OXFORD

Briefings in Bioinformatics, 2022, **23**(**4**), 1–12
https://doi.org/10.1093/bib/bbac247
Advance access publication date: 27 June 2022
Problem Solving Protocol

Online *in silico* validation of disease and gene sets, clusterings or subnetworks with DIGEST

Klaudia Adamowicz, Andreas Maier, Jan Baumbach and David B. Blumenthal
Corresponding author: David B. Blumenthal, Department Artificial Intelligence in Biomedical Engineering, Friedrich-Alexander-Universität Erlangen-Nürnberg, Werner-von-Siemens-Str. 61, 91052 Erlangen, Germany. Tel.: +49 9131 8570676; E-mail: david.b.blumenthal@fau.de

Abstract

As the development of new drugs reaches its physical and financial limits, drug repurposing has become more important than ever. For mechanistically grounded drug repurposing, it is crucial to uncover the disease mechanisms and to detect clusters of mechanistically related diseases. Various methods for computing candidate disease mechanisms and disease clusters exist. However, in the absence of ground truth, *in silico* validation is challenging. This constitutes a major hurdle toward the adoption of *in silico* prediction tools by experimentalists who are often hesitant to carry out wet-lab validations for predicted candidate mechanisms without clearly quantified initial plausibility. To address this problem, we present DIGEST (*in silico* validation of **di**sease and **ge**ne **s**ets, clus**t**erings or subnetworks), a Python-based validation tool available as a web interface (https://digest-validation.net), as a stand-alone package or over a REST API. DIGEST greatly facilitates *in silico* validation of gene and disease sets, clusterings or subnetworks via fully automated pipelines comprising disease and gene ID mapping, enrichment analysis, comparisons of shared genes and variants and background distribution estimation. Moreover, functionality is provided to automatically update the external databases used by the pipelines. DIGEST hence allows the user to assess the statistical significance of candidate mechanisms with regard to functional and genetic coherence and enables the computation of empirical *P*-values with just a few mouse clicks.

Keywords: Systems medicine, in silico validation, Functional and genetic coherence

Introduction

The main objective of systems medicine is to uncover molecular mechanisms driving complex diseases and thereby pave the way for causally effective treatment. A plethora of computational approaches designed to support this overall aim exist, ranging from classical differential gene expression analysis tools [18] over comorbidity pattern analysis frameworks [11] to network-based disease mechanism mining methods [4, 10, 20, 21]. Independently of the concrete algorithmic model, computational systems medicine approaches often return sets, clusterings or subnetworks of genes or diseases as output. For instance, the typical output of differential gene expression analyses are sets of differentially expressed genes (DEGs) for case versus control studies or clusterings of DEGs for different disease subtypes. Comorbidity analyses, on the other hand, often return sets or clusterings of diseases that are predicted to share common molecular mechanisms. Disease mechanism mining approaches typically return small induced subnetworks in protein–protein interaction (PPI) or gene regulatory networks.

In order to have translational potential, pre-clinical studies are necessary to test the hypotheses generated by computational systems medicine approaches in cell lines or animal models. However, such studies are typically resource-intensive, which implies that the hypotheses derived via computational means must have solid prior plausibility in order to convince pre-clinical researchers to invest time and money in follow-up hypothesis testing. Consequently, tools are needed which allow to *in silico* quantify the hypotheses' initial plausibility in a user-friendly way.

A widely used approach for *in silico* validation of hypotheses generated by computational systems medicine approaches is to employ functional annotations available in data sources such as the Kyoto Encyclopedia of Genes and Genomes (KEGG) [14] or the Gene Ontology (GO) [2, 9]. Various tools exist which allow enrichment analyses against these data sources, including DAVID [13], PANTHER [26], g:Profiler [32] and WebGestalt [22]. However, to the best of our knowledge, all existing tools have two limitations: Firstly, only gene sets are accepted as input and neither disease sets nor gene or disease

Klaudia Adamowicz is doctoral researcher at the Institute for Computational Systems Biology at the University of Hamburg.
Andreas Maier is doctoral researcher at the Institute for Computational Systems Biology at the University of Hamburg.
Jan Baumbach is professor and head of the Institute for Computational Systems Biology at the University of Hamburg. He obtained his PhD in Computer Science from Bielefeld University.
David B. Blumenthal is professor and head of the Biomedical Network Science Lab at the Department Artificial Intelligence in Biomedical Engineering of the Friedrich-Alexander-Universität Erlangen-Nürnberg. He obtained his PhD in Computer Science from the Free University of Bozen-Bolzano.
Received: February 28, 2022. **Revised:** May 25, 2022. **Accepted:** May 26, 2022

Table 1. Features of DIGEST and existing online validation tools. Check marks in parentheses indicate that DIGEST provides the respective features via calls to the API of g:Profiler.

	DAVID	PANTHER	g:Profiler	WebGestalt	DIGEST
Tabular results					
Empirical *P*-values	X	X	X	X	✓
Enriched annotations	✓	✓	✓	✓	(✓)
Most frequent annotations	X	X	X	X	✓
Summary figures					
Empirical *P*-values	X	X	X	X	✓
Enriched annotations	X	✓	X	✓	X
Most frequent annotations	X	✓	X	✓	✓
Annotation distribution	X	X	X	X	✓
Annotation graph	X	X	X	✓	X
Publication-ready figures	X	X	X	✓	✓
Supported input types					
Gene set	✓	✓	✓	✓	✓
Gene clustering	X	X	X	X	✓
Gene subnetwork	X	X	X	X	✓
Disease set	X	X	X	X	✓
Disease clustering	X	X	X	X	✓
Disease subnetwork	X	X	X	X	✓
Supported data sources					
Gene ontology	✓	✓	✓	✓	✓
Biological pathways	✓	✓	✓	✓	✓
Regulatory motifs in DNA	✓	X	✓	✓	X
Protein databases	✓	✓	✓	✓	X
Phenotype ontology	✓	X	✓	✓	X
Gene-disease associations	✓	X	X	✓	✓
Variant-disease associations	X	X	X	X	✓
Automated mapping					
Automated gene ID mapping	✓	✓	✓	✓	✓
Gene ID conversion	✓	X	✓	✓	X
Automated disease ID mapping	X	X	X	X	✓
Support beyond *homo sapiens*	✓	✓	✓	✓	X
Orthology-based gene mapping	X	X	✓	X	X
Supported analyses					
Gene set enrichment analysis	✓	✓	✓	✓	(✓)
Reference-free functional coherence	✓	✓	✓	✓	✓
Reference-based functional coherence	X	X	X	✓	✓
Reference-free genetic coherence	X	X	X	✓	✓
Reference-based genetic coherence	X	X	X	X	✓

clusterings or subnetworks are supported. Secondly, functionality to assess the statistical significance of the obtained enrichment scores in comparison to random background models is missing.

In order to fill these gaps, we present DIGEST (short for '*in silico* validation of **di**sease and **ge**ne **s**ets or clus**t**erings') — an *in silico* validation tool for computational systems medicine approaches that allows the validation of different input types in a fully automated and user-friendly way. DIGEST is available as a web service (https://digest-validation.net), as a Python package (https://pypi.org/project/biodigest) and as a REST API (https://api.digest-validation.net). Table 1 provides an overview of its main features in comparison to the existing tools DAVID, PANTHER, g:Profiler and WebGestalt. Note that DIGEST is intended to complement rather than replace these tools. For instance, DIGEST focuses on human diseases and so does not offer support beyond *homo sapiens*. Moreover, DIGEST is not primarily designed for classical gene set enrichment analysis [39], although this feature is indirectly supported via calls to the API of g:Profiler.

Results

Overview of DIGEST and supported input types

The main result of this paper is the DIGEST validation tool itself. Figure 1 provides an overview of its functionality and the supported validation routes. Similar to existing approaches, DIGEST can perform functional analysis on gene sets using GO and KEGG. In addition to the gene set enrichment analysis, gene sets can also be validated for functional coherence via pairwise comparisons of the contained genes' functional annotations. Like WebGestalt, DIGEST also offers the possibility to provide a reference gene set. Alternatively, a reference disease ID can be provided by the user. In addition to analyzing gene sets, DIGEST allows validation of gene clusterings based on clustering quality measures such as the Dunn index [8], the sillhouette

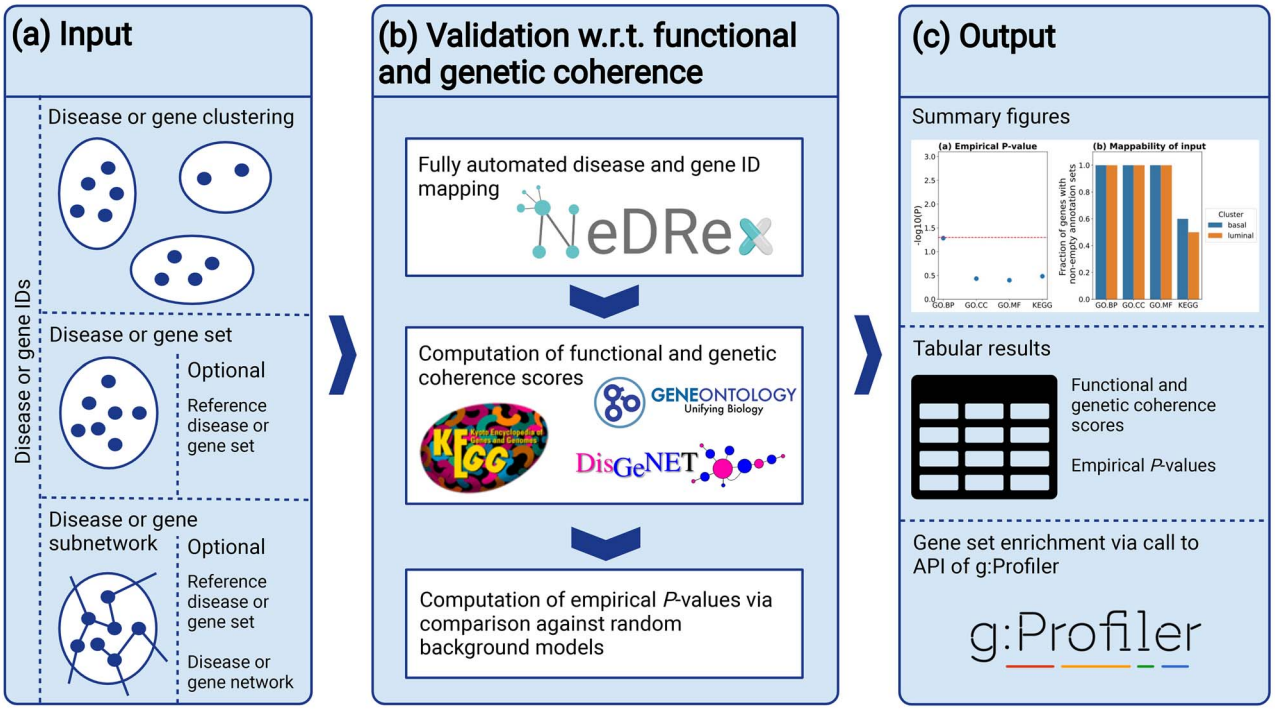

Fig. 1. Overview of validation routes provided by DIGEST. **(A)** As input, DIGEST supports sets, clusterings (i.e. sets of sets) and subnetworks of genes or diseases. For disease and gene sets or subnetworks, a second set or a disease ID can optionally be provided as reference. **(B)** DIGEST queries NeDRex [36] to automatically map the gene or disease IDs provided as input to the required namespaces. Depending on the provided input, functional or genetic coherence scores are computed using GO [2, 9] or KEGG [14] enrichment or genetic associations obtained from DisGeNET [30]. Subsequently, empirical *P*-values are computed by comparing the obtained scores against two random background models. **(C)** The results of the validations are provided in the form of summary figures, as well as in tabular format. Moreover, gene set enrichment analysis is supported via calls to the API of g:Profiler.

score [34] and the Davies–Bouldin index [6]. As in the gene set analysis, functional annotations from GO and KEGG are used for computing the gene distances underlying these measures. Finally, DIGEST can be used to *in silico* validate induced subnetworks in gene–gene networks.

Unlike all existing approaches we are aware of, DIGEST also supports disease IDs as input. As for genes, sets and clusterings of disease IDs are supported, as well as induced subnetworks in disease–disease networks. The input can be validated w. r. t. functional coherence based on annotations obtained from KEGG. Moreover, DIGEST supports a genetic coherence analysis for diseases based on shared genes or variants extracted from DisGeNET [30].

A further novelty of DIGEST w. r. t. existing approaches is that it allows to evaluate whether the obtained scores of functional or genetic coherence are statistically significant. This is important because scores such as the Dunn index or the silhouette score are often difficult to interpret. In order to be able to judge whether an obtained score is 'good' or 'bad', a comparison against a random background is required. Therefore, DIGEST provides random background models for all input types and uses them to compute empirical *P*-values quantifying the significance of the obtained scores.

In the sequel, we showcase how to use DIGEST for validation of gene and disease sets, clusterings and subnetworks. Technical and algorithmic details can be found in the 'Methods' section.

In silico validation of gene sets

Since prior studies have linked lipid and cholesterol metabolism to Alzheimer's disease (AD) [7], Sadegh *et al.* [36] hypothesized that hyperlipidemia-associated genes obtained from OMIM [1] and DisGeNET [30] constitute a promising point of departure for network-based AD drug repurposing. We ran DIGEST's reference-based gene set validation route to assess this hypothesis, using the AD-associated genes as query and the hyperlipidemia-associated genes as reference set (the genes contained in the sets are listed in Supplementary Table 1). To compute empirical *P*-values, we used a random background model that preserves the sizes of the contained genes' annotation sets.

The results are shown in Figure 2. While significant *P*-values were obtained for all three types of GO terms (BP: biological process; CC: cellular component; MF: molecular function), results were not significant for KEGG pathways. This can be explained by the fact that the Jaccard index between KEGG pathways containing, respectively, AD- and hyperlipidemia-associate genes is 0.0 (see the red vertical line in Figure 2b). Nonetheless, the results lend evidence to the conjecture that AD is mechanistically linked to hyperlipidemia and that

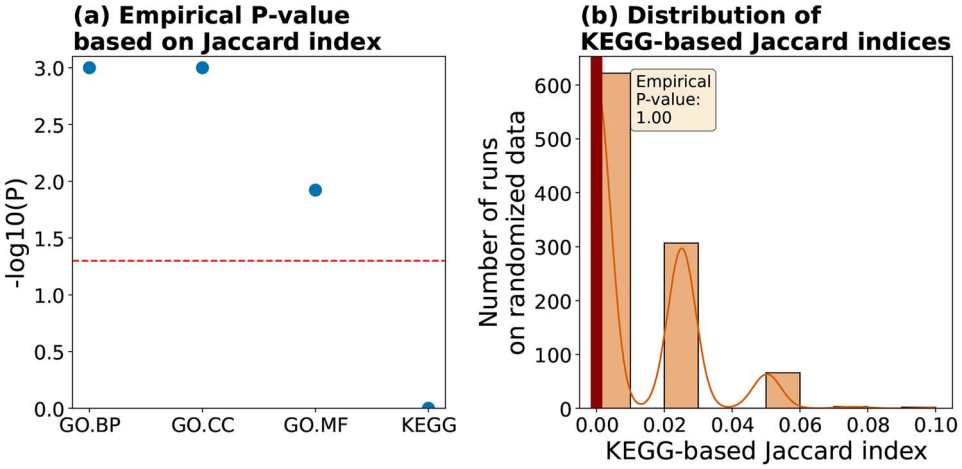

Fig. 2. Results of reference-based gene set validation route for hyperlipidemia-based AD drug repurposing approach suggested by Sadegh *et al.* [36]. **(A)** Empirical *P*-values obtained by comparing the mean Jaccard index of pairwise annotation sets against random background. **(B)** KEGG-based Jaccard indices for input gene set (red vertical line) and gene sets generated by background model.

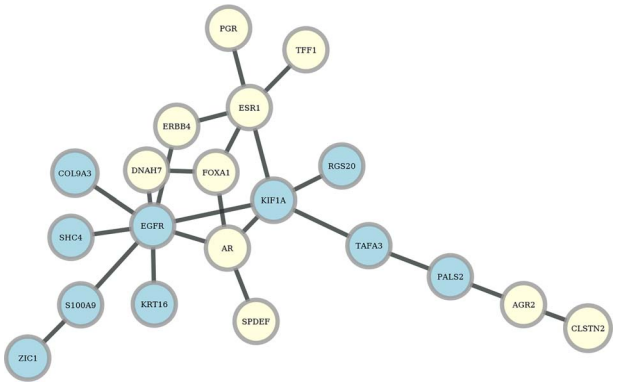

Fig. 3. Candidate pathomechanisms for luminal (blue) and basal (yellow) breast cancer reported in [20]. Genes are connected by an edge if the BioGRID [29] database contains a PPI for their encoded proteins.

Table 2. List of diseases linked to cGMP signaling by Langhauser *et al.* [17] and their respective MeSH IDs.

Disease	MeSH ID
Stroke	D020521
Alzheimer disease	D000544
Dementia	D003704
Atherosclerosis	D050197
Asthma	D001249
Diabetes mellitus type 2	D003924
Parkinson disease	D010300
Heart failure	D006333
Migraine disorders	D008881
Myocardial infarction	D009203
Hypertension	D006973
Obesity	D009765

hyperlidipidemia-associated genes are a promising starting point for *in silico* AD drug repurposing.

In silico validation of gene clusterings

BiCoN [20] is a network medicine tool which allows binary patient stratification based on gene expression data. At the same time, it extracts two subgraphs from PPI networks that explain the patient stratification and hence constitute candidate pathomechanisms. In the original publication, the authors used BiCoN on gene expression data for breast cancer and obtained a patient clustering that almost perfectly matches the luminal versus basal breast cancer sub-types. Figure 3 shows the two candidate pathomechanisms. Here, we interpreted these predicted pathomechanisms as a gene clustering with two blocks, and used DIGEST's gene clustering validation route to assess its functional coherence.

Figure 4 shows the results of the analysis. We did not obtain significant *P*-values for any of the four gene annotation options supported by DIGEST. The results hence provide little evidence for the hypothesis that the two sub-networks extracted by BiCoN indeed correspond to distinct pathomechanisms.

In silico validation of disease sets

Using a network-based systems medicine approach, Langhauser *et al.* [17] identified a set of 12 diseases hypothesized to share a common mechanism linked to cyclic guanosine monophosphate (cGMP) signaling. The set of these presumably mechanistically related diseases, specified in the MeSH namespace as in the original publication, is shown in Table 2.

We used DIGEST's reference-free disease set validation route with random background models that preserve annotation set size to assess the genetic and functional coherence of this set of diseases. The results are shown in Figure 5. For all three annotation types, the disease set exhibits a significant internal coherence. Using DIGEST, we could hence provide further evidence for Langhauser *et al.*'s hypothesis that the diseases listed in Table 2 are mechanistically related.

In silico validation of disease clusterings

To exemplify the use of DIGESTS's disease clustering validation route, we analyzed the clustering of the ICD-10 three-character codes induced by chapter IX 'Diseases of the circulatory system' (see Figure 6). This analysis helps

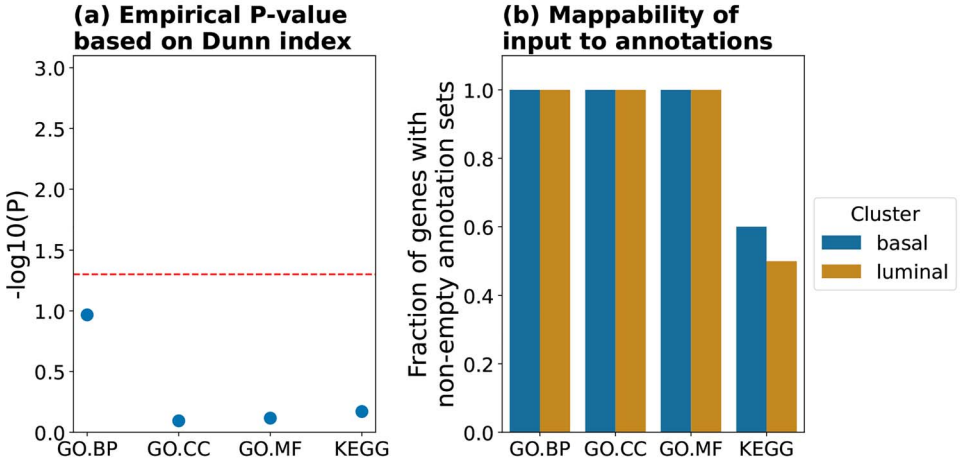

Fig. 4. Results of gene clustering validation route for luminal and basal breast cancer pathomechanisms predicted by Lazareva *et al.* [20]. **(A)** Empirical *P*-values obtained by comparing the clustering's Dunn index against random background. **(B)** Frequency of genes with non-empty sets of assigned functional annotations.

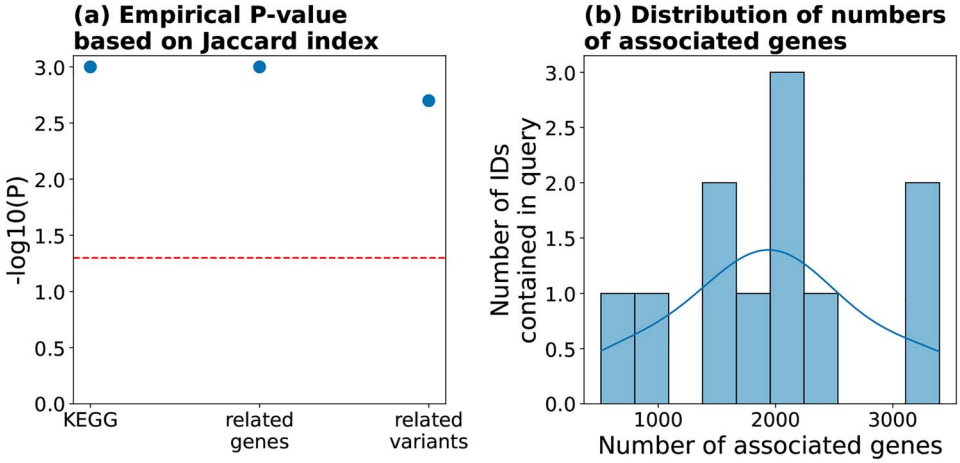

Fig. 5. Results of reference-free disease set validation route for diseases linked to cGMP signaling by Langhauser *et al.* [17]. **(A)** Empirical *P*-values obtained by comparing the mean Jaccard index of pairwise annotation sets against random background. **(B)** Distribution of number of related genes associated with each ID.

IX Diseases of the circulatory system
- **I00–I02** Acute rheumatic fever
- **I05–I09** Chronic rheumatic heart diseases
- **I10–I15** Hypertensive diseases
- **I20–I25** Ischaemic heart diseases
- **I26–I28** Pulmonary heart disease and diseases of pulmonary circulation
- **I30–I52** Other forms of heart disease
- **I60–I69** Cerebrovascular diseases
- **I70–I79** Diseases of arteries, arterioles and capillaries
- **I80–I89** Diseases of veins, lymphatic vessels and lymph nodes, not elsewhere classified
- **I95–I99** Other and unspecified disorders of the circulatory system

Fig. 6. Blocks of chapter IX of the ICD-10 disease ontology obtained from https://icd.who.int/browse10/2019/en/.

to assess to which extent chapter IX of the ICD-10 disease ontology is mechanistically grounded, i.e. whether the contained sub-blocks exhibit a stronger functional and genetic intra- than inter-block coherence.

Figure 7 shows the results of our analysis. Interestingly, we did not observed statistically significant results for any of the three annotation types. Moreover, all of the top 10 most frequently occuring associated genes are present in most of the clusters, indicating that clusters' genetic coherence is low. These findings confirm the increasing awareness in the field that currently used disease ontologies are problematic because they do not mirror the underlying pathomechanisms [28].

In silico validation of gene subnetworks

KeyPathwayMineR [25] is a *de novo* network enrichment tool to predict candidate pathomechanisms starting with a list of DEGs. For the original publication, the authors ran their tool on gene expression data from COVID-19 patients and healthy controls and on a human PPI network obtained from BioGRID to derive a candidate host mechanism involved in COVID-19.

We used DIGEST's reference-free gene subnetwork validation route to assess the internal functional coherence of this candidate mechanism. As shown in Figure 8, we

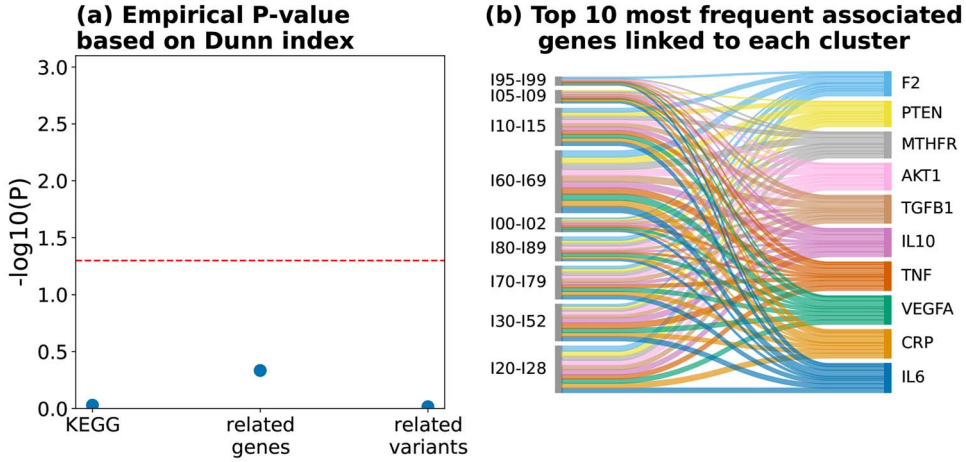

Fig. 7. Results of disease clustering validation route for clustering induced by the blocks of ICD-10 chapter IX. **(A)** Empirical *P*-values obtained by comparing the clustering's Dunn index against random background. **(B)** Sankey plot linking each cluster to the top 10 most frequently occurring associated genes.

did not obtain any significant *P*-values. In fact, even for the GO.BP annotations which led to the smallest *P*-value, the mean Jaccard index obtained for the predicted host mechanism is < 0.05 (Figure 8b). This renders unlikely that the subnetwork identified by Mechteridis *et al.* [25] indeed represents *one* host pathway that is hijacked in SARS-CoV-2 infection (it might still represent several, mechanistically disjoint pathways).

To elucidate possible reasons that might explain the non-significant results, we computed gene-level significance contributions (see 'Methods' for details and Figure 8c for the results obtained for the GO.MF-based *P*-values). That is, we estimated to which extent the individual genes contained in the COVID-19 subnetwork discovered by KeyPathwayMineR positively or negatively affect functional coherence. We then carried out gene set enrichment analysis with g:Profiler on, respectively, the 10 genes with largest positive and largest negative effects on coherence w. r. t. GO.MF annotations (supported in DIGEST via direct links to g:Profiler from the results view).

Interestingly, the 10 genes with largest positive effects are enriched with annotations related to virus response (see Supplementary Figure 1), while the 10 genes with largest negative effects are enriched with more generic annotations (see Supplementary Figure 2). A possible explanation for the non-significant results is hence that KeyPathwayMineR returns a large number of genes with rather generic functions—possibly, because such genes tend to constitute hubs in PPI networks. The fact that the genes with large positive significance contributions assume rather peripheral positions in the subnetwork computed by KeyPathwayMineR provides further evidence for this conjecture (see red nodes in Figure 8c).

Scalability tests

To evaluate the scalability of DIGEST, we compared the API query execution times for the reference-free gene set validation route against functional gene set enrichment queries against the APIs of PANTHER and g:Profiler (results shown in Table 3; for details on the test setup, see 'Methods'). Since DIGEST is the only tool that runs permutation tests, it is not surprising that it is clearly the slowest of the three tools, with a mean query time of around 13 min on gene sets of size 100 if run with $n = 1000$ randomizations. To further improve usability of the web interface, we provide a stable URL to the results immediately after starting the *in silico* validation. By saving this URL, the user does not have to wait for the analysis to finish but can return to the results page at any later point. Note that if run with $n = 1$ randomizations, DIGEST's query is around twice the query time of PANTHER. Setting the number of randomizations to $n = 1$ of course does not make sense for validation purposes, but yields a fairer comparison against PANTHER and g:Profiler, which do not run permutation tests.

Discussion

With DIGEST, we introduce a user-friendly tool for the *in silico* validation of sets, clusterings or subnetworks of genes and diseases. DIGEST supports the fully automated computation of empirical *P*-values for hypotheses generated by computational systems medicine approaches and hence helps to increase the likelihood that promising hypotheses are further carried toward translation in pre-clinical studies.

It is important to emphasize that the *P*-values computed by DIGEST only quantify initial plausibility of hypotheses generated by computational systems medicine approaches. Whether these hypotheses are ultimately valid and indeed correspond to disease mechanisms can only be established via follow-up pre-clinical and clinical studies. DIGEST is hence not intended as a tool to substitute downstream wet-lab validation, but as a tool that allows to further narrow

Table 3. Mean execution times of API queries against PANTHER, g:Profiler and DIGEST for gene set queries with varying numbers of genes (in seconds). DIGEST was run with $n = 1000$ and $n = 1$ randomizations and without estimation of significance contributions.

# Genes	PANTHER	g:Profiler	DIGEST ($n = 1000$)	DIGEST ($n = 1$)
10	79.51	33.89	701.06	149.77
50	80.91	33.67	762.03	148.41
100	88.68	33.98	806.83	148.13

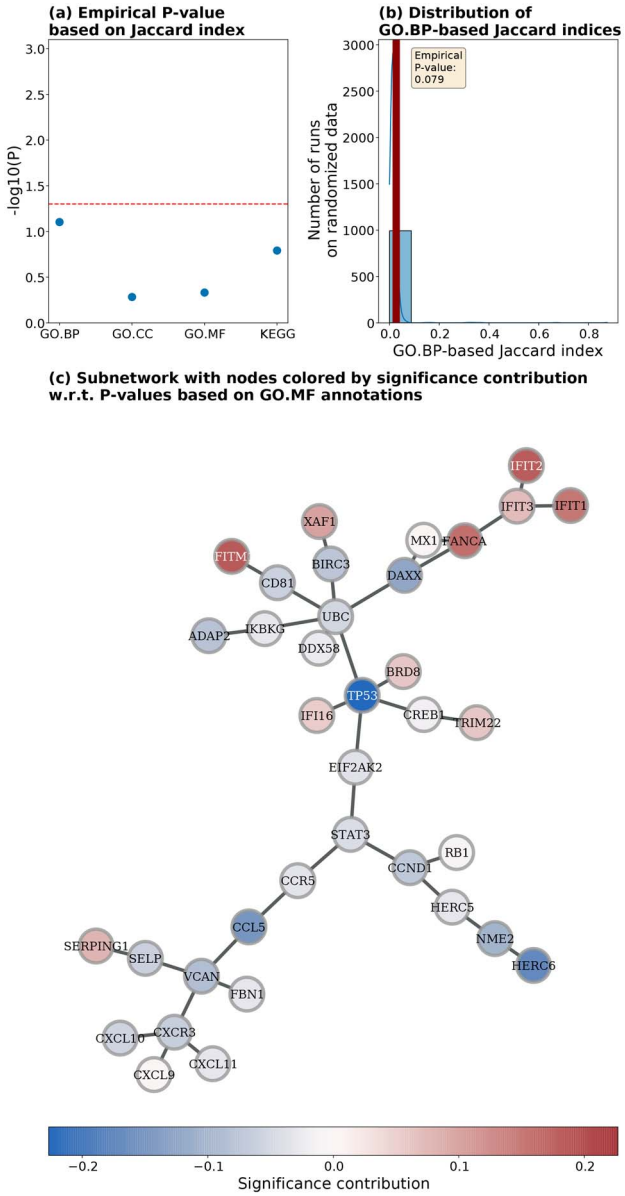

Fig. 8. Results of reference-free gene subnetwork validation route for candidate COVID-19 host mechanism predicted by Mechteridis *et al.* [25]. **(A)** Empirical *P*-values obtained by comparing the mean Jaccard index of pairwise annotation sets against random background. **(B)** Mean GO.BP-based Jaccard indices for input subnetwork (red vertical line) and gene subnetworks generated by random background model. **(C)** Significance contributions toward empirical *P*-values based on GO.MF annotations of individual genes contained in the predicted COVID-19 host mechanism.

down the search space and focus on the most promising hypotheses.

Moreover, the *P*-values returned by DIGEST have to be interpreted carefully, since they are based on the biological knowledge encoded in the KEGG, GO and DisGeNET databases. The scrutinized hypotheses are hence always validated against current knowledge, although they typically aim at generating new insights into pathomechanisms. Moreover, biological databases are known to be heavily affected by study bias [19, 37], i.e. few well-studied genes and diseases are overrepresented w. r. t. the number of associations. In DIGEST, we address this shortcoming using random background models that preserve annotation set sizes. However, this only partially solves the problem, because we cannot distinguish between genes that are 'true biological hubs' and genes that appear to be hubs due to study bias.

Finally, we would like to point out that it is possible that DIGEST's empirical *P*-values are inconsistent (e.g. see results for the drug repurposing use case shown in Figure 2). In such a situation, it is important that the user carefully assesses which of the disease or gene annotation databases used by DIGEST are indeed relevant for their use case. For instance, in the above-mentioned drug repurposing use case, one might decide to disregard the non-significant KEGG-based *P*-value, since it has been argued that KEGG pathways poorly reflect pathomechanisms [28].

Methods

Supported disease and gene namespaces

DIGEST validates sets, clusterings and subnetworks of genes and diseases by comparing suitably defined annotations sets obtained via queries against KEGG, GO and DisGeNET (see subsequent sections for details). Since KEGG, GO and DisGeNET require genes and diseases to be given using specific namespaces (see Supplementary Table 2), DIGEST queries NeDRex [36] to automatically map the input provided by the user to the required namespaces. The following input formats are supported:

- Supported gene namespaces: Entrez [23], Ensembl [12], gene symbols [40], UniProt IDs of encoded proteins [3].
- Supported disease namespaces: Mondo [27], OMIM [1], SNOMED CT [35], UMLS [5], ORPHA [31], MeSH [33], DOID [38], ICD-10 (https://icd.who.int/browse10/2010/en).

For all input types and validation routes, identifiers that cannot be mapped to the required namespace are ignored. Moreover, DIGEST ignores diseases and genes for

which the annotation sets obtained from KEGG, GO or DisGeNET are empty.

Statistical validation

Let X be any input of the supported input types specified above. Moreover, let $S(X)$ be a functional relevance score for X and $\mathcal{M}$ be a background model which allows to generate randomized input $\mathcal{M}(X)$, preserving some of the properties of X. Assume that, w. r. t. S, larger means better. We compute empirical P-values as

$$P := \frac{1 + \sum_{i=1}^{n} [S(X) \leq S(\mathcal{M}(X)_i)]}{n+1}, \tag{1}$$

where n is a user-specified parameter determining the P-value resolution (defaulted to $n := 1000$ in DIGEST) and $[\cdot]$ is the Iverson bracket (i.e. $[\texttt{true}] := 1$ and $[\texttt{false}] := 0$). In the following text, we detail how the scores S and the background models $\mathcal{M}$ are defined for the supported input types. Note that separate P-values are computed for each annotation type. For instance, for reference-free gene set validation (see details below), DIGEST computes empirical P-values based on KEGG pathways, GO cellular components, GO molecular functions and GO biological processes.

Scores and random background for gene sets

DIGEST supports the following three routes for *in silico* validating a set of genes X:

(1) Validation against a reference disease set X_R: Assesses whether the functional relevance of X w. r. t. diseases contained in X_R is statistically significant.
(2) Validation against a reference gene set X_R: Assesses whether the functional similarity between genes contained in X and X_R is statistically significant.
(3) Reference-free validation: Assesses whether the internal functional coherence of the genes contained in X is statistically significant.

For the first two routes, sets $\mathcal{A}_X$ and $\mathcal{A}_R$ of functional annotations are obtained for, respectively, the gene set X and the reference X_R or d_R (see the next paragraph for details). Subsequently, the functional relevance score $S(X)$ is computed either as the Jaccard index $S_{JI}(X) := JI(\mathcal{A}_X, \mathcal{A}_R) := |\mathcal{A}_X \cap \mathcal{A}_R| / |\mathcal{A}_X \cup \mathcal{A}_R|$ or as the overlap coefficient $S_{OC}(X) := OC(\mathcal{A}_X, \mathcal{A}_R) := |\mathcal{A}_X \cap \mathcal{A}_R| / \min\{|\mathcal{A}_X|, |\mathcal{A}_R|\}$, depending on the choice of the user.

For the first validation route, $\mathcal{A}_R$ is defined as union of the set of KEGG pathways associated with diseases $d \in X_R$ and $\mathcal{A}_X$ is defined as the set of KEGG pathways associated with genes $g \in X$ (note that singleton disease sets $X_R = \{d\}$ are allowed). For the second validation route, KEGG and GO are used to obtain the set of functional annotations. In either case, $\mathcal{A}_X$ is defined as the set of annotations associated with genes $g \in X$ or, optionally, as the set of annotations enriched with X as a whole. $\mathcal{A}_R$ is defined either as the set of annotations associated with genes $g \in X_R$ or as the set of annotations enriched with X_R as a whole.

For the third, reference-free validation route, sets $\mathcal{A}_g$ of functional annotations are obtained from GO and KEGG for each gene $g \in X$ independently. Depending on the choice of the user, $S(X)$ is then computed either as the mean Jaccard index or as the mean overlap coefficient over all pairs $(\mathcal{A}_g, \mathcal{A}_{g'})$ of annotation sets for $g, g' \in X$ with $g \neq g'$.

Two random background models are supported: The first model draws genes uniformly without replacement to compute fully randomized gene sets $\mathcal{M}(X)$ of size $|X|$. The second model maintains some information from X and constructs randomized gene sets $\mathcal{M}(X)$ where the distribution of the contained genes' annotation set sizes (approximately) matches the distribution of the annotation set sizes of the genes contained in X.

Scores and random background for gene clusterings

DIGEST supports the following three scoring measures for *in silico* validating gene clusterings $X = C_1, \ldots, C_m$: The Dunn index $S_{DI}(X) := \min_{i \neq j} \delta(C_i, C_j) / \max_i \Delta(C_i)$, the Davies–Bouldin index $S_{DBI}(X) := \sum_{i=1}^{m} [\max_{j \neq i} \Delta(C_i) + \Delta(C_j) / \delta(C_i, C_j)] / m$ and the silhouette score $S_{SS}(X) := \sum_{g \in \bigcup C_i} [(a(g) - b(g)) / \max\{a(g), b(g)\}] / n$. $\Delta(C_i)$ denotes the mean intra-cluster distance (i.e. the mean distance between genes $g, g' \in C_i$ with $g \neq g'$), $\delta(C_i, C_j)$ denotes the mean inter-cluster distance (i.e. the mean distance between genes $g \in C_i$ and $g' \in C_j$), $a(g)$ denotes the mean distance between a fixed gene $g \in C_i$ and all other genes in C_i, and $b(g)$ is defined as $b(g) := \min_{j \neq i} b(g, C_j)$, where $b(g, C_j)$ denotes the mean distance between a fixed gene $g \in C_i$ and the genes contained in C_j.

The gene distances underlying the Dunn index, the Davies–Bouldin index, and the silhouette score are defined as $1 - JI(\mathcal{A}_g, \mathcal{A}_{g'})$ or as $1 - OC(\mathcal{A}_g, \mathcal{A}_{g'})$, depending on whether the user wants to use the Jaccard index or the overlap coefficient as underlying set similarity measure. $\mathcal{A}_g$ and $\mathcal{A}_{g'}$ are functional annotations for the genes g and g' obtained from GO and KEGG. The background model randomly shuffles the assigned clusters between the genes $g \in \bigcup_{i=1}^{m} C_i$, preserving the number of clusters m and the cluster size distribution.

Scores and random background for disease sets

DIGEST supports the following two routes for *in silico* validating a set of diseases X:

(1) Validation against a reference gene set X_R: Assesses whether the functional coherence between the diseases contained in X and the genes contained X_R is statistically significant.
(2) Validation against a reference disease set X_R: Assesses whether the genetic similarity between diseases contained in X and X_R is statistically significant.

(3) Reference-free validation: Assesses whether the internal genetic coherence of the diseases contained in X is statistically significant.

Functional relevance scores for the validation routes are defined in almost exactly the same way as for the validation routes for gene sets explained above. The only difference is that, in addition to KEGG pathways, disease–gene and disease–variant associations also obtained from DisGeNET are used to construct the annotation sets underlying the relevance scores, while no GO annotations are employed.

As for gene sets, two random background models are supported: (1) A fully randomized model which uniformly samples genes sets $\mathcal{M}(X)$ of size $|X|$. (2) A partially randomized model which samples genes sets $\mathcal{M}(X)$ such that the distribution of the sizes of the contained diseases' annotation sets (approximately) matches the distribution of annotation set sizes for X.

Scores and random background for disease clusterings

DIGEST can also be used to *in silico* validate disease clusterings $X = C_1, \ldots, C_m$. Like for gene clusterings, the Dunn index, the Davies–Bouldin index or the silhouette score can be used as clustering quality score $S(X)$. The underlying disease distances are defined in terms of the Jaccard index or the overlap coefficient of disease annotation sets $\mathcal{A}_d$ obtained from DisGeNET (genes or variants) or KEGG. The background model is analogous to the one for gene clusterings, i.e. diseases $d \in \bigcup_{i=1}^{m} C_i$ are randomly shuffled across the m clusters under the constraint that the cluster sizes are preserved.

Scores and random background for induced gene or disease subnetworks

Finally, DIGEST can be used to *in silico* validate subnetworks induced by a gene or disease set provided by the user. More precisely, let $G = (V, E)$ be a gene–gene or disease–disease network and $X \subseteq V$ be the user-provided gene or disease set. Also the network G can be provided by the user. If no network is provided, DIGEST obtains G via queries to NeDRex as follows: If X is a set of genes, DIGEST constructs G as a gene–gene network based on experimentally validated PPIs obtained from IID [15] (i.e. two genes are connected by an edge if their encoded proteins have been shown to physically interact). If X is a set of diseases, G is constructed as a disease–disease network based on shared disease–gene associations obtained from OMIM and DisGeNET (i.e. two diseases are connected by an edge if they have at least one associated gene in common).

To validate the subgraph $G[X]$ induced by X, we compute functional relevance scores $S(X)$ in exactly the same way as for gene and disease sets (see details above). As background model, we compute gene or disease sets $\mathcal{M}(X)$ such that the induced subgraph $G[\mathcal{M}(X)]$ matches $G[X]$ in terms of numbers and sizes of connected components. For this, we first decompose $G[X]$ into its connected components $(X_i)_{i=1}^{k}$. Subsequently, we construct $\mathcal{M}(X)$ via k random network expansions along the edges of G from k randomly selected seed nodes $v_i \in V$. We stop the random expansion from node v_i as soon as the constructed connected component has reached the size $|X_i|$.

Significance contributions of individual genes or diseases

DIGEST also allows to compute the contributions of individual genes or diseases toward (non-)significance (implemented as an optional feature since it increases DIGEST's runtime linearly w. r. t. the size of the input). Let X be any input supported by DIGEST, x be a disease or gene contained in X and $X - x$ be a slightly modified input where x has been removed. We quantify x's significance contribution as

$$\Delta_P(x) := P(X - x) - P(X), \tag{2}$$

where $P(X)$ and $P(X - x)$ are the empirical P-values obtained for the original and modified inputs, respectively. If $\Delta_P(x) > 0$, x has a positive effect on X's genetic or functional coherence. If $\Delta_P(x) < 0$, the opposite is the case.

Note that the gene- or disease-level significance contributions are especially interesting for mixed-message results. If the P-values are extreme ($P(X) \approx 0$ or $P(X) \approx 1$), it may well happen that they are insensitive to removal of individual genes or diseases from the input, which then leads to $\Delta_P(x) = 0$ for many or even all genes or diseases x contained in the input (see Supplementary Figure 3 for such a situation).

Summary figures

DIGEST offers fully automated creation of plots to visualize the results. Let T be all annotation types used for computation of the relevance scores $S(X)$. Moreover, for each annotation type $t \in T$ and each element $x \in X$, let $\mathcal{A}_x^t$ be the set of annotations from t for x and $\Delta_P^t(x)$ be the significance contribution of a gene or disease x on the w. r. t. the P-value based on annotation type t. We provide plots to visualize the calculated empirical P-values and plots to show the mappability of the input. The following plots are generated to visualize the empirical P-value visualization (see Supplementary Figure 3 for an overview):

- A plot showing all calculated empirical P-values for all annotation types $t \in T$ (see Figure 2a, Figure 4a, Figure 5a, Figure 7a and Figure 8a).
- A heatmap showing the significance contributions $\Delta_P^t(x)$ for all annotation types $t \in T$ and the top 15 genes or diseases x contained in the input X with the largest absolute significance contributions across all annotation types ($\mathbf{max}_x \mathbf{max}_t |\Delta_P^t(x)|$, see Supplementary Figure 3b).

- For every annotation type $t \in T$: Heatmaps showing the significance contributions $\Delta_P^{t'}(x)$ for all annotation types $t' \in T$ for (i) the top 10 genes or diseases with the largest positive significance contributions w. r. t. t ($\mathbf{max}_x\, \Delta_P^t(x)$, see Supplementary Figure 3c), and (ii) the top 10 genes or diseases with the largest negative significance contributions w. r. t. t ($\mathbf{min}_x\, \Delta_P^t(x)$, see Supplementary Figure 3d).
- For every annotation type $t \in T$ if X is the node set of an induced subnetwork: A visualization of the subnetwork $G[X]$ where the individual genes' or diseases' significance contributions $\Delta_P^t(x)$ are encoded as node colors (see Figure 8c).
- For every annotation type $t \in T$: A plot showing the relevance score $S(X)$, the distribution of all relevance scores $S(\mathcal{M}(X))$ for the background model, as well as the resulting empirical P-value (see Figure 2b and Figure 8b).

The following plots are generated to visualize mappability (see Supplementary Figure 4 for an overview):

- A plot showing the frequency of elements $x \in X$ with non-empty annotation sets ($\mathcal{A}_x^t \neq \emptyset$) for all annotation types $t \in T$ (see Figure 4b).
- For every annotation type $t \in T$: A plot showing the distribution of annotation set sizes $|\mathcal{A}_x^t|$ for the elements $x \in X$ (see Figure 5b).
- For every annotation type $t \in T$: A Sankey plot linking the elements $x \in X$ to the top 10 most frequently occurring annotation terms of type t. For clusterings, term occurrences are normalized by the sizes of the clusters they are appearing in to avoid overrepresentation of larger clusters (see Figure 7b).

Setup of scalability tests

To evaluate the scalability of DIGEST in comparison to existing tools, we ran queries with identical gene sets against the APIs of DIGEST, PANTHER and g:Profiler (DAVID and WebGestalt were excluded due to lack of APIs that allow query execution time measurements). For this, we generated 3×10 random gene sets of sizes 10, 50 and 100, respectively. We then ran PANTHER's 'Functional enrichment test', g:Profiler's 'g:GOSt' analysis (performs functional enrichment analysis) and DIGEST's reference-free gene set validation route and measured the query execution times. The number of randomizations used for DIGEST's permutation tests was set to $n = 1000$ (default in web interface) and $n = 1$ (yields fairer comparison against tools that do not run permutation tests).

Implementation of Python package

The backend is implemented in Python 3 and pulls the disease ID mappings from the API of NeDRex, the disease attribute mappings (disease-associated genes and variants) from DisGeNET and the shared KEGG pathways from the API of KEGG. Gene set enrichment is computed via GSEApy's Enrichr module [16, 39]. The remaining mappings are obtained via the biothings_client package (https://github.com/biothings/biothings_client.py). A YAML file is provided to easily set up an environment with all dependencies.

Implementation of web interface and REST API

The web interface and REST API components are deployed using Docker and separated into four containers: (1) A backend service running a Django REST framework for request handling and execution of DIGEST's Python package. (2) A Redis service for task queuing and execution of incoming validation requests. (3) A PostgreSQL database to store and manage validation results. (4) A frontend implemented in vue.js, providing a graphical configuration option for execution of DIGEST, as well as the API documentation.

The REST API provides programmatic access to DIGEST's validation routes. Requests receive a unique task ID, used to check the execution status and request results. Moreover, set-up files for DIGEST's Python package can be obtained via the API, which improve the initialization time by loading the latest pre-built scoring matrices and mapping files from the server. Requests submitted through the web interface use the unique task ID to generate a permanently accessibly URL for the result page, allowing users to save it and return to the page at a later point. Besides the obtained empirical P-values, the result page presents the input data, the selected parameter configuration, as well as summary figures. If the user desires to compute gene- or disease-level significance contributions, the necessary computations are scheduled during idle times of the server in order not to block faster jobs. The current status of the computations can be checked at any time using the unique task URL. Moreover, if the user provides an email address, DIGEST notifies them once all computations have been carried out.

Key Points

- Solid prior evidence is necessary to carry hypotheses generated by computational systems medicine approaches toward translation.
- Existing validation tools only support limited input types and have limited statistical analysis capacities.
- DIGEST overcomes this limitation by computing empirical P-values quantifying functional and genetic coherence.

Availability

DIGEST's web interface and REST API are available at https://digest-validation.net and https://api.digest-validation.net, respectively. The Python package is available at https://pypi.org/project/biodigest. The source codes of the Python package, the web interface and

the REST API are freely available at https://github.com/bionetslab/digest, https://github.com/bionetslab/digest-web and https://github.com/bionetslab/digest-api, under the terms of the GNU General Public License, Version 3. A Jupyter notebook for getting familiar with DIGEST's Python package and reproducing the results reported in this paper is available at https://github.com/bionetslab/digest-tutorial. To further increase reproducibility, we generated an AIMe report [24], which is available at https://aime.report/mS7V2J.

Supplementary data

Supplementary data are available online at *Briefings in Bioinformatics*.

Author contributions statement

K. A. and D. B. B. conceived and designed the platform. K. A. developed the Python backend. A. M. implemented the web interface. J. B. and D. B. B. supervised the project. All authors provided critical feedback and discussion and assisted in interpreting the results, writing the manuscript and improving the web service.

Acknowledgements

Figure 1 was created with BioRender.com.

Funding

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreements No. 777111 (A.M., J.B.). This publication reflects only the authors' view and the European Commission is not responsible for any use that may be made of the information it contains. This work was supported by the German Federal Ministry of Education and Research (BMBF) within the framework of the e:Med research and funding concept (grant 01ZX1908A and grant 01ZX1910D) (J.B.). J.B. was partially funded by his VILLUM Young Investigator Grant No. 13154.

References

1. Amberger JS, Bocchini CA, Scott AF, *et al.* OMIM.org: leveraging knowledge across phenotype-gene relationships. *Nucleic Acids Res* 2019;**47**(D1):D1038–43.
2. Ashburner M, Ball CA, Blake JA, *et al.* Gene Ontology: tool for the unification of biology. *Nat Genet* 2000;**25**(1):25–9.
3. Bateman A, Martin MJ, Orchard S, *et al.* UniProt: the universal protein knowledgebase in 2021. *Nucleic Acids Res* 2021;**49**:D480–9.
4. Bernett J, Krupke D, Sadegh S, *et al.* Robust disease module mining via enumeration of diverse prize-collecting Steiner trees. *Bioinformatics* 2022.
5. Bodenreider O. The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic Acids Res* 2004; **32**(D267):1.
6. Davies DL, Bouldin DW. A cluster separation measure. *IEEE Trans Pattern Anal Mach Intell, PAMI-1*:224–227 1979.
7. Di Paolo G, Kim T-W. Linking lipids to alzheimer's disease: cholesterol and beyond. *Nat Rev Neurosci* 2011;**12**(5): 284–96.
8. Dunn JC. A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. *J Cybern* 2008;**3**(32–57):1.
9. Gene Ontology Consortium. The gene ontology resource: enriching a GOld mine. *Nucleic Acids Res* 2021;**49**(D1):D325–34.
10. Ghiassian SD, Menche J, Barabási AL. A DIseAse MOdule Detection (DIAMOnD) algorithm derived from a systematic analysis of connectivity patterns of disease proteins in the human interactome. *PLoS Comput Biol* 2015;**11**(1004120):4.
11. Guo M, Yanan Y, Wen T, *et al.* Analysis of disease comorbidity patterns in a large-scale china population. *BMC Med Genomics* 2019;**12**(Suppl 12):177.
12. Howe KL, Achuthan P, Allen J, *et al.* Ensembl 2021. *Nucleic Acids Res* 2020;**49**(D1):D884–91.
13. Huang DW, Sherman BT, Lempicki RA. Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. *Nat Protoc* 2009;**4**:44–57.
14. Kanehisa M, Sato Y, Kawashima M, *et al.* KEGG as a reference resource for gene and protein annotation. *Nucleic Acids Res* 2016;**44**(D1):D457–62.
15. Kotlyar M, Pastrello C, Malik Z, *et al.* IID 2018 update: context-specific physical protein-protein interactions in human, model organisms and domesticated species. *Nucleic Acids Res* 2019;**47**(D1):D581–9.
16. Kuleshov MV, Jones MR, Rouillard AD, *et al.* Enrichr: a comprehensive gene set enrichment analysis web server 2016 update. *Nucleic Acids Res* 2016;**44**(W1):W90–7.
17. Langhauser F, Casas AI, Dao V-T-V, *et al.* A diseasome cluster-based drug repurposing of soluble guanylate cyclase activators from smooth muscle relaxation to direct neuroprotection. *NPJ Syst Biol Appl* 2018;**4**:8.
18. Law CW, Chen Y, Shi W, *et al.* voom: Precision weights unlock linear model analysis tools for RNA-seq read counts. *Genome Biol* 2014;**15**(2):R29.
19. Lazareva O, Baumbach J, List M, *et al.* On the limits of active module identification. *Brief Bioinform* 2021;**22**(5):bbab066.
20. Lazareva O, Canzar S, Yuan K, *et al.* BiCoN: network-constrained biclustering of patients and omics data. *Bioinformatics* 2021;**37**:2398–404.
21. Levi H, Elkon R, Shamir R. DOMINO: a network-based active module identification algorithm with reduced rate of false calls. *Mol Syst Biol* 2021;**17**(e9593).
22. Liao Y, Wang J, Jaehnig EJ, *et al.* WebGestalt 2019: gene set analysis toolkit with revamped UIs and APIs. *Nucleic Acids Res* 2019;**47**:W199–205.
23. Maglott D, Ostell J, Pruitt KD, *et al.* Entrez gene: gene-centered information at NCBI. *Nucleic Acids Res* 2011;**39**(Database issue):D52–7.
24. Matschinske J, Alcaraz N, Benis A, *et al.* The AIMe registry for artificial intelligence in biomedical research. *Nat Methods* 2021;**18**:1128–31.
25. Mechteridis K, Lauber M, Baumbach J, *et al.* KeyPathwayMineR: De novo pathway enrichment in the R ecosystem. *Front Genet* 2021;**12**:812853.
26. Mi H, Ebert D, Muruganujan A, *et al.* PANTHER version 16: a revised family classification, tree-based classification tool, enhancer regions and extensive API. *Nucleic Acids Res* 2021;**49**: D394–403.

27. Mungall CJ, McMurry JA, Köhler S, *et al.* The monarch initiative: an integrative data and analytic platform connecting phenotypes to genotypes across species. *Nucleic Acids Res* 2017;**45**(D1):D712–22.
28. Nogales C, Mamdouh ZM, List M, *et al.* Network pharmacology: curing causal mechanisms instead of treating symptoms. *Trends Pharmacol Sci* 2022;**43**(2):136–50.
29. Oughtred R, Stark C, Breitkreutz B-J, *et al.* The BioGRID interaction database: 2019 update. *Nucleic Acids Res* 2019;**47**(D1): D529–41.
30. Piñero J, Ramírez-Anguita JM, Saüch-Pitarch J, *et al.* The DisGeNET knowledge platform for disease genomics: 2019 update. *Nucleic Acids Res* 2020;**48**(D1):D845–55.
31. Rath A, Olry A, Dhombres F, *et al.* Representation of rare diseases in health information systems: the orphanet approach to serve a wide range of end users. *Hum Mutat* 2012;**33**(5): 803–8.
32. Uku Raudvere, Liis Kolberg, Ivan Kuzmin, Tambet Arak, Priit Adler, Hedi Peterson, and Jaak Vilo. g:Profiler: a web server for functional enrichment analysis and conversions of gene lists (2019 update). *Nucleic Acids Res.*, **47**:W191–8, 2019.
33. Rogers FB. Medical subject headings. *Bull Med Libr Assoc* 1963;**51**: 114–6.
34. Rousseeuw PJ. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *J Comput Appl Math* 1987;**20**: 53–65.
35. Ruch P, Gobeill J, Lovis C, *et al.* Automatic medical encoding with SNOMED categories. *BMC Med Inform Decis Mak* 2008;**8 Suppl** (1):S6.
36. Sadegh S, Skelton J, Anastasi E, *et al.* Network medicine for disease module identification and drug repurposing with the NeDRex platform. *Nat Commun* 2021;**12**:6848.
37. Schaefer MH, Serrano L, Andrade-Navarro MA. Correcting for the study bias associated with protein-protein interaction measurements reveals differences between protein degree distributions from different cancer types. *Front Genet* 2015;**6**:260.
38. Schriml LM, Mitraka E, Munro J, *et al.* Human disease ontology 2018 update: classification, content and workflow expansion. *Nucleic Acids Res* 2019;**47**:D955–62.
39. Subramanian A, Tamayo P, Mootha VK, *et al.* Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proc Natl Acad Sci U S A* 2005;**102**(43):15545–50.
40. Tweedie S, Braschi B, Gray K, *et al.* Genenames.org: the HGNC and VGNC resources in 2021. *Nucleic Acids Res* 2021;**49**: D939–46.