Prior-free Inference in a Probabilistic Model using Linear Programming

An Application to Protein Structure Prediction

Master's Thesis in Engineering Mathematics

Marcel Lüthi 780213-C933 CTH

Examiner:
Docent Peter Damaschke
Department of Computer Science

Department of Computer Science
Chalmers University of Technology
Göteborg University
2005

Acknowledgements

My foremost thank goes to my supervisor, Prof. Peter Damaschke, for the opportunity to take part in this project and his guidance during the thesis-work. I would also like to thank Dr. Anders Bergkvist, Department of Molecular Biology at Göteborg University, for his explanations, remarks and valuable suggestions. Further my gratitude goes to Prof. Mark Dougherty, Dalarna University, who supported and helped me in my goal to pursue further studies, and Prof. Ivar Gustafsson for giving me the chance to attend this Master Programme.

Throughout my studies and in particular during this thesis-work I benefited from the availability of great free software packages. Many thanks to the Free Software Foundation and all the numerous free software authors.

Abstract

There is a well known correlation between the chemical shift measured on certain atoms of a protein and the local backbone torsion angle. This correlation makes it possible to predict likely torsion angle intervals from given chemical shifts. From a scatterplot of the chemical shifts against the torsion angles, the necessary parameters to apply Bayesian inference can be estimated. In this way likely torsion angle intervals for given chemical shifts can be determined. A linear optimization problem is formulated that chooses a minimal number of torsion angle intervals that are likely to contain the true torsion angle. The intervals are chosen in such a way that a specified error-probability to exclude the true torsion angle is guaranteed. To make the inference independent of the prior distribution in the database, the linear program is formulated so that it yields an optimal strategy for a worst-case prior distribution. The problem and its optimal solutions have several interesting properties that are discussed in this report. Several data-sets obtained by measurements on different nuclei are available for each torsion angle. Combining them leads to better predictions than using a single data-set. Two possible strategies of combining several data-sets to predict torsion angle intervals are discussed and contrasted.

Contents

1 Introduction

Predicting the ternary structure of a protein from its amino-acid sequence is an important problem in molecular biology. This three-dimensional structure can be determined experimentally using different techniques, as for example X-ray crystallography and Nuclear Magnetic Resonance (NMR). The cost for applying these methods is much higher than the cost for sequencing a protein, thus the number of structure resolved proteins is small compared to the sequence data available. But the biological function of a protein is dependent on its three dimensional structure, which makes this problem of predicting or determining a protein structure one of the most important in computational structural biology. In this project we study a linear optimization problem that arises in the context of such a protein structure prediction task. Although the focus of this work lies on the mathematical problem, which might be of independent interest, there will be frequent references to the biological application.

A measure from NMR spectroscopy called *chemical shift* is sensitive on local torsion angles in the protein backbone. Adopting a probabilistic model, we use samples of the chemical shift values for known torsion angles and apply Bayesian inference to reverse this relation. This enables us to predict likely torsion angles from chemical shift values. These predictions can serve as angle-restraints in 3D-structure calculation. The parameters for Bayesian inference are estimated from scatterplots of the chemical shift versus torsion angle. The exact problem setting is discussed in section 4. Then we study the problem of finding an optimal guessing strategy of torsion angle intervals for given chemical shifts. The problem is formulated for an unknown prior distribution, so as to yield the optimal solution in the worst-case. This results in a linear programming model. The mathematical model and its properties are discussed in section 5, which comprises the core of this project work. Several enhancements of the mathematical model and the applied methods are discussed in section 6. Two extensions of the mathematical model are formulated which allow for a compromise of total prior-independence and fixed prior-distributions. Different methods of combining several data-sets to infer torsion angle intervals are presented. To conclude the section, a possible heuristic for partitioning the scatterplot to suit our problem formulation is discussed. In section 7 the developed methods are applied to real data to illustrate the theoretical aspects of the previous sections.

This work is part of an ongoing research project. Many interesting questions arose during the course of this project, which could not be investigated due to the limited scope of this work. Probably the most important of these questions is whether there exists an algorithm to solve this problem apart from those for the general linear programming problem.

1.1 Some remarks about the notation

Most of the symbols introduced later represent vectors and matrices. When it is clear from the context no special notation is used to distinguish between vectors and scalars. Only if the distinction should be stressed are vectors marked with a small arrow (e.g. $\vec{v}$). All the vectors are column vectors. To refer to th i-th row- or column-vector of a matrix x the notation $x_{i(\cdot)}$ or $x_{(\cdot)i}$ is used respectively. The symbol $:=$ is used to indicate a definition.

2 Background

Proteins are polypeptide chains consisting of a large number of amino acid residues. The order in which the 20 different amino acids are arranged is called the primary structure of a protein. The biological function of a protein is often dependent on its three-dimensional structure [Zha01]. A polypeptide chain consists of a regularly repeating part, called the the *backbone*. The peptide bonds between two successive residues is a relatively rigid planar structure, and thus rotation of this bond is restricted. There is however rotational freedom about the single bond that link each C^{α} atom to the N and C atoms of peptide bonds. It is common to denote the torsion angle between C^{α} and N as ϕ and the torsion angle between C and N with ψ [Coz00]. These angles control the three-dimensional structure of the protein backbone.

Nuclear Magnetic Resonance (NMR) can be used to obtain a measure called *Chemical Shift* (δ) on certain isotopes of an atom. The dependence of the chemical shift on the protein structure is a well recognized fact. We use the inverse of this relation to try to infer intervals of torsion angles for a given chemical shift. The chemical shift is a function of the types of atoms present in the molecular environment around the observed nucleus and their relative positions in this environment. The analytical form of this function is strongly non-linear and unknown. A probabilistic approach is adopted to approximate this function. We assume all the contributions to the chemical shifts different from the torsion angles to be random, which expresses our ignorance of these factors [BDG04].

The chemical shifts lists of different proteins were downloaded from the public database BMRB[1] together with the corresponding protein structure from PDB[2]. For each of the nuclei $C^{\alpha}, C^{\beta}, C', H^{N}, H^{\alpha}$ and torsion angle ϕ and ψ, a scatterplot showing the chemical shift against the backbone angle was obtained. Each scatterplot contains between 350 and 2400 points. Examples of such scatterplots for the amino acid *Alanine* and torsion angle ϕ are given in appendix A. In this project these scatterplots are used to estimate probabilities and apply Bayesian inference to determine likely torsion angle intervals for given chemical shifts.

The predicted torsion angle intervals can serve as angle-restraints in 3D-structure calculation programs as for example X-Plor[3]. There are several different approaches to structure calculations. Most of them take advantage of structural restraints, that is defined allowed distance or angle intervals. If a calculated structure contains a distance or an angle outside the allowed interval, a restraint violation occurs. There is often an energy-term associated with restraint violations. The structure calculation heuristic acts to minimize the energy-terms associated with restraint violations. Within the allowed distance or angle interval the energy-term is zero.

That angle-restraints can lead to substantial improvements in the quality of the predicted structure has been shown in [CDB99].

[1] http://www.bmrb.wisc.edu
[2] http://www.rcsb.org/pdb
[3] http://xplor.csb.yale.edu/xplor/

3 Some results from optimization theory

In this section we state some basic results from optimization theory, in particular linear programming. These results provide the theoretical basis the following sections build on. The proofs to the theorems can be found in most textbooks about optimization, as for example [NS96]. The below stated definitions and theorems are given in [AEP04][4]
We consider throughout this section the linear optimization problem

$$\begin{aligned}
&z^* := \max_{x_1,\ldots,x_n} c_1x_1 + c_2x_2 + \ldots, c_nx_n \\
&\text{subject to} \\
&a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n \geq b_1 \\
&a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n \geq b_2 \\
&\qquad\quad \vdots \\
&a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mn}x_n \geq b_m \\
&x_1, x_2, \ldots, x_n \geq 0
\end{aligned}$$

which is conveniently written in matrix form as

$$\begin{aligned}
z^* := &\max_x c^T x \\
\text{s.t. } &Ax \geq b \\
&x \geq 0^n
\end{aligned} \tag{3.1}$$

The function $\max_x c^T x$ is commonly referred to as the *objective function* and z^* the *objective value*. The linear *constraints* $Ax \geq b, x > 0^n$ define a convex polyhedron, which is called the *feasible region*. Consequently an x which fulfills the constraints is called a *feasible solution*.
Some constraints may complicate a problem considerable. An important technique known as *Lagrangian relaxation* is to move the complicating constraints into the objective function.

Definition 3.1 (Lagrangian function). *Let $\lambda \in \mathbb{R}^m$ be an arbitrary vector. We define the* Lagrangian function

$$L(x, \lambda) := c^T x + \lambda^T (Ax - b)$$

Definition 3.2 (Lagrange multiplier). *The vector $\lambda^* \in \mathbb{R}^m$ is called a* Lagrange multiplier *if it is non-negative and if $z^* = \max_{x \geq 0^n} L(x, \lambda^*)$ holds.*

At a Lagrange multiplier λ^* it is possible to obtain optimal solutions to (3.1) from this relaxed problem.

Theorem 3.1 (Lagrange multiplier and global optima). *Let λ^* be a Lagrange multiplier. Then x^* is an optimal solution to* (3.1) *if and only if x^* is feasible in* (3.1) *and*

$$x^* \in \arg\max_{x \geq 0^n} L(x, \lambda^*), \quad \textit{and } \lambda^*(Ax - b)^T = 0^n$$

[4] In [AEP04] the theorems are often stated as minimization problems and where applicable without assuming linearity.

This leads us directly to the problem of finding a Lagrange multiplier, which is called the *Lagrangian dual problem.*

Definition 3.3 (Lagrangian dual problem). *The* Lagrangian dual function *is given by*

$$\theta(\lambda) = \max_{x \geq 0^n} L(x, \lambda)$$

and the Lagrangian dual problem *by*

$$\theta^* := \min_{\lambda > 0^m} \theta(\lambda) \tag{3.2}$$

To every linear program there is a corresponding *linear programming dual (LP-Dual)* formulation that is closely related to the primal problem. We can obtain the LP-dual from the Lagrangian dual by rewriting it as

$$\theta(\lambda) := \max_{x \geq 0^n} \{c^T x + \lambda^T (Ax - b)\} = -b^T \lambda + \max_{x \geq 0^n} (c + A^T \lambda)^T x$$

and observing that

$$\theta(\lambda) = \begin{cases} -b^T \lambda, \text{ if } & -A^T \lambda \geq c \\ \infty & \text{otherwise} \end{cases}$$

The *LP-Dual* formulation becomes then

$$\begin{aligned} y^* := & \min_{\lambda} b^T \lambda \\ \text{s.t. } & A^T \lambda \geq c \\ & \lambda \leq 0^m \end{aligned} \tag{3.3}$$

The following two results establish the relation between the primal and the dual problem.

Theorem 3.2 (Weak duality theorem). *Let x and λ be feasible in* (3.1) *and* (3.3) *respectively, then $c^T x \leq b^T \lambda$.*

The weak duality theorem holds also for non-linear programs. But for linear programs we have an even stronger result:

Theorem 3.3 (Strong duality theorem). *If one of the problems* (3.1) *and* (3.3) *has a finite optimal solution, then so does its dual, and their optimal objective value is equal.*

By the way the LP-Dual was constructed, it is clear that weak and strong duality hold in particular for the Lagrangian dual problem 3.2. As it is possible to obtain the dual-optimal solution from the primal-optimal solution and vice versa, it is sufficient to solve either problem.

Lagrangian duality will take a prominent role in the following discussion. Therefore we state here some further properties and results.
We define the *effective domain* of λ of problem (3.2) as

$$D_\theta := \{\lambda \in \mathbb{R}^m | \theta(\lambda) \leq \infty\}$$

Theorem 3.4 (Convexity of the dual problem). *The effective domain D_θ is convex, and θ is convex on D_θ.*

From this theorem it follows that in the Lagrangian dual function every local minimum is also a global minimum. The Lagrangian dual function is in our case a piecewise linear function and is at the kinks non-differentiable. The optimum is often attained at such a kink. To deal with this situation we introduce the notion of *subgradients* and *subdifferentials*.

Definition 3.4 (Subgradient). *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a convex function. We say that a vector $p \in \mathbb{R}^n$ is a subgradient of f at $x \in \mathbb{R}^n$ if*

$$f(y) \geq f(x) + p^T(y - x),\ y \in \mathbb{R}^n$$

The set of such vectors p defines the *subdifferential* of f at x, and is denoted $\partial f(x)$. In the case of the Lagrangian dual function, a subgradient is always easy to obtain. Define $X(\lambda) := \arg\max_{x \in X} L(x, \lambda), \lambda \geq 0^m$.

Theorem 3.5. *Consider again* (3.1)*. Let $\lambda \geq 0^m$. If $x \in X(\lambda)$, then $Ax - b$ is a subgradient to θ at λ, that is $Ax - b \in \partial\theta(\lambda)$.*

The following theorem is a generalized version of theorem 3.1

Theorem 3.6 (Optimality conditions for the dual problem). *Consider the dual problem* (3.2) *and let $\lambda^* \geq 0^m$. It is then optimal in* (3.2) *if and only if there exists a subgradient $g \in \partial\theta(\lambda^*)$ for which holds that*

$$g(x) \geq 0^m : \qquad \lambda_i^* g_i = 0,\ i = 1, \ldots, m$$

4 From Scatterplots to Bayesian Models

For our inference procedure we are given data in form of scatterplots of the chemical shift δ versus the torsion angles α for a given nucleus. This can be thought of as sampling from an unknown distribution $\mathcal{D}$, where the points in the scatterplots correspond to the drawn samples. In other words the true distribution $\mathcal{D}$ is approximated with the empirical distribution.

The δ-axis is partitioned into m intervals and similarly the torsion angle axis into n intervals. This means that the scatterplot is partitioned into $m \times n$ boxes. In the following we often denote a given chemical shift interval as data-element d and the torsion angle interval as the hypothesis h. The hypothesis that contains the true torsion angle for a given chemical shift is called the *target*.

Let N be the number of points in the scatterplot. For a large enough N, we expect the number of points observed in each box to be proportional to the probability $p(d, h)$. Fixing a hypothesis h, the probability $p(d|h)$ can now be estimated for each data-element d by simply counting the number of points in the corresponding box of the scatterplot. In a similar fashion we can estimate the probability $p(h)$ for each hypothesis h. These are all the ingredients for a Bayesian model. We denote the observed data elements d by indices $k = 1, \ldots, m$ and the hypotheses by $j = 1, \ldots, n$. The probabilities $p(d|h)$ are conveniently arranged in a matrix, referred to as the *likelihood matrix*. The probability to observe data-element k given the hypothesis j is then written as p_{kj}. Furthermore we denote the prior probability of the j-th hypothesis as q_j.

For one torsion angle we have chemical shifts measurements on several, say K, different nuclei. An ample model would therefore be to work with the K

dimensional distributions $p(d^1, \ldots, d^K|h)$ Unfortunately in order to reliably estimate high-dimensional probabilities a vast amount of data is required. For this project we concentrate therefore mainly on the one-dimensional marginal-distribution $p(d^i|h) = \sum_{d^1} .. \sum_{d^{i-1}} \sum_{d^{i+1}} .. \sum_{d^K} p(d^1, \ldots, d^K|h)$. Still most of the theoretical results developed below easily extend to the higher dimensional case. In the case of independence of the probabilities $p(d|h)$ for different nuclei, the joint probabilities equals the product of the marginals. But the chemical shifts from different nuclei given a torsion angle interval are often correlated. This can already be seen by visual inspection of the data (see figure 1), but is also a well known fact from biology [CDB99]. Therefore we would expect to get tighter angle intervals from the predictions if we were able to use higher dimensional distributions.

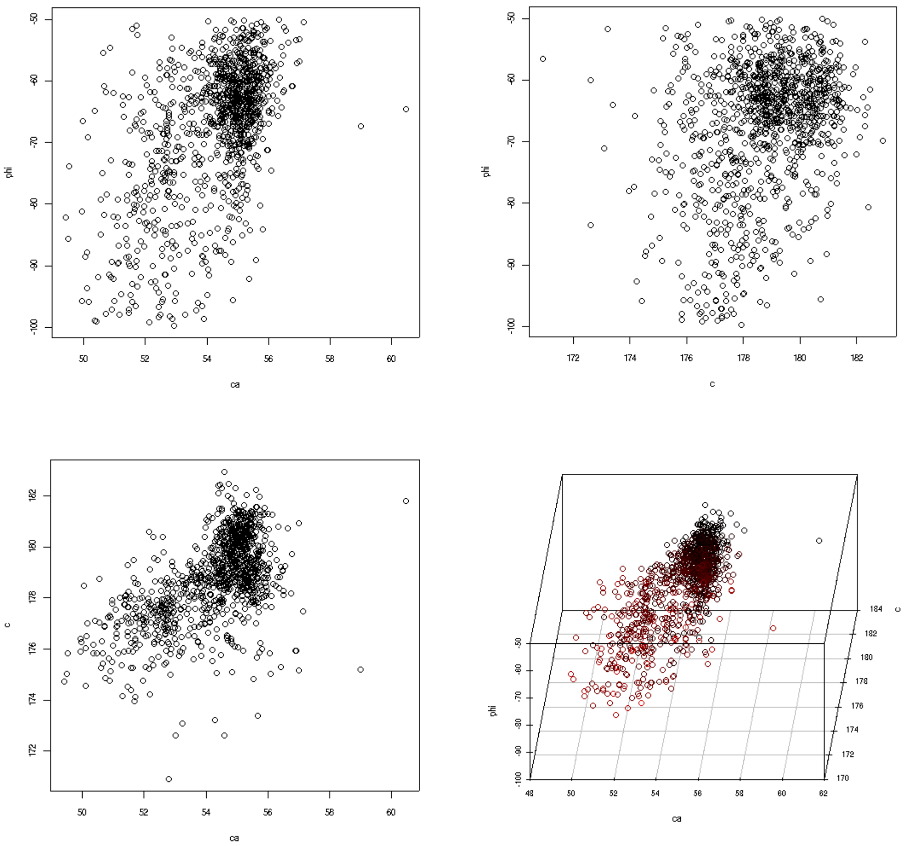


Figure 1: Scatterplots for amino acid Alanine, nuclei c and c^{α}. The first two plots show the chemical shift of c, respectively c^{α}, against ϕ, whereas the latter two plots show the chemical shift of c against c^{α} as well as the joint distribution.

4.1 Partitioning

It should be emphasized that the given data consists only of the scatterplots. We have complete freedom in the choice of partitions, as long as all the parts give rise to sensible estimates of the probabilities $p(d|h)$. Using an adequate partitioning scheme can greatly help to find good predictions.

Consider a scatterplot given in figure 2a), which has for some basic partition the point counts

$$\begin{bmatrix} 0 & 0 & 0 & 7 \\ 8 & 0 & 0 & 1 \\ 1 & 2 & 8 & 0 \\ 0 & 5 & 0 & 0 \end{bmatrix}$$

A good partition for our purposes could be obtained by joining columns 2 and 3 (figure 2b)). The likelihood matrix becomes

$$\begin{bmatrix} 0 & 0 & 0.875 \\ 0.8\overline{8} & 0 & 0.125 \\ 0.1\overline{1} & 0.6\overline{6} & 0 \\ 0 & 0.3\overline{3} & 0 \end{bmatrix}$$

We get another valid partition by splitting the matrix along the second row and column (figure 2c)) to obtain the likelihood matrix

$$\begin{bmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{bmatrix}$$

This latter partition is clearly inferior for our purpose as all the structural information is lost.

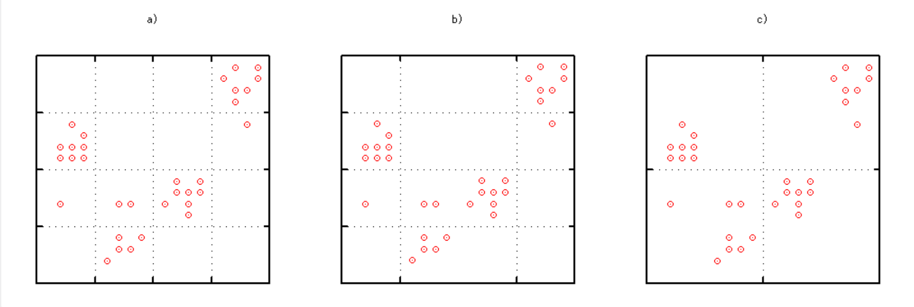


Figure 2: Different ways to partition a scatterplot

There are different factors that characterize a good partition. We could paraphrase it as a good partition should choose the intervals of the two axes in such a way, that adjacent parts have significantly different distributions. The problem of partitioning the data for such an inference task is the subject of the Master's thesis "Data Pre-Processing for LETA-NMR: Prediction of Protein Structure from Raw NMR Data" [Koc05]. The author takes the approach of choosing the parts so that the distribution in each part is uniform. The partition is found by applying statistical tests as well as using different heuristics.

Depending on the induced partition we expect a number of boxes to be void of points. These "holes" in the data are very interesting for the inference, since they represent regions of low probability. But because we are using only a sample of limited size, we can not conclude that these parts really have zero probability and no future sample comes to lie in these regions. This issue of "holes in data" is thoroughly discussed in [BDG04].

5 Mathematical model

As discussed in the previous section we can estimate all the parameters we need for applying Bayesian inference from the scatterplots. Our goal is, for a set of given observations D, to find the most likely hypothesis, that is $\max_h P(h|D)$. The famous Bayes' theorem states that $P(h|D) = \frac{P(h)P(D|h)}{P(D)}$ and hence we can easily find the most likely hypothesis h_{map} by simply solving the problem

$$h_{\text{map}} = \arg\max_h \sum_{d \in D} P(h)P(d|h)$$

or using the notation introduced in the previous section

$$h_{\text{map}} = \arg\max_j \sum_{k \in \{1,\dots,m\}} q_j p_{kj}$$

We can generalize the problem so as to devise a (probabilistic) strategy to find the K hypotheses that contain the target with the highest probability and so to increase the probability that the target is among the chosen hypotheses. Still we are interested in selecting only a few hypotheses that contain the target with high probability. We are faced with the problem of optimizing two conflicting parameters, namely maximizing the number of hypotheses that are discarded and minimizing the probability to throw away the target.

Let x_{kj} denote the probability to discard hypothesis j if data element k is observed. Further we denote the maximum probability to discard the target with ϵ and refer to it as the *error probability*. The expected number of hypotheses which are discarded, that is $\sum_{k=1}^m \sum_{j=1}^n q_j p_{kj} \sum_{i=1}^n x_{ki}$ is referred to as the *exclusiveness* of the strategy. Furthermore, we introduce a weight w_i corresponding to each hypothesis.[5] In the context of our application it is useful to identify the weight w_i with the length of the torsion angle interval that comprises the i-th hypothesis. This accounts for the fact that excluding long torsion angle intervals is advantageous for our goal to get tight angle-restraints.

For a fixed error probability, we can then formulate the problem as the linear program

$$\begin{aligned} \max_x \; & \sum_{k=1}^m \sum_{j=1}^n q_j p_{kj} \sum_{i=1}^n w_i x_{ki} \\ \text{s.t.} \; & \sum_{k=1}^m \sum_{j=1}^n q_j p_{kj} x_{kj} \le \epsilon \\ & 0 \le x_{kj} \le 1, \; \forall k, \forall j \end{aligned} \tag{5.1}$$

[5] In the following the term *exclusiveness* is slightly abused and refers even to this weighted formula.

This is a fractional knapsack problem which is trivial to solve. We order the x_{ki} with decreasing values of $w_i \sum_{j=1}^n q_j p_{kj}/(q_i p_{ki})$ and set the corresponding x_{ki} to 1 until the error probability ϵ is exceeded. The last x_{ki} can be fractional.

It might be too restrictive to assume that the prior-probabilities in the scatterplots are also applicable to the instances we want to predict.[6] We therefore consider the prior distribution to be unknown. Equivalently we can think of an adversary selecting a prior distribution from a set of priors. Our goal is to maximize the exclusiveness under the given error-probability for the worst-possible prior distribution q^*. In the most general case the prior distribution can be any $(q_1, \ldots, q_n)$ such that $\sum_{j=1}^n q_j = 1,\ q_j \geq 0$. This is a simplex in n-dimensional space with vertices $v_1 = (1, 0, \ldots), v_2 = (0, 1, 0, \ldots), \ldots, v_n = (0, 0, \ldots, 1)$.

The key observation is that for a fixed x_{kj} the exclusiveness is a linear function in q_j and so the optimal value is always attained at an extreme point of the feasible set. Each extreme point of the prior set can therefore be identified with a hypothesis j. The following mathematical program reflects this more general situation:

$$
\begin{aligned}
z^* := \max_x \min_j & \sum_{k=1}^m p_{kj} \sum_{i=1}^n w_i x_{ki} \\
\text{s.t.} & \sum_{k=1}^m p_{kj} x_{kj} \leq \epsilon,\ \forall j \\
& 0 \leq x_{kj} \leq 1,\ \forall k, \forall j
\end{aligned}
\tag{5.2}
$$

By introducing a dummy variable u we can rewrite it as a linear program:

$$
\begin{aligned}
z^* := \max_{x,u} & \ u \\
\text{s.t.} & \sum_{k=1}^m p_{kj} \sum_{i=1}^n w_i x_{ki} \geq u,\ \forall j \\
& \sum_{k=1}^m p_{kj} x_{kj} \leq \epsilon\ ,\forall j \\
& 0 \leq x_{kj} \leq 1,\ \forall k, \forall j
\end{aligned}
\tag{5.3}
$$

We refer to the first set of constraints as the *exclusiveness constraints* and the second set of constraints as the *knapsack constraints*.

The linear program (5.3) can be solved with any standard linear programming solver. Still it is worth-while to study it in depth, to gain a better understanding of its behavior and properties.

5.1 Lagrangian dual

While the bounds on x as well as the knapsack constraints are easy to deal with, the exclusiveness constraints complicate the problem considerably. It is

[6] In our application a protein that is to be predicted may be predominated by either of the structural motifs alpha-helix or beta-strand. Such structures would exhibit a significantly different angle frequency than those in the database.

therefore natural to Lagrangian relax these constraints, in order to get a simpler problem.

We define the feasible set $X := \{x | \sum_{k=1}^{m} p_{kj} x_{kj} \leq \epsilon,\ 0 \leq x_{kj} \leq 1\}$. By introducing dual variables $q \geq 0$ we get the Lagrangian function

$$L(x, u, q) = u + \sum_{j=1}^{n} q_j \left(\sum_{k=1}^{m} p_{kj} \sum_{i=1}^{n} w_i x_{ki} - u \right)$$

and the dual function

$$\theta(q) = \max_{x \in X, u} L(x, u, q) \tag{5.4}$$

The Lagrangian dual problem is then given by $\min_q \theta(q)$. The maximization problem in (5.4) can be separated for u and x and we get

$$\theta(q) = \max_u u(1 - \sum_{j=1}^{n} q_j) + \max_{x \in X} \sum_{j=1}^{n} \sum_{k=1}^{m} q_j p_{kj} \sum_{i=1}^{n} w_i x_{ki}$$

We observe that if $\sum_{j=1}^{n} q_j < 1$ in the first term then the problem is unbounded in u and hence $\sum_{j=1}^{n} q_j \geq 1$. Moreover the second term is linearly increasing in q_j. The minimum is therefore attained at $\sum_{j=1}^{n} q_j = 1$. The problem can now be stated as

$$\min_q \theta(q) = \min_q \max_{x \in X} \sum_{j=1}^{n} \sum_{k=1}^{m} q_j p_{kj} \sum_{i=1}^{n} w_i x_{ki} \tag{5.5}$$

From Theorem 3.4 we know that $\theta(q)$ is convex in q. Furthermore its set of optimal solutions (Lagrange multiplier) is convex. By strong duality at an optimal solution q^*, there exists a nonempty set of primal feasible solutions (x^*, u^*) such that $\theta^* = z^*$.

Problem 5.2 and problem 5.5 have an interesting interpretation in the context of matrix games. Where the goal in problem 5.2 is to find the exclusion probabilities x so as to maximize exclusiveness for the worst prior distribution, problem 5.5 represents the adversaries complementary view whose task it is to find a prior distribution that minimizes the best exclusiveness. The optimal dual variables q^* therefore represent a worst-case prior distribution. That the two different views on this problem lead to the same optimal value is in this context a consequence of the famous minimax theorem.

Let q in (5.5) be fixed. Then we have in every column a fractional knapsack problem that can be optimally solved using a greedy strategy, in a similar way as for problem 5.1. The benefit b_{kj} for each element x_{kj} is given by $b_{kj} = w_j \sum_{i=1}^{n} q_i p_{ki}$ and the costs by the likelihood p_{kj}. We define the utility for each element as the benefit/cost ratio $\frac{b_{kj}}{p_{kj}} = \frac{w_j}{p_{kj}} \sum_{i=1}^{n} q_i p_{ki}$. The optimal solution is obtained by letting $x_{kj} = 1$ with decreasing utility until the error probability ϵ is exceeded. The last x_{kj} in every column can be fractional. We define the set of all subproblem solutions as $X(q) := \arg\max_{x \in X} L(x, q)$. The solution obtained using above greedy algorithm are the extreme points of the convex set $X(q)$.

5.2 LP Dual problem

An alternative way of looking at the Lagrangian dual problem (5.5) is that we want to choose q such that the solution obtained from the knapsack problem is minimized, that is to minimize the best profit by weight ratios over all the hypotheses.

In this context it is instructive to look at the LP-dual formulation for the problem (5.3). We introduce dual variables q, μ, y, where q is associated to the exclusiveness constraints, μ to the knapsack constraints and the y correspond to the upper bound constraints on the x. The LP-dual is then given by

$$\begin{aligned}
&\min_{q,\mu,y} \epsilon \sum_{j=1}^{n} \mu_j + \sum_{k=1}^{m} \sum_{j=1}^{n} y_{kj} \\
&\text{s.t.} \ \sum_{j=1}^{n} -q_j w_i p_{kj} + \mu_i p_{ki} + y_{ki} \geq 0, \ \forall k, \forall i \\
&\qquad \sum_{j=1}^{n} q_j = 1 \\
&\qquad q_j, \mu_j, y_{kj} \geq 0, \ \forall k, \forall j
\end{aligned} \tag{5.6}$$

At an optimal solution the variables y_{ik} represent our gain if we were allowed to exceed the upper bound constraint by one unit. The variables μ_j in turn indicates our gain if could exceed the worst exclusiveness by one unit. As already mentioned above, the q_j represents the prior probabilities of the hypothesis j.

We reformulate (5.6) and substitute $\mu'_j = \epsilon \mu_j$:

$$\begin{aligned}
&\min_{q,\mu',y} \sum_{j=1}^{n} (\mu'_j + \sum_{k=1}^{m} y_{jk}) \\
&\text{s.t.} \ \frac{w_i}{p_{ki}} \sum_{j=1}^{n} q_j p_{kj} \leq \frac{\mu'_i}{\epsilon} + \frac{y_{ki}}{p_{ki}}, \ \forall k, \forall i \\
&\qquad \sum_{j=1}^{n} q_j = 1 \\
&\qquad q_j, \mu'_j, y_{kj} \geq 0, \ \forall k, \forall j
\end{aligned} \tag{5.7}$$

We observe that the left hand side of the first constraint corresponds exactly to the profit by weight ratio for a p_{kj} used in the fractional knapsack problem. To obtain an optimal solution the maximal profit by weight ratios, depending on ϵ, have to be minimized. With increasing ϵ more of the y_{ki} will become nonzero, meaning that it would become more attractive to violate the upper bound constraints on the x_{kj}. At the same time, μ_i is decreasing with increasing ϵ. For $\epsilon > 1$ it is always better to increase y, which is in concordance with the above interpretation of the dual variables. We also see from this formulation, that if q^* is optimal in 5.7 then all q for which hold that $\sum_{j=1}^{n} q_j p_{kj} = \sum_{j=1}^{n} q_j^* p_{kj}$ for all k, are also optimal.

5.3 Optimal primal solutions

Not all the subproblem solutions $X(q^*)$ at an optimal q^* to the Lagrangian dual problem are also primal optimal solutions, since due to the relaxed constraints primal feasibility might be violated. We have the following optimality criteria (compare to theorem 3.1 and 3.6): Let $g(x^*, u^*) \in \partial\theta(q^*)$ be a subgradient at q^*. The triple (q^*, x^*, u^*) is optimal if and only if $q_j^* g_j^* = 0$ for $j = 1, \dots, n$ and $g(x^*, u^*) \geq 0$. A subgradient $g(x, u)$ is given by $\sum_{k=1}^{m} p_{kj} \sum_{i=1}^{n} w_i x_{ki} - u$, where $x \in X(q)$ is a knapsack solution and u the corresponding objective value.

We define $X^*(q^*)$ to be the set of primal optimal solutions that fulfill the optimality criteria at q^*. We can show that $X^*(q^*)$ is invariant over all Lagrange multiplier q^* and therefore it is enough to know one q^* to characterize all solutions.

The idea is to show that the value of $g(x^*, u^*)$ attains always a value so that complementarity is fulfilled for any Lagrange multiplier.

Lemma 5.1. *Assume that we have two distinct Lagrange multipliers q' and q'' and there exists a j, such that $q'_j = 0$ and $q''_j > 0$. Then for all $x^* \in X^*(q')$ the corresponding subgradient component satisfies $g_j(x^*, u^*) = 0$.*

Proof. Assume that for such a j there exists a x^* such that $g_j(x^*, u^*) > 0$. The optimality criterion requires $g_j(x^*, u^*) \geq 0$ for all j and hence it holds that $\sum_{k=1}^{m} p_{kj} \sum_{i=1}^{n} w_i x^*_{ki} \geq u^*$.
Since $q''_j > 0$, this x^* plugged in the Lagrangian dual function for q'' yields

$$\sum_{j=1}^{n} q''_j \sum_{k=1}^{m} p_{kj} \sum_{i=1}^{n} w_i x^*_{ki} > \sum_{j=1}^{n} q''_j \sum_{k=1}^{m} u^* = u^*$$

But this contradicts the optimality of q''. □

Proposition 5.1. *Let (x^*, u^*) be an optimal primal solution. Then (x^*, u^*) fulfills the optimality conditions with any Lagrange multiplier q^*.*

Proof. If q^* is unique the proposition holds trivially.

Otherwise choose two arbitrary Lagrange multiplier q' and q''. There are now three different cases for the components q'_j and q''_j to consider:

Both $q'_j = 0$ and $q''_j = 0$: Then every $g_j(x^*, u^*)$ fulfills complementarity.

Both $q'_j > 0$ and $q''_j > 0$: Then to fulfill the optimality conditions we need for any optimal solution $g_j(x^*, u^*) = 0$.

Either $q'_j > 0$ and $q''_j = 0$ or vice versa: By above lemma $g_j(x^*, u^*) = 0$ for every optimal solution.

Hence every (x^*, u^*) that fulfills the optimality conditions for q' also fulfills it for q'' and vice versa. □

Unfortunately it is not clear how to find the primal optimal solutions from the subproblem solutions algorithmically, except by solving again a linear program. We can however find it easily for a special case.

Proposition 5.2. *If for one prior q there is a knapsack solution where all rows sum to the same value, say $\sum_{i=1}^{n} w_i x_{ki} = u$, then q is a Lagrange multiplier and x is a primal optimal solution.*

Proof. We show that q and x satisfy the optimality criteria: Obviously x and q are feasible in the primal and the dual respectively. Since x is a knapsack solution, $x \in \arg\max_x \theta(q)$. It remains to show that complementary slackness holds. The exclusiveness constraints in the primal problem can be written as $\sum_{k=1}^{m} p_{kj} \sum_{i=1}^{n} w_i x_{ki} = \sum_{k=1}^{m} p_{kj} u = u$. Hence the row-sum corresponds to the primal objective value. Then each subgradient component g_j becomes $g_j(x, u) = \sum_{k=1}^{m} p_{kj} \sum_{i=1}^{n} w_i x_{ki} - u = u - u = 0$ and we are done. □

Recall that the x_{kj} define exclusion probabilities. Using these probabilities in further applications would be greatly facilitated if most were either 0 or 1 and hence the strategy to a large extent deterministic. It is therefore good news that we actually can state a bound for the number of fractional elements in an optimal solution.

Proposition 5.3. *For every problem instance there exists an optimal solution, which has at most* $2n$ *fractional elements.*

Proof. This follows geometrically using results from linear programming theory. For a linear program, the optimal solution is always attained at an extreme-point of the feasible set. The feasible set is determined by the values x can attain. If only the constraints $0 \leq x_{kj} \leq 1, \forall k, j$ are given, then the feasible set is a hypercube in $n \times m$-dimensional space. In order for an extreme point with $2n+1$ fractional elements to exists, it has to be bounded by $2n + 1$ hyper-planes. But for this we need at least $2n + 1$ constraints. □

In addition we can also say something about how the fractional elements in such an optimal solution are arranged. For this purpose we construct an undirected bipartite graph $G = (R, C, E)$, with a vertex $v_k \in R$ for each row $x_{k(\cdot)}$ and a vertex $v_j \in C$ for each column $x_{(\cdot)j}$. The graph G contains an edge $e = (v_k, v_j) \in E$ if and only if there is a fractional element at position x_{kj}.

We show that if there is a cycle in the graph $G(R, C, E)$, then we can obtain an optimal solution with one fractional element less. The idea is that if we have a cycle, we can "shift" a quantity between two fractional elements of a column without changing the objective value and even-out the change in row-sums along the remaining vertices of the cycle.

Proposition 5.4. *Let x be an optimal solution with n fractional elements. Let $G(C, R, E)$ be a graph obtained from x using above construction. If G contains a cycle then there exists an optimal solution with $n - 1$ fractional elements.*

Proof. Let G^c be the subgraph of G that forms such a cycle. Each vertex $v_i \in R \cup C$ has at least degree 2. Therefore there exist in each corresponding row/column at least 2 fractional elements. In the following we consider only these fractional elements. We fix two such elements x_{ij} and $x_{ij'}$. There exists another fractional element $x_{i'j}$ in the same column x_{ij}, and theses two elements must have the same utility. We decrease x_{ij} by a sufficiently small amount ρ and increase $x_{i'j}$ by $\rho \frac{p_{ij}}{p_{i'j}}$ without changing the objective value. We want to let the row-sum of all the other rows unchanged, which can be achieved by decreasing the fractional element $x_{i'j''}$ in the same row by $\rho \frac{p_{ij}}{p_{i'j}}$. Again, there is another fractional element $x_{i''j''}$ in column j''. Increasing $x_{i''j''}$ by $\rho \frac{p_{i'j''}}{p_{i''j''}} \frac{p_{ij}}{p_{i'j}}$ yields the original objective value. Continuing in this way along the cycle G^c, we eventually arrive at the fractional element $x_{ij'}$.

The value we increase $x_{ij'}$ using this procedure is exactly ρ. If not, then we had obtained a solution with all row-sums unchanged, except for row i. But this would lead to a different objective value, which is a contradiction. Hence $x_{ij'}$ is increased by the quantity ρ.

Since the objective value as well as the row-sums are unchanged, we have therefore obtained another optimal solution. We can increase ρ and repeat this procedure, until one of the elements along the cycle become either 0 or 1, in which case we obtained a solution with only $n-1$ fractional elements. □

From this proposition we know that there is always a solution that is cycle-free in above sense. It is now easy to obtain a tighter bound for the number of fractional elements for the case $m < n$.

Corrolary 5.1. *There is always an optimal solution with at most $n + m - 1$ fractional elements.*

Proof. Consider again the bipartite graph $G(R, C, E)$, with $|R| = m$ and $|C| = n$. Since there exists a solution for which G is cycle-free, there must be fewer edges than vertices in G. Hence there can be at most $n + m - 1$ fractional elements. □

5.3.1 Lexicographic solution

A subset of all the optimal solutions to (5.3) is of particular interest - the set of *lexicographic solutions*. In the current problem formulation, the goal is to find probabilities x_{kj} that maximize the exclusiveness in the case that the hypothesis leading to the lowest exclusiveness is the target. Among all the optimal solutions to this problem, there is a subset of solutions, which have the property that x maximizes exclusiveness for the hypothesis that leads to the lowest exclusiveness, but also for the second lowest, third lowest etc. We refer to these solutions as the *lexicographic optimal solutions*. The lexicographic optimal solutions are easy to obtain. Given an optimal solution to (5.3), we fix the constraints corresponding to the least exclusiveness, introduce a new variable and optimize over this new problem. If all the exclusiveness constraints have the same value for an optimal solution, then x is the corresponding lexicographic solution.

5.4 More about epsilon

The error probability ϵ is a vitally important parameter in the prediction process. The hypotheses (i.e. torsion angle intervals) that are kept in an optimal solution serve as angle-restraints for the $3D$-structure calculation. To determine an optimal value of ϵ it is important to know how the structure-calculation heuristic uses the angle-restraints and how a wrongly discarded interval affects the algorithm. A more practical approach is however to try out different values of ϵ until the desired trade-off between exclusiveness and error-probability is reached. Independent on how the value of ϵ is determined, it is of great interest to know how ϵ affects the solution as its value is varied.

It is obvious that the exclusiveness is monotonically increasing in ϵ, since for $\epsilon' > \epsilon$, all solutions obtained for ϵ are also feasible for the program with the increased parameter ϵ'. The following proposition shows that it is even concave in ϵ:

Proposition 5.5. *The exclusiveness is a concave function in* ϵ

Proof. We use again Lagrangian duality. At an optimal solution (x^*, u^*, q^*) the Lagrangian dual problem and the primal problem have, by strong duality, the same objective value, and the optimal x^* is a convex combination of the knapsack solutions. We can therefore argue using the knapsack solution.

While ϵ runs through all the value from $0 \le \epsilon \le 1$, the Lagrange multiplier q can change, so we denote the Lagrange multiplier at ϵ by q^ξ. Let $\theta^\xi(\epsilon)$ be the objective value obtained for a fixed q^ξ at ϵ.

Fix a q^ξ. The knapsack solutions are obtained by a greedy algorithm that considers the elements in each row with decreasing utility (i.e. benefit/cost ratio). If we increase ϵ by a small amount it increases in each column the element with the highest utility, which is smaller than one.

The benefit, that is the increase in the function θ^ξ if x_{kj} is increased by δ, is given by $b_{kj}(\delta) = \delta w_j \sum_{i=1}^n q_i p_{ki}$. The corresponding costs are given by δp_{kj}. Hence as long as for growing ϵ the elements the greedy algorithm increases remain all smaller than one, θ^ξ grows linearly in ϵ. θ^ξ is therefore piecewise linear. Since the elements are considered with non-increasing utility and the quantity $\sum_{i=1}^n q_i p_{ki}$ is fixed, the ratio $\frac{w_j}{p_{kj}}$ is non-increasing for growing ϵ. For a fixed q^ξ, the function $\theta^\xi(\epsilon)$ is therefore concave.

The exclusiveness is then given by $excl(\epsilon) = \inf_\xi \theta^\xi(\epsilon)$. Since all $\theta^\xi(\epsilon)$ are concave, we conclude that $excl(\epsilon)$ is concave. □

Intuitively one would think that the x_{kj} also increase monotonically in ϵ. Unfortunately this is not always the case. The following numerical example illustrates what can happen. The example uses the fact that the set of optimal primal solution is the same for all worst-case prior distributions (see proposition 5.1).

Let $p = \begin{bmatrix} 0.1 & 0.5 & 0.8 \\ 0.9 & 0.5 & 0.2 \end{bmatrix}$. In this example, for $\epsilon = 0.1$, a worst case prior distribution, as obtained from solving the LP-dual with a linear programming solver, is given by $q = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$. The profit by weight ratios which are used to calculate the knapsack solutions are given by $\begin{bmatrix} 1 & 0.2 & 0.125 \\ 1 & 1.8 & 4.5 \end{bmatrix}$. The knapsack solutions for column 2 and 3 are unique and hence in every optimal primal solution holds that $x_{(\cdot)2} = \begin{bmatrix} 0 \\ 0.2 \end{bmatrix}$ and $x_{(\cdot)3} = \begin{bmatrix} 0 \\ 0.5 \end{bmatrix}$.

For $\epsilon = 0.2$, a worst case prior distribution is $q = \begin{bmatrix} 0.429 & 0 & 0.571 \end{bmatrix}$. The corresponding profit by weight ratios are $\begin{bmatrix} 5 & 1 & 0.625 \\ 0.5\overline{5} & 1 & 2.5 \end{bmatrix}$. In this case column 1 and 3 are determined by the knapsack solution, with $x_{(\cdot)1} = \begin{bmatrix} 1 \\ \frac{1}{9} \end{bmatrix}$ and $x_{(\cdot)3} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$.

The optimality criterion requires that $\sum_{k=1}^{m} p_{kj} \sum_{i=1}^{n} x_{ki} = u^*$ for $q_j > 0$ and applying this to the example we get the following system of linear equations:

$$0.1(1 + x_{12} + 0) + 0.9(\frac{1}{9} + x_{22} + 1) = u^*$$
$$0.8(1 + x_{12} + 0) + 0.2(\frac{1}{9} + x_{22} + 1) = u^*$$

The optimal objective value u^* is in this case $1.25\overline{5}$. This system of equations has a unique solution which is $x_{12} = 0.25\overline{5}$, $x_{22} = 0.14\overline{4}$. Hence the (unique) optimal solution for $\epsilon = 0.2$ is $x^* = \begin{bmatrix} 1 & 0.25\overline{5} & 0 \\ 0.1\overline{1} & 0.14\overline{4} & 1 \end{bmatrix}$. Compared with the solution for $\epsilon = 0.1$ we see that x_{22} attains a lower value for $\epsilon = 0.2$. The reason why this happens is that when we increase ϵ from 0.1 to 0.2, the upper-bound constraint $x_{11} \leq 1$ becomes binding. The knapsack constraint for $j = 1$ is not binding yet and hence x_{21} is increased. At the same time in column 3, x_{23} becomes 1 while x_{13} remains 0. To fulfill complementarity, the row-sum of x has to be equal for all rows. Hence x_{21} attains a higher value than x_{22} in order to balance the row-sums again. This leads to the situation that x_{22} is decreasing while ϵ is increasing.

6 Enhancements of the method

The method discussed above provides a good basis for inferring likely torsion angle intervals from chemical shifts. Yet the quality of the predictions is influenced by many factors. In this section we discuss several improvements which will lead to more flexibility and better predictions of the angle-restraints.

6.1 Extensions of the models

The mathematical model discussed so far is very general and provides us with a prediction strategy for the worst-case. Empirical tests using different data-sets show that this is in some cases overly pessimistic. The total ignorance of the prior probabilities inherent to the data leads to that all hypotheses are treated with equal respect.

The following example illustrates the problem that can occur. The data is simplified but similar patterns occur in the real data. Consider the data with point-counts $\begin{bmatrix} 100 & 3 & 2 \\ 400 & 2 & 1 \end{bmatrix}$ and the corresponding likelihoods $\begin{bmatrix} 0.2 & 0.6 & 2/3 \\ 0.8 & 0.4 & 1/3 \end{bmatrix}$. The optimal solution for an error probability $\epsilon = 0.2$ is $\begin{bmatrix} 1 & 0.04 & 0 \\ 0 & 0.44 & 0.6 \end{bmatrix}$. For the observation 1, we discard the first hypothesis with probability 1. However, judging by the point-counts we would expect the first hypothesis to be the most-likely target. The problem arises because the second and third hypothesis correspond to relatively rare events. Yet because of our ignorance of the priors, we treat all the hypotheses equally and we get a worst-case prior distribution $q = \begin{bmatrix} 0.142858 & 0 & 0.857142 \end{bmatrix}$.

These rare points do not possess much statistical significance. An reasonable extension of above model should therefore incorporate the information from the prior probabilities, without being too restrictive. In the following we propose two such extensions.

6.1.1 Individual epsilon

Intuitively, if the estimated prior probability for a given hypothesis j is very low, we would like the probabilities $x_{(\cdot)j}$ to discard the hypothesis to be high for all observations. In the extreme case we could think of suppressing this hypothesis completely. One way to achieve this is by assigning an error-probability ϵ'_j to each hypothesis j, such that $\sum_{j=1}^{m} q_j^{\text{est}} \epsilon'_j \leq \epsilon$. This leaves the overall error-probability ϵ unchanged, but provides more flexibility. It has the additional advantage that all the theory developed so far still applies to this new problem.

Taking this idea a step further, the ϵ'_j could be chosen so that exclusiveness is maximized. The following extension of problem 5.3 has this effect.

$$
\begin{aligned}
z^* := \max_{x,u,\epsilon'} \; & u \\
\text{s.t.} \; & \sum_{k=1}^{m} p_{kj} \sum_{i=1}^{n} w_i x_{ki} \geq u, \; \forall j \\
& \sum_{k=1}^{m} p_{kj} x_{kj} \leq \epsilon'_j \;, \forall j \\
& \sum_{j=1}^{m} q_j^{\text{est}} \epsilon'_j \leq \epsilon \\
& 0 \leq x_{kj} \leq 1, \; \forall k, \forall j
\end{aligned}
\tag{6.1}
$$

The parameters q^{est} represent the estimated prior probabilities, while the ϵ'_j are now error probabilities for each hypothesis. The third constraint ensures the maximal error probability ϵ. Note that if q^{est} deviates from the real prior distribution, then this can lead to that the target is excluded with a higher probability than ϵ. Even when the estimated prior distribution is uniform, not all the ϵ'_j will be equal. Consider the likelihood matrix $\begin{bmatrix} 0.1 & 0.5 & 0.8 \\ 0.9 & 0.5 & 0.2 \end{bmatrix}$ and prior $q^{est} \begin{bmatrix} 1/3 & 1/3 & 1/3 \end{bmatrix}$. The optimal solution with $\epsilon = 0.1$ to (6.1) is $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$. The exclusion probabilities for the second hypothesis are in this case all zero, it is optimal to let $\epsilon_2 = 0$ while allowing a greater error-probability on the third hypothesis. It is clear that this extension improves the exclusiveness, since it contains the original problem as a special case.

To be able to better control the values of ϵ'_j, one could introduce another parameter δ, which indicates how much the ϵ'_j can deviate from ϵ. Then the bound $\max\{0, \epsilon - \delta\} \leq \epsilon'_j \leq \min\{\epsilon + \delta, 1\}$ on ϵ'_j controls the deviations. With $\delta = 0$ we are back at the initial problem, where $\epsilon'_j = \epsilon$ for all j.

6.1.2 Restricted prior region

Although in above method the estimated prior distribution is only used to calculate an optimal error-probability for each hypothesis, it still relies on the assumption that the q^{est} is a good estimator for the real prior distribution. It deviates in this respect from the original goal to be independent of a prior distribution.

The following approach allows for a smooth transition between the two extreme cases, fixed prior and total ignorance of prior probabilities. The idea is to specify a region around the estimated prior distribution and optimize over this region of possible priors, instead of over all possible prior distributions.

In the following we denote the unrestricted prior region with $S = \vec{s}_1, \ldots, \vec{s}_n = (1, 0, \ldots 0), \ldots, (0, \ldots, 1)$. If we require the new region to be a simplex, with its faces parallel to S, we get a natural extension of problem (5.3). The midpoint of S is given by $\vec{m} = (\vec{s}_1 + \ldots + \vec{s}_n)/n$. Let π with $0 \leq \pi \leq 1$ be a scaling factor. We obtain the new region $V = v_1, \ldots, v_n$ by scaling S with π and translate it so that its midpoint coincides with $\vec{m}$. It remains to translate it such that its midpoint lies on the estimated prior $\vec{q^{\text{est}}}$. Combining it, we get the following formula for the i-th vertex of the restricted region

$$\vec{v_i} = \pi\vec{s_i} + (\vec{m} - \pi\vec{m}) + (\vec{q^{\text{est}}} - \vec{m})$$

Figure 3 illustrates the procedure in the 3-dimensional case.

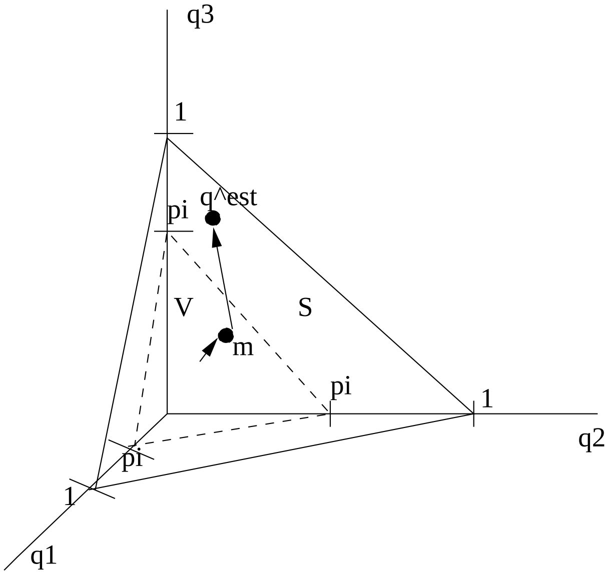


Figure 3: Obtaining the restricted region by scaling and translating S

Depending on q^{est} and π it might happen that some of the vertices v lie outside the feasible region. Since all the vertices sum to one, there exists in this case at least one component of v_i which is negative. The translation started from the midpoint and all the faces of the restricted simplex are parallel to the ones of S. Therefore it suffices to translate it back towards the midpoint $\vec{m}$, until the most negative element of V becomes zero. If i, j denote the indices of the smallest negative element of V, then this is when

$$\vec{v_{ij}} + t(\vec{q^{\text{est}}} - \vec{m})_j = 0$$

for a scalar t. The translation $\vec{v}_i = \vec{v}_i - t(\vec{q^{\text{est}}} - \vec{m})$ translate these vertices back into the feasible region. It is clear from the way the new region was constructed, that it contains the estimated prior as desired.

To extend the linear program formulation (5.3) is straight-forward. As discussed above because of linearity the minimum is always attained at the vertex of the region and hence the problem can be written as

$$
\begin{aligned}
z^* := \max_{x,u}\, & u \\
\text{s.t.}\ & \sum_{j=1}^{n} v_j^l \sum_{k=1}^{m} p_{kj} \sum_{i=1}^{n} w_i x_{ki} \geq u,\ l \in \{1,\dots,n\} \\
& \sum_{j=1}^{n} v_j^l \sum_{k=1}^{m} p_{kj} x_{kj} \leq \epsilon\ , l \in \{1,\dots,n\} \\
& 0 \leq x_{kj} \leq 1,\ \forall k, \forall j
\end{aligned}
\quad , \tag{6.2}
$$

where v^l denotes here the l-th vertex of the restricted region. Note that for the extreme case $\pi = 0$ the region is just the point q^{est} and the problem corresponds to problem (5.1) for fixed priors. The larger the parameter π is chosen, the more independent is the optimization from the prior in the data, and the more rare hypotheses will be considered. For $\pi = 1$ we get back the familiar problem (5.3), where we consider all possible prior distributions.

In spite of the apparent flexibility of this method, it has a few shortcomings. The remarks about the Lagrangian dual in section 5.1 do not extend to the new problem. The interpretation of the dual variables q is not clear anymore, as it might happen that the Lagrange multiplier q^* lies outside the region specified by V. A further motivation for using Lagrangian duality was that when the exclusiveness constraints were relaxed, the remaining problem became a fractional knapsack problem in each column. In this extended problem the knapsack constraints are also more complicated, and we can not simply solve the problem for each hypothesis individually. Furthermore, empirical analysis of the resulting solutions for different data-sets shows that the effect of the parameter π is hard to interpret.

Because of the drawbacks associated with restricting the prior region, it is in practice often best to let $\pi = 1$ and to avoid the too strong influence of hypotheses with a small prior probability by optimizing the ϵ_j as described above. There are however situations where the prior region has to be restricted. For instance consider the case when we have prior probability zero for one of the hypotheses, say h_j. For $\pi = 1$ the worst-case prior probability will be $q_j = 1$ and $q_i = 0$, for $i \neq j$, which yields always a zero objective value (compare to (5.5)). By letting $\pi < 1$ this hypothesis can be excluded from optimization.

6.2 Combining Predictions

As discussed in section 4, for a given torsion angle we have data for different nuclei. The goal is to combine this data in such a way that exclusiveness is increased, and we still can guarantee a given error-probability.

For the following discussion we have to extend our notation. We always assume that K data-sets are given. The observations on the different data-sets are in the following denoted by $k^i, i = 1,\dots,K$, and k without the superscript denotes a given tuple of observations $k = (k^1,\dots,k^K)$. In a similar way a superscript on the likelihood matrices p^i and exclusion probabilities x^i refers to the i-th data-set, whereas when the superscript is omitted we refer to the combined matrix.

There are several ways to combine hypotheses. A natural approach is to solve above optimization problem for each nucleus separately. The solution is a matrix of exclusion probabilities that represents an optimal strategy. Let K such matrices $x^1, \ldots, x^K$ be given. We assume that the angle-axis is equally partitioned into n parts for each x^i. If not this can always be achieved by refining the partitions. The combined exclusion-probabilities for a give k can be calculated as

$$x_{k(\cdot)} = \max\{x^1_{k^1(\cdot)}, \ldots, x^K_{k^K(\cdot)}\}$$

This corresponds to excluding the union of the discarded intervals $\{j | x^i_{kj} > 0\}$ for each $i \in \{1, \ldots, K\}$.

Proposition 6.1. *The error-probability ϵ to discard the target is at most $\sum_{i=1}^K \epsilon^i$ where ϵ^i is the error-probability of the optimal strategy x^i.*

Proof. Let $k = (k^1, \ldots, k^K)$ be a tuple of K observations and assume we are given N such tuples. We apply above strategy to all N tuples. When the target, say hypothesis j, is discarded, we increase a counter n_i, where $i = \arg\max_i\{x^1_{k^1 j}, \ldots x^i_{k^i j} \ldots, x^K_{k^K j}\}$

If we only apply the exclusion probabilities from the i-th matrix x^i to the N observations, then we know that for a large enough N we can expect the number of discarded targets to be $\epsilon^i N$. n_i is clearly smaller or equal to this number and so $n_i \leq \epsilon^i N$.

Hence for a sufficiently large N we have that

$$\sum_{i=1}^K N\epsilon^i = N \sum_{i=1}^K \epsilon^i \geq \sum_{i=1}^K n_i = N\epsilon$$

and therefore $\epsilon \leq \sum_{i=1}^K \epsilon^i$ □

Another possibility for combining several data-sets is to assume that the probabilities $p(d^i|h)$ are independent for all $i \in \{1, \ldots, K\}$. Under this assumption the joint-probabilities equal the product of the marginal probabilities. Then we obtain the joint likelihood matrix p of dimension $(m_1 \cdot \ldots \cdot m_K) \times n$ from the likelihood matrices $p^1, \ldots, p^K$. We introduce a (bijective) mapping $\delta : k^1 \times \ldots \times k^K \to k$, which provides us with a unique row-index k in the matrix p for each tuple $(k^1, \ldots, k^K)$.

The likelihoods p_{kj} are then given by

$$p_{\delta(k^1,\ldots,k^K)j} = p^1_{k^1 j} \cdot \ldots \cdot p^K_{k^K j}$$

Solving the optimization problem with this likelihood matrix yields the optimal exclusion probabilities x_{kj}. As discussed previously the independence assumption does not hold in general. The degree of correlation depends on which data-sets are combined. In practice a big drawback of this method is also that the joint-matrix can become very large and therefore solving the linear program becomes very computational-intensive.

6.3 Partitioning revisited

In section 4.1 we noted that the choice of partition strongly influences the quality of the data for the inference task. Instead of regarding this problem as independent of the optimization problem, we could look at it in light of the objective to maximize exclusiveness (problem 5.3). We can say that a partition of K parts is optimal if it maximizes the exclusiveness. Restricting the maximal number of parts of a partition is necessary since by refining the partitions it is always possible to increase the exclusiveness. But finer partitions make the results less reliable.

As discussed previously if a prior-distribution is known, then the exclusiveness can be calculated by solving the fractional knapsack problem using a greedy strategy. The utility of an element is given by $u_{kj} = w_j/p_{kj} \sum_i q_i p_{ki}$. The term $\sum_i q_i p_{ki}$ is bounded by 1, but w_j/p_{kj} will be big for the elements considered first by the greedy algorithm. Hence a high utility is achieved if the ratio of the area of a box to the number of points in its interior is high. A practical heuristic would therefore be to maximize the area of the union of the largest rectangles, which have at most a fixed number of points M in their interior. The number of points M corresponds to the error bound ϵ in (5.3). This problem is related to the problem of finding maximum empty (hyper-)rectangles, which has been discussed in the data-mining literature (e.g. [EGLM03], [LN03]). An algorithm for the problem of optimal partitioning of data on an interval has been developed by Jackson et al. [JSB$^+$03].

If we assume that one of the axes is readily partitioned and we consider the problem of finding an optimal partition for the other axis, it is possible to state a dynamic programming algorithm for this problem. To make the space of all possible partitions finite, we discretize the axes into a finite number of *cells* $x_i, i = 1, \ldots, N$. Only at the beginning of a cell can a new part be started. Let $P_i^*(k,m)$ denote an optimal partition of $x_1, x_2, \ldots, x_i$ on k parts using m interior points and let $g(P,m)$ be the objective value given partition P and m points . Further we write $P_i|x_{i+1}$ to extend partition P_i by starting a new part at x_{i+1} and $P_i \cup x_{i+1}$ to let x_{i+1} extend the part at x_i. The key idea is the *principal of optimality* which states that an optimal solution to any nontrivial instance of the problem is a combination of optimal solutions to some of its subinstances [BB96]. Applied to this problem it means that an optimal solution $P_i^*(K,M)$ must be a combination of optimal solutions to $P_{i-1}^*(k,m)$ for some k and m. It is clear that this principle applies to our choice of the objective function $g(P,m)$. If $P_{i-1}^*(k,m)$ was not optimal, a higher objective value could be achieved by using the optimal solution. But this contradicts the optimality of $P_i^*(K,M)$.

The algorithm starts with the problem of finding an optimal partition $P_1^*(k,m)$ for all k and m, whose solution is always trivial, since we only need to partition the cell x_1. It then proceeds through all the cells x_i, where the optimal partition P_i^* is calculated using the results obtained for P_{i-1}^* in the previous step. More precisely

$$P_i^*(k,m) = \max\{\max_{m' \in \{1..m\}}\{g(P_{i-1}^*(k-1,m')|x_i,m)\}, g(P_{i-1}^*(k,m) \cup x_i, m\}$$

When the algorithm terminates at x_N then the optimal partitions $P_N^*(k,m)$ for all number of parts k and number of points m are known, in particular $P_N^*(K,M)$.

Algorithm 1 sketches the basic structure of the algorithm. It can easily be seen that the algorithm has complexity $O(N^2KM^2)$ under the assumption that the objective function $g(P, m)$ can be evaluated in linear time in N.

The outlined algorithm works in the same way for many different objective functions $g(P)$, which gives us the flexibility to "tune" the heuristic based on the requirements of specific applications. However for the most obvious choice of $g(P)$, namely determining the objective value to (5.3) using the greedy strategy, the algorithm does not work, since the subproblem solutions are not independent (i.e. one subproblem solution affects the solution of the others). For the same reason the algorithm does not easily extend to finding an optimal solution for two or more axes simultaneously. Detailed analysis of different heuristics and finding an algorithm for higher dimensional cases goes beyond the scope of this thesis work and will be taken up at a later stage of the project.

Algorithm 1 Sketch of the partitioning algorithm with one free dimension

```
P*(k,m) ← {}, ∀k,∀m {The optimal partition for k parts and m points}
for i = 1 to N do
  { P* is only updated once for each i. The intermediate results are stored
  in a temporary variable. }
  P^tmp(k,m) ← P*(k,m), ∀k,∀m
  for k = 1 to K − 1 do
    for m = 0 to M do
      { We first determine the best partition for m points if a new partition
      is started at x_i, i.e. P' = max_{m'∈{1..m}} g(P*_{i−1}(k − 1,m')|x_i,m) }
      P' ← {}
      for m' = 0 to m do
        if g(P*(k,m')|x_i,m) > g(P',m) then
          P' ← P*(k,m')|x_i
        end if
      end for
      { If a better partition than the currently optimal on k + 1 parts and
      m points is found, it becomes the new optimum }
      if g(P',m) > g(P*(k + 1,m) ∪ x_i,m) then
        P^tmp(k + 1,m) ← P'
      else
        { Otherwise a cell x_i is added to extend the part at x_{i−1}}
        P^tmp(k + 1,m) ← P*(k + 1,m) ∪ x_i
      end if
    end for
  end for
  { P* is updated with the new best partitions}
  P*(k,m) ← P^tmp(k,m), ∀k,∀m
end for
```

7 Examples from real data

In this section the developed methods are applied to real data, namely we calculate angle-restraints for the ϕ-angle of amino-acid Alanine. The partitioning scheme of the scatterplots is very simple. Each axis is just divided into a fixed number of intervals of equal length. It would be possible to get tighter angle-restraints by using a more advanced partitioning scheme. But rather than account for the performance of the method, it is tried to underline the theoretical results by practical examples. The scatterplots for all the nuclei are given in appendix A. The grid-lines in the plots correspond to the partition lines. The data-set consisting of 1882 data-elements was partitioned into two sets. Two-third of the data-elements were used to obtain the scatterplots while the remaining 629 elements were used for prediction. All the data elements have similar angle-frequencies, as they are obtained by averaging over different proteins. Therefore the prior-independence will not lead to a significant lower error-rate of the predictions.

We run each of the following examples

- for the standard model with the full prior region ($\pi = 1, \delta = 0$),
- using the estimated prior as a the fixed prior distribution ($\pi = 0, \delta = 0$),
- optimizing the error-bound ϵ^j for each hypothesis ($\pi = 1, \delta = 1$)

and indicate the values:

Exclusiveness The exclusiveness (i.e. the objective value for (5.3)). This value multiplied by 360 yields the expected interval length that will be discarded.

Average discarded interval This is the average of the expected discarded interval over all predicted data-elements.

Expected errors The expected number of errors a committed over all the 629 data-elements using the given exclusion probabilities.

Error in percent The percentage of committed errors over all data-elements.

First we use only the data from a single nucleus H^N and H^α respectively. Table 7.1 and 7.2 show the result for the different methods. It can be seen that a big prize is paid for being independent of the prior distribution - the exclusiveness is much lower for the first method.

ϵ	π	δ	Exclusiveness	Avg. disc. interval	Exp. errors	Error %
0.2	1	0	0.46	166	129	0.21
0.2	0	0	0.73	263	121	0.19
0.2	1	1	0.73	263	130	0.21

Table 7.1: Results for Alanine, ϕ, H^N

In table 7.3 the results are shown when we optimize for nucleus H^α and H^N separately, with $\epsilon = 0.1$ each, and then combine the exclusion probabilities. It can be seen that it leads to better results than using only data from a single nucleus, for the same error-probability. Furthermore this example also illustrates

ϵ	π	δ	Exclusiveness	Avg. disc. interval	Exp. errors	Error %
0.2	1	0	0.48	175	128	0.20
0.2	0	0	0.79	282	126	0.20
0.2	1	1	0.76	276	124	0.20

Table 7.2: Results for Alanine, ϕ, H^α

ϵ	π	δ	Avg. disc. interval	Exp. errors	Error %
0.2	1	0	189	116	0.18
0.2	0	0	259	93	0.15
0.2	1	1	254	91	0.14

Table 7.3: Results for Alanine, ϕ, combining the optimized marginals for H^α and H^N

that the sum of the individual error-bounds provides only an upper bound and is normally not reached.

In table 7.4 the results for the joint-optimization are listed. We see that the error-bound is exceeded for all the parameters. This can be explained by the fact that the independence assumption does not hold for this two data-sets. Observe that in the last row of table 7.4 the parameter ϵ has been adjusted such as to get approximately the same number of errors as above. Comparing it to the result in table 7.3 we see that the length of the discarded interval is larger. This might in a practical setting justify the use of this method even if it is from a theoretical point of view not completely satisfactory.

ϵ	π	δ	Avg. disc. interval	Exp. errors	Error %
0.2	1	0	235	164	0.26
0.2	0	0	290	149	0.24
0.2	1	1	285	149	0.24
0.14	1	0	207	114	0.18

Table 7.4: Results for Alanine, ϕ, optimizing the joint-likelihood matrix obtained from H^α and H^N

We observe that the variant where the error-bound is optimized for each hypothesis ($\pi = 1, \delta = 1$) yields in all the cases slightly better results than letting the priors be fixed ($\pi = 0, \delta = 0$). This despite the fact that both share the same drawback of being prior-dependent. However it is not possible to draw a final conclusion from these tests. To be able to compare the influence of the parameters conclusively, many more tests and especially with data from single proteins would have to be performed.

8 Conclusion

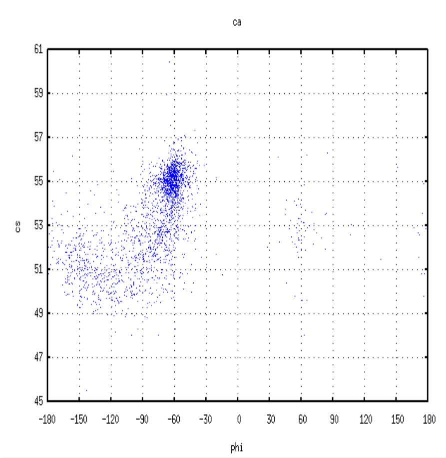
A flexible method for obtaining angle-restraints for 3D-structure calculation was presented. Through the formulation of a linear program, which maximizes exclusiveness for the worst-case prior distribution, maximal independence of the prior probabilities is achieved. This makes it possible to guarantee the specified error-probability even for proteins which have significantly different angle-frequencies than those in the database. The complete independence of the priors comes at a rather high prize. Beside leading to a lower exclusiveness, it can also happen that rare hypotheses which are unlikely to be the target are given too much importance. Therefore it will in practice be necessary to use one of the extended models discussed above, which provide more flexibility and allow to specify the degree of prior-independence. An optimal solution to the linear program provides us with probabilities that determine whether a hypothesis should be part of the angle restraints. It has been shown that most of these probabilities are either 0 or 1, and so the strategy is largely deterministic. By rounding appropriately we can get a deterministic strategy without significantly changing the results.

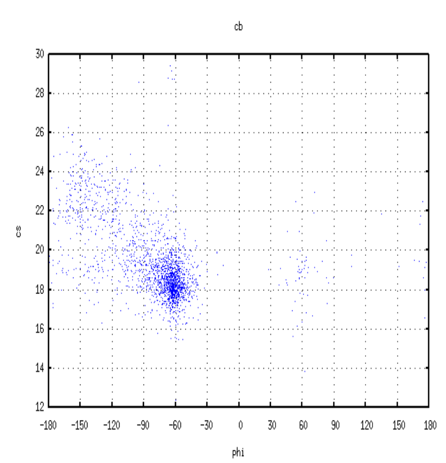
It is important for further application to find a good trade-off between the number of committed errors and the size of the angle-restraints. An optimal value for the error-probability ϵ is hard to find analytically, but it has to be selected based on empirical tests. The concavity of the exclusiveness in ϵ helps to foresee the effect of changing the error-probabiliy. Unfortunately since the exclusion probabilities are not monotone in ϵ it will still be necessary to check many different values to get the desired results.

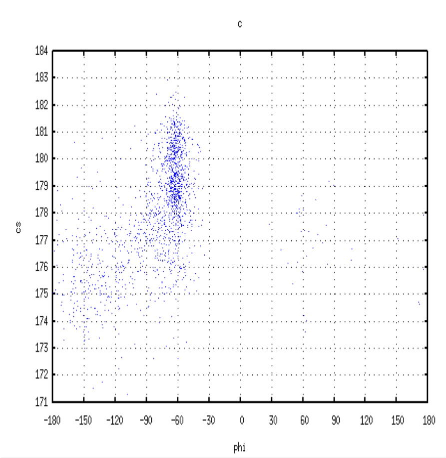
The problem can be solved using widely-available standard-software. Yet it would be helpful to find a specialized algorithm to gain more insight into the problem. Furthermore such an algorithm is expected to have a lower complexity, which would be a big advantage when in a later stage of the project higher dimensional-data are considered and thus the optimization-problems become bigger.

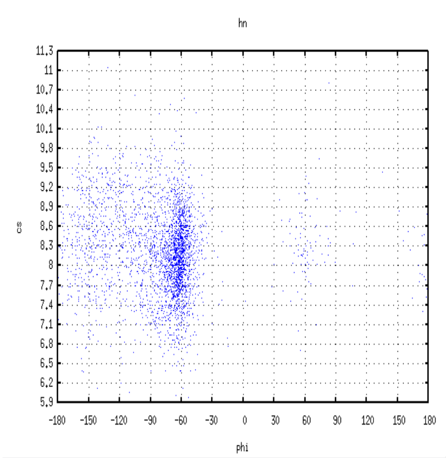
A Scatterplots

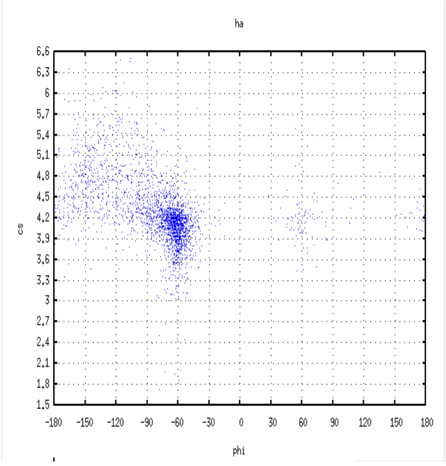
Scatterplots for the amino acid Alanine and torsion angle ϕ, for the nuclei $C^{\alpha}, C^{\beta}, C', H^{N}, H^{\alpha}$, using a simple equidistant partioning scheme.











B Program

A software for solving the optimization problems and inferring angle-restraints has been developed. As input already partitioned scatterplots must be provided. Furthermore the user has to specify values for the parameters ϵ, δ and π and which nuclei to combine for the predictions. The GNU linear programming solver GLPK Version 4.4 [GLP] is used to solve the linear programs. In addition Xerces-c [Xer] is used to parse xml-input. The program is written in C++ . It has been successfully tested on GNU/Linux and SUN/Solaris, but it should be possible to compile and run it on any platform for which GLPK and Xerces are available.

From a conceptual point of view the program consists of two parts, a *solver* and an *inference engine*. As the name implies the *solver* is used to solve the linear programs to find optimal exclusion probabilities. It supports all the different models and extensions discussed above. The *inference engine* implements different algorithm for combining data from several nuclei, and provides to a given tuple of chemical shifts the exclusion probabilities. Which data (nuclei) should be combined is determined by the user. The inference engine also provides the functionality to evaluate its performance on a list of examples.

A detailed description on how to use the software and its internal functioning is provided as a separate document.[7]

C Mathprog Model

GLPK uses a modeling language called GNU/MathProg, intended for describing linear mathematical programming models. It is a subset of the well known modeling language AMPL.
The MathProg model for above described linear program, supporting all extensions, is listed below.

```
/*
 * The MathProg model for the linear program discussed in the report.
 * It supports all the discussed extensions, but not
 * the fixed error probabilities defined by the user.
 */

/* the set of data (chemical shifts) */
set I;

/* the set of hypotheses (torsion angles) */
set J;

/* The likelihoods p_kj to observe k given angle j */
param p{k in I, j in J} >=0 <=1;

/* weight associated to each hypothesis */
param w{j in J} >=0;

/* estimated prior probabilities */
```

[7]The software as well as the user-manual can be obtained from the author by sending an e-mail to marcel.luthi@gmail.com

```
param q{j in J} >=0 <=1;

/* The extreme points of the simplex of possible prior distributions */
param X{i in J, j in J};

/* The error probability epsilon and the maximum deviation of epsilon_j from epsilon */
param epsilon;
param epsilondev;

/*
 * The primal variables:
 * u = min exclusiveness (dummy variable)
 * x_kj = probability to discard hypothesis j if k is observed
 * epshyp = the error bound for each hypothesis
 */
var u >= 0; /* dummy variable */
var x{k in I, j in J} >= 0 <= 1;

/* fix a maximum deviation epsilondev (in the report denoted delta) from epsilon. */
var epshyp{j in J} >= max(epsilon - epsilondev, 0) <= min(epsilon + epsilondev, 1);

/* Objective function */
maximize obj: u;

/* exclusiveness */
s.t. exclusiveness{l in J}:
sum{j in J} X[l,j] * sum{k in I}  p[k,j] * (sum{i in J} w[i] * x[k,i]) >= u;

/*
 * Error bounds
 */
s.t. errorbound{l in J}:
sum{j in J} X[l,j] * sum{k in I} (p[k,j] * x[k,j]) <= epshyp[l];

s.t. globalerror:
sum{j in J} (q[j] * epshyp[j]) <= epsilon;

data;

/*
 * the data is specified in a separate file
 */

end;
```

References

[AEP04] Niclas Andreasson, Anton Evgrafov, and Michael Patriksson. An introduction to optimization. 2004.

[BB96] Gilles Brassard and Paul Bratley. *Fundamentals of Algorithmics*, pages 265–266. Prentice Hall, Inc, 1996.

[BDG04] Anders Bergkvist, Peter Damaschke, and Can Gencer. Methodologies for Range Predictions Based on Scatterplots. 2004.

[CDB99] Gabriel Cornilescu, Frank Delagio, and Ad Bax. Protein backbone angle restraints from searching a database for chemical shift and sequence homology. *Journal of Biomolecular NRM, 13*, pages 289–302, 1999.

[Coz00] Alain J Cozzone. *Nature Encyclopedia of Life Sciences*, chapter Proteins: Fundamental Chemical Properties. Nature Publishing Group, 2000.

[EGLM03] Jeff Edmonds, Jarek Gryz, Dongming Liang, and Renée J. Miller. Mining for empty spaces in large data sets. *Theor. Comput. Sci.*, 296(3):435–452, 2003.

[GLP] GLPK - GNU Linear Programming Kit
http://www.gnu.org/software/glpk.

[JSB+03] Brad Jackson, Jeffrey D. Scargle, David Barnes, Sundararajan Arabhi, Alina Alt, Peter Gioumousis, Elyus Gwin, Paungkaew Sangtrakulcharoen, Linda Tan, and Tun Tao Tsai. An algorithm for optimal partitioning of data on an interval. *CoRR*, math.NA/0309285, 2003.

[Koc05] Rina Sripomo Kocadag. Data Pre-Processing for LETA-NMR: Prediction of Protein Structure from Raw NMR Data, 2005.

[LN03] Ying Liu and Mikhail Nediak. Planar case of the maximum box and related problems. In *Proc. 15th Canad. Conf. Comput. Geom.*, pages 14–18, 2003.

[NS96] Stephen G. Nash and Ariela Sofer. *Linear and Nonlinear Programming*. McGraw-Hill, 1996.

[Xer] Xerces c++, a portable, validating xml-parser developed by
the *Apache XML-Project* - *http://xml.apache.org/xerces-c*.

[Zha01] Hongyu Zhang. *Nature Encyclopedia of Life Sciences*, chapter Protein Tertiary Structures: Prediction from Amino Acid Sequences. Nature Publishing Group, 2001.