Drug Orchestra Summary

Daanish Bhayani, Cao Cong Luan Tran, Victor Urbina, Morgan Williams

09 February 2024

Contents

I. Introduction

DrugOrchestra, which is a deep multi-task learning (MTL) framework, uses computational approaches to jointly analyze datasets on drug responses, drug targets, and drug side effects. DrugOrchestra's goal is to make predictions and gain insight on connections when modeling these tasks so that drug mechanisms may be deciphered and further information provided on future drug development. With the connection among drug response, drug targets, and drug side effects being so strong, the MTL framework's goal is to resolve multiple prediction tasks while concurrently utilizing the similarities and differences across the tasks. DrugOrchestra is especially unique because of its capacity to form predictions on drugs that are unseen. To execute the goals of DrugOrchesta, a molecular structure-based drug representation that is pre-trained from millions of compounds as shared features is used. DrugOrchestra then utilizes a hard parameter sharing deep learning structure for optimization of all three tasks in conjunction.

II. Drug Discovery: Drug Target Dataset

Beginning with the drug target dataset, drug target interactions (DTIs) were obtained from Repurposing Hub, Drugbank, and STITCH. Excluded from every dataset is drugs whose SMILES strings were not present in PubChem, ensuring every compound had a correlating SMILES string representation. Furthermore, the STITCH database excluded DTIs with confidence scores less than 0.9. The amount of drugs in each dataset and overlap of drugs among the datasets can be seen in Figure 1 below whereas the results show that there are 2.7 Repurposing Hub gene targets, 3.6 Drugbank gene targets, and 3.1 STITCH gene targets associated with each drug and 5.7 Repurposing Hub drugs, 6.9 Drugbank drugs, and 15.3 STITCH drugs associated with each target on average.

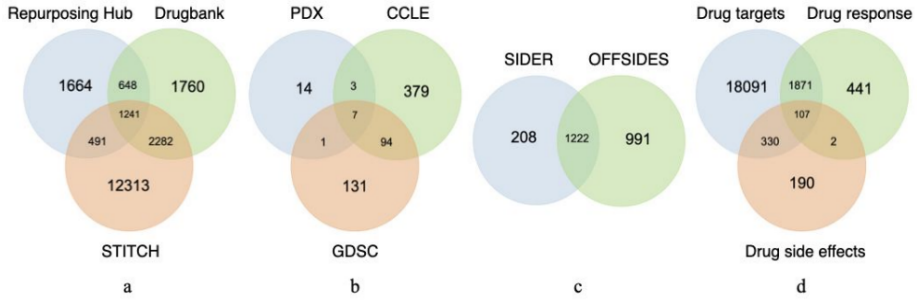


Fig. 1. Overlap of drugs among different datasets and tasks. a-c, Venn diagrams showing the number of drugs within each dataset and the overlap of drugs among different datasets in drug targets (a), drug response (b) and drug side effects (c). **d,** Venn diagram showing the number of drugs with each task and the overlap of drugs among different tasks.

SMILES, which stands for simplified molecular-input line-entry, is used to construct the vector representation of the drugs along with a pre-trained chemical molecular embedding model. The model allows for capturing domain-specific knowledge in the 2-D molecular graph that is depicted by a SMILES string. It was also trained on millions of unlabeled chemical structure data. The target gene features are obtained from pre-trained representations of genes in humans in addition to a variety of model organisms that were determined by the network embedding method named Mashup. Mashup integrates multiple individual networks into a low-dimensional space based on the topological relationship between the nodes. For each gene in the drug-target dataset, there is a map to a gene vector in the pre-trained gene vector file supplied by Mashup. Genes that were not present in the Mashup gene list were excluded.

III. Drug Discovery: Drug Response Dataset

Integrating datasets from Patient-Derived Xenografts (PDX), the Genomics of Drug Sensitivity in Cancer (GDSC), and the Cancer Therapeutics Response Portal (CCLE) resulted in a drug response collection. For drug response prediction values (IC_{50}), gene expression profiles were used as the

feature to indicate the drug response data in GDSC and CCLE. Best Tumor Response value was used to indicate drug response data in PDX. Drugs without a correlating SMILES strings in PubChem were excluded, and the z-score of the original response data was taken as a response value for each dataset. The table below shows the remaining drugs and their overlap among upon filtering.

Table 1. Statistics of the datasets included in our heterogeneous drug discovery dataset

Datasets	Drug	Target/Cell Line/se	Interactions
STITCH	16k	17k	890k
Drugbank	6k	3.1k	234k
Repur	4k	2.2k	120k
PDX	25	400	2k
GDSC	235	1k	190k
CCLE	483	1k	322k
SIDER	1.4k	1k	783k
OFFSIDES	2.2k	1k	830k

The features for the PDX model are established upon the gene expression matrix. This matrix shows relationships amidst cell lines and genes. Common genes that appear in all of the three datasets only are kept in the gene expression matrices of the drug-response datasets. Principal component analysis (PCA) is then performed to filter matrices for gene expression, and to obtain a features matrices that is low-dimensional. To normalize the feature matrices, the z-score is calculated across rows. Each column represents a feature vector for the corresponding cell line.

IV. Drug Discovery: Drug Side Effects

Drug side effects were collected from SIDER and OFFSIDES. Excluded were the drugs without corresponding SMILES strings in PubChem. With OFFSIDES, if drug-side effect associations reported ratio errors of a proportion greater than 0.25, they were excluded. A feature for a given disease is formulated by using associations. For each disease, the feature vector was a one-hot vector. One represented an existing association between a disease and gene while zero was otherwise. Excluded from DisGeNET were side effects that were not included as diseases. The methods of extracting cell line features included using PCA. PCA reduced the disease feature matrix (also the gene-by-disease matrix) to a dimension of 300. Z-score normalization was applied to the low dimensional vectors yielding to the left over side effects. In the matrix, each column represents a reduced feature matrix that is represented by a feature vector for each correlating disease.

V. Methods - DrugOrchestra Architecture

DrugOrchestra structure is made up of a neural network (NN) as the base classifier for each task that contains components of shared layers across all of the tasks and the task-specific layers. Parameters for the shared layers use hard parameter sharing while input features are pre-trained molecular graph-based drug features. Input features are different for each task for specific-task layers. The task-specific layers and shared layers outputs are then concatenated and used to train the task-specific NN for each of the tasks remaining layers. For this particular situation, $d_1, d_2, ..., d_n$ denotes input features for shared layers of n tasks, and input features for task-specific layers are denoted by $f_1, f_2, ..., f_n$ for n tasks, The activations of task i for shared layers is calculated using the formula: $q_k^i = \sigma(W_s^{k-1} q_i^{k-1} + b_s^k)$. In the formula, q_k^i represents the k-th shared hidden layer for the i-th task. σ represents the activation function. q_i^0 is the input drug feature which is represented by d_i. W_s^k and b_s^k are shared representations among the tasks representing weight and bias of the k-th hidden layer. For drug features from different datasets, the same hidden layers is used to process them and the common patterns are captured across tasks. The ReLU function was used as the activation function, σ, for hidden layers. For the activation, *sigma*, for the output layer of classification tasks (drug target prediction and side effect prediction),

the sigmoid function was used, and the linear activation function was used for the activation, *sigma* of the output layer of the regression task (drug response prediction).

For each group that uses task-specific features in the task-specific hidden layers, the activations, a_i^k, are defined with the formula: $a_i^k = \sigma(W_i^k + a_i^{k-1} + b_i^k)$. In the formula, W_i^k is the weight and b_i^k is the bias of the k-th task-specific hidden layer for the i-th task while a_i^0 represents the input task-specific feature, f_i. A key component of the tasks is that they can learn distinct feature representations which preserves the knowledge of each task separately. For the drug hidden layers and the task-specific hidden layers, two hidden layer were used, and a new feature vector, $c_i = [q_i, a_i]$, was obtained. The new obtained feature vector, C_i, for task i was acquired after processing the low-dimensional drug features and the task-specific features. C_i is then given to a task-specific NN along with the ReLU activation function. The output of the remaining task-specific NN is represented by o_i. The classification task has an output value of $o_i = z(C_i)$ and the regression task has an output value of $o_i = c_i$. Lastly, for the classification task the formula for entropy loss is represented by $L_c(d_i, f_i, y_i) = -(1/m)\sum_{k=1}^{m} y_i^k log o(d_i^{(k)}, f_i^{(k)} + (1 - y_i^{(k)}) log(1 - o(d_i^{(k)}, f_i^{(k)}))$. In the formula, m equals the number of training samples in the i-th task while $y_i^(k)$ represents the label of the k-th training sample in the i-th task. For the regression task, the mean squared error loss is used with the given formula: $L_r(d_i, f_i, y_i) = 1/m(\sum_{k=1}^{m}(y_i^{(k)} - o(d_i^{(k)}), f_i^{(k)})^2)$. From each task, minimizing the weighted combination of losses is the overall objective of the learning model. This minimum objective is defined as: $\min_o \sum_{c \in C} \lambda_c L_c(d_c, f_c, y_c) + \sum_{r \in R} \lambda_r L_r(d_r, f_r, y_r)$. In the formula, C refers to a set of classification tasks while R refers to a set of regression tasks. λ_i indicates the weight of a given task i. Another unique feature of DrugOrchestra is that the multi-task learning framework is flexible to include more tasks involving drug features by implicitly augmenting the training data to leverage other tasks to regularize a specific task. With this, over-fitting can potentially bealleviated.

VI. Methods - Dynamic Weight Adjustment

A schema is used that can instinctively adjust each task's weight. The specific weight adjusting strategy, referred to as a dynamic weight adjustment (DWA) strategy, begins by using the two previous epochs to calculate the relative descending rate, ω. The formula is $\omega_k^{t-1} = (L_k^{t-1})/(L_k^{t-2})$. In the formula, t is the t-th epoch, k the k-th task, and L_K^t the loss of the k-th task in the t-th epoch. The second step in the strategy in to calculate the importance weight, λ_k^t, of task k in the t-th epoch. The formula is given as $\lambda_k^t = \frac{e^{\frac{\omega_k^{t-1}}{T}}}{\sum_i e^{\frac{\omega_i^{t-1}}{T}}}$. In the formula, T controls the effect of the task weighting. The smaller the T, the more the tasks determined by ω are aggressively focused on by the optimization process. The calculated weight can then be used to weigh the loss functions of each specific. The strategy allocates the higher weights to the tasks that have lower descending rates, thus boosting the reductions of losses for the tasks. In the experiments performed, T = 2.

VII. Experimental Setup

The multi-task learning framework was compared against the Linear model, the Ensemble model, and Single-task Learning comparison approaches. For the Linear Model, the Support Vector Machine (SVM) is applied to each single dataset with a linear kernel, an 'optimal' learning rate schedule is set, and training stops until convergence. For the Ensemble model, the Random Forest was applied to each single dataset. Implementation uses sklearn library with trees set to 100 and depth set to 10. For Single-task Learning, a single-task deep NN (STL-NN) model was used. The STL-NN utilizes the same DrugOrchestra architecture except it is trained on a single task only. Each comparison approach uses all the available training samples from the corresponding datasets despite being trained on a single task. In addition, the same pre-trained molecular features, input features, and data splits used by DrugOrchestra are used by each approach. Also, the same learning rate and number of epochs as before are used to train DrugOrchestra and STL-NN.

Drug target and drug side effect predictions are classification tasks, so, the area under the curve for Receiver Operating Characteristic (AUROC) and the area under the Precision Recall Curve (AUPRC) are used to assess execution of drug target and drug side effect prediction. Large AUROC and AUPRC

values indicate better performance. The drug response prediction regression task uses the Spearman's Correlation Coefficient (Spearman) and Mean Squared Error (MSE) to evaluate performance. Large SCC values with small MSE values indicate better performance.

Performance gain is used to define transferability among the source dataset (task) and the target dataset (task). The formula is given as $g_{s\to t} = (r_{s\to t} - r_i)/r_t * 100$. The representation of performance of training DrugOrchestra, jointly, on the full source dataset, s, and the training set of data, t, then testing on the test set of dataset t is represented by $r_{s\to t}$. To compute the Tanimoto score among the two drugs, the RDKit package is used, whereas a large Tanimoto score shows similarity between the two drugs in terms of SMILES strings.

Lastly, to evaluate and train the models, a 3-fold cross validation is used. Each time, 2/3 drugs are randomly selected for training while the remaining 1/3 are used for testing. The evaluation setting is able to determine whether the method can be used for unseen drugs due to the lack of overlap among test drugs and training drugs. The process is repeated 10 times with the average results used as the final performance. For the STL-NN model and DrugOrchestra, ADAM optimizer with a learning rate of 10^-3 and batch size of 250 for 20 epochs is used for training. The Xavier approach initialized the parameters of all layers. Binary cross-entropy (BCE) loss is used to computer drug-target prediction tasks, drug side effect prediction tasks, and the MSE loss for drug response prediction tasks.

VIII. Experimental Results - Performance improvement on all three tasks

The results found that 7 out of 8 datasets had significant improvement, with a t-test p-value less than 0.5. DrugOrchestra GDSC achieved 0.375 SCC. This value is higher than the SCC values of 0.315 from STL-NN, 0.235 from SVM and 0.149 from Random Forest. The other 6 datasets outperformed comparison methods significantly as well. An exception, however, is the PDX, whereas DrugOrchestra is superior to RF, but worse than SVM. The linear SVM performance over PDX is contributed to the smaller size of the PDX, which could easily result in over-fitting because of more expensive models. Overall, the improved MTL performance demonstrated effectiveness in transferring knowledge across the three tasks using multi-task learning. DrugOrchestra also had very prominent improvements on drug response prediction, thus suggesting stronger connections among drug response and the two other tasks. DrugOrchestra is especially distinct because it models multiple datasets simultaneously, thereby taking advantage of the shared features to acquire enhanced drug representation.

Datasets	Metrics	SVM	Ramdom Forest	STL-NN	DrugOrchestra	Naïve Bayes	Linear	Logistic	KNN	Decision Tree	Gradient boosting
STITCH	AUROC	0.668±0.048	0.66±0.032	0.827±0.031	**0.833±0.047**	0.823±0.001	-	0.889±0.015	**0.966±0.0003**	0.911±0.001	0.901±0.001
	AUPRC	0.320±0.092	0.349±0.053	0.583±0.034	**0.587±0.088**	0.3594±0.003	-	0.519±0.004	**0.826±0.002**	0.64±0.003	0.61±0.004
Drugbank	AUROC	0.602±0.042	0.616±0.037	0.740±0.034	**0.752±0.012**	0.7302±0.003	-	0.762±0.003	**0.890±0.0003**	0.727±0.001	0.803±0.002
	AUPRC	0.235±0.048	0.269±0.055	0.378±0.063	**0.402±0.023**	0.366±0.005	-	0.405±0.005	**0.626±0.004**	9.285±0.0007	0.49±0.006
Repur	AUROC	0.530±0.010	0.585±0.049	0.691±0.014	**0.703±0.031**	0.711±0.004	-	0.740±0.004	**0.868±0.003**	0.685±0.003	0.779±0.004
	AUPRC	0.125±0.018	0.221±0.066	0.293±0.023	**0.321±0.036**	0.263±0.007	-	0.267±0.002	**0.554±0.006**	0.22±0.003	0.403±0.0007
PDX	SCC	**0.491±0.015**	0.436±0.017	0.468±0.094	0.465±0.014	-	0.342±0.02	-	-	0.17±0.06	0.36±0.01
	MSE	**0.824±0.034**	0.864±0.034	0.856±0.028	0.840±0.031	-	0.878±0.08	-	-	1.64±0.1	0.86±0.06
GDSC	SCC	0.235±0.016	0.150±0.0140	0.315±0.020	**0.375±0.024**	-	**0.07±0.001**	-	-	0.02±0.005	0.15±0.01
	MSE	1.150±0.039	1.110±0.028	0.900±0.038	**0.814±0.061**	-	0.99±0.02	-	-	4.56±0.3	0.97±0.02
CCLE	SCC	0.206±0.016	0.181±0.012	0.394±0.022	**0.410±0.010**	-	**0.09±0.01**	-	-	0.03±0.01	0.21±0.01
	MSE	1.472±0.046	1.147±0.031	0.930±0.026	**0.882±0.037**	-	0.99±0.01	-	-	5.25±0.23	0.95±0.01
SIDER	AUROC	0.591±0.026	0.640±0.036	0.660±0.026	**0.671±0.036**	0.710±0.0009	-	0.694±0.001	**0.801±0.0003**	0.661±0.0002	0.811±0.001
	AUPRC	0.231±0.022	0.295±0.035	0.312±0.016	**0.321±0.034**	0.335±0.001	-	0.378±0.001	**0.438±0.002**	0.232±0.001	0.51±0.001
OFFSIDES	AUROC	0.625±0.009	0.638±0.029	0.660±0.025	**0.686±0.011**	0.643±0.001	-	0.649±0.001	**0.784±0.0004**	0.66±0.0006	0.799±0.0004
	AUPRC	0.399±0.005	0.435±0.013	0.432±0.022	**0.457±0.037**	0.440±0.002	-	0.503±0.001	**0.569±0.0008**	0.398±0.0005	0.647±0.001

We conduct six models to compare the results with those from the paper, such as Naive Bayes, Linear Regression, Logistic Regression, KNN Classification, Decision Tree and Gradient boosting. We use numpy and scikitlearn libraries to perform models and run it repeatedly for 10 times. In the research, they use 3 fold validation technique, so that in our model, we do have to use 3 fold validation to test on the dataset. At the beginning, we forecast that the DrugOrchestra will have a better result. In contrast, KNN classification have a better score on STITCH, Drugbank and Repurposing hub datasets. Linear Regression gets highest score on finding SCC and also lowest score on finding MSE.

We have discussed about why KNN model scores a significantly higher than other models. Some reason can lead to that problems:

- Feature Space: KNN is a non-parametric method, meaning it makes no assumptions about the distribution of data. It relies solely on the feature space and the similarity measure. If the underlying

data distribution is complex and non-linear, KNN might capture it better than linear models like SVM or parametric models like neural networks.

- Data Distribution: KNN performs well when the decision boundary is irregular or when the classes are not easily separable. If the data has a complex structure with regions of varying densities or shapes, KNN might be able to capture this better than other models.

- Amount of Data: KNN doesn't require training time as it simply memorizes the training data. If you have a relatively small dataset, KNN might perform well because it doesn't suffer from overfitting like more complex models might.

- Hyperparameters: The performance of KNN can heavily depend on the choice of the hyperparameter k (number of nearest neighbors). Tuning this hyperparameter carefully might yield better performance. Other models like SVM or neural networks also have hyperparameters that need to be tuned, and their performance can depend on how well these are chosen.

- Imbalanced Data: KNN can perform well on imbalanced datasets since it doesn't make any assumptions about the class distribution. It's sensitive to the local density of points rather than the overall class proportions.

- Preprocessing: The performance of KNN can also depend on the preprocessing steps applied to the data. For example, scaling features can significantly affect KNN's performance since it relies on distance metrics.

Below picture is the EDA (basic statistics for data and labels) of the datasets that we use for computing the results of four models.

Datasets	Samples	Features	Target
STITCH	890628	1100	1
Drugbank	234528	1100	1
Repur	119878	1100	1
PDX	1634	600	1
GDSC	190853	600	1
CCLE	322045	600	1
Sider	783866	600	1
OFFSIDE	829695	600	1

IX. Experimental Results - Transferability across Datasets

Performance gain was used to calculate transferability across datasets. The results, as shown in Table 3 below, show that datasets that belong to the same task have greater transferability in comparison to datasets from different tasks. Drug response average performance gain was 1.9%, 4.71% for drug targets, and 2.75%. These values showed to be higher than the corresponding values of 0.274%, 1.85%, and 0.544% for average performance gain between datasets of different tasks. Because of the closer distribution of the datasets from the same task, drug representation can be shared across them more easily. Table 2 also shows that transferability shows partial reflection on the improvement of DrugOrchestra whereas the GDSC dataset had the highest improvement of 18% from the CCLE dataset, while also showing the highest improvement over STL with 19%. Table 3 also shows negative transferability, mostly among datasets not from the same task. The exception was the transferability from PDX to CCLE. This exception shows to be consistent with the worst performance of DrugOrchestra on PDX. This is depicted in Table 2. The remaining negative values for transferability between the drug side effect dataset and drug target dataset tasks are attributed to inherent differences.

Table 3. Transferability in terms of performance gain across datasets. Tasks in rows are source tasks and tasks in columns are target tasks. The source task is used to help the training of the target task. AUROC and SCC are used as evaluation metrics for the classification task (i.e., drug side effect prediction and drug target prediction) and the regression task (i.e., drug response prediction), respectively.

Datasets	STITCH	Drugbank	Repur	PDX	GDSC	CTRP	SIDER	OFFSIDES
STITCH	0	2.414	2.356	1.399	3.700	1.149	0.953	0.543
Drugbank	**1.090**	0	**2.642**	0.9651	6.579	0.193	0.630	1.501
Repur	0.436	**2.503**	0	1.428	7.377	2.198	0.311	1.761
PDX	-0.255	0.304	0.829	0	1.707	-1.576	-0.083	-0.372
GDSC	0.004	0.504	0.679	2.221	0	**7.261**	0.111	0.382
CTRP	0.205	0.711	0.686	0.485	**18.174**	0	0.344	0.444
SIDER	-0.628	1.128	0.482	1.670	3.318	-2.928	0	**3.733**
OFFSIDES	-0.427	-0.027	-0.084	**2.251**	2.342	-3.970	**1.767**	0

Dataset	STITCH	Drugbank	Repur	PDX	GDSC	CCLE	SIDER	OFFSIDES
STITCH	*0*	*0.0625*	*-0.1167*	1.399	3.7	1.149	0.953	0.543
Drugbank	*0.0625*	*0*	*-0.2087*	0.9651	6.579	0.193	0.63	1.501
Repur	*-0.1167*	*-0.2087*	*0*	1.428	7.377	2.198	0.311	1.761
PDX	-0.255	0.304	0.829	*0*	*-0.3024*	*-0.277*	-0.083	-0.372
GDSC	0.004		0.679	*-0.3024*	*0*	*0.005*	0.111	0.382
CCLE	0.205	0.711	0.686	*-0.277*	*0.005*	*0*	0.344	0.444
SIDER	-0.628	1.128	0.482	1.67	3.318	-2.928	*0*	*0.1555*
OFFSIDES	-0.427	-0.027	-0.084	2.251	2.342	-3.97	*0.1555*	*0*

We re-perform the transferability to get the performance gain across datasets. That means we use one third of a dataset to test, two third of that for training. Now we are gonna use the first two third from 1 data set, the one third will be from a different dataset. The results is in italic and it is quiet different from the paper. Reason for that situation could be the numbers of model we use.

X. Experimental Results - Revealing Transferability Across Tasks

Transferability at the task level for drug side effect and drug target datasets were examined by performing analysis by aggregating datasets from the same task. In Table 4, the results show 4 out of 6 task pairs with positive performance gain using DrugOrchestra. The most prominent improvement is shown when using drug targets to help drug response prediction. Drug targets help understand drug mechanisms as opposed to drug response prediction, making drug target features very important in prediction approaches. Drug phenotypes, drug response and drug side effects were shown to have substantial transferability, thus giving assurance that DrugOrchestra could be united with other datasets to enhance prediction performance on all of the three tasks.

Table 4. Transferability in terms of performance gain across tasks. Datasets belonging to the same task are combined as one dataset for that task. Tasks in rows are source tasks and tasks in columns are target tasks. The source task is used to help the training of the target task. AUROC and SCC are used as evaluation metrics for the classification task (i.e., drug side effect prediction and drug target prediction) and the regression task (i.e., drug response prediction), respectively.

Tasks	Drug targets	Drug response	Drug side effects
Drug targets	0	3.846	-0.002
Drug response	0.006	0	0.276
Drug side effects	-0.007	0.751	0

Table 5. Performance of DrugOrchestra and STL-NN by only considering drugs above a specific Tanimoto threshold. AUROC is used for evaluating drug target datasets and drug side effect datasets, and SCC is used for evaluating drug response datasets.

Methods(Threshold)	Datasets							
	STITCH	Drugbank	Repur	PDX	GDSC	CCLE	SIDER	OFFSIDES
STL-NN(0.6)	0.745±0.055	0.727±0.025	0.634±0.043	0.486±0.019	0.318±0.016	0.295±0.021	0.639±0.050	0.654±0.029
DrugOrchestra(0.6)	**0.785±0.081**	**0.747±0.043**	**0.655±0.042**	**0.501±0.014**	**0.371±0.020**	**0.314±0.026**	**0.646±0.036**	**0.687±0.034**
STL-NN(0.7)	0.775±0.038	0.732±0.034	0.651±0.043	0.468±0.020	0.312±0.014	0.338±0.014	0.643±0.049	0.654±0.024
DrugOrchestra(0.7)	**0.806±0.053**	**0.746±0.052**	**0.668±0.021**	**0.477±0.016**	**0.380±0.016**	**0.358±0.018**	**0.648±0.030**	**0.689±0.023**
STL-NN(0.8)	0.788±0.036	0.741±0.024	0.667±0.031	0.463±0.017	0.316±0.012	0.355±0.013	0.647±0.041	0.652±0.030
DrugOrchestra(0.8)	**0.812±0.049**	**0.749±0.040**	**0.683±0.025**	**0.466±0.016**	**0.382±0.017**	**0.374±0.018**	**0.651±0.028**	**0.686±0.032**
STL-NN(0.9)	0.806±0.047	0.751±0.020	0.679±0.031	**0.467±0.088**	0.322±0.078	0.373±0.011	0.651±0.039	0.654±0.031
DrugOrchestra(0.9)	**0.827±0.027**	**0.756±0.038**	**0.692±0.030**	0.462±0.015	**0.388±0.018**	**0.394±0.016**	**0.658±0.027**	**0.687±0.031**

Tasks	Drug targets	Drug response	drug side effect
Drug targets	0	0.64125	0.66387
Drug response	-0.64125	0	0.022619
drug side effect	-0.663869	-0.022619	0

We re-perform the transferability to get the performance gain across tasks. It means we we concatenate the datasets based on the tasks such as: drug target (STITCH, Drugbank, Repurposing hub), drug response (PDX, GDSC, CCLE) and drug side effects (SIDER, OFFSIDES). And then we re-perform model across datasets. Missing value between column's numbers can be filled in with 0 or mean values. And perform PCA or SVM to reduce dimension because the size of new datasets are quite large.

XI. Experimental Results - Prediction of Unseen Drugs

DrugOrchestra is unique due to its ability to make predictions with unseen drugs. The improvements, seen in table 2, show the test drug set and training drug set have no overlap. There is, however, a possible information leakage that could obscure the use of the method used to predict unseen drugs due to compounds with similar molecular structure that may still show in the test and training set. The investigation of the applicability domain excludes a test set drug if it has a high Tanimoto similarity to any of the drugs in the training set. Through use of different stringent Tanimoto similarity cutoffs, considerable improvement of DrugOrchestra in comparison to STL still showed. In addition, the method shows larger improvement against STL when using a more stringent cutoff, which is indicative of enhanced prediction performance of the prediction method for unseen drugs.

XII. EDA

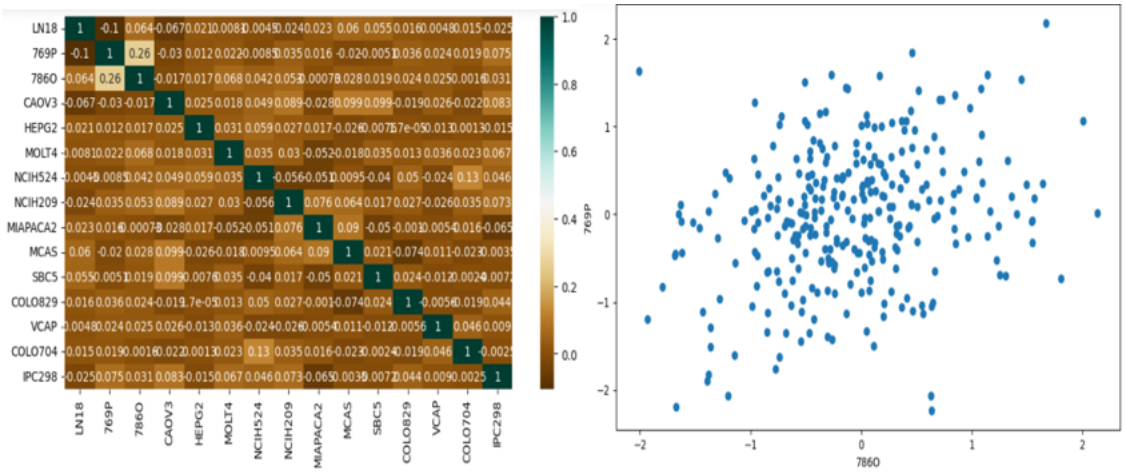
Implementation:

- There are 8 data set: drug target (Repurposing, Drug bank, STITCH), drug response (PDX, GDSC, CCLE) and drug side effect (offside, sider). Each contains a CSV file after taking z-score normalization.

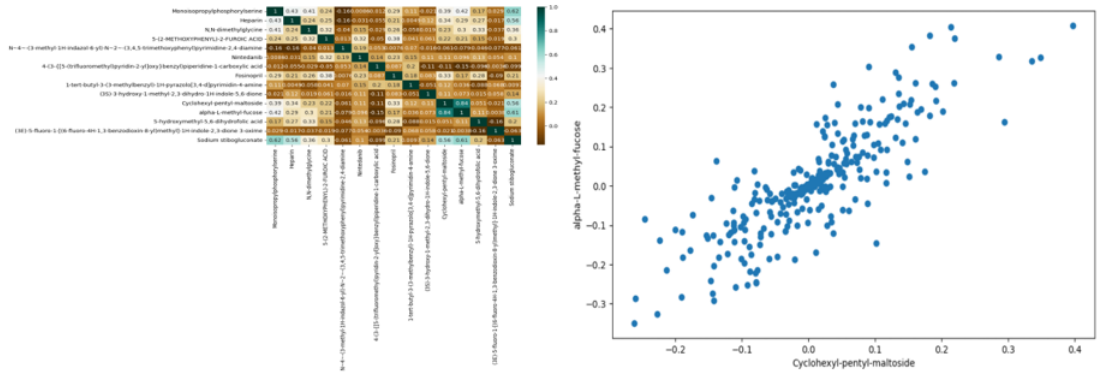
- We will perform EDA on each data set.

Result:

- We consider one example of drug response CCLE. According to the heatmaps, Drug response CCLE has no correlation. The scatter plot also implies there is no relation between 2 random drugs.



On the other hand, we pick the drugbank from drug target data set to analyze the correlation between two random drugs.



From the heat map, we can see there is a strong correlation between Cyclohexyl-pentyl-maltoside and Alpha-L-methyl-fucose reaching 0.84. To confirm it, we draw a scatter plot and we can see that there is a pattern, which is a linear relationship between them.

Cyclohexyl-pentyl-maltoside and Alpha-L-methyl-fucose are related in the sense that they are both chemical compounds used in biochemical research, particularly in the study of membrane proteins and glycobiology.

While they serve different purposes and have different chemical structures, they are both tools utilized by researchers to investigate biological processes. Cyclohexyl-pentyl-maltoside is a detergent used for solubilizing membrane proteins, allowing researchers to study their structure and function in solution. Alpha-L-methyl-fucose, on the other hand, is a modified form of the monosaccharide fucose, which is involved in various cellular processes, particularly in glycoprotein and glycolipid synthesis. Researchers may use Alpha-L-methyl-fucose to study the roles of fucose derivatives in biological systems or to develop glycan-based therapeutics.

XIII. Discussion

DrugOrchestra is a multi-task learning approach that can predict unseen drugs, which is a key advantage compared to other approaches. It improves predictions when the test drug set does not

overlap with the training drug set. However, compounds with similar molecular structures may still appear in the test and training sets, potentially causing information leakage. Using different stringent Tanimoto similarity cutoffs, DrugOrchestra outperforms STL-NN in STITCH and GDSC, and all other datasets, including PDX.

XIV. Contribution

Group 6:

- Cao Cong Luan Tran: EDA, report, Naive Bayes, Gradien Boosting and correct code error.
- Morgan Williams: report, KNN model.
- Daanish Bhayani: Linear model.
- Victor Urbina: Logistic model.

Github