A Dataset Details and Use

A.1 Persistence, Ethics, and License

The 2D version of the dataset is hosted publicly by a third party provider, FigShare for redundancy (for which they determine persistence), https://doi.org/10.6084/m9.figshare.14745234. The rest of the data, including all the 3D structures, is hosted by Argonne's Leadership Computing Center and accessible via a Globus endpoint with documentation hosted by GitHub. The authors are confident the data will be persistent across FigShare, GitHub, ALCF, and Globus.

The authors believe this data presents a minimal ethical risk to the community at large. The proposed dataset contains computer-generated protein-ligand structures and computed scores. Information of this sort, albeit at a smaller scale, is widely available on the web currently, and releasing this particular dataset would not set any new standards (in terms of a qualitative assessment of data type). The authors believe the biggest risk of releasing this dataset would be localized to one's interpretation of resulting models, theories, or endeavours based on inductive reasoning from the data alone—but, this risk is typical of any scientific dataset.

A.2 Data Generation Methods

The original data was generated by Open Eye Scientific's FRED docking protocols, and was aggregated, cleaned, and standardized with naming convetions. The code was generated with this software: github.com/inspiremd/Model-generation.

A.3 Docking Protocol

The training and testing datasets for these experiments were generated using 31 protein receptors, covering 9 diverse SARS-CoV-2 viral target protein conformations, that target (1) 3CLPro (main protease, part of the non-structural protein/ NSP-3), (2) papain like protease (PLPro), (3) SARS macrodomain (also referred to as ADP-ribosyltransferase, ADRP), (4) helicase (NSP13), (5) NSP15 (endoribonuclease), (6) RNA dependent RNA polymerase (RDRP, NSP7-8-12 complex), and (7) methyltransferase (NSP10-16 complex). For each of these protein targets, we identified a diverse set of binding sites along the protein interfaces using two strategies: for proteins that had already available structures with bound ligands, we utilized the X-ray crystallography data to identify where ligand densities are found and defined a pocket bound by a rectangular box surrounding that area; and for proteins that did not have ligands bound to them, we used the FPocket toolkit that allowed us to define a variety of potential binding regions (including protein interfaces) around which we could define a rectangular box. This process allowed us to expand the potential binding sites to include over 90 unique regions for these target proteins. We use the term target to refer to one binding site. The protocol code can be found here: https://github.com/2019-ncovgroup/HTDockingDataInstructions.

A.4 Preparation of Ligand Libraries

Two ligand libraries were prepared. The first was the orderable subset of the Zinc15 database (we refer to this as OZD) and the second was the orderable subset of the MCULE compound database (we refer to this as ORD). The generation of the orderable subsets was primarily a manual activity that involved finding all compounds that are either in stock or available to ship in three weeks across a range of suppliers. Consistent SMILE strings and drug descriptors for the orderable subsets of the Zinc15 and MCULE compound databases were generated as described by Babuji et al [2020]. Drug descriptors for the Zinc15 and MCULE compound databases can be downloaded from the nCOV Group Data Repository at https://2019-ncovgroup.github.io.

A.5 Docking Protocols

We used OpenEye Toolkits for docking six million (6M) small-molecules from the OZD database. For each ligand from the database, we calculate a single Chemgauss4 score as a surrogate for binding affinity. For each ligand in the database (provided as a SMILES string), we create an ensemble of structures, sampled from various proteinization states (tautomers) and 3-D conformers. Typically, this

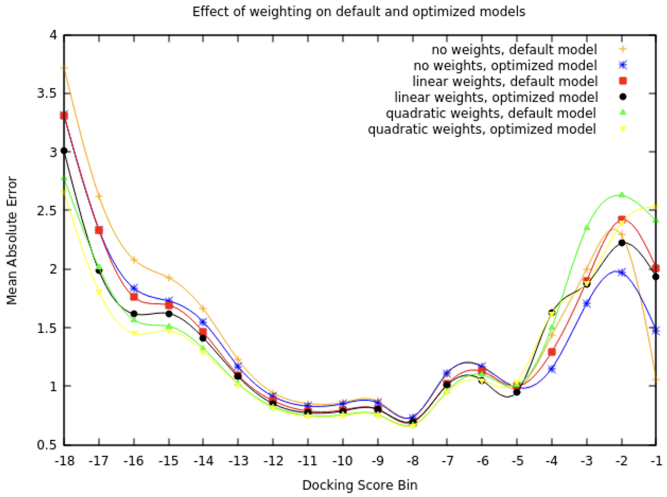

Figure 5: (left) Effects of sample weighting strategies on the default and optimized model.

results in approximately one thousand 3-D structures for each ligand. If the ligand's stereochemistry is ambiguous from the provided SMILES, enantiomers are enumerated prior to conformer generation. Each conformer is then docked to the protein target using FRED or HYBRID depending on the availability of a bound ligand in the crystal structure of the specific target. Scores are calculated over the best pose over the ligand ensemble using the Chemgauss4 scoring function. The scoring function is unitless and aims to measure the fitness of a ligand pose in an active site via a numerical score. Poses with more negative scores are more likely to be correctly docked. In order to produce a single score for each database SMILES entry, the minimum over the ensemble is returned. The typical range of the scoring function is between -20 and 0, though the scoring function range is unbounded.

A.6 Data Cleanliness

Failure analysis of docking runs is available in the SI of [6].

A.7 Model Hyperparameter Optimization

The CANDLE framework was used subsequently used to tune the deep neural network for future training and screening activities. The CANDLE compliant deep neural network was tuned in two phases. The first involved using two CANDLE hyperparameter optimization workflows - mrlMBO and GA. Each differs in the underlying ML techniques used to optimize the hyperparameters. The second phase involved implementing and testing new sample weighting strategies in an attempt to weight the samples at the good end of the distribution more heavily during training. Results of the GA and mlrMBO workflows produced a model architecture that had a 6.6% decrease in the validation mean absolute error and a 2.8% increase in the validation R-squared metrics.

Efforts to decrease the error in the good tail of the distribution (where the docking scores are best) focused on adding sample weights to the model while training. We investigated linear and quadratic weighting strategies. We applied the weighting strategies to both the default model as well as the hyperparameter optimized model. The linear strategy weights the sample proportionally with the docking score, while the quadratic scales with the square of the docking score. These strategies generic in that they can be applied to basically any training target value. To analyze the impact of the weighting strategies, we computed the mean absolute error on bins of predicted scores with a bin interval of one. These results are presented in Figure 5.

A.8 Model Scores

See table 2.

pocket (model)	epochs	loss	mae	r2	val_loss	val_mae	val_r2
3CLPro_7BQY_A_1_F	513	0.338	0.454	0.870	0.426	0.505	0.838
ADRP_6W02_A_1_H	599	1.020	0.782	0.786	1.326	0.876	0.724
NPRBD_6VYO_A_1_F	453	0.302	0.427	0.848	0.356	0.466	0.822
NPRBD_6VYO_AB_1_F	599	0.482	0.540	0.800	0.601	0.597	0.752
NPRBD_6VYO_BC_1_F	427	0.566	0.586	0.899	0.702	0.653	0.876
NPRBD_6VYO_CD_1_F	523	0.474	0.534	0.816	0.602	0.597	0.769
NPRBD_6VYO_DA_1_F	587	0.485	0.541	0.854	0.591	0.595	0.824
NSP10-16_6W61_AB_1_F	283	0.490	0.546	0.902	0.615	0.613	0.878
NSP10-16_6W61_AB_2_F	387	0.523	0.565	0.901	0.655	0.628	0.877
NSP10_6W61_B_1_F	433	0.576	0.590	0.908	0.677	0.631	0.893
Nsp13.helicase_m1_pocket2	338	0.553	0.577	0.867	0.663	0.633	0.843
Nsp13.helicase_m3_pocket2	434	0.485	0.538	0.878	0.582	0.585	0.855
NSP15_6VWW_A_1_F	406	0.526	0.563	0.837	0.640	0.621	0.804
NSP15_6VWW_A_2_F	441	0.336	0.451	0.876	0.417	0.506	0.849
NSP15_6VWW_AB_1_F	599	0.234	0.376	0.829	0.298	0.423	0.784
NSP15_6W01_A_1_F	596	0.473	0.533	0.835	0.595	0.597	0.795
NSP15_6W01_A_2_F	451	0.313	0.434	0.888	0.378	0.475	0.865
NSP15_6W01_A_3_H	530	0.759	0.679	0.784	0.967	0.754	0.727
NSP15_6W01_AB_1_F	470	0.261	0.396	0.829	0.316	0.434	0.796
NSP16_6W61_A_1_H	583	1.044	0.795	0.787	1.339	0.888	0.728
PLPro_6W9C_A_2_F	512	0.343	0.458	0.858	0.427	0.508	0.825
RDRP_6M71_A_2_F	461	0.311	0.430	0.855	0.384	0.479	0.823
RDRP_6M71_A_3_F	498	0.495	0.548	0.859	0.599	0.602	0.830
RDRP_6M71_A_4_F	463	0.382	0.481	0.837	0.465	0.528	0.803
RDRP_7BV1_A_1_F	394	0.312	0.433	0.853	0.378	0.477	0.823
RDRP_7BV1_A_2_F	531	0.497	0.546	0.848	0.603	0.597	0.817
RDRP_7BV1_A_3_F	451	0.453	0.524	0.849	0.550	0.583	0.818
RDRP_7BV1_A_4_F	420	0.385	0.481	0.873	0.476	0.536	0.844
RDRP_7BV2_A_1_F	589	0.304	0.428	0.821	0.369	0.469	0.784
RDRP_7BV2_A_2_F	422	0.441	0.516	0.839	0.562	0.581	0.798
RDRP_7BV2_A_3_F	510	0.466	0.531	0.830	0.579	0.590	0.791

Table 2: Table of released model's training details and validation scores. Released model files and corresponding code is avilable from the project GitHub.

A.9 Modeling Feature Details

1613 Features

Model	epoch	val loss	val MAE	val r^2
V5.1-100K-flatten-2	337	0.80	0.66	0.71
V5.1-100K-random-2	336	0.80	0.66	0.71
V5.1-1M-flatten-2	484	0.60	0.59	0.81
V5.1-1M-random-2	455	0.49	0.52	0.68

1826 Features

Model	epoch	val loss	val MAE	val r^2
V5.1-100K-flatten-2	313	0.97	0.74	0.85
V5.1-100K-random-2	330	0.81	0.67	0.71
V5.1-1M-flatten-2	462	0.60	0.59	0.81
V5.1-1M-random-2	456	0.52	0.54	0.67

Table 3: Impact of including Mordred 3-D descriptors in the training data for the different sampling strategies.

Sample Selection Strategy	epoch	val loss	val mae	val r2
V5.1-100K-flatten	337	0.80	0.66	0.71
V5.1-100K-random	336	0.80	0.66	0.71
V5.1-1M-flatten	484	0.60	0.59	0.81
V5.1-1M-random	455	0.49	0.52	0.68

Difference of 1M Samples - 100K Samples

Sample Selection Strategy	Δ epoch	Δ val loss	Δ val mae	Δ val r2
flatten	147	-0.20	-0.07	0.11
random	119	-0.31	-0.14	-0.03

Table 4: Comparison of 1M samples to 100K samples.