Predicting the Quality of CDR-H3 Antibody Loop Structural Models

Lilian M. Denzler and Andrew C.R. Martin
Institute of Structural and Molecular Biology, Division of Biosciences,
University College of London,
Gower Street,
London WC1E 6BT, UK

December 19, 2022

Abstract

Therapeutic antibodies have shown an unprecedented pace of development and have brought new hope for the treatment of numerous diseases. Bioinformatics 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 (CDRs), all of which can generally be modelled with high accuracy, except for CDR-H3, which has far greater length, sequence and structural variability, making modelling it is considerably harder.

Many approaches for antibody modelling, such as our abYmod software, have been developed. Although such efforts have improved prediction accuracy, the results for CDR-H3 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 user a clearer understanding of the generated 3D-model reliability.

We present a 3D-model quality predictor, combining domain knowledge with machine learning techniques to predict the accuracy of CDR-H3 3D-models generated by antibody modelling software such as abYmod. The newly developed predictor scored a Matthews 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 and specificity. This sets them apart from other pharmaceuticals and makes them effective drugs with endless possibilities in application given their ability to target an immense variety of antigens. In contrast with small drug molecules, antibodies can not only bind pockets, but also flat, concave or 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]. Four of the top 10 best-selling drugs in 2020 were monoclonal antibodies[3].

In order to design therapeutic antibodies rationally, knowledge of their structure is essential. The acquired structural information can be used to modify binding affinity to a target of interest, predicting both the exact binding site and the antibody stability as well as assessing immunogenicity[4]. As experimental structure determination is costly and time consuming, computational predictions of an antibody's structure are used to streamline the process.

The variable fragment (Fv) of an antibody contains the six complementarity determining regions (CDRs, also known as hypervariable loops) which form the antigen binding site. All except one of these loops can be clustered into a limited number of 'canonical structures'[5]. Therefore, modelling these loops with adequate accuracy is commonly achievable[6]. However, the CDR loop 3 of the heavy chain (CDR-H3) has a far greater sequence and length variability due to the processes of V(D)J recombination and somatic hyper-mutation and its structure has remained mostly unclassifiable[7]. The variety in structure is so great, that its structural diversity is remarkable even compared with other protein loops[8]. It was found that over 75% of CDR-H3 loops do not have a sub-Ångström non-antibody structural neighbour, while 30% of CDR-H3 loops have a completely unique structure, compared with under 3% for all non-antibody loops[8].

Apart from being the most structurally diverse, the CDR-H3 loop is also the most important for antigen binding, being located at the centre of the binding site and forming the most contacts with the antigen[1]. It was demonstrated that differences in this loop alone are sufficient to enable otherwise identical antibodies to distinguish between various antigens[9].

According to the Kabat definition, the CDR-H3 loop is made up of the residues H95–H102 (using the Kabat[10], Chothia[5] or Martin[11] numbering schemes) in the heavy chain, with a potential insertion site at position H100. 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 (Figure 4). **Lilian:** Figure missing!

For shorter loops, a higher prediction accuracy can be achieved than for longer CDR-H3 loops. This was also shown in the Antibody Modelling Assessments

(AMA), two blind contests that required researchers to build three-dimensional structural models ('3D-models' — used throughout this text to distinguid from machine-learning models 'ML-models') from antibody sequences. The CDR-H3 loop modelling quality achieved at the contests was, on average, much lower for loops of longer lengths[12, 13].

Several different approaches for generating 3D-models from antibody sequences exist including RosettaAntibody[14, 15], ABodyBuilder[16], PIGSPro[17], Lyra[18], AbLooper[19] and our own abYmod (manuscript in preparation). 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. abYmod `http://abymod.abysis.org/` utilizes extensive canonical class definitions, V_H/V_L angle prediction and a large database of loop structures (LoopDB) for CDR-H3 modelling. 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 class is identified.

abYmod selects light and heavy chain frameworks separately from PDB templates. First these are selected on the basis of the number of matched canonical classes and then on the basis of sequence identity. The V_H/V_L packing angle is currently selected from the parent that has the best sequence identity over both chains, but an improved method is currently in development. Any CDRs where there was no canonical match are then grafted onto the framework. If there is no template of the correct length for CDR-H3, the loop is built using LoopDB, a database of CDR-H3-like loops from all proteins. Finally, Gromacs energy minimization software is used to optimize the 3D-model. This method has proven very effective and preliminary analysis suggests the method achieves comparable results, or outperforms, other modelling software (see Results).

Using these modelling methods, framework regions can generally be predicted with great accuracy (better than 1Å Cα RMSD[13]), 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 CDR-L1,L2,L3,H1,H2 can be identified, they too can be modelled rather well, often within 1Å Cα RMSD. for CDR-H3 loops the The average values are taken from the antibody modelling assessment average is usually above 3Å[12]. **Lilian:** Something odd here — not sure what it's supposed to say!

ABodyBuilder is a modelling server that provides the user with a confidence score for each region (e.g. CDR-H2) of the antibody 3D-model. The given score is the probability that a specific region (e.g. CDR-H2) will be modelled within a specific RMSD threshold[16]. Thus, it can be used to obtain an expected RMSD value for a given probability (default 75%). For 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 owing to the lack of data[16]. ABLooper also provides a confidence metric for the CDR-H3 loop 3D-model, which is estimated by the diversity of a set of predicted conformations for the same loop[19]. However, it remains unclear whether a high prediction diversity score points towards loops with multiple conformations or a low quality 3D-model. Furthermore, it remains unclear how well the generated diversity score reflects 3D-model quality[19].

Modelling the CDR-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 3D-model is within the CDR-H3 region. Therefore, we have produced a user-friendly predictor of CDR-H3 3D-model quality. The predictor will give the user an RMSD-range in Ångströms, in which the generated 3D-model lies with a high probability. This information can guide the user in the antibody engineering process. The user has the choice to determine whether the 3D-model is to be used as is, or whether the 3D-model should be re-worked.

2 Results

The predictive power of any machine learning model ('ML-model') is largely dependent on the quality and size of the dataset on which it was trained. 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[24], was utilised (1924 non-redundant structures). The Cα root-mean-square deviation (RMSD) value between the crystal structures and modelled structures was calculated (see Methods) and was used to classify 3D-models.

The classifier predicts whether a 3D-model has an RMSD of below 2Å, between 2–4Å, or above 4Å. These cutoff values were selected based on the observation that abYmod generally produces a 3D-model with RMSD below 4Å. Incorrectly modelled structures (Figure 1) may be identified by screening for structures estimated to have an RMSD above 4Å. If a very high-quality 3D-model is needed one should also exclude 3D-models with RMSD above 2Å.

The full pipeline for creating the final ML-model starts with feature-set calculation using the antibody sequence. The feature set includes attributes linked to sequence, structure, physical characteristics, interactions, etc., within, as well as outside, the loop. The sequence logo (Figure 2) 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 worse than 4Å were removed. Instances of antibodies in our non-redundant dataset that matched in

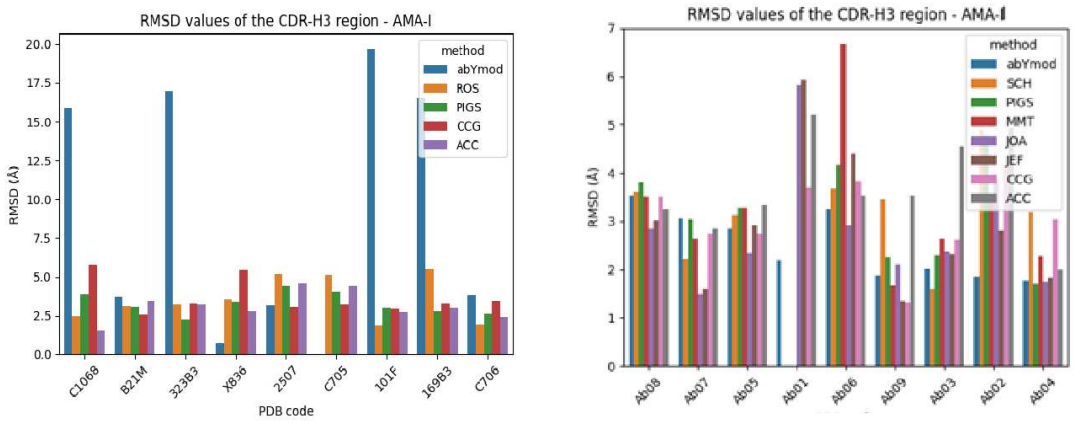

Figure 1: RMSD values of the CDR-H3 loop for structures from the Antibody Modelling Assessment I (2011) and AMAII (2014). abYmod outperforms other modelling software in some instances, but also has much lower accuracy in few outlier cases. Right: Ab01 is the rabbit antibody PDB:4MA3, which was excluded in the CDR-H3 modelling stage in AMAII due to difficulties modelling the overall structure previously. Ab01 is shown for the methods, where generated 3D-models were adequate for RMSD calculation.

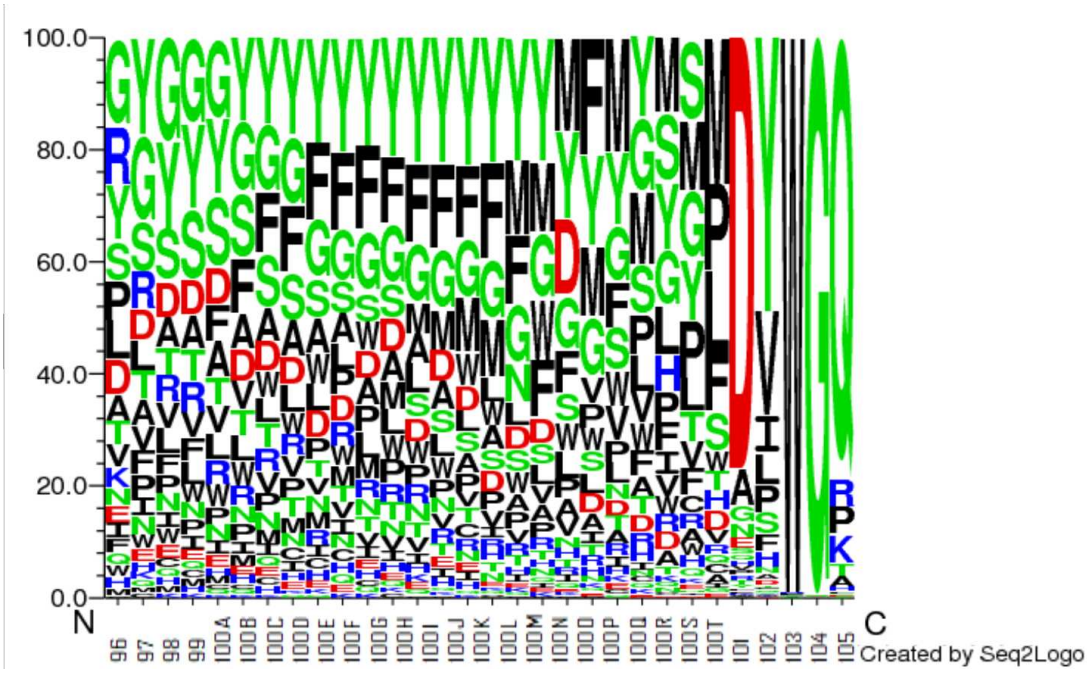

Figure 2: Sequence Logo of the CDR-H3 loop sequence. Data on amino acid occurrence taken from `http://abymod.abysis.org/` Visualized using Seq2Logo using Kabat Numbering

loop sequence were not removed as 3D-Models **Lilian:** Just checking you mean models not structures? Do the structures also differ significantly, or do you really mean that the error varies significantly? of some of these structures with the same loop sequence differ significantly. **Lilian:** If I understand correctly, you mean that CDR-H3 could have the same sequence in multiple structures, but the overall sequence was different? The few large RMSD ranges may stem from low resolution. For example, the loop sequence with the largest RMSD range has multiple structure files linked to it of varying quality, one of which has a resolution of only 3.00Å. Residue differences near the loop may also explain the conformational difference of the loop itself, even if the loop sequence does not differ. Some of these structures are complexed while others are not, which may also affect the loop structure (manuscript in preparation).

The target data (i.e. RMSD values) are 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 (e.g. 1–2Å, 2–3Å, etc.), the optimal size of which must be determined, 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. Initially, this approach was chosen to counteract the skewness of the RMSD distribution. However, this was found to have a negative effect on the final ML-model's predictive power, so uniform classes were used.

The RMSD values are also transformed into a set of binary values according to a list of RMSD thresholds (i.e. a 1 is assigned to above and 0 to below a given threshold). This is done so that binary ML-models can be trained, which will predict the probability e.g. that the 3D-model's RMSD is above 2Å, 2.2Å, 2.4Å, and so on. The number of binary classifiers incorporated into the first layer **Lilian:** You need to describe the overall architecture first so this makes sense. I would have a much simpler version of Figure 3 which just shows the ML architecture has a great effect on the final ML-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 names, these must be encoded before they can be passed to a ML-model. The encoding strategy often determines how efficiently the ML-model learns and how much information can be extracted. Different strategies were employed to represent protein sequences numerically, such as BLOSUM62[27] and NLF[28] encoding (a non-linear Fisher transform of a large set of physicochemical properties). However, the simple four-feature physiochemical encoding strategy[29] was implemented for all ML-models, as it was found to be the most effective, although 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 ML-model over-fitting and can be responsible for an overall decrease in ML-model performance and stability. Each additional input feature forces the ML-model to handle a more complex task, which consumes excess computational power and time and provides more variables leading to over-fitting of the ML-model.

In order to determine the most effective feature selection method, the ML-model was trained on different feature sets selected using manual selection as well as algorithmic selection strategies (see methods). None of the feature selection methods was a best fit for all ML-models. To create a ML-model implementing the encoding and feature selection strategies best suited for the specific ML-structure **Lilian:** What is this?, a number of different combinations were tested, summarized in Table **??**. Additional ML-Models were discarded due to poor performance.

After the data were processed, they were used to train different ML-models. Different types of ML-model were investigated, as the most suited ML-model type has to be determined heuristically. The following list, which includes some of the most commonly used algorithms, was used: 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)[30].

The best ML-model, and its best hyperparameters, were then determined for each binary RMSD target. The set of binary ML-models outputs a number of predictions that give the likelihood of the 3D-model having an RMSD above the threshold value of the respective ML-model. These predictions are then added to the feature set, on which a top-layer classifier is then trained (Figure **??**) **Lilian:** Incorrect reference. Thus, a quasi-voting-system is incorporated into the final classifier, in which a set of weaker classifiers vote on the ML-model quality.

2.2 Hyperparameter Optimization

In the process of hyperparameter optimization, the configuration of ML-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 was defined for each ML-model type, so hyperparameter optimization could be conducted automatically for each ML-model and seamlessly integrated into the full ML-model creation process. Two different methods for hyperparameter optimization were tested. The first was a hybrid approach of randomized search and grid search; the second used a genetic algorithm for optimization. The genetic algorithm was found to achieve slightly better results and was employed for optimizing all ML-models.

Table 1: Summary of Machine-Learning Classifier Performances **Lilian:** Is this just the top-level classifier? You need something providing this sort of info for the final selection for the binary classifiers. What is multi-/single-layer? What is 'First-layer weighted'? Is it needed given that i is always Yes for Multi and No for Single? What is 'SVC'? Is 'Soft Voting' the same as 'Voting(soft)'? What are 'Basic' features?

Features	Feature Selection Method	Parameter Optimization Method	Multi- / Single-layer	First-layer weighted	Classifier Type	MCC
Basic	None	None	Single	No	SVC	0.54
Basic	None	GA†	Single	No	SVC	0.54
All	None	None	Single	No	Random Forest	0.58
Selected	Random Forest	GA	Single	No	Soft Voting	0.59
Selected	Random Forest	GA	Multi	Yes	XGBoost	0.63
Selected	Recursive Feature Elimination	GA	Multi	Yes	XGBoost	0.79
Selected	Recursive Feature Elimination	GA	Multi	Yes	Decision-Tree	0.99
Selected-NL*	Recursive Feature Elimination	GA	Multi	Yes	Voting (soft)	0.92

* No log-file features

† Genetic algorithm

2.3 Machine Learning Model Performance

The overall best final ML-model was composed of several different binary classifiers, (Figure 3) with an extreme gradient boosting (XGBoost) top-layer nominal classifier. Features were selected using a recursive feature elimination algorithm, through which the weakest feature is removed recursively and the model performance is tested. In the final model, 9 features are included: tip_pos, protrusion, length, total_charge, nr_charged, identity, similarity, Hydropathy and Hydropathy_diff (See Table 2). **Lilian:** This appears to be the top level. What about a table showing the features used for each of the binary predictors and its MCC and anything else unique to that classifier (e.g. feature selection method)?

A final MCC value of 0.99 could be achieved for an ML-model using the abYmod log file as input as well as the loop 3D-model file itself. This value slightly dropped to 0.92 if no such log file was given. This is due to the fact that the template sequence abYmod used to generate the 3D-model is unknown in the latter case and therefore, sequence identity, similarity and hydropathy difference cannot be calculated. The energy is also not available. **Lilian:** There is no option to upload the log file on the web site!

The software was tested on a test-set of antibody structures used in the 2014 and 2011 Antibody Modelling Assessments [12, 13]. As the results depicted in Figure 1 show, abYmod generally achieves results similar to, or better than, other modelling programs. However, some outliers with very high RMSD values increase abYmod's RMSD average. The predictor in this work would aim to identify such outlier 3D-models.

3 Methods

3.1 Computing

All machine learning, feature selection and hyperparameter optimization algorithms were implemented in Python. The Scikit-learn library was used for training ML-models, the Yellowbrick[31] 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.

3.2 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. Rows that contained any null values were removed from the dataset (11 rows in total).

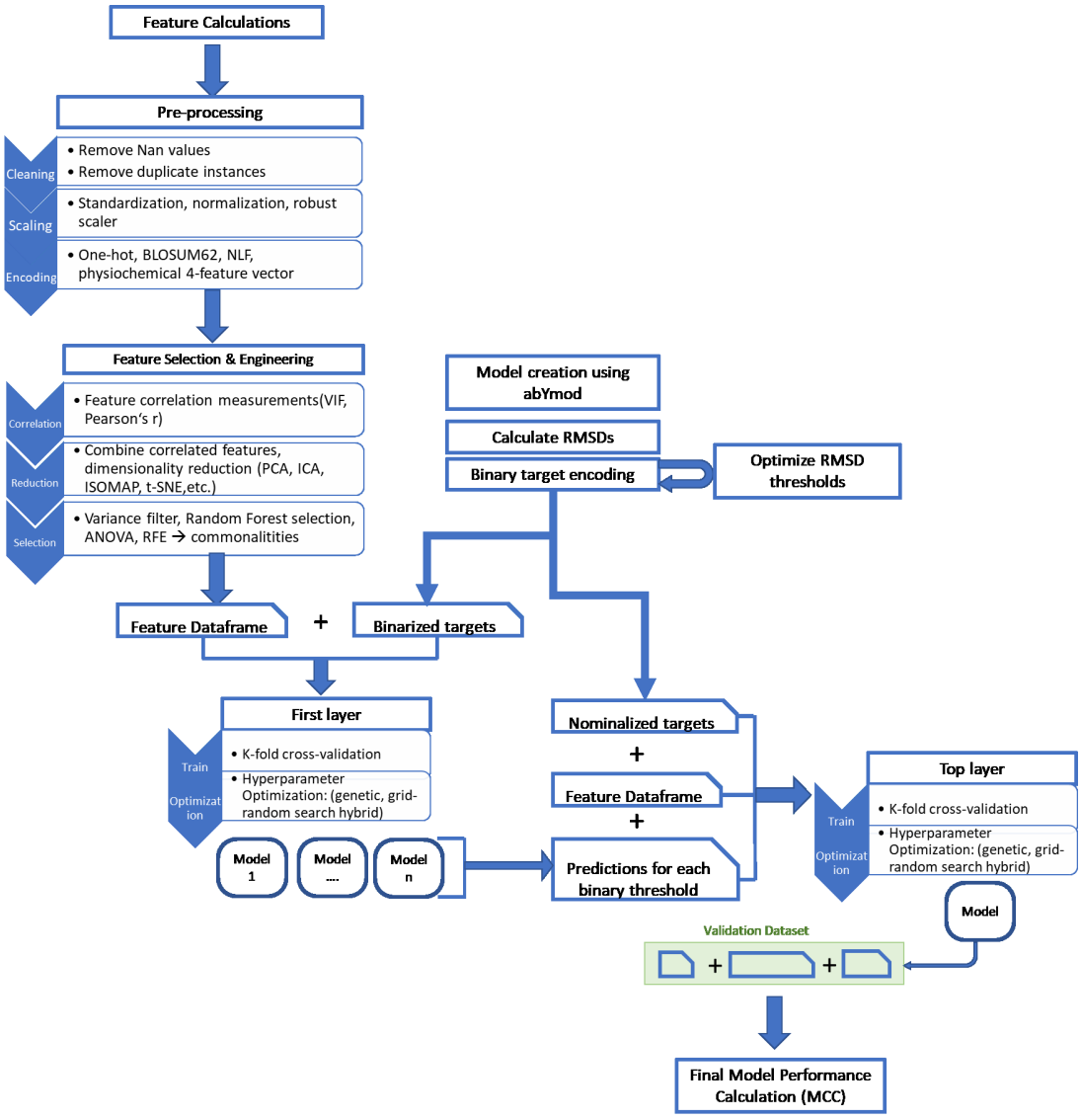

Figure 3: Simplified pipeline for creating the final machine learning model that will predict 3D-model quality by giving its RMSD range. **Lilian:** Make Black and WHite!

3.2.1 Duplicate Screening

Using AbDb's redundancy information it was ensured that no antibodies were present in the dataset more than once.

3.2.2 Scaling

Normalization and Standardization were tested as scaling methods. In normalization the range of the data is fixed between 0 and 1, while in Standardization the data is re-scaled to fit a Gaussian distribution. Both approaches are greatly influenced by outliers and, ideally, such datapoints are 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 was not a viable option. A robust scaler[**?**] **Lilian:** Citation 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. **Lilian:** So which of these scaling methods did you actually use in the end?!

3.2.3 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 were employed: Principal Component Analysis (PCA), Independent Component Analysis (ICA), projection-based methods (t-SNE, Isomap). Three components were used as features. PCA was found to be the most effective dimensionality reduction method.

3.2.4 Physiochemical Feature Encoding

Martin and Abhinandan[29] introduced an encoding using four physiochemical features: the total number of sidechain atoms; the number of sidechain atoms in the shortest path from the Cα to the most distal atom; the Eisenberg consensus hydrophobicity[32]; the charge (using +0.5 for histidine).

NLF-encoding [28] uses multiple physicochemical properties as described by Kawashima *et al.*[**?**] and transforms them using a non-linear Fisher transform (NLF, similar to a PCA) for dimensionality reduction to produce a vector of length 19.

3.3 Dataset-splitting

The final ML-model was evaluated using a test set, separated from the training set at the start in a 30/70 split. The performance of all individual sub-ML-models of the first layer was determined using stratified K-folds cross-validation (K=10) as the dataset is imbalanced, being skewed towards lower RMSD values[33, 34]. The method is different from normal K-folds cross validation as it uses stratified sampling, which is also random, but selections are made to represent class imbalance. This ensures each class is represented, as the percentage of samples for each class is preserved.

3.4 Machine Learning Model Assessment

ML-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 Matthews Correlation Coefficient (MCC)[35] is deemed the most informative, taking the ratios of the four confusion matrix categories into account and 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.[36]

Lilian: The data are missing — it would be good to have some comparison of the performance of the individual predictors and discussion of Table **??**.

3.5 Feature Calculations

Lilian: There is no text in this section!

Table 2: A summary of how different feature values were calculated.

Feature Name	Description	Method of Calculation
Sequence	Amino acid sequence of CDR-H3	Sequence is given in one-letter amino acid codes
Length	Number of residues in CDR-H3	The number of residues are counted
Sequence Identity	Sequence identity of template loop ($SeqA$) and target loop ($SeqB$).	Calculated by abYmod
Sequence Similarity	Sequence similarity of template loop ($SeqA$) and target loop ($SeqB$).	Calculated by abYmod
Loop Protrusion	Distance of loop residue farthest away from the loop base	See Figure 4
Protruding residue	Amino acid code of the most protruding loop residue	See Figure 4
Charge	Total charge of the loop	Sum of charges of all residues in loop
Charge difference	Difference in total charge compared with template sequence	Difference between the two summed changes
Hydrophobicity	Mean Hydrophobicity values of loop	Based on Eisenberg consensus values
Hydrophobicity difference	Sum of absolute differences between loop sequence and template loop	Based on Eisenberg consensus values
Accessibility	Total and average accessibility for the loop	Lee and Richards method implemented using 'pdbsolv' from BiopTools
Sidechain Accessibility	Total and average side-chain accessibility for the loop	Lee and Richards method implemented using 'pdbsolv' from BiopTools
Relative Accessibility	Total and average relative accessibility for the loop	Lee and Richards method implemented using 'pdbsolv' from BiopTools

Relative Sidechain Accessibility	Total and average relative side-chain accessibility for the loop	Lee and Richards method implemented using 'pdbsolv' from BiopTools
'Happiness'	Happiness score, taking accessibility and hydropobicity into account. If a residue is 'happy' it will not be a buried hydrophilic or a surface hydrophobic residue	Hydrophobicity values are normalized to a range of -1 to +1. Mean accessibility values are calculated as above. If hydrophobicity of loop is < 0: $Happiness = 1 + Hydrophobicity(1 - Accessibility)$ Otherwise: $Happiness = 1 - (HydrophobicityAccessibility)$
Number of Contacts	Number of $\leq$ 3.5Å mainchain or sidechain contacts made by residues of the loop Contacts made with residues within and outside the loop are counted separately and as a total. The ratio of inside *vs.* outside is also calculated.	Modified version of 'rangecontacts' from BiopTools.
Energy	Potential energy of the model.	Calculated by Gromacs during the energy minimization step in abYmod.
Lowest BLOSUM62 Scoring Residue Pair	Each possible residue pair in CDR-H3 is scored by their BLOSUM 62 score. The lowest scoring pair's BLOSUM62 value is combined with their residue separation to form the metric.	Separation is the nuber of residues between the worst residue pair (i.e. the lowest BLOSUM62 score achieved by a residue pair), the metric is calculated as: $WorstBLOSUM = -log_2(separation)(worstscore)$

Lilian: You need more description of the BLOSUM separation metric in the methods

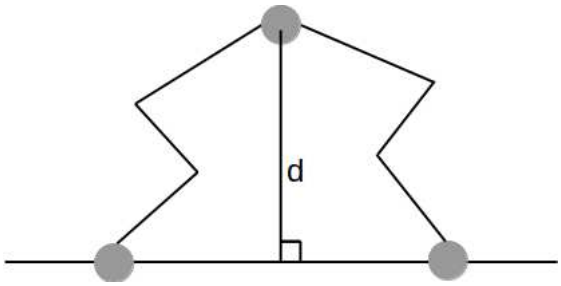

Figure 4: Diagram visualizing the process underlying the protrusion calculation. First, the base residues (i.e. H95 and H102, shown as red spheres) of the CDR-H3 (grey circles) are identified. Then, a line is drawn between the two Cα atoms of these residues. The distance of the Cα-atom of each residue in the CDR-H3 loop to this line is calculated (d). The residue which has the greatest distance to the line is output as one-letter amino acid code and used as feature. The distance d in Å is used as the 'protrusion' feature. **Lilian:** This figure needs to be described in the methods!

4 Discussion

The results suggest that our 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 unfavourable. 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 physicochemical 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 ML-models could be found. However, for our top-layer classifier in our final ML-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 ML-model. A stacked ML-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 3D-model has an RMSD value below 2Å, 2Å–4Å or above 4Å.

We are now looking at incorporating the predictor into the antibody modelling process in the selection of high quality CDR-H3 models given a set of potential decoys.

References

[1] Robert M. MacCallum, Andrew C.R. Martin, and Janet M. Thornton. Antibody-antigen interactions: Contact analysis and binding site topography. *Journal of Molecular Biology*, 262:732–745, 10 1996.

[2] Ruei Min Lu, Yu Chyi Hwang, I. Ju Liu, Chi Chiu Lee, Han Zen Tsai, Hsin Jung Li, and Han Chung Wu. Development of therapeutic antibodies for the treatment of diseases. *Journal of Biomedical Science*, 27:1–30, 1 2020.

[3] Lisa Urquhart. Top companies and drugs by sales in 2020. *Nature Reviews Drug Discovery*, 4 2021.

[4] K. R. Abhinandan and Andrew C.R. Martin. Analyzing the 'degree of humanness' of antibody sequences. *Journal of Molecular Biology*, 369:852–862, 6 2007.

[5] Bissan Al-Lazikani, Arthur M. Lesk, and Cyrus Chothia. Standard conformations for the canonical structures of immunoglobulins. *Journal of molecular biology*, 273:927–948, 11 1997.

[6] Benjamin North, Andreas Lehmann, and Roland L. Dunbrack. A new clustering of antibody CDR loop conformations. *Journal of Molecular Biology*, 406:228–256, 2 2011.

[7] Jessica A. Finn, Julia Koehler Leman, Jordan R. Willis, Alberto Cisneros, James E. Crowe, and Jens Meiler. Improving loop modeling of the antibody complementarity-determining region 3 using knowledge-based restraints. *PLOS ONE*, 11:e0154811, 5 2016.

[8] Cristian Regep, Guy Georges, Jiye Shi, Bojana Popovic, and Charlotte M. Deane. The H3 loop of antibodies shows unique structural characteristics. *Proteins: Structure, Function and Bioinformatics*, 85:1311–1318, 7 2017.

[9] John L. Xu and Mark M. Davis. Diversity in the CDR3 region of V(H) is sufficient for most antibody specificities. *Immunity*, 13:37–45, 2000.

[10] Elvin A. Kabat, Tai Te Wu, Harold M. Perry, Kay S. Gottesman, and C. Foeller. *Sequences of Proteins of Immunological Interest*. U.S. Department of Health and Human Services, Fifth edition, 1991.

[11] K. R. Abhinandan and Andrew C.R. Martin. Analysis and improvements to kabat and structurally correct numbering of antibody variable domains. *Molecular Immunology*, 45:3832–3839, 8 2008.

[12] Juan C. Almagro, Mary Pat Beavers, Francisco Hernandez-Guzman, Johannes Maier, Jodi Shaulsky, Kenneth Butenhof, Paul Labute, Nels Thorsteinson, Kenneth Kelly, Alexey Teplyakov, Jinquan Luo, Raymond Sweet, and Gary L. Gilliland. Antibody modeling assessment. *Proteins: Structure, Function and Bioinformatics*, 79:3050–3066, 11 2011.

[13] Juan C. Almagro, Alexey Teplyakov, Jinquan Luo, Raymond W. Sweet, Sreekumar Kodangattil, Francisco Hernandez-Guzman, and Gary L. Gilliland. Second antibody modeling assessment (AMA-II). *Proteins: Structure, Function and Bioinformatics*, 82:1553–1562, 2014.

[14] Aroop Sircar, Eric T. Kim, and Jeffrey J. Gray. RosettaAntibody: Antibody variable region homology modeling server. *Nucleic Acids Research*, 37:W474–W479, 2009.

[15] Arvind Sivasubramanian, Aroop Sircar, Sidhartha Chaudhury, and Jeffrey J. Gray. Toward high-resolution homology modeling of antibody Fv regions and application to antibody-antigen docking. *Proteins: Structure, Function, and Bioinformatics*, 74:497–514, 2 2009.

[16] Jinwoo Leem, James Dunbar, Guy Georges, Jiye Shi, and Charlotte M. Deane. ABodyBuilder: Automated antibody structure prediction with data–driven accuracy estimation. *mAbs*, 8:1259–1268, 10 2016.

[17] Rosalba Lepore, Pier P. Olimpieri, Mario A. Messih, and Anna Tramontano. PIGSPro: Prediction of immunoglobulin structures v2. *Nucleic Acids Research*, 45:W17–W23, 7 2017.

[18] Michael Schantz Klausen, Mads Valdemar Anderson, Martin Closter Jespersen, Morten Nielsen, and Paolo Marcatili. Lyra, a webserver for lymphocyte receptor structural modeling. *Nucleic Acids Research*, 43:W349, 7 2015.

[19] Brennan Abanades, Guy Georges, Alexander Bujotzek, and Charlotte M. Deane. ABlooper: fast accurate antibody CDR loop structure prediction with accuracy estimation. *Bioinformatics*, 38:1877–1880, 3 2022.

[20] Clara T. Schoeder, Samuel Schmitz, Jared Adolf-Bryfogle, Alexander M. Sevy, Jessica A. Finn, Marion F. Sauer, Nina G. Bozhanova, Benjamin K. Mueller, Amandeep K. Sangha, Jaume Bonet, Jonathan H. Sheehan, Georg Kuenze, Brennica Marlow, Shannon T. Smith, Hope Woods, Brian J. Bender, Cristina E. Martina, Diego Del Alamo, Pranav Kodali, Alican Gulsevin, William R. Schief, Bruno E. Correia, James E. Crowe, Jens Meiler, and Rocco Moretti. Modeling immunity with Rosetta: Methods for antibody and antigen design. *Biochemistry*, 60:825–846, 3 2021.

[21] Brian D. Weitzner and Jeffrey J. Gray. Accurate structure prediction of CDR H3 loops enabled by a novel structure-based C-terminal constraint. *The Journal of Immunology*, 198:505–515, 1 2017.

[22] Yoonjoo Choi and Charlotte M. Deane. FREAD revisited: Accurate loop structure prediction using a database search algorithm. *Proteins: Structure, Function, and Bioinformatics*, 78:1431–1440, 5 2010.

[23] Andrew C.R. Martin and Janet M. Thornton. Structural families in loops of homologous proteins: Automatic classification, modelling and application to antibodies. *Journal of Molecular Biology*, 263:800–815, 11 1996.

[24] Saba Ferdous and Andrew C R Martin. AbDb: antibody structure database — a database of pdb-derived antibody structures. *Database*, 2018, 1 2018.

[25] Martin Christen Frolund Thomsen and Morten 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.

[26] Mark C. Shaner, Ian M. Blair, and Thomas D. Schneider. Sequence logos: A powerful, yet simple, tool. *Proceedings of the Annual Hawaii International Conference on System Sciences*, 1:813–821, 1993.

[27] S. Henikoff and J. G. Henikoff. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences of the United States of America*, 89:10915, 11 1992.

[28] Loris Nanni and Alessandra Lumini. A new encoding technique for peptide classification. *Expert Systems with Applications*, 38:3185–3191, 4 2011.

[29] K. R. Abhinandan and Andrew C.R. Martin. Analysis and prediction of VH/VL packing in antibodies. *Protein Engineering, Design and Selection*, 23:689–697, 9 2010.

[30] Tianqi Chen and Carlos Guestrin. XGBoost: a scalable tree boosting system. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, volume 13-17-August-2016, pages 785–794. Association for Computing Machinery, 8 2016.

[31] Benjamin Bengfort, Rebecca Bilbro, Paul Johnson, Philippe Billet, Prema Roman, Patrick Deziel, Kristen McIntyre, Larry Gray, Anthony Ojeda, Edwin Schmierer, Adam Morris, and Molly Morrison. Yellowbrick v1.3, 2 2021.

[32] David Eisenberg, Robert M. Weiss, Thomas C. Terwilliger, and William Wilcox. Hydrophobic moments and protein structure. *Faraday Symposia of the Chemical Society*, 17:109–120, 1 1982.

[33] Damjan Krstajic, Ljubomir J. Buturovic, David E. Leahy, and Simon Thomas. Cross-validation pitfalls when selecting and assessing regression and classification models. *Journal of Cheminformatics*, 6:10, 3 2014.

[34] Ron Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection, 1995.

[35] Davide Chicco and Giuseppe Jurman. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics*, 21:6, 1 2020.

[36] Giuseppe Jurman, Samantha Riccadonna, and Cesare Furlanello. A comparison of MCC and CEN error measures in multi-class prediction. *PLoS ONE*, 7:41882, 8 2012.