

# Odysseia: Genetic Regulatory Feature Analysis with Interpretable Classification Machine Learning Models

Jack Yu<sup>1,\*1</sup> and Jiawang Tao<sup>1</sup>,

<sup>1</sup>Affiliation, department, city, postcode, country

\*<sup>1</sup>Correspondence: gyu17@alumni.jh.edu

## ABSTRACT

With rapid progress of robust single-cell transcriptome sequencing since last decade, numerous complicate mechanism underlying cell development has been revealed. Yet, single-cell RNA sequencing (scRNA-seq) analysis is widely accepted as main approach to define cell stages and phenotypes. As conversion of somatic cells into induced pluripotency cells already succeeded, identification key genetic factors(GFs) with scRNA-seq for cell reprogramming in biological research and regenerative medicine fields has gained great attention. Herein, we describe *Odysseia*, an interpretable machine learning classifier based single-cell gene expression profile(scGEP) analysis system, that assesses importances of genetic regulatory features in differentiating cell state(CS). Furthermore, combined with regulatory network analysis, extracted factors can help to find specific key GFs in implementing CS conversions. Analyzing 3 published cell reprogramming related reports studying diverge cell types, *Odysseia* correctly extracts GFs acclaimed being capable to induce CS conversion. Overall, *Odysseia* provides an option to obtain guidance information while elucidating mechanism to engineer cellular phenotypes.

## Introduction

Since cell reprogramming of mouse embryonic fibroblasts(MEF) to induced pluripotent stem cell(iPSC) succeed in 2006<sup>1</sup>, several more CS transition methods have been established both in vivo and in vitro<sup>2-4</sup>, spurring prospective development in biomedical area. However, reprogrammed cells could still fail to flawlessly match in vivo counterparts' identities<sup>5</sup>, and precise reprogramming toward cell subtypes remain impossible<sup>6</sup>. As gene expression profiles for each individual cell in query population became accessible with scRNA-seq, substantial progress in accurately defining of cell subtypes with marker gene expression has been made<sup>7,8</sup>. Even though multiple computational methods, like *CellNet*<sup>9</sup>, method of *D'Alessio et al*<sup>10</sup>, and *Mogrify*<sup>11</sup>, are already available and proved informativity of determining key transcription factors(TFs) for CS conversion via carrying out novel human cell conversion experiments, limited generalized method could achieve similar goal while utilizing scRNA-seq based expression profiles instead of microarray based or Cap analysis gene expression (CAGE) based databases. Indeed, constructing scRNA-seq based gene expression database is possible to expand existing methods' applicability in predicting key TFs for cell subtype conversions and elucidating detailed mechanism behind cell reprogramming. But workload of maintaining complicated database covering majority discovered cell subtypes can be massive; hence, developing generalized scRNA-seq based predictive computational method for determining key GFs in CS conversion with existing methods remain challenging. In 2019, *SingleCellNet*<sup>12</sup> is published as successor of *CellNet* in classifying cellular samples with scRNA-seq data but not guiding CS conversion.

Here we present *Odysseia*, a scRNA-seq based expression profile analysis system utilizing interpretable machine learning classifiers, for not only guiding CS conversion method design but also elucidating mechanisms behind CS conversions. Unlike previous described methods, *Odysseia* does not require any background expression database but finding potential key GFs in conversing one CS to another with only expression profiles labeled with binary CS categories as input. Important genetic regulatory pathways(GRPs) being deterministic for classifying CS category will be addressed through assessing and interpreting trained classifiers. Then, key GFs in performing CS conversion will be addressed through analyzing regulons constructed with important GRPs described previously. The main reason to keep this design is due to consideration on the assumption: The key GFs and GRPs may not have converged profiles in either expression or regulatory aspect. Thus, classification model designed to directly find CS deterministic GFs or GRPs can potentially fail to converge on a stable solution. One of the intuitive approach to solve the non-convergence issue is transforming classification question into higher dimension. In our case, instead of asking which factors are CS deterministic, we can transform the question into determining CS category, which is supposed to be a converged and labeled profile among input data, with given GEP. Then, we can analysis the well-performing classifiers and determine which features, equivalent to GRPs, are essential for the classifiers to make correct decisions.

However, in this approach, if a few feature associating with similar functions can be informative enough, CS classifiers

could reach remarkably high accuracy with limited features being heavily weighted. As a result, partial of potential key features will presumably be lost even classifiers being analyzed has demonstrated accuracy over testing data, and informativity of system output will be questionable. **For example, cell growth related genes can also be used as marker genes when separating cells not only in different types but also on distinct cell cycle stages. But, if only cell growth related genes are found after classifier analysis, system output's capability in guiding CS conversion is unlikely being sufficient.** Therefore, a particular classification model designed solving transformed question can still encounter feature lost issue.

In order to overcome feature lost issue as well, *Odysseia* applies multiple classification models with distinct architecture settings in each training iteration then selects well-performing classifiers for further assessment and interpretation. Through interpreting top-performing classifiers trained with different batches of input data, *Odysseia* enhances the feature extraction capability.

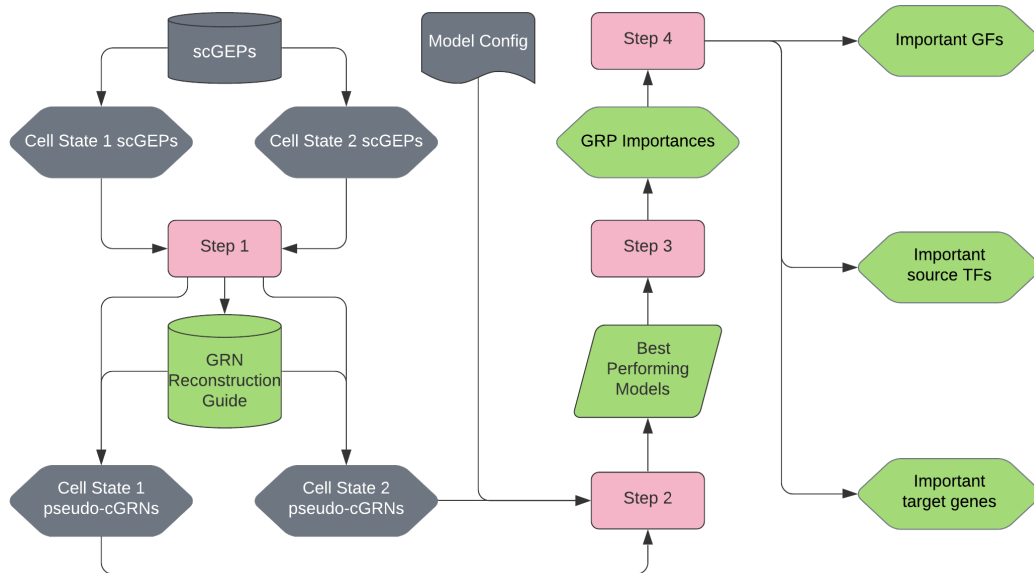
Further details and testing results are discussed in the remainder of this paper which is structured as follow:

- **Method:** firstly, we will summarize the overall system design and describe how it can possibly avoid both non-convergence and feature lost issues.
- **Result:** secondly, we perform extensive analysis using diverse collection of sequencing data to demonstrate that *Odysseia* can be applied into real world problems, especially in finding potential GFs to perform cell reprogramming.
- **Discussion:** third, we discuss about how the outputs of this system can be utilized in further wet lab experiments and what further improvements can be done on this system.
- **Conclusion:** finally, we conclude this paper.

## Methods

Briefly, *Odysseia* consists of 4 main steps which can be summarized below and visualized as Figure 1:

- **Step 1:** Generate pseudo-cellular gene expression profiles(pseudo-cGEPs) and corresponding pseudo-cellular genetic regulatory network(pseudo-cGRN) with genetic regulatory network(GRN) reconstruction guidance.
- **Step 2:** Train classification models with GRNs reconstructed with pseudo-cGEPs then select out well-performing models via accuracy evaluation.
- **Step 3:** Interpret selected classifiers' correct predictions on all generated GRNs.
- **Step 4:** Perform regulon based analysis on important GRPs found from classifier interpretation in step 3 and extract important GFs indicated by the GRPs.



**Figure 1.** The overall workflow of *Odysseia*

## Step 1: Data preprocessing

To reconstruct GRNs based on scGEPs, a computational method measuring interaction strength among gene pairs is mandatory. One of the widely adopted methods is utilizing Pearson correlation coefficient (PCC)<sup>13,14</sup>. Nevertheless, calculating PCC requires sequence of expression levels (ELs) for each gene among interaction rather than single EL. scRNA-seq can provide in one cell. Hence, in order to reconstruct cell-level GRNs, *Odyssea* segments scGEPs under same CS category into subsets with constant size to form pseudo cGEPs, having gene ELs gained from different cells abstracted as sequence of altering ELs among single pseudo cell.

Under circumstances of scGEPs being scarce, sliding window algorithm (SWA) with customized window size and padding stride can be applied to ensure pseudo-cGEP amount being sufficient for later classifier training and assessment processes. With SWA,  $i$ -th pseudo-cGEP can be obtained through:

$$SWA(i) = \{x_{j=j+i*s}^{j+l}\}, j+l < N$$

Here  $N$  denotes for total length of scGEPs gained from scRNA-seq which can be expressed as  $\{x_{j=0}^N\}$ ;  $l$  denotes for window size;  $s$  denotes for padding stride.

Although pseudo-cGRNs will be reconstructed from pseudo-cGEPs respectively, a meta-level process on determining which gene pairs would form regulatory relationships in each pseudo-cGRN can dramatically reduce computational resource requirement than repeatedly testing all possible gene pairs in every pseudo-cGEP. Assuming pseudo-cGEPs have similar expression pattern with comprehensive GEP containing all scGEPs, the GRN reconstruction guidance can be created through analyzing all gene pairs in comprehensive GEP. The overall workflow to create GRN reconstruction guidance can be summarized as Figure 2. In total, two sets of filters must be passed for validating a GRP:

### 1. Gene level:

The main purpose of this filter set is confirming whether a gene's expression pattern is differentiable among binary CS categories or not.

- **Stdev Filter:** The standard deviation (stdev) of gene's EL must be significant. This filter is to confirm query gene not only can reach relatively high EL but also has expression pattern possibly interacting with other GFs. By default, the threshold is set to 1 implying that one gene's expression status must be changing in different circumstances to be considered.

However, stdev threshold would need to be adjusted significantly considering not only significant weight of expression change in studying data but also computational resource capacity.

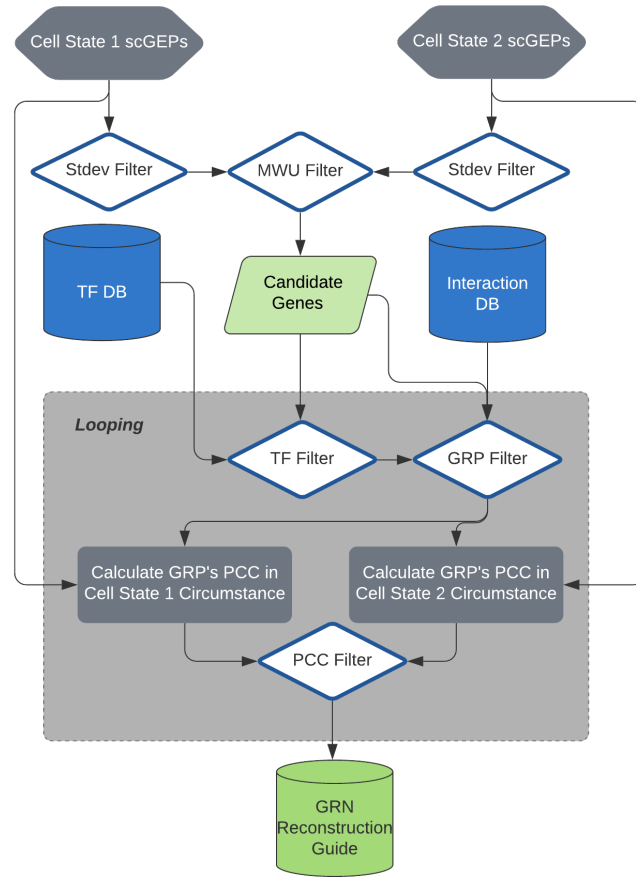
- **MWU Filter:** The expression distribution of selected gene across samples labeled with different CSs must be dissimilar. This filter is to confirm query gene's GEP is likely distinguishable across sample sets under binary CS categories. The Mann-Whitney U rank test (MWU) implemented by *SciPy*<sup>15</sup> will be performed on GEPs of selected gene in binary CS categories. By default, the p-value to reject null hypothesis, expression pattern underlying CS1 is the same as the expression pattern underlying CS2, is set to 0.05.

### 2. Pathway level:

The main purpose of this filter set is confirming whether a GRP is both biologically interpretable and significant or not.

- **TF Filter:** The source input of a GRP must be a recorded TF. If not further specified, TF list will be retrieved from integrated *TRANSFAC*<sup>16</sup> datasets according to the species information in query.
- **GRP Filter:** Regulatory target must have binding ability with source TF. If not further specified, regulatory gene list of given TF will be retrieved from integrated *GTRD*<sup>17</sup> datasets according to the species information in query.
- **PCC Filter:** Expression correlation between source TF and target gene must be significant. The default setting of threshold is 0.2 for absolute value of Pearson correlation coefficient calculated with *SciPy*<sup>15</sup>.

Utilizing GRN reconstruction guidance, pseudo-cGRN will be reconstructed for each pseudo-cGEP. Each gene in the guidance, if also presenting in pseudo-cGEP, need to pass the stdev filter in pseudo-cGEP circumstance before validating corresponding GRPs in the guidance. The pre-validated GRPs also passing PCC filter will take part in eventual pseudo-cGRN.



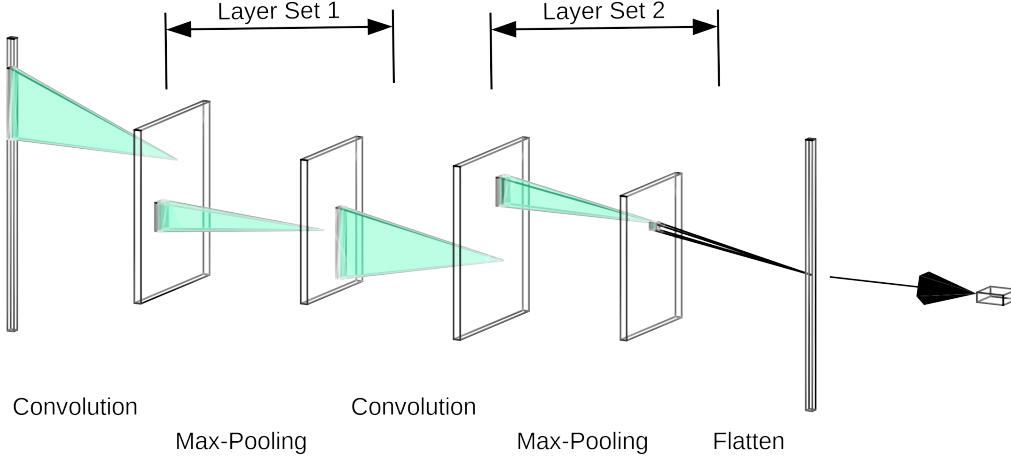
**Figure 2.** Workflow to create GRN reconstruction guidance from comprehensive GEPs. **(1)** Each gene in scGEP set either for CS1 or CS2 need to pass stdev filter confirming its expression pattern is reacting with inconstant gene expression circumstances. **(2)** GEPs of genes passed stdev filter will be input to MWU filter for confirming gene expression pattern is differentiable among binary CS categories. **(3)** All TFs in genes passed MWU filter will be selected out as potential regulatory source TF utilizing given TF database. **(4)** Using GF or protein-protein interaction databae, all genes passed MWU filter will be testified as target genes with potential TF found at previous step to form potential interacting gene pairs. **(5)**, PCCs of potential interacting gene pairs found previously will be calculated under each CS category. A gene pair must have correspond PCC under at least one CS category greater than threshold to be considered as potential GRP.

## Step 2: Classifier selection

According to given model config file, vanilla models will be initialized with specified model type and architecture parameters. While iteratively training classifiers with split pseudo-cGRN sets, *Odyseia* will also perform accuracy tests systematically aiming to select out potentially well-performing models for further analysis. The overall workflow can be summarized as Figure 4. Currently, *Odyseia* supports three types of classification models:

- **SVC:** Support Vector Classifier implemented with *scikit-learn*.<sup>18</sup> All parameters adjustable with *scikit-learn*<sup>18</sup> are supported.
- **GBTree:** Gradient Boosting Trees implemented with *XGBoost*.<sup>19</sup> All parameters adjustable with *XGBoost*<sup>19</sup> are supported.
- **CNN:** Convolutional Neural Network implemented with *Pytorch*.<sup>20</sup>

More specifically, the general architecture designs of *Odyseia*'s integrated CNNs are implemented referring to 1D-CNN and 2D-Hybrid-CNN applied in recent cancer type prediction study<sup>21</sup>. Unlike original 1D-CNN and 2D-Hybrid-CNN, we implemented the models with flexibilities not only on kernel related parameters but also on others including amount of convolution layer set which consists a convolution layer and adjacent max-pooling layer. An example of 1D-CNN with 2 convolution layer sets can be illustrated as Figure 3.



**Figure 3.** 1D-CNN with 2 convolution layer set.

Other CNNs implemented with *Pytorch*<sup>20</sup> can also be analyzed with *Odysseia* but training protocol need to be clarified by user in advance.

By default model config file, 81 model builds consisting of 1 SVC, 16 GBTrees, and 64 CNNs will be used to initialize vanilla models. In each training iteration, each vanilla classifier will be fed with same set of pseudo-cGRNs as training samples and primarily assessed according to prediction accuracy on untrained pseudo-cGRNs. By default setting, at each iteration, 70% of pseudo-cGRNs will be split into training set while 30% will be split into testing set. The classifiers capable of passing accuracy threshold will be kept for further comparisons with other classifiers trained with different sample sets in afterward iterations.

At the end of this step, all available pseudo-cGRNs will be joint as testing data, and all candidate classifiers will be evaluated on their prediction accuracy of fully joint testing data. By default, considering the true randomly choosing can reach an accuracy of 50% in binary classification question, the accuracy threshold for either local accuracy test or fully joint data test is set to 90%.

### Step 3: Classifier interpretation

During last accuracy assessment in Step 2, all classifier's predictions consisting with ground facts will be marked with corresponding input pseudo-cGRNs. Briefly, in this step, *Odysseia* interprets how well-performing classifiers make correct predictions on CS categories of given pseudo-cGRNs. Integrating interpretations of all well-performing classifiers, the generalized importance values of features, which are GRPs in *Odysseia*'s scenario, in determining CS can be approached.

To analysis feature importance, several packages already have internalized methods implemented for machine learning models. For example, GBTree implemented with *XGBoost*<sup>19</sup> can have feature importance approached through calculating average weight gain at each split involving the feature, and linear SVC implemented with *scikit-learn*<sup>18</sup> can have the importance approached with feature coefficient. Regarding the standard differences between these internalized methods, we utilize *softmax* function to normalize feature importance and define the normalized importance calculation function as:

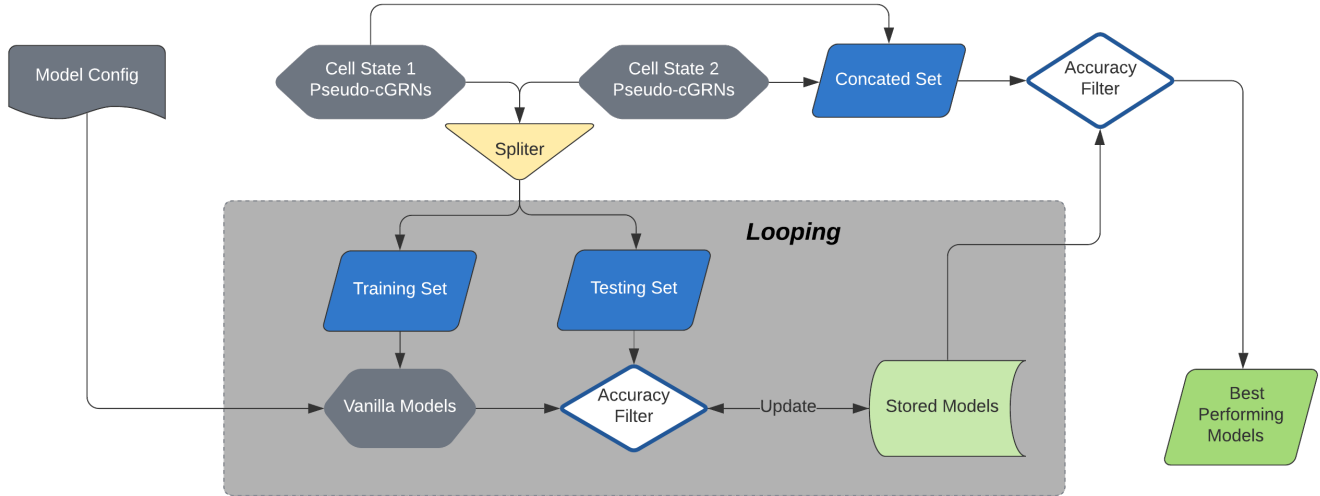
$$T(Z) = \left\{ \frac{f(Z_i)}{\sum_{j=1}^L f(Z_j)} \right\}, i \in Z$$

Here  $Z$  is feature set;  $L$  is total number of features;  $f(X)$  is internalized feature importance calculating function of classification model.

However, for numerous machine learning models, there is no internalized feature importance approaching method, for example NN based models implemented with *Pytorch*<sup>20</sup>, or generalized method cannot be appropriately applied with selected models, such as Kernelized SVC implement with *scikit-learn*<sup>18</sup>. Therefore, we apply the concept of The Shapley value<sup>22</sup> to build an universal approach for determining importance of each feature in any kind of machine learning model. Specific Shapley value calculation or approaching methods are implemented with *SHAP*<sup>23</sup> and applied to different types of classifiers as shown in Table 1.

Let  $N$  denoting total number of correct predictions,  $\phi_{x,j}^i$  and  $\phi_{y,j}^i$  denoting Shapley value of feature  $i$  when predicting sample  $j$  as type  $x$  or type  $y$ , the function determining importance values of feature set  $z$  can be expressed as:

$$T(Z) = softmax\left(\left\{ \sum_{j=1}^N \frac{|\phi_{x,j}^i| + |\phi_{y,j}^i|}{2} \right\}, i \in Z\right)$$



**Figure 4.** Workflow to generate well-performing classifiers. (1) In each training iteration, input pseudo-cGRNs will be split into training set and testing set according to given ratio. By default, the ratio is set to 70% for training and 30% for testing. (2) According to given classification model config file, vanilla classifiers will be initialized and trained with training set. The predictive accuracy of classifiers will be pre-assessed with testing data. Classifiers reaching accuracy threshold, which is set to 0.9 by default, will be kept for further accuracy assessment in afterward training iterations. (3) After all training iterations, all pseudo-cGRNs will be concated as one testing set to assess accuracies of classifiers being kept. Classifiers can still reach accuracy threshold, which is also set to 0.9 by default, will be passed to further analysis.

Model Genera	Model Example	SHAP <sup>23</sup> Method
Tree-Based	GBTree	Tree Explainer
NN-Based without Repetitive Layer Sets	CNN	Deep Explainer
NN-Based with Repetitive Layer Sets	Recurrent Neural Network	Gradient Explainer
Kernel Machines	SVC	Kernel Explainer
Linear Models	Logistic Regression	Linear Explainer

**Table 1.** Machine learning model generas with applicable Shapley value approaching methods

Following analysis all classifiers, each feature's general importance value can be obtained through:

$$g(i) = \sum_{j=1}^C T_j(Z)[i]$$

Here  $i$  denoting  $i$  th feature or GRP,  $C$  denoting total number of classifiers being analyzed,  $T_j(Z)[i]$  denoting  $i$  th feature's importance value calculated through analyzing  $j$  th classifier.

#### Step 4: important GF extraction

With feature list ranked according to importance values, *Odyseia* reconstructs a regulon for top ranked GRPs and extracts influential GFs among the regulon. If analoging regulon or GRN with the graph concept in discrete mathematics, one of the most common methods to analysis influence of a vertex, which is equivalent with GF, is assessing correspond degree of the vertex. Although the regulatory source and target within a GRP can be determined if utilizing interaction database, such as *GTRD*<sup>17</sup>, showing regulatory relationship of interactors, numerous computational prediction method, *GRNBoost2*<sup>24</sup> for example, and comprehensive interaction database, *BioGRID*<sup>25</sup> for example, could not provide faithful indication on regulatory directions. Consequently, based on choice of interaction database for *Odyseia*, the regulon reconstructed cannot be guaranteed to be a directed graph but can surely be analyzed as undirected graph. Therefore, in this step, *odysseia* primarily extracts GFs with high degrees in the regulon regardless in-degree or out-degree. To limit total amount of output important GFs, the default setting of *Odyseia* is reconstructing regulon for top 100 GRPs, and GFs with degree higher than 2 in the regulon will be considered as influential. If regulatory relationships can be clarified, the regulatory sources being capable to regulate multiple important GFs and the regulatory targets being influenced by multiple important GFs can also be extracted according to GRN reconstruction guidance.

In the circumstance of finding GFs can potentially induce CS conversion, referring *Mogrify*<sup>11</sup>, GFs being either CS



deterministic or capable of regulating important GFs shall gain attention. By default setting, if a GF is not listed as important GF, it must be capable to regulate more than 2 listed GFs to be considered as an important regulatory interactor.

## Results

In order to evaluate analytics power, we tested *Odysseia* with 3 previously published CS conversion related scRNA-seq datasets. Considering the differences in information availability and data size among datasets, each dataset was analyzed with adjusted process:

- **MEF and ESC:** This dataset has been used to study cell fate continuum when reprogramming MEF into iPSC.<sup>26</sup> To simulate the process of finding GF combination for reprogramming somatic cell into pluripotent stem cell, we extracted scGEPs labeled MEFs and embryonic stem cells(ESCs) from the dataset and Complete output GF list is included as **Supplementary Table 1**.
- **GSE169054: MEF to Myoblast:** Data from here<sup>27</sup> Complete output GF list is included as **Supplementary Table 2**.
- **GSE185275: iPSC to Neuron:** Data from here<sup>28</sup> Complete output GF list is included as **Supplementary Table 3**.

Even though *Odysseia* is unable to directly guarantee a list of GFs to induce CS conversion, numerous previously discovered key GFs can be successfully extracted and indicate complete combination of GFs known capable to convert CSs. For example...<sup>29</sup>

## Discussion

Considering the variability of available types of data in different studies, *Odysseia's* modules not only can be adjusted upon available resources but also, in order to improve biological interpretability on outputs, provide processed generalized background datasets, such as *GTRD*<sup>17</sup> for finding potential target genes of selected transcription factors when building *GRN* reconstruction guidance. Admittedly, generalized background datasets can still fail to provide substantial support in boundary cases like studying specie is not covered in the dataset. To achieve optimized biological interpretability, substituting generalized dataset with case specific data would be essential.

## Conclusion

Conclusion goes here

## References

1. Takahashi, K. & Yamanaka, S. Induction of pluripotent stem cells from mouse embryonic and adult fibroblast cultures by defined factors. *Cell* **126**, 663–676, DOI: [10.1016/j.cell.2006.07.024](https://doi.org/10.1016/j.cell.2006.07.024) (2006).
2. Buganim, Y. *et al.* Direct reprogramming of fibroblasts into embryonic sertoli-like cells by defined factors. *Cell Stem Cell* **11**, 373–386, DOI: [10.1016/j.stem.2012.07.019](https://doi.org/10.1016/j.stem.2012.07.019) (2012).
3. Qian, L. *et al.* In vivo reprogramming of murine cardiac fibroblasts into induced cardiomyocytes. *Nature* **485**, 593–598, DOI: [10.1038/nature11044](https://doi.org/10.1038/nature11044) (2012).
4. Sekiya, S. & Suzuki, A. Direct conversion of mouse fibroblasts to hepatocyte-like cells by defined factors. *Nature* DOI: [10.1038/nature10263](https://doi.org/10.1038/nature10263) (2011).
5. Earley, A. M., Burbulla, L. F., Krainc, D. & Awatramani, R. Identification of ascl1 as a determinant for human ipsc-derived dopaminergic neurons. *Sci. Reports* **11**, DOI: [10.1038/s41598-021-01366-4](https://doi.org/10.1038/s41598-021-01366-4) (2021).
6. Wang, H., Yang, Y., Liu, J. & Qian, L. Direct cell reprogramming: Approaches, mechanisms and progress. *Nat. Rev. Mol. Cell Biol.* **22**, 410–424, DOI: [10.1038/s41580-021-00335-z](https://doi.org/10.1038/s41580-021-00335-z) (2021).
7. Baron, M. *et al.* A single-cell transcriptomic map of the human and mouse pancreas reveals inter- and intra-cell population structure. *Cell Syst.* **3**, DOI: [10.1016/j.cels.2016.08.011](https://doi.org/10.1016/j.cels.2016.08.011) (2016).
8. Treutlein, B. *et al.* Reconstructing lineage hierarchies of the distal lung epithelium using single-cell rna-seq. *Nature* **509**, 371–375, DOI: [10.1038/nature13173](https://doi.org/10.1038/nature13173) (2014).
9. Cahan, P. *et al.* Cellnet: Network biology applied to stem cell engineering. *Cell* **158**, 903–915, DOI: [10.1016/j.cell.2014.07.020](https://doi.org/10.1016/j.cell.2014.07.020) (2014).

10. D'Alessio, A. C. *et al.* A systematic approach to identify candidate transcription factors that control cell identity. *Stem Cell Reports* **5**, 763–775, DOI: [10.1016/j.stemcr.2015.09.016](https://doi.org/10.1016/j.stemcr.2015.09.016) (2015).
11. Rackham, O. J. *et al.* A predictive computational framework for direct reprogramming between human cell types. *Nat. Genet.* **48**, 331–335, DOI: [10.1038/ng.3487](https://doi.org/10.1038/ng.3487) (2016).
12. Tan, Y. & Cahan, P. Singlecellnet: A computational tool to classify single cell rna-seq data across platforms and across species. *Cell Syst.* **9**, DOI: [10.1016/j.cels.2019.06.004](https://doi.org/10.1016/j.cels.2019.06.004) (2019).
13. Liu, L.-y. D., Hsiao, Y.-C., Chen, H.-C., Yang, Y.-W. & Chang, M.-C. Construction of gene causal regulatory networks using microarray data with the coefficient of intrinsic dependence. *Bot. Stud.* **60**, DOI: [10.1186/s40529-019-0268-8](https://doi.org/10.1186/s40529-019-0268-8) (2019).
14. Song, L., Langfelder, P. & Horvath, S. Comparison of co-expression measures: Mutual information, correlation, and model based indices. *BMC Bioinforma.* **13**, DOI: [10.1186/1471-2105-13-328](https://doi.org/10.1186/1471-2105-13-328) (2012).
15. Virtanen, P. *et al.* SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nat. Methods* **17**, 261–272, DOI: [10.1038/s41592-019-0686-2](https://doi.org/10.1038/s41592-019-0686-2) (2020).
16. Matys, V. Transfac(r) and its module transcompel(r): Transcriptional gene regulation in eukaryotes. *Nucleic Acids Res.* **34**, DOI: [10.1093/nar/gkj143](https://doi.org/10.1093/nar/gkj143) (2006).
17. Kolmykov, S. *et al.* GTRD: an integrated view of transcription regulation. *Nucleic Acids Res.* **49**, D104–D111, DOI: [10.1093/nar/gkaa1057](https://doi.org/10.1093/nar/gkaa1057) (2020). <https://academic.oup.com/nar/article-pdf/49/D1/D104/35364856/gkaa1057.pdf>.
18. Pedregosa, F. *et al.* Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
19. Chen, T. & Guestrin, C. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 785–794 (2016).
20. Paszke, A. *et al.* Pytorch: An imperative style, high-performance deep learning library. In Wallach, H. *et al.* (eds.) *Advances in Neural Information Processing Systems 32*, 8024–8035 (Curran Associates, Inc., 2019).
21. Mostavi, M., Chiu, Y.-C., Huang, Y. & Chen, Y. Convolutional neural network models for cancer type prediction based on gene expression - bmc medical genomics (2020).
22. Roth, A. E. & Shapley, L. S. The shapley value : essays in honor of lloyd s. shapley. *Economica* **101**, 123 (1991).
23. Lundberg, S. M. & Lee, S.-I. A unified approach to interpreting model predictions. In *Proceedings of the 31st international conference on neural information processing systems*, 4768–4777 (2017).
24. Moerman, T. *et al.* Grnboost2 and arboreto: Efficient and scalable inference of gene regulatory networks. *Bioinforma. (Oxford, England)* **35**, DOI: [10.1093/bioinformatics/bty916](https://doi.org/10.1093/bioinformatics/bty916) (2018).
25. Stark, C. Biogrid: A general repository for interaction datasets. *Nucleic Acids Res.* **34**, DOI: [10.1093/nar/gkj109](https://doi.org/10.1093/nar/gkj109) (2006).
26. Guo, L. *et al.* Resolving cell fate decisions during somatic cell reprogramming by single-cell rna-seq. *Mol. Cell* **73**, DOI: [10.1016/j.molcel.2019.01.042](https://doi.org/10.1016/j.molcel.2019.01.042) (2019).
27. Kim, I. *et al.* Integrative molecular roadmap for direct conversion of fibroblasts into myocytes and myogenic progenitor cells. *bioRxiv* DOI: [10.1101/2021.08.20.457151](https://doi.org/10.1101/2021.08.20.457151) (2021).
28. Earley, A. M., Burbulla, L. F., Krainc, D. & Awatramani, R. Identification of ascl1 as a determinant for human ipsc-derived dopaminergic neurons. *Sci. Reports* **11**, DOI: [10.1038/s41598-021-01366-4](https://doi.org/10.1038/s41598-021-01366-4) (2021).
29. Wang, B. *et al.* Induction of pluripotent stem cells from mouse embryonic fibroblasts by jdp2-jhdm1b-mkk6-glis1-nanog-essrb-sall4. *Cell Reports* **27**, DOI: [10.1016/j.celrep.2019.05.068](https://doi.org/10.1016/j.celrep.2019.05.068) (2019).

## Author contributions statement

J.Y.: Methodology, Software, Writing- Original draft preparation J.T.: Writing - Review & Editing

## Additional information

Result, Discussion, Conclusion are not done yet

## Key Resource Table