Supporting Information for

Relative Principal Components Analysis: Application to Analyzing Biomolecular Conformational Changes

Mazen Ahmad, Volkhard Helms, Olga V. Kalinina and Thomas Lengauer

The notation in this supplementary text follows the notation in the main text.

S1. Optimal Relative PCAs via Average-Covariance Sub-Spacing

Let $\boldsymbol{x} = (x_1 \dots x_d)^T$ be a d-dimensional continuous random variable with two samples from two macroscopic states that will be labeled (a) and (b). RPCA aims at finding k latent canonical variables $\boldsymbol{y} = (y_1, y_2 \dots y_k)^T = f(\boldsymbol{x})$ which satisfy:

$$\max D_{kl}(y_i = f(\boldsymbol{x}); a{:}\, b)$$

$$D_{kl}(\boldsymbol{y} = (y_1, y_2 \dots y_k)^T; a{:}\, b) = \sum_{i=1}^{k} D_{kl}(y_i; a{:}\, b) \tag{1.1}$$

Under the model assumption of normality (see the main text), the KL divergence of a normally distributed variable $y_i = \boldsymbol{g}_i^T \boldsymbol{x}$ is given by:

$$D_{kl}\big(y_i = \boldsymbol{g}_i^T \boldsymbol{x}; a{:}\, b\big) = \frac{1}{2}\left[\underbrace{-\ln\frac{\boldsymbol{g}_i^T \boldsymbol{S_b} \boldsymbol{g}_i}{\boldsymbol{g}_i^T \boldsymbol{S_a} \boldsymbol{g}_i} + \frac{\boldsymbol{g}_i^T \boldsymbol{S_b} \boldsymbol{g}_i}{\boldsymbol{g}_i^T \boldsymbol{S_a} \boldsymbol{g}_i} - 1}_{\text{Variance change}} + \underbrace{\frac{\boldsymbol{g}_i^T \Delta\Delta^T \boldsymbol{g}_i}{\boldsymbol{g}_i^T \boldsymbol{S_a} \boldsymbol{g}_i}}_{\text{Average change}}\right] \tag{1.2}$$

$$\lambda_i = \frac{\boldsymbol{g}_i^T \boldsymbol{S_b} \boldsymbol{g}_i}{\boldsymbol{g}_i^T \boldsymbol{S_a} \boldsymbol{g}_i}$$

Kullback emphasized that the generalized eigenvectors and eigenvalues obtained by the simultaneous diagonalization algorithm are stationary points when maximizing the first term of the right-hand side of equation (1.2), which accounts for the contribution to the KL divergence due to variance change [1]. The total KL divergence of the variables y_i is not maximum at these points, due to the additional contributions from the second term of the right-hand side of equation (1.2), when $\Delta \neq \boldsymbol{0}$. One way of avoiding this problem is finding a transformation matrix $\boldsymbol{G} = [\boldsymbol{g}_\mu \quad \boldsymbol{G_v}]$ such that the KL divergence of the variable $y_\mu = \boldsymbol{g}_\mu^T \boldsymbol{x}$ comprises the contribution to the KL divergence due to the second term (the change of the average), while the simultaneous diagonalization algorithm can be used to obtain the optimal variables $\boldsymbol{y_v} = \boldsymbol{G_v^T x}$ because the corresponding second term of equation (1.2) is zero ($\boldsymbol{g}_{i\neq\mu}^T \Delta = 0$).

Optimal relative principle component with respect to KL divergence due to the change of the average ($\boldsymbol{y_\mu = g_\mu^T x}$)

In case of nonsingular $\boldsymbol{S_a}$, the KL divergence of the projection on the vector $\boldsymbol{S_a^{-1}}\Delta$, is known to collect the total contributions to the divergence due to the average change [1]. Now, we will show that the solution in the general case (e.g. singular matrix $\boldsymbol{S_a}$) can be performed analogously using the generalized inverse $\boldsymbol{S_a^-}$ (see below) instead of $\boldsymbol{S_a^{-1}}$.

Maximizing $\boldsymbol{g}_i^T \Delta\Delta^T \boldsymbol{g}_i / \boldsymbol{g}_i^T \boldsymbol{S_a} \boldsymbol{g}_i$ (the contribution due to the change of the averages) from equation (1.2) with respect to $\boldsymbol{g}_i$ yields the required solution:

$$\boldsymbol{g}_\mu = \underset{\boldsymbol{g}_i \in \mathbb{R}^{d\times 1}}{\operatorname{argmax}} \frac{\boldsymbol{g}_i^T \Delta\Delta^T \boldsymbol{g}_i}{\boldsymbol{g}_i^T \boldsymbol{S_a} \boldsymbol{g}_i} \tag{1.3}$$

This maximization becomes much simpler if we first whiten the data [2] using the transformation matrix $\boldsymbol{W}$ (see the main text). The new variable $\boldsymbol{x} \to \boldsymbol{W^T x}$ has a covariance matrix at state *a* which is the identity matrix $(\boldsymbol{W^T S_a W = I})$ and the maximization problem above is turned into maximizing the term:

$$\boldsymbol{q}_\mu = \underset{\boldsymbol{q}_i \in \mathbb{R}^{k\times 1}}{\operatorname{argmax}} \frac{\boldsymbol{q}_i^T \boldsymbol{W^T} \Delta\Delta^T \boldsymbol{W q}_i}{\boldsymbol{q}_i^T \underbrace{\boldsymbol{W^T S_a W}}_{\boldsymbol{I}} \boldsymbol{q}_i} = \underset{\boldsymbol{q}_i \in \mathbb{R}^{k\times 1}}{\operatorname{argmax}} \frac{\boldsymbol{q}_i^T \boldsymbol{W^T} \Delta\Delta^T \boldsymbol{W q}_i}{\boldsymbol{q}_i^T \boldsymbol{q}_i} \tag{1.4}$$

with respect to the vector $\boldsymbol{q_i}$. The final solution $\boldsymbol{g}_\mu$ is the product $\boldsymbol{g}_\mu = \boldsymbol{W q}_\mu$. The solutions $\boldsymbol{q}_\mu$ which maximize the quadratic term in (1.4) are known to be the eigenvectors of the matrix [3] $\boldsymbol{W^T}\Delta\Delta^T\boldsymbol{W}$. Indeed, there is only one eigenvector with non-zero eigenvalue, because the rank of the matrix is 1 ($\Delta\Delta^T$ is formed from a vector). Moreover, the eigenvalues of $\underline{\boldsymbol{W^T}\Delta}\Delta^T\boldsymbol{W}$ are the same as the eigenvalue [4] of $\Delta^T\boldsymbol{W}\underline{\boldsymbol{W^T}\Delta} = \Delta^T\boldsymbol{S_a^-}\Delta$ (a scalar) and it is easy to check (see also Mardia[5]; p. 468) that the corresponding eigenvector is:

$$\boldsymbol{q}_\mu = \boldsymbol{W^T}\Delta \tag{1.5}$$

The generalized eigenvector is the product $\boldsymbol{WW^T}\Delta$ normalized with respect to the covariance matrix $\boldsymbol{S_a}$:

$$\boldsymbol{g}_\mu = \frac{\boldsymbol{S_a^-}\Delta}{\sqrt{(\boldsymbol{S_a^-}\Delta)^T \boldsymbol{S_a} (\boldsymbol{S_a^-}\Delta)}} = \frac{\boldsymbol{S_a^-}\Delta}{\sqrt{\Delta^T \underbrace{\boldsymbol{S_a^-} \boldsymbol{S_a} \boldsymbol{S_a^-}}_{\boldsymbol{S_a^-}} \Delta}} = \frac{\boldsymbol{S_a^-}\Delta}{\sqrt{\Delta^T \boldsymbol{S_a^-}\Delta}} \tag{1.6}$$

$$\boldsymbol{WW^T} = \boldsymbol{U}_k \boldsymbol{D}_k^{-1/2} \boldsymbol{D}_k^{-1/2} \boldsymbol{U}_k^T = \boldsymbol{U}_k \boldsymbol{D}_k^{-1} \boldsymbol{U}_k^T = \boldsymbol{S_a^-}$$

The normality with respect to $\boldsymbol{S_a}$ ($\boldsymbol{g_\mu^T S_a g_\mu} = 1$) is required for the simultaneous diagonalization (see below). The KL divergence due to the change of the mean along this vector is given by half of the Mahalanobis distance:

$$\frac{1}{2}\boldsymbol{g}_\mu^T \Delta\Delta^T \boldsymbol{g}_\mu = \frac{\Delta^T \boldsymbol{S_a^-} \Delta\Delta^T \boldsymbol{S_a^-}\Delta}{2\Delta^T \boldsymbol{S_a^-}\Delta} = \frac{1}{2}\Delta^T \boldsymbol{S_a^-}\Delta \tag{1.7}$$

The corresponding generalized eigenvalue is the variance of $y_\mu = \boldsymbol{g}_\mu^T \boldsymbol{x}$ at the state *b*:

$$\lambda_\mu = \boldsymbol{g}_\mu^T \boldsymbol{S_b} \boldsymbol{g}_\mu \tag{1.8}$$

Optimal relative principle components which do not contribute to the change of the average ($\boldsymbol{y_v} = \boldsymbol{G_v^T x}$)

Given the vector $\boldsymbol{g}_\mu$ which collects the total contributions to the divergence due to the change of the average (above), the remaining generalized eigenvectors $\boldsymbol{G_v}$, which do not have contributions due to the change of the averages (i.e. they are orthogonal to the change of the average $\boldsymbol{g}_{i\neq\mu}^T \Delta = 0$), are computed to be orthogonal to the vector $\boldsymbol{g}_\mu$ with respect to the matrices $\boldsymbol{S_a}$ and $\boldsymbol{S_b}$ so that the KL divergences of their corresponding

variables y_i are additive to the KL divergence of y_m (see the main text). The conditions for $\boldsymbol{G_v}$ to fulfill the restricted simultaneous diagonalization problem can be presented in the equations:

$$\boldsymbol{G^T S_a G} = \begin{bmatrix} \boldsymbol{g_\mu^T} \\ \boldsymbol{G_v^T} \end{bmatrix} \boldsymbol{S_a} [\boldsymbol{g_\mu} \quad \boldsymbol{G_v}] = \begin{bmatrix} \boldsymbol{g_\mu^T S_a g_\mu} & \boldsymbol{g_\mu^T S_a G_v} \\ \boldsymbol{G_v^T S_a g_\mu} & \boldsymbol{G_v^T S_a G_v} \end{bmatrix} = \boldsymbol{I} \tag{1.9}$$

$$\boldsymbol{G^T S_b G} = \begin{bmatrix} \boldsymbol{g_\mu^T} \\ \boldsymbol{G_v^T} \end{bmatrix} \boldsymbol{S_b} [\boldsymbol{g_\mu} \quad \boldsymbol{G_v}] = \begin{bmatrix} \boldsymbol{g_\mu^T S_b g_\mu} & \boldsymbol{g_\mu^T S_b G_v} \\ \boldsymbol{G_v^T S_b g_\mu} & \boldsymbol{G_v^T S_b G_v} \end{bmatrix} = \begin{bmatrix} \lambda_\mu & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Lambda_v} \end{bmatrix} \tag{1.10}$$

$$\boldsymbol{G_v^T} \Delta = \mathbf{0} \ (\boldsymbol{g_{i \neq \mu}^T} \Delta = 0 \ \boldsymbol{;} \ \boldsymbol{G_v} \text{ is orthogonal to the change of the average}). \tag{1.11}$$

The generalized eigenvectors $\boldsymbol{G_v}$ can be constructed using a combination of two transformation matrices:

$$\boldsymbol{G_v} = \boldsymbol{W_v L_v} \tag{1.12}$$

The weighting transformation matrix $\boldsymbol{W_v}$ is constructed here from the eigendecomposition of the projection of $\boldsymbol{S_a}$ onto the subspace which is orthogonal to the vector $\boldsymbol{S_a g_\mu}$ (orthogonal to $\boldsymbol{g_\mu}$ with respect to $\boldsymbol{S_a}$). Let $\boldsymbol{P_{S_a g_\mu}}$ be the projection matrix on the subspace which is spanned by the vector $\boldsymbol{S_a g_\mu}$. Now we can write $\boldsymbol{S_a}$ as the sum of its projection onto this subspace and its complement:

$$\begin{gathered} \boldsymbol{S_a} = \underbrace{\boldsymbol{P_{S_a g_\mu}}}_{\substack{\text{proj. matrix} \\ \text{on } \boldsymbol{S_a g_\mu}}} \boldsymbol{S_a} + \underbrace{\boldsymbol{P_{C,S_a g_\mu}}}_{\substack{\text{complement} \\ \text{proj. matrix}}} \boldsymbol{S_a} = \boldsymbol{S_{a,p}} + \underbrace{\boldsymbol{S_{a,c}}}_{\boldsymbol{U_v D_v U_v^T}} \\ \boldsymbol{P_{C,S_a g_\mu}} = \boldsymbol{I} - \boldsymbol{P_{S_a g_\mu}} \end{gathered} \tag{1.13}$$

It is important to notice that the matrix $\boldsymbol{S_{a,c}}$ is also symmetric and represents the covariance matrix of the projection $\boldsymbol{P_{C,S_a g_\mu} x}$ because $\mathrm{COV}\left(\boldsymbol{P_{C,S_a g_\mu} x}\right) = \boldsymbol{P_{C,S_a g_\mu} S_a P_{C,S_a g_\mu}} = \boldsymbol{P_{C,S_a g_\mu} S_a} = \boldsymbol{S_{a,c}}$ (see reference [2], page 169). The whitening transformation matrix $\boldsymbol{W_v}$ of the matrix $\boldsymbol{S_{a,c}}$ is constructed from the eigendecomposition of $\boldsymbol{S_{a,c}} = \boldsymbol{U_v D_v U_v^T}$:

$$\boldsymbol{W_v} = \boldsymbol{U_v D_v^{-1/2}} \tag{1.14}$$

Here, the eigenvectors in the columns of $\boldsymbol{U_v}$ correspond to the nonzero eigenvalues in the diagonal matrix $\boldsymbol{D_v}$. Now we can show that $\boldsymbol{W_v}$ whitens both of $\boldsymbol{S_{a,c}}$ and $\boldsymbol{S_a}$**:**

$$\begin{gathered} \boldsymbol{W_v^T S_{a,c} W_v} = \boldsymbol{D_v^{-1/2}} \underbrace{\boldsymbol{U_v^T S_{a,c} U_v}}_{\boldsymbol{D_v}} \boldsymbol{D_v^{-1/2}} = \boldsymbol{I} \\ \boldsymbol{W_v^T S_a W_v} = \boldsymbol{W_v^T} \left(\boldsymbol{S_{a,p}} + \boldsymbol{S_{a,c}}\right) \boldsymbol{W_v} = \underbrace{\boldsymbol{W_v^T S_{a,p} W_v}}_{\mathbf{0}} + \underbrace{\boldsymbol{W_v^T S_{a,c} W_v}}_{\boldsymbol{I}} = \boldsymbol{I} \end{gathered} \tag{1.15}$$

because

$$\boldsymbol{W_v^T S_{a,p} W_v} = \boldsymbol{D_v^{-1/2}} \underbrace{\boldsymbol{U_v^T S_{a,p}}}_{\substack{=\mathbf{0} \\ \boldsymbol{U_v \perp S_{a,p}}}} \boldsymbol{U_v D_v^{-1/2}} = \mathbf{0}$$

We have $\boldsymbol{U_v^T S_a g_\mu} = \mathbf{0}$ because the columns of $\boldsymbol{U_v}$ are orthogonal to the vector $\boldsymbol{S_a g_\mu}$**.** Thus, we can write:

$$\boldsymbol{W_v^T S_a g_\mu = D_v^{-1/2}\ \underbrace{U_v^T S_a g_\mu}_{\substack{\mathbf{0} \\ U_v \perp (S_a g_\mu)}}\ = 0} \tag{1.16}$$

The second transformation $\boldsymbol{L_v}$ can be obtained from the eigenvectors of the subspace of $\boldsymbol{W_v^T S_b W_v}$ which is orthogonal to the vector $\boldsymbol{W_v^T S_b g_\mu}$ **:**

$$\boldsymbol{L_V\ \perp W_v^T S_b g_\mu \Leftrightarrow L_V^T W_v^T S_b g_\mu = G_v^T S_b g_\mu = 0} \tag{1.17}$$

Let $\boldsymbol{P}$ be the projection matrix onto the subspace which is spanned by the vector $\boldsymbol{W_v^T S_b g_\mu}$ **.** Now we can write $\boldsymbol{S_b}$ as the sum of its projection onto this subspace and its complement:

$$\boldsymbol{W_v^T S_b W_v = P W_v^T S_b W_v + \underbrace{(I - P_c)}_{\substack{\text{complement} \\ \text{proj. matrix}}} W_v^T S_b W_v = S_{b,p} + \underbrace{S_{b,c}}_{L_V \Lambda_v L_V^T}} \tag{1.18}$$

From the eigendecomposition of $\boldsymbol{S_{b,c}}$ we have:

$$\boldsymbol{S_{b,c} = L_V \Lambda_v L_V^T \Leftrightarrow\ L_V^T S_{b,c} L_V = \Lambda_v} \tag{1.19}$$

$$\boldsymbol{G_v^T S_b G_v = L_V^T W_v^T S_b W_v L_V = L_V^T \underbrace{\left[S_{b,p} + S_{b,c}\right]}_{\substack{W_v^T S_b W_v \\ eq.\ (1.18)}} L_V = \underbrace{L_V^T S_{b,p} L_V}_{\substack{\mathbf{0} \\ L_V \perp S_{b,p}}} + \underbrace{L_V^T S_{b,c} L_V}_{\substack{\Lambda_v \\ eq.(1.19)}} = \Lambda_v} \tag{1.20}$$

Now we can show that the conditions in (1.9) are satisfied:

$$\begin{aligned} \boldsymbol{G^T S_a G} &= \begin{bmatrix} \boldsymbol{g_\mu^T} \\ \boldsymbol{G_v^T} \end{bmatrix} \boldsymbol{S_a [g_\mu \quad G_v]} = \begin{bmatrix} \underbrace{\boldsymbol{g_\mu^T S_a g_m}}_{\substack{1 \\ eq.(1.6)}} & \boldsymbol{g_\mu^T S_a G_v} \\ \boldsymbol{G_v^T S_a g_\mu} & \boldsymbol{G_v^T S_a G_v} \end{bmatrix} \\ &= \begin{bmatrix} 1 & \underbrace{\boldsymbol{g_\mu^T S_a W_v}}_{\substack{\mathbf{0} \\ eq.(1.16)}} \boldsymbol{L_V} \\ \boldsymbol{L_V^T} \underbrace{\boldsymbol{W_v^T S_a g_\mu}}_{\substack{\mathbf{0} \\ eq.(1.16)}} & \boldsymbol{L_V^T} \underbrace{\boldsymbol{W_v^T S_a W_v}}_{\substack{\boldsymbol{I} \\ eq.(1.15) \\ \boldsymbol{L_V^T L_V = I}}} \boldsymbol{L_V} \end{bmatrix} = \boldsymbol{I} \end{aligned} \tag{1.21}$$

The conditions in (1.10) are also satisfied:

$$\boldsymbol{G^T S_b G} = \begin{bmatrix} \boldsymbol{g_\mu^T} \\ \boldsymbol{G_v^T} \end{bmatrix} \boldsymbol{S_b [g_\mu \quad G_v]} = \begin{bmatrix} \underbrace{\boldsymbol{g_\mu^T S_b g_\mu}}_{\substack{\lambda_\mu \\ eq.(1.8)}} & \underbrace{\boldsymbol{g_\mu^T S_b G_v}}_{\substack{\mathbf{0} \\ eq.(1.17)}} \\ \underbrace{\boldsymbol{G_v^T S_b g_\mu}}_{\substack{\mathbf{0} \\ eq.(1.17)}} & \underbrace{\boldsymbol{G_v^T S_b G_v}}_{\substack{\Lambda_v \\ eq.(1.20)}} \end{bmatrix} = \begin{bmatrix} \lambda_\mu & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Lambda_v} \end{bmatrix} \tag{1.22}$$

Finally, we show that the condition (1.11) is satisfied:

In case of nonsingular $\boldsymbol{S_a}$ we directly obtain the condition from equation (1.16):

$$\boldsymbol{W_v^T S_a g_\mu = 0 = W_v^T S_a S_a^{-1} \Delta = W_v^T \Delta = 0} \tag{1.23}$$

Consequently, we have $\boldsymbol{L_V^T W_v^T \Delta = G_v^T \Delta = 0}$

For the general case, we can also start from equation (1.16) to show:

$$\begin{aligned} \boldsymbol{W_v^T S_a g_\mu = 0 =\ W_v^T S_a S_a^- \Delta =\ W_v^T U_k D_k \underbrace{U_k^T U_k}_{I} D_k^{-1} U_k^T \Delta =\ W_v^T U_k \underbrace{D_k D_k^{-1}}_{I} U_k^T \Delta} \\ \boldsymbol{= W_v^T \underbrace{U_k U_k^T \Delta}_{\Delta} =\ W_v^T \Delta = 0} \end{aligned} \tag{1.24}$$

Consequently, we have:

$$\boldsymbol{L_V^T W_v^T \Delta = G_v^T \Delta = 0} \tag{1.25}$$

Here, we have used the eigendecomposition of $\boldsymbol{S_a = U_k D_k U_k^T}$ to write its pseudoinverse $\boldsymbol{S_a^- = U_k D_k^{-1} U_k^T}$. It is important to notice that $\boldsymbol{U_k U_k^T \Delta = \Delta}$, because $\boldsymbol{U_k U_k^T}$ is an orthogonal projection matrix onto the (non-null) subspace. Thus, the projection of the vector Δ on this subspace is identical to Δ (identical to the projection on the total space $\boldsymbol{\Delta = UU^T \Delta}$**):**

$$\boldsymbol{\Delta = UU^T \Delta = \left[U_k U_k^T + U_0 U_0^T\right] \Delta = U_k U_k^T \Delta + \underbrace{U_0 U_0^T \Delta}_{0}} \tag{1.26}$$

S2. Data reconstruction from RPCAs

The influence of the *m*-dimensional ($m < d$) latent variables $\boldsymbol{y_m} = \boldsymbol{G_m^T x}$ on the original *d*-dimensional variable $\boldsymbol{x}$ can be presented via reconstructing the original variable $\widehat{\boldsymbol{x}}$ in the *d*-dimensional subspace which is spanned by the *m* generalized eigenvectors $\boldsymbol{G_m} = [\boldsymbol{g_1} \dots \boldsymbol{g_m}] = \boldsymbol{U}_k \boldsymbol{D}_k^{-1/2} \boldsymbol{L}_m$. The projection $\widehat{\boldsymbol{x}}$ (reconstruction) of a data point $\boldsymbol{x}$ on the subspace, which is spanned by columns of $\boldsymbol{G_m}$, is given by:

$$\begin{gathered} \widehat{\boldsymbol{x}} = \underbrace{\boldsymbol{G}_m \left(\boldsymbol{G}_m^T \boldsymbol{G}_m\right)^- \boldsymbol{G}_m^T}_{\boldsymbol{P_m}} \boldsymbol{x} = \underbrace{\boldsymbol{G}_m \left(\boldsymbol{G}_m^T \boldsymbol{G}_m\right)^-}_{\boldsymbol{R}} \boldsymbol{y_m} \\ \boldsymbol{R} = \boldsymbol{G}_m \left(\boldsymbol{G}_m^T \boldsymbol{G}_m\right)^- \end{gathered} \tag{2.1}$$

Here, $\boldsymbol{P}_m \in \mathbb{R}^{d \times d}$ is the projection matrix [2] on the *d*-dimensional subspace which is spanned by columns of $\boldsymbol{G}_m$. $\boldsymbol{R} \in \mathbb{R}^{d \times m}$ is the reconstruction matrix. It is already known (see reference [2], page 167) that $\boldsymbol{R} = \boldsymbol{G}_m \left(\boldsymbol{G}_m^T \boldsymbol{G}_m\right)^-$ is the generalized (pseudo) inverse of $\boldsymbol{G_m^T} = \boldsymbol{L_m^T D_k^{-1/2}\, U_k^T}$. Moreover, it is easy to check that the generalized (pseudo) inverse of $\boldsymbol{G_m^T}$ ($\boldsymbol{R}$) is given by:

$$\boldsymbol{R} = \left(\boldsymbol{G_m^T}\right)^- = \boldsymbol{U}_k \boldsymbol{D}_k^{\boldsymbol{1/2}} \boldsymbol{L}_m \tag{2.2}$$

$$\begin{gathered} \underbrace{\boldsymbol{L_m^T D_k^{-1/2}\, U_k^T}}_{\boldsymbol{G_m^T}} \underbrace{\boldsymbol{U}_k \boldsymbol{D}_k^{1/2} \boldsymbol{L_m}}_{\boldsymbol{R}} \underbrace{\boldsymbol{L_m^T D_k^{-1/2}\, U_k^T}}_{\boldsymbol{G_m^T}} = \underbrace{\boldsymbol{L_m^T D_k^{-1/2}\, U_k^T}}_{\boldsymbol{G_m^T}} \\ \underbrace{\boldsymbol{U}_k \boldsymbol{D}_k^{1/2} \boldsymbol{L}_m}_{\boldsymbol{R}} \underbrace{\boldsymbol{L_m^T D_k^{-1/2}\, U_k^T}}_{\boldsymbol{G_m^T}} \underbrace{\boldsymbol{U}_k \boldsymbol{D}_k^{1/2} \boldsymbol{L}_m}_{\boldsymbol{R}} = \underbrace{\boldsymbol{U}_k \boldsymbol{D}_k^{1/2} \boldsymbol{L}_m}_{\boldsymbol{R}} \end{gathered} \tag{2.3}$$

KL divergence of the reconstructed variable

The covariance matrices $\boldsymbol{S_{\widehat{x}|a}}$ and $\boldsymbol{S_{\widehat{x}|b}}$ of the reconstructed variable $\widehat{\boldsymbol{x}} = \boldsymbol{R y}_m$ at states *a* and *b* are respectively:

$$\begin{gathered} \boldsymbol{S_{\widehat{x}|a}} = \boldsymbol{R I R^T} = \boldsymbol{R R^T} \\ \boldsymbol{S_{\widehat{x}|b}} = \boldsymbol{R \Lambda}_m \boldsymbol{R^T} \end{gathered} \tag{2.4}$$

Where $\boldsymbol{I} \in \mathbb{R}^{m \times m}$ and $\boldsymbol{\Lambda}_m \in \mathbb{R}^{m \times m}$ are the covariance matrices of $\boldsymbol{y_m}$ at states *a* and *b*, respectively. Now we can show that the transformation matrix $\boldsymbol{G}_m = \boldsymbol{U}_k \boldsymbol{D}_k^{-1/2} \mathbf{L}_m$ simultaneously diagonalizes $\boldsymbol{S_{\widehat{x}|a}}\, \boldsymbol{S_{\widehat{x}|b}}$ into the matrices $\boldsymbol{I}$ and $\boldsymbol{\Lambda}_m$ respectively:

$$\begin{gathered} \boldsymbol{G_m^T S_{\widehat{x}|a} G_m} = \underbrace{\boldsymbol{L_m^T D_k^{-1/2}\, U_k^T}}_{\boldsymbol{G_m^T}} \underbrace{\boldsymbol{U}_k \boldsymbol{D}_k^{1/2} \boldsymbol{L}_m}_{\boldsymbol{R}}\, \boldsymbol{I}\, \underbrace{\boldsymbol{L_m^T D_k^{1/2} U_k^T}}_{\boldsymbol{R^T}} \underbrace{\boldsymbol{U}_k \boldsymbol{D}_k^{-1/2} \boldsymbol{L}_m}_{\boldsymbol{G_m}} = \boldsymbol{I} \\ \boldsymbol{G_m^T S_{\widehat{x}|b} G_m} = \underbrace{\boldsymbol{L_m^T D_k^{-1/2}\, U_k^T}}_{\boldsymbol{G_m^T}} \underbrace{\boldsymbol{U}_k \boldsymbol{D}_k^{1/2} \boldsymbol{L}_m}_{\boldsymbol{R}}\, \boldsymbol{\Lambda}_m\, \underbrace{\boldsymbol{L_m^T D_k^{1/2} U_k^T}}_{\boldsymbol{R^T}} \underbrace{\boldsymbol{U}_k \boldsymbol{D}_k^{-1/2} \boldsymbol{L}_m}_{\boldsymbol{G_m}} = \boldsymbol{\Lambda}_m. \end{gathered} \tag{2.5}$$

Therefore, the KL divergence of the reconstructed variable $\widehat{\boldsymbol{x}}$ is the same as the KL divergence of the corresponding latent variable $\boldsymbol{y}_m$.

S3. Derivation of equations 3.18 and 3.19

$$\begin{aligned}
&\left(\boldsymbol{W^T}(\boldsymbol{x_1}-\boldsymbol{x_2})\right)^{\boldsymbol{T}}\left(\boldsymbol{W^T}(\boldsymbol{x_1}-\boldsymbol{x_2})\right) = (\boldsymbol{x_1}-\boldsymbol{x_2})^{\boldsymbol{T}}\boldsymbol{W}\boldsymbol{W^T}(\boldsymbol{x_1}-\boldsymbol{x_2}) = (\boldsymbol{x_1}-\boldsymbol{x_2})^{\boldsymbol{T}}\boldsymbol{S_a^-}(\boldsymbol{x_1}-\boldsymbol{x_2}) \\
&= (\boldsymbol{x_1}-\boldsymbol{x_2})^{\boldsymbol{T}}\boldsymbol{W}\boldsymbol{W^T}(\boldsymbol{x_1}-\boldsymbol{x_2}) = (\boldsymbol{x_1}-\boldsymbol{x_2})^{\boldsymbol{T}}\boldsymbol{W}\underbrace{\boldsymbol{L}\boldsymbol{L^T}}_{\boldsymbol{I}}\boldsymbol{W^T}(\boldsymbol{x_1}-\boldsymbol{x_2}) \\
&= \left(\boldsymbol{G^T}(\boldsymbol{x_1}-\boldsymbol{x_2})\right)^{\boldsymbol{T}}\left(\boldsymbol{G^T}(\boldsymbol{x_1}-\boldsymbol{x_2})\right) = \sum_{\boldsymbol{i=1}}^{\boldsymbol{k}}\left[\boldsymbol{g_i^T}(\boldsymbol{x_1}-\boldsymbol{x_2})\right]^{\boldsymbol{2}}
\end{aligned} \tag{3.1}$$

$$\begin{aligned}
\left\langle(\boldsymbol{x}-\boldsymbol{\mu_a})^{\boldsymbol{T}}\boldsymbol{S_a^-}(\boldsymbol{x}-\boldsymbol{\mu_a})\right\rangle_b &= \left\langle\left(\boldsymbol{W^T}\boldsymbol{x}-\boldsymbol{W^T}\boldsymbol{\mu_a}\right)^{\boldsymbol{T}}\left(\boldsymbol{W^T}\boldsymbol{x}-\boldsymbol{W^T}\boldsymbol{\mu_a}\right)\right\rangle_b \\
&= \text{trace}\left\langle\left(\boldsymbol{W^T}\boldsymbol{x}-\boldsymbol{W^T}\boldsymbol{\mu_a}\right)\left(\boldsymbol{W^T}\boldsymbol{x}-\boldsymbol{W^T}\boldsymbol{\mu_a}\right)^{\boldsymbol{T}}\right\rangle_b \\
&= \text{trace}\underbrace{\left\langle\left(\boldsymbol{W^T}\boldsymbol{x}-\boldsymbol{W^T}\boldsymbol{\mu_b}\right)\left(\boldsymbol{W^T}\boldsymbol{x}-\boldsymbol{W^T}\boldsymbol{\mu_b}\right)^{\boldsymbol{T}}\right\rangle_b}_{\substack{\text{covariance matrix of}\\ \text{whitened points of state b}}} \\
&+ \text{trace}\left(\boldsymbol{W^T}\boldsymbol{\mu_b}-\boldsymbol{W^T}\boldsymbol{\mu_a}\right)\left(\boldsymbol{W^T}\boldsymbol{\mu_b}-\boldsymbol{W^T}\boldsymbol{\mu_a}\right)^{\boldsymbol{T}} \\
&= \text{trace}\left[\boldsymbol{W^T}\left\langle(\boldsymbol{x}-\boldsymbol{\mu_b})(\boldsymbol{x}-\boldsymbol{\mu_b})^{\boldsymbol{T}}\right\rangle_b\boldsymbol{W}\right] + \underbrace{\text{trace}\left(\boldsymbol{W^T}\Delta\Delta^{\boldsymbol{T}}\boldsymbol{W}\right)}_{\Delta^{\boldsymbol{T}}\boldsymbol{W}\boldsymbol{W^T}\Delta} \\
&= \underbrace{\text{trace}\left(\boldsymbol{W^T}\boldsymbol{S_b}\boldsymbol{W}\right)}_{\sum\lambda_i} + \underbrace{\Delta^{\boldsymbol{T}}\boldsymbol{S_a^-}\Delta}_{\substack{\text{Mahalanobis}\\ \text{distance}}}
\end{aligned} \tag{3.2}$$

S4. Log density ratio as a sufficient statistic

The relationship between the log density ratio and the perturbation is given by equation (2.1) in the main text:

$$\ln \frac{P_b(\boldsymbol{x})}{P_a(\boldsymbol{x})} = \Delta F(a \to b) - U_p(\boldsymbol{x}). \tag{4.1}$$

For a new macroscopic state λ (e.g. an intermediate state between *a* and *b* during the course of an alchemical free energy calculation we obtain by using the log density ratio as a sufficient statistic $T(\boldsymbol{x}) = \ln \frac{P_b(\boldsymbol{x})}{P_a(\boldsymbol{x})}$ in equation (2.5) of the cumulant generating function of the main text:

$$\begin{aligned}
\psi(\lambda) &= \ln \left[\int d\boldsymbol{x} P_a(\boldsymbol{x}) \exp\big(\lambda T(\boldsymbol{x})\big) \right] \\
&= \ln \left[\int d\boldsymbol{x} P_a(\boldsymbol{x}) \exp\Big(\lambda \Delta F(a \to b) - \lambda U_p(\boldsymbol{x})\Big) \right] \\
&= \lambda \Delta F(a \to b) + \underbrace{\ln \left[\int d\boldsymbol{x} P_a(\boldsymbol{x}) \exp\Big(-\lambda U_p(\boldsymbol{x})\Big) \right]}_{-\Delta F(a \to \lambda)} \\
&= \lambda \Delta F(a \to b) - \Delta F(a \to \lambda)
\end{aligned} \tag{4.2}$$

Therefore, using the log density ratio as a sufficient statistic turns the cumulant generating function into the relative free energy, $\psi(\lambda) = \lambda \Delta F(a \to b) - \Delta F(a \to \lambda)$.

S5. Details of Molecular Dynamics simulations

The molecular dynamic simulations were started using the following X-ray structures for complexes of HIV-1 protease with different small molecule inhibitors:

Protein-Ligand complex	PDB code
Wild-type HIV-1 Protease with Tipranavir	1D4Y
Wild-type HIV-1 Protease with Saquinavir	3OXC
HIV-1 mutant I50V with Saquinavir	3CYX

The free (unbound) start structures of the protein were modelled based on the structures of the complexes and by removing the ligand and relaxing the protein via 10 ns of MD simulations. The simulations of the free structures and the complexes were run for 150 and 50 ns, respectively, using the Gromacs 4.6.5 simulation package [6]. The conformational data were collected every 1 ps. The force field Amber99sb [7] was used for the protein while the general Amber force field [8] was used for the ligands. Antechamber [9] was used to obtain the force-field parameters and AM1-BCC was used to determine the partial atomic charges for the ligands. The solvent was represented using the TIP3P water model [10]. The systems were solvated in a rectangular box and the water extended at least 1.2 nm beyond the solute surface. Periodic boundary conditions were used. 500 steps of steepest-descent energy minimization followed by 1000 steps of conjugate gradient optimization were used to energetically minimize each system. The systems were then equilibrated using a 100 ps molecular dynamics simulation with harmonic position restraints using a force constant of 1000 kJ $mol^{-1}nm^{-2}$ for the heavy atoms in the solutes for each system. The production simulations were performed using a leap-frog integrator [11] and a time step of 2.0 fs. Simulations were performed in the isothermal-isobaric ensemble (1 atm and 300 K) by maintaining the temperature and the pressure through a Berendsen bath [12]. The van-der-Waals interactions and short-range electrostatic interactions were computed using a cutoff of 1.2 nm. The long-range electrostatic interactions were computed using the particle-mesh Ewald method [13].

S6. Supplementary Movies

Movie representation of the enhanced and restricted conformational fluctuations of HIV-1 protease upon binding the inhibitor Tipranavir which are presented in figure 5 of the main text. The conformations are reconstructed around the average conformation from the *33*-dimensional latent variable; see equation (3.18) of the main text. The latent variable is interpolated around its average along the selected generalized eigenvectors. The conformation of the ligand is taken from the experimental structure.

Files "increased_top.mpg" and "increased_side.mpg" show, respectively, top and side views of the enhanced conformational fluctuations around the average structure of the bound state along the 33 eigenvectors with the highest generalized eigenvalues ($\lambda_i > 10$).

Files "decreased_top.mpg" and "decreased_side.mpg" show, respectively, top and side views of the highly restricted conformational fluctuations upon association to the ligand. Conformational fluctuations are represented around the average structure of the free state along the 33 eigenvectors with the smallest generalized eigenvalues. These latter fluctuations are highly restricted upon the association ($\lambda_i < 0.009$).

S7. Example of the performance of GPA for superimposing the conformations from an MD ensemble

GPA rotates and translates the rigid conformations in an ensemble such as to minimize the sum of squared distances between all pairs of conformations in the ensemble. GPA performs the following steps:

1. Choose an arbitrary conformation from the ensemble as the reference.
2. Superimpose each of the conformations in the ensemble to the reference by rotating and translating the conformation such as to minimize the squared distance between the conformation and the reference.
3. Compute the average conformation of the resulting ensemble and use it as a new reference
4. Repeat steps 2 and 3 until the convergence of the average conformation (the average structure).

The table below shows an example of applying GPA fitting to superimpose the snapshots from an MD simulation of the complex between wild-type HIV-1 protease and tipranavir. The algorithm converged after three cycles when an RMSD threshold of 0.00001 Å was used. Using the GPA average structure as a reference for superimposing the conformations results in around 30% reduction of the average RMSD between the conformations and the reference when compared to using the starting MD structure as a reference for superimposing the conformations.

Iteration	**RMSD (Å) between the new (computed) average structure and the reference structure**	**Average RMSD (Å) between the conformations and the reference**
1	1.340040	1.870725
2	0.000234	1.310567
3	0.000008	1.310567

S8. Supplementary figure SF1

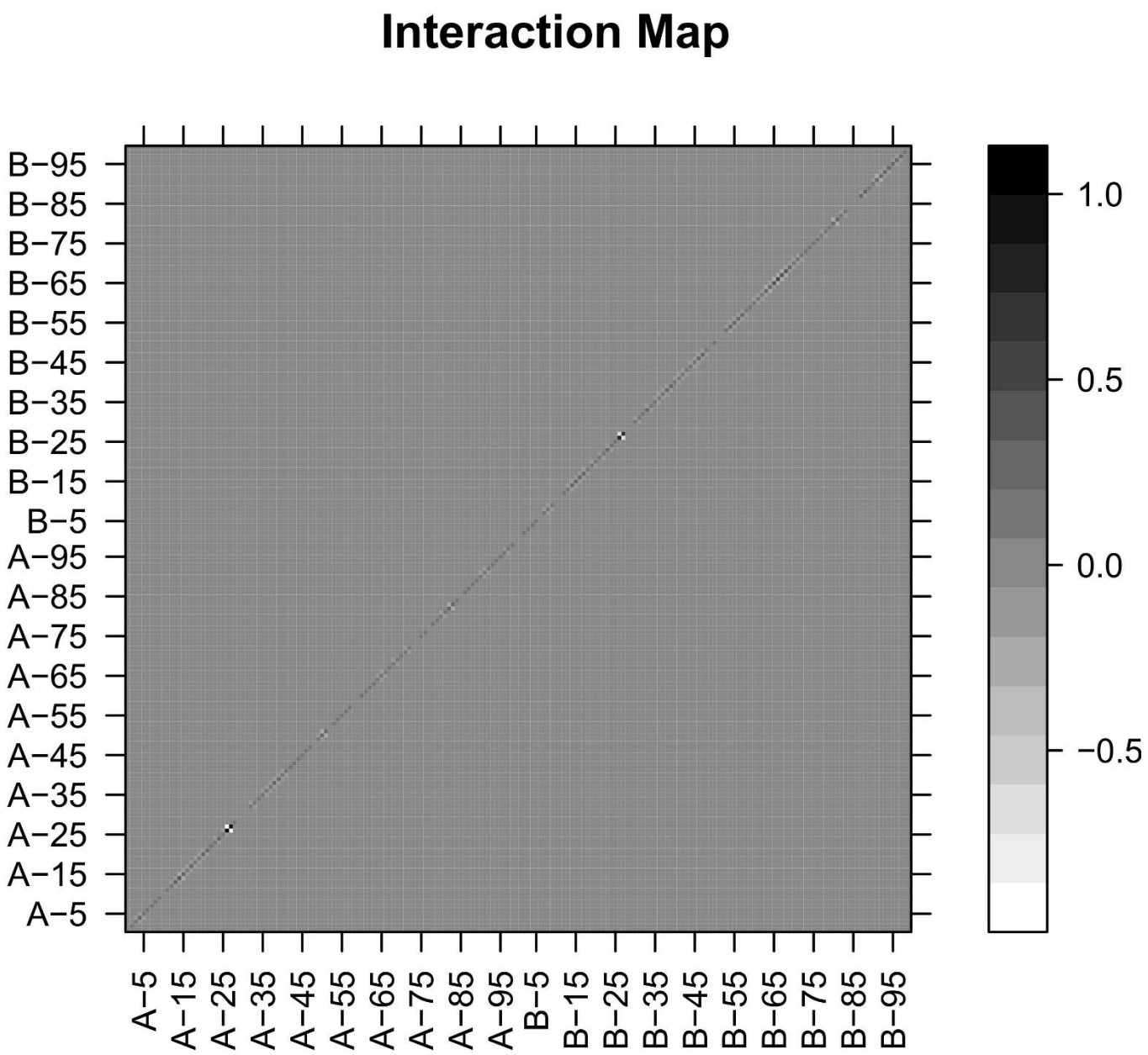

Figure S1. Conformational interaction map between the residues of HIV-1 protease upon binding its inhibitor Tipranavir. The contributions are concentrated around the diagonal elements indicating the important local conformational changes and the propagation of the conformational changes through the neighbor residues.

Supplementary references

(1) Kullback, S. Information Theory and Statistics, Revised edition.; Dover Publications: Mineola, N.Y, 1997.
(2) Harville, D. A. Matrix Algebra From a Statistician's Perspective; Springer Science & Business Media, 2008.
(3) Rao, C. R. Linear Statistical Inference and Its Applications; Wiley, 2002.
(4) Fukunaga, K. Introduction to Statistical Pattern Recognition, Second Edition, 2 edition.; Academic Press: Boston, 1990.
(5) Mardia, K. V.; Kent, J. T.; Bibby, J. M. Multivariate Analysis; Academic Press, 1979.
(6) Van Der Spoel, D.; Lindahl, E.; Hess, B.; Groenhof, G.; Mark, A. E.; Berendsen, H. J. C. GROMACS: Fast, Flexible, and Free. J. Comput. Chem. **2005**, 26 (16), 1701–1718.
(7) Hornak, V.; Abel, R.; Okur, A.; Strockbine, B.; Roitberg, A.; Simmerling, C. Comparison of Multiple Amber Force Fields and Development of Improved Protein Backbone Parameters. Proteins Struct. Funct. Bioinforma. **2006**, 65 (3), 712–725.
(8) Wang, J.; Wolf, R. M.; Caldwell, J. W.; Kollman, P. A.; Case, D. A. Development and Testing of a General Amber Force Field. J. Comput. Chem. **2004**, 25 (9), 1157–1174.
(9) Wang, J.; Wang, W.; Kollman, P. A.; Case, D. A. Automatic Atom Type and Bond Type Perception in Molecular Mechanical Calculations. J. Mol. Graph. Model. **2006**, 25 (2), 247–260.
(10) Jorgensen, W. L.; Chandrasekhar, J.; Madura, J. D.; Impey, R. W.; Klein, M. L. Comparison of Simple Potential Functions for Simulating Liquid Water. J. Chem. Phys. **1983**, 79 (2), 926–935.
(11) Goga, N.; Rzepiela, A. J.; de Vries, A. H.; Marrink, S. J.; Berendsen, H. J. C. Efficient Algorithms for Langevin and DPD Dynamics. J. Chem. Theory Comput. **2012**, 8 (10), 3637–3649.
(12) Berendsen, H. J. C.; Postma, J. P. M.; Gunsteren, W. F. van; DiNola, A.; Haak, J. R. Molecular Dynamics with Coupling to an External Bath. J. Chem. Phys. **1984**, 81 (8), 3684–3690.
(13) Essmann, U.; Perera, L.; Berkowitz, M. L.; Darden, T.; Lee, H.; Pedersen, L. G. A Smooth Particle Mesh Ewald Method. J. Chem. Phys. **1995**, 103 (19), 8577–8593.