Journal of Machine Learning Research 7 (2006) 1339–1355 Submitted 9/05; Revised 2/06; Published 7/06

Linear Programs for Hypotheses Selection in Probabilistic Inference Models

Anders Bergkvist ABK@MOLBIO.GU.SE
Department of Molecular Biology
Göteborg University
P.O. Box 462
40530 Göteborg, Sweden

Peter Damaschke PTR@CS.CHALMERS.SE
Department of Computer Science and Engineering
Chalmers University
41296 Göteborg, Sweden

Marcel Lüthi MARCEL.LUETHI@STUD.UNIBAS.CH
Department of Computer Science
University of Basel
4056 Basel, Switzerland

Editors: Kristin P. Bennett and Emilio Parrado-Hernández

Abstract

We consider an optimization problem in probabilistic inference: Given n hypotheses H_j, m possible observations O_k, their conditional probabilities p_{kj}, and a particular O_k, select a possibly small subset of hypotheses excluding the true target only with some error probability ε. After specifying the optimization goal we show that this problem can be solved through a linear program in mn variables that indicate the probabilities to discard a hypothesis given an observation. Moreover, we can compute optimal strategies where only $O(m+n)$ of these variables get fractional values. The manageable size of the linear programs and the mostly deterministic shape of optimal strategies makes the method practicable. We interpret the dual variables as worst-case distributions of hypotheses, and we point out some counterintuitive nonmonotonic behaviour of the variables as a function of the error bound ε. One of the open problems is the existence of a purely combinatorial algorithm that is faster than generic linear programming.

Keywords: probabilistic inference, error probability, linear programming, cycle-free graphs, network flows

1. Introduction

Suppose that we are given one of m possible *observations* O_k, $k=1,\dots,m$, and n *hypotheses* H_j, $j=1,\dots,n$, each of which might have caused the observed O_k. Moreover we know the conditional probabilities $p_{kj}=P(O_k|H_j)$ to observe O_k if H_j is the true hypothesis, also called the *target*. Since exactly one O_k is observed, the p_{kj} must satisfy $\sum_{k=1}^{m} p_{kj}=1$ for every j. The p_{kj} may come from background knowledge of causal relations, or they may be estimated from statistical data.

Our aim is to devise a strategy that, for any observed O_k, selects a subset of hypotheses so as to minimize two conflicting parameters at the same time: the probability to discard (that is, not to select) the target, and the size of the selection. We imagine that the selected hypotheses are then examined closer, in order to identify the target, whereas we would come back to discarded hypotheses only if we missed the target in our selection. In Section 2 we will state this problem formally as an optimization problem, namely, the minimization of the expected weight of excluded hypotheses, given an error probability bound for each target. We think that the problem is very fundamental and its optimization view could be interesting for any setting where one has to guess hypotheses from data with known conditional distributions.

An obvious application scenario is diagnosis. The probabilities of various syndromes caused by any disease may be known from a database. In each particular case with a given syndrome, one wants to narrow down the set of suspects, that is, of possible diseases to be examined more carefully. But the true hypothesis should, with high probability, not be discarded in the beginning. See the discussion by Szolovits et al. (1988) which refers, however, to complex and structured models rather than "atomic" hypotheses and data.

Our particular motivation however came from a protein structure prediction project. Proteins are sequences of residues, each residue being derived from one of 20 possible amino acids. The 3D structure of the protein backbone is uniquely determined by its torsion angles. Since it is difficult and costly to determine them experimentally, various methods have been developed to infer torsion angles and other structure elements from easier measurable, correlated data, partly with help of sequence homology. Nuclear magnetic resonance (NMR) chemical shifts of nuclei in the amino acids are certain spectroscopic data influenced by the local molecular conformation, see Beger and Bolton (1997); Cornilescu et al. (1999); Wang and Jardetzky (2002); Xu and Case (2002) for more background information. Due to the correlations, it is a natural idea to infer torsion angles from measured chemical shifts. Torsion angle restraints that are narrow but still contain the (unknown) true torsion angle values in the majority of cases are important for correct 3D structure reconstruction of whole protein sequences. Since the correlations are complicated and can hardly be put in a neat formula, we have chosen a statistical approach based on large samples of data. That is, the "local" task of predicting single torsion angle restraints leads to instances of the optimization problem as considered here: Our hypotheses are torsion angles, our observations are measured chemical shifts, both discretized in finitely many intervals, and the p_{kj} are estimated from a database. Our raw data are scatterplots of chemical shift vs. torsion angle values from public databases. The discretization is done in a preprocessing phase, with the aim to partition the scatterplot into a coarse grid where the data points in each rectangle are approximately evenly distributed (so that further splitting would be meaningless). The current partitioning heuristic is described by Christin (2006). Then, we apply different prediction heuristics to the discretized scatterplots, that is, point count matrices. The role of optimization in this application is discussed after the main part of the paper, in Section 7. We have to treat in a semi-automated way a huge number of problem instances: for 6 different nuclei, 20 different amino acids, and 2 torsion angles we get nearly 240 data sets. (A few are empty.) Actually, the number of instances is a multiple of this number when we consider several error probability bounds and their combinations, maybe several discretizations, different sources of raw data, etc. In this sense our application is large-scale, even though the single problem instances are not. On the contrary, we need to reduce every instance to some efficiently solvable optimization problem in order to keep the project feasible. In this paper we will show several beneficial properties of the optimization problem that comply with this goal:

- We end up with a linear program in mn variables, which is a manageable size (see Section 2). Note that, since a selection rule conditional on the observation can be randomized, the possible strategies are described in the first place by variables for the probabilities of all 2^n subsets of hypotheses. However, by the nature of our objective function and by linearity of expectation, we actually need only variables for the probability to exclude any single hypothesis under any observation.

- We can always find an optimal solution where at most $\min\{m,n\}+n$ of the mn variables are strictly between 0 and 1 (Section 4). We conjecture that the actual number of fractional variables is even somewhat smaller. Hence, most decisions are deterministic, which greatly simplifies the practical use of our approach.

- The linear program formulation is quite flexible. We can work, for example, with larger error probabilities for hypotheses that are unlikely to appear as target, or hard to discriminate from others. We will also assign a weight w_j to every hypothesis H_j. Our goal is then to minimize the total weight of selected hypotheses, under given error probability constraints. The weights are just coefficients that do not complicate the problem to solve, but give us further modelling options (see below).

- It is easy to combine predictions from several unrelated observations, if their conditional distributions are available for the considered set of hypotheses. This can further reduce the selected set of hypotheses, for prescribed bounds on the error probability. Since most variables in optimal strategies are 0 or 1, the necessary calculations are fast (Section 6).

- Since the predictors are just linear programs, it is straightforward to implement the approach using standard software packages.

Regarding the weights, in the simplest case all w_j are equal. Otherwise, weight w_j may be used, for example, to indicate the time needed to verify or falsify H_j, so that the total weight of the selected set corresponds to the time to actually identify the target. In the diagnosis example, weights may also be proportional to time or costs to check the hypotheses, however we may divide each by a factor for the seriousness of the disease. In interval prediction applications like protein torsion angle prediction, it is sensible to choose the weight of each interval proportional to the interval length.

In Section 3 we also connect a game-theoretic interpretation of the problem to some Lagrange dual giving the worst-case probability distribution of hypotheses (in the sense that the achievable exlusiveness is minimized). We discuss the use of the dual optimum. Moreover we disprove in Section 4 the tempting conjecture that the exclusion probabilities in optimal strategies are always monotone in the error bounds. Such counterintuitive behaviour suggests that our optimization problem does not exhibit a simple structure that would also allow a simpler algorithm. Despite the fact that linear programs are a standard task being well solvable in practice, it would be interesting to devise a faster, purely combinatorial algorithm for our special class. This would speed up applications with massive sets of instances. We must leave this as an open question. Hints may come from some relationship to flow problems in lossy networks (briefly discussed in Section 5) for which such algorithms exist.

We are not aware of earlier work where optimization has been used for inference in such frameworks. A number of other machine learning tasks, for example, in classification and neural network training, have been cast as linear programs, as in Bennett (1992); Bennett and Mangasarian (1992a,b, 1993); Bradley (1998); Glover (1990). Damaschke (2004) studied target search problems in finite probabilistic inference models with the goal to minimize the expected search time, when switching between the hypotheses (preemptive scheduling of verification jobs) is possible.

One may compare our optimization to very common and simple heuristic inference rules such as the maximum likelihood (ML) and the maximum a-posteriori (MAP) rule: For an observed O_k, ML selects the desired number of hypotheses H_j with the highest p_{kj}. MAP proceeds similarly with the posterior probabilities of the H_j, for prior probabilities given along with the p_{kj}, whereas ML ignores prior probabilities. Both ML and MAP can easily exclude some potential targets H_j completely even though they appear considerably often. This happens if p_{kj} is not among the top values for any k. In our approach we explicitly take care of the probabilities to wrongly discard the target (below denoted ε_j). Similarly to ML we do not make explicit use of prior probabilities, but we can, for example, assign higher ε_j to rare H_j. Finally we optimize the specificity of our hypotheses selection for the desired prescribed error bounds.

2. Hypothesis Selection by Linear Programs

Now we treat our problem more formally. Recall that p_{kj} is the (known) probability to observe O_k if H_j is the target. Based on an observed O_k, a player wants to discard a set of hypotheses that should have large weight but should not contain the target. A *strategy* σ is completely characterized by a probability distribution on the set of subsets (power set) of $\{H_1,\dots,H_n\}$, depending on O_k. It specifies the probability to discard any set. This is in fact the most general form of a strategy, since the selection can be randomized, and the player does not learn more than just O_k. Next, we also make our optimization goal explicit.

Definition 1 *Consider a fixed strategy σ. The* error probability *of σ for target H_j is the probability to discard the target. The* exclusiveness *of σ for any fixed target H_j is the expected total weight of the hypotheses discarded by σ. (Here, randomness comes from the choice of O_k according to the p_{kj} and from σ's randomized choices.) Finally, the exclusiveness of σ is defined to be the worst (smallest) exclusiveness for all H_j.*

HYPOTHESIS SELECTION WITH ERROR BOUNDS *is the following problem: Given an $m\times n$ matrix $P=(p_{kj})$ and error probabilities ε_j for all H_j, devise a strategy with maximum exclusiveness.*

Comments:

(1) By defining the exclusiveness as the minimum over all hypotheses we optimize the guaranteed exclusiveness (in the sense of an expectation in the long run), independently of the frequencies of hypotheses which may be unknown or subject to changes: In the diagnosis example, the relative frequencies of diseases can vary a lot in time, and in torsion angle prediction, the distribution of angles in a protein under consideration is not known in advance. We avoid the explicit use of questionable prior probabilities.

(2) In the simplest case, all ε_j may be equal to some global error probability ε. However, we also allow individual error probabilities. This will not make our problem more complicated, but it gives us the option to assign higher error probabilities to certain hypotheses, and thus to raise the exclusiveness. The choice of the ε_j is up to the application, but, generally speaking, higher ε_j are advisable if H_j is considered unlikely, or if the vector of the p_{kj} for H_j (jth column of P) is in the convex hull of other columns of P, so that none of the O_k is characteristic for H_j alone.

(3) For entries with $p_{kj}=0$ we would immediately discard hypothesis H_j upon observation O_k. Alternatively we may forbid zero entries and consider only instances with positive conditional

probabilities. In applications, typically the p_{kj} are estimated from statistical data, and instead of setting $p_{kj} = 0$ in the absence of cases, it is common in statistical learning methods to apply some correction rules that yield small positive values.

Note that a strategy is described by as many as $m2^n$ variables. However, for maximizing exclusiveness we actually need only mn variables, and this makes the approach feasible. Namely, let x_{kj} be the probability that σ discards hypothesis H_j if O_k has been observed. Let X be the $m \times n$ matrix $X = (x_{kj})$. Matrix X is well-defined, and X is uniquely determined by σ. (The converse is not true: The same X can be "realized" by many different σ, we come back to this point later.)

Theorem 2 *Matrix X of an optimal strategy for* HYPOTHESIS SELECTION WITH ERROR BOUNDS *is the solution to the linear program written below.*

$$\max u \tag{1}$$

$$\forall j: \sum_{k=1}^{m} p_{kj} x_{kj} \leq \varepsilon_j \tag{2}$$

$$\forall j: \sum_{k=1}^{m} p_{kj} \sum_{i=1}^{n} w_i x_{ki} \geq u \tag{3}$$

$$\forall k, j: \ 0 \leq x_{kj} \leq 1 \tag{4}$$

Proof The left-hand side of (2) is obviously the probability to discard H_j if H_j is the target. The left-hand side of (3) is the exclusiveness for H_j, hence (3) says that the exclusiveness for every H_j is at least some u that is maximized in (1). That is, we are maximizing the exclusiveness of the strategy as desired. Constraint (4) just ensures that the x_{kj} are probabilities. ■

Corollary 3 *We can compute an optimal strategy σ for* HYPOTHESIS SELECTION WITH ERROR BOUNDS *through a linear program in only mn variables.*

In particular, it follows that the problem has polynomial time complexity in n, m. We remark that, because of (3), the exclusiveness actually depends only on the weighted sum of variables in each row of X, defined by $x_k := \sum_{i=1}^{n} w_i x_{ki}$. Corollary 3 needs some discussion. Strategy σ is not uniquely determined by X, but it is easy to obtain some σ. To mention only two natural options: We may take a random number *rand* uniformly from interval $[0,1]$ and discard all H_j with $x_{kj} \geq rand$, or we may discard the H_j independently with probabilities x_{kj}. This arbitrariness is not an issue here. Firstly, all σ with the same X have also the same exclusiveness. Thus we will henceforth consider the exclusion probabilities x_{kj} as the strategy variables. Accordingly, we also call a matrix X a *strategy*. Secondly, we will show later that there always exist optimal strategies where only a limited number of variables in X is fractional, so that most decisions are in fact deterministic.

Some applications may prefer hypotheses of some guaranteed weight for every O_k (although this can be rather unnatural, especially when rows of P contain very different numbers of safely discarded small entries p_{kj}). Then, a similar linear program where constraint (3) is replaced with $\forall j: \sum_{i=1}^{n} w_i x_{ki} \geq u$ can be applied.

3. Game-theoretic Interpretation, Knapsack Strategies, and the Dual

Our linear program from Theorem 2 is equivalent to a matrix game between a player who selects hypotheses and an adversary ("Nature") which tries to make a successful choice as difficult as possible. More precisely, the player can choose a strategy X that respects (2) and (4), the adversary chooses a hypothesis, and the payoff to the player is exclusiveness u in (3). The set of possible strategies X is infinite, but we can turn the game into an equivalent finite game, by observing that (a) exclusiveness is linear in X, and (b) all feasible X build a polytope with finitely many vertices. Hence it suffices to consider only these vertices as the player's pure strategies, all other X are convex linear combinations of them. Claim (a) is obvious from the left-hand side of (3), and (b) is clear since the X form a (bounded) feasible region of a linear program. The adversary's mixed strategies can be interpreted as prior probabilities q_j of the H_j. In the following, $q = (q_1, \dots, q_n)$ denotes a vector of prior probabilities. By von Neumann's minmax theorem, there exists a pair X^*, q^* of optimal mixed strategies for both opponents, and the expected payoff for X^*, q^* is the value of the game.

For the moment assume that the player knows $q = (q_1, \dots, q_n)$. The optimal solutions against q are easy to characterize by means of the following definitions. In every column j of X we set up an instance of the fractional knapsack problem, with capacity ε_j and items $k = 1, \dots, m$ having sizes p_{kj} and utilities $w_j \sum_{i=1}^n p_{ki} q_i$; see Martello and Toth (1990) for an introduction to knapsack problems. Note that the fractional knapsack problem is trivially solved by a greedy algorithm: Start from $x_{kj} := 0$ for all k, and then set $x_{kj} := 1$ for k with decreasing utility-to-size ratio $r_{kj} := w_j \sum_{i=1}^n p_{ki} q_i / p_{kj}$, until the capacity is exhausted. The last $x_{kj} > 0$ can be fractional. (Possible division by 0 does not cause problems, cf. Comment (3) below Definition 1. If $p_{kj} = 0$, we get $r_{kj} = \infty$ and also $x_{kj} = 1$. If the whole kth row of P is zero, we can even ignore it right from the beginning.)

Now, we call a matrix X a *knapsack strategy against prior q* if each column of X is an optimal solution to the fractional knapsack problem introduced above.

Proposition 4 *The optimal strategies X against a prior q are exactly the knapsack strategies against that prior q. In particular, if X^* is optimal then X^* is a knapsack strategy against every optimal q^*.*

Proof The first assertion is obvious, since the utility term $w_j \sum_{i=1}^n p_{ki} q_i$ is the coefficient of x_{kj} in the exclusiveness. Let q^* be any optimal strategy of the adversary. A player's strategy achieving the value of the game must be optimal under prior q^*. But since the latter strategies are knapsack strategies against q^*, the second assertion follows. ■

We remark that the converse cannot be concluded: A knapsack strategy against the optimal q^* is not necessarily optimal in the whole game, since it may be worse against other priors. Optimality requires an additional condition that we can get from duality theory of linear programs. The fact that a worst-case prior q^* corresponds to a certain Lagrangian dual might be an interesting structural property in itself:

Proposition 5 *When we dualize constraints (3), then the vector of the n Lagrange multipliers $\lambda_j \geq 0$ in the dual optimal solution is a worst-case prior q^*.*

Proof The Lagrange function is given by

$$L(X,u,\lambda) = u + \sum_{j=1}^{n} \lambda_j \left(\sum_{k=1}^{m} p_{kj} \sum_{i=1}^{n} w_i x_{ki} - u \right).$$

For any fixed vector $\lambda = (\lambda_1, \ldots, \lambda_n)$, the Lagrangian subproblem $\theta(\lambda) = \max_{X,u} L(X,u,\lambda)$ can be separated for u and X:

$$\theta(\lambda) = \max_u u(1 - \sum_{j=1}^{n} \lambda_j) + \max_X \sum_{j=1}^{n} \lambda_j \sum_{k=1}^{m} p_{kj} \sum_{i=1}^{n} w_i x_{ki}.$$

The Lagrangian dual is $\min_\lambda \theta(\lambda)$. We observe that $\sum_{j=1}^{n} \lambda_j \geq 1$, otherwise $\theta(\lambda)$ is unbounded. Since the X term is increasing in the λ_j, and the same matrices X give the maximum when vector λ is multiplied with any positive factor, $\theta(\lambda)$ attains its minimum for some λ with $\sum_{j=1}^{n} \lambda_j = 1$. Thus the Lagrangian dual simplifies to

$$\min_\lambda \theta(\lambda) = \min_\lambda \max_X \sum_{j=1}^{n} \lambda_j \sum_{k=1}^{m} p_{kj} \sum_{i=1}^{n} w_i x_{ki}$$

subject to $\sum_{j=1}^{n} \lambda_j = 1$ and the original constraints (2),(4). Note also that $\theta(\lambda)$ is precisely the exclusiveness for prior λ, thus $\lambda = q^*$. ■

Theorem 6 *(X,q) is a pair of optimal solutions if and only if: X is a knapsack strategy against q, and X has its lowest exclusiveness for all H_j where $q_j > 0$.*

Proof As we have dualized constraints (3), we get from the complementary slackness conditions that (X,q) is optimal if and only if X has optimal exclusiveness against q, and the following alternative holds true for every j: Variable q_j is zero, or the slackness in constraint (3) is zero, which means that X's exclusiveness for target H_j is exactly u (and not larger). Together with Proposition 4 the criterion follows. ■

Note that this optimality criterion can be checked in $O(mn)$ time for given X and q: One just has to solve the fractional knapsack instances for all columns j and to compare the left-hand sides of constraints (3). Since optimality is that easy to check, and the Lagrangian subproblem (fractional knapsack) is trivial, a gradient descent method for the Lagrangian dual is efficient in every step. Therefore it would be interesting to study whether some gradient descent heuristic approaches q^* already in a few iterations. This would be valuable for applications with many instances like our torsion angle prediction project.

Calculating q^* appears to be useful also in another respect: Although we did not explicitly use prior probabilities of the H_j, we know in general which H_j appear frequently or rarely. Now, if q_j^* is large for some rare hypothesis H_j, this indicates that X^* has been optimized for an unlikely distribution of targets. (Recall that X^* has the worst exclusiveness even for all H_j with positive q_j^*.) We may then drop constraint (3) for such indices j and optimize again, in order to raise the exclusiveness for the more frequent targets only. Such modifications are natural and easy to implement, and they may improve the global results in, for example, protein structure prediction. This has to be tested more extensively within the particular applications.

4. Structural Properties of Optimal Solutions

In the following we consider, for simplicity, a special case of our linear program from Theorem 2 where all ε_j are equal to some ε. Regarding the dependency of u from this parameter we have:

Proposition 7 *For any fixed likelihood matrix P, the optimal exclusiveness u is a monotone increasing and concave function in ε.*

Proof *Monotonicity:* Parameter ε appears only in constraints (2). If one raises ε then, obviously, the set of feasible solutions becomes only larger, and since we have a maximization problem, the optimal u increases.

Concavity: Consider the $(mn+2)$-dimensional space with the mn variables x_{kj} and, additionally, ε and u as coordinates. Let F be the feasible region of our linear program in this space, that is, the set of $(mn+2)$-vectors that fulfill constraints (2),(3),(4). Clearly, F is convex. Hence the projection $F|_{\varepsilon,u}$ of F to the ε vs. u plane is convex, too. ($F|_{\varepsilon,u}$ is the set of all pairs (ε,u) for which there exist values of the x_{kj} so that the constraints are satisfied.) Remember that we have to maximize u for a given ε. Geometrically this means to take the point at the upper boundary of $F|_{\varepsilon,u}$ at abscissa ε. Since $F|_{\varepsilon,u}$ is convex, the upper boundary is the graph of a (piecewise linear) concave function. (Figure 1.) ■

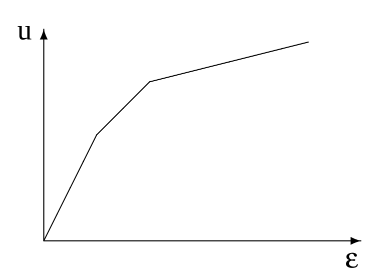

Figure 1. u is monotone and concave in ε. The graph limits the the feasible region from above.

One might expect that also every single variable x_{kj} in the strategy matrix X is monotone in the error bound ε, but this is not true in general. A small example demonstrates the reason. Recall the notations p_{kj} for the probability to observe O_k given H_j, the weighted row sums $x_k := \sum_{i=1}^{n} w_i x_{ki}$, and the utility-to-size ratios $r_{kj} := w_j \sum_{i=1}^{n} p_{ki} q_i / p_{kj}$ from the fractional knapsack problems.

Example 1 *Suppose that all hypotheses have unit weights $w_j = 1$. Consider the following matrix P of conditional probabilities p_{kj}:*

$$P = \begin{bmatrix} 0.1 & 0.5 & 0.8 \\ 0.9 & 0.5 & 0.2 \end{bmatrix}.$$

First let $\varepsilon = 0.1$. For the prior $(q_1,q_2,q_3) = (1,0,0)$ it is easy to check that any knapsack solution has exclusiveness $u = 0.73$. Moreover, against this prior, every knapsack solution with $x_1 \geq x_2$

satisfies the criterion in Theorem 6. Hence

$$X^* = \begin{bmatrix} 0.73+x & 0 & 0 \\ 0.03-x & 0.2 & 0.5 \end{bmatrix}$$

with $0 \le x \le 0.03$ *(arbitrary) is optimal, and* $(1,0,0)$ *is a worst prior. This in turn implies that every optimal* X *must be a knapsack solution against* $(1,0,0)$*. In particular,* $x_{22} = 0.2$ *is enforced.*

Now let $\varepsilon = 0.2$ *instead. Then, every knapsack solution against* $(1,0,0)$ *has* $x_1 < x_2$*, so that prior* $(0,0,1)$ *would be worse. But, similarly, every knapsack solution against* $(0,0,1)$ *has* $x_1 > x_2$*, so that prior* $(1,0,0)$ *would be worse. It follows that* $x_1 = x_2$ *holds in every optimal* X*, and that an optimal* q^* *differs from these two priors. But each prior except the mentioned two gives* $r_{11} > r_{21}$ *and* $r_{23} > r_{13}$*, which determines column 1 and 3 of* X^* *uniquely. Together with* $x_1 = x_2$ *this finally yields (matrix entries rounded to three decimals):*

$$X^* = \begin{bmatrix} 1 & 0.256 & 0 \\ 0.111 & 0.144 & 1 \end{bmatrix}.$$

Note that x_{22} *is smaller than before! The explanation is that* x_{11} *reached 1, thus only* x_{21} *could increase, and* x_{22} *decreased in favour of* x_{12}*, in order to keep* x_1 *and* x_2 *balanced.*

Next we consider arbitrary individual error bounds ε_j again. As announced, we show that our linear programs from Theorem 2 have optimal solutions where only a minority of the mn variables x_{kj} is fractional, that is, properly between 0 and 1. It means that these selection strategies are to a large extent deterministic, which makes them much easier to handle in practice.

Theorem 8 *Any optimal solution being a vertex of the feasible region has at most* $2n$ *fractional variables.*

Proof Some optimal solution X of a linear program is always a vertex of the feasible region. Constraints (4) describe the hypercube in mn-dimensional space where all vertices have coordinates 0 or 1. Furthermore, the number of binding constraints in a vertex X is at least the dimension mn, but only one of any two constraints $x_{kj} \ge 0$, $x_{kj} \le 1$ can be binding. Thus, in a vertex X with more than $2n$ fractional coordinates, more than $2n$ other constraints must be binding. Since we have only $2n$ constraints (2),(3), the assertion follows. ∎

We can also say something about the *positions* of fractional entries in optimal strategy matrices X and get a better bound in case that $m \le n$. Let $B(X)$ be the bipartite graph with vertices r_k for all rows k, and vertices c_j for all columns j, where an edge between r_k and c_j exists iff x_{kj} is a fractional value.

Theorem 9 *There exists an optimal solution* X *with cycle-free* $B(X)$*, and thus at most* $m+n-1$ *fractional entries.*

Proof Suppose that $B(X)$ contains a cycle C with vertices $r_1, c_1, r_2, c_2, \ldots, r_l, c_l$ (in this cyclic order). That is, edges $r_i c_i$, $c_i r_{i+1}$ and $c_l r_1$ exist, where $2l$ is the length of the cycle. Note that the indexing of rows and columns in X is arbitrary, hence we may rename them such that, without loss of generality, indices in C are $1,2,\ldots,l$ as defined above.

Let d be some real number that we fix later. We change the matrix entries corresponding to the edges in C by the following procedure. First, define $d_{11} = d$ and replace x_{11} with $x_{11} - d_{11}$. Define $d_{21} = \frac{p_{11}}{p_{21}} d_{11}$ and replace x_{21} with $x_{21} + d_{21}$. Obviously, the error bound constraint (2) remains valid for column $j = 1$. Next, define $d_{22} = \frac{w_1}{w_2} d_{21}$ and replace x_{22} with $x_{22} - d_{22}$. The effect is that $x_k := \sum_{i=1}^{n} w_i x_{ki}$ remains unchanged for row $k = 2$. We walk the cycle C and continue in this way. The general step is: Define $d_{ii} = \frac{w_{i-1}}{w_i} d_{i,i-1}$ and replace x_{ii} with $x_{ii} - d_{ii}$, then define $d_{i+1,i} = \frac{p_{ii}}{p_{i+1,1}} d_{ii}$ and replace $x_{i+1,i}$ with $x_{i+1,i} + d_{i+1,1}$. Following this scheme we finally we update x_{1l}, according to $l + 1 \bmod l = 1$.

Note that all these changes neither affect the left-hand sides of constraints (2) nor the weighted row sums x_k defined above, with x_1 as the only exception. If x_1 has not decreased, constraints (3) remain satisfied, too. If x_1 has decreased, we use $-d$ instead of d, so that x_1 now increases. Since all x_{kj} on edges of C are fractional, constraints (4) also remain valid for small enough $|d|$. Hence we get a new feasible solution for any d which has the suitable sign and small enough absolute value. Finally we adjust our d so that some entry in C becomes exactly 0 or 1.

Hence we can destroy some cycle C of fractional entries. Since the optimal value u is monotone in the x_k, the new solution X is no worse. Applying the same procedure repeatedly we destroy all such cycles. Since every step also properly decreases the number of fractional entries, the process terminates with an X as desired. Since a cycle-free graph has fewer edges than vertices, the bound $m + n - 1$ follows.

We remark that this proof gives also a polynomial algorithm that computes a cycle-free optimal solution. ∎

The 3×2 instances from Example 1 admit optimal solutions with $n = 3$ fractional entries.

An obvious question is whether our combinatorial bounds are already tight. More precisely: Given numbers m, n, let $f(m,n)$ denote the largest number such that there exists an $m \times n$ instance P of HYPOTHESIS SELECTION WITH ERROR BOUNDS where every optimal solution X needs at least $f(m,n)$ fractional variables. We have shown $f(m,n) \leq \min\{m,n\} + n$, and it is trivial to give general examples where the number of fractional variables must be n, so that $f(m,n) \geq n$. On the other hand, note that the “fractional knapsack” property of optimal solutions does not imply $f(m,n) \leq n$: Knapsack solutions are not always unique and may allow several fractional variables in a column j (namely if several r_{kj} are equal), and since a knapsack solution against a dual optimal q^* is not necessarily already optimal, we may have to take a solution with more fractional variables. We must leave the exact $f(m,n)$ as an open problem.

5. Is There a Faster Algorithm?

In this more informal section we briefly discuss another open problem: to devise a purely combinatorial algorithm for our class of linear programs that is faster than a generic linear program solver. We point out two ways, but also the reasons why these attempts have not been successful so far.

(1) Example 1 in the previous section shows (besides non-monotonicity of the x_{kj} in the error bounds) that, in an optimal solution, the x_{kj} in a row k are in general not simply filled up to 1 in increasing order of the p_{kj}. This is an effect of the columnwise error constraints. Nevertheless, intuition tells that larger x_{kj} are mostly assigned to smaller p_{kj}. Exceptions are structurally limited,

due to the following discussion. Let us call two matrix entries in the same row or column a *monotone pair* if the values of these entries in X and P stand in the *same* relation (larger or smaller). For an input P and a strategy X, define a directed graph $C(X)$ whose vertices are the columns, with a directed arc from i to j if $p_{ki} > p_{kj}$ and $x_{ki} > x_{kj}$ holds for some k. The directed graph $R(X)$ whose vertices are rows is defined similarly. By an argument similar to the proof of Theorem 9 we can show the existence of an optimal solution X where $B(X)$ is cycle-free and also $C(X)$ and $R(X)$ are free of *directed* cycles. Hence there is some topological order of the rows and columns such that all monotone pairs in rows and columns decrease in the same direction, for example, to the right and downwards, respectively. However this does not limit the *number* of monotone pairs. Moreover, the topological orders are not obvious from P, and even if we knew them, we could not compute the optimal X from them in a simple way. In summary, the observation above did not lead us to an efficient algorithm.

(2) Another idea is a reduction to flow problems in bipartite lossy networks. For that problem which has many other applications in transportation and finance (for example, currency exchange), purely combinatorial polynomial-time algorithms have been given by Tardos and Wayne (1998); Wayne (2002). However, the idea works only for a variant of HYPOTHESIS SELECTION WITH ERROR BOUNDS with “observation-wise” exclusiveness demands instead of a global exclusiveness objective: Recall again the weighted row sums $x_k := \sum_{i=1}^n w_i x_{ki}$. For given parameters ε_j and y_k for all j and k, respectively, we may raise the following existence problem: Is there a solution X with error probabilities at most ε_j for all H_j, and $x_k \geq y_k$ for all O_k? This problem is easily seen to be a flow problem in a bipartite lossy network with arc capacities 1 and gain factors $1/p_{kj}$; see Tardos and Wayne (1998); Wayne (2002) for the definitions. In contrast, a reduction from HYPOTHESIS SELECTION WITH ERROR BOUNDS does not seem to exist, for the intuitive reason that flow variables cannot be “copied” in order to “participate” in several linear combinations of the x_k. Still, algorithmic techniques similar to those used for flows in lossy networks might be applicable. We have to leave this subject for future research.

6. Combining Data

Suppose that we have several matrices $P^{(1)}, P^{(2)}, \ldots$ of conditional probabilities for the same set of hypotheses but for different types of observations, such as different groups of symptomes in diagnosis, or chemical shifts of several nuclei in protein torsion angle prediction. We do not assume that the joint distribution of vectors of observations is known: Since the number of vectors is the product of $m^{(1)}, m^{(2)}, \ldots$, there may be not enough cases in the database that would allow meaningful probability estimates for all these vectors. Still, combining these data sets can further narrow down the selected hypotheses (if the observations “complement each other” well), and at the same time preserve guaranteed error bounds. For ease of presentation we describe the method for two matrices, but it can be readily extended to any number.

Proposition 10 *Let P and P' be the conditional probability matrices of size $m \times n$ and $m' \times n$, respectively, and $\varepsilon_j, \varepsilon'_j$ the error bounds of two instances of* HYPOTHESIS SELECTION WITH ERROR BOUNDS *for the same set of hypotheses H_j, $j = 1, \ldots, n$. Furthermore let X and X' be strategies for these two instances that respect the given error bounds. For any pair of observations O_k, O'_l from both instances (where $k = 1, \ldots, m$ and $l = 1, \ldots, m'$), we define for all H_j the exclusion probabilities $x_{kj} + x'_{lj} - x_{kj}x'_{lj}$. Then the resulting $mm' \times n$ matrix is a strategy for combined observations with upper bound $\varepsilon_j + \varepsilon'_j$ on the probability to wrongly discard target H_j.*

Proof In fact, the combined strategy is designed so that we discard any H_j if at least one of the predictors X or X' does. The decision to discard a hypothesis is taken independently in both instances. Hence, if O_k, O'_l are observed, we keep H_j with probability

$$(1-x_{kj})(1-x'_{lj}) = 1-(x_{kj}+x'_{lj}-x_{kj}x'_{lj}).$$

On the other hand, since the probability of a union of events is at most the sum of probabilities of the single events, we discard target H_j with *at most* the probability $\varepsilon_j + \varepsilon'_j$. ■

Proposition 10 gives only a guarantee on the error probabilities. However, concavity of exclusiveness (see Proposition 7) suggests that combining two predictors with half error bound in general improves the exclusiveness. For concrete instances and a desired total error probability we may try various partitions into summands, with some resonable step length, and take the combination that works best. We also remark that, since by Theorem 8 and 9 most strategy variables x_{kj}, x'_{lj} are 0 or 1, the calculations are fast.

If several $P^{(i)}$ are available (for example, in our protein structure application, the chemical shifts of 6 nuclei, and also from neighbored residues), then exhaustive search is expensive, but we may choose to combine only the most informative data, that is, only those $P^{(i)}$ with largest exclusiveness.

Finally, a deliberately very simple, symmetric toy example with two hypotheses of equal weight illustrates the principle of combining predictions.

Example 2

$$P = \begin{bmatrix} 1 & 0.5 \\ 0 & 0.5 \end{bmatrix} \; P' = \begin{bmatrix} 0.5 & 1 \\ 0.5 & 0 \end{bmatrix}$$

We choose $\varepsilon = 0.2$ *for both hypotheses in both instances. Then the optimal solutions for the separate instances are, rather obviously:*

$$X = \begin{bmatrix} 0.2 & 0.4 \\ 1 & 0 \end{bmatrix} \; X' = \begin{bmatrix} 0.4 & 0.2 \\ 0 & 1 \end{bmatrix}.$$

In the first instance, the exclusiveness is only 0.6 if H_1 *is the target (since always* O_1 *is observed), and for* H_2 *we get exclusiveness 0.8 (average of both rows), the second instance is symmetric. If we use the information from both instances, we can improve the exclusiveness for the same* $\varepsilon = 0.2$. *First we optimize both instances separately, but now with half error bound* 0.1*:*

$$X = \begin{bmatrix} 0.1 & 0.2 \\ 1 & 0 \end{bmatrix} \; X' = \begin{bmatrix} 0.2 & 0.1 \\ 0 & 1 \end{bmatrix}.$$

For the four combinations $(k,l) = (1,1),(1,2),(2,1),(2,2)$ *we compute new exclusion probabilities as specified above:*

$$\begin{bmatrix} 0.28 & 0.28 \\ 0.1 & 1 \\ 1 & 0.1 \\ 1 & 1 \end{bmatrix}.$$

Since target H_1 *causes the pair of observations* $(1,1)$ *or* $(1,2)$*, each with probability* 0.5*, the exclusiveness is* $0.5 \cdot (0.56+1.1) = 0.83$*, and target* H_2 *yields the same exclusiveness by symmetry. This is considerably better than the worst case above, and even slightly larger than the best case above.*

7. Application Note: Protein Torsion Angle Prediction

We studied properties of a class of linear programs for hypotheses selection in probabilistic inference which is hopefully of fundamental interest. We were led to the problem class by a concrete challenge: a project where we are comparing different methods for predicting protein torsion angles from NMR chemical shifts, see Section 1. Characteristic features of our scatterplot data are large empty regions with almost no data points, in them clouds of data points with a variety of shapes and different densities. Optimization assists in the creation of a predictor:

Any prediction heuristic has to take a measured chemical shift value and output predicted torsion angle values. In a statistical approach it is sensible to precompute the predictions, based on the sampled data. The actual application is then a simple table look-up, done by an auxiliary program. The main relevant question for spectroscopists is the achievable confidence when predicting torsion angle intervals of a prescribed length (error probability vs. exclusiveness, in our terminology). Then they can make their specific decisions using this tradeoff. Besides the actual predictions, the optimization results also quantify how informative the chemical shifts of different nuclei (or their combinations) are for this purpose.

Basic heuristics working purely "row-wise" (MAP, ML, or similar) do not pay attention to error probabilities for specific hypothesis intervals and easily discard certain torsion angles completely, despite a considerable frequency of occurrence. Hence such heuristics generate systematically misleading predictions when these neglected ranges of torsion angles appear. Even worse, they can appear more frequently in a protein under consideration than in the database: Recall that the scatterplots are sampled from a large collection of various proteins so that we know only average torsion angle frequencies. A more even distribution of errors to different torsion angles gives more robustness against varying torsion angle frequencies. We can also expect that the global structure reconstruction process itself works smoother if the local restraints have balanced errors: Most wrong sequences of torsion angles, that is, sequences with errors injected, are already geometrically impossible, which gives us a chance to correct such occasional errors.[1] Since the precise effects are hard to know beforehand, free parameters ε_j seem to be a valuable feature.

A simple MAP heuristics, for example, would take the measured chemical shift value and select the torsion angle ranges (columns) with highest point densities in the row containing the measured value. Other preferences may be taken into account, for instance, one interval is easier to handle as a restraint than a union of several intervals. In either case, a selection yields a matrix X and a vector of error probabilities ε_j which are typically low (high) in densely (sparsely) populated columns. Now we can adjust error probabilities for individual torsion angle intervals in any desired direction and re-optimize.

As an illustration we discuss an arbitrary example (point count matrix) from the real data: Aspartic Acid, nucleus C^{α}, and torsion angle ϕ, partitioned into homogeneous regions using the method from Christin (2006):

1. As a linguistic analogy, typos scattered in a text can be erased promptly, whereas systematic errors make words unrecognizable, or even smuggle in other words that fit in the context but were not intended.

$$\begin{bmatrix} 1 & 0 & 7 & 12 & 4 & 1 & 1 & 0 \\ 0 & 0 & 0 & 19 & 18 & 1 & 2 & 0 \\ 5 & 4 & 22 & 212 & 116 & 16 & 4 & 0 \\ 2 & 3 & 21 & 90 & 32 & 3 & 5 & 0 \\ 10 & 6 & 38 & 93 & 28 & 7 & 39 & 3 \\ 98 & 86 & 304 & 193 & 39 & 11 & 63 & 5 \\ 34 & 43 & 86 & 27 & 4 & 0 & 7 & 3 \\ 22 & 60 & 67 & 18 & 1 & 1 & 2 & 2 \\ 3 & 12 & 19 & 6 & 4 & 0 & 2 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 40 & 30 & 30 & 20 & 15 & 80 & 45 & 100 \end{bmatrix}.$$

The bottom line gives the torsion angle interval lengths in degrees, that is, our weights. The frequencies of hypotheses in the database are (in percent, rounded):

$$\begin{bmatrix} 8.5 & 10.5 & 27.5 & 33.0 & 12.0 & 2.0 & 6.0 & 0.5 \end{bmatrix}.$$

Suppose we want to predict torsion angle intervals of about 60 degrees and start with a naive MAP heuristic that takes the intervals of exactly 60 degrees with maximum density in each row. It leads to the following matrix X (entries indicate the discarded fractions of intervals, values are rounded):

$$\begin{bmatrix} 1 & 1 & 0.17 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0.69 & 1 & 1 \\ 1 & 1 & 0.17 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0.17 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0.17 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0.67 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 100.0 & 33.0 & 2.5 & 7.5 & 19.5 & 99.0 & 100.0 & 100.0 \end{bmatrix}.$$

The bottom line indicates the error bounds ε_j in percent (rounded). For the prior probabilities from the database, the overall error probability would be about 26%. Optimization with the same ε_j yields only marginal changes:

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0.69 & 1 & 1 \\ 1 & 1 & 0.18 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0.16 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0.17 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0.67 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0.06 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

Columns 6 and 8 are completely discarded, which is reasonable because only 2.5% of cases are to be expected in the corresponding large intervals. The separate cluster in column 7 which appeared with 6% is always discarded, too. We may accept this error when we prefer a single predicted interval to a union of two (see the remark above). The most relevant part is the dense region in columns 1 to 5. Observe that also column 1 is completely discarded, even though it contains a considerable cluster of points. This makes up for 8.5 of the 26% total error. Let us reduce ε_1 and see how this affects the predictions. For instance, after changing ε_1 to 0.4, optimization (followed by raising some sporadic $x_{kj} < 1$ to 1 when p_{kj} is small) yields this X:

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0.69 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0.76 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0.26 & 0.7 & 0 & 0.03 & 1 & 1 & 1 & 1 \\ 0.16 & 0 & 0.16 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

We remark that the optimal dual solution moved from $q_3 = 1$ to $q_1 = 1$. The (expected) length of predicted intervals increased to 84 degrees. On the other hand, the global error went down to 21%, and we can afford to raise the very small initial ε_3. For instance, with $\varepsilon_3 = 0.2$ we are back to the initial total error of 26%, now with an expected hypothesis length of 77 degrees and the following X:

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0.69 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0.06 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0.46 & 0.7 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0.16 & 0 & 0.4 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

Interestingly, this step pressed the predictions in rows 7,8 more to the lower left corner, while x_{61} became higher again. The apparent reason is that, in row 6, the element in column 1 has strong competitors in columns 3 and 4. Hence it is not predicted definitely, even though $p_{61} > p_{71}, p_{81}$. The result in row 6 suggests to choose either columns 1-2 or 3-4. In order to avoid predicting intervals of excessive lengths in some rows, we may cut the longest intervals down, for example, at the end with the smallest increase of error. In our example, the longest predicted interval, with 93 degrees, appears in row 4. Changing x_{42} to 1 increases ε_2 marginally to 0.345 but shortens this interval to 65 degrees. In row 8 we may cut at column 3, etc.

This example merely served to demonstrate that desirable improvements can be made after quick manual checking, while the main calculations are left to any linear programming tool. For processing hundreds of instances with different desired interval lengths we fix the initial ε_j (subject to a proportional factor which is used for the error vs. exclusiveness tradeoff) also by other plausible heuristics: ε_j proportional to the weight-by-frequency ratio, equal ε_j, and combinations of them. However, since no simple automatic rule seems to be satisfactory for *all* diverse shapes of scatterplots (apparently bad examples exist for each), some minor intervention as shown above is required.

Acknowledgments

The first author has been supported by the Knut and Alice Wallenberg Foundation, Carl Tryggers Foundation, and Assar Gabrielsson Foundation. The second author has been partially supported by the Swedish Research Council (Vetenskapsrådet), grant no. 621-2002-4574.

We thank the anonymous referees for their helpful suggestions.

References

R. D. Beger and P. H. Bolton. Protein ϕ and ψ dihedral restraints determined from multidimensional hypersurface correlations of backbone chemical shifts and their use in the determination of protein tertiary structures. *Journal of Biomolecular NMR*, 10:129–142, 1997.

K. P. Bennett. Decision tree construction via linear programming. In *Proceedings of the 4th Midwest AI and Cognitive Science Sociecty Conference*, pages 97–101, 1992.

K. P. Bennett and O. L. Mangasarian. Neural network training via linear programming. In *Advances in Optimization and Parallel Computing*, pages 56–67. North-Holland, 1992a.

K. P. Bennett and O. L. Mangasarian. Robust linear programming discrimination of two linearly inseparable sets. *Optimization Methods and Software*, 1:23–34, 1992b.

K. P. Bennett and O. L. Mangasarian. Multicategory separation via linear programming. *Optimization Methods and Software*, 3:27–39, 1993.

P. S. Bradley. *Mathematical Programming Approaches to Machine Learning and Data Mining*. PhD thesis, University of Wisconsin, 1998.

C. Christin. Scatterplot partitioning algorithm for LETA-NMR. Master's thesis, International Master's Programme in Bioinformatics, Chalmers University, Göteborg (Sweden), 2006.

G. Cornilescu, F. Delaglio, and A. Bax. Protein backbone angle restraints from searching a database for chemical shift and sequence homology. *Journal of Biomolecular NMR*, 13:289–302, 1999.

P. Damaschke. Scheduling search procedures. *Journal of Scheduling*, 7:349–364, 2004.

F. Glover. Improved linear programming models for discriminant analysis. *Decision Sciences*, 21: 771–785, 1990.

S. Martello and P. Toth. *Knapsack Problems: Algorithms and Computer Implementations*. Wiley, 1990.

P. Szolovits, R. S. Patil, and W. B. Schwartz. Artificial intelligence in medical diagnosis. *Annals of Internal Medicine*, 108:80–87, 1988.

E. Tardos and K. D. Wayne. Simple generalized maximum flow algorithms. In *Proceedings of the 6th Integer Programming and Combinatorial Optimization Conference, Lecture Notes in Computer Science*, volume 1412, pages 310–324, 1998.

Y. Wang and O. Jardetzky. Probability-based protein secondary structure identification using combined NMR chemical-shift data. *Protein Science*, 11:852–861, 2002.

K. D. Wayne. A polynomial combinatorial algorithm for generalized minimum cost flow. *Mathematical Operations Research*, 27:445–459, 2002.

X. P. Xu and D. A. Case. Probing multiple effects on ^{15}N, $^{13}C^{\alpha}$, $^{13}C^{\beta}$ and $^{13}C'$ chemical shifts in peptides using density functional theory. *Biopolymers*, 65:408–423, 2002.