Protocol

Protocol to identify host-viral protein interactions between coagulation-related proteins and their genetic variants with SARS-CoV-2 proteins

Here, we describe a bioinformatics pipeline that evaluates the interactions between coagulation-related proteins and genetic variants with SARS-CoV-2 proteins. This pipeline searches for host proteins that may bind to viral protein and identifies and scores the protein genetic variants to predict the disease pathogenesis in specific subpopulations. Additionally, it is able to find structurally similar motifs and identify potential binding sites within the host-viral protein complexes to unveil viral impact on regulated biological processes and/or host-protein impact on viral invasion or reproduction.

Publisher's note: Undertaking any experimental protocol requires adherence to local institutional guidelines for laboratory safety and ethics.
Protocol

Protocol to identify host-viral protein interactions between coagulation-related proteins and their genetic variants with SARS-CoV-2 proteins

David D. Holcomb,1,4,5,* Katarzyna I. Jankowska,1,4 Nancy Hernandez,1 Kyle Laurie,1 Jacob Kames,1 Nobuko Hamasaki-Katagiri,1 Anton A. Komar,2 Michael DiCuccio,3 and Chava Kimchi-Sarfaty1,6,*

1Center for Biologics Evaluation and Research, Office of Tissues and Advanced Therapies, Division of Plasma Protein Therapeutics, Food and Drug Administration, Silver Spring, MD, USA
2Center for Gene Regulation in Health and Disease, Department of Biological, Geological and Environmental Sciences, Cleveland State University, Cleveland, OH, USA
3National Center of Biotechnology Information, National Institutes of Health, Bethesda, MD, USA
4These authors contributed equally
5Technical contact
6Lead contact
*Correspondence: holcombddf@gmail.com (D.D.H.), chava.kimchi-sarfaty@fda.hhs.gov (C.K.-S.)
https://doi.org/10.1016/j.xpro.2022.101648

SUMMARY

Here, we describe a bioinformatics pipeline that evaluates the interactions between coagulation-related proteins and genetic variants with SARS-CoV-2 proteins. This pipeline searches for host proteins that may bind to viral protein and identifies and scores the protein genetic variants to predict the disease pathogenesis in specific subpopulations. Additionally, it is able to find structurally similar motifs and identify potential binding sites within the host-viral protein complexes to unveil viral impact on regulated biological processes and/or host-protein impact on viral invasion or reproduction.

For complete details on the use and execution of this protocol, please refer to Holcomb et al. (2021).

BEFORE YOU BEGIN

The clinical data from COVID-19 patients clearly display the strong association between coagulation-related proteins and COVID-19 infections (Levi et al., 2020). Elevated circulating D-dimer levels, a prolonged prothrombin time (PT) and low ADAMTS13 plasma levels have been observed in patients with COVID-19 (Al-Samkari et al., 2020), (Levi et al., 2020) (Mancini et al., 2020) (Bazzan et al., 2020). Nevertheless, the detailed mechanism of how the virus dysregulate the coagulation pathway is not fully understood. While there are numerous factors including age, weight and various medical underlying conditions that increase the risk for severe COVID-19 and the extent of thrombotic events or coagulopathy after infection (Zhou et al., 2020) (National Center for Immunization and Respiratory Diseases NCIRD; Division of Viral Diseases, 2022) the interactions between viral proteins and coagulation related proteins may also be responsible for the prothrombotic phenotype that is seen in COVID-19 patients. Some proteins associated with the coagulation cascade that can be directly or indirectly affected by viral proteins has been already identified (Ortega-Bernal et al., 2022), (Pfefferle et al., 2011).

The enrichment analysis combined with human proteomics data revealed the direct interactions of virus proteins with coagulation cascade proteins including fibrinogen (FGA and FGG) or protein S (PROS1) (Ortega-Bernal et al., 2022).
The interaction of coagulation factor X (FX) that cleaved the SARS-CoV-2 spike protein has been experimentally validated (Du et al., 2007), (Kastenhuber et al., 2022).

Through computational modeling, we previously examined the binding of Poly(A) Binding Protein Cytoplasmic 4 (PABPC4), Serine/Cysteine Proteinase Inhibitor Clade G Member 1 (SERPING1) and Vitamin K epOxide Reductase Complex subunit 1 (VKORC1) with SARS-CoV-2 proteins and have shown that some of these protein genetic variants may impact COVID-19 severity (Holcomb et al., 2021). In addition, it has been suggested that VKORC1 variant 1629A confers protection against thrombotic complications of COVID-19 and that differences in its allele frequency are partially responsible for the differences in COVID-19 severity in Asians population (Janssen and Walk, 2020).

These studies strongly suggest that in addition to the protein-protein interaction between host-viral system also the proteins’ genetic variation should be considered to fully understand the source of coagulation pathway dysregulation during COVID-19 infection.

The protocol presented here describes a step-by-step bioinformatics approach to identify host-viral protein interactions. This pipeline identifies the host-proteins that may interact with the virus/virally encoded proteins. The selected host-protein variants are then identified in various subpopulations by searching numerous databases including COVID-19 host genetics initiative (HGI) GWAS data, dbSNP or ClinVar. All identified missense and synonymous variants are rated based on conservation score, change in mRNA minimum free energy, and codon usage using numerous in-silico tools. The selected host-viral proteins are then aligned to find structurally similar sites and to identify potential binding sites.

The identification of host-viral protein interactions may help to understand the origin of coagulopathy during or after COVID-19 infection. The evaluation of host-viral protein interactions for different host protein variants can help to understand the distinct disease pathogenesis in minority groups and subpopulations. Moreover, by applying sequence and structural alignments, the pipeline finds and visualizes the similarities in the specific human and viral proteins structures and predicts the specific regions of protein-protein interaction in the viral-host system.

The identified interactions may help to understand the viral impact on regulated biological process and/or protein impact on viral invasion or reproduction which can further help to better understand the outcomes of host-viral interactions critical in development of potential novel drugs and treatment options.

**Institutional permissions**
All resources are publicly available. Please cite any sources used for any publications resulting from this protocol.

**Hardware**
We strongly suggest at least 16 GB of memory and 2 cores. At least 150 GB of hard drive space will be required. In addition, a network connection is required to query webservers. Linux is strongly recommended, particularly an Ubuntu or Debian distribution.

**Installing code**

© Timing: 10 days

1. Python dependencies are the backbone of the code. Most of the code is written in Python.
   a. Python version 3.7.12 (Van Rossum and Drake, 2009).
b. Jupyter Notebook 3.4.3 (Kluyver et al., 2016) is useful for the bulk of the calculations. They have been automated to run in Jupyter Notebook. The code can be run in any environment but has been developed and documented for Jupyter Notebook.
   i. The easiest installation method uses Anaconda (Anaconda Documentation, 2022).

```bash
conda install jupyterlab
```

c. Entrez from BioPython 1.69 (Cock et al., 2009) is necessary to query NCBI servers for information on sequences and variants.

```bash
conda install -c conda-forge biopython
```

d. Update your Entrez email and API key. If you do not have an API key for Entrez services, more information is provided here: https://ncbiinsights.ncbi.nlm.nih.gov/2017/11/02/new-api-keys-for-the-e-utilities/.

e. The following Python libraries are needed: Requests, Pandas, Prody, Numpy, BeautifulSoup, urllib, xlrd, subprocess, multiprocessing. Additional libraries and versions used are included in requirements.txt [troubleshooting 1].

```bash
conda install requests, pandas, prody, numpy, beautifulsoup, urllib, xlrd, subprocess, multiprocessing
```

2. Associated code has many dependencies. Any further software requirements are listed here, and there are no further hardware requirements [troubleshooting 2].

a. Clustal Omega 1.2.4 (Sievers et al., 2011) is necessary for aligning sequences to generate multiple sequence alignments and conservation scores.
   i. Download source code from http://www.clustal.org/omega/#Download, and extract.
   ii. Move into the extracted directory, then configure, make, and make install.

```bash
./configure
make
sudo make install
```

   iii. The executable should be installed to a directory in the path, preferably in /usr/bin. If not placed in /usr/bin, you will need to change clustalo_cmd in tools.align_sequences.

b. KineFold (Xayaphoummine et al., 2005) randomly estimates mRNA folding energy for a given nucleotide sequence. This is useful for comparing wild-type and variant sequences.
   i. Download the KineFold executable from http://kinefold.curie.fr/download.html.
   ii. Extract the downloaded file and move into a directory in the $PATH.

c. NUPACK 3.0.6 (Zadeh et al., 2010) also computes mRNA stability.
   i. Register at the NUPACK website, and wait to receive a password.
   ii. Download source files from NUPACK website http://www.nupack.org/downloads/source.
   iii. Add the bin directory to your $PATH variable.

d. RNAfold (Lorenz et al., 2011) also computes mRNA stability.
   i. Download the ViennaRNA (version 2.4.10) source code from https://www.tbi.univie.ac.at/RNA/ and extract the file.
   ii. Move into the extracted directory, then ./configure, make, and make install.

```bash
./configure
make
sudo make install
```
e. HH-Suite 3.3.0 (Steinegger et al., 2019). This is required for NetSurfP.
   i. The easiest way to install is from conda.

```
conda install -c bioconda hhsuite
```

ii. Alternately, may be installed from GitHub.

```
git clone https://github.com/soedinglab/hh-suite.git
```

f. Uniclust (Mirdita et al., 2017). This is required for HH-Suite and NetSurfP.
   i. Download Uniclust30 version 2018_08 for HHsuite from http://gwdu111.gwdg.de/~combiol/uniclust/2018_08/.
   ii. More up-to-date versions may be used, but have not been tested.

g. Update the uniclust and hhsuite_model variables in compute_features.py to the appropriate locations.

h. NetSurfP 2.0 (Klausen et al., 2019) estimates the accessible surface area of the protein at different amino acids in the protein sequence. This is useful for determining which amino acids are accessible to solvent or binding.
   i. Install from pip.

```
pip install netsurfp2
```

   ii. Alternately, request download from Downloads, Version 2.0 - Any at https://services.healthtech.dtu.dk/service.php?NetSurfP-2.0.
   iii. After downloading and extracting file, follow instructions in README.md.
   iv. Finally, install HH-Suite and Uniclust as described above.

i. NetPhos 3.1b (Blom et al., 1999) is used to estimate the phosphorylation potential of a protein at each position in the sequence.
   i. Request download from Downloads, using the appropriate OS or system at https://services.healthtech.dtu.dk/service.php?NetPhos-3.1.
   ii. Extract the package and move into the resulting directory.
   iii. Install as directed by the readme file.
   iv. Add the executable directory to your $PATH, or copy the executable to a bin already in your $PATH.

j. NetOGlyc 3.1 (Hansen et al., 1998) is used to estimate the O-linked glycosylation potential of a protein at each position in the sequence.
   i. Request download from Downloads, using the appropriate OS or system at https://services.healthtech.dtu.dk/service.php?NetOGlyc-4.0.
   ii. Extract the package and move into the resulting directory.
   iii. Install as directed by the readme file.
   iv. Add the executable directory to your $PATH, or copy the executable to a bin already in your $PATH.

k. NetNGlyc 1.0 (Gupta and Brunak, 2002) is used to estimate the N-linked glycosylation potential of a protein at each position in the sequence.
   i. Request download from Downloads, using the appropriate OS or system at https://services.healthtech.dtu.dk/service.php?NetOGlyc-4.0.
   ii. Extract the package and move into the resulting directory.
   iii. Install as directed by the readme file.
   iv. Add the executable directory to your $PATH, or copy the executable to a bin already in your $PATH.
I. Coarse-grained co-translational folding model (Jacobs and Shakhnovich, 2017), but primarily the rare-codon enrichment calculation.
   i. An adapted version of this code is included with the rest of the code for this protocol. This program is useful for locating regions in the nucleotide sequence that are enriched in rare codons, and where such enrichment is conserved across multiple species.

m. Change the path to rc_enrichment in compute_features.py to the appropriate directory where the coarse-grained co-translational folding model is installed.

```python
sys.path.insert(0, '/media/home/workspace/Submission/rc_enrichment')
```

n. Pymol 1.8.4.0 is useful for visualizing protein crystal structures. This may elucidate similar structural motifs or important binding sites between proteins.
   i. The easiest way to install is from the default package manager.
   
   ```bash
   sudo apt-get install pymol
   ```

   ii. Copy the text in the gray box from here https://pymolwiki.org/index.php/InterfaceResidues and save as InterfaceResidues.py.

o. Rosetta 3.10 is useful for modeling and designing protein structures. It’s also helpful for docking proteins to identify potential binding sites.
   i. Apply for an academic license or purchase a license here: https://www.rosettacommons.org/software/license-and-download.
   ii. When you receive an academic license, as well as a username and password, enter them here: https://www.rosettacommons.org/software/academic/.
   iii. Download rosetta_src_3.10_bundle.tgz, rosetta_bin_linux_3.10_bundle.tgz and rosetta_3.10_user_guide.tgz and extract them to an appropriate directory.

p. Update all variables at the top of analyze_all_variants.ipynb, including the directory containing the files (also used as output).

**Note:** Unless specified otherwise, all programs should be installed in a directory added to the $PATH variable (or placed in a main bin directory).

### Other resources

**Timing:** 8 h

3. EVmutation (Hopf et al., 2017).
   a. An adapted version of the python wrappers is included in EVmutation folder.
   b. Requires PLMC.
   i. Can be downloaded from GitHub with

   ```bash
git clone https://github.com/debbiemarkslab/plmc.git
```

   ii. Then, inside the either install with

   ```bash
make all
```

   if using a single core Linux system. If using openMP, install with

   ```bash
make all-openmp
```
iii. Finally, either add the plmc directory to your $PATH or copy the plmc executable to a directory in your $PATH.

4. RemuRNA \cite{Salari2013}.
   a. Can be installed from conda with
   
   ```bash
   conda install -c bioconda remurna
   ```

5. mFold 3.6 \cite{Zuker2003}.
   a. Download mFold 3.6 from \url{http://www.unafold.org/mfold/software/download-mfold.php}.
   b. After extracting and moving into the resulting directory.
   
   ```bash
   ./configure
   make
   sudo make install
   ```

---

## Download necessary datasets

*Timing: Days, dependent on download speeds*

6. Download the newest BIOGRID \cite{Oughtred2021} release (or other protein-protein interaction database such as STRING \cite{Szklarczyk2019}).
   a. From BIOGRID website, select downloads at top.
   b. Select Current-Release.
   c. Select BIOGRID-ALL-x.x.xxx.tab3.zip, where x.x.xxx is the version. For replication purposes, we used BIOGRID-ALL-4.4.209.tab3.zip.
   d. Download file and unzip to location specified in biogrid_file in analyze_all_variants.ipynb.

7. Download all host proteins involved in desired biological process (for example coagulation) from Gene Ontology.
   a. Search your biological function on AmiGO \cite{Carbon2009} at \url{http://amigo.geneontology.org/amigo/landing}. A view of the website, including the query and one possible result is shown in Figure 1.
   b. Select Ontology.
   c. Choose the closest matching term (blood coagulation).
   d. Filter for only host proteins (Homo sapiens).
   e. Select Download.
   f. Add Gene/product (bioentity_label) to top of the Selected fields.

Download and save text to location in go_file variable in analyze_all_variants.ipynb.

8. Download latest GWAS output.
   a. At COVID-19 HGI \cite{COVID-19HostGeneticsInitiative2021} website, select Downloads, then latest release with data.
   b. Select a dataset, and download the GRCh37 liftover and associated tbi file. For replication, we used the Release 6 version of the A2_ALL_leave_23andme study.
   c. Decompress the file to the location in the gwas_file variable.

9. As implied by previous steps, set the variables biogrid_file, go_file, and gwas_file appropriately in the analyze_all_variants.ipynb. The initialization of analyze_all_variants.ipynb is shown in Figure 2.
10. Set the host, host_organism, and virus variables appropriately: host and virus should be set to the taxonomy IDs of the host and virus organisms.

⚠️ CRITICAL: analyze_all_variants.ipynb contains variables representing the locations of these datasets, as well as the prefix to be used for output files, and the host and viral names and taxonomy IDs. Please set these variables in the notebook because you attempt to run it, as the program will need the appropriate locations of the files to read them. You can find them near the top of the notebook.

**KEY RESOURCES TABLE**

| REAGENT or RESOURCE | SOURCE | IDENTIFIER |
|---------------------|--------|------------|
| Deposited data      |        |            |
| BIOGRID 4.4.209     | (Oughtred et al., 2021) | RRID: SCR_007393 |
| Database of protein-protein interaction information. Used in step 1. | |

(Continued on next page)
| REAGENT or RESOURCE | SOURCE | IDENTIFIER |
|---------------------|--------|------------|
| AmiGO               | (Carbon et al., 2009) | RRID: SCR_002143 |
| COVID-19 HGI Release 6 | (COVID-19 Host Genetics Initiative, 2021) | RRID: SCR_022272 |
| dbsNP               | NCBI (Sherry et al., 2001) | RRID: SCR_002338 |
| ClinVar             | NCBI (Landrum et al., 2018) | RRID: SCR_006169 |
| Google Scholar      | Google | RRID: SCR_008878 |
| Entrez              | NCBI (Sayers et al., 2022) | RRID: SCR_016640 |
| Protein Data Bank   | Research Collaboratory for Structural Bioinformatics (Berman et al., 2000) | RRID: SCR_012820 |
| Codon and Codon-Pair Usage Tables (September 2021) | (Alexaki et al., 2019) | RRID: SCR_018504 |
| ESEfinder           | (Cartegni et al., 2002, 2003) | RRID: SCR_007088 |
| FAS ESS             | (Wang et al., 2004) | RRID: SCR_022517 |
| exonScan            | (Wang et al., 2004, Yeo and Burge, 2004) | RRID: SCR_022516 |
| I-TASSER            | (Yang et al., 2015) | RRID: SCR_014627 |
| Dali                | (Holm, 2019) | RRID: SCR_013433 |
| ZDock               | (Pierce et al., 2014) | RRID: SCR_022518 |

**Software and algorithms**

| REAGENT or RESOURCE | SOURCE | IDENTIFIER |
|---------------------|--------|------------|
| Python 3.7.12       | (Van Rossum and Drake, 2009) | RRID: SCR_008394 |
| Biopython 1.69      | (Cock et al., 2009) | RRID: SCR_007173 |
| BLAST               | (Altschul et al., 1990) | RRID: SCR_004870 |
| Clustal Omega 1.2.4 | (Sievers et al., 2011) | RRID: SCR_001591 |
| NetNGlyc 1.0        | (Gupta and Brunak, 2002) | RRID: SCR_001570 |
| NetOGLyC 3.1        | (Hansen et al., 1998) | RRID: SCR_009026 |
| NetPhos 3.1b        | (Blom et al., 1999) | RRID: SCR_017975 |
| NetSurfP 2.0        | (Klausen et al., 2019) | RRID: SCR_018781 |
| Coarse-grained Co-translational Folding Analysis and Rare Codon Enrichment | (Jacobs and Shakhnovich, 2017) | RRID: SCR_022271 |

(Continued on next page)
### STEP-BY-STEP METHOD DETAILS

Most steps are managed by `analyze_all_variants.ipynb`. If the dependencies are properly installed and the variables at the top of `analyze_all_variants.ipynb` are set, this notebook will run most of the following computations. In addition, locations of the code segments corresponding to each step are stated. A view of the menu of the notebook is shown in Figure 3.

Details are included here to further assist at each step. Additionally, timing is included here, but is not important and will be highly variable depending on the datasets involved.

#### Identify host proteins that bind to viral proteins and are involved in the biological process

**Timing:** 1 day

This step identifies proteins that both interact with viral proteins and are involved in a particular biological process (blood coagulation, in this instance). These proteins may indicate an impact of viral invasion on the relevant biological process, and variants in the genes encoding these proteins may impact viral binding.

1. In the second cell of `analyze_all_variants.ipynb`, Interactions are filtered from the BIOGRID release to include only those that involve host and viral proteins.
2. In the third cell, genes involved in the biological process (coagulation) are extracted from the AmiGO output, and only those included in the previous step are included.
3. In the fourth cell, each gene from the previous step is queried in NCBI’s Gene database, and the most appropriate RefSeq accession identifier is selected.

---

### REAGENT or RESOURCE SOURCE IDENTIFIER

| REAGENT or RESOURCE | SOURCE | IDENTIFIER |
|---------------------|--------|------------|
| %MinMax | Tool to predict rare codon clustering. Used in second step. Used in steps 4b and c. | (Rodriguez et al., 2018) RRID: SCR_022268 |
| Ensembl Variant Effect Predictor | Tool to predict effect of genetic variant and also contains minor allele frequencies. Used in steps 4b and c. | (McLaren et al., 2016) RRID: SCR_007931 |
| AL2CO conservation scores | Describes multiple conservation scores. Used in steps 4 and 5. | (Pei and Grishin, 2001) RRID: SCR_022267 |
| RNAfold 2.4.10 | Tool to compute mRNA stability. Used in steps 4b and c. | (Lorenz et al., 2011) RRID: SCR_008550 |
| NUPACK 3.0.6 | Tool to compute mRNA stability. Used in steps 4b and c. | (Zadeh et al., 2010) RRID: SCR_022274 |
| KineFold | Tool to compute mRNA stability. Used in steps 4b and c. | (Kayaphoummine et al., 2005) RRID: SCR_022273 |
| ESRseq hexamer score | Scores 6-mers for splicing potential. Used in step 5. | (Ke et al., 2011) RRID: SCR_022270 |
| HEXplorer score (ZEI and ZWS) | Scores 6-mers for splicing potential. Used in step 5. | (Erkelenz et al., 2014) RRID: SCR_022269 |
| Rosetta 3.10 | Software suite used for prediction of protein structure and docking. Used in step 9. | (Kahraman et al., 2013) RRID: SCR_015701 |
| Pymol 1.8.4.0 | Software to visualize proteins, protein complexes, and other molecules. Used in step 8. | (Pymol, 2020) RRID: SCR_000305 |
| Other | Dell Precision 7730 laptop with Intel Core i7-8850H, 32 GB memory, 500 GB solid state drive | Dell |

---

**Dell Precision 7730 laptop with Intel Core i7-8850H, 32 GB memory, 500 GB solid state drive**

**Dell**
Find sequences, synonymous variants, and missense variants for each gene

Timing: 2 days

This step finds the associated nucleotide sequences for each gene automatically identified in the previous section, including pre-spliced sequences, transcript fed to ribosome, and open reading frame read by ribosome. Additionally, this will find all synonymous and missense variants for each gene in dbSNP. This step will run BLAST and Clustal Omega to generate multiple sequence alignments of homologous nucleotide and amino acid sequences to compute conservation scores for all variants. Then, this step will query the Variant Effect Predictor to score the effect of any missense variants. Additionally, this step will use codon and codon pair usage from CoCoPUTs (Codon/Codon Pair Usage Tables (CoCoPUTs) (fda.gov)) to score the change in codon and codon pair usage using relative synonymous codon usage (RSCU), relative synonymous codon pair usage (RSCPU), and % MinMax. Additionally, variants are scored with KineFold and NUPACK to access impact on mRNA stability. This step will also score missense variants for the potential effect on post-translational modifications with NetPhos, NetOGLy, and NetNGlyc. Finally, NetSurfP is used to predict the accessible surface area of the protein to determine if variants occur on the surface or the interior of the protein.
Variants in conserved regions or variants that effect biochemical features of the DNA or the protein may impact viral binding or invasion. Synonymous variants may impact protein expression and protein conformation in some circumstances (Sauna and Kimchi-Sarfaty, 2011; Jankowska et al., 2022).

4. This occurs in the fifth cell of analyze_all_variants.ipynb. For each gene and associated RefSeq accession ID:
   a. run `bio_tools.get_all_seqs` with the associated gene name and RefSeq accession ID.
   ```python
   seqs = bio_tools.get_all_seqs(genename, nm, write=True)['seqs']
   ```
   b. Then, find and score all synonymous variants.
   ```python
   tmpd = get_dbSNP.get_syn(genename, seqs['ORF'], nm)
   ```
   c. Finally, find and score all missense variants.
   ```python
   tmpd = get_dbSNP.get_missense(genename, seqs['ORF'], nm)
   ```

**Query GWAS data and score associated variants**

© Timing: 2 days

This step will find all variants in the host genes identified in steps 1 and 2 with a p-value less than 0.05, and then score the variants. As part of this process, we gather all identifiers for each variant, including the dbSNP accession ID, as well as transcript coordinate and genomic coordinate HGVS identifiers. This is useful when we query them in databases such as dbSNP, ClinVar, and Google Scholar. Because these variants almost exclusively appear in introns and UTRs, we score them with splicing predictors such as ESRseq scores, HEXplorer scores, ESEfinder, FAS ESS, and ExonScan.

These variants may impact viral binding to the host protein, or expression of the protein and impact of the viral binding. While the code is specific to COVID-19 HGI formatting and column names, other GWAS studies may be used by changing the relevant column names.

5. This occurs in the sixth cell of analyze_all_variants.ipynb. Give the GWAS file location and the gene names to the `gwas_pipeline` function. All other columns in the GWAS data can be changed, but are set to defaults for COVID-19 HGI datasets. [troubleshooting 3][troubleshooting 4].

```python
gwas_data = parse_gwas.gwas_pipeline(gwas_file, genes, prefix=gwas_prefix, index_col="SNP", p_col="all_inv_var_meta_p", chromosome="CHR", position="POS", ref="REF", alt="ALT", plim=0.05, translation=gwas_prefix + ".trans")
```
Query Dali for structurally similar proteins

© Timing: 3 days

This step will identify any proteins that are structurally homologous to the viral protein and that bind to the host protein. This may elucidate specific motifs common to viral and host proteins, giving insight to the way viral and host proteins bind. Moreover, if the resulting homologs are involved in another biological process, this may indicate other effects of viral proteins on the host.

6. For each viral protein that interacts with one of the host proteins identified in steps 1 and 2:
   a. Query the protein sequence for the viral protein on NCBI’s protein database: https://www.ncbi.nlm.nih.gov/protein/. An example query output is given in Figure 4.

   Note: In some instances, it may be useful to filter for the organism, as shown in Figure 4.

   b. Query the protein in the Protein Data Bank.
i. If you can’t find a quality structure, move on to step c.

ii. Otherwise, you may download the structure and skip to step d.

Note: You may query the protein name, or the sequence. A good structure should be of high quality, high resolution (low Angstroms), complete (full sequence included), and a realistic environment.

c. Submit the protein sequence from step A to I-TASSER to be modeled.

Note: If you do not have an account, you will have to register an account. For replication, we generally ignore the options for modeling. A view of the I-TASSER interface is shown in Figure 5.
d. Download the file corresponding to the produced structure.
e. Submit this file to Dali under the PDB search header.
f. When your results are finished from Dali, open a python terminal, then type:

```python
import compute_features
compute_features.parse_dali(html, gene, path=directory)
```

where 'html' is the link given from the Dali run with the results, ‘gene’ is the name of the viral protein, and ‘directory’ is the location to write to.

7. Afterwards, run the final three cells of analyze_all_variants.ipynb. The third-to-last and second-to-last cells create helping dictionaries. The last cell cross-references structural homologs against proteins known to interact with the host protein of interest.

### Align structures to identify shared motifs

© Timing: 1 day

This step will visualize the shared structural motifs between viral proteins and homologous host proteins that also bind to other host proteins. In particular, this step will identify the regions containing these structural motifs, which may particularly impact binding between host and viral proteins. This section doesn’t involve variants identified in prior sections.

8. For each viral protein that interacts with one of the host proteins associated with your biological process of interest. An example output is shown in Figure 6.
   a. Open the I-TASSER modeled structure from the previous section in Pymol.
   b. In Pymol, fetch the associated homologous structures. For each homologous structure identified for the viral protein, ABCD_E.

```bash
fetch ABCD
```

c. In the ‘all’ section on the right, click ‘H’, then ‘everything’.
d. In the ‘all ‘section on the right, click ‘S’, then ‘cartoon’.
e. Select ‘Display’, then ‘Sequence’.
f. For each homologous structure, highlight all chains not involved (everything except for E here). Then, in the ‘(sele)’ section on the right, click ‘H’, then ‘everything’.
g. For each homologous structure, highlight the chain involved (E). Then, in the Pymol terminal,

```bash
super ABCD, I-TASSER_model
```

to structurally align the homolog and the I-TASSER viral protein model.
h. Save the Pymol session or the image, as needed.

### Dock viral and host proteins

© Timing: 10 days

This step will indicate likely binding sites of viral and host proteins. This may further indicate vital regions that will impact viral binding and its influence on the biological process. Variants included in these regions may strongly impact binding of viral and host proteins.

9. For each viral protein and interacting host protein.
a. Find the protein sequence of the host protein and model this structure in I-TASSER as described above.

b. Submit both the host protein and viral protein structures to ZDock.

c. When ZDock is finished, load each resulting PDB file into Pymol and manually curate each docking for feasibility.

Note: If necessary, a literature review may indicate vital positions that are involved in binding.

d. For each acceptable output docking from ZDock, give to Rosetta Prepack as

```
/PATH/TO/ROSETTA/EXEC5/docking_prepack_protocol.linuxgccrelease -nstruct 1 -detect_disulf true -rebuild_disulf true -ex1 -ex2aro -overwrite -ignore_zero_occupancy false -in:file:s $PDB -unboundrot $pdb -partners A_B -suffix $SUFFIX -out:file:scorefile ${SUFFIX}.sc
```
Where /PATH/TO/ROSETTA/EXECS/ is the global path to the Rosetta executables, $PDB$ is the path to the docking file from ZDOCK, A and B are the names of the chains (viral and host proteins), $SUFFIX$ is the suffix of the output files you want to use.

e. Then, use:

```
/path/to/rosetta/execs/docking_protocol.linuxgccrelease -nstruct 1 -out:pdb_gz true -ex1 -ex2aro -in:file:$PDB -partners A_B -suffix $SUFFIX -out:file:scorefile ${SUFFIX}.sc
```

Again, set $PDB$, A and B, and $SUFFIX$ appropriately.

```
python scores_to_csv.py --directory directory --col total_score --prefix viral_gene_host_gene
```
f. Use scores_to_csv to compile the score files:
   where directory is the directory containing all the score files (.sc) generated by the previous step
   and viral_gene_host_gene is the names of the viral and host genes (you can modify this as desired).

g. Read through scores.tsv. The last number in each score/PDB file name indicates the Rosetta
   model generated, and the models with the lowest total_score should be selected.

h. For each of the optimal docking models, load the PDB into Pymol.

i. Visualize as desired. We suggest hiding everything then showing cartoon, and displaying
   sequence as described previously. Then, next to ‘all’ on the right, select ‘C’, then ‘by chain’,
   then ‘by chain’. A view of the Pymol interface is shown in Figure 7.

j. Click ‘File’, then ‘Run’, then browse to the location of ‘InterfaceResidues.py’.

k. Then, in the Pymol terminal, enter.
   where docked_struct is the docking of the viral and host proteins.

l. All highlighted positions are amino acids that interact between the two proteins and may war-
   rant further focus.

**EXPECTED OUTCOMES**

The code described here provides five primary outputs related to host proteins potentially interact-
ning with the virus: (i) Identification of host proteins that interact with viral proteins. (ii) The subset of
synonymous variants of the interacting host genes. (iii) The subset of missense variants of the inter-
acting host genes. (iv) Variants of the interacting host genes that are significantly impactful per the
GWAS study. (v) Structural alignments of viral proteins with structural homologs that also interact
with the host proteins.

This pipeline identifies coagulation related-proteins that interact with SARS-CoV-2 proteins but can
be easily modified and adjusted for the search of interactions (and their potential impact) between
proteins involved in any specific/defined biological pathway and/or process and proteins encoded
by any type of virus. The pipeline identifies the host-proteins that may interact with the virus/virally
encoded proteins, including selected host-protein variants (found in various subpopulations) by
screening numerous databases including COVID-19 host genetics initiative (HGI) GWAS data,
dbSNP or ClinVar. It should be noted here that, the subset of GWAS data includes only data with
p-value less than 0.05 (unless specified differently). This may exclude many variants, or even some
genes of interest entirely.

All identified missense and synonymous variants are then rated. The subset of synonymous variants
includes minor allele frequencies, conservation scores, rare codon enrichment, change in mRNA
minimum free energy from multiple tools, many codon and codon pair usage scores, and the num-
ber of values that are in the extreme. The subset of missense variants includes the same scores as the
synonymous variants, except that the conservation scores are computed for amino acid sequences,
rather than nucleotide sequences. Evaluation of host-viral interaction for different host protein var-
iants can help understand the distinct disease pathogenesis in minority groups. The Jupyter Note-
book rendering of the synonymous variants and scores is shown in Figure 8.

In the pipeline outcome, in addition to the GWAS data, the identified variants have genomic
and transcriptomic HGVS coordinates, 501 nucleotide flanking sequence (250 nucleotides on
left and right) for wild-type and mutant sequences, summaries of results from splicing tools,
minor allele frequencies in different populations, as well as any information on clinical significance
or publications referencing the variant. The rendering of the computed GWAS data is shown in
Figure 9.
Additionally, the pipeline will produce a list of host proteins that are structurally homologous to the viral proteins and also bind to the associated host proteins. This step identifies viral proteins that may target same host proteins via competition binding thus disturbing the host protein function and the processes that the protein regulates. Additionally, structural alignments of these proteins may elucidate the structurally homologous regions. Finally, the docked structures may imply a more significant impact on variants in amino acids near the protein-protein interface. The sequence and structural alignments may help to identify the host proteins that are critical for viral replication and highlight the host proteins that can be affected by viral invasion. Predicting the host-viral interaction can help in understanding the mechanisms of viral infection and potential targets for therapeutic interventions.

**Figure 8.** Visualization of the scored synonymous variants in coagulation-involved human proteins that interact with SARS-CoV-2 proteins (the output of the step 4)

This shows the identifier of the variant (with the gene location) as the row name. Primarily minor allele frequencies for different populations from GnomAD are shown. Not all columns are shown here.

**Figure 9.** Visualization of the filtered and processed GWAS data of coagulation-involved human proteins that interact with SARS-CoV-2 proteins (the output of the step 5)

This shows the location of the variant, and effect size of the variant on severe COVID-19 status. Additional columns are now shown.
interactions between the host protein and its variants with the viral proteins are critical in development of potential treatments.

**QUANTIFICATION AND STATISTICAL ANALYSIS**

For all synonymous and non-synonymous variants, quantify the number of computed features that are above the $95^{th}$ percentile or below the $5^{th}$ percentile. For some features such as post-translational modifications that are zero for most amino acids, use the $95^{th}$ percentile of nonzero values. Variants with more extreme values may be more impactful on the protein structure/function, or more common in the population and more impactful to the population at large. This occurs in the eighth and ninth cell of `analyze_all_variants.ipynb`.

Further analysis of impactful variants is usually manually curated for interesting, extreme, and common variants.

**LIMITATIONS**

While protein-protein interactions databases tend to be reliable, they may include potentially spurious results that have not been peer reviewed. To overcome this limitation, sources for protein-protein interactions should be manually curated to verify for credibility.

In addition, this protocol is exclusively in silico, so results should be interpreted as such. In vitro validation on the results may be helpful.

**TROUBLESHOOTING**

**Problem 1**

Any step: Python package not found.

**Potential solution**

Additional Python packages may not be installed. If Anaconda is installed, these can usually be installed with

```
conda install [package name]
```

The repository may need to be specified or added in Anaconda to find the appropriate package. Conda-forge and Bioconda are common repositories for many dependencies.

**Problem 2**

Permission denied when attempting to get sudo privileges for

```
sudo make install
```

**Potential solution**

You may move the executables to a bin in the $PATH, or add the directory containing the executable to your $PATH variable.

The easiest way to add this directory to your path variable is to include this statement in your `~/.bashrc` file. To do this, edit the `~/.bashrc` file. Add the line

```
export PATH="/PATH/TO/EXECUTABLE/F ILE:$PATH"
```
to the end of the file. Then, run

```
source ~/.bashrc
```

to update your $PATH variable.

**Problem 3**

Step 5: Columns expected but not found in GWAS result file.

**Potential solution**

In the gwas_pipeline function, certain columns are expected in the GWAS file, including the p-value of the variant impact, the chromosome, position, reference and alternate alleles. If the titles of these columns are different from the defaults in gwas_pipeline, feed the appropriate column names as arguments to gwas_pipeline. If these columns are not included in the GWAS file, it may be necessary to process the GWAS file to create these columns.

**Problem 4**

Step 5: Very little data in GWAS results, or the same to transcriptomic HGVS identifier used for many different variants in results.

**Potential solution**

The gwas_pipeline function includes a section to translate genomic HGVS identifiers to transcriptomic HGVS identifiers. If the batch Mutalyzer translation file is not used or available, the program will manually pass each variant to Mutalyzer, frequently resulting in duplicate entries and errors.

To solve this, specify a translation file by passing as a translation argument to gwas_pipeline. Then when the program pauses, feed the list of genomic HGVS identifiers to batch Mutalyzer and save the result as your translation file.

**RESOURCE AVAILABILITY**

**Lead contact**

Further information and requests for resources and reagents should be directed to and will be fulfilled by the lead contact, Chava Kimchi-Sarfaty (chava.kimchi-sarfaty@fda.hhs.gov).

**Materials availability**

All materials are publicly available from the U.S. Food and Drug Administration GitHub. All other required materials have been outlined above.

**Data and code availability**

Data are publicly available from BIOGRID, Gene Ontology, and COVID-19 HGI.

Code is publicly available from the U.S. Food and Drug Administration GitHub here: https://doi.org/10.5281/zenodo.6862872.

**ACKNOWLEDGMENTS**

We thank Dr. Zuben Sauna, CBER/FDA for editorial assistance. This work was supported by funds from the Center for Biologics Evaluation and Research, US Food and Drug Administration operating funds. The work was also funded by CBER, FDA special COVID-19 funds. Additionally, here are the contributors to the COVID19 HGI project: Acknowledgements (covid19hg.org).
Declarations of Interest

The authors declare no competing interests.

References

Alexaki, A., Kames, J., Holcomb, D.D., Athey, J., Santana-Quintero, L.V., Lam, P.V.N., Hamasaki-Katagiri, N., Osipova, E., Simonyan, V., Bar, M., and Kimchi-Sarfaty, C. (2019). Codon and codon-pair usage tables (CoCoPUTs): facilitating genetic variability analyses and recombinant gene design. J. Mol. Biol. 431, 2434–2441.

Al-Samkari, H., Karp Leaf, R.S., Dzik, W.H., Carlson, J.C., Fogerty, A.E., Waheed, A., and Rosovsky, R.P. (2020). COVID-19 and coagulation: bleeding and thrombotic manifestations of SARS-CoV-2 infection. Thrombosis Hemostasis 113, 489–500.

Altschul, S.F., Gish, W., Miller, W., Myers, E.W., and Lipman, D.J. (1990). Basic local alignment search tool. J. Mol. Biol. 215, 403–410.

Anaconda Documentation. (2022). (Anaconda Inc.) https://docs.anaconda.com/.

Bazzan, M., Montaruli, B., Sciascia, S., Cosseddu, D., Notaro, C., and Roccatello, D. (2020). Low ADAMTS13 plasma levels are predictors of mortality in COVID-19 patients. Intern. Emerg. Med. 15, 861–863.

Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., and Bourne, P.E. (2000). The protein Data Bank. Nucleic Acids Res. 28, 235–242.

Blom, N., Gammeltoft, S., and Brunak, S. (1999). Sequence and structure-based prediction of eukaryotic protein phosphorylation sites. J. Mol. Biol. 294, 1351–1362.

Carbon, S., Ireland, A., Mungall, C.J., Shu, S., Marshall, B., and Lewis, S., AmiGO Hub, Web Presence Working Group (2009). AmiGO: online access to ontology and annotation data. Bioinformatics 25, 288–289.

Cartegni, L., Chew, S.L., and Krainer, A.R. (2002). Listeners to silence and understanding nonsense: exonic mutations that affect splicing. Nat. Rev. Genet. 3, 285–298.

Cartegni, L., Wang, J., Zhu, Z., Zhang, M.Q., and Krainer, A.R. (2003). ESEfinder: a web resource to identify exonic splicing enhancers. Nucleic Acids Res. 31, 3568–3571.

Cock, P.J.A., Antao, T., Chang, J.T., Chapman, B.A., Cox, C.J., Dalke, A., Friedberg, I., Hamelryck, T., Kauff, F., Wilczynski, B., et al. (2009). Biopython: freely available Python tools for computational molecular biology and bioinformatics. Bioinformatics 25, 1422–1423.

Du, L., Kao, R.Y., Zhou, Y., He, Y., Zhao, G., Wang, C., Jiang, S., Yuan, K.Y., Jin, D.Y., and Zheng, B.-J. (2007). Cleavage of spike protein of SARS coronavirus by protease factor Xa is associated with viral infectivity. Biochem. Biophys. Res. Commun. 359, 174–179.

Erkelens, D., Theiss, S., Otto, M., Widera, M., Peter, J.O., and Schaaf, H. (2014). Genomic HE XPlo ring allows landscaping of novel potential splicing regulatory elements. Nucleic Acids Res. 42, 10681–10697.

Gupta, R., and Brunak, S. (2002). Prediction of glycosylation across the human proteome and the correlation to protein function. Pac. Symp. Biocomput. 7, 310–322.

Hansen, J.E., Lund, O., Tolstrup, N., Gooley, A.A., Williams, K.L., and Brunak, S. (1998). NetOglyc: prediction of mucin type O-glycosylation sites based on sequence context and surface accessibility. Glycocon. J. 15, 115–130.

Holcomb, D., Alexaki, A., Hernandez, N., Hunt, R., Laurie, K., Kames, J., Hamasaki-Katagiri, N., Komar, A.A., DiCuccio, M., and Kimchi-Sarfaty, C. (2021). Gene variants of coagulation related proteins that interact with SARS-CoV-2. PLoS Comput. Biol. 17, e1008005-25.

Holm, L. (2019). DALI and the persistence of protein similarity. Protein Sci. 29, 128–140.

Hopf, T.A., Ingraham, J.B., Poelwijk, F.J., Scharfe, C.P.I., Springer, M., Sander, C., and Marks, D.S. (2017). Mutation effects predicted from sequence co-variation. Nat. Biotechnol. 35, 128–135.

COVID-19 Host Genetics Initiative (2021). Mapping the human genetic architecture of COVID-19. Nature 660, 472–477.

Jacobs, W.M., and Shakhnovich, E.I. (2017). Evidence of evolutionary selection for cotranslationally folding. Proc. Natl. Acad. Sci. USA 114, 11344–11349.

Jankowska, K.I., Meyer, D., Holcomb, D.D.F., Kames, J., Hamasaki-Katagiri, N., Kateni, U.K., Hunt, R.C., Ibba, J.C., and Kimchi-Sarfaty, C. (2022). Synonymous ADAMTS13 variants impact molecular characteristics and contribute to variability in active protein abundance. Blood Adv. https://doi.org/10.1182/bloodadvances.2022007655.

Janssen, R., and Walk, J. (2020). Vitamin K epoxide reductase complex subunit 1 (VKORC1) gene polymorphism as determinant of differences in COVID-19 related disease severity. Med. Hypotheses 144, 110218.

Kahraman, A., Herzog, F., Leitner, A., Rosenberger, G., Aebersold, R., and Malmstro¨m, L. (2013). Cross-link guided molecular modeling with ROSETTA. PLoS One 8, e73411.

Kastenhuber, E.R., Mercadante, M., Nilsson-Payant, B., Johnson, J.L., Jaimes, J.A., Muecksch, F., Weisblum, Y., Bram, Y., Chandar, V., Whittaker, G.R., and Cantley, L. (2022). Coagulation factors directly cleave SARS-CoV-2 spike and enhance viral entry. Elife 11, e77444.

Ke, S., Shang, S., Kalachikov, S.M., Morozova, I., Yu, L., Russo, J.J., Ju, J., and Chasin, L.A. (2011). Quantitative evaluation of all exomers as exonic splicing elements. Genome Res. 21, 1360–1374.

Klausen, M.S., Jespersen, M.C., Nielsen, H., Jensen, K.K., Juritz, V.I., Sandlerby, C.K., Sommer, M.O.A., Winther, O., Nielsen, M., Petersen, B., and Marcattil, P. (2019). NetSurfP-2D: improved prediction of protein structural features by integrated deep learning. Proteins 87, 520–527.

Klyuver, T., Ragan-Kelley, B., Fernandez Perez, G.B., Perez, F., Bussonnier, M., Frederic, J., and Schmidt, eds. (IOS Press), pp. 87–90.

Landrum, M.J., Lee, J.M., Benson, M., Brown, G.R., Chao, C., Chitipiralla, S., Gu, B., Hart, J., Hoffman, D., Jang, W., and Maglott, D.R. (2018). ClinVar: improving access to variant interpretations and supporting evidence. Nucleic Acids Res. 46, D1062–D1067.

Levi, M., Thachil, J., Iba, T., and Levy, J.H. (2020). Coagulation abnormalities and thrombosis in patients with COVID-19. Lancet. Haematol. 7, e438–e440.

Lorenz, R., Bernhart, S.H., Hower, Z. Sieder, C., Tafer, H., Flamm, C., Stadler, P.F., and Hofacker, I.L. (2011). ViennaRNA package 2.0. Algorithms Mol. Biol. 6, 26.

Mancini, I., Baronciani, L., Artoni, A., Colpani, P., Biganzoli, M., Cozzi, G., Novembri, C., Boscoli Anzoletti, M., De Zan, V., Pagliari, M.T., and Peyvandi, F. (2020). The ADAMTS13 von Willebrand factor axis in COVID-19 patients. J. Thromb. Haemost. 19, 513–521.

McLaren, W., Gil, L., Hunt, S.E., Riat, H.S., Ritchie, G.R.S., Thomar, A., Flicek, P., and Cunningham, F. (2016). The ensemble variant effect predictor. Genome Biol. 17, 122.

Mirza, M., von den Driesch, L., Galiez, C., Martin, M.J., Soding, J., and Stennegger, M. (2017). Unicluster databases of clustered and deeply annotated protein sequences and alignments. Nucleic Acids Res. 45, D170–D176.

National Center for Immunization and Respiratory Diseases (NCIRD); Division of Viral Diseases (2022). Science brief: evidence used to update the list of underlying medical conditions associated with higher risk for severe COVID-19. CDC COVID-19 Science Briefs, A Hall, ed. (Centers for Disease Control and Prevention (US)).

Ortega-Bernal, D., Zarate, S., Martinez-Cardenas, M.D.L.A., and Bojalil, R. (2022). An approach to cellular tropism of SARS-CoV-2 through protein...
protein interaction and enrichment analysis. Sci. Rep. 12, 9599.

Oughtred, R., Rust, J., Chang, C., Breitkreutz, B.-J., Stark, C., Willems, A., Boucher, L., Leung, G., Kolas, N., Zhang, F., and Tyers, M. (2021). The BioGRID database: a comprehensive biomedical resource of curated protein, genetic, and chemical interactions. Protein Sci. 30, 187–200.

Pei, J., and Grishin, N.V. (2001). AL2CO: calculation of positional conservation in a protein sequence alignment. Bioinformatics 17, 700–712.

Pfefferle, S., Schöpf, J., Kögl, M., Friedel, C.C., Müller, M.A., Carbajo-Lozoya, J., Stellberger, T., von Dall’Armi, E., Herzog, P., Kallies, S., et al. (2011). The SARS-coronavirus-host interactome: identification of cyclophilins as target for pan-coronavirus inhibitors. PLoS Pathog. 7, e1002331.

Pierce, B.G., Wiehe, K., Hwang, H., Kim, B.-H., Vreven, T., and Weng, Z. (2014). ZDOCK server: interactive docking prediction of protein-protein complexes and symmetric multimers. Bioinformatics 30, 1771–1773.

Pymol. (2020). (L. Schrödinger, Producer) Retrieved from the PyMOL Molecular Graphics System, Version 2.0.

Rodriguez, A., Wright, G., Emrich, S., and Clark, P.L. (2018). %MinMax: a versatile tool for calculating and comparing synonymous codon usage and its impact on protein folding. Protein Sci. 27, 356–362.

Salari, R., Kimchi-Sarfaty, C., Gottesman, M.M., and Przytycka, T.M. (2013). Sensitive measurement of single-nucleotide polymorphism-induced changes of RNA conformation: application to disease studies. Nucleic Acids Res. 41, 44–53.

Sauna, Z.E., and Kimchi-Sarfaty, C. (2011). Understanding the contribution of synonymous mutations to human disease. Nat. Rev. Genet. 12, 683–691.

Sayers, E.W., Bolton, E.E., Brister, J.R., Canese, K., Chan, J., Comeau, D.C., Connor, R., Funk, K., Kelly, C., Kim, S., and Sherry, S.T. (2022). Database resources of the national center for biotechnology information. Nucleic Acids Res. 50, D20–D26.

Sherry, S.T., Ward, M.H., Kholodov, M., Baker, J., Phan, L., Smigielski, E.M., and Sirotkin, K. (2001). dbSNP: the NCBI database of genetic variation. Nucleic Acids Res. 29, 308–311.

Sievers, F., Wilm, A., Dineen, D., Gibson, T.J., Karplus, K., Li, W., Lopez, R., McWilliam, H., Remmert, M., Söding, J., and Higgins, D.G. (2011). Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega. Mol. Syst. Biol. 7, 539.

Steinegger, M., Meier, M., Mirdita, M., Vöhringer, H., Hausberger, S.J., and Söding, J. (2019). HH-suite3 for fast remote homology detection and deep protein annotation. BMC Bioinf. 20, 473.

Zadeh, J.N., Steenberg, C.D., Bois, J.S., Wolfe, B.R., Pierce, M.B., Khan, A.R., Dirks, R.M., and Pierce, N.A. (2010). NUPACK: analysis and design of nucleic acid systems. J. Comput. Chem. 32, 170–173.

Zhou, Y., Chi, J., Lv, W., and Wang, Y. (2020). Obesity and diabetes as high-risk factors for severe coronavirus disease 2019 (Covid-19). Diabetes Metab. Res. Rev. 37, e3377.

Zuker, M. (2003). Mfold web server for nucleic acid folding and hybridization prediction. Nucleic Acids Res. 31, 3406–3415.