Analysis of compounds activity concept learned by SVM using robust Jaccard based low-dimensional embedding

Do we really have 90% accuracy in drug compounds virtual screening?

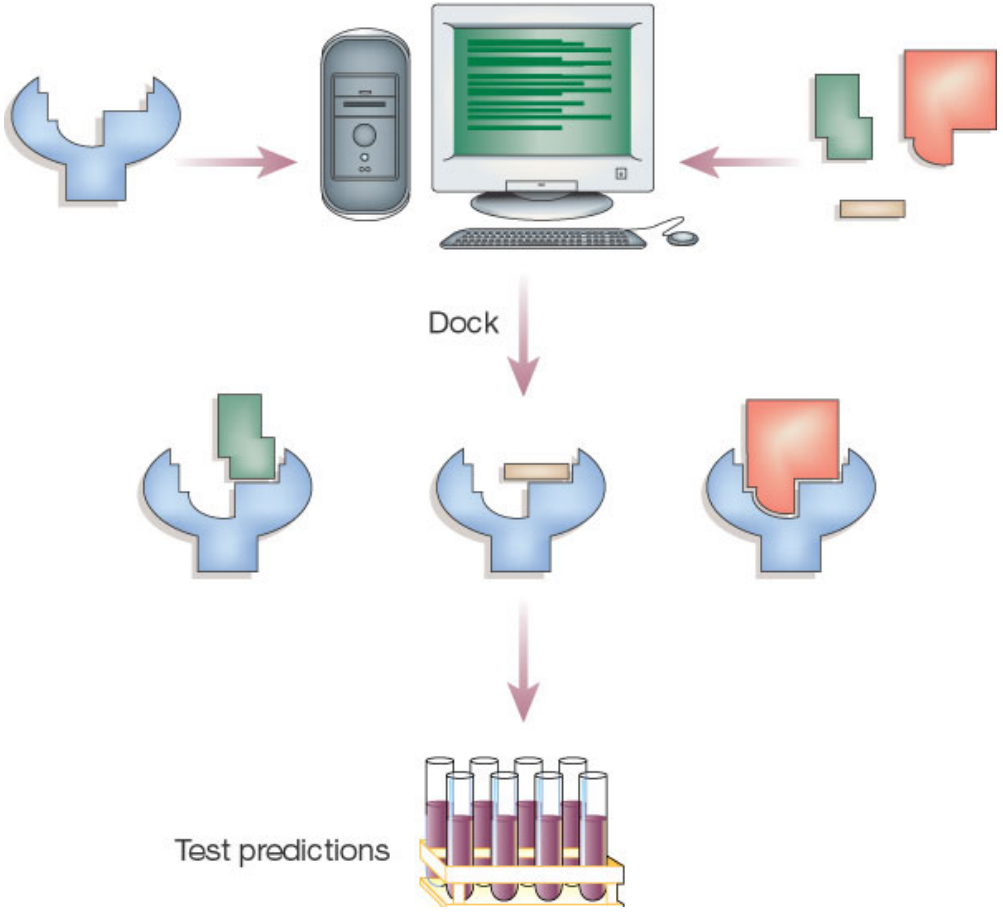
Stanisław Jastrzębski, Wojciech Marian Czarnecki

Presentation plan

- Virtual screening problem
- Data characteristics and hypothesis
- Model description
- Results

Virtual screening

Virtual screening (VS) is a technique used in drug discovery to search for small molecules that are likely to bind to a drug target.
If we know both active and inactive compounds, then commonly used technique is machine learning, especially Support Vector Machine (SVM).

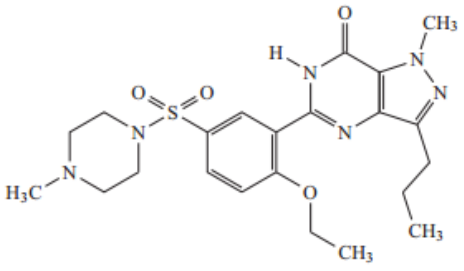


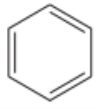
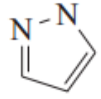
Problem description

- Binary classification where we want to predict if compound will bind to a receptor (target)
- Our goal is to predict new active compounds to aid choosing candidates for real tests (not in computer). There is a need to learn some complex patterns present in compounds that are active to match good candidates (we want to discover new drugs that are preferable cheaper or better)

Fingerprint representation

- Each compound is represented as a real vector (fingerprint) in which each position represents for instance certain chemical structure present within the compound.



keys				|N|≥ 3	…	
fingerprint	1	0	1	1	…	1

- For instance MACCS fingerprint contains one if there is a ring of size 4 in the molecule
- In our paper we used 8 different fingerprints for statistical robustness

How are the datasets created?

- The data is collected much differently than usually in Machine Learning as we have to manually test each compound

How are the datasets created?

- The data is collected much differently than usually in Machine Learning as we have to manually test each compound
- Active compounds are enormously rare but negative results are rarely published. **The dataset does not reflect underlying distribution**.

How are the datasets created?

- The data is collected much differently than usually in Machine Learning as we have to manually test each compound
- Active compounds are enormously rare but negative results are rarely published. **The dataset does not reflect underlying distribution**.
- The compounds in the dataset are similar It usually doesn't make much sense to test completely novel compound. It is hard to learn something complex if we have such a datset
- Common practice is to include artifically generated compounds (*Directory of Useful Decoys*)

Research question

Hugely popular model in this area is SVM with RBF kernel. The nontypical dataset creation method and reported high accuracy results inclined us to pose following question:

> *Is Support Vector Machine with RBF kernel learning any complex data patterns exploiting compound activity or does it degenerate to nearly nearest neighbour search?*

Research question

Hugely popular model in this area is SVM with RBF kernel. The nontypical dataset creation method and reported high accuracy results inclined us to pose following question:

> *Is Support Vector Machine with RBF kernel learning any complex data patterns exploiting compound activity or does it degenerate to nearly nearest neighbour search?*

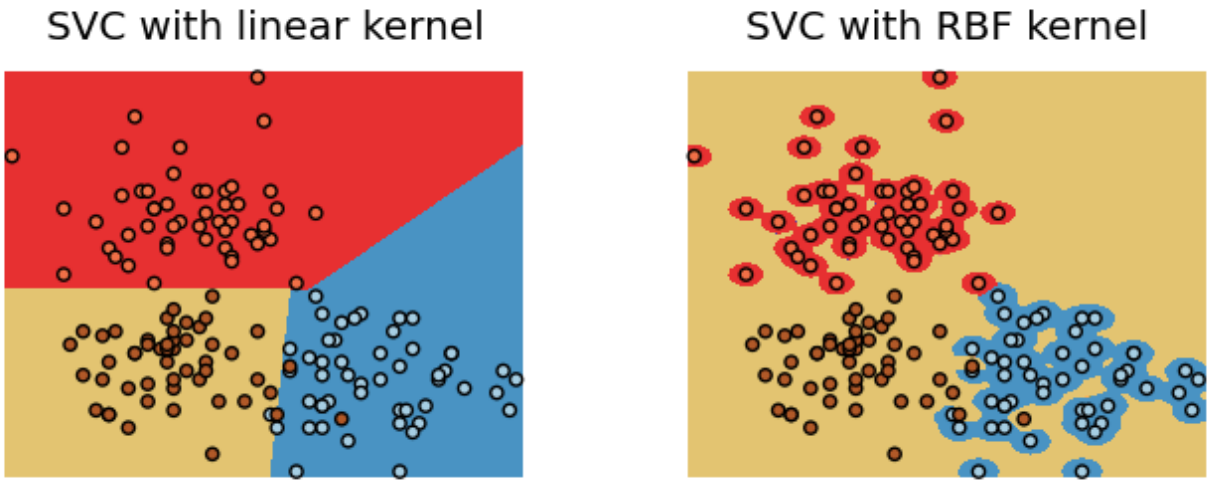
This would be a strong indicator that we might have to look differently at this problem and use different methodology.

Few remarks

- Nearest neighbour search is not a bad model obviously (even optimal given infinite data), but it is not useful in our case

Few remarks

- Nearest neighbour search is not a bad model obviously (even optimal given infinite data), but it is not useful in our case
- SVM RBF can degenerate to nearest neighbour search if it has a large number of memorized support vectors: $f(x') = \sum_i y_i \alpha_i K(x_i, x') + b$, where $K(x, x')$ is RBF kernel.
- If SVM is forced to memorize all the training examples to encode target it means there is no much regularity in the data.



Methodology

- It is not trivial to test whether given model is learning anything complex.

Methodology

- It is not trivial to test whether given model is learning anything complex.
- We decided to test if model using only local knowledge can be comparable in terms of (weighted) accuracy

Methodology

- It is not trivial to test whether given model is learning anything complex.
- We decided to test if model using only local knowledge can be comparable in terms of (weighted) accuracy
- Of course it is not enough, so we investigated that further raising additional arguments in favor of this hypothesis

Jaccard Distance

We are constructing local embedding so we need a **metric**. Our choice is Jaccard similarity measure

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|},$$

which has many interesting properties and is very suitable for fingerprint comparison.

Local embedding

We construct 8 dimensional fingerprint representation. This is a huge dimensionality reduction (fingerprints have on average 1000 dimensions).

Definition

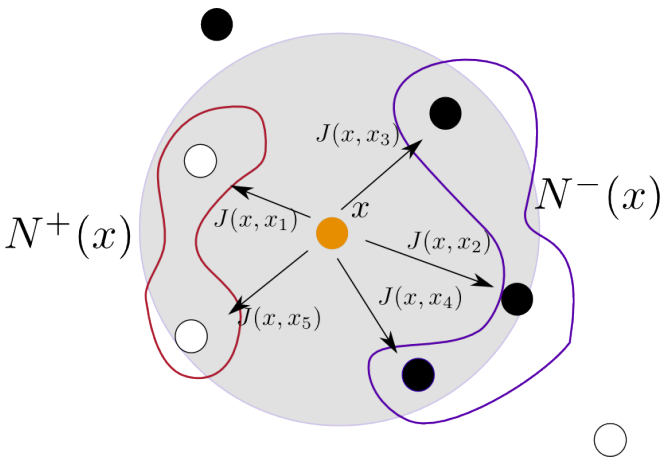
For a given dataset and arbitrary similarity measure S we define a Local Statistics Embedding (LSE) as

$$\tau_k(x) = [\quad \begin{matrix} |N^-(x)| \\ \min S(N^-(x)) \end{matrix} \quad \begin{matrix} |N^+(x)| \\ \min S(N^+(x)) \end{matrix} \quad \begin{matrix} \text{mean } S(N^-(x)) \\ \max S(N^-(x)) \end{matrix} \quad \begin{matrix} \text{mean } S(N^+(x)) \\ \max S(N^+(x)) \end{matrix} \quad]',$$

where $N^l(x)$ is a sequence of samples with label l of the k nearest neighbours of x in terms of S.

Local embedding

Example embedding of a point with $k = 5$ and using some similarity denoted as J, positive samples are white and negative are black.



$$\tau_5(x) = \begin{bmatrix} 3 \\ 2 \\ \frac{J(x,x_1)+J(x,x_2)+J(x,x_4)}{3} \\ \frac{J(x,x_3)+J(x,x_5)}{2} \\ \min\{J(x,x_3), J(x,x_5)\} \\ \min\{J(x,x_1), J(x,x_2), J(x,x_4)\} \\ \max\{J(x,x_3), J(x,x_5)\} \\ \max\{J(x,x_1), J(x,x_2), J(x,x_4)\} \end{bmatrix}$$

Model

- As our model we used composition of local embedding and linear SVM (similar results obtained using logisitc regression as well)

Model

- As our model we used composition of local embedding and linear SVM (similar results obtained using logisitc regression as well)
- For efficiency we applied approximate neighbour search called Local Sensitive Hashing (without it the complexity is quadratic as we have to check all pairs to construct the embedding)

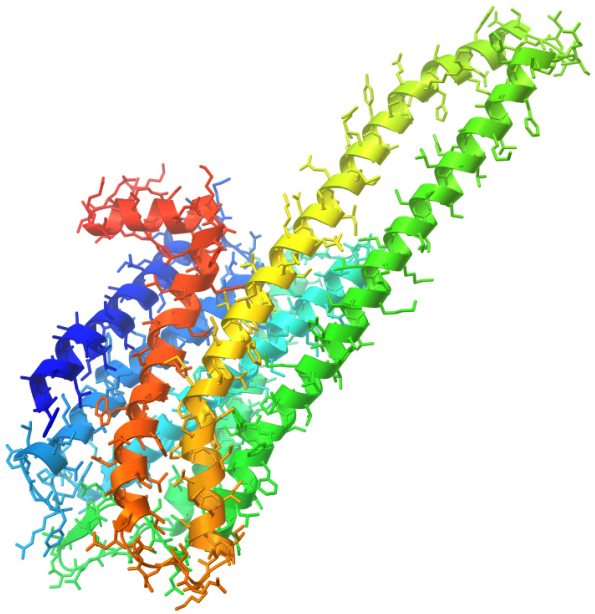
Local Sensitive Hashing

Local Sensitive Hashing is a technique for nearest neighbour search.

- General idea is to find a small list of candidate pairs for nearest neighbours
- We have a family H of hashing functions
- $d(x, y)$ is a metric
- If $d(x, y) \leq e$ with high probability $h(x) = h(y)$
- We hash objects to multiple buckets and then scan for neighbours only from selected buckets
- For our application we had to construct a chain of LSH with different thresholds

Datasets used in our work

- 10 receptors (targets) were tested
- For each receptor compounds were represented using 8 different fingerprints
- 80 separate datasets in total (receptor + representation pair)



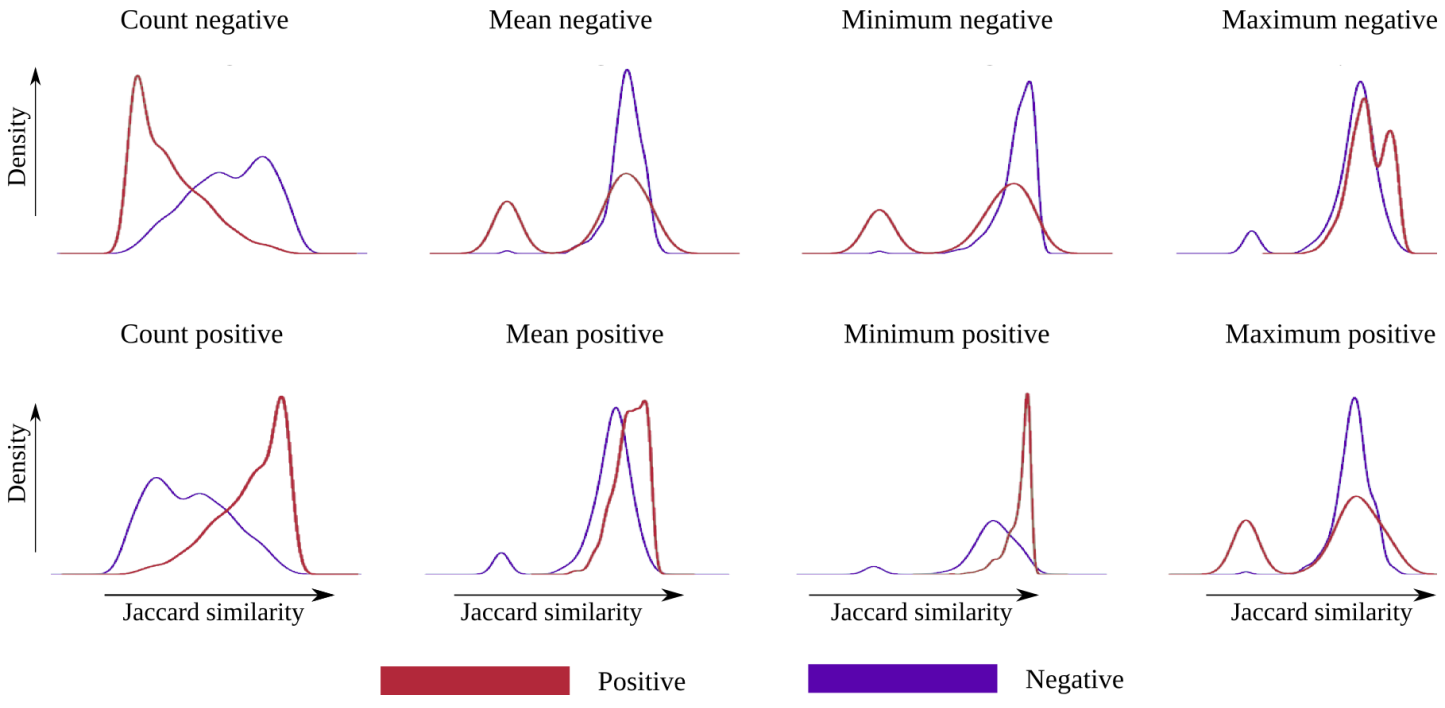
Models tested

- SVM with RBF kernel (SVM RBF)
- SVM with RBF Nystroem kernel approximation
- SVM with Jaccard kernel (SVM Jaccard)
- kNN
- Local Statistics Embedding + SVM (LSE + SVM)
- Local Statistics Embedding + Logistic Regression (LSE + LR)

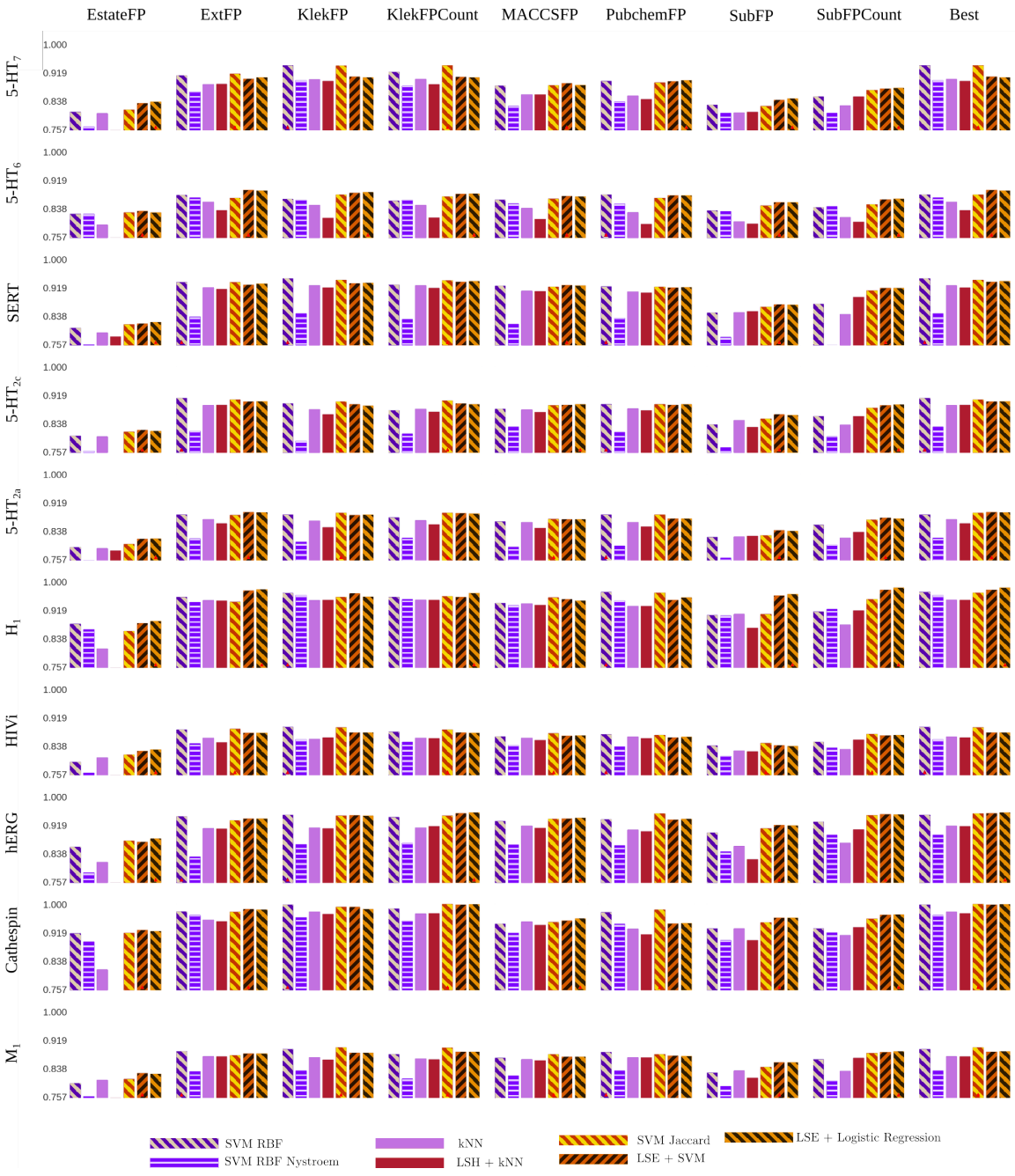
If our hypothesis is true our models (the last two) should be comparable with SVM RBF.

Feature discrimination for LSE

Despite its simplicity Local Statistics Embedding provides a good discrimination.



Results


EstateFP
ExtFP
KlekFP
KlekFPCount
MACCSFP
PubchemFP
SubFP
SubFPCount
Best
5-HT7
5-HT6
SERT
5-HT2c
5-HT2a
H1
HIVi
hERG
Cathespin
M1
1.000
0.919
0.838
0.757
SVM RBF
SVM RBF Nystroem
kNN
LSH + kNN
SVM Jaccard
LSE + SVM
LSE + Logistic Regression

Results

- Our model using *only local information* is better than SVM RBF in 80% of cases

Results

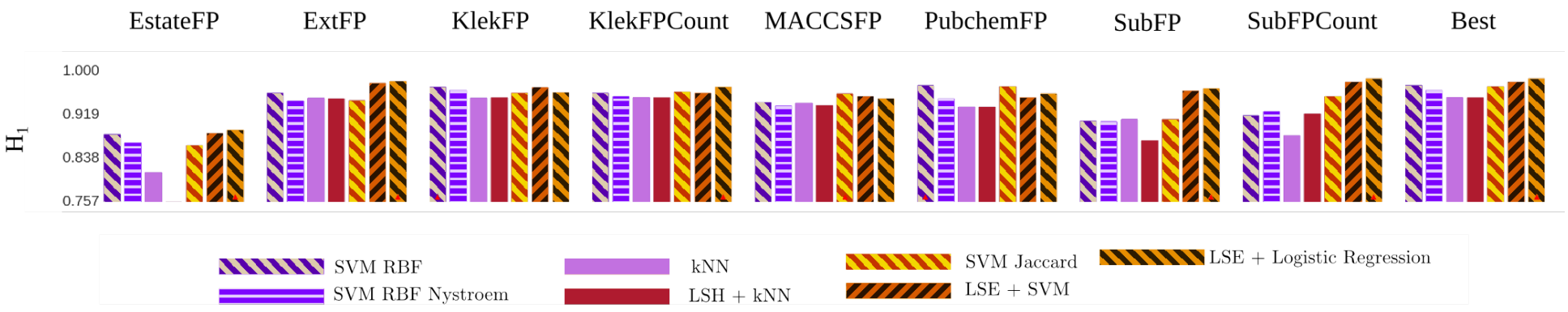
- Our model using *only local information* is better than SVM RBF in 80% of cases
- SVM RBF is memorizing on average 90% of training cases that makes prediction very expensive (similarly other SVMs tested)

Results

- Our model using *only local information* is better than SVM RBF in 80% of cases
- SVM RBF is memorizing on average 90% of training cases that makes prediction very expensive (similarly other SVMs tested)
- SVM RBF and our models have highly correlated outputs

Results

- Our model using *only local information* is better than SVM RBF in 80% of cases
- SVM RBF is memorizing on average 90% of training cases that makes prediction very expensive (similarly other SVMs tested)
- SVM RBF and our models have highly correlated outputs
- We performed several tests, for instance we looked at SVM RBF performance on incorrectly classified examples by our model.



Performance for target H_1

Training time and prediction time

Model	Model parameters	Training time [h]	Testing time [h]
SVM RBF	$\sim 2000 \cdot d$	233.27	21.8
SVM RBF Nystroem	$\sim 2000 \cdot h$	67.10	2.5
SVM Jaccard	$\sim 1000 \cdot d$	11.69	0.4
LSE + SVM	$\sim 100 \cdot 8$	27.60	0.0
LSE + LR	$\sim 100 \cdot 8$	16.1	0.0

Model complexities as measured by the number of parameters used during classification of the new point. d is fingerprint size and h is Nystroem feature space size (in our experiments set to 100).

Summary

- We have proven the hypothesis that SVM with RBF Kernel degenerates to nearly neighbour search on this dataset. So we are using fairly complex model which degenerates to trivial nearest neighbour search.

Summary

- We have proven the hypothesis that SVM with RBF Kernel degenerates to nearly neighbour search on this dataset. So we are using fairly complex model which degenerates to trivial nearest neighbour search.
- The most probable reason for this is a strong violation of the i.i.d. assumption during dataset generation. The dataset is not reflecting the underlying true dataset distribution.

Summary

- We have proven the hypothesis that SVM with RBF Kernel degenerates to nearly neighbour search on this dataset. So we are using fairly complex model which degenerates to trivial nearest neighbour search.
- The most probable reason for this is a strong violation of the i.i.d. assumption during dataset generation. The dataset is not reflecting the underlying true dataset distribution.
- We have shown nearly equivalent model in terms of both achieved results and represented knowledge

Summary

- We have proven the hypothesis that SVM with RBF Kernel degenerates to nearly neighbour search on this dataset. So we are using fairly complex model which degenerates to trivial nearest neighbour search.
- The most probable reason for this is a strong violation of the i.i.d. assumption during dataset generation. The dataset is not reflecting the underlying true dataset distribution.
- We have shown nearly equivalent model in terms of both achieved results and represented knowledge
- As an additional result we proposed a fast linear classifier allowing for fast online training

Summary

- We have proven the hypothesis that SVM with RBF Kernel degenerates to nearly neighbour search on this dataset. So we are using fairly complex model which degenerates to trivial nearest neighbour search.
- The most probable reason for this is a strong violation of the i.i.d. assumption during dataset generation. The dataset is not reflecting the underlying true dataset distribution.
- We have shown nearly equivalent model in terms of both achieved results and represented knowledge
- As an additional result we proposed a fast linear classifier allowing for fast online training
- This research suggests serious flaws in many virtual screening methods

Future directions

- Trying wider range of datasets and methods (including Database of Useful Decoys)
- Suggesting new methods of measuring model quality
- Using linear model in virtual screening leveraging online training capabilities