An Inductive Logic Programming Approach to Validate Hexose Binding Biochemical Knowledge

Houssam Nassif[1,2], Hassan Al-Ali[3], Sawsan Khuri[4,5], Walid Keirouz[6], and David Page[1,2]

[1] Department of Computer Sciences,
[2] Department of Biostatistics and Medical Informatics,
University of Wisconsin-Madison, USA
[3] Department of Biochemistry and Molecular Biology,
[4] Center for Computational Science, University of Miami,
[5] The Dr. John T. Macdonald Foundation Department of Human Genetics,
University of Miami Miller School of Medicine, Florida, USA
[6] Department of Computer Science, American University of Beirut, Lebanon

Abstract. Hexoses are simple sugars that play a key role in many cellular pathways, and in the regulation of development and disease mechanisms. Current protein-sugar computational models are based, at least partially, on prior biochemical findings and knowledge. They incorporate different parts of these findings in predictive black-box models. We investigate the empirical support for biochemical findings by comparing Inductive Logic Programming (ILP) induced rules to actual biochemical results. We mine the Protein Data Bank for a representative data set of hexose binding sites, non-hexose binding sites and surface grooves. We build an ILP model of hexose-binding sites and evaluate our results against several baseline machine learning classifiers. Our method achieves an accuracy similar to that of other black-box classifiers while providing insight into the discriminating process. In addition, it confirms wet-lab findings and reveals a previously unreported Trp-Glu amino acids dependency.

Key words: ILP, Aleph, rule generation, hexose, protein-carbohydrate interaction, binding site, substrate recognition

1 Introduction

Inductive Logic Programming (ILP) has been shown to perform well in predicting various substrate-protein bindings (e.g., [9, 26]). In this paper we apply ILP to a different and well studied binding task.

Hexoses are 6-carbon simple sugar molecules that play a key role in different biochemical pathways, including cellular energy release, signaling, carbohydrate synthesis, and the regulation of gene expression [24]. Hexose binding proteins belong to diverse functional families that lack significant sequence or, often, structural similarity [16]. Despite this fact, these proteins show high specificity to their hexose ligands. The few amino acids (also called residues) present at

the binding site play a large role in determining the binding site's distinctive topology and biochemical properties and hence the ligand type and the protein's functionality.

Wet-lab experiments discover hexose-protein properties. Computational hexose classifiers incorporate different parts of these findings in black-box models as the base of prediction. No work to date has taken the opposite approach: given hexose binding sites data, what biochemical rules can we extract with no prior biochemical knowledge, and what is the performance of the resulting classifier based solely on the extracted rules?

This work presents an ILP classifier that extracts rules from the data without prior biochemical knowledge. It classifies binding sites based on the extracted biochemical rules, clearly specifying the rules used to discriminate each instance. Rule learning is especially appealing because of its easy-to-understand format. A set of if-then rules describing a certain concept is highly expressive and readable [18]. We evaluate our results against several baseline machine learning classifiers. This inductive data-driven approach validates the biochemical findings and allows a better understanding of the black-box classifiers' output.

2 Previous Work

Although no previous work tackled data-driven rule generation or validation, many researchers studied hexose binding.

2.1 Biochemical Findings

From the biochemical perspective, Rao et al. [21] fully characterized the architecture of sugar binding in the Lectin protein family and identified conserved loop structures as essential for sugar recognition. Later, Quiocho and Vyas [20] presented a review of the biochemical characteristics of carbohydrate binding sites and identified the planar polar residues (Asn, Asp, Gln, Glu, Arg) as the most frequently involved residues in hydrogen bonding. They also found that the aromatic residues Trp, Tyr, and Phe, as well as His, stack against the apolar surface of the sugar pyranose ring. Quiocho and Vyas also pinpointed the role of metal ions in determining substrate specificity and affinity. Ordered water molecules bound to protein surfaces are also involved in protein-ligand interaction [15].

Taroni et al. [29] analyzed the characteristic properties of sugar binding sites and described a residue propensity parameter that best discriminates sugar binding sites from other protein-surface patches. They also note that simple sugars typically have a hydrophilic side group which establishes hydrogen bonds and a hydrophobic core that is able to stack against aromatic residues. Sugar binding sites are thus neither strictly hydrophobic nor strictly hydrophilic, due to the dual nature of sugar docking. In fact, as García-Hernández et al. [11] showed, some polar groups in the protein-carbohydrate complex behave hydrophobically.

2.2 Computational Models

Some of this biochemical information has been used in computational work with the objective of accurately predicting sugar binding sites in proteins. Taroni et al. [29] devised a probability formula by combining individual attribute scores. Shionyu-Mitsuyama et al. [23] used atom type densities within binding sites to develop an algorithm for predicting carbohydrate binding. Chakrabarti et al. [5] modeled one glucose binding site and one galactose binding site by optimizing their binding affinity under geometric and folding free energy constraints. Other researchers formulated a signature for characterizing galactose binding sites based on geometric constraints, pyranose ring proximity and hydrogen bonding atoms [27, 28]. They implemented a 3D structure searching algorithm, COTRAN, to identify galactose binding sites.

More recently, researchers used machine learning algorithms to model hexose binding sites. Malik and Ahmad [17] used a Neural Network to predict general carbohydrate as well as specific galactose binding sites. Nassif et al. [19] used Support Vector Machines to model and predict glucose binding sites in a wide range of proteins.

3 Data Set

The Protein Data Bank (PDB) [2] is the largest repository of experimentally determined and hypothetical three-dimensional structures of biological macromolecules. We mine it for proteins crystallized with the most common hexoses: galactose, glucose and mannose [10]. We ignore theoretical structures and files older than PDB format 2.1. We eliminate redundant structures using PISCES [30] with a 30% overall sequence identity cut-off. We use Swiss-PDBViewer [14] to detect and discard sites that are glycosylated or within close proximity to other ligands. We check the literature to ensure that no hexose-binding site also binds non-hexoses. The final outcome is a non-redundant positive data set of 80 protein-hexose binding sites (Table 1).

We also extract an equal number of negative examples. The negative set is composed of non-hexose binding sites and of non-binding surface grooves. We choose 22 binding-sites that bind hexose-like ligands: hexose or fructose derivatives, 6-carbon molecules, and molecules similar in shape to hexoses (Table 2). We also select 27 other-ligand binding sites, ligands who are bigger or smaller than hexoses (Table 2). Finally, we specify 31 non-binding sites: protein surface grooves that look like binding-sites but are not known to bind any ligand (Table 3).

We use 10-folds cross-validation to train, test and validate our approach. We divide the data set in 10 stratified folds, thus preserving the proportions of the original set labels and sub-groups.

Table 1. Inventory of the hexose-binding positive data set

Hexose	PDB ID	Ligand	PDB ID	Ligand	PDB ID	Ligand
Glucose	1BDG	GLC-501	1ISY	GLC-1471	1SZ2	BGC-1001
	1EX1	GLC-617	1J0Y	GLC-1601	1SZ2	BGC-2001
	1GJW	GLC-701	1JG9	GLC-2000	1U2S	GLC-1
	1GWW	GLC-1371	1K1W	GLC-653	1UA4	GLC-1457
	1H5U	GLC-998	1KME	GLC-501	1V2B	AGC-1203
	1HIZ	GLC-1381	1MMU	GLC-1	1WOQ	GLC-290
	1HIZ	GLC-1382	1NF5	GLC-125	1Z8D	GLC-901
	1HKC	GLC-915	1NSZ	GLC-1400	2BQP	GLC-337
	1HSJ	GLC-671	1PWB	GLC-405	2BVW	GLC-602
	1HSJ	GLC-672	1Q33	GLC-400	2BVW	GLC-603
	1I8A	GLC-189	1RYD	GLC-601	2F2E	AGC-401
	1ISY	GLC-1461	1S5M	AGC-1001		
Galactose	1AXZ	GLA-401	1MUQ	GAL-301	1R47	GAL-1101
	1DIW	GAL-1400	1NS0	GAL-1400	1S5D	GAL-704
	1DJR	GAL-1104	1NS2	GAL-1400	1S5E	GAL-751
	1DZQ	GAL-502	1NS8	GAL-1400	1S5F	GAL-104
	1EUU	GAL-2	1NSM	GAL-1400	1SO0	GAL-500
	1ISZ	GAL-461	1NSU	GAL-1400	1TLG	GAL-1
	1ISZ	GAL-471	1NSX	GAL-1400	1UAS	GAL-1501
	1JZ7	GAL-2001	1OKO	GLB-901	1UGW	GAL-200
	1KWK	GAL-701	1OQL	GAL-265	1XC6	GAL-9011
	1L7K	GAL-500	1OQL	GAL-267	1ZHJ	GAL-1
	1LTI	GAL-104	1PIE	GAL-1	2GAL	GAL-998
Mannose	1BQP	MAN-402	1KZB	MAN-1501	1OUR	MAN-301
	1KLF	MAN-1500	1KZC	MAN-1001	1QMO	MAN-302
	1KX1	MAN-20	1KZE	MAN-1001	1U4J	MAN-1008
	1KZA	MAN-1001	1OP3	MAN-503	1U4J	MAN-1009

4 Problem Representation

In this work, we first extract multiple chemical and spatial features from the binding site. We then apply ILP to generate rules and classify our data set.

4.1 Binding Site Representation

We view the binding site as a sphere centered at the ligand. We compute the center of the hexose-binding site as the centroid of the coordinates of the hexose pyranose ring's six atoms. For negative sites, we use the center of the cavity or the ligand's central point. The farthest pyranose-ring atom from the ring's centroid is located 2.9 Å away. Bobadilla et al. [4] consider atomic interactions to be significant within a 7 Å range. We thereby fix the binding site sphere radius to 10 Å. Given the molecule and the binding site centroid, we extract all atoms within the sphere. We include water molecules and ions present in the

Table 2. Inventory of the non-hexose-binding negative data set

PDB ID	Cavity Center	Ligand	PDB ID	Cavity Center	Ligand
Hexose-like ligands					
1A8U	4320, 4323	BEZ-1	1AI7	6074, 6077	IPH-1
1AWB	4175, 4178	IPD-2	1DBN	pyranose ring	GAL-102
1EOB	3532, 3536	DHB-999	1F9G	5792, 5785, 5786	ASC-950
1G0H	4045, 4048	IPD-292	1JU4	4356, 4359	BEZ-1
1LBX	3941, 3944	IPD-295	1LBY	3944, 3939, 3941	F6P-295
1LIU	15441, 15436, 15438	FBP-580	1MOR	pyranose ring	G6P-609
1NCW	3406, 3409	BEZ-601	1P5D	pyranose ring	G1P-658
1T10	4366, 4361, 4363	F6P-1001	1U0F	pyranose ring	G6P-900
1UKB	2144, 2147	BEZ-1300	1X9I	pyranose ring	G6Q-600
1Y9G	4124, 4116, 4117	FRU-801	2B0C	pyranose ring	G1P-496
2B32	3941, 3944	IPH-401	4PBG	pyranose ring	BGP-469
Other ligands					
11AS	5132	ASN-1	11GS	1672, 1675	MES-3
1A0J	6985	BEN-246	1A42	2054, 2055	BZO-555
1A50	4939, 4940	FIP-270	1A53	2016, 2017	IGP-300
1AA1	4472, 4474	3PG-477	1AJN	6074, 6079	AAN-1
1AJS	3276, 3281	PLA-415	1AL8	2652	FMN-360
1B8A	7224	ATP-500	1BO5	7811	GOL-601
1BOB	2566	ACO-400	1D09	7246	PAL-1311
1EQY	3831	ATP-380	1IOL	2674, 2675	EST-400
1JTV	2136, 2137	TES-500	1KF6	16674, 16675	OAA-702
1RTK	3787, 3784	GBS-300	1TJ4	1947	SUC-1
1TVO	2857	FRZ-1001	1UK6	2142	PPI-1300
1W8N	4573, 4585	DAN-1649	1ZYU	1284, 1286	SKM-401
2D7S	3787	GLU-1008	2GAM	11955	NGA-502
3PCB	3421, 3424	3HB-550			

Table 3. Inventory of the non-binding surface groove negative data set

PDB ID	Cavity Center	PDB ID	Cavity Center	PDB ID	Cavity Center
1A04	1424, 2671	1A0I	1689, 799	1A22	2927
1AA7	579	1AF7	631, 1492	1AM2	1277
1ARO	154, 1663	1ATG	1751	1C3G	630, 888
1C3P	1089, 1576	1DXJ	867, 1498	1EVT	2149, 2229
1FI2	1493	1KLM	4373, 4113	1KWP	1212
1QZ7	3592, 2509	1YQZ	4458, 4269	1YVB	1546, 1814
1ZT9	1056, 1188	2A1K	2758, 3345	2AUP	2246
2BG9	14076, 8076	2C9Q	777	2CL3	123, 948
2DN2	749, 1006	2F1K	316, 642	2G50	26265, 31672
2G69	248, 378	2GRK	369, 380	2GSE	337, 10618
2GSH	6260				

binding groove [15, 19, 20]. We discard hydrogen atoms since most PDB entries lack them. We do not extract residues.

For every extracted atom we record its PDB-coordinates, its charge, hydrogen bonding, and hydrophobicity properties, and its atomic element and name. Every PDB file has orthogonal coordinates and all atom positions are recorded accordingly. We compute atomic properties as done by Nassif et al. [19]. The partial charge measure per atom is positive, neutral, or negative; atoms can form hydrogen bonds or not; hydrophobicity measures are considered as hydrophobic, hydroneutral, or hydrophilic. Finally, every PDB-atom has an atomic element and a specific name. For example, the residue histidine (HIS) has a particular Nitrogen atom named ND1. This atom's element is Nitrogen, and name is ND1. Since ND1 atoms only occur in HIS residues, recording atomic names leaks information about their residues.

4.2 Aleph Settings

We use the ILP engine Aleph [25] to learn first-order rules. We run Aleph within Yap Prolog [22]. To speed the search, we use Aleph's heuristic search. We estimate the classifier's performance using 10-fold cross-validation.

We limit Aleph's running time by restricting the clause length to a maximum of 8 literals, with only one in the head. We set the Aleph parameter *explore* to true, so that it will return all optimal-scoring clauses, rather than a single one, in a case of a tie. The consequent of any rule is $bind(+site)$, where *site* is predicted to be a hexose binding site. No literal can contain terms pertaining to different binding sites. As a result, *site* is the same in all literals in a clause.

The literal describing the binding site center is:

$$point(+site, -id, -X, -Y, -Z) \tag{1}$$

where *site* is the binding site and *id* is the binding center's unique identifier. X, Y, and Z specify the PDB-Cartesian coordinates of the binding site's centroid.

Literals describing individual PDB-atoms are of the form:

$$point(+site, -id, -X, -Y, -Z, -charge, -hbond, -hydro, -elem, -name) \tag{2}$$

where *site* is the binding site and *id* is the individual atom's unique identifier. X, Y, and Z specify the PDB-Cartesian coordinates of the atom. *charge* is the partial charge, *hbond* the hydrogen-bonding, and *hydro* the hydrophobicity. Lastly, *elem* and *name* refer to the atomic element and its name (see last paragraph of previous section).

Clause bodies can also use distance literals:

$$dist(+site, +id, +id, \#distance, \#error)\,. \tag{3}$$

The *dist* predicate, depending on usage, either computes or checks the *distance* between two points. *site* is the binding site and the *id*s are two unique point identifiers (two PDB-atoms or one PDB-atom and one center). *distance* is their

Euclidean distance apart and *error* the tolerated distance error, resulting in a matching interval of $distance \pm error$. We set *error* to 0.5 Å.

We want our rules to refer to properties of PDB-atoms, such as "an atom's name is ND1", or "an atom's charge is not positive". Syntactically we do this by relating PDB-atoms' variables to constants using "equal" and "not equal" literals:

$$equal(+setting, \#setting)\,, \tag{4}$$

$$not\_equal(+feature, \#feature)\,. \tag{5}$$

$feature$ is the atomic features *charge*, *hbond* and *hydro*. In addition to these atomic features, *setting* includes *elem* and *name*.

Aleph keeps learning rules until it has covered all the training positive set, and then it labels a test example as positive if *any* of the rules cover that example. This has been noted in previous publications to produce a tendency toward giving more false positives [6, 7]. To limit our false positives count, we restrict coverage to a maximum of 5 training-set negatives. Since our approach seeks to validate biological knowledge, we aim for high precision rules. Restricting negative rule coverage also biases generated rules towards high precision.

5 Results

The Aleph testing set error averaged to 32.5% with a standard deviation of 10.54%. The confidence interval is $[24.97\%, 40.03\%]$ at the 95% confidence level. Refer to Table 5 for the 10-folds cross-validation accuracies.

To generate the final set of rules, we run Aleph over the whole data set. We discard rules with $pos\_cover - neg\_cover \leq 2$. Even though Aleph was only looking at atoms, valuable information regarding amino acids can be inferred. For example ND1 atoms are only present within the amino acid His, and a rule requiring the presence of ND1 is actually requiring His. We present the rules' biochemical translation while replacing specific atoms by the amino acids they imply. The queried site is considered hexose binding if any of these rules apply:

1. It contains a Trp residue and a Glu with an OE1 Oxygen atom that is 8.53 Å away from an Oxygen atom with a negative partial charge (Glu, Asp amino acids, Sulfate, Phosphate, residue C-terminus Oxygen).
   [Pos cover = 22, Neg cover = 4]

2. It contains a Trp, Phe or Tyr residue, an Asp and an Asn. Asp and an Asn's OD1 Oxygen atoms are 5.24 Å apart.
   [Pos cover = 21, Neg cover = 3]

3. It contains a Val or Ile residue, an Asp and an Asn. Asp and Asn's OD1 Oxygen atoms are 3.41 Å apart.
   [Pos cover = 15, Neg cover = 0]

4. It contains a hydrophilic non-hydrogen bonding Nitrogen atom (Pro, Arg) with a distance of 7.95 Å away from a His's ND1 Nitrogen atom, and 9.60 Å away from a Val or Ile's CG1 Carbon atom.
   [Pos cover = 10, Neg cover = 0]

5. It has a hydrophobic CD2 Carbon atom (Leu, Phe, Tyr, Trp, His), a Pro, and two hydrophilic OE1 Oxygen atoms (Glu, Gln) 11.89 Å apart.
   [Pos cover = 11, Neg cover = 2]

6. It contains an Asp residue B, two identical atoms Q and X, and a hydrophilic hydrogen-bonding atom K. Atoms K, Q and X have the same charge. B's OD1 Oxygen atom share the same Y-coordinate with K and the same Z-coordinate with Q. Atoms X and K are 8.29 Å apart.
   [Pos cover = 8, Neg cover = 0]

7. It contains a Ser residue, and two NE2 Nitrogen atoms (Gln, His) 3.88 Å apart.
   [Pos cover = 8, Neg cover = 2]

8. It contains an Asn residue and a Phe, Tyr or His residue, whose CE1 Carbon atom is 7.07 Å away from a Calcium ion.
   [Pos cover = 5, Neg cover = 0]

9. It contains a Lys or Arg, a Phe, Tyr or Arg, a Trp, and a Sulfate or a Phosphate ion.
   [Pos cover = 3, Neg cover = 0]

Most of these rules closely reproduce current biochemical knowledge. One in particular is novel. We will discuss rule relevance in Section 7.2.

6 Experimental Evaluation

We evaluate our performance by comparing Aleph to several baseline machine learning classifiers.

6.1 Feature Vector Representation

Unlike Aleph, the implemented baseline algorithms require a constant-length feature vector input. We change our binding-site representation accordingly. We subdivide the binding-site sphere into concentric shells as suggested by Bagley and Altman [1]. Nassif et al. [19] subdivided the sphere into 8 layers centered at the binding-site centroid. The first layer had a width of 3 Å and the subsequent 7 layers where 1 Å each. Their results show that the layers covering the first 5 Å, the subsequent 3 Å and the last 2 Å share several attributes. We thereby subdivide our binding-site sphere into 3 concentric layers, with layer width of

5 Å, 3 Å and 2 Å respectively. For each layer, our algorithm reports the total number of atoms in that layer and the fraction of each atomic property (charge, hydrogen-bonding, hydrophobicity). For example, feature "layer 1 hydrophobic atoms" represents the fraction of the first layer atoms that are hydrophobic.

The ILP predicate representation allows it to implicitly infer residues from atomic names and properties. We use a weakly expressive form to explicitly include amino acids in the feature vector representation. Amino acids are categorized into subgroups, based on their structural and chemical properties [3]. We base our scheme on the representation adopted by Nassif et al. [19], grouping histidine, previously a subclass on its own, with the rest of the aromatic residues. Histidine can have roles which are unique among the aromatic amino acids. We group it with other aromatics because it was not selected as a relevant feature in our previous work. Gilis et al. [12] report the mean frequencies of the individual amino acids in the proteomes of 35 living organisms. Adding up the respective frequencies, we get the expected percentage p of each residue category. We categorize the residue features into "low", "normal" and "high". A residue category feature is mapped to "normal" if its percentage is within $2 \times \sqrt{p}$ of the expected value p. It is mapped to "low" if it falls below, and to "high" if it exceeds the cut-off. Table 4 accounts for the different residue categories, their expected percentages, and their cut-off values mapping boundaries. Given a binding site, our algorithm computes the percentage of amino acids of each group present in the sphere, and records its nominal value. We ignore the concentric layers, since a single residue can span several layers.

Table 4. Residue grouping scheme, expected percentage, and mapping boundaries

Residue Category	Amino Acids	Expected Percentage	Lower Bound	Upper Bound
Aromatic	His, Phe, Trp, Tyr	10.81%	4.23%	17.39%
Aliphatic	Ala, Ile, Leu, Met, Val	34.19%	22.50%	45.88%
Neutral	Asn, Cys, Gln, Gly, Pro, Ser, Thr	31.53%	20.30%	42.76%
Acidic	Asp, Glu	11.91%	5.01%	18.81%
Basic	Arg, Lys	11.55%	4.75%	18.35%

The final feature vector is a concatenation of the atomic and residue features. It contains the total number of atoms and the atomic property fractions for each layer, in addition to the residue features. It totals 27 continuous and 5 nominal features.

6.2 Baseline Classifiers

This section details our implementation and parametrization of the baseline algorithms. Refer to Mitchell [18] and Duda et al. [8] for a complete description of the algorithms.

***k*-Nearest Neighbor** The scale of the data has a direct impact on k-Nearest Neighbor's (kNN) classification accuracy. A feature with a high data mean and small variance will a priori influence classification more than one with a small mean and high variance, regardless of their discrimination power [8]. In order to put equal initial weight on the different features, the data is standardized by scaling and centering.

Our implementation handles nominal values by mapping them to ordinal numbers. It uses the Euclidean distance as a distance function. It chooses the best k via a leave-one-out tuning method. Whenever two or more k's yield the same performance, it adopts the larger one. If two or more examples are equally distant from the query, and all may be the kth nearest neighbor, our implementation randomly chooses. On the other hand, if a decision output tie arises, the query is randomly classified.

We also implement feature backward-selection (BSkNN) using the steepest-ascent hill-climbing method. For a given feature set, it removes one feature at a time and performs kNN. It adopts the trial leading to the smaller error. It repeats this cycle until removing any additional feature increases the error rate. This implementation is biased towards shorter feature sets, going by Occam's razor principle.

Naive Bayes Our Naive Bayes (NB) implementation uses a Laplacian smoothing function. It assumes that continuous features, for each output class, follow the Gaussian distribution. Let X be a continuous feature to classify and Y the class. To compute $P(X|Y)$, it first calculates the normal z-score of X given Y using the Y-training set's mean μ_Y and standard deviation s_Y: $z_Y = (x - \mu_Y)/s_Y$. It then converts the z-score into a $[0, 1]$ number by integrating the portions of the normal curve that lie outside $\pm z$. We use this number to approximate $P(X|Y)$. This method returns 1 if $X = \mu$, and decreases as X steps away from μ:

$$P(X|Y) = \int_{|z_Y|}^{\infty} normalCurve + \int_{-|z_Y|}^{-\infty} normalCurve\,. \tag{6}$$

Decision Trees Our Decision Tree implementation uses information gain as a measure for the effectiveness of a feature in classifying the training data. We incorporate continuous features by dynamically defining new discrete-valued attributes that partition the continuous attribute value into a discrete set of intervals. We prune the resulting tree using a tuning set. We report the results of both pruned (Pr DT) and unpruned decision trees (DT).

Perceptron Our perceptron (Per) implementation uses linear units and performs a stochastic gradient descent. It is therefore similar to a logistic regression. It automatically adjusts the learning rate, treats the threshold as another weight, and uses a tuning set for early stopping to prevent overfitting. We limit our runs to a maximum of 1000 epochs.

Sequential Covering Sequential Covering (SC) is a propositional rules learner that returns a set of disjunctive rules covering a subset of the positive examples. Our implementation uses a greedy approach. It starts from the empty set and greedily adds the best attribute that improves rule performance. It discretizes continuous attributes using the same method as Decision Trees. It sets the rule coverage threshold to 4 positive examples and no negative examples. The best attribute to add is the one maximizing:

$$|entropy(parent) - entropy(child)| * numberOfPositives(child)\,. \quad (7)$$

6.3 Baseline Classifiers Results

We apply the same 10-folds cross-validation to Aleph and all the baseline classifiers. Table 5 tabulates the error percentage per testing fold, the mean, standard deviation and the 95% level confidence interval for each classifier.

Table 5. 10-folds cross-validation test error percentage, mean, standard deviation and the 95% level confidence interval for the baseline algorithms and Aleph

Fold	kNN	BSkNN	NB	DT	Pr DT	Per	SC	Aleph
0	25.0	25.0	43.75	31.25	37.5	43.75	31.25	25.0
1	25.0	25.0	25.0	31.25	25.0	43.75	31.25	37.5
2	18.75	18.75	25.0	12.5	25.0	25.0	25.0	25.0
3	18.75	18.75	37.5	6.25	12.5	31.25	12.5	50.0
4	25.0	37.5	37.5	25.0	37.5	25.0	12.5	31.25
5	31.25	31.25	37.5	31.25	18.75	37.5	31.25	18.75
6	31.25	18.75	25.0	37.5	31.25	37.5	25.0	25.0
7	31.25	25.0	37.5	25.0	31.25	31.25	37.5	43.75
8	18.75	18.75	31.25	25.0	12.5	31.25	31.25	25.0
9	31.25	31.25	50.0	50.0	31.25	43.75	25.0	43.75
mean	25.63	25.0	35.0	27.5	26.25	35.0	26.25	32.5
standard deviation	5.47	6.59	8.44	12.22	9.22	7.34	8.23	10.54
lower bound	21.71	20.29	28.97	18.77	19.66	29.76	20.37	24.97
upper bound	29.54	29.71	41.03	36.23	32.84	40.24	32.13	40.03

Our SC implementation learns a propositional rule that covers at least 4 positive and no negative examples. It then removes all positive examples covered by the learned rule. It repeats the process using the remaining positive examples. Running SC over the whole data set generates the following rules, sorted by coverage. Together they cover 63 positives out of 80. A site is hexose-binding if any of these rules apply:

1. **If** layer 1 negatively charged atoms density > 0.0755
   and layer 2 positively charged atoms density < 0.0155

and layer 3 negatively charged atoms density > 0.0125
[Pos cover = 32]

2. **If** layer 1 non hydrogen-bonding atoms density < 0.559
and layer 1 hydrophobic atoms density > 0.218
and layer 3 hydrophilic atoms density > 0.3945
[Pos cover = 14]

3. **If** layer 1 negatively charged atoms density > 0.0665
and layer 1 hydroneutral atoms density < 0.2615
and layer 1 non hydrogen-bonding atoms density > 0.3375
and layer 3 atoms number < 108.5
[Pos cover = 12]

4. **If** layer 1 negatively charged atoms density > 0.0665
and layer 2 atoms number > 85.5
and layer 1 negatively charged atoms density < 0.3485
[Pos cover = 5]

7 Discussion

Despite its average performance, the main advantage of ILP is the insight it provides to the underlying discrimination process.

7.1 Aleph's Performance

Aleph's error rate of 32.5% has a p-value < 0.0002, according to a two-sided binomial test. Random guessing would return 50%, since the number of positives and negatives are equal. According to a paired t-test at the 95% confidence level, the difference between Aleph and each of the baseline algorithms is not statistically significant. Aleph's mean error rate (32.5%) and standard deviation (10.54%) are within the ranges observed for the baseline classifiers, $[25\%, 35\%]$ and $[5.47\%, 12.22\%]$ respectively (see Table 5).

Aleph's error rate is also comparable to other general sugar binding site classifiers, ranging from 31% to 39%, although each was run on a different data set (Table 6). On the other hand, specific sugar binding sites classifiers have a much better performance (Table 6). COTRAN [27] galactose-binding site classifier achieves a 5.09% error while Nassif et al. [19] glucose-binding site classifier reports an error of 8.11%. This may suggest that the problem of recognizing specific sugars is easier than classifying a family of sugars.

7.2 Aleph Rules Interpretation

Contrary to black-box classifiers, ILP provides a number of interesting insights. It infers most of the established biochemical information about residues and

Table 6. Error rates achieved by general and specific sugar binding site classifiers. Not meant as a direct comparison since the data sets are different.

Program	Error (%)	Method and Data set
General sugar binding sites classifiers		
ILP hexose predictor	32.50	10-folds cross-validation, 80 hexose and 80 non-hexose or non-binding sites
Shionyu-Mitsuyama et al. [23]	31.00	Test set, 61 polysaccharide binding sites
Taroni et al. [29]	35.00	Test set, 40 carbohydrate binding sites
Malik and Ahmad [17]	39.00	Leave-one-out, 40 carbohydrate and 116 non-carbohydrate binding sites
Specific sugar binding sites classifiers		
COTRAN [27]	5.09	Overall performance over 6-folds, totaling 106 galactose and 660 non-galactose binding sites
Nassif et al. [19]	8.11	Leave-one-out, 29 glucose and 35 non-glucose or non-binding sites

relations just from the PDB-atom names and properties. We hereby interpret Aleph's rules detailed in Section 5.

Rules 1, 2, 5, 8 and 9, rely on the aromatic residues Trp, Tyr and Phe. This highlights the docking interaction between the hexose and the aromatic residues [17, 27, 28]. The aromatic residues stack against the apolar sugar pyranose ring which stabilizes the bound hexose. His is mentioned in many of the rules, along-side other aromatics (5, 8) or on its own (4, 7). Histidine provides a similar docking mechanism to Trp, Tyr and Phe [20].

All nine rules require the presence of a planar polar residue (Asn, Asp, Gln, Glu, Arg). These residues have been identified as the most frequently involved in the hydrogen-bonding of hexoses [20]. The hydrogen bond is probably the most relevant interaction in protein binding in general.

Rules 1, 2, 3, 5 and 6 call for acidic residues with a negative partial charge (Asp, Glu), or for atoms with a negative partial charge. The relative high negative density observed may be explained by the dense hydrogen-bond network formed by the hexose hydroxyl groups.

Some rules require hydrophobic atoms and residues, while others require hydrophilic ones. Rule 5 requires both and reflects the dual nature of sugar docking, composed of a polar-hydrophilic aspect establishing hydrogen bonds and a hydrophobic aspect responsible for the pyranose ring stacking [29].

A high residue-sugar propensity value reflects a high tendency of that residue to be involved in sugar binding. The residues having high propensity values are the aromatic residues, including histidine, and the planar polar residues [29]. This fact is reflected by the recurrence of high propensity residues in all rules.

Rules 8 and 9 require, and rule 1 is satisfied by, the presence of different ions (Calcium, Sulfate, Phosphate), confirming the relevance of ions in hexose binding [20].

Rule 6 specifies a triangular conformation of three atoms within the binding-site. This highlights the relevance of the binding-site's spatial features. On the other hand, we note the absence of the site's centroid literal from the resulting Aleph rules. The center is merely a geometric parameter and does not have any functional role. In fact, the binding site center feature was not used in most computational classifying approaches. Taroni et al. [29] and Malik and Ahmad [17] ignore it, Shionyu-Mitsuyama et al. [23] use the pyranose ring $C3$ atom instead, and Nassif et al. [19] indirectly use it to normalize distances from atoms within the binding pocket to atoms in the ligand. Only Sujatha and Balaji [27] explicitly refer to the center. These results confirm that the biochemical composition of the binding-site and the relative 3-dimensional positioning of its atoms play a much more important role in substrate specificity than the exact location of the ligand's center.

Rao et al. [21] report a dependency between PHE/TYR and ASN/ASP in the Lectin protein family. This dependency is reflected in rules 2 and 8. Similarly, rule 1 suggests a dependency between TRP and GLU, a link not previously identified in literature. This novel relationship merits further investigation and highlights the rule-discovery potential of ILP.

7.3 Baseline Algorithms Insight

In addition to providing a basis for comparison with Aleph, the baseline algorithms shed additional light on our data set and hexose binding site properties.

Naive Bayes and Perceptron return the highest mean error rates, 35.0%. Naive Bayes is based on the simplifying assumption that the attribute values are conditionally independent given the target value. This assumption does not hold for our data. In fact, charge, hydrogen-bonding and hydrophobicity values are correlated. Like Naive Bayes, Perceptron correctly classifies linearly separable data. Its high error stresses the fact that our data is not linearly separable.

On the other hand, backward-selection kNN algorithm, with the lowest mean error rate (25.0%), outperforms all other classifiers. This provides further evidence that similar sites, in terms of biochemical and spatial properties, are good predictors of binding [13]. kNN's good performance further highlights both the correlation between our features, and the data's non-linearity.

Like Aleph, Sequential Covering's rules provide insight into the discriminating process. Unlike Aleph's first-order logic rules, propositional rules are less expressive and reflect a smaller number of biochemical properties. We hereby interpret SC's rules detailed in Section 6.3.

Although SC uses an explicit representation of residues, it completely ignores them in the retained rules. Only atomic biochemical features influence the prediction. This may be due to the fact that it is the binding-site's atoms, rather

than overall residues, that bind to and stabilize the docked hexose. These atoms may not be mapped to conserved specific residues.

Another general finding is that most rule antecedents are layer 1 features. This reflects the importance of the atoms lining the binding-site, which establish direct contact with the docking hexose. Layers 2 and 3 are farther away and hence have weaker interaction forces.

Only four amino acid atoms have a partial negative charge in our representation, in addition to the infrequent Sulfate and Phosphate Oxygens [19]. The first rule, covering most of the positive examples, clearly suggests a binding-site with a high density of negatively charged atoms. The first and third antecedents explicitly specify layers with a negatively charged atomic density above some thresholds. The second one implicitly states so by opting for a non-positively charged layer. The relative high negative density observed may be explained by the dense hydrogen-bond network formed by the hexose hydroxyl groups [20]. The fourth rule is similar. It imposes a bond on the first layer's negative charge, between 0.0665 and 0.3485. Although it is well established in the literature that hexose forms hydrogen bonds through its hydrogen's partial positive charge [31], the binding site itself is not known to be negatively charged. Rule 4 captures this distinction, requiring the binding site to have an above-average density of negatively charged atoms, while still setting upper-bounds.

The second rule requires a high density of hydrophobic atoms in layer 1, and a high density of hydrophilic atoms in layer 3. This reflects the dual hydrophobic-hydrophilic nature of hexose binding [29]. It also indirectly specifies a high hydrogen-bonding density by imposing an upper limit for the non-hydrogen bonding atoms.

The third rule is a combination of the other ones. First it demands a slightly negative first layer. Second, it requires hydroneutral atoms. Third it implicitly asks for hydrogen bonding atoms by setting a relatively low threshold for the much more abundant non-hydrogen bonding atoms. It is worth to note that the number of atoms in the third layer is capped by 108. This may reflect a particular spatial-arrangement of atoms in hexose binding sites.

8 Conclusion

In this work, we present the first attempt to model and predict hexose binding sites using ILP. We investigate the empirical support for biochemical findings by comparing Aleph induced rules to actual biochemical results. Our ILP system achieves a similar accuracy as other general protein-sugar binding sites black-box classifiers, while offering insight into the discriminating process. With no prior biochemical knowledge, Aleph was able to induce most of the known hexose-protein interaction biochemical rules, with a performance that is not significantly different than several baseline algorithms. In addition, ILP finds a previously unreported dependency between Trp and Glu, a novel relationship that merits further investigation.

Acknowledgments. This work was partially supported by US National Institute of Health (NIH) grant R01CA127379-01. We would like to thank Jose Santos for his inquiries that led us to improve our Aleph representation.

References

1. Bagley, S.C., Altman, R.B.: Characterizing the microenvironment surrounding protein sites. Protein Science 4(4), 622–635 (1995)
2. Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., Bourne, P.E.: The protein data bank. Nucleic Acids Research 28(1), 235–242 (2000)
3. Betts, M.J., Russell, R.B.: Amino acid properties and consequences of substitutions. In: Barnes, M.R., Gray, I.C. (eds.) Bioinformatics for Geneticists, pp. 289–316. John Wiley & Sons, West Sussex, UK (2003)
4. Bobadilla, L., Nino, F., Narasimhan, G.: Predicting and characterizing metal-binding sites using Support Vector Machines. In: Proceedings of the International Conference on Bioinformatics and Applications. pp. 307–318. Fort Lauderdale, FL (2004)
5. Chakrabarti, R., Klibanov, A.M., Friesner, R.A.: Computational prediction of native protein ligand-binding and enzyme active site sequences. Proceedings of the National Academy of Sciences of the United States of America 102(29), 10153–10158 (2005)
6. Davis, J., Burnside, E.S., de Castro Dutra, I., Page, D., Ramakrishnan, R., Santos Costa, V., Shavlik, J.: View Learning for Statistical Relational Learning: With an application to mammography. In: Proceedings of the 19th International Joint Conference on Artificial Intelligence. pp. 677–683. Edinburgh, Scotland (2005)
7. Davis, J., Burnside, E.S., de Castro Dutra, I., Page, D., Santos Costa, V.: An integrated approach to learning Bayesian Networks of rules. In: Proceedings of the 16th European Conference on Machine Learning. pp. 84–95. Porto, Portugal (2005)
8. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. Wiley-Interscience, New York, second edn. (2001)
9. Finn, P., Muggleton, S., Page, D., Srinivasan, A.: Pharmacophore discovery using the Inductive Logic Programming system PROGOL. Machine Learning 30(2-3), 241–270 (1998)
10. Fox, M.A., Whitesell, J.K.: Organic Chemistry. Jones & Bartlett Publishers, Boston, MA, 3rd edn. (2004)
11. García-Hernández, E., Zubillaga, R.A., Chavelas-Adame, E.A., Vázquez-Contreras, E., Rojo-Domínguez, A., Costas, M.: Structural energetics of protein-carbohydrate interactions: Insights derived from the study of lysozyme binding to its natural saccharide inhibitors. Protein Science 12(1), 135–142 (2003)
12. Gilis, D., Massar, S., Cerf, N.J., Rooman, M.: Optimality of the genetic code with respect to protein stability and amino-acid frequencies. Genome Biology 2(11), research0049 (2001)
13. Gold, N.D., Jackson, R.M.: Fold independent structural comparisons of protein-ligand binding sites for exploring functional relationships. Journal of Molecular Biology 355(5), 1112–1124 (2006)
14. Guex, N., Peitsch, M.C.: SWISS-MODEL and the Swiss-PdbViewer: An environment for comparative protein modeling. Electrophoresis 18(15), 2714–2723 (1997)

15. Kadirvelraj, R., Foley, B.L., Dyekjær, J.D., Woods, R.J.: Involvement of water in carbohydrate-protein binding: Concanavalin A revisited. Journal of the American Chemical Society 130(50), 16933–16942 (2008)
16. Khuri, S., Bakker, F.T., Dunwell, J.M.: Phylogeny, function and evolution of the cupins, a structurally conserved, functionally diverse superfamily of proteins. Molecular Biology and Evolution 18(4), 593–605 (2001)
17. Malik, A., Ahmad, S.: Sequence and structural features of carbohydrate binding in proteins and assessment of predictability using a Neural Network. BMC Structural Biology 7, 1 (2007)
18. Mitchell, T.M.: Machine Learning. McGraw-Hill International Editions, Singapore (1997)
19. Nassif, H., Al-Ali, H., Khuri, S., Keirouz, W.: Prediction of protein-glucose binding sites using Support Vector Machines. Proteins: Structure, Function, and Bioinformatics 77(1), 121–132 (2009)
20. Quiocho, F.A., Vyas, N.K.: Atomic interactions between proteins/enzymes and carbohydrates. In: Hecht, S.M. (ed.) Bioorganic Chemistry: Carbohydrates, chap. 11, pp. 441–457. Oxford University Press, New York (1999)
21. Rao, V.S.R., Lam, K., Qasba, P.K.: Architecture of the sugar binding sites in carbohydrate binding proteins—a computer modeling study. International Journal of Biological Macromolecules 23(4), 295–307 (1998)
22. Santos Costa, V.: The life of a logic programming system. In: de la Banda, M.G., Pontelli, E. (eds.) Proceedings of the 24th International Conference on Logic Programming. pp. 1–6. Udine, Italy (2008)
23. Shionyu-Mitsuyama, C., Shirai, T., Ishida, H., Yamane, T.: An empirical approach for structure-based prediction of carbohydrate-binding sites on proteins. Protein Engineering 16(7), 467–478 (2003)
24. Solomon, E., Berg, L., Martin, D.W.: Biology. Brooks Cole, Belmont, CA, 8th edn. (2007)
25. Srinivasan, A.: The Aleph Manual, 4th edn. (2007), `http://www.comlab.ox.ac.uk/activities/machinelearning/Aleph/aleph.html`
26. Srinivasan, A., King, R.D., Muggleton, S.H., Sternberg, M.J.E.: Carcinogenesis predictions using ILP. In: Proceedings of the 7th International Workshop on Inductive Logic Programming. pp. 273–287. Prague, Czech Republic (1997)
27. Sujatha, M.S., Balaji, P.V.: Identification of common structural features of binding sites in galactose-specific proteins. Proteins: Structure, Function, and Bioinformatics 55(1), 44–65 (2004)
28. Sujatha, M.S., Sasidhar, Y.U., Balaji, P.V.: Energetics of galactose- and glucose-aromatic amino acid interactions: Implications for binding in galactose-specific proteins. Protein Science 13(9), 2502–2514 (2004)
29. Taroni, C., Jones, S., Thornton, J.M.: Analysis and prediction of carbohydrate binding sites. Protein Engineering 13(2), 89–98 (2000)
30. Wang, G., Dunbrack, R.L.: PISCES: A Protein Sequence Culling Server. Bioinformatics 19(12), 1589–1591 (2003)
31. Zhang, Y., Swaminathan, G.J., Deshpande, A., Boix, E., Natesh, R., Xie, Z., Acharya, K.R., Brew, K.: Roles of individual enzyme-substrate interactions by alpha-1,3-galactosyltransferase in catalysis and specificity. Biochemistry 42(46), 13512–13521 (2003)