Development and application of ligand-based computational methods for *de-novo* drug design and virtual screening

By

Alexander Richard Geanes

Thesis

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfillment of the requirements

for the degree of

MASTER OF SCIENCE

in

Chemistry

December, 2016

Nashville, Tennessee

Approved:

Prof. Jens Meiler, Ph.D.

Prof. Craig Lindsley, Ph.D.

To my wife, Amanda, for her extraordinary support and patience.

And to my parents and brothers, who fostered my scientific curiosity

and kept me on my toes.

ACKNOWLEDGMENTS

I would like to thank my advisors, Jens Meiler and Craig Lindsley for all that they have done for me both scientifically and professionally. They were both fantastically supportive during my time at Vanderbilt, allowed me to participate in great research, and were able to provide great insight into many scientific challenges. I would also like to thank the Vanderbilt Institute of Chemical Biology, the Chemistry-Biology Interface program, and the National Science Foundation Graduate Research Fellowship Program for funding my research during this time.

I would also like to thank my mother and my father, who raised me to have a scientific mindset and a curiosity about how things work; I would not have been able to do this without them. I would like to thank my brothers who were both friends and sources of inspiration for me. Most of all I would like to thank my wife, Amanda, who gave me a tremendous amount of support during graduate school. She was the one who listened and helped when I ran into the many hurdles that inevitably come up in projects in the fields of science.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Development of BCL::EvoGen, A *De-novo* Algorithm for Focused Library Design

1.1 Introduction and Background

Computer-aided drug discovery (CADD) is a broad term that represents the use of computational power to enhance the drug and molecular discovery process, and made its appearance shortly after the first computers were available to researchers [1]. In recent years, the use of CADD techniques has increased with the advent of cheap and widespread computer power and the public availability of biological data through sites such as PubChem and ChEMBL [2][3]. These databases have allowed for the development of new techniques and solutions to drug discovery problems that could not have been addressed before. Computer-aided drug discovery has played a role in the discovery of a number of pharmaceutical candidates and approved drugs including dorzolamide, captopril, saquinavir, and others [4][5]

Ligand-based CADD (LB-CADD) methods are a subset CADD methods which rely on knowledge of small molecule ligands for a biological target. LB-CADD methods are based on the *similar property principle* which postulates that structurally similar molecules are expected to exhibit similar properties [6]. For this reason LB-CADD methods are sometimes considered an indirect method for predicting biological properties since explicit interactions with biomolecules are not modeled [7]. This is in contrast to structure-based CADD methods (SB-CADD) which require knowledge of the structure of the biological target of interest and use explicitly-modeled interactions for evaluation. LB-CADD methods are useful when little structural information about a biological target is known, and especially when obtaining this information is difficult, such as for many membrane-bound proteins. An additional advantage of LB-CADD methods is that they are often faster than SB-CADD methods, and are therefore often used to screen databases of hundreds of thousands or mil-

lions of compounds [8]. While it is not clear whether either class of CADD method is superior to the other, there are many cases where LB-CADD methods have proven more effective than SB-CADD methods for discovering quality small molecule ligands despite their indirect predictive nature [9][10].

1.1.1 Encoding Molecular Information

Central to ligand-based drug discovery methods are molecular descriptors. The term ”molecular descriptor” is a generic term used to refer to different methods of numerically encoding molecular features which may have varying levels of complexity and can be structural or physicochemical in nature. Over the years a wide variety of descriptors have been proposed, all of which seek to balance information content with calculation and storage efficiency. The information that can be encoded by molecular descriptors can include molecular shape, volume, surface areas, inter-atomic distances, electronegativities, partial charges, presence of substructures or functional groups, and many other properties. A number of reviews and comprehensive volumes on the subject of molecular descriptors have been published [11][12] and development of new descriptors is an ongoing area of research.

Descriptors may be derived from a number of sources including empirical data, graph theoretical methods, or computational simulations. In addition, descriptors can be classified according to the dimensionality of the information that they encode. 1D descriptors include single-valued quantities that describe the entire molecular structure such as molecular weight or volume. 2D descriptors are derived from topological features of the molecule such as atom connectivity; this can be further extended to 2.5D descriptors which also take into account stereochemistry. 3D descriptors encode information about the molecular geometry which may be derived from relative spatial arrangements of atoms [13]. Oftentimes 2- and 3-dimensional descriptors will use lower dimensional information to weight their results. For example 2- or 3-dimensional distribution of partial charges around a molecular

structure can be computed in this manner. Higher dimensional descriptors, including 4-dimensional descriptors, have also been proposed which account for non-static molecular features, such as by encoding information across multiple conformations for each molecule [14].

One of the most popular class of molecular descriptors are substructure fingerprint descriptors. These descriptors encode molecular substructures as either a bit string or (less commonly) a vector containing fragment counts across a molecule. The actual method for encoding these can vary, and range from detecting the presence of each a fixed set of pre-determined substructures (such as for MACCS keys [15]), or may be generated based on the chemical structure of query compounds (such as Daylight or Morgan fingerprints [16]). These classes of fingerprints are often used for similarity comparisons of molecules [17]. Since fingerprint descriptors rely on direct substructural features of compounds, they can be much easier to understand and visualize relative to physicochemical or geometric descriptors.

1.1.2 Ligand-Based Scoring Functions

1.1.2.1 Similarity

The simplest of metrics used to score compounds are those based on similarity of compounds. These metrics usually use one or a few compounds as templates, and a similarity value is computed between these template molecules and molecules of interest. Similarity methods are most often coupled to fingerprint descriptors, with the similarity value often being calculated as a Jaccard index or Tanimoto coefficient between the bit strings U and V of the different molecules, given by [17]:

$$J(U,V) = \frac{|U \wedge V|}{|U \vee V|} = \frac{|U \wedge V|}{|U| + |V| - |U \wedge V|} \tag{1.1}$$

When used with multiple template models, a number of strategies have been used to

calculate the final similarity measure including score summation, rank summation, statistical Z-scores, and group fusion [17],[18],[19]. While straightforward to implement and interpret, fingerprint similarity metrics do not take into account spatial information about a molecule which may be important for binding to a receptor, and detecting novel classes of ligands may be difficult due to a dependence on specific atom identity and connectivity [20]. Despite this, 2D fingerprint similarity measures are still one of the most popular methods for comparing molecular similarity for virtual drug discovery.

1.1.2.2 Pharmacophore Mapping

The IUPAC has formally defined a pharmacophore as "an ensemble of steric and electronic features that is necessary to ensure the optimal supramolecular interactions with a specific biological target and to trigger (or block) its biological response" [21]. In other words, a pharmacophore can be considered as the spatial arrangement of functional groups in a molecule that confer biological activity, which are features that medicinal chemists often use to direct synthesis during medicinal chemistry campaigns.

Computational pharmacophore mapping is a technique wherein the structural features of biologically active compounds are compared to elucidate the important pharmacophore features. Ligand-based pharmacophore mapping techniques often rely on the superimposition and alignment of several known active compounds to generate pharmacophore hypotheses, usually with an attempt to balance specificity of the model with its generalizability [13]. Molecular structures may be rigidly or flexibly aligned for pharmacophore mapping, with methods employing the former approach often much faster than their flexible counterparts, but at the cost of reduced accuracy. Pharmacophore mapping methods often employ property-based matching using molecular fields to determine the optimal alignment of two molecules [22]. Pharmacophore methods offer an advantage over simple similarity comparisons by simultaneously leveraging the information from multiple sources to arrive at a relatively abstract consensus model which may be more informative than a simple fin-

gerprint. In particular, pharmacophore models are capable of matching abstract molecular features and do not necessarily rely on strict topological similarities, which makes them useful for scaffold hopping in medicinal chemistry projects [23].

PHASE is a commonly-used field-based pharmacophore mapping program which uses explicitly modeled pharmacophore centers to determine molecular similarities [24]. PHASE generates multiple conformations of each molecule in an active ligand set and then searches these conformations for common pharmacophores based on geometrical distances. PHASE was recently used to discover a novel class of nicotinamide phosphoribosyltransferase (NAMPT) inhibitors which have clinical relevance to cancer treatment and for inflammation [25]. In this study a series of LB- and SB-CADD methods were used to prioritize a compound databases containing between 750,000 and 3 million compounds to select 102 compounds for testing. This screening campaign resulted in the discovery of two novel series of NAMPT with potencies in the nanomolar range.

Surflex-Sim is a shape-based pharmacophore matching method that leverages molecular fields which encode hydrophobic and electrostatic features of molecular surfaces to align query molecules [26]. The Surflex-Sim method first generates hypotheses using a small number of known active compounds which are flexibly aligned so as to optimize the overlap of surface features of these molecules. These hypotheses are then held static, and query molecules are aligned in a similar fashion to the hypotheses. Surflex-Sim was used in the discovery of T-type calcium channel inhibitors for scaffold hopping and potency improvement from known structures [27]. In this study, Surflex-Sim was used to align a known compound from Merck and a preliminary hit compound from a medicinal chemistry campaign. The alignment identified suboptimal features of the hit compound and provided key insight that allowed the design of a new structurally novel compound with high potency.

1.1.2.3 Quantitative Structure-Activity Relationship Modeling

Quantitative structure-activity relationship (QSAR) models are mathematical descriptions of relationships between molecular features or descriptors and the physicochemical properties associated with them. The Hansch-Fujita approach is one of the earliest examples of QSAR modeling, wherein various electronic, steric, and hydrophobic features of molecules were used to predict biological activities [28]. Many other techniques which have combined a wide range of molecular descriptors have been proposed since these early approaches, and today QSAR models may be built using pharmacophore maps and other straightforward approaches up to complex mathematical models requiring large amounts of computational power.

Machine learning (ML) involves the use of mathematical and computational algorithms to extract meaningful information from data without human intervention [29]. In recent years, machine learning has gained popularity in the fields of computer science, statistics, and data science as many analysis techniques were developed to deal with the vast amounts of data in these fields. Within the context of drug discovery, machine learning is frequently used to discover and examine correlations between molecular features and biological activity, and to compare groups of molecules with similar properties [30][31][32][33].

Supervised machine learning methods are techniques wherein a limited amount of data is used to tune a mathematical model to approximate a desired function space or task. Learning can be accomplished as either classification (i.e. predicting active or inactive) or regression (e.g. predicting pIC_{50} values) depending on necessity. In the field of drug discovery these methods can be used to predict the biological activities or other physicochemical properties, such as toxicity and metabolism, of a given chemical structure [34]. These methods have come to the forefront as potentially some of the most promising tools in the computational drug discovery arena [35].

Neural network models were used to discover novel metabotropic glutamate receptor 5 (mGlu5) negative allosteric modulator (NAM) compounds using information from

a high-throughput screen for the same target [36]. In this case the neural network was used to screen a database of around 700,000 commercially available compounds, of which 749 were ordered and experimentally tested. From these experiments, two sub-micromolar compounds with novel structural features were discovered and used for a subsequent medicinal chemistry campaign.

The Merck Molecular Activity Challenge was a Kaggle competition hosted in 2012 to determine the most promising method for ligand-based activity prediction [37]. The winner of the contest used a combination of a deep artificial neural networks and gaussian-boosted decision tree ensembles for their predictions [38]. Interestingly it was found that multi-task neural networks which predict on many outputs simultaneously tended to perform better than combining the predictions of individual neural network models. The use of this combination of models resulted in a set of predictions which exceeded the baseline predictive ability provided by Merck by almost 15 percent and illustrated the power that deep learning methods may have for drug discovery.

1.1.3 *De-novo* Drug Design

De-novo molecular design algorithms are computational routines which search chemical space for chemical structures which are likely to have desirable pharmacological or physicochemical properties. In recent years some attention has been paid to the development of these methods as a non-traditional way of exploring chemical space. Chemical space has been estimated to contain somewhere between 10^{60} to 10^{100} chemically feasible drug-like molecules, whereas the largest available compound libraries used for traditional high-throughput screening (HTS) contain somewhere on the order of 10^9 compounds [39]. *De-novo* design methods can be used to close this large size gap between possible and available chemical compounds, with the *de-novo* design acting, in essence, as a complementary method to traditional HTS to explore chemical space that would otherwise not be possible.

De-novo design methods require at least two components: a scoring function to evaluate

the quality of a proposed molecule, and a structural modification routine. Some of the earliest *de-novo* design methods used atom-based structural modification methods to build compounds one atom at a time [40][41]. While these methods are theoretically able to sample all possible chemical space, they also face enormous challenges resulting from a combinatorial explosion of possibilities. A specific example of this is the difficulty in designing chemically feasible and drug-like molecules, especially when extra constraints are not put in place to prevent this [39].

Fragment-based methods are a superset of atom-based methods, and have been used to overcome some of the biggest problems faced by atom-based approaches. Instead of modifying chemical structures by single atoms, fragment-based methods append groups of atoms (consisting of single atoms in some circumstances) to other structures. Use of these fragments has the effect of encoding some pieces of chemical knowledge and intuition and are better suited to the generation of chemically feasible molecules than single-atom modifications. In particular these additional rules have been used to account for the challenging prospect of designing drug-like and chemically feasible compounds, which remains one of the biggest challenges for *de-novo* design methods [39][42]. All recent *de-novo* methods use a fragment-based modification approach in some form or another, all of which seek to implicitly encode chemical knowledge and improve the quality of designed molecules using these more advanced methods.

One example of a method designed to improve synthetic accessibility of compounds is the retrosynthetic combinatorial analysis procedure (RECAP) [43]. RECAP uses molecular fragments or ”building blocks” which are chosen by fragmenting a set of known molecules using chemical knowledge. A set of 11 bond types which correspond to easily-formed chemical bonds were selected as the fragmentation criteria, though the authors explicitly noted that the method was designed such that this list could be modified. These fragments can then be combined together to form new structures with a higher synthetic accessibility than naîvely designed molecules. This technique has grown in popularity and has been

ported to modern cheminformatics platforms such as RDKit [44].

Another class of fragment-based modifications that have recently gained attention are those which use reaction-based rules for chemical modification. In addition to ensuring that designed compounds have reasonable chemical structures, reaction-based modifications have the added advantage that they provide an explicit (albeit putative) synthetic pathway for the generated compounds. A notable example of a program which uses this approach is the DOGS algorithm [45], a *de-novo* design algorithm aimed at suggesting drug-like compounds for drug discovery campaigns. Structural modification in the DOGS algorithm proceeds first via a reaction-search stage to determine the most promising possible modifications for a molecule using a set of reactions and dummy functional groups. Once the best reaction for a given step is chosen, it is followed by an exhaustive enumeration of the space represented by reacting all candidate. Notably, DOGS has been successfully applied as part of a pipeline for the discovery of novel Polo-like kinase 1 inhibitors [46].

1.1.4 The BioChemical Library and the EvoGen Algorithm

Reaction-based *de-novo* design methods are a promising avenue for exploring synthetically feasible chemical space in a rapid and efficient manner. Reaction-based methods are ideal for providing a tentative synthetic pathway to access molecular structures of interest, a feature that could help medicinal chemists judge the feasibility of synthesizing a molecule.

The BioChemical Library (BCL) is a scientific C++ library developed at Vanderbilt University which includes capabilities for protein folding and cheminformatics. One of the main goals of the BCL is to provide an integrated environment for CADD methods. A stochastic *de-novo* design algorithm named the BCL::EvoGen has been added to the BCL to integrate *de-novo* drug design with other advanced features of the library. This algorithm is capable of leveraging the internal functionality of the BCL to automatically generate conformers of designed molecules, calculate physicochemical properties, and predict biological activities using several modeling capabilities.

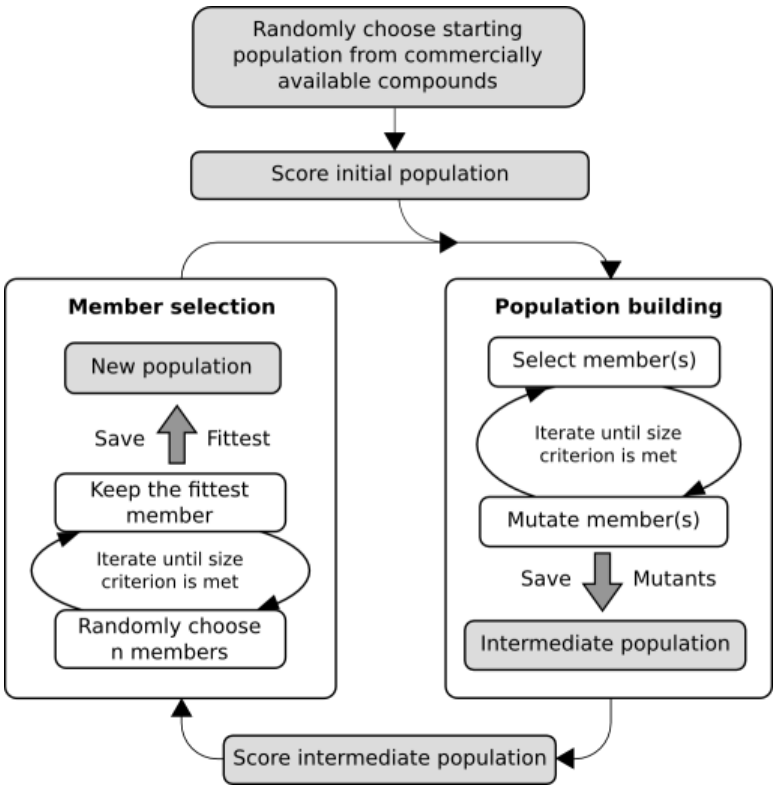


Figure 1.1: Diagram of EvoGen operation. The algorithm begins by selecting a random set of compounds from a user-provided library and subsequently scoring them to provide a starting point for the algorithm. The construction loop then either uses candidate molecules from the previous iteration to build new structures preferentially using high-scoring structures from the parent population or inserts new building blocks from an external library. These structures are then scored, and down-sampled to the appropriate population size via tournament selection. This process repeats until a pre-determined number of iterations has been reached.

1.2 Results and Discussion

1.2.1 EvoGen Algorithm

The EvoGen algorithm incorporates reaction-based structure modification into a stochastic search algorithm to enable the automatic generation of drug-like focused libraries. The algorithm consists of an initialization and setup phase, a structural modification and scoring loop, and termination phase once run criteria have been met (Figure 1.1).

The algorithm reads four data sources on startup: a reaction library and corresponding

reagent library, a compound library for future additions, and an initial set of molecules that can be used to start the algorithm. In most cases the last two files will contain the same set of molecules. In addition, the desired population size, number of iterations to execute the algorithm, and the desired set of models to use for scoring must be be provided.

At startup, the EvoGen algorithm reads the initial molecule set and, in the event that it contains more molecules than the population size, the initial population of molecules is randomly sampled from this set. The scoring function (described below) is initialized using the provided models and the weighting function described above. The scoring function is then used to score the initial set of molecules to provide a starting point for the iterative loop.

The main part of the algorithm is the construction loop which executes until the specified number of iterations have transpired. This loop consists of three phases: candidate generation, candidate scoring, and candidate selection.

During candidate generation and candidate selection, compounds from the previous or current population must be chosen for modification or selection, respectively. Tournament selection was implemented as an efficient method to probabilistically select compounds based on their fitness scores. Tournament selection randomly selects a percentage of the full population, assigns a probability of selection to each member based on its fitnesses relative to the subset, and then selects the n-th compound probabilistically. Tournament sizes are specified as the percentage of the whole population that should be considered for a single tournament round, and are therefore always numbers between 0 and 1.

Candidate generation is the sampling phase of the algorithm wherein new chemical structures are generated as derivatives of parent molecules in the previous generation, or by including compounds sampled from the addition library. The number of candidate solutions is by default ten times the desired population size to provide a large sampling space. The two operations that are possible during this step are the reaction and the addition operations.

The reaction operation uses one molecule selected from the parent generation, selects

a random reaction and set of reaction partners, and executes the reaction to build a new candidate compound. The addition operation selects a random molecule from the addition database and treats it as a new candidate solution. Compound scores are not used for the selection in this step since each selected compound will be scored (and possibly eliminated) in a later step of the construction loop. The purely random selection therefore should not greatly affect the search behavior of the algorithm when high-scoring candidates are present, but provides an opportunity to sample diverse chemical space during periods where only low-scoring solutions are present. During any of these operations it is ensured that no duplicate compounds are generated so that each newly generated population consists of unique compounds, though it is possible for the same molecule to be generated multiple times throughout the run.

At the end of the candidate generation step, compounds from the parent generation may be included with the newly generated candidate compounds based on a retirement policy which specifies which parents are to be included. In the current algorithm there are three available policies. The two straightforward implementations are the **A**ll policy (abbreviated as policy A), wherein all parents are discarded after a single iteration, and the **N**one policy (policy N) wherein no parents are discarded (i.e. all parent molecules are included as candidate compounds). The **P**robabilistic policy (policy P) provides a trade-off between the two extremes of the A and N policies, and discards parent compounds probabilistically based on their age, i.e. how many iterations the compound has been present. The probability of discarding a parent compound is given by the equation:

$$p(a) = 1 - exp\,(-Ca) \tag{1.2}$$

Where a is the compound's age in generations, and C is a constant used to adjust the expected lifetime of a single compound. A value of -0.5 was chosen for C so that the expected lifetime of compounds in these studies were approximately 4 generations. This value was chosen since it would allow for several generations where high-scoring compounds could

be optimized to search the local space around apparently good candidate solutions, while still forcing compound turnover relatively frequently to avoid stagnation. In addition, the score of the compound is not considered in this calculation because, during later steps, low-scoring compounds will be pruned from the population anyway; adding an additional term for the compound's score would do little to improve the selection while introducing another degree of freedom into the algorithm.

Following candidate generation, *candidate scoring* uses the combination of raw scoring function and the weighting function described above to assign fitnesses to each candidate solution. During this step, all molecules are also preprocessed to match requirements to be scored using the models. In the studies presented here this involved ensuring that a single low-energy three-dimensional structure was generated and hydrogens were added to the compounds (see section 1.3.4).

The final phase of the construction loop is *candidate selection* in which the full set of candidate solutions (including those from the parent generation) is downsampled to the appropriate population size for use in the next iteration of the construction loop. Note that this step does not differentiate between newly generated solutions and those included from the previous generation. The downsampling is done using tournament selection to choose an appropriate number of compounds from the oversampled set. Once these compounds have been chosen, the newly pruned population is written to a file and is then used as the input for the next iteration of the loop. This process then repeats for a fixed number of iterations.

1.2.2 Reaction Library

The reactions used in the EvoGen algorithm were compiled from a combination of literature sources including [47] and [45], and from in-house medicinal chemistry knowledge. A total of 93 reactions were selected based on their frequency of use and their ability to introduce important structural features such as heterocycles. These reactions included link-

Property	Mean	Std. Dev.
Weight	194	58.6
Heavy atoms	12.8	3.5
Girth[a]	8.0	2.1
LogP[b]	1.5	4.9
Rotatable bonds	2.7	2.0
H-bond acceptors	3.0	1.6
H-bond donors	1.0	1.0
Complexity[c]	0.32	0.26

Table 1.1: Calculated properties for the EvoGen reagent library.
a) Girth is defined as the largest distance between two atoms of the molecule
b) Calculated LogP using an atom-based code, approximately 3% of the library has a LogP below -5
c) Complexity measure is similar to that used in [49]

ing reactions such as amide and sulfonyl chloride couplings, cross-coupling reactions such as the Suzuki and Sonogashira reactions, and heterocycle formation reactions.

1.2.3 Reagent Library

The reagent library was obtained using a list of commercially available compounds from the Sigma Aldrich building block catalog deposited in the ZINC database [48]. The full library contained approximately 72,000 small molecules. Filtering was performed to remove any compounds that did not match at least one of the reactions from the reaction library, had more than 3 and fewer than 20 heavy atoms, contained no permanently charged groups (not counting net-zero charged groups such as NO_2), and did not contain aliphatic chains over 6 carbons long. These filtering steps resulted in a final set of approximately 26,000 small molecule building blocks. A summary of different molecular properties for these building blocks is given in Table 1.1

1.2.4 Scoring Function Design

It was found early on in testing the EvoGen algorithm that it was necessary to place some conservative restrictions on specific molecular features in order to aid raw scoring functions that don't implicitly include such restraints in order to improve the drug-likeness of results. It was found in early tests that certain scoring criteria would give impractically large molecules with favorable scores, which could then accumulate in populations and make the affected runs effectively useless. To combat this, a slotted weighting function was used to scale raw scores according to molecular weight given by equation 1.3:

$$f(w) = \begin{cases} 0 & w < 100 \\ \frac{1}{2}\left(1 - \cos(\frac{\pi(w-150)}{50})\right) & 100 < w < 150 \\ 1 & 150 < w < 550 \\ \frac{1}{2}\left(\cos(\frac{\pi(w-550)}{50})\right) & 550 < w < 600 \\ 0 & 600 < w \end{cases} \tag{1.3}$$

Where w is molecular weight. The output of this function is multiplied by the raw model score to provide a corrected, optimization-appropriate score. The multiplier effectively decreases a compound's score once the molecular weight falls outside a preferred region of 200-550 Da. Once the weight drops below 150 Da or exceeds 600 Da the score will be set to zero. These values were chosen because very small compounds are not likely of interest to drug discovery scientists and very large molecules will be difficult to synthesize and are unlikely to be truly synthesizable or active. A hard weight cutoff was also tested, but this proved too severe a restriction as very similar molecules would have drastically different scores by virtue of one exceeding the weight threshold by only a few Daltons.

1.2.5 Quantitative Structure-Activity Relationship Models

Machine learning QSAR models consisting of artificial neural networks were built to benchmark the EvoGen algorithm. Inhibitors of cyclin-dependent kinase 2 (CDK2) and metabotropic glutamate receptor 5 (mGlu5) NAMs were used as the two biological targets for these studies. This choice was based on the availability of a large amount of data for these targets, and the distinct cellular location of the two proteins. CDK2 is a soluble protein that is found in the cytosol, whereas mGlu5 is a membrane-bound G-protein coupled receptor. Data for mGlu5 was obtained from in-house medicinal chemistry sources, and the CDK2 data was obtained from publicly available sources in PubChem, ChEMBL, and literature [3],[50],[51],[52]

Performance metrics and receiver-operator characteristic (ROC) curves of the cross-validation results are shown in Figure 1.2. The integrated areas under the curve (AUC) of the ROC curves and the average enrichment values indicate that these models are capable of prioritizing active compounds over inactives at a rate substantially higher than random chance.

1.2.6 Analysis of Active Compounds

In order to evaluate the best similarity metric that could be used to compare *de-novo* designed molecular structures with actives for the different targets, the known active compounds for both CDK2 and mGlu5 NAMs were compared against both themselves and against random compounds sampled from the ZINC drug-like dataset [48]. For a set of N actives for a single target this would result in an Nx(N-1) similarity matrix. When M random molecules are considered this would result in an MxN similarity matrix.

Several statistical measures were calculated using these data, including maximum similarity, mean similarity across all known actives, and mean similarity of the top 10 most similar known actives. Upon inspection of density plots of the different statistical mea-

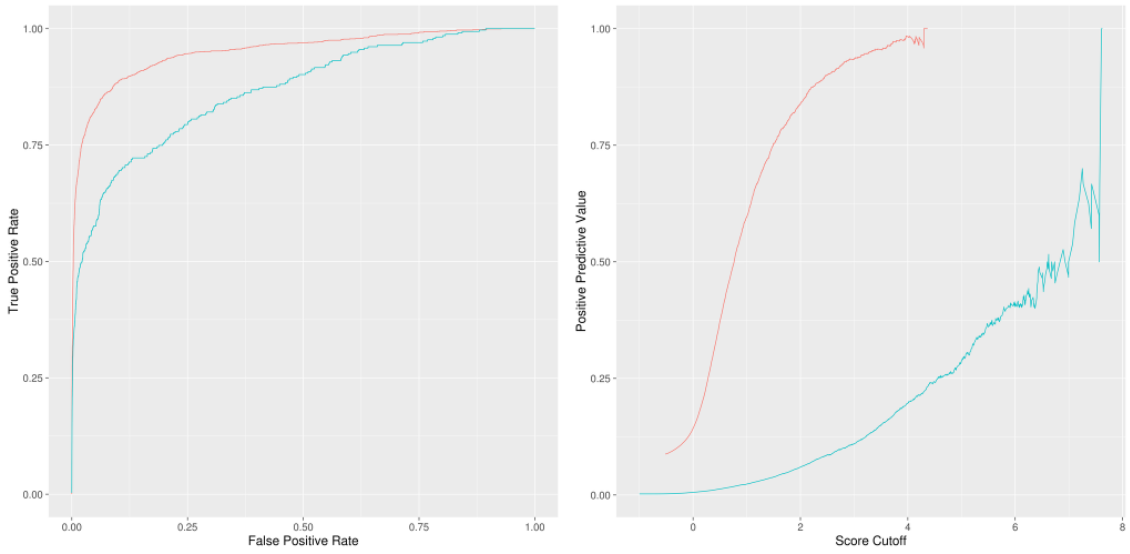


Figure 1.2: ROC curves (left) of CDK2 (red) and mGlu5 models (blue) and comparison of score cutoff values versus positive predictive value (right). Area under the ROC curve, CDK2: 0.95; mGlu5: 0.85. Avg. enrichment up to 10% FPR, CDK2: 10.2; mGlu5, 20.3

sures it was found that the behavior for the CDK2 and mGlu5 targets differed substantially (Figure 1.3). Whereas the CDK2 dataset showed a difference between actives and random compounds for almost every metric, it was found that the only metric that gave a substantial difference between known actives and random compounds was maximum similarity.

Maximum similarities are relatively narrow for the CDK2 dataset, with most compounds showing similarity Tanimoto scores of between 0.8 and 1.0, with a maximum density occurring at around 0.95, indicating that most compounds in the dataset are very similar to at least one other compound. This can be contrasted with the random compound similarity distribution with a narrow distribution peaking around 0.47 and very low densities past scores of 0.6. The mGlu5 data shows a more diffuse similarity profile, with active compounds showing a bimodal distribution with maxima around 0.52 and 0.86 and a wide score distribution surrounding each. Random compounds show a maximum similarity to known actives at 0.5, indicating that there is a portion of the mGlu5 actives dataset that is largely indistinguishable to random compounds in terms of its 2-dimensional profile. From

these data it can be concluded that similarity values above 0.7 will represent compounds that have a substantial active-like topology for either dataset, whereas values between 0.4 and 0.6 are reasonably likely to represent non-active-like compounds.

1.2.7 Random Sampling For Baseline Comparison

Random compounds were sampled from the ZINC [48] Drug-like database and were scored using the same models used for CDK2 and mGlu5 *de-novo* runs. Similarities to known active compounds were calculated using RDKit [44]. The absolute behavior of these plots varies somewhat between the two models but similar trends are observed.

Repeatedly sampling random compounds and tracking the cumulative best-scoring compounds for both targets reveals trends that rapidly increase over the first few samples followed by a plateau with periodic but irregular increases in score. The CDK2 scores differ from the mGlu5 scores in that score increases are generally much larger and it takes little time to reach a global maximum, whereas mGlu5 scores continue to increase regularly across all 100 samples (Figure 1.4). This further indicates that mGlu5 provides a particularly rigorous target for these methods, as high scores are uncommon when randomly sampling drug-like compound sets. Interestingly, for either target the average similarity of the top-scoring molecules does not increase monotonically with the score values, indicating that simple 2-dimensional similarity cannot fully account for the favorable scores of these compounds and that they possess some degree of novelty relative to the known active compounds.

1.2.8 Molecular Design Benchmarking

In order to determine obvious patterns which arose from different choices of parameters, a grid search over the tournament sizes and the parent retirement policies was done using a single run for each set of parameters and model set. Tournament values of 0.1, 0.5, and 0.9 (representing the sampling percentage used for each tournament round) were

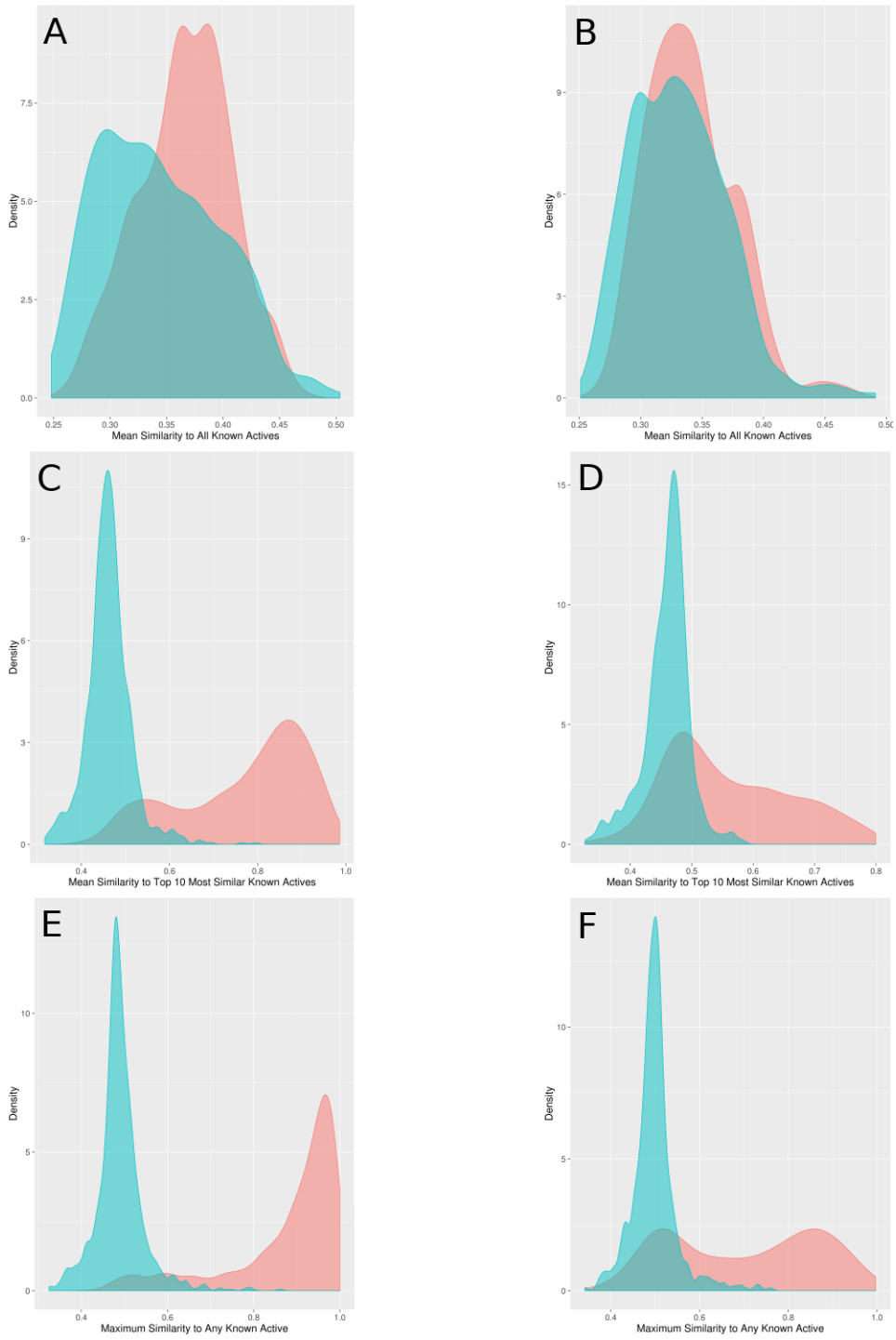


Figure 1.3: Density plot comparison of known active and random compounds to other known actives for CDK2 and mGlu5 targets according to metric. Red: known actives to other known actives. Blue: Random compounds to known actives. A) CDK2, mean similarity to all known actives, B) mGlu5, mean similarity to all known actives, C) CDK2, mean similarity to top 10 most similar actives, D) mGlu5, mean similarity to top 10 most similar actives, E) CDK2, maximum similarity to any known active, F) mGlu5, maximum similarity to any known active. Maximum similarity to known actives is the only metric which shows significant signal for both CDK2 and mGlu5.

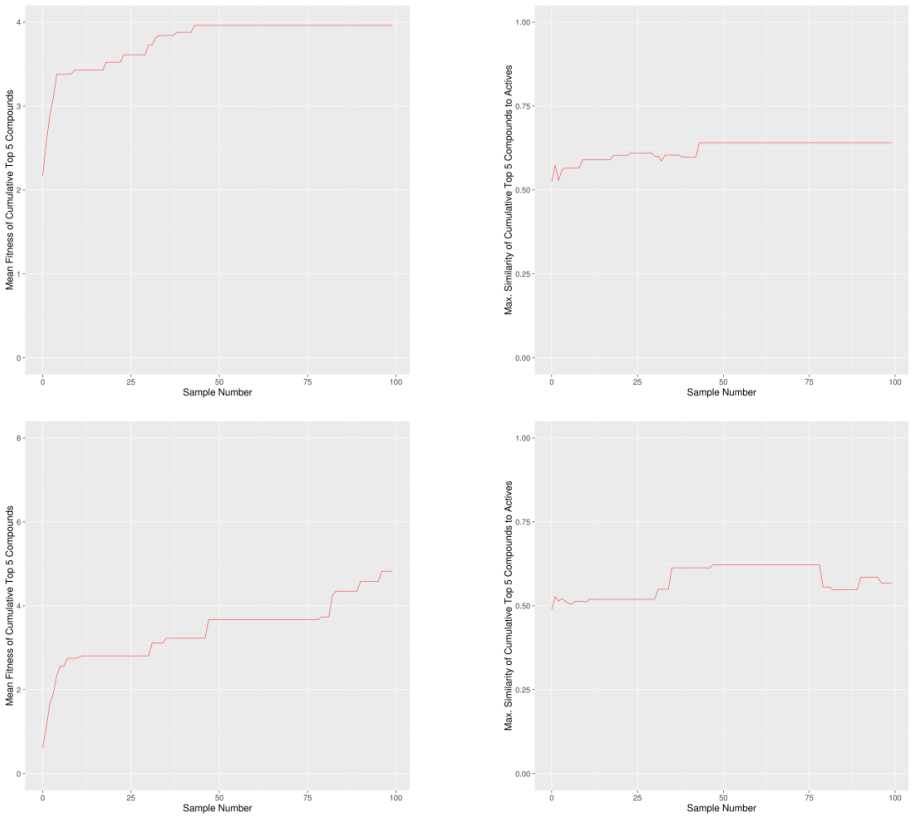


Figure 1.4: Cumulative fitness and similarity metrics using a growing number of randomly sampled compounds. Top row: comparisons to CDK2 dataset, bottom row: comparisons to mGlu5 data set. Left column) cumulative best fitnesses, right column) maximal similarity of the highest-scoring compound to any known active.

chosen to represent small, medium, and large tournament sizes, respectively. In addition, retirement policies A, N, and P were investigated. It was found that tournament sizes had an almost negligible effect on the quality metrics investigated, and the only major effect arose from parent retirement policies.

Based on these preliminary results, it was decided that retirement policy effects should be investigated more thoroughly. For these benchmarks, population sizes were held constant at 100 members, the optimizations were run for 100 iterations, and the tournament sizes were fixed at 0.5 for both selection and replacement. Each time the algorithm was run, it was done so with random starting point to ensure that general behaviors could be determined from statistical anomalies.

The overall performance of the algorithm under each parameter set was investigated to determine its fitness for focused library design. The metrics of interest were the number of active-scoring compounds and their relative percentage of the total number of unique compounds generated per run, fitness score distributions, and diversity measurements. Diversity was considered by comparing the 2-dimensional similarity of compounds within single runs and between multiple independent runs. The 2-dimensional similarities were calculated using the RDKit [44].

1.2.8.1 Characteristics of Active-Scoring Compounds

To assess the performance of the EvoGen algorithm with regards to its ability to generate compounds likely to be active, the EvoGen algorithm was run 10 times for each set of parameters, then designed compounds were classified as active-scoring if they achieved a fitness score higher than the median of active compounds during model training (Table 1.2). Using these cutoffs, all active-scoring molecules from each run were extracted, duplicates removed, and the groups were compared.

It should be noted that while the EvoGen algorithm will not include duplicate molecules within a single population, duplicate molecules may be seen between iterations. This can

Table 1.2: Median scores of active compounds during model training. These scores are used as the cutoff values above which *de-novo* designed compound are considered to have active-like scores

Dataset	Median Active Score
CDK2	3.05
mGlu5	1.74

be caused either by coincidentally forming a new molecule using reaction modification, or when a parent molecule is transferred to a subsequent population as may happen when using retirement policies N or P.

Figure 1.5 shows the distributions of unique and active-scoring compounds with respect to each retirement policy. Within these 10 runs, policies A and policy P generate roughly equivalent numbers of active-scoring compounds (Figure 1.5b), with policy A producing slightly more for the CDK2 dataset and policy P producing slightly more for the mGlu5 dataset. Policy N produces the fewest active-scoring compounds, at a mean value approximately 50 percent lower than the A and P policies. However, when considering the percentage of active compounds relative to the total number of unique structures generated per run, policy N produces a much larger percentage of active molecules (approximately 75 percent of cases), versus either policy A or P (between 10 and 50 percent, respectively, Figure 1.5c).

Figure 1.6 shows how the choice of retirement policy affects the number of unique structures on a per-iteration basis which can explain the above results. Policy A (not shown), by virtue of never saving parent molecules, always results in a 100 percent rate of unique compounds barring random chance duplication. However, policies N and P both start off with high structure generation rates which quickly decrease and plateau at low levels.

Regardless of retirement policy, the first iteration of a run will result in a number of compounds obtained by reaction-based modification. Given the statistically low scores of

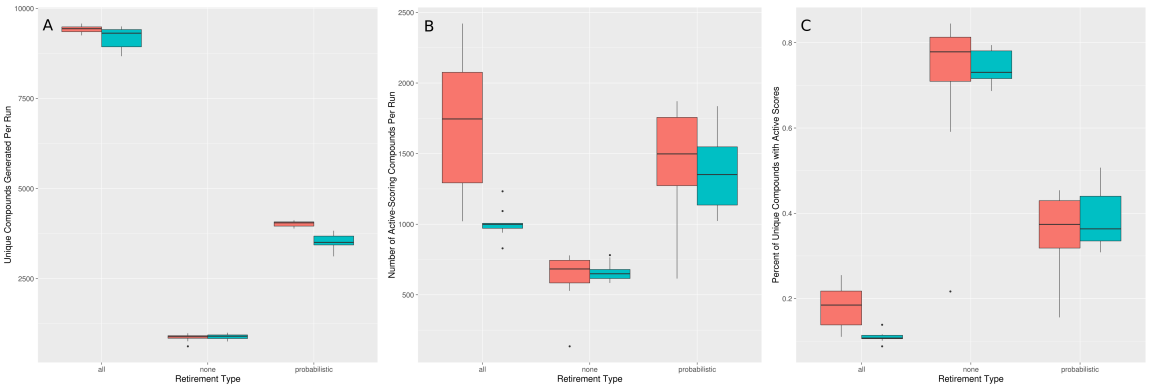


Figure 1.5: Metrics for different retirement policies for CDK2 (red) and mGlu5 (blue) datasets. A) Number of unique compounds generated per run, B) number of active-scoring compounds generated per run, C) percentage of unique compounds which are active-scoring.

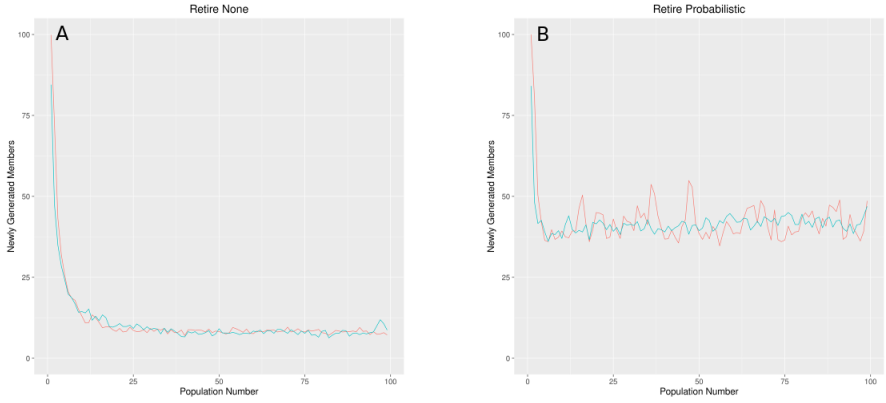


Figure 1.6: Number of unique structures per iteration for CDK2 (red) and mGlu5 (blue) runs using policy N (A) and P (B). The algorithm initially generates many compounds that are higher scoring than the random initial population. The score threshold for a molecule's inclusion in subsequent populations rapidly rises, which results in the rejection of most compounds and a plateau in the graphs. Policy A results in nearly all unique compounds from one population to the next and so is not included.

the randomly sampled starting populations and the fact that a large portion of the new compounds will result from the modification of compounds with the most favorable features, these newly generated compounds are likely to improve in score over the starting population and will be retained through the selection step. With policies N and P, once this process repeats, the subsequent newly generated compounds will also compete with high-scoring parent compounds from previous generations and will therefore have a reduced probability of continuing on than had they been compared against random starting populations. This means that as the algorithm progresses it becomes harder for structural modification to result in improvements, and therefore the number of unique compounds generated will drop quickly after the first iteration for both policies P and N. Policy P will eventually discard some of these very high scoring parent compounds based on their age, which will periodically give lower-scoring candidates a higher chance to survive the selection phase. This difference between policy N and P results in a larger steady state turnover for policy P and hence a larger number of unique structures generated overall.

The breadth of chemical space that was explored during each individual run was calculated by comparing similarities of active-scoring compounds with themselves. To calculate metrics for individual molecules, each molecule was compared against all other active-scoring compound from the same run, and the average value of the top 10 highest similarity scores was calculated. This metric was chosen over a simple maximum similarity since it considers many relationships between molecules simultaneously and is therefore more likely to reflect a true structural diversity across the whole dataset.

Figure 1.7 illustrates the relationships between fitness and this similarity metric. Policy A results in the highest spread in similarities (and therefore the highest diversity) and the lowest mean similarity between the three, followed by policy P, and then by policy N. Fitnesses for the different retirement types follow expected trends, with policy N producing proportionately more high-scoring compounds than policies N and P, though the numbers of compounds generated using policy N are much lower. Despite this, both policies A

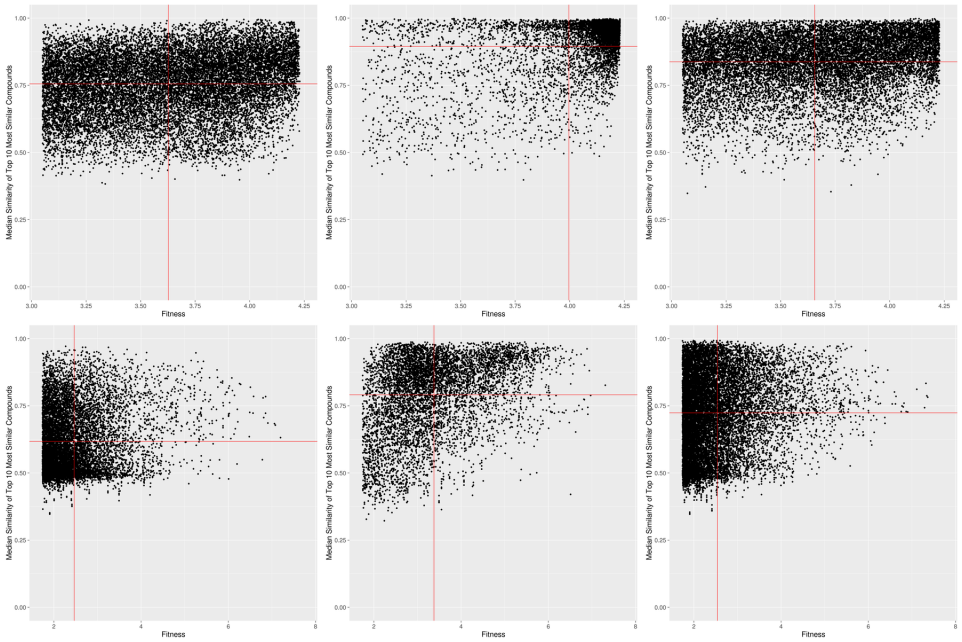


Figure 1.7: Intra-run similarities versus fitness for CDK2 (top row) and mGlu5 (bottom row). Left column) policy A, middle column) policy N, right column: policy P. Red lines indicate the mean value of fitness (vertical) or similarity (horizontal) for each graph.

and P can also produce compounds with fitnesses similar to those for policy N across both targets. The mean score values further show that policy N produces the highest scores on average, followed by policy P and then policy A. Similarities are also consistent with the relative turnover rates for the three policies from which one would expect policy A to have the highest diversity of the three policies. Interestingly there is only a slight correlation between 2-dimensional similarity and fitness, indicating that the models do not rely entirely on topological information for scoring compounds.

Another consideration was the ability of the algorithm to generate different solutions with different starting populations. Similar to the metrics used for single-run diversity measurements, inter-run diversity measurements were made by comparing active-scoring compounds of a single run with the active-scoring compounds of the other nine runs (Figure 1.8). Again, the top 10 highest similarity values per compound were averaged to provide a per-molecule similarity value. Interestingly, mean similarity values were not substantially

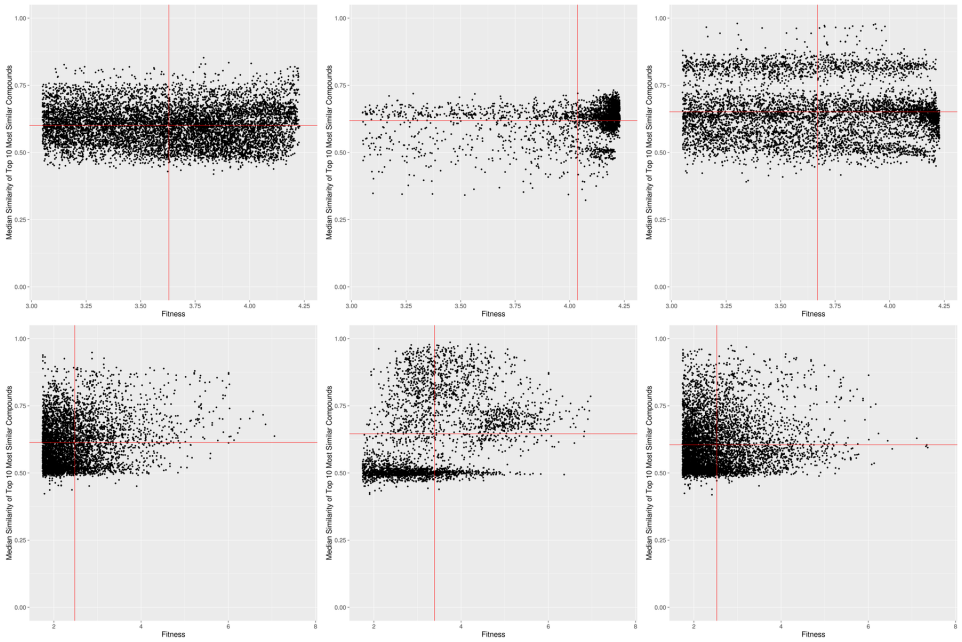


Figure 1.8: Inter-run similarities versus fitness for CDK2 (top row) and mGlu5 (bottom row). Left column) policy A, middle column) policy N, right column: policy P. Red lines indicate the mean value of fitness (vertical) or similarity (horizontal) for each graph.

different between the three retirement types, within a range of 0.60-0.65 for both CDK2 and mGlu5 datasets. This indicates that all three methods are equally good at generating diverse compounds when run up to 10 times. Interestingly, the difference in similarities of molecules between independent runs does not result in substantial changes in the mean score values of compounds, which would suggest that score distributions from a single run will reflect score distributions in future runs.

Based on the contrast in similarity values between compounds within single runs to those in multiple runs, it is beneficial to re-run the EvoGen algorithm several times to generate diverse sets of compounds. However, it is likely that chemical exploration will reach a finite limit once a certain number of repeats have been done. To determine where this point of diminishing returns begins, ninety additional runs were performed using each parameter set and the active-scoring compounds were collected. Active-scoring compounds from five randomly chosen runs were used as reference points, and the active-scoring compounds

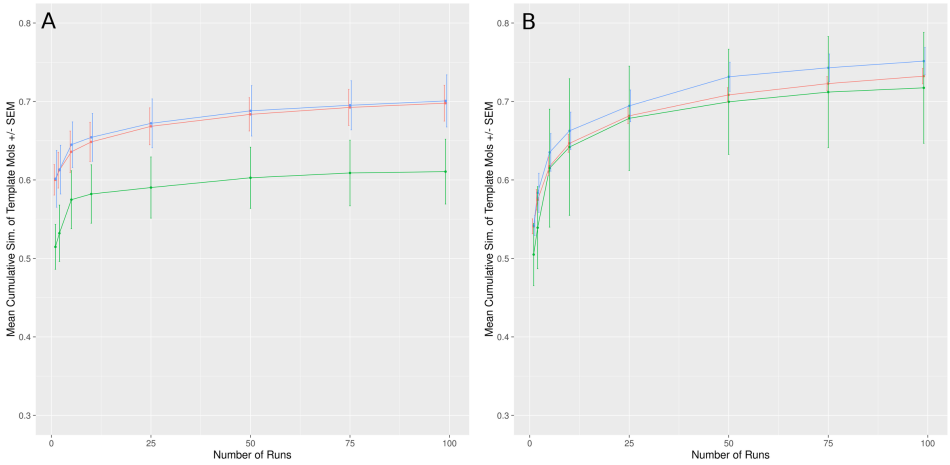


Figure 1.9: Relationship between number of runs and mean top-10 similarity metrics for active-scoring compounds for retirement policies A (red), N (green), and P (blue). A) CDK2, B) mGlu5. Average similarity metric reaches a plateau at around 50 populations independent of target or retirement policy, indicating this is likely universal behavior. All molecules after approximately 50 runs will likely be very similar to some of the compounds generated in earlier runs.

from 1, 2, 5, 10, 50, 75, and 99 populations were combined as comparison sets. Note that compounds were grouped in a cumulative fashion so that, for example, when compounds were grouped from 10 populations this included the same populations that were used when 5 populations were grouped plus five additional populations. When 99 populations were used, this included active-scoring compounds from all of the runs except for the one which was used for the reference compounds.

Figure 1.9 compares the mean value of the top-10 similarity metrics to the number of independent runs executed. Here it appears that there is a relatively rich exploration of chemical space early on regardless of retirement type as indicated by low similarity values. A point of diminishing returns occurs relatively quickly with very few novel structures generated after approximately 50 runs. Regardless of retirement policy, the mean top-10 similarity approaches a maximum at around 50 runs, indicating that at this point almost all of the compounds in the population are highly similar to several other compounds. This would imply that very little novel chemical space is explored past the 50-run mark, and re-running the algorithm past this point would likely be relatively unproductive. These

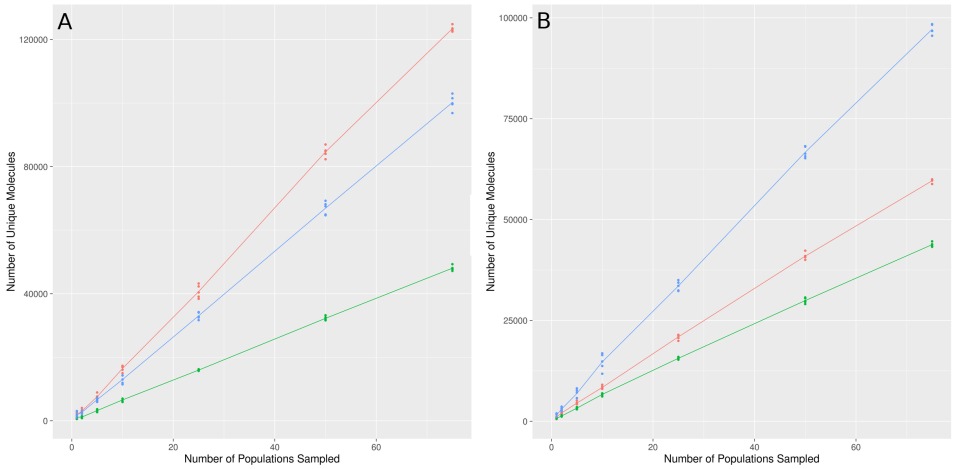


Figure 1.10: Plots of number of runs compared to the number of unique compounds generated for retirement policies A (red), N (green), and P (blue). A) CDK2, B) mGlu5. Number of unique compounds scales linearly with number of runs up to 75 runs independent of retirement policy. Policy N grows the slowest, with policy A and P producing compounds at a much higher rate.

results suggest that re-running the EvoGen algorithm with random starting points up to a few dozen times may be beneficial for exploring novel chemical space.

To further illustrate the claim that chemical exploration begins to repeat already-sampled space, it is useful to investigate the numbers of unique compounds generated for different numbers of cumulative runs (Figure 1.10). The sum total of unique compounds continues to climb in a linear fashion at least up to run 75 runs, indicating that the algorithm is not strictly regenerating previously discovered compounds. Since new compounds continue to be generated, if the newly generated compounds were substantially different from those that had been generated in other runs one would expect the similarity metrics shown in Figure 1.9 to stay relatively constant near the starting values. However, the increasing similarity metrics that are observed indicate that the algorithm will begin to generate many compounds that are highly similar to those seen in earlier repeats, and therefore exploration of novel chemical space will slow after several repeats.

1.2.8.2 Chemical Synthesizability of Active-Scoring Compounds

A major question regarding computationally generated chemical structures is whether they are synthetically feasible or not. A score for determining synthetic accessibility named SAScore [49] as implemented in the RDKit [44] was used to determine whether the generated compounds were synthetically reasonable. Active-scoring compounds from 10 of the EvoGen runs were combined and the resultant distributions of SAScores were plotted relative to known active compounds for both targets (Figure 1.11a and 1.11b). In addition, SAScore density maps of the top 10 percent of highest active-scoring compounds were also plotted to determine if there was a substantial difference in distributions as fitnesses increased (Figures 1.11c and 1.11d).

SAScores of all active-scoring compounds show a maximum densities between 3 and 4 for CDK2 (though there is an additional local maximum for policy P around 4.5), and around 3 for mGlu5. The density profiles shift somewhat toward lower values when only the top ten percent of fittest active-scoring compounds are considered, and this also removes the extra maximum in the CDK2 policy P data. Maximum densities for both targets differ from those for known actives by roughly a full SAScore point, indicating that the designed compounds are more complex than the known actives. According to the original SAScore benchmarks [49], SAScore densities for catalog molecules used for virtual screening experiments are highest at values around 3, and bioactive molecules between 3 and 4. Both comparisons presented here overlap well with these ranges and indicate that the molecules generated by the EvoGen algorithm are quantitatively similar to generic bioactive and catalog molecules despite the difference to known actives for each target. The disparity between SAScores of known actives and designed compounds indicates that further refinements of post-run filtering should be investigated to determine general methods for ensuring more target-specific molecular profiles can be built. Distributions of SAScore compared to compound fitness (Figure 1.12) indicate that these filtering criteria could potentially be done using SAScore itself, and also indicate that choosing easy-to-synthesize

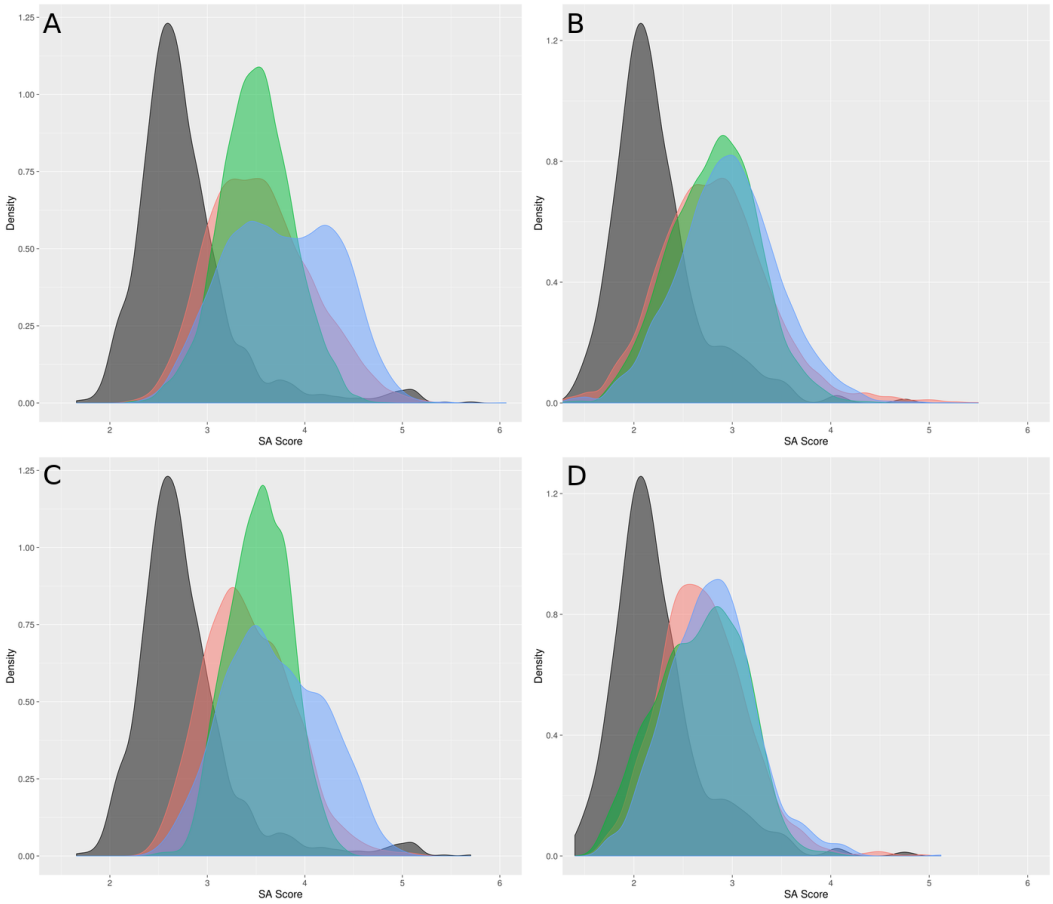


Figure 1.11: Density plots of SAScore for known active (grey) and designed molecules (Policy A: red, policy N: green, policy P: blue). Different plots show results for different targets and subsets of active-scoring compounds. A) CDK2, all active-scoring, B) mGlu5, all active-scoring, C) CDK2, highest 10% of active-scoring, D) mGlu5, highest 10% of active-scoring.

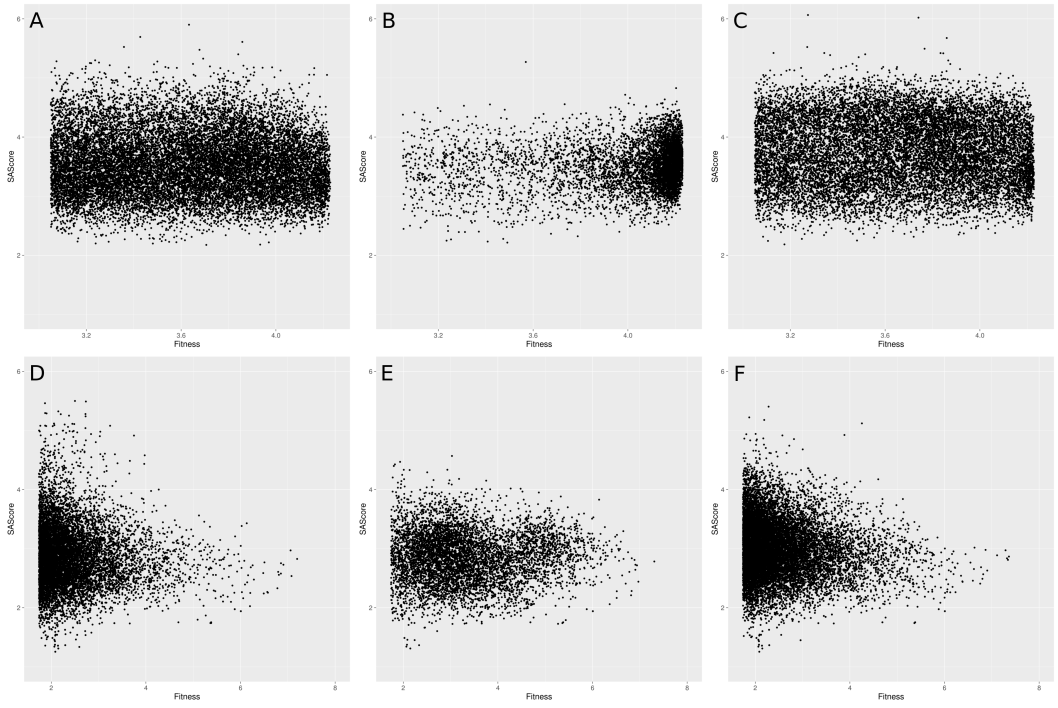


Figure 1.12: Comparisons of SAScore with fitnesses of active-scoring compounds for different retirement policies. A) CDK2, policy A, B) CDK2, policy N, C) CDK2, policy P, D) mGlu5, policy A, E) mGlu5, policy N, F) mGlu5, policy P. High-scoring compounds from all retirement policies across the different targets have a range of SAScores, indicating that picking out a subset of these compounds with high synthesizability should be possible.

molecules should not preclude the selection of high-scoring compounds.

1.2.8.3 Per-Population Compound Fitnesses

To characterize the way that the algorithm samples chemical space using each retirement policy, each of the 10 runs for each parameter set and target were investigated on a per-population basis. Mean fitness of the top 10 fittest compounds per population were calculated and inspected (Figure 1.13). Policy A resulted in the most sporadic behavior of the three policies, with scores fluctuating rapidly with both CDK2 and mGlu5 models. For the CDK2 runs, the scores quickly approached the saturation level of the model, but then promptly fell back to lower values. For the mGlu5 models no populations were able to attain a high-scoring level, though certain peaks during different runs were able to achieve

levels much higher than average. In addition, there was no upward trend in the baseline of the runs, indicating that while the algorithm was exploring chemical space, it was not performing optimization in any sort of targeted manner.

Policy N displayed complementary behavior to policy A, with an essentially monotonic increase in mean score for both CDK2 and mGlu5 datasets. This result was unsurprising given that it is very likely that the highest-scoring compounds will be passed from generation to generation regardless, with only a small chance of removal from the tournament selection procedure. The accumulation of high-scoring compounds illustrates how this mode effectively forces the algorithm to behave as an optimization algorithm. As a result of this behavior, increases in mean fitnesses were relatively small and resulted in a score plateau except during certain periods where large score jumps occurred over the course of a few iterations. In addition, there was a large disparity in behavior of individual runs relative to the average trend when using this policy. Four of the five runs shown in Figure 1.13 for CDK2 reach the saturation level of the CDK2 models, but a single run leveled off at a score value that was substantially lower than the other four runs. Similar behavior was true of the mGlu5 datasets, but with more pronounced differences between the runs. In some cases with mGlu5 the maximal scores differed by more than a whole score unit. This indicates that policy N will often trap the algorithm in a local optimum which is difficult to escape.

Policy P shows behavior that is intermediate between the A and N policies, with a somewhat dampened but sporadic behavior. Like the policy A, the mean fitnesses of the top 10 fittest compounds oscillated between peaks and troughs for both the CDK2 and mGlu5 datasets, but with a more smooth transition between each population. The peak values for the CDK2 datasets often reached the saturation level of the model, as they did using policy A. Unlike results from policy A, however, these peak periods lasted much longer, in some cases for 10-12 iterations. Similar trends were found with the mGlu5 sets as well, with smoother oscillations between high and low values than those seen with policy A. Similar to policy A, the policy P mean values were also relatively low compared

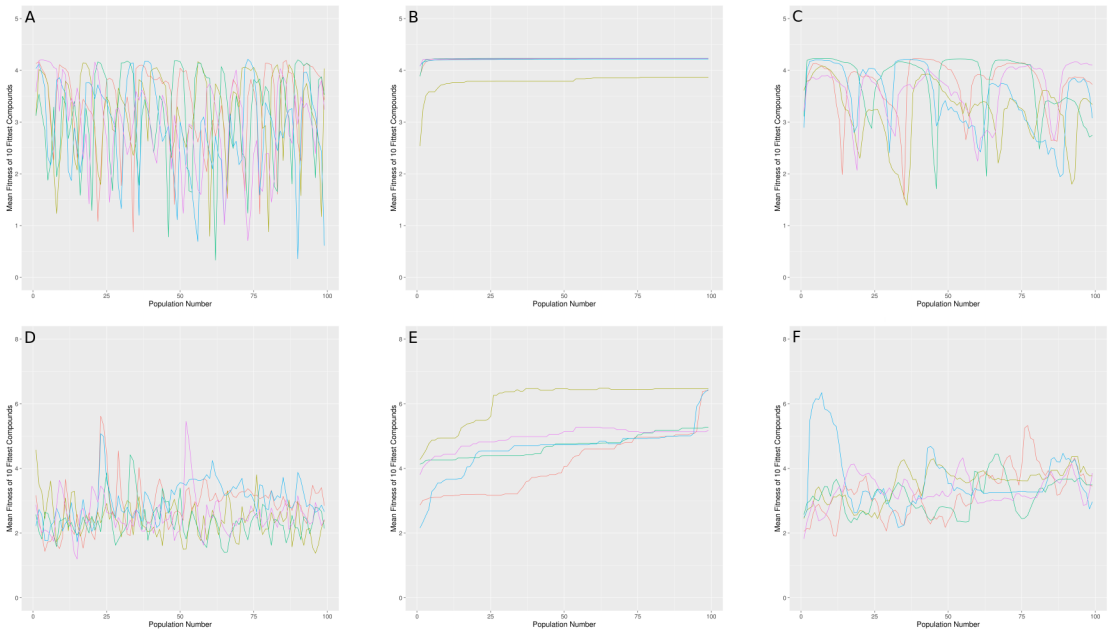


Figure 1.13: Mean fitness of the top 10 fittest compounds per population for CDK2 and mGlu5 targets according to retirement policy. A) CDK2, policy A, B) CDK2, policy N, C) CDK2, policy P, D) mGlu5, policy A, E) mGlu5, policy N, F) mGlu5, policy P. Colors represent different independent runs of the Evo-Gen algorithm. Policy A has the most sporadic mean fitness plots, and policy N has the smoothest. Policy P shows some sporadic behavior with relatively smooth transitions.

to those seen with policy N indicating that the majority of solutions had relatively low scores. Interestingly, policy P results also showed a globally upward trend in mean score as the algorithm progressed which is a distinct feature compared to policy A. The combination of this upward trend with the persistence of the sporadic fitnesses indicates that policy P is able to simultaneously optimize compounds for model score while maintaining an ability to explore chemical space outside of what would be accepted using a greedy search and thus did function as an effective intermediate to policies A and N.

1.2.8.4 Within-Population Diversity of Designed Compounds

Consideration of how much space is explored on a per-population basis was investigated quantitatively by calculating the similarity of each molecule in each population with

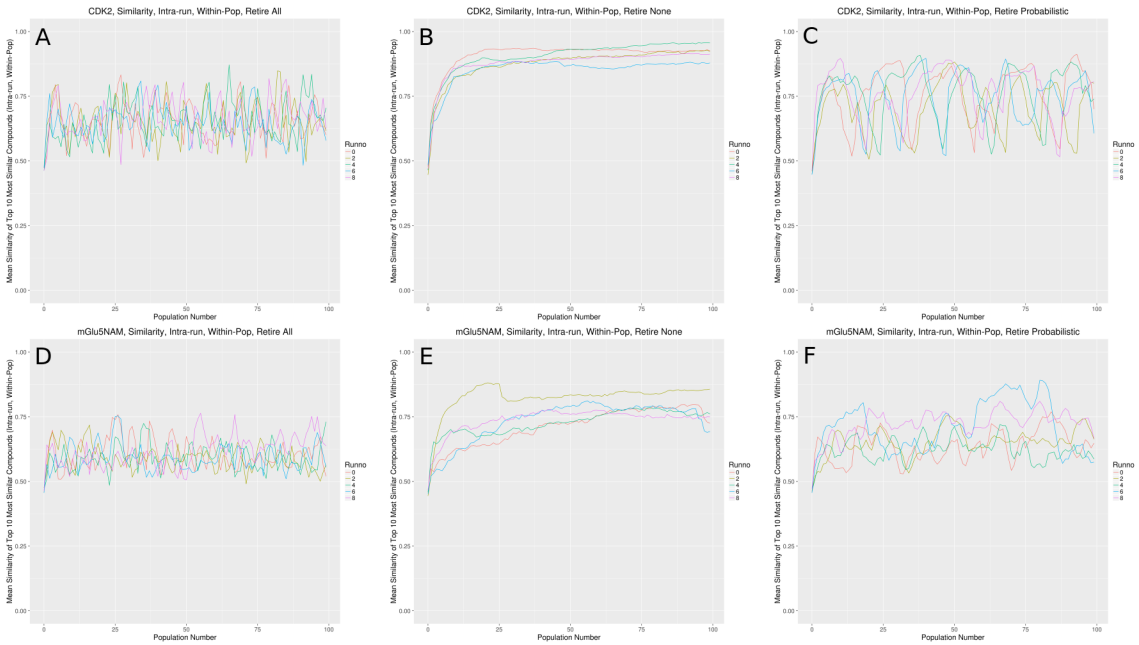


Figure 1.14: Within-run similarities by population number. A) CDK2, policy A, B) CDK2, policy N, C) CDK2, policy P, D) mGlu5, policy A, E) mGlu5, policy N, F) mGlu5 policy P. Colors represent multiple independent runs using the same set of parameters.

every other molecule in that population (disregarding self-comparisons). These similarity measures for each molecule were then consolidated by considering the mean similarity of the top 10 most similar molecules in the population. Recall that this measure acts as a trade-off between maximum similarity and mean similarity which would be bias similarity scores to higher or lower values, respectively.

The mean values of the top-10 similarity metrics per population are plotted according to iteration number is shown in Figure 1.14. Similar to behaviors seen for fitnesses, policy A shows very sporadic behavior, policy N very regular monotonic behavior, and policy P is an intermediate between these two. The maximum values for any population for policies A and P are less than the steady state levels seen for policy N, but lower similarity values are also seen. This indicates that on a per-population basis policies A and P are capable of generating relatively chemically diverse compounds whereas policy N will sample more compounds from the same scaffold pool.

1.2.8.5 Diversity of Designed Compounds Between Runs

In order to address the question of how running the algorithm multiple times provides substantially different results than a single run, active-scoring molecules from each run were compared with those generated from an independently generated run on a per-population basis. A similar metric using the top 10 most similar compounds was used here as well, but with the exception that the calculated similarities were for a molecule in one population with each molecule of the same population number in the other run. Plots of the mean of these values per population is shown in Figure 1.15. The computed similarity values are much lower than the intra-run values shown in Figure 1.14 across all retirement policies and confirms that running repeated EvoGen runs will produce different compounds. Similarity values are generally higher for the CDK2 optimizations than they are for the mGlu5 optimizations, but otherwise the patterns in both sets are comparable.

Policy A results in a sporadic oscillation of inter-run similarities much like the other metrics already discussed. The mean similarities oscillate around a value of about 0.5 and rarely exceed 0.62. These values are only slightly higher than average similarities of the random initial populations. This is even more true for molecules generated for mGlu5, which exhibit smaller oscillations around the low initial values.

Use of policy N in CDK2 optimizations results in a rapid increase in similarity followed by a plateau at a level somewhat higher than the initial population similarities. This behavior can be explained by the fact that molecules in later iterations are very similar to each other within a single run, and the turnover rate is relatively low at these points. Favorable molecules generated in each run will also share molecular features by virtue of the similar property principle. Therefore, as two runs independently converge their similarities should increase relative to the beginning of the run. For a single run, the deviation in similarities decreases with policy N which, when combined with a slight increase in common substructural features, gives the appearance of an increasing inter-run similarity. This same phenomenon manifests in the policy P results, which oscillate in a fashion similar to policy

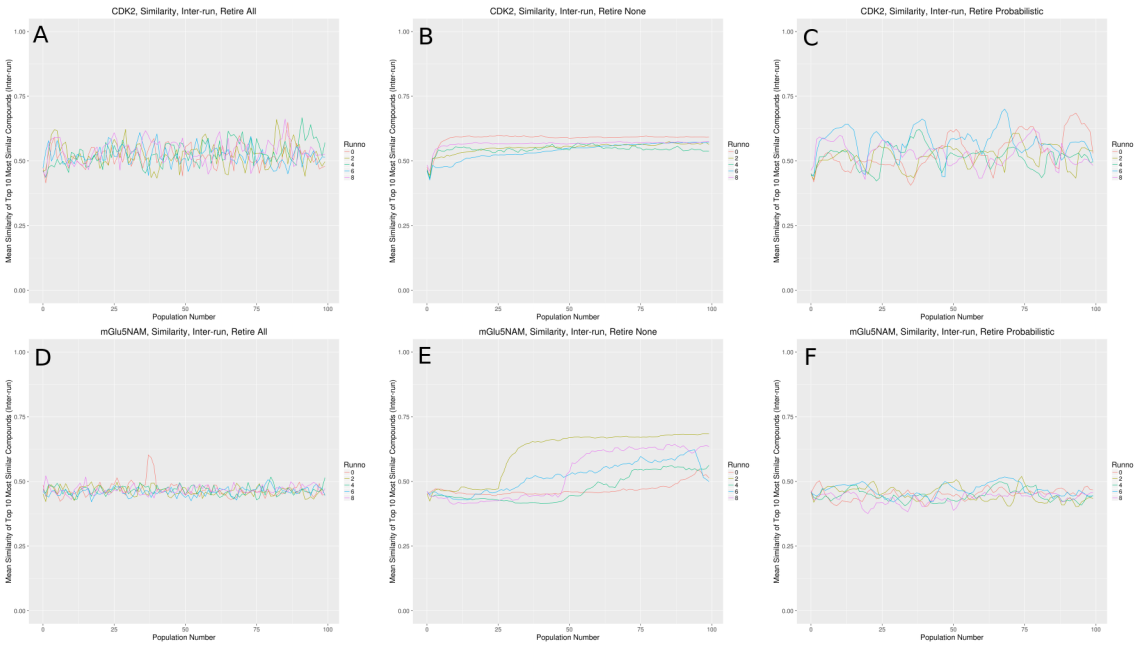


Figure 1.15: Inter-run similarities by population number. A) CDK2, policy A, B) CDK2, policy N, C) CDK2, policy P, D) mGlu5, policy A, E) mGlu5, policy N, F) mGlu5 policy P. Colors indicate independent runs using the same set of parameters.

A, but show localized regions where inter-run similarities are high.

1.2.8.6 Evaluation of Optimization Capabilities

In order to better understand the global optimization ability of the EvoGen algorithm with each retirement policy, a cumulative account of the top 5 fittest compounds between runs was also investigated (Figure 1.16). In these cases, regardless of retirement policy, optimization for CDK2 resulted in an almost immediate saturation of the score level and little information could be discerned from fitness alone. Despite this, one of the runs which used policy N was unable to increase its cumulative maximum score to the values seen for the other runs and indicates that use of this policy will predispose the algorithm to become trapped in local optima. Further metrics which are discussed below revealed additional information for the CDK2 target.

The mGlu5 optimizations, which had a much larger dynamic range, were more infor-

mative and showed discernible differences between the retirement policies. Overall the different runs were often able to achieve similar score levels for the top-scoring compounds, usually with maximal values around 6. The overall shapes of all curves were similar, beginning usually with a rapid increase that leveled off after a few iterations. Policy A showed smaller and more frequent score increases than the other two policies, which is a result of the more stochastic nature of its search function. Policy N features fewer score jumps of larger magnitude interspersed with long plateaus, which again is likely a result of an optimization around a local maximum which is occasionally disturbed by random chance discovery of higher-scoring compounds. Results for policy P more closely resemble those of policy A than they do policy N, but with a smoother fluctuation which increases by larger amounts at a time than does policy A. Interestingly, policy P resulted in the highest-scoring set of compounds of any of the three policies, with the mean fitness values of these compounds in the range of 7.3-7.4.

In all of these cases it appears that each retirement policy is capable of discovering some high-scoring compounds, though the specific results can vary from run to run. It can therefore be concluded that choice of retirement policy does not play a large role in determining the upper bounds of molecule scores, and so any retirement policy could be used to discover these high-scoring compounds.

In order to determine how the diversity of all compounds develops across an entire run and thereby profile the algorithm's structural focus across different runs, a similarity metric was used wherein the mean value of each compound's similarity to the top 10 most similar compounds generated in all previous generations was calculated (Figure 1.17). This metric was motivated by similar factors to the above similarity metrics. Trends in each of the policies are similar to the results described for other metrics, with policy A resulting in a very rapid oscillation of similarity, policy P following a slower oscillation, and policy N almost monotonically increasing during the course of the optimizations.

This metric differs from the within-population similarity measure described above, as

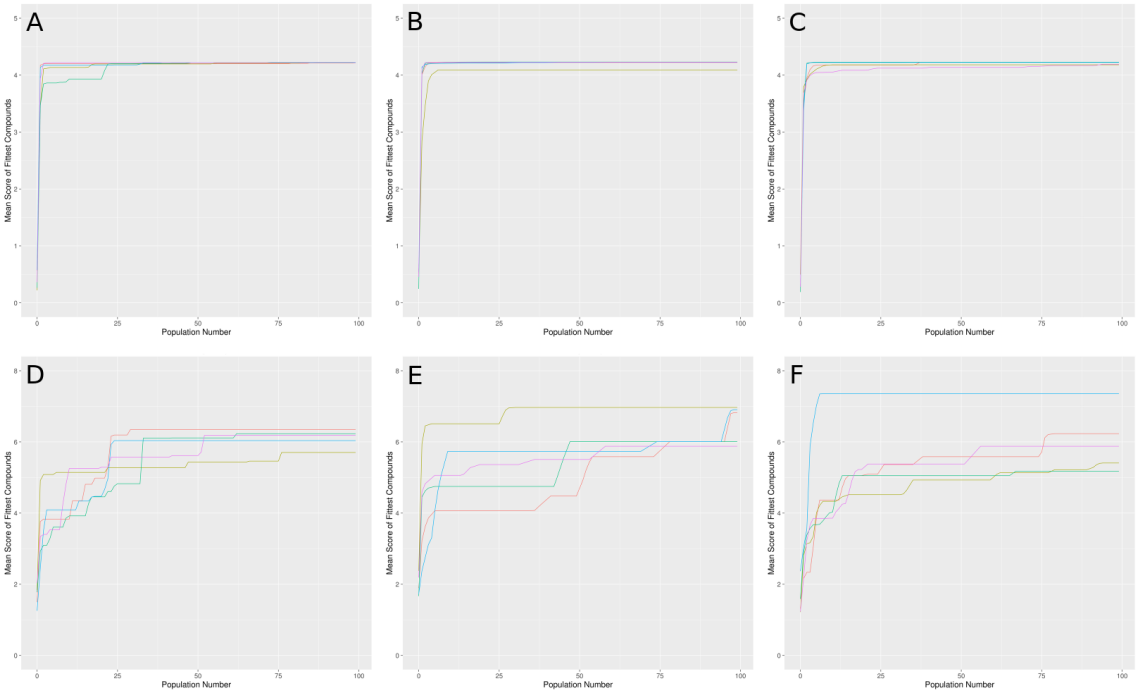


Figure 1.16: Mean fitnesses of the cumulative top 5 fittest compounds by population. A) CDK2, policy A, B) CDK2, policy N, C) CDK2, policy P, D) mGlu5, policy A, E) mGlu5, policy N, F) mGlu5, policy P. Colors represent multiple independent runs using the same parameters.

the number of compounds which may be compared at each iteration is increased relative to the one before it. This factor results in a slight increase on average of the similarity scores relative to the within-population metrics. In addition, this modification to the scoring metric results in an expected upward slope of the baseline similarity regardless of retirement policy; this is as a result of an increasing sample space as the run progresses, increasing the likelihood that compounds with similar structures would have been generated.

An interesting feature that was highlighted with this metric are the relatively large drops in similarity metric for runs 0 and 6 of mGlu5 near the end of the run (blue and red lines, Figure 1.17) and the drop in similarity near iteration 25 of run 2 (yellow line). These drops in similarity correspond to large jumps in score shown in Figure 1.13 and act as a highlight for how discovery of distinct structural motifs can give rise in a short time to a large bump in scores. The further observation there are no drops in score of a similar magnitude with this policy illustrates that the discovery of a new structural feature must also give rise to a corresponding increase in score in order to have a large impact on the overall flow of the algorithm.

1.2.8.7 Diversity Relative to Known Active Space

A main feature of the EvoGen algorithm is its ability to generate novel compound ideas for focused libraries for each target. In order to accomplish this goal it is necessary that at least some of the proposed high-scoring compounds differ from the known active compounds in terms of their 2-dimensional similarity. To quantify the algorithms ability to produce structurally novel compounds, the top 50 compounds from each run were collected and their similarities were calculated to the known actives. The maximum similarity of each compound to known Figure 1.18. Recall that the comparison of random compounds to known actives, and active compounds with themselves (section 1.2.6) indicated that similarity values above 0.7 would represent highly active-like compounds, and values from 0.4-0.6 have structures that are substantially different from active compounds.

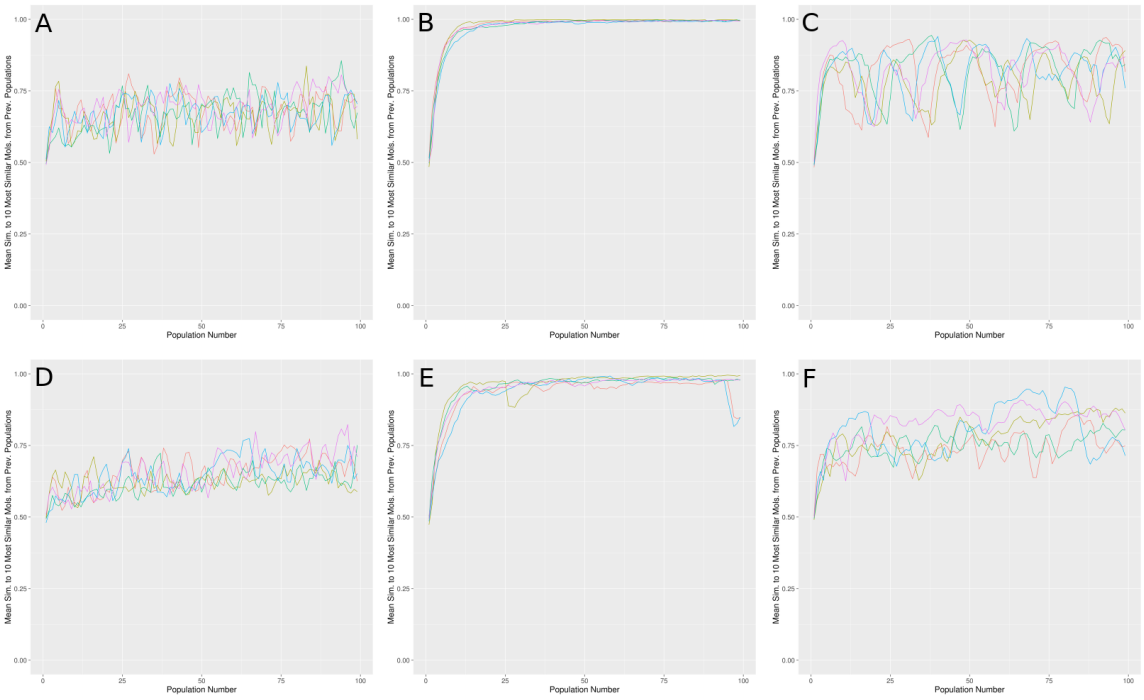


Figure 1.17: Similarity of cumulative best-scoring compounds by population to all generated molecules. A) CDK2, policy A, B) CDK2, policy N, C) CDK2, policy P, D) mGlu5, policy A, E) mGlu5, policy N, F) mGlu5, policy P. Colors represent multiple independent runs using the same parameters.

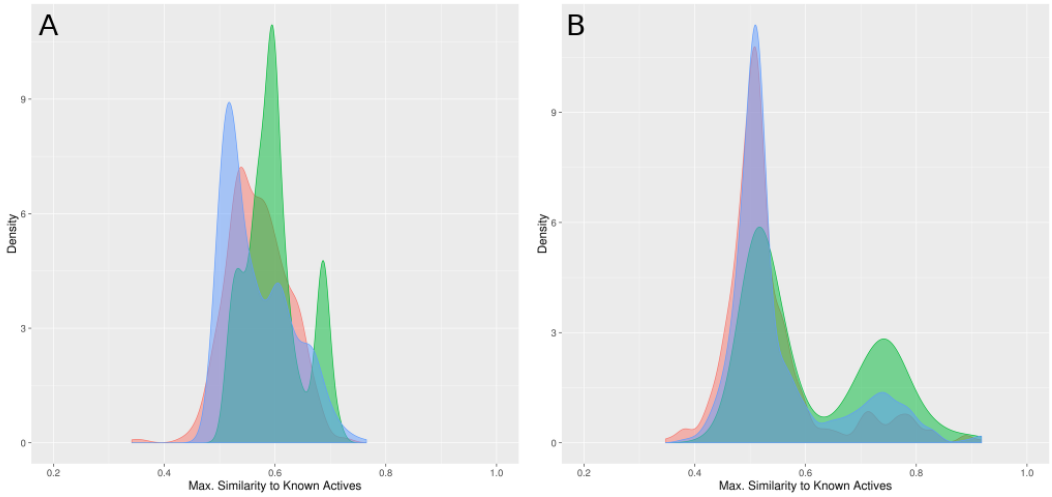


Figure 1.18: Similarity comparisons of top 100 highest scoring compounds with known actives for policies A (red), N (green), and P (blue) according to dataset. A) CDK2, B) mGlu5.

The CDK2 dataset shows some differences between the results for retirement policies A and P as compared to N; there is a substantially larger percentage of compounds from policy N which are at or above a similarity value of 0.6 to the known actives, whereas most compounds from policies A and P have similarity values below 0.6. All policies contain some compounds with similarity values that approach or exceed the 0.7 threshold, but policy N contains a high density of compounds near these values indicating that it is much more likely for policy N to generate compounds that resemble actives than either of the other two policies.

Similarity trends for high-scoring mGlu5 compounds all share the same shape, which is a bimodal distribution similar to that seen when comparing mGlu5 actives to themselves (Figure 1.18). All policies show that the majority of high-scoring compounds have similarities very near 0.5, which is approximately where random compounds fell in the comparison to known actives. It should also be noted that all three policies also result in some compounds with similarities that exceed 0.7, though a substantial portion of molecules generated by policy N fall in this range with a nearly 1:2 ratio of active-like to non-active-like scores.

From these data it can be concluded that while all three retirement policies are capable of generating chemically novel structures with high scores, policy N is much more likely to optimize compounds into similar structures than either policy A or P. Similarly, policies A and P are better suited to the generation of more diverse compounds which are also capable of exhibiting high scores. As such, the choice of retirement policy could be made to adjust to user needs.

1.3 Methods

1.3.1 Structure Modification Algorithm

The main complexity-building step of the EvoGen algorithm is the reaction operation. This operation takes a query molecule as input and performs a random reaction with a random set of reaction partners to generate a more complex final product. Once a query molecule has been selected for this process, the reaction is executed as a series of steps:

1. Determine possible reactions via substructure search
2. Randomly select one of the available reactions
3. Populate a list of available reaction partners
4. Randomly select a reaction partner for every reactant not satisfied by the query molecule

The information required for these steps is stored as a hierarchical tree that can be used to increase the lookup time of possible reactions and partner reagents. On program startup the substructures that define reaction elements are read and stored along with information about the reaction(s) that they were read from. A substructure search is then done to establish relationships between substructural elements; for example, a primary amine would be considered a substructure of a secondary amine. These relationships are then organized into a set of spanning trees to connect all related reagent structures together. Use of this spanning tree provides a rapid search mechanism for possible reactions for a given molecule and may prevent large chunks of the tree from being searched in the event a smaller substructure is not found. Once the reagent structure tree has been built, each molecule from a reagent library is used as a query for the tree. The query molecule's information is then cached in each matching reagent substructure, allowing for a rapid lookup of compounds that will satisfy a given structure.

1.3.2 Reaction Storage Format

Reaction files are encoded using the RXN file format. In order to maximize compatibility with common reagent libraries, any reactions which included compounds that involve metal ions were based on precursor compounds. For example, Grignard reagents are represented using alkyl halide version of the compound instead of the alkyl magnesium halide.

RXN files consist of three sections, namely the RXN header, reactant data, and product data. The RXN header provides a name for the reaction and a count of the numbers of reactants and products in the reaction. Reactant and product data are determined according to the counts given in the RXN header, and molecular structures are encoded using the Molfile format with individual entries prefixed with a '$MOL' delimiter. Reaction information is specified using a field in the Molfile atom lines, wherein each atom which participates in a reaction are given a unique non-zero integer identifier which is used to correlate atoms between products and reactants.

1.3.3 Algorithmic Implementation of Chemical Reactions

To execute an in-silico reaction, the algorithm must determine substructural features from compounds which match their respective reactants and combine them together in the appropriate way. A substructure search is performed on each reactant to determine which atoms represent the reactive portion of the molecule. Any atoms that match the reactive substructure are removed from the reactant and, if they are connected to atoms which are not part of the reactive substructure, replaced with a placeholder atom containing their reaction atom mapping. In a subsequent step, each placeholder atom is then replaced with the appropriate atom from the product structure. This procedure effectively transfers substituents from the reactant molecule to the product substructures (Figure 1.19).

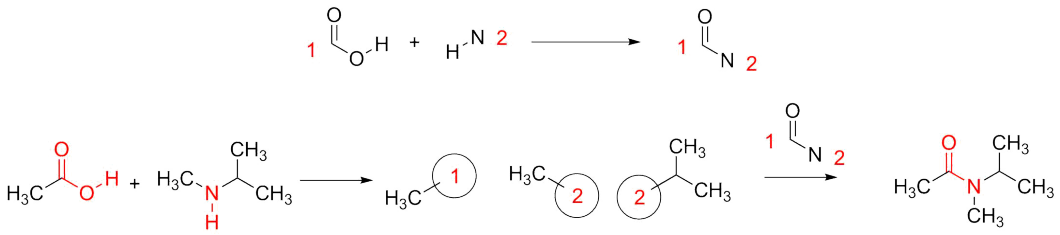


Figure 1.19: Visual representation of the chemical reaction algorithm using an amide coupling reaction (top). A query molecule (acetic acid, far left) and a randomly chosen partner (N,N-isopropylmethylamine) are combined to form the product (N-methylisopropylacetamide). Reactants are searched for substructures matching the reaction (red highlights, left). Matching substructures are removed and marked with dummy atoms to indicate their attachments (middle). Finally, the product is assembled using the determined substituents (red highlights, right).

1.3.4 Model Training

Chemical information was encoded using a standard set of descriptors used by our group [53] which resulted in 1315 numerical values per molecule. Before descriptor generation, datasets were cleaned by discarding small ions from each molecule, discarding any molecules with undefined Gasteiger atom types, and generating a single low-energy conformation for each molecule using CORINA [54]. The descriptors that were used included scalar values including molecular weight and predicted LogP values, and 2- and 3-dimensional autocorrelation functions weighted by a number of chemical properties including partial charge and polarizability.

ANN models were trained for both mGlu5 NAMs and CDK2 inhibitors to predict the activity classification of molecules given molecular descriptor values as input. For each model set, a five-fold cross-validation procedure utilizing both independent and monitoring datasets was used to prevent overtraining and provide an unbiased metric for the performance of each model [53], which resulted in a total of 20 neural network models per target. The neural network architectures consisted of either 8 or 32 hidden nodes (for CDK2 and mGlu5 respectively) and were trained using dropout to further prevent over-training.

1.4 Conclusions

The EvoGen algorithm can be tuned to operate for optimization or for chemical space exploration depending largely on the choice of the parent retirement policy. It was found that a number of high-scoring compounds could be generated regardless of retirement policy. Policy A, wherein no parents were passed unmodified to subsequent populations, consistently exhibited largely stochastic behavior and performed as an effective method for exploring chemical space relative to the other two policies. Policy N showed a much higher tendency for optimization and enumeration around a single chemical scaffold, with high similarities both within single populations and between populations. Policy P operated as an intermediate between these two, with the ability to both explore chemical space but also indications that it is effectively able to optimize compounds over short time intervals. Consideration of the exploration of chemical space indicates that policy N has a tendency of discovering compounds with a much higher similarity to known actives than the other two policies. Despite this, all policies appear to be capable of producing compounds that are also chemically distinct from known actives as well. Comparison of measures of synthetic accessibility indicates that the designed molecules are of similar complexity to generic catalog and screening compounds, though they are more complex than known active compounds for the chosen targets. Overall the data indicates that running the algorithm with retirement policy P multiple times is likely the best approach in most cases, as it will generate a large number of unique compounds with a large fraction of high-scoring compounds.

Chapter 2

Application of Virtual Screening for the Discovery of Novel Muscarinic Receptor M_5 Antagonists

The original research presented in this chapter was previously published in [55]. Alexander Geanes contributed the computational modeling and screening and the much of the synthetic chemistry work surrounding compound VU0549108.

2.1 Introduction and Background

2.1.1 Muscarinic Acetylcholine Receptor Structure and Function

Acetylcholine is a neurotransmitter that is associated with myriad physiological processes including movement and cognition, and is the major endogenous ligand of two classes of receptors in the nervous system, namely the ionotropic nicotinic acetylcholine receptors (nAChRs) and the metabotropic muscarinic acetylcholine receptors (mAChRs) [56]. Nicotinic receptors are multimeric ligand-gated ion channels which, upon activation by an agonist, act as cation channels permeable to Na^+, K^+, and Ca^{2+} [57]. Muscarinic receptors are class A G-protein coupled receptors (GPCRs) found widely distributed in both the central and the peripheral nervous systems (CNS and PNS, respectively) [56]. The mAChRs are metabotropic receptors, meaning that, unlike nAChRs, they attenuate neuronal activity via intermediate messengers and do not act as ion channels themselves. There are five mAChR isoforms, denoted M_1-M_5 which vary in their expression levels and biological function. Structurally, all mAChRs contain a 7-α-helical bundle that spans the cell membrane, four extracellular loops, and four intracellular loops. It has been found that the intracellular loop three is responsible for G-protein specificity (which is the longest of the four intracellular loops), whereas intracellular loop two is responsible for the strength of G-protein binding [58][59]. The acetylcholine binding pocket has been found to reside

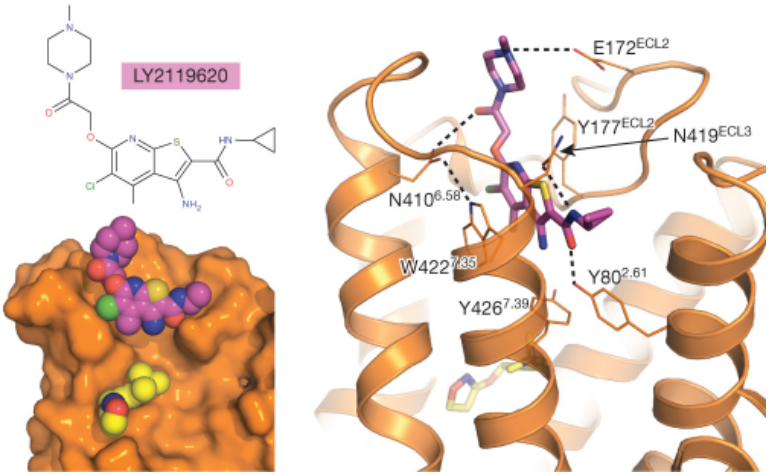


Figure 2.1: Structure and location of the orthosteric and allosteric sites of muscarinic acetylcholine receptor M_2. Left) The agonist iperoxo (yellow) is bound in the orthosteric site, and the allosteric modulator LY2119620 (purple) is bound in the allosteric site above iperoxo. Right) an illustration of the interaction of LY2119620 with the extracellular loops (PDB entry 4MQT; Figure reproduced from [60])

in the transmembrane bundle, and allosteric binding sites are known to exist in an external vestibule above the orthosteric site and interact with the extracellular loops [60] (Figure 2.1).

Subtypes M_1, M_3, and M_5 couple to G_q, whereas M_2 and M_4 preferentially couple to $G_{i/o}$ [61]. Activation of M_1, M_3, and M_5 is known to activate phospholipase C, and activation of M_2 and M_4 inhibits adenylyl cyclase activity [62]. These receptors have also been associated with modulation of ion channels including NMDA receptors and voltage-gated calcium channels [63],[64].

Muscarinic receptors exhibit a number of physiological roles depending on isoform and location, with functions ranging from regulating heart rate and glandular secretion to CNS function. The most prominent mAChR in the forebrain is the M_1 receptor which has been associated with cognition and has been pursued as a possible target for the treatment of several neurological diseases such as Alzheimer's and Schizophrenia [56]. The M_2 and M_3 receptors are found largely in the peripheral nervous system though they are also found in the CNS. M_2 is found in the heart and is known to be responsible for slowing heart rate

after stimulation by the sympathetic nervous system [56]. M_3 is associated with secretory gland function and smooth muscle contraction [56]. Activity at the M_2 and M_3 has been attributed to side effects observed in potential pharmaceuticals that target mAChRs, though these receptors are also responsible for some of the favorable actions in clinically useful antimuscarinic agents [56]. M_4 is prevalent in the CNS, and been implicated as a possible target for antipsychotic medications [56]. The M_5 isoform is the most recently discovered of the five and has seen little attention from the scientific community until recently, likely because it is expressed at the lowest levels of any mAChR (less than 2% of total mAChR expression in rat brain [65]). M_5 is found sparingly in many areas of the body, but its expression levels are highest in the hippocampus and the midbrain, including the ventral tegmental area (VTA) and substantia nigra pars compacta (SNc) [66].

2.1.2 Physiological Associations of M_5

Despite its low overall expression rate, the high density of M_5 in specific areas of the brain makes it of potential interest as a drug target for diseases including stroke and drug addiction. It has been found that M_5 is responsible for ACh-mediated blood vessel dilation in the cerebrum [67]. M_5 knockout mice showed deficits in hippocampus-dependent cognitive tasks and reduced hippocampal long-term potentiation (LTP) [68]. These findings suggest that targeting this receptor could be useful in the treatment of ischemic stroke and cognitive deficiencies under certain conditions. Several studies have indicated that M_5 plays an important role in dopaminergic transmission in the midbrain and could be a novel therapeutic target in the treatment of addiction and drug-seeking behavior [69]. Most drugs of abuse (e.g. morphine, cocaine, and amphetamines) are associated with the activation of reward circuitry in the brain and trigger an increase in dopamine levels in the nucleus accumbens (nAcc)[70]. Stimulation of neurons in the VTA/SNc with acetylcholine induces dopamine release in the nAcc (the terminal point of VTA/SNc neurons). M_5 is effectively the only mAChR expressed in VTA and SNc cells and is therefore the most likely target responsible

for ACh-mediated dopamine release in these regions. When compared to wild-type mice, $M_5^{-/-}$ mice also exhibit a decreased cocaine-conditioned place preference, reduced cocaine self-administration, and lessened opioid withdrawal symptoms when treated with an opioid antagonist. However, the $M_5^{-/-}$ mice showed no significant difference in the observed analgesic effect of the drug as compared to wild-type mice.

2.1.3 Allosteric Modulators of M_5

While studies with gene-knockout mice have provided a wealth of evidence that the M_5 receptor plays a significant role in a number of physiological processes, possible subtle physiological changes or compensatory expression of other proteins mean that these results do not necessarily translate into definitive information about the role of M_5 in biological systems. To definitively associate these effects with the M_5 receptor, M_5-selective small molecule probes are necessary to selectively attenuate M_5 activity in otherwise normal biological systems. Many compounds are known which act at the M_5 receptor either as agonists or antagonists, but these are usually preferential to other mAChR subtypes or are completely non-specific and thereby cannot be used to determine M_5 function. However, within the last few years the first M_5-selective antagonist, negative allosteric modulator (NAM), and positive allosteric modulator (PAM) compounds were discovered by the Vanderbilt Center for Neuroscience Drug Discovery as part of the molecular libraries probe discovery effort [71],[72],[73],[74]

The compound ML375 (1) is the first example of a highly M_5-selective negative allosteric modulator reported in the literature. The precursor to ML375, which contained the core 2,3-dihydro-1H-imidazo[2,1-a]isoindol-5(9bH)-one core (Figure 2.2), was discovered using a functional high throughput screen to simultaneously test for M_5 agonists, PAMs, and antagonists/NAMs [71]. After resynthesis and testing from fresh powder, the HTS hit compound showed an IC_{50} of 3.49 μM in cell lines expressing human M_5, with very little activity at any other mAChR (>30 μM for $M_{1\text{-}4}$). Optimization of this structure across sev-

Figure 2.2: Development of the first M_5-selective NAM from the HTS hit compound (left) to ML375 (center). The stereochemistry proved essential for NAM activity at M_5, with only the S-isomer showing M_5 inhibition. Figure reproduced from [71]

eral rounds of parallel synthesis resulted in the molecular library probe compound ML375 with modifications to the peripheral aryl groups. It was found that restricting the stereochemistry of the compound to only the S-isomer resulted in an improved IC_{50} of 300 nM, with the R-isomer showing effectively no activity at any mAChR. Experiments indicated the compound showed a favorable DMPK profile and CNS penetration. However, the high lipophilicity of the compound led to a very low fraction unbound (f_u) of 0.029 and 0.003 in rat plasma and brain, respectively. Improvements were sought to alleviate this problematic character, and further modification resulted in VU600181, which featured and improved IC_{50} (hM_5: 264 nM) and f_u compared to ML375. Unfortunately, these improvements also resulted in a much higher clearance, close to hepatic bloodflow, in both and human and rat. Some improvements to human intrinsic clearance were made by exchanging a methoxy group with a deuterated methoxy group, but these results did not translate to improved human hepatic clearance or rat DMPK profiles.

2.2 Virtual Screening for M_5 Antagonists and NAMs

After the discovery of ML375 and ML381, it was still of interest to expand the chemical space of known M_5-selective antagonists and NAMs since these scaffolds offered a relatively limited space for exploration due to steep SAR and problems with ancillary phar-

macology. An analysis of the data indicated that pursuing a computational modeling and virtual screening campaign for the discovery of M_5 NAMs would likely be the most fruitful, since a substantial amount of information was generated during the ML375 discovery effort and the rigid isoindolone core would ease molecular modeling relative to the flexible ML381 chemotype. This approach was also chosen as it reduced the need to screen large numbers of compounds (i.e. a traditional HTS campaign) and efforts could instead be focused on compounds likely to present M_5 activity.

2.2.1 Artificial Neural Network Modeling

The medicinal chemistry effort surrounding the ML375 scaffold resulted in 68 active compounds with varying levels of potency and 145 inactive compounds. This information made it possible to build artificial neural network (ANN) quantitative structure-activity relationship (QSAR) models to correlate molecular features with biological activity. The chosen approach involved encoding molecular features using descriptors to encode molecular fingerprints to describe both 2- and 3-dimensional molecular information into a feature vector that could then be used for machine learning and mathematical analysis.

Molecular descriptor calculation, ANN training, and model analyses were performed using the BioChemical Library (BCL) developed at Vanderbilt University (Butkiewicz 2013). The dataset was prepared by removing any ions from structures, adding hydrogens, neutralizing charges, and removing duplicate entries. A single three-dimensional conformation was generated for each structure using Corina version 3.60 [54]. Descriptors which encoded 1D (scalar values), 2D (connectivity), and 3D (shape) information were calculated for each structure. Scalar descriptors included number of hydrogen bond donors and acceptors, calculated LogP, and topological polar surface area. 2- and 3-D information was encoded using autocorrelation functions weighted by properties such as partial charge and polarizability and are described in more detail in [53]. These descriptors resulted in 1315 numerical values for each structure. Calculated descriptor vectors were labeled with

the respective human M_5 pIC_{50} value, or 0 if the compound was inactive. A more thorough explanation of these descriptors can be found in [53]

A feed-forward neural network with a densely connected 32-node hidden layer and a single-valued output layer was trained using this feature set. For training, error values were calculated by treating pIC_{50} values as binary values based on whether pIC_{50} was greater than 5 (active) or less than 5 (inactive). A 5-fold cross validation procedure using monitoring and independent sets and dropout was used to prevent overtraining and to evaluate model performance [53]. Receiver-operator characteristic (ROC) curves and figures of merit are given in Figure 2.3.

2.2.2 Shape-Based Modeling with Surflex-Sim

The conformational rigidity of the ML375 scaffold made shape-based modeling an attractive option to complement QSAR models. ML375 and its analog VU6000181 (VU181) were selected for the generation of a 3-dimensional binding hypothesis. These two compounds were aligned using the flexible alignment feature of Surflex-Sim from Sybyl 2.1.1 [26]. Default parameter values for the algorithm were used with the exception that ring flexibility was considered during the alignment. The highest-scoring hypothesis from the alignment was used for virtual screening (Figure 2.3). A receiver-operator characteristic (ROC) curve for this hypothesis generated by aligning and scoring the remaining 211 compounds from the M_5 NAM dataset with the flexible screening (pscreen) feature of Surflex-Sim is shown in Figure 2.3.

In addition, 1 and 2 were selected for the generation of a 3-dimensional binding hypothesis. These two compounds were aligned using the flexible alignment feature of Surflex-Sim from SYBYL-X 2.1.1. Default parameter values for the algorithm were used with the exception that ring flexibility was considered during the alignment. The highest-scoring hypothesis from the alignment was used for virtual screening. A receiver-operator characteristic (ROC) curve for this hypothesis generated by aligning and scoring the remaining

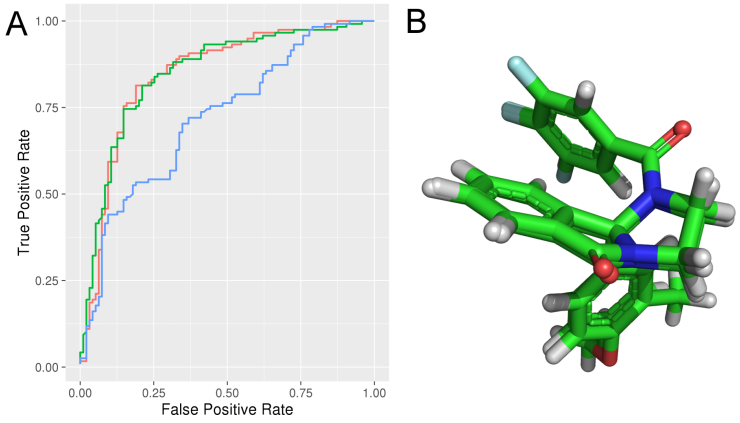


Figure 2.3: A) Receiver-Operator Characteristic Curves of QSAR (green), Surflex-Sim (blue) and Consensus (red) models. B) Highest-scoring Surflex-Sim model of VU0600181 and ML375, used for shape-based scoring. Area under the ROC curve, QSAR: 0.86, Surflex: 0.74. Avg. enrichment up to 10% FPR, QSAR: 1.67, Surflex: 1.47. Figure reproduced from [55].

211 compounds from the M_5 NAM dataset with the flexible screening (pscreen) feature of Surflex-Sim is shown in Figure 2.3A. The shape based model was also able to prioritize active compounds over inactives at a rate higher than random chance. The shape-based model resulted in similar predictive ability to the QSAR model for high-scoring compounds, though the QSAR models appeared to outperform the shape model across the whole dataset.

2.2.3 Virtual Screening for M_5 NAMs and Antagonists

Each model was used to select 30 compounds from an in-house 98,000-compound screening library for experimental testing. The character of the top compounds from each model differed substantially from each other, though there were many substructural commonalities in the compounds of both sets. The QSAR model preferentially chose compounds from the same scaffold pool, whereas the shape-based method contained more diverse chemical structures. From this set of 60 compounds, two compounds from the shape-based set, VU101217 and VU627194, showed weak antagonist activity at M_5, exhibiting around 40 percent inhibition of M_5 response at 30 µM (Figure 2.4). This indicated

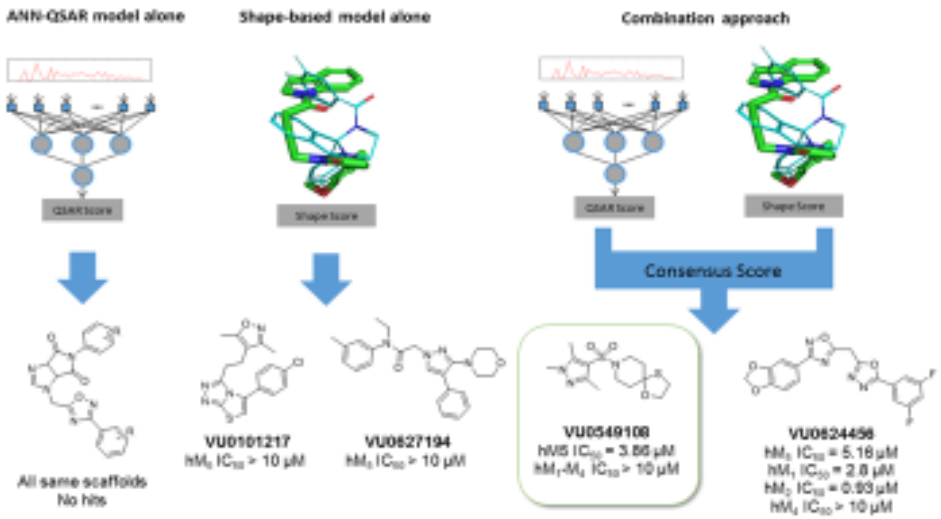


Figure 2.4: Graphical summary of M_5 NAM/antagonist virtual screening. Individual models differed in the character of the compounds that they selected, with QSAR models selecting many compounds from the same scaffold and Surflex models selecting compounds from more diverse scaffolds. The best hit compound was retrieved by combining these two approaches. Figure reproduced from [55].

that a more diverse selection could be powerful for compound discovery than an exhaustive enumeration of SAR around a high scoring scaffold.

To leverage the predictive ability of both models, a larger screen using a consensus approach was performed. To address the issue of low compound diversity when prioritizing by QSAR score alone, the workflow involved first scoring the full 98,000-compound database with the ANN-QSAR models and clustering the highest-scoring 10 percent of the data to select a diverse set of high-scoring chemical structures. To enable clustering, a random subset was sampled from the prioritized compounds, Murcko scaffolds [75] were generated for each selected compound, and the maximum common substructure between each pair of Murcko scaffolds was calculated. Ring and chain fragments from each scaffold were also added to supplement the database. Fingerprint vectors were computed by searching for the presence of each substructure in the prioritized compounds. The distance metric between pairs of compounds was calculated as the Tanimoto [17] coefficient between fingerprint vectors.

Table 2.1: Sources and counts of compounds used to explore SAR around VU0549108 and VU0624456

Selection approach	Source	Number of compounds
2D similarity (VU108)	In-house library	46
2D similarity (VU456)	In-house library	37
VU456 analogs	Commercial	67
Virtual screening (consensus)	In-house library	237

A consensus score was calculated by scoring the top two highest QSAR-scoring compounds from each cluster (or a single compound if the cluster size was 1) with the shape-based method and combining the individual model scores. 986 compounds were aligned to the ML375 binding hypothesis in this manner using the Surflex-Sim pscreen algorithm. The QSAR and Surflex scores were normalized such that their ranges fell between 0 and 1, and both normalized scores were added together to provide the final consensus score for each molecule. The 320 highest scoring compounds according to consensus score were selected and submitted for pharmacological screening against mAChR M_5. Using the two-model consensus approach, two compounds VU0549108 (4) and VU0624456 demonstrated significant M_5 antagonist activity and were confirmed with 10-point concentration response curves. VU0549108 and VU0624456 exhibited maximum M_5 inhibition of 50 and 70 percent, respectively. In order to rapidly explore SAR surrounding the initially more attractive VU0624456 scaffold, SAR-by-catalog was performed by ordering a set of analogs from commercial sources. In addition, compounds that had a high similarity to the two hit compounds (2D fingerprint similarity with Tanimoto > 0.5), and a second set of virtually screened compounds using updated models were chosen from the in-house screening library to follow up these results (Table 2.1). Single point screening revealed several possible antagonists which were subsequently confirmed using 10-point CRCs against M_5. However, only analogs of the original hits showed significant M_5 activity, and none proved more potent than the original compounds.

These data prompted the resynthesis of VU0549108 (4) to reconfirm mAChR activity

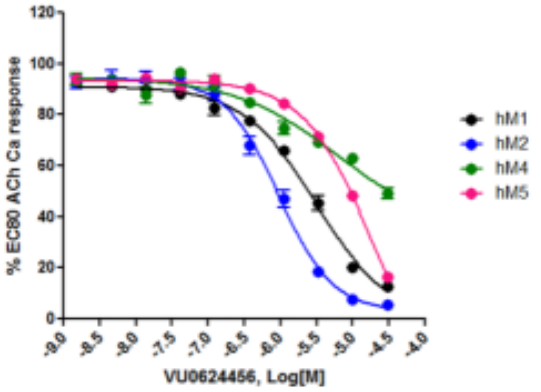


Figure 2.5: Concentration response curves for VU0624456 against $hM_{1\text{-}5}$. Figure reproduced from [55].

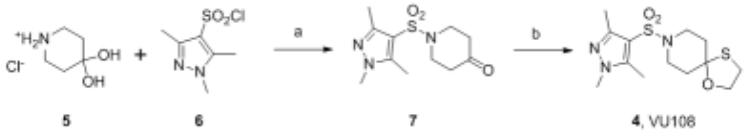


Figure 2.6: Synthesis of VU0549108 (4). Reagents and Conditions: (a) Triethylamine, DCM, rt 12 hrs., 76%; (b) 2-mercaptoethanol, BF_3 OEt_2, DCM, 12 hrs. rt 79%. Figure reproduced from [55].

from fresh powder, as the lack of mAChR selectivity deprioritized VU0624456 (Figure 2.5). VU0549108 (4) was readily prepared in two steps (Scheme 2.6). Starting from commercial piperidine hydrate 5, treatment with sulfonyl chloride 6 provides 7 in 76% yield. Condensation with 2-mercaptoehthanol under Lewis acid catalysis affords 4 in 79% yield. This expedited route was also employed for analog synthesis.

The resynthesized 4 proved to be a functional inhibitor of M_5 (Figure 2.7), with an IC_{50} of 6.2 µM

(pIC_{50} = 5.23±0.09, ACh min 12.6±2.5) and modest selectivity versus $M_{1\text{-}4}$ (IC_{50}s $>$ 10 µM). The divergence from classical orthosteric antagonist chemotype, coupled with the observed selectivity, led us to perform radioligand binding assays to assess if 4 was an allosteric ligand (NAM) or an atypical orthosteric ligand. Here, employing the standard [3H]-NMS ligand (Figure 2.7),1-3 and compared to atropine, 4 proved to interact with

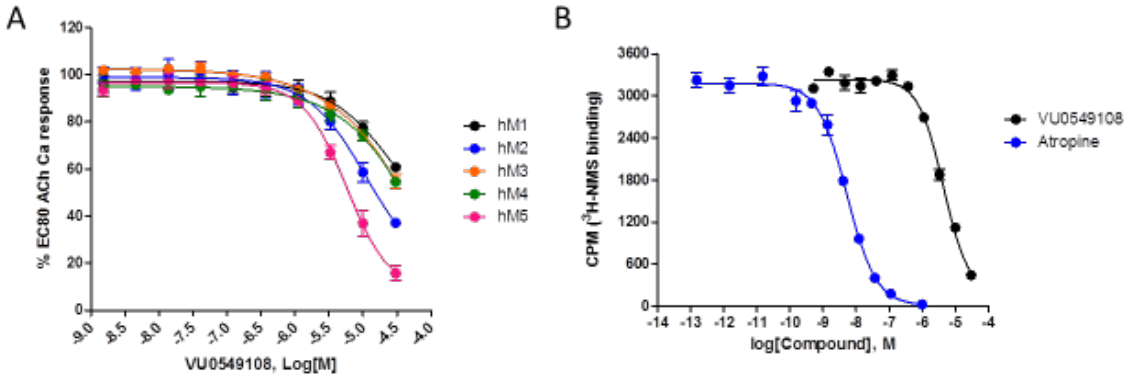


Figure 2.7: Molecular pharmacology profile of VU108 (4). A) Concentration-response curves of 4 for human hM_5, as well as $hM_{1\text{-}4}$ ($IC_{50} > 10$ μM), n=3); B) [^{3}H]-N-methylscopolamine (NMS) competition binding (n=3) in membranes prepared from human M_5-expressing cells, showing competitive displacement (Ki=2.7 μM). Figure reproduced from [55].

or modulate with orthosteric site with a K_i of 2.7 μM (atropine control, K_i = 2.7 nM) [71],[72],[73],[76],[77],[78]

However, the effect on NMS binding could be due to cooperativity with the orthosteric site by binding of 4 to an allosteric site, or it could reflect direct, competitive interaction at the orthosteric site. Based on 4 being a small, non-basic chemotype, distinct form prototypical mAChR antagonists, and some measure of mAChR selectivity, more potent analogs are required to definitively address the mode of inhibition of M_5. Moreover, shape-based alignment of 1 and 4 (Figure 5), while showing reasonable overlap for achiral 4, significant lipophilic regions are not occupied. This hypothesis shows an alignment of one sulfonyl oxygen and the free pyrazole nitrogen in 4 align with the carbonyl groups in 1, and the 4 oxathiolane oriented along the 9b-4-chlorophenyl group of 1. Since the 4-chlorophenyl group confers stereochemistry to ML375, this three-point pharmacophore could hold the structure in a position that allows 4 to mimic the stereochemistry of 1 without itself being chiral, and suggests why the virtual screen identified 4.

A small library was synthesized to further explore SAR surrounding the 4 scaffold in hopes of increasing M_5 inhibitory potency. Two points were considered for modification,

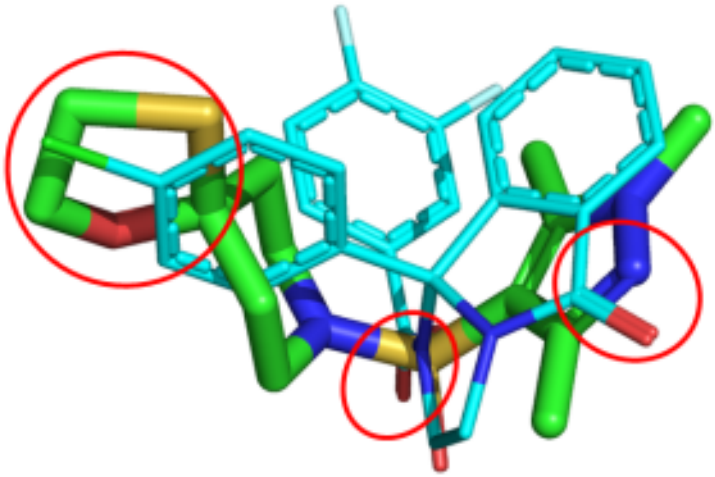
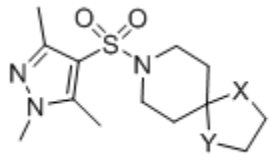


Figure 2.8: Overlay of 4 (green) on 1 (cyan). Hydrogen bond acceptors at the pyrazole and sulfonyl groups of 4 align with corresponding hydrogen bond acceptors in 1, and the oxathiolane of 4 overlaps with the 4-chlorophenyl moiety of 1. The overlap of these features could explain why the virtual screen selected 4. Figure reproduced from [55].

namely the oxathiolane spirocycle, analogs 8 (Table 2.2) and the heterocyclic sulfonamide congeners, 9 (Table 2.3). SAR was steep,15-17 with all analogs displaying IC_{50}s > 10 μM; however, the EC_{80} was diminished. The 1,3-oxathiolane (4) was critical for activity, as the parent piperidinone, 1,3-dithiolane (8a), 1,3-dioxolane (8b) and spiro furan (8c) analogs all showed weak inhibition of M_5. Similarly, only the 1,3,5-pyrazole sulfonamide of 4 maintained M_5 inhibition.

2.3 Conclusions

Muscarinic acetylcholine receptors are pharmacologically important GPCRs found in the central and peripheral nervous system. Different mAChR isoforms are associated with a large variety of physiological phenomenon. The mAChR M_5 isoform is suspected to play a role in dopaminergic transmission in the midbrain, which makes it a possible target for novel treatments for drug addiction. Here we used a combination of computational

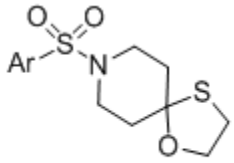


8

Table 2.2: Structure and Activities of Analogs 8

Compound	**X**	**Y**	**hM_5 Percent Inhib.**
4 (VU108)	S	O	88% (hM_5 IC_{50}=5.23 μM)
8a	S	S	86%
8b	O	O	45%
8c	O	CH_2	37%

Table 2.3: Structure and Activities of Analogs 9



9

Compound	**Ar**	**hM_5 Percent Inhib.**
4 (VU108)	1,3,5-TriMe-1H-4-pyrazole	88% (hM_5 IC_{50}=5.23 μM)
9a	1-Me-1H-4-pyrazole	27%
9b	1,3-DiMe-1H-4-pyrazole	Inactive
9c	1,5-DiMe-1H-4-pyrazole	83%
9d	3-Cl-4-F-phenyl	83%
9e	3,5-DiF-phenyl	62%
9f	3,4,5-TriF-pheyl	55%
9g	3,5-DiMe-4-isoxazole	40%

techniques to leverage information from past drug discovery efforts to discover chemically novel M_5 subtype-selective antagonists. Two compounds, VU0624456 and VU0549108 were identified as part of a virtual screen using these techniques. VU0549108 demonstrated moderate selectivity for M_5 over the other four mAChRs, with a potency of 5.23 μM against M_5. The SAR surrounding both VU0549108 and VU0624456 was found to be very steep, with small modifications greatly reducing the potency of the compounds.

REFERENCES

[1] Baskin, I. I.; Winkler, D.; Tetko, I. V. *Expert Opinion on Drug Discovery* **2016**, *11*, 785–795.

[2] Wang, Y.; Bryant, S. H. NCBI PubChem BioAssay Database. 2014; http://www.ncbi.nlm.nih.gov/books/NBK190601/.

[3] Bento, A. P.; Gaulton, A.; Hersey, A.; Bellis, L. J.; Chambers, J.; Davies, M.; Krger, F. A.; Light, Y.; Mak, L.; McGlinchey, S.; Nowotka, M.; Papadatos, G.; Santos, R.; Overington, J. P. *Nucleic Acids Research* **2014**, *42*, D1083–D1090.

[4] Talele, T.; Khedkar, S.; Rigby, A. *Current Topics in Medicinal Chemistry* **2010**, *10*, 127–141.

[5] Clark, D. E. *Expert Opinion on Drug Discovery* **2006**, *1*, 103–110.

[6] Maggiora, G. M.; Shanmugasundaram, V. In *Chemoinformatics*; Walker, J. M., Bajorath, J., Eds.; Humana Press: Totowa, NJ, 2004; Vol. 275; pp 1–50.

[7] Acharya, C.; Coop, A.; Polli, J. E.; MacKerell, A. D. *Current computer-aided drug design* **2011**, *7*, 10–22.

[8] Drwal, M. N.; Griffith, R. *Drug Discovery Today: Technologies* **2013**, *10*, e395–e401.

[9] Chen, Z.; Li, H.-l.; Zhang, Q.-j.; Bao, X.-g.; Yu, K.-q.; Luo, X.-m.; Zhu, W.-l.; Jiang, H.-l. *Acta Pharmacologica Sinica* **2009**, *30*, 1694–1708.

[10] Vilar, S.; Karpiak, J.; Costanzi, S. *Journal of Computational Chemistry* **2009**, NA–NA.

[11] Todeschini, R., Consonni, V., Mannhold, R., Kubinyi, H., Folkers, G., Eds. *Molecular Descriptors for Chemoinformatics*; Methods and Principles in Medicinal Chemistry; Wiley-VCH Verlag GmbH & Co. KGaA: Weinheim, Germany, 2009; Vol. 41.

[12] Randic', M.; Jerman-Blazîc', B.; Trinajstic', N. *Computers & Chemistry* **1990**, *14*, 237–246.

[13] Sliwoski, G.; Kothiwale, S.; Meiler, J.; Lowe, E. W. *Pharmacological Reviews* **2013**, *66*, 334–395.

[14] Andrade, C. H.; Pasqualoto, K. F. M.; Ferreira, E. I.; Hopfinger, A. J. *Molecules* **2010**, *15*, 3281–3294.

[15] Durant, J. L.; Leland, B. A.; Henry, D. R.; Nourse, J. G. *Journal of Chemical Information and Computer Sciences* **2002**, *42*, 1273–1280.

[16] Riniker, S.; Landrum, G. A. *Journal of Cheminformatics* **2013**, *5*, 26.

[17] Willett, P. *Drug Discovery Today* **2006**, *11*, 1046–1053.

[18] Sastry, G. M.; Inakollu, V. S. S.; Sherman, W. *Journal of Chemical Information and Modeling* **2013**, *53*, 1531–1542.

[19] Williams, C. *Molecular Diversity* **2006**, *10*, 311–332.

[20] Thimm, M.; Goede, A.; Hougardy, S.; Preissner, R. *Journal of Chemical Information and Computer Sciences* **2004**, *44*, 1816–1822.

[21] Wermuth, C. G.; Ganellin, C. R.; Lindberg, P.; Mitscher, L. A. *Pure and Applied Chemistry* **1998**, *70*.

[22] Wolber, G. *Drug Discovery Today* **2008**, *13*, 23–29.

[23] Schuffenhauer, A. *Wiley Interdisciplinary Reviews: Computational Molecular Science* **2012**, *2*, 842–867.

[24] Dixon, S. L.; Smondyrev, A. M.; Knoll, E. H.; Rao, S. N.; Shaw, D. E.; Friesner, R. A. *Journal of Computer-Aided Molecular Design* **2006**, *20*, 647–671.

[25] Clark, D. E.; Waszkowycz, B.; Wong, M.; Lockey, P. M.; Adalbert, R.; Gilley, J.; Clark, J.; Coleman, M. P. *Bioorganic & Medicinal Chemistry Letters* **2016**, *26*, 2920–2926.

[26] Jain, A. N. *Journal of Medicinal Chemistry* **2004**, *47*, 947–961.

[27] Xiang, Z. et al. *ACS Chemical Neuroscience* **2011**, *2*, 730–742.

[28] Hansch, C.; Fujita, T. *Journal of the American Chemical Society* **1964**, *86*, 1616–1626.

[29] Smola, A.; Vishwanathan, S. V. N. *Introduction to Machine Learning*; Cambridge University Press, 2008.

[30] Chen, B.; Harrison, R. F.; Papadatos, G.; Willett, P.; Wood, D. J.; Lewell, X. Q.; Greenidge, P.; Stiefl, N. *Journal of Computer-Aided Molecular Design* **2007**, *21*, 53–62.

[31] Fernández, M.; Caballero, J. *Chemical Biology & Drug Design* **2006**, *68*, 201–212.

[32] Reker, D.; Rodrigues, T.; Schneider, P.; Schneider, G. *Proceedings of the National Academy of Sciences* **2014**, *111*, 4067–4072.

[33] Brown, R. D.; Martin, Y. C. *Journal of Chemical Information and Computer Sciences* **1996**, *36*, 572–584.

[34] Mayr, A.; Klambauer, G.; Unterthiner, T.; Hochreiter, S. *Frontiers in Environmental Science* **2016**, *3*.

[35] Lavecchia, A. *Drug Discovery Today* **2015**, *20*, 318–331.

[36] Mueller, R. et al. *ChemMedChem* **2012**, *7*, 406–414.

[37] Merck Molecular Activity Challenge | Kaggle. https://www.kaggle.com/c/MerckActivity.

[38] Dahl, G. E.; Jaitly, N.; Salakhutdinov, R. *arXiv preprint arXiv:1406.1231* **2014**,

[39] Schneider, G.; Fechner, U. *Nature Reviews Drug Discovery* **2005**, *4*, 649–663.

[40] Lewis, R. A. *Journal of Computer-Aided Molecular Design* **1990**, *4*, 205–210.

[41] Nishibata, Y.; Itai, A. *Tetrahedron* **1991**, *47*, 8985–8990.

[42] Boda, K.; Johnson, A. P. *Journal of Medicinal Chemistry* **2006**, *49*, 5869–5879.

[43] Lewell, X. Q.; Judd, D. B.; Watson, S. P.; Hann, M. M. *Journal of Chemical Information and Computer Sciences* **1998**, *38*, 511–522.

[44] RDKit: Open-source cheminformatics. http://www.rdkit.org/.

[45] Hartenfeller, M.; Zettl, H.; Walter, M.; Rupp, M.; Reisen, F.; Proschak, E.; Weggen, S.; Stark, H.; Schneider, G. **2012**, *8*, e1002380.

[46] Spânkuch, B.; Keppner, S.; Lange, L.; Rodrigues, T.; Zettl, H.; Koch, C. P.; Reutlinger, M.; Hartenfeller, M.; Schneider, P.; Schneider, G. *Angewandte Chemie International Edition* **2013**, *52*, 4676–4681.

[47] Roughley, S. D.; Jordan, A. M. *Journal of Medicinal Chemistry* **2011**, *54*, 3451–3479.

[48] Irwin, J. J.; Sterling, T.; Mysinger, M. M.; Bolstad, E. S.; Coleman, R. G. *Journal of Chemical Information and Modeling* **2012**, *52*, 1757–1768.

[49] Ertl, P.; Schuffenhauer, A. *Journal of Cheminformatics* **2009**, *1*, 8.

[50] Brasca, M. G. et al. *Journal of Medicinal Chemistry* **2009**, *52*, 5152–5163.

[51] McIntyre, N. A.; McInnes, C.; Griffiths, G.; Barnett, A. L.; Kontopidis, G.; Slawin, A. M. Z.; Jackson, W.; Thomas, M.; Zheleva, D. I.; Wang, S.; Blake, D. G.; Westwood, N. J.; Fischer, P. M. *Journal of Medicinal Chemistry* **2010**, *53*, 2136–2145.

[52] Bradley, E. K.; Miller, J. L.; Saiah, E.; Grootenhuis, P. D. J. *Journal of Medicinal Chemistry* **2003**, *46*, 4360–4364.

[53] Mendenhall, J.; Meiler, J. *Journal of Computer-Aided Molecular Design* **2016**, *30*, 177–189.

[54] Gasteiger, J.; Rudolph, C.; Sadowski, J. *Tetrahedron Computer Methodology* **1990**, *3*, 537–547.

[55] Geanes, A. R.; Cho, H. P.; Nance, K. D.; McGowan, K. M.; Conn, P. J.; Jones, C. K.; Meiler, J.; Lindsley, C. W. *Bioorganic & Medicinal Chemistry Letters* **2016**, *26*, 4487–4491.

[56] Dencker, D.; Thomsen, M.; Wrtwein, G.; Weikop, P.; Cui, Y.; Jeon, J.; Wess, J.; Fink-Jensen, A. *ACS Chemical Neuroscience* **2012**, *3*, 80–89.

[57] Itier, V.; Bertrand, D. *FEBS Letters* **2001**, *504*, 118–125.

[58] Burstein, E. S.; Spalding, T. A.; Brann, M. R. *Biochemistry* **1998**, *37*, 4052–4058.

[59] Burstein, E. S.; Spalding, T. A.; Brann, M. R. *The Journal of Biological Chemistry* **1998**, *273*, 24322–24327.

[60] Kruse, A. C. et al. *Nature* **2013**, *504*, 101–106.

[61] Kruse, A. C.; Kobilka, B. K.; Gautam, D.; Sexton, P. M.; Christopoulos, A.; Wess, J. **2014**, *13*, 549–560.

[62] Volpicelli, L. A.; Levey, A. I. *Progress in Brain Research* **2004**, *145*, 59–66.

[63] Marino, M. J.; Rouse, S. T.; Levey, A. I.; Potter, L. T.; Conn, P. J. *Proceedings of the National Academy of Sciences of the United States of America* **1998**, *95*, 11465–11470.

[64] Shapiro, M. S.; Gomeza, J.; Hamilton, S. E.; Hille, B.; Loose, M. D.; Nathanson, N. M.; Roche, J. P.; Wess, J. *Life Sciences* **2001**, *68*, 2481–2487.

[65] Eglen, R. M.; Nahorski, S. R. *British Journal of Pharmacology* **2000**, *130*, 13–21.

[66] Abrams, P.; Andersson, K.-E.; Buccafusco, J. J.; Chapple, C.; Groat, W. C.; Fryer, A. D.; Kay, G.; Laties, A.; Nathanson, N. M.; Pasricha, P. J.; Wein, A. J. *British Journal of Pharmacology* **2006**, *148*, 565–578.

[67] Yamada, M.; Lamping, K. G.; Duttaroy, A.; Zhang, W.; Cui, Y.; Bymaster, F. P.; McKinzie, D. L.; Felder, C. C.; Deng, C.-X.; Faraci, F. M.; Wess, J. *Proceedings of the National Academy of Sciences* **2001**, *98*, 14096–14101.

[68] Araya, R. et al. *Neurobiology of Disease* **2006**, *24*, 334–344.

[69] Berizzi, A. E.; Gentry, P. R.; Rueda, P.; Den Hoedt, S.; Sexton, P. M.; Langmead, C. J.; Christopoulos, A. *Molecular Pharmacology* **2016**, *90*, 427–436.

[70] Wise, R. A. *Annual Review of Neuroscience* **1996**, *19*, 319–340.

[71] Gentry, P. R.; Kokubo, M.; Bridges, T. M.; Kett, N. R.; Harp, J. M.; Cho, H. P.; Smith, E.; Chase, P.; Hodder, P. S.; Niswender, C. M.; Daniels, J. S.; Conn, P. J.; Wood, M. R.; Lindsley, C. W. *Journal of Medicinal Chemistry* **2013**, *56*, 9351–9355.

[72] Kurata, H.; Gentry, P. R.; Kokubo, M.; Cho, H. P.; Bridges, T. M.; Niswender, C. M.; Byers, F. W.; Wood, M. R.; Daniels, J. S.; Conn, P. J.; Lindsley, C. W. *Bioorganic & Medicinal Chemistry Letters* **2015**, *25*, 690–694.

[73] Gentry, P. R.; Kokubo, M.; Bridges, T. M.; Noetzel, M. J.; Cho, H. P.; Lamsal, A.; Smith, E.; Chase, P.; Hodder, P. S.; Niswender, C. M.; Daniels, J. S.; Conn, P. J.; Lindsley, C. W.; Wood, M. R. *Journal of Medicinal Chemistry* **2014**, *57*, 7804–7810.

[74] Gentry, P. R. et al. *ChemMedChem* **2014**, n/a–n/a.

[75] Bemis, G. W.; Murcko, M. A. *Journal of Medicinal Chemistry* **1996**, *39*, 2887–2893.

[76] Melancon, B. J.; Hopkins, C. R.; Wood, M. R.; Emmitte, K. A.; Niswender, C. M.; Christopoulos, A.; Conn, P. J.; Lindsley, C. W. *Journal of Medicinal Chemistry* **2012**, *55*, 1445–1464.

[77] Conn, P. J.; Lindsley, C. W.; Meiler, J.; Niswender, C. M. *Nature Reviews. Drug Discovery* **2014**, *13*, 692–708.

[78] Lindsley, C. W.; Emmitte, K. A.; Hopkins, C. R.; Bridges, T. M.; Gregory, K. J.; Niswender, C. M.; Conn, P. J. *Chemical Reviews* **2016**, *116*, 6707–6741.