ADAPTING MOLECULAR FOUNDATION MODELS FOR MACROCYCLIC PEPTIDE AFFINITY PREDICTION VIA ROBUST PROXY REHEARSAL

Anonymous authors
Paper under double-blind review

ABSTRACT

Adapting molecular foundation models to macrocyclic peptide affinity prediction requires improving target-domain performance without sacrificing their existing small-molecule capabilities. When the original pretraining corpus is unavailable, public affinity measurements can serve as a source proxy, but their protein-family composition and coverage may differ from the true source distribution. We study retention under this imperfect source information. We show that reducing absolute worst-family risk does not guarantee retention, whereas controlling degradation relative to the pretrained model characterizes worst-case retention over arbitrary family proportions under full coverage and matched within-family distributions. A conditional bound further isolates the effects of finite sampling, within-family mismatch, and missing coverage. Motivated by this analysis, we introduce Robust Proxy Rehearsal (RPR), which combines target supervision with reference-relative source penalties and adaptively emphasizes protein families that degrade during adaptation. We instantiate RPR with Boltz-2 and evaluate macrocycle adaptation, small-molecule retention, and transfer to two external peptide benchmarks. In our primary comparison, RPR improves over uniform replay, increasing Spearman ρ from 0.508 to 0.611 on the source domain and from 0.216 to 0.312 on the target domain, while also improving both external benchmarks. Additional runs show that these gains vary across hyperparameters, delineating when reference-relative rehearsal can improve the adaptation–retention trade-off under an imperfect proxy.

1 INTRODUCTION

Binding-affinity prediction helps prioritize protein–ligand pairs for experimental validation in drug discovery (Passaro et al., 2025). Macrocyclic peptides are an attractive modality because they can combine antibody-like affinity and selectivity with the tissue penetration and tunable pharmacokinetic properties associated with smaller therapeutics. They extend the chemical space of conventional small molecules, can engage traditionally challenging protein surfaces, and offer opportunities for oral drug development (Driggers et al., 2008; Doak et al., 2014; Vinogradov et al., 2019). These advantages have drawn growing attention from the pharmaceutical industry. Molecular foundation models are a natural starting point for predicting their affinities. The AlphaFold family (Jumper et al., 2021) has attracted substantial attention for biomolecular structure prediction, but the latest AlphaFold 3 (Abramson et al., 2024; Google DeepMind, 2024) provides only an inference pipeline for structure prediction, without a fine-tuning pipeline or an affinity head. Boltz-2 (Passaro et al., 2025), as a representative open-weight alternative, couples structure prediction with a learned affinity predictor and releases its model weights, and an open-source framework supports fine-tuning its affinity head (Amini et al., 2026). However, its affinity training focuses on small molecules with at most 50 heavy atoms, so extending it to the larger macrocyclic peptides requires further fine-tuning.

However, because fine-tuning modifies the predictor originally trained on small molecules, updates that improve peptide affinity prediction can degrade its small-molecule performance—a form of catastrophic forgetting (McCloskey & Cohen, 1989; Kirkpatrick et al., 2017). Such forgetting is costly. Small molecules remain the dominant modality in drug discovery, and programs may consider both peptide and small-molecule candidates against the same protein target, motivating a single predictor that remains reliable on both modalities. Moreover, knowledge learned from small molecules may

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Table 1: **All methods use labeled target-domain data but differ in whether and how they use source-domain information.** Source-domain information comprises data or information derived from such data (✓: required; ×: not required). Pretrained parameters alone do not count as source-domain information. No method has access to Boltz-2's actual affinity corpus (Passaro et al., 2025), which is curated and not redistributed. FT: target-only fine-tuning; L^2-SP: L^2 starting point; EWC: elastic weight consolidation; OD: output distillation; UR: uniform replay; AGEM: averaged gradient episodic memory; RPR: robust proxy rehearsal. See Section 1 for detailed descriptions of the methods.

Method	Source info.?	Form (if required)	Mechanism
FT (Amini et al., 2026)	×	—	Fine-tune on target only
L^2-SP (Xuhong et al., 2018)	×	—	Penalise $\|\theta - \theta_0\|$
EWC (Kirkpatrick et al., 2017)	✓	Fisher matrix from proxy data	Penalise $\|\theta - \theta_0\|_F$ with diagonal Fisher F
OD (Li & Hoiem, 2017)	✓	teacher outputs on proxy data	Match teacher outputs
UR (Chaudhry et al., 2019)	✓	proxy data	Draw half of each batch from proxy
AGEM (Chaudhry et al., 2018)	✓	gradients on proxy data	Project conflicting gradients
RPR (ours)	✓	grouped proxy data	Reference-relative worst-group rehearsal

provide transferable protein–ligand interaction patterns that are difficult to learn from scarce peptide data alone. We therefore study a joint adaptation and retention problem: improving affinity prediction for macrocyclic peptides while maintaining the pretrained model's capabilities for small molecules.

Prior Boltz-2 adaptations have addressed protein–protein affinity (King et al., 2025) and project-specific small-molecule potency (Amini et al., 2026), but focus primarily on the new task. Our joint objective also requires retaining the original capability, which target-only fine-tuning (FT) does not provide. Continual learning offers two families of such methods (Table 1). Parameter-space methods keep weights near pretrained values: L^2 starting-point regularization (L^2-SP) (Xuhong et al., 2018) penalizes the distance to them, and elastic weight consolidation (EWC) (Kirkpatrick et al., 2017) weights this penalty by a Fisher matrix estimated on source data. Data-dependent methods use source examples directly: output distillation (OD) (Li & Hoiem, 2017) matches a frozen teacher's predictions on them, uniform replay (UR) (Chaudhry et al., 2019) interleaves them with target examples, and averaged gradient episodic memory (AGEM) (Chaudhry et al., 2018) projects out updates that conflict with their gradients. Each family has a limitation in our setting: parameter constraints limit how far weights move rather than how much predictions degrade, while data-dependent methods safeguard only the portion of the source domain that their available examples can faithfully represent.

The ideal source examples would therefore be Boltz-2's original affinity training data, which by construction represent the source domain, but two obstacles prevent their use. First, the curated data cannot be directly downloaded: although Boltz-2 describes its affinity curation in detail (Passaro et al., 2025), neither its code repository nor its Hugging Face organization releases it. Second, rebuilding the described corpus exceeds the compute and storage of most academic laboratories. It comprises about 4.6M protein–ligand examples (1.2M measured affinities and 3.4M binary labels), each needing a co-folding pass to produce the trunk features the affinity head consumes. At a measured 20 seconds per complex on an A100 GPU, this amounts to over 25,000 GPU-hours (more than four months on a dedicated eight-GPU node) and about 1 PB of uncropped embeddings, before multiple-sequence alignments add a further $\geq$1,800 GPU-hours (Appendix D). Therefore, in practice we can only rehearse a tractable proxy built from public affinity databases, whose protein coverage and composition may differ from those of the original corpus. This raises our central question: *how should an imperfect proxy be used during fine-tuning to retain the model's small-molecule capability?*

Our key idea is to use the proxy not to preserve its average performance, but to identify and protect the parts of the source capability that deteriorate most during adaptation. We group proxy examples into protein families, operationalized as sequence-similarity clusters, and measure each family's degradation relative to the frozen pretrained model. This reference-relative view is crucial: unlike group distributionally robust optimization (group DRO) (Sagawa et al., 2019), which minimizes the largest absolute group loss, it distinguishes deterioration caused by fine-tuning from groups that were already difficult before adaptation (Figure 1b). Building on reference-relative retention and minimax regret (Li et al., 2024; Agarwal & Zhang, 2022), we show in Section 2 that, when the proxy covers every source family and the within-family distributions match, the largest family-wise degradation exactly characterizes retention under unknown family proportions; we also formalize how proxy–source mismatch limits this guarantee. This motivates Robust Proxy Rehearsal (RPR;

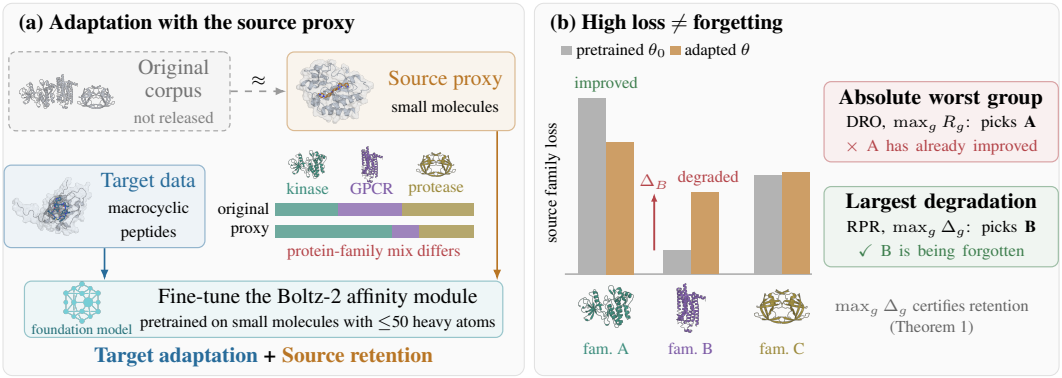


Figure 1: **Adapting affinity prediction while retaining source-domain capability.** (a) Boltz-2's curated affinity corpus is not publicly released, so adaptation to macrocyclic peptides (e.g., a PD-L1 inhibitor, PDB 5O45) rehearses a reconstructed small-molecule proxy (e.g., Abl kinase with imatinib, PDB 1IEP) whose protein-family mix can differ from the original source. (b) Illustrative example: the family with the highest absolute loss (A) has improved, whereas a low-loss family (B) has degraded. An absolute worst-group objective targets A; the largest degradation relative to the pretrained model flags B, and Theorem 1 shows that this quantity controls retention under unknown family proportions. Figure 2 describes the RPR procedure. Family icons: kinase (1IEP), GPCR (4EIY), protease (1HVR).

Figure 2), which interleaves target and proxy examples while adaptively emphasizing protein families whose losses have increased most relative to the pretrained reference. Because reference losses are cached once, RPR needs neither the pretraining corpus nor a second frozen model during adaptation.

In summary, we make three contributions. First, a *retention criterion*: we formulate affinity-model adaptation when the pretraining corpus is unavailable, and establish reference-relative worst-family degradation as the retention objective under unknown family proportions, in contrast to the absolute worst-group risk of group DRO. Second, a *proxy-transfer analysis*: we characterize what breaks when only an imperfect proxy is available, separating finite-sample uncertainty, within-family mismatch and missing coverage, and identifying which terms are unobservable without the original corpus. Third, a *practical instantiation*: we introduce RPR, which combines target learning with reference-relative source penalties and adaptive family weighting, and study empirically when this criterion helps, across macrocyclic peptides, small molecules and two external peptide benchmarks under matched target exposure, against fine-tuning, parameter regularization, distillation, replay and gradient projection.

2 Problem Formulation and Retention Analysis

Adaptation should improve macrocycle prediction without losing what the pretrained model already does well on small molecules. Because the original corpus cannot be used (Section 1), the only source evidence is a proxy whose family composition and coverage relative to that corpus are uncertain (Figure 1a). We formalize this setting as target-risk minimization under a constraint on source degradation relative to the pretrained model, with protein families as units, and ask whether reducing the loss of the hardest source family suffices and which quantity characterizes retention when family proportions are unknown. Theorem 1 answers both: reducing the largest absolute group risk is not sufficient, whereas the largest degradation relative to the pretrained model exactly characterizes worst-case retention. This result motivates the training signal of RPR (Section 3); Proposition 2 in Appendix C.3 specifies what else is needed to carry proxy retention to the unavailable original source.

Problem formulation. Let P_S and P_T denote the source small-molecule and target macrocycle distributions. An observation Z contains a protein–ligand input and its affinity supervision, including measurement qualifiers when relevant. For predictor f_θ and a fixed loss $\ell_\theta(Z)$, define

$$R_P(\theta) = \mathbb{E}_{Z\sim P}[\ell_\theta(Z)], \qquad \Delta_P(\theta) = R_P(\theta) - R_P(\theta_0). \tag{1}$$

The reference θ_0 is the fixed pretrained model, following the reference-relative view of retention in Li et al. (2024). Negative Δ_P denotes improvement, and positive Δ_P denotes degradation. For a

prescribed tolerance $\epsilon \geq 0$, the population goal is

$$\min_{\theta \in \Theta} R_{P_T}(\theta) \quad \text{subject to} \quad \Delta_{P_S}(\theta) \leq \epsilon. \tag{2}$$

Strict target improvement additionally requires $\Delta_{P_T}(\theta) < 0$. The reference is feasible when $\theta_0 \in \Theta$, but the existence of a strictly improving feasible model is not automatic. A loss-risk guarantee does not by itself guarantee improvement in every reported metric, such as Spearman correlation or AUC.

We use protein families to denote groups of related proteins. Each protein–ligand observation is assigned to the group containing its protein. These groups are fixed before training and serve as the units for assessing source retention; Appendix E specifies their construction. Let $g = 1, \ldots, G$ index the groups covered by the proxy, with true source conditional laws P_g and proxy conditional laws Q_g. Write $R_g = R_{P_g}$ and $\Delta_g = \Delta_{P_g}$. The empirical proxy contains samples from Q_g, not access to P_g, and the family proportions in the source need not equal their proxy proportions. We first consider full proxy coverage, whereas Appendix C.3 allows an unobserved component of the original source.

Is improving the hardest family enough? Because the source family proportions are unknown, a natural robust strategy is to control the worst family. Group DRO (Sagawa et al., 2019) does so by minimizing the largest *absolute* group risk. Applied to rehearsal, it yields

$$\min_{\theta} \max_{g} \widehat{R}_{Q_g}(\theta), \qquad \min_{\theta} \left\{ \widehat{R}_T(\theta) + \lambda \max_{g} \widehat{R}_{Q_g}(\theta) \right\}, \tag{3}$$

where hats denote empirical risks and $\lambda \geq 0$. Both objectives rank families by how hard they are, not by how much they have changed. Consider two families whose source losses move from 0.8 to 0.6 (family A) and from 0.1 to 0.3 (family B). The absolute worst-group objective focuses on A, which remains the hardest family although it improved, and overlooks B, which has lost previously strong performance. If B accounts for 90% of the source distribution, the largest absolute loss falls from 0.8 to 0.6 while the average source loss rises from 0.17 to 0.33. This example separates a family being *difficult now* from becoming *worse than before* and motivates the first part of the result stated below.

Retention under unknown family proportions. Let $\Pi_G = \{p \in \mathbb{R}^G_+ : \sum_g p_g = 1\}$ and $P_p = \sum_g p_g P_g$. Here the group conditional distributions are fixed; only their proportions vary.

Theorem 1 (Absolute group risk and reference-relative retention)**.** *The following statements hold.* (i) *There exist two group distributions P_1, P_2, a bounded squared loss, reference and candidate predictors f_{θ_0}, f_θ, and a mixture $P_S = p_1 P_1 + p_2 P_2$ such that*

$$\max_{g \in \{1,2\}} R_g(\theta) < \max_{g \in \{1,2\}} R_g(\theta_0), \qquad \Delta_{P_S}(\theta) > 0. \tag{4}$$

Thus, reducing the maximum absolute group risk does not generally guarantee source retention. (ii) *For any reference and candidate predictors with finite group risks,*

$$\sup_{p \in \Pi_G} \Delta_{P_p}(\theta) = \max_{1 \leq g \leq G} \Delta_g(\theta). \tag{5}$$

Consequently, $\Delta_{P_p}(\theta) \leq \epsilon$ for every $p \in \Pi_G$ if and only if $\Delta_g(\theta) \leq \epsilon$ for every group g.

The example above makes both parts concrete. Families A and B change by $\Delta_A = -0.2$ and $\Delta_B = +0.2$. Part (i) is the example itself: the largest absolute risk falls from 0.8 to 0.6, yet the source degrades whenever B is common. Part (ii) quantifies how badly. At the stated proportions the source degradation is $0.1\Delta_A + 0.9\Delta_B = 0.16$, and over all possible proportions it is largest, at $0.2 = \max_g \Delta_g$, when the source consists of B alone. Tracking $\max_g \Delta_g$ therefore flags B whatever the unknown proportions, whereas tracking $\max_g R_g$ flags A, already improved. Part (i) does not claim that absolute-risk optimization always causes forgetting, only that it cannot certify retention.

Why subtract the reference risk? Subtracting the pretrained risk protects established capability while leaving every family free to improve. The identity is exact when proportions range over the full simplex. If reliable knowledge restricts them to a smaller set $\mathcal{W} \subseteq \Pi_G$, the supremum over $\mathcal{W}$ is less conservative, since gains in some families can then offset losses in others. Unlike minimax regret (Agarwal & Zhang, 2022), which compares each group with its own optimal predictor, the reference in our setting is the pretrained model whose small-molecule capabilities are to be retained.

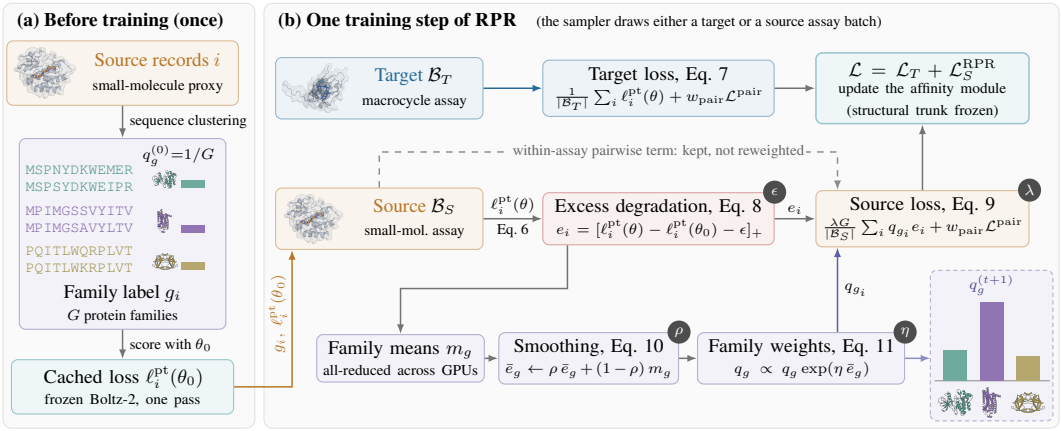


Figure 2: **Robust Proxy Rehearsal (RPR).** (a) Before training, source-proxy records are clustered into protein families with uniform initial weights, and each record's pointwise loss under the frozen pretrained model is cached once, so training needs neither the original corpus nor a second model. (b) Each step draws a target or a source assay batch. Target batches keep the original affinity loss. For source batches, each record's loss is compared with its cached reference to give the positive excess degradation e_i; per-family means are all-reduced across GPUs, smoothed, and used to update the family weights q, which reweight e_i in the source loss, so a family that keeps degrading (here GPCR) gains weight. The pairwise term is not reweighted. Dark circles indicate where ϵ, λ, ρ, and η enter.

Scope of the theorem. Theorem 1 characterizes which predictors retain source performance, not how to find them. Controlling a fixed average does not in general guarantee this, although uniform replay may still satisfy every family constraint. It therefore motivates focusing rehearsal on the high-degradation families without prescribing a reweighting scheme or step size η. The theorem also concerns the true source-family conditionals, whereas training samples the proxy laws Q_g and the proxy may not cover the entire source. Proposition 2 in Appendix C.3 shows that transferring proxy retention to the source additionally incurs finite-sample uncertainty, within-family distribution mismatch, and uncovered source mass. The latter two are unobservable without the original corpus, so the proposition delineates the proxy objective rather than certifying retention in our experiments.

3 Robust Proxy Rehearsal

RPR turns the family-wise, reference-relative criterion of Theorem 1 into a training procedure (Figure 2). The criterion motivates the method rather than being optimized exactly: we pass from the population quantity $\max_g \Delta_g$ to a per-record surrogate, and then to a stochastic estimator of that surrogate driven by the stream sampler. We state each step below and quantify both gaps. We instantiate it with a pretrained Boltz-2 affinity predictor f_{θ_0}, macrocyclic peptides as target and the small-molecule proxy as source. We freeze the structural backbone and fine-tune the affinity modules. The fixed pretrained reference supplies a cached pointwise loss for every source record, so training needs neither the original corpus nor a second reference forward pass. All methods start from θ_0 with the same target data; Table 1 summarizes how each uses the source information, and in what form.

Reference-relative rehearsal objective. For a record i, write the pointwise part of the Boltz-2 affinity loss as

$$\ell_i^{\mathrm{pt}}(\theta) = w_{\mathrm{abs}}\ell_i^{\mathrm{abs}}(\theta) + w_{\mathrm{bin}}\ell_i^{\mathrm{bin}}(\theta), \tag{6}$$

and let $\mathcal{L}^{\mathrm{pair}}(\theta;\mathcal{B})$ denote the within-assay pairwise term for batch $\mathcal{B}$. On a target batch $\mathcal{B}_T$, we retain the original affinity objective

$$\mathcal{L}_T(\theta;\mathcal{B}_T) = \frac{1}{|\mathcal{B}_T|}\sum_{i\in\mathcal{B}_T} \ell_i^{\mathrm{pt}}(\theta) + w_{\mathrm{pair}}\mathcal{L}^{\mathrm{pair}}(\theta;\mathcal{B}_T). \tag{7}$$

Table 2: **Target and source proxy streams.** Counts after the filtering of Appendix D, before the family-disjoint split of Section 4. Target ligands contain a ≥ 12-atom ring, source ligands span 15–45 heavy atoms, and protein families are MMseqs2 clusters at 30% identity. Storage covers the per-record MSAs and uncropped embeddings for training; MSA search also needs a 1.4 TB local ColabFold database. Pro.: protein; Lig.: ligand; Emb.: embedding.

	Pro.–Lig. Pairs	Pro.	Lig.	Pro. Families	MSA (GB)	Emb. (TB)
Target (macrocyclic peptides, >50 heavy atoms)	19,959	837	10,419	355	15.2	3.3
Source proxy (small molecules, ≤ 50 heavy atoms)	19,993	518	16,974	248	12.1	3.9
Shared between streams	—	274	0	248	—	—
Total	39,952	1,081	27,393	355	27.3	7.2

For a source record i in family g_i, we instead measure positive degradation relative to the cached reference loss,

$$e_i(\theta) = \left[\ell_i^{\mathrm{pt}}(\theta) - \ell_i^{\mathrm{pt}}(\theta_0) - \epsilon\right]_+, \tag{8}$$

where $\epsilon \geq 0$ is a retention tolerance. Given family weights $q \in \Pi_G$, the loss on a source batch $\mathcal{B}_S$ is

$$\mathcal{L}_S^{\mathrm{RPR}}(\theta; \mathcal{B}_S, q) = \frac{\lambda G}{|\mathcal{B}_S|} \sum_{i \in \mathcal{B}_S} q_{g_i} e_i(\theta) + w_{\mathrm{pair}} \mathcal{L}^{\mathrm{pair}}(\theta; \mathcal{B}_S). \tag{9}$$

The factor G makes uniform initialization, $q_g = 1/G$, recover unit source weight before scaling by λ. Equation 8 penalizes each record separately, whereas Theorem 1 constrains the family mean Δ_g, in which improvements offset degradations. The per-record form is the more conservative of the two: the pointwise bound $\ell_i^{\mathrm{pt}}(\theta) - \ell_i^{\mathrm{pt}}(\theta_0) \leq \epsilon + e_i(\theta)$ gives $\Delta_g \leq \epsilon + \mathbb{E}_g[e_i(\theta)]$ in expectation; driving the penalty to zero therefore certifies $\Delta_g \leq \epsilon$, but a small positive penalty does not, and the source pairwise term, which is retained without family reweighting, carries no such bound. Alternating target and source assay batches gives a stochastic objective that combines Equations 7 and 9, with their frequencies determined by the stream sampler. For the pairwise term, the independent unit is an assay block sampled with weights matching the analyzed family risk, because the quadratically many within-assay comparisons are not independent.

Adaptive family weights and optimization. At step t, let $\mathcal{I}_g^t$ contain the source records from family g observed across workers. For an observed family, we compute its batch mean excess $m_g^t = |\mathcal{I}_g^t|^{-1} \sum_{i \in \mathcal{I}_g^t} e_i$ and update an exponential moving average

$$\bar{e}_g^t = \rho \bar{e}_g^{t-1} + (1 - \rho) m_g^t. \tag{10}$$

The first observation initializes $\bar{e}_g$ directly. We then apply an exponentiated-gradient update to observed families and renormalize,

$$q_g^{t+1} = \frac{q_g^t \exp\big(\eta\, \mathbf{1}\{\mathcal{I}_g^t \neq \varnothing\} \bar{e}_g^t\big)}{\sum_{h=1}^G q_h^t \exp(\eta\, \mathbf{1}\{\mathcal{I}_h^t \neq \varnothing\} \bar{e}_h^t)}. \tag{11}$$

Families with larger smoothed degradation receive more weight in the source objective. The coefficient λ sets the source-retention strength, η controls adaptation of the family weights, and ρ controls smoothing. Setting $\eta = 0$ keeps q uniform while retaining the relative excess and the pairwise term.

Relative to uniform replay, RPR changes which source errors are prioritized, not whether source data are used: it upweights observed loss increases rather than average source difficulty. Relative to absolute group DRO, it compares each source record with the pretrained reference. Because the per-record positive-part penalty and online family weights only approximate the largest family degradation, we assess retention empirically: Section 4 isolates these components by comparing uniform replay, absolute and reference-relative objectives, fixed and adaptive family weights, and the source sampler (Table 4), with all remaining settings held constant. Appendix C.4 gives a joint sufficient condition for target improvement and source retention given additional target calibration data, although neither feasibility nor a strict improvement is guaranteed for every adaptation problem.

4 Experiments

Experimental setup. Both data streams follow Boltz-2's own affinity curation, so that each departs from its training data only where intended (Appendix D). The target stream collects macrocyclic

Table 3: **Main results.** All methods start from Boltz-2 (Passaro et al., 2025); the first row is the released model. Fam. ρ is the median Spearman correlation over test families with at least ten records (36 source and 29 target families); the other metrics use all test records. External columns give ρ on PpI_ba, nc-PpI_ba and both combined (All). In each column, bold marks the best of the eight rows.

	Small-molecule source ($n = 1909$)					Macrocycle target ($n = 2379$)					External ($n = 1533$), $\rho \uparrow$		
Method	Fam. $\rho \uparrow$	$\rho \uparrow$	$r \uparrow$	RMSE↓	AUC↑	Fam. $\rho \uparrow$	$\rho \uparrow$	$r \uparrow$	RMSE↓	AUC↑	PpI_ba	nc-PpI_ba	All
Pretrained Boltz-2 (no FT)	0.356	0.455	0.450	1.308	0.738	0.050	0.283	0.293	1.512	0.625	0.200	0.190	0.157
FT (Amini et al., 2026)	0.249	0.467	0.440	1.357	0.723	0.144	0.237	0.233	1.591	0.613	0.273	0.313	0.300
L^2-SP (Xuhong et al., 2018)	0.261	0.289	0.324	1.527	0.664	0.151	0.161	0.173	1.684	0.577	0.194	0.232	0.202
EWC (Kirkpatrick et al., 2017)	0.340	0.284	0.321	1.523	0.656	0.219	0.251	0.255	1.653	0.623	0.179	0.219	0.170
OD (Li & Hoiem, 2017)	0.359	0.400	0.409	1.365	0.721	0.165	0.240	0.262	1.526	0.604	0.246	0.286	0.229
UR (Chaudhry et al., 2019)	0.307	0.508	0.477	1.335	0.759	0.114	0.216	0.229	1.559	0.593	0.152	0.261	0.181
AGEM (Chaudhry et al., 2018)	0.292	0.534	0.488	1.526	0.768	0.137	0.149	0.177	1.730	0.558	0.214	0.291	0.237
RPR (ours)	**0.485**	**0.611**	**0.593**	**1.144**	**0.814**	**0.255**	**0.312**	**0.324**	**1.497**	**0.640**	**0.316**	**0.323**	**0.317**

Table 4: **Ablation study.** Each RPR row above the last changes one component of the full method in the last row, which is RPR of Table 3; all entries are Spearman ρ. q signal: family statistic driving the weight update; Sampler: source sampling scheme; Loss: rehearsed per-record term. Fam. ρ and All are as in Table 3; bold marks the best per column. Abbreviations: rel., relative excess; abs., absolute loss; unif., fixed uniform weights; —, no family weighting; Src. pw., the source's pairwise loss term.

					Source		Target		External		
Method	q signal	Sampler	Loss	Src. pw.	Fam. ρ	ρ	Fam. ρ	ρ	PpI_ba	nc-PpI_ba	All
UR (no q signal)	—	assay	abs.	✓	0.307	0.508	0.114	0.216	0.152	0.261	0.181
Group DRO (absolute q signal)	abs.	assay	abs.	✓	0.350	0.543	0.094	0.286	0.276	0.328	0.278
RPR (uniform q signal)	unif.	assay	rel.	✓	0.437	0.517	0.079	0.250	0.210	0.207	0.197
RPR (absolute q signal)	abs.	assay	rel.	✓	0.339	0.472	**0.268**	**0.349**	0.185	0.198	0.113
RPR (family-balanced sampler)	rel.	family	rel.	✓	0.433	0.550	0.219	0.231	0.241	**0.368**	0.272
RPR (absolute loss)	rel.	assay	abs.	✓	0.380	0.556	0.116	0.296	0.281	0.334	0.276
RPR (no source pairwise term)	rel.	assay	rel.	×	0.390	0.568	0.174	0.309	0.276	0.346	0.276
RPR (all components)	rel.	assay	rel.	✓	**0.485**	**0.611**	0.255	0.312	**0.316**	0.323	**0.317**

peptide affinities from ChEMBL and BindingDB, the databases Boltz-2 curates from, and keeps peptide-like ligands (at least three amide bonds, 500–3000 Da) whose largest ring has at least 12 atoms and which have more than 50 heavy atoms, the size above which Boltz-2 discards molecules; every target ligand therefore lies outside the range its affinity head was fitted on. The source proxy applies Boltz-2's recipe to the same databases, keeping ligands of at most 50 heavy atoms from single-protein assays that pass its hit-to-lead criteria, and draws whole assays breadth-first over the 248 protein families (MMseqs2 clusters at 30% identity) shared with the target until it matches the target in size (19,993 and 19,959 pairs; Table 2). These are the families over which RPR reweights its rehearsal. The target stream is split by protein family, so no target test protein shares a family with target training: 15,516 pairs over 197 families for training, 2,064 over 90 for validation and 2,379 over 68 for test; the source-proxy training stream still covers 50 of the 68 test families. Source retention is measured on 1,909 held-out source records under a ligand-block split. External transfer uses the two peptide–protein affinity regression sets of PepBenchmark (Zhang et al., 2026), PpI_ba (1,256 records) and nc-PpI_ba (277), neither of which is a macrocycle benchmark. All methods start from Boltz-2 and train for 10 epochs at equal target exposure, with rehearsal methods also drawing source records 1:1, and use a single training seed throughout. All reported results use one checkpoint-selection protocol (protocol IV): it selects each checkpoint by the median within-family Spearman correlation on the target validation set and applies it unchanged to every test set. We report this family-level statistic on the test sets (Fam. ρ) with Spearman ρ, Pearson r, RMSE and AUC over all test records. These metrics treat the 23% of target test labels that are bounds ($>$ or $<$) as exact values; Appendix Table 11 gives exact-label and censoring-aware versions. Baselines follow Table 1. RPR uses $\lambda = 1$, $\epsilon = 0$ and $\rho = 0.9$, with $\eta = 0.4$ in Tables 3 and 4, from the step-size sweep in Table 6, and $\eta = 1.0$ elsewhere. Appendix E details the split, its family overlap and implementations.

Main results. In Table 3, RPR improves both source retention and target adaptation relative to uniform replay. It raises source Fam. ρ from 0.307 to 0.485 and ρ from 0.508 to 0.611, raises target Fam. ρ from 0.114 to 0.255 and ρ from 0.216 to 0.312, and lowers both RMSEs. On exact target

Table 5: **Hyperparameter study on source ratio.** Ratios are target:source draws. Target draws stay at 7,758 per epoch while source draws shrink, so training shortens from 7,760 steps (1:1) to 3,880 (1:0); the last row instead keeps 7,760 steps, raising target exposure. Metrics and bold as in Table 3.

	Small-molecule source ($n = 1909$)					Macrocycle target ($n = 2379$)					External ($n = 1533$), $\rho\uparrow$		
Method	Fam. $\rho\uparrow$	$\rho\uparrow$	$r\uparrow$	RMSE↓	AUC↑	Fam. $\rho\uparrow$	$\rho\uparrow$	$r\uparrow$	RMSE↓	AUC↑	PpI_ba	nc-PpI_ba	All
RPR (1:1)	**0.460**	**0.594**	**0.579**	**1.184**	**0.808**	0.188	0.346	0.364	1.489	0.659	**0.276**	0.299	0.279
RPR (1:0.75)	0.456	0.525	0.503	1.234	0.776	0.177	0.349	0.367	1.510	0.667	0.251	0.312	0.252
RPR (1:0.5)	0.267	0.506	0.493	1.257	0.761	0.163	**0.392**	**0.387**	**1.418**	**0.694**	0.258	0.258	0.235
RPR (1:0.5, 7,760 steps)	0.325	0.532	0.526	1.208	0.776	**0.225**	0.303	0.329	1.483	0.645	**0.276**	0.274	0.239
RPR (1:0.25)	0.328	0.507	0.478	1.307	0.753	0.135	0.159	0.182	1.579	0.563	0.214	**0.325**	0.252
FT (1:0)	0.249	0.467	0.440	1.357	0.723	0.144	0.237	0.233	1.591	0.613	0.273	0.313	**0.300**

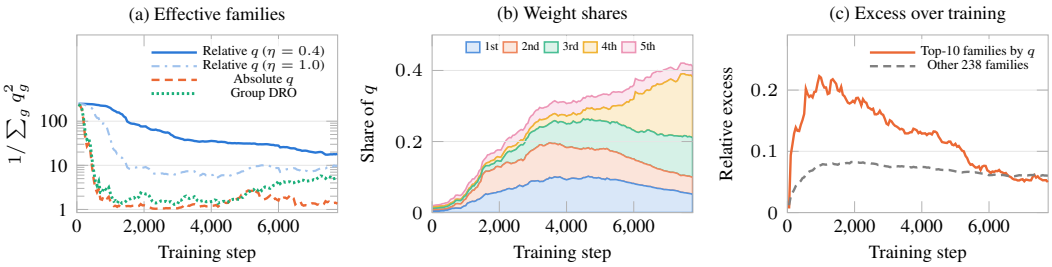


Figure 3: **Ablation study on family weights.** RPR of Table 3, with q logged every 50 steps over the 248 source families. (a) Relative weights narrow steadily to about 18 effective families, and faster with a larger step size, whereas absolute signals (Table 4) collapse onto one or two within 500 steps. (b) Shares of the five most-weighted families, ranked by mean q over training, which reach about 40% of the weight, with none above 18%. (c) The ten most-weighted families first degrade more than the rest and then fall to their level once rehearsed, averaging 0.068 versus 0.062 from step 5,000 on.

labels alone it keeps the highest ρ, Fam. ρ, r and AUC, but its RMSE (1.424) exceeds that of the pretrained model (1.371) and UR (1.383; Appendix Table 11). It also exceeds the pretrained model in both domains simultaneously. Among the baselines, the parameter-space methods L^2-SP and EWC fall below the pretrained model on the source (0.289 and 0.284), whereas UR and AGEM retain more source correlation (0.508 and 0.534) but lose target correlation relative to it (0.216 and 0.149). On the external benchmarks RPR reaches 0.316, 0.323 and 0.317, against 0.152, 0.261 and 0.181 for uniform replay, although FT keeps a higher AUC on the nc-PpI_ba benchmark (Appendix Table 10).

Ablation study. Table 4 changes one component of the main configuration at a time, and answers three questions. *Is the reference the right comparator?* Making both the weight signal and the rehearsed loss absolute, as in group DRO, lowers source and target Fam. ρ to 0.350 and 0.094; an absolute loss alone lowers the target value to 0.116. An absolute q signal is the one change that raises target Fam. ρ (0.268), but at the cost of source Fam. ρ (0.339) and combined external ρ (0.113 against 0.317), so the relative comparator trades target accuracy for retention and transfer. *Does adapting the weights matter?* Freezing them at uniform lowers target Fam. ρ to 0.079. *Is the gain merely more source data?* No: uniform replay and family-balanced sampling reach 0.114 and 0.219, and dropping the source pairwise term gives 0.174. Every change also lowers source Fam. ρ, from 0.485 to 0.339–0.437. Figure 3 shows how they behave: the family weights depart from uniformity gradually and narrow to about 18 effective families, while an absolute signal collapses onto one or two within 500 steps. The five most-weighted families reach about 40% of the weight, and the ten most-weighted, which degrade most early in training, fall back to the level of the rest once rehearsed. As predicted by Theorem 1, the weights follow those families whose losses have grown the most.

Hyperparameter study. These studies change one setting at a time around $\eta = 1.0$, $\lambda = 1$, $\epsilon = 0$ and $\rho = 0.9$. (i) *Source ratio.* Table 5 shrinks source draws at fixed target exposure. Removing rehearsal (1:0) lowers source Fam. ρ from 0.460 to 0.249; 1:0.75 nearly keeps it (0.456), while 1:0.5 and 1:0.25 fall to 0.267–0.328, and training 1:0.5 for 7,760 steps recovers only part of the gap (0.325), so the source share, not only training length, drives retention. (ii) *Step size.* Table 6 varies η: $\eta = 0.4$ gives the highest source and target Fam. ρ and combined external ρ, $\eta = 0.1$ is close on the target (0.249), and $\eta = 0$ drops the target value to 0.079, so the effect of η is not monotone. (iii) *Rehearsal*

Table 6: **Hyperparameter study on step size.** Each row changes only the step size η of the family-weight update, and $\eta = 0$ freezes the weights at uniform. Metrics and bold are as in Table 3.

	Small-molecule source ($n = 1909$)					Macrocycle target ($n = 2379$)					External ($n = 1533$), $\rho\uparrow$		
Method	Fam. $\rho\uparrow$	$\rho\uparrow$	$r\uparrow$	RMSE↓	AUC↑	Fam. $\rho\uparrow$	$\rho\uparrow$	$r\uparrow$	RMSE↓	AUC↑	PpI_ba	nc-PpI_ba	All
RPR ($\eta = 0$)	0.437	0.517	0.510	1.239	0.774	0.079	0.250	0.268	1.558	0.617	0.210	0.207	0.197
RPR ($\eta = 0.1$)	0.464	0.600	0.573	1.167	0.810	0.249	0.247	0.259	1.555	0.625	0.275	**0.343**	0.299
RPR ($\eta = 0.2$)	0.359	0.567	0.555	1.204	0.794	0.094	0.061	0.105	1.693	0.512	0.215	0.306	0.253
RPR ($\eta = 0.4$)	**0.485**	**0.611**	**0.593**	**1.144**	**0.814**	**0.255**	0.312	0.324	1.497	0.640	**0.316**	0.323	**0.317**
RPR ($\eta = 0.6$)	0.295	0.562	0.538	1.211	0.786	0.182	0.341	0.345	1.510	**0.665**	0.302	0.314	0.298
RPR ($\eta = 0.8$)	0.400	0.514	0.488	1.270	0.765	0.105	0.218	0.216	1.740	0.603	0.187	0.303	0.205
RPR ($\eta = 1.0$)	0.460	0.594	0.579	1.184	0.808	0.188	**0.346**	**0.364**	**1.489**	0.659	0.276	0.299	0.279

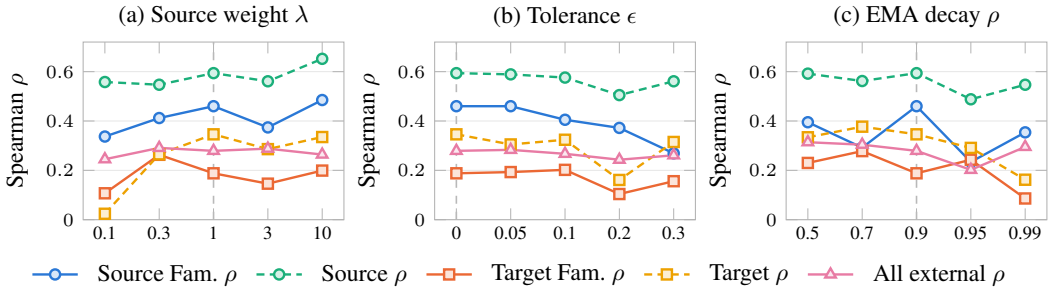


Figure 4: **Hyperparameter study on λ, ϵ and ρ.** Each point is a separately trained run in which one hyperparameter varies while the others stay at their defaults (dashed line). Circles mark the source, squares the target and triangles the external set; solid lines show Fam. ρ and dashed lines ρ over all test records, as in Table 3. Appendix Table 14 lists each value, its selected step and every test-set ρ.

weight, tolerance and EMA decay. Figure 4 varies λ, ϵ and ρ. Source ρ stays within 0.488–0.652, but source Fam. ρ spans 0.234–0.485, and the target statistics are fragile: target ρ collapses to 0.024 with weak rehearsal ($\lambda = 0.1$) and falls to about 0.16 at $\epsilon = 0.2$ or $\rho = 0.99$, where target Fam. ρ drops to 0.104 and 0.086. No positive tolerance ϵ improves source or target ρ over $\epsilon = 0$, and settings that beat the defaults on single statistics, such as $\lambda = 10$ or $\rho = 0.7$, are isolated single runs not chosen by validation, and so we keep the defaults used in Tables 3 and 4; Appendix Table 14 lists every run.

Further analysis. Appendix F reports the complete metrics behind every main-text table (Tables 10–14), including a re-evaluation of the target test set with censored labels handled explicitly (Table 11). The remaining appendices cover statements on AI use, ethics, reproducibility and limitations (Appendix A), related work (Appendix B), the proofs and the conditional proxy-retention bound (Appendix C), the construction of both data streams and the external sets, with the compute and storage each stage cost (Appendix D), and the split, protocol and implementation used throughout (Appendix E), with the method-specific terms and hyperparameters of every method listed in Table 9.

5 Conclusion

We studied adapting a molecular foundation model to macrocyclic peptide affinity when its original training corpus is unavailable and only an imperfect source proxy remains. Controlling the absolute worst-family risk cannot certify retention, whereas degradation relative to the pretrained model characterizes worst-case retention over unknown family proportions; Robust Proxy Rehearsal turns this into a training signal. With Boltz-2, RPR improved macrocycle and small-molecule prediction together and transferred to external peptide benchmarks in our main run, and its adaptive weights concentrated rehearsal on the families that degraded, which then recovered. These gains vary across hyperparameters; Appendix A.4 adds that retention is measured on a rebuilt proxy, the external benchmarks are not macrocycle-specific, the theory does not certify that RPR attains the constraint, and one model is adapted to one modality. Appendices A.1–A.3 cover language-model use, data terms and reproduction. Establishing when RPR reliably helps, and for which foundation models, target modalities and proxy corpora, is the main goal of our future work, together with repeated runs.

REFERENCES

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630(8016):493–500, 2024.

Alekh Agarwal and Tong Zhang. Minimax regret optimization for robust machine learning under distribution shift. In *Conference on Learning Theory*, pp. 2704–2729. PMLR, 2022.

Samad Amini, Simone Sciabola, and Ye Wang. Affinity fine-tuning of boltz-2: An open framework for protein-ligand potency prediction in drug discovery. *bioRxiv*, pp. 2026–05, 2026.

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. *arXiv preprint arXiv:1812.00420*, 2018.

Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K Dokania, Philip HS Torr, and Marc'Aurelio Ranzato. On tiny episodic memories in continual learning. *arXiv preprint arXiv:1902.10486*, 2019.

Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. Diffdock: Diffusion steps, twists, and turns for molecular docking. *arXiv preprint arXiv:2210.01776*, 2022.

Bradley Croy Doak, Björn Over, Fabrizio Giordanetto, and Jan Kihlberg. Oral druggable space beyond the rule of 5: insights from drugs and clinical candidates. *Chemistry & biology*, 21(9): 1115–1142, 2014.

Edward M Driggers, Stephen P Hale, Jinbo Lee, and Nicholas K Terrett. The exploration of macrocycles for drug discovery—an underexploited structural class. *Nature Reviews Drug Discovery*, 7(7): 608–624, 2008.

Google DeepMind. AlphaFold 3: Official inference implementation. `https://github.com/google-deepmind/alphafold3`, 2024. Accessed September 22, 2026.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873):583–589, 2021.

Ji Won Kim. *Fine-tuning Boltz for Antibody-Antigen Binding Prediction*. PhD thesis, Massachusetts Institute of Technology, 2025.

James King, Lewis Cornwall, Andrei Cristian Nica, James Day, Aaron Sim, Neil Dalchau, Lilly Wollman, and Joshua Meyers. On fine-tuning boltz-2 for protein-protein affinity prediction. *arXiv preprint arXiv:2512.06592*, 2025.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114 (13):3521–3526, 2017.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International conference on machine learning*, pp. 5637–5664. PMLR, 2021.

Donggyu Lee, Sangwon Jung, and Taesup Moon. Continual learning in the presence of spurious correlation. *arXiv preprint arXiv:2303.11863*, 2023.

Gang Li, Wendi Yu, Yao Yao, Wei Tong, Yingbin Liang, Qihang Lin, and Tianbao Yang. A retention-centric framework for continual learning with guaranteed model developmental safety. *arXiv preprint arXiv:2410.03955*, 2024.

Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.

Christopher A Lipinski, Franco Lombardo, Beryl W Dominy, and Paul J Feeney. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Advanced drug delivery reviews*, 23(1-3):3–25, 1997.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, 30, 2017.

Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pp. 109–165. Elsevier, 1989.

Saro Passaro, Gabriele Corso, Jeremy Wohlwend, Mateo Reveiz, Stephan Thaler, Vignesh Ram Somnath, Noah Getz, Tally Portnoi, Julien Roy, Hannes Stark, et al. Boltz-2: Towards accurate and efficient binding affinity prediction. *BioRxiv*, 2025.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. Experience replay for continual learning. *Advances in neural information processing systems*, 32, 2019.

Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731*, 2019.

Hannes Stärk, Octavian Ganea, Lagnajit Pattanaik, Regina Barzilay, and Tommi Jaakkola. Equibind: Geometric deep learning for drug binding structure prediction. In *International conference on machine learning*, pp. 20503–20521. PMLR, 2022.

Martin Steinegger and Johannes Söding. Mmseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature biotechnology*, 35(11):1026–1028, 2017.

Alexander A Vinogradov, Yizhen Yin, and Hiroaki Suga. Macrocyclic peptides as drug candidates: recent progress and remaining challenges. *Journal of the American Chemical Society*, 141(10): 4167–4181, 2019.

LI Xuhong, Yves Grandvalet, and Franck Davoine. Explicit inductive bias for transfer learning with convolutional networks. In *International conference on machine learning*, pp. 2825–2834. PMLR, 2018.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in neural information processing systems*, 33: 5824–5836, 2020.

Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International conference on machine learning*, pp. 3987–3995. Pmlr, 2017.

Jiahui Zhang, Rouyi Wang, Kuangqi Zhou, Tianshu Xiao, Lingyan Zhu, Yaosen Min, and Yang Wang. Pepbenchmark: A standardized benchmark for peptide machine learning. *arXiv preprint arXiv:2604.10531*, 2026.

Appendix for "Adapting Molecular Foundation Models for Macrocyclic Peptide Affinity Prediction via Robust Proxy Rehearsal"

APPENDIX CONTENTS

A AI Use, Ethics, Reproducibility, and Limitations. 12

A.1 AI use statement 12

A.2 Ethics statement 13

A.3 Reproducibility statement 13

A.4 Limitations and future directions 13

B Related Work 14

B.1 Molecular and biomolecular foundation models 14

B.2 Fine-tuning and adaptation strategies 14

B.3 Continual learning and retention 15

B.4 Affinity prediction and macrocyclic peptides 16

B.5 What is different here 16

C Proofs and Scope of the Retention Analysis 16

C.1 Setup and motivation 16

C.2 Proof of Theorem 1 17

C.3 Conditional transfer from the proxy to the source 17

C.4 A sufficient condition for both adaptation goals 19

C.5 Interpretation of assumptions and limits 19

D Data construction and earlier split 20

D.1 How the source proxy's size was decided 23

D.2 External benchmarks: which sets, and why those 24

E Experimental details 24

F Complete experimental results 25

A AI USE, ETHICS, REPRODUCIBILITY, AND LIMITATIONS

A.1 AI USE STATEMENT

Large language models were used as general-purpose assistants for this work, in three roles. They helped review and refactor experiment-management code, such as the job submission, evaluation and result-aggregation scripts around the training runs. They were used for editorial assistance on the manuscript: rewriting for concision, checking that reported numbers in the text match the tables generated from the result files, and copy-editing. They were also used as a search aid when locating related work, whose claims we then verified against the primary sources. All research contributions were made by the authors: the problem formulation, the retention analysis and its proofs, the design of RPR, the data-construction rules, the experimental protocol, and the interpretation of results. Every number reported in this paper is produced by the evaluation pipeline from stored model predictions,

and every citation was checked against the cited paper. The authors take full responsibility for the contents of this paper, including any text that was drafted with model assistance.

A.2 Ethics statement

This work uses only public, non-human-subject data: measured protein–ligand affinities from ChEMBL, BindingDB, PDBBind and PPB-Affinity, the public PepBenchmark (Zhang et al., 2026) evaluation sets, and the openly released Boltz-2 weights (Passaro et al., 2025). No personal or identifying data is involved, and no human or animal experiments were performed. We comply with the terms of use of each database and of the Boltz-2 release, and we do not redistribute any data we are not licensed to redistribute; our released artifacts consist of record identifiers, construction scripts and model code, from which the datasets can be rebuilt.

Affinity prediction for macrocyclic peptides is a dual-use capability. The intended use is to prioritise candidates for experimental validation in therapeutic discovery, which reduces the number of syntheses and assays required. The same models could in principle be used to design molecules that bind targets for harmful purposes. We judge the marginal risk here to be limited: the model ranks candidates for measured targets rather than designing novel agents, its accuracy is far from sufficient to replace experimental validation, and the underlying affinity data and the pretrained model are already public. We also note a misuse risk specific to our claims.

A.3 Reproducibility statement

We describe every step needed to reproduce this work. The data-construction rules, including the databases queried, the filters in the order applied, the deduplication and standardisation steps, and the resulting counts, are given in Appendix D; the two upstream featuriser defects we had to repair are documented there as well, since leaving them unfixed silently removes 5.1% of the target benchmark. The family-disjoint split, the checkpoint-selection rule (protocol IV), the training budget, the loss weights and the hyperparameters of every method are given in Appendix E. The method itself is stated in Section 3 as equations rather than prose, and all proofs are in Appendix C.

Every experiment in this paper uses fixed training seeds that are reported with the results, and checkpoint selection is performed by a rule applied automatically to the validation set rather than by manual choice. The evaluation pipeline writes per-record predictions to disk, and all reported metrics are recomputed from those files, so the tables can be regenerated without retraining. We report the full hyperparameter sweeps rather than only the configurations that worked (Tables 6 and 14), and we state which configuration was taken from the step-size sweep. Code for data construction, training and evaluation, together with the record identifiers defining each split, will be released.

Training the main comparison, the ablations and the sweeps reported here took on the order of 10^4 GPU-hours on NVIDIA A100-80GB GPUs, excluding the alignment and embedding precomputation whose storage is quantified in Table 2. Reproducing the full set of experiments therefore requires substantial compute; reproducing a single comparison, such as RPR against uniform replay at one seed, requires about two four-GPU days.

A.4 Limitations and future directions

One model, one target modality. All experiments adapt a single foundation model, Boltz-2, to a single target modality. Whether the same conclusions hold for other affinity predictors, for other shifts in ligand chemistry, or for structure rather than affinity prediction is untested here. Applying the analysis to settings where the pretraining corpus is partially available, so that proxy mismatch can be measured rather than assumed, is the most direct extension of this work.

Sensitivity to training randomness. Each configuration is trained once, at about 60 GPU-hours per run. The hyperparameter studies in Table 6 and Figure 4 show that the target statistics move appreciably with the step size and the rehearsal weight, which suggests that the margins in Table 3 should be read as indicative rather than as precise effect sizes. Repeating the comparison over several seeds would place error bars on those margins, and we regard this as the natural next step rather than a settled question.

Retention is measured on a proxy, not on the pretraining corpus. Boltz-2's curated affinity corpus is not publicly available, so retention is measured on a proxy we rebuilt from the same databases under the same documented rules. All retention numbers therefore concern that proxy. Proposition 2 states what else would be needed to carry proxy retention to the original source, and two of its terms, within-family mismatch and uncovered source mass, are unobservable without the corpus itself.

B Related Work

This appendix places the paper in four bodies of work: molecular foundation models, strategies for adapting them, continual learning and retention, and affinity prediction for macrocyclic peptides. We close by stating what distinguishes the setting studied here.

B.1 Molecular and biomolecular foundation models

Structure prediction has been reshaped by large pretrained models. The AlphaFold family established the paradigm for protein structure (Jumper et al., 2021), and AlphaFold 3 extended it to general biomolecular complexes including protein–ligand interactions (Abramson et al., 2024). Its released implementation, however, provides an inference pipeline without a fine-tuning path or an affinity head (Google DeepMind, 2024), which makes it unsuitable as a starting point for the adaptation problem we study.

Boltz-2 (Passaro et al., 2025) is the model we adapt. It couples a structure trunk with a learned affinity module trained on measured protein–ligand affinities, and it releases weights openly. Two properties matter for this paper. First, its affinity curation is documented in enough detail to be re-executed: the databases queried, the assay-quality criteria, the unit conventions, and the size bound of 50 heavy atoms. This is what makes a principled source proxy possible at all, and Appendix D follows those rules step by step. Second, the curated corpus itself is not distributed, which is precisely the constraint our problem formulation assumes.

Affinity prediction sits alongside a larger body of work on protein–ligand structure prediction, from which these models inherit their architectures. Regression-based geometric methods such as EquiBind predict a bound pose directly (Stärk et al., 2022), while DiffDock reformulates docking as generative sampling over ligand poses (Corso et al., 2022). These predict geometry rather than affinity, and are therefore complementary to the capability we protect: our constraint is on a scalar the affinity head outputs, not on the structure the trunk produces.

Work adapting Boltz-2 has so far targeted new tasks rather than retention. King et al. (2025) fine-tune it for protein–protein affinity, Kim (2025) for antibody–antigen binding, and Amini et al. (2026) provide an open framework for affinity fine-tuning on project-specific potency data, which is the implementation our target-only baseline uses. All optimise performance on the new task; none measures what happens to the model's original small-molecule capability, which is the quantity this paper places under constraint.

B.2 Fine-tuning and adaptation strategies

Adapting a pretrained model to a new domain admits several families of methods, and they differ in what they constrain.

Parameter-space constraints. L^2-SP regularises the fine-tuned weights toward the pretrained starting point (Xuhong et al., 2018), and elastic weight consolidation weights that penalty by a diagonal Fisher estimate so that parameters the reference model relies on move less (Kirkpatrick et al., 2017). These methods need no source data at training time beyond, for EWC, a Fisher estimate. Their limitation in our setting is structural: they bound how far weights travel, not how much predictions degrade, and the map from parameter distance to predictive degradation is neither uniform across families nor known in advance. Table 3 shows both falling below the pretrained model on source correlation while also losing target performance.

Function-space constraints. Output distillation keeps the adapted model close to the reference in function space by matching the reference's predictions on inputs drawn from the source domain (Li & Hoiem, 2017). This constrains predictions rather than weights, which is closer to what retention means, but it protects only the region the available inputs cover, and it treats all such inputs alike regardless of how much each has degraded.

Data-space methods. Rehearsal interleaves source examples with target examples (Chaudhry et al., 2019; Rolnick et al., 2019), which directly supervises the behaviour to be retained when labelled source data exist. Uniform replay, the standard form, applies the same objective to every source record. Our method belongs to this family and changes only which source errors are prioritised.

Restricting what can change. A different response to forgetting is to reduce the capacity that adaptation may use at all. Low-rank adaptation freezes the pretrained weights and trains a low-rank update (Hu et al., 2021), which restricts the direction of the update and makes the adaptation cheap to store and revert. This is orthogonal to our contribution: it constrains the hypothesis class, whereas we constrain the objective, and the two can be combined. It also does not by itself decide *which* source behaviour to protect when the protected domain is heterogeneous, which is the question Theorem 1 answers.

Gradient-space methods. Gradient episodic memory constrains updates so that the loss on stored examples does not increase (Lopez-Paz & Ranzato, 2017); AGEM relaxes this to a single averaged constraint and is far cheaper (Chaudhry et al., 2018), and it is the variant we run. Related work projects conflicting task gradients to reduce interference (Yu et al., 2020). These methods constrain the direction of each update using source gradients, which is a local condition and is enforced per step rather than over the whole adaptation trajectory.

B.3 Continual learning and retention

Catastrophic forgetting, the loss of previously acquired capability when a network is trained on new data, has been studied since McCloskey & Cohen (1989), and a large literature has grown from it; Lee et al. (2023), for instance, analyse how spurious correlations propagate across a task sequence and propose a simple baseline against them. Most of this literature measures forgetting over a sequence of tasks whose data the learner has seen, and reports average accuracy and a forgetting measure across them.

The classical responses fall into the families listed above and were largely developed on task-incremental image classification: importance-weighted parameter penalties estimated from the Fisher (Kirkpatrick et al., 2017) or accumulated along the optimisation path (Zenke et al., 2017), and function-space constraints that keep the new model's outputs close to the old model's on available inputs (Li & Hoiem, 2017). Our setting differs from that benchmark in three ways that matter: there is no task sequence, only one adaptation step from a public checkpoint; the protected data are not stored but reconstructed; and the protected domain has internal structure, the protein families, over which the composition is unknown.

Two threads are directly relevant here. The first is retention as an explicit constraint rather than an average. Li et al. (2024) formulate model development as improving a target task subject to bounds on the loss increase relative to a reference model on protected tasks, give a retention generalisation bound and a constrained optimisation algorithm, and instantiate it for CLIP adaptation. Our problem formulation in Section 2 adopts this reference-relative view. What we add is the consequence of the reference data being unavailable: retention must be estimated on a reconstructed proxy whose family proportions, within-family distribution and coverage may all differ from the original, which is what Proposition 2 accounts for.

The second thread is robustness to unknown group proportions. Group distributionally robust optimisation minimises the worst-group risk over a set of groups (Sagawa et al., 2019), which protects against unknown mixing weights. Theorem 1 shows why the *absolute* worst-group risk is the wrong functional for retention: it ranks groups by current difficulty rather than by change, so it can certify nothing about degradation. Benchmarks such as WILDS have made group shift a standard axis of evaluation (Koh et al., 2021), and in most of that literature the groups are observed attributes and the objective is accuracy under shift; here the groups are protein families, and the objective is degradation

against a fixed reference rather than accuracy itself. Minimax regret (Agarwal & Zhang, 2022) also measures each group against a comparator, but that comparator is the group's own optimal predictor; in our setting the comparator is fixed and given, namely the pretrained model whose behaviour is to be preserved. Because the comparator is fixed, the mixture risk stays linear in the family proportions, so the supremum in Equation 5 is attained at a vertex of the simplex and reduces to a maximum over families.

B.4 Affinity prediction and macrocyclic peptides

Binding-affinity prediction ranks protein–ligand pairs to prioritise experimental validation, and it is the capability Boltz-2's affinity module provides (Passaro et al., 2025). Macrocyclic peptides are an attractive modality because they combine antibody-like affinity and selectivity with the tissue penetration and tunable pharmacokinetics associated with smaller therapeutics (Driggers et al., 2008; Vinogradov et al., 2019). They sit outside the descriptor ranges that guided small-molecule discovery for decades (Lipinski et al., 1997), in the region characterised as beyond the rule of five (Doak et al., 2014), and that is exactly why the models fitted on small molecules do not transfer to them for free: Boltz-2's affinity curation discards molecules above 50 heavy atoms, while our target corpus has a median of 80.

Evaluation resources for peptides are comparatively scarce. PepBenchmark (Zhang et al., 2026) standardises 35 peptide datasets across activity, ADME and toxicity, of which only the PepPI group measures peptide–protein binding. Since we adapt a regression head, we use its two regression sets for affinity regression, taking both rather than selecting on score; neither is macrocycle-specific, as Appendix D quantifies.

B.5 What is different here

The setting combines three constraints that are individually familiar and jointly unaddressed. The pretraining corpus is unavailable, so retention cannot be measured where it is defined and must be estimated on a proxy with unknown mismatch. The protected domain is naturally partitioned into protein families whose proportions in the original corpus are unknown, so an average over the proxy is not a safe surrogate. And the reference model is fixed and public, which makes reference-relative degradation both the right quantity and a measurable one. Within these constraints, our contribution is to characterise which functional certifies retention (Theorem 1), to turn it into a rehearsal signal that costs no more than uniform replay, and to report honestly how far the resulting empirical gains do and do not generalise.

C Proofs and Scope of the Retention Analysis

This appendix restates the retention problem in full, explains why the functional we analyse is the one that matters, proves Theorem 1, and then gives the conditional result that carries proxy retention to the unavailable source. Section 2 states the same material compactly; nothing here supersedes it.

C.1 Setup and motivation

The problem. A pretrained model θ_0 performs well on a source domain P_S, here small-molecule affinity, and is to be adapted to a target domain P_T, here macrocyclic peptides. Adaptation must improve target performance without losing source performance. With $R_P(\theta)$ the risk under P and $\Delta_P(\theta) = R_P(\theta) - R_P(\theta_0)$ the degradation relative to the pretrained model, the goal is Equation 2: minimise $R_{P_T}(\theta)$ subject to $\Delta_{P_S}(\theta) \leq \epsilon$.

Why this is not a standard continual-learning problem. Two features of our setting break the usual assumptions. First, P_S is not available. Boltz-2's curated affinity corpus is not distributed (Appendix D), so the constraint $\Delta_{P_S}(\theta) \leq \epsilon$ cannot be evaluated; only a proxy Q rebuilt from public databases can be. Second, the source domain is not homogeneous. It decomposes into protein families whose proportions in the original corpus are unknown, and the proxy's proportions need not match them. An average over the proxy is therefore not a safe surrogate for the constraint: a method can lower the proxy average while degrading badly on families the proxy under-represents.

Why the functional must be reference-relative. Given unknown family proportions, the natural robust strategy is to control the worst family. The question is which per-family quantity to control. Group DRO (Sagawa et al., 2019) controls the largest *absolute* risk $\max_g R_g(\theta)$, which ranks families by how hard they are now. Retention is a statement about *change*, so the corresponding quantity is the largest *degradation* $\max_g \Delta_g(\theta)$. These differ whenever difficulty and degradation are not aligned, which is the normal case: a family that was always hard stays hard after improving, while a family the pretrained model handled well can deteriorate and still look easy in absolute terms. Theorem 1 makes both halves of this precise. Part (i) shows that reducing $\max_g R_g$ certifies nothing about Δ_{P_S}, and part (ii) shows that $\max_g \Delta_g$ is exactly the worst-case degradation over all mixtures, so bounding it is equivalent to satisfying the constraint for every possible family composition. The proof of part (i) is a two-point construction, given below; the proof of part (ii) is the observation that a supremum of a linear functional over a simplex is attained at a vertex.

What the result does not say. The theorem characterises which predictors retain source performance; it does not say how to find them, and it concerns the true family conditionals rather than the proxy laws we train on. Both gaps are addressed, as far as they can be, in Appendices C.3 and C.5.

C.2 Proof of Theorem 1

Proof. Part (i). Let the input space be $\{a, b\}$, let the response be identically zero, and consider squared loss $\ell(f(x), 0) = f(x)^2$ with predictions in $[0, 1]$. Let P_1 and P_2 be point masses at $(a, 0)$ and $(b, 0)$, respectively, and define

$$f_{\theta_0}(a) = \sqrt{0.8}, \quad f_{\theta_0}(b) = \sqrt{0.1}, \qquad f_\theta(a) = \sqrt{0.6}, \quad f_\theta(b) = \sqrt{0.3}.$$

The group-risk vectors are therefore $(0.8, 0.1)$ and $(0.6, 0.3)$, so their maxima decrease from 0.8 to 0.6. For $P_S = 0.1P_1 + 0.9P_2$,

$$R_{P_S}(\theta_0) = 0.1(0.8) + 0.9(0.1) = 0.17, \qquad R_{P_S}(\theta) = 0.1(0.6) + 0.9(0.3) = 0.33.$$

Hence $\Delta_{P_S}(\theta) = 0.16 > 0$, proving the claimed non-implication.

Part (ii). For any $p \in \Pi_G$, finite group risks and linearity of expectation imply

$$\Delta_{P_p}(\theta) = \sum_{g=1}^{G} p_g \big[R_g(\theta) - R_g(\theta_0)\big] = \sum_{g=1}^{G} p_g \Delta_g(\theta) \le \max_g \Delta_g(\theta).$$

Since the number of groups is finite, choose $g^\star \in \arg\max_g \Delta_g(\theta)$. The simplex vertex with $p_{g^\star} = 1$ attains equality. Taking the supremum proves (5). If every group degradation is at most ϵ, every mixture degradation is at most ϵ. Conversely, if every mixture satisfies this tolerance, applying the condition at each simplex vertex gives $\Delta_g(\theta) \le \epsilon$ for every g. □

The quantifier over all mixtures is essential. For a fixed mixture, positive and negative group degradations may cancel. Restricting admissible proportions to $\mathcal{W} \subseteq \Pi_G$ yields the support function $\sup_{p \in \mathcal{W}} \sum_g p_g \Delta_g(\theta)$; this is no greater than the full-simplex maximum, but is valid only if the actual proportions belong to $\mathcal{W}$. Also, $\max_g [R_g(\theta) - R_g(\theta_0)]$ is generally different from $\max_g R_g(\theta) - \max_g R_g(\theta_0)$, as the preceding counterexample shows.

C.3 Conditional transfer from the proxy to the source

Theorem 1 assumes access to the true family conditionals. The proxy instead provides samples from Q_g and may miss parts of the source. We represent incomplete coverage as

$$P_S = (1 - \omega) \sum_{g=1}^{G} p_g P_g + \omega P_U, \quad p \in \Pi_G, \quad 0 \le \omega \le 1, \tag{12}$$

where P_U is the uncovered component and ω its source probability mass, distinct from the optimization step size η. Assume $0 \le \ell_\theta(z) \le B$ for the reference and all candidates. This applies, for example, to a clipped evaluation loss, but not automatically to an unbounded training loss. Let $\mathrm{TV}(P_g, Q_g) := \sup_A |P_g(A) - Q_g(A)| \le \delta_g$, where the laws include both inputs and supervision.

For a finite set $\mathcal{F} = \{\theta^{(1)}, \dots, \theta^{(K)}\}$ fixed independently of calibration data, observe $n_g \geq 1$ independent calibration units $Z_{gi} \sim Q_g$ in each group. Define

$$\widehat{\Delta}_g^Q(\theta) = \frac{1}{n_g} \sum_{i=1}^{n_g} \big[\ell_\theta(Z_{gi}) - \ell_{\theta_0}(Z_{gi})\big], \qquad u_g = B\sqrt{\frac{2\log(2GK/\alpha)}{n_g}}, \tag{13}$$

for $\alpha \in (0,1)$. The reference is also fixed independently of these units. Training may generate the candidates, but calibration must not shape candidate generation.

Proposition 2 (Conditional source-retention bound from a proxy)**.** *Under the preceding assumptions, with probability at least $1-\alpha$ over the proxy calibration samples, simultaneously for every $\theta \in \mathcal{F}$,*

$$\Delta_{P_S}(\theta) \leq \underbrace{(1-\omega) \max_{1\leq g\leq G} \left\{\widehat{\Delta}_g^Q(\theta) + u_g + 2B\delta_g\right\} + \omega B}_{U_S(\theta)}. \tag{14}$$

Hence $U_S(\theta) \leq \epsilon$ is sufficient for $R_{P_S}(\theta) \leq R_{P_S}(\theta_0) + \epsilon$ on this event. The bound holds uniformly over the covered-family proportions $p \in \Pi_G$.

The terms u_g, $2B\delta_g$, and ωB represent sampling uncertainty, within-family mismatch, and missing coverage, respectively. More independent samples reduce the first term, but not necessarily the other two. Since the original corpus is unavailable, δ_g and ω require assumptions or sensitivity analysis; the proposition alone does not provide an observable certificate.

Proof. Write $d_\theta(z) = \ell_\theta(z) - \ell_{\theta_0}(z)$. Boundedness implies $d_\theta(z) \in [-B, B]$.

Step 1: simultaneous sampling control. For fixed g and $\theta \in \mathcal{F}$, Hoeffding's inequality applied to n_g independent variables of range length $2B$ gives

$$\Pr\left(\left|\widehat{\Delta}_g^Q(\theta) - \mathbb{E}_{Q_g} d_\theta\right| > t\right) \leq 2\exp\left(-\frac{n_g t^2}{2B^2}\right).$$

At $t = u_g$ from (13), this probability is at most $\alpha/(GK)$. A union bound over all GK group–candidate pairs therefore gives an event $\mathcal{E}$ of probability at least $1-\alpha$ on which

$$\mathbb{E}_{Q_g} d_\theta \leq \widehat{\Delta}_g^Q(\theta) + u_g \quad \text{for every } g, \theta. \tag{15}$$

This argument can be conditioned on training data and training randomness provided the calibration units retain the stated laws and independence. Independence between different groups is not required for the union bound.

Step 2: transfer within a covered family. For any measurable $h \in [0,1]$, the layer-cake representation gives

$$|\mathbb{E}_P h - \mathbb{E}_Q h| = \left|\int_0^1 [P(h > t) - Q(h > t)]\, dt\right| \leq \mathrm{TV}(P, Q).$$

Apply this to $h = (d_\theta + B)/(2B)$ for $B > 0$. The constant offset cancels between probability measures, so

$$\mathbb{E}_{P_g} d_\theta \leq \mathbb{E}_{Q_g} d_\theta + 2B\,\mathrm{TV}(P_g, Q_g) \leq \mathbb{E}_{Q_g} d_\theta + 2B\delta_g.$$

Combined with (15), on $\mathcal{E}$ this yields

$$\Delta_g(\theta) \leq \widehat{\Delta}_g^Q(\theta) + u_g + 2B\delta_g. \tag{16}$$

For $B = 0$, all risks and degradations vanish and the proposition is immediate.

Step 3: account for uncovered source mass. The mixture decomposition (12) gives

$$\begin{aligned}
\Delta_{P_S}(\theta) &= (1-\omega)\sum_g p_g \Delta_g(\theta) + \omega \mathbb{E}_{P_U} d_\theta \\
&\leq (1-\omega)\sum_g p_g \left[\widehat{\Delta}_g^Q(\theta) + u_g + 2B\delta_g\right] + \omega B \\
&\leq (1-\omega)\max_g \left[\widehat{\Delta}_g^Q(\theta) + u_g + 2B\delta_g\right] + \omega B.
\end{aligned}$$

The first inequality uses (16) and $d_\theta \le B$ on the unobserved component. The last uses the simplex constraint on p. All inequalities hold simultaneously for the candidate set and every covered-group mixture on $\mathcal{E}$. If the final upper bound is at most ϵ, the retention conclusion follows directly from the definition of Δ_{P_S}. □

C.4 A SUFFICIENT CONDITION FOR BOTH ADAPTATION GOALS

The source bound alone does not establish target improvement. A separate calibration argument can state precisely what additional evidence suffices.

Corollary 3 (Joint target improvement and source retention)**.** *Use Proposition 2 at failure probability α_S. In addition, let $m \ge 1$ independent target calibration units $Z_j^T \sim P_T$ be independent of the fixed reference and candidate set, with the same loss bound B. Define*

$$\widehat{\Delta}_T(\theta) = \frac{1}{m}\sum_{j=1}^{m} d_\theta(Z_j^T), \qquad v_T = B\sqrt{\frac{2\log(2K/\alpha_T)}{m}}.$$

With probability at least $1 - \alpha_S - \alpha_T$, every candidate satisfying

$$U_S(\theta) \le \epsilon, \qquad \widehat{\Delta}_T(\theta) + v_T < 0$$

simultaneously satisfies

$$R_{P_S}(\theta) \le R_{P_S}(\theta_0) + \epsilon, \qquad R_{P_T}(\theta) < R_{P_T}(\theta_0).$$

This remains true for a candidate selected using these calibration results.

Remark 1 (An acceptance condition, not guaranteed feasibility). The corollary permits selecting the best target candidate among those meeting the stated source condition, but does not guarantee that any candidate passes. The known reference always has zero true degradation; a loose proxy bound need not certify it, and the reference cannot satisfy strict target improvement. If mismatch and coverage are not bounded credibly, U_S remains a conditional quantity rather than an operational certificate. These conditions describe a possible calibration protocol, not an additional step already used in the experiments.

Proof. The same Hoeffding and union-bound argument as Step 1 above gives, with probability at least $1 - \alpha_T$, $\Delta_{P_T}(\theta) \le \widehat{\Delta}_T(\theta) + v_T$ simultaneously for all K candidates. Intersect this event with the source event of Proposition 2. A union bound gives probability at least $1 - \alpha_S - \alpha_T$ for the intersection; independence between the source and target events is unnecessary. On that intersection, the two acceptance inequalities imply the two population conclusions. Simultaneity over the candidate set permits subsequent data-dependent selection. □

C.5 INTERPRETATION OF ASSUMPTIONS AND LIMITS

What constitutes an independent unit. For per-complex loss, the proposition assumes independently sampled complexes. For a loss containing within-assay differences, one can instead let Z be an independently sampled assay block and $\ell_\theta(Z)$ its bounded aggregate loss. Then n_g counts independent blocks, not compounds or pairwise differences. The resulting population risk is over that block sampling distribution; it is not automatically the pair-weighted risk of a different benchmark. Dependence across assays or repeated proteins requires an appropriate dependence-aware concentration argument.

Adaptive fitting and calibration. The finite-candidate guarantee applies to candidates generated independently of the calibration samples. Reusing those samples to adjust model weights, design new hyperparameter trials, or repeatedly expand the candidate set violates this assumption. Such reuse requires a uniform-complexity or adaptive-data-analysis argument. Calibration is conceptually distinct from the final held-out evaluation.

Mismatch is over the joint law. The total-variation term concerns protein–ligand inputs and affinity supervision together. Sequence similarity or a feature-distance estimate does not automatically upper-

bound it. Different assay mechanisms can contribute to δ_g even for similar proteins. If only ranges for ω and δ_g are assumed, one must maximize the bound over the admissible ranges rather than treat unknown quantities as measured.

Missing coverage and metric scope. Without additional assumptions, $\mathbb{E}_{P_U} d_\theta \leq B$ cannot be improved uniformly: losses zero for the reference and B for the candidate on P_U attain it while leaving all covered-group observations unchanged. This does not imply that every update fails on uncovered data; it identifies the limit of a guarantee based only on covered data. Finally, bounded-loss retention need not imply preservation of Spearman correlation, enrichment, or an unbounded regression loss. Those metrics require separate analysis or empirical evaluation.

D DATA CONSTRUCTION AND EARLIER SPLIT

Every step below is fixed by an external constraint or a stated rule; none is tuned against results.

Target domain: which databases, and why those. Boltz-2's appendix states that it curates continuous affinity data *"primarily from ChEMBL and BindingDB"*, amounting to 1.2M measured pairs over 2k protein targets. We search the *same two* databases for macrocyclic peptides, so that the target domain is drawn from the same provenance as the distribution the model was fitted on and extends its ligand-size range. We additionally consider two databases that report affinity *together with* an experimental structure — PDBBind and PPB-Affinity — so that the benchmark is not narrower than it needs to be. From these four we keep only records satisfying, in order:

1. **Peptide-like**: ≥ 3 amide bonds and $500 \leq \mathrm{MW} \leq 3000$. An operational definition, stated rather than tuned.
2. **Assay normalisation**: values converted to $\log_{10} \mu$M following Boltz-2's convention; on conflict, assay types are preferred Kd $>$ Ki $>$ IC_{50} $>$ EC_{50} (thermodynamic before functional) and records carrying an experimental structure win ties.
3. **Deduplication** by (canonical SMILES, protein sequence).
4. **Heavy atoms** > 50**.** *This bound is Boltz-2's, not ours*: its curation *"filter[s] out molecules with more than 50 heavy atoms"*, so every target example lies outside the range its affinity head was fitted on.
5. **Standardisation**, applied exactly as Boltz-2's affinity path applies it: the ChEMBL structure-curation pipeline, which strips counterions and co-crystallised fragments and keeps the largest component. Two of its rules remove ligands outright — molecules containing a metal or more than seven borons are excluded (87 pairs, largely ^{177}Lu/^{188}Re/^{68}Ga-chelated radiopharmaceutical peptides), and 373 further pairs fall below the 50-heavy-atom bound once their counterions are removed. We apply the size bounds to the standardised molecule, i.e. to what the model actually sees.
6. **Heavy atoms** ≤ 256, the affinity module's token-cropping capacity. This removes 155 pairs, the largest reaching 423 heavy atoms.
7. **Availability**: the protein must have a computed alignment. We compute alignments in-house for every protein in the pool, so unlike the size bounds this step removes nothing.
8. **Actually macrocyclic**, applied last: largest ring ≥ 12 atoms, the standard bound in the beyond-rule-of-five literature. This matters more than it sounds. The peptide-like rule alone admits linear peptides carrying an ordinary substituent ring: 64% of the pool's ligands have a largest ring of 1–7 atoms, a phenyl or a piperidine. A benchmark named after macrocycles cannot be two thirds linear ones, so we require the ring, which removes 40,364 pairs and leaves a median ring size of 23.

Steps 1–4 yield 60,964 pairs over 2,769 proteins that are by construction outside Boltz-2's training range; steps 5–7 remove a further 615, and the macrocycle requirement (step 8) removes 40,364 more. This leaves 19,985 pairs over 844 proteins and 10,442 distinct ligands in 358 families at 30% identity. Removing the 26 pairs that overlap the external benchmarks gives the target stream of Table 2: **19,959** pairs over **837** proteins and **10,419** ligands in 355 families.

What the distribution looks like, and why we do not reshape it. The rules above fix the distribution; we report it rather than balance it. Heavy-atom counts span 51–213 with a median of 80: 27.2% of pairs fall in 51–70, 33.0% in 70–90, and 39.9% above 90, with 34.4% inside the 85–135 window that clinical oral macrocycles occupy. Contributions are ChEMBL 67.4%, BindingDB 31.6%, PDBBind 1.1%; PPB-Affinity passes the earlier filters but none of its entries are macrocyclic. Labels carry a qualifier in 18.5% of pairs (17.6% ">", 0.9% "<"), which we treat as censored rather than discard.

The protein side is severely long-tailed, and this shapes the method more than the size distribution does. The median protein contributes 3 pairs while the mean is 23.7; the largest single protein contributes 795, and 273 of 844 proteins (32%) appear exactly once. Both streams are sampled assay by assay, following Boltz-2's Algorithm 4: assays are drawn in proportion to the interquartile range of their affinities and five compounds are taken from each, so that within-assay differences supervise the pairwise term; family structure enters through the split and, for the source, through the family weights of RPR.

Two silent losses we had to repair first. Building the pool surfaced two defects in the upstream featuriser that discard ligands without failing the run — each logs a traceback, skips the molecule, and reports success. Both are size-dependent, so they disproportionately affect the macrocycles this benchmark is about, and both remove a chemotype rather than a random sample. First, ligand atoms are named `SYMBOL+`*index* and validated against a four-character field; since indices run over all atoms including hydrogens, any two-letter element past index 99 overflows. Every ligand here exceeds 100 atoms with hydrogens, so this rejected 1,209 pairs, 78% of them chlorinated. We encode the index in base 36 (`CL118` → `CL3A`), which keeps the element symbol, stays unique, and leaves the names of all other molecules byte-identical. Second, standardisation parses with `sanitize=False` and then calls a fragment chooser that requires implicit valences, raising on 1,928 pairs; priming the property cache repairs all of them. Left unfixed, the two defects would have silently removed 5.1% of the benchmark with a systematic bias toward halogenated and multi-fragment ligands — a bias invisible in any success/failure count, since every job still reports success.

Earlier ligand-disjoint split. Splitting on individual pairs leaks: a ligand measured against several proteins, or an assay whose compounds are scattered across splits, puts training information in the test set. We therefore split *blocks*. Two target records are linked when they share a ligand or a known assay, and each connected component is assigned whole; the 13.5% of target records with no recoverable assay are never linked on that key. The source stream links on the ligand alone, because its test set asks whether the model has forgotten small molecules, not whether it generalises to a new experiment; we report that 99.7% of source test records come from assays seen in training and read that number as retention on unseen compounds of seen assays. No ligand and no known target assay crosses a split boundary in either stream.

Blocks are allocated family-stratified, where families are MMseqs2 30%-identity clusters: within a family, each block goes to whichever of validation or test has the larger fraction of its quota left, and blocks that fit neither go to training, so nothing is discarded and neither held-out set is systematically served the other's leftovers. Target quotas are capped at $\min(60, 40\%)$ per family. The cap matters: without it one family supplies a quarter of the test set and the effective number of independent units collapses. Families are used for this cap and for stratification. The result is $81.7/9.0/9.4$ for the target stream and $80.9/9.5/9.5$ for the source. This protocol measures lead optimisation — ranking new ligands against a target — and not extrapolation to unseen families. The experiments keep it for the source stream only; the target stream uses the family-disjoint split of Appendix E.

What the source distribution actually is. Boltz-2 documents its affinity curation precisely, which lets us define the proxy by following its recipe rather than guessing. For the head we adapt — the regression head that predicts a value, not the binary binder/decoy head — its appendix states that continuous affinity data are curated *"primarily from ChEMBL and BindingDB"*, keeping *"only assays that target a single protein and are categorized as either biochemical or functional, excluding any labeled as low-confidence or unreliable"*, with all values *"standardized to* $\log_{10}$ *scale derived from values measured in* μM*"* and molecules above 50 heavy atoms removed. ChEMBL is the primary source; BindingDB is used to *"retain only records not already covered by ChEMBL"*. Table 1 of

that paper puts the result at **1.2M measured pairs over 2k protein targets and 600k compounds** (1.45M before a structural-quality filter that discards complexes with iPTM < 0.75).

What we could actually build, and why it is smaller. Two obstacles rule out rehearsing Boltz-2's own affinity corpus. *Availability.* Although the Boltz-2 paper states that code, weights and data are available from its repository (`https://github.com/jwohlwend/boltz`), none of the public release channels provides the curated affinity set (all checked in September 2026). The repository's data documentation covers only the Boltz-1 structural data (PDB and OpenFold distillation sets) and links no affinity data. The Hugging Face organization (`https://huggingface.co/boltz-community`) hosts the Boltz-1 and Boltz-2 weights and lists no public datasets. A GitHub request for the affinity training, validation and test data (issue #660, March 2026) was closed as "not planned" without a reply. A third-party reconstruction that follows the preprint (SandboxAQ's AQAffinity) releases assay lists but defers its queries and preprocessing scripts. The recipe itself is given in prose, and two of its steps — randomly shuffled synthetic decoys and an assay filter scored by Boltz-2's own confidence module — cannot be reproduced exactly. *Scale.* Table 7 combines the corpus size that Boltz-2 reports with unit costs we measured in our own pipeline. Rebuilding the full corpus would require over 25,000 GPU-hours of co-folding, about 1 PB of uncropped trunk embeddings, and at least 1,800 GPU-hours of alignment before any training begins. We therefore follow the same recipe at a fraction of the scale: 19,993 pairs over 518 proteins, about 0.4% of the corpus.

Table 7: Cost of rebuilding Boltz-2's affinity training corpus. Corpus sizes are taken from Table 1 of Passaro et al. (2025) (after its structural-quality filter; before it in parentheses); protein targets are its clusters at 90% identity, summed over sources. Unit costs are measured in our pipeline on NVIDIA A100-80GB GPUs with the inference settings used for our proxy (3 recycling steps, 50 sampling steps, one sample): co-folding time is the median interval between consecutive predictions in 12 inference jobs; embedding size is the median over 200 uncropped embeddings written by Boltz-2 (mean 0.34 GB); alignment cost is 182.9 GPU-hours for 396 sequences using GPU MMseqs2 against a local ColabFold database. Boltz-2 crops features to the binding pocket, so its own storage is smaller than the uncropped estimate.

Quantity	Unit cost	Full corpus
Protein–ligand examples	—	4.62M (7.17M)
with measured affinity values	—	1.20M (1.45M)
with binary binder/decoy labels	—	3.42M (5.72M)
Protein targets (90% clusters)	—	≈3.9k (5.9k)
Co-folding for trunk features	20 s per complex	25,700 (39,800) GPU-h
Uncropped trunk embeddings	0.20 GB per complex	0.94 (1.43) PB
Multiple-sequence alignments	0.46 GPU-h per sequence	≥1,800 (≥2,700) GPU-h
Alignment database	—	1.4 TB
Our proxy (Table 2)	—	19,993 pairs, 518 proteins

Two rules then fix its shape. First, the *assay* is the unit. Boltz-2 draws each batch as five compounds from one assay and supervises their pairwise differences, because affinity values are not comparable across assays — an IC_{50} depends on substrate concentration, and the Cheng–Prusoff correction needs metadata the public records rarely carry — so a difference taken inside one assay cancels what a raw value cannot. We therefore sample whole assays, never individual pairs, and apply the hit-to-lead criteria of Boltz-2's Appendix A.2.1: single-protein assays with at least ten measurements, at least ten distinct activity values, a unique-to-total ratio above 0.2, and an activity standard deviation of at least 0.25 log units. Assays are keyed by measurement type as well as identifier, since BindingDB's assay id is a DOI and one paper routinely reports IC_{50} and K_i side by side.

Second, we aim for a protein count comparable to the target's, so that neither stream is simply broader than the other. How far that is achievable is itself a measurement: clustering both sides at 30% identity, the target occupies 355 families, and only 248 of them yield source assays that survive those criteria. We draw whole assays breadth-first over those 248 families — families, not proteins, are what rehearsal protects and what the group-DRO objective ranges over — until the pair count matches

the target's (19,993 against 19,959), so that a difference in results cannot be attributed to one stream carrying more data.

The remaining 107 families — 30.1% of those the target perturbs — cannot be rehearsed at all, having either no small-molecule counterpart anywhere in ChEMBL or BindingDB or one whose assays carry too little activity spread to supply a difference signal. One plausible reason is that macrocyclic peptides are developed against interfaces that small-molecule campaigns avoid, though our data only establish that no record in these databases passes the filters, not why. Either way it is the premise of the method in quantitative form — uniform rehearsal and group-robust rehearsal both obtain direct source supervision only for the 69.9% of target families covered by the proxy. Any benefit for the remaining families must be established empirically; group reweighting alone does not supply their missing supervision (Section 2). Target pairs in unreachable families are kept: the assay-level criteria judge *source* measurements, and the macrocycle data for the same family are different experiments on different chemistry. Appendix D.1 gives the derivation.

Source retention set. The main tables measure retention on the 1,909-record source test split, held out by ligand blocks from the reconstructed source data. Protein and assay overlap with source training records is possible. These results measure retention on held-out proxy-domain compounds; they do not directly measure risk on the unavailable original pretraining distribution.

What this study itself cost. Table 8 breaks down our own compute and storage by stage. Two properties of this pipeline drive the cost. First, the trunk features are precomputed once per protein–ligand record and reused across every training run, which is why the one-off embedding stage dominates storage while contributing little compute. Second, each training run is short in wall-clock terms but is repeated many times: the main comparison, the mechanism ablations and the four hyperparameter sweeps together account for the bulk of the GPU-hours reported here. The figures cover the experiments reported in this paper and exclude exploratory runs that were discarded. They are also specific to a corpus of 39,952 pairs; the same pipeline applied to Boltz-2's full affinity corpus would incur the costs in Table 7, about two orders of magnitude larger in storage (0.94 PB against 7.2 TB).

Table 8: Compute and storage for this study, by stage, on NVIDIA A100-80GB GPUs. Alignment and embedding are one-off preprocessing over the 39,952 pairs of Table 2 and are reused by every run; training and evaluation are repeated per configuration. Per-run figures are medians over completed jobs. Storage follows the symbolic links of the consolidated corpus, so shared records are counted once.

Stage	Per run	Runs	Total
Multiple-sequence alignments	0.46 GPU-h per sequence	1,081 sequences	≈500 GPU-h
Trunk embeddings (uncropped)	20 s per complex	39,952 complexes	≈220 GPU-h
Fine-tuning (10 epochs, 4 GPUs)	≈15 h, 60 GPU-h	272 runs	≈16,300 GPU-h
Evaluation (4 test sets, 1 GPU)	≈1.8 GPU-h	130 runs	≈240 GPU-h
Total compute	—	—	≈17,300 GPU-h
Alignment storage	—	—	27.3 GB
Embedding storage	—	—	7.2 TB
Alignment database	—	—	1.4 TB

D.1 How the source proxy's size was decided

The target set is fixed at 19,985 pairs over 844 proteins; the proxy's protein count is not set to match it, because no mechanism makes 844 the right number for a rehearsal set. Two alternatives were considered and rejected. Matching Boltz-2's own ratio (1.2M pairs over 2k targets, i.e. ∼600 pairs per target) would give ∼100 proteins — defensible, but an order of magnitude narrower than the target, and narrowness is what the method is measured against. Matching 844 exactly would force ∼24 pairs per protein, which means splitting assays apart; since Boltz-2 batches five compounds from one assay and supervises only within-assay differences, that would disable the supervision.

What does have a mechanism is coverage. Fine-tuning perturbs the model where the target lives, and rehearsal can only protect a family it holds data for — which is also how the method is built,

since the group-DRO groups are MMseqs2 30% families. We therefore cluster the 844 target proteins jointly with the 3,522 source proteins carrying at least one single-protein assay of ≥ 10 compounds, using `--min-seq-id 0.3 -c 0.5 --cov-mode 1` (the flags of Boltz-2 Appendix D.2.1), ignoring alignment availability so that compute cost does not decide which families exist. The target occupies 358 families, of which 248 yield source assays passing the A.2.1 hit-to-lead criteria. Assays are then drawn whole, breadth-first over those 248 families, until the pair budget is met; going breadth-first over *proteins* instead overshoots it by half, since 1,426 proteins each owed a whole ten-compound assay come to 30,981 pairs. The quality criteria are applied to source assays only. Target pairs whose family lost its source counterpart are kept — they are different experiments on different chemistry, and their labels show the same spread as the rest of the target set — since filtering the benchmark by what the proxy happens to cover would hide the families for which direct source rehearsal is unavailable. Assays are drawn whole and uniformly under a fixed seed — the IQR weighting of Boltz-2's Algorithm 4 is applied at training time, and applying it here as well would bias the corpus toward activity cliffs twice.

D.2 External benchmarks: which sets, and why those

PepBenchmark (Zhang et al., 2026) standardises 35 peptide datasets — 29 canonical and 6 non-canonical — across seven groups spanning activity, absorption, distribution, metabolism and excretion (ADME), and toxicity. Only one group, PepPI, measures peptide–protein binding, and it holds three datasets: `PpI` (44,148 pairs, binary), `PpI_ba` (1,433, regression on $-\lg M$) and `nc-PpI_ba` (277, regression). We adapt a regression head and report ρ and RMSE, so the binary set is not a valid target; we therefore take *both* regression sets and neither is chosen for how it scores. "Non-canonical" in that benchmark means a peptide that "incorporate[s] cyclic structures or non-canonical amino acid monomers": in `nc-PpI_ba` these are hydrocarbon staples (7%), disulfide bridges (5%), N-methylation (3%) and halogenation (2%) — the modifications medicinal chemists make for protease stability and permeability, which is the chemistry our target domain is drawn from. We note plainly that neither set is a macrocycle benchmark: `PpI_ba` contains no ring of twelve or more atoms and `nc-PpI_ba` only 22.7%. They test whether adaptation carries to peptide chemistry *outside* the macrocycle window, which supports transfer beyond the macrocycle subset rather than macrocycle-specific generalisation.

E Experimental details

This appendix gives the data split and the training and selection protocol summarised in Section 4; Appendix D describes the data themselves.

Protein grouping and split. We operationally define protein families as sequence clusters constructed with MMseqs2 (Steinegger & Söding, 2017), using a minimum sequence-identity threshold of 30% (`--min-seq-id 0.3`) and coverage settings `-c 0.5 --cov-mode 1`. These are sequence-based groups rather than curated biological family annotations. The target corpus is partitioned by these protein groups, so no target test protein has a family represented in target training: 15,516 pairs over 197 families for training, 2,064 over 90 for validation and 2,379 over 68 for test. Three families larger than 8% of the corpus are held in training rather than allowed to land in a held-out set, where they would dominate it. The source stream keeps its own ligand-block split and is unchanged. The two streams share protein families: the source-proxy training stream covers 50 of the 68 target test families and 2,205 of the 2,379 target test records, and 57 target test proteins appear in it with identical sequences. Target test families are therefore unseen by the target stream, not by the model's training as a whole.

Training and selection protocol. All experiments use equal target exposure (`eq`) and a single training seed unless stated otherwise. Target-only FT, L^2-SP, and EWC draw 7,758 examples per epoch; rehearsal methods draw 15,516 with a 1:1 source–target mixture. Thus all methods receive about five target passes, while rehearsal methods additionally draw 77,580 source examples. This matches target exposure, not total draws or computation. Target pairwise supervision is enabled, and the absolute, pairwise, and binary loss weights are 0.5, 2.0, and 1.0. Early stopping is disabled, so every run completes all of its 10 epochs.

Protocol IV selects among retained checkpoints, including the final checkpoint, by median within-family Spearman correlation on the complete target validation set with augmentation disabled. Families need at least ten records and a defined correlation to enter the median. Each selected checkpoint is used unchanged on all four test sets. Table 3 reports the same family-level statistic on the test sets (Fam. ρ) together with metrics over all test records. Root-mean-square error (RMSE) includes all labeled records. The pretrained reference needs no adaptation or any checkpoint selection.

RPR uses $\lambda = 1$, $\epsilon = 0$ and $\rho = 0.9$ throughout. Tables 3 and 4 use $\eta = 0.4$ from the step-size sweep; the other hyperparameter studies use $\eta = 1.0$, and Table 6 and Figure 4 report every run around it.

Implementation details. All methods run in the same Boltz-2 training code and differ only in the terms of Table 9, whose values are used in every experiment. Only the two affinity heads are trained (about 8.7M parameters); the trunk and confidence module stay frozen. Optimization uses Adam ($\beta_1 = 0.9$, $\beta_2 = 0.95$, $\epsilon = 10^{-8}$, no weight decay) with peak learning rate 10^{-4}, reached after 200 linear warm-up steps and multiplied by 0.95 every 2,000 steps after step 2,000. Each of 4 GPUs processes 5 complexes per step (effective batch 20) with at most 352 tokens per complex, bf16 mixed precision and gradient-norm clipping at 10. The affinity loss uses the weights above, with a Huber absolute term ($\delta = 0.5$), a focal binary term ($\alpha = 0.25$, $\gamma = 2$) and a within-assay pairwise term capped at 50 pairs.

FT, L^2-SP and EWC train on target batches only. L^2-SP and EWC add a penalty on the trainable parameters relative to their pretrained values θ_0. For EWC, the diagonal Fisher is estimated at θ_0 as the mean squared gradient of the training affinity loss over 3,000 source-proxy training batches and rescaled to mean one, so that λ has the same scale as in L^2-SP. The rehearsal methods (OD, UR, AGEM and RPR) share the replay stream of UR, which alternates target and source assay batches 1:1, and OD and AGEM keep the labelled affinity loss on source batches. OD adds a squared-error term pulling the predictions of both affinity heads on source batches toward those of a frozen copy of the pretrained heads. AGEM follows AGEM (Chaudhry et al., 2018): it keeps the 32 most recent source batches as episodic memory, computes a reference gradient on one randomly drawn memory batch before each optimizer step, averages it across GPUs, and projects the update onto the half-space in which the inner product with this reference gradient is non-negative. RPR caches the pointwise loss of every source record at θ_0 in one pass before training and applies Equations 8–9 over the $G = 248$ source families.

Table 9: **Method-specific terms and hyperparameters.** All methods share the optimization settings above. θ_0: pretrained parameters; F_i: normalized diagonal Fisher; f_θ, f_{θ_0}: predictions of the trained and the frozen pretrained affinity heads.

Method	Additional term	Hyperparameters
FT	—	—
L^2-SP	$\lambda \sum_i (\theta_i - \theta_{0,i})^2$	$\lambda = 1$
EWC	$\lambda \sum_i F_i (\theta_i - \theta_{0,i})^2$	$\lambda = 2$; Fisher from 3,000 source-proxy batches
OD	$\lambda \lVert f_\theta - f_{\theta_0} \rVert_2^2$ on source batches	$\lambda = 1$; source:target = 1:1
UR	labelled loss on source batches	source:target = 1:1
AGEM	AGEM gradient projection	memory of 32 source batches; source:target = 1:1
RPR	Equation 9	$\lambda = 1$, $\epsilon = 0$, $\rho = 0.9$, $G = 248$; $\eta = 0.4$ (Tables 3 and 4), $\eta = 1.0$ (other analyses)

F COMPLETE EXPERIMENTAL RESULTS

The main text reports a selected subset of metrics for each table so that the comparisons stay readable. This appendix gives the complete results for every run. Tables 10 and 13 add Pearson r, RMSE and AUC on the external benchmarks, Table 11 re-evaluates the target test set with censored labels handled explicitly, Table 12 adds all metrics for the ablation, and Table 14 lists every hyperparameter-sensitivity run. All runs use protocol IV selection and equal target exposure, except the longer 1:0.5 run of Table 5.

Table 10: **External benchmarks, all metrics.** The checkpoints of Tables 3 and 6, protocol IV. Table 3 gives only Spearman ρ for these sets; this table adds Pearson r, RMSE and AUC. PpI_ba has 1,256 scored records, nc-PpI_ba 277, and All combines them ($n = 1533$); no family-level statistic is defined, since neither benchmark is grouped by protein family. Bold marks the best value per column.

	PpI_ba ($n = 1256$)				nc-PpI_ba ($n = 277$)				All ($n = 1533$)			
Method	$\rho\uparrow$	$r\uparrow$	RMSE↓	AUC↑	$\rho\uparrow$	$r\uparrow$	RMSE↓	AUC↑	$\rho\uparrow$	$r\uparrow$	RMSE↓	AUC↑
Pretrained Boltz-2 (no FT)	0.200	0.208	1.879	0.578	0.190	0.177	1.962	0.603	0.157	0.148	1.894	0.557
FT (Amini et al., 2026)	0.273	0.294	1.618	0.616	0.313	0.255	1.904	0.681	0.300	0.299	1.673	0.635
L^2-SP (Xuhong et al., 2018)	0.194	0.189	1.916	0.570	0.232	0.169	2.175	0.614	0.202	0.184	1.965	0.577
EWC (Kirkpatrick et al., 2017)	0.179	0.158	1.649	0.558	0.219	0.147	2.145	0.614	0.170	0.140	1.749	0.558
OD (Li & Hoiem, 2017)	0.246	0.230	1.854	0.597	0.286	0.269	1.843	0.639	0.229	0.204	1.852	0.589
UR (Chaudhry et al., 2019)	0.152	0.153	1.948	0.544	0.261	0.230	1.915	0.625	0.181	0.171	1.942	0.563
AGEM (Chaudhry et al., 2018)	0.214	0.201	2.200	0.576	0.291	0.243	2.054	0.642	0.237	0.212	2.175	0.593
RPR (ours)	0.316	0.346	1.605	0.637	0.323	0.289	1.800	0.672	0.317	0.325	1.642	0.640
RPR ($\eta = 0$)	0.210	0.192	1.946	0.584	0.207	0.222	1.879	0.600	0.197	0.182	1.934	0.579
RPR ($\eta = 0.1$)	0.275	0.284	1.686	0.614	0.343	0.322	1.767	0.661	0.299	0.300	1.701	0.627
RPR ($\eta = 0.2$)	0.215	0.232	1.503	0.586	0.306	0.281	1.789	0.636	0.253	0.268	1.558	0.605
RPR (ours, $\eta = 0.4$)	0.316	0.346	1.605	0.637	0.323	0.289	1.800	0.672	0.317	0.325	1.642	0.640
RPR ($\eta = 0.6$)	0.302	0.326	1.616	0.633	0.314	0.293	1.804	0.662	0.298	0.300	1.652	0.631
RPR ($\eta = 0.8$)	0.187	0.149	2.267	0.569	0.303	0.292	1.956	0.650	0.205	0.167	2.214	0.581
RPR ($\eta = 1.0$)	0.276	0.305	1.729	0.615	0.299	0.237	1.851	0.665	0.279	0.281	1.752	0.619

Table 11: **Censoring-aware target evaluation.** The checkpoints of Table 3 on the target test set. Of its 2,379 labels, 1,831 are exact and 548 are bounds (516 with $>$, 32 with $<$); the source proxy keeps exact labels only. Exact: metrics over exact labels, with Fam. ρ over the 26 families that keep at least ten exact records. Censoring-aware: a bound contributes zero error when the prediction satisfies it and its distance to the bound otherwise; AUC uses the 2,299 records whose class (active below 1 μM) is fixed by the label and its qualifier. Sat.: share of bounds satisfied by the prediction. Bold marks the best value per column.

	Exact labels ($n = 1831$)					Censoring-aware ($n = 2379$)		
Method	Fam. $\rho\uparrow$	$\rho\uparrow$	$r\uparrow$	RMSE↓	AUC↑	RMSE↓	AUC↑	Sat.
Pretrained Boltz-2 (no FT)	0.045	0.276	0.289	**1.371**	0.628	1.487	0.637	16%
FT	0.147	0.247	0.264	1.428	0.615	1.567	0.619	13%
L^2-SP	0.184	0.167	0.201	1.562	0.580	1.643	0.588	19%
EWC	0.232	0.243	0.272	1.683	0.621	1.616	0.634	31%
OD	0.171	0.233	0.259	1.384	0.601	1.502	0.612	16%
UR	0.170	0.225	0.255	1.383	0.607	1.529	0.605	16%
AGEM	0.126	0.153	0.206	1.439	0.563	1.700	0.569	12%
RPR (ours)	**0.274**	**0.312**	**0.327**	1.424	**0.645**	**1.470**	**0.648**	18%

Table 12: **Ablation study, all metrics.** The runs of Table 4, which reports only Fam. ρ and ρ; this table adds Pearson r, RMSE and AUC on both internal test sets, together with the combined external ρ. Bold marks the best value per column.

	Small-molecule source ($n = 1909$)					Macrocycle target ($n = 2379$)					External
Method	Fam. $\rho\uparrow$	$\rho\uparrow$	$r\uparrow$	RMSE↓	AUC↑	Fam. $\rho\uparrow$	$\rho\uparrow$	$r\uparrow$	RMSE↓	AUC↑	All $\rho\uparrow$
UR (no q signal)	0.307	0.508	0.477	1.335	0.759	0.114	0.216	0.229	1.559	0.593	0.181
Group DRO (absolute q signal)	0.350	0.543	0.513	1.221	0.772	0.094	0.286	0.297	1.507	0.637	0.278
RPR (uniform q signal, $\eta = 0$)	0.437	0.517	0.510	1.239	0.774	0.079	0.250	0.268	1.558	0.617	0.197
RPR (absolute q signal)	0.339	0.472	0.451	1.261	0.746	**0.268**	**0.349**	**0.356**	1.496	**0.679**	0.113
RPR (family-balanced sampler)	0.433	0.550	0.534	1.197	0.796	0.219	0.231	0.261	1.589	0.603	0.272
RPR (absolute loss)	0.380	0.556	0.537	1.198	0.781	0.116	0.296	0.322	**1.489**	0.630	0.276
RPR (no source pairwise term)	0.390	0.568	0.569	1.175	0.799	0.174	0.309	0.322	1.524	0.649	0.276
RPR (all components, $\eta = 0.4$)	**0.485**	**0.611**	**0.593**	**1.144**	**0.814**	0.255	0.312	0.324	1.497	0.640	**0.317**

Table 13: **Source-ratio study, external benchmarks.** The runs of Table 5, which reports only Spearman ρ on the external sets. Columns as in Table 10. Bold marks the best value per column.

	PpI_ba ($n = 1256$)				nc-PpI_ba ($n = 277$)				All ($n = 1533$)			
Target:source	$\rho\uparrow$	$r\uparrow$	RMSE↓	AUC↑	$\rho\uparrow$	$r\uparrow$	RMSE↓	AUC↑	$\rho\uparrow$	$r\uparrow$	RMSE↓	AUC↑
RPR (1:1)	0.276	0.305	1.729	0.615	0.299	0.237	1.851	0.665	0.279	0.281	1.752	0.619
RPR (1:0.75)	0.251	0.237	2.114	0.604	0.312	0.280	1.901	0.662	0.252	0.229	2.077	0.606
RPR (1:0.5)	0.258	0.260	1.546	0.614	0.258	0.250	1.816	0.633	0.235	0.228	1.598	0.601
RPR (1:0.5, 7,760 steps)	0.276	0.303	1.598	0.628	0.274	0.257	1.865	0.647	0.239	0.244	1.650	0.607
RPR (1:0.25)	0.214	0.198	1.578	0.582	0.325	0.290	1.790	0.659	0.252	0.240	1.618	0.604
FT (1:0)	0.273	0.294	1.618	0.616	0.313	0.255	1.904	0.681	0.300	0.299	1.673	0.635

Table 14: **All hyperparameter-sensitivity runs of Figure 4.** Protocol IV, Spearman ρ. One hyperparameter varies while the others stay at the $\eta = 1.0$ configuration ($\eta = 1.0$, $\lambda = 1$, $\epsilon = 0$, $\rho = 0.9$), which is repeated in each block and marked by $\star$. "Sel." is the selected step and "All" is as in Table 3. Bold marks the best value per column.

Hyperparameter	Value	Sel.	Source	Target	PpI_ba	nc-PpI_ba	All
Step size η	0	2,716	0.517	0.250	0.210	0.207	0.197
	0.1	6,596	0.600	0.247	0.275	0.343	0.299
	0.2	2,328	0.567	0.061	0.215	0.306	0.253
	0.4	7,760	0.611	0.312	0.316	0.323	**0.317**
	0.6	5,044	0.562	0.341	0.302	0.314	0.298
	0.8	388	0.514	0.218	0.187	0.303	0.205
	1.0*	7,760	0.594	0.346	0.276	0.299	0.279
Source weight λ	0.1	2,328	0.558	0.024	0.214	0.286	0.245
	0.3	7,760	0.547	0.265	0.276	0.333	0.291
	1*	7,760	0.594	0.346	0.276	0.299	0.279
	3	3,492	0.561	0.286	0.286	**0.403**	0.288
	10	7,760	**0.652**	0.335	0.257	0.267	0.264
Tolerance ϵ	0*	7,760	0.594	0.346	0.276	0.299	0.279
	0.05	7,760	0.589	0.305	0.277	0.289	0.283
	0.1	7,760	0.576	0.324	0.270	0.296	0.267
	0.2	388	0.505	0.161	0.219	0.344	0.243
	0.3	7,760	0.561	0.315	0.264	0.263	0.261
EMA decay ρ	0.5	5,044	0.592	0.334	0.315	0.339	0.314
	0.7	5,044	0.562	**0.377**	**0.322**	0.307	0.304
	0.9*	7,760	0.594	0.346	0.276	0.299	0.279
	0.95	4,268	0.488	0.291	0.189	0.286	0.202
	0.99	2,328	0.547	0.162	0.280	0.335	0.295