Predicting the Quality of CDRH3 Antibody Loop Structural Models

Lilian M Denzler[a], Andrew CR Martin[a]

[a]*Department of Structural and Molecular Biology, University College of London, Gower Street, London, WC1E 6BT, UK*

Abstract

Therapeutic antibodies have shown an unprecedented pace of development and have brought new hope for the treatment of numerous diseases. The bioinformatic tools for modelling antibody structures have become invaluable for antibody engineering and the development of therapeutic antibodies. The antigen-binding site consists of six hypervariable loops, also known as the Complementary Determining Regions (CDR), all of which can be modelled with adequate accuracy, except for one. It remains markedly difficult to model the third CDR loop of the antibody heavy chain. The CDRH3 differs in length, has far greater sequence variability and has such a great structural diversity that modelling it is considerably harder. Many sophisticated approaches for antibody modelling, such as the abYmod software, have been developed. Although such efforts have improved prediction accuracy the results for the CDRH3 loop are still inconsistent and require further improvement. Providing a confidence score for the structure predictions would aid in differentiating well-modelled structures from incorrectly modelled structures, giving the abYmod user a clearer understanding of the generated model reliability. We present a model quality predictor, combining domain knowledge with machine learning techniques to predict the accuracy of CDRH3 models generated by antibody modelling software such as abYmod. The newly developed predictor scored a Mathews Correlation Coefficient of 0.99, and can thus be described as highly reliable. The predictor is made available at http://www.bioinf.org.uk/abs/qualiloop/.

1. Introduction

Antibodies are highly specialized proteins of the immune system that are produced in response to a foreign substance, called an antigen. A mature antibody binds a specific antigen with high affinity, while only weakly interacting with other antigens, or not at all. This high affinity, high specificity sets it apart from other pharmaceuticals. Furthermore, in contrast with small drug molecules, antibodies can not only bind pockets, but also flat, concave or even convex surfaces [1]. Their unique characteristics have enabled researchers to develop efficient antibody drugs for treating cancers, autoimmune disorders, infectious diseases and many more [2]. Their ability to target an immense variety of antigens allows for endless possibilities in application. The global market size was valued at USD 130.9 billion in 2020, estimated to grow 223.7 billion by the end of 2025 at a compound annual growth rate of 11.31% [3]. Four of the top 10 best-selling drugs in 2020 were monoclonal antibodies [4]. In order to rationally design therapeutic antibodies, knowledge of their structure is essential. The acquired structural information can be used to increase binding affinity to a target of interest, predicting both the exact binding site and the antibody stability as well as assessing immunogenicity [5]. As experimental structure determination is very costly and time consuming, computational predictions of an antibody's structure are used to streamline the process. Antibodies consist of a heavy and a light chain, which are linked by disulphide bonds. The N-terminal domain of each makes up the variable fragment (Fv), which contains the complementarity determining regions (CDRs). The antigen binding site is composed of six CDRs, also known as hypervariable loops. All except one of these loops can be clustered into a limited number of canonical structures. Therefore, modelling these loops with adequate accuracy is commonly achievable [6],[7]. However, the CDR loop 3 of the heavy chain (CDRH3) has a far greater sequence variability due to the processes of V(D)J recombination and somatic hyper-mutation and its structure has remained unclassifiable [8]. The variety in structure is so great, that its structural diversity is remarkable even compared to other protein loops [9]. It was found that over 75% of CDRH3 loops do not have a sub-Angstrom non-antibody structural neighbour, as well as that 30% of CDRH3 loops have a completely unique structure compared with under 3% for all other loops on average [9]. Apart from being the most structurally diverse, the H3 loop is also the most important for antigen binding, being located at the center of the binding site and forming the most contacts to the antigen [10]. In fact, it was demonstrated that differences in this loop alone were sufficient to enable otherwise identical antibodies to distinguish between various antigens [11]. According to the Kabat definition, the CDRH3 loop is made up of the residues 95-105 (Kabat numbering scheme[12]) in the heavy chain, with a potential insertion site at position 100. The possibility of such an insertion of a varying number of residues leads to a large range of loop lengths, with bovine antibodies being exceptionally long 1.

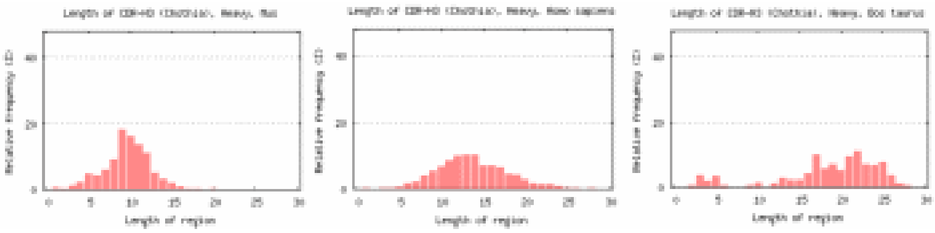


Figure 1: Distribution of CDR-H3 loop lengths in mouse (left), human (centre) and cow (right). Mouse and human antibody CDRH3s have a unimodal, yet almost normal distribution with a range of ca. 4-28 and 4-38 respectively. The length cut-off for CDRH3s depicted above is 30 amino acids. Bovine antibodies with significantly longer CDR-H3 loops than depicted here exist, reaching a length of 67 amino acids and above. [13]

For shorter loops, a higher prediction accuracy can be achieved than for longer CDRH3 loops. This was also shown by the Antibody Modelling Assessments (AMA), two blind contests that required researchers to build structural models from antibody sequences. The CDR-H3 loop modelling quality achieved at the contests was on average much lower for loops of longer lengths [14],[15]. Several different approaches for generating three-dimensional structure models from antibody sequences exist such as RosettaAntibody [16],[17], ABodyBuilder[18], PIGSPro[19] and abYmod, UCL's in-house software developed by Prof. Martin. One of the most used methods is RosettaAntibody, which implements template selection and ab initio CDR-H3 loop modelling using loop fragments and employing specific angle restraints which bias the conformational space towards so-called 'kinked' loops[20],[21]. In contrast, ABodyBuilder uses a database search algorithm (FREAD[22]) for CDR loop modelling. In this project abYmod is used, which is to be found at http://abymod.abysis.org/. The program utilizes extensive canonical class definitions, VH/VL angle prediction and a large database of loop structures (LoopDB) for CDR-H3 modelling to achieve optimal results. Upon inputting an antibody sequence, abYmod assigns the canonical class using a set of key residues [23]and where an exact match is not possible, a nearest match is made. Then, the program identifies the 10 best overall-matching PDB files according to sequence identity for the light and heavy chain. Of these, the best template is then identified for each CDR using sequence similarity and identity. In general, modelling the antibody using the single best overall-matching template works best and the CDR-specific templates are used if there is no canonical match. The VH/VL packing angle is then determined either using machine learning or the chosen template structure for one of the chains. CDR specific templates, if selected, are grafted onto the framework. If there is no template of the correct length for CDR-H3, the loop is built using the LoopDB database, containing CDRH3-like loops from all proteins. Finally, Gromacs energy minimization software is used to optimize the model. This method has proven very effective and preliminary analysis suggests the method achieves comparable results or outperforms other modelling software (see results section). Using these mentioned modelling methods, framework regions can generally be predicted with great accuracy (with better than 1Å RMSD[15]), as one can often find a very similar structure for the homology modelling process. However, the CDR loops are not as easily predicted due to their great diversity. If the canonical conformation of CDR loops 1-5 can be identified, they too can be modelled rather well, often within 1Å RMSD, for CDR-H3 loops the average is usually above 3Å [14]. ABodyBuilder is a modelling server that provides the user with a confidence score for each region (e.g.CDR-H2) of the antibody model. The given score is the probability that a specific region (e.g. CDR-H2) will be will be modeled within a specific RMSD threshold [18]. Thus, it can be used to obtain an expected RMSD value for a given probability (default 75%). For the CDR-H3 this score is calculated as a function of the loop length. The confidence scorer is described as robust, but less accurate in the case of CDR loops due to the lack of data.[18] ABLooper also provides a confidence metric for the CDR-H3 loop model, which is estimated by the diversity of a set of predicted conformations for the same loop[24].However, it remains unclear whether a high prediction diversity score points towards loops with multiple conformations or a low quality model. Furthermore, it remains unclear how well the generated diversity score reflects model quality[24].

Modelling the H3 loop is a hurdle for in silico development of therapeutic antibodies. Currently, there is no definite, reliable way to determine how accurate a generated structural model is within the H3 region. Therefore, the aim of this project will be to produce a user-friendly predictor of H3 model quality. The predictor will give the user an RMSD-range in Ångstöms, in which the generated model lies with a high probability. Making such a confidence score available via the web interface of the in-house modelling software abYmod is a future goal. Such a score is not provided by most modelling programs and would thus be a novel addition. This information can guide the user in the antibody engineering process. The user has the choice to determine whether the model is to be used in the intended way, or whether the model should be re-worked.

2. Results

To asses the predictive power regarding the CDR-H3 loop, the software was tested on a test-set of antibody structures used in the 2014 and 2011 Antibody Modelling Assessments [15],[14]. As the results depicted in 2 show, abYmod achieves results similar to, or better than other modelling programs. However, the outliers with very high RMSD values increase abYmod's RMSD average. The predictor in this work would aim to identify such outlier models. The predictive power of any machine learning model is largely dependent on the quality and size of the dataset it was trained on. As this is a non-linear, complex, multi-class classification problem, a substantial amount of data was required. Thus, an extensive, verified dataset of antibody structures called abYbank/AbDb[14], established by Prof. Andrew Martin, was utilised (1924 non-redundant structures). The root-mean-square deviation (RMSD) value, a measure of distance between backbone C- atoms of superimposed crystal structures and modelled structures, is calculated (see methods). This metric for model quality was used to classify models. The full pipeline for creating the final machine learning model that will predict model quality by giving its RMSD range starts with a feature-set calculation using the antibody sequence. The feature set includes many different attributes linked to sequence, physical characteristics, interactions, etc. within as well as outside of the loop. There is a plethora of information to train our classifier on, including packing quality, protrusion, hydrophobicity, pseudo-energy (see protrusion, hydrophobicity, pseudo-energy (see methods for details). The sequence logo 3 visualizes amino acid occurrence within the loop sequence, elements of which can be extracted as features [25],[26].

After creating the feature dataset, it is pre-processed (cleaning, scaling, encoding, see methods for details). Structures with a resolution below 4Å were removed given their low quality. Identical whole antibody structures

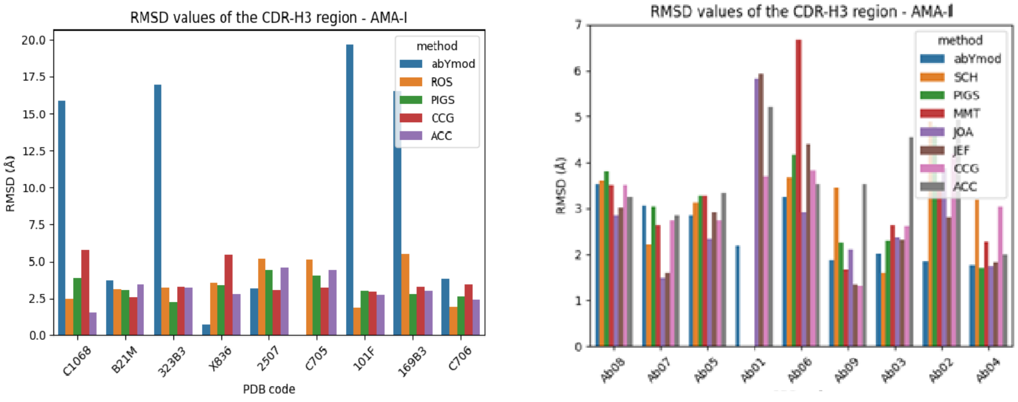


Figure 2: RMSD values of the CDR-H3 loop for structures from the Antibody Modelling Assessment I (2011) and AMAII (2014). abYmod outperforms other modelling software in some instances, but also has much lower accuracy in few outlier cases. (left: an abYmod structure for C705 could be generated, yet the RMSD calculation failed. Right: Ab01 is the rabbit antibody PDB:4MA3, which was excluded in the CDR-H3 modelling stage in AMAII due to difficulties modelling the overall structure previously. Ab01 is shown for the methods, where generated models were adequate for RMSD calculation.)

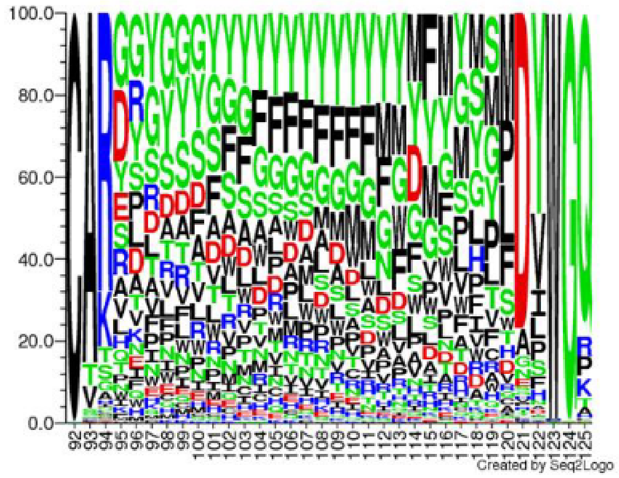


Figure 3: Sequence Logo of the CDRH3 loop sequence. Data on amino acid occurrence taken from http://abymod.abysis.org/. Visualized using Seq2Logo.

were removed from the dataset, while instances of different antibodies that matched in loop sequence were not removed. Models of some of these structures with the same loop sequence differ significantly. The few large RMSD ranges may stem from low resolution, e.g. the highest datapoint contains a structure with a resolution of 3.00 Å. Residue differences near the loop may also explain the conformational difference. Some of these structures are complexed while others are not, which may also affect the loop structure.

The dataset was also screened for any models which abYmod created using a template sequence from LoopDB, a database of CDR-H3 like loops from all proteins. This was not the case for any of our structures. The target data (i.e. RMSD values) are then transformed from numerical values to nominal values so that they can be used for classification. In order to define these nominal categories, the total RMSD range must be divided into categories. This is done either by creating uniform classes i.e. 1-2Å, 2-3Å (the optimal size of which must be determined), etc. or by creating balanced classes. When creating balanced classes, the upper and lower thresholds of a category are chosen in such a way that each class contains an equal number of instances. This approach is chosen to counteract the skewness of the RMSD distribution. However, this was found to negatively affect the final model's predictive power. Therefore, uniform classes were used. They are also transformed into a set of binary values according to a list of RMSD thresholds. This is done so that binary models can be trained, which will predict the probability e.g. that the model's RMSD is above 2Å, 2.2Å, 2.4Å, and so on. The number of binary classifiers incorporated into the first layer have a great effect on the final model, the general trend being that the more binary classifiers are used, the better the nominal prediction.

2.1. Feature Encoding and Selection

As some features are in the form of amino acid codes, these must be encoded before they can be passed to a machine learning model. The encoding strategy often determines how efficiently the model learns and how much information can be extracted. Different strategies were employed to represent followed by BLOSUM62 and NLF encoding. The physiochemical encoding strategy was implemented for all models, being the most effective. However, PCA-3 BLOSUM62, a dimensionality-reduced BLOSUM62 encoding method achieved comparable results. Feature selection was conducted to improve the ML model's learning capacity. A high-dimensional feature dataset bears the risk of introducing excessive noise, facilitating model overfitting and can be responsible for an overall decrease in model performance and stability. Each additionally inputted feature forces the model to handle a more complex task, which consumes excess computational power and time and leads to overfitting of the model.

Our model is trained on different feature sets selected using manual selection as well as algorithmic selection strategies, in order to determine the most effective feature selection method. None of the feature selection methods was a best fit for all models.

After the data is processed, it can be fed into different machine learning models. Different model types are investigated, as the most suited model-type has to be heuristically determined. We decided on the following list, which includes some of the most commonly used algorithms: logistic regression, linear discriminant analysis, K-nearest neighbours classifier, decision tree classifier, Gaussian NB, random forest classifier, support vector machine, probability-based voting (also known as soft voting) and extreme gradient boosting (XGBoost)[27].

The best model, and its best hyperparameters, are then determined for each binary RMSD target. The set of binary models outputs a number of predictions that give the likelihood of the model having an RMSD above the threshold value of the respective model. These predictions are then added to the feature set, which a top-layer classifier is then trained on. Thus, a quasi-voting-system is incorporated into the final classifier, in which a set of weaker classifiers vote on the model quality.

2.2. Hyperparameter Optimization

In the process of hyperparameter optimization, the configuration of model parameters which results in best performance is selected. This is usually a computationally expensive and manual procedure. In an effort to automate this process, a population is defined for each model type, so hyperparameter optimization can be conducted automatically for each model and seamlessly integrated into the full model creation process. Two different methods for hyperparameter optimization were tested. The first is a hybrid approach of randomized search and grid search, the second uses a genetic algorithm for optimization. The genetic algorithm was found to achieve slightly better results and was selected for all models.

2.3. Model Performance

The overall best final model is composed of several different binary classifiers, with an extreme gradient boosting (XGBoost) top-layer nominal classifier. Features were selected using random forest feature selection. A final

MCC value of 0.99 could be achieved for a model using the abYmod log file as input as well as the loop model file itself. This value slightly dropped to 0.92 if no such log file was given. This is mainly due to the fact that the template sequence abYmod used to generate the model is unknown in the latter case. The classifier predicts whether a model has an RMSD of below 2Å, between 2-4Å,or above 4Å. These cut-off values were selected based on the observation that abYmod generally produces a model with RMSD below 4Å. Incorrectly modelled structures (2) may be identified by screening for structures estimated to have an RMSD above 4Å. If a very high-quality model is needed one should also exclude models above 2Å.

3. Discussion

The results suggest that our presented classifier can differentiate between well-modelled and less well-modelled CDR-H3 loop structures. An MCC value of 0.99 was achieved, which underlines this ability for accurate discrimination. Different methods for data pre-processing, feature encoding, feature selection and hyperparameter optimization were tested. Feature encoding methods that were very high-dimensional (one-hot-encoding, BLOSUM62, NLF) were found to be unfavorable. Dimensionality reduction methods (Principal Component Analysis (PCA), Independent Component Analysis (ICA), projection-based methods e.g. t-SNE) were used on BLOSUM62 encoded matrices, which lead to significant improvement. However, a physiochemical encoding strategy was most effective. The selection of features incorporated in the training set seemed to be most important for effective learning. A multitude of methods were tested. No one-fit-for-all method for the different models could be found. However, for our top-layer classifier in our final model recursive feature elimination worked best. A set of commonly used machine learning algorithms were tested, and the best ML models were incorporated into the final ensemble model. A stacked model approach (consisting of 23 binary classifiers and a single top-layer nominal classifier) was shown to outperform single ML models. An MCC value of 0.99 was achieved for a classifier predicting whether an input-model has an RMSD value below 2Å, 2Å-4Å or above 4Å. Given that abYbank/AbDb is soon to be expanded by an additional ca. 2000 structures, classifier performance on models without a known abYmod-template sequence may be improved by a larger dataset. It is conceivable that the described predictor may also be incorporated in the antibody modelling process as a low-quality filter in the future, flagging certain structures for re-modelling.

In a future research project residue patterns in correlation with RMSD may be analyzed. Possibly, one might identify certain sequence patterns that make accurate modelling with abYmod more difficult. Furthermore, separate classifiers according to loop length can be built. Given that loop length is the most important determinant of model quality, this approach may yield some insight into the challenges of modelling shorter vs longer loops. One could also conduct an analysis of the predictor's behaviour when abYmod is forced to use LoopDB-based modelling. This might shed light on whether the ML model presented in this paper is biased towards abYmod's used source of template structures. It would also give an indication of how well the predictor would work in combination with other modelling software.

References

[1] J. C. A. Janeway, P. Travers, M. Walport, M. J. Shlomchik, The interaction of the antibody molecule with specific antigen (2001).
URL `https://www.ncbi.nlm.nih.gov/books/NBK27160/`

[2] R. M. Lu, Y. C. Hwang, I. J. Liu, C. C. Lee, H. Z. Tsai, H. J. Li, H. C. Wu, Development of therapeutic antibodies for the treatment of diseases (1 2020). doi:10.1186/s12929-019-0592-z.
URL `https://doi.org/10.1186/s12929-019-0592-z`

[3] Antibodies market size, share trends, growth, forecast — 2020 to 2025.
URL `https://www.marketdataforecast.com/market-reports/antibodies-market`

[4] L. Urquhart, Top companies and drugs by sales in 2020, Nature Reviews Drug Discovery (4 2021). doi:10.1038/d41573-021-00050-6.

[5] K. R. Abhinandan, A. C. Martin, Analyzing the "degree of humanness" of antibody sequences, Journal of Molecular Biology 369 (2007) 852–862. doi:10.1016/j.jmb.2007.02.100.

[6] B. North, A. Lehmann, R. L. Dunbrack, A new clustering of antibody cdr loop conformations, Journal of Molecular Biology 406 (2011) 228–256. doi:10.1016/j.jmb.2010.10.030.

[7] B. D. Weitzner, R. L. Dunbrack, J. J. Gray, The origin of cdr h3 structural diversity, Structure 23 (2015) 302–311. doi:10.1016/j.str.2014.11.010.

[8] J. A. Finn, J. K. Leman, J. R. Willis, A. Cisneros, J. E. Crowe, J. Meiler, Improving loop modeling of the antibody complementarity-determining region 3 using knowledge-based restraints, PLOS ONE 11 (2016) e0154811. doi:10.1371/journal.pone.0154811.
URL https://dx.plos.org/10.1371/journal.pone.0154811

[9] C. Regep, G. Georges, J. Shi, B. Popovic, C. M. Deane, The h3 loop of antibodies shows unique structural characteristics, Proteins: Structure, Function and Bioinformatics 85 (2017) 1311–1318. doi:10.1002/prot.25291.
URL /pmc/articles/PMC5535007/?report=abstract https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5535007/

[10] R. M. MacCallum, A. C. Martin, J. M. Thornton, Antibody-antigen interactions: Contact analysis and binding site topography, Journal of Molecular Biology 262 (1996) 732–745. doi:10.1006/jmbi.1996.0548.

[11] J. L. Xu, M. M. Davis, Diversity in the cdr3 region of v(h) is sufficient for most antibody specificities, Immunity 13 (2000) 37–45. doi:10.1016/S1074-7613(00)00006-6.
URL https://pubmed.ncbi.nlm.nih.gov/10933393/

[12] E. Kabat, T. T. Wu, H. Perry, C. Foeller, K. Gottesman, (1992).
URL

[13] T. T. Wong, Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation, Pattern Recognition 48 (2015) 2839–2846. doi:10.1016/j.patcog.2015.03.009.

[14] J. C. Almagro, M. P. Beavers, F. Hernandez-Guzman, J. Maier, J. Shaulsky, K. Butenhof, P. Labute, N. Thorsteinson, K. Kelly, A. Teplyakov, J. Luo, R. Sweet, G. L. Gilliland, Antibody modeling assessment, Proteins: Structure, Function and Bioinformatics 79 (2011) 3050–3066. doi:10.1002/prot.23130.

[15] J. C. Almagro, A. Teplyakov, J. Luo, R. W. Sweet, S. Kodangattil, F. Hernandez-Guzman, G. L. Gilliland, Second antibody modeling assessment (ama-ii) (2014). doi:10.1002/prot.24567.
URL https://pubmed.ncbi.nlm.nih.gov/24668560/

[16] A. Sircar, E. T. Kim, J. J. Gray, Rosettaantibody: Antibody variable region homology modeling server, Nucleic Acids Research 37 (2009) W474–W479. doi:10.1093/nar/gkp387.
URL `https://jhu.pure.elsevier.com/en/publications/rosettaantibody-antibody-va`

[17] A. Sivasubramanian, A. Sircar, S. Chaudhury, J. J. Gray, Toward high-resolution homology modeling of antibody f ¡sub¿v¡/sub¿ regions and application to antibody-antigen docking, Proteins: Structure, Function, and Bioinformatics 74 (2009) 497–514. doi:10.1002/prot.22309.
URL `http://doi.wiley.com/10.1002/prot.22309`

[18] J. Leem, J. Dunbar, G. Georges, J. Shi, C. M. Deane, Abodybuilder: Automated antibody structure prediction with data–driven accuracy estimation, mAbs 8 (2016) 1259–1268. doi:10.1080/19420862.2016.1205773.
URL `https://www.tandfonline.com/doi/full/10.1080/19420862.2016.1205773`

[19] R. Lepore, P. P. Olimpieri, M. A. Messih, A. Tramontano, Pigspro: Prediction of immunoglobulin structures v2, Nucleic Acids Research 45 (2017) W17–W23. doi:10.1093/nar/gkx334.
URL `http://biocomputing.it/pigspro.`

[20] C. T. Schoeder, S. Schmitz, J. Adolf-Bryfogle, A. M. Sevy, J. A. Finn, M. F. Sauer, N. G. Bozhanova, B. K. Mueller, A. K. Sangha, J. Bonet, J. H. Sheehan, G. Kuenze, B. Marlow, S. T. Smith, H. Woods, B. J. Bender, C. E. Martina, D. D. Alamo, P. Kodali, A. Gulsevin, W. R. Schief, B. E. Correia, J. E. Crowe, J. Meiler, R. Moretti, Modeling immunity with rosetta: Methods for antibody and antigen design, Biochemistry 60 (2021) 825–846. doi:10.1021/ACS.BIOCHEM.0C00912/SUPPL$_{F}ILE/BI0C00912_{S}I_{0}02.ZIP.$
URL`/pmc/articles/PMC7992133/ /pmc/articles/PMC7992133/?report=abstract https`

[21] B. D. Weitzner, J. J. Gray, Accurate structure prediction of cdr h3 loops enabled by a novel structure-based c-terminal constraint, The Journal of Immunology 198 (2017) 505–515. doi:10.4049/jimmunol.1601137.
URL `http://www.jimmunol.org/content/198/1/505`

[22] Y. Choi, C. M. Deane, Fread revisited: Accurate loop structure prediction using a database search algorithm, Proteins: Structure, Function, and Bioinformatics 78 (2010) 1431–1440. doi:10.1002/prot.22658.
URL `http://doi.wiley.com/10.1002/prot.22658`

[23] A. C. Martin, J. M. Thornton, Structural families in loops of homologous proteins: Automatic classification, modelling and application to antibodies, Journal of Molecular Biology 263 (1996) 800–815. doi:10.1006/jmbi.1996.0617.

[24] B. Abanades, G. Georges, A. Bujotzek, C. M. Deane, Ablooper: fast accurate antibody cdr loop structure prediction with accuracy estimation, Bioinformatics 38 (2022) 1877–1880. doi:10.1093/BIOINFORMATICS/BTAC016.
URL https://academic.oup.com/bioinformatics/article/38/7/1877/6517780

[25] M. C. F. Thomsen, M. Nielsen, Seq2logo: A method for construction and visualization of amino acid binding motifs and sequence profiles including sequence weighting, pseudo counts and two-sided representation of amino acid enrichment and depletion, Nucleic Acids Research 40 (7 2012). doi:10.1093/nar/gks469.
URL https://pubmed.ncbi.nlm.nih.gov/22638583/

[26] M. C. Shaner, I. M. Blair, T. D. Schneider, Sequence logos: A powerful, yet simple, tool, Proceedings of the Annual Hawaii International Conference on System Sciences 1 (1993) 813–821. doi:10.1109/HICSS.1993.270609.

[27] T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, Vol. 13-17-August-2016, Association for Computing Machinery, 2016, pp. 785–794. doi:10.1145/2939672.2939785.

[28] B. Bengfort, R. Bilbro, P. Johnson, P. Billet, P. Roman, P. Deziel, K. McIntyre, L. Gray, A. Ojeda, E. Schmierer, A. Morris, M. Morrison, Yellowbrick v1.3 (2 2021). doi:10.5281/ZENODO.4525724.
URL https://zenodo.org/record/4525724

[29] D. Eisenberg, R. M. Weiss, T. C. Terwilliger, W. Wilcox, Hydrophobic moments and protein structure, Vol. 17, The Royal Society of Chemistry, 1982, pp. 109–120. doi:10.1039/FS9821700109.
URL https://pubs.rsc.org/en/content/articlehtml/1982/fs/fs9821700109
https://pubs.rsc.org/en/content/articlelanding/1982/fs/fs9821700109

[30] D. Krstajic, L. J. Buturovic, D. E. Leahy, S. Thomas, Cross-validation pitfalls when selecting and assessing regression and classification models, Journal of Cheminformatics 6 (2014) 10. doi:10.1186/1758-2946-6-10.
URL https://jcheminf.biomedcentral.com/articles/10.1186/1758-2946-6-10

[31] R. Kohavi, A study of cross-validation and bootstrap for accuracy estimation and model selection (1995).
URL `http://robotics.stanford.edu/ ronnyk`

[32] B. W. Matthews, Comparison of the predicted and observed secondary structure of t4 phage lysozyme, BBA - Protein Structure 405 (1975) 442–451. doi:10.1016/0005-2795(75)90109-9.
URL `https://www.scienceopen.com/document?vid=4306d5d3-4410-4930-a5c6-b19620a483ca`

Appendix A. Experimental Procedures

Appendix A.1. Computing

All machine learning and, feature selection and hyperparameter optimization algorithms were implemented in Python. The Scikit-learn library was used for training models, the Yellowbrick[28] library was utilized for visualization. All code is available at https://github.com/LilianDenzler/qualiloop . The code was run under CentOS 7 on an 8-core virtual machine on an Intel Xeon 4208 CPU with 16Gig RAM.

Appendix A.2. Data Pre-Processing and Preparation

Handling Null Values and Duplicates: The dataset containing target RMSD values, and the calculated features was screened for null values. If a feature column contained more than 5% null values, it was dropped (none removed). Rows that contained any null values were removed from the dataset (11 rows in total).

Appendix A.2.1. Duplicate Screening

Using AbDb's redundancy information it was ensured that no antibodies were present in the dataset more than once. The dataset is additionally screened for duplicate instances.

Appendix A.2.2. Scaling

Normalization and Standardization are tested as scaling methods. Both approaches are greatly influenced by outliers, and such datapoints are ideally removed for optimal scaling. Here we define outliers as datapoints that lie over 1.5 times the interquartile range (IQR) below the first quartile or above the third quartile. The IQR is defined as the range between quartile 1, i.e. the median of the lower half of the data, and quartile 3, i.e. the median of the upper half of the data. However, across all features there are a total of 632 outlier values and removing such a large number of datapoints is not a viable option. A robust scaler was also used, which uses statistics that are robust to outliers. The median is set to zero and numerical features are scaled to the interquartile range.

Appendix A.2.3. BLOSUM 62 encoding

The BLOSUM62 matrix reflects the frequencies of amino acid substitutions within a locally aligned, conserved regions of proteins with at least 62% similarity. Each amino acid is represented by a row (or column) of the BLOSUM62 matrix. Dimensionality reduction techniques are employed: Principal Component Analysis (PCA), Independent Component Analysis (ICA), projection-based methods (t-SNE, Isomap). Three components were used as features.

Appendix A.2.4. Physiochemical Feature Encoding

In a paper by L. Nanni and A.Lumini a new encoding technique is presented which was developed for machine learning classifiers. Many physiochemical properties are calculated and transformed using a non-linear Fisher transform for dimensionality reduction. A vector of length 19 is produced for each amino acid34. In a paper on designing a neural network for predicting the packing angle of the light and heavy variable chain of an antibodies, A.C.R. Martin and K.R. Abhinandan introduce an encoding method that produces a four-dimensional physiochemical feature vector25. The amino acid properties used are 1) the total number of side-chain atoms, 2) the number of side-chain atoms in the shortest path from C to the most distal atom, 3) the Eisenberg consensus hydrophobicity[29], 4) the charge.

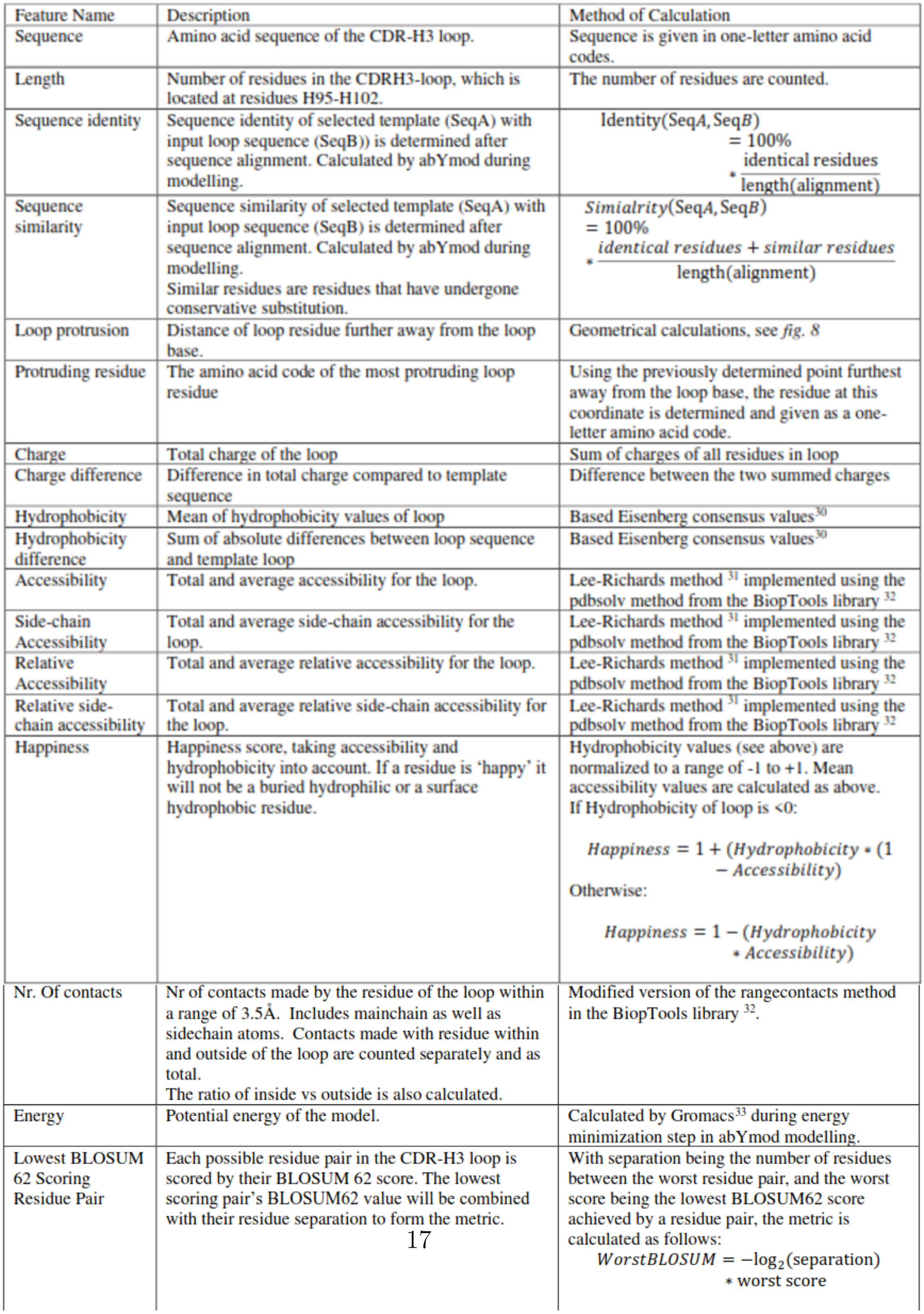
Appendix A.3. Dataset-splitting

The final model was evaluated using a test set, separated from the training set at the start in a 30/70 split (lock-box principle) 35. The performance of all individual sub-models of the first layer are determined using stratified K-folds cross-validation (k=10) as the dataset is imbalanced, being skewed towards lower RMSD values. The method is differentiable from K-folds cross validation as it uses stratified sampling instead of random sampling. This ensures each class is represented, as the percentage of samples for each class are preserved. A validation set, usually used for testing during the optimization stage will be omitted in favour of stratified K-folds cross-validation (k=10) [30] [31].

Appendix A.4. Model Assessment

Model assessment must be considered at two levels as performance metrics of binary and multi-class classifiers are calculated differently and must thus be considered separately. The Mathews Correlation Coefficient (MCC) [32] is deemed the most informative, taking the ratios of the four confusion matrix categories into account 39and is thus more reliable than the F1 score and accuracy . It is also consistent for both binary and multi-class problems and therefore well suited for our purpose.

Appendix A.5. Feature Calculations



Feature Name	Description	Method of Calculation
Sequence	Amino acid sequence of the CDR-H3 loop.	Sequence is given in one-letter amino acid codes.
Length	Number of residues in the CDRH3-loop, which is located at residues H95-H102.	The number of residues are counted.
Sequence identity	Sequence identity of selected template (SeqA) with input loop sequence (SeqB)) is determined after sequence alignment. Calculated by abYmod during modelling.	$\text{Identity}(\text{Seq}A, \text{Seq}B) = 100\% * \frac{\text{identical residues}}{\text{length(alignment)}}$
Sequence similarity	Sequence similarity of selected template (SeqA) with input loop sequence (SeqB) is determined after sequence alignment. Calculated by abYmod during modelling. Similar residues are residues that have undergone conservative substitution.	$\mathit{Similarity}(\text{Seq}A, \text{Seq}B) = 100\% * \frac{\mathit{identical\ residues} + \mathit{similar\ residues}}{\text{length(alignment)}}$
Loop protrusion	Distance of loop residue further away from the loop base.	Geometrical calculations, see *fig. 8*
Protruding residue	The amino acid code of the most protruding loop residue	Using the previously determined point furthest away from the loop base, the residue at this coordinate is determined and given as a one-letter amino acid code.
Charge	Total charge of the loop	Sum of charges of all residues in loop
Charge difference	Difference in total charge compared to template sequence	Difference between the two summed charges
Hydrophobicity	Mean of hydrophobicity values of loop	Based Eisenberg consensus values[30]
Hydrophobicity difference	Sum of absolute differences between loop sequence and template loop	Based Eisenberg consensus values[30]
Accessibility	Total and average accessibility for the loop.	Lee-Richards method [31] implemented using the pdbsolv method from the BiopTools library [32]
Side-chain Accessibility	Total and average side-chain accessibility for the loop.	Lee-Richards method [31] implemented using the pdbsolv method from the BiopTools library [32]
Relative Accessibility	Total and average relative accessibility for the loop.	Lee-Richards method [31] implemented using the pdbsolv method from the BiopTools library [32]
Relative side-chain accessibility	Total and average relative side-chain accessibility for the loop.	Lee-Richards method [31] implemented using the pdbsolv method from the BiopTools library [32]
Happiness	Happiness score, taking accessibility and hydrophobicity into account. If a residue is 'happy' it will not be a buried hydrophilic or a surface hydrophobic residue.	Hydrophobicity values (see above) are normalized to a range of -1 to +1. Mean accessibility values are calculated as above. If Hydrophobicity of loop is <0: $\mathit{Happiness} = 1 + (\mathit{Hydrophobicity} * (1 - \mathit{Accessibility})$ Otherwise: $\mathit{Happiness} = 1 - (\mathit{Hydrophobicity} * \mathit{Accessibility})$
Nr. Of contacts	Nr of contacts made by the residue of the loop within a range of 3.5Å. Includes mainchain as well as sidechain atoms. Contacts made with residue within and outside of the loop are counted separately and as total. The ratio of inside vs outside is also calculated.	Modified version of the rangecontacts method in the BiopTools library [32].
Energy	Potential energy of the model.	Calculated by Gromacs[33] during energy minimization step in abYmod modelling.
Lowest BLOSUM 62 Scoring Residue Pair	Each possible residue pair in the CDR-H3 loop is scored by their BLOSUM 62 score. The lowest scoring pair's BLOSUM62 value will be combined with their residue separation to form the metric.	With separation being the number of residues between the worst residue pair, and the worst score being the lowest BLOSUM62 score achieved by a residue pair, the metric is calculated as follows: $\mathit{WorstBLOSUM} = -\log_2(\text{separation}) * \text{worst score}$

Figure A.4: A summary of how different feature values were calculated.

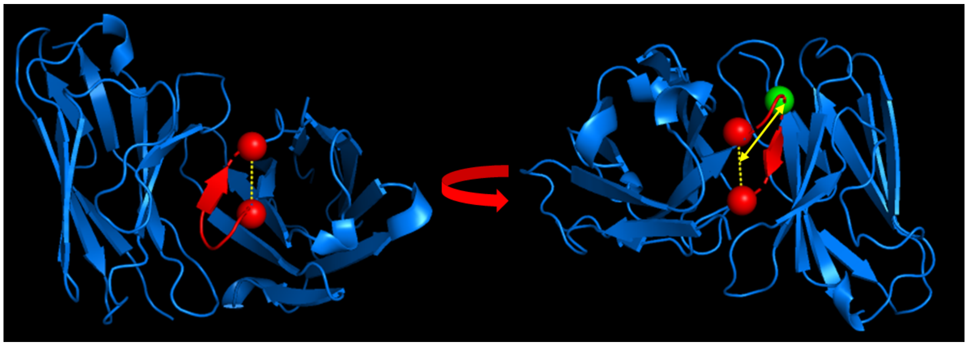


Figure A.5: Diagram visualizing the process underlying the protrusion calculation. First, the base residues (i.e H95 and H102, shown as red spheres) of the CDR-H3 (shown in red) are identified. Then, a line is drawn between the two C atoms of these residues (yellow dashed line). The distance of the C-atom of each residue in the CDR-H3 loop to this line is calculated. The residue which has the greatest distance to the line (shown as green sphere) is outputted as one-letter amino acid code and used as feature. The distance in Å (depicted as yellow arrow) is used as the 'protrusion' feature.

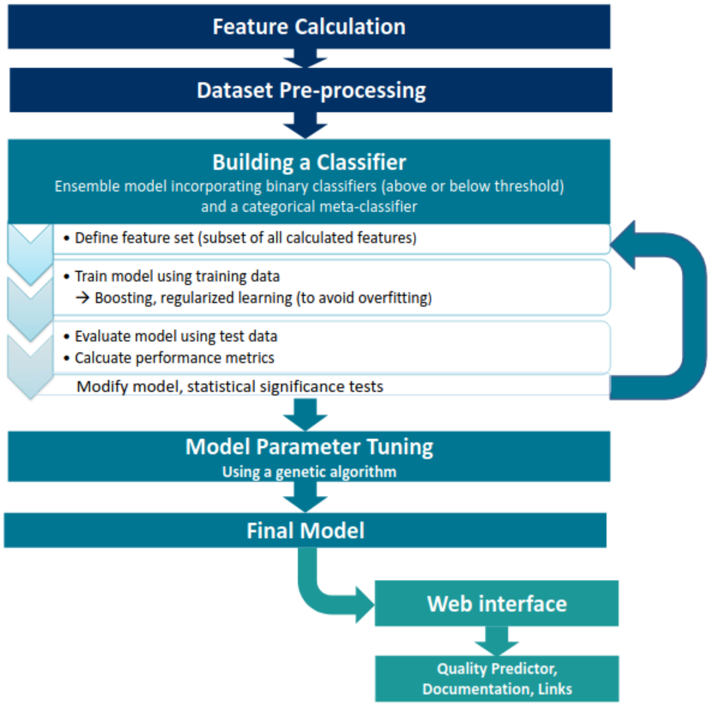

Feature Calculation
Dataset Pre-processing
Building a Classifier
Ensemble model incorporating binary classifiers (above or below threshold) and a categorical meta-classifier
• Define feature set (subset of all calculated features)
• Train model using training data
→ Boosting, regularized learning (to avoid overfitting)
• Evaluate model using test data
• Calcuate performance metrics
Modify model, statistical significance tests
Model Parameter Tuning
Using a genetic algorithm
Final Model
Web interface
Quality Predictor, Documentation, Links

Figure A.6: Simplified pipeline for creating the final machine learning model that will predict model quality by giving its RMSD range.