2013

A Feature-Based Approach to Estimate Protein-Protein Electrostatic Binding Energetics

Amelia Kreienkamp
akreienk@wellesley.edu

A FEATURE-BASED APPROACH TO ESTIMATE PROTEIN-PROTEIN ELECTROSTATIC BINDING ENERGETICS

Amelia Kreienkamp

Advisor: Mala Radhakrishnan

Submitted in partial fulfillment of the prerequisite for honors in chemistry

April 2013

ACKNOWLEDGEMENTS

I would like to thank my advisor, Professor Mala Radhakrishnan, for allowing me to work in her lab. Although it has been incredibly challenging, I have learned so much, and I am truly grateful for the opportunity she has given me.

I would also like to thank our collaborator, Finale Doshi-Velez, for the support she has offered during the course of this project. She has been an invaluable resource on regression techniques, calculating features, and countless other aspects of the project, and I am grateful for her expertise and guidance.

I am also grateful to my thesis committee, Didem Vardar-Ulu (Chemistry), Christopher Arumainayagam (Chemistry) and Sharon Elkins (Religion). I really appreciate each of you taking the time to serve on my committee. I would also like to thank the chemistry department of Wellesley at large, for making my Wellesley experience so worthwhile.

I would also like to thank my friends at Wellesley for their help and support. Special thanks to YingYi Zhang '13, who wrote numerous pieces of code that have been instrumental to this project. Another special thanks to Helena Qi '14, who helped in the initial stages of this project and who read various parts of my thesis. I would also like to thank Lucy Liu '13 and Connie Chen '15, for their support and companionship during the summer, and especially Lucy for all her support as a fellow "thesis-ing" senior. And thank you to all Radhakrishnan lab members past and present.

I am especially grateful for the funding sources that have allowed me to pursue research at Wellesley, including the National Science Foundation Research Experience for Undergraduates, the Susan Todd Horton Class of 1910 Trust, and Wellesley College itself.

In closing, I would like to thank my parents, my brother, and all my family for their love and support.

ABSTRACT

Quickly and accurately calculating the electrostatic free energy change that occurs when proteins bind is essential to the analysis of protein-protein interactions. This free energy change, quantified as ΔG_{elec}, expresses how energetically favorable it is for two proteins to come together. Calculating ΔG_{elec} for a protein complex usually involves finding a numerical solution to the computationally intensive Poisson equation. More approximate methods can save time and computational cost. Our approach involves training a computer via regression-based machine learning techniques to predict ΔG_{elec} using simple, structural features of the protein complex itself. Work on both shape-simplified model systems and protein shapes suggests that this approach may be successful in efficiently and accurately predicting ΔG_{elec} for protein complexes.

TABLE OF CONTENTS

INTRODUCTION

I. Importance of electrostatic interactions

From metabolism to the actions of the immune system, many cellular processes critical to life depend on protein-protein interactions. Much experimental work has focused on the energetics of binding, which are governed by electrostatic interactions, hydrophobic effects, and van der Waals forces. Especially critical to molecular recognition and binding are electrostatic forces, the interactions between polar or charged groups. Electrostatic interactions are particularly suited to computational studies, such as the one presented in this work.

This study focuses on calculating ΔG_{elec}, which quantifies how energetically favorable it is for two proteins to bind based on the electrostatic component of binding alone. A quantitative, computationally inexpensive model for binding energetics would aid our understanding of molecular interaction. The work presented here uses regression-based machine learning techniques to train a model to predict electrostatic binding free energies based on structural features of a protein complex.

A. Protein binding

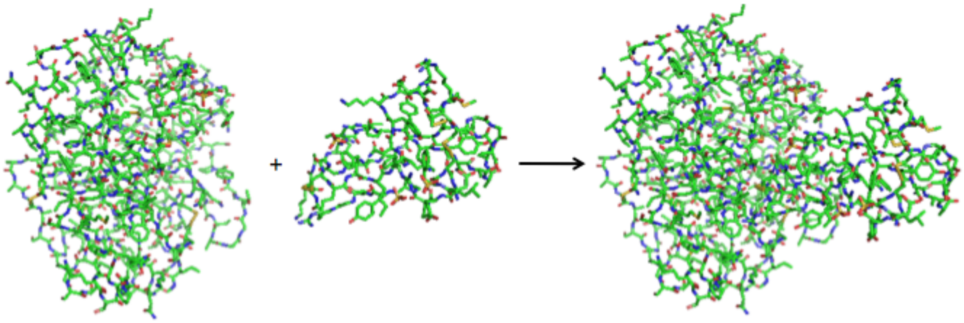

Figure 1.1. Schematic of protein binding: a *ligand* (typically the smaller partner) binding to a *receptor* to form a *complex*.

Made up of only 20 different types of amino acids, proteins have remarkable similarities but also vast differences in structure and function. Proteins are polymers of amino acids linked by peptide bonds. The peptide bond linkages make up the backbone of the protein. Proteins can differ in their number of amino acids (called residues) and in their amino acid composition, as each amino acid contains a variable region called a side chain that confers vastly different properties to the structure. Despite such apparent simplicity in their composition, proteins can adopt complex secondary and tertiary structures that are very different from one protein to the next. Perhaps because of this, proteins differ vastly in their binding behavior – some bind promiscuously to multiple targets, while others are highly specific. This behavior can in part be explained by electrostatics, which impacts both protein specificity and affinity.

The complex role of electrostatics in protein binding has been extensively reviewed.[1, 2] Electrostatic interactions can impact protein stability. A study of protein-protein complexes with amino acid mutations at the binding interface found that disease-causing mutations tend to destabilize the protein complex electrostatically, while non-disease-causing mutations do not – suggesting that electrostatic interactions between interfacial residues can impact stability.[3] Electrostatic interactions are not always energetically favorable during protein binding,[4] but are critical to molecular specificity and affinity.

Much research has focused on short range electrostatic interactions between binding partners.[5] Polar and charged residues tend to be conserved at the binding interface, acting as "hot spot" residues.[6, 5] When such "hot spot" residues are removed, either computationally or experimentally (i.e., alanine scanning), their favorable contribution to the electrostatic binding free energy is lost.

However, the vast diversity in protein shape and character [7] makes it difficult to predict which residues are "hot spots".[8] While hydrogen bonding is important for specificity[9], it is sometimes difficult to tell using scientific intuition which structural contacts are important and contribute favorably to binding free energy.[10] Despite this difficulty, certain trends have been elucidated that give insight into important features in protein binding. Kumar and Nussinov studied the electrostatic contribution of ion pairs to the overall electrostatic free energy.[11] They found that favorability of the ion pairs depends on the local environment of the ion pair as well as the geometrical orientation of the two side chains in relationship to each other. Furthermore, protein binding occurs in an aqueous solvent, which complicates calculating the interaction between these polar and charged groups.[1] Kundrotas and Alexov showed that proteins with smaller interfaces tend to have a higher proportion of charged and polar residues than do those with larger interfaces, suggesting that electrostatics play a larger role in proteins with smaller interfaces.[12] They proposed that the extent to which desolvation, the stripping of favorable interactions with solvent from the binding interface, is compensated by hydrogen bonds and ion pairs is important to determining the impact of electrostatics on binding.

Electrostatic interactions also play a role at long distances. Residues on the periphery of the interface are sometime more important than those on the interface itself. While research has shown that net charge becomes important at long distances,[1] simply changing the net charge is not enough to enhance binding affinity.[13] Joughin and Tidor identified specific noncontacting residues that can increase binding affinity of the protein TEM1 β-lactamase to its inhibitor,[13] which suggests that electrostatic interactions between non-interfacial residues are important. Electrostatic interactions, both at long and short distances, are important to binding affinity and specificity.

B. Modeling protein binding

Modeling protein binding is challenging for many reasons. Although quantum mechanics, which treats electrons as delocalized particles, would be the most accurate way to model binding, such treatment for large macromolecules like proteins is computationally infeasible. Molecular mechanics, a model in which atoms are treated as localized particles with a charge at the center, is more practical. The accuracy of computer-calculated free energies depends on multiple factors, including the resolution of the X-ray crystal structure or other structural model used, the accuracy of appropriate parameters such as partial atomic charge, and the assignment of the appropriate protonation state on titratable residues.[1]

Additionally, the energetics of binding are difficult to calculate because the reaction occurs in the aqueous phase, often with a certain salt concentration at a particular pH.[1] In reality, proteins are surrounded by countless water molecules, which can take on multiple orientations and are thus difficult to model computationally. By modeling water implicitly as a constant dielectric, one can account for the polarization of water without modeling each molecule explicitly. An additional challenge is that proteins are dynamic, rarely locked into shape as the static images of X-ray crystal structures suggest. Furthermore, although in this work, proteins are modeled as rigid structures during binding, in reality the bound form of the protein within a complex may not resemble its unbound conformation. Such challenges make it difficult to model protein binding computationally, so it is important to understand the limitations of our model when interpreting our results.

In nature, proteins are surrounded by solvent, which contains highly polarizable water molecules and ions. The charges on proteins interact with the charges on water, which contains both a partially negative oxygen atom and two partially positive hydrogen atoms. Upon binding, the water molecules that were interacting with the proteins' binding interfaces must forfeit these interactions. However, the charges on one protein can now interact favorably with the charges on the other protein – a phenomenon called *interaction*. Protein binding, therefore, is a delicate balancing act between these two terms: the generally favorable interaction of binding partners, and the unfavorable act of pushing solvent molecules aside (termed *desolvation*).

II. Free energy calculations

A. The continuum electrostatic model

Methods to calculate free energies have been extensively researched and reviewed.[14] Continuum electrostatics is a well-established method to model free energies. *In vacuo*, charges can be described by classical electrostatics.[15] The Poisson equation, below, describes charges in a vacuum, and can be used to solve for electrostatic potential. In this equation, $\phi(r)$ is the electrostatic potential as a function of position, and $\rho(r)$ is the charge distribution as a function of position.

$$-\nabla \bullet \nabla \phi(r) = \frac{\rho(r)}{\varepsilon_0}$$

Furthermore, for charges *in vacuo*, the electric field is a superposition (a sum) of the individual electric fields produced by each of the charges.

$$E(r) = -\nabla\phi(r)$$

The electrostatic energy of a field of charges Q is the electrostatic work required to bring the charges together from an infinite distance apart. This is described by the following equation.

$$U = \int_0^Q \phi(q)dq$$

Because the electrostatic potential ϕ is proportional to charge, ϕ(q) can be written as Cq, or charge q multiplied by a constant C. The electrostatic energy can also be expressed using the equation below:

$$U = \int_0^Q (Cq)dq = \frac{1}{2}CQ^2$$

For two point charges *in vacuo*, the interaction energy of the two charges can be described by the Poisson equation, which reduces to Coulomb's law. This interaction U represents the work done to bring the charges together from infinite separation.

$$U(r_{12}) = \frac{q_1 q_2}{4\pi\varepsilon_0 r_{12}}$$

In this equation, r_{12} is the distance between the two point charges. This equation shows that as two charges get closer together (r_{12} decreases), the magnitude of their interaction energy U increases.

In most biologically relevant systems, however, charges do not exist *in vacuo*, but rather in solutions that contain solvent. In this case, a spatially varying dielectric constant D(r) is used to account for how water or other species screen (or dampen) the interactions between charges. A uniform dielectric constant cannot be used because of the sudden change in dielectric constant at the protein boundary. The Poisson equation can be amended as below to account for this spatially varying dielectric.

$$-\nabla\cdot\,[D(r)\nabla\phi(r)] = \frac{\rho(r)}{\varepsilon_0}$$

Furthermore, the solvent often contains not just water but mobile ions such as salt, which further screen electrostatic interactions. These ions carry charge, which interfere with the interactions between the protein charges themselves. Debye-Hückel theory extends the Poisson equation into the Poisson-Boltzmann equation, which implicitly accounts for the extra sources of point charges. The Boltzmann factor of $e^{-\beta\phi(r)q}{}_i$ increases or decreases the estimation of the local concentration (c_r) of ions compared to their bulk concentration (c_{bulk}).

$$c_i(r) = c_{i,bulk}e^{-\beta\phi(r)q_i}$$

In a system that contains N types of ions each with specified charge and bulk concentration, the nonlinear Poisson-Boltzmann equation can be used.

$$-\varepsilon_0\nabla\bullet\left[D(r)\nabla\phi(r)\right] = \rho^f(r) + \sum_{i=1}^{N} q_i c_{i,bulk}(r)e^{-\beta q_i\phi(r)}$$

In this equation, the electric fields generated by the system of charges are not equal to the sum of the fields generated by each individual charge, because each charge experiences a different degree of solvent screening. This means that the electrostatic potential is not proportional to charge. Because the potential does not vary linearly with the source charges $\rho^f(r)$, it is very difficult to find a numerical solution to this equation.

However, when the magnitude of the charges is small, we can assume that the electrostatic potential is small. This approximation can convert the nonlinear equation into a linear one: [15]

$$-\varepsilon_0\nabla\bullet\left[D(r)\nabla\phi(r)\right] = \rho^f(r) - \varepsilon_0 D(r)\kappa^2(r)\phi(r)$$

In this instance:

$$\kappa^2 = \frac{\beta}{D\varepsilon_0}\sum_{1}^{N} c_{i,bulk}q_i^2$$

The linearized Poisson-Boltzmann equation (LPBE) is more convenient to use. It assumes linear response: the electrostatic field generated by the system of charges is equal to the sum of the fields generated by the individual charges, meaning that the electrostatic potential is proportional to charge and that the principle of superposition again applies. This means that the total electrostatic energy can be obtained by summing the product of charge and potential for each charge, and dividing by ½.

$$G = \sum_i \frac{1}{2} q_i \phi_i$$

In this equation, the factor of ½ stems from two sources: 1). It avoids double counting the electrostatic effects of two charges feeling the impact of the other, and 2). It reflects the cost of generating a reaction field, which is ½ the interaction energy of the charge with the solvent. Furthermore, this reaction energy is a free energy, because it includes the entropic cost associated with re-orienting the solvent molecules. [16] This term G is called the Gibbs free energy.

We are interested in calculating the free energy change that occurs when proteins bind. Thus, we aim to find the change in Gibbs free energy that occurs when proteins go from their unbound to their bound states. We are interested in systems with multiple charges on two partners, and quantifying how the charges on one partner (the “ligand”) interact with the charges on the other partner (the “receptor”).

For an individual charge, the solvation energy is quadratically related to the magnitude of the charge itself, assuming linear response – that is, that the reaction field generated by the solvent in response to each charge is proportional to the magnitude of the charge itself.[16] For interacting charges, the interaction energy is proportional to the product of both charges *i* and *j*. The energy can thus be rewritten as a sum of charge squared and charge pairs:[16]

$$G = \sum_i \frac{1}{2} q_i c_{ii} q_i + \sum_i \sum_j \frac{1}{2} q_i c_{ij} q_j$$

This equation can also be rewritten in matrix form. A vector of charge *q*, containing all the charges on the protein, can be multiplied by a potential matrix M consisting of the proportionality constants ½ c_{ij}.[16]

$$G = q^T M q$$

As proteins bind, the difference in their free energies is described by the difference in the Gibbs free energy of the unbound and bound states. The lower in energy a state is, the more stable it is. If the bound state is lower in energy (more negative) than the unbound state, ΔG_{elec} will be negative, and the process is energetically favorable. This can be written mathematically as:

$$\Delta G = G_{bound} - G_{unbound} = q^T (M_{bound} - M_{unbound}) q = q^T (M_{diff}) q$$

In this case, *q* represents all the charges on the ligand and the receptor. This potential matrix can be split up into matrices of potential for the ligand, receptor, and complex of ligand-receptor. We partition M_{diff} into separate matrices so that we can quantify different phenomena in protein binding: *desolvation* and *interaction*. This is summarized in the following equations:

$$\Delta G = q^T (M_{diff}) q = [q_L^T q_R^T] \begin{bmatrix} L & \frac{1}{2} C \\ \frac{1}{2} C^T & R \end{bmatrix} \begin{bmatrix} q_L \\ q_R \end{bmatrix} = q_L^T L q_L + q_R^T R q_R + q_L^T C q_R$$

Above, vectors of charge for the ligand (q_L) and receptor (q_R) are multiplied by matrices of unit potential for the ligand, receptor, and complex. Each term quantifies the particular phenomenon involved in binding:

ΔG_{elec} = Ligand Desolvation Penalty + Receptor Desolvation Penalty + Complex Interaction

B. Numerical solutions to the LPBE

Because proteins have an irregular dielectric boundary, it is impossible to solve the linearized Poisson-Boltzmann equation analytically.[15] Numerical methods, which discretize the problem into smaller systems that can be solved by matrices, must be used to solve the equation. Two commonly used numerical methods include the boundary element method and the finite difference method. In the boundary element method, the dielectric boundary of a protein is discretized into flat panels, and the surface charge is determined for each panel. In contrast, in the finite difference method, a cubic lattice is laid over a protein, discretizing space. At each grid point, the charge is defined. The electrostatic potential is solved for at each of the grid points:

$$\phi_i = \frac{\sum_1^6 \varepsilon_0 D_j \phi_j + q_i / h}{\sum_1^6 \varepsilon_0 D_j + \varepsilon_0 D_i \kappa_i^2 h^2}$$

In this equation, ϕ_i is the potential for each grid point *i*, while *j* represents the indices of the six neighboring grid points, and *h* is the length of one grid line.[15]

As it is computationally expensive to solve the Poisson-Boltzmann equation, many different approaches have been taken to speed up the solution process. These different approaches range from physics-based approximations to more empirical methods based on physicochemical properties.

C. Physics-based approximations to the LPBE

The broad applicability of free energy calculations necessitates improving ways to calculate such free energies. In cases such as molecular dynamics and Monte Carlo simulations, rapidly calculating ΔG is essential.[17] One physics-based approximation is the

Generalized Born (GB) approximation. This model approximates a protein as a set of spheres with an internal dielectric constant ε_{in} surrounded by solvent with dielectric constant ε_{out}, which is advantageous because the Poisson equation can be solved analytically for spheres.[16] This method aims to calculate the individual polarization energy G_i^{self} of each charge in a system in the absence of all other charges, in order to find the overall polarization energy G^{pol} of the system.[18]

$$G^{pol} = -\frac{1}{2}\left(\frac{1}{\varepsilon_{in}} - \frac{1}{\varepsilon_{out}}\right)\sum_{i,j}\frac{q_i q_j}{\left(r_{i,j}^2 + \alpha_i\alpha_j \exp\left(-\frac{r_{i,j}^2}{c\alpha_i\alpha_j}\right)\right)^{1/2}}$$

The constant c is an empirical coefficient, originally defined as 4,[19] while q_i is each atom's atomic charge and $r_{i.j}$ is the distance between each atom *i* and *j*. G^{pol} is written above as a sum over the Born radii α_i, which can themselves be written in terms of their individual self-polarization energies G_i^{self}.[18]

$$\alpha_i = -\frac{1}{2}\left(\frac{1}{\varepsilon_{in}} - \frac{1}{\varepsilon_{out}}\right)\frac{q_i^2}{G_i^{self}}$$

The Born radius α_i is a term that accounts for each atom's degree of burial within the solvent.

Because calculating the Born radii for each atom still requires finding G_i^{self}, it is still computationally expensive. A further approximation simplifies G_i^{self} into a volume integral.[18]

$$G_i^{self} = -\frac{1}{8\pi}\left(\frac{1}{\varepsilon_{in}} - \frac{1}{\varepsilon_{out}}\right)\int_V \overline{D}^2 dV$$

In this equation, $\overline{D} = q\overline{r}/|r|^3$ is the electric displacement vector. The integration is done over the volume outside the dielectric cavity.[18] This changes the equation for Born radii into the following Coulomb approximation:

$$\frac{1}{\alpha_i} = \frac{1}{4\pi}\int_V \frac{1}{r^4}\,dV$$

Instead of integrating over the volume outside the dielectric cavity, this equation can be re-formulated to integrate over the volume of the atom itself (W), excluding the volume of the electrostatic radius (the distance from the atom center to the edge of the electron cloud) of the atom.

$$\frac{1}{\alpha_i} = \frac{1}{R_i^{es}} - \frac{1}{4\pi}\int_W \frac{1}{r^4}\,dV$$

The GB approximation can be further simplified into the **surface-Generalized Born** approximation, by turning the volume integral into a surface integral.[18, 20] This can be used to calculate the individual polarization energies of each atom.[18]

These physics-based methods are effective in approximately calculating free energies, but more rigorous solutions take more time. The above formulation of the Surface GB method predicts solvation energies with a root mean square error of 0.13 kcal/mol relative to a well-established polarizable continuum method.[18] Many implementations of the GB method overestimate Born radii.[21] These methods are disadvantageous because of the tradeoff between accuracy and computational cost.

D. Empirical methods based on physicochemical properties

One alternative to physics-based modeling is the identification of “features” of binding that impact free energy calculation. Although much research has focused on feature identification in drug-protein systems, work remains to be done on protein-protein systems. Feature identification and free energy calculation are often done in the field of rational drug design, which often involves virtual screening of large-scale databases to select potential drugs (lead compounds) for a molecular target. These lead compounds are often identified by pharmacophore modeling and/or molecular docking.

Pharmacophore modeling

Pharmacophore modeling has become an important tool in rational drug design and has been extensively reviewed.[22] A pharmacophore is the set of features that is necessary for a molecule to bind to its target. [23] Such features often involve hydrogen-bond acceptors or donors, hydrophobes, negatively and positively charged ionizable groups, and aromatic ring structures.[24] Pharmacophore modeling involves both ligand-based and structure based methods.

Ligand-based methods involve characterizing common chemical features of a set of ligands binding to one molecular target.[22] A computational technique that allows the ligand to occupy a range of conformational spaces is used, followed by methods that characterize the chemical features of all the ligands in all the various conformations. This allows for the production of a set of key features that are apparently vital to bind to that particular target.[22] Ligand-based pharmacophore modeling methods are commercially available, often involving both pharmacophore identification as well as quantitative-structure-activity-relationship (QSAR) model development and 3D database screening.[24] QSAR links experimental activity with each feature, enabling prediction of active compounds.[24]

Structure-based pharmacophore modeling involves analysis of the interactions of the ligand with the active site of a protein complex. However, because this method depends on the 3D structure of the complex, it is unusable in cases in which no ligands are known to bind to the target.[22] In structure-based pharmacophore modeling, following structure preparation of the protein and identification of the binding site, pharmacophore features are defined and selected.[25] The most commonly used features in structure-based methods are hydrogen bond acceptors and donors, ionizable groups, lipophilic regions and aromatic rings. [25] A variety of methods exist to select proper combinations of features, including energy-based predictions that predict the interaction of the ligand with the protein. Probe docking selects features by docking the ligand with the protein, and selecting the features that lead to high interaction energies.[25]

Molecular docking

Molecular docking involves predicting which orientation a molecule prefers when binding to a target. Scoring functions are used to predict the strength of the binding interaction.[26] These scoring functions tend to fall into three categories: force field calculations, empirical methods, and knowledge-based statistical potentials.[27] Force field scoring functions calculate the potential energy of a complex based on the sum of non-bonded and bonded energy terms, which are physics-based and computationally intensive.[17] Empirical scoring functions calculate binding free energies by summing physicochemically relevant terms, such as hydrogen bonding, hydrophobic interactions, van der Waals interactions and conformational entropy.[27] Knowledge-based statistical potentials compare the features of a protein complex to those in a database, such as the distribution of atom-atom distances between the ligand and the receptor, and from this comparison predict an energy.[27] Because force field scoring methods are very similar to the physics-based methods described above, for brevity I will discuss only knowledge-based functions.

Knowledge-based scoring functions rely on a set of data. Wallqvist and Covell described an approach that classified the surfaces of buried ligand atoms in a set of enzyme-inhibitor complexes and predicted ΔG_{bind}.[28] They used computational geometry to examine the shape of each protein's surface and derived a scoring system for each pair of geometrically-overlapping segments between partners, and from this calculated ΔG_{bind} for the complex. The minimum energy was used to determine the appropriate configuration of the complex. Knowledge-based scoring functions have thus been used to calculate free energies in the past.

Pharmacophore modeling employs the use of features to make predictions about drug-target interaction, while molecular docking uses scoring functions to calculate free energies in order to determine the correct orientation of the drug in the binding site. Combined, both molecular docking and pharmacophore modeling are useful in identifying potential drugs during drug discovery. We aim to use a similar feature-based approach to estimate protein-protein binding free energies.

III. Our approach

Past research has focused on one of two extremes: mathematical, physics-based approximations to solving the Poisson equation, or empirical methods to calculate binding free energies. This work aims to combine the machine-learning aspects of empirical methods with the physics-based insights of more mathematical models. We define certain "features" of binding based on human intuition of the underlying physical models. We then use regression to relate these features to binding free energies.

Our method takes advantage of the fact that ΔG_{elec} for the association of two proteins in solution can be decomposed into three *terms*: the ligand desolvation penalty, the receptor desolvation penalty, and the complex interaction. As a reminder, the desolvation penalties are the costs associated with stripping the protein of its favorable interactions with water from the binding interface. Interaction is the hopefully energetically favorable term that arises from the charges on the ligand interacting favorably with the receptor.

$$\Delta G_{elec} = q_L L q_L + q_R R q_R + q_L C q_R$$

From this equation, each term can be written out more fully as a vector of charge multiplied by a matrix of unit potential multiplied again by a vector of charge. For example, the ligand desolvation penalty can be rewritten:

$$LDP = \begin{bmatrix} q_1 & q_2 \end{bmatrix} \begin{bmatrix} L_{11} & L_{12} \\ L_{21} & L_{22} \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}$$

The matrix elements $L_{11} \ldots L_{22}$ are unit potentials. The *diagonal* matrix elements (L_{11} and L_{22}) correspond to charge 1 and charge 2 on the ligand, respectively. The *off-diagonal* matrix elements L_{12} and L_{21} represent the interaction of charge 1 with charge 2, and thus by definition should be equal to each other.

The above matrix multiplication to calculate the term can be rewritten as a sum of individual charge potentials and the pairwise atom potentials. For the ligand desolvation penalty, in the case of two charges:

$$LDP = q_1^{\ 2} L_{11} + q_1 q_2 L_{21} + q_1 q_2 L_{12} + q_2^{\ 2} L_{22}$$

The potentials $L_{11} \ldots L_{22}$ can be calculated using a numerical solution to the Poisson Boltzmann equation, such as the finite difference method. In a similar fashion, we can write the entire ΔG_{elec} equation as a weighted sum of individual charge features (ie, q_1^2) and pairwise charge features (ie, $q_1 q_2$).

$$\Delta G = \sum_i c_{ii} q_i^{\ 2} + \sum_{i,j} c_{i,j} q_i q_j$$

In the above equation, if charges *i* and *j* are on the same partner, c_{ii} and c_{ij} are the diagonal and off-diagonal matrix elements of the L and R matrices. If *i* and *j* are on separate partners, c_{ij} are the elements of the C matrix. This would multiply out to produce the same result as using the equation:

$$\Delta G_{elec} = q_L L q_L + q_R R q_R + q_L C q_R$$

Our work aims to bypass solving the Poisson Boltzmann equation by estimating either these unit potentials (matrix elements) or terms (LDP, RDP, CI) from the features that we define. In other words, we aim to estimate c_{ii} and c_{ij} using features. Using these mathematical relationships, we can perform regression either on the matrix elements or the terms, writing each as a linear combination of the features. Thus c_{ii} and c_{ij} are coefficients that are calculated by performing regression on the features:

$$c_{ii} = \sum_k \alpha_k x_k^{ii}$$

$$c_{ij} = \Sigma_k \beta_k x_k^{ij}$$

In these equations, x is the feature(s) of interest, and α and β are the coefficients assigned during regression.

Picking physically relevant features is obviously crucial to the success of this approach. Our aim for this project is to be able to predict the three components of ΔG_{elec} to a high degree of accuracy. In picking features, we aim to select terms that intuitively seem important to desolvation and interaction. For desolvation, there are both single atom and pairwise features, while for interaction there are only pairwise features. An example of a single atom feature important to desolvation is the distance of the charge to the interface. An analogous pairwise feature is the average of two charges' distance to the interface.

After defining the features, regression will be used to assign a weight (*coefficient*) to each feature, which will estimate how "important" each feature is to binding. Both linear and nonlinear regression methods may prove useful. However, up to this point, linear regression, in which a change in x is directly proportional to a change in y, has proven to be sufficient for our model, so this work presents only simple linear regression.

A. Linear regression

In linear regression, an input vector X_j is used to predict an output Y. In our case, the input vector contains the features. The linear regression model is as follows.[29]

$$f(X) = \beta_0 + \sum_{j=1}^{p} X_j \beta_j$$

In the above equation, β_j are the unknown parameters (coefficients) that regression assigns to the feature vector X_j. In our regression, we generally set the error term or noise β_0 equal to 0, so that we can use just β_j to predict Y. (However, in some instances β_0 was included so that we could standardize the features to have a mean of 0 and standard deviation of 1 - see the discussion section for explanation.)

In linear regression, the parameters β_j are predicted to minimize the sum of squared errors (RSS):[29]

$$RSS = \sum_{j=1}^{p} (y_i - f(x_i))^2 = \sum_{j=1}^{p} (y_i - \sum_{j=1} \beta_j x_{ij})^2$$

A similar type of regression, not currently included in this work but intended as part of future work, is LASSO regression, which minimizes a similar but different sum to predict the parameters β_j. Using LASSO is advantageous because inflicts a penalty if the error is too large, which allows it to limit the number of non-zero coefficients and thus the number of features. LASSO selects fewer features with similar error.

$$RSS = \sum_{j=1}^{p} (y_i - f(x_i))^2 + \lambda \sum_{j=1} \left| \beta_j \right|$$

This tries to minimize the coefficients as well as the error. When λ is zero, this simplifies to normal linear regression. However, if λ is too large, this forces the coefficients to be zero. The trick is to pick the correct value of λ that will select out important features (non-zero parameters) without forcing all the parameters to be zero. Using LASSO is a future step that will limit the number of coefficients, and thus the number of features used.

B. Nonlinear regression

Other methods exist that are nonlinear. Although we did not use these except in preliminary studies, these methods may potentially prove useful in future. Nonlinear methods include k nearest neighbors.

K nearest neighbors begins by classifying data based on its "nearest neighbor" features in the training data. If k = 1, the data is put into the same class as its nearest neighbor. In other words, it is predicted to have the same value as its nearest neighbor. If k = n, each of the values of the n nearest neighbors are averaged to predict the value of that data point.

Though such techniques are useful, this work utilizes a simple linear-regression-based approach; though the problem is linear in the charge products, this work investigates whether a model that is linear in features can provide adequate accuracy. Up to this point, simple linear regression has been sufficient.

C. Summary

This work aims to predict ΔG_{elec} using only structural features of a protein-protein complex. We trained a regression-based model on free energies calculated from a numerical solver to the Poisson equation. We tested this approach on theoretical model systems and on the irregular molecular shapes of protein-protein complexes. After randomly creating charge distributions for the systems, we calculated both matrix potentials and the overall free energies with a Finite Difference Method solver to the LPBE. We then calculated features for each system that intuitively seemed important to desolvation or interaction. We regressed on this data using the features as input data, and predicted free energies from these features. Our work is a novel approach to estimate protein-protein electrostatic binding free energies. We aim to combine our human insights into protein binding with machine learning techniques in order to calculate ΔG_{elec} quickly and efficiently. We hope that this work provides a stepping-stone for other work combining machine-learning techniques with protein-protein free energy calculations, and in general improves the body of knowledge of protein-protein interactions.

METHODS

The goal of this study was to develop a model to predict the electrostatic component of the free energy of binding (ΔG_{elec}) using structural features of a protein-protein complex. Using both a model system and protein-protein complexes, we first created random charge distributions on the systems of interest and performed continuum electrostatics calculations on these systems to obtain a "known" ΔG_{elec}. Then, we defined and calculated features of the complexes that we hypothesized would be important to ΔG_{elec}. Regression was then used to assign weights to each feature, which allowed us to predict ΔG_{elec} from the features.

A. Structure preparation

This work utilized both a model system and proteins to test our problem theoretically and on existing biomolecular shapes.

a). Model system

To create the atoms inside which charges would be placed on a model system, atoms of 1.2-Å radius were placed to form the hollow outline of a box-shaped ligand that bound within a cavity on a box-shaped receptor (refer to Figure 2.1). Then, the system was filled in with larger atoms of 2-Å radius inside both ligand and receptor. While these atoms carried no charge, charges were later added inside these atoms to create systems with overall charge.

b). Protein shapes

This method was also tested on more biologically relevant shapes of those protein-protein complexes. Each atom was assigned a partial atomic charge of zero.

Later, random locations within the low dielectric continuum of the complex were assigned charges between 1 and -1. 400 charges were placed on each complex. Natural charge distributions were not used because proteins consist of thousands of charges. Calculation of the matrix elements for each of these charges would prevent us from being able to study a variety of complex shapes.

The crystal structures of 9 protein complexes were selected from the Protein Data Bank (PDB) and prepared for use.[1] A list of their PDB codes is shown below.

PDB ID	Complex	Number of ligand atoms	Number of receptor atoms
3BTK[30]	Trypsin-BPTI	2092	580
1BRS[31]	Barnase-barstar	1141	912
2O60[32]	Calmodulin- neuronal nitric oxide synthase complex	1461	199
3D65[33]	Textilinin-1-trypsin	2091	577
2XTT[34]	Schistocerca gregaria protease inhibitor 1 - trypsin	331	2091
1CM1[35]	Calmodulin - calmodulin-dependent protein kinase II-alpha	1418	188
1TAW[36]	Trypsin - amyloid beta-protein precursor	2092	547
2BCX[37]	Calmodulin – ryanodine receptor peptide	1409	310
2F3Y[38]	Calmodulin – IQ domain of cardiac Ca(v)1.2 calcium channel	1447	249

Table 2.1. PDB codes of proteins used.

These particular protein complexes were selected because they had been previously prepared by another student in our laboratory, YingYi Zhang '13. These particular proteins were sufficient for our purposes, as we wanted structures with overall different geometries. Each structure was resolved to a resolution of 2.5 Å or better, to ensure maximum resolution. First, non-essential waters not critical to the structure's function,

[1] 9 structures were selected, but 10 points will be shown on the graphs. One complex, 1TAW, was accidentally trained on twice using different charge distributions.

other solvent molecules, and non-biological atoms that were merely a byproduct of crystallization were removed from the structures. The orientation of the carbonyl and the amine on both Asn and Gln residues, and the tautomerization state and orientation of each His side chain were assessed and optimized. Hydrogens and, if needed, missing density were built in for each residue using CHARMM.[39]

B. Random charge distributions

It was essential to create charge distributions that were off-center of the atoms themselves, so that we could train on the same structure multiple times. If only the atom centers were charged, the free energy would be linear because the unit potential would remain the same, and multiplication by charge would produce energy. Charges were randomly placed inside both the proteins and the model system, subject to certain constraints.

a). Model system

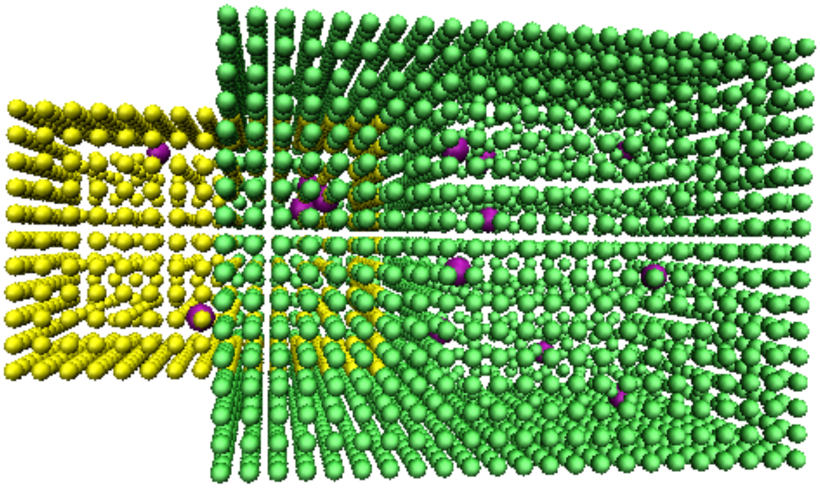

Figure 2.1. Schematic of the ligand (left, yellow) binding to a cavity inside the receptor (right, green) model system. Charges are shown in purple.

Charges were randomly placed inside the 2-Å radius atoms. Each charge was defined as an "atom" of zero radius, with a randomly generated partial atomic charge between 1 and -1. To avoid charges being too close together, each charge was constrained to be at least 1 Å away from the other charges. We purposefully used the smaller atoms to outline the box and only placed charges on the inner, larger atoms in order to constrain each charge to be at least 1 Å away from the dielectric boundary.

b). Protein shapes

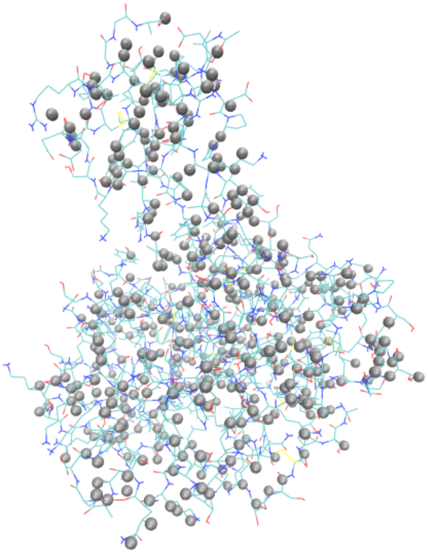

Figure 2.2. Representative random charge distribution for the protein complex trypsin-BPTI. Charges are shown as gray spheres.

A random charge distribution was created for each protein-protein complex. For each system, the original atoms were made neutral by assigning them a partial atomic charge of 0. Then, 400 charges of zero radius were randomly placed inside these atoms, off-center of the atom itself and constrained to be at least 1 Å apart from each other. Each charge was assigned a partial atomic charge between 1 and -1.

Constraining charges to be at least 1 Å away from the dielectric boundary required identifying the solvent-exposed atoms. All solvent exposed atoms were identified using a script written by YingYi Zhang '13. This script divided the local area of each atom up into 8 boxes and counted the number of atom contacts in each box. If the number of boxes with 2 or fewer contacts was 3 or greater, the atom was considered to be solvent exposed. To place a charge, a script randomly selected an atom and placed a charge inside it. If a charge was placed inside a solvent-exposed atom, an additional script checking to make sure that every atom was at least 1 Å away from the dielectric boundary was run (YingYi Zhang '13). This script calculated the distance from the charge to the center of the atom in which it was placed. If that distance was more than 1 Å, that charge was rejected and another charge was placed inside another atom. (In the future, solvent exposed atoms will be double-checked using CHARMM, which calculates the solvent exposed surface area for each atom in the bound and unbound states.[39] If the surface area is non-zero in the bound or unbound states (or both), the atom is solvent exposed. This method will be used as a check against our laboratory method. The double-checking did not occur for the placement of these charges.)

In some complexes, charges were biased to be located near the interface to encourage a larger desolvation term. To bias the charges, all atoms within 10 Å of the partner were considered "interface". Charges could either be placed by selecting an atom located either within one protein atom of the complex or within one atom of the interface, making it more than 50% likely that a charge would land on the interface.

In some complexes, an additional charge complementarity bias was included to encourage a favorable complex interaction term. For each charge placed on the ligand, the closest charge on the receptor was biased to be opposite in sign to that charge, 80%

on average. This did help in increasing the favorability of the interaction, making it more negative.

C. Continuum electrostatics calculations

We calculated the potential of the bound and unbound states of the complexes using a Finite Difference Method (FDM) solver to the Poisson equation, as the standard against which our approximate method would be compared.[40] Although the FDM is itself approximate, it is a rigorous numerical solution to the Poisson equation and is often the benchmark used to evaluate other, more approximate models.[41] Two types of calculations were carried out: 1). An overall binding free energy calculation, calculating the overall *terms* of LDP, RDP, CI and thus ΔG, and 2). An explicit potential *matrix elements* calculation, calculating the L, R, and C potential matrices for each complex in going from the unbound to bound states. Note that if the L, C or R potential matrix is multiplied by vectors of the appropriate charges, it equals the overall terms.

The Finite Difference Method solver laid down a cubic lattice consisting of 201 grids points along each dimension over the protein complex, for approximately 3-4 grids per Å for each protein. Then, it solved for the electrostatic potential at each point on the grid. The inner protein dielectric constant was set at 4, reflecting the relatively low polarizability of protein, while the outer dielectric was set at the comparatively high value of 80. Calculations were carried out using a probe radius of 1.4 Å (the size of a water molecule) that rolls over the protein surface to identify the area where water cannot penetrate, and zero ionic concentration. The grid was translated three times to recalculate the potential using a slightly different lattice. These values were averaged to calculate potential.

D. Feature definition

The heart of the project was the definition of "features" for each complex, which were each assigned a weighted coefficient during regression as a measure of its importance. From these coefficients and the features, we were able to predict ΔG_{elec}. In approaching this problem, we trained separately on desolvation and interaction, assuming that if we predicted both of these terms accurately, the sum of the predictions would accurately predict ΔG_{elec}.

We used our human intuition to devise features that would be important to desolvation or interaction. For desolvation, we identified both *single atom features*, or features that directly relate to only one charge, and *pairwise atom features*, features that capture the pairwise interaction between two charges. Because interaction is by definition pairwise, interaction features were only pairwise atom features.

Desolvation

For desolvation, we assumed that charges pay a bigger penalty when they are solvent exposed in the unbound state and close to the binding interface so that they are highly buried upon binding. So, our features for each charge aimed to capture the degree of solvent exposure that is sacrificed when the protein binds.

First, we defined one single atom feature as the distance of a charge to the binding interface. As before, the atoms within 10 Å of the other partner were considered interface. Then, to capture the fact that one charge may be close to multiple atoms on the binding interface, we defined a burial term as the reciprocal distance of a charge to all the atoms on the partner within a certain distance. We took this term at various distances, from 3-10 Å.

Then we defined one other single atom feature to approximate the local geometry around the charge. The local area around a charge was divided into eight boxes, and the number of atom contacts in each box was counted. The feature was taken as the number of empty, or solvent exposed, boxes.

We then defined pairwise atom features, which in some instances were analogs of the single atom features defined earlier. Others were independent of the single atom features, such as the distance between two charges. Analogs of the single atom features included the arithmetic and geometric means of two charges' distance to the interface. Multiple pairwise features were created as analogs of the burial terms. We considered that the degree of interaction between two charges would scale inversely as the distance between them. For the pairwise burial term, we multiplied two charges' burial terms and divided by the distance between them. In this way, two charges far apart but both close to the interface would have a large burial term, but two charges close to the interface but close together would have an even larger burial term. We then further expanded the pairwise atom features by using a variety of geometric and arithmetic means on the burial terms.

A comprehensive list of the features used is listed below.

Desolvation features	
Single atom features	**Pairwise atom features**
Distance to interface	Arithmetic mean of distance to interface: ½ (x+y)
	Geometric mean of distance to interface: √xy
	Distance between charges
Burial term: reciprocal distance to all atoms within x Å of interface* BT = $1/r_1+1/r_2 + \ldots + 1/r_n$	$\frac{\text{Burial term}_{\text{charge1}} \text{ x Burial term}_{\text{charge2}}}{\text{Distance between charges}}$
	$\frac{\text{Burial term}_{\text{charge1}} + \text{Burial term}_{\text{charge2}}}{\text{Distance between charges}}$
Number of solvent exposed boxes	Average number of solvent-exposed boxes

Table 2.2. Desolvation features.
*x = 3-10 Å

Complex interaction

A similar strategy was used to define features to predict complex interaction. These features were by definition pairwise.

$$CI = q_L^T C q_R$$

Because q_L is a vector of length *m*, and q_R is a vector of length *n*, C must be a matrix of length *m*x*n*.

Similar to how we approached desolvation, features aimed to capture human intuition about how two charges interact. First, because two charges interact strongly when they are close to each other, one feature was defined as the distance between a ligand charge and a receptor charge. Because the interaction increases as the distance decreases, an additional feature was added that was the reciprocal of this distance. Secondly, a high degree of solvent screening lessens the interaction between two charges. Thus, an additional feature was the average of two charges' closest distances to a solvent-exposed atom. A feature was added to capture the degree of solvent screening: pairwise "solvent exposure" terms. For each charge, the reciprocal distances to each solvent-exposed atom within a certain “cutoff” distance were added. A large term meant the charge was close to many solvent-exposed atoms. To obtain the pairwise feature, the solvent exposure terms of the two charges were added or multiplied, and then divided by the distance between them. An additional feature called the “inverse solvent exposure” term took the inverse of the solvent exposure term for each charge, so that the feature decreased as the level of solvent exposure increased (resembling how solvent exposure decreases interaction). The pairwise feature was obtained similarly. Additionally, a last set of features were included simply because they were readily accessible: the same burial features used in the desolvation section, in spite of the fact that complex interaction can only occur when the complex is interacting (because when the proteins are bound, the binding interface does not exist in our model, but is only part of the interior of the low-dielectric cavity). A comprehensive list of each feature used is shown below.

Interaction features	**Interaction features**	**Interaction features**
Pairwise “burial” features	**Pairwise “solvent exposure” features**	**Pairwise “inverse solvent exposure” features**
Arithmetic mean of distance to interface: ½ (x+y)	Arithmetic mean of distance to solvent-exposed atom: ½ (x+y)	Arithmetic mean of distance to solvent-exposed atom: ½ (x+y)
Geometric mean of distance to interface: √xy	Geometric mean of distance to solvent-exposed atom: √xy	Geometric mean of distance to solvent-exposed atom: √xy
Distance between charges	Distance between charges	Distance between charges
Inverse distance between charges	Inverse distance between charges	Inverse distance between charges
$\frac{\text{B term}_{charge1} \text{ x } \text{B term}_{charge2}}{\text{Distance between charges}}$	$\frac{\text{SE term}_{charge1} \text{ x } \text{SE term}_{charge2}}{\text{Distance between charges}}$	$\frac{\text{ISE term}_{charge1} \text{ x } \text{ISE term}_{charge2}}{\text{Distance between charges}}$
$\frac{\text{B term}_{charge1} + \text{B term}_{charge2}}{\text{Distance between charges}}$	$\frac{\text{SE term}_{charge1} + \text{SE term}_{charge2}}{\text{Distance between charges}}$	$\frac{\text{ISE term}_{charge1} + \text{ISE term}_{charge2}}{\text{Distance between charges}}$

Table 2.3. Complex interaction features.

*B term = Burial term: reciprocal distance to all atoms within 3-10 Å of interface

**SE term = Solvent exposure term: reciprocal distance to all atoms within 3-10 Å of protein edge

**ISE term = Inverse solvent exposure term: reciprocal of solvent exposure term

We stress that the features defined in this work are not necessarily the "best" features to use. Although we show that the features used can reasonably predict desolvation and interaction, it is entirely possible that other features could predict them better. Our work instead is a proof of principle that a feature-based approach can be used to predict ΔG_{elec}.

E. Regression

Regression techniques were then used to assign coefficients to each feature. As a reminder, the electrostatic component of binding is comprised of three terms (LDP, RDP, and CI), which can be written as:

$$\Delta G_{elec} = q_L^T L q_L + q_R^T R q_R + q_L^T C q_R$$

Because of this mathematical relationship, two methods can be used to carry out the regression: 1). Regression on the L, R, and C matrix elements themselves and 2). Regression on the overall LDP, RDP, and CI.

Simple linear regression was used in both approaches. As a reminder, linear regression minimizes the sum of squared errors to estimate the coefficients β_j.

$$RSS = \sum_{j=1}^{p} (y_i - \sum_{j=1} \beta_j x_{ij})^2$$

We first use "training data" with known $(x_1,y_1)\ldots(x_N,y_N)$ to estimate the parameters β_j. Then, from these parameters, we can multiply by known x values of "testing data" to obtain output y.

Predictions were first obtained by training on all the data, using those coefficients to predict the same data. To verify that the coefficients did have predictive value, cross-validation was performed by training on a subset of the data and predicting the rest of the data (the "testing data"). 10-fold cross-validation was conducted, training on 90% of the data and predicting the remaining 10%. This technique was performed a total of 10 times to make predictions for all the data. All results shown are performed using 10-fold cross-validation, except as specified.

Regression on the matrix elements

We can regress on the L, C, and R matrix elements to predict matrix elements. In this case, x_N is each feature we define, while y_N is the matrix potential. A table summarizes:

X_N (feature)	**Y_N (matrix potential)**	**Type of feature and matrix potential**
Single atom feature	Desolvation diagonal matrix element	Single atom
Pairwise atom feature	Desolvation off-diagonal matrix element	Pairwise atom
Pairwise atom feature	Complex interaction matrix element	Pairwise atom

Table 2.4. Regression on matrix potentials.

We can multiply these predicted matrix elements by charge to get energy, allowing us to predict LDP, RDP, and CI, and thus ΔG_{elec}.

Regression on the term

In the second case of regression on the term itself, we can write each term as a sum of charge squared and charge pairs. For example, the ligand desolvation penalty can be written as a sum of the single atom charges squared times the single atom features x_i and the charge pair products times the pairwise atom features x_{ij}, where α and β are coefficients for the charge squared and charge pairs respectively.

$$LDP = \sum_i \alpha q_i^2 x_i + \sum_{i,j} \beta q_i q_j x_{ij}$$

x_N is the sum of each feature multiplied by charge pairs, while y_N is the term itself.

An example makes this more concrete. If we have a ligand with 3 charges, for the ligand desolvation penalty we obtain a 3x3 unit potential matrix, and have a vector of charge q_L with size 3x1. We can obtain a vector of charge squared by making a 1x3 vector of q_L^T: $[q_1^2 \;\; q_2^2 \;\; q_3^2]$. We have a matrix of single atom features where each feature is a 3x1 vector that describes the feature for each atom, so the matrix of features is 3x*m* where *m* is the number of features. Multiplying q_L^T, the 1x3 matrix, by this 3x*m* feature matrix gives a 1x*m* matrix. We can do an analogous process for the pairwise features, where the charge vector in this case is instead $[q_1q_2 \;\; q_1q_3 \;\; q_2q_3]$.

Then, we can do the regression, using x_N as a concatenation of the charge squared-single atom feature matrix (1x*m*) described above and the charge pair-pairwise feature matrix (1x*n*) above, while y_N is the LDP itself. This regression assigns coefficients to each feature, producing *m*+*n* coefficients.

Regression protocol

We performed regression using a variety of input data, summarized in Table 5.

Training data	Testing data
Model system	Model system
Model system	Proteins
Proteins	Proteins
Proteins	Model system

Table 2.5. Summary of training data and testing data for regression.

Regression was conducted using R.[42]

Data analysis

To analyze each feature's importance, the data was standardized to account for the relative size of the input. This involved standardizing each feature so that the mean of the data was zero, and the standard deviation was one. The mean of each feature for a set of data was subtracted from each feature value, and each value was divided by the standard deviation of the feature. This accounted for the relative size of the input itself, and standardized the coefficients so that they could be compared. We examined the size of the coefficient itself, as large coefficients meant that the feature had a large impact on the prediction.

RESULTS

Regression on Poisson equation-derived matrix elements or terms was used to predict electrostatic binding free energies from features of the complex. Regression and data plotting was performed in R.[42] This section will proceed as follows: first, the model system results will be described, including the two approaches to regression (matrix elements and terms) for desolvation and interaction, and the coefficients obtained in both approaches. The next section will describe the results of the protein regression. The last section will compare the coefficients obtained when training on either the model system or the proteins, and also describe the data obtained when training on one system and testing on the other. A list of the features used in each case is listed at the end of Results.

As seen above, ΔG_{elec} can be written as follows:

$$\Delta G_{elec} = q_L^T L q_L + q_R^T R q_R + q_L^T C q_R$$

L, R, and C are unit potential matrices consisting of *matrix elements*. Each unit potential matrix, when multiplied by charge, gives the *term*: ligand desolvation penalty (LDP), receptor desolvation penalty (RDP), and complex interaction (CI). We can regress on either the matrix elements or the term, using features to predict ΔG_{elec}.

MODEL SYSTEM

Desolvation

The two terms of interest in desolvation are the ligand desolvation penalty and the receptor desolvation penalty.

$$LPD = q_L^T L q_L$$
$$RDP = q_R^T R q_R$$

To predict desolvation, two approaches to regression are possible: training on the matrix elements, and training on the term. Each will be discussed separately, and then coefficients from both approaches will be compared. In summary, training on the matrix elements produced predicted matrix elements that correlated well with the original values. Multiplying these matrix elements by charge to predict desolvation penalties improved the correlation. However, regression on the term produced the best correlations of all.

Matrix elements

The L and R matrices are by definition square and symmetric, consisting of diagonal matrix elements and off-diagonal matrix elements.

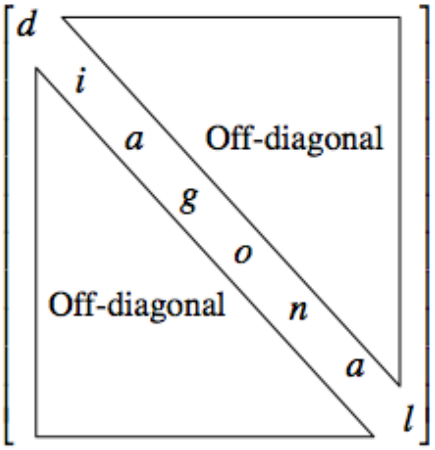

Figure 3.1. Schematic of a square symmetric matrix. The diagonal and off-diagonal matrix elements are indicated. Note that the two halves of the off-diagonal matrix elements are identical.

Diagonal matrix elements depend on the environment of single charges, so the features used in the regression are single atom features. Off-diagonal matrix elements depend on properties of interacting charges, so the features used are pairwise.

In the figures below, R indicates the correlation of the predicted values with the expected "exact" values. The "exact" values are the values obtained from the Finite Difference Method solution to the Poisson equation.[40] When R = 1, the data is perfectly correlated. When R = 0, there is no correlation. The root mean square error (RMSE) between the predicted and expected values is also indicated. This is the absolute difference between the two data sets, and thus measures the amount of error in the prediction. The following figures are the results from 10-fold cross-validation, except as specified.

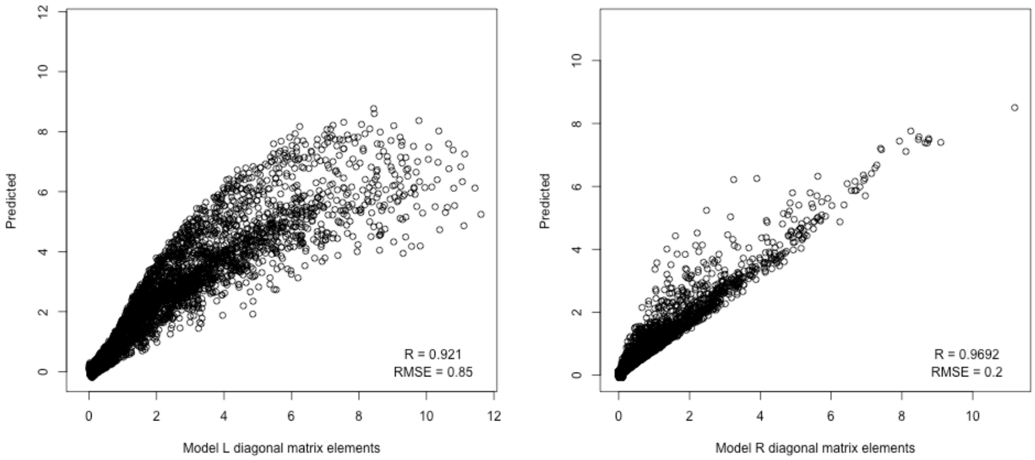

Figure 3.2. Predicted vs. exact L (left) and R (right) diagonal matrix elements. Features trained on: List 1 (see end of section). RMSE units = kcal/mol/e^2.

The predicted L diagonal matrix elements are well correlated. The predicted values appear to underestimate when the matrix elements are large, indicating that the model underestimates how much of a desolvation penalty charges pay when they are highly buried upon binding. However, predictions are close to the expected when the matrix elements are small.

Interestingly, the predicted R diagonal matrix elements (Figure 3.2 right) behave differently. Two observations must be noted: the R diagonal matrix elements are systematically overestimated, and more ligand charges pay a high desolvation penalty than receptor charges. These implications will be addressed in the discussion section.

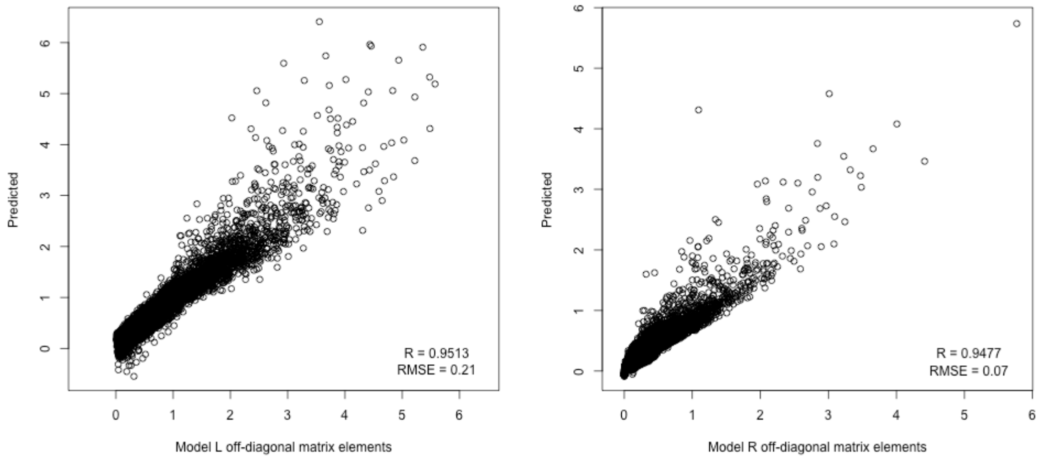

Figure 3.3. Predicted vs. exact L (left) and R (right) off-diagonal matrix elements. Features trained on: List 1. RMSE units = kcal/mol/e^2.

The predicted diagonal matrix elements of L and R behave differently, so it is unsurprising that the predicted off-diagonal matrix elements do as well, because the pairwise features are derived from the single atom features. The same trend for the diagonal matrix elements is seen for the off-diagonal matrix elements: error in the large matrix elements for the L matrix elements, and the same systematic overestimation for the R off-diagonal matrix elements. This is most likely because the pairwise features were almost entirely pairwise analogs of single atom features. In the case of the L off-diagonal matrix elements, almost all the large elements are overestimated, unlike the diagonal matrix elements, which were underestimated. However, the R off-diagonal matrix elements show nearly the same behavior as the diagonal: overestimating larger matrix elements.

These matrix elements can be used to construct an “approximate” L or R matrix, which can be multiplied against charge to give LDP or RDP.

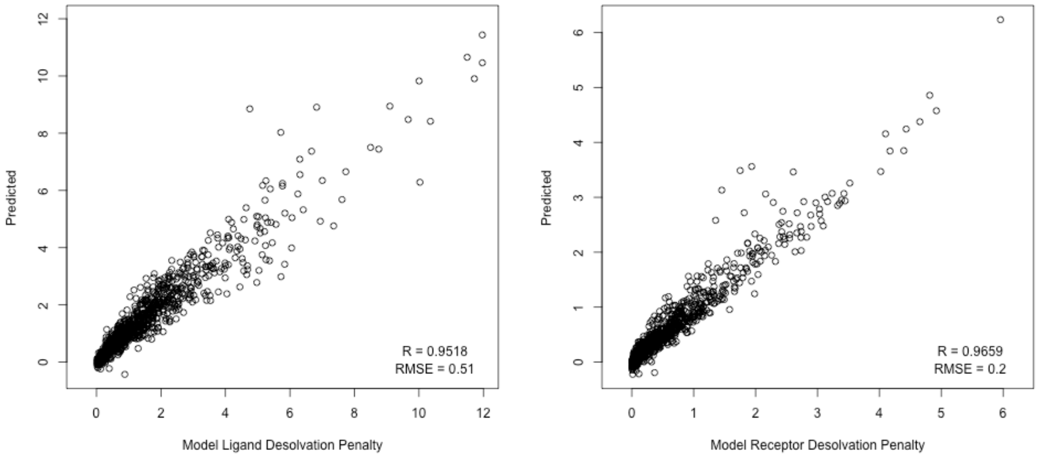

Figure 3.4. Predicted vs. exact desolvation penalties for ligand and receptor, after training on the matrix elements. RMSE units = kcal/mol.

Given the error in larger predicted matrix elements, it is surprising that multiplication of the approximate matrix by charge gives an energetic penalty so close to the expected. It is possible that the error in each of the individual matrix elements cancel out when multiplied by charge.

Term regression

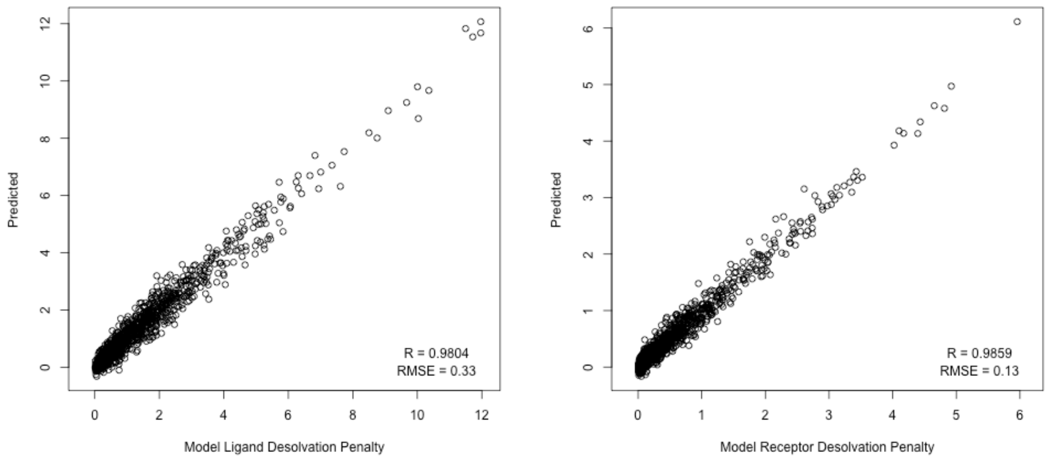

Figure 3.5. Regression on the desolvation penalties. Features trained on: List 1. RMSE units = kcal/mol.

Regression on the desolvation penalties produced high correlations for both ligand and receptor. This suggested that regression on the term was a more promising approach than regression on the matrix elements. Earlier, we suggested that the error in the individual predicted matrix elements cancels out when multiplied by charge. However, regression on the desolvation penalties produces a better fit, which implies the previous approach still suffers from some compounding of error, though not as much as one might expect. This requires further investigation.

Coefficients

The features were each scaled to make the mean 0 and the standard deviation 1. This standardizes the coefficients to allow for more direct comparisons between features. Magnitude indicates relative importance, and direction indicates in what way the feature affects the prediction.

	Model system L	Model system R
(Intercept)	1.9072	0.3640
DI	0.4338	0.1184
BT:3	0.3337	0.5282
BT:4	0.9507	-0.2252
BT:5	-0.0505	0.0587
BT:6	-0.0078	0.0917
BT:7	-0.3233	0.0963
BT:8	1.0081	0.1453
BT:9	0.9509	0.1541
BT:10	-0.2962	0.1353
Solvent boxes	0.0000	0.0000

Table 3.1. Diagonal feature coefficients for L (left) and R (right) after training on matrix elements. Highest positive coefficient (excluding the y intercept) is shown in blue, negative in red.

The single atom feature coefficients for ligand and receptor for desolvation show little correlation. The largest coefficient for the ligand was the burial term for all atoms within 8 angstroms. The largest coefficient for the receptor was the burial term for all atoms within 3 angstroms. The most negative coefficient for the ligand was the burial term at 7 angstroms, while the receptor was a burial term at 4 angstroms. This difference is not entirely surprising, given that the receptor has a different shape from the ligand. However, the difference in both magnitude and direction make interpretation of the relative importance of features difficult.

	Model system L	Model system R
(Intercept)	0.6329	0.1166
Dist between	-0.1282	-0.0263
Geom. DI	-0.1974	-0.0708
Arith. DI	0.1610	0.0304
PWBprod:3	3.1074	0.2636
PWBprod:4	-3.4784	-0.1688
PWBprod:5	1.1078	0.0166
PWBprod:6	0.4497	-0.0037
PWBprod:7	-0.4450	-0.0040
PWBprod:8	0.0889	-0.0072
PWBprod:9	-0.0308	0.0038
PWBprod:10	0.0081	-0.0001
PWBsum:3	-3.0078	-0.1147
PWBsum:4	3.9630	0.1037
PWBsum:5	-1.1558	0.0010
PWBsum:6	-0.7231	0.0170
PWBsum:7	0.3551	0.0219
PWBsum:8	0.3303	0.0081
PWBsum:9	0.0867	0.0261
PWBsum:10	-0.0929	0.0068

Table 3.2. Off-diagonal feature coefficients for ligand (left) and receptor (right) after training on matrix elements. Highest positive coefficient shown in blue, negative in red.

The pairwise atom coefficients, on the other hand, do appear more correlated. The directionality of the coefficients is largely the same, apart from the final six coefficients, which are all small in magnitude. The largest coefficient for the ligand was the pairwise burial sum using a cutoff of 4 angstroms, while the receptor was a pairwise burial product using a cutoff of 3 angstroms. The most negative coefficient for both the ligand and the receptor was the pairwise burial product, using a cutoff of 4 angstroms. Furthermore, note that the magnitudes of the receptor coefficients are smaller than those of the ligand. This is possibly because the receptor is larger, so the charges inside experience less desolvation. The implications of these coefficients will be discussed in the discussion section.

	Model system L	Model system R
(Intercept)	-0.0727	0.0099
DI	-0.0198	0.0022
BT:3	-0.0458	-0.0032
BT:4	0.1073	0.0167
BT:5	-0.0271	0.0174
BT:6	-0.0098	0.0371
BT:7	-0.0284	-0.0210
BT:8	0.0469	0.0831
BT:9	0.2572	0.1483
BT:10	-0.2310	0.3059
Solvent boxes	0.0000	0.0000
Dist between	0.0929	0.0041
Geom. DI	-1.4899	-0.1393
Arith. DI	1.6206	0.1459
PWBprod:3	0.0042	0.0150
PWBprod:4	-0.0082	-0.0113
PWBprod:5	0.0109	-0.0079
PWBprod:6	0.0029	-0.0139
PWBprod:7	-0.0373	0.0655
PWBprod:8	0.0595	-0.0725
PWBprod:9	-0.0239	-0.2126
PWBprod:10	-0.1115	6.0294
PWBsum:3	-0.1875	0.1063
PWBsum:4	0.4491	-0.0063
PWBsum:5	-0.4585	-0.2233
PWBsum:6	-0.0666	0.5359
PWBsum:7	0.5110	-0.2809
PWBsum:8	-0.2807	0.0506
PWBsum:9	0.4895	0.6044
PWBsum:10	0.1438	0.2516

Table 3.3. Feature coefficients for ligand (left) and receptor (right) after training on the term. Highest positive coefficient shown in blue, negative in red.

Interaction

Complex interaction occurs when ligand and receptor charges interact in the bound complex. For a system of *n* ligand charges and *m* receptor charges, the C matrix is an *n*x*m* matrix, involving only pairwise interactions. Thus, the features used are pairwise. Similar to desolvation, we can train on the C matrix elements or the complex interaction term.

Matrix elements

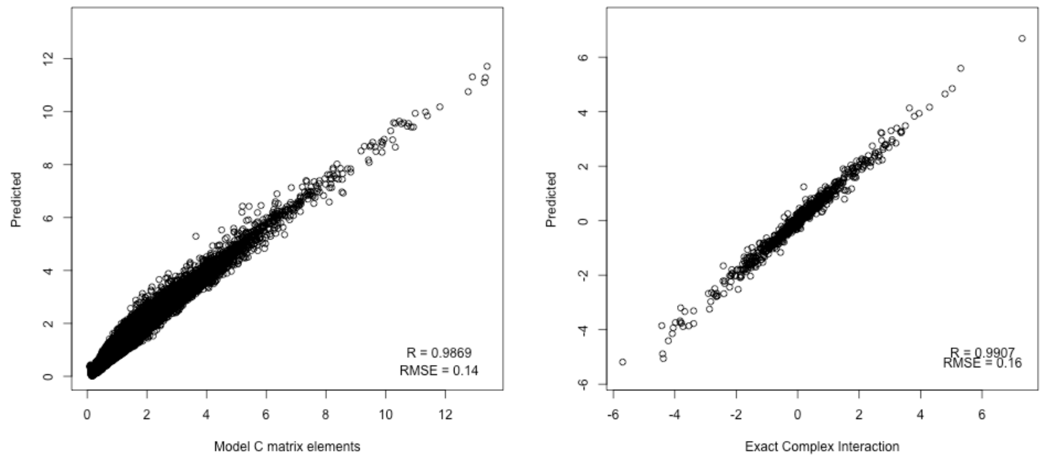

Figure 3.6. Predicted C matrix elements (left) and complex interaction (right) for model system. Features trained on: List 3 (burial terms). RMSE units (left) = kcal/mol/e^2. RMSE units (right) = kcal/mol.

Note that above, the features used were similar to the pairwise features for desolvation, which were largely comprised of burial terms. In this case, the interactions were between charges on the ligand and the receptor, rather than the same partner. But because there are no “binding partners” in the complex, interaction should theoretically not depend on the binding interface at all. We tried these features simply because they were readily accessible. However, we found that these features produced good predictions for complex interaction.

We trained on an additional set of features: solvent exposure terms analogous to burial terms (the reciprocal distance of the charge to all solvent-exposed atoms within a certain distance). The larger this term, the more solvent exposed a charge was, therefore the more its interaction should be screened and the more the matrix element should decrease.

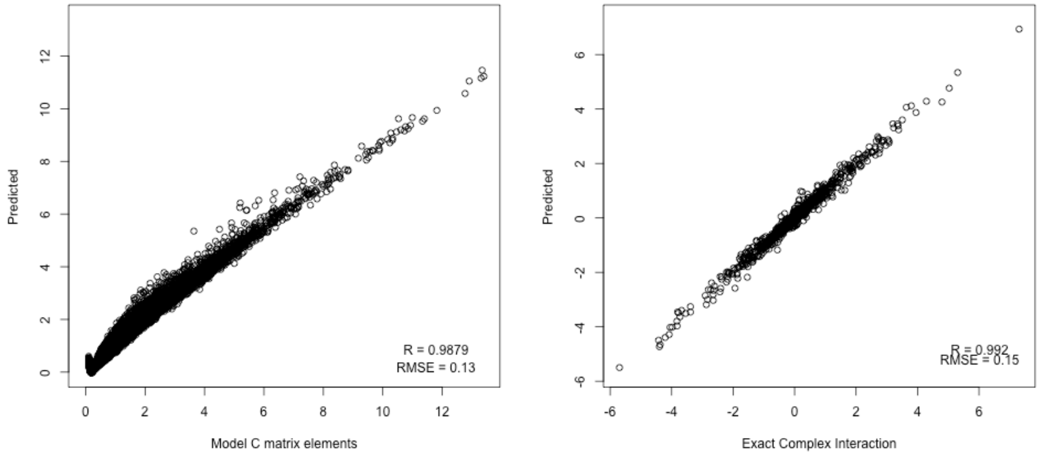

Figure 3.7. Predicted C matrix elements (left) and complex interaction (right) for model system. Features trained on: List 4 (solvent exposure terms). RMSE units (left) = kcal/mol/e^2. RMSE units (right) = kcal/mol.

The results showed that this term was also successful in capturing the matrix elements for this system.

We then tested a third set of features that were termed inverse solvent exposure terms. These were very similar to the solvent exposure terms, except that the inverse of the solvent exposure term was taken for each charge. The larger this term, therefore, the less solvent exposed a charge was, thus the less its pairwise interaction term should be screened.

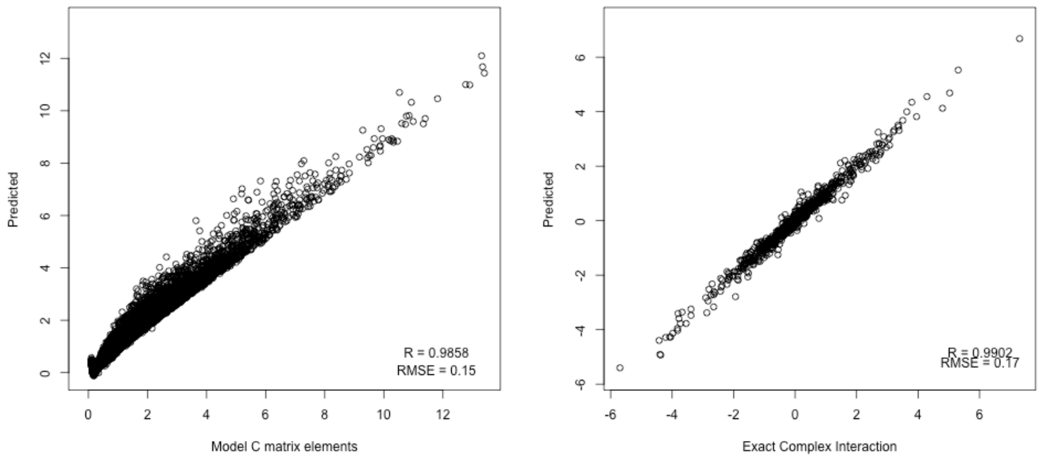

Figure 3.8. Predicted C matrix elements (left) and complex interaction (right) for model system. Features trained on: List 5 (inverse solvent exposure terms). RMSE units (left) = kcal/mol/e^2. RMSE units (right) = kcal/mol.

These predicted values also showed good agreement with the expected values. The matrix elements were slightly overestimated, compared to the predicted matrix elements shown in the previous graph (using the solvent exposure features).

Note that two features were included in all three sets of features (Lists 3-5): the distance between charges, and the inverse of the distance between two charges. These features most likely influence the predictions greatly.

Term regression

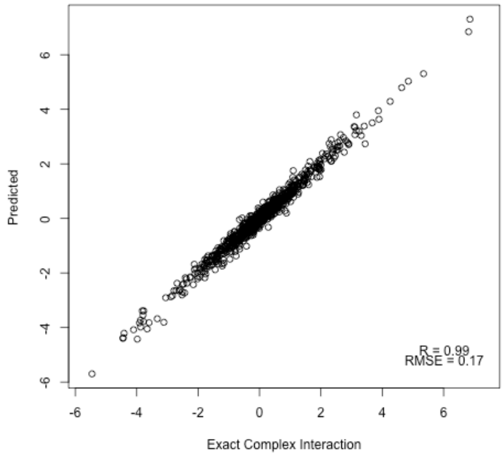

Figure 3.9. Predicted complex interaction after regression on the term. Features trained on: List 3 (burial terms). RMSE units = kcal/mol.

Using the List 3 burial term features, regression on the complex interaction resulted in a good correlation with the expected values, although the matrix elements approach was slightly better in this case (R = 0.9907).

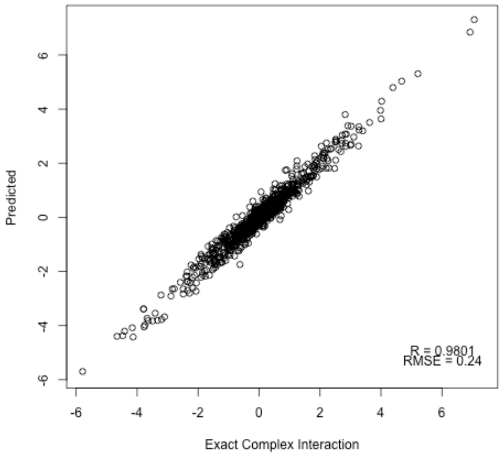

Figure 3.10. Predicted complex interaction after regression on the term. Features trained on: List 4 (solvent exposure terms). RMSE units = kcal/mol.

Regression on the term using the solvent exposure features produced a similar fit, but it was again not as successful as the matrix elements approach (R = 0.992).

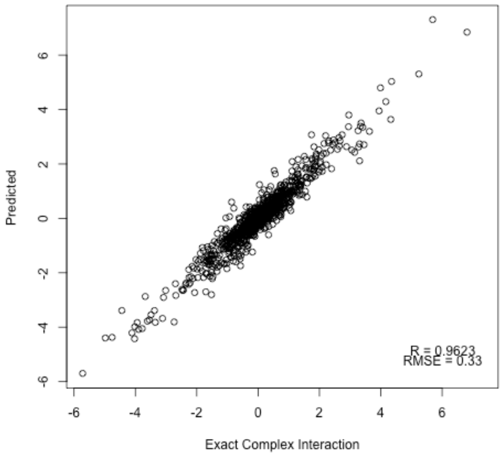

Figure 3.11. Predicted complex interaction after regression on the term. Features trained on: List 5 (inverse solvent exposure terms). RMSE units = kcal/mol.

Regressing on the term using the inverse solvent exposure produced good correlations. The data was less correlated than 1). the matrix elements approach and 2). the previous two sets of features.

Taken together, the results suggest that the matrix elements approach is more promising in the case of interaction. Furthermore, although each of the three sets of features produced good correlations, the features in List 3 were surprisingly successful, and best in the term regression approach. Somehow, these burial terms must approximate features important to interaction. Perhaps charges close to the interface with a large burial term are close to charges on the opposite partner, which would increase the interaction between them. This will be addressed further in the discussion.

Coefficients

All coefficients are shown after standardizing the features by setting the mean equal to 0 and the standard deviation equal to 1.

The tables below show that the inverse distance between charges resulted in large coefficients when regressing on the matrix elements.

	Matrix elements	Term
(Intercept)	0.6684	0.0246
Dist bw charges	0.3715	-1.0760
Inverse dist	0.9904	0.3880
Geom. DI	0.1862	0.9265
Arith. DI	-0.0811	0.0919
PWBprod:3	0.4843	1.8721
PWBprod:4	-0.2452	-1.4179
PWBprod:5	-0.0140	0.3318
PWBprod:6	0.0300	0.0511
PWBprod:7	-0.0258	-0.1092
PWBprod:8	0.0357	0.0598
PWBprod:9	-0.0233	-0.0611
PWBprod:10	-0.0015	-0.0004
PWBsum:3	0.0223	-1.9335
PWBsum:4	0.0368	3.1522
PWBsum:5	0.0719	-1.1323
PWBsum:6	-0.1037	-0.2106
PWBsum:7	-0.0793	0.1269
PWBsum:8	0.1200	0.1312
PWBsum:9	-0.0223	0.1780
PWBsum:10	-0.0414	-0.1682

Table 3.4. Coefficients for matrix elements regression (left) and complex interaction regression (right) for the model system. Features trained on: List 3 (burial terms).

In the matrix elements approach, the most positive coefficient when using the burial terms was the inverse distance between two charges. This makes sense, because the interaction of two charges increases as the distance between them decreases. The most negative coefficient when training on the burial features was the pairwise burial product at a cutoff of 4 angstroms. In the term approach, the most positive and negative coefficients were those for the pairwise burial sum terms at 4 and 3 angstroms, respectively.

	Matrix elements	Term
(Intercept)	0.6684	0.0246
Dist bw charges	0.7101	-0.0453
Inverse dist	1.0990	0.4408
Geom. DS	0.0319	0.0908
Arith. DS	-0.0053	-0.0748
SEprod:3	0.5631	12.5566
SEprod:4	-0.3770	-12.4905
SEprod:5	-0.0672	1.9510
SEprod:6	0.0195	2.5783
SEprod:7	-0.1320	-1.6351
SEprod:8	0.0242	-0.2230
SEprod:9	-0.0238	0.2947
SEprod:10	0.0095	-0.0807
SEsum:3	0.0380	-21.6931
SEsum:4	0.0290	23.6577
SEsum:5	0.0638	-3.6298
SEsum:6	0.1458	-2.3141
SEsum:7	0.0762	1.8341
SEsum:8	0.0284	0.1264
SEsum:9	-0.0435	-0.3558
SEsum:10	-0.0321	0.0527

Table 3.5. Coefficients for matrix elements regression (left) and complex interaction regression (right) for the model system. Features trained on: List 4 (solvent exposure terms).

Similarly, when training on the matrix elements using the solvent exposure terms, the most positive coefficient was the inverse distance between two charges. The most negative coefficient was the pairwise solvent exposure product at a cutoff of 4 angstroms. When training on the term, the solvent exposure sum at 3 angstroms and 4 angstroms were the most negative and positive, respectively.

	Matrix elements	Term
(Intercept)	0.6684	0.0246
Dist bw charges	0.7995	-0.6990
Inverse dist	1.4353	4.2963
Geom. DS	0.0402	0.2502
Arith. DS	0.0135	-0.2343
ISEprod:3	-0.1197	7.2374
ISEprod:4	0.0887	-7.1258
ISEprod:5	-0.0830	1.4299
ISEprod:6	-0.0030	0.0478
ISEprod:7	-0.0041	-0.0367
ISEprod:8	-0.0047	0.0001
ISEprod:9	-0.0016	0.0046
ISEprod:10	-0.0015	0.0174
ISEsum:3	0.0148	-11.8221
ISEsum:4	0.0571	9.2444
iSEsum:5	0.0586	-1.5849
ISEsum:6	-0.0021	-0.0416
ISEsum:7	0.0039	0.0325
ISEsum:8	0.0056	-0.0074
ISEsum:9	0.0086	0.0071
ISEsum:10	0.0078	-0.1274

Table 3.6. Coefficients for matrix elements regression (left) and complex interaction regression (right) for the model system. Features trained on: List 5 (inverse solvent exposure terms).

Again, when using the inverse solvent exposure features and training on the matrix elements, the most positive coefficient was the inverse distance between two charges. The most negative coefficient was the pairwise solvent exposure product at a cutoff of 3 angstroms. Note the small magnitude of the inverse solvent exposure feature coefficients. When training on the term, the most positive and negative coefficient was the pairwise solvent exposure sums at 4 and 3 angstroms. In this case, the solvent exposure sums and products features had low magnitudes of coefficients, while the lower cutoff terms had much higher coefficients.

Predicting ΔG_{elec}

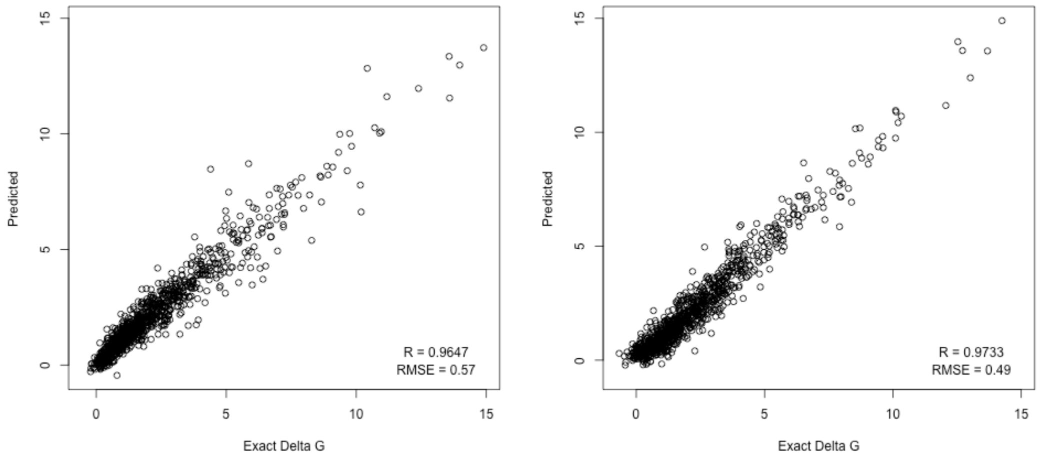

Figure 3.12. ΔG_{elec} after regressing on matrix elements (left) and terms (right). Features trained on: List 1, List 2, and List 3. RMSE units = kcal/mol.

Summing desolvation penalties and complex interaction yielded good correlations in both cases. In both types of regression, the RMSE was less than 0.6 kcal/mol. In all, the model system results suggest that both types of regression are a promising approach to predicting ΔG_{elec}.

PROTEIN-SHAPED SYSTEMS

The same approach used above on the model system can be taken using actual protein complex geometries. After randomly placing charges inside 10 different protein-protein complexes, Poisson-equation derived matrix potentials and free energies were calculated. These matrix potentials and free energy terms can be regressed on, using the features to predict ΔG_{elec}.

Desolvation

Matrix elements

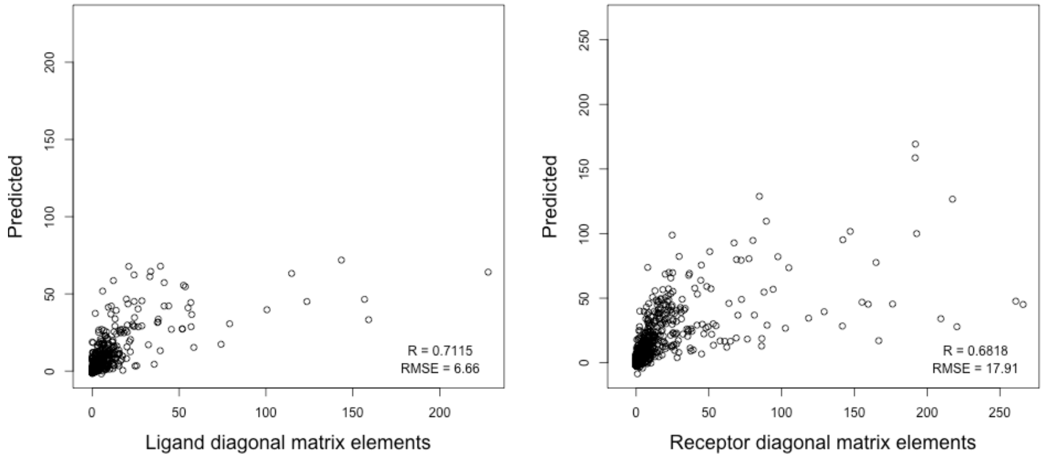

Figure 3.13. Representative predicted diagonal matrix elements from 10 protein-protein complexes. Receptor and ligand are arbitrarily defined. Features trained on: List 1. RMSE units = kcal/mol/e^2.

The ligand and receptor diagonal matrix elements show a similar correlation to each other, but the correlation is much worse compared to the model system. Note that there are more high magnitude diagonal matrix elements on the receptor than on the ligand. The extremely high nature of some of the matrix elements is unphysical. This will be addressed later in the discussion.

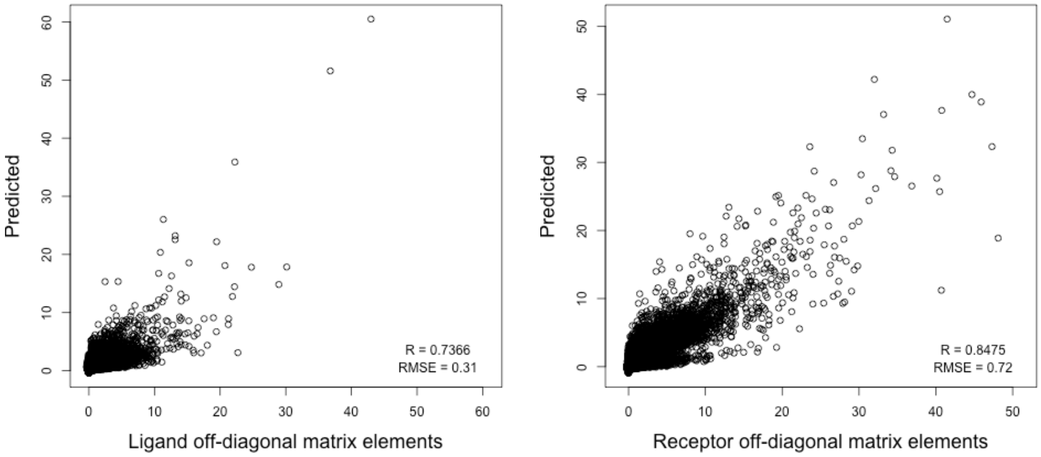

Figure 3.14. Representative predicted off-diagonal matrix elements from 10 protein-protein complexes. Receptor and ligand are arbitrarily defined. Features trained on: List 2. RMSE units = kcal/mol/e^2.

Both the ligand and the receptor off-diagonal matrix elements show a similar correlation. Overall, the correlation is worse than the model system. Note again that the receptor off-diagonal matrix elements tend to be higher in magnitude than those of the ligand. The implications of this will be mentioned in the discussion section.

These predicted matrix elements can be used to construct "approximate" L and R matrices. When multiplied by charge, these yield ligand and receptor desolvation penalties, shown below.

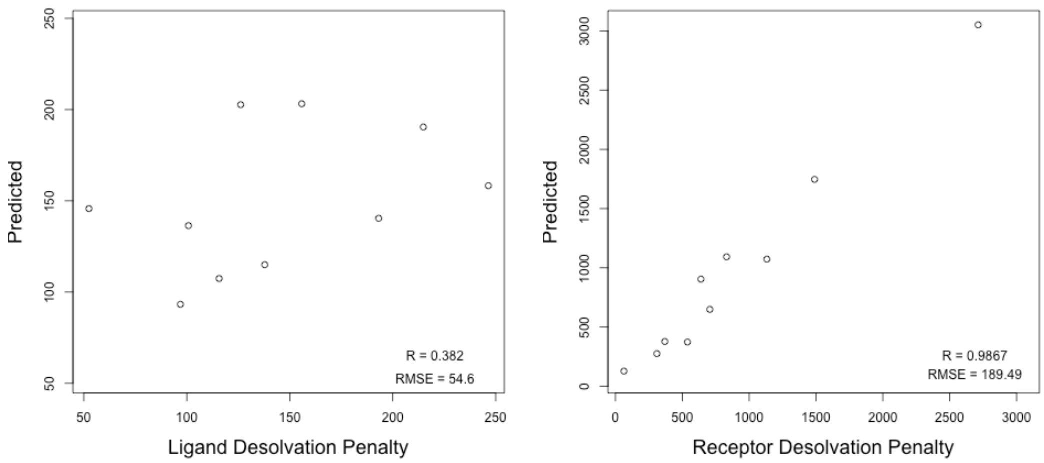

Figure 3.15. Predicted vs. exact desolvation penalties for ligand and receptor, after training on the matrix elements. RMSE units = kcal/mol.

The ligand desolvation penalty shows a much worse correlation than the receptor desolvation penalty. This is probably because the receptor desolvation penalties are much higher magnitude, and one point appears to dominate the receptor graph. The root mean square error for the receptor is 189 kcal/mol, while it is only 54 kcal/mol for the ligand desolvation penalty.

Term regression

Regression can also be performed using the ligand and receptor desolvation penalties.

The following figure, unlike those shown previously, which show 10-fold cross-validation, shows the predictions for training data after regressing on the term.

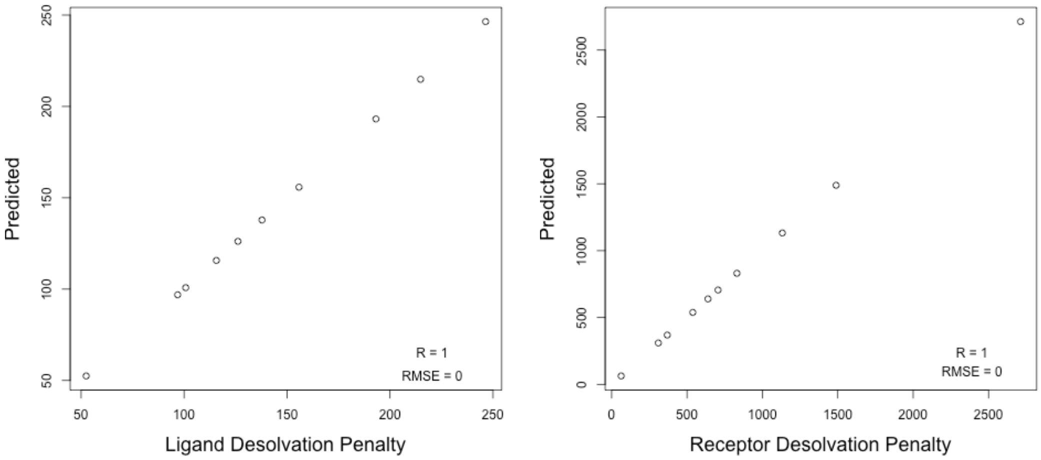

Figure 3.16. Predicted vs. exact desolvation penalties for ligand (left) and receptor (right), after training on the term and predicting that same term.

A perfectly linear fit is observed when training on the term data and predicting the term. The implications of this will be addressed in the discussion, but note that this indicates over-fitting, which is when the number of parameters (features) exceeds the number of observations (terms).

When data is overfit, the training predictions are often very good, but the testing data predictions are much worse. The results from the testing data, shown below, suggest overfitting as well. When 10-fold cross-validation is performed, the following results are obtained.

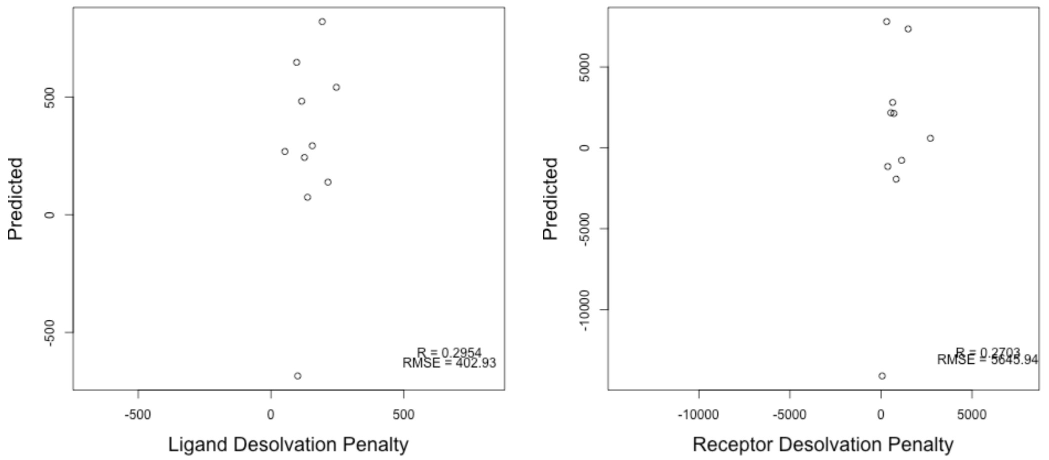

Figure 3.17. Predicted vs. exact desolvation penalties for ligand (left) and receptor (right), after training on the term and predicting that same term. RMSE units = kcal/mol.

The predicted values deviate largely from the expected result, with correlations below 0.3. Taken together, these results highly suggest overfitting. For this reason, we focused on regressing on the matrix elements for the proteins, rather than the term.

Coefficients

Only the coefficients produced by regressing on the matrix elements will be discussed, because regressing on the term had such low predictive value.

	Proteins L	Proteins R
(Intercept)	1.9088	7.8735
DI	0.2217	1.1876
BT:3	-1.1159	4.0957
BT:4	0.2647	-4.7312
BT:5	2.4031	5.4283
BT:6	0.6675	4.1852
BT:7	-2.2356	0.7400
BT:8	2.3247	-8.1887
BT:9	2.7394	14.0177
BT:10	2.5447	4.1364
Solvent boxes	-0.3312	0.2758

Table 3.7. Single atom desolvation coefficients, training on the off-diagonal matrix elements. Features trained on: List 1. Highest positive coefficient shown in blue, negative in red.

The two most positive coefficients are the same: the burial terms using a cutoff of 9 angstroms.

	Proteins L	Proteins R
(Intercept)	0.1157	0.3995
Dist between	-0.0353	-0.0893
Geom. DI	-0.1063	0.0952
Arith. DI	0.1705	0.0817
PWBprod:3	-0.3541	1.9294
PWBprod:4	0.3702	-2.3645
PWBprod:5	-0.0575	1.2857
PWBprod:6	-0.1101	-0.4225
PWBprod:7	0.1677	0.1757
PWBprod:8	-0.0239	-0.2730
PWBprod:9	-0.0100	0.0603
PWBprod:10	0.0265	0.0165
PWBsum:3	0.7327	0.0583
PWBsum:4	-0.5248	0.9622
PWBsum:5	0.1041	-0.7498
PWBsum:6	0.0405	0.6584
PWBsum:7	0.0223	-0.4929
PWBsum:8	0.0028	0.2761
PWBsum:9	-0.0050	0.0830
PWBsum:10	0.0186	0.0425

Table 3.8. Pairwise desolvation coefficients, training on the off-diagonal matrix elements. Features trained on: List 1. Highest positive coefficient shown in blue, negative in red.

Interaction

Matrix elements

The plots below are generated using the features listed in Lists 3-5 as specified. Each of these features contains the following features in common: distance between charges, and inverse distance between charges.

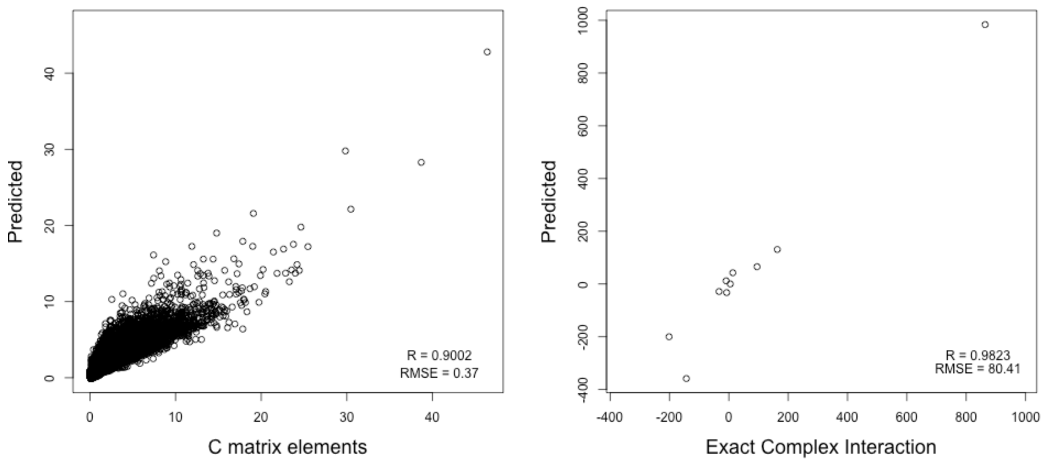

Figure 3.18. Predicted C matrix elements (left) and complex interaction (right) for proteins. Features trained on: List 3 (burial terms). RMSE units (left) = kcal/mol/e^2. RMSE units (right) = kcal/mol.

Training on the burial terms produces a good correlation with the expected values of the matrix elements. When multiplied by charge, the correlation is remarkably good but appears to be dominated by the highest-magnitude value. Regression on the burial terms does give a root mean square error of approximately 80 kcal/mol.

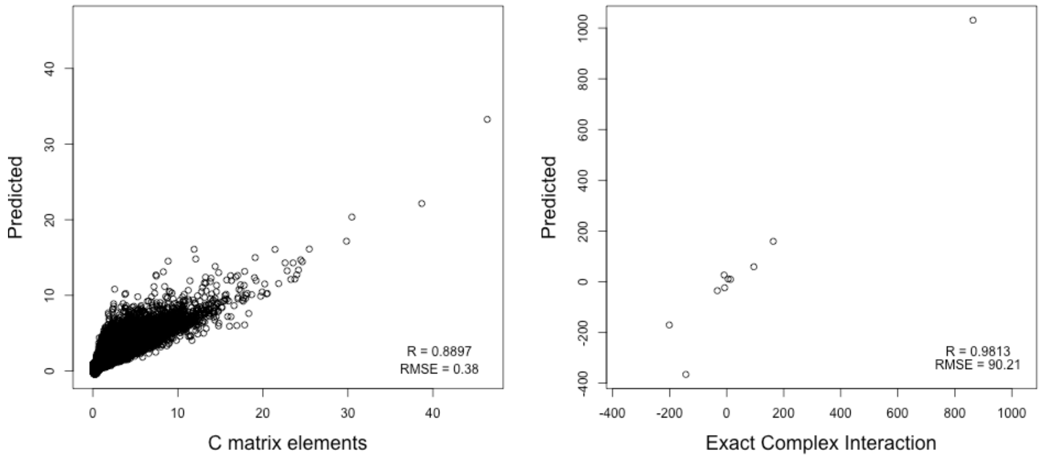

Figure 3.19. Predicted C matrix elements (left) and complex interaction (right) for proteins. Features trained on: List 4 (solvent exposure terms). RMSE units (left) = kcal/mol/e^2. RMSE units (right) = kcal/mol.

Training on the solvent exposure terms underestimates the matrix elements at higher values.

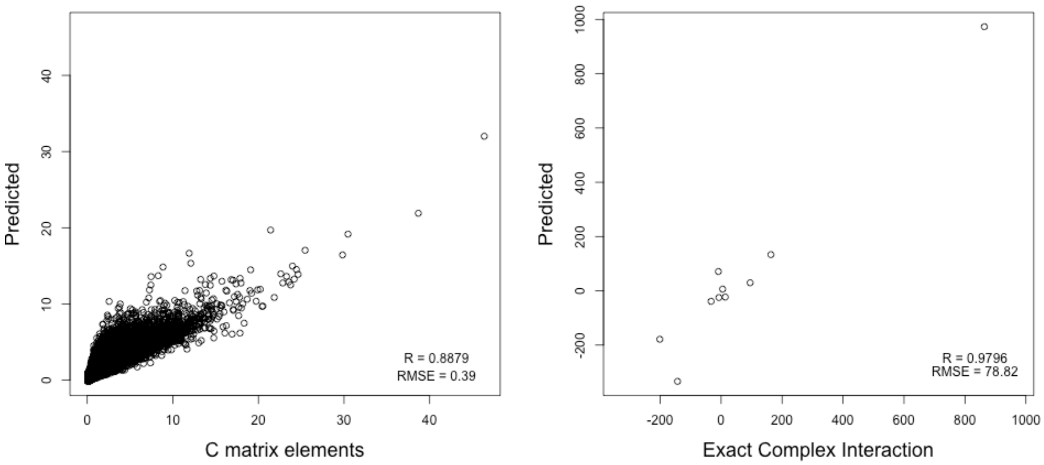

Figure 3.20. Predicted C matrix elements (left) and complex interaction (right) for proteins. Features trained on: List 5 (inverse solvent exposure terms). RMSE units (left) = kcal/mol/e^2. RMSE units (right) = kcal/mol.

Training on the inverse solvent exposure terms gives essentially the same result as the solvent exposure terms. The prediction is worse for the higher-magnitude matrix elements.

Coefficients

The coefficients for the three sets of features used are shown below. In all cases, the inverse distance between charges was the most positive coefficient.

	Matrix elements
(Intercept)	0.4473
Dist bw charges	0.3075
Inverse dist	0.6752
Geom. DI	0.0000
Arith. DI	0.0555
PWBprod:3	0.1310
PWBprod:4	-0.2833
PWBprod:5	0.2104
PWBprod:6	-0.0375
PWBprod:7	0.0479
PWBprod:8	-0.0090
PWBprod:9	0.0062
PWBprod:10	0.0540
PWBsum:3	0.3568
PWBsum:4	0.1336
PWBsum:5	-0.2878
PWBsum:6	0.0452
PWBsum:7	-0.0538
PWBsum:8	0.1719
PWBsum:9	-0.0589
PWBsum:10	-0.0230

Table 3.9. Coefficients for burial terms (List 3).

For the burial features, the most positive coefficient is the inverse distance between charges, while the most negative is the pairwise burial sum at a cutoff of 5 angstroms.

	Matrix elements
(Intercept)	0.4473
Dist bw charges	0.4579
Inverse dist	1.0546
Geom. DS	0.1659
Arith. DS	-0.2262
SEprod:3	-0.4674
SEprod:4	0.3139
SEprod:5	0.1151
SEprod:6	-0.0322
SEprod:7	0.0505
SEprod:8	-0.0043
SEprod:9	0.0025
SEprod:10	-0.0068
SEsum:3	0.8730
SEsum:4	-0.4771
SEsum:5	-0.4087
SEsum:6	0.1999
SEsum:7	-0.2118
SEsum:8	-0.0005
SEsum:9	0.0045
SEsum:10	-0.0097

Table 3.10. Coefficients for solvent exposure terms (List 4).

The most positive coefficient is the inverse distance between charges when training on the solvent exposure terms. The most negative is the pairwise solvent exposure sum at a cutoff of 4 angstroms.

	Matrix elements
(Intercept)	0.4473
Dist bw charges	0.4627
Inverse dist	1.0314
Geom. DS	0.0300
Arith. DS	-0.0356
ISEprod:3	-0.0110
ISEprod:4	-0.0057
ISEprod:5	0.0078
ISEprod:6	0.0215
ISEprod:7	0.0032
ISEprod:8	-0.0004
ISEprod:9	-0.0025
ISEprod:10	0.0054
ISEsum:3	-0.0219
ISEsum:4	0.0440
ISEsum:5	0.0249
ISEsum:6	0.0511
ISEsum:7	0.0200
ISEsum:8	0.0166
ISEsum:9	-0.0257
ISEsum:10	-0.0389

Table 3.11. Coefficients for inverse solvent exposure terms (List 5).

The most positive coefficient is the inverse distance between charges when training on the inverse solvent exposure terms. The most negative is the pairwise inverse solvent exposure sum at a cutoff of 10 angstroms. The implications of these coefficients will be addressed in the discussion.

Predicting ΔG_{elec}

The matrix elements regression was the best approach to take with the proteins, because of issues overfitting the terms.

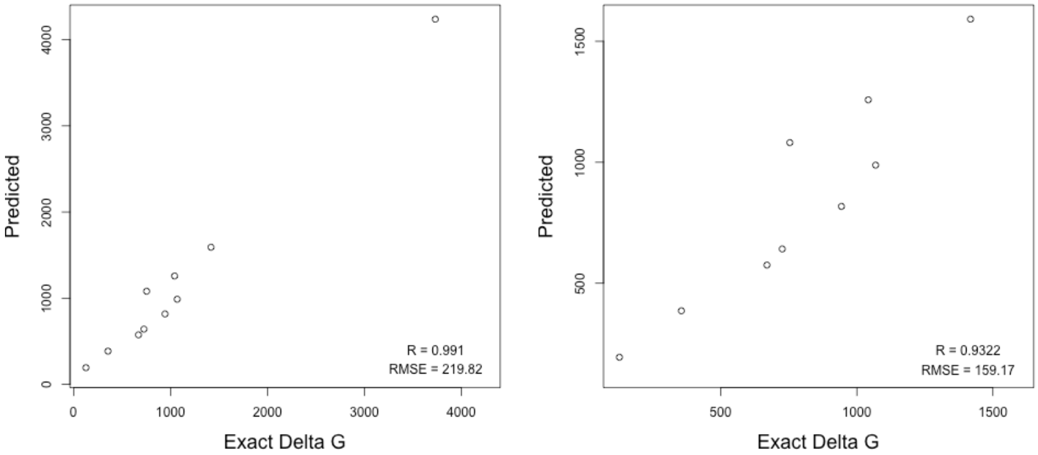

Figure 3.21. Predicted ΔG_{elec} after training on the matrix elements. Features used: List 1, List 2, and List 3 (best complex interaction results). Left: training on 10 structures. Right: same data with outlier removed. RMSE units (right) = kcal/mol.

The predicted ΔG_{elec} is well correlated with the expected. However, one point dominates the graph because its magnitude is physically unrealistic. With this point removed, the predictions are still well correlated, and the absolute error decreases somewhat.

CROSS-SYSTEMS

Thus far, we have trained on both a model system and protein shapes. It is interesting to compare how the coefficients on one system carry over to the other. For the sake of comparison, only the coefficients produced by regressing on the matrix elements will be compared, because the coefficients from the regression on the term for the proteins had low predictive value.

Desolvation

For the sake of comparison, the coefficients from both systems for desolvation are listed below. The implications of these coefficients will be discussed further in the discussion section.

	Model system L	Model system R	Proteins L	Proteins R
(Intercept)	1.9072	0.3640	1.9088	7.8735
DI	0.4338	0.1184	0.2217	1.1876
BT:3	0.3337	0.5282	-1.1159	4.0957
BT:4	0.9507	-0.2252	0.2647	-4.7312
BT:5	-0.0505	0.0587	2.4031	5.4283
BT:6	-0.0078	0.0917	0.6675	4.1852
BT:7	-0.3233	0.0963	-2.2356	0.7400
BT:8	1.0081	0.1453	2.3247	-8.1887
BT:9	0.9509	0.1541	2.7394	14.0177
BT:10	-0.2962	0.1353	2.5447	4.1364
Solvent boxes	0.0000	0.0000	-0.3312	0.2758

Table 3.12. Single atom desolvation coefficients, training on the diagonal matrix elements. Features trained on: List 1.

In looking at these coefficients, the magnitude is clearly highly variable from system to system.

Note that one feature, the number of solvent-exposed boxes, is always zero in the model system. This is because all charges were constrained to be within the model system, which was outlined by atoms. Splitting up the local geometry around a charge into boxes and counting the number of atom contacts never produced an empty box, so the feature was always zero. This likely affects the fit when applying the model system coefficients to the proteins and vice versa.

	Model system	Model system	Proteins L	Proteins R

	L	R		
(Intercept)	0.6329	0.1166	0.1157	0.3995
Dist between	-0.1282	-0.0263	-0.0353	-0.0893
Geom. DI	-0.1974	-0.0708	-0.1063	0.0952
Arith. DI	0.1610	0.0304	0.1705	0.0817
PWBprod:3	3.1074	0.2636	-0.3541	1.9294
PWBprod:4	-3.4784	-0.1688	0.3702	-2.3645
PWBprod:5	1.1078	0.0166	-0.0575	1.2857
PWBprod:6	0.4497	-0.0037	-0.1101	-0.4225
PWBprod:7	-0.4450	-0.0040	0.1677	0.1757
PWBprod:8	0.0889	-0.0072	-0.0239	-0.2730
PWBprod:9	-0.0308	0.0038	-0.0100	0.0603
PWBprod:10	0.0081	-0.0001	0.0265	0.0165
PWBsum:3	-3.0078	-0.1147	0.7327	0.0583
PWBsum:4	3.9630	0.1037	-0.5248	0.9622
PWBsum:5	-1.1558	0.0010	0.1041	-0.7498
PWBsum:6	-0.7231	0.0170	0.0405	0.6584
PWBsum:7	0.3551	0.0219	0.0223	-0.4929
PWBsum:8	0.3303	0.0081	0.0028	0.2761
PWBsum:9	0.0867	0.0261	-0.0050	0.0830
PWBsum:10	-0.0929	0.0068	0.0186	0.0425

Table 3.13. Pairwise desolvation coefficients, training on the off-diagonal matrix elements. Features trained on: List 2.

The model system features share the same most negative coefficient: the pairwise burial product using a cutoff of 4 angstroms. That feature also had the most negative coefficient in the protein receptor model. The model system and protein receptors both had the most positive coefficient as the pairwise burial product with a cutoff of 3 angstroms.

Applying the Model System Coefficients to Proteins

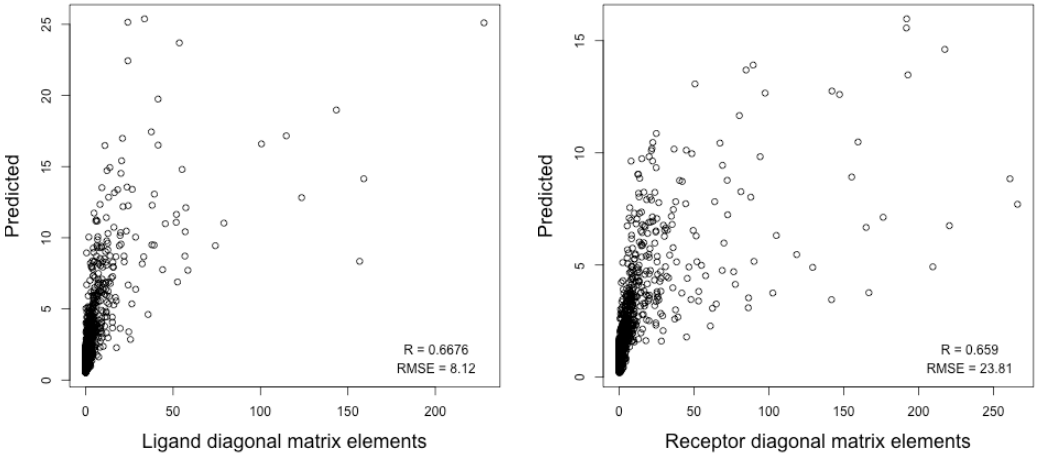

Figure 3.22. Using the model system ligand diagonal coefficients to predict protein L (left) and R (right) diagonal matrix elements. Ligand and receptor here are arbitrarily defined. RMSE units = kcal/mol/e^2.

While the correlation is good, the magnitude is greatly underestimated. This makes sense, because the proteins had much higher matrix elements than the model system, indicating much higher desolvation penalties.

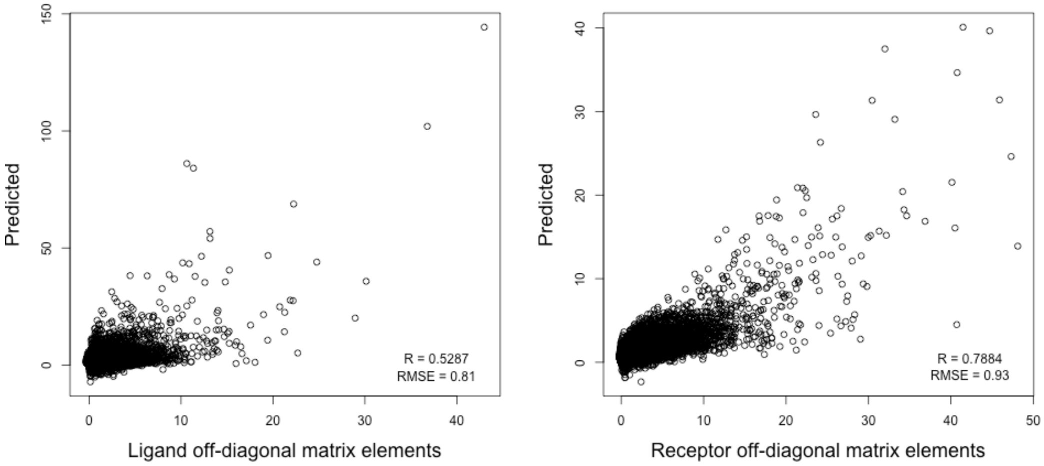

Figure 3.23. Using the model system ligand off-diagonal coefficients to predict protein L (left) and R (right) off-diagonal matrix elements. RMSE units = kcal/mol/e^2.

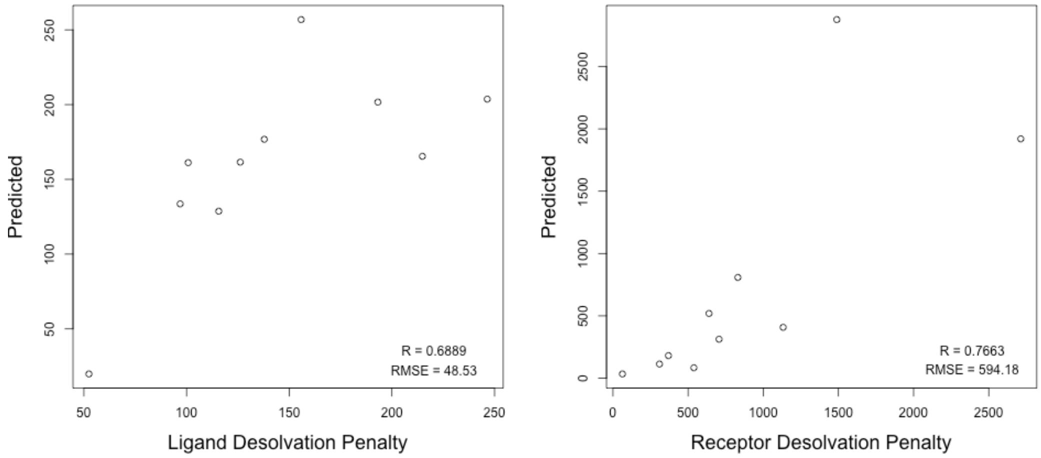

Figure 3.24. Predicting ligand desolvation penalty (left) and receptor desolvation penalty (right) after using the ligand model system coefficients. RMSE units = kcal/mol.

When these predicted matrix elements are multiplied by charge to produce desolvation penalties, the error is very high for both ligand and receptor (RMSE = 48 and 594 kcal/mol for ligand and receptor, respectively). This is not surprising given that the matrix elements themselves were underestimated so drastically. In all, these results suggest that training on a shape-simplified system like the model system does not produce coefficients good enough to predict energies for protein shapes, although there is reasonable correlation in some cases.

Applying the Protein Coefficients to the Model System

The graphs below show that applying the protein coefficients to the model system result in vastly overestimating the magnitude of the matrix elements, and the magnitude of the desolvation penalties.

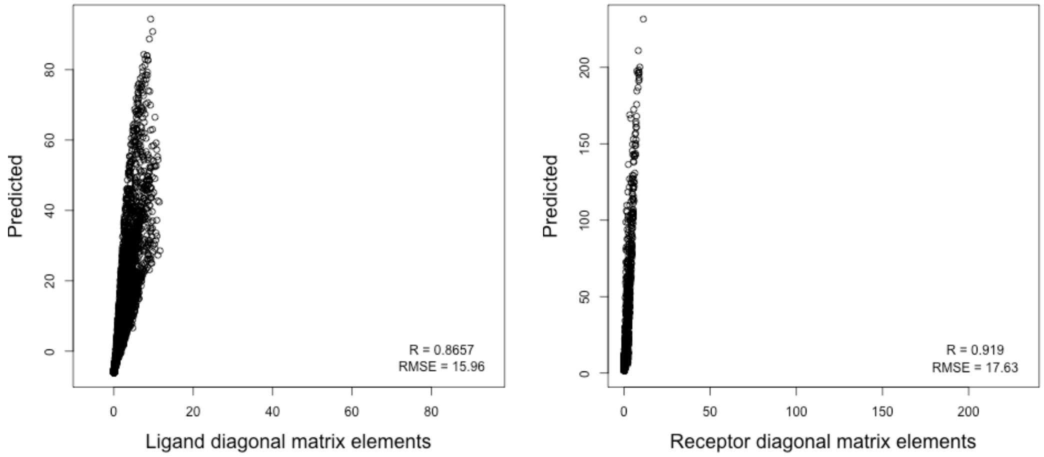

Figure 3.25. Using the protein diagonal coefficients to predict model system L (left) and R (right) diagonal matrix elements.

Using the protein coefficients results in predictions that are surprisingly well correlated with the expected values. However, the predictions are vastly overestimated.

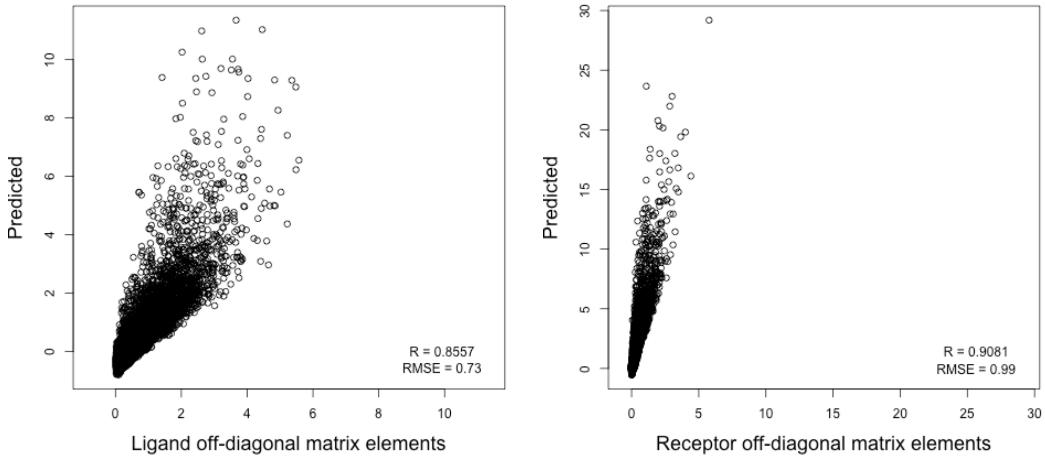

Figure 3.26. Using the protein off-diagonal coefficients to predict model system L (left) and R (right) off-diagonal matrix elements.

The same trend is seen for the off-diagonal matrix elements, though not to the same degree because the magnitudes of the matrix elements are so small. The implications of this will be addressed in the discussion.

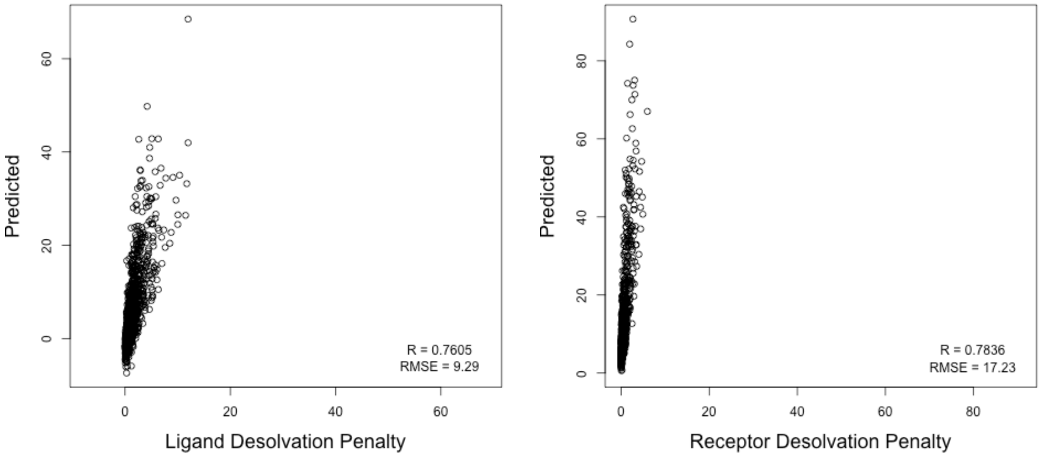

Figure 3.27. Predicting model system ligand desolvation penalty (left) and receptor desolvation penalty (right) after using the protein coefficients.

Using the predicted matrix elements to predict the desolvation penalty resulted in a reasonably good correlation. However, the fit is not as good as the fit for the matrix elements themselves.

SUMMARY

In all, the results suggest that training on the systems of interest make the best predictions for those same systems. Because we want to identify the set of features and their coefficients that could best predict ΔG_{elec} for protein-protein complexes, we will most likely need to train on natural charge distributions to obtain the optimal result.

FEATURES

List 1. Single atom desolvation features

Number	Single atom feature
1	Distance to interface
2	BT: 3
3	BT: 4
4	BT: 5
5	BT: 6
6	BT: 7
7	BT: 8
8	BT: 9
9	BT: 10
10	Number of solvent-exposed boxes

List 2. Pairwise atom desolvation features

Number	Pairwise Feature
1	Distance between charges
2	Geometric distance to interface
3	Arithmetic distance to interface
4	PWBproduct: 3
5	PWBproduct: 4
6	PWBproduct: 5
7	PWBproduct: 6
8	PWBproduct: 7
9	PWBproduct: 8
10	PWBproduct: 9
11	PWBproduct: 10
12	PWBsum: 3
13	PWBsum: 4
14	PWBsum: 5
15	PWBsum: 6
16	PWBsum: 7
17	PWBsum: 8
18	PWBsum: 9
19	PWBsum: 10

*PWBproduct = BT1*BT2/distance between charges

*PWBsum = BT1+BT2/distance between charges

List 3. Pairwise atom features for complex interaction: burial.

Number	Pairwise Feature
1	Distance between charges
2	Geometric distance to interface
3	Arithmetic distance to solvent-expoPWBd atom
4	PWBproduct: 3
5	PWBproduct: 4
6	PWBproduct: 5
7	PWBproduct: 6
8	PWBproduct: 7
9	PWBproduct: 8
10	PWBproduct: 9
11	PWBproduct: 10
12	PWBsum: 3
13	PWBsum: 4
14	PWBsum: 5
15	PWBsum: 6
16	PWBsum: 7
17	PWBsum: 8
18	PWBsum: 9
19	PWBsum: 10

List 4. Pairwise atom features for complex interaction: solvent exposure

Number	Pairwise Feature
1	Distance between charges
2	Geometric distance to solvent-exposed atom
3	Arithmetic distance to solvent-exposed atom
4	SEproduct: 3
5	SEproduct: 4
6	SEproduct: 5
7	SEproduct: 6
8	SEproduct: 7
9	SEproduct: 8
10	SEproduct: 9
11	SEproduct: 10
12	SEsum: 3
13	SEsum: 4
14	SEsum: 5
15	SEsum: 6
16	SEsum: 7
17	SEsum: 8
18	SEsum: 9
19	SEsum: 10

*SEproduct = SE1*SE2/distance between charges

*SEsum = SE1+SE2/distance between charges

List 5. Pairwise atom features for complex interaction: inverse solvent exposure

Number	Pairwise Feature
1	Distance between charges
2	Geometric distance to solvent-exposed atom
3	Arithmetic distance to solvent-exposed atom
4	Inverse SEproduct: 3
5	Inverse SEproduct: 4
6	Inverse SEproduct: 5
7	Inverse SEproduct: 6
8	Inverse SEproduct: 7
9	Inverse SEproduct: 8
10	Inverse SEproduct: 9
11	Inverse SEproduct: 10
12	Inverse SEsum: 3
13	Inverse SEsum: 4
14	Inverse SEsum: 5
15	Inverse SEsum: 6
16	Inverse SEsum: 7
17	Inverse SEsum: 8
18	Inverse SEsum: 9
19	Inverse SEsum: 10

DISCUSSION

We begin this section with a few caveats to interpreting the data. Next, implications from the results will be addressed. Then, future steps will be suggested to build off this work. Overall, the results showed that this regression-based model is a promising approach to predicting ΔG_{elec}. More work must be done to address certain data fitting issues, and expand the structures trained on to more biologically relevant systems.

Caveats

There are a few concerns with this project that will be addressed by future students. One is that, in this work, certain energies for the proteins were physically unrealistic. A second concern is that the regression model may be overfitting the data. Thirdly, standardization of the data was carried out so that coefficients could be directly compared; however, this constrained the way the regression was carried out, forcing the inclusion of a y-intercept. Last, it is important to note that the systems trained on thus far consist of randomly placed charges inside protein shapes. Natural systems are of interest in the future.

Magnitude of certain protein terms

In certain cases, the magnitude of desolvation penalties for the proteins was extremely high (>1000 kcal/mol). This number is extremely large and physically unrealistic. Upon investigation into why certain values were so high, charges were discovered that were placed less than 1 angstrom away from the dielectric boundary. In cases when the charge was less than one angstrom away from protein edge and located on the binding interface, the L and R diagonal matrix elements were observed to be far above the maximum expected value of approximately 39.45 kcal/mol/e^2 for a sphere of radius 1 Å that is completely, 100% desolvated [calculations performed by M. Radhakrishnan and J. Bardhan]. This was due to the fact that not all solvent-exposed atoms were identified when placing charges, and charges were only constrained to be at least 1 angstrom away from the dielectric boundary if the atom it was placed inside was a solvent-exposed atom. In the future, CHARMM will be used to ensure that all the solvent exposed atoms are identified correctly.[39] For the purposes of this work, it is important to keep in mind that the training data includes values that are physically unrealistic. However, the error in charge placement only occurred in a subset of the data, and the vast majority of charges are indeed physically realistic.

Overfitting

A potential danger in this project was overfitting, which occurs when the model describes noise rather than the relationship between variables. This can occur when there are too many parameters to fit the number of observations. When overfitting occurs, the model has poor predictive value. To eliminate the possibility of overfitting, we performed 10-fold cross-validation, using 90% of the observations to predict the remaining 10% of the data for a total of 10 times. Because the number of observations, either the terms or the matrix elements, was often so much greater than the number of features, it is unlikely our data was overfitted for the model system. However, the protein data was accumulated using large numbers of charges but fewer overall runs, leading to many matrix elements but few terms. When we trained on all the desolvation terms and predicted those same terms, a perfectly linear fit was observed, which is suspicious in itself. However, when we performed 10-fold cross-validation, the predictions deviated largely from the expected result, strongly suggesting that the model was overfitted. Because the proteins carried hundred of charges, the quantity of data required to regress on the terms for the protein is large, requiring thousands of single atom and pairwise atom features multiplied by thousands of charge pairs for just ten data points. This makes it difficult to train on large numbers of terms. It is possible that we could randomly place smaller charge distributions on more proteins, producing more terms, which would hopefully solve the overfitting issue. Overfitting is a concern, particularly when it comes to the protein data.

Data standardization

Ideally, linear regression ought to be carried out by constraining the y-intercept to be zero, such that the coefficients solely determine the prediction. The plots shown in this work are produced after standardizing the coefficients, which necessitated including a y-intercept. Standardizing the data, subtracting the mean and dividing by the standard deviation, was essential so that coefficients could be compared to each other. However, the y-intercept adds a certain amount of noise to the system, so the coefficients are not sole predictors of the output data.

Biological significance

The proteins trained on in this work were created through random charge distributions. Biologically relevant systems will be investigated in later work. While we hope that this theoretical model will carry over to natural systems, it is too early to assume that it will. Furthermore, the matrix elements of natural charge distributions are much smaller than those used in this work. A good model will need to produce much smaller RMSE values than those seen in this work, less than 1 kcal/mol. Additionally, natural proteins contain thousands of charges. To develop a model based on these natural charge distributions, these charges must be used. To make this more computationally feasible, it is possible to select a subset of these charges as training data. However, this may not adequately sample the different types of interactions present in real proteins.

Implications

Given the early stage of this project, it is important not to over-interpret the coefficients, which are still preliminary. However, examination of the coefficients can elucidate important aspects of predicting binding energies. First, the magnitude and direction of the coefficients in each system can indicate the importance of each feature. Secondly, the robustness of the coefficients between systems can yield insights into important structural differences between systems.

Coefficients

We can compare the magnitude and direction of the coefficients both between and among systems. A positive coefficient means that a large value of the feature increases the matrix element for that charge, while a negative coefficient means that a large value of the feature decreases the matrix element.

Model system

1). Desolvation

The model system coefficients were somewhat difficult to interpret, because they did not indicate that the same features were important to both ligand and receptor. However, the structural differences between the receptor and the ligand (e.g., the binding cavity located on the receptor) may account for these differences.

The receptor burial term coefficients were largest at smaller cutoffs than the ligand, indicating that the charge's nearest surroundings were most important to predicting its desolvation penalty. While the largest coefficient for the ligand was the burial term for all atoms within 8 angstroms, the largest coefficient for the receptor was the burial term for all atoms within 3 angstroms. Similarly, the most negative coefficient for the ligand was the burial term at 7 angstoms, while a burial term at 4 angstroms was most negative for the receptor. This difference is not entirely surprising, given that the receptor has a different shape from the ligand. It is possible that for the receptor, the burial terms at higher cutoffs were not sensitive enough to the presence of the binding cavity. In other words, smaller cutoffs may do better at accounting for the presence of the binding cavity, because atoms on the ligand would be closer to charges on the receptor. However, the difference in both magnitude and direction make interpretation of the relative importance of features difficult.

For the pairwise features, the largest coefficient for the ligand was the pairwise burial sum using a cutoff of 4 angstroms, while the receptor was a pairwise burial product using a cutoff of 3 angstroms. The most negative coefficient for both the ligand and the receptor was the pairwise burial product, using a cutoff of 4 angstroms. These are both features that examine the close surrounding environment of the charge. Since these features are so similar, yet the coefficients are so different in direction, it is possible that they somehow compensate for the other. Perhaps the positive coefficient overestimates the matrix element, while the negative coefficient helps underestimate, helping the matrix element be predicted correctly.

2). Interaction

Three sets of features were used to predict complex interaction: burial terms, solvent exposure terms, and inverse solvent exposure terms. Each had essentially the same success in the prediction.

Despite not being intuitively important features, the burial features produced a good fit. This result may perhaps indicate that the burial features are in fact good predictors of interaction. The coefficients for the burial features showed good agreement when regressing on the matrix elements and the term. There may technically be no "ligand" or "receptor" when the two are in a complex, but the burial features may approximate how close the charge is to the atoms on the partner, and thus, how closely it can interact with the opposite charge.

Two additional sets of features were used to approximate the level of solvent screening that each charge feels. These features resulted in approximately the same fit as the burial features. The solvent exposure features, which add up the reciprocal distances to the solvent-exposed atoms within certain cutoff distances, are large when a charge is solvent exposed. However, a highly solvent-exposed charge will have its interactions dampened because of the solvent. Thus, because of the concern that the directionality of the solvent exposure features was wrong, a third set of features was added that took the reciprocal of the initial solvent exposure terms. However, we found that all three sets of features predicted interaction well. It is important to note that all three sets used had certain features in common, most notably the inverse of the distance between two charges, which was strikingly predicted to be the most positive coefficient in all cases.

The fact that all three sets of features were successful in predicting interaction was perhaps due to the fact that they all contained this feature in common.

One might expect that the inverse solvent exposure feature coefficients would be opposite in sign to the solvent exposure feature coefficients. After all, the directionality is reversed. However, this was not observed. This warrants future study.

Protein shapes

Due to issues with overfitting during regression on the term, the following discussion only pertains to regression on the matrix elements.

1). Desolvation

The two most positive coefficients for desolvation were the same for the ligand and receptor: the burial terms using a cutoff of 9 angstroms. This was encouraging, given that the ligand and receptor have no inherent structural differences and should have similar coefficients. This result is similar to what was seen for the model system ligand, in which the burial term within 8 angstroms was the most positively correlated.

However, in the model system, the most positive receptor coefficient was the burial term within 3 angstroms. Previously, we hypothesized that because of the more irregular shape of the receptor, the local geometry around the charge was most important, leading to the large coefficient for the 3 angstrom burial term. One might expect that because protein shapes have such irregular geometry, the more immediate surroundings (ie, burial term within 3 angstroms) would be most important. Yet, the larger distance cutoffs were observed to be more important. It is possible that because the protein shapes are larger, charges are further from the partner, with relatively few charges being within

10 or fewer angstroms of the partner. Features at larger cutoffs may be more important simply because more charges have features defined for them.

2). Interaction

Similar to the results for the model system, each of the three sets of features resulted in predictions with good correlations. In all cases, the most positive coefficient was the inverse distance between charges. When training on the burial features, the most negative coefficient was the pairwise burial sum at a cutoff of 5 angstroms. The most negative is the pairwise solvent exposure sum at a cutoff of 4 angstroms, which meant that high levels of solvent exposure decreased the matrix elements at those values. This is physically realistic because the solvent will screen the interaction. In the third set of features, the most negative coefficient was the inverse solvent exposure term at a cutoff of 10 angstroms. This is puzzling, because a large value of this feature meant that the atom was not solvent exposed, and the negative coefficient meant that a large value reduced the prediction for the matrix element. However, the actual magnitude of this coefficient was relatively low (-0.03), suggesting that it was not that important to the prediction.

Predictions and fit

Model system

The model system results showed that it is possible to regress on both the matrix elements and the terms. Several issues need to be addressed.

In the model system for desolvation, the R diagonal matrix elements are systematically overestimated, while the L diagonal matrix elements are not. This is interesting, given that the model was trained directly on the receptor matrix elements themselves. The only difference between the ligand and receptor is the receptor’s binding cavity, which must be affecting the level of desolvation for the charges inside. This indicates that perhaps the features do not do a good job of quantifying the geometry around the receptor. However, one would expect that the presence of the binding cavity would increase the actual value of the matrix element, and that the feature would underestimate it.

Furthermore, more ligand charges pay a high desolvation penalty than receptor charges. This could be an artifact of the system. The ligand was much smaller than the receptor, so the likelihood of a charge being closer to the solvent and binding interface was greater.

A surprising result occurred when training on the matrix elements: the fit for the predicted term (multiplying the predicted matrix elements by charge) was better than the predicted matrix elements themselves. While one might expect the error in each of the individual matrix elements to compound, the error appears to instead cancel out. This observation also requires further investigation.

We observed that training on the term itself sometimes produced a greater overall fit than training on the matrix elements. It is difficult to say which is a better approach. On the one hand, the error in each of the matrix elements appears to cancel out when multiplied by charge. The features can be directly and intuitively correlated to the matrix elements, so that is also a promising approach. However, regression on the term would intuitively seem to be a more promising approach, because it avoids any potential source of error in the matrix elements. Both methods show promise, and both will be investigated more in the future.

Protein shapes

Training on protein shapes, unsurprisingly, produced worse correlations than the model system. This is to be expected, given that the shape of the model system was so simple compared to the irregular geometry of proteins. Two points must be discussed further: the correlations and the impact of certain physically unrealistic values for desolvation.

Both the ligand and the receptor diagonal and off-diagonal matrix elements show a similar correlation, which makes sense because the "ligand" and "receptor" were arbitrarily defined. However, there were more high-magnitude matrix elements in the receptor rather than the ligand. Because the model underestimated the value of these matrix elements, predicting the term from these matrix elements was much worse for the ligand than for the receptor in terms of overall fit, but better in terms of root mean square error.

This is likely because the actual magnitude of the receptor desolvation penalty was so much greater than the ligand desolvation penalty, due to the error in placing charge that was discussed previously. The fact that the receptor desolvation penalty tended to be high is most likely an artifact of the system. The arbitrarily defined "receptors" tended to be listed second in the original data file and tended to be smaller overall. Furthermore, in 80% of cases, charges were biased to be located towards the interface. This resulted in a higher probability that a charge would be placed inside a solvent-exposed atom, causing the higher magnitude matrix elements and thus desolvation penalties.

Model system and protein shapes

In order to compare the model system and the proteins, the features were scaled to one standard deviation and a mean of zero. When applying the coefficients from one system to the other, the features were scaled based on the first system. This likely had a large effect, because the magnitude of the model system features was much greater than those of the proteins. The model system is smaller, so more charges are closer to the interface, making the features greater in magnitude. We hypothesize that this greater magnitude of features in turn makes the magnitude of the coefficients smaller, leading to the results seen.

While the correlations of using the coefficients of one system to predict the others are reasonable, the error is very high. This is most likely due to the fact that the magnitude of the matrix elements is much higher in the protein-shaped systems. This result is both because protein charges are more solvent exposed because of the irregular geometry, and because of the occasional error in identifying solvent-exposed atoms. More features ought to be incorporated that better quantify the level of solvent exposure.

Future work

Future work will take many directions, including data acquisition, feature refinement, and additional regression techniques.

The first step that needs to be taken is the correct identification of solvent-exposed atoms. This work used a script that did identified most but not all solvent-exposed atoms. Because placement of charges depended on the identification of the atoms on the dielectric boundary, certain charges were placed too close to the edge, resulting in unphysical values for desolvation. This can be corrected in two ways: 1).

Using CHARMM[39] to identify solvent exposed atoms, or 2). Improving the accuracy of the script used in this work, possibly by adjusting the parameters used to define the boxes around each atom.

In the future, when calculating the Finite Difference Method standard, the number of grid points per dimension ought to be varied to ensure the same grids per angstrom across all training data. This work utilized the same number of grid points per dimension and approximately the same number of grids per angstrom, but since the absolute binding free energy is sensitive to the grid spacing, it is best to be consistent.

Furthermore, for ease of data analysis, fewer charges ought to be placed on proteins and more runs ought to be generated. This would help eliminate the term regression issue, which was that there were so many matrix elements and features, but not enough terms. Secondly, training on more diverse crystal structures would be optimal. This work utilized proteins that were previously prepared for a specificity-promiscuity study, and many of the partners were in common. Thirdly, training on the natural charge distributions of the proteins would allow us to see 1). How our protein model translates to natural systems, and 2). Test the hypothesis that it is most optimal to train and test on the same-shaped system, and 3). See how different the coefficients produced from the natural charge distributions are from those produced in this work.

In the future, features ought to be expanded and refined. One additional feature to which we have made a preliminary start is an additional geometry feature that aims to capture the degree of concavity and convexity around a charge. If a charge protrudes out into the solvent (ie, it is on a convex surface) and is close to the interface, it will pay a larger desolvation penalty than a charge that is located on a concave surface. This feature will draw a line from a charge to the nearest interfacial atom on the partner. Inner and outer cylinders using multiple radii can be drawn around this line, encircling all atoms on the protein that fall within that radius.

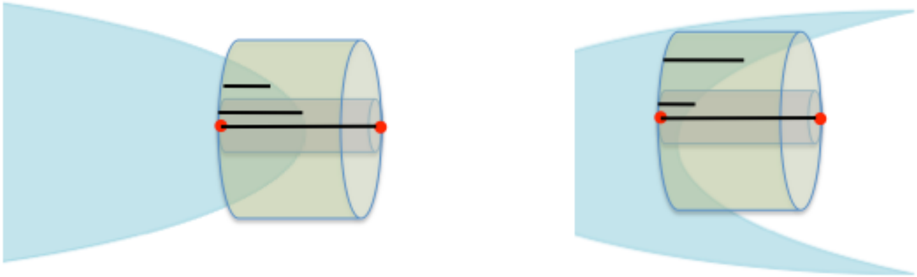

Figure 4.1. Convexity and concavity feature schematic. Red dots indicate charge; one on the protein (blue) and partner (white space). An inner cylinder (pink) and outer cylinder (green) circle atoms that fall within that cylinder on the protein (blue). The two black lines above the line connecting the charges represent the average horizontal projection onto that line of atoms in the cylinder. Left: a charge in a *convex* environment will have a shorter average "outer cylinder" horizontal component than the inner. Right: a charge in a *concave* environment will have a longer average "outer cylinder" horizontal component than the inner.

The term can be taken as either the difference in the number of atoms in each cylinder, or as the average horizontal projection onto the line. This feature will better quantify the local geometry around the charge. In the future, hopefully better features can be added that are both effective and computationally efficient.

As the number of features increases, additional regression techniques ought to be used to limit the number of features. This will help avoid overfitting, and allow determination of "important" features. LASSO, for example, is a technique that imposes an additional penalty on the error that forces coefficients to be zero. Using this technique will result in fewer features with similar error.

In this work, coefficients with a large magnitude were considered important. However, certain features may have a small but consistent effect on the prediction. To identify those variables, the confidence value of each feature ought to be examined, to see which features have a high degree of confidence.

Future work will compare the accuracy of our model to those of more established methods such as Surface-Generalized Born.[18] Both the accuracy and the efficacy of our model ought to be compared to other methods, as the goal of this project is to develop a accurate yet fast model to predict ΔG_{elec}.

Summary

In this work, a feature-based approach to estimate protein-protein electrostatic binding energetics was investigated. This work aims to replace a Poisson-equation numerical solver with a regression model that uses features to predict ΔG_{elec}. The results suggested that this may be a promising approach to estimate ΔG_{elec}, although work is ongoing to continue to improve the models for potential accuracy on actual protein-protein complexes.

REFERENCES

1. Zhang, Z.; Witham, S.; Alexov, E., On the role of electrostatics in protein–protein interactions. *Physical Biology* **2011,** *8* (3), 035001.

2. Sheinerman, F. B.; Norel, R.; Honig, B., Electrostatic aspects of protein-protein interactions. *Current Opinion in Structural Biology* **2000,** *10*, 6.

3. Teng, S.; Madej, T.; Panchenko, A.; Alexov, E., Modeling Effects of Human Single Nucleotide Polymorphisms on Protein-Protein Interactions. *Biophysical Journal* **2009,** *96* (6), 2178-2188.

4. Sheinerman, F. B.; Honig, B., On the Role of Electrostatic Interactions in the Design of Protein–Protein Interfaces. *Journal of Molecular Biology* **2002,** *318* (1), 161-177.

5. Moreira, I. S.; Fernandes, P. A.; Ramos, M. J., Hot spots-A review of the protein-protein interface determinant amino-acid residues. *Proteins: Structure, Function, and Bioinformatics* **2007,** *68* (4), 803-812.

6. Hu, Z.; Ma, B.; Wolfson, H.; Nussinov, R., Conservation of polar residues as hot spots at protein interfaces. *Proteins: Structure, Function, and Genetics* **2000,** *39*, 11.

7. Conte, L. L.; Chothia, C.; Janin, J., The atomic structure of protein-protein recognition sites. *Journal of Molecular Biology* **1999,** *285*, 22.

8. DeLano, W. L., Unraveling hot spots in binding interfaces: progress and challenges. *Current Opinion in Structural Biology* **2002,** *12*, 6.

9. James, L. C.; Tawfik, D. S., The specificity of cross-reactivity: Promiscuous antibody binding involves specific hydrogen bonds rather than nonspecific hydrophobic stickiness. *Protein Science* **2009,** *12* (10), 2183-2193.

10. Kortemme, T., A simple physical model for binding energy hot spots in protein-protein complexes. *Proceedings of the National Academy of Sciences* **2002,** *99* (22), 14116-14121.

11. Kumar, S.; Nussinov, R., Relationship between ion pair geometries and electrostatic strengths in proteins. *Biophysical Journal* **2002,** *83* (3), 1595-1612.

12. Kundrotas, P. J.; Alexov, E., Electrostatic Properties of Protein-Protein Complexes. *Biophysical Journal* **2006,** *91* (5), 1724-1736.

13. Joughin, B. A.; Green, D. F.; Tidor, B., Action-at-a-distance interactions enhance protein binding affinity. *Protein Science* **2005,** *14* (5), 1363-1369.

14. (a) Wang, J.; Hou, T.; Xu, X., Recent Advances in Free Energy Calculations with a Combination of Molecular Mechanics and Continuum Models. *Current Computer-Aided Drug Design* **2006,** *2*, 8; (b) Kollman, P. A.; Massova, I.; Reyes, C.; Kuhn, B.; Huo, S.; Chong, L.; Lee, M.; Lee, T.; Duan, Y.; Wang, W.; Donini, O.; Cieplak, P.; Srinivasan, J.; Case, D. A.; Cheatham, T. E., Calculating structures and free energies of complex molecules: Combining molecular mechanics and continuum models. *Accounts Chemical Research* **2000,** *33*, 8.

15. Gilson, M. K., Introduction to continuum electrostatics, with molecular applications. *self-published at <ce_www1a.pdf>* **2006**.

16. Radhakrishnan, M. L., Designing electrostatic interactions in biological systems via charge optimization or combinatorial approaches: insights and challenges with a continuum electrostatic framework. *Theoretical Chemistry Accounts* **2012,** *131* (8).

17. Huang, N.; Kalyanaraman, C.; Bernacki, K.; Jacobson, M. P., Molecular mechanics methods for predicting protein?ligand binding. *Physical Chemistry Chemical Physics* **2006,** *8* (44), 5166.

18. Romanov, A. N.; Jabin, S. N.; Martynov, Y. B.; Sulimov, A. V.; Grigoriev, F. V.; Sulimov, V. B., Surface Generalized Born Method- A Simple, Fast, and Precise Implicit Solvent Model beyond the Coulomb Approximation. *Journal of Physical Chemistry* **2004,** *108* (43), 5.

19. Still, W. C.; Tempczyk, A.; Hawley, R. C.; Hendrickson, T., Semianalytical Treatment of Solvation for Molecular Mechanics and Dynamics. *Journal of the American Chemical Society* **1990,** *112*, 2.

20. Ghosh, A.; Rapp, C. S.; Friesner, R. A., Generalized Born model based on a surface integral formulation. *Journal of Physical Chemistry B* **1998,** *102*, 7.

21. Grycuk, T., Deficiency of the Coulomb-field approximation in the generalized Born model: An improved formula for Born radii evaluation. *The Journal of Chemical Physics* **2003,** *119* (9), 4817.

22. Yang, S.-Y., Pharmacophore modeling and applications in drug discovery: challenges and recent advances. *Drug Discovery Today* **2010,** *15* (11-12), 444-450.

23. Wermuth, C. G.; Ganellin, C. R.; Lindberg, P.; Mitscher, L. A., Glossary of Terms Used in Medicinal Chemistry. *Pure and Applied Chemistry* **1998,** *70* (5), 14.

24. Dixon, S. L.; Smondyrev, A. M.; Knoll, E. H.; Rao, S. N.; Shaw, D. E.; Friesner, R. A., PHASE: a new engine for pharmacophore perception, 3D QSAR model development, and 3D database screening: 1. Methodology and preliminary results. *Journal of Computer-Aided Molecular Design* **2006,** *20* (10-11), 647-671.

25. Sanders, M. P. A.; McGuire, R.; Roumen, L.; de Esch, I. J. P.; de Vlieg, J.; Klomp, J. P. G.; de Graaf, C., From the protein's perspective: the benefits and challenges of protein structure-based pharmacophore modeling. *MedChemComm* **2012,** *3* (1), 28.

26. Warren, G. L.; Andrews, C. W.; Capelli, A. M.; Clarke, B.; LaLonde, J.; Peishoff, C. E.; Head, M. S., A Critical Assessment of Docking Programs and Scoring Functions. *Journal of Medicinal Chemistry* **2006,** *49*, 19.

27. Ballester, P. J.; Mitchell, J. B. O., A machine learning approach to predicting protein-ligand binding affinity with applications to molecular docking. *Bioinformatics* **2010,** *26* (9), 1169-1175.

28. Wallqvist; Covell, D. G., Docking enzyme-inhibitor complexes using a preference-based free-energy surface. *Proteins: Structure, Function, and Genetics* **1996,** *25*, 16.

29. Hastie, T. T., Robert; Friedman, Jerome, *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. 2nd ed.; Springer Science+Business Media, LLC: 2009.

30. Helland R, O. J., Sundheim O, Dadlez M, Smalås AO., The crystal structures of the complexes between bovine beta-trypsin and ten P1 variants of BPTI. *Journal of Molecular Biology* **1999,** *287* (5), 923-42.

31. Buckle, A. M., Schreiber, G., Fersht, A.R., Protein-protein recognition: crystal structural analysis of a barnase-barstar complex at 2.0-A resolution. *Biochemistry* **1994,** *33* (30), 8878-8889.

32. Valentine, K. G., Ng, H.L., Schneeweis, L., Kranz, J.K., Frederick, K.K., Alber, T., Wand, A.J., Crystal structure of calmodulin-neuronal nitric oxide synthase complex. **2006**.

33. Millers, E.-K. I., Lavin, M.F., de Jersey, J., Masci, P.P., Guddat, L.W., Crystal structure of Textilinin-1, a Kunitz-type serine protease inhibitor from the Australian Common Brown snake venom, in complex with trypsin. **2008**.

34. Wahlgren WY, P. G., Kardos J, Porrogi P, Szenthe B, Patthy A, Gráf L, Katona G., The catalytic aspartate is protonated in the Michaelis complex formed between trypsin and an in vitro evolved substrate-like inhibitor: a refined mechanism of serine protease action. *Journal of Biological Chemistry* **2011,** *286* (5), 3587-96.

35. Wall, M. E., Clarage, J.B., Phillips Jr., G.N., Motions of calmodulin characterized using both Bragg and diffuse X-ray scattering. *Structure* **1997,** *5*, 1599-1612.

36. Scheidig, A. J., Hynes, T.R., Pelletier, L.A., Wells, J.A., Kossiakoff, A.A., Crystal structures of bovine chymotrypsin and trypsin complexed to the inhibitor domain of Alzheimer's amyloid beta-protein precursor (APPI) and basic pancreatic trypsin inhibitor (BPTI): engineering of inhibitors with altered specificities. *Protein Science* **1997,** *6* (9), 1806-24.

37. Maximciuc, A. A., Putkey, J.A., Shamoo, Y., Mackenzie, K.R., Complex of calmodulin with a ryanodine receptor target reveals a novel, flexible binding mode. *Structure* **2006,** *14*, 1547-1556.

38. Fallon, J. L., Halling, D.B., Hamilton, S.L., Quiocho, F.A., Structure of calmodulin bound to the hydrophobic IQ domain of the cardiac Ca(v)1.2 calcium channel. *Structure* **2005,** *13*, 1881-1886.

39. Brooks BR, B. C., Mackerell AD Jr, Nilsson L, Petrella RJ, Roux B, Won Y, Archontis G, Bartels C, Boresch S, Caflisch A, Caves L, Cui Q, Dinner AR, Feig M, Fischer S, Gao J, Hodoscek M, Im W, Kuczera K, Lazaridis T, Ma J, Ovchinnikov V, Paci E, Pastor RW, Post CB, Pu JZ, Schaefer M, Tidor B, Venable RM, Woodcock HL, Wu X, Yang W, York DM, Karplus M, CHARMM: The biomolecular simulation program. *Journal of Computational Chemistry* **2009,** *30* (10), 1545-1614.

40. Altman, M. D., Computational Ligand Design and Analysis in Protein Complexes Us- ing Inverse Methods, Combinatorial Search, and Accurate Solvation Modeling. **2006**.

41. Baker, N. A., Improving implicit solvent simulations: a Poisson-centric view. *Current Opinion in Structural Biology* **2005,** *15* (2), 137-143.

42. Team, R. D. C. *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing: Vienna, Austria, 2008.