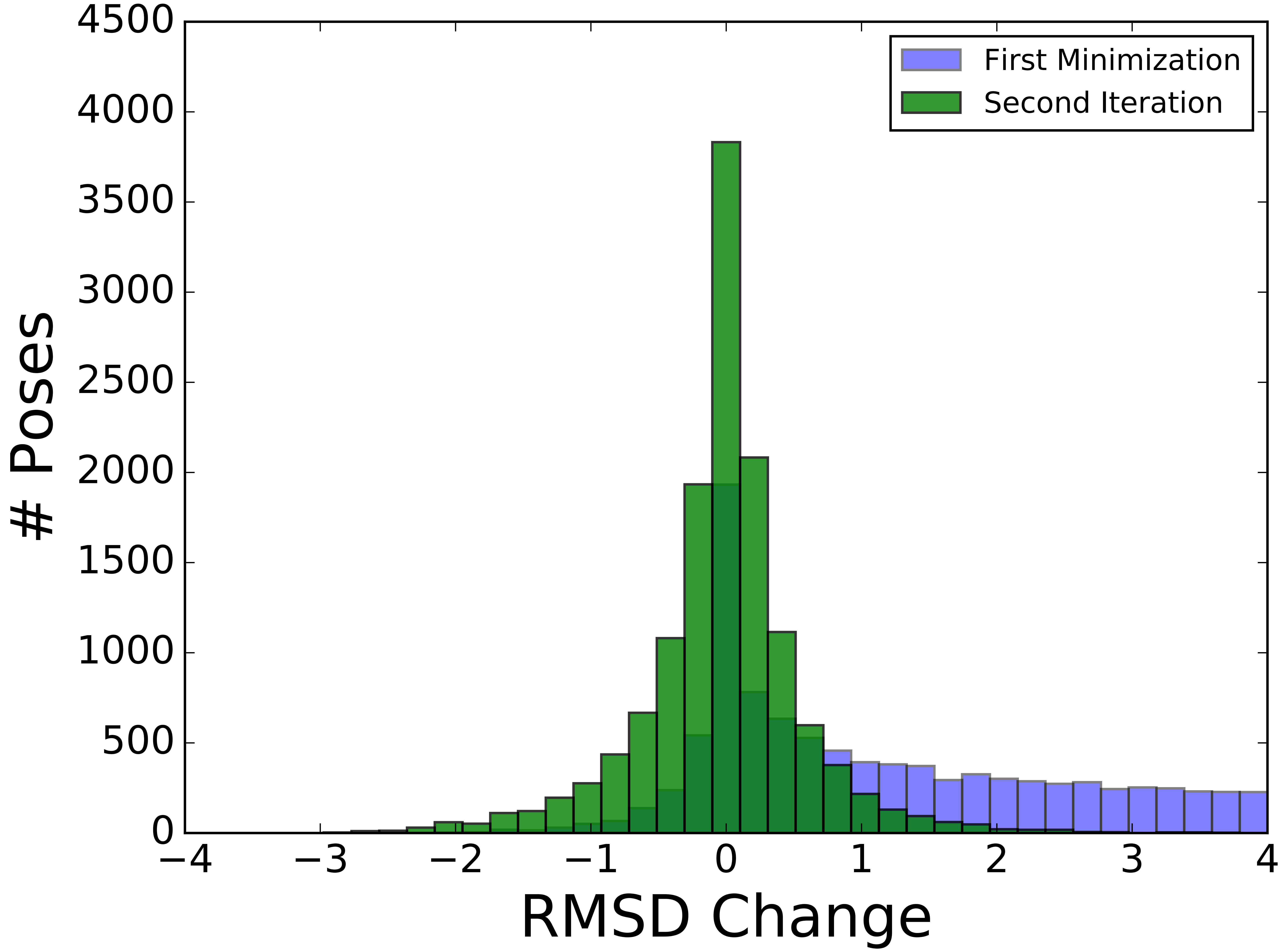


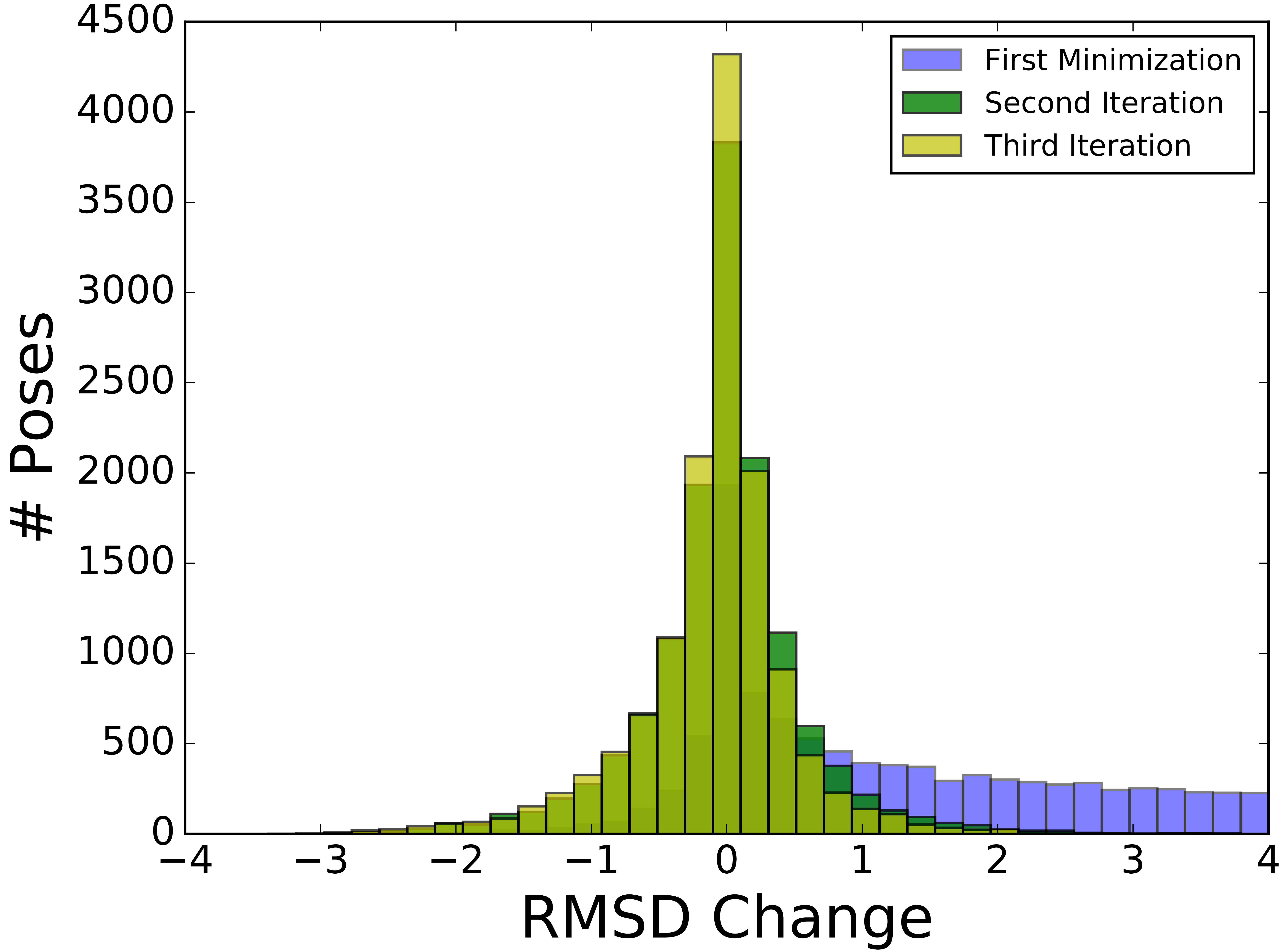







4500
4000
3500
3000
2500
2000
1500
1000
500
0
# Poses
First Minimization
−4
−3
−2
−1
0
1
2
3
4
RMSD Change


First Minimization
Second Iteration
4500
4000
3500
3000
2500
2000
1500
1000
500
0
# Poses
−4
−3
−2
−1
0
1
2
3
4
RMSD Change



First Minimization
Second Iteration
Third Iteration
# Poses
0
500
1000
1500
2000
2500
3000
3500
4000
4500
−4
−3
−2
−1
0
1
2
3
4
RMSD Change

# The Future

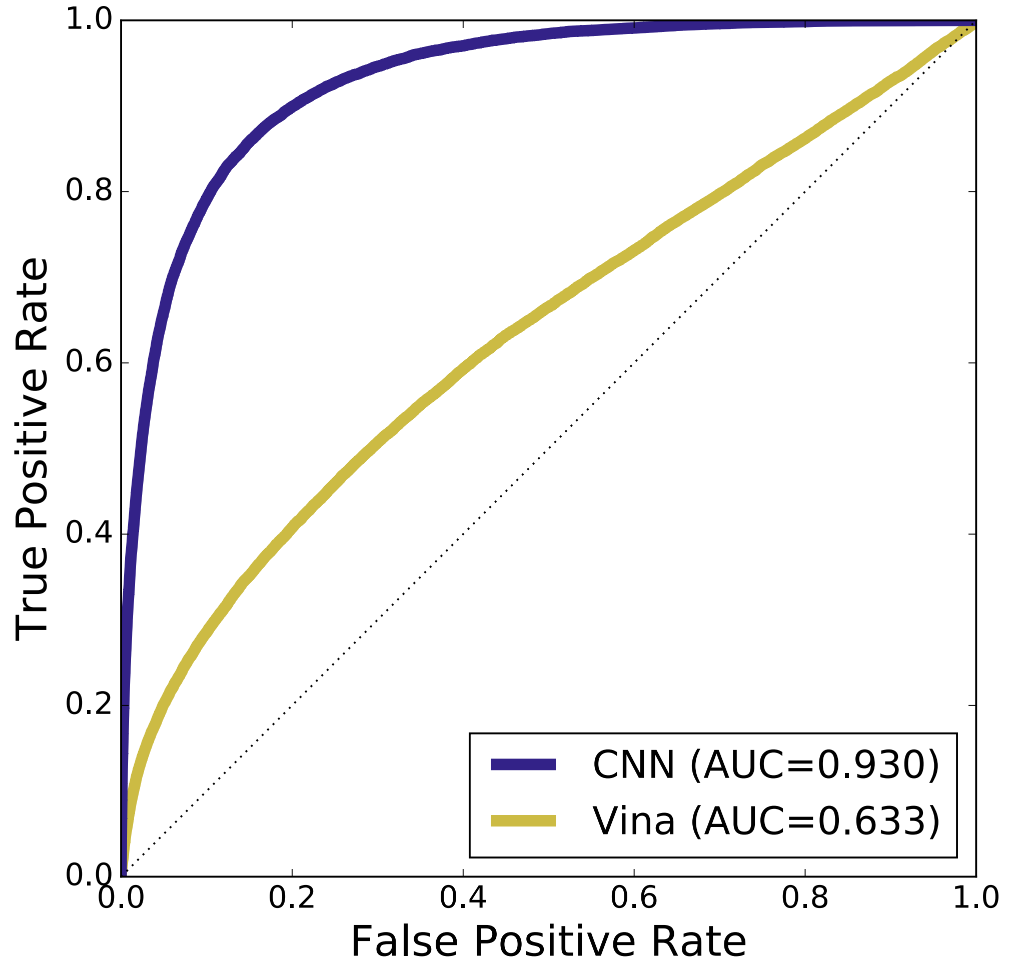
Pose Selection

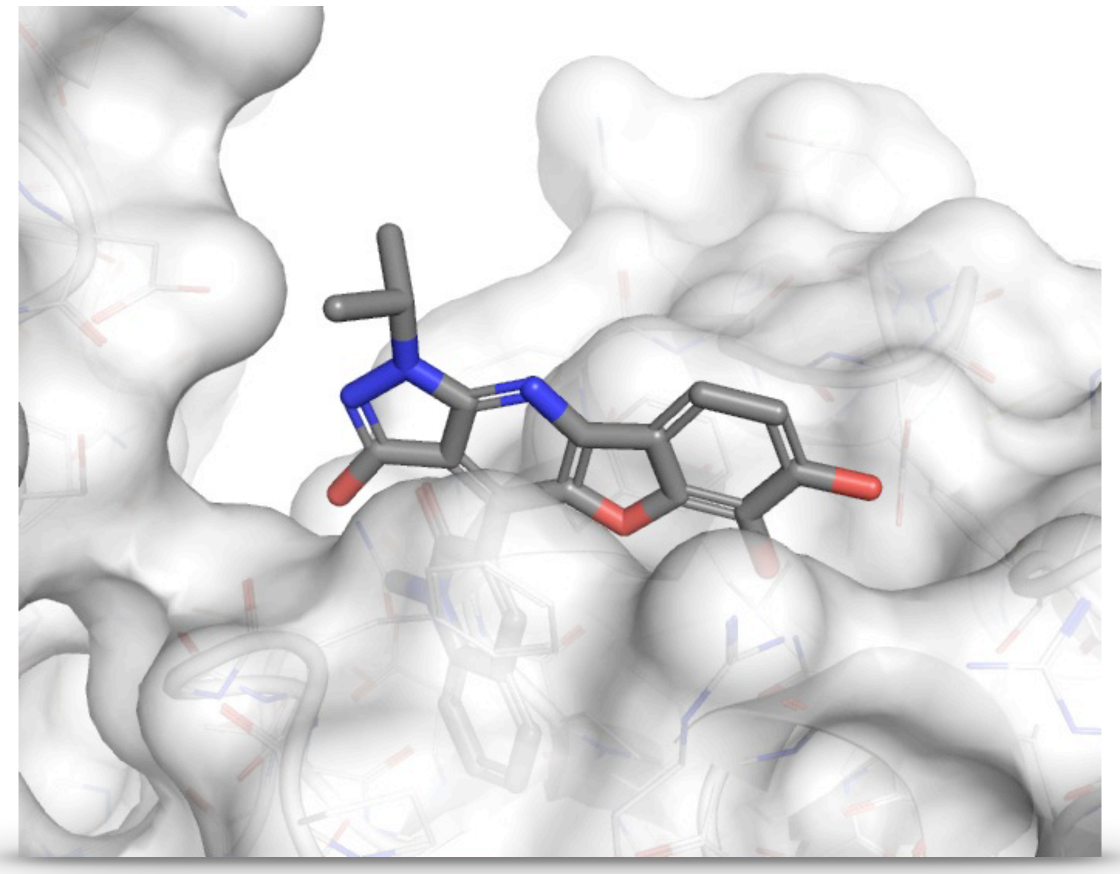
Iterative Training

Pose *Generation*

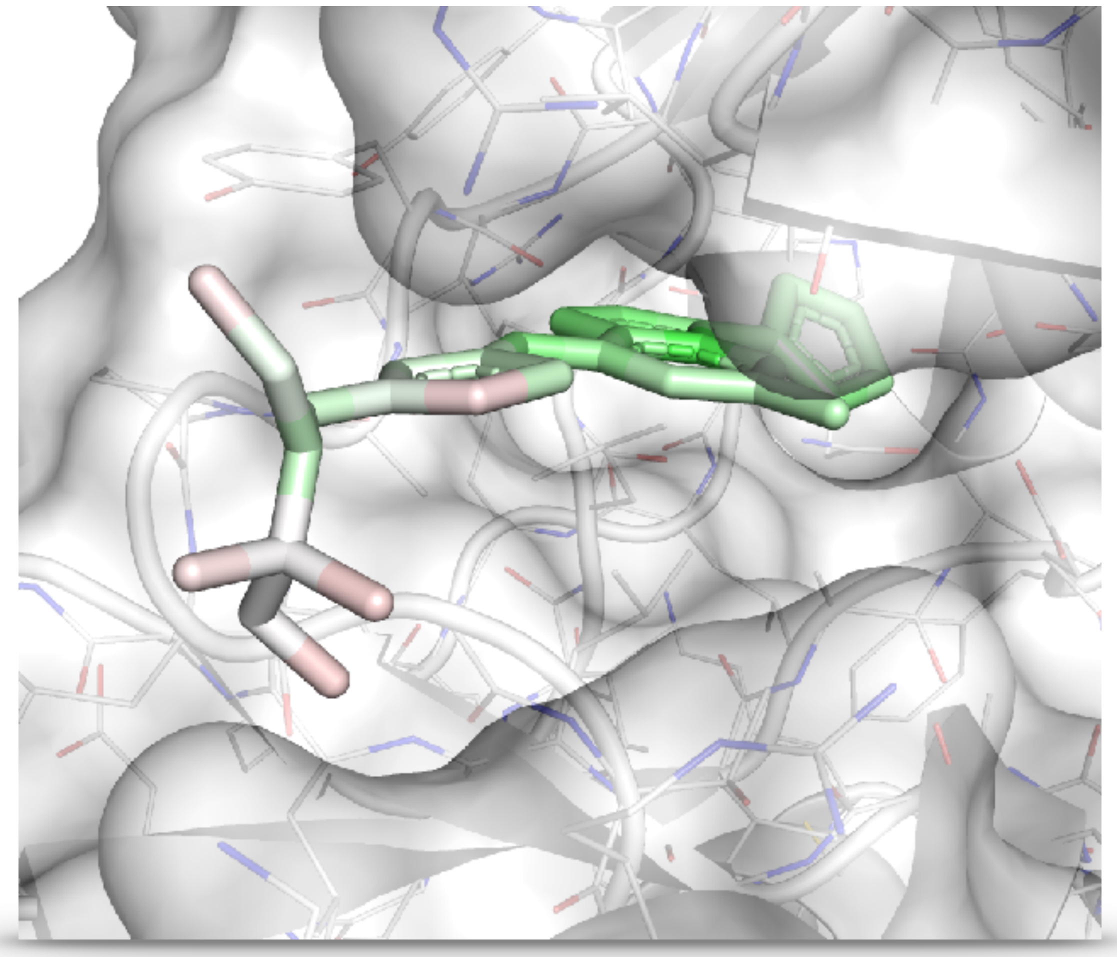
Iterative Training

*Compound* Generation



**Virtual Screening**



**Lead Optimization**

# The Future

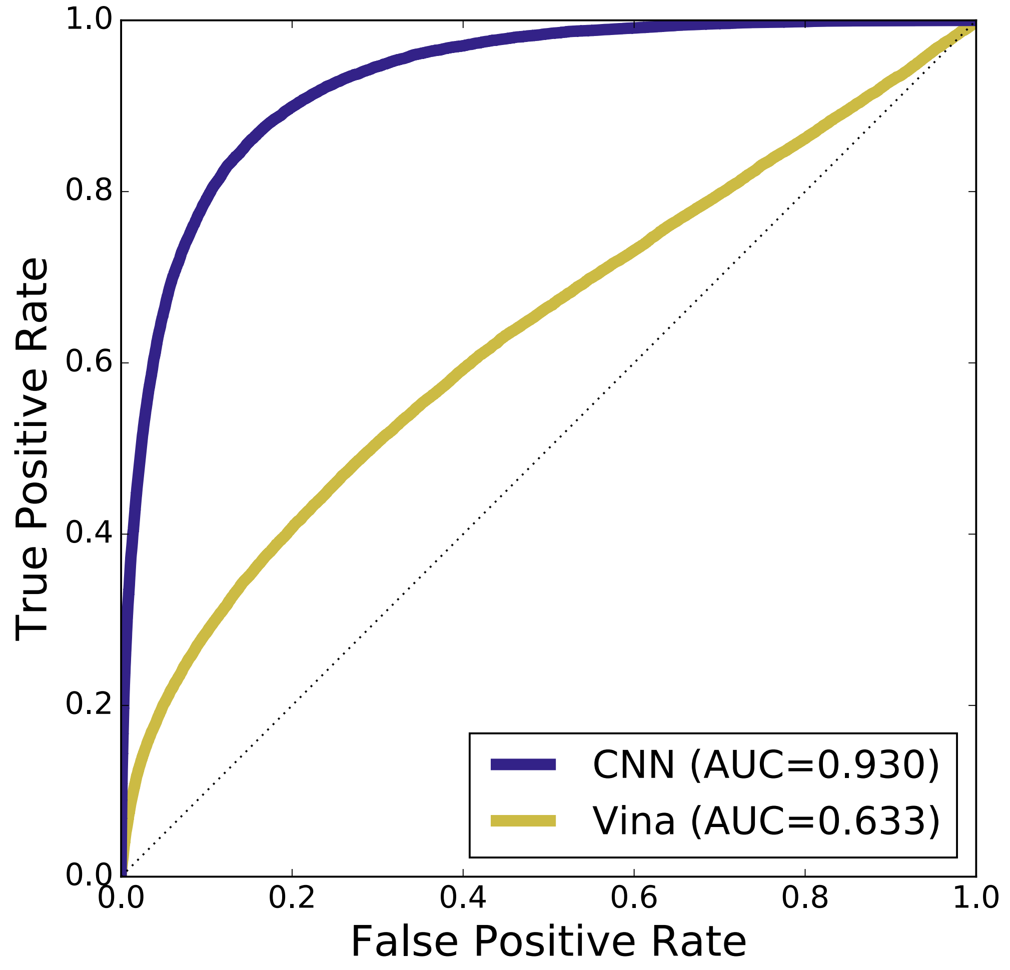
Pose Selection

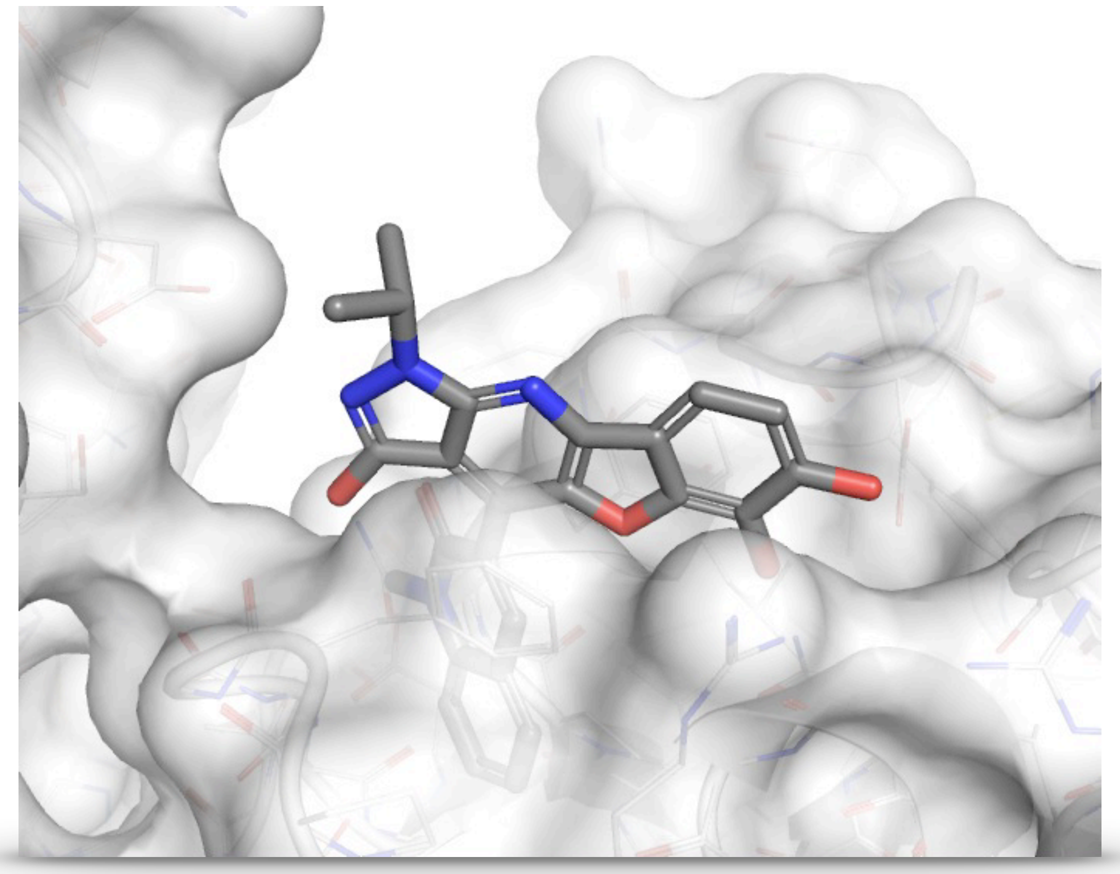
Iterative Training

Pose *Generation*

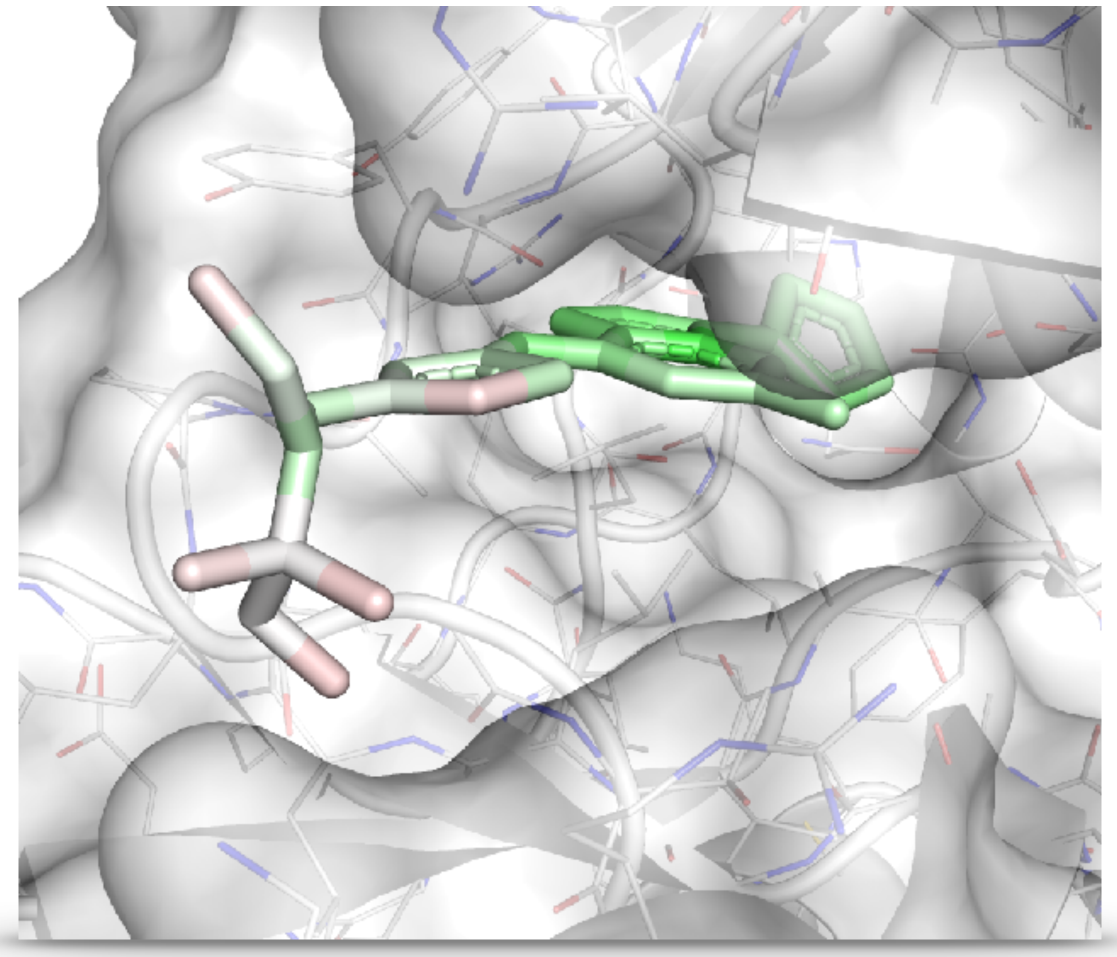
Iterative Training

*Compound* Generation

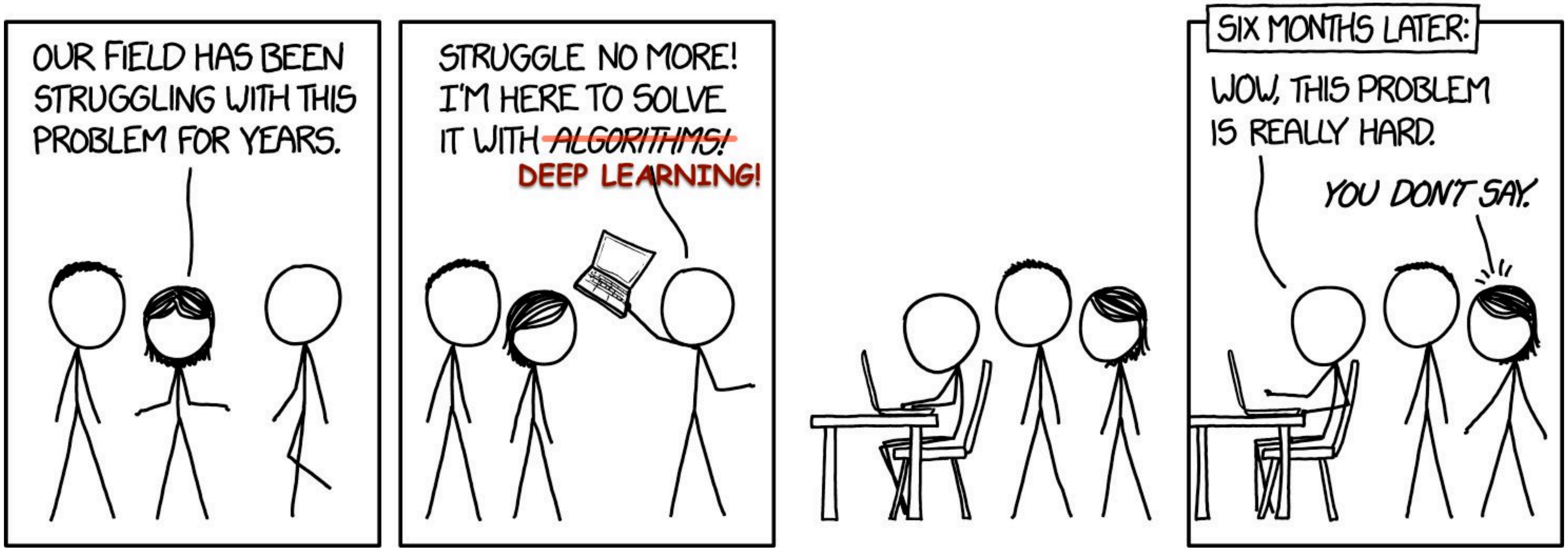


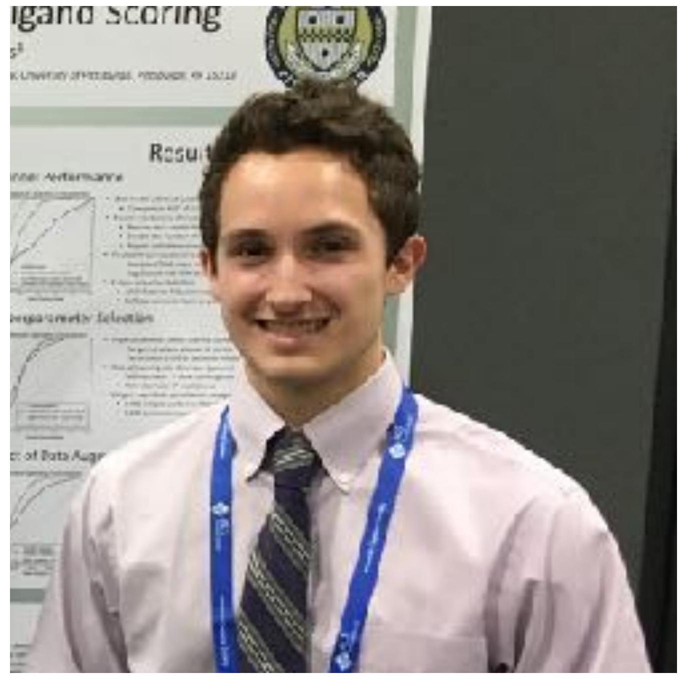
**Virtual Screening**

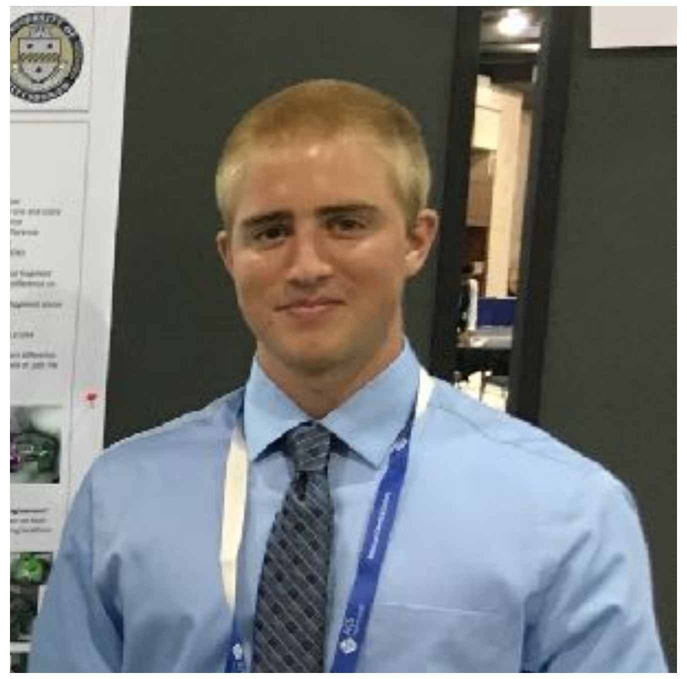


**Lead Optimization**

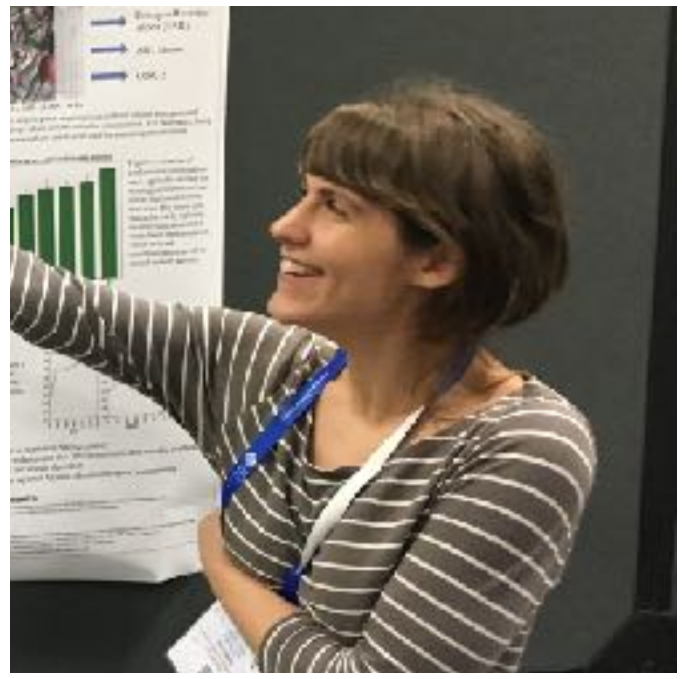

OUR FIELD HAS BEEN STRUGGLING WITH THIS PROBLEM FOR YEARS.
STRUGGLE NO MORE! I'M HERE TO SOLVE IT WITH ~~ALGORITHMS!~~
DEEP LEARNING!
SIX MONTHS LATER:
WOW, THIS PROBLEM IS REALLY HARD.
YOU DON'T SAY.

# Acknowledgements


Matt Ragoza


Josh Hochuli


Elisa Idrobo


Jocelyn Sunseri

**Group Members**
Jocelyn Sunseri
Matt Ragoza
Josh Hochuli
Pulkit Mittal
Alec Helbling
Tamar Skaist
Christopher Dunstan

Department of Computational and Systems Biology

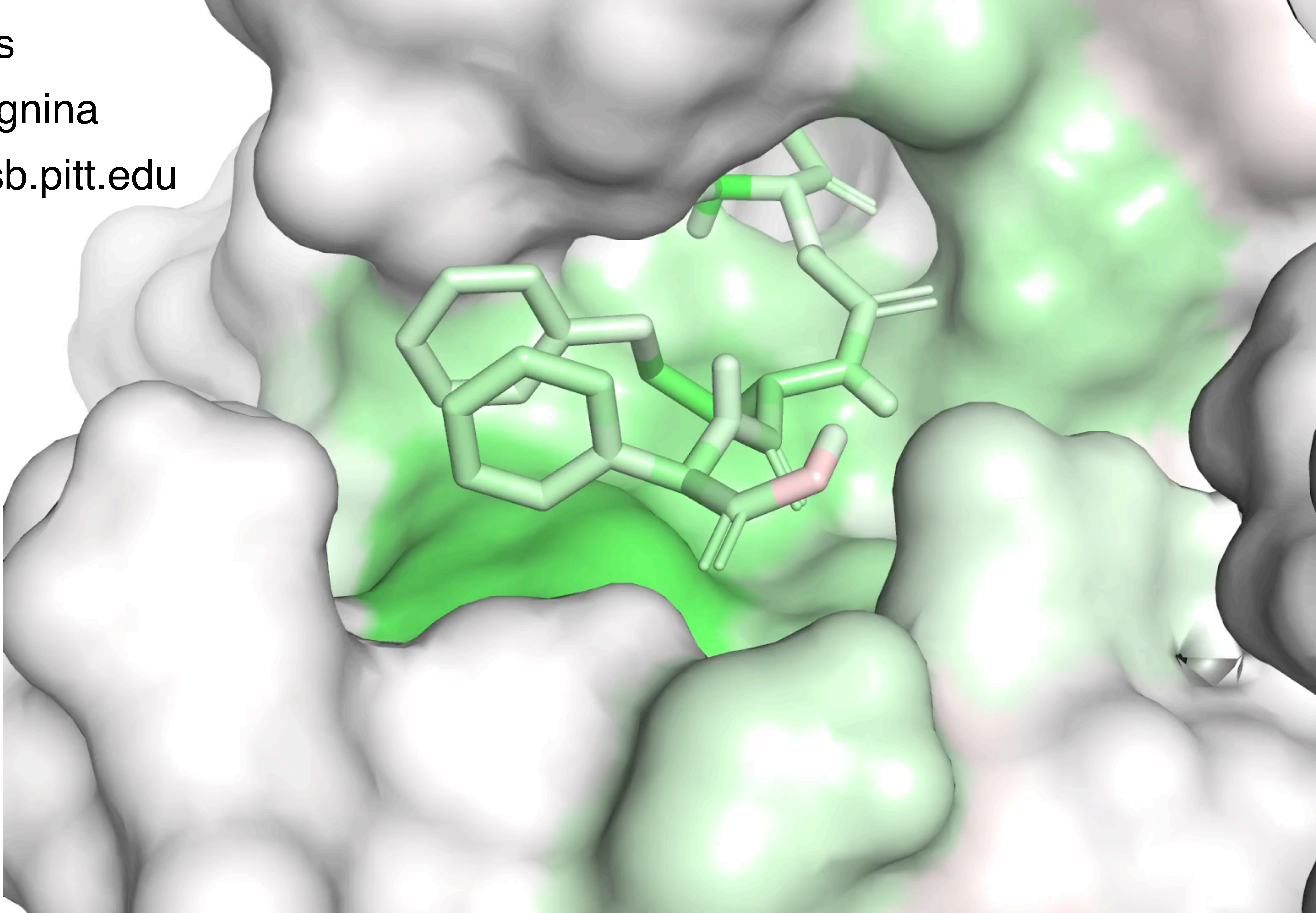
TECBio REU

National Institute of General Medical Sciences
R01GM108340

@david_koes

github.com/gnina

http://bits.csb.pitt.edu



@david_koes

github.com/gnina

http://bits.csb.pitt.edu