Problem: The accuracy of protein embedding clustering is fundamentally limited by its reliance on a single, global similarity threshold (ST), which determines the cut off place of clustering protein embeddings. This static approach fails to account for the inherent variability in evolutionary divergence and sequence conservation among different protein families, leading to suboptimal group assignments. This is supported by the figure below, which shows that different homologous protein families break apart at different STs. Indicating that using a single global ST may prove impossible when clustering different homolog groups.

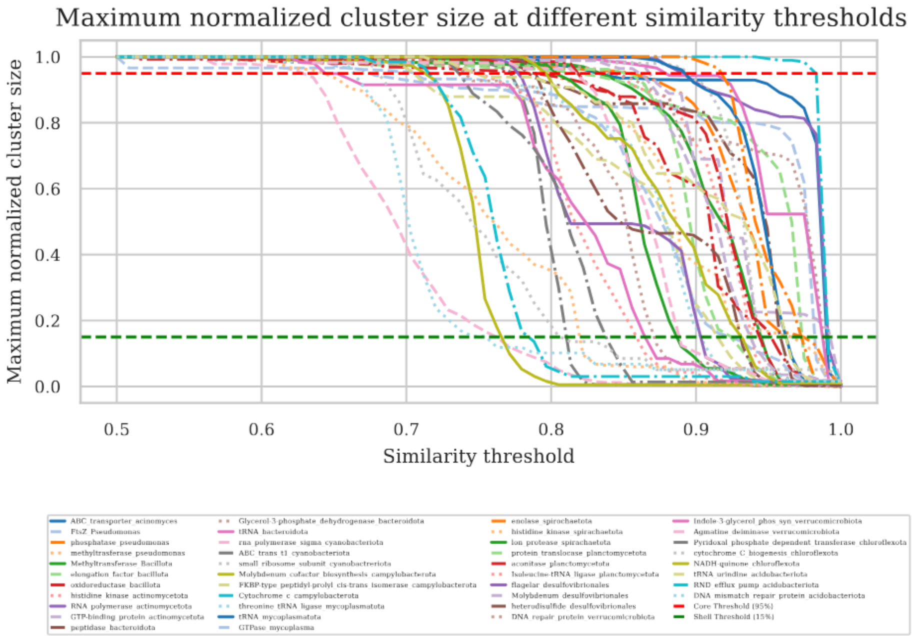


Fig. 3. Impact of similarity thresholds on cluster formation. The figure demonstrates how different thresholds lead to varying classifications, with a value of 1 producing separate clusters (singletons) and a value of 0 merging all genes into one cluster.

Hypothesis: The optimal similarity threshold is not a global constant but a local property of the protein embedding space. We hypothesize that this property can be modeled based solely on the position in the embedding.

Aim 1: Characterize Optimal STs. For a set of known BUSCO homolog groups, we will computationally determine the optimal ST for each group, defined as the highest threshold under which the group forms a cluster with ≥95% of members assigned to a single cluster.
Aim 2: Model the ST Landscape. We will train a Gaussian Process regression model (GP) to learn the mapping from a protein's embedding to the optimal ST of its parent group.
Aim 3: Validate the Adaptive Clustering Approach. We will evaluate the model using leave-one-group-out cross-validation (CV), where a GP is trained on all but a held-out BUSCO group and used to predict that ST for that BUSCO group.

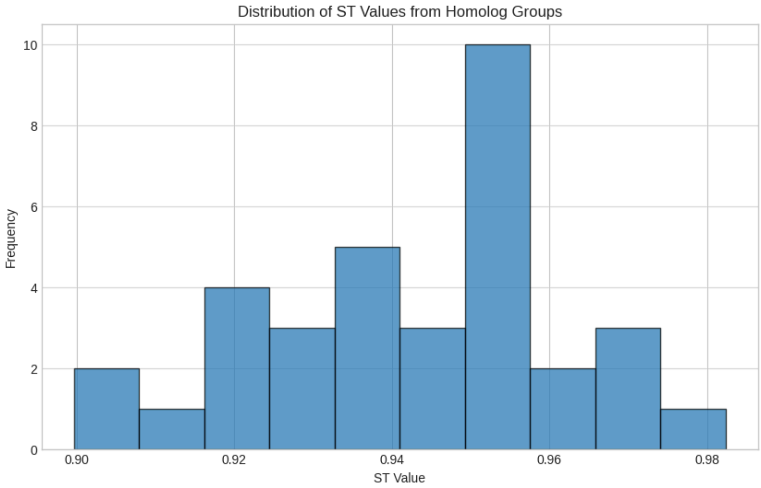
Proposed solution

Given the $O(n^3)$ time complexity of GPs, where n is the number of data points, applying a GP model directly to our full dataset is computationally infeasible. The dataset consists of 350,000 protein embeddings, each with 1024 dimensions, distributed across 32 distinct homologous protein families.

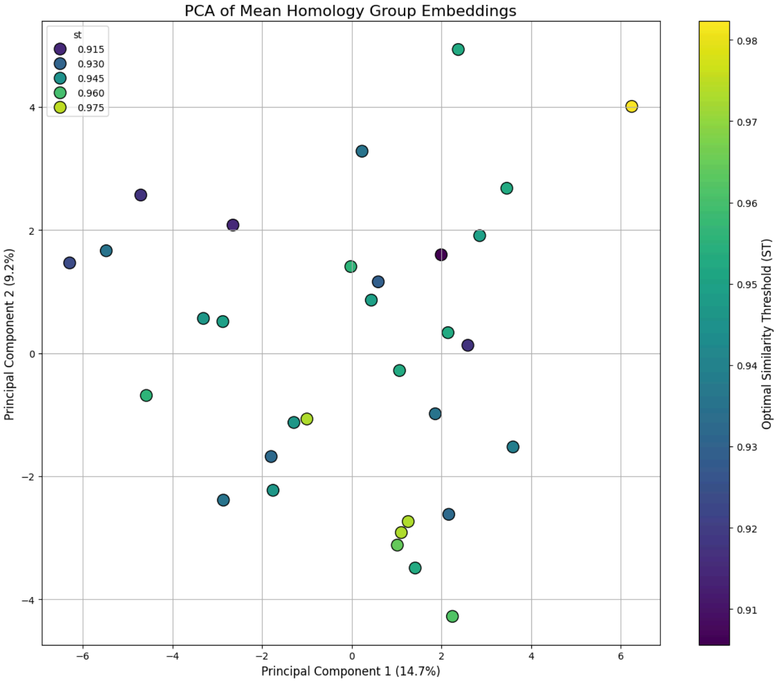
To manage the computational cost, we apply:
First, the number of proteins within each family was reduced to 20. This was achieved by applying either k-means clustering to the embeddings within each family and using the resulting cluster centroids as representative data points. or random sampling 20 embeddings for each group.
Second: we reduced the dimensionality of the protein embeddings by Principal Component Analysis (PCA) from 1024-dimension down to 50 dimensions. This lower-dimensional representation successfully captured 92% of the variance in the original data.

Inspection of data

The distribution of optimale ST can be seen in the figure below. An important observation is that the range of ST is in a short range of the full possible range between 0 and 1.



Mean embedding and optimal ST for each homolog group, projected onto the first two principal components.
A key observation here is the absence of a clear pattern between the embeddings, as represented by the two principal components that explain the most variance.



Baseline

To determine if GP is a good model for solving our problem (*computing the variant, the optimal ST, using homolog protein embeddings as covariant*), do we have to establish a baseline which we can compare to.

Two simple, computationally inexpensive, baselines are chosen. First, just using the mean of all the homolog groups but the group analysed is used as the predicted optimale ST.

The second baseline is a fitting linear regression model, using the same covariants as the later GP.

Rational behind some decisions

Why transformation?

Because the range of variants *the optimale ST* is small: 0,08 is the monotonic logit transformation applied to make this range larger.

Why could GP be a good solution to the problem?

Because a GP is a non-parametric model, meaning that the model has minimal assumptions regarding the underlying data distribution. Does it make sense to use, based on the observation seen in the plot with the placement of homolog groups and their optimal ST on the two first PCs. This is in contrast to the second baseline model linear regression which is a parametric model, with the assumption that the data follows $y=w^tx+b$.
From a bayesian point of view, the benefit of using a GP, that it provides a posterior distribution, which is very powerful. Meaning that for a *new protein embedding* x, the model predicts a Gaussian distribution for its optimal ST, y: $p(y|x,D)=N(\mu ,\sigma^2)$
Where: D is the data, μ is the predicted mean, and σ^2 is the variance or uncertainty.

Why use leave-one-group-out CV as evaluation?

It is known from previous work that the embeddings within a homolog group have a small distance to each other in the embedding space. It is also known that the relative distance in the embedding space between groups is large (i.e., the ratio of the distance within each group to the distance between groups is a small number).
If one were to split the data using, for example, k-fold stratified splits where the stratification is applied to each homolog group, this would result in each group having a portion (e.g., 20% for a 5-fold split) in each test set, while the remainder would be in the training set. This setup means the model will largely rely on interpolation to predict the target property, as the test set contains examples that are highly similar to those in the training set. Arguably, this evaluation strategy is acceptable if the dataset is so large that it densely covers a substantial portion of the relevant embedding space. In such a scenario, most new, unseen data would also likely fall near existing training examples, making interpolation a realistic measure of the model's real-world performance.
Therefore, to avoid an interpolation evaluation, a more stringent strategy based on predicting entirely unseen groups must be used. A method like leave-one-group-out cross-validation is chosen because it provides an estimate of the model's ability to extrapolate and generalize to novel data, even though this typically results in lower measured performance.

Data

Simply because of the size of data, will the 350K fasta files not be uploaded to github, instead will all the cluster files (one for each homolog group) be uploaded.

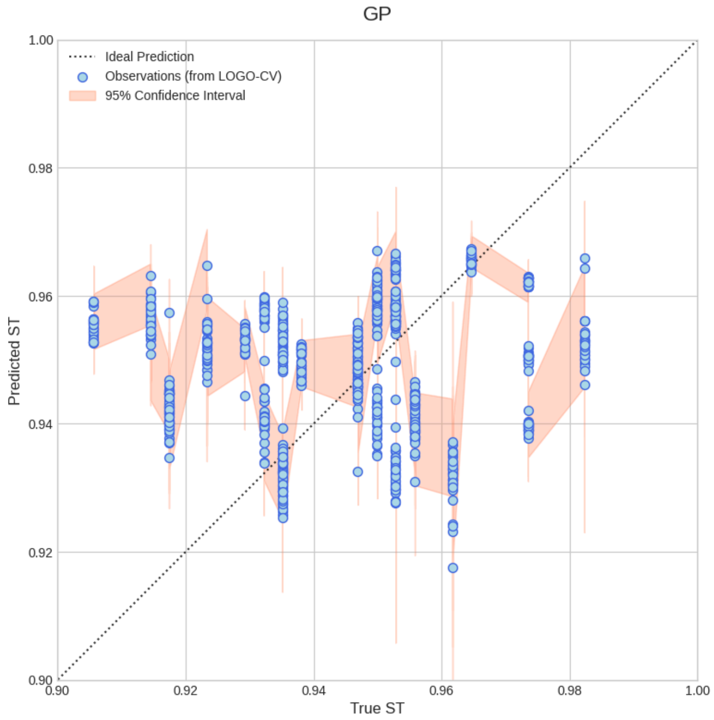
The same for the full 350k embedding files will not be uploaded. Instead will the 20 centroid embeddings per homolog group be uploaded. And the

Results

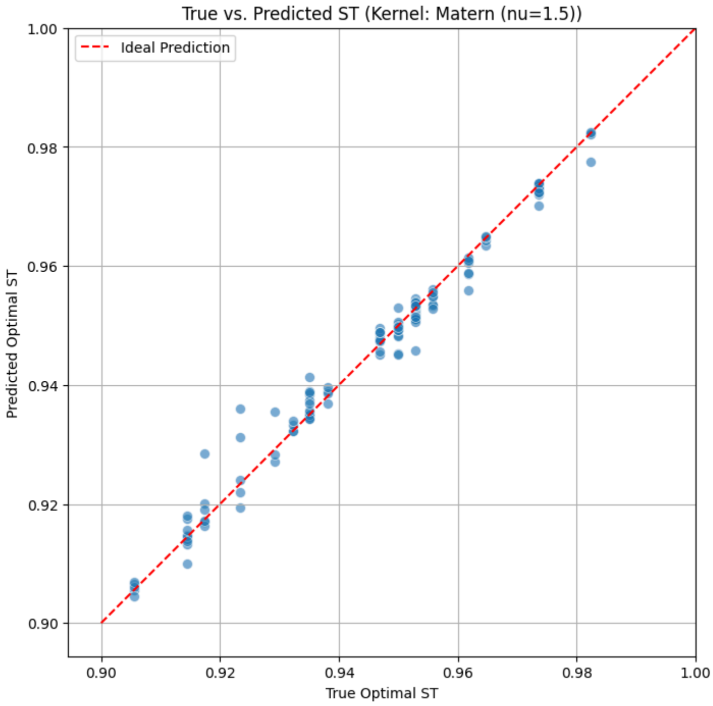
The performance of the GP model and two baseline models was rigorously evaluated using Leave-One-Group-Out cv. The primary goal was to predict the optimal similarity threshold (ST) for a held-out homolog group based on its protein embeddings.

The results indicate that the more complex models struggled to generalize better than the simplest mean model.

Fitting a GP achieved an MSE of **0,000427**.



Biased evaluation, meaning randomly splitting the data, achieves close to perfect score:

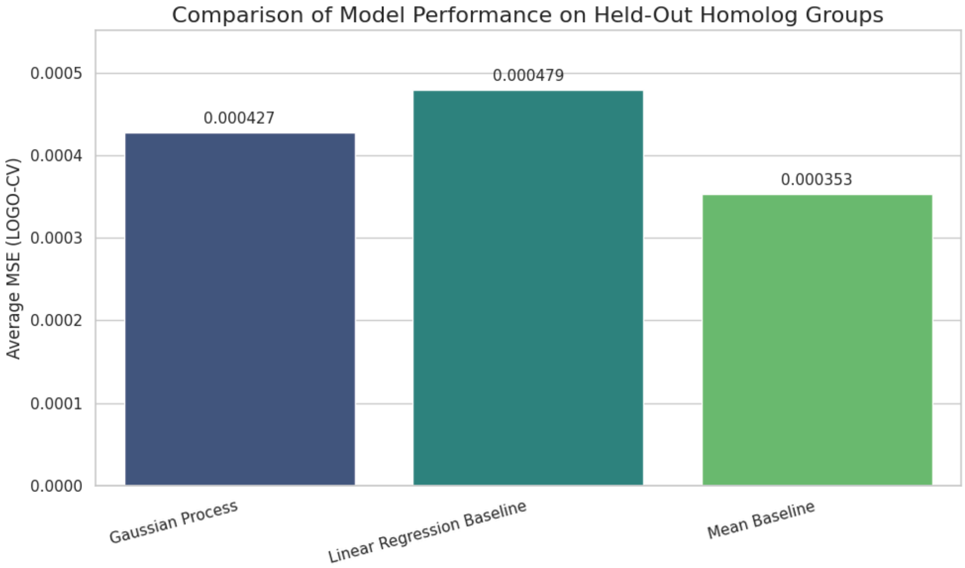


Linear model

When fitting a linear model to the same data. Do we see similar results as for the GP with a MSE of **0,000479**

Mean baseline

When just taking the mean of the ST, and using that as the predicting for every homolog group, do we see surprisingly the best result, of **0,000353**



Conclusion

The initial hypothesis stated that the optimal ST is a local property of the protein embedding space and could be modeled from an embedding's position.

The results of this analysis lead us to reject this hypothesis.

The key finding is that neither the non-parametric GP model nor the Linear Regression model could effectively learn the relationship between homolog group protein embeddings and optimal STs for unseen homolog groups. Both models were outperformed by a simple Mean Baseline, which entirely disregards the embedding information.

When, this is said, may GP perform better if the amount of data is increased. There is a great bias of just using 32 homolog groups to represent the whole bactria embedding space.

The next step will be to use the Kaggle CAFA5 dataset, which consist of all proteins separated in gene ontology class.