

A Guide Through Oxford University's ABodyBuilder2 to Aid in Antibody Structure Prediction
Jason Langlais

Dr. Frédéric Chain
BIOL.4062 Bioinformatic Tools in Sequence Analysis
December 16, 2023

ABodyBuilder2:

<https://opig.stats.ox.ac.uk/webapps/sabdab-sabpred/sabpred/abodybuilder2/>

Datasets:

<https://useast.ensembl.org/biomart/martview/1a7f0df76abb584bc9f725acd96fda34>

<https://useast.ensembl.org/biomart/martview/1a7f0df76abb584bc9f725acd96fda34>

<https://useast.ensembl.org/biomart/martview/1a7f0df76abb584bc9f725acd96fda34>

Video:

<https://www.youtube.com/watch?v=9fA4be8rvIM>

Purpose:

The purpose of this tutorial is to show the ABodyBuilder2 online tool for predicting antibody structure of the variable region using an amino acid sequence. This guide will go step-by-step to show how to gather an amino acid sequence for the heavy and light chain of an antibody's variable region and to input these sequences into ABodyBuilder2. This manual will also show different aspects of ABodyBuilder2 that can aid in analysis of the predicted structure.

Background:

Antibody prediction is an essential process to help understand the adaptive immune response and potential therapeutics mediated through antibody use [1]. The diversity of antibody structure is achieved through genetic recombination and somatic hypermutations that produce a wide variety of antigen binding sites. These are produced in response to different pathogen antigens and additionally facilitate changes in the constant region of antibodies for different effector functions against a pathogen invasion [2].

The basic structure of an antibody is two heavy chains and two light chains held together by disulfide bonds which together form two different regions: the variable region, and more importantly the complementary determining regions (CDRs) that reside in the variable region, are responsible for binding to the target epitopes, while the constant region is responsible for binding to immune cells and pathogens to elicit an immune response or complement activation [3]. The heavy chain is made up of three segments encoded by human chromosome 14: variable (V), diversity (D), and junctional (J). The light chain is made up of the V and J segments encoded by chromosome 2 and 22 [4]. All these segments come together to make their respective chains, and then combine to make the full antibody.

The main tool used in this project for antibody structure prediction is the University of Oxford's SAbPred online tool [ABodyBuilder2](#). This tool uses the ANARCI (antigen receptor numbering and receptor classification) sequence numbering scheme, which assigns numbers to the antibody's heavy and light chain residues to compare it to other annotated amino acid sequences of known antibody structures [5]. This allows understanding of how the input heavy and light chain sequences may interact with one another using the known structures as examples [1]. ANARCI is then used in the online application ABodyBuilder2 to obtain predicted structures. This tool also uses ABangle, which predicts the orientation of the variable heavy and light chains to each other using the orientation of other known variable chains [6]. After using ANARCI to predict the structure based on amino acid chemistry and interactions, the program uses OpenMM, a biomolecular modeling toolkit that helps simulate the behavior and properties of molecules, to refine the model by removing stereochemical errors [7,8].

Tools and Databases:

The main tool used in this project is ABodyBuilder2. R will be used to obtain DNA sequences from the VDJ coding genes found on human chromosomes 2 and 14. Different transcripts have been randomly selected from each of the segments from Ensembl, and the DNA sequence of each transcript has been combined and translated into the amino acid sequences for the heavy and light chains seen below. These culminated sequences are what will be taken from R and submitted into ABodyBuilder2.

Heavy Chain Sequence:

LPLLIKTSRAQTLQLWEKSPSPRIPRSFHSVISTEHRGLTMEFGLSWVFLVAIIKGVQCQV
QLVESGGGLVKPGGSLRLSCAASGFTFSDYYMSWIRQAPGKGLEWVSYISSSSSYTNYA
DSVKGRFTISRDNANKSLYLQMNSLRAEDTAVYYCARDAFDIWGQGTMTVTVSSVWGP
CGLPHEQMPPGPLVPASSFTAAVGAGARGIPGRVVFACVTVLGVRRRDQADRPGIVGAT
VEMATI

Light Chain Sequence:

RVGKKYLQLSASAEELWGVCTMAWTPLLFLTLLHCTGSLSQLVLTQSPSASASLGASVK
LTCTLSSGHSSYAIAWHQQQPEKGPRYLMKLNSDGSHSKGDGIPDRFSGSSSGAERYLTI
SSLQSEDEADYYCQTWGTGIVFGGGTQLIIL

Manual:

Gathering Amino Acid Sequences of VDJ coding regions of the Heavy and Light Chains:

1. Start by opening [R](#) and installing and running the following packages:
 - a. The devtools package needs to be installed to obtain a previous version of the dbplyr package

```

7
8 ~ {r package library}
9 library("dplyr")
10 library("biomaRt")
11 library("Biostrings")
12 library("bioseq")
13 library("dbplyr")
14 ~
15
16 ~ {r}
17 #Needed for an earlier version of a package used to complete this
18 install.packages("devtools")
19 devtools::install_version("dbplyr", version = "2.3.4")
20 ~
21

```

2. Use the following script to set the dataset to the Ensembl *Homo sapiens* genome:

```

22 Establishes using Ensembl's homo sapien catalog
23 ```{r Database and Mart}
24 mart<-useMart("ensembl",dataset = "hsapiens_gene_ensembl")
25 useDataset(dataset="hsapiens_gene_ensembl",mart)
26 ^

```

3. Create dataframes for each of the VDJ coding segments using the getBM() command and combine them into one dataframe using the rbind() function:

```

28 Creating dataframes of VDJ segment coding genes
29 ```{r VDJ Dataframes}
30 DGenes<-getBM(mart,
31               attributes=c("ensembl_gene_id","chromosome_name","gene_biotype","transcript_length"),
32               filters = "biotype",
33               value="IG_D_gene")
34 VGenes<-getBM(mart,
35               attributes=c("ensembl_gene_id","chromosome_name","gene_biotype","transcript_length"),
36               filters = "biotype",
37               value="IG_V_gene")
38 JGenes<-getBM(mart,
39               attributes=c("ensembl_gene_id","chromosome_name","gene_biotype","transcript_length"),
40               filters = "biotype",
41               value="IG_J_gene")
42 VDJ<-rbind(DGenes,VGenes,JGenes)
43 ^

```

4. Extract the transcripts of different VDJ segments using the getSequence() function, obtaining VDJ transcripts from chromosome 14 for the heavy chain and V and J transcripts from chromosome 2 for the light chain. This will require going into the VDJ dataframe created in the last step and picking out different Ensembl Gene IDs for the corresponding VDJ segments:
 - a. This example gathers two short J and D segments for the heavy chain. to get a decent size sequence closer to the proper sequence size of an antibody

```

56 Creating variables of nucleotide sequences of VDJ coding genes for combination
57 ▾ ``{r Heavy Chain - VDJ genes from chromosome 14}
58 V1Hseq<-getSequence(id="ENSG00000282322",
59                      type="ensembl_gene_id",
60                      seqType="cdna",
61                      mart=mart)
62 J1Hseq<-getSequence(id="ENSG00000242887",
63                      type="ensembl_gene_id",
64                      seqType="cdna",
65                      mart=mart)
66 J2Hseq<-getSequence(id="ENSG00000242472",
67                      type="ensembl_gene_id",
68                      seqType="cdna",
69                      mart=mart)
70 D1Hseq<-getSequence(id="ENSG00000282323",
71                      type="ensembl_gene_id",
72                      seqType="cdna",
73                      mart=mart)
74 D2Hseq<-getSequence(id="ENSG00000282674",
75                      type="ensembl_gene_id",
76                      seqType="cdna",
77                      mart=mart)
78 ▴ ``
80 ▾ ``{r Light Chain - VJ genes from chromosome 22}
81 V1Lseq<-getSequence(id="ENSG00000211637",
82                      type="ensembl_gene_id",
83                      seqType="cdna",
84                      mart=mart)
85 J1Lseq<-getSequence(id="ENSG00000211680",
86                      type="ensembl_gene_id",|
87                      seqType="cdna",
88                      mart=mart)
89 ▴ ``

```

5. Obtain the DNA sequence from each of the selected sequences and translate them into an amino acid sequence:
 - a. The `dna()` command will gather the nucleotide sequence of the chosen transcripts and the `seq_translate()` command will translate it into an amino acid sequence

```

80 Getting the nucleotide sequence of each gene we've obtained and combining them into one sequence
81 `r Creating variable Region 1 (both heavy and light chain)}
82 dna(Seq_1=V1Hseq$cdna)
83 tv1H<-seq_translate(dna(Seq_1=V1Hseq$cdna))
84
85 dna(Seq_1=J1Hseq$cdna)
86 dna(seq_1=J2Hseq$cdna)
87 tJ1H<-seq_translate(dna(Seq_1=J1Hseq$cdna))
88 tJ2H<-seq_translate(dna(seq_1=J2Hseq$cdna))
89
90 dna(Seq_1=D1Hseq$cdna)
91 dna(Seq_1=D2Hseq$cdna)
92 tD1H<-seq_translate(dna(Seq_1=D1Hseq$cdna))
93 tD2H<-seq_translate(dna(Seq_1=D2Hseq$cdna))
94
95 dna(Seq_1=V1Lseq$cdna)
96 dna(Seq_1=J1Lseq$cdna)
97 tv1L<-seq_translate(dna(Seq_1=V1Lseq$cdna))
98 tJ1L<-seq_translate(dna(Seq_1=J1Lseq$cdna))
99 `r

```

6. Combine the translated amino acid sequences of the heavy chain and light chain using the cat() command so that there are two individual sequences for separate chains:
 - a. This command will output the amino acid sequence that can be copy and pasted into ABodyBuilder2 seen in a later step

```

100 Combining the translated sequences
101 `r Translated Heavy Chain}
102 cat(tv1H,tJ1H,tJ2H,tD1H,tD2H)
103 `r
104 `r Trasnlated Light Chahin}
105 cat(tv1L,tJ1L)
106 `r

```

Using ABodyBuilder2 to Create an Antibody Structure (Covered in Video)

1. Open a web browser to search for and open "SAbPred ABodyBuilder2"
 - b. ABodyBuilder is an older version of the tool, and will eventually no longer be available

2. Scroll down to the “Sequence submission form” section

> Sequence submission form

Heavy chain sequence: ⓘ [load example](#)

Light chain sequence: ⓘ [load example](#)

Job name: ⓘ
my_tv_model

☒ IMGT ☐ Kabat ☐ Chothia ☐ Martin

Model

3. Here, you can input amino acid sequences for the heavy chain and light chain sequences
4. Copy and paste the heavy and light chain amino acid sequences generated in R into their respective box
 - a. Asterisks in the sequence represent stop codons and should be deleted.
5. Under the sequence submission boxes the name of the file can be changed in the “Job name” box. Use the name ‘Example_Ab’
6. Under the “Job name” box are the four different CDR labelling schemes, make sure the IMGT option is selected as this labels CDR regions based on other vertebrate antibodies
7. Click the pink “Model” button towards the bottom of the webpage

> Sequence submission form

Heavy chain sequence: ⓘ [load example](#)

LPLLIKTSRACTLQLWEKSPSPRIFSHSVISTEHRGLTMEFGLSWVFLVAIIKGVQCQVOLVESGGGLVKPGGSLRLSCAASGFTPSDYIMSWIRQAPGKGLEWVSYISSSSYTNYADSVKGRFTISRDNKNSLYLQMNSLRAEDTAVYYCARDAFDIWGQGTMTVSSVWGPGCLPHEQMPPGPLVPASSFTAAGAGARGIPGRVFACTVLGVRRRDQADRPQVIGATVEMATI

Light chain sequence: ⓘ [load example](#)

RVGKKYLOLSASAEIWGVCYMAWTFLLFTLLHCTGSLQLVLTQSPASASLGASVRLTCTLSGHSYAIAWHQQQPEKGRVYLMKLNDSGSHSKGDGIFDRFGSSSGAERYLISLQSEDEADYYCOTWGTGIVFGGTQIIL

Job name: ⓘ
Example_Ab

☒ IMGT ☐ Kabat ☐ Chothia ☐ Martin

Model

8. Once “Model” is selected, the “Job Status” will appear showing the run status of the learning model

9. Once the job has finished, under the “Job status” section is the “Results section” containing the downloadable Fv model, which allows for deeper chemical analysis such as the atoms making up a residue, XYZ coordinates, and temperature factors

> Job Status

Job ID: 20231215_0058940

Job name: Example_Ab

Job status: finished!

Log file:

```

ABodyBuilder2
A Method for Antibody Structure Prediction
Author: Brennan Abanades Kenyon
Supervisor: Charlotte Deane

Sequences loaded successfully.
Heavy and light chains are:
H: LPLIKTSRAQTLQWEKSPSPRIIPSHSVISTENHRLTHEGLSWFLVAIZIKGVQCQQLVESGGGLVKPGGSLRLSCAASGFTFSQYVSWMRQAPKGLDWVSYSISSSVTYADSVKGRFTISRDNAMKSLYLQMSLRAEDTAVVYCARDADFQWQGTHTVTYSVWGPCPLPHEQHPGPLVPASSFT
AVGAGARLIPSPFYVACVTLQVRMDQAPRIQVATVDATE
L: RVGKXYLQSAKSLWVCTMTAFTPLFLTLHCTSSLSQVLVTQSPSASASLGASIKLTCTLSGSHSYAIAHQQQPEKIDPRYLKLNKSDGSKSDGQDPDFSGSSSGAERYLTSSLSQSEDEADIVYCQWGTGIVFGGTQLIIL

Running sequences through deep learning model...
Antibody modelled successfully, starting refinement.
Extracted modelling details
  
```

> Results

[View Model and Annotations](#)

Downloads

Renumbered Fv model (PDB format):

[Download](#)

10. Under the downloadable file, there is a “Modelling Score” section which shows the overall prediction error of each chain and the individual prediction error for each CDR loop of the variable region. This prediction error is a level of certainty that the residues have been placed and oriented correctly, a lower score representing a better model
11. After modelling, click the “View Model and Annotations” button in the “Results” section to view the predicted 3D structure

> Results

[View Model and Annotations](#)

Downloads

Renumbered Fv model (PDB format):


[Download](#)

Modelling scores

Antibody region	Prediction error
Framework H-chain	0.29
CDR-H1	0.42
CDR-H2	0.28
CDR-H3	0.23
Framework L-chain	0.39
CDR-L1	0.39
CDR-L2	0.33
CDR-L3	0.29

12. The model is presented in different colors. After the model prediction is run, it presents the structure in three different colors: light blue for the variable heavy chain, pink for the variable light chain, and dark blue for the CDR region
13. Under the structure are different filtering options to obtain different views of the model
14. Ensure the CDR definition is IMGT. The display options view the model in different ways, cartoon is used for this because it shows the sequence direction through arrows on the model. There are several different annotation options that will change the colors of the model for different region identifications
 - a. Regions: shows the heavy chain, light chain, and CDR segments
 - b. Secondary Structure: shows the conformation of the amino acid secondary structure
 - c. Prediction error: shows the prediction error throughout the structure
 - d. Sequence liabilities: shows sequences that are known to cause structuring problems in antibody development. Crosses out liabilities show there is no concern for them in you model
 - e. Solvent exposed: which parts of the structure are internal or external
 - f. Domains: shows the variable heavy and light chain regions without the CDR regions

> Model Viewer



Please note the WebGL plugin needs to be enabled to use PV Viewer.

CDR definitions

- ☐ Kabat
- ☐ Chothia
- ☒ IMGT
- ☐ North/AAHs
- ☐ Contact

Display options

- ☐ Spacefill
- ☐ Wire
- ☐ Ball&stick
- ☒ Cartoon
- ☐ Spin on/off

Annotation options

- ☒ Regions
- ☐ Secondary structure
- ☐ Prediction error
- ☐ Sequence liabilities
- ☐ Solvent exposed
- ☐ Domains

VH

VL

IMGT CDRs

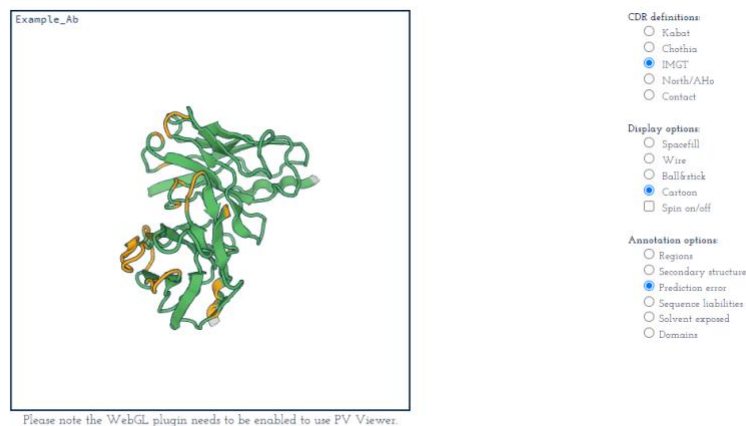
Your model should be shown above (we recommend using google-chrome to load this page; also have WebGL enabled). Alternatively, download your [model](#) (PDB file).

15. Under the region identification bar, click the hyperlink on the word “model” in blue to download a PDB file

a. This file can be used for structure analysis and shows:

- i. Row 1 – ATOM shows the order of atoms that make up the residue
- ii. Row 2 – where in the order of atoms is making up the residue (1 is the first atom in the whole sequence, 10 would be the tenth atom in the whole amino sequence)
- iii. Row 3 – What atom is in the numbered positions (N for nitrogen, H for hydrogen, C for carbon, and other letters and numbers show different conformations)
- iv. Row 4 – the three-letter abbreviation for the amino acid in the sequence
- v. Row 5 – chain identifier (in this case heavy or light)
- vi. Row 6 – the residue sequence number
- vii. Row 7-9 – the XYZ coordinates of the residue
- viii. Row 10 – occupancy (how many conformations there could be of the residue)
- ix. Row 11 – temperature factor (measure of different electron densities for the residue)
- x. Row 12 – element symbol of the atoms in the residue

16. The “Prediction Error” section shows the ANARCI labelled sequence highlighted by their prediction error, which can be edited using the threshold slider at the top of the section. The IMGT CDR regions are highlighted in blue.



Your model should be shown above (we recommend using google-chrome to load this page; also have WebGL enabled.). Alternatively, download your [model](#) (PDB file).

> Prediction Error

Thresholds:



Conclusion:

ABodyBuilder2 is an easy-to-use tool that allows for the creation of a predictive model of antibody structure. The analysis options allow the structure to be viewed in different modes to show things such as prediction error of each residue and the different sequence liabilities that could lead to misfunction or disassembly of the antibody. This tool also generates different CDR regions which can show potential changes in the structure between different types of organisms. Although this is a useful tool, there are many more out there that can be used to generate and analyze other aspects of antibodies, such as analyzing antibody-epitope binding to see what epitopes the antibody can bind to. Although it was not shown in this project, the PDB file can also be used to compare antibody structures and different liabilities and conformations in the structures.

Work Cited

1. *Fast, accurate antibody structure prediction from deep learning on massive set of natural antibodies* / *Nature Communications*. (n.d.). Retrieved November 1, 2023, from <https://www.nature.com/articles/s41467-023-38063-x>
2. Fernández-Quintero, M. L., Kokot, J., Waibl, F., Fischer, A.-L. M., Quoika, P. K., Deane, C. M., & Liedl, K. R. (n.d.). Challenges in antibody structure prediction. *MAbs*, 15(1), 2175319. <https://doi.org/10.1080/19420862.2023.2175319>
3. Aziz, M., Iheanacho, F., & Hashmi, M. F. (2023). Physiology, Antibody. In *StatPearls*. StatPearls Publishing. <http://www.ncbi.nlm.nih.gov/books/NBK546670/>
4. *Antibody Immunoglobulin Diversity*. (2011, January 18). News-Medical.Net. <https://www.news-medical.net/health/Antibody-Immunoglobulin-Diversity.aspx>
5. Dunbar, J., & Deane, C. M. (2016). ANARCI: Antigen receptor numbering and receptor classification. *Bioinformatics*, 32(2), 298–300. <https://doi.org/10.1093/bioinformatics/btv552>
6. Dunbar, J., Fuchs, A., Shi, J., & Deane, C. M. (2013). ABangle: Characterising the VH-VL orientation in antibodies. *Protein Engineering, Design & Selection: PEDS*, 26(10), 611–620. <https://doi.org/10.1093/protein/gzt020>
7. Abanades, B., Wong, W. K., Boyles, F., Georges, G., Bujotzek, A., & Deane, C. M. (2023). ImmuneBuilder: Deep-Learning models for predicting the structures of immune proteins. *Communications Biology*, 6(1), Article 1. <https://doi.org/10.1038/s42003-023-04927-7>
8. *OpenMM - An Open Source Molecular Simulation Toolkit* / *Exxact Blog*. (n.d.). Retrieved December 14, 2023, from <https://www.exxactcorp.com/blog/molecular-dynamics/what-is-openmm>