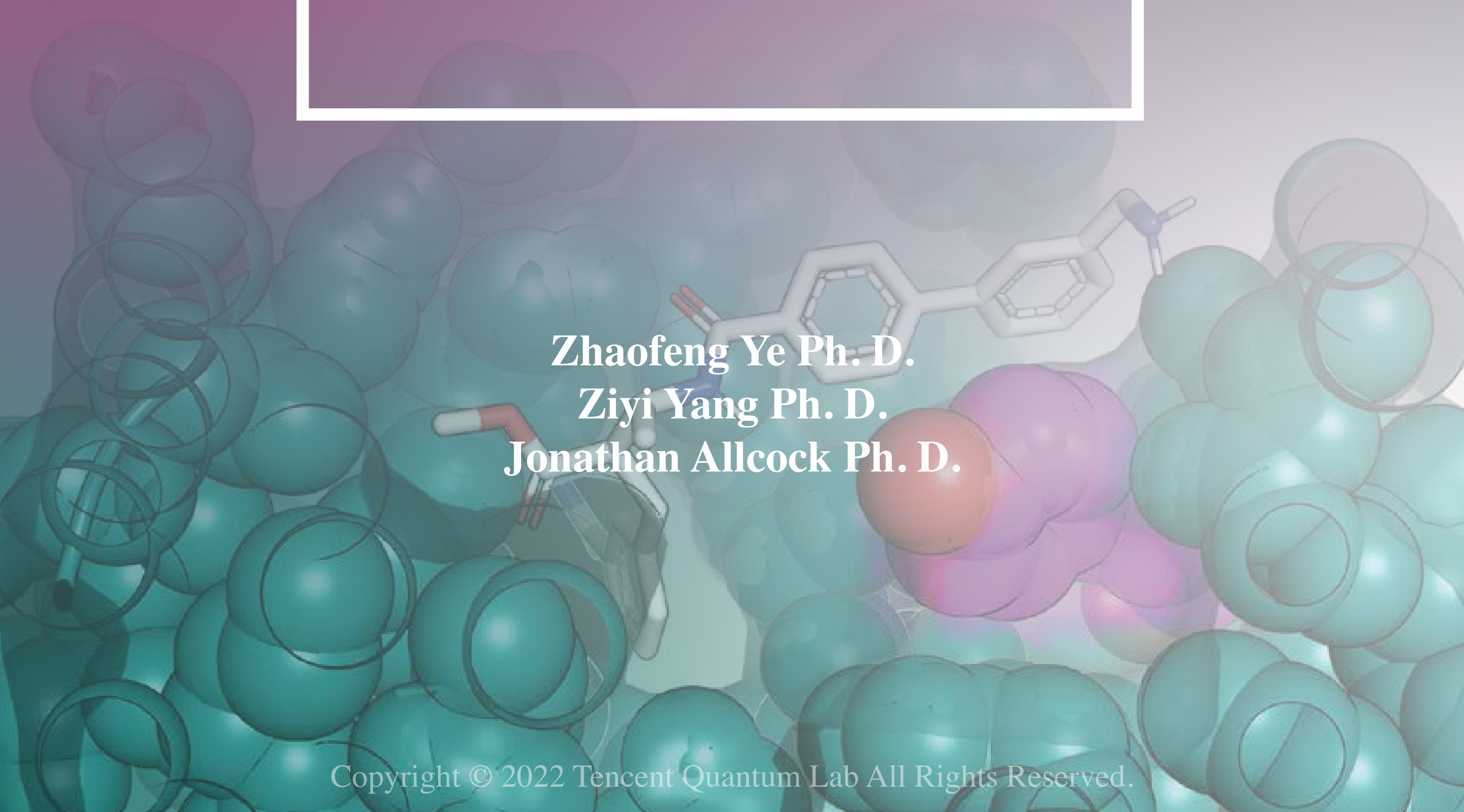
Tencent
Quantum Lab

Tutorial

for

Mutation-induced drug resistance DataBase

(MdrDB)



Zhaofeng Ye Ph. D.
Ziyi Yang Ph. D.
Jonathan Allcock Ph. D.

Table of Contents

1. Introduction

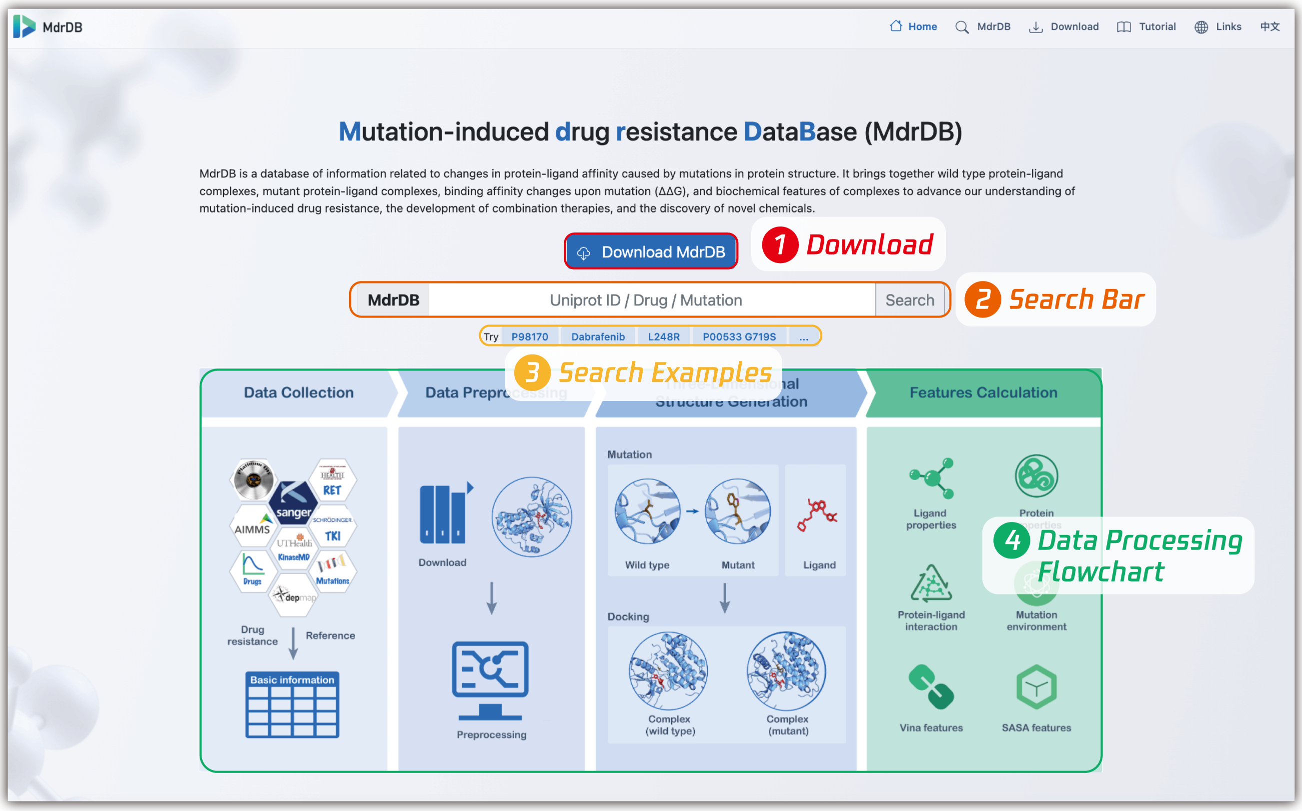
1.1. Background

MdrDB is a database of information related to changes in protein-ligand affinity caused by mutations in protein structure. It brings together wild type protein-ligand complexes, mutant protein-ligand complexes, binding affinity changes upon mutation (ΔΔG), and biochemical features of complexes to advance our understanding of mutation-induced drug resistance, the development of combination therapies, and the discovery of novel chemicals.

The goal of MdrDB is to collate the effects of mutation-induced protein structural changes on binding to small molecules. The database combines protein structures and annotations from the **Protein Data Bank (PDB)** and Uniprot, with drug data from **PubChem** and experimentally measured drug effects on wild type proteins and mutants from the **Genomics of Drug Sensitivity in Cancer (GDSC)**[1] and other databases. MdrDB provides wild type structures, mutant protein structures, wild type protein-ligand complex structures, and mutant protein-ligand complex structures for protein mutation studies and drug resistance modeling. A variety of mutation types are accounted for: in addition to single-point substitution mutations, complex mutations such as deletion mutations, insertion mutations, insertion-deletion (indel) mutations, and multi-site mutations are also included in the database.

1.2. Homepage

The MdrDB homepage introduces the database and the pipeline by which it was constructed. The homepage comprises three sections: (i) data search and download, (ii) about MdrDB, and (iii) statistics.



Mutation-induced drug resistance DataBase (MdrDB)

MdrDB is a database of information related to changes in protein-ligand affinity caused by mutations in protein structure. It brings together wild type protein-ligand complexes, mutant protein-ligand complexes, binding affinity changes upon mutation (ΔΔG), and biochemical features of complexes to advance our understanding of mutation-induced drug resistance, the development of combination therapies, and the discovery of novel chemicals.

Figure 1. The data search and download section of the MdrDB homepage.

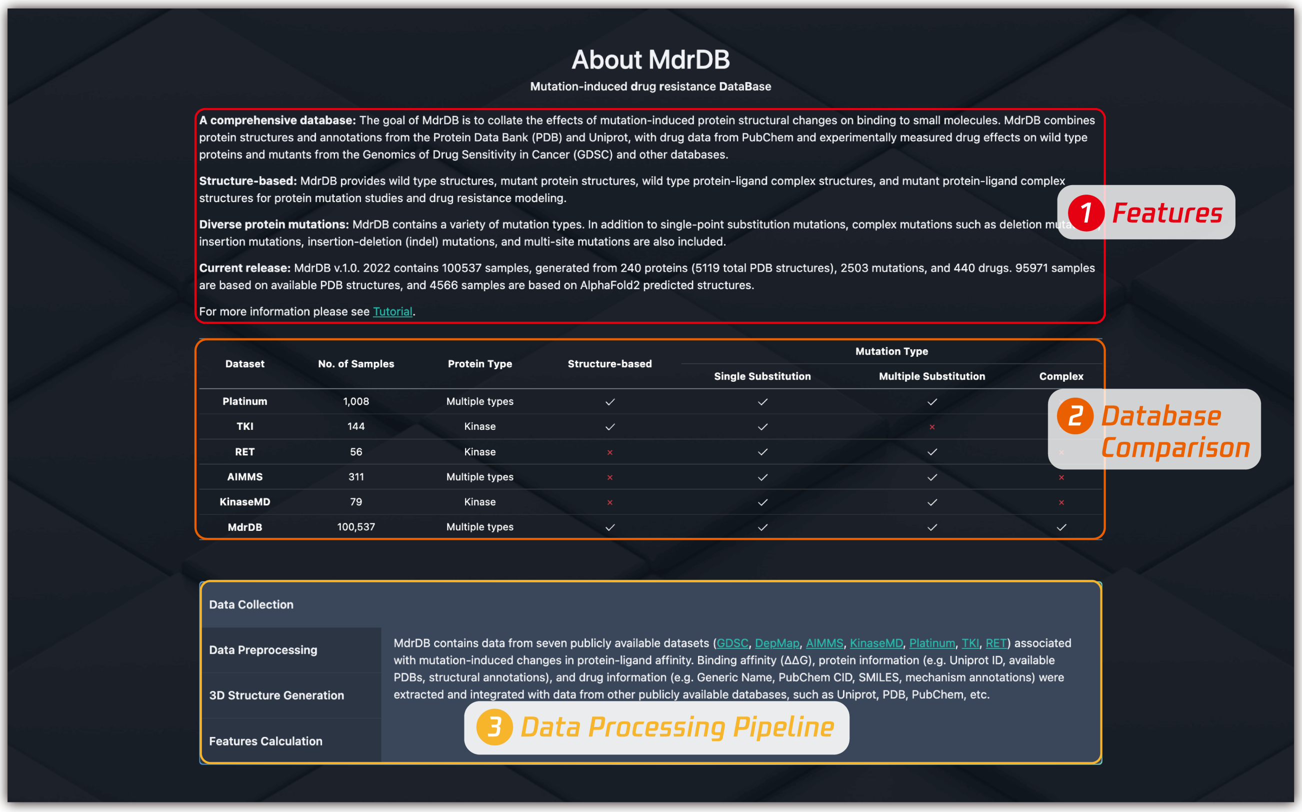
1.2.1. Data search and download

- Clicking the `Download MdrDB` button will navigate to the data download page. (**Fig. 1** ①)
- Users can enter `Uniprot ID`, `drug name` or `mutation information` in the input box for data retrieval. For more complex queries, see Section 2.2 on Advanced Search. (**Fig. 1** ②)
- Several search examples are shown below the search bar. Click `P98170`, `Dabrafenib`, or `L248R` to have a try! (**Fig. 1** ③)
- A flowchart below the search bar outlines the database preparation procedure. For more information see Section 1.2.2. (**Fig. 1** ④)

1.2.2. About MdrDB

This section introduces the key features of MdrDB and gives details on the data preparation pipeline.

- MdrDB is a `comprehensive`, `structure-based` database, and contains `diverse protein mutations`. (**Fig. 2** ①)
- The current release is **MdrDB v.1.0.2022** (https://quantum.tencent.com/MdrDB/).
- A comparison of MdrDB to various other drug resistance related databases is given. (**Fig. 2** ②)
- The data pipeline consists of four steps: `data collection`, `data preprocessing`, `3D structure generation`, and `features calculation`. For more details, see Section 5. (**Fig. 2** ③)



About MdrDB

Mutation-induced drug resistance DataBase

A comprehensive database: The goal of MdrDB is to collate the effects of mutation-induced protein structural changes on binding to small molecules. MdrDB combines protein structures and annotations from the Protein Data Bank (PDB) and Uniprot, with drug data from PubChem and experimentally measured drug effects on wild type proteins and mutants from the Genomics of Drug Sensitivity in Cancer (GDSC) and other databases.

Structure-based: MdrDB provides wild type structures, mutant protein structures, wild type protein-ligand complex structures, and mutant protein-ligand complex structures for protein mutation studies and drug resistance modeling.

Diverse protein mutations: MdrDB contains a variety of mutation types. In addition to single-point substitution mutations, complex mutations such as deletion mutations, insertion mutations, insertion-deletion (indel) mutations, and multi-site mutations are also included.

Current release: MdrDB v.1.0. 2022 contains 100537 samples, generated from 240 proteins (5119 total PDB structures), 2503 mutations, and 440 drugs. 95971 samples are based on available PDB structures, and 4566 samples are based on AlphaFold2 predicted structures.

For more information please see Tutorial.

Dataset	No. of Samples	Protein Type	Structure-based	Mutation Type		
				Single Substitution	Multiple Substitution	Complex
Platinum	1,008	Multiple types	✓	✓	✓	×
TKI	144	Kinase	✓	✓	×	×
RET	56	Kinase	×	✓	✓	×
AIMMS	311	Multiple types	×	✓	✓	×
KinaseMD	79	Kinase	×	✓	✓	×
MdrDB	100,537	Multiple types	✓	✓	✓	✓

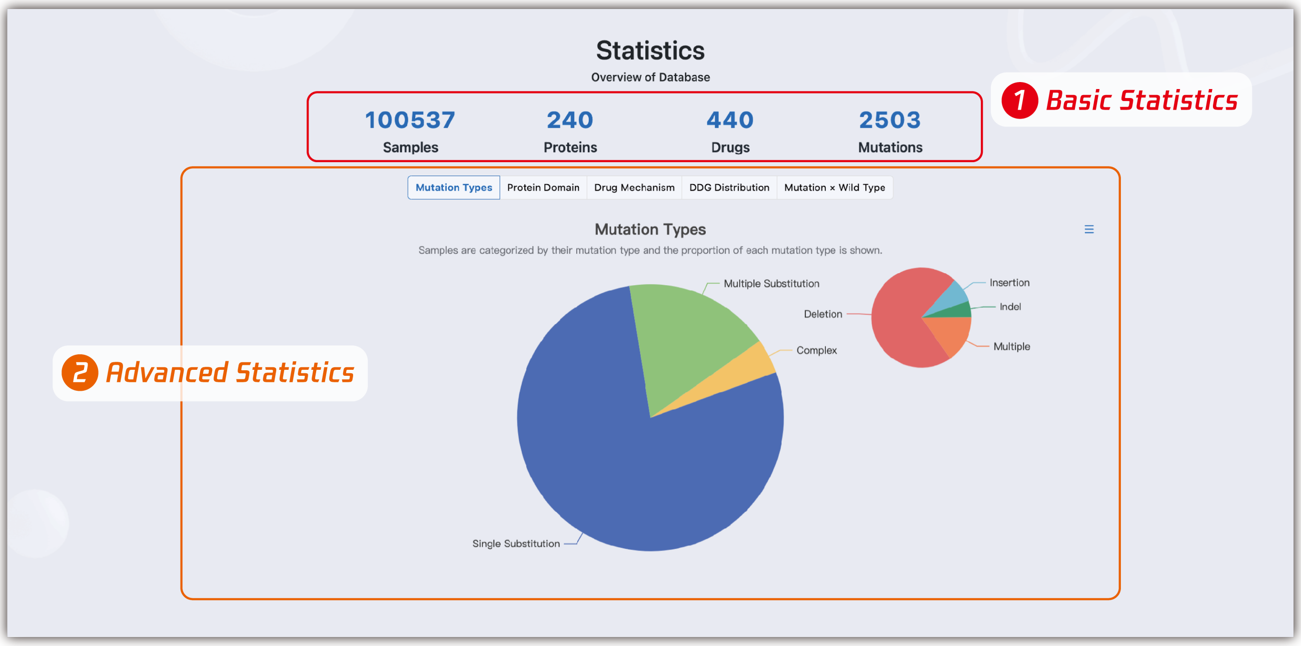
MdrDB contains data from seven publicly available datasets (GDSC, DepMap, AIMMS, KinaseMD, Platinum, TKI, RET) associated with mutation-induced changes in protein-ligand affinity. Binding affinity (ΔΔG), protein information (e.g. Uniprot ID, available PDBs, structural annotations), and drug information (e.g. Generic Name, PubChem CID, SMILES, mechanism annotations) were extracted and integrated with data from other publicly available databases, such as Uniprot, PDB, PubChem, etc.

Figure 2. The About MdrDB section of the homepage.

1.2.3. Statistics

This section of the hompage gives a number of statistics related to MdrDB.

- Basic Statistics are listed at the top of this section. (**Fig. 3** ①)
- More detailed statistics can be obtained in the Advanced Statistics section. (**Fig. 3** ②)
    - Click `Mutation Type` to view pie charts corresponding to the various mutation types in MdrDB. The smaller pie chart on the right hand side gives a breakdown of the `Complex` mutation types shown in the main pie chart on the left.
    - Click `Protein Domain` to view a bar chart of proteins in MdrDB, grouped by protein domain. The x-axis shows different protein domains, and the y-axis is the corresponding sample count on a logarithmic scale.
    - Click `Drug Mechanism` to view a bar chart of drugs in MdrDB grouped by pharmacological mechanism. The x-axis shows different drug mechanisms, and the y-axis is the corresponding sample count on a logarithmic scale.
    - Click `DDG Distribution` to view a histogram of protein mutation-induced ligand binding affinity changes measured as △△G (kcal/mol). The dashed vertical line indicates a threshold △△G value of 1.36 kcal/mol, which corresponds to a 10-fold decrease in drug binding affinity to the mutant protein.
    - Click `Mutation x Wild Type` to view a heatmap of amino acid changes upon substitution mutation in MdrDB. The y-axis corresponds to residues in wild type proteins and the x-axis corresponds to residues in mutant proteins. The sample counts -- grouped by residue types -- are shown besides the axes. Two double-ringed pie charts additionally show the ratio of each residue type in wild type and mutant proteins. The inner ring shows the 20 natural amino acids, and the outer ring groups these amino acids into five types according to the physical and chemical properties of their side chains.



Statistics

Overview of Database

100537 Samples

240 Proteins

440 Drugs

2503 Mutations

Mutation Types

Samples are categorized by their mutation type and the proportion of each mutation type is shown.

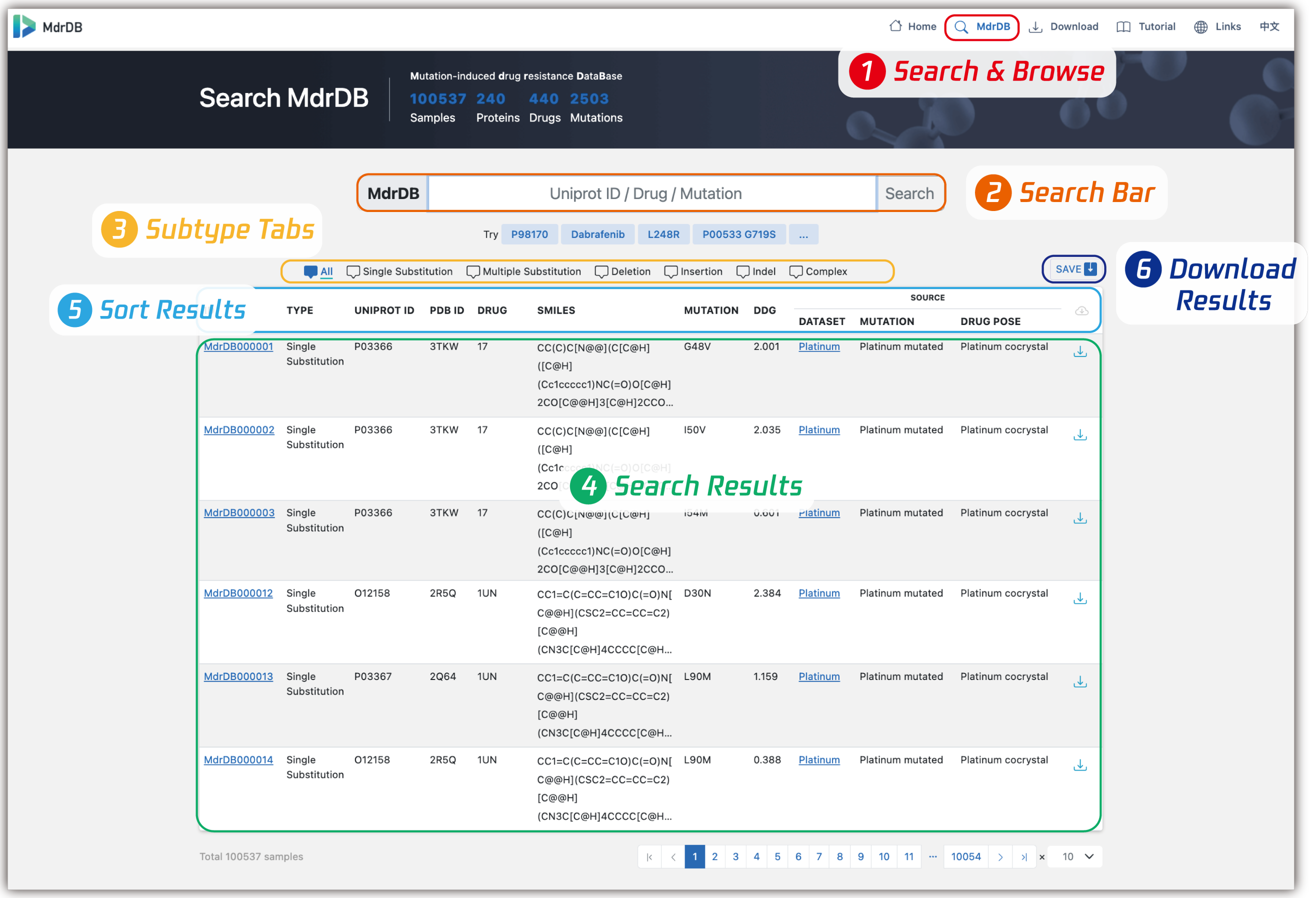
Figure 3. The statistics section of the MdrDB homepage.

2. Browse and Search

2.1. Browse

Clicking the `MdrDB` button in the navigation bar will take you to the **search and browse page**. (**Fig. 4** ①)

- Users can directly browse all samples in MdrDB. Mutations of a given type (e.g. single substitution, multiple substitution, deletion etc...) can be filtered by clicking on the corresponding tab. (**Fig. 4** ③)
- Users can also search samples by keyword using the search bar. (**Fig. 4** ②)
- The search/browse results give the following basic information for each sample: sample ID, UniProt ID, PDB ID, mutation string, drug name, drug SMILES and △△G value. For more detailed information, click on the corresponding line to navigate to the **sample display page**. (**Fig. 4** ④)
- The search/browse results can be sorted with values by clicking the corresponding headers on the table (**Fig. 4** ⑤) and downloaded by clicking the button on the upper right of the table. (**Fig. 4** ⑥)



	TYPE	UNIPROT ID	PDB ID	DRUG	SMILES	MUTATION	DDG	SOURCE		
								DATASET	MUTATION	DRUG POSE
MdrDB000001	Single Substitution	P03366	3TKW	17	CC(C)C[N@@](C[C@H] ([C@H] (Cc1ccccc1)NC(=O)O[C@H] 2CO[C@@H]3[C@H]2CCO...	G48V	2.001	Platinum	Platinum mutated	Platinum cocrystal
MdrDB000002	Single Substitution	P03366	3TKW	17	CC(C)C[N@@](C[C@H] ([C@H] (Cc1ccccc1)NC(=O)O[C@H] 2CO	I50V	2.035	Platinum	Platinum mutated	Platinum cocrystal
MdrDB000003	Single Substitution	P03366	3TKW	17	CC(C)C[N@@](C[C@H] ([C@H] (Cc1ccccc1)NC(=O)O[C@H] 2CO[C@@H]3[C@H]2CCO...	I54M	0.601	Platinum	Platinum mutated	Platinum cocrystal
MdrDB000012	Single Substitution	O12158	2R5Q	1UN	CC1=C(C=CC=C1O)C(=O)N[C@@H](CSC2=CC=CC=C2) [C@@H] (CN3C[C@H]4CCCC[C@H...	D30N	2.384	Platinum	Platinum mutated	Platinum cocrystal
MdrDB000013	Single Substitution	P03367	2Q64	1UN	CC1=C(C=CC=C1O)C(=O)N[C@@H](CSC2=CC=CC=C2) [C@@H] (CN3C[C@H]4CCCC[C@H...	L90M	1.159	Platinum	Platinum mutated	Platinum cocrystal
MdrDB000014	Single Substitution	O12158	2R5Q	1UN	CC1=C(C=CC=C1O)C(=O)N[C@@H](CSC2=CC=CC=C2) [C@@H] (CN3C[C@H]4CCCC[C@H...	L90M	0.388	Platinum	Platinum mutated	Platinum cocrystal

Total 100537 samples

Figure 4. The browse and search page of the MdrDB website.

2.2. Sample display

On the **sample display page**, users can view detailed sample information, with information grouped into 10 blocks.

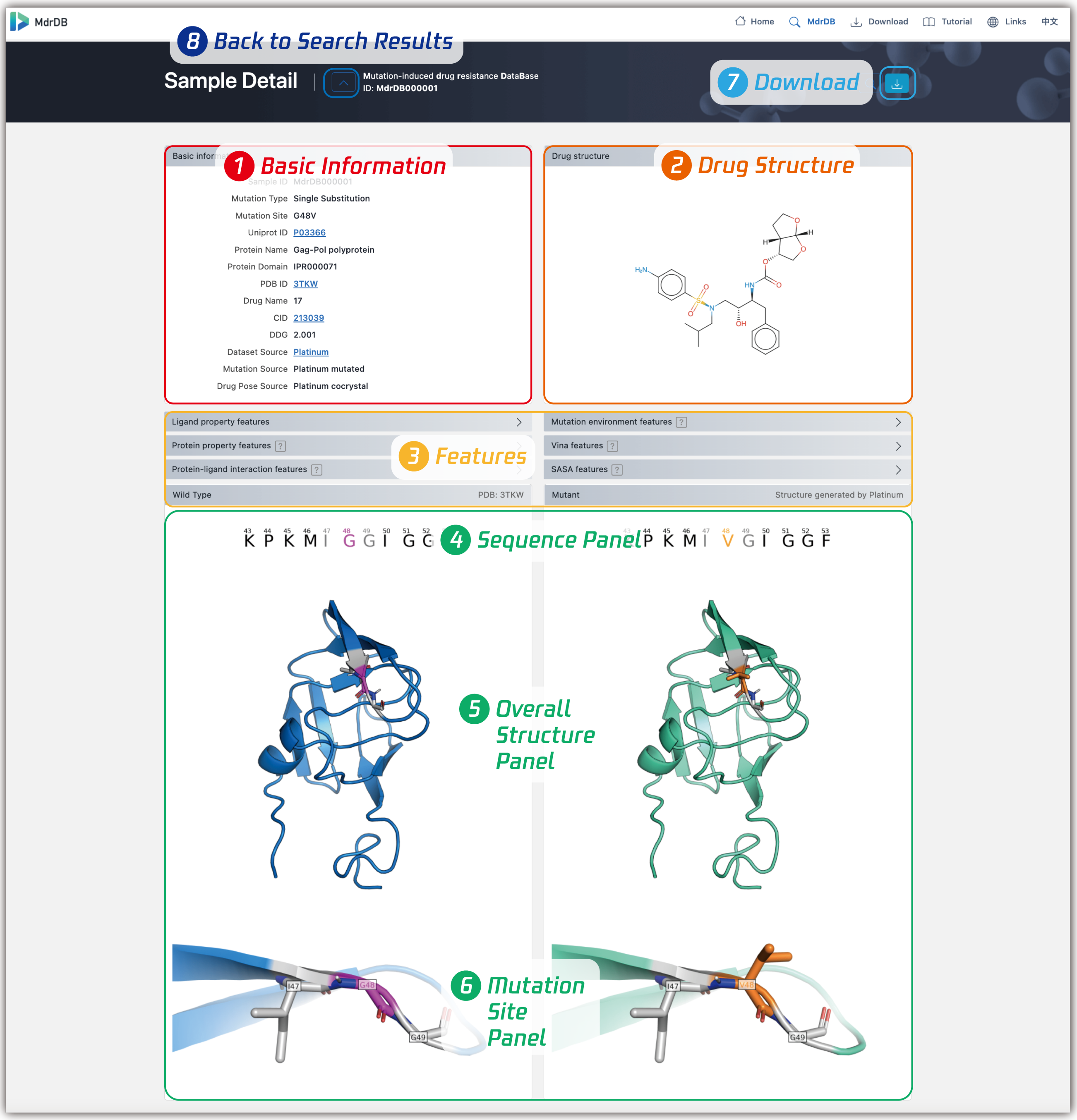
- The `Basic information` block displays the information shown in the **search and browse page**. In addition, links to **UniProt** (`Uniprot ID`), **RCSB PDB** (`PDB ID`), and **PubChem** (`CID`) are given, which navigate to the corresponding pages in these databases. (**Fig. 5** ①)
- The `Drug structure` block shows the 2D structure of the drug. (**Fig. 5** ②)
- There are six blocks which correspond to different features. In total, 146 features are given: (i) 18 features that reflect ligand properties, (ii) 12 features that represent the wild type and mutant protein differences, (iii) 21 features that describe mutation environments, (iv) 6 features that model protein-ligand interactions, (v) 59 features of VINA energy functions and (vi) 30 features related to solvent accessibility of both ligand and protein. Click each block heading to view details of their corresponding features. Except for the ligand property features, all values (X) that appear in the other five blocks correspond to the difference between the values for the mutant complex (X_{mt}) and the wild type complex (X_{wt})^2. (**Fig. 5** ③)

$$X = X_{mt} - X_{wt}$$

- The `Wild type` and `Mutation` blocks show the sequence, overall protein structure and mutation site panels. Mutation site residues are colored **magenta** in wild type proteins, and **orange** in mutant proteins. The two residues that are next to the mutation site are colored **grey**.
  - In the sequence panel, residue sequences and indices are shown. (**Fig. 5** ④)
  - In the overall protein structure panel, proteins are oriented to show all residues and to give a global view of where the mutations are located. (**Fig. 5** ⑤)
  - In the mutation site panel, the residues near mutation sites are oriented to show details of the mutation site before and after mutation. (**Fig. 5** ⑥)

  In addition, the source of the structures are documented. For the wild type protein, a `PDB ID` or `AlphaFold2` is shown. For the mutant protein, a label is assigned in `Platinum mutated`, `TKI mutated`, `Pymol` or `AlphaFold2`. For more information, please check Section 5.4.

- Users can download structure and feature files for each sample by clicking the download button on the upper right of the page (**Fig. 5** ⑦). The downloaded folder contains 6 files:
  - drug_3d.sdf
  - protein_wt.pdb
  - protein_mt.pdb
  - wt_complex.pdb
  - mt_complex.pdb
  - feature.tsv
- Users can go back to the search result page by clicking the icon near the top of the page, just to the right of "Sample Detail". (**Fig. 5** ⑧)



Sample Detail

Mutation-induced drug resistance DataBase
ID: MdrDB000001

Basic information

Sample ID MdrDB000001
Mutation Type Single Substitution
Mutation Site G48V
Uniprot ID P03366
Protein Name Gag-Pol polyprotein
Protein Domain IPR000071
PDB ID 3TKW
Drug Name 17
CID 213039
DDG 2.001
Dataset Source Platinum
Mutation Source Platinum mutated
Drug Pose Source Platinum cocrystal

Drug structure

Ligand property features
Protein property features
Protein-ligand interaction features
Mutation environment features
Vina features
SASA features

Wild Type PDB: 3TKW
Mutant Structure generated by Platinum

Figure 5. The sample display page of the MdrDB website.

2.3. Basic search

Basic search can be performed via the search bar on the **homepage** or **browse and search page**. `MdrDB ID`, `Uniprot ID`, `Mutation string` and `drug name` can be searched directly. Alternative names of drugs which are documented in PubChem can also be used for searching a drug.

Search Content	Keywords	Case Sensitive	Example
MdrDB ID	ID	No	mdrd000001, MdrDB087681
Protein	UNIPROT_ID	No	P00533, p15056, O12158
Mutation string (see Section 4)	MUTATION	Yes	Single substitution: I50V Deletion: N486_P490delNVTAP Insertion: T599_V600insT Indel: V487_P492delinsA Multiple: E88G+N92L A750P+L747_E749delLRE
Drug names (including alternative names)	DRUG	No	dabrafenib, 1195765-45-7, GSK2118436A, CHEBI:75045, UNII-QGP4HA4G1B

2.4. Advanced search

MdrDB supports several types of customized search.

2.4.1. Search with wildcards

The `*` wildcard symbol can be used in queries to indicate unspecified characters or values. MdrDB currently supports two kinds of wildcard search:

- `(*)string(*)`: at the beginning and/or at the end of a string.
- `string(*)string`: one `*` in the middle of a string.

For more examples, see Section 2.4.2.

2.4.2. Advanced keywords

In addition to the keywords given in Section 2.3, a number of **advanced keywords** can also be used by prepending the search term with an appropriate prefix (see table below). The general format of a query using these advanced keywords thus takes the form `prefix:search_content`.

Search Content	Keywords	Case Sensitivity	Prefix	Example
Mutation types[1]	TYPE	No	**T:**	T:*substitution , T:deletion
PDB	PDB_ID	No	**P:**	P:*G9* , P:4G9R
SMILES string	SMILES	Yes	**S:**	S:*ccccc* , S:COC1=C(C=C(C=C1)OCCCCC(=O)O)CC2=CN=C(N=C2N)N
△△G	DDG	range[2]	**DDG:**	DDG:(-10, -6] , DDG:[5.5,6.3]
Source database	SAMPLE_SOURCE	No	**SD:/DS:**	DS:platinum, DS:GDSC[3]
Mutation generation method	MUTATION_SOURCE	No	**SM:/MS:**	MS:pymol*,MS:alphafold2[4]
Drug generation method	DRUG_POSE_SOURCE	No	**SDP:/DPS:/DP:**	DPS:tki*,DPS:Docked[5]

[1] *For a specific type, users can also check the corresponding tabs shown in Fig. 4 ③.*

[2] *DDG is a continuous variable. The search values supported here are a range of values.* `()` *and* `[]` *are used to represent the value range. For example, if the searched value is* `(a,b)`*, the returned results would be values* `a < i < b`*. If the searched value is* `[a,b]`*, the returned results would be values* `a ≤ i ≤ b`*. If the searched value is* `(a,b]`*, the returned results would be values* `a < i ≤ b`*.*

[3] *The supported search words would be:* `platinum`*,* `tki`*,* `gdsc`*,* `ret`*,* `aimms`*,* `depmap`*,* `kinasemd`*.*

[4] *The supported search words would be:* `pymol__mutagenesis__wizard`*,* `alphafold2`*,* `platinum__mutated`*,* `tki__mutated`*.*

[5] *The supported search words would be:* `platinum__cocrystal`*,* `tki__cocrystal`*,* `tki__docked`*,* `docked`*.*

Note that the basic keywords of Section 2.3 can also be written in a similar format by specifying a prefix, although this is optional.

Search Content	Keywords	Case Sensitivity	Prefix	Example
MdrDB ID	ID	No	**I:**	I:MdrDB087681, I:I:MdrDB00000*
Protein	UNIPROT_ID	No	**U:**	U:P00533, U:P*
Mutation string	MUTATION	Yes	**M:**	M:V316A, M:I*, M:*A, M:A*G
Drug names[1]	DRUG	No	**D:**	D:*nib, D:*-*, D:GSK*

[1] *Alternative drug names are not yet supported with wildcard search.*

2.4.3. Multi-keyword search

Multiple keywords can be searched at the same time by separating keywords with a `space`. When using multi-keyword search with wildcards, we suggest explicitly specifying the prefixes for all keywords. Some examples:

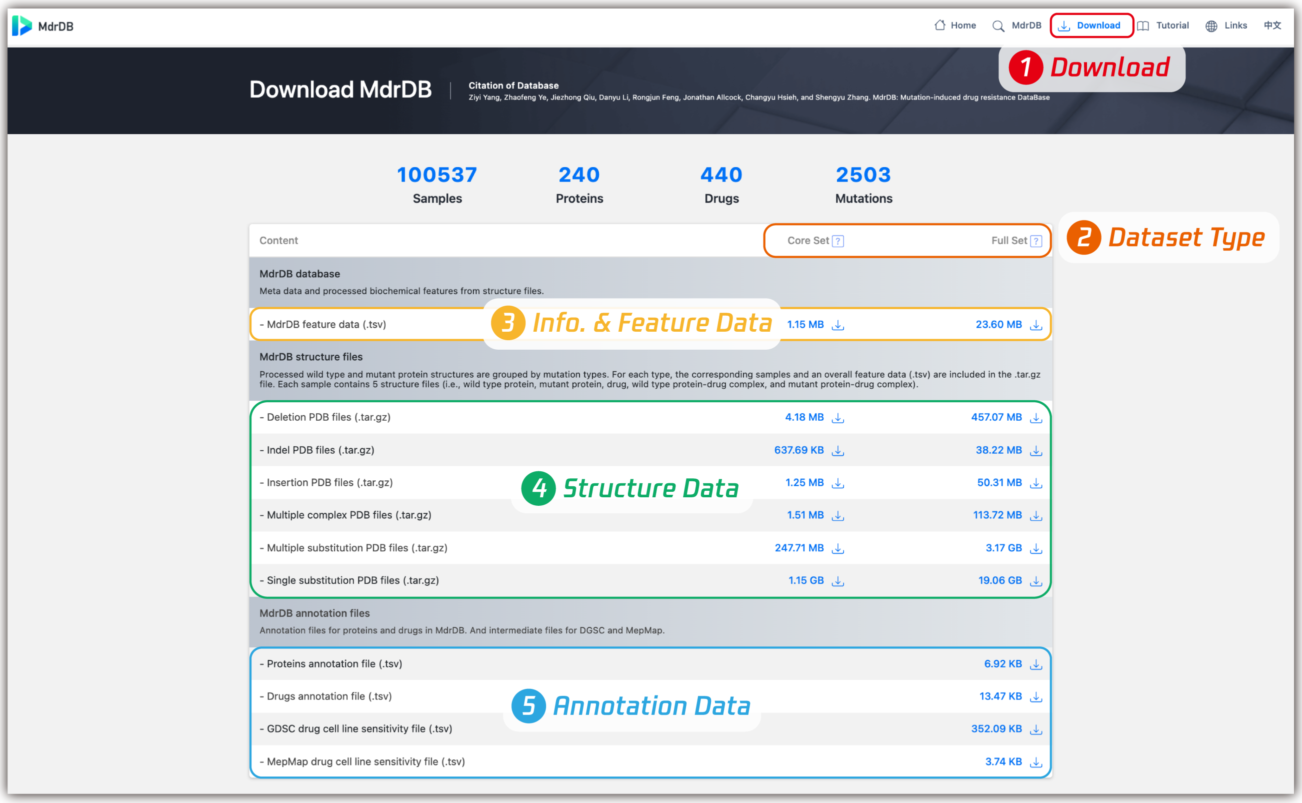
- P98170 R443C
- U:P98170 P:5* T:deletion
- D:*nib S:*CC=CC=C*

3. Download

Click the `Download` button in the navigation bar to enter the **download page**.(**Fig. 6** ①)

- MdrDB provides two dataset types to users for downloading: (**Fig. 6** ②)
  - `MdrDB_CoreSet`: Non-repetitive **'uniprot-mutation-drug'** samples whose features are averaged over all corresponding PDB features.
  - `MdrDB_FullSet`: All **'uniprot-pdb-mutation-drug'** samples, whose features are calculated based on each PDB structure.
- `MdrDB database` block: basic information, meta data and biochemical features of each sample in .tsv format. (**Fig. 6** ③)

- `MdrDB structure files` block: the processed structure files of MdrDB. Structure files are grouped by mutation type. For each type, the corresponding samples and an overall table (.tsv) are included in the .tar.gz file. (**Fig. 6** ④) Each individual sample contains five structure files, which can also be downloaded directly from the Sample Detail page (**Fig. 5** ⑦).
- `MdrDB annotation files` block: the annotation files (.tsv) in MdrDB. (**Fig. 6** ⑤)



Download MdrDB | Citation of Database
Ziyi Yang, Zhaofeng Ye, Jiezhong Qiu, Danyu Li, Rongjun Feng, Jonathan Allcock, Changyu Hsieh, and Shengyu Zhang. MdrDB: Mutation-induced drug resistance DataBase

100537	240	440	2503
Samples	Proteins	Drugs	Mutations

Content	Core Set ?	Full Set ?
MdrDB database Meta data and processed biochemical features from structure files.		
- MdrDB feature data (.tsv)	1.15 MB	23.60 MB
MdrDB structure files Processed wild type and mutant protein structures are grouped by mutation types. For each type, the corresponding samples and an overall feature data (.tsv) are included in the .tar.gz file. Each sample contains 5 structure files (i.e., wild type protein, mutant protein, drug, wild type protein-drug complex, and mutant protein-drug complex).		
- Deletion PDB files (.tar.gz)	4.18 MB	457.07 MB
- Indel PDB files (.tar.gz)	637.69 KB	38.22 MB
- Insertion PDB files (.tar.gz)	1.25 MB	50.31 MB
- Multiple complex PDB files (.tar.gz)	1.51 MB	113.72 MB
- Multiple substitution PDB files (.tar.gz)	247.71 MB	3.17 GB
- Single substitution PDB files (.tar.gz)	1.15 GB	19.06 GB
MdrDB annotation files Annotation files for proteins and drugs in MdrDB. And intermediate files for DGSC and MepMap.		
- Proteins annotation file (.tsv)		6.92 KB
- Drugs annotation file (.tsv)		13.47 KB
- GDSC drug cell line sensitivity file (.tsv)		352.09 KB
- MepMap drug cell line sensitivity file (.tsv)		3.74 KB

Figure 6. The MdrDB download page.

4. Protein mutation grammar

4.1. Mutation types

There are six types of mutation defined in MdrDB:

- `Single substitution`: a single amino acid is replaced with a different amino acid.
- `Mutiple substitution`: several single substitutions occuring at different locations.
- `Deletion`: one or more amino acids are deleted from the original sequence.
- `Insertion`: one or more amino acids are inserted into the original sequence.
- `Indel`: one or more amino acids are deleted from the original sequence (if multiple amino acids are deleted they must form a contiguous sequence), and one or more new amino acids are added at the deletion site.
- `Complex`: single or multiple substitutions with additional insertions, deletions and indels.

For more information on protein and nucleic acid mutations see http://atlasgeneticsoncology.org/Educ/NomMutID30067ES.html and https://varnomen.hgvs.org/recommendations/protein/variant/substitution/

4.2 General grammar

Searching using the MUTATION keyword (see Section 2.3 requires mutations to be specified according to a particular grammar. The associated patterns are listed in the table below, along with examples, for each type of mutation. The placeholders [aa] refers to 'amino acid', while [resi] refers to 'residue index' number.

Mutation Type	Pattern	Example	Explanation
Single substitution	**[aa][resi][aa]**	P252R	at position 252, P replaced by R
Multiple substitution	**[single_sub_1]+ [single_sub_2]+...**	L11T+E56G	at position 11, L replaced by T; at position 56, E replaced by G.
Deletion	**[aa][resi]del[aa]**	K15delK	at position 15, K is deleted
Deletion	**[aa_1][resi_1]_[aa_2] [resi_2]del[aa_seq]**	R84_L86delRLL	from R at position 84 to L at position 86, the sequence RLL is deleted
Insertion	**[aa_1][resi_1]_[aa_2] [resi_2]ins[aa_seq]**	R84_L85insAA	between R at position 84 and L at position 85, the sequence AA is inserted
Indel	**[aa][resi]delins[aa_seq]**	V97delinsAWS	V at position 97 is deleted, a new sequence AWS is inserted
Indel	**[aa_1][resi_1]_[aa_2] [resi_2]delins[aa_seq]**	V97_Q99delinsAWS	from V at position 97 to Q at position 99, the original sequence is deleted, a new sequence AWS is inserted

Complex	**[mut_1]+[mut_2]+...**	L11T+E56G+R84_L86delRLL	at position 11, L replaced by T;
			at position 56, E replaced by G.
			from postion 84 to position 86, the sequence RLL is deleted

4.3. Some common mistakes

When searching for mutations using the protein mutation grammar, take care to avoid the following common mistakes.

- **Overlapping**: one position in the protein sequence should not be edited multiple times. (e.g. L85G+R84_L86delRLL)
- **AA Mismatch**: the amino acid does not match the residue in the protein sequence at the specified position.
- **AA Type**: the letter representing the amino acid does not correspond to one of the 20 natural amino acids. (e.g. L85X)
- **Wrong Pattern**: the mutation string does not match the grammar for a specific type of mutation. (e.g. R84L86delRLL)
- **Sequence Mismatch**: in deletion, the deleted sequence should match the length and residues in the original protein sequence. (e.g. R84_L86delRL, R84_L86delPTL)
- **Wrong Ordering**: the two amino acids should be ordered from smallest residue index to largest residue index. (e.g. R84_S83delRS, R84_S83insAA, V97_L96delinsWWW)

5. Methods and other information

The data gathering and processing procedure used to generate MdrDB is outlined in this section. MdrDB collates data from seven publicly available sources: **GDSC**[1], **DepMap**[3], **AIMMS**[4], **KinaseMD**[5], **Platinum**[6], **TKI**[7] and **RET**[8]. An overall flowchart can be found in **Fig. 7**. While the other five datasets contain △△G and mutation information, GDSC and DepMap do not and must therefore first undergo an additional processing step.

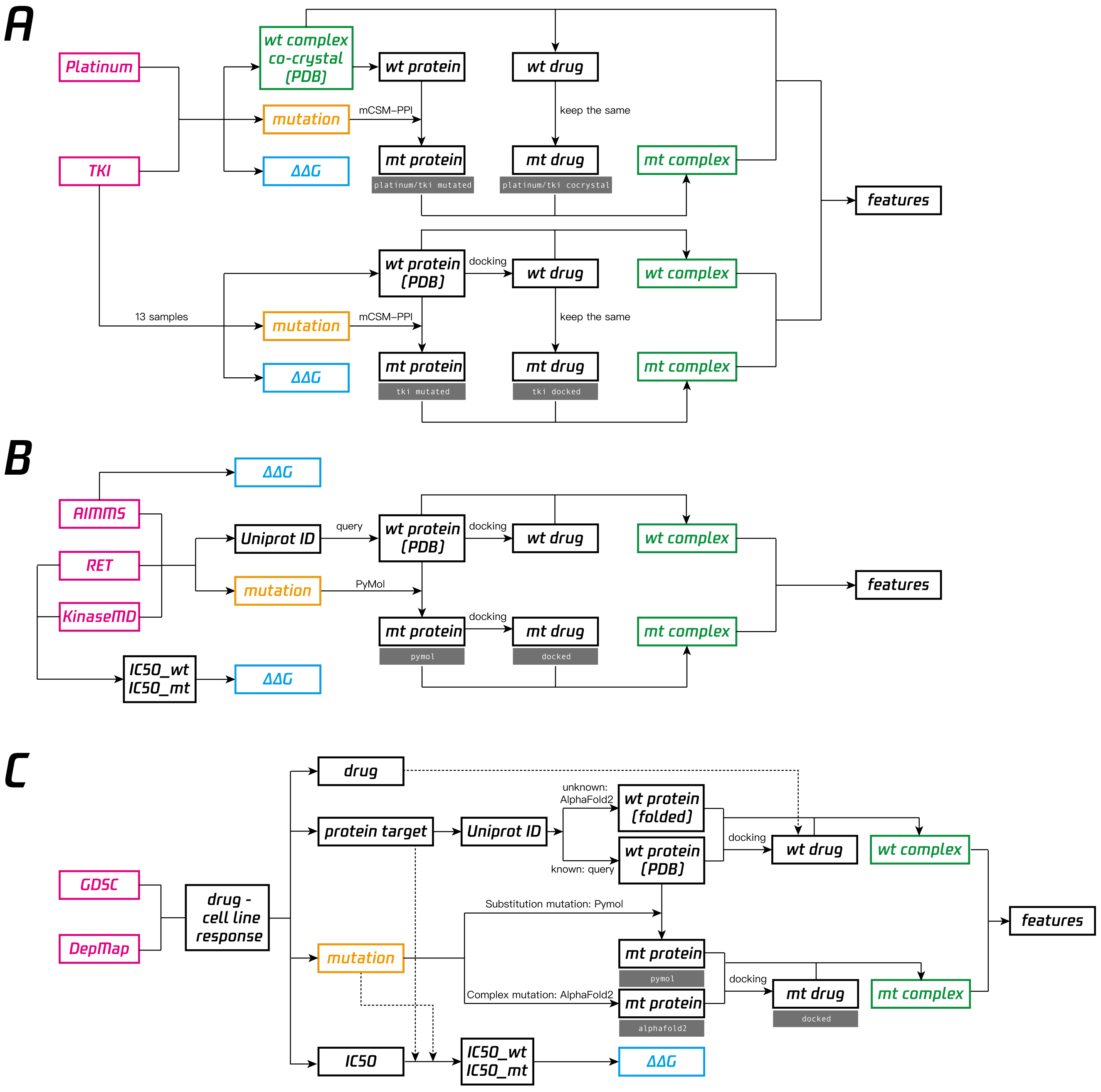


Figure 7. The data processing flowchart for the MdrDB database.

5.1. GDSC/DepMap raw data processing

In this step, information on **mutations on proteins in cell lines** and **cell line responses (IC50) to specific drugs** were integrated to generate a table containing drug affinities for wild type protein-drug and mutant protein-drug complexes. These affinities were then used to calculate the △△G.

Here we take GDSC2020 as an example to explain how we processed the data. All GDSC2020 data was download from https://www.cancerrxgene.org/.

5.1.1. Mutation information

- Protein mutation information was gathered for all cell lines. For a specific protein, cell lines that did not have mutations on them were considered to be control (wild type samples), while cell lines with mutations were considered to be mutant.
- Mutations were grouped by type (substitution, deletion, etc.) and mutations that contain a stop codon (*) were filtered out.
- A dictionary was built up, where the key is (protein, cell_line), and the value is the mutation.

5.1.2. Drug cell line response information

- A dictionary was built up, where the key is (protein, cell_line, drug), and the value is the IC50. For drugs with multiple known target proteins, each protein was considered individually. If, in the cell line, only one of these proteins is mutated, the sample was kept. Otherwise, the sample was skipped.
- For each mutant sample (protein, cell_line, drug), the corresponding mutation string was generated by merging all mutations with '+', IC50s were averaged, and a wild type sample was assigned.

5.1.3. Preparation of final data

- The △△G for each sample was calculated using the wild type and mutant IC50 according to the formula[9]:

$$\Delta\Delta G = -k_B T \log(\frac{IC50(MT)}{IC50(WT)})$$

- The UniProt ID was identified for each sample according to the protein name.
- The samples for different mutation types (Section 4.1) were split into separate tables.
- The final header of the table is:

UNIPROT_ID	TARGET	MUTATION	DRUG	LN_WT	LN_MT	DDG

5.2. PDB file downloading

- For each UniProt ID, all associated PDBs were identified with the RCSB REST API (https://data.rcsb.org/redoc/index.html). The `.pdb` or `.cif` for 3D structures and `.fasta` for sequences were downloaded from RCSB PDB (https://www.rcsb.org/).
- The SMILES for all drugs were identified using the PubChem PUG REST API (https://pubchemdocs.ncbi.nlm.nih.gov/pug-rest). Several drugs that could not be directly identified via PubChem were manually checked and assigned.

5.3. Structure file preprocessing

- For each sample, the protein PDB files and drug files were prepared.
- For protein PDB files, all water, solvent, and ions were removed. Then, the protein chains and ligands were split into separate files. Each chain was annotated and only the chains corresponding to the protein were kept. If multiple chains exist for the protein, the longest one was kept. The largest ligand was kept for further docking box generation.
- Each mutation for a protein was checked against all available PDBs. If the mutation sites could be found in the PDB, the mutation and drug would be assigned to the PDB.
- For the drugs, ions and salts in the SMILES were removed and the structures were neutralized. Then the SMILES were rewritten into canonical format. The 3D structures were first generated using openbabel 3.1.1[10] with the `--gen3D` flag. Then, the non-polar hydrogens were added to the generated structures with the `--addpolarH` flag.

5.4. Mutant structure generation

- There are 4 ways in which mutant structures in MdrDB were obtained:
  - For Platinum and TKI samples, both Platinum and TKI provided the mutant strutures themselves by modifying the PDB structures with mCSM-PPI. In these cases we have listed their mutant structures directly. We have marked such samples as `Platinum mutated` and `TKI mutated` respectively.
  - For samples from other datasets, as previously mentioned, we obtained mutant structures using either Pymol or AlphaFold 2. We have marked such samples with the labels `Pymol` and `AlphaFold2` respectively.
- Two tools were used for mutant structure generation: pymol-open-source v2.5.0 (https://github.com/schrodinger/pymol-open-source)[11] and AlphaFold 2.0 (https://github.com/deepmind/alphafold)[12].
- The pymol Mutagenesis Wizard module (https://pymolwiki.org/index.php/Mutagenesis) makes a mutation by replacing a residue with a new amino acid type, samples several rotamers from the rotamer library and generates several non-clashing conformations. Then, the most likely rotamer is chosen as the mutated residue.
- For AlphaFold 2.0, we used the protein amino acid sequence as the input to predict the structures. A length threshold of 2000 was set for computing resource considerations. For msa searching, `reduced_db` was used. For inference, the `model_ptm` models were used. Five models were generated and the one with highest averaged plddt value was chosen as the predicted structure for further procedures.
- For proteins with known PDBs containing the mutation sites:
  - For single substitution and multiple substitution mutations, pymol was used for mutant protein generation.
  - For deletion, insertion, indel and complex mutations, AlphaFold 2 was used for mutant protein structure prediction. For fair comparison and feature calculation, post-processing was carried out to keep the residue numbers the same except at the mutated sites.
- For proteins with no known PDBs containing the mutation sites:
  - AlphaFold 2 was used for both wild type protein and mutant protein structure prediction. Structures with an average plddt larger than 70 for the whole structure were kept, which was a

confidence threshold for the predicted structures in AlphaFold 2. In addition, if a mutated site was located on a region that was poorly predicted, the sample was discarded.

- After post-processing, the mutant protein was aligned with the wild type protein. The alignment was carried out by only taking the backbone atoms into consideration. A file tree can be built with the files mentioned above for each sample (**Fig. 8**).

```
DATASET/
    UNIPROT_ID/
        PDB_ID/
            # wildtype protein
            {PDB_ID}_{chain_ID}_pro.pdb
            {PDB_ID}_{chain_ID}_pro.fasta
            # optional ligand
            {PDB_ID}_lig.sdf
            MUTATION_NAME/
                # mutation protein
                mutant.pdb
                COMPOUND_NAME/
                    # drug
                    drug.sdf
```

Figure 8. The processed file tree for structure files for each sample.

5.5. Molecular docking

- The drug poses were obtained in 4 ways:
    - For Platinum and most TKI samples, Platinum and TKI provided the drug poses for the wild type proteins from known cocrystal structures, and used the same poses for mutant proteins. For these samples we provided their drug poses directly. These are marked `Platinum cocrystal` and `TKI cocrystal` respectively.
    - For 13 of the TKI samples, they docked the drugs to wild type proteins and used the same poses for mutant proteins. Again, we used their drug poses directly. These are marked as `TKI docked`.
    - For samples from other datasets, we docked the drugs to both wild type and mutant proteins. These are marked `Docked`.
- The molecular docking is carried out using smina (https://sourceforge.net/projects/smina/)[13], with default docking parameters used:
    - If the wild type PDB contained a known in-pocket ligand, then `--autobox_ligand` was selected.
    - If no ligand was present, the whole protein was used to generate the docking box.
- After docking, the conformation with the best smina score (the first conformation) was kept.

5.6. Feature calculation

- For biochemical feature calculations, the procedures in ***'Predicting Kinase Inhibitor Resistance: Physics-Based and Data-Driven Approaches'*** [2] (https://pubs.acs.org/doi/full/10.1021/acscentsci.9b00590) were used.

5.7. Data annotation

- For the protein annotations, we used the Interpro API (https://interpro-documentation.readthedocs.io/en/latest/download.html#interpro-application-programming-interface-api) [14] to query the `protein family`, `homologous superfamily` and `domain` information.
- For the drug annotations, we used the PubChem PUG REST API (https://pubchemdocs.ncbi.nlm.nih.gov/pug-rest)[15] to query the `CID`, `Depositor-Supplied Synonyms`, `FDA machanism`, `MeSH` and `Drug Classes` information.

5.8. Data tracking

Due to differences between the source databases, the procedure for users to track back to an entry in the contributing databases varies:

- For Platinum and TKI samples, the original entry can be tracked by using the PDB ID, mutation and drug information. The Platinum database is shut down now and there is no database id for the entrys. Users can found the processed data in the supplementary files for Platinum and TKI in https://pubs.acs.org/doi/full/10.1021/acscentsci.9b00590.
- For RET and AIMMS samples, the data was attached with the paper in a table. The original entries could be tracked with Uniprot id, mutation and drug information.
- For KinaseMD, no IDs were provided. The original entries could be found using protein (Uniprot id), mutation in substructure, and drug information.
- For GDSC and DepMap samples, there are no direct corresponding entries. However, the drug sensitivity data could be queried using the drug names. In addition, we've added the intermediate table for these two datasets in the download files, which documented the IC50 values and cell line information that we used for ddG calculation. Users can use this to track back to the original IC50 in corresponding databases.

References

1. Yang, W., Soares, J., Greninger, P., Edelman, E.J., Lightfoot, H., Forbes, S., Bindal, N., Beare, D., Smith, J.A., Thompson, I.R. and Ramaswamy, S., 2012. Genomics of Drug Sensitivity in Cancer (GDSC): a resource for therapeutic biomarker discovery in cancer cells. *Nucleic acids research*, *41*(D1), pp.D955-D961.
2. Aldeghi, M., Gapsys, V. and de Groot, B.L., 2019. Predicting kinase inhibitor resistance: physics-based and data-driven approaches. *ACS central science*, *5*(8), pp.1468-1474.
3. Tsherniak, A., Vazquez, F., Montgomery, P.G., Weir, B.A., Kryukov, G., Cowley, G.S., Gill, S., Harrington, W.F., Pantel, S., Krill-Burger, J.M. and Meyers, R.M., 2017. Defining a cancer dependency map. *Cell*, *170*(3), pp.564-576.
4. Wu, F.X., Wang, F., Yang, J.F., Jiang, W., Wang, M.Y., Jia, C.Y., Hao, G.F. and Yang, G.F., 2020. AIMMS suite: a web server dedicated for prediction of drug resistance on protein mutation. *Briefings in bioinformatics*, *21*(1), pp.318-328.
5. Hu, R., Xu, H., Jia, P. and Zhao, Z., 2021. KinaseMD: kinase mutations and drug response database. *Nucleic Acids Research*, *49*(D1), pp.D552-D561.
6. Pires, D.E., Blundell, T.L. and Ascher, D.B., 2015. Platinum: a database of experimentally measured effects of mutations on structurally defined protein–ligand complexes. *Nucleic acids research*, *43*(D1), pp.D387-D391.
7. Hauser, K., Negron, C., Albanese, S.K., Ray, S., Steinbrecher, T., Abel, R., Chodera, J.D. and Wang, L., 2018. Predicting resistance of clinical Abl mutations to targeted kinase inhibitors using alchemical free-energy calculations. *Communications biology*, *1*(1), pp.1-14.
8. Liu, X., Shen, T., Mooers, B.H., Hilberg, F. and Wu, J., 2018. Drug resistance profiles of mutations in the RET kinase domain. *British journal of pharmacology*, *175*(17), pp.3504-3515.
9. Barlow, K.A., Ó Conchúir, S., Thompson, S., Suresh, P., Lucas, J.E., Heinonen, M. and Kortemme, T., 2018. Flex ddG: Rosetta ensemble-based estimation of changes in protein–protein binding affinity upon mutation. *The Journal of Physical Chemistry B*, *122*(21), pp.5389-5399.
10. O'Boyle, N.M., Banck, M., James, C.A., Morley, C., Vandermeersch, T. and Hutchison, G.R., 2011. Open Babel: An open chemical toolbox. *Journal of cheminformatics*, *3*(1), pp.1-14.
11. DeLano, W.L., 2002. Pymol: An open-source molecular graphics tool. *CCP4 Newsl. Protein Crystallogr*, *40*(1), pp.82-92.
12. Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A. and Bridgland, A., 2021. Highly accurate protein structure prediction with AlphaFold. *Nature*, *596*(7873), pp.583-589.
13. Koes, D.R., Baumgartner, M.P. and Camacho, C.J., 2013. Lessons learned in empirical scoring with smina from the CSAR 2011 benchmarking exercise. *Journal of chemical information and modeling*, *53*(8), pp.1893-1904.
14. Hunter, S., Apweiler, R., Attwood, T.K., Bairoch, A., Bateman, A., Binns, D., Bork, P., Das, U., Daugherty, L., Duquenne, L. and Finn, R.D., 2009. InterPro: the integrative protein signature database. *Nucleic acids research*, *37*(suppl_1), pp.D211-D215.
15. Kim, S., Chen, J., Cheng, T., Gindulyte, A., He, J., He, S., Li, Q., Shoemaker, B.A., Thiessen, P.A., Yu, B. and Zaslavsky, L., 2019. PubChem 2019 update: improved access to chemical data. *Nucleic acids research*, *47*(D1), pp.D1102-D1109.