SI Text

1) Data preprocessing.

We downloaded the amino acid multiple sequence alignment (MSA) of HIV-1 Clade B gp160 sequences from the Los Alamos National Laboratory (LANL) HIV sequence database (www.hiv.lanl.gov; accessed 14th May, 2016). To control sequence quality, we excluded sequences (i) labeled by LANL as 'problematic', (ii) with ≥ 2% gaps at non hyper-variable residues (https://www.hiv.lanl.gov/content/sequence/VAR_REG_CHAR/variable_region_characterization_explanation.html), (iii) with > 10 consecutive insertions at residues where there is an amino acid at every other sequence or (iv) which are outliers as determined by principal component analysis (PCA) (as described in (1)). We then removed sequences labeled or predicted to be CXCR4-tropic, where the prediction was performed using (2). A total of 20043 sequences remained after excluding the above sequences, with these sequences belonging to a total of 1918 patients.

To control residue quality, we excluded residues which are (i) 100% conserved, (ii) contain >90% gaps or are ambiguous, or (iii) are in the hyper-variable regions due to the poor quality of the alignment. We then included eight additional "artificial" residues whose states reflect the number of residues and N-linked glycans in the four hyper-variable regions we have excluded. This resulted in a total of L=815 residues. The ambiguous residues were then imputed with a simple mean imputation. The MSA can be represented by a matrix $\mathbf{X}$ where the lth row is a vector of amino acids $\mathbf{x}_\ell = [x_{\ell 1}, \ldots, x_{\ell L}]$ representing the lth sequence, where the amino acids at residue i are identified as either consensus ($x_{\ell i} = 0$) or the kth dominant (most frequent) mutant ($x_{\ell i} = k$), for $k=1,..,\ q_i$ where q_i denotes the number of mutants at residue i.

2) Computational model and framework

2.1 Maximum entropy model. We consider a probabilistic model designed to reproduce the single and double mutant probabilities

$$f_i(a) = \frac{1}{P}\sum_{k=1}^{B} w_k \delta(x_{ki}, a) \qquad \text{(S1)}$$

$$f_{ij}(a,b) = \frac{1}{P}\sum_{k=1}^{B} w_k \delta(x_{ki}, a)\delta(x_{kj}, b)$$

where $f_i(a)$ denotes the frequency of amino acid a at residue i and $f_{ij}(a,b)$ the joint frequency of amino acids a and b at residues i and j respectively, as observed from the MSA $\mathbf{X}$. Also, B is the number of sequences after data preprocessing, δ is the Kronecker delta function

$$\delta(a,b) = \begin{cases} 0 & \text{if } a \neq b \\ 1 & \text{if } a = b \end{cases}, \qquad \text{(S2)}$$

P=1918 is the number of patients, and w_k is one divided by the number of MSA sequences contributed by the patient from which sequence k was extracted. The weights w_k ensure that the different number of sequences extracted per patient do not bias the calculation of the single and double mutant probabilities.

A vast family of probabilistic models can reproduce the single and double mutant probabilities in Eq. S1. Among such models, we consider the 'least biased' model that maximizes the entropy of the sequence distribution. For a given sequence $\mathbf{x} = [x_1, x_2, \ldots, x_L]$, this maximum entropy model assigns the probability

$$p_{\mathbf{h},\mathbf{J}}\left(\mathbf{x}\right)=\frac{\exp\left(-E_{\mathbf{h},\mathbf{J}}\left(\mathbf{x}\right)\right)}{Z} \tag{S3}$$

$$E_{\mathbf{h},\mathbf{J}}\left(\mathbf{x}\right)=\sum_{i=1}^{L}h_i\left(x_i\right)+\sum_{i=1}^{L}\sum_{j=i+1}^{L}J_{ij}\left(x_i,x_j\right) \tag{S4}$$

where **h** denotes the set of all fields, **J** denotes the set of all couplings, Z is the partition function (a normalization ensuring the distribution has total mass of unity), while $E(\mathbf{x})$ is the "energy" of strain **x**. The field $h_i(x_i)$ and coupling $J_{ij}(x_i, x_j)$ parameters are chosen such that the single and double mutant probabilities of the model match those in the data, i.e.,

$$f_i\left(a\right)=\sum_{\mathbf{x}}\delta\left(x_i,a\right)\ p_{\mathbf{h},\mathbf{J}}\left(\mathbf{x}\right) \tag{S5}$$

$$f_{ij}\left(a,b\right)=\sum_{\mathbf{x}}\delta\left(x_i,a\right)\delta\left(x_j,b\right)\ p_{\mathbf{h},\mathbf{J}}\left(\mathbf{x}\right).$$

This choice can be formulated as the following convex optimization problem (see e.g (3))

$$\left(\mathbf{h}^*,\mathbf{J}^*\right)=\arg\min_{\mathbf{h},\mathbf{J}}\mathrm{KL}\left(p_0\parallel p_{\mathbf{h},\mathbf{J}}\right) \tag{S6}$$

where

$$p_0\left(\mathbf{x}_k\right)=\frac{1}{P}\sum_{k=1}^{B}w_k\,p_{\mathbf{X}}\left(\mathbf{x}_k\right)$$

with $p_{\mathbf{X}}\left(\mathbf{x}_k\right)$ denoting the frequency of observing strain $\mathbf{x}_k$, (i.e., the fraction of rows in $\mathbf{X}$ corresponding to $\mathbf{x}_k$) and

$$\mathrm{KL}\left(p_0\parallel p_{\mathbf{h},\mathbf{J}}\right)=\sum_{\mathbf{x}}p_0\left(\mathbf{x}\right)\ \ln\ \frac{p_0\left(\mathbf{x}\right)}{p_{\mathbf{h},\mathbf{J}}\left(\mathbf{x}\right)}\ . \tag{S7}$$

This represents the Kullback-Leibler divergence between p_0 and $p_{\mathbf{h},\mathbf{J}}$.

Although the optimization problem in Eq. S6 is convex and thus can be solved by BML algorithms, the number of mutants and length of HIV proteins can be large (Fig. 1) leading to two related computational issues. First, it yields a very large number of parameters, resulting in a slow convergence. Specifically, the number of fields to be estimated is $\sum_{i=1}^{L}q_i$ while the number of couplings is $\sum_{i=1}^{L}q_i\sum_{j=i+1}^{L}q_j$. Thus the total number of parameters is given by $\sum_{i=1}^{L}\left(q_i+1\right)\sum_{j=i+1}^{L}q_j$, which can be very large for HIV, as shown in Fig. 1. This is most extreme for gp160, which has about an order of magnitude more parameters than every other protein.

Secondly, calculating the gradient required by BML algorithms can be computationally prohibitive due to the large number of terms in the partition function. Specifically, the partition involves $\prod_{i=1}^{L}\left(q_i+1\right)$ terms, which increases exponentially with the number of residues L. This makes exact computation intractable, and therefore necessitates the use of MCMC simulations to approximate the gradient. For gp160 however, proper initialization of the algorithm is essential otherwise the running time is prohibitive. To alleviate these two problems, we first reduce the number of mutants that are modeled explicitly, then apply a method to find an accurate estimate of the parameters used to initialize the BML algorithm.

2.2 Reducing the number of mutants that are explicitly modeled. To limit overfitting and to reduce the number of mutants that we consider per residue, we group the low-frequency mutants together. Specifically, we model only the top k_i most frequent mutants, whilst grouping the remaining $q_i - k_i$ mutants, such that the corresponding entropy with grouping achieves a certain fraction ϕ of the entropy without grouping (4). Mathematically, for residue *i*, this involves choosing the smallest integer k_i such that

$$S_i(k_i) \geq \phi S_i(q_i) \tag{S8}$$

where

$$S_i(k_i) = -\sum_{a=0}^{k_i} f_i(a) \ln f_i(a) - \overline{f}_i \ln \overline{f}_i \tag{S9}$$

and

$$\overline{f}_i = \sum_{a=k_i+1}^{q_i} f_i(a) \quad . \tag{S10}$$

To choose ϕ, we introduce the measure

$$\beta_i(\phi) = \frac{\sum_{a=1}^{q_i} \left(f_i(a) - \overline{f}_i(a) \right)^2}{\sum_{a=1}^{q_i} \frac{f_i(a)\left(1 - f_i(a)\right)}{P}} \tag{S11}$$

where

$$\overline{f}_i(a) = \begin{cases} f_i(a) & \text{if } a < k_i + 1 \\ \overline{f}_i & \text{if } a = k_i + 1 \\ 0 & \text{if } a > k_i + 1 \end{cases}$$

and we recall that *P* is the number of patients. The conceptual idea behind Eq. S11 is to introduce as much bias due to grouping (numerator) as possible within the limits afforded by the fluctuations in the amino acid frequencies (denominator). By noting that k_i is a function of ϕ, to achieve a balance between overfitting and underfitting, we choose ϕ such that the mean value of $\beta_i(\phi)$, taken over all residues *i=1,.,.L*, is approximately one. This selection criteria leads to the choice of $\phi^* = 0.95$ (Table S2), leading to a significant reduction in the number of model parameters. For this choice of ϕ, for the *i*th residue, we denote $\tilde{q}_i = k_i + 1$ as the modified number of mutants after combining, resulting in a new MSA matrix $\tilde{\mathbf{X}}$. This method, which we later validate for gp160 (Table S3), results in a six-fold reduction in the number of parameters.

2.3 Efficiently obtaining accurate initialization parameters through MPF. To obtain an efficient and accurate initialization for BML, we apply the principle of minimum probability flow (MPF). MPF was originally designed for the Ising model (5) which has two states per residue. Here, we consider however a Potts generalization, allowing for an arbitrary number of states per residue. To this end, we construct a binary MSA based on the amino acid MSA. Specifically, each amino acid at the *i*th residue is represented by $\tilde{q}_i$ binary digits. The *j*th most frequent amino acid is then represented by the $\tilde{q}_i$-bit binary representation of 2^{j-1}, and thus the consensus sequence is the all-zero vector. For example, if $\tilde{q}_i = 2$, then the consensus amino acid, most common and least common mutant are denoted respectively as 00, 01 and 10. We will denote $\mathbf{Y}$ as the binary matrix corresponding to the amino acid matrix $\tilde{\mathbf{X}}$, with *i*th row denoted by $\mathbf{y}_i$.

The key problem to solving Eq. S6 is that $p_{\mathbf{h},\mathbf{J}}$ contains the intractable partition function *Z*. The goal of MPF is to replace

$p_{\mathbf{h},\mathbf{J}}$ with an alternate probability mass function (PMF) which is also parametrized by the fields and couplings, but which results in a tractable objective problem (to obtain a low-complexity solution), and at the same time retain similar 'features' as $p_{\mathbf{h},\mathbf{J}}$ (to obtain an accurate solution). To simultaneously achieve these goals, a continuous-time Markov chain is considered whose states correspond to the $M = \prod_{i=1}^{L} (\tilde{q}_i + 1)$ possible sequences. The probability of each state at time zero is set to the empirical PMF p_0, and the master equation describing this Markov chain is given by

$$\frac{\mathrm{d}}{\mathrm{d}t} p_{\mathbf{h},\mathbf{J};t}(\mathbf{y}_i) = \sum_{j=1, j\neq i}^{M} \Gamma_{ij}\, p_{\mathbf{h},\mathbf{J};t}(\mathbf{y}_j) - \sum_{j=1, j\neq i}^{M} \Gamma_{ji}\, p_{\mathbf{h},\mathbf{J};t}(\mathbf{y}_i) \tag{S12}$$

where $p_{\mathbf{h},\mathbf{J};t}(\mathbf{y}_i)$ denotes the probability of $\mathbf{y}_i$ at time t, and with $p_{\mathbf{h},\mathbf{J};t} = p_0$. The solution is given by

$$p_{\mathbf{h},\mathbf{J};t}(\mathbf{y}_i) = \left[\exp(t\boldsymbol{\Gamma})\mathbf{p}_0\right]_i \tag{S13}$$

where $[\mathbf{y}]_i$ denotes the ith element of the vector $\mathbf{y}$. The matrix $\boldsymbol{\Gamma}$ is the $M \times M$ transition rate matrix with (i,j)th element Γ_{ij} designed such that

$$\lim_{t\to\infty} p_{\mathbf{h},\mathbf{J};t}(\mathbf{y}_i) = p_{\mathbf{h},\mathbf{J}}(\mathbf{y}_i) \tag{S14}$$

with details of this matrix given in (5). Thus for any given $\mathbf{h}$ and $\mathbf{J}$, the transition rate matrix is designed to ensure that as time increases, the PMF 'evolves' towards $p_{\mathbf{h},\mathbf{J}}$.

The next step is to choose a t which can result in a tractable optimization problem. MPF replaces $p_{\mathbf{h},\mathbf{J}}$ in Eq. S6 by $p_{\mathbf{h},\mathbf{J};t}$ with t small. This choice has the advantage that the resulting optimization problem is convex and easy to solve. The resulting KL divergence between p_0 and $p_{\mathbf{h},\mathbf{J};t}$ expanded as a Taylor series around $t=0$ results in the linear approximation:

$$\mathrm{KL}\left(p_0 \,\|\, p_{\mathbf{h},\mathbf{J};t}\right) = t\mathrm{K}_{\mathbf{h},\mathbf{J}} + o(t) \tag{S15}$$

where

$$\mathrm{K}_{\mathbf{h},\mathbf{J}} = \sum_{b=1}^{B}\sum_{i=1}^{L}\sum_{a=1}^{q_i} \mathrm{e}\,\mathrm{xp}\left(\frac{1}{2}\left(\left(2y_{b,(i-1)L+a} - 1\right)\sum_{j=1}^{L}\sum_{c=1}^{q_j} y_{b,(j-1)L+c} J_{ij}(a,c) - h_i(a)\right)\right) \tag{S16}$$

with $y_{b,n}$ denoting the (b,n)th entry of $\mathbf{Y}$.

Based on this representation, the estimate of the parameters can be found by minimizing the objective function $\mathrm{K}_{\mathbf{h},\mathbf{J}}$. To account for the fact that the number of samples per parameter is quite small, even after grouping the mutants, we introduce a L2 regularization term to the objective function. Moreoever, it is reasonable to assume that a significant proportion of the couplings are either zero or negligible, as most residues are physically seperated in the protein native state. As such, we also introduce a L1 regularization term, which enforces parameter sparsity. Including the L1 and L2 regularization terms, we obtain the following MPF proxy for the original optimization problem (Eq. S6):

$$\left(\mathbf{h}^{\mathrm{MPF}}, \mathbf{J}^{\mathrm{MPF}}\right) = \arg\min_{\mathbf{h},\mathbf{J}} \mathrm{K}_{\mathbf{h},\mathbf{J}} + \lambda_1 \sum_{i=1}^{L}\sum_{a=1}^{q_i}\sum_{j=i+1}^{L}\sum_{b=1}^{q_j} \left|J_{ij}(a,b)\right| + \lambda_2 \sum_{i=1}^{L}\sum_{a=1}^{q_i}\sum_{j=i+1}^{L}\sum_{b=1}^{q_j} J_{ij}(a,b)^2 \tag{S17}$$

where λ_1 and λ_2 are the L1 and L2 regularization parameters respectively. Note that this objective function is convex and does not contain the intractable partition function. In particular, it involves only a quadratic number of terms in $\tilde{L}$, and can be easily optimized using standard gradient descent algorithms. We run gradient descent multiple times in parallel to obtain

different field and coupling parameter sets, with each set corresponding to a different set of regularization parameters.

2.4 Refinement through BML: Each field and coupling parameter set that solves Eq. S17 is used to initialize a BML algorithm using MCMC simulations to approximate the gradient. BML was implemented by a modified RPROP algorithm (6), which was run for each parameter set. The couplings which were set to zero due to the L1 regularization in Eq. S17 were fixed to zero during each iteration of the BML algorithm. Out of these different parameter sets, we choose the one, as in (7, 8), such that

$$\varepsilon_1 = \frac{1}{\tilde{L}} \sum_{i=1}^{L} \sum_{a=1}^{\tilde{q}_i} \frac{\left(f_i^{\text{model}}(a;\lambda_1,\lambda_2) - f_i(a,\phi^*) \right)^2}{\frac{f_i(a,\phi^*)\left(1 - f_i(a,\phi^*)\right)}{P}} \approx 1$$

$$\varepsilon_2 = \frac{1}{\sum_{k=1}^{L} \tilde{q}_k \sum_{l=k+1}^{L} \tilde{q}_l} \sum_{i=1}^{L} \sum_{a=1}^{\tilde{q}_i} \sum_{j=i+1}^{L} \sum_{b=1}^{\tilde{q}_j} \frac{(f_{ij}^{\text{model}}(a,b;\lambda_1,\lambda_2) - f_{ij}(a,b,\phi^*))^2}{\frac{f_{ij}(a,b,\phi^*)\left(1 - f_{ij}(a,b,\phi^*)\right)}{P}} \approx 1 \qquad \text{(S18)}$$

where $f_i^{\text{model}}(a;\lambda_1,\lambda_2)$ and $f_{ij}^{\text{model}}(a,b;\lambda_1,\lambda_2)$ are the model single and double mutant probabilities using regularization parameters λ_1 and λ_2 using the refined parameters from BML, and $f_i(a,\phi^*)$ and $f_{ij}(a,b,\phi^*)$ are the single and double mutant probabilities after grouping with combining factor ϕ^*. Note that the condition in Eq. S18 is to ensure that the regularization parameters are chosen to balance overfitting and underfitting the single and double mutant probabilities.

2.5 Summary of computational framework: The computational framework can be summarized by the following steps:

1. Combining mutants: From the pre-processed MSA, combine the mutants according to Eq. S8, with the combining factor chosen such that the mean of Eq. S11 over all sites is approximately one. Form a new MSA based on these combined mutants.
2. Minimum probability flow: Convert this new MSA into the binary matrix **Y,** which serves as an input to the optimization problem in Eq. S17. Solve this problem by any gradient descent method to obtain a parameter set. Repeat this multiple times for different regularization parameters, to obtain multiple field and parameter sets.
3. Refinement of parameters: Run any gradient descent algorithm multiple times in parallel to solve (Eq. S6), with each time using a different parameter set from step 2. Choose the parameter set in accordance with Eq. S18.

3) Fitness verification

To verify that the inferred prevalence landscape correlates well with the fitness landscape, we compared with in vitro experimental fitness measurements from literature. These were obtained by infectivity or competition assays using CCR5-tropic strains. In Fig. S1 we plot the normalized logarithm of these fitness measurements $\ln \frac{f_i}{f_{WT}}$ vs the normalized energy $E_i - E_{WT}$ for each experiment, showing in each case a strong negative correlation. Simply calculating the Spearman correlation by aggregating the fitness measurements collated from all papers is not meaningful, as the experiments and fitness measures are not consistent across these papers. Thus to account for this inconsistency, we instead consider the average Spearman correlation, a common approach in meta-analysis (9), which calculates the weighted average of the individual Spearman correlation ρ_i, with the weights equal to the number of fitness measurements N_i, and thus given by

$$\bar{\rho} = \frac{\sum_{i=1}^{7} N_i \rho_i}{\sum_{i=1}^{7} N_i} \quad . \tag{S19}$$

To further validate our computational framework, we applied it to p24, an internal protein in HIV. The inferred prevalance landscape is observed to also have a strong correlation with in vitro experimental fitness measurements (Fig. S2), and is the relative energy of the different strains is qualitavely the same as those observed in (10). These experiments were obtained by site-directed mutagensis of single, double and triple mutations into the HIV-1 NL4-3 reference strain (10).

4) Predicting residues in contact

The couplings can also be used to infer residues which are in contact in the protein native structure, and serves as a direct verification that the inferred couplings contain useful biological information (11–13). To determine the association between residues based on the inferred couplings, we consider the Direct Information (DI) (11), a measure which only takes into account the couplings between residues, and is thus a reflection of only direct associations between residues. The DI between residues *i* and *j* is given by

$$\mathrm{DI}_{i,j} = \sum_{a=1}^{\tilde{q}_i} \sum_{b=1}^{\tilde{q}_j} f_{ij}^{\mathrm{dir}}(a\ ,b) \ln \frac{f_{ij}^{\mathrm{dir}}(a,b)}{f_i(a)\, f_j(b)} \tag{S20}$$

where

$$f_{ij}^{\mathrm{dir}}(a\ ,b) = \frac{\exp\left(J_{ij}(a,b) + \tilde{h}_i(a) + \tilde{h}_j(b)\right)}{Z_{ij}}$$

with $\tilde{h}_i(a)$ and $\tilde{h}_j(b)$ chosen such that

$$f_i(a) = \sum_{b=1}^{\tilde{q}_j} f_{ij}^{\mathrm{dir}}(a\ ,b) \tag{S21}$$

$$f_j(b) = \sum_{a=1}^{\tilde{q}_i} f_{ij}^{\mathrm{dir}}(a\ ,b)$$

and Z_{ij} is a normalization constant. To reduce the effects of sampling noise and phylogenetic effects, we then modify the DI using the standard average product correction (APC) (12), which we will refer to as DI-APC. Note that the DI-APC is only a function of the couplings, and not the fields.

As residues in contact are likely to be strongly coupled, we expect that the DI-APC between these residues to be high, assuming that our inferred couplings is correct (11). We show this to be the case in Fig. S3, which plots the true positive rate (TPR) – the fraction of contacts in the top *x* predicted pairs (based on the DI-APC) which are actually in contact based on distance between the CA atoms (based on the PDB crystal structure) – vs the top *x* pairs. We observe in all cases an excellent prediction of the contacts compared to a random prediction. There are however apparent false positives which are observed (Table S5), which may be due to conformational changes that occurs when gp160 interacts with CD4 and other co-receptors.

5) Comparison with other models. We compared the ability of our MPF-BML model to predict fitness rank ordering and protein contacts with three other computationally-efficient methods (Fig. S4): MPF without BLM refinement, a fields-only model calculated to exactly fit only the single mutant probabilities, and mean-field Direct Coupling Analysis (mfDCA) (13). We observe that our model has the largest prediction accuracy in terms of both fitness rank ordering and protein. The fields-only model does quite a good job at predicting fitness rank ordering, however it fails to predict protein contacts, whereas mfDCA does well at identifying contacts, but performs poorly at predicting fitness ordering.

6) Envelope trimer geometry and surface residues. Spatial information about the envelope trimer was obtained from

Protein Data Bank ID 5D9Q. Although this is a crystal structure of SOSIP, it is assumed that the coordinates are representative of the native gp160 structure. Each residue was assigned a center of mass as the weighted average of the coordinates of its atoms (excluding hydrogen, which is not included in the crystal structure), with weights equal to the respective atomic masses. The centers of mass were used when determining residue neighborhoods below. Each residue was also assigned a solvent accessible surface area (SASA) using the PyMOL software (www.pymol.org) get_area() function, using a 1.4 solvent radius parameter, for each residue in the crystal structure. SASA values per residue were converted to relative solvent accessibility (RSA) values by normalizing by the respective SASA values per residue in a Gly-X-Gly tripeptide construct, as published by (14). Residues with RSA >0.2, used as a threshold in (15), are considered surface residues while the remaining residues are considered "buried."

7) Fitness costs. Assuming that the prevalence landscape inferred in this work accurately reflects intrinsic fitness, a measure of the difficulty for a virus to make a certain set of mutations can be quantified by the "fitness cost," or the change in energy (Eq. S4) upon introducing these mutations. We first consider the fitness cost of a mutation from the consensus amino acid to a non-consensus amino acid at a single residue. Before presenting the expression for this, it is convenient to introduce the fitness cost for amino acid *a* at residue *i* as

$$\Delta E_{i,\mathbf{h}_\mathrm{f},\mathbf{J}_\mathrm{f}}(a) = \sum_{\mathbf{x},x_i=0} \left(E_{\mathbf{h}_\mathrm{f},\mathbf{J}_\mathrm{f}}(\mathbf{x}') - E_{\mathbf{h}_\mathrm{f},\mathbf{J}_\mathrm{f}}(\mathbf{x}) \right) p_{\mathbf{h}_\mathrm{f},\mathbf{J}_\mathrm{f}}(\mathbf{x}) \tag{S22}$$

where $\mathbf{h}_\mathrm{f}$ and $\mathbf{J}_\mathrm{f}$ represent the final parameters after BML refinement, and the summation is over all sequences $\mathbf{x}$ having the consensus amino acid ('0') at the *i*th residue. Moreover, for any given $\mathbf{x}$, we define $\mathbf{x}'$ as the corresponding strain, but with amino acid *a* at residue *i*. Thus, Eq. S22 can be intepreted as the fitness cost from the consensus amino acid to amino acid *a*, averaged over all sequence backgrounds. From this, we can then obtain the fitness cost for residue *i* obtained by averaging Eq. S22 over all non-consensus amino acids:

$$\overline{\Delta E_i} = \frac{\sum_{a=1}^{\tilde{q}_i} \Delta E_{i,\mathbf{h}_\mathrm{f},\mathbf{J}_\mathrm{f}}(a) \exp\left[-\Delta E_{i,\mathbf{h}_\mathrm{f},\mathbf{J}_\mathrm{f}}(a) \right]}{\sum_{a=1}^{\tilde{q}_i} \exp\left[-\Delta E_{i,\mathbf{h}_\mathrm{f},\mathbf{J}_\mathrm{f}}(a) \right]} \quad . \tag{S23}$$

Here, we use Boltzmann averaging to emphasize low-fitness-cost pathways.

The above expression for the fitness cost is for the case in which each residue is treated in isolation. This is now extended to the case of long length scales, in which each residue is averaged along with all surface residues within a 12.5-Angstrom radius (based on residue center-of-mass). For a residue *i*, let us denote the set of all such proximal surface residues as $\mathcal{E}_i$. Then the fitness cost is averaged over all possible mutations within the neighborhood of (and including) residue *i*, given by:

$$\overline{\Delta E_{\mathcal{E}_i}} = \frac{\sum_{j\in\mathcal{E}_i} \sum_{a=1}^{\tilde{q}_j} \Delta E_{j,\mathbf{h}_\mathrm{f},\mathbf{J}_\mathrm{f}}(a) \exp\left[-\Delta E_{j,\mathbf{h}_\mathrm{f},\mathbf{J}_\mathrm{f}}(a) \right]}{\sum_{j\in\mathcal{E}_i} \sum_{a=1}^{\tilde{q}_j} \exp\left[-\Delta E_{j,\mathbf{h}_\mathrm{f},\mathbf{J}_\mathrm{f}}(a) \right]} \quad . \tag{S24}$$

For all calculations in this work involving fitness costs, averages over sequence background were performed using an MCMC sample of 33,330 sequences, assuming all mutations are from the consensus amino acid at that residue to a non-consensus amino acid. The MCMC sample used a burn-in period of 1e7 steps and a thinning parameter of 3e3 steps.

8) CATNAP analysis. CATNAP is a tool hosted by Los Alamos HIV Database that compiles published data in the form of antibody-virus neutralization values (IC-50). We used this tool to extract all IC-50 values for any bnAb-virus pairs in which the viral sequence is known and also included in the clade B sequences used to fit our landscape; we assumed that these sequences are viable (i.e. the corresponding viruses can grow and replicate in vivo). For any particular antibody, there

exists a panel of viruses against which the IC-50 has been measured. CATNAP uses Fisher's exact test (16) to determine whether any particular amino acids at any residue are statistically associated with either high or low IC-50 in the viral panel. We tagged any such residue as "IC-50-affecting." In other words, mutations at these binding-associated residues can potentially allow a bnAb that previously did not bind to the virus to bind upon mutation, or vice versa. Thus, these residues are likely candidates for escape from antibody pressure.

The CATNAP tool was used to analyze binding properties of the following antibodies: 10-1074, 1B2530, 8ANC131, 8ANC134, B12, B13, F105, JM4, NIH45-46, PGT128, PGT135, PGT151, VRC-PG04, VRC01, VRC07, VRC13, VRC16, VRC18, VRC23, VRC27.

For a particular residue *i*, the average biochemical similarity, $\bar{s}_i$, between the observed non-consensus amino acids and the respective consensus amino acid for that residue was obtained using the following formula:

$$\bar{s}_i = \frac{\sum_{j=1}^{n_{panel}} s_i(x_{ji}, c_i)[1-\delta(x_{ji}, c_i)]}{\sum_{j=1}^{n_{panel}} [1-\delta(x_{ji}, c_i)]} \qquad \text{(S25)}$$

where n_{panel} is the number of sequences in the CATNAP panel, x_{ji} is the amino acid at residue *i* of the j-th panel sequence, c_i is the consensus amino acid at residue *i*, $s_i(a, b)$ is the similarity between amino acids *a* and *b*, as defined in (17), and δ is the Kronecker delta function which is unity if the two arguments are equal. This quantity reflects the average similarity between consensus and non-consensus amino acids within the CATNAP panel.

Supplementary Figures and Tables

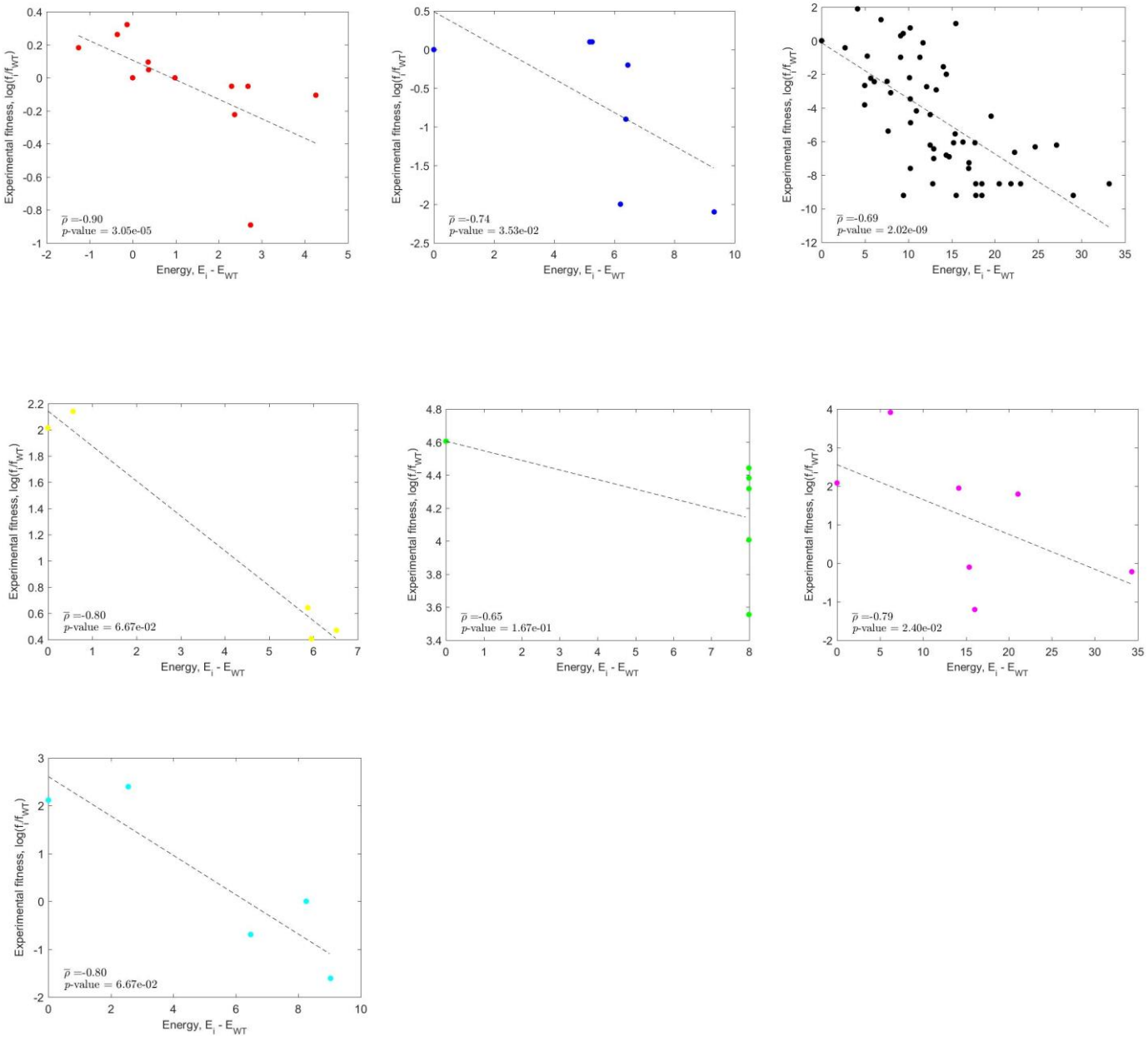


Fig. S1. Logarithm of fitness vs. energy normalized by the reference strain for the seven experiments collected from literature using a combining factor $\phi = 0.95$, for gp160. A strong negative correlation is observed for all experiments. The experimental values are taken clockwise respectively from (18), (19), (20), (21), (22), (23) and (21).

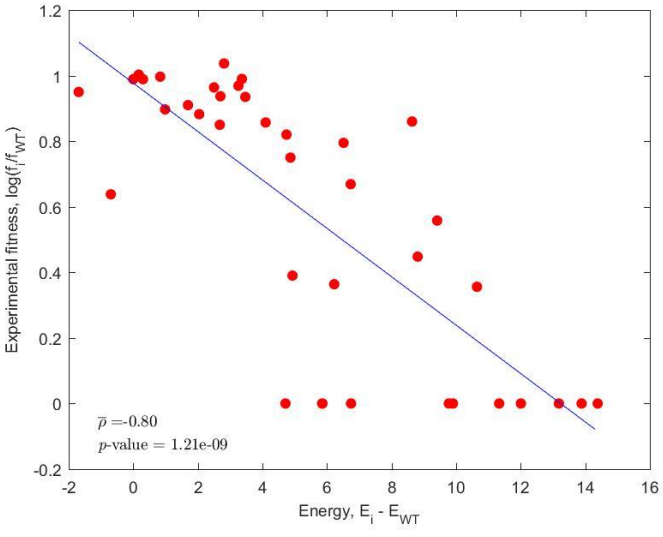


Fig. S2. Logarithm of fitness vs. energy normalized by the reference strain for p24. A strong negative correlation is observed.

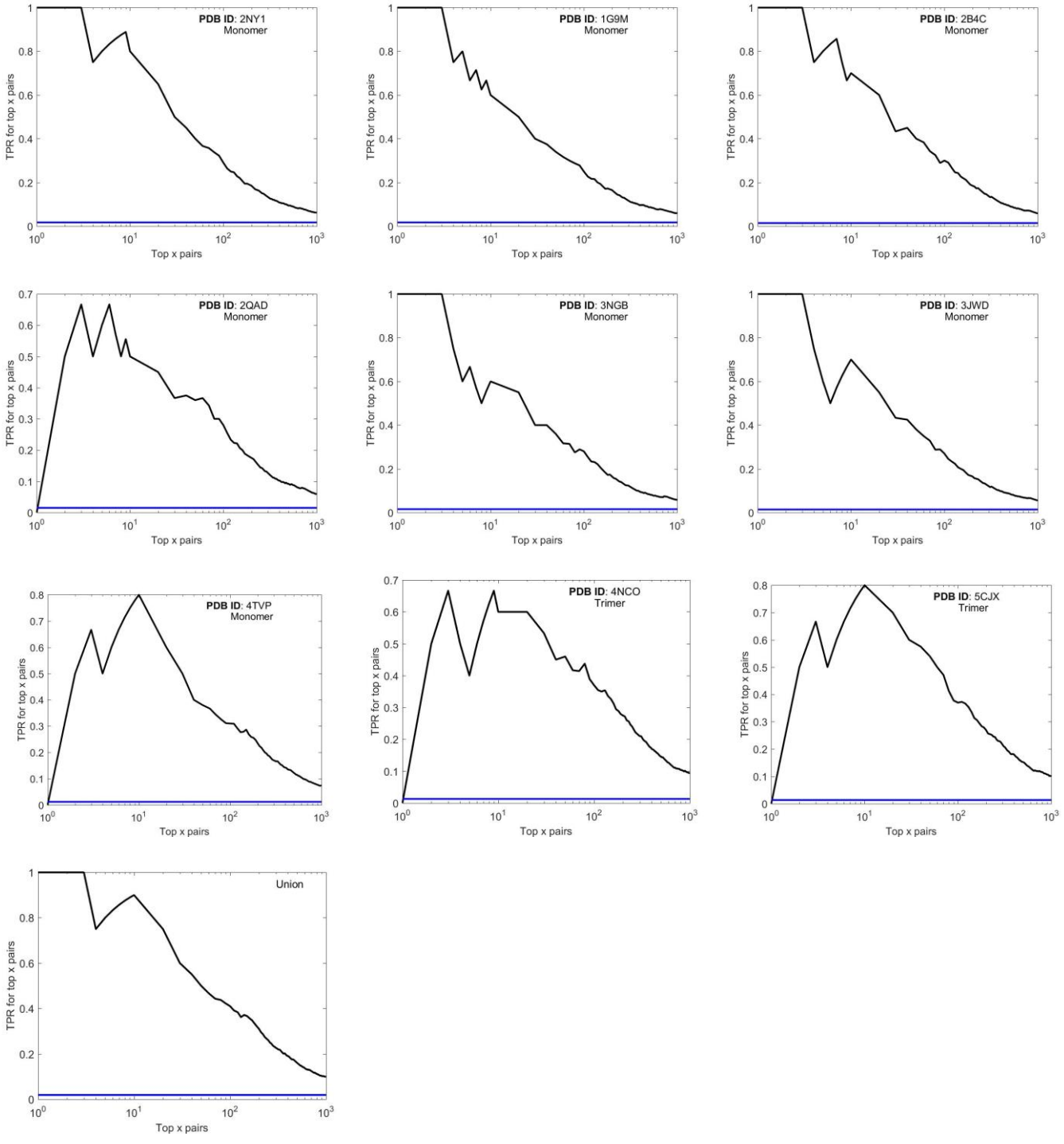
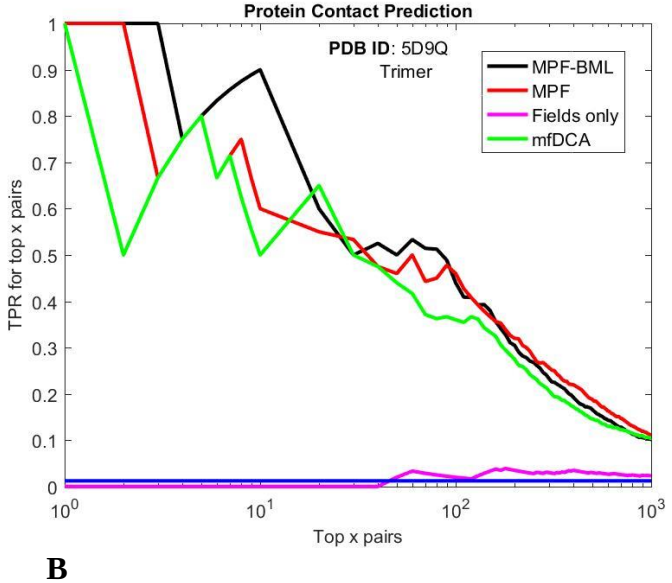


Fig. S3. Fraction of contacts in the top *x* predicted pairs which are actually in contact (true positive rate (TPR)) vs the top *x* pairs (black line). Also plotted is the total number of contacts divided by the total number of pairs (horizontal blue line), which represents the probability of choosing a pair in contact purely by chance. Only pairs > 5 distance apart in sequence space are considered, and out of these pairs, two residues are assumed to be in contact if they are < 8Å apart. The last figure shows the TPR curve using the union of all contact maps.

Method	Weighted Correlation of Fitness vs Energy (fitness rank ordering prediction)
MPF-BML	-0.7353
MPF	-0.6918
Fields only	-0.6870
mfDCA	-0.4378



A **B**

Fig. S4. A) Average weighted correlation between experimental fitness and predicted energy. B) Fraction of contacts in the top *x* predicted pairs which are actually in contact (true positive rate (TPR)) vs the top *x* pairs (black line). Predictions are made by four different models: MPF/BML (the model using our framework), MPF (the model using our framework without BML), a fields only model (fields are matched only to the single mutant probabilities and couplings are zero), and mfDCA (mean-field approach) (13).

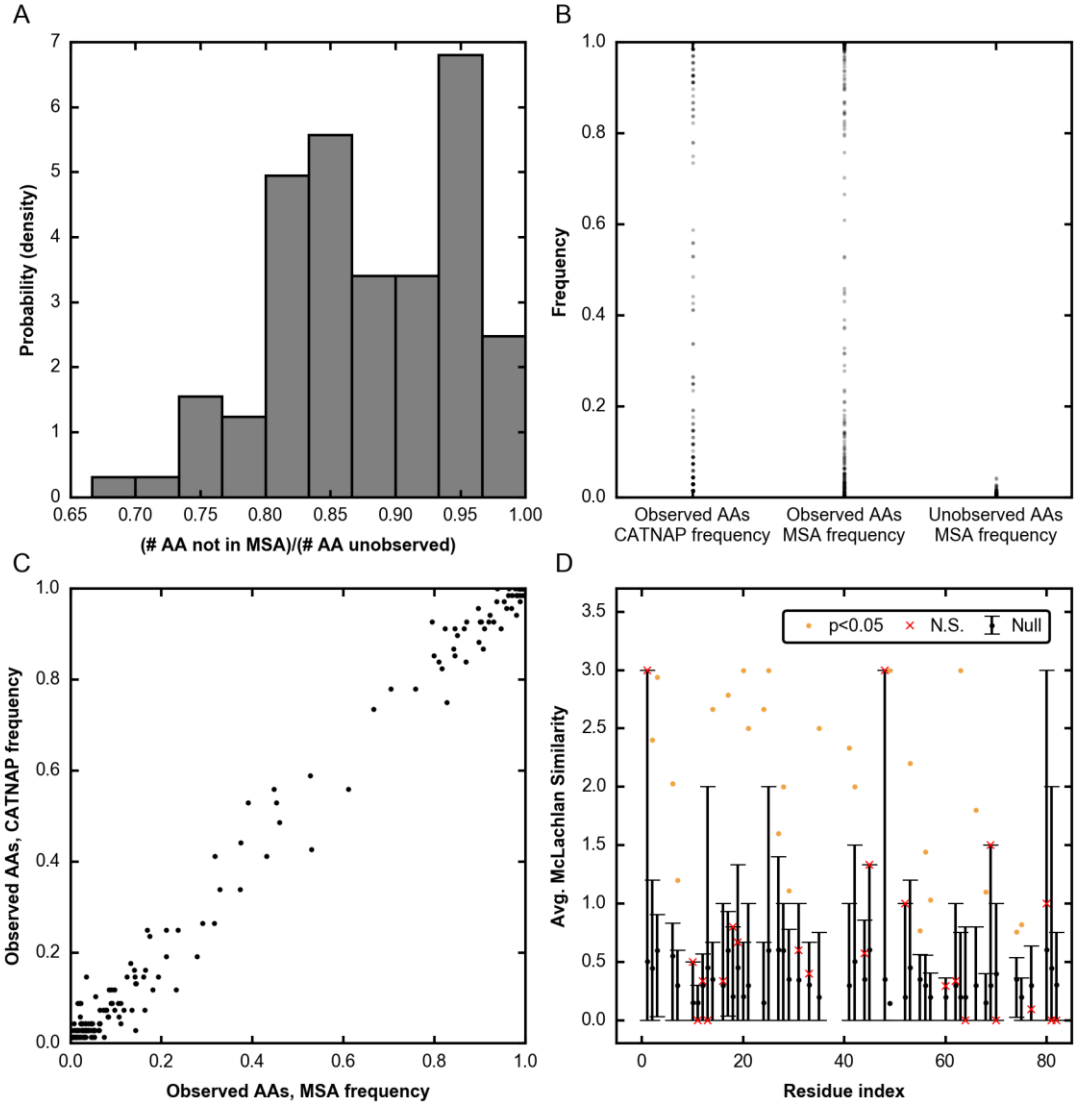


Fig. S5. Further analysis of CATNAP data. A) Histogram, over the residues in the CD4bs bnAb footprints, of the fraction of amino acid (AA) mutations that are not observed in the MSA used to fit the landscape, out of all mutations not observed in the CATNAP data. B) Frequencies of amino acid mutations classified in three ways: 1. "Observed AAs, CATNAP frequency" consists of the CATNAP frequencies of mutations observed in CATNAP; 2. "Observed AAs, MSA frequency" consists of the MSA frequencies of mutations observed in CATNAP; and 3. "Unobserved AAs, MSA frequency" consists of the MSA frequencies of mutations unobserved in CATNAP. C) Comparison of MSA and CATNAP frequencies for amino acid mutations observed in CATNAP. D) Average McLachlan similarity between consensus and non-consensus amino acids at non-IC-50 observed in the CATNAP panels for the bnAbs studied. The values are compared with a null distribution in which the non-consensus amino acids found in the CATNAP viruses are replaced with random amino acids chosen uniformly from all possible non-consensus amino acids. The error bars plotted are from the minimum to the 95th percentile. For residues having average similarity in the top 95th percentile of null model, this average is plotted as an orange dot, while the remaining (not significant) are plotted as red crosses. Missing data indicates a residue that had no non-consensus amino acids represented in the CATNAP data.

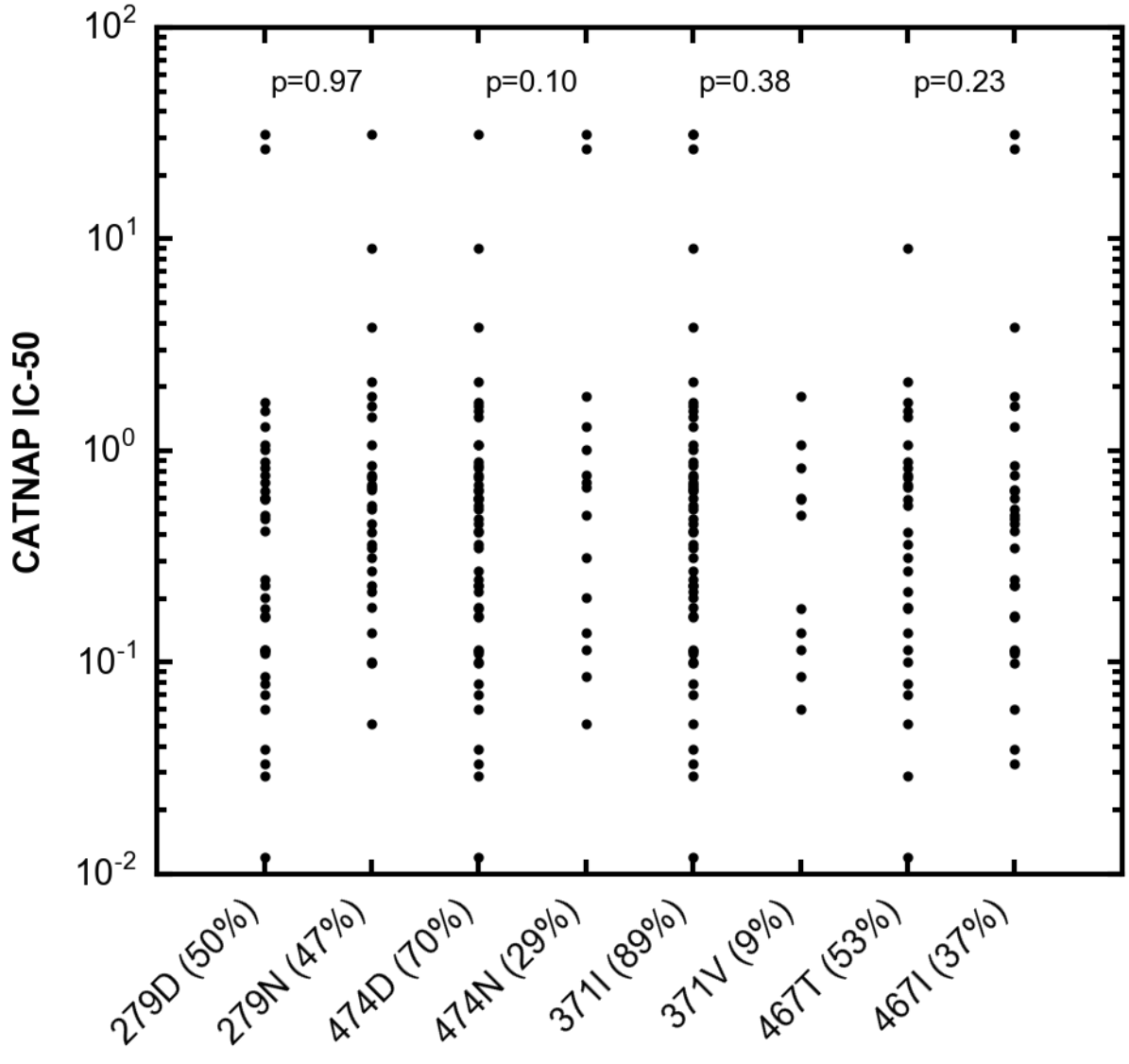


Fig S6. IC-50 values in the CATNAP panel for bnAb VRC01, for four residues (279, 474, 371, 467) which have low predicted fitness cost. The IC-50 values are split into two groups based on whether the sequence has the consensus or most common mutant amino acid, labeled on the x-axis; e.g. 279D (50%) indicates sequences in the panel in which residue 279 has amino acid D, which is present in 50% of sequences in the MSA used to fit our landscape. P-values are shown for each residue, and indicate that the distribution of IC-50 values for sequences containing either consensus or most common mutant amino acid at a particular residue are statistically indistinguishable using a two-tailed t-test.

Protein	**No. param**	**Avg. ent.**	**Length, *L***	**Avg. no. mut.**
gp160	28,262,831	0.46	856	7.78
RT	3,153,651	0.12	440	4.71
p24	1,143,462	0.09	231	5.55
Integrase	1,141,235	0.12	288	4.25
p17	729,218	0.26	132	8.18
nef	569,292	0.36	205	5.50
vif	519,117	0.30	192	5.30
protease	323,981	0.33	99	7.17
p15	238,351	0.14	120	4.77
rev	198,462	0.39	116	6.25
p6	155,691	0.33	52	9.82
p7	132,960	0.22	55	8.45
vpu	99,951	0.45	82	7.29
p2	10,358	0.49	14	9.64
p1	9,914	0.13	16	8.06

Table S1. Number of parameters, average entropy (over all residues), length and average number of mutants (over all residues) using the HXB2 sequence as a reference sequence, and based on the amino acid multiple sequence alignment (MSA) of HIV-1 Clade B gp160 sequences from the Los Alamos National Laboratory HIV sequence database (www.hiv.lanl.gov; accessed 11th Dec, 2016). gp160 has the largest number of parameters (without combining mutants) and approximately 9× more parameters than the protein (RT) with the second largest number. As gp160 has the second largest entropy, it also has the largest number of parameters after appropriate combining of mutants (e.g. using (S13)).

ϕ	**Mean of** $\beta_i(\phi)$	**Number of parameters**
0	78.8285	332,520
0.1	78.8285	332,520
0.2	77.87298	334,150
0.3	75.52216	337,420
0.4	66.94252	353,190
0.5	54.21058	384,050
0.6	41.60982	441,170
0.7	24.29135	593,120
0.8	10.77352	965,830
0.9	3.010148	2,394,300
0.95	0.816736	4,415,500
1	0	26,070,000

Table S2. The mean of $\beta_i(\phi)$ for different ϕ, and the corresponding number of parameters. We observe that the mean is close to one when $\phi = 0.95$.

Exp. *i*	**Ref.**	N_i	$\phi = 0$	$\phi = 0.2$	$\phi = 0.4$	$\phi = 0.6$	$\phi = 0.8$	$\phi = 0.95$	$\phi = 1$
1	(18)	12	-0.57	-0.59	-0.74	-0.79	-0.84	-0.90	-0.85
2	(19)	7	-0.20	-0.20	-0.31	-0.27	-0.40	-0.74	-0.70
3	(20)	56	-0.60	-0.62	-0.59	-0.61	-0.67	-0.69	-0.63
4	(21)	5	-0.56	-0.87	-0.87	-0.56	-1.00	-0.80	-0.80
5	(22)	6	-0.65	-0.65	-0.65	-0.65	-0.65	-0.65	-0.85
6	(23)	7	-0.71	-0.71	-0.75	-0.71	-0.71	-0.79	-0.43
7	(21)	5	-0.67	-0.67	-0.67	-0.67	-0.80	-0.80	-0.80
		$\bar{\rho} =$	-0.58	-0.61	-0.62	-0.62	-0.70	-0.74	-0.68

Table S3. For different combining factors ϕ, and for seven different experiments, the Spearman correlation between measured fitness and the energy as calculated from the prevalence landscape is shown (fourth to tenth column). The parameter N_i denotes the number of fitness measurements for the *i*th experiment. The last row shows the weighted Spearman correlation (Eq. S19). This correlation is strongest when $\phi = 0.95$ (for which, there is a strong correlation for each individual experiment), providing validation for the data-driven choice of the combining factor ϕ.

Residue i	Residue j	Distance (Å)	Function
389	417	9.3763	Both in V4 Loop
65	208	9.0257	?
192	426	21.6352	192 in V2 Loop, 426 is a CD4 contact
178	195	14.2069	Both in V2 Loop
279	474	15.1180	Both are CD4 contacts
164	195	9.5545	Both in V2 Loop
178	194	9.3138	Both in V2 Loop
170	178	24.2928	Both in V2 Loop

Table S4. The eight residue-pairs predicted by our landscape to be in contact, but are not in contact according to PDB structure 5D9Q. The distance between residues and functional domains are indicated. We observe that with the exception of one residue-pair, all pairs are either in the V2 or V4 loop, or are CD4 contacts.

Residue type	HXB2 #	Neutralization CD4-Ig (%)	Binding VRC01 (%)	Escape cost
High fitness **cost, low affinity**	**368**	ND	3	7.35
	367	ND	28	6.84
	457	ND	29	6.57
High fitness **cost, moderate affinity**	**473**	ND	79	6.90
	458	ND	84	6.69
	366	ND	50	6.44
	469	ND	62	6.44
	427	ND	76	6.02
	280	ND	66	5.68
Other	**456**	55	80	6.10
	276	29	193	5.61
	430	>10000	98	5.15
	459	722	63	4.95
	282	5	68	4.68
	97	184	88	4.67
	455	22	65	4.37
	365	32	149	4.18
	283	5	78	3.04
	476	5	55	2.86
Low fitness cost, low affinity	**371**	ND	30	3.12
	474	11	20	2.79
	279	2	0	2.13
	467	ND	28	1.99

Table S5. VRC01 bnAb contact residues (as determined by (24)), characterized by experimental loss of neutralization sensitivity to CD4-Ig and loss of binding affinity for respective Alanine scanning mutants, reported as percent neutralization sensitivity or binding affinity of mutant compared to wild type. ND indicates that the Alanine mutant did not support entry. The "high fitness cost, low affinity" residues greatly diminish VRC01 binding affinity (<33%) when mutated to Alanine, and are predicted to have high fitness cost. The "high fitness cost, moderate affinity" residues are less detrimental to affinity, but do not support entry, which is reflected in our high predicted fitness costs. The "low fitness cost, low affinity" residues result in diminished VRC01 affinity upon mutation to Alanine, but are paradoxically predicted to have low fitness cost despite their low sensitivity to neutralization by CD4-Ig. This is explained in the main text by the high prevalence of the most-common mutant amino acid for each of these residues.

References

1. Shekhar K, et al. (2013) Spin models inferred from patient-derived viral sequence data faithfully describe HIV fitness landscapes. *Phys Rev E - Stat Nonlinear, Soft Matter Phys* 88:1–10.
2. Jensen MA, et al. (2003) Improved coreceptor usage prediction and genotypic monitoring of R5-to-X4 transition by motif analysis of human immunodeficiency virus type 1 env V3 loop sequences. *J Virol* 77(24):13376–88.
3. Ferguson AL, et al. (2013) Translating HIV sequences into quantitative fitness landscapes predicts viral vulnerabilities for rational immunogen design. *Immunity* 38(3):606–617.
4. Barton JP, Leonardis E De, Coucke A, Cocco S (2016) ACE: Adaptive cluster expansion for maximum entropy graphical model inference. *Bioinformatics* 32(20):3089–3097.
5. Sohl-Dickstein J, Battaglino P, DeWeese MR (2011) Minimum Probability Flow learning. *Proceedings of the 28th International Conference on Machine Learning*, pp 905–912.
6. Riedmiller M, Braun H (1993) A direct adaptive method for faster backpropagation learning: The RPROP algorithm. *IEEE International Conference on Neural Networks*, pp 586–591.
7. Barton JP, et al. (2016) Relative rate and location of intra-host HIV evolution to evade cellular immunity are predictable. *Nat Commun* 7(5):1–10.
8. Barton J, Cocco S (2013) Ising models for neural activity inferred via selective cluster expansion: structural and coding properties. *J Stat Mech Theory Exp* (3):1–57.
9. Field A (2001) Meta-analysis of correlation coefficients: A Monte Carlo comparison of fixed- and random-effects methods. *Psychol Methods* 6(2):161–180.
10. Mann JK, et al. (2014) The fitness landscape of HIV-1 gag: Advanced modeling approaches and validation of model predictions by in vitro testing. *PLoS Comput Biol* 10(8):1–11.
11. Weigt M, White RA, Szurmant H, Hoch JA, Hwa T (2009) Identification of direct residue contacts in protein-protein interaction by message passing. *Proc Natl Acad Sci* 106(1):67–72.
12. Dunn SD, Wahl LM, Gloor GB (2008) Mutual information without the influence of phylogeny or entropy dramatically improves residue contact prediction. *Bioinformatics* 24(3):333–340.
13. Morcos F, et al. (2011) Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proc Natl Acad Sci* 108(49):E1293-301.
14. Sneath PHA (1966) Relations between chemical structure and biological activity in peptides. *J Theor Biol* 12(2):157–195.
15. Jardine JG, et al. (2016) Minimally mutated HIV-1 broadly neutralizing antibodies to guide reductionist vaccine design. *PLoS Pathog* 12(8):1–33.
16. Fisher R (1922) On the interpretation of $\chi 2$ from contingency tables, and the calculation of P. *J R Stat Soc* 85(1):87–94.
17. McLachlan AD (1972) Repeating sequences and gene duplication in proteins. *J Mol Biol* 64(2):417–437.
18. Troyer RM, et al. (2009) Variable fitness impact of HIV-1 escape mutations to cytotoxic T lymphocyte (CTL) response. *PLoS Pathog* 5(4):30–34.
19. Liu Y, et al. (2014) A sensitive real-time PCR based assay to estimate the impact of amino acid substitutions on the competitive replication fitness of human immunodeficiency virus type 1 in cell culture. *J Virol Methods* 189(1):157–166.
20. da Silva J, Coetzer M, Nedellec R, Pastore C, Mosier DE (2010) Fitness epistasis and constraints on adaptation in a human immunodeficiency virus type 1 protein region. *Genetics* 185(1):293–303.
21. Lobritz MA, Marozsan AJ, Troyer RM, Arts EJ (2007) Natural variation in the V3 crown of human immunodeficiency virus type 1 affects replicative fitness and entry inhibitor sensitivity. *J Virol* 81(15):8258–69.
22. Kassa A, et al. (2009) Identification of a human immunodeficiency virus type 1 envelope glycoprotein variant

resistant to cold inactivation. *J Virol* 83(9):4476–88.

23. Anastassopoulou CG, et al. (2007) Escape of HIV-1 from a small molecule CCR5 inhibitor is not associated with a fitness loss. *PLoS Pathog* 3(6):720–732.

24. Li Y, et al. (2011) Mechanism of neutralization by the broadly neutralizing HIV-1 monoclonal antibody VRC01. *J Virol* 85(17):8954–8967.