Published in final edited form as:
J Comput Aided Mol Des. 2016 February ; 30(2): 177–189. doi:10.1007/s10822-016-9895-2.

Improving Quantitative Structure-Activity Relationship Models using Artificial Neural Networks Trained with Dropout

Jeffrey Mendenhall and **Jens Meiler**[*]
Department of Chemistry, Center for Structural Biology, Institute of Chemical Biology, Vanderbilt University, Nashville, TN 37235

Abstract

Dropout is an Artificial Neural Network (ANN) training technique that has been shown to improve ANN performance across canonical machine learning (ML) datasets. Quantitative Structure Activity Relationship (QSAR) datasets used to relate chemical structure to biological activity in Ligand-Based Computer-Aided Drug Discovery (LB-CADD) pose unique challenges for ML techniques, such as heavily biased dataset composition, and relatively large number of descriptors relative to the number of actives. To test the hypothesis that dropout also improves QSAR ANNs, we conduct a benchmark on nine large QSAR datasets. Use of dropout improved both Enrichment false positive rate (FPR) and log-scaled area under the receiver-operating characteristic curve (logAUC) by 22–46% over conventional ANN implementations. Optimal dropout rates are found to be a function of the signal-to-noise ratio of the descriptor set, and relatively independent of the dataset. Dropout ANNs with 2D and 3D autocorrelation descriptors outperform conventional ANNs as well as optimized fingerprint similarity search methods.

Keywords

Artificial Neural Network (ANN); dropout; Quantitative Structure Activity Relationship (QSAR); BioChemicalLibrary (BCL); Machine Learning (ML); Ligand-Based Computer-Aided Drug Discovery (LB-CADD)

Background and Significance

Quantitative Structure Activity Relationship (QSAR) models are an established means of Ligand-Based Computer-Aided Drug Discovery (LB-CADD), i.e. finding novel compounds that bind to a particular protein target, given a dataset of known binders and non-binders [1]. Physicochemical properties are encoded using spatial and topological representations (descriptors) of the local atomic environments within the molecule. To model the non-linear relation between chemical structure and biological activity for a particular protein target, a machine learning method, such as an ANN, is trained to predict binding or activity at a particular protein target.

[*]To whom correspondence should be addressed: Jens Meiler, Ph.D., Associate Professor, Vanderbilt University, Department of Chemistry, 7330 Stevenson Center, Station B 351822, Nashville, TN 37235, jens@meilerlab.org, Phone: +1 (615) 936-5662, Fax: +1 (615) 936-2211.

Neuronal dropout [2] has been repeatedly demonstrated to improve ANN performance on classification tasks including speech recognition and handwritten digit classification [3,4]. With the dropout training technique, a fraction of neurons is effectively silenced (set to zero) during each training epoch. Dropout is known to improve the generalizability of ANNs by preventing co-adaptation of hidden-layer neurons [2]. With this training feature, each hidden neuron must build an independent mapping from feature space onto output space. Thereby, dropout in the hidden layer helps to prevent ANNs from memorizing the input data (overtraining). In previous QSAR ANN models, overtraining effects have been mitigated by model selection – selecting the ANN from those generated during training that performed best on a monitoring dataset, or early-termination – stopping training when overtraining is evident against a separate monitoring dataset [1,5–10]. Dropout is thought to produce better generalizing ANNs that circumvent the necessity for model selection [2].

Dropout is often employed in large ANNs with several hidden layers, known as “deep” ANNs. The effects of dropout have not been investigated extensively in ANNs with a single hidden layer, and when using heavily class-biased datasets and inhomogenous descriptor sets (descriptors with unrelated units) that are commonplace in QSAR modeling. Dropout has been used in previous QSAR modeling in the context of large multi-task QSAR setting that is uncommon in practice [11]. Dropout was used by the winning entry in a Merck-sponsored QSAR competition. However, it remains unclear how much dropout contributed to this success, and whether the results will extend to other targets [12]. In the present work, using the BioChemicalLibrary (BCL) [6], we explore whether the success of dropout extends to single-layer, single-target, QSAR models in LB-CADD. We systematically optimize the fraction of neurons dropped in the hidden (D_{hid}) and input (D_{inp}) layers prior to each forward-propagation pass starting from typical values D_{hid} = 50% and D_{inp} = 0% [2].

Methods

Dataset Preparation

To mitigate ligand biases and other dataset-dependent effects, we employ an established QSAR benchmark comprised of nine diverse protein targets. The datasets each contain at least 100 confirmed active molecules and more than 60,000 inactive molecules [6]. The datasets were re-curated to eliminate a few dimers and higher-order molecular complexes that had previously been included in the virtual screening, and to add molecules that were previously excluded due to difficulties in calculating descriptors. Structural duplicates and duplicates created during the process of curation (e.g. due to desalting) were also re-checked and eliminated when present [13]. SMILES strings for all active and inactive molecules are available on www.meilerlab.org/qsar_benchmark_2015.

Conformations were generated with Corina version 3.49 [14], with the driver options *wh* to add hydrogens and *r2d* to remove molecules for which 3d structures cannot be generated.

Three descriptor sets used to encode chemical structure

To understand whether dropout is broadly useful for ANN-based QSAR ML methods, three descriptor sets were used. These descriptor sets differ in size, encoding (binary vs. floating point), conformational dependence, as well as redundancy and orthogonality (Table 2).

The *benchmark* descriptor set (BM) includes scalar, topological, and conformation-dependent molecular encodings Scalar descriptors include those described in [6], with the addition of number of rings, aromatic rings, and molecular girth. Topological and conformational descriptors include 2D and 3D-autocorrelations of atomic properties used in [6]. In total, the benchmark set contains 3853 descriptors, 11 of which are scalar, 770 are 2D / topological, and 3072 are 3D (Table 2). The descriptor set differs from that used in Butkiewicz, Lowe et al. 2013 primarily with the introduction of an enhanced 2D and 3D-autocorrelations descriptor that accounts for atom property signs (Sliwoski, Mendenhall et al., in this issue) [15], and the use of min and max to compute binned-values for 2D and 3D autocorrelations, in addition to the traditional use of summation. The BM descriptor set was used for most testing because its size and information content are most similar to commercially-available descriptor sets such as DRAGON [16] and CANVAS [17].

The *short-range* (SR) descriptor set differs from the benchmark set primarily in that the maximum distance considered for the 3D-autocorrelations was reduced from 12 Å to 6 Å. For faster training, the SR set used a smaller set of atom properties (6 vs. 14), which preliminary testing suggested were sufficient to reproduce the performance of the full set. In total, the SR descriptor set contains 1315 descriptors: 24 scalar, 235 topological (2D-autocorrelations), and 1056 spatial (3D-autocorrelations).

A QSAR-tailored variant of the PubChem Substructure Fingerprint descriptor set [23], referred to here-after as the *substructure* (SS) descriptor set, was used to determine whether dropout benefits a binary, fingerprint-based descriptor set. This set contains all but a few of the 881 binary values in the PubChem substructure fingerprint, v1.3. The omitted bits of the fingerprint contain transition metals for which we lack Gasteiger atom types, which is a requirement for the SR and BM sets. Secondarily, when counting rings by size and type, we considered saturated rings of a given size distinctly from aromatic rings of the same size. Lastly, we added sulfonamide to the list of SMARTS queries due to their frequency in drug-like molecules. In total, the SS set contains 922 binary-valued descriptors.

Substructure Searching with Fingerprint Descriptors

The Schrodinger Canvas software suite was used to create MolPrint2D and MACCS fingerprints and search for nearest matches. MolPrint2D was used with ElemRC atom types, consistent with the optimal settings found in a recent benchmark [17]. The closest match for each molecule in a dataset was identified using the Buser metric as implemented in the Canvas package.

ANN Training

Simple propagation [24] was used with η (learning rate) of 0.05. The learning rate η scales the weight adjustment computed during back propagation before applying it to the ANN weights. The momentum parameter α scales the second derivative of the weight change, which is used to accelerate stochastic descent [24], and is tested in this study at values of either 0 or 0.5. In multi-layer ANNs, $\alpha \geq 0.5$ is thought to improve the sampling of ANN weight space [2]. Thirty-two neurons in a single-hidden layer were used throughout the benchmark except where otherwise noted.

Dropout was implemented as in [2] in the machine learning module of the BCL software, source and executables for which are available free of charge for academic use from www.meilerlab.org/bcl_academic_license. After training the ANN, the weights matrix for each layer is multiplied by $1 - d_i$, where d_i is the fraction of neurons dropped in layer i, for scaling purposes. The dropout mask and weights were updated after every feature presentation (online-learning).

The output layer contained neurons with sigmoidal activation output to the range [0,1]. Experimental pK_d or pIC_{50} values were scaled via min-max scaling to [0.1,0.9] to avoid transfer function saturation.

The benchmark was conducted on the Advanced Computing Center for Research and Education (ACCRE) at Vanderbilt University, consuming approximately 500,000 CPU hours.

ANN Performance Evaluation

ANN performance was evaluated by computing the area under the log-linear receiver operating characteristic curve (*log AUC*) [25] between false positive rates (*FPR*) of 0.001 and 0.1, to f ocus on early detection of actives. The *log AUC* values are normalized by the integral of the ideal true-positive rate curve over the same FPR range, such that an optimal classification model obtains a *log AUC* of 1, while a naïve model obtains a *log AUC* of $\frac{\int_{0.001}^{0.1} x \, d\log_{10} x}{\int_{0.001}^{0.1} 1 \, d\log_{10} x} = \frac{\int_{-3}^{-1} 10^u \, du}{\int_{-3}^{-1} 1 \, du} \approx 0.0215$. *log AUC* values are averaged across each of the twenty models in a 5x4-fold cross-validation on its test set, and across the nine datasets in the benchmark.

ANN performance was further assessed by computing enrichment at 1% FPR (Enr_1), averaged across all 20 models in a given cross-validation.

To obtain confidence intervals and standard deviations for each metric, each test set was bootstrap sampled (with replacement) 200 times. *log AUC*s and enrichments were computed for each sample, and across all samples the average and standard deviations was computed. The standard deviation of mean metric values across the benchmark datasets was computed using the equation: $\sigma_\mu = \sqrt{\frac{\sum_{i=1}^{9} \sigma_i^2}{9^2}}$, where σ_i^2 is the variance of the metric on the i-th dataset [26]. T-tests of paired results on the benchmark sets was performed with normalization by the maximum *log AUC* obtained for a given dataset [27]. Performance metrics and confidence intervals were computed using BCL v3.4, model:ComputeStatistics application. Paired t-tests were performed with scipy, version 0.16.0.

Cross-Validation

Five-fold cross-validation was used throughout the present investigation. Specifically, by splitting the dataset into fifths, training on four of the parts (e.g. the training set), and making predictions on the test set. Each of the five parts was a test set for one set of four models. The *log AUC*s and enrichments for each model in a cross-validation on its test set were averaged.

For conventional ANNs, model selection required a second partitioning of the training dataset to compose a monitoring dataset. To reduce bias from an arbitrary choice of the monitoring dataset, four models were trained per test set using a disparate chunk of the training data instead as a monitoring set. When not using a monitoring dataset, the full training set was used to train four ANNs with different starting random seeds.

Optimization of Training Parameters for Dropout and Conventional ANNs

For evaluation of the dropout method, we tested 24 combinations of ANN training and regularization features and parameters for each of the nine datasets in the presence and absence of dropout. The options tested included input scaling method, shuffling [28], active:inactive presentation ratio ($A{:}I_{ratio}$), and model selection. The options were tested at selected combinations of input dropout rate (D_{inp}: 0.0 and 0.25) and hidden dropout rate (D_{hid}: 0.0, 0.25, 0.5, 0.75). An unbiased evaluation of the improvements offered by dropout required testing each type of ANNs under optimal training conditions.

QSAR datasets conventionally suffer from class imbalance (active/inactive) due to the selectivity of the protein targets themselves. Often less than 0.1% of the compounds in the primary screen exhibit significant activity. Class imbalance beyond a roughly 10:1 ratio between the majority and minority class are known to decrease AUC metrics [29]. Previous QSAR modeling techniques have upsampled actives when training ANNs such that every presentation of an inactive molecule to the ANN was followed by the presentation of an active molecule ($\mathrm{A{:}I_{ratio}}$=1:1), with each active molecule being presented thousands of times for each inactive [6,7,30]. We here considered whether a lower $\mathrm{A{:}I_{ratio}}$ would better preserve ANN generalization for either dropout or conventional ANNs.

Overtraining is often mitigated by tracking performance of each ANN during training on a monitoring dataset, which is distinct from both the training and test sets. Training is halted when no improvement is seen after a specified number of iterations, or a specified maximum number of iterations is reached, and the best performing model on the monitoring dataset is selected as the final model (model-selection). Dropout ANNs used in image-processing and related applications appear immune to overtraining and so a monitoring dataset is ordinarily unnecessary when training them. We test whether model-selection is beneficial for dropout ANNs used for QSAR datasets. The influence of scaling was investigated by rescaling to [−1,1] using min-max scaling ($x^{'}=\frac{2x-\mathrm{x}_{min}-\mathrm{x}_{max}}{\mathrm{x}_{max}-\mathrm{x}_{min}}$) of inputs [6], or Z-score scaling ($x^{'}=\frac{x-\mu}{\sigma}$) [28].

Utility of each ANN setting *F* was measured by Δ*log AUC* = max(*log AUC*) − max (*log AUC*(*F*)), where *log AUC*(*F*) is the set of models trained with setting *F*, and similarly

ΔENR_1 for enrichment at 1% FPR. Δ*log AUC* and ΔENR_1 account for the interdependency of optimal options by taking the difference between the best *log AUC* for any options set and the best *log AUC* observed for an options set with a particular option. Significance was assessed by performing a paired t-test between max(*log AUC*) and max(*log AUC*(*F*)) on all models trained with a given setting.

Results & Discussion

Dropout prevents overtraining in QSAR ANNs Independent of Backpropagation Method

A set of ANNs for each of the benchmark datasets for 1000 iterations was trained using either dropout (D_{hid}: 0.5, D_{inp}: 0.25) or no dropout, with the SR descriptor set (Fig. 1), to test for convergence and overtraining. Dropout successfully prevented overtraining even out to 1000 epochs, which is far beyond the maximum of 36 epochs required to achieve convergence on any of the benchmark datasets. The ability of dropout to prevent overtraining is consistent with results from literature on well-balanced datasets [2,4].

Use of momentum (α=0.5) led to a small increase in overtraining in conventional ANNs used for QSAR. Likewise, α was set to 0 for a conservative estimate of the benefit of dropout for all parameter optimizations.

Batch update - updating weights after computing the gradient across the whole dataset - with simple back-propagation learning has been suggested to be more powerful than online learning (where weights are updated every iteration), based on theoretical considerations [31]. We tested batch update on this QSAR benchmark with learning rate set to $\frac{12}{DatasetSize}$. At rates above $\frac{30}{DatasetSize}$, ANNs often failed to train beyond the first epoch. This sensitivity to the learning rate has been noted previously as a weaknesses of batch update [32]. Convergence required approximately 900 epochs, despite using α of 0.9 to improve convergence rate. ANNs trained with batch update were not significantly better than ANNs trained online, and their slow rate of convergence made them unsuitable for benchmarking the extensive ANN features tested in this benchmark. Previous work has found that batch update is usually inferior to online learning in convergence rate across a host of applications [32,33]. While the optimal independent *log AUC* were ~2% larger for conventional batch ANNs than conventional online-learning ANNs, this difference is neither significant nor does it persist when model selection is used.

Resilient propagation (RProp) is an alternative to simple propagation that utilizes second order derivative information in an attempt to accelerate convergence [34]. We found that under dropout conditions, RProp gave equivalent results to simple propagation, yet required over 1000 iterations to converge. When using RProp, the dropout mask was updated after every feature presentation, while the weights were updated only after each epoch.

Optimized Training Conditions for Dropout and Conventional ANNs

A grid search was conducted over 192 combinations of ANN training and regularization features and parameters for each of the nine datasets. The options tested included input scaling method, shuffling, active:inactive presentation ratio (*A:I*$_{ratio}$), and model selection. The options were tested at selected combinations of input dropout rate (D_{inp}: 0.0 and 0.25) and hidden dropout rate (D_{hid}: 0.0, 0.25, 0.5, 0.75). The optimal options differed significantly for the set of ANNs trained without dropout (D_{inp}= D_{hid}=0) vs. those trained with any dropout, but did not differ within the set of ANNs trained with different dropout rates (p > 0.1).

The BM descriptor set was used throughout the options grid search.

Z-score scaling proved critical for dropout ANN performance. The importance of scaling may be related to the definition of dropout. When a neuron is "dropped," its output value is set to0. If 0 is an unusual value due to a skewed descriptor distribution, as can occur with MinMax scaling, the ANN may have difficulty making use of the descriptor. For example, if a descriptor x satisfies $x \in [0,1]$; $\mu_x = 0.9$; $\mu_{x,actives} = 0.5$, $\sigma_{x,actives} = 0.1$, then when the descriptor is presented to the ANN (rescaled to the range [−1,1]), a dropped input will be equivalent to an active input, and the ANN will be unable tell whether the descriptor was dropped or came from an active compound. Z-score scaling ($x' = \frac{x-\mu}{\sigma}$) is thought to mitigate the influence of outliers in the data [28], though our results indicated no significant improvement in results for conventional ANNs.

A:I$_{ratio}$ of 1:1 consistently yielded inferior results in this benchmark for both dropout and conventional ANNs. For conventional ANNs, a 1:100 ratio provided a small benefit (~2–3%) over a 1:1 or 1:10 ratio. Dropout ANNs showed a similar improvement with a 1:10 ratio. A lower ratio also reduces training time significantly (1.9x faster for a 1:10 ratio).

Our results indicated a significant improvement in both *logAUC* and Enr$_1$ for individual dropout ANNs lacking model selection. As expected, conventional ANNs showed a small improvement using model selection. ANNs employing either hidden dropout or input-layer dropout were significantly better than equivalent ANNs trained without dropout (p < 0.001).

logAUC is more Robust than Enrichment to Bootstrap Resampling

Based on bootstrapping of independent results, reported *logAUC* is subject to a standard deviation of +/− 2.2% – 3.3% (M: 2.5%, SD: 0.2%) of the base value. Enr$_1$ was subject to a significantly larger bootstrap error of +/− 2.6% – 5.2% (M: 3.1%, SD: 0.4%) (p < 0.01). This verifies that logAUC is more robust to minor changes in dataset composition and model ranking. Limitations of enrichment as an objective function have been noted elsewhere [35], but it remains useful for comparison with other methods. When reporting percentage improvements, *logAUC* is used except where otherwise noted.

Dropout Optimization

A grid search for optimal D_{inp} and D_{hid} values was conducted for the benchmark datasets. ANNs were trained with dropout rates (D_{hid} & D_{inp}) sampled between 0 and 0.5 (for D_{hid}) or 0.95 (for D_{inp}) at a step size of 0.05. The upper limit of 0.5 was chosen for D_{hid} based on the options-based grid-search, wherein we noted that D_{hid} of 0.25 usually provided modestly better results than 0.5, and significantly better than D_{hid} of 0.75 (p < 0.05).

The optimization of D_{inp} and D_{hid} across the benchmark datasets is shown in

Fig. 2. The global optimum is at 60% D_{inp} and 25% D_{hid}, though the results are generally robust to D_{inp} between 0.4 – 0.7 and D_{hid} between 0.1 – 0.5.

Fig. 3 illustrates the dependence of *logAUC* on D_{inp} and D_{hid} for each dataset. The numbers next to each dataset ID indicate the value of *logAUC* at D_{inp}=D_{hid}=0, followed by the maximal *logAUC* value. The *logAUC* at the dataset-specific optima of D_{inp} and D_{hid} averages 2.5% higher than that obtained using the overall optimal parameters (D_{inp}=0.6, D_{hid}=0.25), however, the differences in optimal dropout values are not significantly different across the datasets, relative to uncertainty in the *logAUC* value itself ($p > 0.1$). Importantly, this suggests that the dropout optimization need not be repeated for novel QSAR targets using this descriptor set.

In the input layer (where the descriptors are fed in), dropout hinders the ability of the ANN to describe complex functions. The descriptor sets employed here incorporated several hundred descriptors with substantial redundancy. In such a setting, dropout in the input layer prevents the ANN from relying on any two particular descriptors being present at the same time. We anticipated that it would be useful when training with dropout to include several representations of each molecular property. For example, atomic charge could be represented using either VCharge [21] or σ/π-charge computed by partial equalization of orbital electronegativity [20]. Thereby, the omission of any particular representation of a given property was not detrimental to the description of the molecule.

Importance of Descriptor Set to Optimal Dropout Fractions & Comparison with Fingerprint Methods

D_{inp} was assessed for the BM, SR, and SS descriptor sets, while keeping D_{hid} fixed to the optimal value of 0.25.

The performance of ANNs trained without dropout for each descriptor set was analyzed both with and without model selection. Both the SS and SR descriptor sets were relatively insensitive to D_{inp} between $0.05 \le D_{inp} \le 0.6$, while the BM set showed a strong dependence on D_{inp} with a maximum at 0.6. Figure 3 shows the results. The SR descriptor set, with optimized dropout parameters, was 4.8% better compared to the BM descriptor set overall, while the SS set averaged 11.3% worse. At $D_{inp} > 0.6$, the performance of the SR and SS sets declined rapidly.

The impact of the descriptor set on the optimal ANN D_{inp} motivated a similar test for D_{hid}, with D_{inp} fixed at 0.05. Figure 4 depicts the results. The curves for D_{hid} are substantially similar ($R^2 > 0.8$ for all pairs), suggesting that D_{hid} can be optimized for ANN architecture independently of the dataset. In all three datasets, the region $0.2 < D_{hid} < 0.6$ is optimal.

While the SR descriptor set was best used in conjunction with a very low input dropout rate ($D_{inp,optimal}$=0.05), the BM set required a high rate of input dropout ($D_{inp,optimal}$=0.60). Moreover, the relatively small SR set outperformed the BM set. The sets differ in spatial boundaries for 3DA calculation as well as atom properties, raising the question of which of these changes is responsible for the increase in $D_{inp,optimal}$. Using the BM set's atom properties with the SRs 3DA extent, *logAUC* decreased insignificantly (~1.5%, $p > 0.1$)

relative to results on the SR set. The different atom properties between the BM and SR sets was, likewise, not a significant factor in altering the sensitivity of the descriptor set performance to D_{inp}.

We propose that the optimal D_{inp} is indicative of the signal to noise ratio in the descriptors. Sensitivity to D_{inp} was much greater in the BM set ($0.5 < D_{inp,optimal} < 0.75$, +9.1% improvement in *logAUC* from optimizing $D_{inp}>0$) than the SS or SR sets ($0.0 < D_{inp,optimal} < 0.65$, 2.4% improvement in *logAUC* from optimizing $D_{inp}>0$). The descriptors in the BM set that are not in the SS set are heavily dependent on the specific molecular conformation that was used, in particular due to the 3DA bins beyond 6Å (only present in the BM set). For a molecule with typical flexibility, a pair of atoms at 10Å in the average conformation may be anywhere between 8Å–12Å depending on the rotamer, so there is substantial uncertainty involved when assigning these atom pairs to one of the 0.25 Å bins of a 3DA. The SS set is purely fragment-based and thus none of the descriptors has significant uncertainty, and so the optimal D_{inp} is < 0.25.

The influence of the dataset on the optimal dropout parameters has further implications for the design of novel descriptors. In particular, benchmarking novel descriptors may require optimizing D_{inp} for each descriptor set on a benchmark dataset, to enable fair comparison across descriptor sets with differing levels of uncertainty. For large QSAR datasets (> 60K compounds) with 800 – 3800 descriptors, as used in this study, D_{inp} of 0.5 and D_{hid} of 0.25 appear to be good starting points. $D_{inp} > 0.6$ may be appropriate for descriptors sets with more than 4k values and for descriptors associated with high degrees of uncertainty, such as long distance 3DAs.

Our benchmark suggests the utility of optimizing dropout parameters for well-established benchmark descriptor sets for QSAR such as WHIM, CPSA, and 3D-Morse [36].

ANN training may benefit from setting input dropout probabilities for each descriptor column according to the uncertainty in their value. This should be less important for canonical machine learning problems such as number recognition, where every descriptor has the same units (e.g. pixel intensity) and levels of uncertainty. Nevertheless, there is evidence that variable levels of dropout can improve ANN regularization even on traditional MNIST-style benchmarks [37]. For 3D-conformational descriptors, it may be fruitful to use the conformational ensemble for each ligand to derive the probability that each 3D-descriptor column is substantially different from its nominal value as an input-specific dropout probability.

Probing the Role of Input Dropout on AID435034

For further analysis of the role of input dropout, AID435034 was chosen as a representative dataset based on the similarity of its' D_{inp}, D_{hid} heatmap and the benchmark average.

The SR descriptor set was padded with as many Gaussian-noise descriptor columns as it has true descriptors to form the SR+Noise set. We optimized D_{inp} for the SR+Noise set, with D_{hid} fixed at 0.25. Figure 5 shows that with $D_{inp}=0$, less than half the original performance of the original SR set is recovered (0.13 versus 0.29). The optimized value of D_{inp} shifts from 0.05 for the SR descriptor set to 0.85 in the SR+Noise set. The added noise also leads to a significantly greater dependence of *logAUC* on D_{inp}, with only a 3.1% gain in *logAUC* in the base SR set from optimizing D_{inp} vs. leaving it at 0, while with the added noise columns a 67% increase in *logAUC*. The SR+Noise set performed 34% worse after optimization of D_{inp} than the SR set. We posit that the addition of noise columns is particularly detrimental to QSAR datasets due to the small number of actives and the probability that, given a large enough number of noise columns, one of them by chance has a significant correlation with activity for a few actives in a given dataset.

The effect of redundant descriptors on the optimal D_{inp} was investigated by duplicating every descriptor in the SR descriptor set (SRx2). Optimizing D_{inp} yielded a curve with the same peak *logAUC* as in the SR set on AID435034, but with the overall curve shifted by +5% (Figure 5). The same magnitude of shift was also observed when optimizing the duplicated BM descriptor set on AID435034 (data not shown).

Lastly, the effects of zero-padding the SR descriptor set were considered by doubling the SR descriptor set size by padding it with zeros (Figure 5: SR + 0s). The resulting curve is qualitatively indistinguishable from the results on the unperturbed SR set. The lack of effect of zero-padding columns on the optimal input dropout rates additionally supports the notion that $D_{inp,optimal}$ is primarily a function of the signal-to-noise ratio in the dataset.

Comparison with Fingerprint Methods

Dropout ANNs trained on MACCS keys improved *logAUC* by 38% relative to simple similarity searches, and 26% relative to conventional ANNs. MolPrint2D fingerprints using the optimal settings and comparison metric proved superior to conventional ANNs trained with the SR descriptor set, but inferior to dropout ANNs by ~18% in *logAUC*. Nevertheless, MolPrint2D fingerprints were superior in training dropout ANNs on one dataset (AID488997). It may prove fruitful to use the MolPrint2D fingerprints as additional descriptors to train ANNs in future work, based on the results with dropout ANNs using MACCS keys. These results further validate a smaller benchmark that found similar improvement using ANNs on fingerprint-style descriptors [5]. We expect, however, that fingerprint-type will be relatively limited in terms of identifying novel active scaffolds, a purpose for which we anticipate that conformational and electrostatic descriptors such as those used here will have a clear advantage.

Sensitivity to choice of atom properties

For the BM descriptor set, we used atom properties from prior studies [6,38], specifically: Identity, Polarizability, Electronegativities (σ, π, lone-pair) and partial charges (σ, π, σ+π, VCharge) for use in 2DA and 3DA functions [20,39,40]. The charge properties can take on positive and negative values, and likewise were used exclusively in the sign-sensitive 2DA and 3DAs [15].

For the SR descriptor set, a reduced set of four atom properties was used: σ charge, $\begin{cases} 1, & \text{Hydrogen} \\ -1, & \text{Heavy Atom} \end{cases}$, $\begin{cases} 1, & \text{In an aromatic ring} \\ -1, & \text{Not in an aromatic ring} \end{cases}$. Testing proved that this set yielded slightly better (1.5% on average *logAUC*) performance than that of the BM set, while being substantially smaller in size. Testing with sixteen different sets of up to twenty different atom properties produced *logAUC* that remained within 5% of this optimal set so long as the set contained σ charge and $\begin{cases} 1, & \text{Hydrogen} \\ -1, & \text{Heavy Atom} \end{cases}$ or a similar descriptor capable of describing hydrogenation and steric bulk (results not shown), suggesting that our results are relatively insensitive to atom properties. The additional atom properties that were tested included splitting charges based on whether they were associated with a heavy or light atom, as well as simpler versions of the charges where each atomic charge was converted to 0, −1, or +1 depending on specific cutoffs derived from the distribution of each of the charge types. While these descriptors improved results on specific datasets by up to 10% in some cases, the improvements did not hold across the benchmark.

Caveats and Limitations

ANNs trained in this study had a single hidden layer with thirty-two neurons. To understand the relative influence of neural architecture on our results, we used our optimized parameters to train a larger ANN, keeping other parameters set to their optimal values from the benchmark. Using a hidden layer size of 256 neurons resulted in an insignificant change ($p > 0.1$) in *logAUC* values (+1.3% +/− 3.2%) across the benchmark datasets. While larger ANNs or additional hidden layers could improve the outcomes of this study, this exploration leads us to expect that the benefit of larger ANNs will be small relative to the use of dropout itself.

ANNs were trained to one hundred iterations. For ANNs without dropout, optimal performance was obtained between 2 and 49 iterations ($\mu=14$, $\sigma^2=15$) using the SR descriptors, depending on the dataset. With an input and hidden layer dropout fraction of 25%, convergence to within 99% of the final *logAUC* required between 8 and 26 iterations ($\mu=15$, $\sigma^2=6$). Increasing the input dropout rate to 50% further increased the number of iterations for convergence to an average of 35 (range: 20 – 69). Input dropout rates of 75% and higher likewise may require more than one hundred iterations converge to within 1% of their optimal *logAUC* on some datasets. Given that this is well above the optimal input dropout rate found in this study for all descriptor sets, it appears unlikely that our optimization would differ significantly if performed with further iterations.

Conclusions

When training ANNs on large QSAR datasets, dropout is important in both the input and hidden layers. Compared to conventional ANNs trained with no model selection or early termination, dropout ANNs improves *logAUC* by an average of 36%, and enrichment by 29%. Compared to ANNs trained with model selection, dropout ANNs still outperforms conventional ANNs by an average of 22% across the descriptors sets considered here. Dropout ANNs outperformed optimized similarity searching methods based on MolPrint2D fingerprints by 18%. The dropout technique thus places ANNs at the forefront of QSAR modeling tools.

Applying dropout to the ANN hidden layer at the fractions prescribed by this benchmark will provide a starting point for further optimization on QSAR datasets of commercial and academic interest, and further highlight the need for widespread dissemination of contemporary machine learning techniques into broader disciplines.

References

1. Sliwoski G, Kothiwale S, Meiler J, Lowe EW Jr. Computational methods in drug discovery. Pharmacol Rev. 2014; 66(1):334–395.10.1124/pr.112.007336 [PubMed: 24381236]
2. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research. 2014; 15:1929–1958.
3. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems. 2012:1097–1105.
4. Deng, L.; Hinton, G.; Kingsbury, B. New types of deep neural network learning for speech recognition and related applications: An overview. Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on; 2013; IEEE; p. 8599-8603.
5. Myint KZ, Wang L, Tong Q, Xie XQ. Molecular fingerprint-based artificial neural networks QSAR for ligand biological activity predictions. Mol Pharm. 2012; 9(10):2912–2923.10.1021/mp300237z [PubMed: 22937990]
6. Butkiewicz M, Lowe EW Jr, Mueller R, Mendenhall JL, Teixeira PL, Weaver CD, Meiler J. Benchmarking ligand-based virtual High-Throughput Screening with the PubChem database. Molecules. 2013; 18(1):735–756.10.3390/molecules18010735 [PubMed: 23299552]
7. Mueller R, Dawson ES, Niswender CM, Butkiewicz M, Hopkins CR, Weaver CD, Lindsley CW, Conn PJ, Meiler J. Iterative experimental and virtual high-throughput screening identifies metabotropic glutamate receptor subtype 4 positive allosteric modulators. J Mol Model. 2012; 18(9):4437–4446.10.1007/s00894-012-1441-0 [PubMed: 22592386]
8. Sliwoski G, Lowe EW, Butkiewicz M, Meiler J. BCL::EMAS--enantioselective molecular asymmetry descriptor for 3D-QSAR. Molecules. 2012; 17(8):9971–9989.10.3390/molecules17089971 [PubMed: 22907158]
9. Hartman JH, Cothren SD, Park SH, Yun CH, Darsey JA, Miller GP. Predicting CYP2C19 catalytic parameters for enantioselective oxidations using artificial neural networks and a chirality code. Bioorg Med Chem. 2013; 21(13):3749–3759.10.1016/j.bmc.2013.04.044 [PubMed: 23673224]
10. Ahmadi M, Shahlaei M. Quantitative structure-activity relationship study of P2X7 receptor inhibitors using combination of principal component analysis and artificial intelligence methods. Res Pharm Sci. 2015; 10(4):307–325. [PubMed: 26600858]
11. Dahl GE, Jaitly N, Salakhutdinov R. Multi-task neural networks for QSAR predictions. 2014 arXiv preprint arXiv:14061231.
12. Dahl, G. [Accessed Aug 14, 2015] Deep Learning How I Did It: Merck 1st place interview. 2012. http://blog.kaggle.com/2012/11/01/deep-learning-how-i-did-it-merck-1st-place-interview/
13. Cherkasov A, Muratov EN, Fourches D, Varnek A, Baskin II, Cronin M, Dearden J, Gramatica P, Martin YC, Todeschini R, Consonni V, Kuz'min VE, Cramer R, Benigni R, Yang C, Rathman J, Terfloth L, Gasteiger J, Richard A, Tropsha A. QSAR modeling: where have you been? Where are you going to? J Med Chem. 2014; 57(12):4977–5010.10.1021/jm4004285 [PubMed: 24351051]
14. Sadowski J. A hybrid approach for addressing ring flexibility in 3D database searching. J Comput Aid Mol Des. 1997; 11(1):53–60.
15. Berenger F, Voet A, Lee XY, Zhang KYJ. A rotation-translation invariant molecular descriptor of partial charges and its use in ligand-based virtual screening. J Cheminformatics. 2014; 6:Artn 23.10.1186/1758-2946-6-23
16. Todeschini, R.; Consonni, V. Methods and principles in medicinal chemistry. 2. Vol. 41. Wiley-VCH; Weinheim: 2009. Molecular descriptors for chemoinformatics.

17. Sastry M, Lowrie JF, Dixon SL, Sherman W. Large-scale systematic analysis of 2D fingerprint methods and parameters to improve virtual screening enrichments. J Chem Inf Model. 2010; 50(5):771–784.10.1021/ci100062n [PubMed: 20450209]

18. Xing L, Glen RC. Novel methods for the prediction of logP, pK(a), and logD. J Chem Inf Comput Sci. 2002; 42(4):796–805. [PubMed: 12132880]

19. Ertl P, Rohde B, Selzer P. Fast calculation of molecular polar surface area as a sum of fragment-based contributions and its application to the prediction of drug transport properties. J Med Chem. 2000; 43(20):3714–3717. [PubMed: 11020286]

20. Gasteiger J, Marsili M. Iterative Partial Equalization of Orbital Electronegativity - a Rapid Access to Atomic Charges. Tetrahedron. 1980; 36(22):3219–3228.10.1016/0040-4020(80)80168-2

21. Gilson MK, Gilson HS, Potter MJ. Fast assignment of accurate partial atomic charges: an electronegativity equalization method that accounts for alternate resonance forms. J Chem Inf Comput Sci. 2003; 43(6):1982–1997.10.1021/ci034148o [PubMed: 14632449]

22. Miller KJ. Additivity Methods in Molecular Polarizability. J Am Chem Soc. 1990; 112(23):8533–8542.10.1021/Ja00179a044

23. PubChem. [Accessed May 05 2014] PubChem Substructure Fingerprint. 2009. ftp://ftp.ncbi.nlm.nih.gov/pubchem/specifications/pubchem_fingerprints.pdf

24. Rumelhart DE, Hinton GE, Williams RJ. Learning Representations by Back-Propagating Errors. Nature. 1986; 323(6088):533–536.10.1038/323533a0

25. Mysinger MM, Shoichet BK. Rapid context-dependent ligand desolvation in molecular docking. J Chem Inf Model. 2010; 50(9):1561–1573.10.1021/ci100214a [PubMed: 20735049]

26. Weisstein, EW. Normal Sum Distribution. Wolfram Research, Inc; 2000. http://mathworld.wolfram.com/NormalSumDistribution.html [Accessed November 1, 2015]

27. Valcu M, Valcu CM. Data transformation practices in biomedical sciences. Nat Methods. 2011; 8(2):104–105.10.1038/nmeth0211-104 [PubMed: 21278720]

28. LeCun, Y.; Bottou, L.; Orr, G.; Müller, K-R. Efficient BackProp. In: Orr, G.; Müller, K-R., editors. Neural Networks: Tricks of the Trade, vol 1524. Lecture Notes in Computer Science. Springer; Berlin Heidelberg: 1998. p. 9-50.

29. Prati RC, Batista GE, Silva DF. Class imbalance revisited: a new experimental setup to assess the performance of treatment methods. Knowledge and Information Systems. 2014:1–24.

30. Batista GEAPA, Prati RC, Monard MC. Balancing strategies and class overlapping. Advances in Intelligent Data Analysis Vi, Proceedings. 2005; 3646:24–35.

31. Nakama T. Theoretical analysis of batch and on-line training for gradient descent learning in neural networks. Neurocomput. 2009; 73(1–3):151–159.10.1016/j.neucom.2009.05.017

32. Wilson DR, Martinez TR. The general inefficiency of batch training for gradient descent learning. Neural Netw. 2003; 16(10):1429–1451.10.1016/S0893-6080(03)00138-2 [PubMed: 14622875]

33. Wu W, Wang J, Cheng M, Li Z. Convergence analysis of online gradient method for BP neural networks. Neural Netw. 2011; 24(1):91–98.10.1016/j.neunet.2010.09.007 [PubMed: 20870390]

34. Igel C, Husken M. Empirical evaluation of the improved Rprop learning algorithms. Neurocomputing. 2003; 50:105–123. Pii S0925-2312(01)00700-7. 10.1016/S0925-2312(01)00700-7

35. Jain AN, Nicholls A. Recommendations for evaluation of computational methods. J Comput Aid Mol Des. 2008; 22(3–4):133–139.10.1007/s10822-008-9196-5

36. Gasteiger, J. Handbook of chemoinformatics : from data to knowledge. Wiley-VCH; Weinheim: 2003.

37. Ba J, Frey B. Adaptive dropout for training deep neural networks. Advances in Neural Information Processing Systems. 2013:3084–3092.

38. Mueller R, Rodriguez AL, Dawson ES, Butkiewicz M, Nguyen TT, Oleszkiewicz S, Bleckmann A, Weaver CD, Lindsley CW, Conn PJ, Meiler J. Identification of Metabotropic Glutamate Receptor Subtype 5 Potentiators Using Virtual High-Throughput Screening. ACS Chem Neurosci. 2010; 1(4):288–305.10.1021/cn9000389 [PubMed: 20414370]

39. Marsili M, Gasteiger J. Pi-Charge Distribution from Molecular Topology and Pi-Orbital Electronegativity. Croat Chem Acta. 1980; 53(4):601–614.

40. Gilson MK, Gilson HSR, Potter MJ. Fast assignment of accurate partial atomic charges: An electronegativity equalization method that accounts for alternate resonance forms. J Chem Inf Comput Sci. 2003; 43(6):1982–1997.10.1021/Ci034148o [PubMed: 14632449]

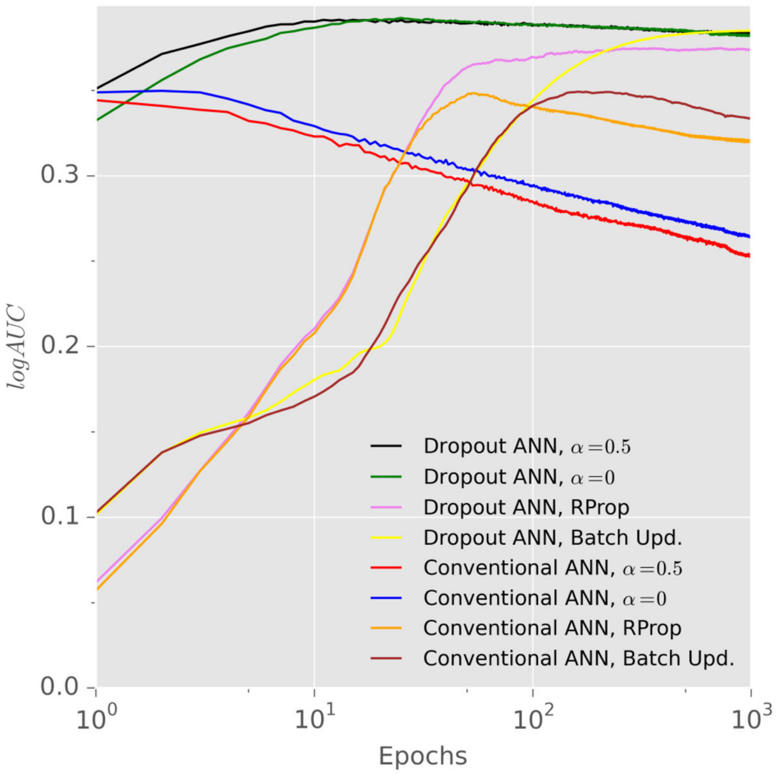

Fig. 1.

Average *logAUC* by training epoch for dropout and conventional networks. Conventional ANNs converge within 10 iterations on all datasets and over-train afterwards. Dropout ANNs converge within 25 iterations on all datasets, reach higher *logAUC* values and do not over-train, even after 1000 iterations. Using momentum (α) improves convergence at the expense of slightly lower *logAUC*. Batch update and resilient propagation exhibit slower convergence but similar overtraining without dropout, and reach essentially the same peak performance as observed with online-learning.

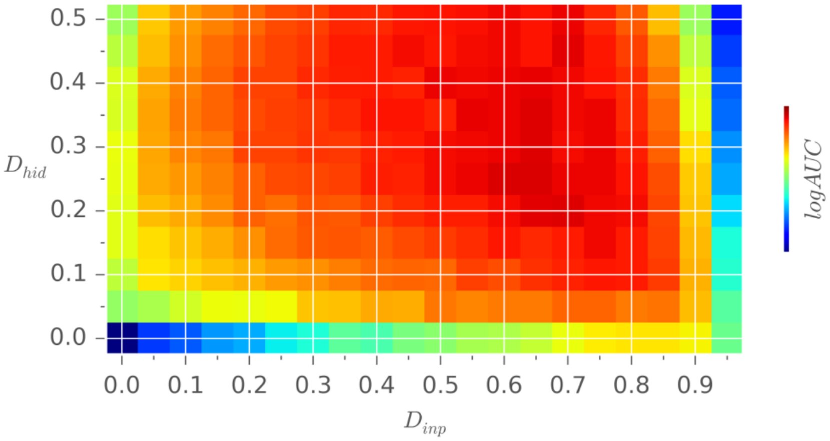

Fig. 2.

Optimization of input and hidden layer dropout rates, averaged over the benchmark datasets. Optimum is at D_{inp}=0.6, D_{hid}=0.25, with an average *logAUC* of 0.377. The weakest performance is seen with D_{inp}=D_{hid}=0 (e.g. no dropout), with an average *logAUC* of 0.255.

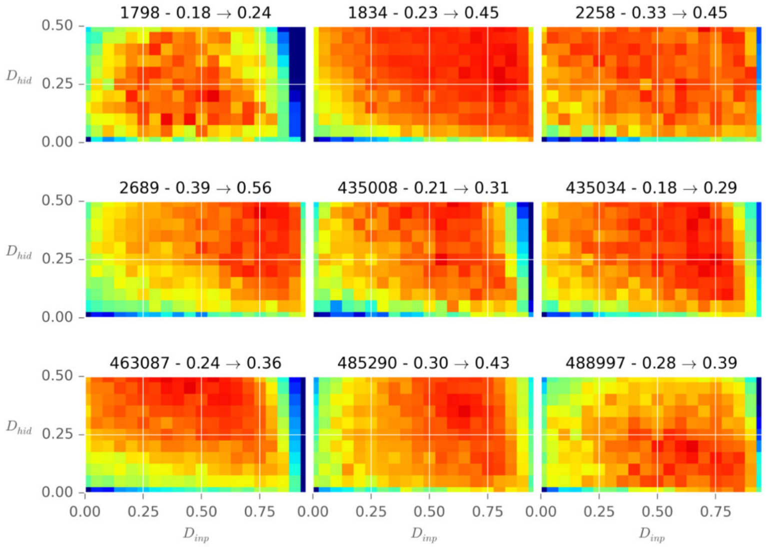

Fig. 3.

Per-dataset optimization of *logAUC*. Title bar for each dataset indicates ID - *logAUC* of the dataset trained without dropout → *logAUC* after dropout rate optimization.

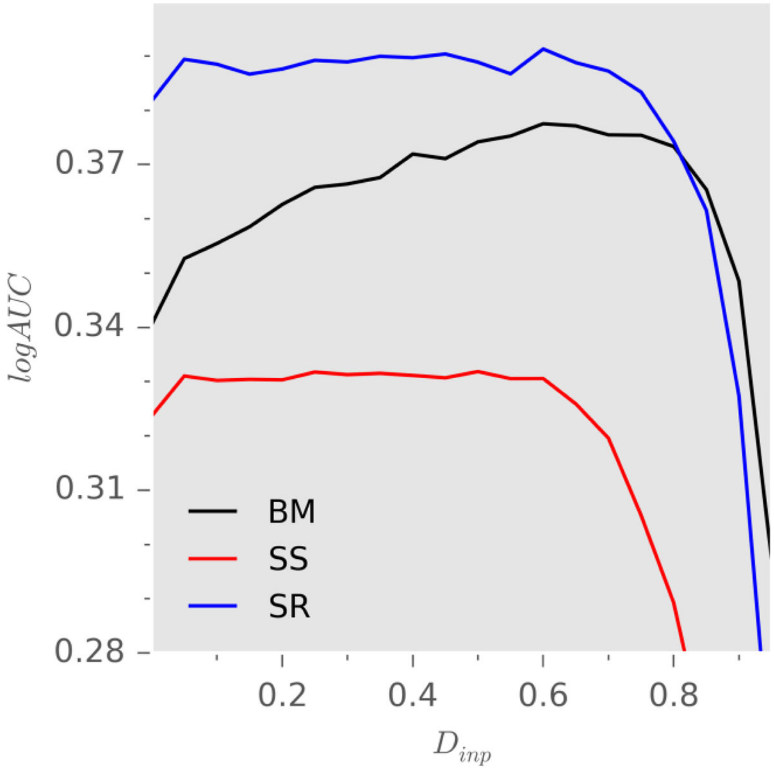

Fig. 4.

Dependence of D_{inp} optimization on descriptor set. D_{hid} = 0.25 for all descriptor sets. BM – Benchmark descriptor set, SS – substructure descriptor set, SR – short-range descriptor set.

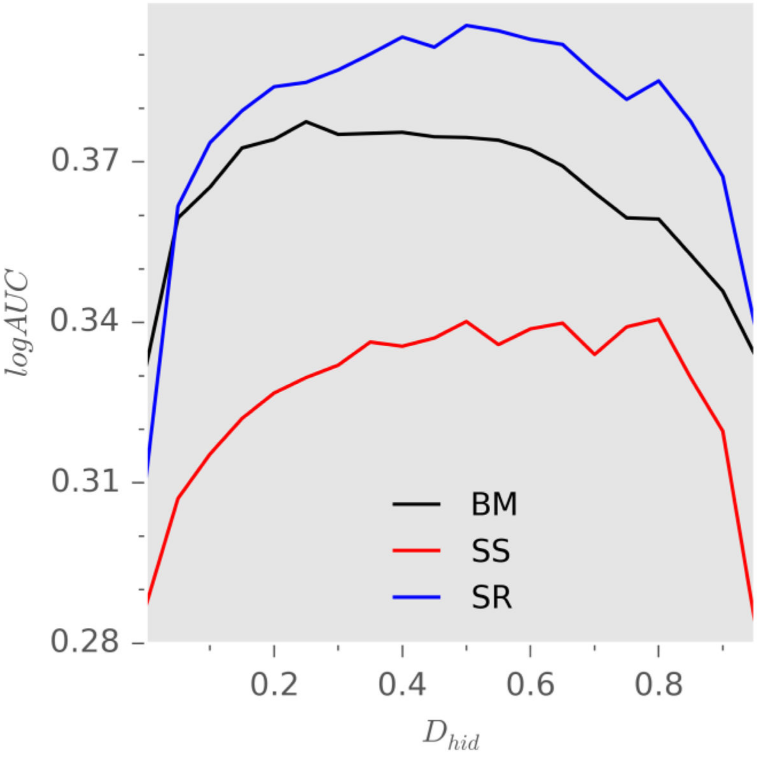

Fig. 5.

Dependence of D_{hid} optimization on descriptor set. Optimal values of D_{inp} used for each descriptor set from Figure 2 (D_{inp} = 0.05 for SS & SR, D_{inp} = 0.6 for BM).

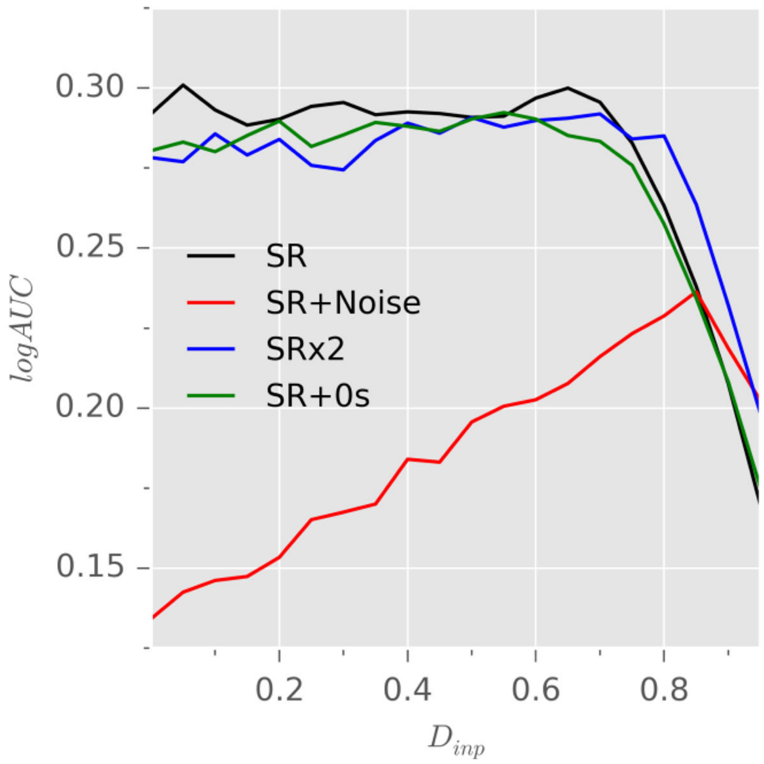

Fig. 6.

Effect of artificial descriptor set defects on D_{inp} optimization. SR - short range descriptor set, SR+Noise – SR descriptors & an equal number of Gaussian noise columns, SRx2 – SR descriptors, repeated (all descriptors represented twice), SR+0s – SR descriptors, with all descriptors paired with an all-0 column.

Table 1

Datasets used in the benchmark. The PubChem Summary ID (SAID) is used to refer to the datasets throughout this manuscript

Protein Class – Target	PubChem SAID	# Active Molecules	# Inactive Molecules
GPCR – Orexin1 Receptor Antagonists	435008	233 (0.11%)	217925
GPCR – M1 Muscarinic Receptor Agonists	1798	187 (0.30%)	61646
GPCR – M1 Muscarinic Receptor Antagonists	435034	362 (0.59%)	61394
Ion Channel – Kir$_{2.1}$ K+ Channel Inhibitors	1843	172 (0.06%)	301321
Ion Channel – KCNQ2 K+ Channel Potentiators	2258	213 (0.07%)	302192
Ion Channel – Cav3 T-type Ca^{2+} Inhibitors	463087	703 (0.70%)	100172
Transporter – Choline Transporter Inhibitors	488997	252 (0.08%)	302054
Kinase Inhibitor –Serine/Threonine Kinase 33 Inhib.	2689	172 (0.05%)	319620
Enzyme –Tyrosyl-DNA Phosphodiesterase Inhib.	485290	281 (0.08%)	341084

Table 2

Complete list of descriptors in the BM and SR descriptor sets. For signed 2DAs and 3DAs, unsigned atom properties (Polarizability, Identity, VdW Surface Area) were multiplied by −1 for hydrogen atoms, to enhance the information content of these descriptors.

Scalar Descriptors	Descriptor Set
Molecular Weight	BM & SR
# hydrogen bond donors and acceptors	
LogP - Octanol/water coefficient [18]	
Total charge of molecule	
# of rotatable bonds	
# of aromatic rings	
# of rings	
TPSA - Total polar surface area of molecule [19]	
Bond Girth - maximum # of bonds between two atoms	
Girth - Widest diameter of molecule (Å)	
# of atoms in largest & smallest rings	SR
# of atoms in aromatic rings	
# of bridge atoms in fused rings	
# of bridge atoms in fused aromatic rings	
Min, Max, Std, Absolute sum of σ charges [20]	
Min, Max, Std, Absolute sum of V charges [21]	

2DAs

11 bonds - 12 values for each atom property

Binning Kernel	Atom Properties	Descriptor Set
Sum	Identity (1), Polarizability [22], VdW Surface Area	BM
Max, Min	Polarizability, VdW Surface Area	BM
Sum	Identity (1), IsInAromaticRing, IsAromaticRingBridgeAtom, VCharge on Hydrogen, σ charge on Hydrogen	SR

Signed 2DAs

BM: 11 bonds - 36 values / atom property

SR: 5 bonds - 18 values / atom property

Binning Kernel	Atom Properties	Descriptor Set
Sum, Max	σ charge, V-Charge, Polarizability	BM & SR
Max	VdW-SA, VdW-SA weighted σ charge, V-Charge, and Polarizability	BM
Sum	π charge, σ+π charge, VdW-SA weighted σ charge, V-Charge, and Polarizability; VdW-SA, Discretized σ charge (Q < −0.15, −0.15 < Q < 0.15, Q > 0.15), Discretized π charge (Q < −0.1, −0.1 < Q < 0.1, Q > 0.1), Discretized V-charge (Q < −0.25, −0.25 < Q < 0.25, Q > 0.25)	BM
Sum, Max	Identity, σ charge on heavy atoms, V-Charge on heavy atoms	SR

3DAs

BM: 0.25 Å bins, 12 Å max – 48 values / atom property

SR: 0.25 Å bins, 6 Å max – 24 values / atom property

Binning Kernel	Atom Properties	Descriptor Set
Sum, Max, Min	Identity (Sum only), Polarizability, VdW SA, VdW-weighted Polarizability	BM
Sum, Max	IsInAromaticRing, IsAromaticRingBridgeAtom, VCharge on Hydrogen, σ charge on Hydrogen	SR

Signed 3DAs

BM: 0.25 Å bins, 12 Å max – 144 values / atom property

SR: 0.25 Å bins, 6 Å max – 72 values / atom property

Binning Kernel	Atom Properties	Descriptor Set
Sum, Max	σ charge, V-Charge, IsH (1 for H, −1 for Heavy atoms)	BM & SR
Sum, Max	π charge, σ+π charge, VdW-weighted σ, π, σ+π, and V-Charge	BM
Sum, Max	σ charge on heavy atoms, V-Charge on heavy atoms, Polarizability	SR

Table 3

Options optimization results. Reported ***log AUC*(*F*)** is subject to a standard deviation of +/− 2.1% − 2.9% of the reported value (0.007 – 0.009), assessed using bootstrap resampling as described in Methods. ***Enr*$_1$(*F*)** are similarly subject to a standard deviation of +/− 3.0 – 3.4% (0.7–1.0). Significance is reported at at $p < 0.05$ (*), or $p < 0.025$ (**). Percentage improvement is computed from raw data before rounding.

		max (*log AUC*(*F*))		**max (*Enr*$_1$(*F*))**	
Parameter	**Value**	**Dropout ANN**	**Conventional ANN**	**Dropout ANN**	**Conventional ANN**
Scaling	*MinMax*	0.33	**0.30**	30	27
	Z-Score	**0.37** ↑ 10%**	**0.30**	**35** ↑14%**	**28**
Model Selection	*Yes*	0.34	**0.30* ↑6.2%**	31	**28* ↑7.0%**
	No	**0.37** ↑8.4%**	0.28	**35** ↑11%**	26
$A{:}I_{ratio}$	*1:100*	0.35	**0.30* ↑3.4%**	33	28
	1:10	**0.37** ↑2.8%**	0.29	**35* ↑3.3%**	27
	1:1	0.35	0.28	33	27
Shuffling	*Yes*	0.37	0.30	35	28
	No	0.37	0.29	35	27

Table 4

Optimized training conditions for dropout and conventional ANNs

Parameter	Optimum for dropout ANNs	Optimum for conventional ANNs
Scaling	Z-Score	Z-Score
Model Selection	No	Yes
Active:Inactive Ratio	1:10	1:100
Shuffling	Yes	Yes

Table 5

Improvements in ***logAUC*** observed across the three descriptors with dropout. Italicized numbers show the improvement relative to the baseline conditions of no dropout or model selection and a 1:10 A:I presentation ratio. Significance is reported at at $p < 0.05$ (*), or $p < 0.01$ (**). Reported ***logAUC*** is subject to a standard deviation of +/− 2.4% – 2.9% of the reported value (0.008 – 0.009), assessed using bootstrap resampling as described in Methods.

***log AUC* by training condition** **Average model results**			
ANN Parameters			
D_{inp}	0	0	0.6
D_{hid}	0	0	0.25
Model Selection	−	+	−
A:I ratio	1:10	1:100	1:10
Descriptor set		*log AUC*	
BS	0.26	0.30** *+16%*	0.37** *+46%*
SR	0.29	0.33** *+14%*	0.39** *+35%*
SS	0.26	0.27* *+2.6%*	0.33** *+26%*
Descriptor set		**Enr_1**	
BS	27	28* *+2.6%*	35** *+31%*
SR	28	31** *+12%*	37** *+33%*
SS	25	25	31** *+23%*
Similarity Search			
Fingerprint	*log AUC*		**Enr_1**
MACCS	0.24	23	
MolPrint2D	0.33	33	

Table 6

Per-dataset ***logAUC***s and Enr_1 for selected ANN training parameters, and comparison with results from similarity searches using MACCS and MolPrint2D fingerprints. The first column corresponds to the options used in [6], the second is the optimal conventional ANN. The third column is the optimal training condition for the BM descriptor set. The fourth column is the optimal SR and SS descriptor set results. MACCS and MolPrint2D fingerprint search methods provided for comparison (final two columns).

***logAUC* / Enr$_1$ by training condition** **Average model results**						
ANN Parameters					**Fingerprints**	
D$_{inp}$	0	0	0.6	0.05	MACCS with Tanimoto similarity	MolPrint2D with Buser similarity
D$_{hid}$	0	0	0.25	0.25		
Model Sel.	+	+	−	−		
A:I ratio	1:1	1:100	1:10	1:10		
Scaling	MinMax	Z-Score	Z-Score	Z-Score		
Shuffle	−	+	+	+		
Descriptor Set	BM	BM	BM	SR		
Dataset						
1798	0.16 / 15	0.19 / 17	0.23 / 22	0.26 / 23	0.14 / 14	0.20 / 21
1834	0.35 / 34	0.36 / 34	0.44 / 43	0.44 / 43	0.26 / 22	0.34 / 31
2258	0.36 / 37	0.35 / 34	0.44 / 44	0.48 / 48	0.25 / 24	0.38 / 37
2689	0.39 / 42	0.44 / 43	0.55 / 55	0.56 / 57	0.41 / 45	0.52 / 54
435008	0.22 / 15	0.22 / 19	0.31 / 26	0.30 / 28	0.17 / 12	0.28 / 28
435034	0.20 / 17	0.22 / 16	0.28 / 22	0.30 / 26	0.18 / 18	0.22 / 19
463087	0.20 / 19	0.26 / 21	0.35 / 30	0.40 / 34	0.25 / 21	0.32 / 29
485290	0.33 / 33	0.35 / 33	0.41 / 40	0.44 / 44	0.26 / 26	0.38 / 36
488997	0.21 / 25	0.28 / 27	0.39 / 38	0.38 / 37	0.31 / 29	0.41 / 42
Average	**0.27 / 26**	**0.30 / 27**	**0.37 / 36**	**0.39 / 38**	**0.24 / 23**	**0.34 / 33**