

A Model-based Approach for Mining Membrane Protein Crystallization Trials

Sitaram Asur* Srinivasan Parthasarathy* Pichai Raman† Matthew Eric Otey*

Abstract

Crystallization has been proven to be an essential step in macromolecular structure verification. Unfortunately, the bottleneck is that the crystallization process is quite complex. It can take any time from weeks to years to obtain diffraction-quality crystals, even under the right conditions. Other issues include the time and cost involved in taking trials and the presence of very few positive samples in a wide and largely undetermined parameter space.

Any help in directing scientists' attention to the hot spots in the conceptual crystallization space would lead to increased efficiency in crystallization trials. This work is an application case study on mining membrane protein crystallization trials to predict novel conditions with a high likelihood of leading to crystallization. We use suitable supervised learning algorithms to model the data-space and predict a novel set of crystallization conditions. Our preliminary wet laboratory results are very encouraging and we believe this work shows great promise. We conclude with a view of the crystallization space, based on our results, which should prove useful for future studies in this area.

1 Introduction

The study of membrane proteins is one of prime importance in all branches of proteomics. They are integral to cellular function acting as mediators between the cell and its environment. These remarkable proteins play important roles in energy transduction, cell signaling, and maintaining the integrity of the cells internal environment. However, there is still very little known about their function since many of their structures remain unknown. Since structure leads to function, discovering the structure of these remarkable proteins will help lead to understanding their function and will aid in creating drugs for a host of diseases. However, compared to sol-

uble proteins, there is a dearth of membrane proteins with known structure. In order to obtain the structure (with high resolution) of a protein, many scientists rely on the powerful technique of X-ray diffraction, which requires a protein crystal. However, obtaining good quality crystals of membrane proteins is an arduous task when compared to water soluble proteins. This is due to the fact that membrane proteins typically get trapped as an intractable aggregate during the crystallization process, limiting access to their structure [5].

The science of crystallization is still quite preliminary and there is very limited knowledge on what actually causes crystallization to occur. Hence, crystallographers are forced to systematically sift through a wide parameter space (for example, physio-chemical, biophysical, biological parameters) to grow crystals with good diffraction characteristics. This trial-and-error approach (not unlike searching for needles in a haystack) has been shown to be difficult due to the phenomenally large cost and time requirements to perform the crystallization experiments.

As a consequence, the set of conditions currently employed is based almost entirely on earlier experimental successes [16]. These conditions, while not random, are not specifically designed for a particular protein. From a statistical perspective, this amounts to over-sampling certain regions in the multi-dimensional crystallization space. Such screens represent what is known as a sparse matrix. These sparse matrices assume that different proteins will crystallize under the same conditions. This assumption is not completely valid [16]. Therefore, researchers have attempted to vary one or two of the chemical components from successful combinations to obtain new favourable conditions. Unfortunately this has met with mixed success after requiring many trials to get a few good crystals.

The process of protein crystallization involves using a protein/lipid membrane that is mechanically mixed and brought to correct water content and temperature. At this stage, suitable chemical reagents are added and protein crystals are then allowed to form. The reagents can be grouped into classes such as precipitant, additive, buffer, and detergent. The temperature, type

*Department of Computer Science and Engineering, Ohio State University

†Department of Biophysics, Ohio State University

and concentration of the lipid and reagents are of utmost importance in protein crystallization. These physio-chemical conditions and reagents together form the crystallization screen. Additionally, the current hypothesis is that the optimum conditions (those that cause the best resolution crystals) are protein-specific. Overall, it is fairly difficult to obtain crystals of any quality. With this in mind, it stands to reason that if we can produce a greater number of conditions that do in fact bring about crystals, we can assume some percentage of them will have crystals of good diffraction quality. Obtaining multiple crystals is also good since this adds to robustness and reliability of the results. Specifically, the end goal is to develop a screen (with different crystallization conditions) that is optimal for a particular protein and maximizes the number of high-resolution crystals.

In this case study, we consider the crystallization space to be broadly classified into three areas, mapped to classes 0, 1 and 2. These are analogous to the three levels, clear, precipitate and crystalline, proposed by Kimber *et al.* [9]. The ‘hot spots’ are the areas that yield protein crystals (class 2). A large part of the space consists of clear or ‘no-hit’ areas that are not conducive for the production of crystals (class 0). There are also areas that do not yield crystals but produce protein precipitates (class 1).

Some researchers [16, 17] discuss the virtues of random sampling on the crystallization space. We believe a more structured and intelligently designed approach will lead to success. In this work, we examine the use of suitable supervised learning algorithms to examine relationships or correlations between the input parameters (protein properties, crystallization conditions) and model the response output (crystals, precipitates or no crystals) for existing trials and then close the loop to identify interesting ‘hot spots’ (areas with high potential for yielding good quality crystals) in the space for future trials. We use the model learnt to predict the outcomes for a randomly sampled set of conditions. We then perform stratified sampling based on our model, incorporating physio-chemical constraints, to obtain new sets of conditions to test in the wet laboratory. Our premise is that this method is more structured and a more profitable option than random sampling. Preliminary wet lab experiments seem to validate this premise. Our results also allow us to hypothesize a view of the crystallization space. We provide details of this hypothesis at the end of the paper. To summarize, the main contributions of this paper are:

- Application of classification algorithms to model the protein crystallization space.

- Model-based prediction and stratified sampling to obtain novel conditions with high probability of yielding crystals.
- A Hypothetical view of the crystallization space based on our results.

2 Background on Protein Crystallization

In this section, we provide some background on the crystallization process and discuss some related work in this area.

2.1 Cubic Phase (*In meso*) Crystallization

Crystallization is essentially a phase separation technique in a thermodynamically stable system, with the favourable outcome being the formation of a crystal. There are a host of techniques currently employed to crystallize proteins. The basis of this project rests on the laurels of a relatively new technique for membrane protein crystallization known as the Cubic Phase or *in meso* method [5]. This is the technique from which all our data is derived. The cubic phase technique is based on the assumption that the protein to be crystallized is initially reconstituted into the lipid bilayer of the cubic phase [5]. The essential steps involved in this technique are adding a protein/lipid membrane that is mechanically mixed and brought to correct water content and temperature to form the cubic phase. At this stage, additives and precipitants are added and protein crystals can then form in a time span that extends from hours to months. This can be done manually or with the aid of a robot for high throughput crystallization. While the technique itself seems straightforward, the way in which the *in meso* method crystallizes proteins is still not well understood. There is a great deal of speculation as to how this method works. The crux of the method rests on the understanding of the peculiar phase behavior of lipids. Lipids have two standard phases, liquid and solid. However, they also possess a third set of phases known as liquid-crystalline phases. These phases represent configurations of the lipid molecules in aqueous medium that arise due to the amphipathic nature of lipids and the hydrophobic effect. A set of lipid phases is shown in Figure 1.

The lipid phases change with the water content and temperature and this is plotted out in a Temperature/Composition (T/C) diagram. An understanding of these phases is of utmost importance since in this method one must achieve the cubic phase, hence utilizing the right proportion of water (to the protein/lipid blend) and the right temperature is key in order to get to the appropriate phase. The idea is that during the mixing process, the proteins start off solubilized in detergent

micelles but then reconstruct into the lipid bilayer with the introduction of dry lipid. The lipidic phase they are thrust into is the cubic phase. These phases (Pn3m, Ia3d, and Im3m) are shown in Figure 1. With the addition of salt the curvature of this phase increases, which in turn causes the protein to leave and associate in a transient lamellar phase, also shown in the figure. The belief is that as proteins leave the lamellar phase, they arrange in a highly ordered fashion and form crystals.

There exist a large number of variables in this technique such as additive structure, additive concentration, detergent type, protein structure, etc. It is not known which of these parameters are instrumental in obtaining a favourable outcome, realizing a crystal. Furthermore, researchers have varying, inconsistent and largely incomplete information as to why proteins crystallize in the first place. As a consequence, the set of conditions scientists currently employ is based almost entirely on earlier successes and the chemicals readily available currently. These conditions, while not random, are not specifically designed for a particular protein. Therefore, the likelihood of getting crystals from these screens for novel proteins is incredibly small.

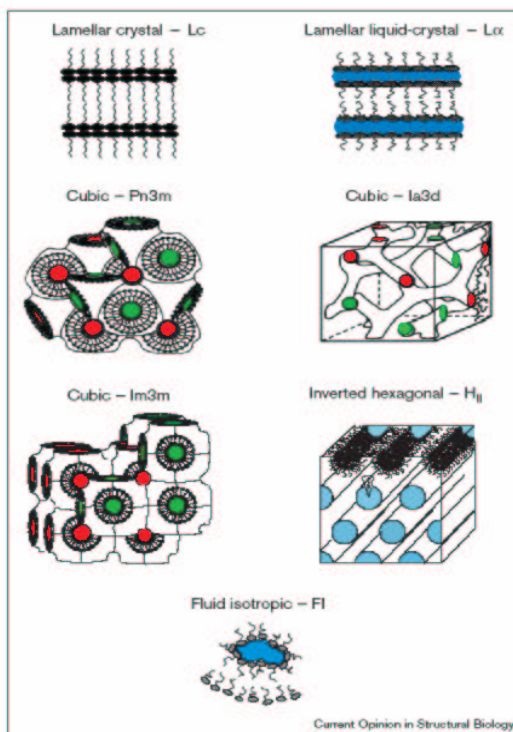


Figure 1: Lipid Phases - Cubic (Pn3m, Ia3d, Im3m), lamellar liquid-crystal (L), and inverted hexagonal (HII) phases are the liquid crystalline phases. Fluid isotropic (FI) is a liquid phase and lamellar crystal (Lc) is a solid phase [5].

2.2 Related Work The work by Samudzi and colleagues [11] postulates that the response surface was composed of a set of disjoint clusters, rather than a single coherent cluster. Subsequently, they apply a clustering algorithm on the Biological Macromolecule Crystallization Database, which is a large collection of successful crystallization trial conditions. Their initial attempt revealed interesting qualitative relationships between recorded parameters but did not yield how this information could be used in the design of future experiments. A limitation of these experiments (along with others at the time), is that the data used consisted of only successful trials.

Several researchers [6, 9, 16], argue convincingly that a comprehensive information repository for crystal growth experiments (both positive and negative trials) is fundamental to the computational analysis of trials. This stored information is necessary to discover general rules or principles underlying the growth process for crystals, as well as to guide the reasoning algorithm for planning experiments. As noted by the above researchers, the application of data mining and knowledge discovery algorithms to such datasets is still in its infancy. Carter and Carter [7] were one of the first to propose the use of statistical sampling techniques for this problem. Segelke [17] assesses crystallization screens in terms of sampling and shows the advantages of random sampling. Others [16] believe that random sampling may not be the best solution since it does not use the available prior knowledge effectively. However, most approaches taken by crystallographers still rely on either random or stratified sampling of the crystallization space. Rupp [16], also argues that in a high throughput environment, with a large number of data points and limited prior knowledge, a semi-automated machine learning/data mining driven approach is absolutely essential. In spite of these works discussing the use of data mining algorithms, to the best of our knowledge there has been no prior work in this direction.

3 Overview of our Approach

The protein crystallization space has been conceptualized as an n-dimensional hypercube [16] with axes represented by the chemical components and other parameters. The various crystallization condition trials are obtained by sampling this space. Our strategy for mining the protein crystallization space is a closed loop consisting of four stages, represented in the flowchart in Figure 2.

- **Experimental Data:** The data obtained from prior experiments is used as training data. This data consists of sets of conditions that have been employed before in crystallization trials. An issue

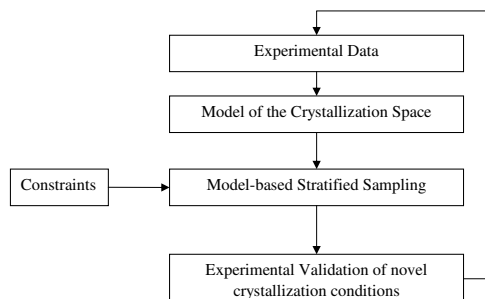


Figure 2: Mining the Protein Crystallization space

with using prior data is that it has been obtained almost completely from the same regions in the crystallization space. These regions have been over-sampled repeatedly. Another issue is the large bias present in the dataset, with a significant majority of the samples resulting in failures. We discuss the characteristics of the dataset in detail in the next section.

- Modeling the space:** The empirical training data is used to build supervised models on the protein crystallization space. We believe that supervised learning algorithms such as classifiers are useful for this problem as they can use the training data and known class values to partition the space efficiently. Hence, we apply traditional classifiers and build an ensemble using the best classifiers to increase the precision of prediction. It is important to note that this approach will initially be limited since the empirical data currently available represents only a few regions. A large amount of the space is presently unknown. However, our strategy is dynamic and incremental. As we iterate, more regions of the space will be added into our training data for modeling. We present details of our modeling technique in Section 5.
- Model-based Stratified sampling with constraints of the condition space:** We use the model of the data-space to lead us to the right regions for sampling, and the classifiers that we trained to predict class values of novel conditions. We perform stratified sampling on the predicted conditions to overcome the over-sampling issue. The objective is to discover new regions in the space that have not been visited earlier and that have high potential for yielding crystals. Our approach is iterative and incremental. At each iteration, we broaden our search space. We use stratified sampling on our results in order to maintain balance.

We also need to minimize the number of conditions to be tested and ensure a high success-rate (reduce false positives). We leverage this by using a relatively strict metric for prediction.

At the same time, we need to consider constraints (physio-chemical, physical and biological) of the crystallization space. These constraints may be a factor of the conditions or internal parameters such as temperature and solubility. We present details of our sampling scheme in Section 6.

- Experimental Wet lab Validation of novel conditions:** Once novel conditions have been discovered, we need to test them experimentally. One of the issues with experimentation is the expense, in terms of time and effort, for each crystallization trial. We validate the sampled conditions, again considering constraints of the crystallization process. The results from this step feeds back into the first step of the next loop. Our experimental validation results are presented in Section 7.

4 Dataset Properties

The initial data that was used to build the models for prediction was a set of screens of 3 proteins - outer membrane Cobalamin transporter (BtuB), bacteriorhodopsin (bR) and LH2 with a set of 3 lipids - 9.9 MAG, 7.7 MAG, and 9.7 MAG and a set of 480 standard conditions that originate from Hampton research, a company that specializes in developing products for biological macromolecular crystallization (<http://www.hamptonresearch.com/>). Each protein/lipid mix (there are 5 combinations out of the possible 9) was put through these conditions on a set of five 96 well plates. Each plate was then manually scored with a number from 0-9, indicating the phase/protein condition. This designation is referred to as the crystal rating.

The chemical conditions include a main buffer, a precipitant and one or more additives. The purpose of the buffer component in a screen is to cover a certain pH range (and thus charge distribution) on the protein, independent of the other components and the pH of the original protein solution. Buffers with different pH values can thus be considered different. There are two major types of precipitants, high molecular weight poly-alcohols (like PEGs) and salts. The additives used may be buffers, precipitants or any chemical that might help crystallization. Each sample in the dataset contains a crystal rating. The crystal ratings are formulated as follows, 0-2 means lamellar or dispersed phase, 3-5 indicates protein precipitate, and 6-9 indicates the

Crystal classes	Number of Samples	Percentage
0	1995	86.7
1	170	7.3
2	136	6

Table 1: Crystal class percentages

formation of crystals. The number of 0's in the dataset are very high and as the rating increases, the number of samples having that value decreases. The number of samples rated 9 is very low. To perform adequate classification, we require a better distribution. Hence, values between 0 and 2 are assigned to class 0, between 3 and 5 are assigned to class 1, and values between 6 and 9 are assigned to class 2. Class 2 is the desired class indicating the formation of crystals.

There are a total of 2,301 trial conditions (5 protein/lipid combinations * 480 conditions each - 99 elements where no data taken) with various protein, lipid, buffer, additive, precipitant combinations. The percentages of the three discretized ratings in the dataset are given in Table 1. It can be seen that data is significantly biased with around 87% of the samples classified as 0. In this work, we treat the data samples as normal categorical data. This is a safe assumption since in this application, two buffers with different pH values can be expected to behave differently. Each sample is a feature vector of size 6 consisting of:

- Protein - Btub, bR or LH2
- Lipid - 9.9 MAG, 7.7 MAG or 9.7 MAG
- Buffer - Eg. 0.1 M Na Acetate pH 4.6
- Main Precipitant - Eg. 0.5 M Magnesium Formate
- Additive - Eg. 2 M Na Chloride
- Class Value - 0, 1 or 2

5 Modeling the Protein Crystallization space

As we mentioned earlier, the protein crystallization space can be represented as an n-dimensional hypercube with axes represented by the chemical components and other parameters. The regions in this space that yield crystals are called 'hot spots'. For a given protein, there exist a large number of conditions which do not lead to precipitates. In some conditions, proteins precipitate but do not form crystals. We use supervised learning (classification) to model the protein crystallization space using the empirical data. We believe that classification is a good method to partition the data space and predict class values for new samples.

5.1 Supervised Learning Algorithms: In this section, we present details of the supervised learning algorithms we use.

Naive Bayes Classifiers: Naive Bayes classifiers are based on Bayes' rule of conditional probability. It uses all attributes and allows them to make contributions to the decision as if they were all equally important and independent of one another. The classifier can be formally defined as

$$(5.1) C(F) = \operatorname{argmax}_c p(C = c) \prod_{i=1}^n p(F_i = f_i | C = c)$$

where c is the class and f_i are the features(attributes).

Decision Tables: Decision table classifiers are rule-based classifiers that are typically used not only for prediction but visualization of the attribute space [15]. A decision table generally has two components, a scheme and a body. The scheme is the list of attributes that are used to predict the class variable. The body consists of a set of assigned values for each attribute in the scheme. The class variables that are of the same type fall into a broad category called cells. The dataset is sorted by the broadest possible field (or field with the least number of attribute types). From here, the set of rows with the same type in each attribute are grouped together in a cell. Generally, the rules of constructing a decision table involve mapping all possible combinations of the attribute space to class values. This ensures that every single input vector will have been assigned some designation. The program then simply runs through the table with the input vector to determine which class variable is appropriate.

Random Forests: A Random Forest is an ensemble classification technique which is popular due to its high accuracy. In this method, several classification trees are constructed by sampling with replacement from the original training data. In order to find the best split at a node in the tree, m random attributes are chosen and the one with the best split among them is used. Furthermore, the trees that are constructed are not pruned. Classification is done using each tree to separately classify the test data. Finally, the majority of the votes from each tree is chosen to be the prediction on the test sample.

Classification Based on Associations: This is a technique based on association rule-based classification [8], which can be used effectively on discrete datasets. Association rules identify collections of data attributes that are statistically related in the underlying

data. An association rule is of the form $X \rightarrow Y$ where X and Y are disjoint conjunctions of attribute-value pairs. The support of the rule is the observed frequency of X and Y , $Pr(X, Y)$. The confidence of the rule is the observed frequency of Y given X , $Pr(Y|X)$. Given a database of transactions, a minimal confidence threshold, and a minimal support threshold, the goal of association rule mining is to find all association rules whose confidences and supports are above the corresponding thresholds. In this case, each row of the dataset can be considered to be a separate transaction, with the values in each column being the items for that transaction. The Apriori algorithm [2] is a commonly used algorithm for mining association rules. The algorithm discovers rules for dependencies between the elements that are frequent, i.e, satisfy some minimum support and minimum confidence constraints. We then use these frequent rules to perform classification. We have tried different values of minimum support and confidence thresholds. We find that using low support (5%) and high confidence (60%) thresholds are adequate for discovering association rules even for large datasets.

Nearest Neighbor Voting: In this technique, we build separate nearest-neighbor classifiers for each attribute. For each attribute i with value v_i , we identify the k rows in the dataset that contain a value closest to the value v_i . Then we use the class values predicted by these k rows to compute a single vote value. We take the mode of the k classifications as the single vote value. This process is repeated for each attribute (v_i) resulting in several single vote values. To tally the vote values, we once again use the mode to predict the class of that sample.

Support Vector Machines: Support Vector Machines (SVMs) [13, 18] are based on the concept of decision planes that define decision boundaries. A decision plane separates a set of objects having different class memberships. Support Vector Machines are particularly suited to handling classification tasks that involve complex decision planes, as opposed to linear classification. They work by constructing hyperplanes in a multidimensional space. The classifier maps the input vectors to a higher dimensional space, after which it finds a linear separating hyperplane with the maximal margin in the high-dimensional space.

For our experiments, we used two popular SVM packages, SVMLight [13] and BSVM [12]. SVMLight works efficiently for two-class problems while BSVM performs well for multi-class classification problems. We used the default linear kernel function in our

experiments.

PNRule: PNRule, proposed by Joshi et al [14], is a rule-based classifier designed to handle skewed class distributions. PNRule works in two phases. In the P phase, it discovers positive rules that cover the target class. In the N phase, it generates rules on the negative class to eliminate false positives from the samples covered in the P phase. The rules are based on single attribute values. The test samples are run through the positive and negative rules. Accordingly, a test sample is classified positive only if it is found to satisfy a positive rule and no negative rules.

5.2 Metric: Since our goal is to discover novel trial conditions using classification, we are really interested in measuring how many of the positively predicted samples are actually positive. In other words, the precision of prediction is the key. If pos_{pred} are the samples that are predicted to be positive and pos_{actual} are the samples that are actually positive, the precision is given by

$$(5.2) \quad Precision = \frac{\|pos_{pred} \cap pos_{actual}\|}{\|pos_{pred}\|}$$

We would like to point out that the accuracy of classification in this case is not particularly useful. This is due to the fact that a naive classifier that predicts class 0 for every sample will yield a high accuracy of 87% (due to the significant bias in the dataset).

5.3 Classification Results

5.3.1 The Bias Problem: We split up the crystallization dataset randomly into training and test sets. As we mentioned earlier, the crystallization dataset consists of a large majority (87%) of negative samples. This causes significant bias and affects classifiers such as CBA and Nearest Neighbor and causes all the predictions to be of class 0.

The problem of learning with biased data has been addressed in several works in the data mining literature. As we mentioned earlier, PNRule, the rule-based classifier was proposed to handle skewed class distributions. We have implemented PNRule but find that for our dataset, the negative rules we discover cover all the samples. Hence, we cannot obtain any positive predictions. We have tried varying the negative rules based on recall, as was suggested, but do not obtain any improvement in the results.

The main methods suggested for balancing skewed training data include downsampling the non-target class, upsampling the target class and generating new samples of the target class. SMOTE [10] is a technique that generates new samples of the target class using existing positive samples that are close to each other.

This is possible only if there is a valid distance metric to find nearest neighbors in the set of samples, which is not true for our data. Also, in our data, the minority class is very sparse with respect to the majority class. Hence, the application of SMOTE results in a mixture of the classes (over-generalization) which is very hard to separate. Batista et al [4] evaluated different techniques for balancing training data and found that random over-sampling of the target class performs well in most cases. Using this notion, we develop an ensemble approach to eliminate the bias problem. We generate several random sub-samples of the negative class and merge each of them with over-sampled positive examples. This results in several balanced subsets of the original data. We then train our classifiers on each sub-dataset separately and use each of them to predict the class values of the test data. Finally, we use majority vote decision fusion to combine the predictions of each of the individual classifiers. We obtain much better results using this approach, although it does not completely eliminate the bias problem.

5.3.2 3-class Prediction: We predict class values using all the classifiers we reviewed earlier. We perform 5-fold cross-validation. The best individual classifier is the Decision Table Classifier with a precision of 58%. The other classifiers mis-classified several samples of class 1 as class 2. The results are presented in Table 2.

5.3.3 2-class Prediction: Although, we obtain a precision of 58% for the Decision Table Classifier, most of the classifiers had trouble separating the samples in the 3-class case. An interesting observation we made with the results is that a large number of samples belonging to class 1 were falsely identified as class 2. We leverage this observation as follows. In the training phase, we consider all samples of class 1 to be of class 2. Although this does not remove the bias, it increases the percentage of samples belonging to class 2 from 6% to 13%. We believe that this leads to better partitioning by the classifiers. We therefore predict once again on the test sets, using this assumption.

We find the improvement in precision to be substantial. Every individual classifier is found to predict more accurately under this scenario. The results of 5-fold cross-validation by all the classifiers are presented in the 3rd column in Table 2. CBA produced dramatic improvement (15% to 65%). The Naive Bayes technique also improved phenomenally (although its performance is still below par). The three best classifiers are, in order, Decision Table, CBA and Support Vector Machines.

Algorithm	Precision (3-class)	Precision (2-class)	Percentage Improvement
Naive Bayes	4%	44%	1000%
Decision Table	58%	72%	24.14%
Random Forest	35%	48%	37.14%
Bagging	52%	60%	15.38%
CBA	15%	65%	333.33%
NNV	12%	21%	75%
SVM	39.5%	65%	64.5%

Table 2: Individual Classifier Results

5.3.4 Ensemble Classification: We constructed an ensemble classifier using these three individual classifiers to improve the precision of prediction. If x_i is the test sample, and p_j where $j=1..3$ are the the predictions from the three individual classifiers, the ensemble prediction is given by

$$Ens(x_i, p_1, p_2, p_3) = \begin{cases} 2 & \text{if } p_1 = p_2 = p_3 = 2; \\ 0 & \text{if } p_1 = 0 \cup p_2 = 0 \cup p_3 = 0. \end{cases} \quad (5.3)$$

When we use the ensemble classifier to predict values for the test-sets we obtain a precision close to 100%. However, the number of positively predicted samples is very low (5-10). This is due to our constraint that all three individual classifiers need to predict a positive result for a sample to be classified positive. This assumption can be relaxed. Accordingly, we proceed to choose samples which any two of the classifiers predicted as positive. This gives us a larger number of positive samples (20-30) and a precision of 86% on the test data after cross validation.

6 Model-based Stratified Sampling

As mentioned earlier, our goal in this work is not only to model the crystallization space but to discover novel regions to sample for positive conditions. Earlier works focussed entirely on randomly sampling the crystallization space. Random sampling alone does not ensure success. We believe a more intelligently designed approach can yield better performance. Random sampling maximizes the variance of the data space. We propose a more principled approach, applying domain knowledge to sampling, similar to ideas proposed by Bailey-Kellogg and Ramakrishnan [3].

We employ a two-stage stratified sampling technique in this regard. In the first step, we generate a large number of random samples. We ensure that these samples are sufficiently different from the samples in the training data. The samples are then pruned using the help of a domain expert, to enforce physio-chemical constraints such as compatibility between chemicals. We use the classifiers with the best performance to predict the class values of these samples based on the training data. The classifiers partition the samples into regions

of class 0, 1 and 2. In the second step, we perform stratified sampling on the results of the classifiers. We propose two schemes that can be used for stratified sampling, depending on the context:

- In the 2-class scheme, we over-sample regions that are predicted to be class 2 or class 1 and under-sample the region that is predicted to be class 0.
- In the 3-class scheme, we over-sample regions predicted to be class 2 and under-sample the other two classes. The proportion of samples chosen that are of class 0 is less than the proportion of samples chosen that belong to class 1.

The sampled conditions are then fed to an automated robot that conducts the experiments using these conditions. During this process, constraints on internal parameters, such as temperature and solubility (K_{sp} values), are applied. Currently, this process is manually done by domain experts. We use two kinds of filters in the process, one to remove incompatible chemical combinations and improbable factor levels (excessive precipitant concentrations, high PEG concentrations etc) and the other to remove conditions that are not very novel.

7 Experimental Wet lab Validation

We proceeded to generate a large set of random samples using the conditions from the Hampton kit. We used the protein BtuB and the lipid was 7.7 MAG. The rest of the conditions were randomly generated from the data, i.e., a random buffer, precipitant, and additive were chosen from the set of unique elements in the Hampton kits. Each vector was compared with the 480 Hampton kit conditions to ensure there were no duplicates.

Since the time for experimentation is a major bottleneck in this technique, we chose to perform some preliminary experiments using the predictions of a single classifier. We chose the decision table classifier, since it outperformed all the others for 3-class classification. The set of feature vectors was run through the algorithm and each sample was assigned a crystal rating. Samples were chosen to for experimental validation using the 3-class scheme.

We obtained 96 conditions and conducted crystallization experiments in our laboratory using these. We used the buffers, precipitants, and salts available in the Hampton Research kit. The rest of the required reagents were prepared to specified concentrations and pH (when applicable) in house. The protein was combined with the lipid to form the cubic phase using mechanical mixing. A robot was then used to mix the reagents and dispense both the well conditions and the

protein/lipid combination. All screens were set in 96 well plates which were scanned at various time intervals for crystals using a light microscope.

Since the decision table classifier yields a precision of 58% for the 3-class problem, taking account for experimental errors, we expected the upper bound on the number of crystals obtained to be around 40. True to form, there were 37 conditions of the 96 that were found to produce crystals. Among the hits, the crystals ranged in size from 50 microns to 90 microns (Figures 3 and 4). Interestingly, a large number of the negative trials yielded protein precipitates (class 1).

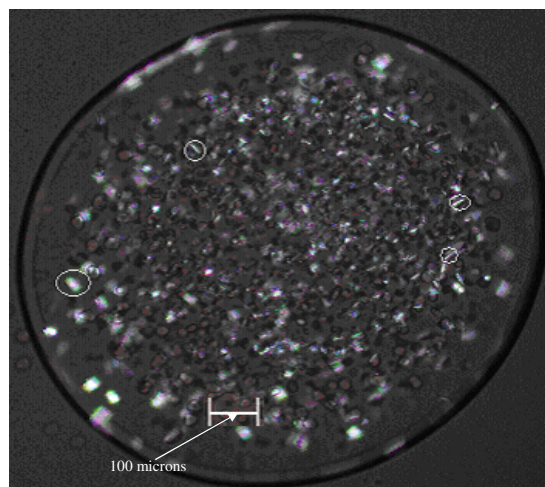


Figure 3: BtuB crystals produced from decision table crystallization. Crystals are circled in white

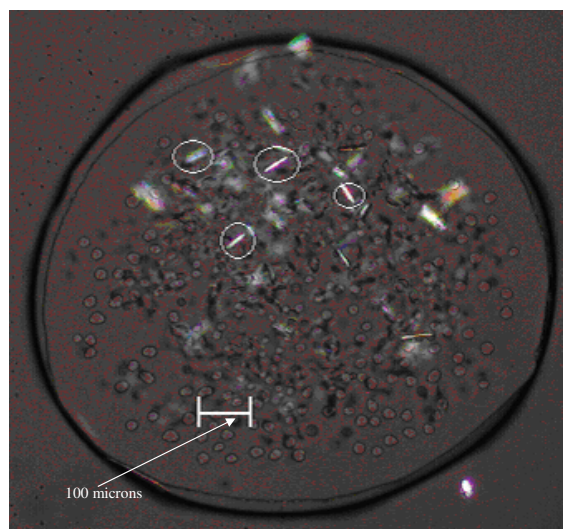


Figure 4: More BtuB crystals produced from decision table crystallization. Crystals are circled in white

We tested our ensemble 2-class classifier on this set of experimentally determined samples. When we

used the ensemble on the 96 conditions, we obtained 13 positive predictions. Since the precision of the ensemble classifier was 86%, once again considering experimental errors, we expected to get crystals in at most 8 or 9 of these trials. We were pleased to get crystals in 8 of the 13 trials. Furthermore, we were pleasantly surprised to obtain precipitates (class 1) in the negative trials. This is equivalent to a 100% precision in the 2-class scenario. Given that the number of crystals generally obtained from crystallization screens are very few and the trials typically consume a large amount of time, our results are useful. To illustrate this, we compare the average number of positive samples from each of the 5 Hampton protein-lipid combinations with our results using the samples predicted by the Decision Table classifier. The difference can be observed in Figure 5. H1-H5 represent the 5 protein-lipid combinations in the Hampton screen kit. The average number of positive samples/96 wells for all 5 Hampton screens is around 5, whereas we predict 37 positive conditions using just one screen of 96 conditions. We would also like to point out that the conditions we obtained were adequately novel when compared to the conditions in the Hampton kit which over-sampled the same space.

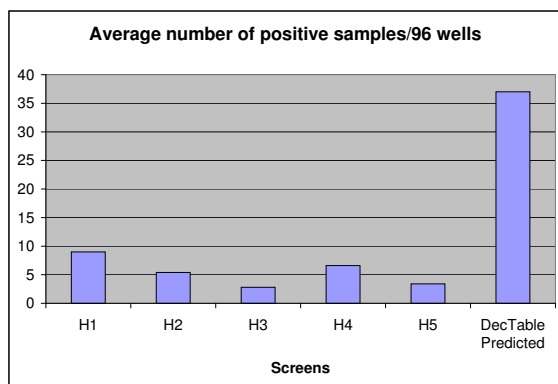


Figure 5: Comparison between the Hampton screens and our predicted samples

To follow-up, we generated a very large set of random vectors and used our ensemble classifier to predict on them. We obtained close to 200 conditions using the predictions from the classifier. We have set up trials on these conditions and expect to obtain favourable results.

7.1 Discussion Our preliminary results are encouraging. However, it is important to note some limitations to this study. While applying data mining algorithms to build models for crystallization conditions prediction is a good approach and provides many benefits, we are

limiting ourselves by using the Hampton crystallization dataset which does not represent a truly random or evenly distributed sample of the crystallization space.

For instance, the Hampton kit contains crystallization trials information for three proteins. Using this information, we can predict conditions for only these three proteins. Since different proteins react differently to the same condition, it may not be practical to make predictions for other proteins. The same holds true for the chemicals that have been used. We can create new combinations of these chemicals and predict for them. However, it is impossible to predict for sets of chemicals that have not been used in the dataset. Thus, to accurately sample it is important to develop a screen that covers the entire space. In this work, we advocate an incremental approach to this problem, with each iteration of the loop leading us slowly towards greener pastures for sampling.

A related issue is the large amount of time required for experimentation. Despite refinements over the years, this still remains the greatest bottleneck in the crystallization process. This minimizes the amount of experimental validation that can be performed. In our work, we have tried to minimize the number of conditions to be tested and decrease the false positive rate.

An important observation we have made from this study is that conventional distance metrics cannot adequately capture the distance between similar conditions in the protein crystallization space. This is evidenced by the poor performance of the Nearest Neighbor algorithm. This demonstrates a need for distance metrics that are sensitive to the domain, as suggested by Aggarwal in [1]. A suitable distance function, in this case, would need to consider the physio-chemical characteristics of the reagents used as well as correlations between them.

The difference in precision between the 2-class classification and 3-class classification indicates that regions that yield precipitates (class 1) are close to hot spots. This was supported by the fact that the 3-class classifiers mis-classified a large number of samples belonging to class 1 as class 2 samples. Even in the wet lab experiments, we were surprised to find that a large number of samples predicted to be class 2 belonged to class 1. Our observations suggest the following view on the protein crystallization space:

- Areas fertile for crystallization (hot spots - class 2) are often well separated. This is somewhat evidenced by the poor performance of the nearest neighbor classifier.
- These areas are surrounded by areas which are

not good enough to produce crystals but yield precipitates (class 1). A large part of the space comprises of no-hit areas which do not yield any crystal (class 0).

We can therefore hypothesize that the crystallization space is of a continuous nature with 2's turning into 1's and then 0's. A representation of the crystallization space consistent with this idea is shown in Figure 6. This representation should be prove to be useful for future studies.

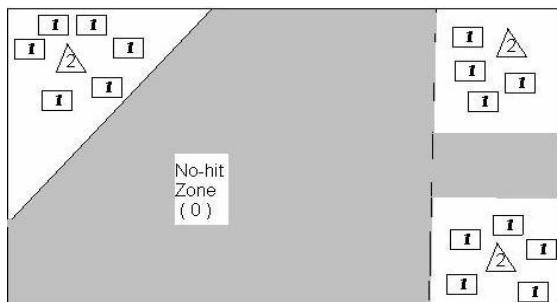


Figure 6: Crystallization Space

8 Conclusion

In this paper, we utilize supervised learning techniques to explore the properties of the protein crystallization space and to identify potential hot spots of protein crystallization. This problem has baffled scientists for many years due to a limited understanding of the crystallization space, and the cost of performing crystallization experiments. In this work, we presented a strategy for mining the crystallization space effectively for novel conditions. Our hypothesis that the crystallization space is conducive to the use of classifiers is borne out by our classification results. Our wet lab experimental results, although preliminary, show great promise. We are presently conducting experiments on a larger scale. In future, we plan to develop alternative distance metrics for the crystallization space to increase the quality of our classification techniques in hopes of refining our preliminary map of this space and finding more hot spots.

9 Acknowledgements

We would like to thank Dr. Martin Caffrey, Vadim Cherezov and the Caffrey lab for facilities and help provided for this work. This work is supported primarily by the DOE Early Career Principal Investigator Award No. DE-FG02-04ER25611 and also by NSF CAREER Grant IIS-0347662.

References

- [1] Charu C. Aggarwal. Towards systematic design of distance functions for data mining applications. *SIGKDD 2003*, pages 9–18, 2003.
- [2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. *VLDB*, 1994.
- [3] Chris Bailey-Kellogg and Naren Ramakrishnan. Ambiguity directed sampling for qualitative analysis of sparse data from spatially-distributed physical systems. *IJCAI*, 2001.
- [4] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard. A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data. *SIGKDD Explorations Newsletter*, 6(1):20–29, 2004.
- [5] M. Caffrey. Membrane protein crystallization. *Journal of Structural Biology*, 142:108–132, 2003.
- [6] C. A. Cumbaa and I. Jurisica. Automatic classification and pattern discovery in high-throughput protein crystallization trials. *Journal of Structural and Functional Genomics*, 6(2-3):195–202, 2005.
- [7] Carter Jr. C.W. and C.W. Carter. Protein crystallization usingincomplete factorial experiments. *J. Biol. Chem*, 254:12219–12226, 1979.
- [8] B. Liu et al. Integrating classification and association rule mining. *Knowledge Discovery and Data Mining*, pages 80–86, 1998.
- [9] M. Kimber et al. Data mining crystallization databases: knowledge-based approaches to optimizing protein crystal screens. *Proteins*, 51:562–568, 2003.
- [10] N. V. Chawla et al. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence and Research*, 16:321–357, 2002.
- [11] Samudzi et al. Cluster analysis of biological macromolecular crystallization database. *Journal of Crystal Growth*, 123:47–58, 1994.
- [12] C. W. Hsu and C. J. Lin. A comparison on methods for multi-class support vector machines. *IEEE Transactions on Neural Networks*, 13:415–425, 2002.
- [13] T. Joachims. Making large-scale svm learning practical. *Advances in Kernel Methods - Support Vector Learning*, B. Schlkopf and C. Burges and A. Smola (ed.), MIT-Press, 1999.
- [14] M. V. Joshi, R. C. Agarwal, and V. Kumar. Mining needle in a haystack: classifying rare classes via two-phase rule induction. *SIGMOD Record (ACM Special Interest Group on Management of Data)*, 30(2):91–102, 2001.
- [15] R. Page and R. C. Stevens. The power of decision tables. *European Conference on Machine Learning*, 1995.
- [16] B. Rupp. Maximum likelihood crystallization. *Journal of Structural Biology*, 142:162–169, 2003.
- [17] B. Segelke. Efficiency analysis of sampling protocols in protein crystallization screening. *Journal of Crystal Growth*, 232:553–562, 2001.
- [18] Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*, Springer, 1995.