M1 BIM Project Report

Exploration of Kinase Protein Sequence Datasets for Generative Modeling

ANTON JESUTHASAN Christine & DAVOUDI Sajjad
2024-2025

Advisors: Giovanni Peinetti, Roberto Netti
Laboratory of Computational and Quantitative Biology (LCQB)
Faculté des Sciences et d'Ingénierie - Sorbonne Université

Contents

Abstract

Protein kinases are essential regulators of cellular signaling cascades, mediating phosphorylation events that control various biological processes. Despite the availability of hundreds of thousands of sequences associated with the kinase domain (Pfam PF00069), a large fraction remain unannotated, and redundant representations complicate conventional supervised learning approaches. This report presents an unsupervised, supervised and semi-supervised deep learning framework to explore kinase sequence space, generate synthetic variants, and improve automated subfamily classification. The work comprises five major phases: preprocessing and encoding of kinase sequences, dimensionality reduction and clustering, development and evaluation of deep generative models (primarily variational autoencoders), synthetic sequence evaluation and augmentation, and finally a classifier design that integrates pseudo-labeled data. Along this project, we leveraged insights from recent literature on generative deep learning for biosequences, transferring methods originally applied in protein engineering to the domain of kinase sequence modeling. [1] [3]

.

1 Introduction

Protein kinases play an essential role in cellular communication by phosphorylating target substrates. Kinases are involved in the regulation of various cellular processes including metabolism, growth, and apoptosis. The amino acid sequences that encode kinase function also encode evolutionary and structural information. However, the kinase family (PF00069) comprises hundreds of thousands of sequences, of which only a small subset ($< 0.5\%$) is labeled regarding subfamily classification. This label scarcity, combined with high redundancy in structural motifs, poses significant challenges to conventional supervised prediction models. In recent years, unsupervised deep learning—as implemented via generative models such as variational autoencoders (VAEs)—has emerged as a promising tool for exploring high-dimensional protein sequence spaces and generating synthetic sequences that preserve key functional characteristics [1]. Furthermore, the application of semi-supervised techniques such as pseudo-labeling has shown potential to enhance classifier performance by integrating synthetic data with limited annotated examples [3].

The aim of this project is to exploit the available PF00069 kinase sequence data by building a pipeline that (i) preprocesses and encodes sequences into

one-hot representations, (ii) performs dimensionality reduction to reveal the intrinsic manifold structure via PCA and t-SNE, (iii) clustering the data using methods like k-means and Gaussian Mixture Models (GMMs), (iv) develops multiple VAE architectures – including standard dense, transformer-based, convolutional, and LSTM-based models – to generate synthetic sequences, and (v) establishes a classification framework that leverages both real and pseudo-labeled data to predict kinase subfamilies. Two classification approaches are applied in parallel: a Logistic Regression model trained on the few available labeled sequences, and an MLP classifier trained on the augmented dataset, incorporating synthetic and pseudo-labeled sequences, to improve subfamily prediction accuracy. The generated synthetic sequences are evaluated for reconstruction accuracy, intra-set diversity, and biological plausibility using tools such as HMMER. This integrative approach not only expands the training dataset but also addresses the inherent challenges of high-dimensional sequence variation and class imbalance [1], [22].

2 Materials and Methods

2.1 Dataset Construction, Alignment Process, One-Hot Encoding

A comprehensive dataset comprising approximately 878,000 kinase sequences was extracted from the PF00069 family on UniProt, with the HMM profile sourced from Pfam. For context, UniProt is a protein sequence database with annotations, while Pfam is a protein family database based on the alignment of conserved protein domains.

The sequences were first aligned using HMMER and then processed with custom Shell scripts to remove insertions, gapped sequences, and duplicates, as we aimed to identify conserved domains within the large kinase family. This preprocessing step resulted in a uniform sequence length of 263 amino acids per sequence. Both labeled sequences (around 3,000 entries spanning 235 distinct kinase subclasses) and unlabeled sequences were retained in the dataset to facilitate unsupervised, supervised and semi-supervised learning.

Each amino acid was encoded as a one-hot vector in a 21-dimensional space (20 amino acids plus one gap character), resulting in tensor representations with a shape of $(M, 263, 21)$, which were then flattened to dimension $(M, 5523)$ for models that require vectorized inputs, where M represents the number of sequences. This encoding strategy preserves both sequential

order and residue identity while enabling the application of deep learning architectures.

2.2 Dimensionality Reduction

Due to the high dimensionality of the one-hot encoded kinase sequences (263 $\times$ 21 = 5523 features), initial dimensionality reduction was performed using Principal Component Analysis (PCA). This allowed us to retain the most informative components while significantly reducing computational complexity. Using just two or three principal components, we could effectively visualize the distribution of sequences and gain a deeper understanding of how they are organized in the sequence space. These visualizations highlighted local patterns and potential clusters that would otherwise be hidden in the original high-dimensional space. To further expose local neighborhood structures and potential nonlinear relationships, we applied t-distributed Stochastic Neighbor Embedding (t-SNE) to the data.

2.3 Clustering and Classification

2.3.1 Unsupervised Techniques: K-Means and Gaussian Mixture Models

We applied both k-means clustering and Gaussian Mixture Models (GMMs) to the flattened sequence embeddings in order to explore the structure of the sequence space and reveal latent patterns. Both algorithms rely on the Expectation-Maximization (EM) principle, an iterative approach used to estimate the parameters by maximizing the likelihood of the data, while considering the most probable cluster structure. In k-means, this principle is applied in a simplified way, while GMMs use probabilistic assignments based on Gaussian distributions.

K-means is a clustering algorithm that partitions the data into k distinct groups based on similarity. It aims to minimize the within-cluster variance by assigning each data point to the nearest cluster centroid. The algorithm works iteratively: it begins by randomly initializing k centroids, then assigns each point to its closest centroid. After that, the centroids are updated as the mean of the points assigned to each cluster. These steps are repeated until convergence (when the assignments no longer change significantly or after reaching a maximum number of iterations).

Gaussian Mixture Models (GMMs) is a probabilistic clustering algorithm

that assumes the data is generated from a mixture of several Gaussian distributions. It estimates the probability that each point belongs to each Gaussian component and updates the parameters of the Gaussian components to better fit the data distribution.

To determine the optimal number of clusters, we used the elbow method and the silhouette score. The elbow method evaluates the inertia (total within-cluster sum of squares) as a function of k, the number of clusters. As k increases, the inertia naturally decreases, but at a certain point, we observe the "elbow" in the curve, which suggests a reasonable trade-off between how compact the clusters are and how easy the results are to interpret. The silhouette score is a metric used to evaluate the quality of clusters created by clustering algorithms. It measures how well samples are clustered with other samples that are similar to each other. It takes into account the average distance within clusters and the average distance to the nearest cluster, with a score ranging from -1 to 1. A score close to 1 means the samples are well-clustered, a score close to 0 indicates overlapping clusters, and a negative score suggests that the samples may have been assigned to the wrong cluster.

2.3.2 Supervised Techniques: Logistic Regression

We aimed to train a Logistic Regression classifier using the 15 most represented labels, which corresponded to 1,757 sequences out of the 3,058 labeled data points available to us. The goal was for the model to predict the subfamily of a given sequence. Given that we had 15 classes, the model was multinomial with a softmax activation function.
Since the number of labeled data was relatively small compared to the entire dataset, we evaluated our trained logistic regression model on a sample of 10,000 random sequences selected from the full dataset (around 878,000 sequences). As we did not have labels for all of these sequences, we queried Uniprot for their predicted labels (since, even in the absence of experimental annotations, Uniprot provides predicted functions for most proteins based on homology, bioinformatics predictions, or other annotations). Finally, we compared the predicted labels from our classifier with the labels from Uniprot.

2.4 Generative Models: Variational Autoencoders

Multiple Variational Autoencoder (VAE) architectures were developed to explore generative modeling in kinase sequence space. In each VAE model, the encoder compresses the flattened one-hot representation into a 32-dimensional

latent vector, while the decoder reconstructs the sequence from the latent space. The following architectures were implemented:

1. Standard VAE: A fully connected network that flattens the input and uses dense layers for both encoding and decoding. This model served as a baseline for reconstruction accuracy and synthetic sequence quality.

2. Transformer-VAE: An architecture that integrates a transformer-based encoder and decoder, leveraging self-attention mechanisms to capture long-range dependencies inherent in sequential data. The transformer component processes unaligned sequences without requiring multiple sequence alignments (MSA), thus retaining the ability to represent subtle variations across kinases [22].

3. ConvVAE: A convolutional VAE using one-dimensional convolutional layers that specifically exploit local sequence motifs and preserve spatial correlation among residues.

To evaluate architectural trade-offs, we implemented and compared multiple VAE variants, aiming to identify the model with the best balance between performance and computational efficiency. Each model utilized a composite loss function combining binary cross-entropy for sequence reconstruction and Kullback–Leibler (KL) divergence to encourage a structured latent space aligned with a standard Gaussian prior. For consistency and comparability, all architectures were trained on an identical set of 10,000 randomly selected kinase sequences. While their reconstruction performance was broadly similar, the standard fully connected VAE demonstrated notably faster training and lower computational overhead, making it the most practical choice for scaling. Based on these findings, we selected the standard VAE and trained it on a significantly larger subset of 200,000 sequences to leverage its representational capacity and generalization potential.

2.5 Evaluation Metrics and Synthetic Sequence Validation

A unified evaluation pipeline was established to assess both reconstruction accuracy and the biological plausibility of the generated synthetic sequences. Performance metrics included:

1. Reconstruction Accuracy: Measured by the mean squared error (MSE) between the original sequence and its reconstruction by the VAE, and quantified using Hamming distance as a sequence similarity measure.

2. Intra-set Diversity: The percentage difference among generated sequences was computed to ensure that the synthetic library covered a broad region of the latent space. A high intra-diversity (70% difference) was targeted to capture the natural variability of kinase sequences.

3. Similarity to Real Data: Minimum Hamming distance calculations between each synthetic sequence and the closest real sequence were performed to quantify how well the generated sequences resemble naturally occurring kinases.

PCA visualization of the latent representations of real versus generated sequences further provided qualitative insights into the placement of synthetic data within the natural sequence manifold.

2.6 Semi-Supervised Learning Strategy

A two-stage classification pipeline was developed to automatically annotate kinase subfamilies. First, the trained VAE encoder was used to embed the one-hot encoded sequences into a low-dimensional latent space. This embedding, which captures both conserved and divergent features of kinase sequences, served as the input to a Multilayer Perceptron (MLP) classifier. The classifier was initially trained on a set of 1,700 labeled sequences, representing 15 distinct kinase subfamilies.

Given the limited labeled data, one approach that we tried was pseudo-labeling and then performing semi-supervised learning. A high-confidence threshold (≥ 0.80) was applied to the already trained MLP model to assign pseudo-labels to a subset of unlabeled kinase sequences, thereby augmenting the training set and balancing class representation. The augmented dataset was constructed via stratified sampling to enforce a maximum of 1,000 sequences per class and was subsequently used to retrain the classifier. This semi-supervised augmentation led to improvements in overall classification accuracy as well as better performance on minority classes.

3 Results and Discussion

3.1 Clustering Performance

3.1.1 Estimation of the Optimal Number of Clusters

We worked with a sample of 10,000 randomly selected sequences due to the large size of the dataset. We began by performing an elbow analysis using both k-means inertia and the silhouette score to determine the optimal number of clusters, testing values of k from 1 to 100. As shown in Figure 1, the automated elbow detection identified $k = 10$ as the optimal number of clusters. However, the silhouette score appears to reach its maximum around $k = 5$. Based on these results, we can assume that the optimal number of clusters lies between 5 and 10.

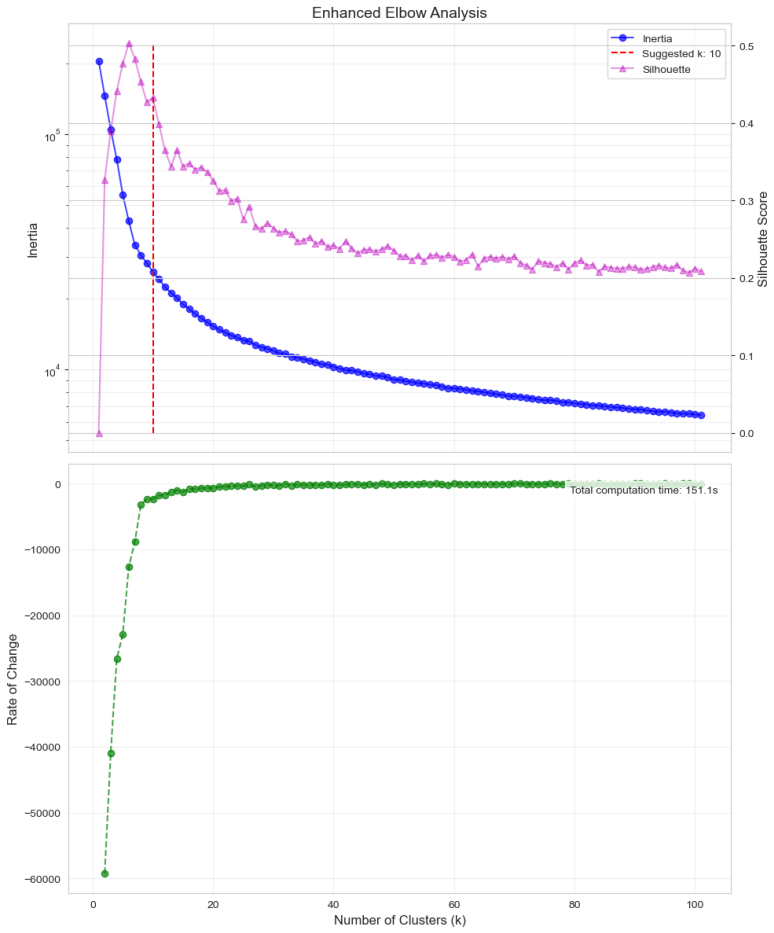
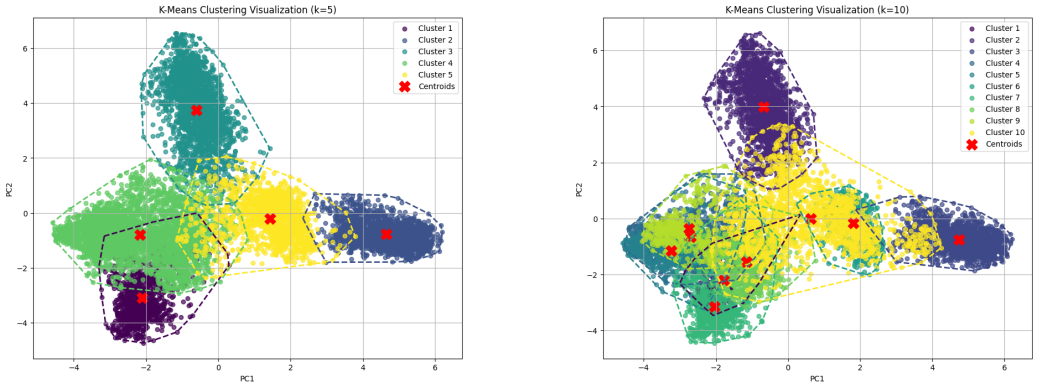


Figure 1: Elbow analysis based on inertia and silhouette score to identify the optimal number of clusters (k).

3.1.2 Projection of K-means and GMM Clusters onto PCA Space

Next, we aimed to visualize the k-means clusters. To do so, we sampled 10,000 sequences (flattened to a shape of (10000, 5523)) and applied dimensionality reduction using PCA with 10 components. We then projected the resulting cluster assignments onto the PCA space for various values of k. Initially, we plotted the clusters on the first two principal components (PC1 vs. PC2), but observed some overlap between clusters, particularly for higher values of k. Figure 2 presents the cluster projections for $k = 5$ and $k = 10$, where slight overlapping can already be noticed in the case of $k = 10$.

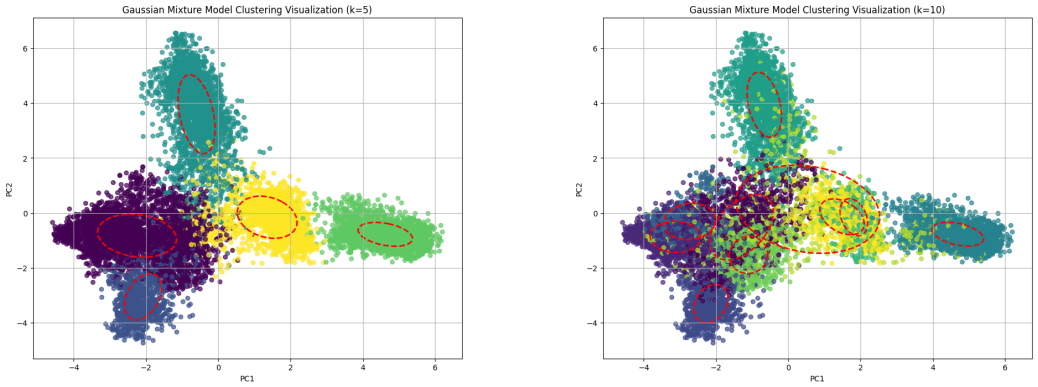


(a) $k = 5$ (b) $k = 10$

Figure 2: K-means cluster projections for $k = 5$ and $k = 10$ on the first two principal components (PC1 vs. PC2). Some overlap is visible, particularly for $k = 10$.

To further explore the distribution of clusters in the reduced space, we created pairwise scatter plots for the first five principal components (e.g., PC1 vs. PC2, PC1 vs. PC3, etc.). For identical component comparisons (e.g., PC1 vs. PC1), we used density histograms. As illustrated in Figure 14, included in the appendix, some overlap remains visible across several component pairs for $k = 10$, suggesting that the clusters are not entirely well-separated in the PCA-reduced space.

We applied the same analysis using GMM clustering and observed similar cluster patterns for $k = 5$ as with k-means. For $k = 10$, some overlap between the clusters remains visible, indicating that the clusters are still not entirely distinct in the reduced PCA space, as shown in Figure 3.



(a) $k = 5$ (b) $k = 10$

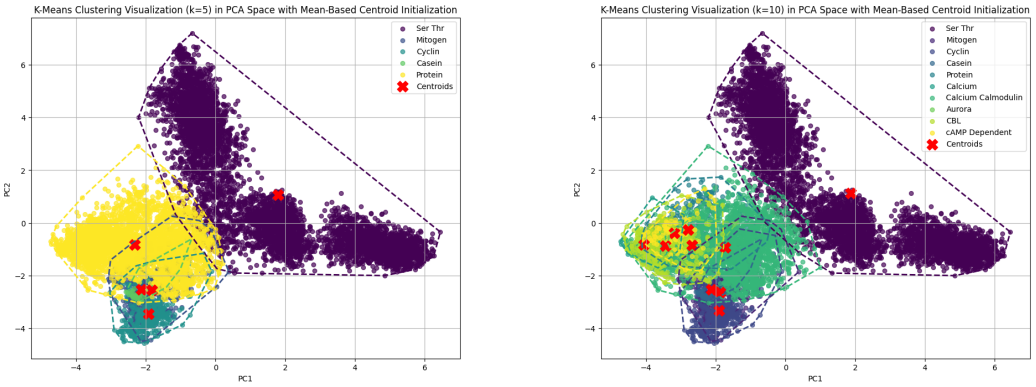
Figure 3: GMM cluster projections for $k = 5$ and $k = 10$ on the first two principal components (PC1 vs. PC2). Some overlap is visible, particularly for $k = 10$.

3.1.3 PCA Projection of K-means Clusters with Mean-Based Centroid Initialization

Then, using the labeled dataset in our possession (approximately 3,000 reviewed sequences from Swiss-Prot), we performed k-means clustering with the same number of clusters as previously ($k = 5$ and $k = 10$). However, instead of using random initialization, we initialized the centroids using the means of the most represented labeled sequences. For $k = 5$, the initial centroids were set as the mean of the sequences from the five most represented functional labels; similarly, for $k = 10$, we used the ten most represented labels.

To achieve this, we grouped sequences by function (e.g., serine/threonine kinases, mitogen-activated kinases, etc.). We used one-hot encoded and flattened sequence matrices, as in previous steps, and focused on the 15 most represented labels. These labels were selected because each had at least 20 associated protein sequences, whereas the remaining 220 labels had very few proteins (typically fewer than 10). As a result, we reduced the label set from 235 to 15, covering a total of 1,757 sequences— a number close to the optimal number of clusters we had previously identified, and far more manageable than the original 235. For $k = 5$, the cluster centroids were initialized with the mean vectors of sequences corresponding to the five most frequent functional labels: serine/threonine kinases, mitogen-activated kinases, cyclin-dependent kinases, casein kinases, and protein kinases. For $k = 10$, the initialization included the previous five, along with: calcium-dependent ki-

nases, calcium/calmodulin-dependent kinases, ribosomal protein S6 kinases, aurora kinases, and CBL-interacting protein kinases.



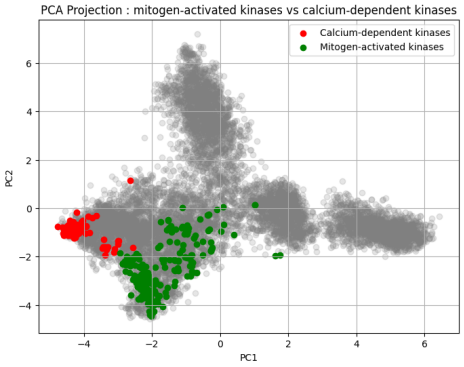
(a) $k = 5$ (b) $k = 10$

Figure 4: K-means cluster projections for $k = 5$ and $k = 10$ in PCA space with mean-based centroid initialization.

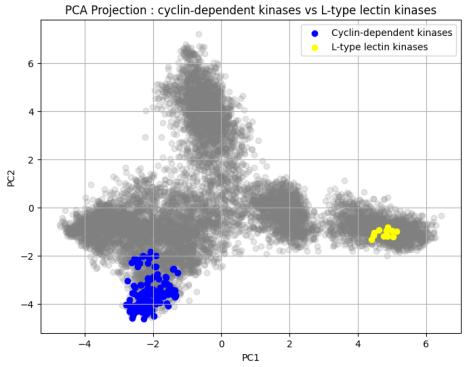
The results of the K-means clustering with mean-based centroid initialization showed clusters that differ from those obtained with random initialization, as illustrated in Figure 4. These clusters show considerably more overlap: while one large cluster is well-separated, the others show significant overlap. Since the mean-based centroid initialization ensures that the starting points are more representative of the data's structure, we can infer that most of the sequences are likely to be similar to one another.

3.1.4 PCA Projections with Labeled Data

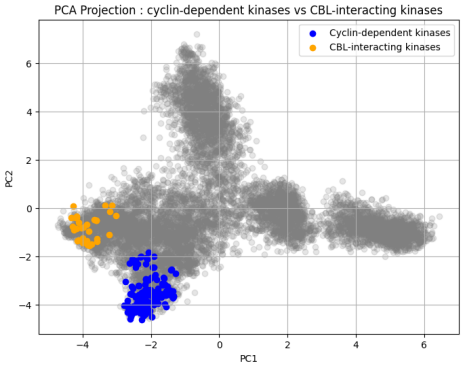
As most of the 15 functional labels contained relatively few sequences (except for serine/threonine kinases, which had 944 out of 1,757 sequences), and due to significant overlap between them, we decided to project the sequences in pairs onto the PCA space. This allowed us to observe their distribution in the PCA space and determine whether functionally related sequences tend to form distinct clusters. In Figure 5,we observe that sequences associated with each functional label cluster in specific regions of the PCA space. This suggests that sequences assigned the same functional label by researchers share underlying features that are effectively captured in the PCA space. These observations help explain why the k-means clustering results did not yield well-separated clusters: either the sequences have only minor variations across functions, or the limited number of labeled sequences reduces the clustering resolution.



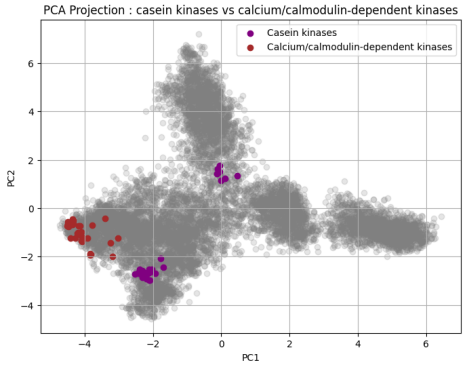
(a) Mitogen-activated vs Calcium-dependent kinases



(b) Cyclin-dependent vs L-type Lectin kinases



(c) Cyclin-dependent vs CBL-interacting kinases



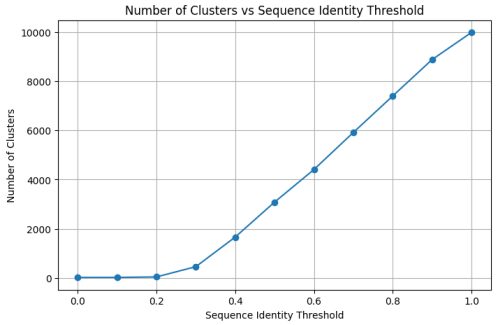
(d) Casein vs Calcium Calmodulin-dependent kinases

Figure 5: PCA projection of labeled data in pairs: Mitogen-activated kinases vs Calcium-dependent kinases, Cyclin-dependent kinases vs L-type Lectin kinases, Cyclin-dependent kinases vs CBL-interacting kinases, Casein kinases vs Calcium Calmodulin-dependent kinases.

3.1.5 Clustering Analysis of Kinase Sequences Based on Sequence Identity

In this final clustering experiment, we aimed to evaluate the similarity of kinase sequences within our dataset. To do so, we computed the sequence identity within a sample of 10,000 random sequences taken from the original dataset of 800,000 sequences. We varied the sequence identity threshold to examine how it affected the clustering results. The plot and table in Figure 6 show the relationship between the number of clusters formed and the

sequence identity threshold. As observed, the number of clusters increases significantly as the sequence identity threshold increases. This suggests that the kinase sequences in our dataset are not highly similar to each other. Based on the number of clusters observed at different thresholds, we conclude that the average sequence identity between the sequences is likely to be below 20%. This result aligns with our expectations, as protein families such as kinases often exhibit considerable divergence, with an estimated 80% sequence divergence, which corresponds to less than 20% sequence identity.



(a) Number of Clusters vs Sequence Identity Threshold

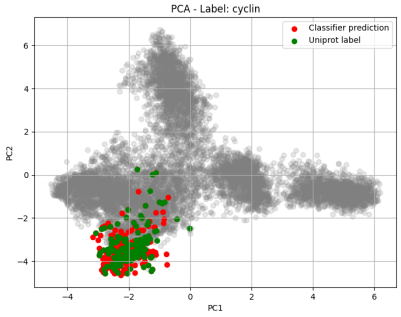
Threshold	Clusters formed
0.00	10
0.10	11
0.20	32
0.30	450
0.40	1657
0.50	3084
0.60	4412
0.70	5922
0.80	7409
0.90	8893
1.00	10000

(b) Clusters Formed Based on Sequence Identity Threshold

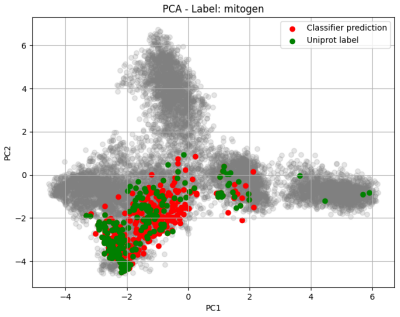
Figure 6: Analysis of kinase sequence clustering based on sequence identity.

3.2 Sequence Classification

We evaluated our logistic regression classifier, trained on 1,757 labeled sequences, by predicting the labels of 10,000 random sequences sampled from the initial dataset. We compared these predicted labels with the reference annotations retrieved from UniProt via query requests. Then, we projected the sequences onto a PCA space to visually assess the clustering behavior. As shown in Figure 7, sequences assigned the same label, whether predicted by our classifier or annotated by UniProt, tend to cluster in similar regions of the PCA space. This suggests that our classifier has learned a biologically meaningful representation of the data and generalizes well to unseen sequences.



(a) PCA projection of sequences labeled as Cyclin.



(b) PCA projection of sequences labeled as Mitogen.

Figure 7: PCA projections of sequences labeled as Cyclin and Mitogen, using both classifier predictions and UniProt annotations. Clustering consistency across both sources suggests strong classifier performance.

3.3 Generative Modeling via VAE

To model the kinase sequence space and generate realistic yet diverse variants, we trained a fully connected Variational Autoencoder (VAE) on 200,000 one-hot encoded sequences sampled from the PF00069 family. This architecture, selected for its training efficiency and reliable convergence, comprised three dense layers in both encoder and decoder, with a 32-dimensional latent space. The VAE architecture comprises two network modules:

- **The Encoder:** This network takes as input the fixed-length representation of the kinase sequences and maps them to a latent space defined by a multivariate Gaussian distribution. The encoder therefore outputs the mean (μ) and variance (σ^2) parameters that characterize the latent distribution, such that each kinase sequence is represented as

$$\mathbf{z} = \mu + \sigma \odot \varepsilon,$$

  where ε is a noise vector drawn from a standard normal distribution $\mathcal{N}(0, I)$ [12].

- **The Decoder:** Given a sample from the latent space, the decoder reconstructs the kinase protein sequence. By optimizing the reconstruction error—commonly the cross-entropy loss when working with one-hot encoded sequences, or mean squared error for continuous representations—the decoder learns the mapping from the latent variables back to the original sequence space [4].

Model performance was evaluated across multiple axes. First, *binary reconstruction accuracy* reached a high average of **97.65%**, indicating the model was able to accurately regenerate input sequences from latent codes. At the sequence level, *Levenshtein distance* between original and reconstructed sequences averaged ˜109 edits per 263-length sample, capturing structural but not excessive divergence. The *cosine similarity* between latent vectors before and after reconstruction was also high (˜0.989), confirming the latent space's consistency and stability.

We then explored the *organization of the latent space for unlabeled data.* Applying **k-means clustering (k=5)** on 10,000 randomly selected latent vectors revealed well-separated, isotropic clusters (Fig. 8), suggesting that even without supervision, the VAE learned to organize kinase sequences in a semantically meaningful manner. These findings support the hypothesis that unsupervised models can capture domain-specific structure purely from sequence information.

Next, we overlaid **the top five kinase classes** from the labeled dataset onto the same PCA plane. Although some classes like tyrosine and serine/threonine kinases showed partial spatial separation, the overlapping nature of the clusters highlighted the functional and evolutionary continuum of kinase subtypes.(Fig. 9)

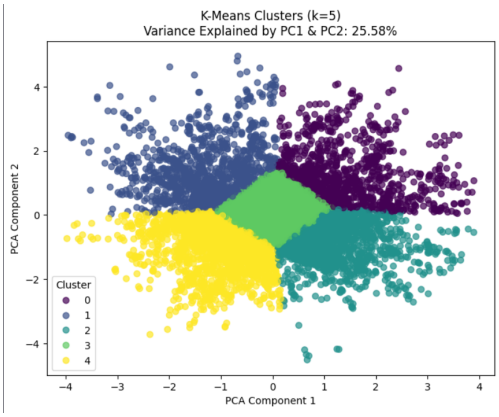


Figure 8: K-means clustering (k=5) applied to 10,000 latent vectors from unlabeled kinase sequences. Clusters show clear separation in the PCA-reduced space.

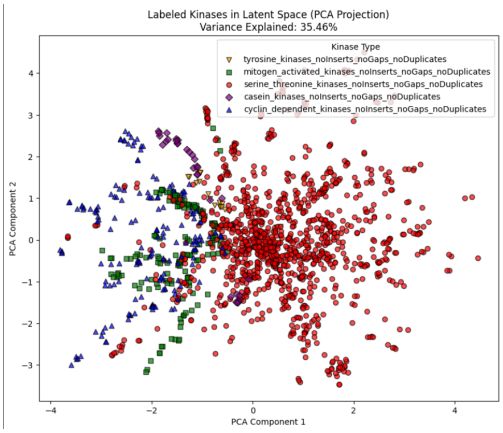


Figure 9: PCA projection of VAE latent vectors for labeled kinase classes. Each point is colored by kinase type. Partial separation is visible among major functional groups.

Together, these quantitative metrics and visual analyses demonstrate that the trained VAE effectively compresses and organizes kinase sequence space,

making it a suitable foundation for downstream generation tasks. In the next section, we shift focus to the **actual generation of novel sequences**, evaluate their quality using statistical criteria, and assess whether the VAE can synthesize kinase-like variants that are both diverse and functional.

3.4 Synthetic Sequence Evaluation

Synthetic Sequence Evaluation was performed in two main parts. In the first part, we generated synthetic kinase sequences directly from the VAE latent space by sampling random vectors, while in the second part, we produced synthetic sequences by reconstructing real kinase inputs. For both approaches, the sequences underwent the same preprocessing steps—namely, one-hot encoding followed by flattening into fixed-dimension vectors—and then were subjected to Principal Component Analysis (PCA) to project them onto a lower-dimensional subspace that captures the maximum variance inherent in the original distribution of kinase sequences .

3.4.1 Latent Sampling Evaluation

We generated 10,000 synthetic sequences by decoding random vectors sampled from the VAE's latent space. These vectors were drawn from a standard multivariate Gaussian distribution, consistent with the prior imposed during training.

To assess how realistic these sequences were, we computed two distance-based metrics using the aligned and fixed-length representations. We first computed the average pairwise Hamming distance among synthetic sequences (intra-diversity), and the average minimum Hamming distance from each generated sequence to its closest real counterpart. These metrics provide insight into how *diverse* the generated sequences are, and how *biologically plausible* they appear relative to natural sequences.

The results were:

- **Average intra-Hamming distance (generated)**: 183.4 / 263 (70%)
- **Average min distance to natural**: 128.9 / 263 (49%)

These values suggest that the generated sequences are highly diverse (avoiding mode collapse), but they remain at a moderate distance from the real data manifold. This could reflect either novel yet plausible variants or overly noisy samples.

For visual assessment, we performed PCA on 100,000 real sequences and the 10,000 latent-sampled synthetic sequences. All sequences were one-hot

encoded and flattened to vectors of length 5523 before dimensionality reduction. The PCA plot (Figure 10) shows that the synthetic sequences broadly overlap with the distribution of real kinase sequences. This behavior indicates that the VAE's latent space captures meaningful features of kinase sequences.

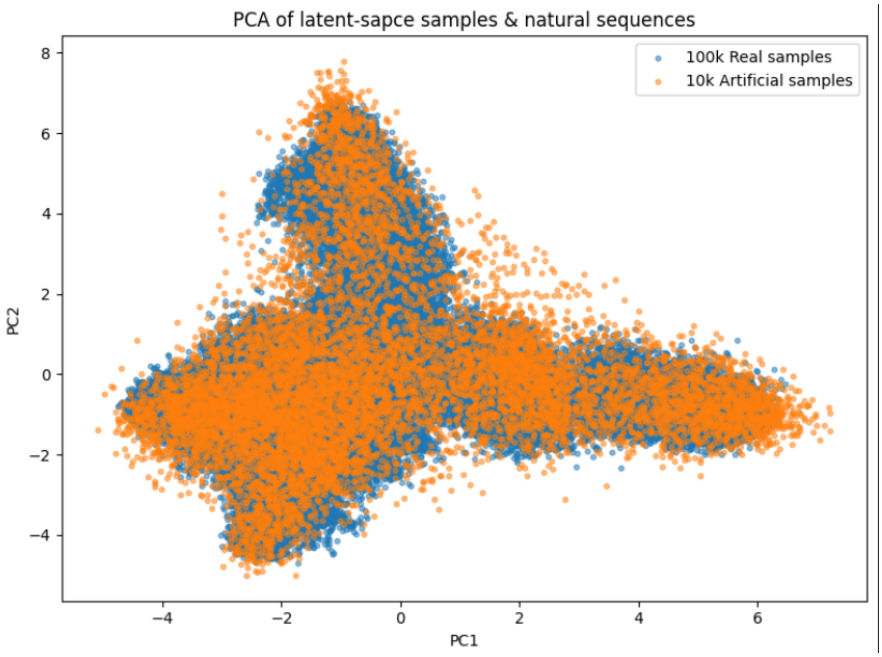


Figure 10: PCA projection of 100k real sequences (blue) and 10k latent-sampled synthetic sequences (orange).

3.4.2 Reconstruction-Based Generation

To evaluate how well the model could reconstruct known sequences with controlled variability, we selected 1,000 real kinase sequences at random and produced 10 reconstructions per input, resulting in 10,000 reconstructed sequences. The decoder was conditioned on each encoded input while sampling from the posterior to introduce variability.

After applying the same one-hot encoding and PCA procedure used in the previous evaluation, we projected these 10,000 reconstructed sequences alongside 100,000 real sequences. As shown in Figure 11, the reconstructed sequences exhibit close spatial alignment with the natural ones, forming dense clusters that remain confined to the original manifold. This suggests that when guided by real examples, the decoder maintains strong structural fidelity, regenerating sequences that preserve domain-relevant motifs

and residue patterns.

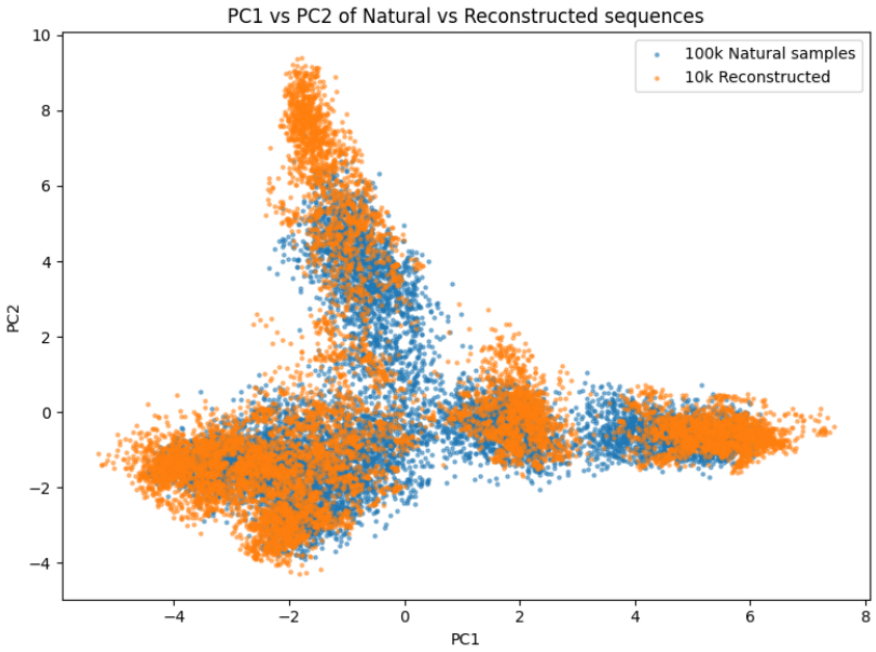


Figure 11: PCA projection of 100k real sequences (blue) and 10k reconstructed sequences (orange) generated from 1k natural inputs.

The combination of distance-based metrics and PCA projections indicates that the VAE effectively learns the structure of kinase sequence space. Latent-sampled sequences show high internal diversity and remain within a biologically relevant range relative to real sequences. Reconstruction-based sequences further confirm the model's ability to regenerate kinase-like sequences that closely follow the natural distribution. Overall, these evaluations demonstrate that the VAE generates synthetic sequences that preserve meaningful biological characteristics and are consistent with known kinase domain structure.

3.5 MLP Classifier Performance and Semi-Supervised Augmentation

The first Multilayer Perceptron (MLP) classifier was trained on 1,757 labeled kinase sequences spanning 15 subfamilies. The model was evaluated on a hold-out test set consisting of 170 sequences and achieved strong results. As shown in Figure 12 , the model reached an accuracy of 82.1% and

a weighted F1-score of 0.826. Despite the limited training data, the classifier performed well across most classes, though some confusion occurred in minority subfamilies.

Table 1: Summary of classification performance on test set (352 samples).

Metric	**Precision**	**Recall**	**F1-Score**
Macro Average	0.7998	0.6863	0.7214
Weighted Average	0.8845	0.8977	0.8834
Accuracy	**89.77%**		

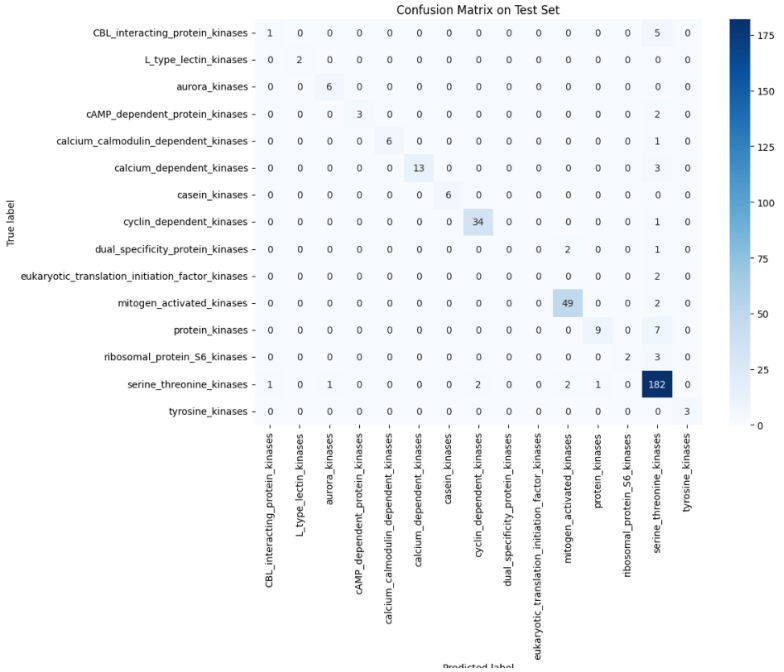


Figure 12: Confusion matrix for the first MLP classifier trained on 1,757 labeled sequences.

3.5.1 Augmentation via Pseudo-Labeling and VAE-Generated Sequences

To improve model generalization and mitigate class imbalance, we augmented our training dataset by combining pseudo-labeled and VAE-generated sequences. First, high-confidence predictions (threshold ≥ 0.80) from the initial MLP were used to assign pseudo-labels to previously unlabeled sequences. Second, synthetic sequences were generated using the trained VAE decoder and added to the dataset with appropriate labels.

The final dataset consisted of 3,271 sequences in the training set, 818 sequences in the validation set, and 1,700 sequences in the test set (the original labeled data from the initial training stage).

We then trained a second MLP classifier on the new training set and evaluated it on the same 1,700 labeled sequences. The model demonstrated improved performance, achieving a test accuracy of 89.8% and a weighted F1-score of 0.883, as shown in table 2. The results reflect a more balanced and consistent performance across all kinase subfamilies.

Table 2: Summary of classification performance on test set (1,757 samples).

Metric	**Precision**	**Recall**	**F1-Score**
Macro Average	0.7700	0.8953	0.8132
Weighted Average	0.8504	0.8213	0.8265
Accuracy	**82.13%**		

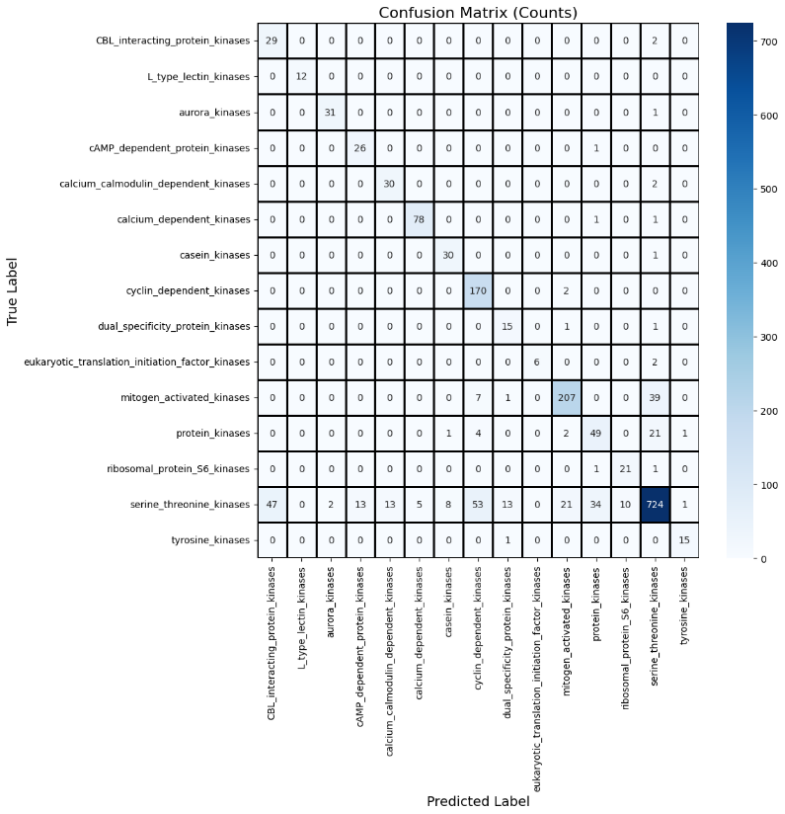


Figure 13: Confusion matrix for the second MLP classifier trained on augmented data and evaluated on the original labeled set.

These results show that the second model, trained on pseudo-labeled and VAE-generated sequences, maintains strong performance when evaluated on a larger test set. However, whether this augmentation strategy provides a consistently reliable improvement requires further investigation.

3.6 Limitations and Future Work

Several limitations emerged during the development of this project. First, the labeled dataset was highly imbalanced: while certain kinase subfamilies were represented by hundreds of labeled sequences, others contained only a few. This imbalance negatively impacted classifier performance for minority classes and limited the reliability of supervised learning outcomes.

Second, the overall number of labeled sequences was small relative to the size of the unlabeled kinase dataset. This constrained the potential of purely supervised methods and necessitated the use of semi-supervised strategies such as pseudo-labeling and data augmentation.

Third, identifying kinase-specific sequence motifs proved challenging using unsupervised clustering approaches. The subtlety and local variability of functionally relevant patterns are not always captured well by global distance-based methods such as k-means or Gaussian Mixture Models, especially when applied to one-hot encoded or flattened representations.

Future work may focus on addressing these limitations by integrating more sophisticated representations (e.g., transformer embeddings or motif-aware encodings), incorporating domain-specific prior knowledge, or applying targeted feature selection. Additionally, expanding the labeled dataset and employing more advanced techniques for motif detection—such as attention-based models or domain-informed contrastive learning—may improve both clustering quality and classifier generalization, particularly for rare subfamilies.

4 Conclusion

To conclude, we conducted a systematic analysis on both labeled and unlabeled data. Clustering and modeling the unlabeled data proved to be a challenging task due to the large size of the dataset. However, our clustering analysis revealed the presence of 5 to 10 distinct clusters. Although these clusters were not immediately apparent, our analysis supports their existence in the unsupervised setting.

For the supervised part, the logistic regression classifier yielded strong performance and demonstrated good generalization to unseen data. We believe that with more labeled data, the model's performance could be further improved and potentially enable even more robust predictions.

Beyond logistic regression, we developed a generative modeling pipeline based on a Variational Autoencoder (VAE), which was able to learn meaningful latent representations of kinase sequences. The VAE was used to generate synthetic data through both latent sampling and reconstruction. Our evaluations showed that the synthetic sequences preserved core biological features and maintained structural alignment with natural kinase data in PCA space. Hamming distance metrics also confirmed their plausibility and diversity.

These generative capabilities were leveraged in a semi-supervised learning framework. We combined pseudo-labeled sequences and VAE-generated sequences to augment the training set and retrain a Multilayer Perceptron classifier. The final classifier had consistent performance across kinase subfamilies, suggesting that data augmentation using generative and semi-supervised methods is a promising approach to mitigate label scarcity.

Overall, this project demonstrated how deep learning methods—both generative and discriminative—can be applied to large-scale, partially labeled biological datasets. The VAE proved effective not only as a compression and generation model but also as a source of informative representations for classification. Future work should focus on refining motif detection, expanding labeled data, and incorporating domain-specific priors to enhance interpretability and functional prediction in kinase subfamily classification.

5 Appendix

5.1 Additional PCA Cluster Projections

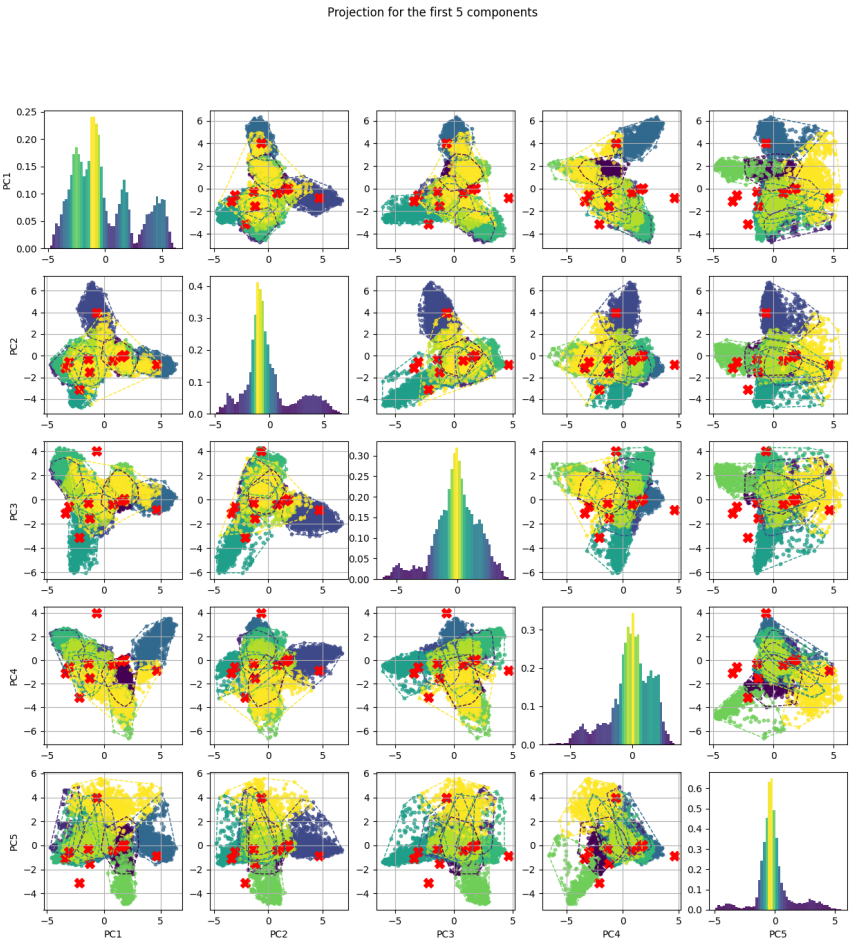


Figure 14: Pairwise PCA projections for $k = 10$, showing PC1 to PC5. Overlapping clusters are still visible.

6 References

[1] Srisairam Achuthan et al. "Leveraging deep learning algorithms for synthetic data generation to design and analyze biological networks". In: *Journal of Biosciences* (2022).

[2] Lubos Cipak. "Protein Kinases: Function, Substrates, and Implication in Diseases". In: *International Journal of Molecular Sciences* (2022).

[3] Iman Deznabi et al. "DeepKinZero: Zero-Shot Learning for Predicting Kinase-Phosphosite Associations Involving Understudied Kinases". In: *bioRxiv* (2019).

[4] Carl Doersch. "Tutorial on variational autoencoders". In: *arXiv preprint arXiv:1606.05908* (2016).

[5] EMBL-EBI. *Pfam: PF00069 – Protein kinase domain*. https://www.ebi.ac.uk/interpro/entry/pfam/PF00069/. 2024.

[6] Hao Fu et al. "Learning protein fitness models from evolutionary and assay-labeled data". In: *Nature Biotechnology* (2024). URL: https://pubmed.ncbi.nlm.nih.gov/38720073/.

[7] Yang Hao et al. "Developing a Semi-Supervised Approach Using a PU-Learning-Based Data Augmentation Strategy for Multitarget Drug Discovery". In: *International Journal of Molecular Sciences* (2024).

[8] Alex Hawkins-Hooker et al. "Generating functional protein variants with variational autoencoders". In: *bioRxiv* (2020).

[9] Jesse Horne and Diwakar Shukla. "Recent Advances in Machine Learning Variant Effect Prediction Tools for Protein Engineering". In: *Industrial & Engineering Chemistry Research* (2022).

[10] Hyosoon Jang et al. "De novo drug design through gradient-based regularized search in information-theoretically controlled latent space". In: *Journal of Computer-Aided Molecular Design* (2024).

[11] Keras Team. *Keras Documentation*. https://keras.io/. 2024.

[12] Diederik P Kingma and Max Welling. "Auto-encoding variational Bayes". In: *Proceedings of the 2nd International Conference on Learning Representations (ICLR)* (2014).

[13] Yan Li et al. "Searching for protein variants with desired properties using deep generative models". In: *BMC Bioinformatics* (2023).

[14] Xinran Lian et al. "Deep learning-enabled design of synthetic orthologs of a signaling protein". In: *bioRxiv* (2022).

[15] Evgenii Lobzaev and Giovanni Stracquadanio. "Dirichlet latent modelling enables effective learning and sampling of the functional protein design space". In: *Nature Communications* (2024).

[16] Alireza Makhzani et al. "Adversarial Autoencoders". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2015. URL: https://proceedings.neurips.cc/paper/2015/file/8d55a249e6baa5c06772297520da2051-Paper.pdf.

[17] Edward M. et al. Marcotte. "A kinase chemogenomic set enables new biology and drug discovery". In: *Nature* 613 (2023), pp. 647–654. URL: https://www.nature.com/articles/s41586-022-05575-3.

[18] Mehrsa Mardikoraem et al. "Generative models for protein sequence modeling: recent advances and future directions". In: *Briefings in Bioinformatics* (2023).

[19] MIT Kinase Library. *About the Kinase Library – MIT*. https://kinase-library.mit.edu/about. 2024.

[20] William R Pearson. "An Introduction to Sequence Similarity ("Homology") Searching". In: *Current Protocols in Bioinformatics* (2013).

[21] PhosphoSitePlus. *Kinase Library – PhosphoSitePlus*. https://kinase-library.phosphosite.org/kinase-library/about. 2024.

[22] Emre Sevgen et al. "ProT-VAE: Protein Transformer Variational AutoEncoder for Functional Protein Design". In: *bioRxiv* (2023).

[23] Aditya Shah and Sungroh Yoon. "A survey of deep learning techniques for protein sequence classification". In: *arXiv preprint* (2022). URL: https://arxiv.org/pdf/2207.06678.

[24] Ankita Shreya. *Protein Sequence Classification*. https://medium.com/@ankita9shreya/protein-sequence-classification-c83a6cb38548. 2022.

[25] Alexey Strokach and Philip M. Kim. "Deep generative modeling for protein design". In: *Current Opinion in Structural Biology* (2022).

[26] Wikipedia contributors. *Autoencoder*. https://en.wikipedia.org/wiki/Autoencoder. 2024.

[27] Wikipedia contributors. *Multilayer perceptron*. https://en.wikipedia.org/wiki/Multilayer_perceptron. 2024.

[28] Wikipedia contributors. *Variational autoencoder*. https://en.wikipedia.org/wiki/Variational_autoencoder. 2024.