Final Research Report

Course Code: BIOX7011  
Student Name: Yu Sun  
Student Number: 45105764  
Supervisor: [Jan Engelstaedter, Affiliation]  
Date: [Submission Date]

# Abstract

This section provides a concise summary of the entire project (200–300 words). It should clearly outline the research aims, the main methodologies used, the key findings, and the major conclusions derived from the study.

# 1. Introduction

Since the start of 21st century Antimicrobial Resistance (AMR) has become one of the most pressing threats to global public health. The World Health Organization estimates more than 10 million people will die annually by 20250 from AMR-related infections [1].AMR not only makes it more difficult to treat infections, but also significantly raises healthcare costs and length of hospitalization, placing greater pressure on public health systems in low-income countries [2]. Among the many drug-resistant pathogens, drug-resistant tuberculosis (DR-TB) is of particular concern. approximately 450,000 people worldwide will have rifampicin-resistant tuberculosis (RR-TB) in 2022, with the majority of cases also showing resistance to isoniazid, thus constituting multidrug-resistant tuberculosis (MDR-TB) [3].

# Although Mycobacterium tuberculosis is the most intensively studied model for rifampicin resistance, rifampicin is exposed to a wide range of bacteria, both in clinical use and through environmental contamination. Therefore, this study aims to predict cross-species mutation patterns, which may provide insights into the evolutionary convergence of antibiotic resistance.

As humans truly explore the Earth, they encounter more than just E. coli or M. tb. Therefore, if humans rely on a limited set of RIF drugs to combat a wide variety of bacteria in nature, they are likely to exhibit varying adaptability or resistance. Therefore, this project aims to predict as many potential RIF-resistant mutations as possible in unconventional or less widely studied bacterial species.

# Materials and Methods

Materials：

1

你如何系统性地检索文献；

使用了哪款软件（ASReview）、版本号、核心功能；

使用了哪些关键词；

主动学习的流程（机器模型与人工复核如何结合）；

输出结果数量。

2

你具体使用了哪些语言模型（Sentence-BERT、SciBERT）；

微调方式（HuggingFace Transformers，PyTorch 环境）；

性能评估方法（5-fold CV + Precision/Recall/F1）；

软件库。

3

数据整合步骤（DOI 匹配、去重、跨物种突变映射）；

提及使用的辅助工具（Excel、Python、Biopython、Clustal Omega 等）；

输出文件（newrifmutdata.xlsx）。

4

输入数据矩阵（X\_dense\_high / X\_dense\_midhigh）；

使用的降维算法与聚类算法；

软件与库；

筛选 top methods 的依据（平均轮廓系数）。

5

说明：

目的（验证模型能否学习已知突变规律）；

方法（PU-learning, Random Forest）；

软件与实现（Scikit-learn）；

评估策略（Mask–then–Recover + Recall@K）。

Methods:

2. Methods

2.1 Data Sources and Literature Screening

2.1.1 Literature Search and Active Learning (ASReview)

To systematically collect literature reporting on rifampicin (RIF) resistance mutations, this study first tested multiple keyword combinations on the Web of Science website, including "RIF," "resistance," and "mutant screen," to search for as many articles as possible containing experimentally generated RIF resistance mutation data.

The search query was: ((rifampicin OR rifampin) AND (resistance OR resistant) AND (mutation OR polymorphism OR variant)).

Then, at the recommendation of my supervisor, I used the open-source active learning platform ASReview (version 1.3.1) for semi-automated screening.

After initially importing the literature, ASReview continuously updated the ranking using an active learning algorithm to maximize the model's efficiency in identifying relevant literature. The researchers manually annotated the first 109 articles in the system, obtaining 46 relevant samples as the initial training set. This was then combined with three rounds of model disagreement sampling, ultimately yielding a total of 335 labeled articles.

2.2 Language Model Fine-tuning and Document Classification

To further automatically identify articles with potential drug-resistant mutations, two models based on deep language representation were used: Sentence-BERT (all-MiniLM-L6-v2) and SciBERT (allenai/scibert\_scivocab\_uncased).

Model fine-tuning was performed in Python 3.10, relying on HuggingFace Transformers (v4.30) and PyTorch (v2.0).

Stratified 5-fold cross-validation was performed on each model on the 334 annotated samples, and the average precision, recall, and F1 score were calculated. The SciBERT model performed best (Precision = 0.41 ± 0.07, Recall = 0.83 ± 0.07, F1 = 0.54 ± 0.04) and was therefore used to predict the remaining unannotated articles.

2.3 Integration and Standardization of New Mutation Data

After comparing the candidate articles predicted by the model with the existing mutation database, a total of 37 duplicate articles, 170 newly added articles, and 11 articles without DOI records were identified.

For newly added articles, the mutation site, amino acid substitution pattern, and species of origin were extracted.

If the mutation originated from a species other than E. coli, Clustal Omega (v1.2.4) was used for sequence alignment, and the mutation position was mapped to the corresponding site in E. coli.

The resulting standardized mutation table (latestnewdata.xlsx) includes a uniformly formatted amino acid position (AA\_pos), mutation pattern (AA\_change), species name, and reference number.

2.4 Unsupervised ML

To identify the clustering structure of mutation spectra across species, the authors first constructed a binary mutation matrix X\_dense (rows = species, columns = mutations) using the Lab mutant dataset generated in the previous step.****To make the results more readable and intuitive, the authors filtered the species × mutation matrix (X\_dense) for species with low confounder scores (i.e., <0.3).****

Besids,after multiple attempts at plotting, the authors removed data from several species because their mutation data had too low overlap with other species. This was likely due to a misalignment in the coordinate system constructed using E. coli as the standard, or other issues beyond the scope of this project, leading to manual filtering.

UMAP (uwot v0.1.15) was used for nonlinear dimensionality reduction (n\_neighbors=15, min\_dist=0.3, seed=123). Four clustering algorithms were then applied to the embedding space: HDBSCAN (dbscan v1.1.11), DBSCAN, k-means, and GMM (mclust v6.0).

Three distance metrics (Euclidean, Manhattan, and Cosine) were combined to generate 12 schemes. The average silhouette score is calculated for each combination. The top three methods are selected based on the scores: Cosine–HDBSCAN, Euclidean–GMM, and Euclidean–HDBSCAN. The clustering results are visualized on their UMAP embeddings.

2.5 Supervised ML

Using the clustering results and confounder score from unsupervised clustering, the authors narrowed the species and mutations used for training and prediction. They then trained a supervised learning model to test its ability to learn the distribution patterns of drug-resistant mutations.

Using the Positive–Unlabeled Learning (PU-learning) framework, a binary classification model based on Random Forest (scikit-learn v1.3) was trained using known drug-resistant mutations as positive examples and unlabeled mutations as background samples. Furthermore, since the ALJE team had already collected a considerable number of "non-laboratory" mutations, the authors excluded these mutations from their predictions to ensure the "novelty" of the final results. A "novel mutation" is defined as one that does not appear in our compiled set of previously observed non-lab mutations after mapping all records to a unified E. coli rpoB amino-acid coordinate system.

To evaluate the generalization ability of the model, a "mask-then-recover" strategy was employed: some positive mutations were randomly masked and then observed to see if the model could recover these mutations in the top K predictions.

Top-K Strategy: Select the top K mutations with the highest predicted probability for each species. Threshold Strategy: Select mutations with p\_true ≥ τ(0.7). Generate a list of candidate mutations for subsequent analysis and experimental verification.

Model performance was measured using Recall@K, Precision, and ROC-AUC metrics, and was analyzed using the ROI-based validation dataset. The results were averaged from 100 repeated experiments.

2.6 Visualization and statistical analysis

所有可视化分析均在 R (version 4.3.1) 环境中完成，主要使用 ggplot2 (v3.4.4)、ComplexHeatmap (v2.16) 与 UpSetR (v1.4.0)。

热图用于展示突变分布与聚类一致性；

UpSet 图用于展示突变交集结构；

统计检验包括 Fisher 精确检验与 Benjamini–Hochberg FDR 校正。

# Results

****Clustering results are systematically compared all combinations of three distance metrics (Euclidean, Manhattan, and Cosine) and four clustering algorithms (HDBSCAN, k-means, DBSCAN, and GMM). Each combination was first embedded in two dimensions using UMAP with uniform parameters (n\_neighbors=15, min\_dist=0.3, and seed=123). Clustering was then performed on the embedded space (minPts and eps were used for HDBSCAN/DBSCAN, k=4 for k-means, and optimality was automatically chosen for GMM). To mitigate noise, the average silhouette coefficient was calculated only on valid samples (with positive cluster labels and ≥2 clusters) as a quality metric. The top\_k methods were selected from the highest to lowest silhouette coefficients. If restrict\_metric is set, the optimization is performed within the specified metric; otherwise, the optimization is performed across all combinations. Scores and labels for all solutions were exported and archived to ensure reproducibility.****

****For the top-selected methods, we present their UMAP projections and corresponding cluster heatmaps (all mutations and Top-30 versions) to compare the consistency and differences in clustering achieved by different algorithms in the mutation spectrum space.****

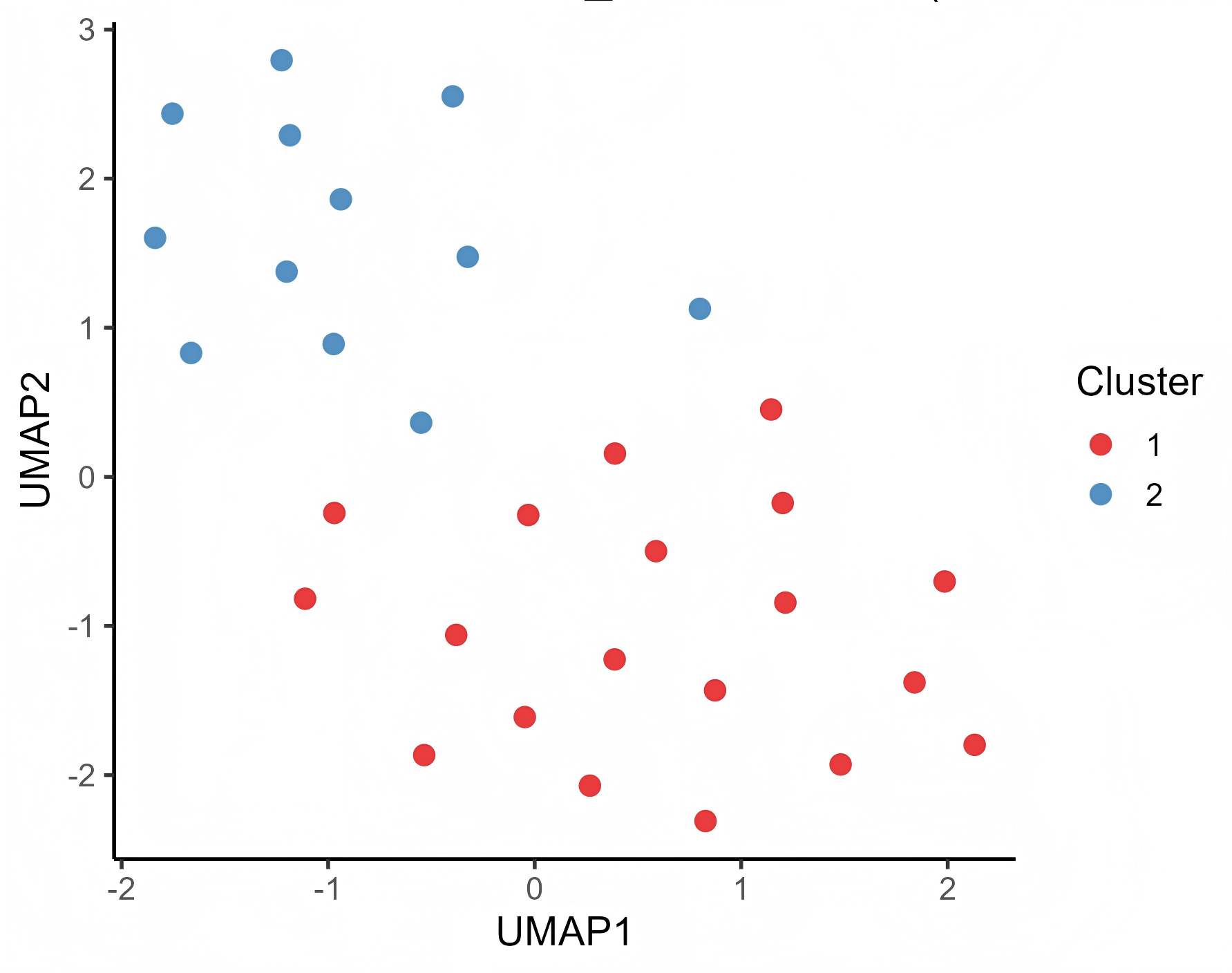
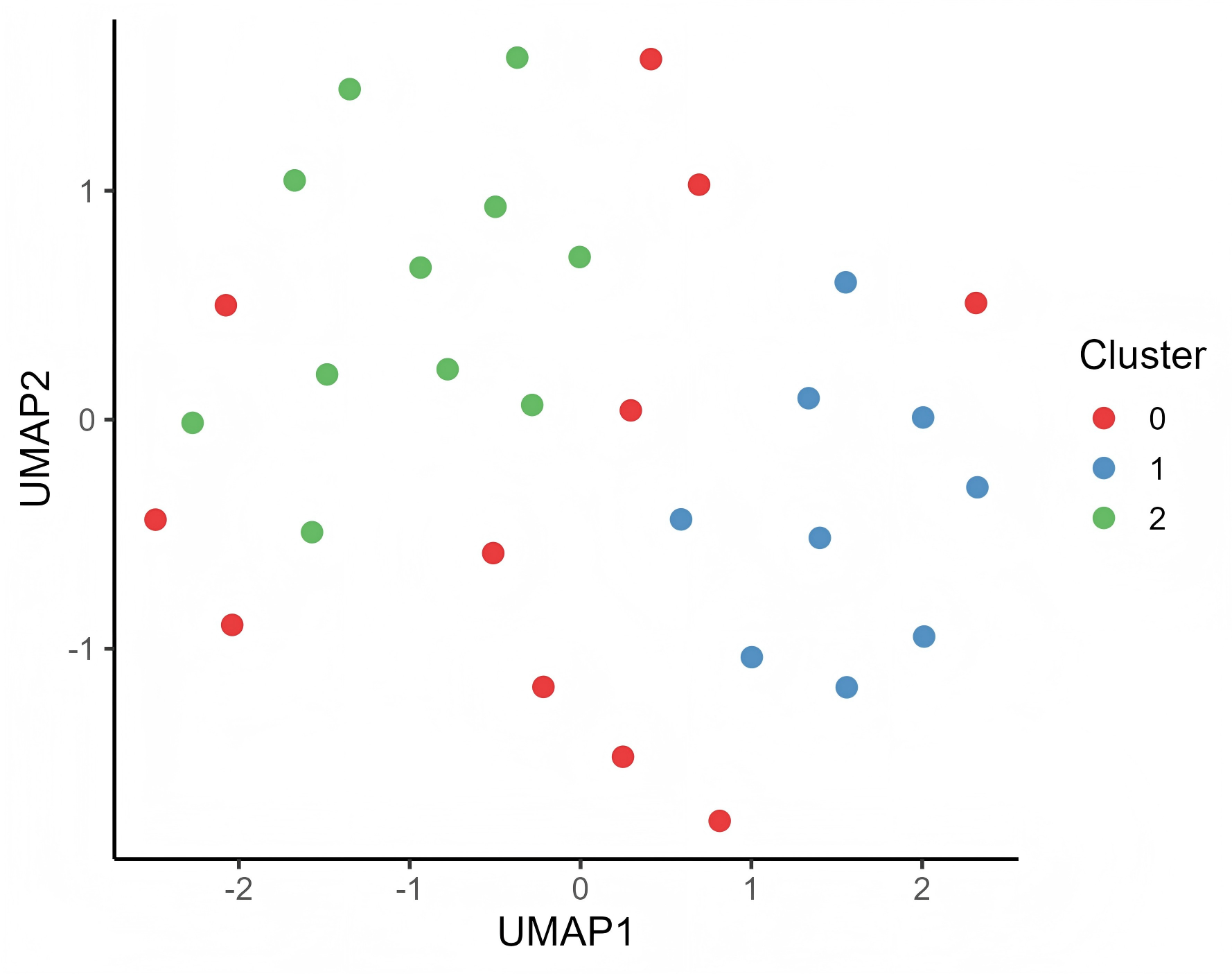
****Based on the parameters of the X\_dense\_midhigh matrix, this study performed multi-algorithm clustering and visualization analysis of mutation distribution patterns across species. To verify the robustness and consistency of the clustering results, we used three top-performing clustering methods: Cosine–HDBSCAN, Euclidean–GMM, and Euclidean–HDBSCAN.****

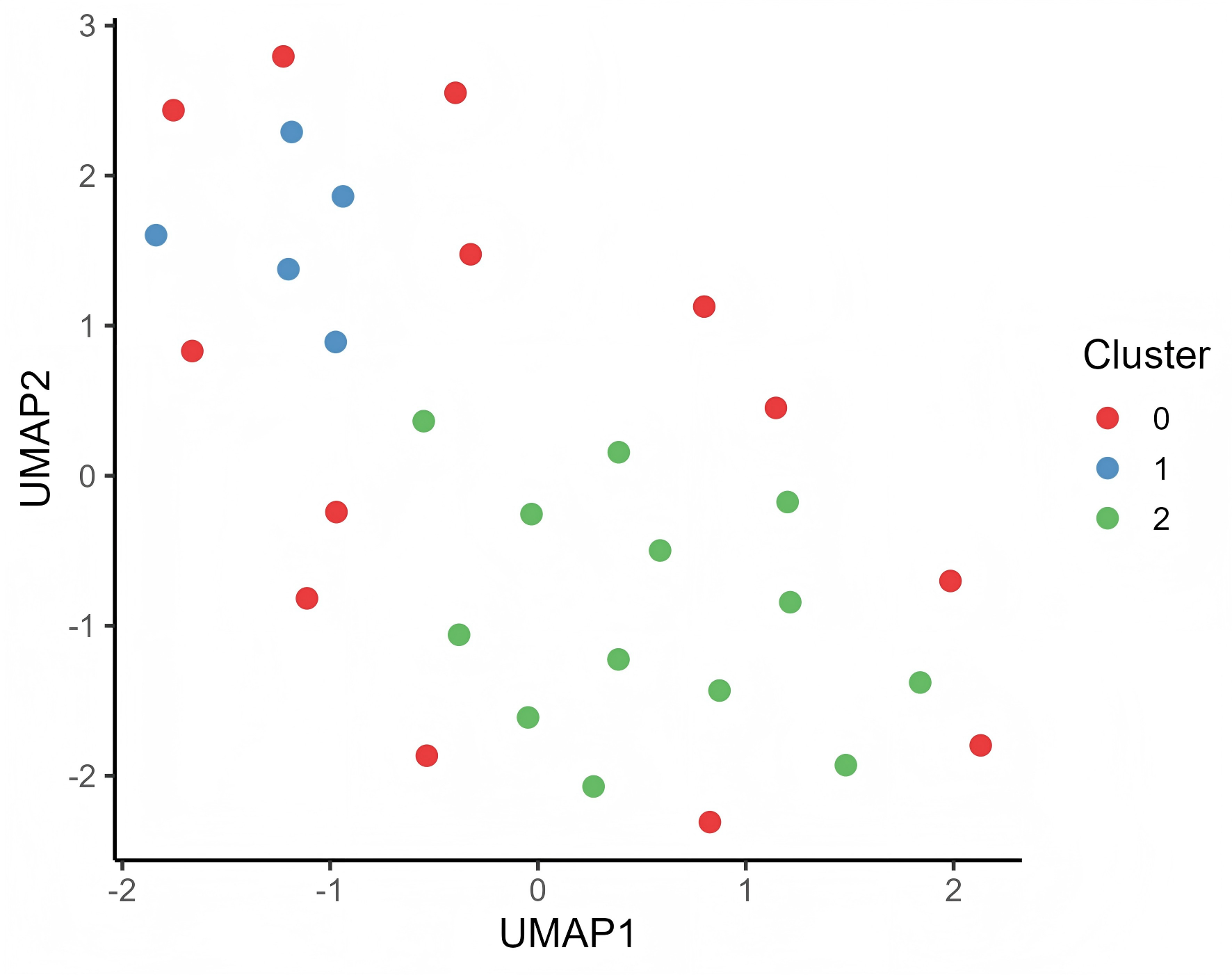
****The HDBSCAN algorithm automatically identifies clusters and excludes noise points in non-spherical data, while the GMM (Gaussian Mixture Model) captures the underlying continuous distribution of the data through probability density modeling. All three methods take a standardized mutation presence matrix as input. The dimensionality reduction and visualization uses the UMAP (Uniform Manifold Approximation and Projection) algorithm to preserve local topological relationships between samples. The UMAP scatter plot ( ) shows the distribution and cluster boundaries of each species in the two-dimensional latent space using the three methods. Different colors represent different cluster labels, which can intuitively reflect the similarity of mutational profiles between species. For example, Cosine–HDBSCAN effectively groups species with highly similar mutational profiles into the same cluster; Euclidean–GMM shows a relatively regular distribution with clear boundaries; and Euclidean–HDBSCAN demonstrates higher resolution when dealing with intermediate or noisy species. Overall, the cluster structures obtained by the three methods are highly consistent, indicating that mutational patterns are stable and reproducible across different metric spaces.****

****The heatmap ( ) further illustrates the shared mutations among species within each cluster. Rows represent species, columns represent mutation sites, and colors indicate the presence or absence of mutations (1/0). The color bars on the sides correspond to the cluster numbers in the UMAP clustering results. It can be seen that different clusters exhibit distinct complementarity or specificity in mutation distribution. For example, Cluster 0 is primarily concentrated in RRDR core mutations (such as D516, H526, and S531), while Cluster 2 is enriched in marginal or low-frequency mutations (such as Q148R and L533R). These clustering characteristics indicate that the mutation spectrum has certain phylogenetic and evolutionary differentiation characteristics.****

Clustering Framework Overview

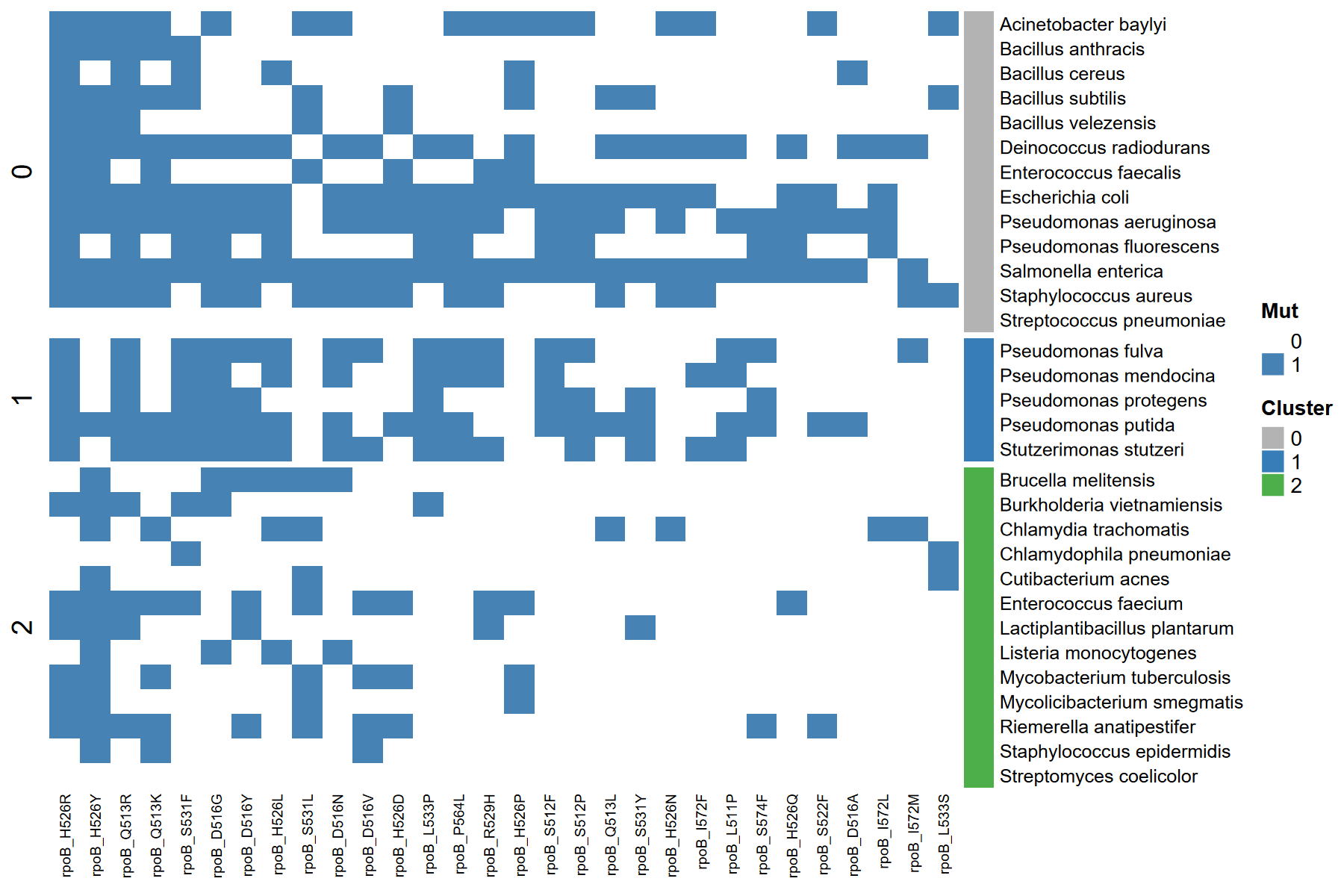
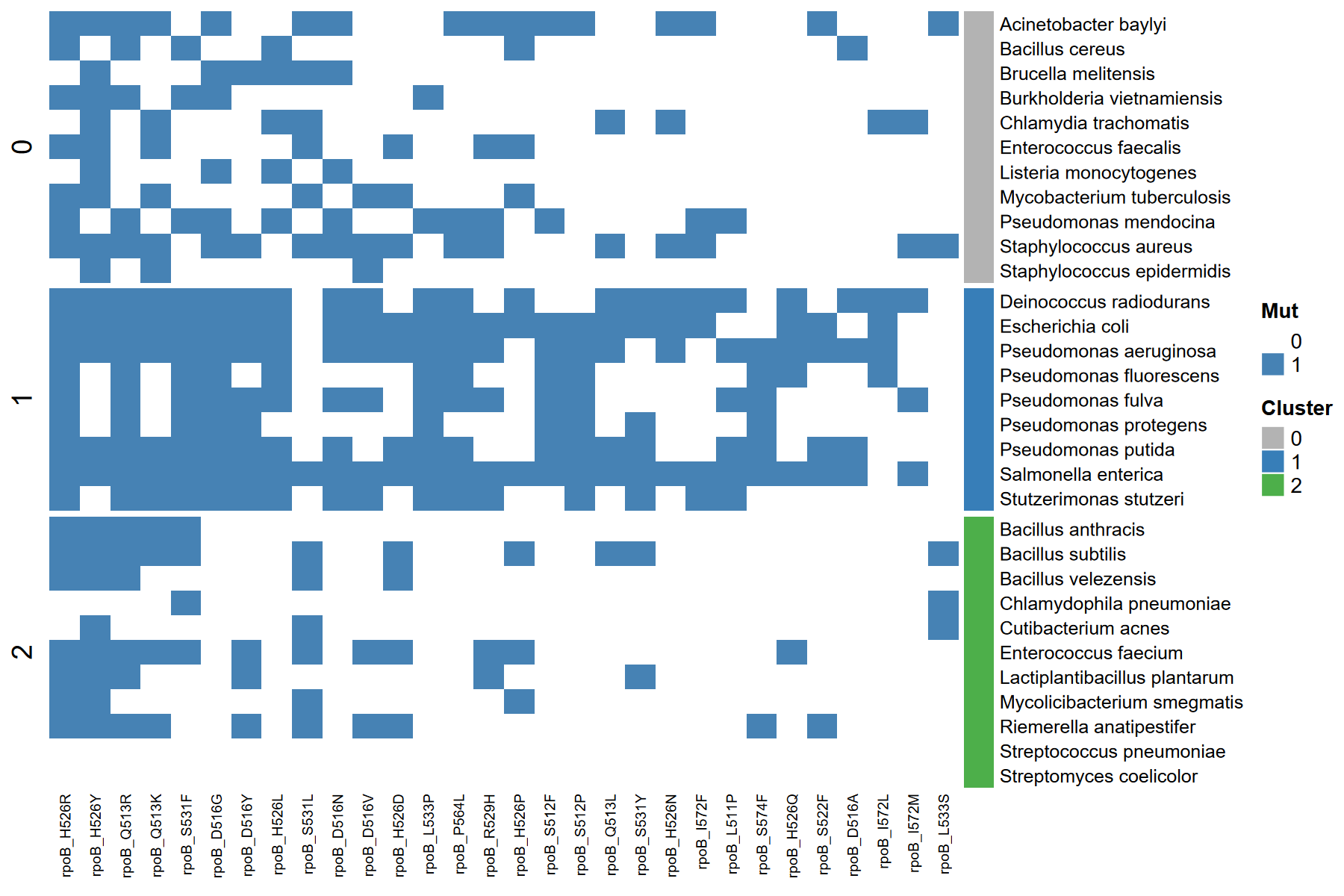
Based on the X\_dense\_midhigh matrix, the authors used three top-performing clustering methods: Cosine–HDBSCAN, Euclidean–GMM, and Euclidean–HDBSCAN. All three methods take a normalized mutation presence matrix as input and employ the Unified Mapping (UMAP) algorithm for dimensionality reduction and visualization, preserving local topological relationships between samples.

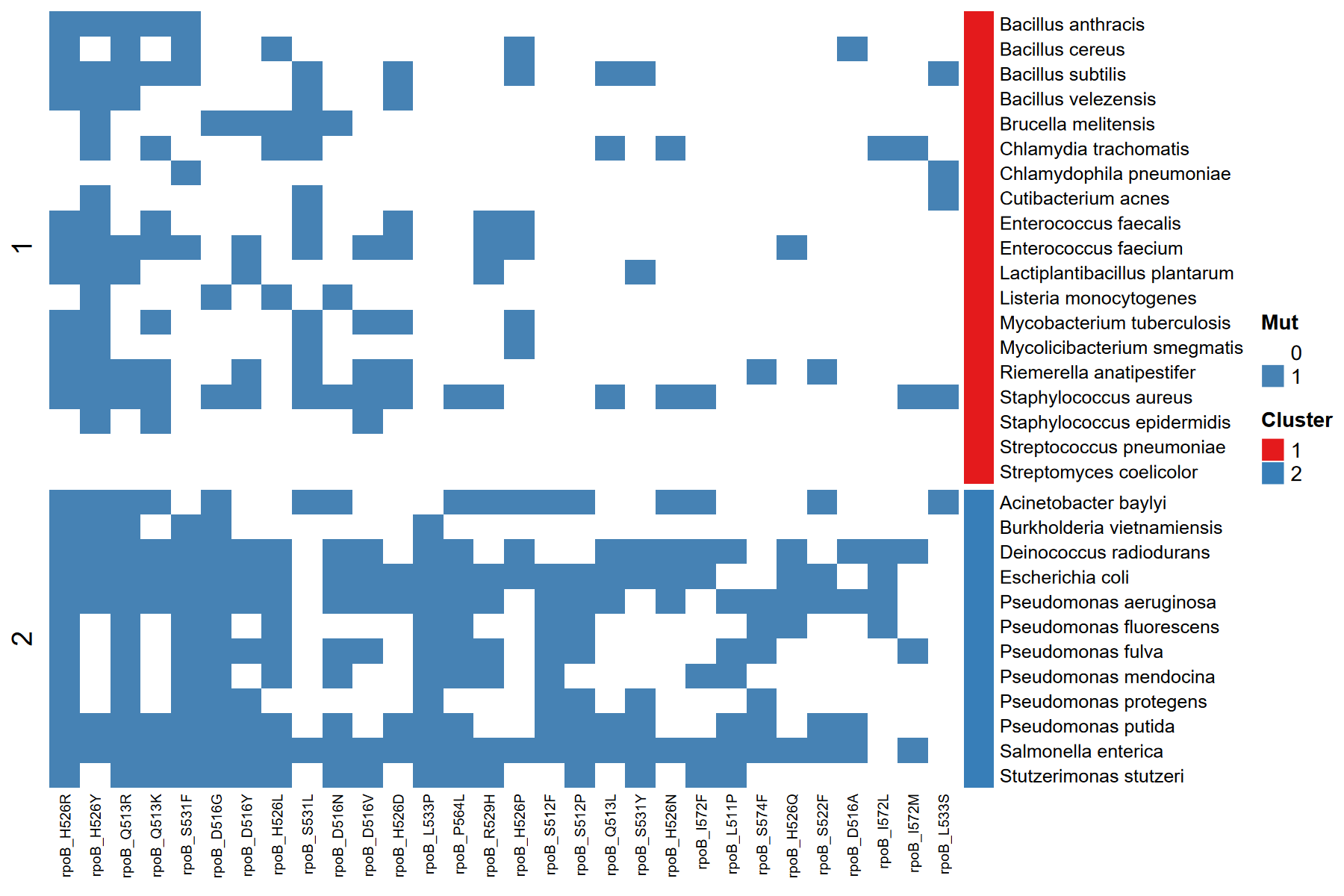
The UMAP scatter plots show the distribution and cluster boundaries of each species in the two-dimensional latent space using the three methods, with different colors representing different cluster labels.  




**Mutation Distribution and Intra-Cluster Characteristics**

The heatmap further illustrates the shared mutations among species within each cluster. Rows represent species, columns represent mutation sites, and colors indicate the presence or absence of mutations.

The color bars on the sides correspond to the cluster numbers in the UMAP clustering results. Different clusters exhibit distinct complementarities or specificities in their mutation distributions.  




**Analysis of the best clustering methods reveals that the best clustering method for mid-high-level data is cosine–HDBSCAN, followed by euclidean–GMM.**

**Both algorithms tend to capture density differences and multimodal distributions.**

**This "density difference" can be seen in the changes in the UpSet graph:**

**Top 10 species → one or two density peaks (highly shared mutations);**

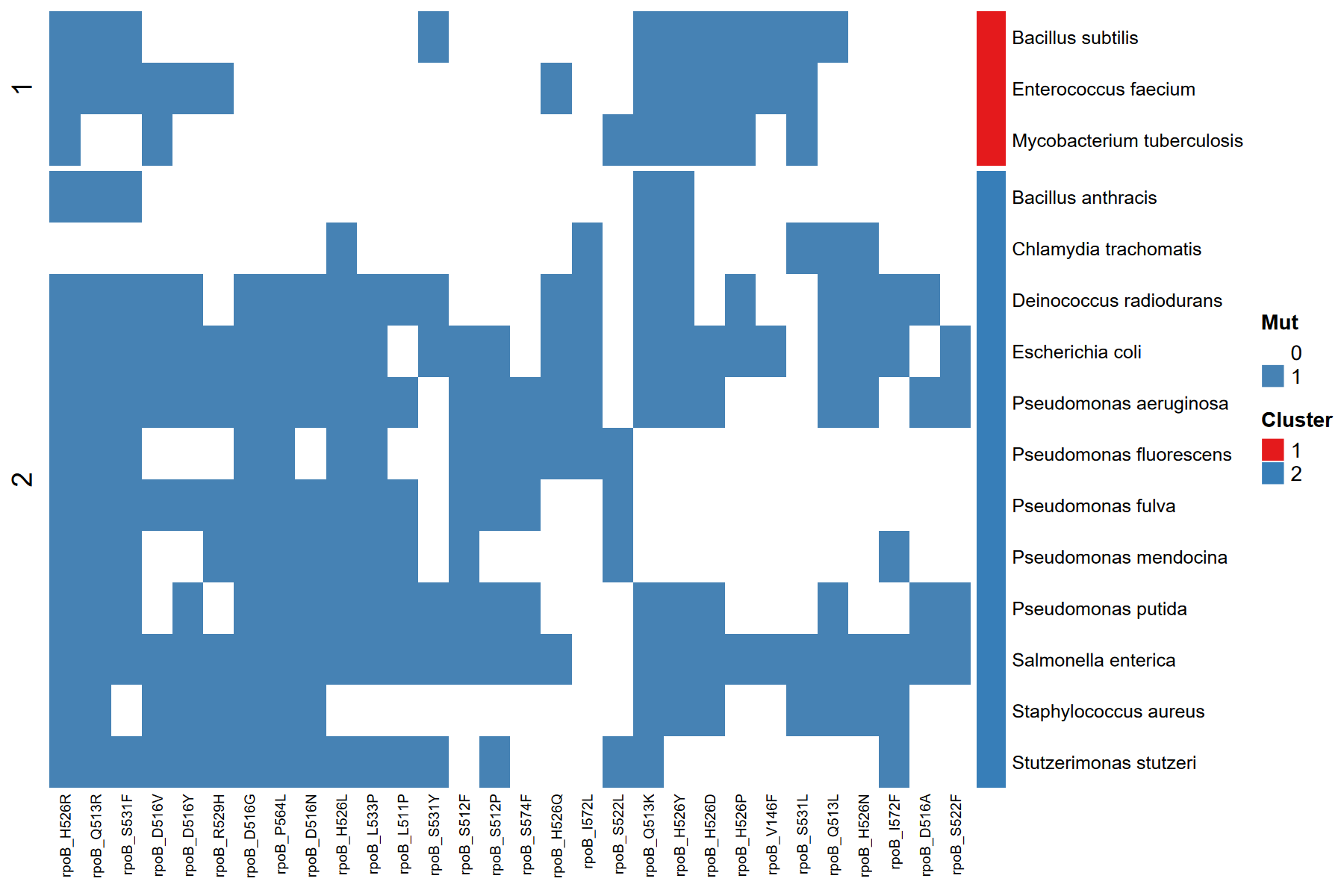
**Mid-high → multiple flat peaks (local commonality but overall sparseness).**

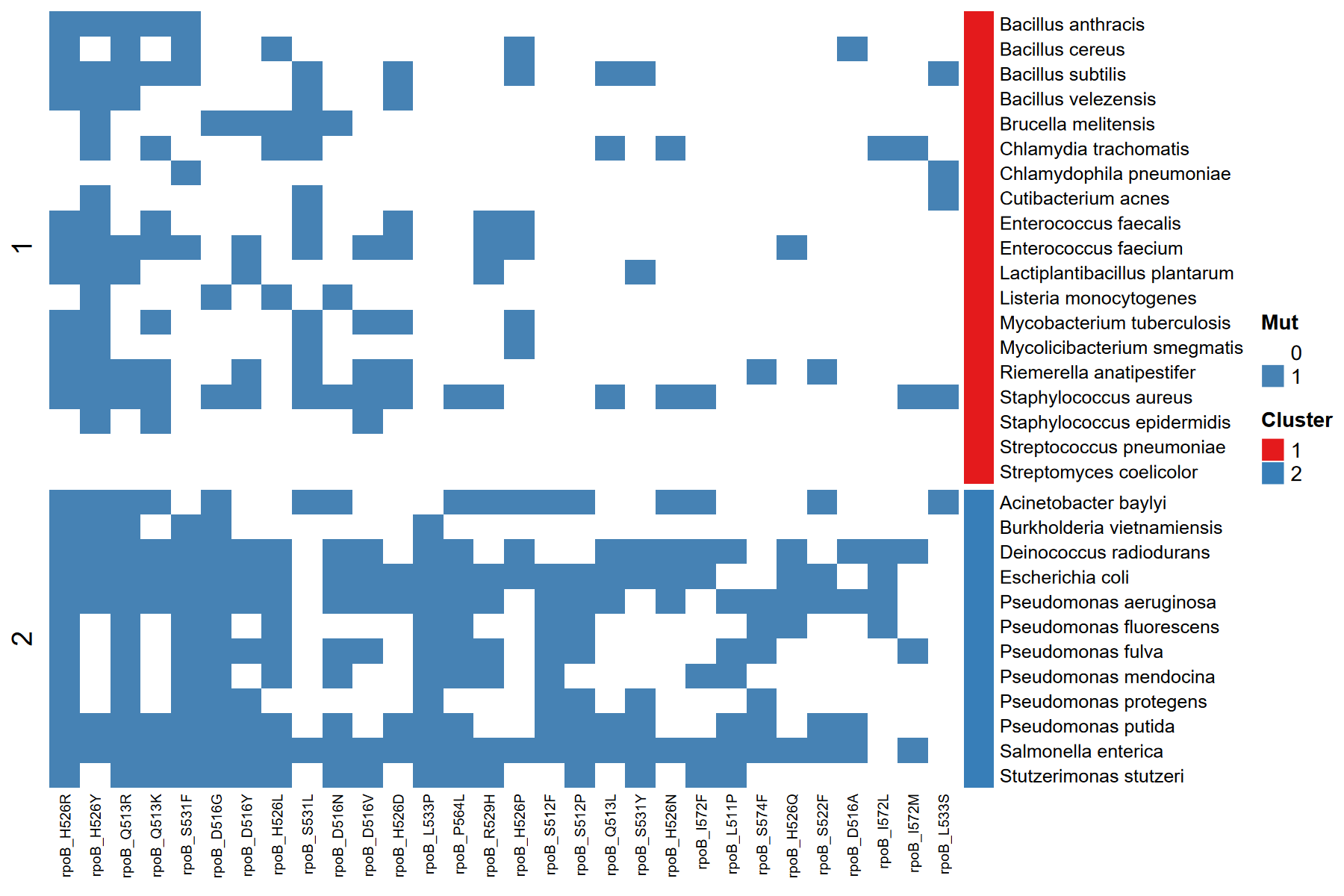
**This suggests that the broadened species dimension of the UpSet graph already reflects the tendency of your cluster structure to become sparser at the mutation level.**

****Comparison of Clustering Patterns at Different Confounder Levels****

**To ensure that clustering results are not significantly biased towards highly studied species versus less studied ones, the authors specifically compared clustering results for data with high confounder and mid-high confounder levels.**

**Despite varying noise levels, the clustering structures obtained by the three methods are generally consistent, demonstrating that mutation patterns are robust and reproducible across different metric spaces.**

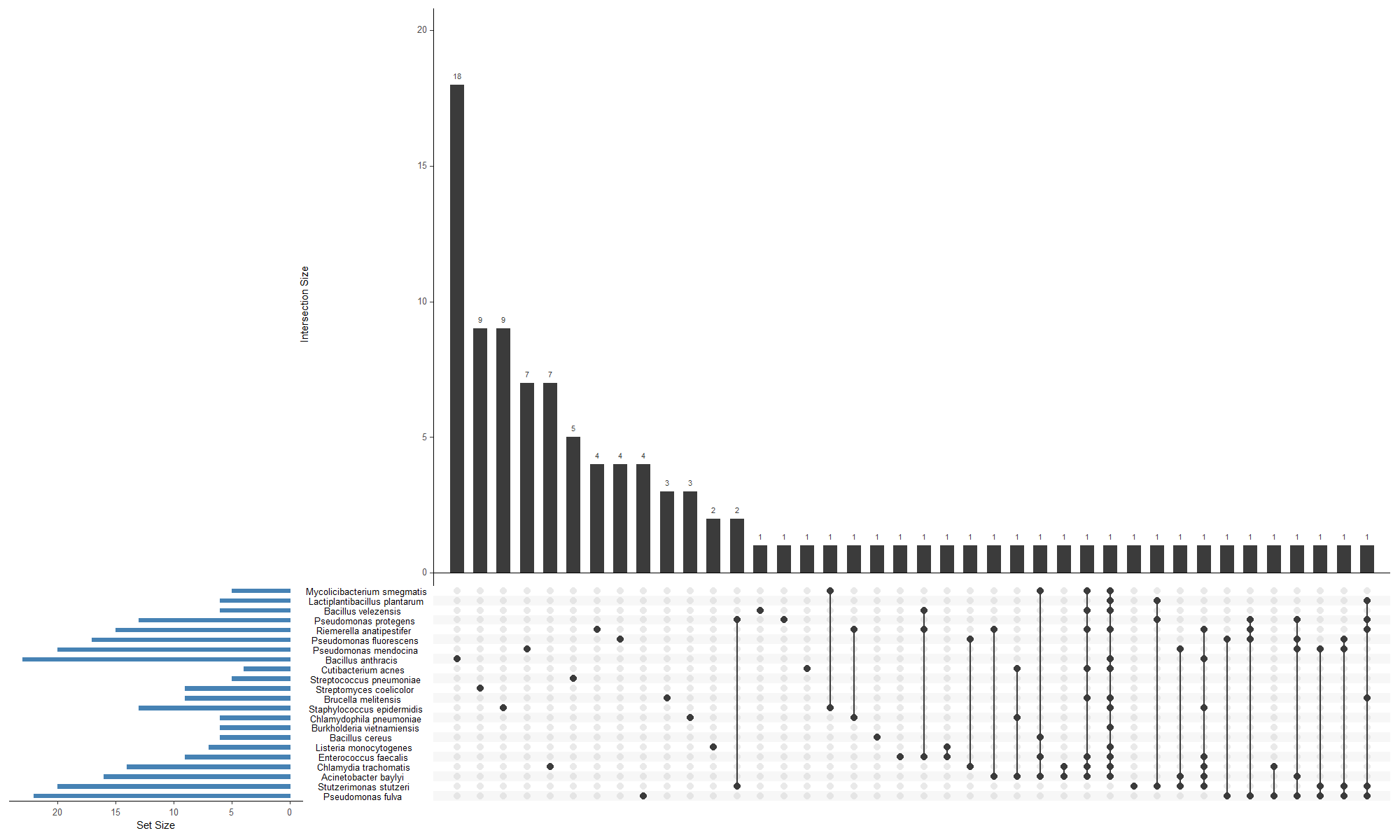
****

****

****Mutation Intersection Relationships (UpSet Analysis)****

**UpSet plots demonstrate the shared mutation relationships between species. Species are set elements, and the bar graphs represent the size of the intersections.**

**In the mid-high dataset, some bacterial genera (e.g., Pseudomonas and Bacillus) share multiple high-frequency rpoB mutation sites, suggesting convergence in resistance mechanisms across lineages.**

********

The blue horizontal bar on the left (Set Size) represents the total number of mutations in each species (set), that is, the number of mutations possessed by that species.

The longer the blue bar, the greater the total number of mutations in that species.

The dot matrix at the bottom (Sets) represents the intersection of different species combinations.

Each vertical column corresponds to a species combination; a black dot indicates that the species is included in the set, and a gray dot indicates that it is not.

A line connecting multiple black dots in a column indicates that those species share the same set of mutations.

The black bar above (Intersection Size) represents the number of mutations shared by that species combination.

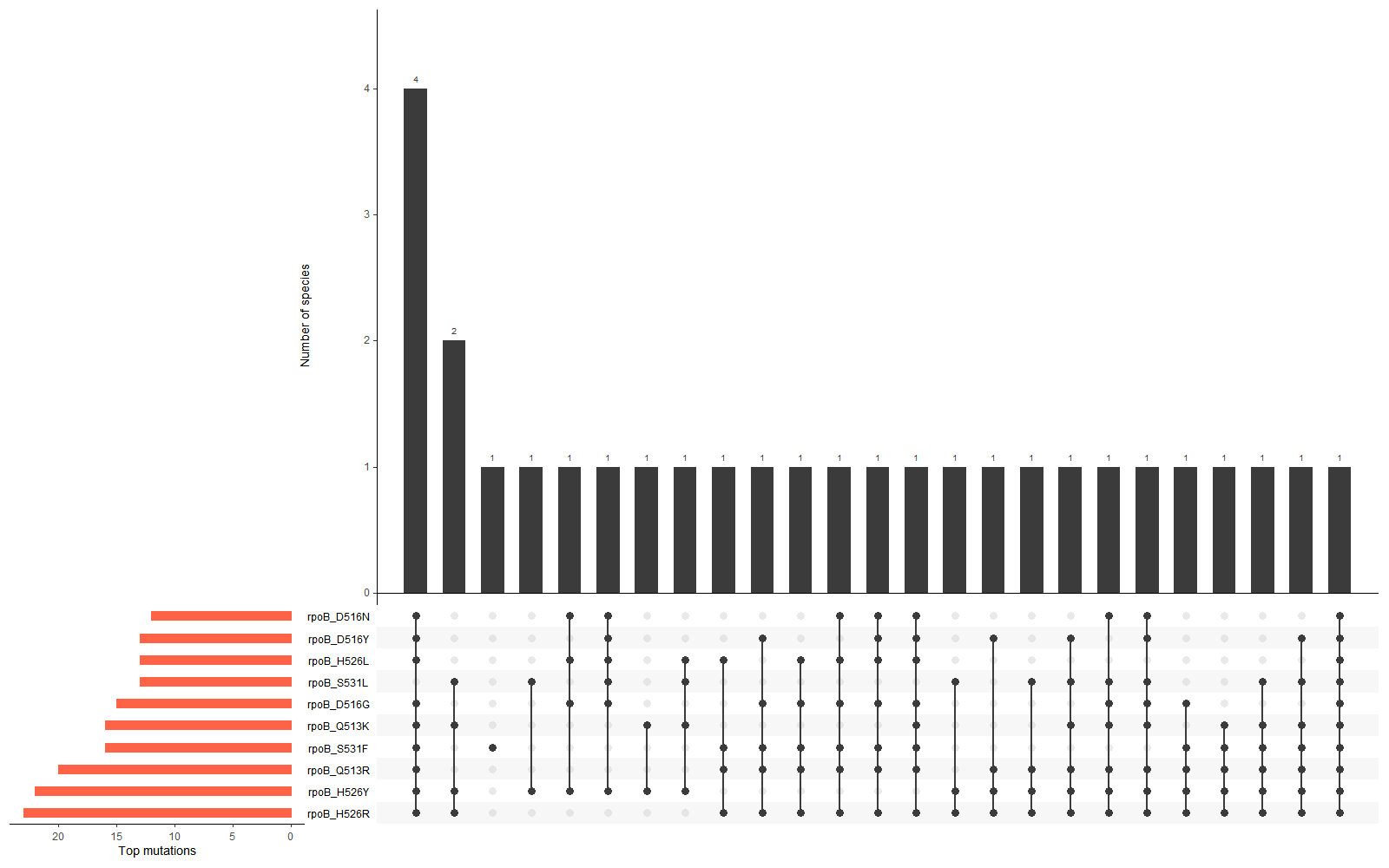
The taller the bar, the more mutations are shared among the species in that group.

Some mutations are shared by a very large number of species (for example, 20 species share certain mutations).

**Only a few mutations are widely distributed across multiple species, while most mutations occur in only a few species.**

The distribution of mutations across species is significantly uneven.

Certain core mutation combinations are common across multiple species, potentially reflecting conserved drug resistance mechanisms.

********

**The red horizontal bar on the left (Selected mutation frequency) indicates the number of species in which each mutation occurs. Longer red bars indicate more common mutations.**

**The bottom matrix (Sets): Each column represents a mutation combination (i.e., which mutations co-occur in the same species).**

**A black dot indicates that the mutation is included in the set, and a line connecting the dots indicates that these mutations co-occur.**

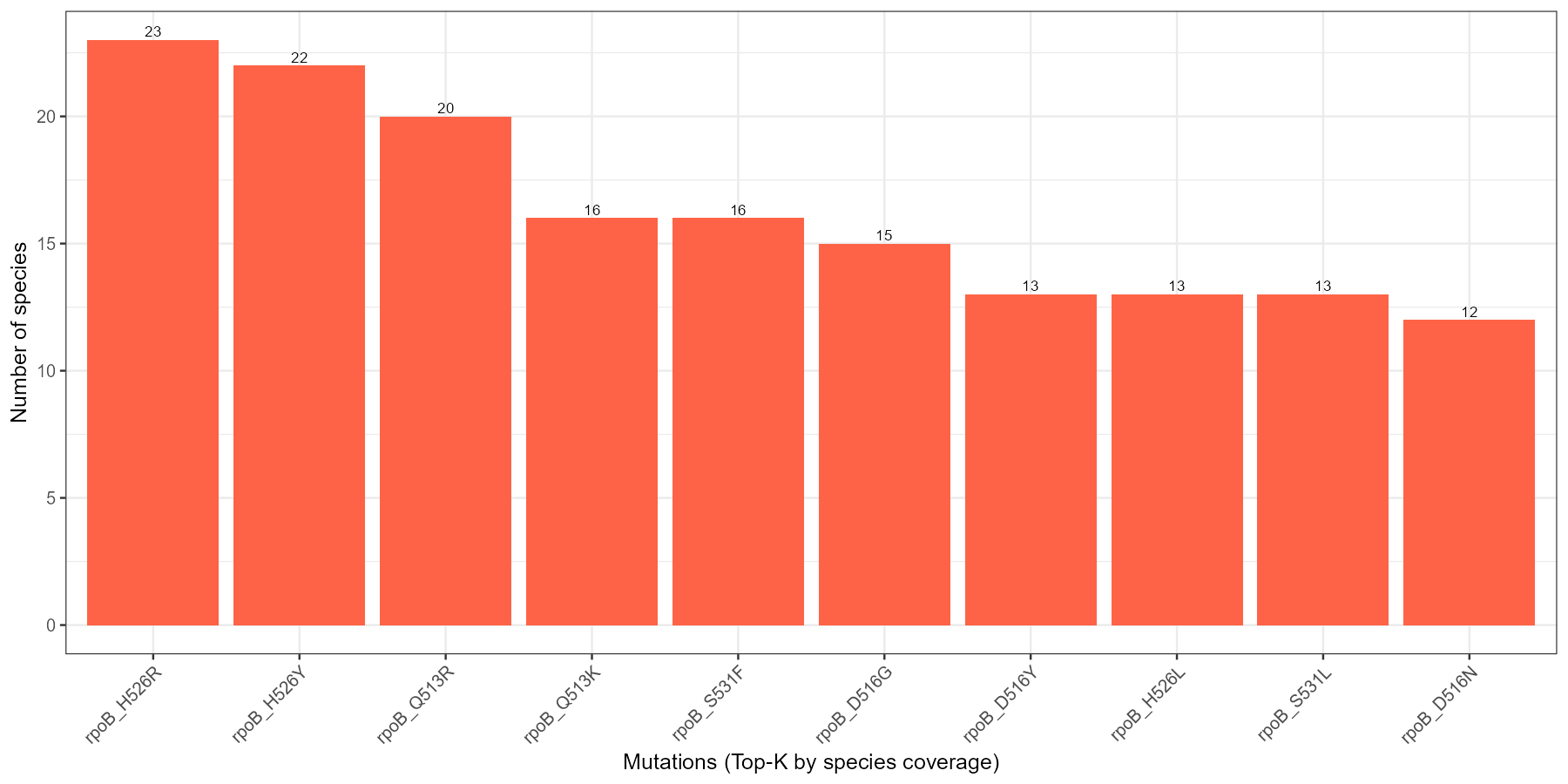
**The upper bars (Number of species): Show how many species share that set of mutations.**

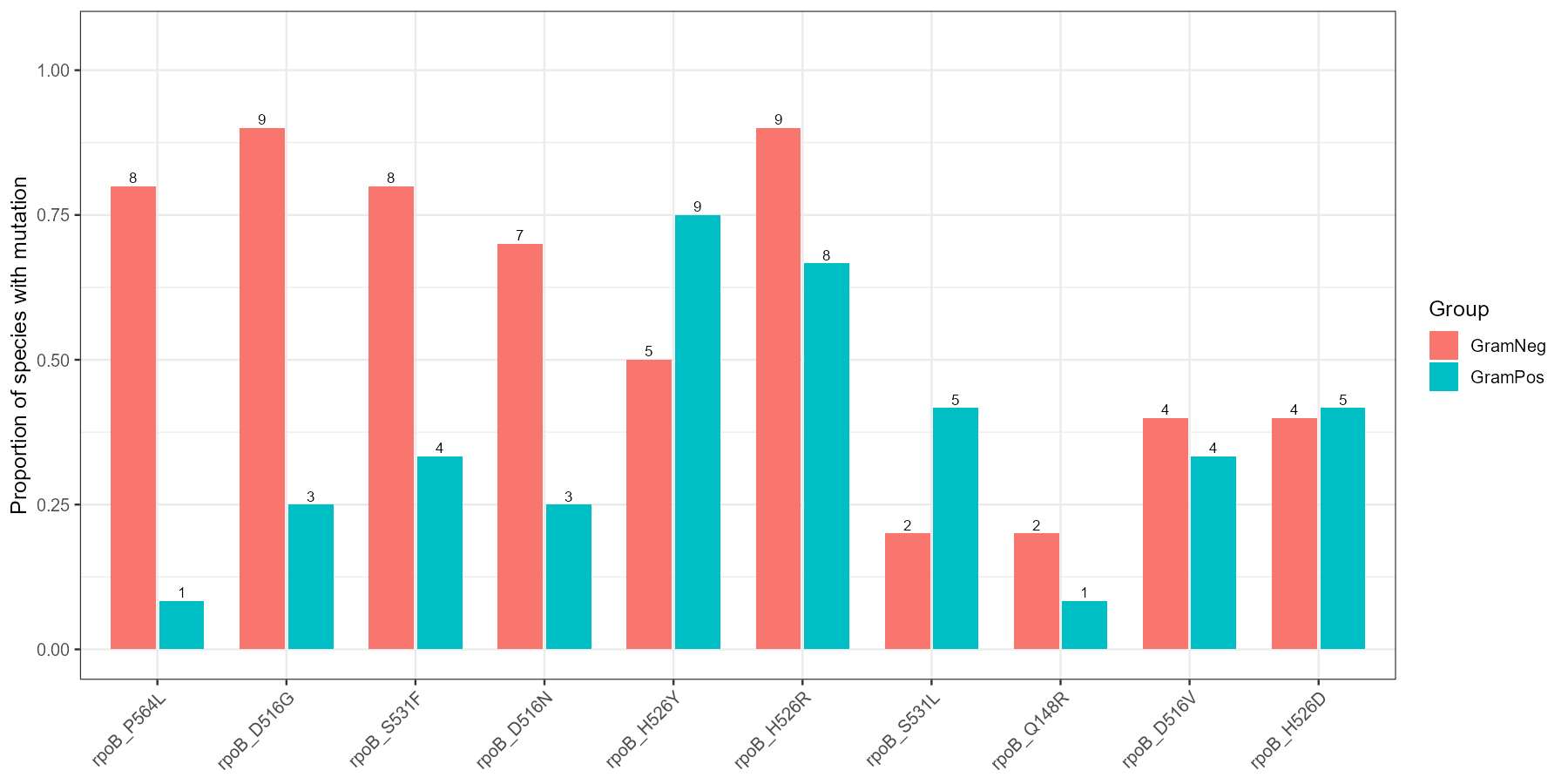
**The taller the bar, the higher the bar on the far left, is 4, indicating that four species share the mutation combination shown in that column (typically including the classic combination of S531F and H526Y).**

**The remaining bars are mostly 1 or 2 in height, indicating that most mutation combinations co-occur in only a very small number of species.**

**rpoB\_S531F and rpoB\_H526Y are the most frequently occurring mutations, each appearing in over 20 species. They are typical rifampicin resistance loci.**

****Core sites of resistance mutations are highly conserved, but multi-site co-occurrence patterns are species-specificGram-positive and -negative differentiation analysis.****

****

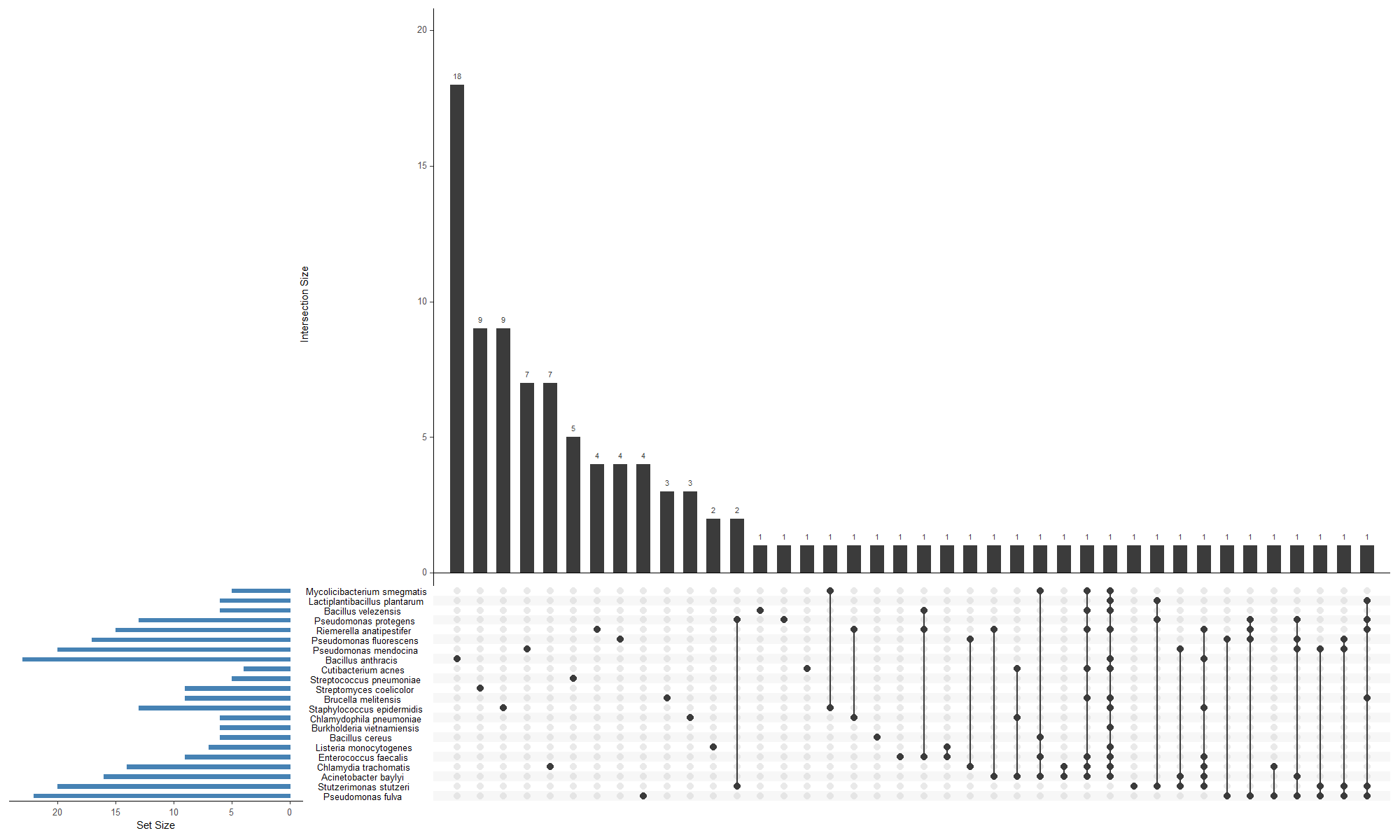
****

**Leveraging the highly recognized Gram-positive/-negative differentiation method, the authors further compared mutation patterns in the mid-high data by Gram staining group (Gram+/Gram–). They found that the two bacterial groups differed in their preferences at typical rpoB loci.**

Most classical RRDR mutations (e.g., P564L, D516G, S531F) were enriched in Gram-negative species, whereas a few peripheral variants (S531L, Q148R) showed relative enrichment in Gram-positive taxa.

This suggests that Gram-negative species rely predominantly on canonical RRDR substitutions conferring strong resistance, whereas Gram-positive taxa accumulate peripheral or compensatory variants that may fine-tune rifampicin susceptibility with reduced fitness costs.

**This grouping trend is consistent with the phylogenetic clustering reported by Bolourchi et al. (2025), indicating that mutational spectrum structure exhibits reproducible evolutionary clustering across species.**



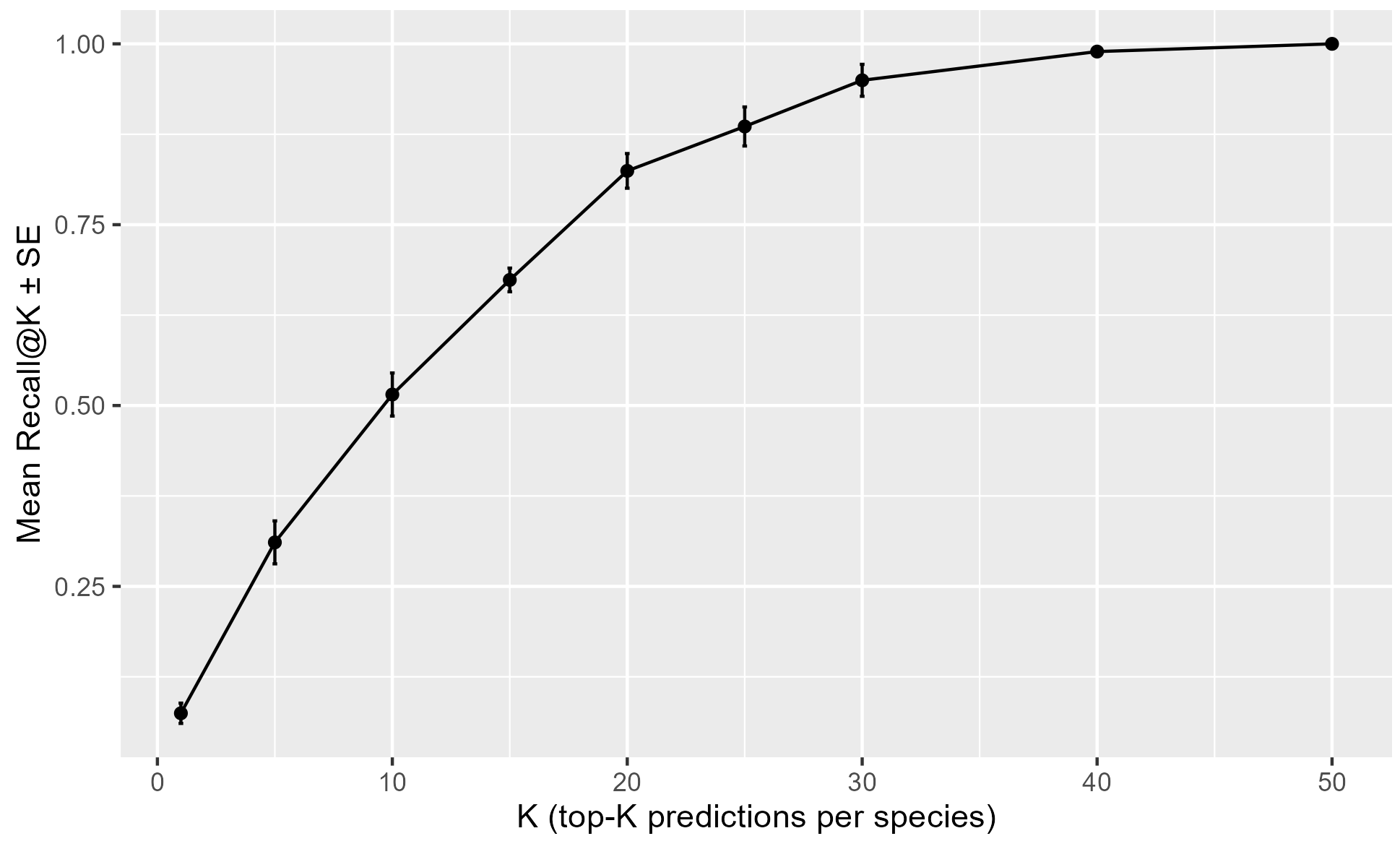
4. Supervised Learning Results

4.1 Overview of PU-learning framework

- Briefly explain model, data, and Mask–then–Recover design.

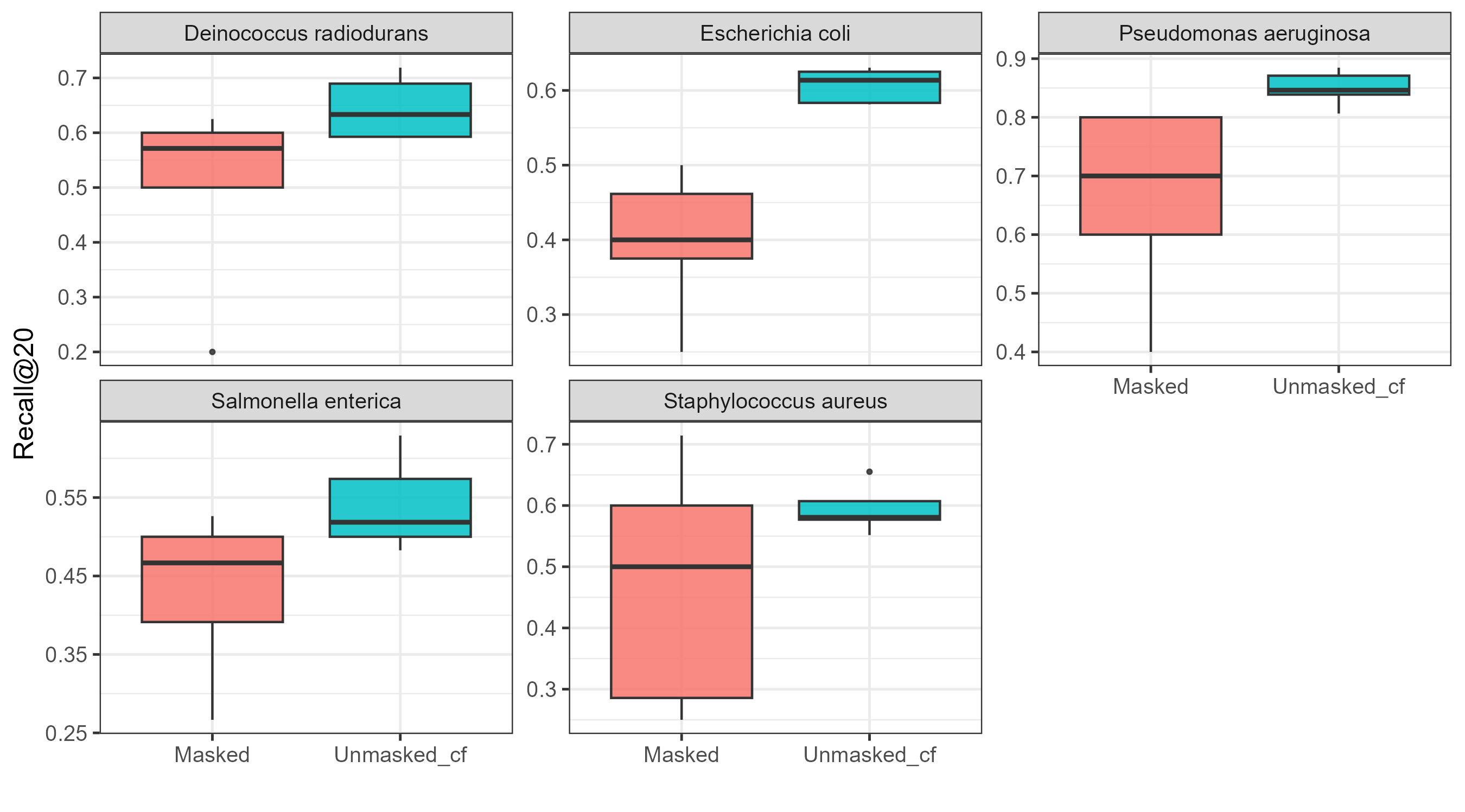
4.2 Model performance and recovery ability

- Recall@K curves (Fig. X)



|  |  |  |  |
| --- | --- | --- | --- |
| Species | Masked | Unmasked\_cf | Delta |
| Brucella abortus | 0.16666666666666666 | 1 | 0.8333333333333334 |
| Listeria monocytogenes | 0.1 | 0.6733333333333333 | 0.5733333333333334 |
| Brucella melitensis | 0.48 | 0.935 | 0.45500000000000007 |
| Helicobacter pylori ATCC | 0.43333333333333335 | 0.8742063492063492 | 0.44087301587301586 |
| Neisseria meningitidis | 0.5 | 0.9 | 0.4 |

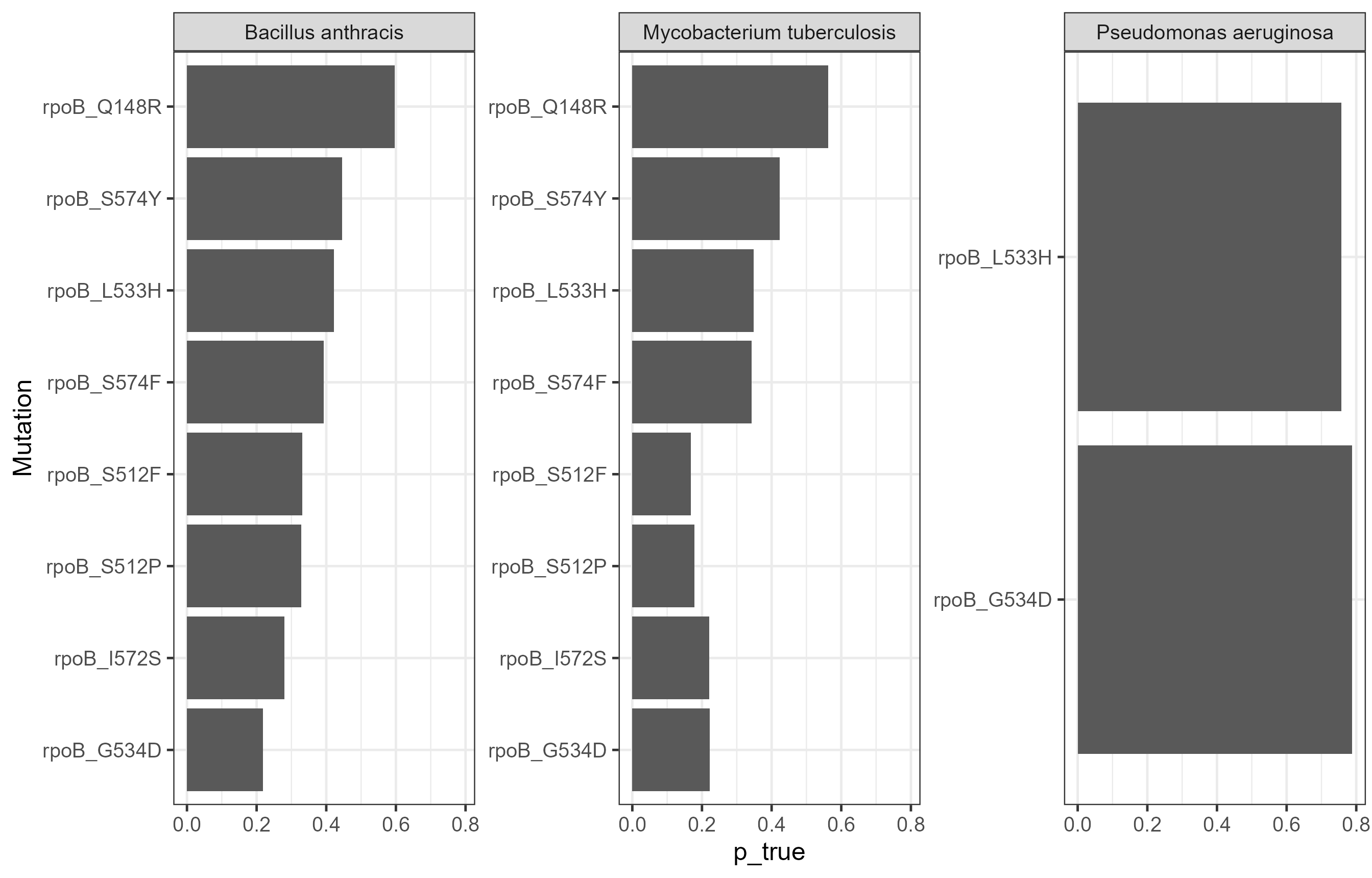
- Interpretation: model generalizes well to resistance-associated mutations.



4.3 Candidate prediction and novelty analysis

- Probability distribution and novelty filtering (Fig. Y)

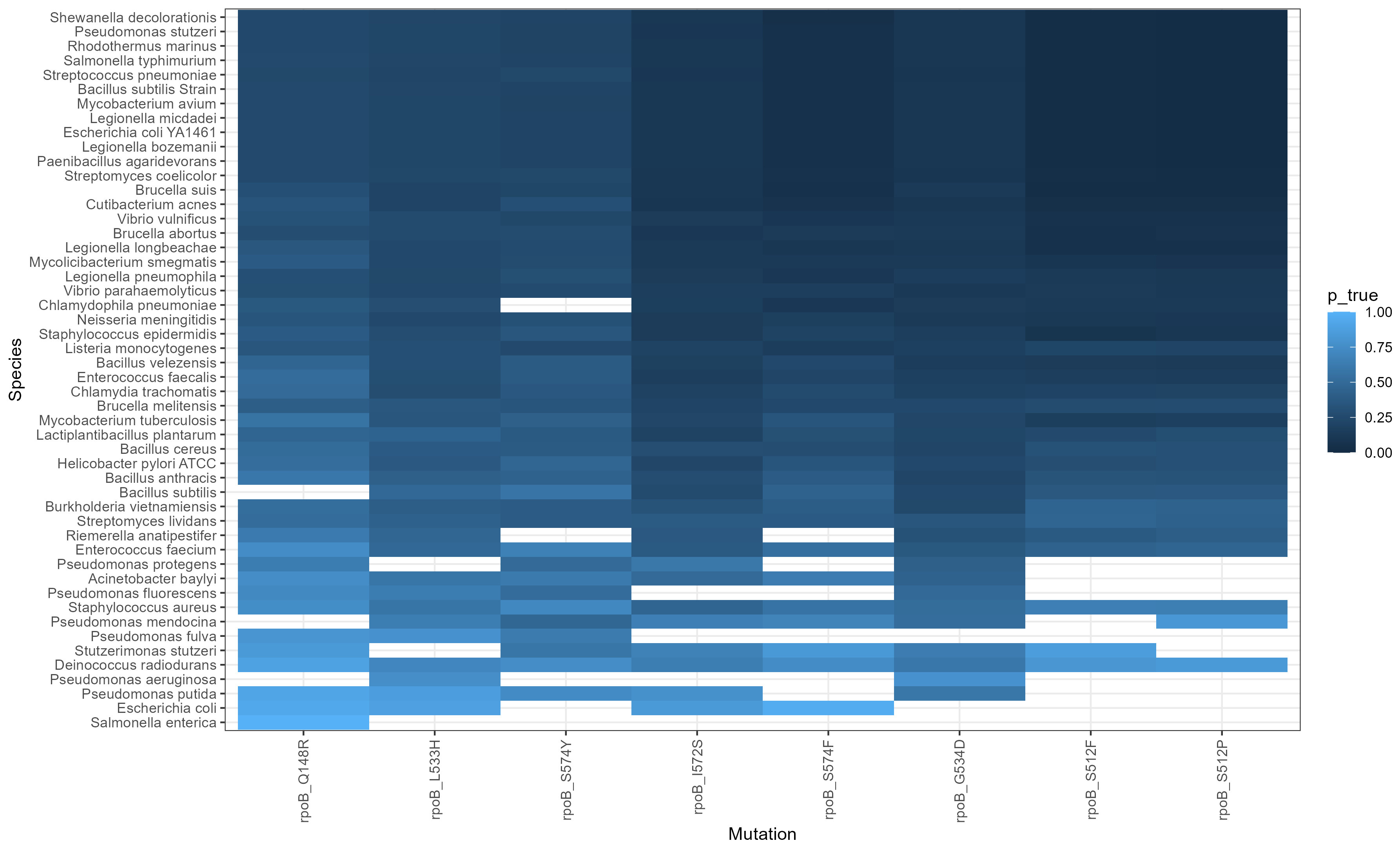
- Highlight representative novel sites (e.g., L533R, Q148R).



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Species | Novel\_Count | Mean\_Prob | Candidate1\_Mutation | Candidate2\_Mutation | Candidate3\_Mutation |
| Deinococcus radiodurans | 8 | 0.748578812656373 | rpoB\_Q148R | rpoB\_S512P | rpoB\_S512F |
| Staphylococcus aureus | 8 | 0.6132635026502922 | rpoB\_Q148R | rpoB\_S574Y | rpoB\_S512F |
| Enterococcus faecium | 8 | 0.5098281033263364 | rpoB\_Q148R | rpoB\_S574Y | rpoB\_S574F |
| Streptomyces lividans | 8 | 0.4192693982589603 | rpoB\_Q148R | rpoB\_S512F | rpoB\_L533H |
| Burkholderia vietnamiensis | 8 | 0.3989763259749059 | rpoB\_Q148R | rpoB\_S512F | rpoB\_S512P |

4.4 Species-level mutation prediction pattern

- Heatmap showing predicted mutation profiles per species (Fig. Z)



4.5 Biological interpretation

- Summarize Gram+/– divergence and functional implications.

- Connect unsupervised and supervised findings.

# Discussion

**Can ASreview be optimized?**

Currently, ASreview does not offer batch automatic labeling, relying entirely on users to manually assign labels (relevant/irrelevant). Its AI algorithm is only used to rank recommended articles. Users must rely on ASreview's recall regression curve for the current project to determine whether to stop reading subsequent articles deemed "irrelevant." The author believes this process is still cumbersome and should be combined with a language understanding model to automatically perform all subsequent screening after a small amount of manual labeling, or to leave only a few "suspicious" articles for users to judge.

**Why are RIF mutation sites highly disproportionate in some species compared to others?**

Vibrio parahaemolyticus, V. vulnificus, Streptomyces lividans, Brucella suis & Brucella melitensis

In the PU-learning and mask and recall processes, how does masking affect the final prediction model?

**Is random masking the optimal approach?**

Can ASreview's recommendation prioritization or other language vocabulary analysis models be combined to perform high-value, targeted masking?

**Are the final predictions too credible?**

# 5. Conclusions

Summarize the most significant findings and their implications. State whether the research aims were achieved and highlight potential applications or follow-up studies.

# 6. References

All cited literature should appear here in a consistent referencing style (APA, Harvard, or a scientific journal format). Example:  
Foster, L., Mouse, M., & Christ, J. (1972). The effect of hypoxia on free divers. J. Irrep. Res., 23, 490–512.

# 7. Figures and Tables

All figures and tables should be labeled and inserted close to where they are first mentioned in the text. Each figure should include a descriptive caption below, while tables should have a title above.

# 8. Appendices

Include any supplementary materials, extended data tables, code snippets, or detailed methodology not essential to the main text.