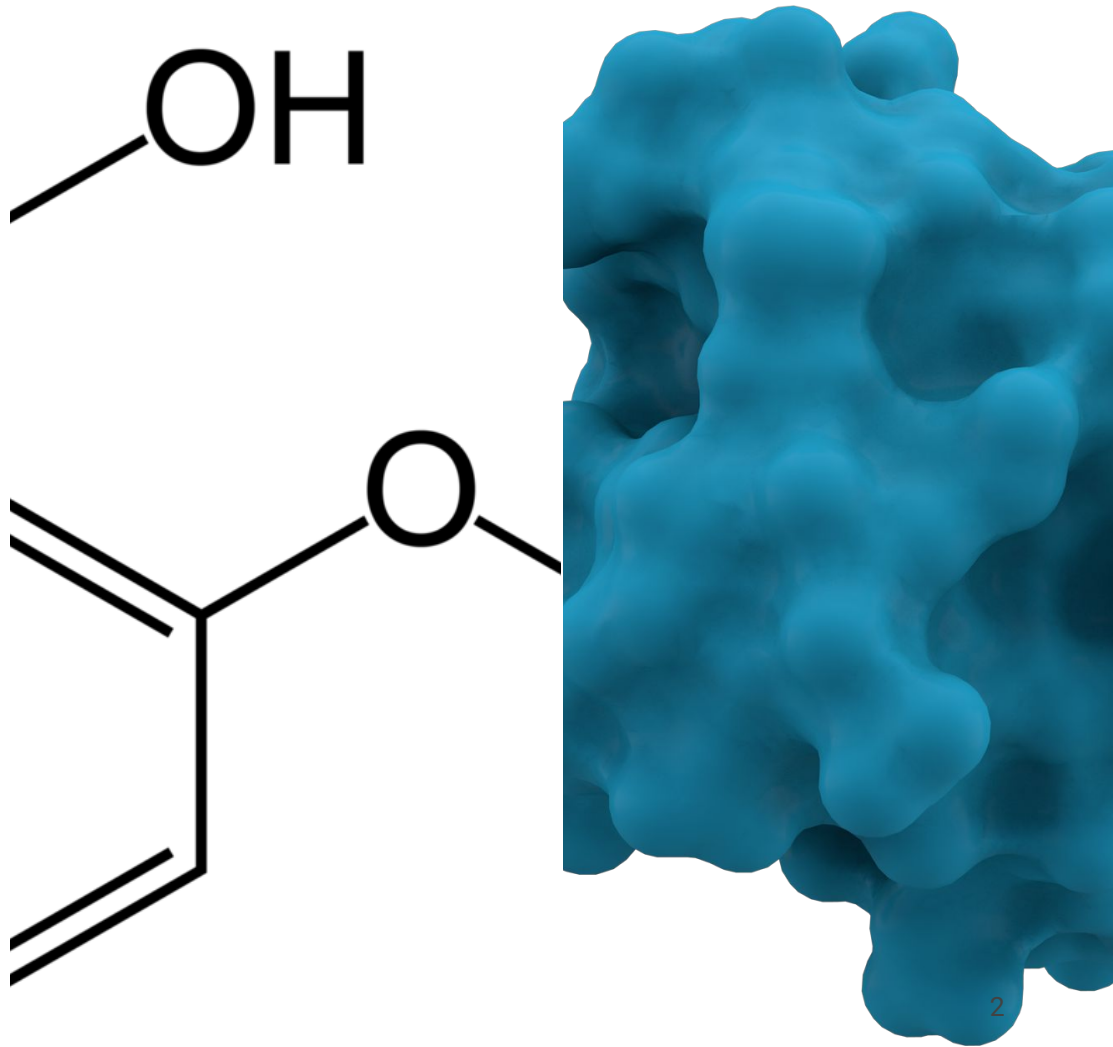


# Scalable Feature Selection and Extraction with Applications in Kinase Polypharmacology

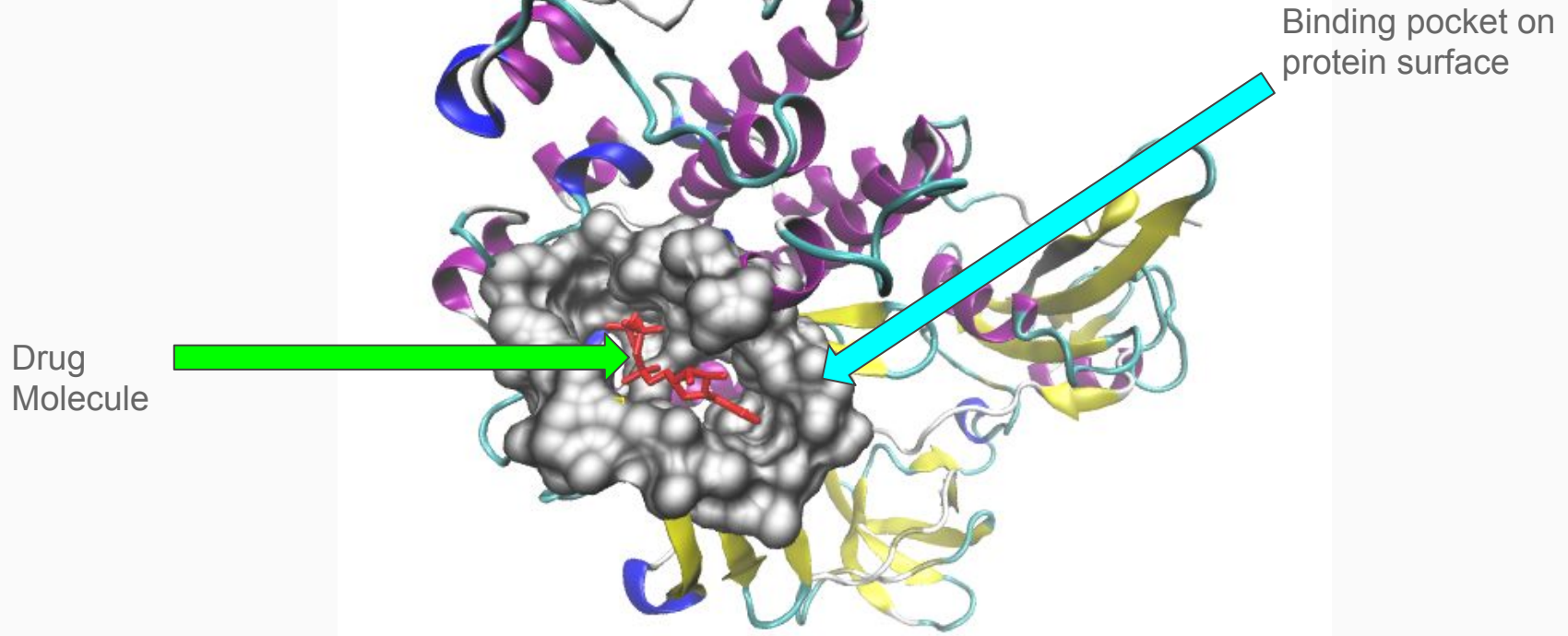
Derek Jones



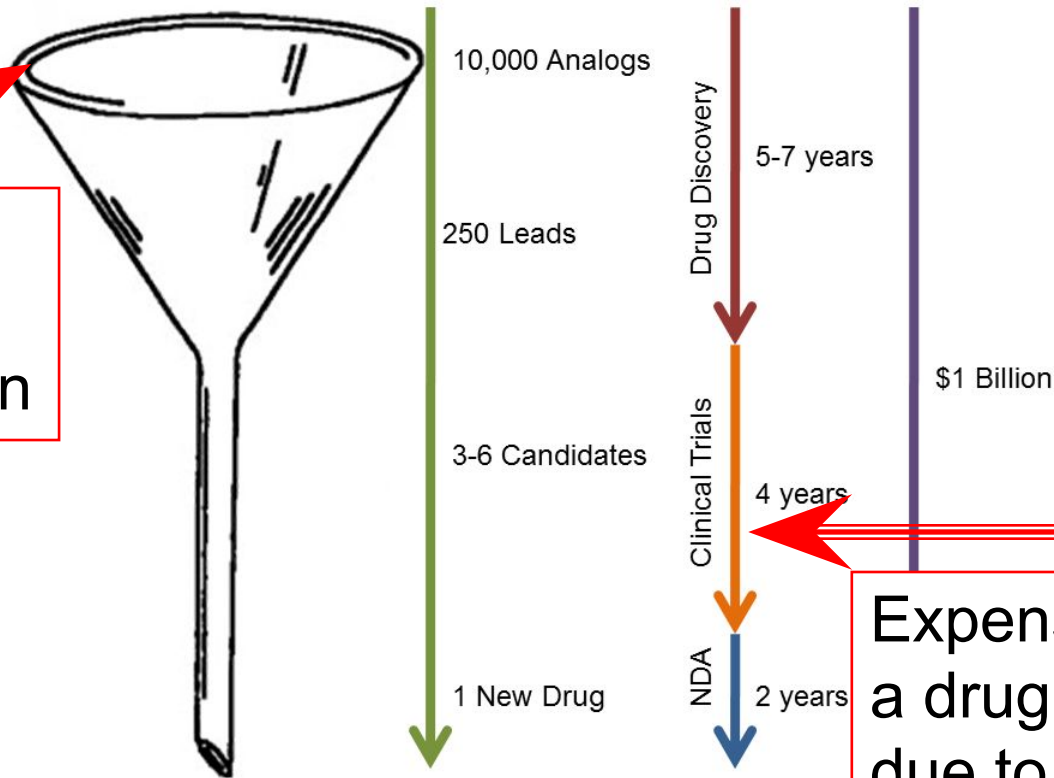
# Drug Discovery



# Drug-Protein Binding Interaction



Predict here  
using virtual  
toxicity screen



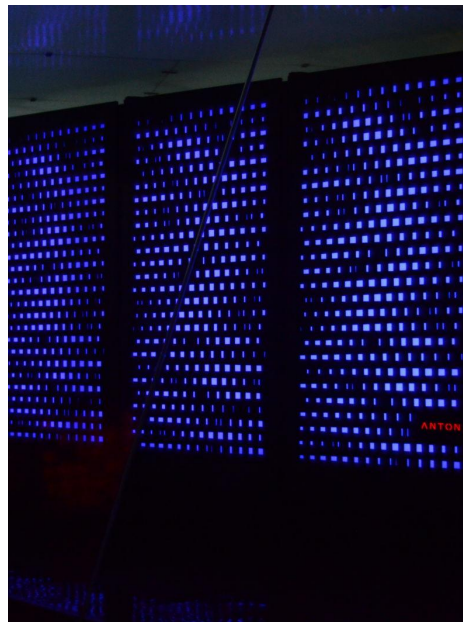
Expensive stage for  
a drug to fail. Often  
due to ADRs.

## Pipeline to New Drugs

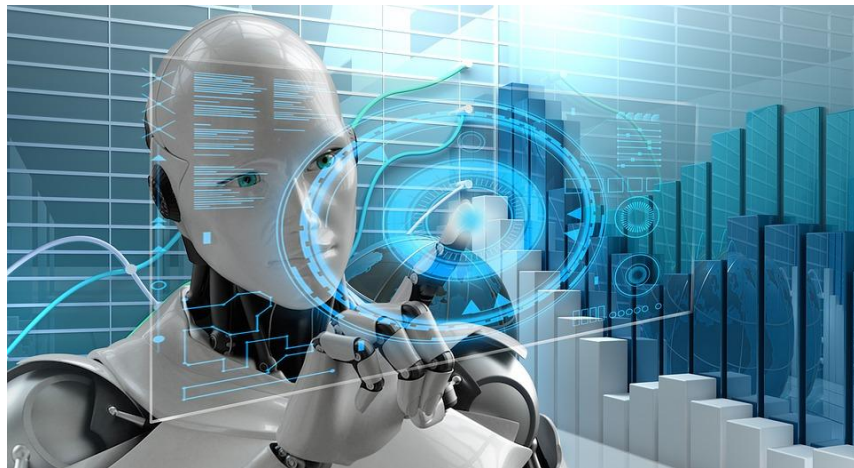
<http://www.chemistry-blog.com/2012/01/04/ted-talk-medicine-for-the-99-hes-about-99-wrong/>

# Challenges in Drug Discovery

- **Massive** search space is estimated to be between  $10^{23}$  -  $10^{60}$
- **High-performance compute** power required
- **Limited access** to large quantities of data



# Improving Drug Discovery with Machine Learning



## Good News:

- More high quality open-source data is becoming available
- Advancements in machine learning/deep learning and computational power

## Our contribution:

- Reduce the amount of domain expertise to reason about drug-protein binding interactions via data-driven approaches

# Data-Driven Feature Selection in Binding Affinity Models

“Polypharmacology Within the Full Kinome: a Machine Learning Approach”, Derek Jones, Jeevith Bopaiah, Fatemah Alghamedy, Nathan Jacobs, Heidi Weiss, W.A. de Jong, Sally Ellingson, AMIA 2018 Informatics Summit, San Francisco

---

# Motivation

- Protein kinases represent a large proportion of proteins that have essential functions
- Kinases are difficult to target selectively
- Potential treatments could likely lead to toxic off-target effects



# Our Contributions

- Expand the scope of previous multi-protein models by using a more comprehensive set of features
- Compiled a large, high-dimensional dataset
- Automatic feature selection algorithm to discard less informative information to handle the dimensionality

# What are we trying to learn?

$$P(\text{binding} \mid \text{protein } p, \text{drug molecule } m) = ?$$

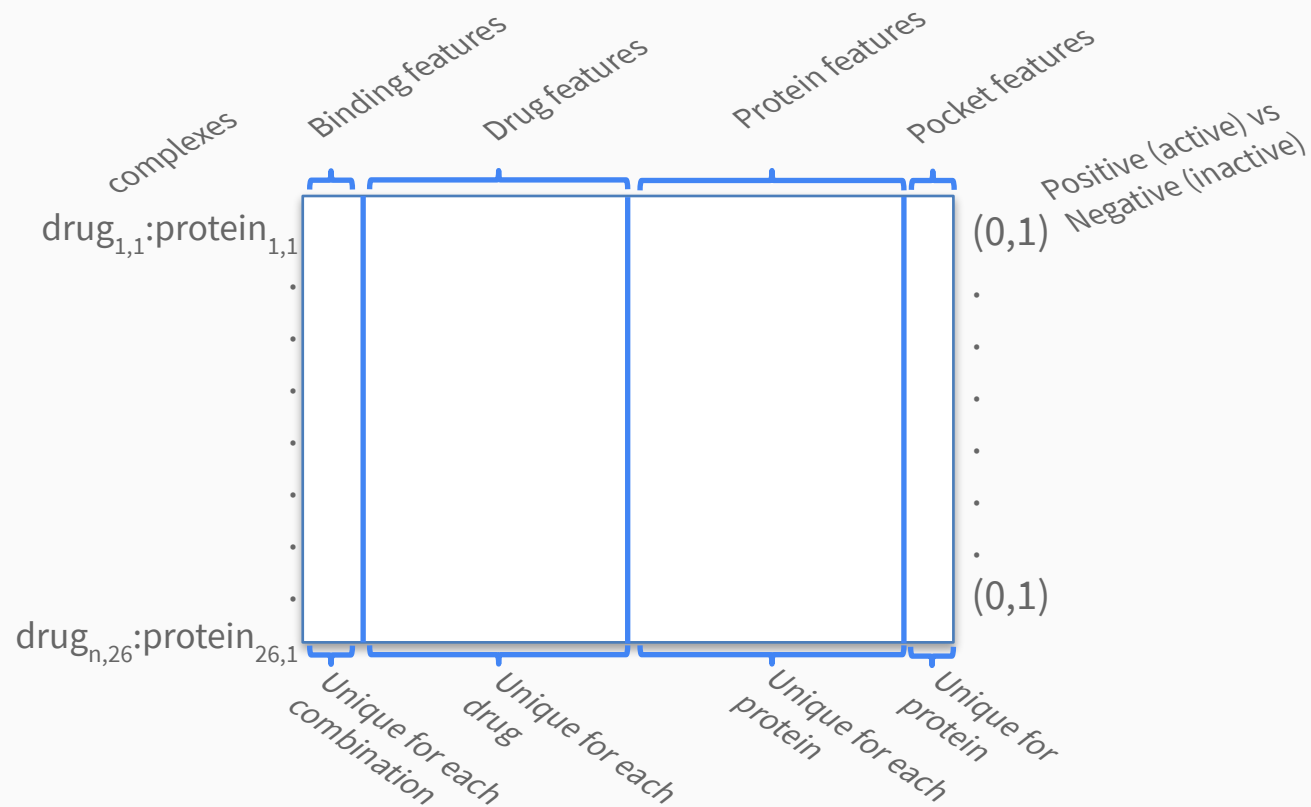
\*We let active binding compounds be called the positive class and the inactive binding compounds referred to as the negative class

# Methods: Data Collection

We derive our dataset of drug-protein pairs from the *Directory of Useful Decoys-Extended* (DUD-E) (Mysinger et. al)

- 1532 protein features derived from their sequences using a number of feature extraction servers
- 3850 drug features collected from the Dragon 7 molecular descriptor software
- 11 docking features collected from VinaMPI molecular docking software
- 38 binding pocket features using the prank software

# Dataset Description



# Methods: Data Preprocessing

- 189 features with some portion of missing values
- Eliminated 21 features with 23.1%-99.9% missing values, all drug features
- Remaining 167 features with  $\leq 5\%$  missing, impute use the column mean

# Methods: Data Preprocessing

- Full dataset after preprocessing:
  - 5,410 features
  - 361,786 examples
  - 1:50 ratio between positive/negative class labels
- Created training/testing sets using a stratified 80/20 split, keeping the proportion of positive class examples consistent across each split (~2%)

# Methods: Feature Selection and Classification

- We use the Random Forest (RF) as our classifier
- RFs are composed of an ensemble of shallow decision trees trained on subsets of the full features
- RFs can be robust to label imbalance and overfitting
- Can extract GINI impurities to quantify “feature importance”, allowing the ability to perform automatic feature selection

---

**Algorithm 1** Feature Selection

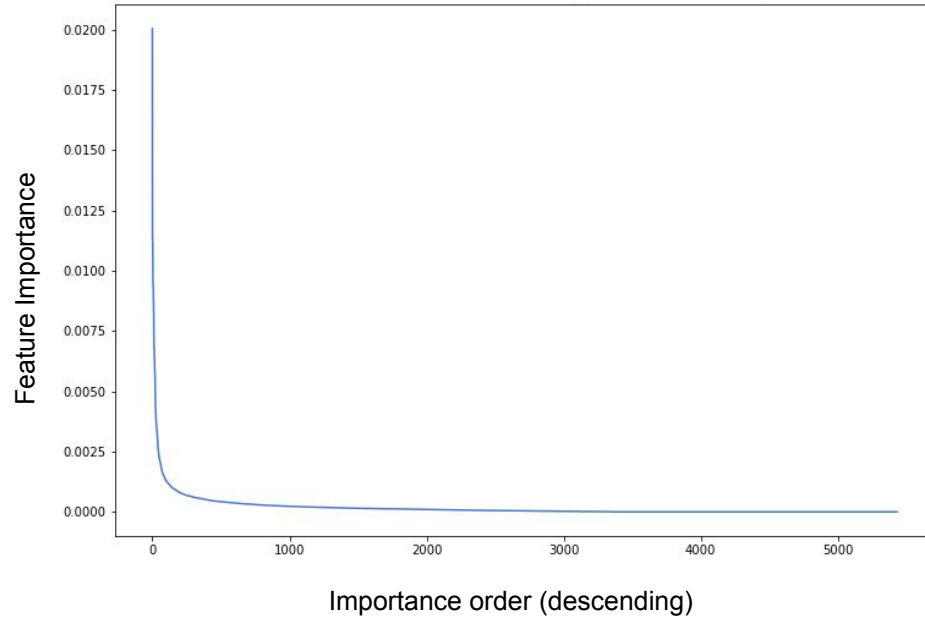
---

```
1: procedure SELECTION FOREST( $F, max\_steps$ )  
    $features\_to\_keep = F$   
2:   while  $features\_to\_keep \neq \emptyset$  and  $step < max\_steps$  do  
3:      $X, y = load\_data(features=features\_to\_keep)$   
4:      $X_{train}, X_{test}, y_{train}, y_{test} = train\_test\_split(X, y)$   
5:      $best\_forest = RandomizedGridSearch(RandomForest, X_{train}, y_{train}).best\_estimator$   
6:      $feature\_importances = best\_forest.importances$   
7:      $features\_to\_keep = feature\_importances > \frac{1}{|feature\_importances|}$   
8:      $step+ = 1$ 
```

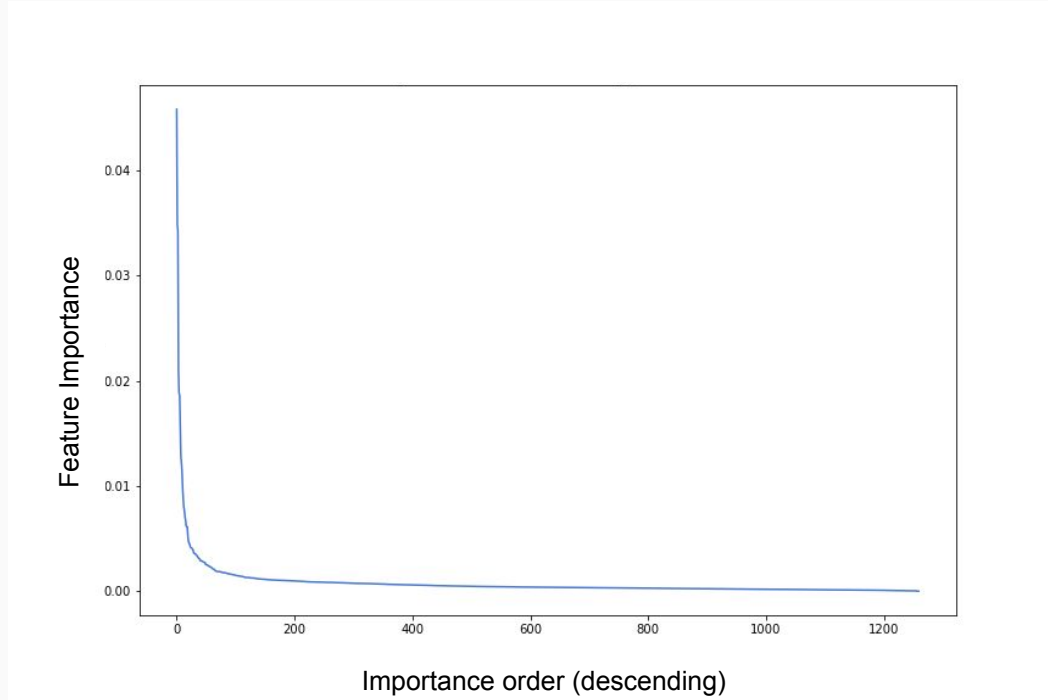
---



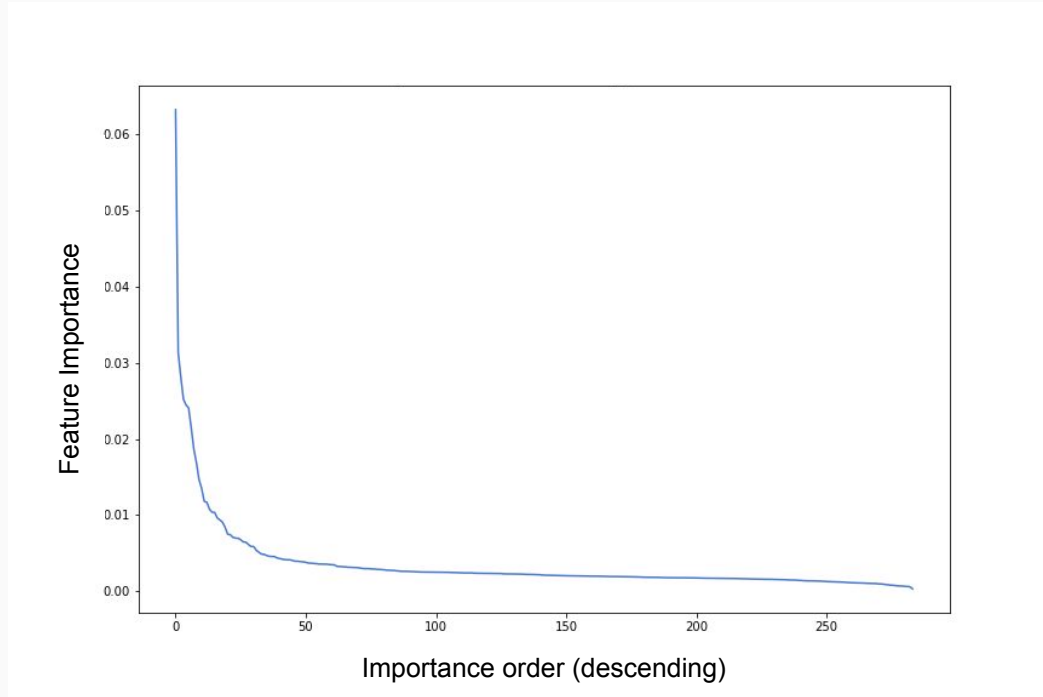
# Feature Selection



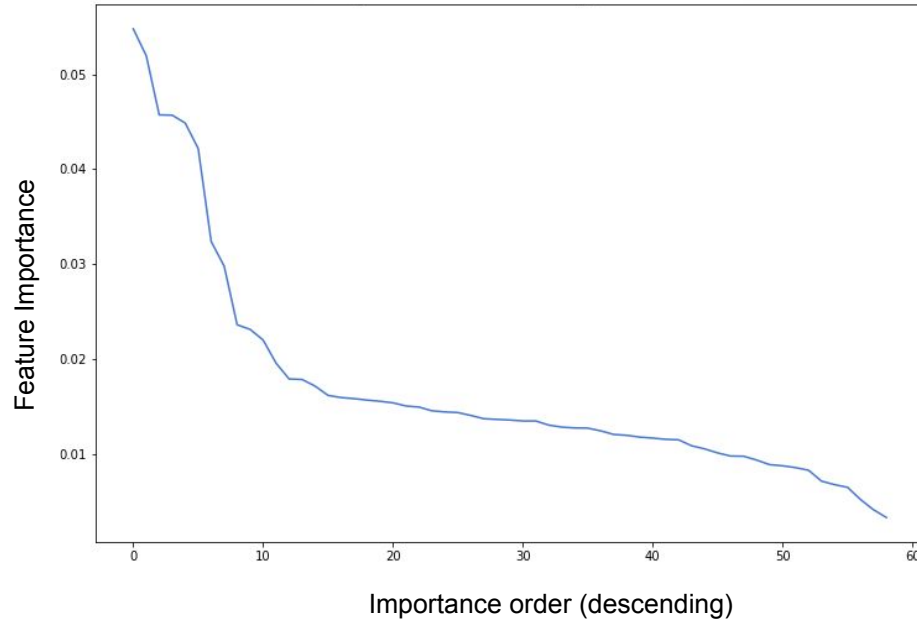
# Feature Selection



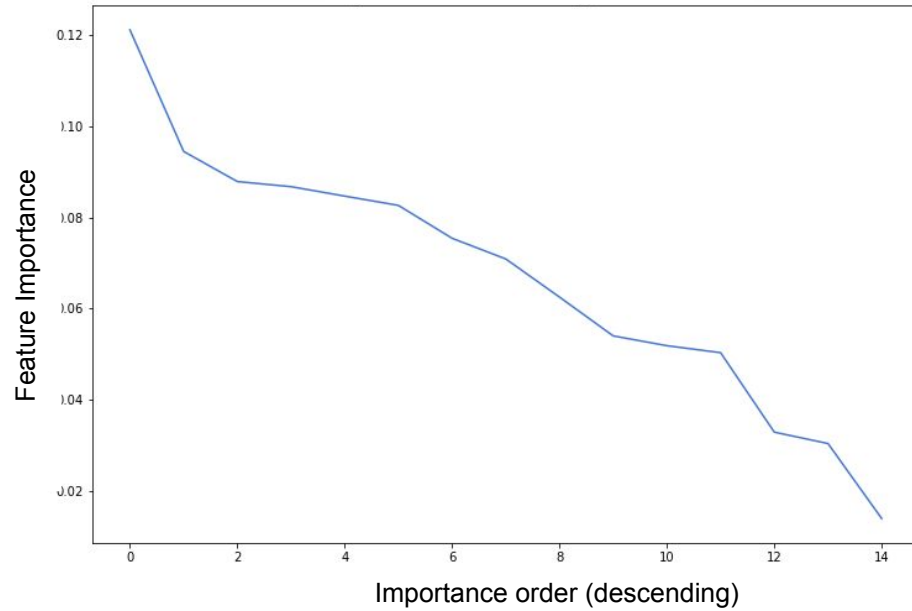
# Feature Selection



# Feature Selection



# Feature Selection



# Methods: Evaluation

- Feature Set 1 (FS1): selected using entire feature set
- Feature Set 2 (FS2): selected using only protein and binding pocket features
- Feature Set 3 (FS3): selected using only drug features
- Feature Set 4 (FS4): contains only docking features

<b>Model</b>	<b>Feature Set</b>	<b>Model</b>	<b>Feature Set</b>	<b>Model</b>	<b>Feature Set</b>
1	FS1	3	FS1 + FS3	5	FS1 + FS2 + FS3 + FS4
2	FS4	4	FS1 + FS3 + FS4	6	all features

# Evaluation Metrics

Name	Definition	Formula
Youden's index	Performance of dichotomous test. The value 1 indicates a perfect test and -1 indicates a useless test.	$\frac{TP}{TP+FN} + \frac{TN}{TN+FP} + 1$
F1	Harmonic mean of precision and recall	$\frac{2TP}{2TP+FP+FN}$
Precision	Positive predictive value	$\frac{TP}{TP+FP}$
Recall	True positive rate	$\frac{TP}{TP+FN}$

# Classification Performance

Model	Class	Precision	Recall	F1-Score	Class	Precision	Recall	F1-Score
1	0	1.00	1.00	1.00	1	0.91	0.81	0.86
2	0	0.98	0.99	0.99	1	0.4	0.28	0.33
3	0	1.00	1.00	1.00	1	0.83	0.91	0.87
4	0	0.99	1.00	1.00	1	0.97	0.80	0.88
5	0	1.00	1.00	1.00	1	0.83	0.92	0.87
6	0	1.00	1.00	1.00	1	0.87	0.90	0.89
Docking	0	0.98	0.59	0.73	1	0.04	0.64	0.07



Model	Feature Set	Model	Feature Set	Model	Feature Set
1	FS1	3	FS1 + FS3	5	FS1 + FS2 + FS3 + FS4
2	FS4	4	FS1 + FS3 + FS4	6	all features



# Conclusions

- We found that the features selected from our method greatly increased the performance of binding predictions
- Our methods achieve the goal of robustness to class imbalance and are able to extract the important features for the task in a data-driven way
- We observe near 60% increase in the success rate for discovering active compounds over molecular docking

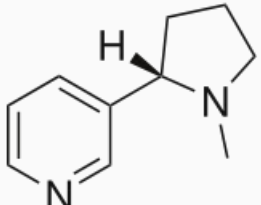
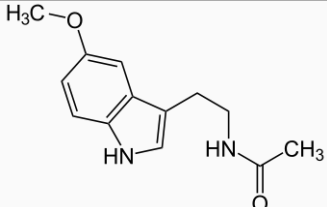
# Distributed Learning of Molecular Feature Representations

---

# Motivation

- Common for modern screening pipeline methods in computational drug discovery to require explicit feature extraction
- These features are gathered independently of the end task
- Process that still requires some degree of domain expertise

# Representing Molecular Structures

Molecule	Structure	SMILES
Nicotine		<chem>CN1CCC[C@H]1c2cccnc2</chem>
Melatonin		<chem>CC(=O)NCCC1=CNc2c1cc(OC)cc2</chem> <chem>CC(=O)NCCc1c[nH]c2ccc(OC)cc12</chem>

wikipedia:[https://en.wikipedia.org/wiki/Simplified\\_molecular-input\\_line-entry\\_system](https://en.wikipedia.org/wiki/Simplified_molecular-input_line-entry_system)

# Related Work: Molecular Descriptors

- Output of some computational process “which transforms chemical information encoded within a symbolic representation of a molecule into a useful number...”
- Examples may include a measure of a molecules solubility, the number of a certain type of bond, drug-like index, etc;

# Related Work: Molecular Fingerprinting

- Compute a unique binary valued vector that identifies a molecule based upon the atoms that it contains and their features
- Often used for computing similarity between molecules

---

## Algorithm 1 Circular fingerprints

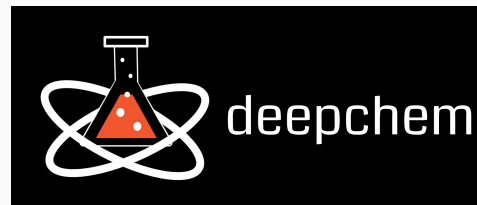
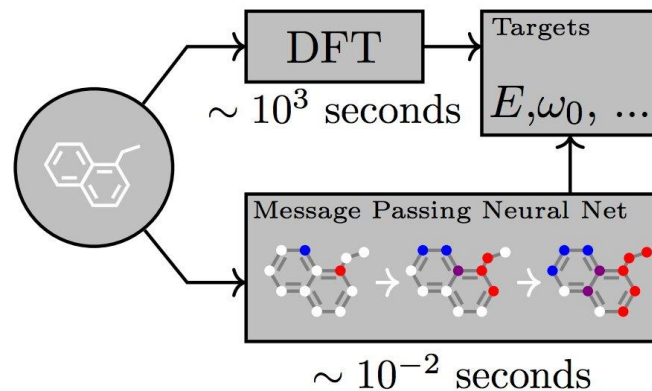
---

```
1: Input: molecule, radius  $R$ , fingerprint length  $S$ 
2: Initialize: fingerprint vector  $\mathbf{f} \leftarrow \mathbf{0}_S$ 
3: for each atom  $a$  in molecule
4:    $\mathbf{r}_a \leftarrow g(a)$   $\triangleright$  lookup atom features
5: for  $L = 1$  to  $R$   $\triangleright$  for each layer
6:   for each atom  $a$  in molecule
7:      $\mathbf{r}_1 \dots \mathbf{r}_N = \text{neighbors}(a)$ 
8:      $\mathbf{v} \leftarrow [\mathbf{r}_a, \mathbf{r}_1, \dots, \mathbf{r}_N]$   $\triangleright$  concatenate
9:      $\mathbf{r}_a \leftarrow \text{hash}(\mathbf{v})$   $\triangleright$  hash function
10:     $i \leftarrow \text{mod}(r_a, S)$   $\triangleright$  convert to index
11:     $\mathbf{f}_i \leftarrow 1$   $\triangleright$  Write 1 at index
12: Return: binary vector  $\mathbf{f}$ 
```

---

# Related Work: Neural Fingerprinting and Molecular Graph Convolutions

- Duvenaud et. al. propose a fingerprinting method that uses a differentiable function
- Variations on this idea have been proposed and provided via deepchem
- Gilmer et. al. present a comprehensive survey of current techniques under the name “message passing neural networks” or MPNNs



# Dataset Description

- Collect the drug molecules and their targets from DUD-E
- Training/Testing sets generated using an 80/20 stratified split
- Training/Validation sets generated using a 90/10 stratified split
- Dragon 7 software to extract molecular properties

	smiles	molecular properties	active/decoy
drug <sub>1,1</sub> :protein <sub>1,1</sub>			(0,1)
.			.
.			.
.			.
.			.
.			.
.			.
drug <sub>n,26</sub> :protein <sub>26,1</sub>			(0,1)

partition	# examples	% active
Train	259,952	2.495
Validation	28,884	2.662
Test	72,209	2.504



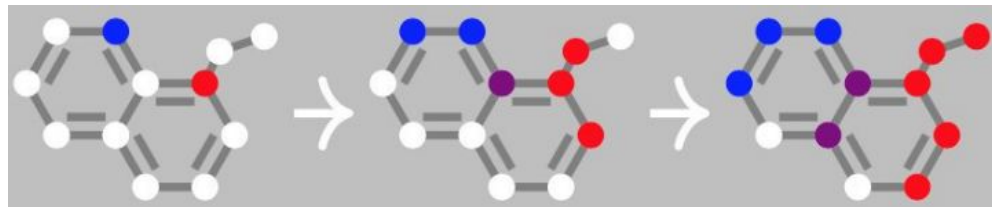
# Network Description

- Message Passing Phase:

- $m_v^{t+1} = \sum_{w \in N(v)} M_t(h_v^t, h_w^t, e_{vw})$
- $h_v^{t+1} = U_t(h_v^t, m_v^{t+1})$

- Readout Phase:

- $\hat{y} = R(\{h_v^T \mid v \in G\})$



We define:

- $M_t(h_v^t, h_w^t, e_{vw}) = h_v^t \parallel h_w^t \parallel e_{vw}$
- $U_t(h_v^t, m_v^{t+1}) = \text{ReLU}(m_v^{t+1})$
- $\hat{y}_{reg} = O(R(\{h_v^T \mid v \in G\}))$
- $\hat{y}_{class} = \text{softmax}(O(R(\{h_v^T \mid v \in G\})))$

# Network Training

- Practical issues with the MPNN motivated the need to think of ways to improve the speed of training
- We use the HOGWILD! Asynchronous SGD algorithm as a potential solution to this problem

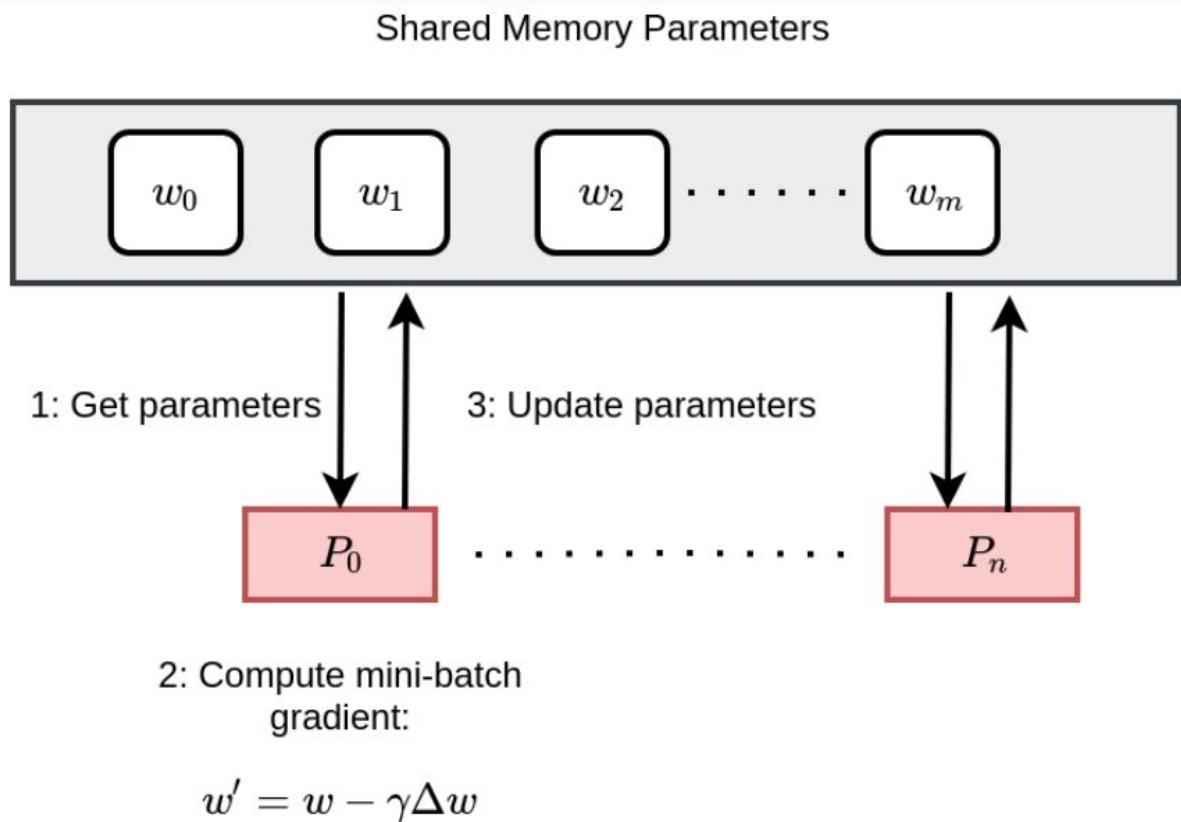
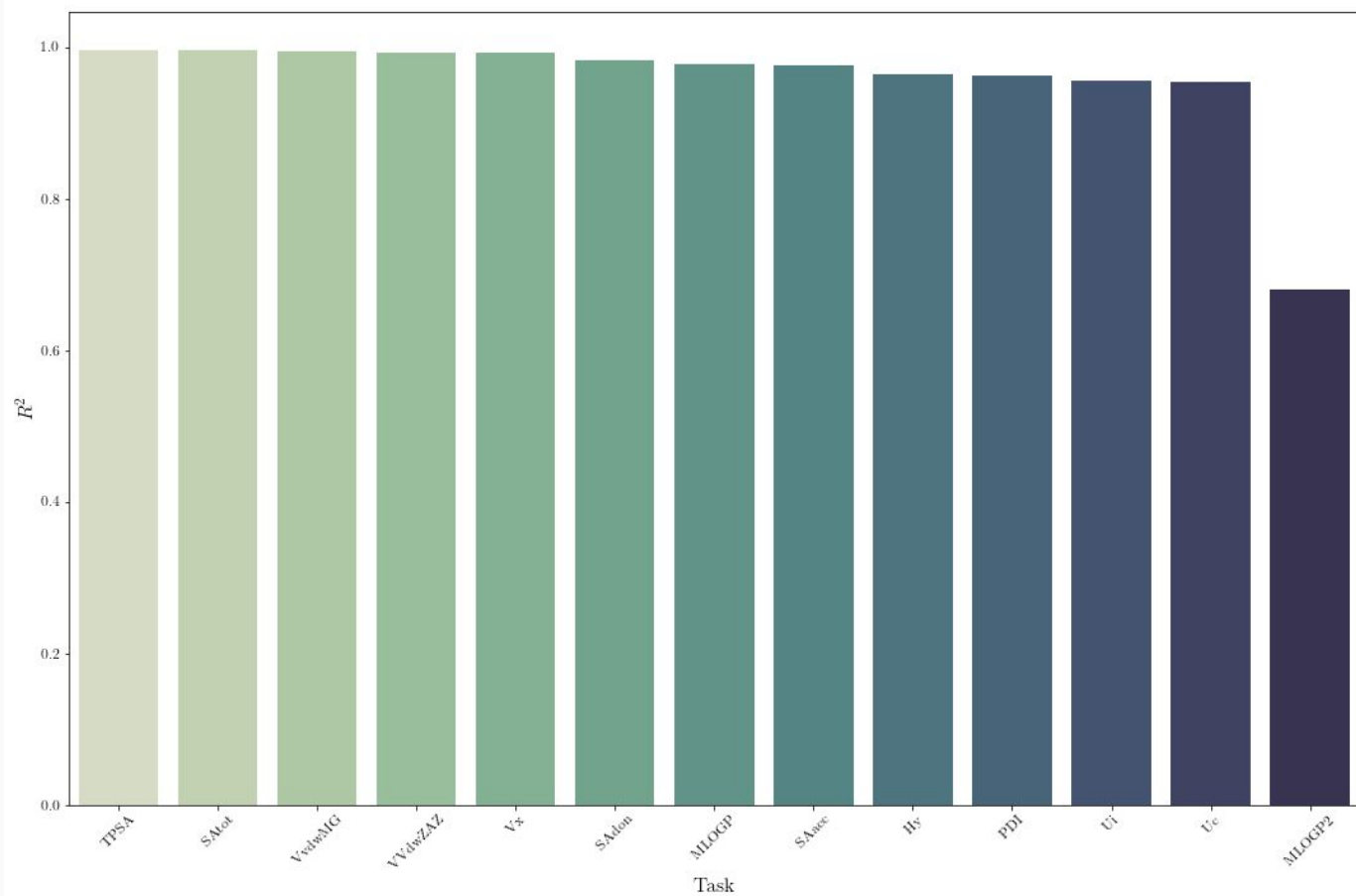


Figure 3.1: Visualization of the Hogwild! training algorithm

# Results: Experiment 1 - Predicting Molecular Properties

- In our first set of experiments, we are interested in the MPNNs ability to learn meaningful feature representations
- We use the set of molecular properties from the Dragon 7 software as our targets
- Idea: If this can be done, perhaps more complicated tasks are possible
- Minimize the MSE using the ADAM optimizer with learning rate =  $1e-3$



# Results: Experiment 2 - Binding Affinity Classification

- Goal 1: can we achieve comparable performance to our previous methods using only the drug molecular structures as our input?
- Goal 2: can we use a distributed training algorithm to reduce training time?
- Evaluated 100 settings of hyperparameters to find best set, then vary the number of training processes
- Minimize the cross entropy loss using ADAM optimizer with HOGWILD! training

Old Method

Model	Class	Precision	Recall	F1-Score	Class	Precision	Recall	F1-Score
1	0	1.00	1.00	1.00	1	0.83	<b>0.92</b>	0.87
2	0	0.98	0.98	0.98	1	0.26	0.30	0.28
3	0	1.00	1.00	1.00	1	0.83	<b>0.92</b>	0.87
4	0	1.00	1.00	1.00	1	0.83	0.91	0.87
5	0	1.00	1.00	1.00	1	0.84	<b>0.92</b>	<b>0.88</b>
6	0	1.00	1.00	1.00	1	<b>0.85</b>	0.89	0.87
Docking	0	0.99	0.58	0.73	1	0.04	0.67	0.07

Table 1: Random Forest Binding Classification Results

New Method

$n$ processors	Class	Prec.	Recall	F1-score	Class	Prec.	Recall	F1
1	0	0.99	1.00	1.00	1	0.89	0.78	0.83
2	0	1.00	1.00	1.00	1	0.91	0.80	0.85
3	0	1.00	1.00	1.00	1	0.92	0.82	0.87
4	0	1.00	1.00	1.00	1	0.91	0.83	0.87
5	0	1.00	1.00	1.00	1	<b>0.93</b>	0.79	0.85
6	0	1.00	1.00	1.00	1	0.87	<b>0.87</b>	0.87
7	0	1.00	1.00	1.00	1	0.90	0.86	<b>0.88</b>
8	0	1.00	1.00	1.00	1	0.89	0.84	0.87
9	0	1.00	1.00	1.00	1	0.87	<b>0.87</b>	0.87
10	0	1.00	1.00	1.00	1	<b>0.93</b>	0.80	0.85
11	0	1.00	1.00	1.00	1	0.88	<b>0.87</b>	0.87

Table 2: MPNN Binding Classification Results

---

# Conclusions

- Each of the MPNN models exceeds the maximum precision score of the Random Forest method
- We can match the best F1-score with the previous method
- By using only the molecular structures of the drug molecules, we have demonstrated that we can learn optimized features that drastically reduce the computational overhead and size of the dataset used for making this prediction



---

# Future Directions

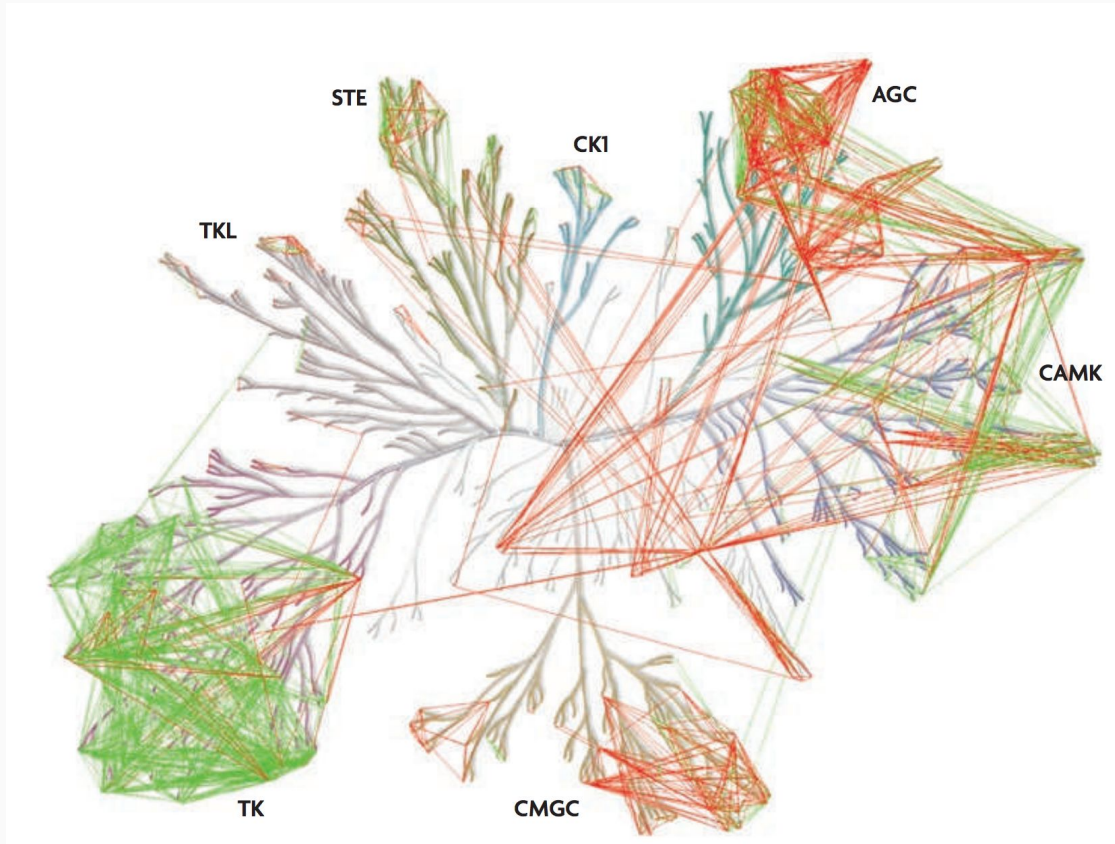
- Visualize the learned features
- HOGWILD! can work but is it the best approach to scaling the MPNN?
- Can we find more clever ways to speed up the training? Architecture or Optimization?
- How about using synchronous distributed optimization algorithms?
- What about multi-task learning with unscaled targets?

# Acknowledgements

- UK Markey Cancer Center
- UK CCS
- National Energy Research Scientific Computing Center (NERSC)
- National Institutes of Health (NIH)

# Questions?

# Selectivity in Kinases



- Pairs of kinases that were potentially inhibited by a common inhibitor (green lines) were used to determine a sequence similarity cut-off to predict pairs of kinases that can be inhibited by a common inhibitor (red lines)