

# Supporting Information (S1 Text)

Jumper et al.

## Derivation of contrastive divergence

We derive the contrastive divergence method as a series of approximations to the problem of best approximating the probability distribution of observed PDB structures using a force field of an imperfect, fixed form. The initial part is a standard derivation of the maximum likelihood method, adapted to make clear its connection to protein molecular dynamics, while the end of the derivation makes clear the relaxation to obtain contrastive divergence as an approximation to maximum likelihood.

We begin by assuming that we have a large collection of protein sequences  $\{s_a\}$  and their associated Boltzmann distributions  $p_{s_a}^{\text{true}}(X_a)$  under physiological conditions, where  $a$  represents an arbitrary label to enumerate the proteins and  $X_a$  represents the configuration of the protein. Note that the “true” Boltzmann distribution is an unobservable idealization of the conformational ensemble of a protein under physiological conditions, and we further idealize that the true Boltzmann distribution is derived from from an extremely-complicated true potential  $V_{s_a}^{\text{true}}$  by statistical mechanics,

$$p_{s_a}^{\text{true}}(X_a) = \frac{\exp(-V_{s_a}^{\text{true}}(X_a))}{\exp(-G_{s_a}^{\text{true}})} \quad (1)$$

$$G_{s_a}^{\text{true}} = -\log \int e^{-V_{s_a}^{\text{true}}(X)} dX. \quad (2)$$

The subscript  $s_a$  indicates that both the potential  $V_{s_a}^{\text{true}}$  and free energy  $G_{s_a}^{\text{true}}$  depend on the sequence of the protein. We may think of this as an artifact of working in the coarse-grained coordinates of the backbone trace, where the energy  $V_{s_a}^{\text{true}}$  really represents the free energy of the backbone coordinates after integrating away the solvent and side chain degrees of freedom. An analogous situation occurs in parameterizing all-atom molecular dynamics, where the “energy” of the system really represents the free energy of the system after integrating over the electronic degrees of freedom. Our goal is to define a parametric  $V_s^{\text{approx}}(X)$  that approximates the  $V_s^{\text{true}}$  for any sequence  $s$ . We drop the subscript  $s$  below where there is no possibility for confusion.

For an approximate potential  $V^{\text{approx}}$ , it is almost certain that  $V^{\text{approx}}$  does not have enough flexibility in its functional form to match all of the Boltzmann distributions  $p_a$  for any sequence  $s_a$ . We instead find a  $V^{\text{approx}}$  that is “close” to  $V^{\text{approx}}$ . Defining the Boltzmann distribution of  $V^{\text{approx}}$  in the same manner as that of  $V^{\text{true}}$ ,

$$p_{s_a}^{\text{approx}}(X_a) = \frac{\exp(-V_{s_a}^{\text{approx}}(X_a))}{\exp(-G_{s_a}^{\text{approx}})} \quad (3)$$

$$G_{s_a}^{\text{approx}} = -\log \int e^{-V_{s_a}^{\text{approx}}(X)} dX, \quad (4)$$

we may use the Kullback-Leibler (KL) divergence to measure the similarity of the associated Boltzmann

distributions,

$$\begin{aligned}
& \text{KL}(p^{\text{true}}, p^{\text{approx}}) \\
&= \int p^{\text{true}}(X) \log \frac{p^{\text{true}}(X)}{p^{\text{approx}}(X)} dX \\
&= \langle -\log p^{\text{approx}}(X) + \log p^{\text{true}}(X) \rangle_{\text{true}} \\
&= \langle (V^{\text{approx}}(X) - G^{\text{approx}}) - \\
&\quad (V^{\text{true}}(X) - G^{\text{true}}) \rangle_{\text{true}}.
\end{aligned} \tag{5}$$

In the last equation, we note that the KL divergence is simply the average energy difference between the true and approximate potentials (after subtracting the free energies to normalize the probabilities), where the average is taken over the true Boltzmann distribution. The key fact when minimizing KL divergence is that if the approximate distribution lacks the freedom to exactly match the true distribution, then the minimizing distribution will be weaker than the true distribution (i.e. less sharp) to avoid assigning highly unfavorable energy to configurations that are likely in the true distribution.

Dropping constant terms, we may instead minimize

$$\langle V^{\text{approx}}(X) - G^{\text{approx}} \rangle_{\text{true}}, \tag{6}$$

since the remaining term  $\langle V^{\text{true}}(X_a) - G^{\text{true}} \rangle_{\text{true}}$  is independent of the approximating potential. This expectation value is still intractable since we do not know  $p^{\text{true}}$ , but we can approximate,

$$p^{\text{true}}(X) \approx p^{\text{empirical}}(X) = \frac{1}{M} \sum_{a=1}^M \delta(X - X_a), \tag{7}$$

where  $\delta$  is the Dirac delta function and  $M$  is the number of proteins. This gives the objection function,

$$\begin{aligned}
& \langle V^{\text{approx}}(X) - G^{\text{approx}} \rangle_{\text{empirical}} \\
&= \frac{1}{M} \sum_{a=1}^M \int \delta(X - X_a) (V^{\text{approx}}(X) - G^{\text{approx}}) dX \\
&= \frac{1}{M} \sum_{a=1}^M (V^{\text{approx}}(X_a) - G^{\text{approx}}).
\end{aligned} \tag{8}$$

Minimizing the expression Eq 8 is exactly the method of maximum likelihood. The derivation given above illustrates two points via the connection to KL divergences. The first is that, if  $V^{\text{approx}}$  is insufficiently detailed, the model's ensemble will be overly broad to ensure no experimental conformation has high energy under  $V^{\text{approx}}$ . The second point is that with only a finite number of samples,  $p^{\text{empirical}}$  may be a poor approximation to  $p^{\text{true}}$ , which would allow  $V^{\text{approx}}$  to wrap itself tightly near the  $\delta$ -functions associated with each sample. This is the origin of overfitting in maximum-likelihood models.

We can now take the derivative with respect to an arbitrary force field parameter  $\alpha_i$  in preparation to perform gradient descent on Eq 8. The gradient is given by

$$\begin{aligned}
& \frac{d}{d\alpha_i} \frac{1}{M} \sum_{a=1}^M (V^{\text{approx}}(X_a) - G^{\text{approx}}) \\
&= \frac{1}{M} \sum_{a=1}^M \left( \frac{dV^{\text{approx}}}{d\alpha_i}(X_a) - \frac{dG^{\text{approx}}}{d\alpha_i} \right) \\
&= \frac{1}{M} \sum_{a=1}^M \left( \frac{dV^{\text{approx}}}{d\alpha_i}(X_a) - \left\langle \frac{dV^{\text{approx}}}{d\alpha_i}(X) \right\rangle_{\text{approx}} \right),
\end{aligned} \tag{9}$$

where we have used the standard statistical mechanics identity  $dG/d\alpha_i = \langle dV/d\alpha_i \rangle$ . While we have obtained a concrete expression for gradient descent in Eq 9, we still have a major stumbling block. Computing the

expectation of the derivative of the potential at  $X_a$  is straightforward given a functional form for  $V^{\text{approx}}$ , but obtaining even a reliable approximation for  $\left\langle \frac{dV^{\text{approx}}}{d\alpha_i}(X) \right\rangle_{\text{approx}}$  is extraordinarily difficult. To approximate the expectation value on a rough protein energy landscape, we would need Boltzmann samples from our current approximating potential. Even obtaining the single most likely configuration for our approximating potential is equivalent to finding the native state of the model, and this is very difficult for realistic pairwise potentials. Instead, we require the Boltzmann ensemble for all the proteins in our training set, and must update those Boltzmann ensembles as we use gradient descent to optimize the approximating potential. This represents an extreme expense and is unrealistic for anything but the simplest models of proteins. Note also that we cannot simply construct a large list of structures at some time and reweight those structures according to the potential, since the potential is constantly changing. Reweighting ensembles is only valid over very small neighborhoods of parameter space, and this procedure would depend on being able to generate an exhaustive survey of candidate structures in an exponentially large space.

The contrastive divergence method [1] approximates the maximum likelihood procedure using an empirical observation. We do not need an accurate approximation to Eq 9, so long as the derivative points in direction of parameter space that improves the potential accuracy (i.e. any direction is acceptable as long as it is not uphill). Hinton proposes replacing Boltzmann average  $\left\langle \frac{dV^{\text{approx}}}{d\alpha_i}(X) \right\rangle_{\text{approx}}$  with a finite-time Fokker-Planck average over a very short period of time *for a simulation that originates at the data point  $X_a$* . In the Monte Carlo (MC) dynamics that the original authors use, even one MC step is sufficient to produce acceptable optimization of their model. In our case, we replace their small number of MC steps with a short time simulation using replica exchange Langevin dynamics. As the duration of the simulation is increased, our derivative estimate will converge to the true derivative Eq 9. Our paper empirically demonstrates that equilibrating each model within only a local region around the crystallographic native state is sufficient for a good folding model, so long as a large and diverse collection of protein structures are jointly optimized.

## Optimization and simulation details

The following temperature ranges are used for the replica exchange simulations with 14-16 replicas per simulation. These temperatures are chosen to use the minimal temperature range that approximately span the thermal melting transition for each protein using information from an earlier set of replica exchange simulations. The temperatures of the simulation initialized from the crystal structure and those initialized from extended structures use the same temperature range.

The force is integrated using Verlet integration with a time step of 0.009 time units. Temperature is maintained using a Langevin thermostat with a thermalization timescale of 0.135 time units.

Simulation times in all figures are given in millions of *Upside* time units (approximately  $10^8$  force evaluations).

## Handling crystallographic artifacts

The derivation of contrastive divergence makes the assumption that the conformations  $X_a$  are equilibrium samples from the Boltzmann distribution of each protein, but in reality, we must work with crystal structures of proteins. It has been shown that the static diversity of crystal structures for different proteins conveys significant information about the dynamic ensembles of individual proteins [3]. Crystal structures deviate in a number of systematic ways from equilibrium samples, but we are most concerned about crystal packing artifacts, crystallizability bias, and errors in published structures.

We expect that our bias in working only with crystallizable sequences, thus missing intrinsically disordered regions from training, likely biases the resulting potential to disfavor coil states. The loop-stabilizing effects of crystal packing somewhat counteract this effect, as it allows longer loop regions to exist in crystal structures.

## Training data and optimization

The contrastive divergence training is conducted with 456 crystal structures from the Protein Data Bank. The initial selection of structures uses the PISCES server [4] to select proteins with X-ray resolution less than 2.2 Å and pairwise sequence similarity less than 30%. In structures with multiple chains, a single chain is

Table A: Temperatures for replica exchange

Protein	$T_{\min}$	$T_{\max}$
alpha3d	0.972	1.123
BBA	0.757	1.499
BBL	0.844	1.202
cspA	0.925	0.975
gpW	0.850	1.000
homeodomain	0.978	1.153
hyp	1.000	1.050
lambda	0.879	1.172
NTL9	0.776	1.139
NuG2 (Shaw)	0.975	1.050
protein B	0.885	1.172
protein G	0.776	1.295
protein L	0.883	1.109
top7	1.090	1.130
ubiquitin	0.951	1.060
WW domain	0.704	1.295
T0765	0.970	1.000
T0769	0.940	0.980
T0771	0.975	1.050
T0773	0.850	0.980
T0803	0.920	1.020
T0816	0.980	1.020
T0855	0.980	1.050

chosen by the PISCES server. To avoid non-globular proteins or proteins with strong interactions with other subunits in the structure, random sample consensus linear regression [5] is used to identify outliers based on the relationship between  $\log N_{\text{res}}$  and  $\log R_g$ . Only chains with between 50 and 100 residues are used to encourage fast relaxation during the contrastive divergence simulations. All proteins homologous to proteins in the benchmark folding set are eliminated from the training set. Additionally, all proteins with backbone gaps, either missing residues due to diffuse electron density or non-standard amino acids that *Upside* does not handle, are also excluded from the training set.

The final training set of 456 proteins is divided into 38 groups of 12 proteins each, called minibatches. The Adam optimizer, a popular machine learning algorithm [6] is used to perform gradient descent on the objective function, using the contrastive divergence pseudo-gradient in place of the true maximum likelihood gradient. The Adam parameters used are  $\beta_1 = 0.8$ ,  $\beta_2 = 0.96$  and  $\epsilon = 10^{-6}$ . The  $\alpha$  parameter is varied based on the type of term to ensure stability,  $\alpha_{\text{SC}} = 0.5$ ,  $\alpha_{\text{env}} = 0.1$ ,  $\alpha_{\text{HBond}} = 0.02$ , and  $\alpha_{\text{sheet}} = 0.03$ . The  $\alpha$  parameters are multiplied by 0.25 for the fine-tuning optimization.

Regularization and derivative propagation for contrastive divergence optimization are handled using the Theano python library for automatic differentiation [7].

## Burial definition

To handle the uncertainty of rotameric states that can affect the count, the count for different rotameric states are weighted by the prior probabilities of the rotamer states, given by

$$N_i = \sum_{\substack{j \\ |i-j|>2}} \sum_{\chi_i} p(\chi_i) S(|y_i(\chi_i) - y_i^{C_\beta}| - (8 \text{ \AA}), (1 \text{ \AA})) S(\text{angle}(y_i(\chi_i) - y_i^{C_\beta}, d_i^{C_\beta}) + 0.1, 1), \quad (10)$$

where  $S$  is sigmoid-like cutoff function,  $y^{C_\beta}$  is the position of the  $C_\beta$ , and  $d^{C_\beta}$  is the  $C_\alpha$ - $C_\beta$  bond direction.

Table B: Protein sequences and structures

Name	PDB ID	Length	Sequence
alpha3d	2a3d	73	MGSWAEFKQRLAAIKTRLQALGGSEAELAAFEKEIAA FESELQAYKKGKNPEVEALRKEAAAIRDELQAYRHN
BBA	1fme	28	EQYTAKYKGRTRNEKELRDFIEKFKGR
BBL	2wxc	47	GSQNNDALSPAIRLLAEWNLDASAIKGTGVGGRLTREDVEKHLAKA
cspA	1mjc	69	SGKMTGIVKWFNADKGFGITPDDGSKDVFVHFSAIQND GYKSLDEGQKVSFTIESGAKGPAAGNVTSL
gpW	2l6q	62	MVRQEELAAARAALHDLMTGKRVATVQKDGRRVEFTATSV SDLKKYIAELEVQTMGTQRRRG
homeodomain	2p6j	52	MKQWSENVEEKLKEFVKRHRITQEELHQYQA RLGLNEEAIRQFFEEFEQRK
hyp	1whz	70	MWMPPRPEEVARKLRLRGFVERMAKGGHRLYTHPDGRIVV VPFHSGELPKGTFKRILRDAGLTEEEFHNL
lambda	1lmb	80	PLTQEQLDARRLKAIEYKKNELGLSQESVADKMGMGQS GVGALFNGINALNAYNAALLAKILKVSVEEFSPSIAREIY
NTL9	2hba	39	MKVIFLKDVKGMGKKGEIKNVADGYANNFLFKQGLAIEA
NuG2 (Shaw)	1mi0	57	MDTYKLVIVLNGTTFYTTTEAVDAATAEKVFKQY ANDAGVDGEWTYDAATKTFTVTE
protein B	1prb	53	TIDQWLLKNAKEDAIAELKKAGITSDFYFNAINKAKTVEEVNALKNEILKAHA
protein G	1pga	56	MTYKLILNGKTLKGETTTEAVDAAT AEKVFKQYANDNGVDGEWTYDDATKTFTVTE
protein L	2ptl	61	VTIKANLIFANGSTQTAEFKGTFEKATSE AYAYADTLKKNNGEYTVDVADKGYTLNIFAG
top7	1qys	92	DIQVQVNIDDNGKNFDYTYTVTTESELQKVLNELMDYIKKQGAKRV RISITARTKKEAEKFAAILIKVFAELGYNDINVTFDGDTVTEGQL
ubiquitin	1ubq	76	MQIFVKTLTGKTITLEVEPSDTIENVKAKIQDKEGIPP DQQLIFAGKQLEDGRTLSDYNIQESTLHLVLRRLRG
WW domain	2f2l	33	KLPPGWEKRMSADGRVYFNFHITNASQWERPSG
T0765		76	QQQEATLAIRPVGGIGMPDGFVWHHLANGIRFKSIT PQKDGLLIKFDSTAQGAAAKEVLGRALPHGYIIALLE
T0769		112	MLTVEVEVKITADDENKAEIIVKRVIDEVEREVQKQYP NATITRTLRRDDGTVELRIKVKADTEEKAKSIIKLIEE RIEEELRKRDPNATITRTVRTEVGSSWSLEHHHHHH
T0771		178	TEESLEGTVIYKKTTFEVDGYTYQCDVDDGSQFVTLYNK ENKLTYEKIVYKDTGKTYIGSWSSNVIEYDRFMSQQADFI VDQAFTKAMADEIGKTELMITMLLSPNTGEVMEVNFNFFT EPYAKVPLHVYREIEVKLKEQIHFKPIEEGKQLNYIMLA WMQKPQGKLPLPPPGSL
T0773		77	MVDLKIDVSDDEEAEKIIREIREQWPKATVTRTNGDIKLD AQTEKEAEKMEKAVKVKPNATIRKTGGSLEHHHHHHH
T0803		134	SNINKAKVASVESDYSSIKSAALSYSNTNIPVTPDGQT GLNVLETYMESLPDKADIGGEYKLIKVGNKLVLQIGKDGE GVTLTEAQSAKLLSDIGKDKIYTGVTGDNFGEQLKDTTKI DNKALYIVLIDNTV
T0816		68	MITYKLLDELKKEIGPIAKIFLNKAMESLGYDDVDDSNYK EILSVLKMKNKELREYVEIVEERLEKEG
T0855		115	FTDQQIGVLAGLAISPEWLKQNIANQLVYGIVKPSDTPV AGVDDYSYLVAAADDQDGTIIFKAEGQTVIHKYTSQRNTK LKAKALTLSQLKKEFYQTRSQKREVDDYVAGLRTE

Mutations from the listed PDB structures are indicated in red. The NuG2 sequence is from reference [2].

## Other potential terms

The hydrogen bonding interaction is computed from the distance between hydrogen and oxygen  $r_{\text{HO}}$ , the separation vector  $d_{\text{HO}}$  as well as the hydrogen and oxygen bond vectors  $d_{\text{H}}$  and  $d_{\text{O}}$ , respectively. The strength of a single interaction is given by

$$h = f(r_{\text{HO}})g(d_{\text{HO}} \cdot d_{\text{O}})g(-d_{\text{HO}} \cdot d_{\text{H}}) \quad (11)$$

$$f(r) = \text{sigmoid}\left(\frac{r - 1.4 \text{ \AA}}{0.1 \text{ \AA}}\right) \text{sigmoid}\left(\frac{2.5 \text{ \AA} - r}{0.125 \text{ \AA}}\right) \quad (12)$$

$$g(c) = \text{sigmoid}\left(\frac{c - 0.682}{0.05 \text{ \AA}}\right). \quad (13)$$

The interactions are combined to ensure that each hydrogen bond can have at most one full partner using

$$V_{\text{hbond}} = E_{\text{hbond}} \sum_i \left( 2 - \prod_{j \neq i} (1 - h_{ij}) - \prod_{j \neq i} (1 - h_{ji}) \right), \quad (14)$$

which is inspired by treating each  $h_{ij}$  as the probability that residue  $i$  donates a hydrogen bond to residue  $j$ . The energy is then linear in the probability has at least one hydrogen bond donor and linear in the probability it has at least one hydrogen bond acceptor. The parameter  $E_{\text{hbond}}$  is learned by contrastive divergence.

The backbone steric interactions are imposed with approximately 3 Å soft sphere repulsion among the N, C $^\alpha$ , C, and C $^\beta$  (placed) atoms.

## References

1. Hinton GE. Training products of experts by minimizing contrastive divergence. *Neural computation*. 2002;14(8):1771–1800.
2. Lindorff-Larsen K, Piana S, Dror RO, Shaw DE. How fast-folding proteins fold. *Science*. 2011;334(6055):517–520.
3. Jha AK, Colubri A, Freed KF, Sosnick TR. Statistical coil model of the unfolded state: resolving the reconciliation problem. *Proceedings of the National Academy of Sciences of the United States of America*. 2005;102(37):13099–13104.
4. Wang G, Dunbrack RL. PISCES: a protein sequence culling server. *Bioinformatics*. 2003;19(12):1589–1591.
5. Fischler MA, Bolles RC. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*. 1981;24(6):381–395.
6. Kingma D, Ba J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*. 2014;.
7. Theano Development Team. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*. 2016;abs/1605.02688.