

# EVALUATING PROTEIN SELECTION IN CRITICAL CARE PROTEOMICS

A UNIFIED VALIDATION FRAMEWORK  
FOR ARDS IN SEPSIS

LISA FREDRIKSSON, LOVE MATTSSON

Master's thesis  
2026:E61



LUND INSTITUTE OF TECHNOLOGY  
Lund University

Faculty of Engineering  
Centre for Mathematical Sciences  
Mathematical Statistics

Evaluating Protein Selection in  
Critical Care Proteomics:  
A Unified Validation Framework  
for ARDS in Sepsis

---

Utvärdering av proteinselektion  
inom proteomik i intensivvården:  
Ett enhetligt valideringsramverk  
för ARDS vid sepsis

---

Love Mattsson & Lisa Fredriksson



## Abstract

Identifying proteomic patterns associated with acute respiratory distress syndrome (ARDS) in sepsis patients is a central challenge in critical care research. Standard univariate approaches may miss non-linear associations and higher-order interactions that reflect the true complexity of the syndrome, yet systematic evaluation of alternative methods is lacking. This thesis introduces a unified evaluation framework for protein selection methods, combining simulation validation on synthetic data with planted signals of known type, classifier validation on held-out patient data, and pathway enrichment analysis. The framework is applied to a cohort of 512 adult sepsis patients from the SWECRIT multicentre Swedish ICU registry, of whom 73 developed moderate or severe ARDS, using proteomic measurements from the SomaScan 11K Assay.

Three selection methods are evaluated: Welch's  $t$ -test, mutual information with permutation-based significance, and random forest feature importance via SHAP values. The  $t$ -test is restricted by construction to mean-shift regulation, but identifies a small set of stable proteins associated with mitochondrial electron transport. Mutual information can in principle detect arbitrary univariate signals, but is power-limited at the cohort sample size and recovers few proteins at moderate effect sizes; nevertheless, its top-ranked proteins yield the strongest biological enrichment signal, with ketone body metabolism as the most robustly supported pathway. RF+SHAP also targets arbitrary univariate signals and produces the strongest classifier performance, with mean AUC up to 0.667 and consistent improvement over random selection, but lacks a principled mechanism for FDR control. No method recovers pure pairwise interaction signals, confirming that all approaches operate at the univariate level. The divergence between classifier and enrichment performance across methods demonstrates that complementary validation strategies are necessary to fully characterise a selection method.

## Sammanfattning

Att identifiera proteinmönster associerade med lungsyndromet acute respiratory distress syndrome (ARDS) hos patienter med underliggande sepsis är en central utmaning inom intensivvårdsforskning. Konventionella univariata metoder riskerar att missa icke-linjära associationer och högre ordningens interaktioner som speglar syndromets sanna komplexitet, och systematisk utvärdering av alternativa metoder saknas. Detta examensarbete introducerar ett enhetligt utvärderingsramverk för proteinselektionsmetoder, som kombinerar simuleringsvalidering på syntetiska data med planterade signaler av känd typ, klassificeringsvalidering på hållen testdata, samt vägberikning som biologisk validering. Ramverket tillämpas på en kohort bestående av 512 vuxna sepsispatienter från SWECRIT-registret, ett flercenterstudie vid svenska intensivvårdsavdelningar, varav 73 utvecklade måttlig eller svår ARDS. Proteinuttryck mättes med SomaScan 11K-plattformen.

Tre selektionsmetoder utvärderas: Welchs  $t$ -test, ömsesidig information med permutationsbaserat signifikanstest, samt random forest-baserad variabelvikt via SHAP-värden.  $t$ -testet är per konstruktion begränsat till medelvärdesförskjutningar, men identifierar en liten uppsättning stabila proteiner kopplade till mitokondriell elektrontransport. Ömsesidig information kan i princip detektera godtyckliga univariata signaler, men är styrkebegränsad vid den aktuella kohortstorleken och återfinner få proteiner vid måttliga effektstorlekar; trots detta ger dess topprankade proteiner den starkaste biologiska berikningssignalen, med ketonkroppsmetabolism som det mest robusta resultatet. RF+SHAP riktar sig också mot godtyckliga univariata signaler och uppvisar den starkaste klassificeringsförmågan, med ett medel-AUC upp till 0,667 och konsekvent förbättring jämfört med slumpmässig selektion, men saknar en principiell mekanism för FDR-kontroll. Ingen metod återfinner rena parvisa interaktionssignaler, vilket bekräftar att samtliga utvärderade metoder opererar på univariat nivå. Divergensen mellan klassificerings- och berikningsresultat över metoder visar att komplementära valideringsstrategier är nödvändiga för att fullständigt karakterisera en selektionsmetod.

## Acknowledgements

Research, we have come to learn, is something like fumbling in a darkened room, catching onto glimmers, thoughts, and half-formed ideas that lead you somewhere until they either fade or light up a path you had not anticipated. This project has consisted of many such glimmers, most of them leading somewhere other than we first imagined, but none the less important for that. We also discovered, perhaps unsurprisingly, that the most productive discussions often began with someone saying *“I don’t really know, but...”*, a phrase we have come to regard with deep affection.

We owe a great deal of gratitude to our supervisors Andreas Jakobsson and Attila Frigyesi for sparking thoughts, broadening our ways of thinking, and encouraging us to make bold guesses when our more theoretical instincts were holding us back. We also wish to thank Lund University, and the education in Engineering Physics and Mathematics that made this work possible.

Finally, we wish to thank our friends for their company and encouragement throughout this process, and the north foyer of the Mathematics building and the café at Studiecetrum for providing the reassuringly solid brick-walled square metres within which most of this thesis came to life.

# Contents

|          |                                                                         |           |
|----------|-------------------------------------------------------------------------|-----------|
| <b>1</b> | <b>Introduction</b>                                                     | <b>7</b>  |
| <b>2</b> | <b>Background</b>                                                       | <b>8</b>  |
| 2.1      | Sepsis and ARDS . . . . .                                               | 8         |
| 2.2      | Proteomic Approaches in Critical Care . . . . .                         | 8         |
| <b>3</b> | <b>Data</b>                                                             | <b>9</b>  |
| 3.1      | SomaLogic – SomaScan . . . . .                                          | 9         |
| 3.2      | Cohort and Data . . . . .                                               | 9         |
| 3.3      | Data Exploration . . . . .                                              | 9         |
| 3.3.1    | Class distribution and imbalance . . . . .                              | 9         |
| 3.3.2    | Clinical covariates . . . . .                                           | 10        |
| 3.3.3    | Proteomic data properties . . . . .                                     | 11        |
| <b>4</b> | <b>Methodology Framework</b>                                            | <b>13</b> |
| 4.1      | Overview . . . . .                                                      | 13        |
| 4.2      | Cohort Preparation . . . . .                                            | 13        |
| 4.3      | Split Generation . . . . .                                              | 13        |
| 4.4      | Protein Selection . . . . .                                             | 14        |
| 4.4.1    | Ranked vs. native list . . . . .                                        | 14        |
| 4.5      | Final Protein Selection . . . . .                                       | 14        |
| 4.5.1    | Stability selection and diagnostics . . . . .                           | 14        |
| 4.6      | Validation . . . . .                                                    | 15        |
| 4.7      | Cross-Method Comparison . . . . .                                       | 15        |
| 4.8      | Reproducibility . . . . .                                               | 15        |
| <b>5</b> | <b>Validation Framework</b>                                             | <b>16</b> |
| 5.1      | Simulation Validation . . . . .                                         | 16        |
| 5.1.1    | Data generation . . . . .                                               | 16        |
| 5.1.2    | Evaluation . . . . .                                                    | 21        |
| 5.2      | Classifier Validation . . . . .                                         | 21        |
| 5.2.1    | Repeated stratified random subsampling . . . . .                        | 21        |
| 5.2.2    | Adjusting for correlated test sets . . . . .                            | 22        |
| 5.2.3    | Reporting cross-method results . . . . .                                | 23        |
| 5.2.4    | Classifiers . . . . .                                                   | 23        |
| 5.2.5    | Class imbalance and evaluation metrics . . . . .                        | 24        |
| 5.2.6    | Hyperparameter configuration . . . . .                                  | 25        |
| 5.3      | Enrichment Analysis Validation . . . . .                                | 25        |
| 5.3.1    | A biological explainability score from enrichment results . . . . .     | 26        |
| 5.3.2    | Permutation null and reporting . . . . .                                | 26        |
| 5.3.3    | An example with biological explainability score . . . . .               | 27        |
| <b>6</b> | <b>Random Protein Selection</b>                                         | <b>28</b> |
| 6.1      | Results . . . . .                                                       | 28        |
| 6.1.1    | Simulation validation . . . . .                                         | 28        |
| 6.1.2    | Classifier validation . . . . .                                         | 29        |
| 6.1.3    | Enrichment validation . . . . .                                         | 30        |
| <b>7</b> | <b>Differential Expression Analysis via Welch’s <math>t</math>-test</b> | <b>31</b> |
| 7.1      | Welch’s $t$ -test . . . . .                                             | 31        |
| 7.2      | Multiple Testing Correction . . . . .                                   | 31        |
| 7.3      | Limitations . . . . .                                                   | 31        |
| 7.4      | Results . . . . .                                                       | 31        |
| 7.4.1    | Simulation validation . . . . .                                         | 31        |
| 7.4.2    | Classifier validation . . . . .                                         | 34        |
| 7.4.3    | Stability diagnostics . . . . .                                         | 36        |
| 7.4.4    | Enrichment validation . . . . .                                         | 36        |
| <b>8</b> | <b>A Mutual Information Based Selection</b>                             | <b>40</b> |

|           |                                                              |           |
|-----------|--------------------------------------------------------------|-----------|
| 8.1       | Mutual Information . . . . .                                 | 40        |
| 8.1.1     | Entropy . . . . .                                            | 40        |
| 8.1.2     | Mutual information . . . . .                                 | 40        |
| 8.1.3     | Estimating mutual information . . . . .                      | 41        |
| 8.2       | Permutation Testing to Determine Significance . . . . .      | 41        |
| 8.2.1     | Reducing computational complexity . . . . .                  | 41        |
| 8.3       | Advantages and Limitations . . . . .                         | 42        |
| 8.4       | Results . . . . .                                            | 43        |
| 8.4.1     | Simulation validation . . . . .                              | 43        |
| 8.4.2     | Classifier validation . . . . .                              | 45        |
| 8.4.3     | Stability diagnostics . . . . .                              | 45        |
| 8.4.4     | Enrichment validation . . . . .                              | 46        |
| <b>9</b>  | <b>Random Forest Selection through SHAP values</b>           | <b>51</b> |
| 9.1       | Random Forest and SHAP . . . . .                             | 51        |
| 9.1.1     | TreeSHAP and feature perturbation . . . . .                  | 51        |
| 9.2       | Hyperparameters . . . . .                                    | 52        |
| 9.3       | Selection . . . . .                                          | 52        |
| 9.3.1     | Top- $k$ selection . . . . .                                 | 52        |
| 9.3.2     | Native selection via stability selection . . . . .           | 52        |
| 9.4       | Advantages and Limitations . . . . .                         | 52        |
| 9.5       | Results . . . . .                                            | 53        |
| 9.5.1     | Simulation validation . . . . .                              | 53        |
| 9.5.2     | Classifier validation . . . . .                              | 55        |
| 9.5.3     | Stability diagnostics . . . . .                              | 55        |
| 9.5.4     | Enrichment validation . . . . .                              | 56        |
| <b>10</b> | <b>Cross-Method Comparison</b>                               | <b>60</b> |
| 10.1      | Simulation Results . . . . .                                 | 60        |
| 10.2      | Classifier Results . . . . .                                 | 62        |
| 10.3      | Enrichment Results . . . . .                                 | 64        |
| <b>11</b> | <b>Discussion</b>                                            | <b>66</b> |
| 11.1      | Method-Specific Findings and Cross-Method Patterns . . . . . | 66        |
| 11.2      | Framework Strengths and Limitations . . . . .                | 68        |
| 11.3      | Clinical and Biological Implications . . . . .               | 68        |
| 11.4      | Future Work . . . . .                                        | 69        |
| <b>12</b> | <b>Conclusions</b>                                           | <b>71</b> |
|           | <b>Populärvetenskaplig Sammanfattning</b>                    | <b>76</b> |

# 1 Introduction

Sepsis and acute respiratory distress syndrome (ARDS) are among the most severe conditions encountered in intensive care, together accounting for a substantial proportion of critical care mortality worldwide [SDS<sup>+</sup>16, BW22]. Despite decades of research, no targeted therapies have been approved for either condition, and treatment remains largely supportive.

Proteomic data offers a window into the molecular state of a patient at the time of measurement, and has emerged as a promising source of biomarkers for disease stratification and outcome prediction in critical care. However, in clinical proteomics studies, the analytical methods used to identify informative proteins are predominantly univariate, evaluating each protein independently for mean-level differences between groups. Proteins do not act in isolation but as parts of interconnected signalling networks, and disease-relevant patterns may emerge from interactions between proteins rather than from shifts in individual protein levels, a structure that univariate methods cannot detect regardless of whether they are linear or not [BGL11]. Understanding these patterns is a long-term goal of proteomic research in critical care, though revealing new biology is not achievable without first establishing which selection methods are reliable and what they are capable of detecting.

A further challenge is that evaluating protein selection methods in clinical data is fundamentally difficult: there is no ground truth for which proteins are truly ARDS-associated, making it hard to distinguish genuine biological signal from statistical artifact. This motivates the development of a principled validation framework as a research contribution in its own right.

The aim of this thesis is twofold. First, to develop such a validation framework, combining simulated data with planted signals of known type and effect size, classifier performance on real data, and biological enrichment analysis, enabling controlled and fair comparison of arbitrary selection methods. Second, to apply this framework to evaluate a range of selection methods, from the standard univariate *t*-test to information-theoretic and supervised learning approaches, characterizing what each method can and cannot detect rather than optimizing predictive performance. Higher-order interaction effects, such as synergistic protein combinations, are identified as a theoretically motivated direction but are left for future work. The immediate goal is methodological: to establish which selection strategies are reliable, interpretable, and robust enough to serve as a foundation for future biological investigation.

## 2 Background

### 2.1 Sepsis and ARDS

Sepsis is a life-threatening condition caused by a dysregulated immune response to infection [SDS<sup>+</sup>16]. Rather than the infection itself, it is the host’s own inflammatory response that drives organ dysfunction and, in severe cases, death. Sepsis is the most common precipitant of acute respiratory distress syndrome (ARDS) [BW22], and the two conditions frequently co-occur in critically ill patients. ARDS is characterized by severe respiratory failure, bilateral lung infiltrates, and impaired gas exchange, and carries a mortality rate of 30–45 % [BW22]. Both conditions are highly heterogeneous at the molecular level: patients meeting the same clinical criteria may differ substantially in their underlying protein expression patterns, reflecting distinct biological processes that are not captured by clinical classification alone [SP23]. Identifying which molecular patterns are robustly associated with disease is therefore a central challenge in critical care research.

### 2.2 Proteomic Approaches in Critical Care

Proteins are central mediators of the biological processes that drive sepsis and ARDS. As functional molecules that reflect the current physiological state of the body, proteins are well-suited for studying disease mechanisms and identifying patient-relevant biomarkers [LF19]. Large-scale proteomic platforms such as SomaScan now make it possible to measure thousands of proteins simultaneously from a single blood sample, providing a rich source of molecular information for clinical research [Som23].

The standard analytical approach is to evaluate proteins individually, using univariate statistical tests such as the  $t$ -test to identify proteins whose mean expression differs between patient groups [LF19, KCTR15]. This approach is straightforward to interpret and has identified several candidate biomarkers in sepsis and ARDS research. However, it is limited to detecting linear, additive effects and cannot capture the complex regulatory relationships that govern biological signalling. Proteins do not act in isolation but as parts of interconnected pathways, and disease-relevant patterns may emerge from interactions between proteins rather than from shifts in individual protein levels, a structure that univariate methods cannot detect regardless of whether they are linear or not [BGL11]. A further challenge is the high-dimensional nature of proteomic data, where the number of measured proteins far exceeds the number of patients ( $p \gg n$ ), increasing the risk of false positives and making robust feature selection necessary [LF19, KCTR15]. Validating whether an identified protein set reflects true biological signal rather than an artefact of the data is inherently difficult, particularly in small clinical cohorts where overfitting and selection bias are persistent risks [LF19].

## 3 Data

### 3.1 SomaLogic – SomaScan

Protein abundances were measured using the SomaScan 11K Assay v5.0 (SomaLogic, Boulder, CO), a highly multiplexed, aptamer-based proteomics platform capable of quantifying over 10,000 unique human proteins from small sample volumes [Som23]. The assay relies on Slow Off-rate Modified Aptamer (SOMAmer) reagents, which are chemically modified single-stranded DNA molecules that bind target proteins with high specificity.

Protein levels were measured across three serial dilutions (1:5, 1:200, and 1:20,000), enabling a wide dynamic range of detection spanning up to  $10^{10}$  in plasma. The output is reported in relative fluorescence units (RFU), which are proportional to the amount of target protein in the original sample.

Raw RFU values were  $\log_{10}$ -transformed prior to analysis to stabilize variance and approximate normality. Data standardization, including hybridization normalization and signal normalization, was applied by SomaLogic to reduce systematic technical variation [Som23].

### 3.2 Cohort and Data

The study sample is drawn from the SWECRIT cohort, a multicentre Swedish ICU registry, and includes adult patients with confirmed sepsis. Two groups were considered: sepsis patients who developed moderate or severe ARDS, and sepsis patients without ARDS, defined according to the Berlin criteria [RRTT<sup>+</sup>12]. Patients without confirmed sepsis, as well as those with mild ARDS or missing outcome labels, were excluded. This resulted in a final cohort of 512 patients, of whom 73 had ARDS and 439 did not. Notably, the largest source of attrition was the missing label drop, which removed 856 patients, the majority of the initial cohort, in a single filtering stage (Table 1). If this missingness is not random but related to patient characteristics or disease severity, the resulting cohort may not be representative of the broader ICU population, which should be considered when generalising the findings.

Protein abundances were measured across 10,776 SOMAmer reagent features, covering over 10,000 unique human proteins, using the SomaScan 11K Assay v5.0, as described in Section 3.1. The data were used as provided after SomaLogic’s standard preprocessing pipeline, and no additional normalization was applied unless otherwise stated.

Available covariates include age, sex, ARDS severity (moderate or severe), ICU site, and survival or ICU length of stay. Their distributions are summarized in Table 2, and discussed further in section 3.3.2.

Table 1: Cohort construction. Patients were sequentially filtered to retain sepsis patients with moderate or severe ARDS and sepsis patients without ARDS. Mild ARDS was excluded to sharpen the clinical contrast between groups.

| Step                                                    | Patients retained | Removed |
|---------------------------------------------------------|-------------------|---------|
| Initial dataset                                         | 1 449             | —       |
| After missing label drop                                | 593               | 856     |
| After non-sepsis filter ( <i>Sepsis</i> = <i>TRUE</i> ) | 521               | 72      |
| After mild-ARDS exclusion                               | 512               | 9       |
| <b>Final cohort</b>                                     | <b>512</b>        | —       |

### 3.3 Data Exploration

#### 3.3.1 Class distribution and imbalance

The final cohort, described in Table 2, consists of 512 patients: 73 with moderate or severe ARDS and 439 without ARDS, corresponding to an ARDS prevalence of 14.3%. Among ARDS patients, 48 (65.8%) were classified as moderate and 25 (34.2%) as severe. This approximately 1:6 class imbalance is addressed explicitly in all classification analyses. The cohort composition and severity breakdown are shown in Figure 1.

Table 2: Baseline characteristics of the final cohort ( $n = 512$ ). Continuous variables are reported as mean  $\pm$  SD. Percentages are column proportions within each ARDS group.  $p$ -values are from Welch’s  $t$ -test (age) and Fisher’s exact test (sex); *n.s.* denotes  $p > 0.05$ .

|                                           | All<br>( $n = 512$ ) | non-ARDS<br>( $n = 439$ ) | ARDS<br>( $n = 73$ ) | $p$         |
|-------------------------------------------|----------------------|---------------------------|----------------------|-------------|
| <i>Demographics</i>                       |                      |                           |                      |             |
| Age, yr (mean $\pm$ SD)                   | 65.5 $\pm$ 14.46     | 65.6 $\pm$ 14.61          | 64.9 $\pm$ 13.65     | <i>n.s.</i> |
| Female, $n$ (%)                           | 197 (38.5)           | 169 (38.5)                | 28 (38.4)            | <i>n.s.</i> |
| <i>ARDS severity (ARDS patients only)</i> |                      |                           |                      |             |
| Moderate, $n$ (%)                         | —                    | —                         | 48 (65.8)            | —           |
| Severe, $n$ (%)                           | —                    | —                         | 25 (34.2)            | —           |
| <i>ICU site</i>                           |                      |                           |                      |             |
| Helsingborg, $n$ (%)                      | 134 (26.2)           | 124 (28.2)                | 10 (13.7)            | —           |
| Lund, $n$ (%)                             | 220 (43.0)           | 186 (42.4)                | 34 (46.6)            | —           |
| Malmö, $n$ (%)                            | 158 (30.9)           | 129 (29.4)                | 29 (39.7)            | —           |

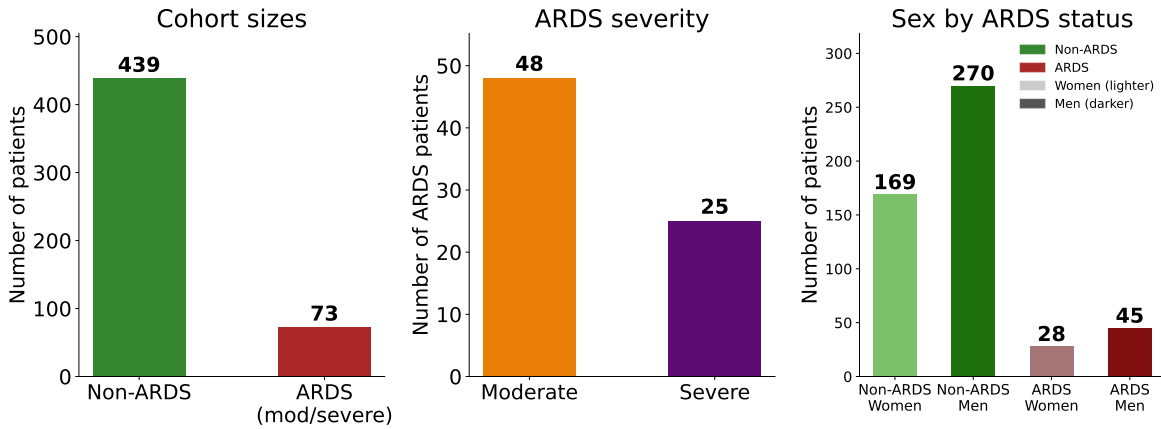


Figure 1: Cohort overview. *Left*: patient counts by ARDS status. *Middle*: ARDS severity distribution among ARDS patients. *Right*: sex distribution by ARDS status.

### 3.3.2 Clinical covariates

Age distributions were similar between groups, with a mean of 65 years in ARDS patients and 66 years in non-ARDS patients (Figure 2a), suggesting age is not a primary driver of proteomic differences. The cohort consisted of approximately 39 % women and 61 % men, with similar proportions across ARDS and non-ARDS groups (Table 2).

Patients were recruited from three ICUs: Helsingborg, Lund, and Malmö (Figure 2b). ARDS prevalence varied substantially across sites, ranging from 7.5 % at Helsingborg to 18.4 % at Malmö, which may reflect differences in patient severity or referral patterns. Age, sex, and ICU site are potential confounders that are not adjusted for in the protein selection analyses. Since all methods are evaluated on the same data and the same splits, any systematic bias introduced by confounders affects all methods approximately equally and does not distort the comparison between them. However, the absolute findings, that is, which specific proteins are identified as ARDS-associated, may be confounded, and should be interpreted with caution.

It should be noted that several potential confounders are not systematically available in the current dataset, these include factors such as the underlying cause of sepsis, time from sepsis onset to sampling, comorbidity burden, and concurrent treatments. This represents a limitation and is discussed where relevant.

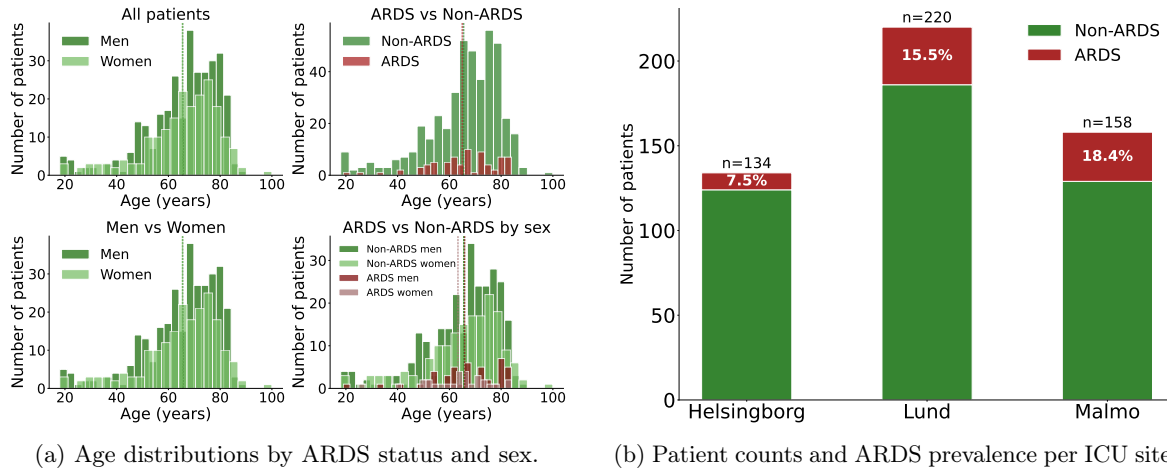


Figure 2: Clinical covariate distributions. (a) Age distributions stratified by ARDS status and sex; dashed lines indicate group means. (b) Patient counts and ARDS prevalence per ICU site.

### 3.3.3 Proteomic data properties

SomaScan expression values are  $\log_{10}$ -transformed across the  $\sim 11,000$  aptamers, shown in Figure 3. No additional transformation was applied to the expression values prior to analysis unless otherwise stated. No missing values were observed in the protein expression data, and therefore no imputation was required.

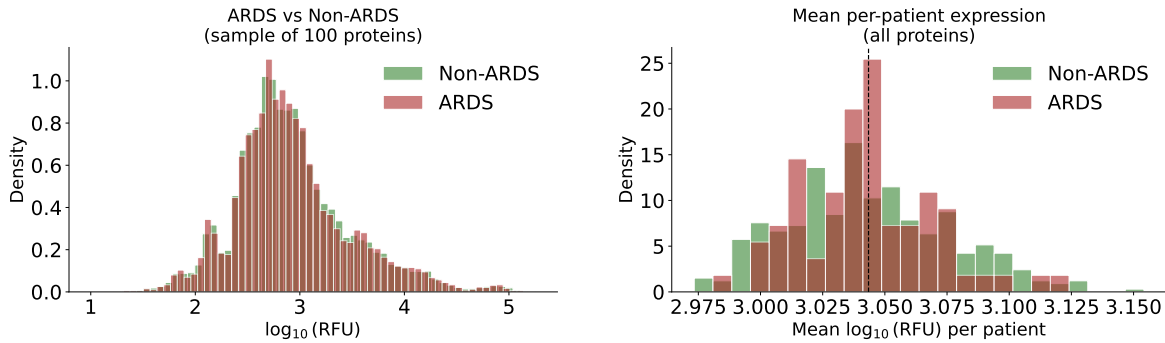


Figure 3: Distribution of SomaScan protein expression values ( $\log_{10}$  RFU). *Left*: expression distributions for ARDS and non-ARDS patients across a sample of 100 proteins. *Right*: mean per-patient expression across all proteins; dashed line indicates the grand mean.

Figure 4 illustrates the expression distributions of six proteins identified as informative by subsequent selection methods. Even for the most strongly associated proteins, the distributions of ARDS and non-ARDS patients overlap considerably, and no single protein cleanly separates the two groups. This highlights a fundamental challenge in proteomic classification: individual proteins carry limited discriminative information. Notably, dysregulation is not unidirectional, while SDHAF2, BLOC1S2, and ALDH5A1 show lower expression in ARDS patients, GOLM1 and MFAP5 show higher expression, reflecting the biological complexity of the disease.

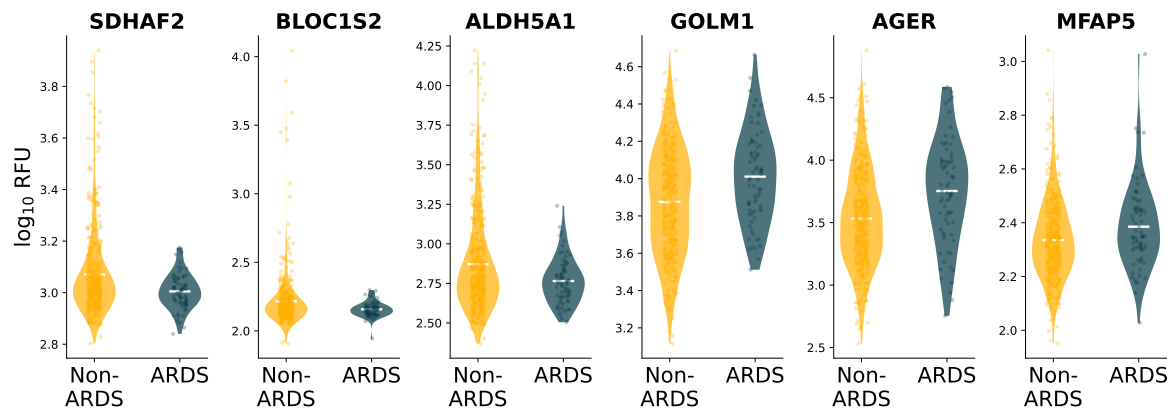


Figure 4: Expression distributions of six proteins across ARDS and non-ARDS patients. White markers indicate the mean; jittered points show individual observations. Values are on the  $\log_{10}$  RFU scale. Despite visible differences in mean expression for some proteins, all distributions show substantial overlap between groups.

## 4 Methodology Framework

This section describes the general framework used to identify proteins associated with ARDS in the sepsis cohort and evaluate protein selection methods. The framework is designed to support comparison of multiple protein selection methods, including methods sensitive to non-linear regulation and higher-order interactions, under a unified evaluation protocol. Specific selection methods are introduced in their respective sections; this section defines only the shared pipeline.

### 4.1 Overview

The pipeline is organized as a sequence of independent stages. Each stage consumes the output of the previous one and writes its own results, so that any single stage can be re-run without re-executing upstream work and so that adding or replacing a selection method does not affect the rest of the pipeline. The same pipeline is applied to every selection method to ensure comparability. Adding a new method only requires implementing the selection procedure; the rest of the pipeline is shared.

1. **Cohort preparation:** apply inclusion and exclusion criteria to the SWECRIT cohort and produce a cleaned analysis table.
2. **Split generation:** build a shared collection of stratified train/test splits over patients, used by every selection method, so that paired comparisons across methods are well-defined at the split level.
3. **Protein selection:** apply a given selection method to each training split, and once to the full cohort to produce the final reported ranking.
4. **Classifier validation:** train and evaluate fixed-hyperparameter classifiers on each split using the proteins selected on that split.
5. **Enrichment validation:** run pathway over-representation analysis on the final selected set, with a custom enrichment score and a permutation-based null distribution.
6. **Simulation validation:** apply the same selection method to synthetic data with planted signals of known type and measure recall and false discovery proportion (FDP) per signal type.
7. **Cross-method aggregation:** consolidate the validation outputs into comparison across methods.

### 4.2 Cohort Preparation

The preparation stage performs only sample-level filtering. No transformation of expression values is applied; SomaScan values are assumed to be normalized upstream.

The procedure can be summarized as:

1. Load the SomaScan expression matrix  $X \in \mathbb{R}^{n \times p}$  and clinical labels.
2. Apply inclusion and exclusion criteria: keep only patients that are either (a) sepsis-positive and ARDS-negative, or (b) sepsis-positive and ARDS-positive with severity *not mild* (moderate or severe).
3. Resolve ARDS severity by reconciling the available severity indicator columns; rows with missing or contradictory severity information are dropped.
4. Verify there are no missing values in  $X$ ; abort if any are found so that handling can be documented before any downstream step.

The output is a cleaned pair  $(X, y)$  with  $y \in \{0, 1\}^n$  (with 1 denoting ARDS), which serves as the input to all subsequent stages.

### 4.3 Split Generation

All downstream stages operate on a shared set of  $B$  stratified train/test splits, generated once and reused across methods. This shared structure is what makes per-split paired comparisons across methods well-defined: identical fold indices mean that any performance difference between two methods

on split  $b$  is attributable to the methods themselves rather than to which patients happened to fall in the test set.

The default configuration is  $B$  splits at an 80/20 train/test ratio, stratified on  $y$ , with per-split seeds derived deterministically from a single global seed. The statistical rationale for repeated subsampling and the choice of  $B$  are deferred to Section 5.2, where they are tied to the classifier validation procedure that motivates them.

## 4.4 Protein Selection

A protein selection method is treated as a procedure that, given a training set  $(X^{\text{train}}, y^{\text{train}})$ , returns a complete ranking of all  $p$  proteins, a per-protein score and  $q$ -value (adjusted  $p$ -value), and a Boolean significance mask defining the method’s native selection. The framework treats each method as a black box satisfying this interface; outputs are validated for consistency before being recorded.

Methods covered in this thesis:

- **Random selection** (null baseline): select random proteins, used to anchor expectations under no signal.
- **Two-sample  $t$ -test** with Benjamini–Hochberg FDR correction (standard baseline).
- **Mutual information** with a permutation-based null.
- **Random forest SHAP** importance ranking with a stability-controlled native threshold for final selection, detailed in Section 4.5.1, and 9.3.2.

### 4.4.1 Ranked vs. native list

For each method we distinguish a ranked list from a native list. The ranked list orders all proteins by some score (e.g. adjusted  $p$ -value); the top- $k$  entries form the top- $k$  selection. The native list is the subset of proteins that satisfy a method-specific significance (e.g. adjusted  $p$ -value below  $\alpha$ ). The term “native” reflects that this list is determined by the selection method itself, without imposing an external size constraint such as a fixed  $k$ . The difference between these is like asking the selection method, which proteins do you believe to be relevant, contra if you had to choose  $k$  proteins, which would you choose. Downstream stages evaluate both views: classifier and simulation validation report results at top- $k$  for  $k \in \{10, 25, 50, 100\}$  and at the native cutoff when possible.

## 4.5 Final Protein Selection

The final reported protein set  $S^{\text{final}}$  for a method, used for enrichment validation (Section 5.3) and downstream interpretation, is obtained by running the selection method once on the full cohort as this will provide the highest possible power. When the resulting native set has at least a minimum size it is used directly; otherwise a top- $k$  fallback is applied so that downstream enrichment has a sufficiently large gene list.

### 4.5.1 Stability selection and diagnostics

Since repeated random subsampling is performed, we can analyze the stability of the proteins selected in the splits. The selection frequency for protein  $j$  is

$$f_j = \frac{1}{B} \sum_{b=1}^B \mathbb{1}[j \in S_b].$$

If a given selection method does not have a native selection we could use stability selection [MB10], where the final set of proteins is defined by

$$S^{\text{final}} = \{j : f_j \geq \tau\},$$

where  $\tau$  is a stability threshold. The original stability selection framework of [MB10] derives theoretical bounds on the false discovery rate under the assumption that each subsample contains exactly  $\lfloor n/2 \rfloor$  observations [MB10]. Since we draw subsamples of size  $0.8n$  to ensure sufficient training data for the classifiers, these theoretical guarantees do not strictly apply. Nevertheless, the underlying principle holds: proteins with a genuine association with ARDS should appear consistently across subsamples

regardless of the exact subsample size, and larger subsamples are if anything more stable, making the selection frequencies a conservative diagnostic rather than a liberal one. Even if a method has a native selection, we still report the selection frequencies as a stability diagnostic of the final selected proteins.

## 4.6 Validation

Three validation procedures are applied. Detailed definitions are given in Section 5.

- **Simulation validation:** the method is applied to synthetic datasets in which a small number of proteins are designated as true signals, with the rest acting as noise embedded in a correlated covariance structure. Several signal types are planted, ranging from purely linear effects to non-linear and interaction-based effects, at varying effect sizes. Because ground truth is known, recall and false discovery proportion (FDP) can be measured directly. Simulation validation confirms whether a given method is able to detect a given type of signal under controlled conditions. See Section 5.1.
- **Classifier validation:** on each split, a small set of standard classifiers spanning linear and non-linear families is trained on the selected proteins and evaluated on the held-out test set, with classifier hyperparameters fixed across methods so that differences are attributable to the selection step rather than to model tuning. Performance is reported across several feature-set definitions (top- $k$  at multiple  $k$ , and the method’s native set). The factorial design over selectors, classifiers, and feature-set definitions tests whether non-linear classifiers recover signal that linear ones miss. Classifier validation determine whether a method selects proteins in our data that are more ARDS informative than random proteins. See Section 5.2.
- **Enrichment validation:** pathway over-representation analysis (ORA) is run on the final selected set  $S^{\text{final}}$  against several pathway libraries, with the measured SomaScan proteins as background. In addition to standard ORA statistics, we summarize each library with a custom Biological Explainability Score (BES). To calibrate these scores against what would be expected by chance, a permutation null is constructed by repeatedly applying the same procedure to random gene lists of matching size drawn from the background. Enrichment validation gives an idea of the biological relevance of the proteins identified. See Section 5.3.

None of these validation procedures is sufficient alone; we must combine them to support evidence that a given method is relevant. For example, a method may identify a specific type of signal in the simulation, suggesting that it can detect such signals, but without the other validation procedures there is no guarantee that this type of signal exists in the real data. Classifier validation gives no biological interpretation of the proteins selected, and enrichment validation will favor already-studied proteins.

## 4.7 Cross-Method Comparison

A dedicated comparison stage consolidates the validation outputs across methods. Per-method sections report results for that method only, while cross-method tables and figures, AUC and AUC-PR across selectors and classifiers, simulation recall and FDP heatmaps, and BES comparison across libraries appear in the comparison section. Because all methods are run on the same splits, classifier scores are paired across methods at the split level, and uncertainty for paired-split differences is reported with the Nadeau–Bengio correction for repeated train/test resampling. Paired permutation tests are used to assess whether classifier performance differs significantly from the random selection baseline.

## 4.8 Reproducibility

- All random seeds are fixed and logged. Per-split and per-repeat seeds are derived deterministically from a single global seed, so that the entire pipeline can be reproduced from that seed alone.
- The same  $B$  splits are used across all selection methods, so paired comparisons per split are well-defined.
- Each stage’s outputs are saved, so any single stage can be re-run without re-executing upstream stages, and downstream stages can be re-run for a single method without disturbing the others.
- Adding a new selection method requires implementing only the selection procedure introduced in Section 4.4; the rest of the pipeline is shared.

## 5 Validation Framework

Given a set of proteins from a protein selection method, we require some way of validating the set, in the sense that we want to know that the selected proteins are truly associated with ARDS and not just artifacts of the selection procedure. We consider three complementary validations: classifier validation, enrichment validation, and simulation validation. Classifier and enrichment validation are applied to selections produced from the real SWECRIT data; simulation validation is applied to synthetic data with planted signals of known type and is used to characterise what each selection method is actually able to detect.

### 5.1 Simulation Validation

Classifier and enrichment validation operate on the real SWECRIT data, where the ground truth is unknown. Simulation validation complements them by evaluating each selection method on synthetic data with planted signals of known type, location, and strength. This makes it possible to quantify which kinds of structure each method actually detects, and at what rate it produces false positives.

Simulation validation establishes, per method, the answer to: which signal types is the method capable of detecting, at what effect size, and with what false-discovery cost. The results inform interpretation of the real-data findings: a method’s real-data selections should be read in light of the signal types it is empirically able to recover.

#### 5.1.1 Data generation

Each simulated dataset consists of  $n$  samples and  $p$  features, with a binary label  $Y$  and class prevalence  $\Pr(Y = 1)$ . Unless stated otherwise, we use

$$n = 500, \quad p = 11,000, \quad \pi = \Pr(Y = 1) = 0.15,$$

so that the simulated data resemble the size and class imbalance of the real data, as described in Section 3. In addition to this, we also run a simulation with  $n = 2500$  samples to determine if the  $n = 500$  samples limitation is a power issue or a fundamental method limitation.

We use  $\mathbf{X} \in \mathbb{R}^{n \times p}$  for the (final) data matrix, with  $n$  samples and  $p$  features. Throughout,  $i \in \{1, \dots, n\}$  indexes samples (rows) and  $j \in \{1, \dots, p\}$  indexes features (columns). We write

- $\mathbf{X}_{i,\cdot} \in \mathbb{R}^p$  for the row vector of sample  $i$ ,
- $\mathbf{X}_{\cdot,j} \in \mathbb{R}^n$  for the column vector of feature  $j$ ,
- $X_{i,j} \in \mathbb{R}$  for the scalar entry at sample  $i$ , feature  $j$ .

The corresponding random variables are denoted  $X_j$  (the random variable for feature  $j$ ) and  $Y$  (the binary class label), with realisation  $Y_i$  for sample  $i$ . Bold symbols denote vectors and matrices; light italic symbols denote scalars and scalar random variables. The base (pre-injection) matrix is denoted  $\tilde{\mathbf{X}}$ , with identical indexing conventions. Rows of  $\mathbf{X}$  are i.i.d. across samples; the within-row covariance structure across features is specified below.

The data generation consists of two steps:

1. Construct the base observation matrix  $\tilde{\mathbf{X}} \in \mathbb{R}^{n \times p}$  by drawing rows independently,

$$\tilde{\mathbf{X}}_{i,\cdot} \stackrel{\text{i.i.d.}}{\sim} \mathcal{N}(\mathbf{0}, \Sigma), \quad i = 1, \dots, n,$$

where  $\Sigma \in \mathbb{R}^{p \times p}$  is the covariance matrix. Unless stated otherwise,  $\Sigma$  is block-diagonal with blocks of size 20. Each block has ones on the diagonal and  $\rho$  on the off-diagonals, with  $\rho \sim \mathcal{U}(0.3, 0.8)$  drawn once per block. This implies  $\tilde{X}_{i,j} \sim \mathcal{N}(0, 1)$  marginally, and

$$\text{Cov}(\tilde{X}_{i,j}, \tilde{X}_{i,k}) = \rho \quad \text{if features } j, k \text{ lie in the same block,}$$

while entries from different rows remain independent. We also construct a binary label vector  $\mathbf{Y}$  by fixing the number of  $Y_i = 1$  to  $n_1 = \text{round}(n \cdot \Pr(Y = 1))$  and randomly assign  $n_1$  ones to  $\mathbf{Y}$ .

2. Randomly inject a class-dependent signal into a small number of features in a way that preserves most of the correlation structure. The result is the final observation matrix  $\mathbf{X} \in \mathbb{R}^{n \times p}$ . For noise features,  $\mathbf{X}_{:,j} := \tilde{\mathbf{X}}_{:,j}$ , while for planted features,  $\mathbf{X}_{:,j}$  is constructed from  $\tilde{\mathbf{X}}_{:,j}$  as described below.

This procedure yields a simple correlation structure into which arbitrary signals can be planted at a controlled set of features.

For each signal type, a small number of planted features (or feature pairs, in the XOR case) are included at multiple effect sizes, which we will denote  $\delta$ ; the remaining features are noise. The signal types are:

- **Linear:**  $X_j \mid Y$  is Gaussian with class-dependent mean,  $X_j \mid Y = y \sim \mathcal{N}(\mu_y, 1)$ .
- **Bounded variance:**  $X_j \mid Y$  is a clipped Gaussian with class-dependent variance,  $X_j \mid Y = y \sim \text{clip}(\mathcal{N}(0, \sigma_y), -c, c)$ .
- **U-shape:**  $X_j \mid Y = 1$  is bimodal (mixture of two Gaussians with means  $\pm a$ );  $X_j \mid Y = 0$  is a single Gaussian centred at 0.
- **Threshold:**  $X_j$  is informative for  $Y$  only above (or below) a fixed cutoff; below the cutoff the conditional distributions of the two classes coincide.
- **Pairwise XOR interaction:** a pair of features  $(X_j, X_k)$  such that  $Y$  depends on  $\text{sign}(X_j) \cdot \text{sign}(X_k)$ , while neither feature alone has any marginal association with  $Y$ .

For all signal injections below, the starting point is the base column  $\tilde{\mathbf{X}}_{:,j} \in \mathbb{R}^n$  for the planted feature  $j$  (or columns  $\tilde{\mathbf{X}}_{:,j}, \tilde{\mathbf{X}}_{:,k}$  for the XOR pair). Marginally,  $\tilde{\mathbf{X}}_{:,j} \sim \mathcal{N}(\mathbf{0}, I_n)$ , but is correlated with any given feature  $k$  in the same block, with  $\text{Cov}(\tilde{X}_j, \tilde{X}_k) = \rho$ . Each injection preserves this within-row correlation structure unless explicitly stated otherwise. Each  $\tilde{X}_{i,j}$  and  $X_{i,j}$  should be seen as observations of  $\tilde{X}_j$  and  $X_j$  respectively.

If nothing else is stated we use three features of each effect size and signal type, with effect sizes  $\{0.3, 0.6, 1\}$ . This gives a total of  $6 \times 3 \times 3 = 54$  features with planted signals, where we have counted XOR-pair as two features with planted signals. This means that the planted signals are present in approximately 0.5% ( $54/11,000$ ) of all features.

Below are detailed descriptions of how the signals are generated, followed by figures that display simulated examples of each signal type in Figure 6. Figure 5 empirically shows that the correlation structure is preserved after the injections.

**Linear:** To plant the linear signal in column  $j$ ,  $\tilde{\mathbf{X}}_{:,j}$ , we let

$$X_{i,j} = \begin{cases} \tilde{X}_{i,j} + \delta/2, & \text{if } Y_i = 1 \\ \tilde{X}_{i,j} - \delta/2, & \text{if } Y_i = 0 \end{cases}$$

and thus,

$$\begin{aligned} X_j \mid Y = 1 &\sim \mathcal{N}(\delta/2, 1) \\ X_j \mid Y = 0 &\sim \mathcal{N}(-\delta/2, 1). \end{aligned}$$

**Bounded variance:** To plant the bounded variance signal in column  $j$ ,  $\tilde{\mathbf{X}}_{:,j}$ , we let

$$X_{i,j} = \begin{cases} \text{clip}(\tilde{X}_{i,j}(1 + \delta), -c, c), & \text{if } Y_i = 1 \\ \text{clip}(\tilde{X}_{i,j}, -c, c), & \text{if } Y_i = 0 \end{cases}$$

and thus,

$$\begin{aligned} X_j \mid Y = 1 &\sim \text{clip}(\mathcal{N}(0, (1 + \delta)^2), -c, c) \\ X_j \mid Y = 0 &\sim \text{clip}(\mathcal{N}(0, 1), -c, c). \end{aligned}$$

We use  $c = 3$  if nothing else is stated.

**U-shape:** To plant the U-shaped signal in column  $j$ ,  $\tilde{X}_{\cdot,j}$ , we let  $F_1$  denote the distribution of  $\mathcal{N}(0, 1)$  and  $F_2$  the distribution of the Gaussian mixture,  $f_2(x) = \frac{1}{2}(\mathcal{N}(-a, \sigma^2) + \mathcal{N}(a, \sigma^2))$ , where we let  $\sigma = 0.5$  as this gives a fairly distinct bimodal distribution for high effect sizes  $\delta$ , and  $a = 1.5\delta$  for effect size  $\delta$ . We then let

$$X_{i,j} = \begin{cases} \frac{F_2^{-1}(F_1(\tilde{X}_{i,j}))}{\sqrt{a^2 + \sigma^2}}, & \text{if } Y_i = 1 \\ \tilde{X}_{i,j}, & \text{if } Y_i = 0 \end{cases}$$

and thus,

$$\begin{aligned} X_j | Y = 1 &\sim \frac{1}{\sqrt{a^2 + \sigma^2}} F_2 \\ X_j | Y = 0 &\sim \mathcal{N}(0, 1) \end{aligned}$$

where we have divided by  $\sqrt{a^2 + \sigma^2}$  in order to match the variance, i.e.  $\text{Var}(X_j | Y = 1) = \text{Var}(X_j | Y = 0) = 1$ .

The mixture CDF  $F_2$  has no closed-form inverse. We compute  $F_2^{-1}(u)$  scalarwise via Brent's bracketed root-finder [Bre73], as implemented in `scipy.optimize.brentq` [VGO<sup>+</sup>20], applied to  $g(x) = F_2(x) - u$  on the bracket  $[-20, 20]$  with tolerance  $10^{-12}$ . The function  $g$  is continuous and strictly monotone (it is a positively weighted sum of two normal CDFs minus a constant), so the iteration converges to the unique root within the bracket. Inputs  $u = F_1(\tilde{X}_{i,j})$  are clipped to  $[10^{-15}, 1 - 10^{-15}]$  before inversion, both to keep the root within the bracket and to avoid evaluating  $F_2^{-1}$  at the singular endpoints.

**Threshold:** The threshold signal is only active above a fixed threshold  $\tau = 0.5$ . Let  $\mathcal{A} = \{i : \tilde{X}_{i,j} > \tau\}$ , draw scores  $s_i = 2\delta Y_i + \varepsilon_i$  with  $\varepsilon_i \sim \mathcal{N}(0, 1)$  for  $i \in \mathcal{A}$ , denote by  $r_i \in \{1, \dots, |\mathcal{A}|\}$  the rank of  $s_i$  within  $\{s_k\}_{k \in \mathcal{A}}$ , and write the order statistics of  $\{\tilde{X}_{k,j}\}_{k \in \mathcal{A}}$  as  $\tilde{X}_{(1),j} \leq \dots \leq \tilde{X}_{(|\mathcal{A}|),j}$ . We then let

$$X_{i,j} = \begin{cases} \tilde{X}_{(r_i),j}, & \text{if } i \in \mathcal{A} \\ \tilde{X}_{i,j}, & \text{if } i \notin \mathcal{A}. \end{cases}$$

The rearrangement on  $\mathcal{A}$  is a permutation of the existing values, so the marginal distribution of  $X_j$  is preserved, both globally and within  $\mathcal{A}$ . The class label  $Y$  enters only through the rank coupling on  $\mathcal{A}$ : rows with  $Y = 1$  tend to receive larger order statistics. The signal vanishes at  $\delta = 0$  and strengthens monotonically with  $\delta$ , while remaining strictly localised to the upper tail  $\tilde{X}_{i,j} > \tau$ .

**XOR-pair:** The XOR-pair signal is planted in two columns  $a$  and  $b$  jointly. Column  $a$  is left unchanged,  $X_{\cdot,a} = \tilde{X}_{\cdot,a}$ . For column  $b$  we proceed in three steps. First, select the  $m = \text{block size} - 1$  columns most correlated in absolute Pearson sense with  $\tilde{X}_{\cdot,b}$ , collect them into a matrix  $P$ , and decompose

$$\tilde{X}_{\cdot,b} = P\hat{\beta} + R, \quad \hat{\beta} = (P^\top P)^{-1} P^\top \tilde{X}_{\cdot,b},$$

where  $R$  is the OLS residual. Second, define a target sign per row,

$$t_i = \begin{cases} -\text{sign}(\tilde{X}_{i,a}), & \text{if } Y_i = 1 \\ \text{sign}(\tilde{X}_{i,a}), & \text{if } Y_i = 0, \end{cases}$$

and draw  $\sigma_i = t_i$  with probability  $q = \frac{1}{2} + \frac{1}{2}\delta$  (independently across  $i$ , with  $\delta$  clipped to  $[0, 1]$ ) and  $\sigma_i = -t_i$  otherwise. Third, set

$$X_{i,b} = (P\hat{\beta})_i + \sigma_i |R_i|.$$

The fitted part  $P\hat{\beta}$  is left untouched, so the linear dependence of  $X_b$  on its block partners is preserved; the signal lives entirely in the sign of the residual. By construction  $\text{sign}(R_i) \text{sign}(\tilde{X}_{i,a})$  equals  $+1$  with probability  $p$  when  $Y_i = 0$  and  $-1$  with probability  $p$  when  $Y_i = 1$ , so the pair  $(X_a, X_b)$  carries information about  $Y$  only through the *interaction* of their signs: each marginal is uninformative.  $\delta = 0$  recovers  $p = 1/2$  (no signal),  $\delta = 1$  gives  $p = 1$  (deterministic XOR).

## Comments on data generation

The data generation described above deserves a few comments.

First, the block-diagonal correlation structure is a simplification. Real proteomic data likely contains hub proteins that correlate with many others, proteins that correlate weakly with most, and proteins that are essentially uncorrelated with the rest. We nevertheless adopt the block-diagonal structure for tractability and interpretability.

Second, several of the injected signals alter the marginal variance of  $X_j$ , which means a trivial variance filter, one that ignores  $Y$  entirely, can in principle separate them from noise. Let  $\pi = \Pr(Y = 1) = 0.15$ , chosen to approximate the observed ARDS prevalence in the SWECRIT cohort. For the linear signal, the law of total variance gives

$$\text{Var}(X_j) = \underbrace{\mathbb{E}[\text{Var}(X_j | Y)]}_{=1} + \underbrace{\text{Var}(\mathbb{E}[X_j | Y])}_{=\pi(1-\pi)\delta^2} = 1 + \pi(1-\pi)\delta^2.$$

For the bounded-variance signal, the conditional means are both zero, so

$$\text{Var}(X_j) = \mathbb{E}[\text{Var}(X_j | Y)] = (1 - \pi) + \pi(1 + \delta)^2$$

before clipping. Clipping at  $\pm c = \pm 3$  reduces this by a small amount for  $\delta \leq 0.3$  and by up to roughly 10% of the total variance at  $\delta = 1$ .

In contrast, the threshold, U-shape and XOR injections preserve the marginal variance by construction: the threshold signal rearranges existing values within  $\mathcal{A}$ , and the XOR construction leaves  $|R_i|$  unchanged and modifies only its sign.

The magnitude of the variance inflation varies. At  $\pi = 0.15$ , the linear signal yields  $\text{Var}(X_j) \approx 1 + 0.128\delta^2$ , which is small relative to the sampling variability of  $\widehat{\text{Var}}(X_j)$  at  $n = 500$  for  $\delta \leq 0.3$ . The bounded-variance signal produces a substantially larger shift. A variance filter is therefore expected to recover the bounded-variance signal with little difficulty, while linear signals at the smallest effect size sit near the noise floor. The U-shape, threshold, and XOR signals are invisible to a variance filter by construction.

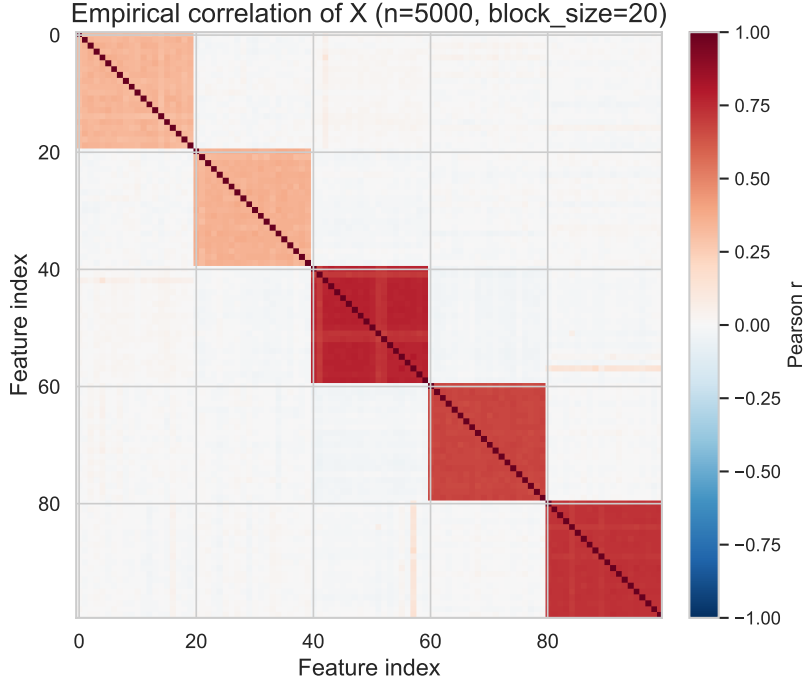


Figure 5: Empirical correlation between features using  $n = 5000$ ,  $p = 100$  and a block size of 20, showing that the most of the correlation is preserved even after signal injections.

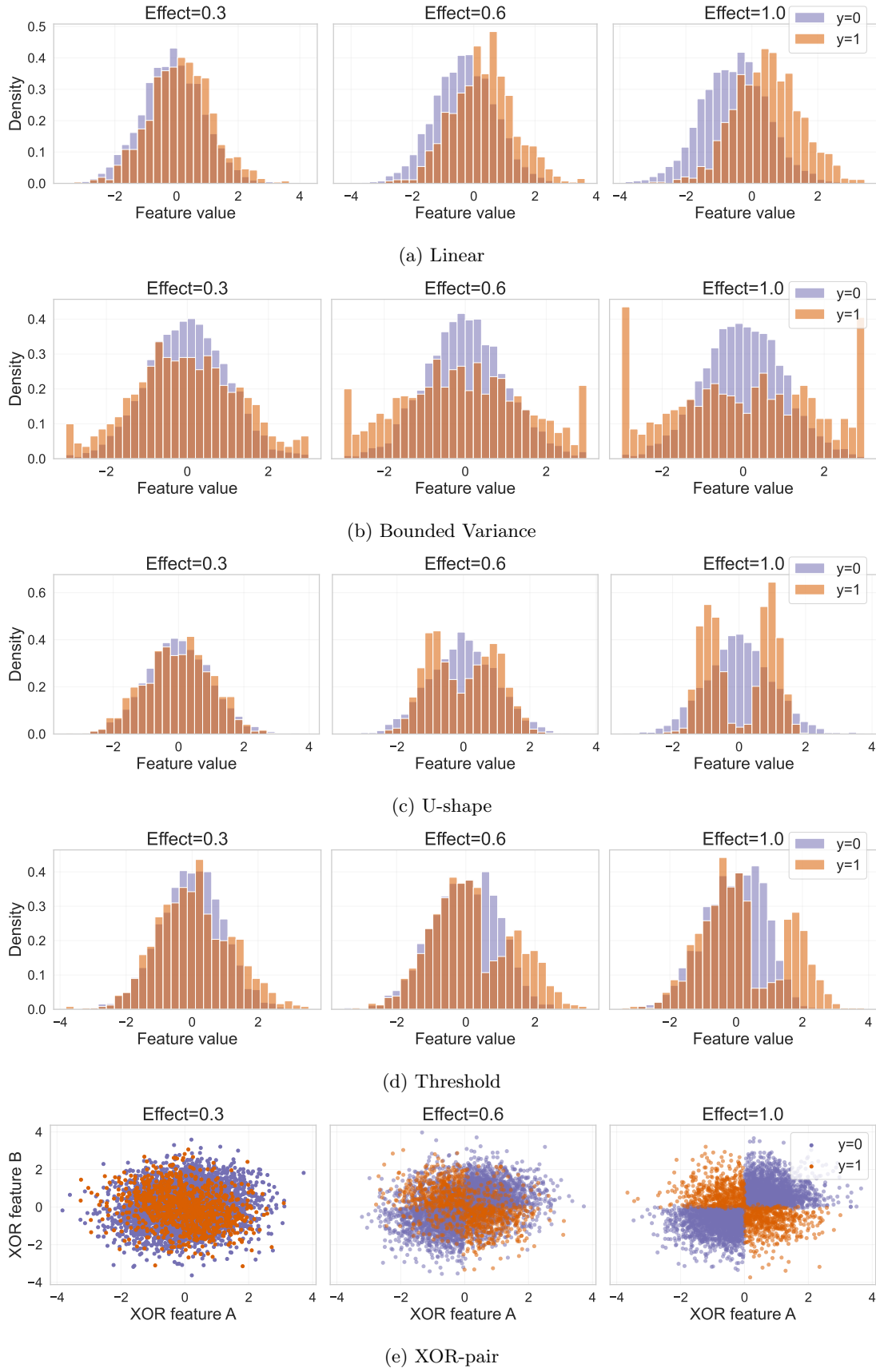


Figure 6: Simulated data ( $n = 5000$ ) showing each signal type at three effect sizes, with the class-conditional distributions for  $y = 0$  and  $y = 1$  overlaid. The XOR pair is shown as a joint distribution, since its signal is not visible marginally.

### 5.1.2 Evaluation

Each selection method is run on the simulated dataset and produces a ranked or native list of features. Performance is reported as:

- **Per-signal-type recall:** the fraction of planted features of each signal type that appear in the method’s top- $k$  and native selection. Recall, or true positive rate, is defined as [JWH<sup>+</sup>23]:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

where TP denotes true positives and FN denotes false negatives.

- **False discovery Proportion (FDP):** the fraction of selected features that are noise (i.e. neither planted nor part of a planted interaction pair). FDP is defined as:

$$\text{FDP} = \frac{\text{FP}}{\text{TP} + \text{FP}}$$

where FP denotes false positives. Usually we use the false discovery rate (FDR), defined as  $\text{FDR} = \mathbb{E}[\text{FDP}]$ , since there is no way to be certain which hypotheses are true on a real data set [JWH<sup>+</sup>23]. However, since we have simulated data we know the ground truth and can, therefore, easily calculate the FDP.

The top- $k$  cutoff matches the values used in classifier validation ( $k \in \{10, 25, 50, 100\}$ ). Each configuration (signal type, effect size) is repeated over  $R$  independently generated datasets (default  $R = 50$ ); recall and FDP are reported as mean and standard deviation across repeats. Note that given the specifications in Section 5.1.1, there are a total of 54 true signals, which means that a top- $k$  selection with  $k = 100$  will not be able to give a FDP lower than 0.46.

## 5.2 Classifier Validation

Classifier validation determines whether a classifier can differentiate patients with sepsis and ARDS from patients with sepsis only, using only the proteins in a given selected set. The reasoning is that if the selection method genuinely identifies ARDS-informative proteins, then on unseen data a classifier built on the proteins it selects from training data should discriminate sepsis+ARDS from sepsis-only.

Classifier validation establishes, per method, the answer to: does the method select ARDS informative protein. This should therefore be regarded as more of a validation of the selection method itself, rather than the actual proteins selected.

### 5.2.1 Repeated stratified random subsampling

The classifier validation procedure operates on the shared splits introduced in Section 4.3. This subsection justifies the choice of repeated stratified subsampling over alternatives and details the per-split workflow.

For the classifier we need to train a classifier on some training data and then test its performance on some held-out test data. However, due to the small sample size and class imbalance, a single train/test split is unstable. Also, to get a confidence interval of the classifier performance a single train/test split is not sufficient. Instead, the framework uses repeated stratified random subsampling, also known as Monte Carlo cross-validation (MCCV) [XL01, Sim07]. This means that we create multiple random splits of the data into training and validation, or, in this case, test, data. The advantage over standard stratified K-fold cross-validation is that the proportion of training and test split is not dependent on the number of splits, thereby, allowing for more splits while keeping the same split proportion. A disadvantage with MCCV is that the testfolds will not be disjoint, which is discussed in Section 5.2.2.

We use the following:

- **Number of splits:**  $B = 50$  (default).
- **Split ratio:** 80% train / 20% test, stratified on  $y$ .
- **Seeds:** fixed and recorded for reproducibility.

For each split  $b = 1, \dots, B$ :

1. Partition  $(X, y)$  into  $(X_b^{\text{train}}, y_b^{\text{train}})$  and  $(X_b^{\text{test}}, y_b^{\text{test}})$ .
2. Run the selection method on  $(X_b^{\text{train}}, y_b^{\text{train}})$  to obtain a selected protein set  $S_b$ .
3. Train classifiers (Section 5.2.4) on  $(X_b^{\text{train}}[:, S_b], y_b^{\text{train}})$ .
4. Evaluate on  $(X_b^{\text{test}}[:, S_b], y_b^{\text{test}})$ .

By training a predefined classifier on each resampled train set  $(X_b^{\text{train}}, y_b^{\text{train}})$  and evaluate its performance on the corresponding test set  $(X_b^{\text{test}}, y_b^{\text{test}})$ , we get a distribution of test-set performance values per classifier for each selection method.

The same split indices are reused across selection methods (seeded), so that performance differences reflect differences between methods rather than between splits. Classifier performance is reported as the mean and standard deviation of the test-fold metric across the  $B$  splits.

The full design is factorial: each selection method is paired with each classifier, so that we can examine whether a given classifier is better able to exploit the proteins selected by a given method (e.g. whether logistic regression performs comparably to random forest on  $t$ -test selections, while a non-linear classifier is required to extract signal from non-linear mutual-information selections).

### 5.2.2 Adjusting for correlated test sets

With  $B = 50$  random 80/20 splits on  $n \approx 500$  patients, every patient appears in roughly 10 test folds. Thus, per-split test metrics are likely positively correlated, so the naive standard deviation across splits underestimates the standard error of the mean generalization estimate.

Let  $a_b$  denote the test metric on split  $b$ ,  $b = 1, \dots, B$ , with  $\text{Var}(a_b) = \sigma^2$  for all  $b$  and  $\text{Corr}(a_b, a_{b'}) = \rho$  for all  $b \neq b'$ . The mean metric over the  $B$  splits is

$$\bar{a} = \frac{1}{B} \sum_{b=1}^B a_b.$$

Its variance decomposes into a diagonal and off-diagonal contribution:

$$\text{Var}(\bar{a}) = \frac{1}{B^2} \text{Var}\left(\sum_{b=1}^B a_b\right) = \frac{1}{B^2} \left[ \sum_{b=1}^B \text{Var}(a_b) + \sum_{b \neq b'} \text{Cov}(a_b, a_{b'}) \right].$$

The diagonal contains  $B$  terms each equal to  $\sigma^2$ ; the off-diagonal contains  $B(B-1)$  terms each equal to  $\rho\sigma^2$ . Substituting,

$$\text{Var}(\bar{a}) = \frac{1}{B^2} [B\sigma^2 + B(B-1)\rho\sigma^2] = \frac{\sigma^2}{B} + \frac{B-1}{B}\rho\sigma^2.$$

In the limit  $B \rightarrow \infty$  the first term vanishes but the second tends to  $\rho\sigma^2$ :

$$\text{Var}(\bar{a}) \rightarrow \rho\sigma^2 \quad \text{as } B \rightarrow \infty.$$

The variance of the mean metric therefore does not shrink as  $\sigma^2/B$ ; it is bounded below by  $\rho\sigma^2$ , which is the irreducible variance induced by the shared patients across overlapping test folds. The naive standard error  $s/\sqrt{B}$  ignores this bound and is asymptotically too small by a factor of  $\sqrt{B\rho}$  for large  $B$ .

Variance underestimation in repeated random subsampling is a known problem, and corrected estimators have been proposed to address it [NB03]. The Nadeau–Bengio correction for repeated random subsampling gives

$$\text{SE}_{\text{NB}} = s \sqrt{\frac{1}{B} + \frac{n_{\text{test}}}{n_{\text{train}}}},$$

where  $s^2$  denotes the sample variance, which for  $B = 50$  and 80/20 splits is approximately  $0.52 s$ , whereas the naive  $s/\sqrt{B} \approx 0.14 s$ . The naive estimate is roughly  $3.7\times$  too small.

In the results we report two complementary summaries per metric: (i) the sample standard deviation across the  $B$  splits, characterising split-to-split variability, and (ii) the Nadeau–Bengio standard error,

an upward-adjusted estimate that accounts for the correlation structure between splits. The correction is derived under specific assumptions, in particular an exchangeability structure on the folds, and is approximate rather than exact in our setting. Confidence intervals on the mean are constructed from the latter; the former should not be interpreted as a standard error.

### 5.2.3 Reporting cross-method results

We note that the same split indices are reused across selection method, making per-split metrics paired across methods. This means that if we want to determine if the performance of a method significantly better than that of another, we can then perform a paired test.

For the paired test we use a permutation test [Goo05, PS10a, HG18], which is both non-parametric and exact. Let  $a$  be the metric of interest, for example AUC, and let  $a_b^{(i)}$  denote the computed metric for method  $i$  and split  $b$ . For each of the  $B$  split we compute

$$d_b = a_b^{(i)} - a_b^{(ii)}, \quad b = 1, \dots, B.$$

Under the null hypothesis we have that method  $i$  and method  $ii$  perform identically, i.e. that the labels  $i$  and  $ii$  attached to the metric within each split are exchangeable.

The permutation testing scheme is as follows:

1. Compute observed  $\bar{d}_{obs} = \frac{1}{B} \sum_b d_b$
2. For  $r = 1, \dots, R$ 
  - For each split  $b$ , draw  $\epsilon_b^{(r)} \in \{-1, +1\}$  with equal probability
  - Compute  $\bar{d}^{(r)} = \frac{1}{B} \sum_b \epsilon_b^{(r)} d_b$
3. Compute permutation  $p$ -values via the two-sided test:

$$p_{\text{perm}} = \frac{1 + \#\{|\bar{d}^{(r)}| \geq |\bar{d}_{obs}|\}}{1 + R}$$

where the  $+1$  correction ensures  $p_{\text{perm}} > 0$  and guarantees  $\Pr(p_{\text{perm}} \leq \alpha) \leq \alpha$  under the null for any  $\alpha$  [PS10b]. The sign-flip is equivalent to exchanging  $(i) \leftrightarrow (ii)$  labels within a split, because if  $(i)$  and  $(ii)$  are interchangeable then  $d_b$  and  $-d_b$  are equally likely.

Under  $H_0$  of within-split exchangeability of method labels,  $d_b$  and  $-d_b$  have the same conditional distribution given the data. This invariance is per-split, so the joint distribution of  $\{d_b\}_{b=1}^B$  is invariant under independent sign flips:  $\{d_b\} \stackrel{d}{=} \{\epsilon_b d_b\}$  for any  $\epsilon_b \in \{-1, +1\}$ . The sign-flip permutation samples directly from this null distribution conditional on  $\{|d_b|\}$ . Any cross-split correlation in  $\{|d_b|\}$ , induced by overlapping test patients, is held fixed by this conditioning, so it does not need to be modelled separately.

The procedure above tests a single pair of methods. When several methods are compared, the resulting collection of  $p$ -values is subject to multiple testing: comparing  $m$  methods pairwise yields  $\binom{m}{2}$  tests (e.g.  $\binom{4}{2} = 6$  for four methods). We do not currently apply a multiple-testing correction across these pairwise comparisons and reported  $p$ -values should therefore be interpreted as unadjusted.

### 5.2.4 Classifiers

Three classifiers are evaluated: logistic regression, random forest, and XGBoost, implemented using `scikit-learn` [PVG<sup>+</sup>11] except for XGBoost which uses the `xgboost` library [CHB<sup>+</sup>26]. These are chosen to represent increasing model complexity, from a linear baseline to non-linear ensemble methods. This allows the classifier validation to reveal not only whether a selected protein set carries predictive signal, but also whether that signal is well-captured by a linear model or requires non-linear modelling to exploit.

Logistic regression serves as a linear baseline. It can only exploit proteins whose association with ARDS is approximately linear and additive; a protein whose association with ARDS is non-linear, or whose effect emerges only in combination with another protein, will generally not improve logistic regression

performance even if it is genuinely informative. Random forest and XGBoost are included precisely for this reason: both are tree-based ensemble methods capable of capturing non-linear associations and interaction effects, and an improvement over the linear baseline suggests that the selected proteins carry signal of this kind. Random forest builds an ensemble of decorrelated trees through bagging and random feature subsets; XGBoost constructs trees sequentially by correcting the residual errors of previous trees [CG16]. Together, the three classifiers span the range from simple linear models to expressive non-linear ensembles, so that any form of protein-ARDS association can in principle be exploited for prediction [JWH<sup>+</sup>23].

### 5.2.5 Class imbalance and evaluation metrics

The cohort contains substantially more non-ARDS than ARDS patients, a class imbalance of approximately 6:1. A naive classifier predicting non-ARDS for every patient would achieve roughly 85 % accuracy while detecting zero ARDS cases, making accuracy an uninformative metric. To address this at training time, logistic regression and random forest use `class_weight=‘balanced’`, which rescales the per-sample loss inversely proportional to class frequencies. XGBoost uses `scale_pos_weight =  $n_{\text{neg}}/n_{\text{pos}}$` , computed from the training fold of each cross-validation split to avoid data leakage.

Performance is evaluated using AUC and AUPR, both of which are threshold-independent summary statistics derived from the classifier’s predicted probability scores rather than from a single decision threshold [JWH<sup>+</sup>23].

**AUC:** Area under the curve (AUC), specifically the area under the receiver operating characteristic (ROC) curve, also called ROC-AUC or AUC-ROC. The ROC curve is built from the classifier’s output: a binary classifier assigns each patient a predicted probability score and classifies them as ARDS positive if the score exceeds a chosen decision threshold. As this threshold decreases, more patients are classified as positive, increasing both the true positive rate and the false positive rate. The receiver operating characteristic (ROC) curve traces this trade-off by plotting the true positive rate (TPR, also called recall or sensitivity; see Section 5.1.2) against the false positive rate

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}},$$

as the threshold varies continuously from 1 to 0. The area under the ROC curve (AUC, also referred to as ROC-AUC or AUC-ROC) summarises classifier performance across all thresholds and has a useful probabilistic interpretation: it equals the probability that a randomly chosen ARDS patient receives a higher predicted score than a randomly chosen non-ARDS patient [JWH<sup>+</sup>23]. A random classifier achieves  $\text{AUC} = 0.5$  and a perfect classifier achieves  $\text{AUC} = 1.0$ . Because the FPR is computed with respect to the true negative count, AUC is not inflated by a large majority class and remains interpretable under class imbalance.

**AUPR:** Area under the precision-recall (PR) curve, also called AUC-PR. The PR curve offers a complementary view that is more sensitive to minority-class performance. It plots precision

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

against recall as the decision threshold varies. Precision measures the fraction of patients flagged as ARDS-positive that truly have ARDS, and is therefore directly affected by class imbalance: with few positive cases it is easy to accumulate false positives, which drives precision down even when recall is high. AUPR consequently penalises a classifier that achieves high recall only at the cost of many false positives, making it a stricter and more informative indicator of minority-class performance than AUC under imbalance [SR15]. The no-skill baseline for AUPR equals the positive class prevalence ( $\approx 0.15$ ), reflecting the precision a random classifier would achieve; any useful model must exceed this value. With approximately 15 ARDS patients in each test fold, the statistical resolution of AUPR is limited and estimates should be interpreted with caution.

Together, AUC captures overall discriminative ability while AUPR emphasises performance on the minority ARDS class. Reporting both provides a more complete picture of classifier performance under class imbalance than either metric alone.

### 5.2.6 Hyperparameter configuration

Hyperparameters are fixed across all evaluated feature sets and selection methods so that performance differences reflect the selected proteins rather than classifier tuning. Principled tuning would require nested cross-validation in which protein selection is repeated on each inner fold, which is computationally infeasible for some of the selection methods considered. The classifier serves as a validation instrument for the selected protein method rather than an end product; absolute performance figures are therefore less important than relative differences across classifiers and selection methods, and differences across classifiers should be interpreted as indicators of signal type, where comparable performance between logistic regression and tree-based models suggests that the selected proteins carry primarily linear signal.

For the tree-based models, `max_depth=5` limits tree complexity to reduce overfitting given the limited number of ARDS cases, while `n_estimators=100` provides a sufficient ensemble size for stable estimates. Logistic regression is wrapped in a `StandardScaler` pipeline so that L2 regularisation penalises all protein coefficients on an equal footing, regardless of differences in feature variance. The full configurations are as follows.

#### Logistic Regression (sklearn):

- `StandardScaler` preprocessing (zero mean, unit variance)
- `class_weight`: balanced
- `solver`: lbfgs
- `max_iter`: 1000

#### Random Forest (sklearn):

- `class_weight`: balanced
- `n_estimators`: 100
- `max_depth`: 5

#### XGBoost (xgboost):

- `scale_pos_weight`:  $n_{\text{neg}}/n_{\text{pos}}$ , computed per training fold
- `n_estimators`: 100
- `max_depth`: 5
- `eval_metric`: auc

## 5.3 Enrichment Analysis Validation

To assess whether the identified protein groups reflect biologically meaningful patterns rather than statistical artifacts, pathway enrichment analysis is employed as an external validation strategy. This approach is well established in omics research [RIV<sup>+</sup>19, GLL<sup>+</sup>23, ZWL<sup>+</sup>22].

Enrichment analysis is a computational method for identifying whether a set of proteins significantly overlaps with annotated gene sets representing prior biological knowledge [HSL09]. Here, over-representation analysis (ORA) is performed using the GSEAPy Python library [FLP23], querying gene-set libraries from Enrichr [CTK<sup>+</sup>13, KJR<sup>+</sup>16], available at <https://maayanlab.cloud/Enrichr/>. Enrichr provides hundreds of curated gene-set libraries spanning pathways, biological processes, and molecular functions. Given a submitted list of proteins, ORA computes enrichment significance using Fisher’s exact test, which evaluates whether the overlap between the submitted list and a given gene set is greater than expected by chance, relative to a background set consisting of all proteins measured in the study. The  $p$ -values are adjusted for multiple comparisons using the Benjamini-Hochberg procedure [BH95]. A combined score  $c = -\log(p) \cdot \text{oddsRatio}$  is also reported, where the odds ratio quantifies the degree of overlap relative to the background [Ma’25].

### 5.3.1 A biological explainability score from enrichment results

Given a set of significantly enriched terms or pathways  $\mathcal{P}$  obtained from enrichment analysis (for example GO, Reactome, or KEGG), a biological explainability score (BES) is defined as

$$\text{BES} = \sum_{i \in \mathcal{P}} w_i e_i, \quad (1)$$

where  $e_i$  denotes the enrichment strength of term  $i$  and  $w_i$  is a weight. Here,  $w_i = 1$  for all  $i$ . In principle, the weights could be chosen to account for redundancy among terms, term specificity, or gene set coverage. However, since Fisher’s exact test already accounts for overlap size and term size when computing the  $p$ -value, a uniform weight is considered sufficient for the purposes of this study. Redundancy among terms is not eliminated, but the  $z$ -score provides a relative normalisation by comparing against a null distribution that is subject to the same redundancy structure; the correction is therefore approximate rather than exact, as discussed further in Section 5.3.2.

The enrichment strength is defined as

$$e_i = \min(-\log_{10}(q_i), c), \quad (2)$$

where  $q_i$  is the adjusted  $p$ -value of term  $i$  and  $c$  is a cap imposed to prevent extremely small  $p$ -values from dominating the score. If nothing else is stated, then  $c = 10$ .

**Natural-language interpretation.** The purpose of this score is to quantify how well a gene list can be explained by known biological functions or pathways. A gene list is considered more biologically explainable when it maps to terms that are strongly significant. The term  $\min(-\log_{10}(q_i), c)$  rewards stronger enrichment while preventing a few extremely small adjusted  $p$ -values from overwhelming the score. Consequently, a high biological explainability score indicates that the observed genes can be summarized by a relatively coherent, specific, and well-supported set of biological processes, whereas a low score suggests that the enrichment signal is weak.

**Caveat: novelty versus explainability.** BES rewards gene lists that map onto known, well-annotated biology. This is in tension with one of the aims of the present study, which is to identify novel non-linear and higher-order effects that may involve poorly annotated proteins. A selection method that recovers genuinely novel mechanisms can score lower on BES than a method that re-identifies well-known sepsis or ARDS markers, simply because the former is penalised by the structure of existing annotation databases. BES should therefore be read as a sanity check that some fraction of the selected proteins is recognisable known biology, and not as a quality measure for novelty-oriented selection methods. A low BES with high classifier performance is not by itself evidence against a method.

### 5.3.2 Permutation null and reporting

The raw value of BES is not directly interpretable: it depends on the size of the input gene list  $|G|$ , on the annotation density of the chosen library, and on structural properties of the library itself. Two methods may produce different raw BES values for reasons unrelated to biological quality. To enable comparison, BES is reported relative to a permutation null distribution computed under random gene-list selection from the assayed background.

The null is constructed as follows. Fix the input list size  $|G|$  and the background set  $B$  (the  $\sim 11,000$  assayed proteins). For  $b = 1, \dots, B_{\text{perm}}$  (with  $B_{\text{perm}} = 500$  as default):

1. Sample  $G^{(b)} \subset B$  uniformly at random with  $|G^{(b)}| = |G|$ .
2. Run the same ORA pipeline on  $G^{(b)}$  against the same library, with the same significance threshold for inclusion in  $\mathcal{P}$ , the same cap  $c$ .
3. Compute  $\text{BES}(G^{(b)})$ .

This yields a null distribution  $\{\text{BES}(G^{(b)})\}_{b=1}^{B_{\text{perm}}}$ . Because the null depends only on  $|G|$  and the library, it is computed once per  $(|G|, \text{library})$  pair and reused across selection methods; making it independent of the selection method itself and thus ensuring a fair comparison between methods. With  $B_{\text{perm}} = 500$ , the minimum attainable empirical  $p$ -value is  $1/501 \approx 0.002$ .

For each real gene list, BES is reported as three values:

- the raw score  $\text{BES}(G)$ ;
- a  $z$ -score  $z = (\text{BES}(G) - \mu_0)/\sigma_0$ , where  $\mu_0$  and  $\sigma_0$  are the mean and standard deviation of the null distribution;
- an empirical  $p$ -value  $p_{\text{emp}} = (1 + \#\{b : \text{BES}(G^{(b)}) \geq \text{BES}(G)\}) / (1 + B_{\text{perm}})$ .

The  $z$ -score is the primary value used for cross-method comparison, since it removes dependence on  $|G|$  and on library-specific baselines. It should be noted that  $z$  is a relative effect size, not a  $p$ -value, and no assumption of normality is made; inferential statements are based on  $p_{\text{emp}}$  alone. The raw BES and  $p_{\text{emp}}$  are reported alongside for completeness.

### 5.3.3 An example with biological explainability score

To showcase the raw BES,  $z$ -score and  $p_{\text{emp}}$  we consider an example. In Enrichr it is possible to retrieve a set of the top- $k$  genes that co-express with a specific gene of interest. Here, ORA is applied to the top- $k$  genes co-expressed with STAT3, as identified by ARCHS4 RNA-seq gene-gene co-expression, for  $k = 10, 20, 40$ .

Figure 7 shows the raw BES,  $z$ -score and  $p_{\text{emp}}$  for  $k = 10, 20, 40$ . The raw BES increases with  $k$ , which is expected since a larger gene list tends to produce more significantly enriched terms. However, the  $z$ -score need not increase monotonically: a larger list also yields a larger  $\sigma_0$ , so that the gain in raw BES may be offset by increased null variance, resulting in a lower  $z$ -score despite a higher raw BES. This illustrates why raw BES values are not directly comparable across lists of different sizes, and why the  $z$ -score is used for cross-method comparison.

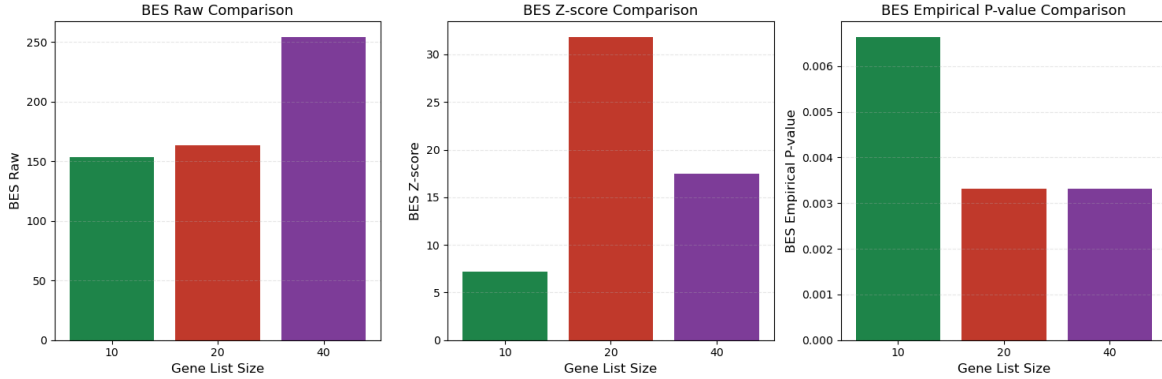


Figure 7: BES raw score,  $z$ -score, and empirical  $p$ -value ( $p_{\text{emp}}$ ) for the top  $k = 10, 20, 40$  genes co-expressed with STAT3. The raw score increases with  $k$ , while the  $z$ -score peaks at  $k = 20$  and decreases at  $k = 40$ , illustrating that raw BES values are not directly comparable across list sizes.

## 6 Random Protein Selection

We first consider a method of randomly selecting proteins. The reason we consider this is that it will allow us to see whether our downstream methods perform better than random proteins. We consider selecting a random set of proteins of different sizes, e.g. 10, 25, 50, 100. We can then see how well a classifier is able to perform using these random sets of proteins, this will act as a null baseline. We may also analyze recall and FDP in the simulated data when using random selection. Since random selection has no inherent significance threshold, we ignore the native list.

### 6.1 Results

#### 6.1.1 Simulation validation

The simulated data follow the specifications stated in Section 5.1.1. Figures 8 and 9 show recall across signal types, effect sizes, and values of  $k$ . As expected for the random selector, recall is near zero for all configurations and shows no dependence on signal type or effect size, since the selector ignores the data entirely. The small non-zero values reflect the probability of selecting a true signal protein by chance, which scales approximately as  $k/p$ , where  $p$  is the total number of proteins. For example, with  $k = 100$ ,  $p = 11,000$ , and 54 planted signal proteins, the expected recall per signal is approximately  $100/11,000 \approx 0.009$ , consistent with the values of 0.01–0.02 observed in Figure 9.

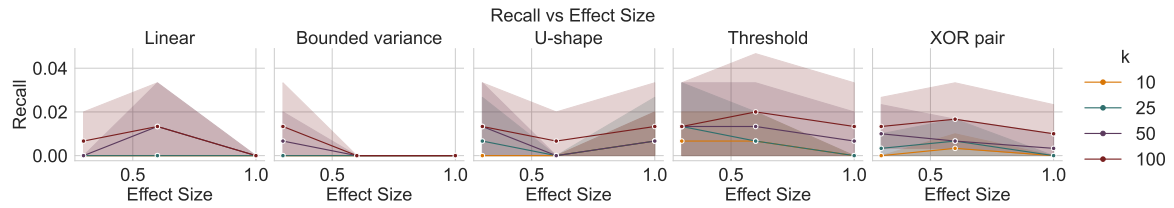


Figure 8: Recall for each signal type at different effect sizes for random selection with 95%-confidence interval. Recall remains near zero across all configurations, with no dependence on signal type or effect size.

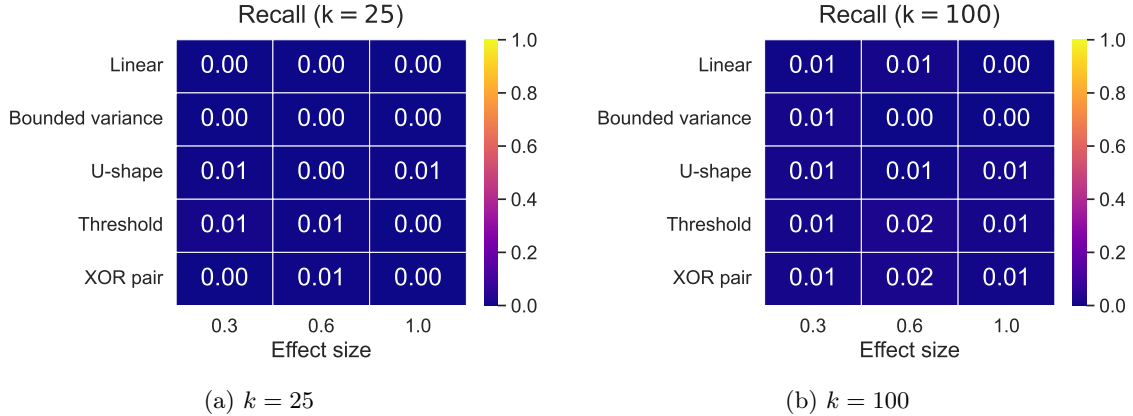


Figure 9: Recall heatmap for each signal type at different effect sizes for random selection. Here,  $k$  denotes the number of random proteins selected.

Figure 10 and Table 3 show how the FDP depends on the value of  $k$ . The FDP remains at approximately 1.0 across all values of  $k$ , since the overwhelming majority of selected proteins are noise features. This is the expected behaviour for a selector that does not use information from the data.

These results confirm that the random selector provides a lower bound on the performance any data-driven method should exceed. Any selector with non-trivial recall above  $k/p$ , or FDP meaningfully below 1.0, is by definition extracting some information from the data.

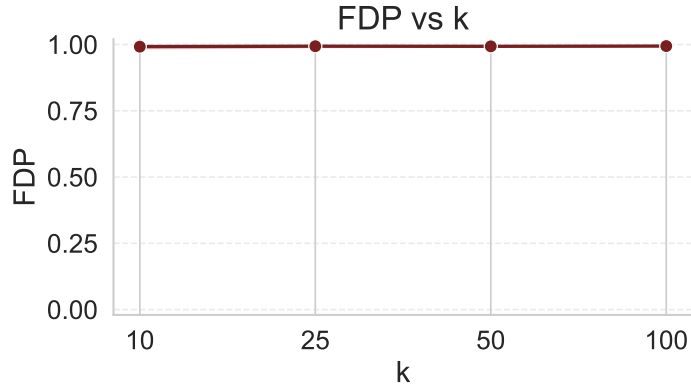


Figure 10: FDP for each  $k$  using random selection with 95%-confidence interval.

Table 3: FDP mean and standard deviation across  $k$  for random selection.

| $k$ | FDP mean | FDP std |
|-----|----------|---------|
| 10  | 0.992    | 0.027   |
| 25  | 0.994    | 0.015   |
| 50  | 0.993    | 0.011   |
| 100 | 0.994    | 0.007   |

### 6.1.2 Classifier validation

Table 4 and Figure 11 summarise classifier performance across all values of  $k$  on the real cohort data using random feature selection. AUC scores range from 0.515 to 0.604 and AUPR scores from 0.184 to 0.229. Although these values exceed the random classifier of 0.5 and the no-skill AUPR baseline ( $\approx 0.14$ ), the Nadeau–Bengio corrected standard errors are fairly large relative to the deviations from baseline. The gradual increase in AUC with  $k$  is likely attributable to the larger feature sets giving the classifier more opportunity to interpolate near the training data, rather than to any genuine signal in the selected proteins. These results confirm the role of the random selector as a lower-bound baseline: any data-driven selection method should yield classifier performance meaningfully above the levels observed here. The fact that random selection produces modestly above-chance AUC at large  $k$  illustrates that classifier metrics alone, without comparison to such a baseline, can give a misleading impression of predictive signal.

Table 4: Classifier performance across feature selection  $k$ . Bold denotes the best classifier for each  $k$  per metric. Split std is the standard deviation across cross-validation splits;  $SE_{NB}$  is the Nadeau–Bengio corrected standard error.

| Classifier                | $k$ | AUC Mean     | AUC Split Std | AUC $SE_{NB}$ | AUPR Mean    | AUPR Split Std | AUPR $SE_{NB}$ |
|---------------------------|-----|--------------|---------------|---------------|--------------|----------------|----------------|
| LogisticRegression        | 10  | 0.517        | 0.079         | 0.041         | 0.184        | 0.055          | 0.029          |
| <b>RandomForest</b>       | 10  | <b>0.524</b> | 0.067         | 0.035         | <b>0.194</b> | 0.057          | 0.029          |
| XGBoost                   | 10  | 0.515        | 0.071         | 0.037         | 0.187        | 0.049          | 0.026          |
| LogisticRegression        | 25  | 0.549        | 0.076         | 0.039         | 0.205        | 0.057          | 0.029          |
| <b>RandomForest</b>       | 25  | <b>0.553</b> | 0.084         | 0.044         | <b>0.206</b> | 0.053          | 0.028          |
| XGBoost                   | 25  | 0.524        | 0.076         | 0.039         | 0.185        | 0.038          | 0.020          |
| <b>LogisticRegression</b> | 50  | <b>0.593</b> | 0.060         | 0.031         | <b>0.220</b> | 0.049          | 0.026          |
| RandomForest              | 50  | 0.586        | 0.079         | 0.041         | 0.218        | 0.058          | 0.030          |
| XGBoost                   | 50  | 0.540        | 0.083         | 0.043         | 0.194        | 0.049          | 0.025          |
| LogisticRegression        | 100 | 0.602        | 0.071         | 0.037         | <b>0.229</b> | 0.066          | 0.034          |
| <b>RandomForest</b>       | 100 | <b>0.604</b> | 0.083         | 0.043         | 0.227        | 0.061          | 0.031          |
| XGBoost                   | 100 | 0.563        | 0.072         | 0.037         | 0.201        | 0.045          | 0.024          |

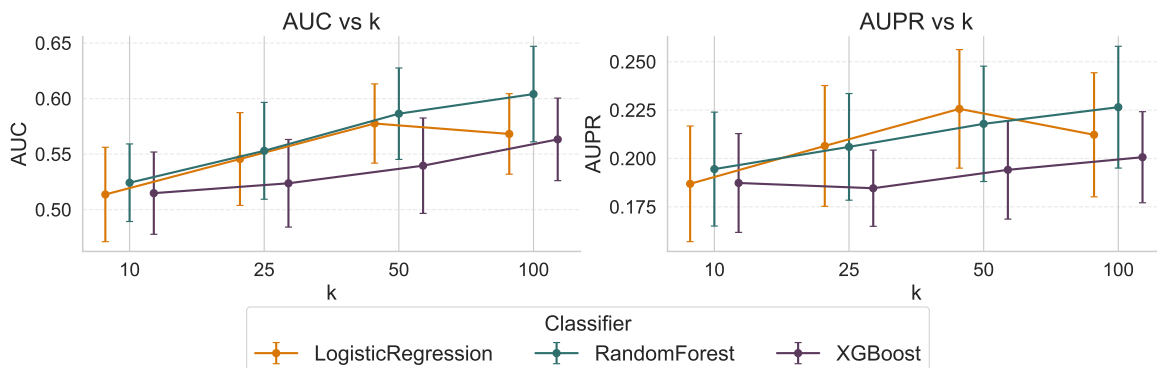


Figure 11: Mean AUC (left) and AUPR (right) as a function of  $k$  for each classifier under random feature selection, with Nadeau–Bengio corrected standard errors. Performance increases gradually with  $k$  up to  $k = 100$ .

### 6.1.3 Enrichment validation

As expected, random protein selection yields no significant enrichment in any of the three libraries, with a biological explainability score of zero and empirical  $p$ -values of 1.00 throughout, as shown in Table 5. This confirms that randomly selected proteins do not generally map onto known biological pathways, which serves as an important sanity check for the enrichment validation framework. Methods that produce meaningful enrichment results can therefore be interpreted as capturing genuine biological signal rather than noise.

Table 5: Enrichment analysis results for random protein selection ( $|G| = 20$ ). BES denotes the biological explainability score. No significant terms were identified in any library, consistent with the expectation that randomly selected proteins carry no biological signal.

| Library                    | BES Raw | BES $z$ -score | $p_{\text{emp}}$ | Significant terms |
|----------------------------|---------|----------------|------------------|-------------------|
| GO Biological Process 2025 | 0.00    | −0.24          | 1.00             | 0                 |
| KEGG 2026                  | 0.00    | −0.11          | 1.00             | 0                 |
| Reactome Pathways 2024     | 0.00    | −0.11          | 1.00             | 0                 |

## 7 Differential Expression Analysis via Welch’s $t$ -test

Differential expression analysis (DEA) is the most common approach for protein selection in clinical proteomics studies [LF19, KCTR15]. The general framework of DEA identifies proteins with significantly different abundances between two groups by testing each protein individually for a mean-level difference. In this thesis, DEA via Welch’s  $t$ -test serves as the methodological baseline against which subsequent selection methods are compared, as it represents the standard practice in the field. Its limitations are discussed in Section 7.3.

### 7.1 Welch’s $t$ -test

When comparing protein expression between two independent groups, such as ARDS and non-ARDS patients, a natural starting point is the two-sample  $t$ -test. However, in proteomics data the two groups often differ in both sample size and variance, which makes the equal-variance assumption of the standard  $t$ -test unreasonable. Welch’s  $t$ -test addresses this by estimating the variance separately for each group, making it more reliable under the conditions present in our data [Wel47]. For each protein, Welch’s  $t$ -test produces a  $p$ -value that reflects how statistically significant the observed mean difference between the two groups is. Proteins are then ranked by their adjusted  $p$ -value, or  $q$ -value, in ascending order, and the top- $k$  proteins are selected for downstream analysis. For the native selection, all proteins with a  $q$ -value below 0.05 are retained.

### 7.2 Multiple Testing Correction

When testing roughly 11,000 proteins simultaneously, the probability of obtaining false positives by chance alone becomes substantial, and multiple testing correction is therefore necessary. Rather than controlling the family-wise error rate (FWER) through conservative corrections such as Bonferroni, we apply the Benjamini-Hochberg (BH) procedure to control the false discovery rate (FDR), which offers higher power in high-dimensional settings such as ours [BH95].

### 7.3 Limitations

Although DEA via Welch’s  $t$ -test is a fast and interpretable approach, it rests on several assumptions that limit its ability to capture the full complexity of proteomic data. The test assumes that protein expression is approximately normally distributed within each group. A natural non-parametric alternative is the Mann-Whitney U-test, which makes no distributional assumptions [MW47]. However, in our data the two methods produced equivalent protein rankings, consistent with the known robustness of Welch’s  $t$ -test to moderate departures from normality at larger sample sizes [LDEC02]. We therefore proceed with Welch’s  $t$ -test, as it is more widely used in the proteomics literature and its  $p$ -values are more directly interpretable.

More fundamentally, the  $t$ -test is restricted to detecting univariate linear regulation, meaning it can only identify proteins whose mean expression level shifts monotonically between groups. Non-linear regulation patterns, such as U-shaped, threshold, or saturation effects, are outside the scope of what the test can detect. Furthermore, proteins are tested independently, which ignores the fact that proteins operate in interconnected pathways rather than in isolation. Biological effects that emerge only through combinations of proteins, analogous to XOR logic, are therefore missed entirely. These limitations motivate the use of the non-linear and higher-order methods explored in subsequent sections.

### 7.4 Results

#### 7.4.1 Simulation validation

**Samples:**  $n = 500$

The simulated data follow the specifications described in Section 5.1.1. Figure 12 shows recall as a function of effect size across all five signal types and values of  $k$ . The  $t$ -test reliably detects linear signals, recall increases steeply with effect size and approaches 1.0 for all values of  $k$  at moderate-to-large effect sizes. This is expected, as the  $t$ -test is designed precisely to detect mean-level shifts between groups, which is the defining characteristic of the linear signal type. For all non-linear signal types: bounded variance, U-shape, threshold, and XOR pair, recall remains near zero across all effect sizes and all values of  $k$ . This confirms the theoretical limitation discussed in Section 7.3: the  $t$ -test cannot detect

proteins whose association with ARDS does not manifest as a shift in mean expression.

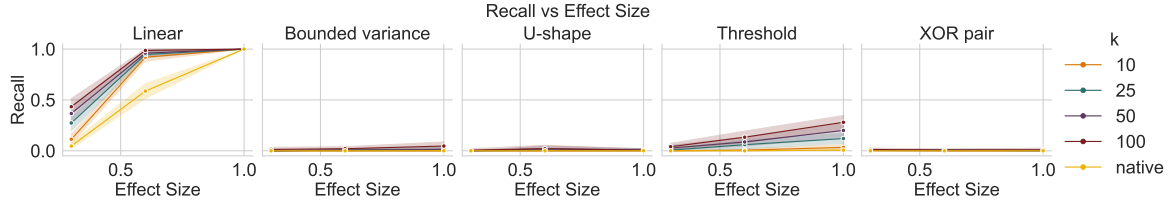


Figure 12: Recall as a function of effect size for each signal type and value of  $k$ , using  $n = 500$ . The  $t$ -test recovers linear signals reliably at moderate-to-large effect sizes but fails to detect any non-linear signal type regardless of effect size or  $k$ .

Figure 13 shows heatmaps of the mean recall across signal types and effect sizes. The native selector detects only linear signals, and only when the effect size is sufficiently large. This reflects the fact that, at smaller effect sizes, some truly linear proteins fail to reach significance and are therefore excluded at the given sample size ( $n = 500$ ). When the selector is forced to return  $k = 25$  or  $k = 100$  features, most linear signals are recovered. A small number of threshold signals are also detected, since the threshold signal produces a modest shift in the mean.

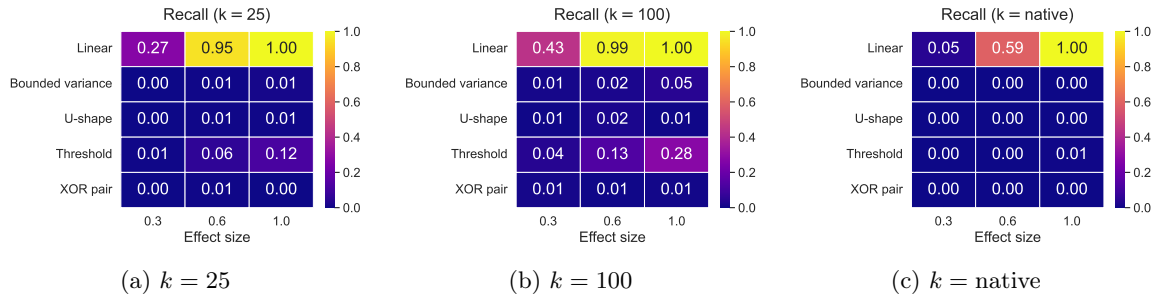


Figure 13: Recall heatmaps for the  $t$ -test across signal types and effect sizes at  $k = 25$ ,  $k = 100$ , and native selection, using  $n = 500$ . Linear signals are recovered reliably at moderate-to-large effect sizes, while non-mean-shift signals (bounded variance, U-shape, XOR pair) remain undetected regardless of  $k$ .

Figure 14 and Table 6 summarise the false discovery proportion across values of  $k$ . The FDP increases monotonically with  $k$  for fixed top- $k$  selections: at  $k = 10$  the mean FDP is 0.40, rising to 0.91 at  $k = 100$ . This reflects the fact that as more proteins are selected beyond the true linear signals, an increasing proportion are noise features that happen to rank highly by chance. The native selection achieves a dramatically lower mean FDP of 0.046 with an average of 5.22 proteins selected per simulation repeat, demonstrating that the BH correction effectively controls false discoveries when effect sizes are sufficient for proteins to reach significance. On average, 5 proteins are selected as significant. The relatively higher standard deviation ( $\pm 0.087$ ) compared to the fixed- $k$  methods reflects variability in how many proteins clear the significance threshold across repeated datasets, since the number of selected proteins varies between runs rather than being fixed. Since the numerator  $|\text{FP}|$  is a small discrete count, often zero and occasionally one or two, the distribution of FDP across repetitions is right-skewed rather than approximately Gaussian, and the reported standard deviation should not be interpreted as the half-width of a symmetric confidence interval.

Taken together, these simulation results confirm that the  $t$ -test is a reliable and precise method for detecting linear protein-ARDS associations when effect sizes are moderate to large, but is blind to all other forms of biological signals. The native selection offers substantially better precision than any fixed top- $k$  cutoff, at the cost of lower recall when effect sizes are small. These results establish the  $t$ -test as a well-characterised baseline against which subsequent methods can be compared.

**Samples:**  $n = 2500$

Increasing the sample size from  $n = 500$  to  $n = 2500$  allows a clean separation between power-limited and structurally-limited detection for the  $t$ -test. Figures 15 and 16 confirm that linear-signal detection

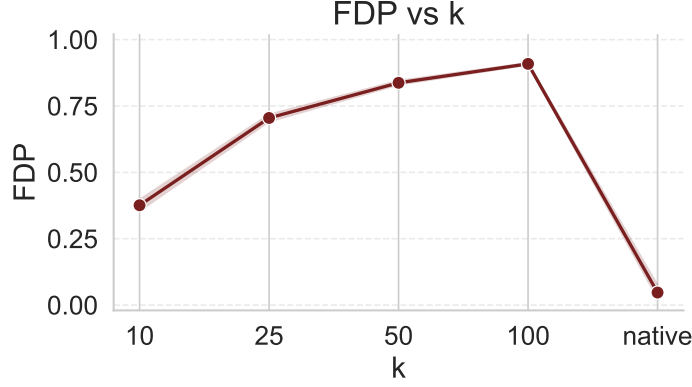


Figure 14: Mean FDP as a function of  $k$  for  $t$ -test selection. FDP increases monotonically with  $k$  for fixed top- $k$  cutoffs, while the native selection achieves a substantially lower FDP through BH-based significance thresholding. Native selects on average 5.22 proteins.

Table 6: FDP mean and standard deviation across values of  $k$  using  $n = 500$ . Native selects on average 5.22 proteins.

| $k$    | FDP mean | FDP std |
|--------|----------|---------|
| 10     | 0.376    | 0.082   |
| 25     | 0.705    | 0.050   |
| 50     | 0.837    | 0.030   |
| 100    | 0.909    | 0.017   |
| native | 0.047    | 0.091   |

essentially saturates across all  $k$  for moderate-to-large effect sizes. The previously power-limited regime at the smallest effect size is now largely recoverable under native selection, indicating that the weak performance observed at  $n = 500$  for small linear effects was sample size, not method.

The more informative result concerns the threshold signal. The incidental mean shift induced by the rank-coupling on the upper tail, which produced only partial detection at  $n = 500$ , becomes a reliable detection route at  $n = 2500$ : larger  $n$  sharpens the estimated mean within the active region  $\mathcal{A}$  and pushes the induced shift above the BH significance threshold. Given sufficient data, what was previously characterised as incidental threshold detection becomes a primary detection route.

In contrast, bounded variance, U-shape, and XOR-pair signals remain at recall near zero across all  $k$  and effect sizes. These are not power-limited failures but structural blind spots of the  $t$ -test: bounded variance produces no mean shift, U-shape is symmetric around zero in the class-conditional mean, and XOR has no marginal signal by construction. The persistence of zero recall at  $n = 2500$  confirms that no realistic sample size will allow Welch’s  $t$ -test to recover these signal types.

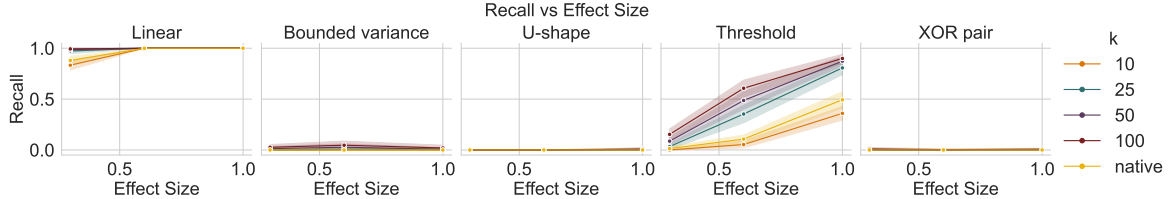


Figure 15: Recall as a function of effect size for each signal type and value of  $k$ , using  $n = 2500$ . Linear and threshold signals are for all effect sizes while threshold signals require larger effect sizes.

The FDP behaviour in Figure 17 and Table 7 show that the BH-controlled native selection remains well-calibrated at  $n = 2500$ , with the native FDP at the nominal level and the average selection size roughly doubling to 11.1 proteins. Fixed top- $k$  FDP drops sharply at small  $k$ , reflecting that with

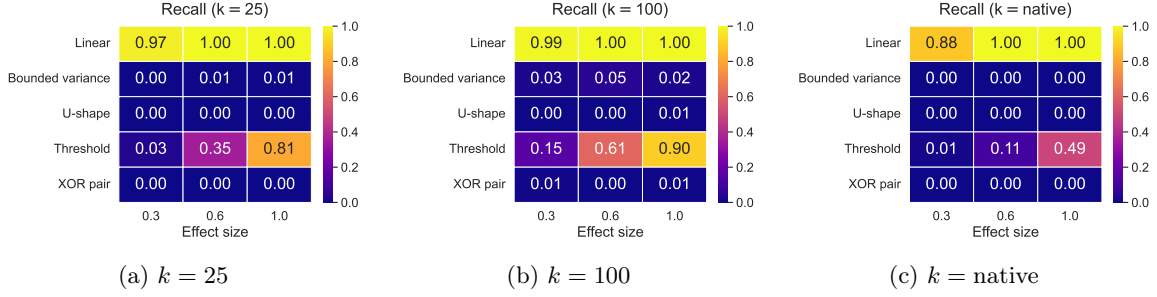


Figure 16: Recall heatmaps for the  $t$ -test across signal types and effect sizes at  $k = 25$ ,  $k = 100$ , and native selection, using  $n = 2500$  samples. Linear signals reach near-perfect recall across all effect sizes. Bounded variance, U-shape, and XOR pair signals remain undetected, reflecting the structural blind spots of the  $t$ -test.

sufficient power the top of the  $t$ -test ranking is dominated by true signals rather than noise, while the monotonic increase in FDP with larger  $k$  persists as expected.

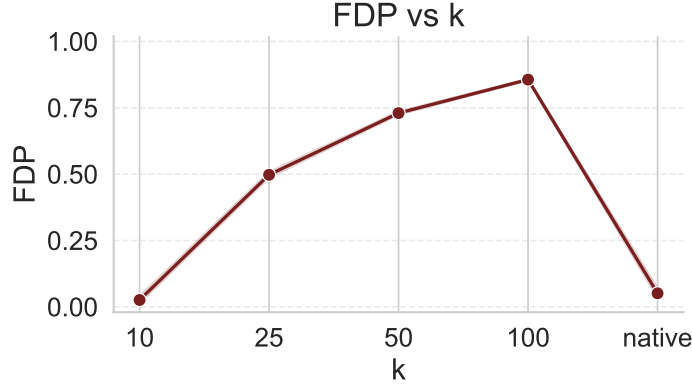


Figure 17: Mean FDP as a function of  $k$  for  $t$ -test selection, using  $n = 2500$  samples. FDP increases monotonically with  $k$  for fixed top- $k$  cutoffs, while the native selection achieves a substantially lower FDP. Native selects on average 11.1 proteins.

Table 7: FDP mean and standard deviation across values of  $k$  using  $n = 2500$ . Native selects on average 11.1 proteins.

| $k$    | FDP mean | FDP std |
|--------|----------|---------|
| 10     | 0.026    | 0.056   |
| 25     | 0.498    | 0.045   |
| 50     | 0.730    | 0.026   |
| 100    | 0.856    | 0.014   |
| native | 0.051    | 0.074   |

Taken together, the  $n = 2500$  results support the characterisation of the  $t$ -test as a reliable and well-calibrated detector of mean-shift signals, including signals that produce only a partial mean shift such as the threshold construction used here. The structural inability to detect non-mean-shift regulation is confirmed as a fundamental limitation rather than an artefact of small sample size.

#### 7.4.2 Classifier validation

Table 8 and Figure 18 summarise classifier performance across all values of  $k$  on the real cohort data. Overall, AUC scores range from 0.516 to 0.618, above the random classifier baseline of 0.5 (not to confuse with the random selector baseline), and AUPR scores range from 0.180 to 0.237, above the no-skill baseline equal to the ARDS prevalence ( $\approx 0.14$ ). Both metrics indicate modest but consistent

discriminative ability across all configurations.

Table 8: Classifier performance across feature selection  $k$ . Bold denotes the best classifier for each  $k$  per metric. Split std is the standard deviation across cross-validation splits;  $SE_{NB}$  is the Nadeau–Bengio corrected standard error.

| Classifier                | k      | AUC Mean     | AUC Split Std | AUC $SE_{NB}$ | AUPR Mean    | AUPR Split Std | AUPR $SE_{NB}$ |
|---------------------------|--------|--------------|---------------|---------------|--------------|----------------|----------------|
| <b>LogisticRegression</b> | 10     | <b>0.572</b> | 0.070         | 0.036         | <b>0.210</b> | 0.056          | 0.029          |
| RandomForest              | 10     | 0.567        | 0.070         | 0.036         | 0.199        | 0.047          | 0.024          |
| XGBoost                   | 10     | 0.538        | 0.081         | 0.042         | 0.194        | 0.055          | 0.029          |
| LogisticRegression        | 25     | 0.585        | 0.074         | 0.038         | 0.213        | 0.057          | 0.030          |
| <b>RandomForest</b>       | 25     | <b>0.587</b> | 0.072         | 0.037         | <b>0.220</b> | 0.064          | 0.033          |
| XGBoost                   | 25     | 0.571        | 0.081         | 0.042         | 0.206        | 0.059          | 0.030          |
| LogisticRegression        | 50     | 0.588        | 0.065         | 0.034         | <b>0.222</b> | 0.054          | 0.028          |
| <b>RandomForest</b>       | 50     | <b>0.592</b> | 0.070         | 0.036         | 0.216        | 0.056          | 0.029          |
| XGBoost                   | 50     | 0.580        | 0.070         | 0.036         | 0.210        | 0.056          | 0.029          |
| LogisticRegression        | 100    | 0.596        | 0.067         | 0.035         | 0.230        | 0.057          | 0.030          |
| <b>RandomForest</b>       | 100    | <b>0.618</b> | 0.072         | 0.037         | 0.231        | 0.062          | 0.032          |
| <b>XGBoost</b>            | 100    | 0.612        | 0.072         | 0.037         | <b>0.237</b> | 0.077          | 0.040          |
| <b>LogisticRegression</b> | native | <b>0.562</b> | 0.056         | 0.029         | <b>0.202</b> | 0.050          | 0.026          |
| RandomForest              | native | 0.537        | 0.067         | 0.035         | 0.187        | 0.048          | 0.025          |
| XGBoost                   | native | 0.516        | 0.070         | 0.036         | 0.180        | 0.042          | 0.022          |

Performance improves gradually as  $k$  increases from 10 to 100, with the best AUC of 0.618 achieved by random forest at  $k = 100$  and the best AUPR of 0.237 by XGBoost at the same  $k$ . This trend is visible in Figure 18 and is consistent with the simulation results: adding more top-ranked proteins increases the chance of including true linear signals, at the cost of also including more noise features as shown by the rising FDP in Table 6.

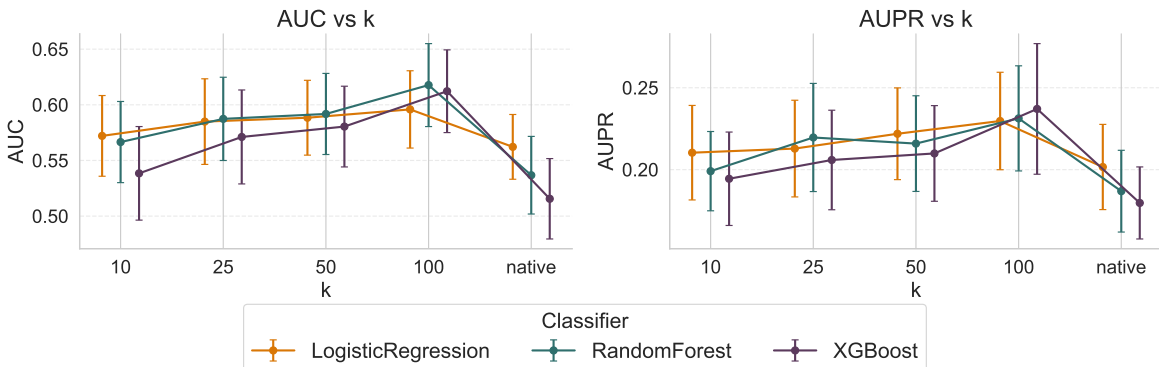


Figure 18: Mean AUC (left) and AUPR (right) as a function of  $k$  for each classifier, with Nadeau–Bengio corrected standard errors. Performance increases gradually with  $k$  up to  $k = 100$  before dropping at the native selection.

The native selection shows a notable drop in performance relative to  $k = 100$ : AUC falls to at most 0.562 and AUPR to 0.202, both achieved by logistic regression. This is in apparent tension with the low FDP of the native selection (0.046), and can be explained by the recall results from the simulation: at smaller effect sizes, true linear proteins do not reach significance and are excluded, leaving the native selection with fewer informative features overall. The trade-off between precision and recall is therefore directly visible in the classifier performance: a selection with low FDP is not necessarily a better input for a classifier if the resulting list is too small to exploit.

Across all configurations, the linear baseline (logistic regression) performs comparably to or better than the non-linear classifiers, particularly for the native selection where it outperforms both random forest and XGBoost on both metrics. This is consistent with the simulation finding that  $t$ -test selection recovers only linear signals: if the selected proteins associate with ARDS linearly, a linear classifier is sufficient to exploit them, and the added complexity of tree-based models provides no benefit.

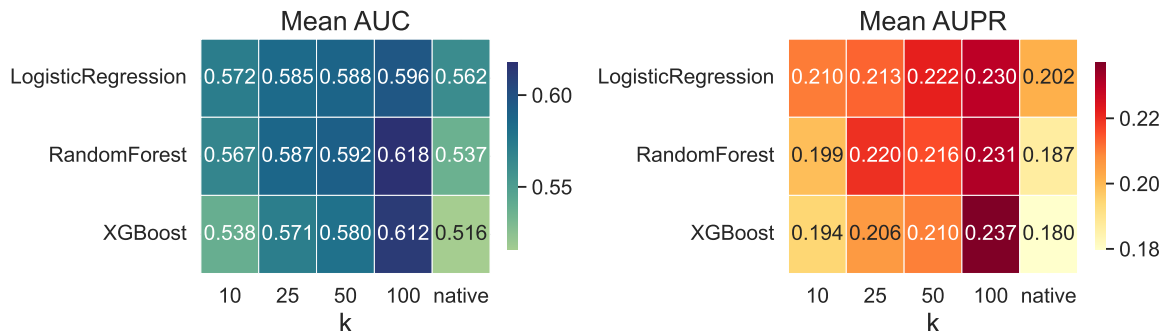


Figure 19: Heatmaps of mean AUC (left) and mean AUPR (right) across classifiers and values of  $k$ . Random forest achieves the highest AUC at  $k = 100$  (0.618) and XGBoost the highest AUPR at  $k = 100$  (0.237), but differences between classifiers are small throughout.

The heatmaps in Figure 19 confirm that differences between classifiers are small and consistent across all values of  $k$ . No classifier systematically dominates, and the variation across  $k$  is larger than the variation across classifiers at any fixed  $k$ . These results suggest that the choice of classifier matters less than the number and quality of selected features, which is the expected behaviour when the classifier is used as a validation instrument rather than as an optimised predictive model.

### 7.4.3 Stability diagnostics

Table 9 shows the selection frequency of the union of the 20 most frequently selected proteins across cross-validation splits, for each value of  $k$  and the native selection. Several patterns are worth noting.

First, native selection frequencies are generally low, with most proteins selected in fewer than 20% of splits. This reflects the joint effect of 0.8 subsampling and the BH significance threshold: small changes in the training sample are sufficient to push borderline proteins below the significance cutoff, illustrating why a single run on the full dataset may yield an unstable protein list. The higher variance of the native FDP observed in Table 6 is consistent with this instability.

Second, top- $k$  selection is considerably more stable. At  $k = 100$ , most proteins achieve frequencies above 0.70, indicating that the relative ranking of proteins is consistent across subsamples even when absolute significance varies. This suggests that the  $t$ -statistic itself is a stable signal, and that instability in the native selection is driven by the significance threshold rather than by the ranking.

Third, a small set of proteins, most notably SDHAF2, BLOC1S2, and ALDH5A1, appear consistently across all values of  $k$ , including the native selection. These proteins rank highly and clear the significance threshold regardless of which 80% subsample is used, and can be considered the most robustly supported proteins in the  $t$ -test selection. Accordingly, all three are included in the final protein list.

Finally, the final selection produced by running the  $t$ -test with native selection on the full dataset aligns closely with the most stable proteins identified through repeated subsampling. Note that GOLM1 appears twice in Table 9, reflecting the fact that SomaScan measures multiple SOMAmer reagents per protein; the two entries correspond to distinct aptamers targeting the same protein, which may differ in their binding affinity and therefore in their statistical behaviour across subsamples. Proteins included in the final list generally correspond to those with high selection frequencies already at small  $k$ , providing empirical support for the reliability of the full-data result.

### 7.4.4 Enrichment validation

The 12 proteins natively selected by the Welch's  $t$ -test are listed in Table 10, ranked by  $q$ -value. All but two proteins (GOLM1 and MAFF) show negative scores, indicating lower expression in ARDS patients relative to non-ARDS patients. SDHAF2 is the most robustly selected protein, with a resample frequency of 0.96 and the lowest  $q$ -value by a considerable margin ( $q = 0.0002$ ). Several proteins at the bottom of the list share the same  $q$ -value ( $q = 0.0339$  and  $q = 0.0478$ ), reflecting ties in the BH adjustment, and have low resample frequencies, suggesting they represent weaker or more borderline signals.

Table 9: Selection frequency of the top-20 proteins per native/ $k$  union (30 proteins total) across cross-validation splits, for each value of  $k$  and the native selection. Higher frequency indicates more consistent selection across splits. The *final* column indicates whether the protein is included in the final cohort analysis, presented in Table 10.

| Protein   | native      | k=10        | k=25        | k=50        | k=100       | Final |
|-----------|-------------|-------------|-------------|-------------|-------------|-------|
| SDHAF2    | <b>0.96</b> | <b>1.00</b> | <b>1.00</b> | <b>1.00</b> | <b>1.00</b> | ✓     |
| BLOC1S2   | 0.60        | <b>0.84</b> | <b>0.98</b> | <b>1.00</b> | <b>1.00</b> | ✓     |
| ALDH5A1   | 0.44        | 0.54        | <b>0.88</b> | <b>0.96</b> | <b>1.00</b> | ✓     |
| HSF4      | 0.32        | 0.40        | 0.58        | <b>0.90</b> | <b>0.98</b> | ✓     |
| COX5B     | 0.28        | 0.30        | 0.70        | <b>0.92</b> | <b>0.98</b> | ✓     |
| RGN       | 0.22        | 0.28        | 0.38        | 0.60        | <b>0.94</b> | ✓     |
| CDC25A    | 0.16        | 0.26        | 0.42        | 0.64        | <b>0.90</b> | ✗     |
| MAFF      | 0.16        | 0.22        | 0.54        | <b>0.78</b> | <b>0.88</b> | ✓     |
| SNTA1     | 0.20        | 0.20        | 0.56        | <b>0.84</b> | <b>1.00</b> | ✓     |
| GOLM1     | 0.14        | 0.20        | 0.52        | <b>0.76</b> | <b>0.90</b> | ✓     |
| CNN1      | 0.14        | 0.20        | 0.38        | 0.56        | <b>0.78</b> | ✗     |
| PHF11     | 0.20        | 0.20        | 0.32        | 0.46        | <b>0.76</b> | ✗     |
| MAGEA6    | 0.20        | 0.20        | 0.20        | 0.20        | 0.24        | ✗     |
| GYG2      | 0.18        | 0.18        | 0.50        | 0.70        | <b>0.96</b> | ✓     |
| ACY3      | 0.14        | 0.18        | 0.46        | 0.68        | <b>0.88</b> | ✓     |
| AGER      | 0.10        | 0.16        | 0.48        | 0.68        | <b>0.90</b> | ✗     |
| HAO1      | 0.12        | 0.16        | 0.30        | 0.48        | 0.70        | ✗     |
| USH1C     | 0.14        | 0.14        | 0.26        | 0.40        | 0.64        | ✗     |
| TSSK2     | 0.08        | 0.14        | 0.24        | 0.26        | 0.26        | ✗     |
| GOLM1     | 0.06        | 0.12        | 0.30        | 0.60        | <b>0.82</b> | ✗     |
| BPHL      | 0.10        | 0.04        | 0.30        | 0.50        | 0.74        | ✗     |
| ADAMTS10  | 0.06        | 0.08        | 0.30        | 0.50        | 0.74        | ✗     |
| RTF1      | 0.06        | 0.08        | 0.30        | 0.48        | 0.70        | ✗     |
| ACADL     | 0.14        | 0.08        | 0.28        | 0.54        | <b>0.82</b> | ✓     |
| CHRNA3    | 0.06        | 0.10        | 0.28        | 0.52        | <b>0.82</b> | ✗     |
| PIK3R1    | 0.08        | 0.06        | 0.26        | 0.50        | <b>0.76</b> | ✗     |
| STAT1     | 0.08        | 0.10        | 0.24        | 0.48        | <b>0.80</b> | ✗     |
| NOTCH2NLB | 0.10        | 0.02        | 0.14        | 0.46        | <b>0.80</b> | ✗     |
| APBB1     | 0.14        | 0.12        | 0.24        | 0.48        | <b>0.78</b> | ✗     |
| SHISA6    | 0.12        | 0.06        | 0.12        | 0.28        | 0.62        | ✗     |

Table 10: Final natively selected proteins from Welch’s  $t$ -test ( $n = 12$ , mapped symbols = 12). Proteins are ranked by  $q$ -value. Score denotes the mean difference in expression between ARDS and non-ARDS patients. Resample frequency reflects how often each protein was natively selected across the 50 resampling splits.

| Rank | Symbol  | Gene name                                                   | Score  | $q$ -value | Resample freq. |
|------|---------|-------------------------------------------------------------|--------|------------|----------------|
| 1    | SDHAF2  | Succinate dehydrogenase complex as-<br>sembly factor 2      | −0.066 | 0.0002     | 0.96           |
| 2    | BLOC1S2 | Biogenesis of lysosomal organelles com-<br>plex 1 subunit 2 | −0.058 | 0.0093     | 0.60           |
| 3    | ALDH5A1 | Aldehyde dehydrogenase 5 family mem-<br>ber A1              | −0.106 | 0.0194     | 0.44           |
| 4    | HSF4    | Heat shock transcription factor 4                           | −0.039 | 0.0211     | 0.32           |
| 5    | COX5B   | Cytochrome c oxidase subunit 5B                             | −0.039 | 0.0212     | 0.28           |
| 6    | ACY3    | Aminoacylase 3                                              | −0.112 | 0.0339     | 0.14           |
| 7    | RGN     | Regucalcin                                                  | −0.051 | 0.0339     | 0.22           |
| 8    | SNTA1   | Syntrophin alpha 1                                          | −0.041 | 0.0339     | 0.20           |
| 9    | GYG2    | Glycogenin 2                                                | −0.037 | 0.0339     | 0.18           |
| 10   | GOLM1   | Golgi membrane protein 1                                    | +0.135 | 0.0478     | 0.14           |
| 11   | MAFF    | MAF bZIP transcription factor F                             | +0.109 | 0.0478     | 0.16           |
| 12   | ACADL   | Acyl-CoA dehydrogenase long chain                           | −0.027 | 0.0478     | 0.14           |

Table 11 summarises the enrichment analysis across the three pathway libraries. Only GO Biological Process 2025 returned any significant terms; KEGG 2026 and Reactome Pathways 2024 yielded no enrichment. The overall BES is not statistically significant relative to the permutation null ( $p_{\text{emp}} = 0.194$ ), indicating that the selected protein set does not map onto a single well-defined biological process beyond two specific proteins, a negative result that is itself informative about the biological coherence of the  $t$ -test selection.

Table 11: Enrichment analysis results for the  $t$ -test selected proteins ( $|G| = 12$ ). Only GO Biological Process 2025 yielded significant terms. BES denotes the biological explainability score.

| Library                    | BES Raw | BES $z$ -score | $p_{\text{emp}}$ | Significant terms |
|----------------------------|---------|----------------|------------------|-------------------|
| GO Biological Process 2025 | 3.47    | −0.17          | 0.194            | 2                 |
| KEGG 2026                  | 0.00    | −0.20          | 1.000            | 0                 |
| Reactome Pathways 2024     | 0.00    | −0.11          | 1.000            | 0                 |

Despite the non-significant overall BES, two individual GO terms related to mitochondrial electron transport reach significance at  $q < 0.05$ , both driven by SDHAF2 and COX5B (Table 12). Both proteins show consistently negative scores, suggesting reduced mitochondrial activity in ARDS patients. The involvement of mitochondrial dysfunction in ARDS is biologically plausible: impaired cellular energy metabolism and disruption of the electron transport chain are recognised features of sepsis-induced organ failure [BBH<sup>+</sup>02]. However, with an overlap of only two proteins driving both terms and no significance at the global BES level, this finding should be interpreted with caution and would require independent replication to be considered confirmatory. Moreover, the two terms do not constitute independent biological evidence but rather reflect term redundancy in the GO annotation structure, as both are driven by the same protein pair SDHAF2 and COX5B.

Table 12: Significant GO Biological Process terms ( $q < 0.05$ ) for the  $t$ -test selected proteins. Both terms are driven by the same two proteins, SDHAF2 and COX5B, which are both involved in mitochondrial function.

| <b>Term</b>                                            | <b><math>q</math>-value</b> | <b><math>p</math>-value</b> | <b>Overlap</b> | <b>Genes</b>     |
|--------------------------------------------------------|-----------------------------|-----------------------------|----------------|------------------|
| Aerobic Electron Transport Chain (GO:0019646)          | 0.0184                      | 0.0004                      | 2/64           | SDHAF2,<br>COX5B |
| Mitochondrial ATP Synthesis Coupled Electron Transport | 0.0184                      | 0.0004                      | 2/65           | SDHAF2,<br>COX5B |

## 8 A Mutual Information Based Selection

Mutual information (MI) has been widely used in feature selection frameworks as it may measure arbitrary dependencies between random variables [VE13, Bat94]. While many of these feature selection methods build on selecting the most relevant feature, while also minimizing the redundancy, i.e., keeping features as independent as possible, in a so called Minimum Redundancy Maximum Relevance (mRMR) [PLD05], we would like to select two proteins even if they are correlated. Here, we explore a mutual information based protein selection on proteomics data that uses permutation test to determine significance.

### 8.1 Mutual Information

Mutual information is a probabilistic measure of mutual dependence between two random variables  $X$  and  $Y$  [CT06]. Mutual information is not limited linear dependencies, but is instead more general and is able to determine how different the joint distribution of  $(X, Y)$  is from the marginal distribution of  $X$  and  $Y$ .

#### 8.1.1 Entropy

Entropy can be seen as the average level of uncertainty or information associated with a random variable [CT06]. For a discrete random variable  $X$  which may take values in  $\mathcal{X}$  and is distributed according to the probability mass function  $p : \mathcal{X} \rightarrow [0, 1]$ , its entropy is defined as

$$H(X) = - \sum_{x \in \mathcal{X}} p(x) \log p(x) = \mathbb{E} \left[ \log \frac{1}{p(X)} \right].$$

For a continuous random variable  $X$ , with probability density function  $f$  with the support  $\mathcal{X}$ , we instead define the differential entropy  $h(X)$  as

$$h(X) = - \int_{\mathcal{X}} f(x) \log f(x) dx = \mathbb{E} [-\log f(x)].$$

We can similarly define the conditional entropy of  $X$  given  $Y$  as

$$H(X | Y) = - \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x, y) \log \frac{p(x, y)}{p(x)} = \mathbb{E} [-\log p(X | Y)]$$

or likewise the conditional differential entropy

$$h(X | Y) = - \int_{\mathcal{X}, \mathcal{Y}} f(x, y) \log \frac{f(x, y)}{f(x)} dx dy = \mathbb{E} [-\log f(X | Y)]$$

where  $\mathcal{X}$  and  $\mathcal{Y}$  denote the supports of  $X$  and  $Y$ .

For the conditional entropy, and conditional differential entropy, we have that  $H(X | Y) = 0$  if and only if  $X$  is completely determined by  $Y$ , while  $H(X | Y) = H(X)$  if and only if  $X$  and  $Y$  are independent.

#### 8.1.2 Mutual information

For two discrete random variables  $X$  and  $Y$  the mutual information is given as

$$I(X; Y) = \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x, y) \log \left( \frac{p(x, y)}{p(x)p(y)} \right)$$

and for two continuous random variables, we have

$$I(X; Y) = \int_{\mathcal{X}, \mathcal{Y}} f(x, y) \log \left( \frac{f(x, y)}{f(x)f(y)} \right) dx dy.$$

We can write this in terms of entropy, and differential entropy, using  $H(\cdot)$  for both, as

$$\begin{aligned} I(X; Y) &= H(X) - H(X | Y) \\ &= H(Y) - H(Y | X) \\ &= H(X) + H(Y) - H(X, Y) \end{aligned}$$

where  $H(X)$  and  $H(Y)$  are marginal entropies,  $H(X | Y)$  and  $H(Y | X)$  are conditional entropies, and  $H(X, Y)$  is the joint entropy.

Mutual information is both symmetric  $I(X; Y) = I(Y; X)$  and non-negative  $I(X; Y) \geq 0$  with equality when joint distribution coincides with the product of the marginals, i.e., when  $X$  and  $Y$  are independent.

If we want to calculate the mutual information between a continuous variable  $X$  with density  $f_X(x)$  and a discrete variable  $Y$  with mass function  $p_Y(y)$ , we can do so via [Ros14]

$$I(X; Y) = \sum_{y \in \mathcal{Y}} p_Y(y) \int_{\mathcal{X}} f_X(x | y) \log \frac{f_X(x | y)}{f_X(x)} dx. \quad (3)$$

### 8.1.3 Estimating mutual information

Since mutual information involves the unknown mass and density function, as well as integrals over these, it is not a trivial task to estimate. However, this can be solved using nonparametric methods based on entropy estimation from  $k$ -nearest neighbors distances [KSG04, Ros14, KL87], which has been implemented in `sklearn.feature_selection.mutual_info_classif` [PVG<sup>+</sup>11], which uses  $k = 3$  neighbors by default.

## 8.2 Permutation Testing to Determine Significance

While mutual information can provide a measure of dependence between two variables, we have no way of determining significance from pure mutual information values. To resolve this, we can use a permutation test [Goo05, Pit37, Ern04] to get  $p$ -values for each protein.

Let  $X \in \mathbb{R}^{n \times p}$  be a data matrix of  $n$  observations and  $p$  features, and  $y \in \{0, 1\}^n$  a binary label vector. For each feature  $j \in \{1, \dots, p\}$ , define the test statistic  $T_j = I(X_j; y)$ , an estimate of the mutual information between feature  $j$  and the label. The null hypothesis for feature  $j$  is

$$H_{0,j} : X_j \perp y,$$

i.e. the feature carries no information about the label. Under  $H_{0,j}$ , any permutation of  $y$  is equally likely given  $X_j$ . This exchangeability property is the foundation of permutation testing. The goal is to identify the set of features for which  $H_{0,j}$  can be rejected at a controlled false discovery rate (FDR), using the Benjamini–Hochberg (BH) procedure [BH95].

In a standard permutation test, we would, for each feature  $j$ , generate permuted label vectors  $y^{(1)}, y^{(2)}, \dots, y^{(B)}$  sequentially and compute permuted statistics  $T_j^{(b)} = I(X_j; y^{(b)})$ . The standard Monte Carlo  $p$ -value after  $B$  permutations is given by

$$\hat{p}_j = \frac{1 + \sum_{b=1}^B \mathbb{1}\{T_j^{(b)} \geq T_j\}}{B + 1} \quad (4)$$

where the  $+1$  correction ensures  $\hat{p}_j > 0$  and guarantees  $\Pr(\hat{p}_j \leq \alpha) \leq \alpha$  under the null for any  $\alpha$  [PS10b].

### 8.2.1 Reducing computational complexity

The smallest attainable  $p$ -value from a permutation test is  $1/(B + 1)$ . To resolve significance under multiple testing correction across  $p$  tests at level  $\alpha$ , the per-feature  $p$ -value must reach below the relevant adjusted threshold. The strictest such threshold corresponds to Bonferroni correction, which requires the  $p$ -value to fall below  $\alpha/p$ , yielding the requirement

$$B \geq p/\alpha - 1. \quad (5)$$

Under Benjamini–Hochberg (BH) correction, the threshold is data-dependent: a feature ranked  $k$ -th in ascending order of  $p$ -values can survive with a  $p$ -value up to  $\alpha k/p$ , so the required  $B$  depends on the (unknown) number of truly significant features. The Bonferroni bound is therefore sufficient but not necessary for BH-significance, and we use it as a conservative target. With  $p = 10,776$  features and  $\alpha = 0.05$ , this gives  $B \geq 215,529$  permutations per feature, which is computationally infeasible. To address this we use an adaptive permutation test as described by Besag & Clifford [BC91].

**Procedure** The adaptive stopping rule of Besag & Clifford reduces computation by terminating early for clearly null features. Fix a target hit count  $h$  (e.g.  $h = 5$ ) and a maximum number of permutations  $B_{\max}$ :

1. Set  $b \leftarrow 0$ , hits  $\leftarrow 0$ .
2. Draw  $y^{(b)}$ , compute  $T_j^{(b)}$ .
3. If  $T_j^{(b)} \geq T_j$ , increment hits.
4. If hits =  $h$ : stop, report  $\hat{p}_j = h/b$ .
5. If  $b = B_{\max}$  and hits <  $h$ : stop, report  $\hat{p}_j = (\text{hits} + 1)/(B_{\max} + 1)$ .
6. Otherwise, increment  $b$  and return to step 2.

Besag & Clifford prove that  $\hat{p}_j = h/b$  is a valid  $p$ -value under the null regardless of the data-dependent stopping rule. The primary estimator  $\hat{p}_j = h/b$  has resolution floor  $h/B_{\max}$ , achieved when the procedure runs to  $B_{\max}$  permutations and just barely reaches  $h$  hits. If fewer than  $h$  hits are observed within  $B_{\max}$  permutations, the procedure falls back to the standard Monte Carlo estimator  $\hat{p}_j = (\text{hits} + 1)/(B_{\max} + 1)$ , which has resolution floor  $1/(B_{\max} + 1)$ . The requirement

$$B_{\max} \geq \frac{h \cdot p}{\alpha} \quad (6)$$

ensures that features near the Bonferroni significance threshold can be resolved via the primary estimator rather than fall through to the fallback.

**Computational cost** For a null feature with true  $p$ -value  $\approx 0.5$ , the expected stopping time is  $2h$ . For a true positive, the procedure runs to  $B_{\max}$ . With  $p_0$  null features and  $p_1 = p - p_0$  true positives, the total number of single-feature MI evaluations is approximately

$$C \approx p_0 \cdot 2h + p_1 \cdot B_{\max}. \quad (7)$$

Thus, if we believe that the great majority of features are null features, while only a few are true positives, we can greatly reduce the complexity.

### 8.3 Advantages and Limitations

The clear advantage of the mutual information based selection described above is its ability to, in theory, discover arbitrary dependencies between a protein and ARDS. Still, the selection method described above is still univariate and will, therefore not be able to detect higher-order interactions, such as XOR-pairs. However, there exists a generalization of mutual information for more than two variables called interaction information, sometimes multivariate mutual information, first introduced in 1954 by McGill [McG54] which has been used in feature selection methods [VE14, LK13]. It should be noted that this quantity will be noisier than normal mutual information, and if mutual information does not work, it is unlikely that interaction information will provide an improvement.

Another limitation is the computational complexity required for the permutation tests. Using the adaptive permutation above will greatly reduce the required compute, as we probably will terminate early for the majority of proteins. However, if there is still a large number of proteins that are true positives, or at least close to being true positives, we might still need a large number of permutations. An alternative could be to use the generalised Pareto distribution (GPD) that provides a parametric model for the tail of the null distribution generated from permutations, allowing extrapolation beyond the empirical resolution  $1/(B + 1)$  from a much smaller number of permutations. The Pickands–Balkema–de Haan theorem [III75, BdH74] states that for a broad class of distributions  $F$ , the conditional distribution of exceedances over a threshold  $u$  converges to a GPD as  $u$  approaches the upper endpoint of  $F$ . Another alternative would be to apply  $B$  joint label permutations to all  $p$  features simultaneously; the resulting  $B \times p$  null matrix then contains  $N = Bp$  values. Then, if the null distribution after standardisation is approximately similar across features, these values can be pooled into a single reference distribution, providing empirical resolution of  $1/(Bp + 1)$  without any parametric tail model.

## 8.4 Results

### 8.4.1 Simulation validation

The simulated data follow the specifications described in Section 5.1.1, with the exception that we generate only 10 simulated datasets instead of 50.

**Samples:**  $n = 500$

Figures 20 and 21 show how recall varies with effect size, signal type, and value of  $k$  for the mutual information based selector at  $n = 500$ . The selector is able to detect arbitrary univariate signals, both linear and non-linear. However, the detection power is fairly low across all signal types, particularly for the native selector. This suggests that this method, designed to detect any signal, may perform worse on each individual signal type than a method targeted at that type alone. The low overall recall is further explained by the fact that, on average, only 4.7 features are selected. As discussed in Section 8.3, this selector cannot detect higher-order interactions such as XOR pairs, which is confirmed in the figures.

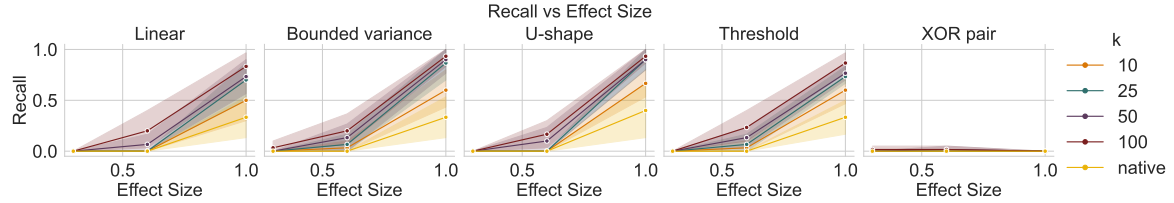


Figure 20: Recall as a function of effect size for each signal type and value of  $k$  using MI selection, with  $n = 500$  samples. Linear, bounded variance, threshold, and U-shape signals are recovered at moderate-to-large effect sizes, with recall increasing with  $k$ . XOR pair signals remain unrecovered at any effect size.

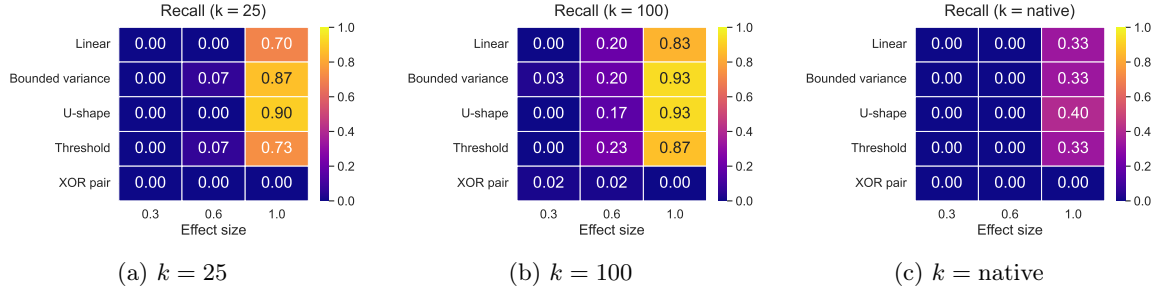


Figure 21: Recall heatmaps for MI selection across signal types and effect sizes at  $k = 25$ ,  $k = 100$ , and native selection, using  $n = 500$  samples.

Figure 22 and Table 13 summarize the FDP across values of  $k$ . As expected, the FDP increases monotonically with  $k$  for fixed top- $k$  selections, again reflecting that an increasing proportion of selected features are noise. The native selector achieves a substantially lower FDP of 0.077, with an average of 4.7 proteins selected per simulation repeat, indicating that the permutation test combined with the BH correction effectively controls false discoveries.

Table 13: FDP mean and standard deviation across values of  $k$  using  $n = 500$ . Native selects on average 4.7 proteins.

| $k$    | FDP mean | FDP std |
|--------|----------|---------|
| 10     | 0.270    | 0.116   |
| 25     | 0.600    | 0.057   |
| 50     | 0.774    | 0.039   |
| 100    | 0.866    | 0.023   |
| native | 0.077    | 0.106   |

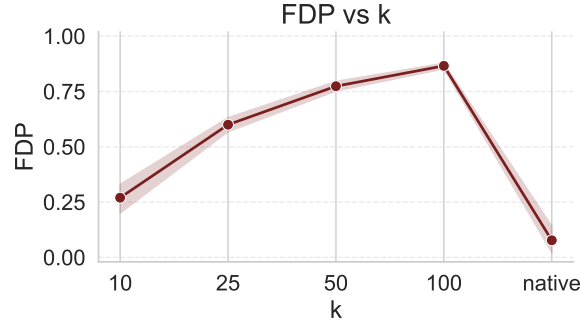


Figure 22: Mean FDP as a function of  $k$  for MI selection, with 95 % confidence intervals, using  $n = 500$  samples. FDP increases monotonically with  $k$  for fixed top- $k$  cutoffs, while the native selection achieves a substantially lower FDP. Native selects on average 4.7 proteins.

These results show that the mutual information based method with its native selector gives reliable features that are arbitrarily regulated. However, it seems that for  $n = 500$  samples, the power of the method is quite low, with fairly low recalls even for higher effect sizes.

**Samples:**  $n = 2,500$

The  $n = 2500$  results address the central question raised by the weak performance at  $n = 500$ : whether mutual information is fundamentally limited or merely power-limited under the cohort-matched sample size. Figures 23 and 24 support the latter interpretation. Linear, bounded variance, U-shape, and threshold signals are all recovered with near-perfect recall at the largest effect size across all values of  $k$  including native selection. This validates the characterization of the MI selector as capable of detecting arbitrary univariate dependencies in principle, with the limitation at  $n = 500$  attributable to the sample-hungry nature of the  $k$ -NN entropy estimators rather than to the method itself.

XOR-pair signals remain at recall near zero at  $n = 2500$ , confirming that the inability to detect higher-order interactions is structural to the univariate MI criterion rather than a power issue. This is the cleanest demonstration in the simulation results that pure interaction effects require fundamentally different selection criteria, such as the interaction information generalisation discussed in Section 8.3.

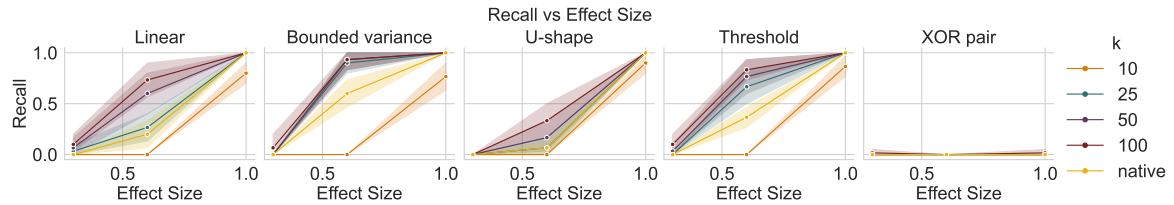


Figure 23: Recall heatmaps across signal types and effect sizes at  $k = 25$ ,  $k = 100$ , and native selection, using  $n = 2500$  samples.

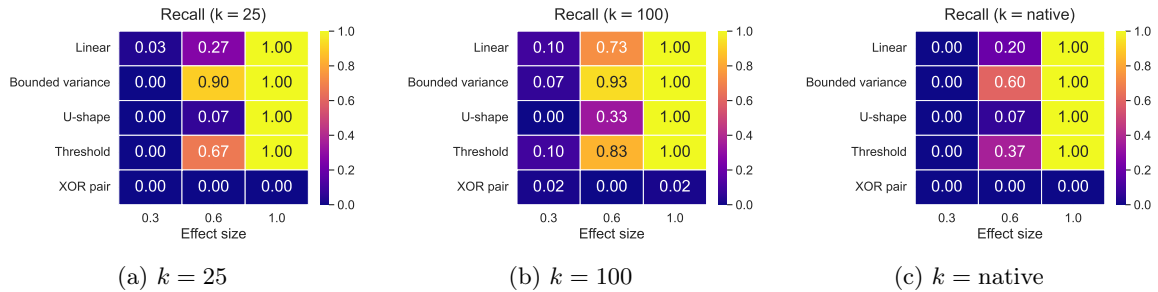


Figure 24: Recall heatmaps for different signal types and effect sizes, using  $n = 2500$  samples.

The native selection size roughly triples to 16.1 proteins on average, and the native FDP drops to well below the nominal level, Table 14. The permutation-based significance machinery is therefore

well-calibrated; it was simply unable to reach significance at  $n = 500$  for the effect sizes considered. The FDP at small  $k$  collapses to zero, indicating that with sufficient power the top of the MI ranking becomes essentially noise-free.

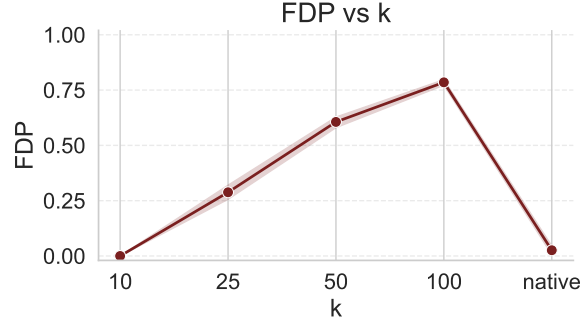


Figure 25: Mean FDP with 95% confidence interval against different  $k$  using  $n = 2500$  samples. Native selects on average 16.1 proteins.

Table 14: FDP mean and standard deviation across values of  $k$  using  $n = 2500$ . Native selects on average 16.1 proteins.

| $k$    | FDP mean | FDP std |
|--------|----------|---------|
| 10     | 0.000    | 0.000   |
| 25     | 0.288    | 0.053   |
| 50     | 0.606    | 0.040   |
| 100    | 0.785    | 0.021   |
| native | 0.026    | 0.034   |

Taken together, the  $n = 2500$  results confirm that mutual information can in principle detect arbitrary univariate signal types, but reliable detection requires both sufficient sample size and sufficient effect size. Even at  $n = 2500$ , recall at the smallest effect size remains limited across signal types, indicating that weakly regulated proteins remain difficult to detect regardless of estimator power. The method is therefore best characterized as fundamentally flexible in the type of signal it can detect, but practically constrained by the combination of sample size and effect size encountered in critical care cohorts.

#### 8.4.2 Classifier validation

As no proteins were selected by the native selector in (almost) any of the cross-validation splits, results are reported only for  $k = 10, 25, 50, 100$  in this section. Table 15 and Figure 26 summarize classifier performance across all reported values of  $k$  on the real cohort data. AUC scores range from 0.531 to 0.599 and AUPR scores from 0.193 to 0.232. These values indicate modest but consistent discriminative ability across all configurations.

Performance improves gradually with  $k$ , reaching the best AUC of 0.599 (random forest at  $k = 100$ ) and the best AUPR of 0.232 (random forest at  $k = 50$ ). The improvement from  $k = 50$  to  $k = 100$  is marginal on both metrics, suggesting that additional features beyond  $k = 50$  contribute little signal, probably due to larger top- $k$  cutoffs mostly adding noise.

Across almost all configurations, random forest performs marginally better than logistic regression, while XGBoost underperforms both. No classifier systematically dominates by a large margin, and the variation across  $k$  is comparable to or larger than the variation across classifiers at any fixed  $k$ . The fact that random forest now consistently outperforms logistic regression is consistent with the MI selector’s ability to retain non-linear signals that benefit from a non-linear classifier.

#### 8.4.3 Stability diagnostics

Table 16 shows the selection frequency of the union of the 20 most frequently selected proteins across cross-validation splits, for each value of  $k$  and the native selection, using mutual information as the ranking criterion.

Table 15: Classifier performance across feature selection  $k$ . Bold denotes the best classifier for each  $k$  per metric. Split std is the standard deviation across cross-validation splits;  $SE_{NB}$  is the Nadeau–Bengio corrected standard error.

| Classifier          | $k$ | AUC Mean     | AUC Split Std | AUC $SE_{NB}$ | AUPR Mean    | AUPR Split Std | AUPR $SE_{NB}$ |
|---------------------|-----|--------------|---------------|---------------|--------------|----------------|----------------|
| LogisticRegression  | 10  | 0.533        | 0.088         | 0.045         | 0.200        | 0.065          | 0.034          |
| <b>RandomForest</b> | 10  | <b>0.554</b> | 0.087         | 0.045         | <b>0.205</b> | 0.058          | 0.030          |
| XGBoost             | 10  | 0.531        | 0.080         | 0.041         | 0.193        | 0.051          | 0.026          |
| LogisticRegression  | 25  | 0.560        | 0.093         | 0.048         | 0.216        | 0.071          | 0.037          |
| <b>RandomForest</b> | 25  | <b>0.568</b> | 0.086         | 0.045         | <b>0.223</b> | 0.074          | 0.038          |
| XGBoost             | 25  | 0.540        | 0.083         | 0.043         | 0.202        | 0.054          | 0.028          |
| LogisticRegression  | 50  | 0.588        | 0.085         | 0.044         | 0.229        | 0.073          | 0.038          |
| <b>RandomForest</b> | 50  | <b>0.593</b> | 0.075         | 0.039         | <b>0.232</b> | 0.070          | 0.036          |
| XGBoost             | 50  | 0.546        | 0.069         | 0.036         | 0.207        | 0.054          | 0.028          |
| LogisticRegression  | 100 | 0.587        | 0.088         | 0.045         | <b>0.227</b> | 0.064          | 0.033          |
| <b>RandomForest</b> | 100 | <b>0.599</b> | 0.071         | 0.037         | 0.222        | 0.061          | 0.032          |
| XGBoost             | 100 | 0.551        | 0.071         | 0.037         | 0.204        | 0.057          | 0.030          |

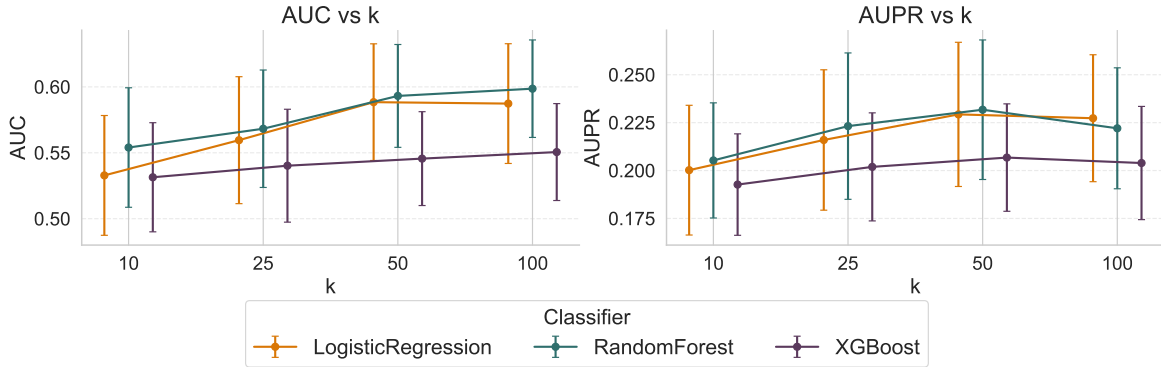


Figure 26: Mean AUC (left) and AUPR (right) as a function of  $k$  for each classifier, with Nadeau–Bengio corrected standard errors. Performance generally increases gradually with  $k$  up to  $k = 50$  before plateauing.

First, the native selector is effectively empty: only a single protein (BDH1) is selected in more than zero splits, and even then in only 2 % of splits, i.e., in 1 out of 50 splits. This is consistent with the low recall and small native selection size observed in the simulation results, and explains the absence of native results in the classifier validation.

Second, top- $k$  selection frequencies are modest. At  $k = 100$ , the most stable protein (HDGFL1) reaches a frequency of 0.80, and most proteins fall below 0.50. This indicates that the MI ranking is itself unstable across subsamples, not merely that the significance threshold is unstable.

Third, no protein is consistently selected across all values of  $k$ . The proteins that rank highly at  $k = 100$  (HDGFL1, CHCHD10, CBLN4) achieve much lower frequencies at  $k = 10$ , suggesting that the top of the MI ranking is sensitive to the specific subsample.

Finally, the table show that no proteins from MI stability table are included in the final cohort analysis, which is because no proteins were natively selected on the full data.

#### 8.4.4 Enrichment validation

Since no proteins were natively selected by the mutual information based selector, we instead consider the top- $k$  proteins on the full data with  $k = 10$ . The 10 proteins are listed in Table 17, ranked by  $q$ -values and mutual information score. None of the proteins reach significance after BH correction. The  $q$ -values are identical across all 10 proteins ( $q = 0.764$ ), reflecting ties introduced by the BH adjustment when the underlying  $p$ -values are at the resolution floor of the adaptive permutation test. As a consequence, no proteins are included in the final cohort analysis based on native selection, and the enrichment analysis below should be interpreted as exploratory rather than confirmatory.

Table 16: Selection frequency of the top-20 proteins per native/ $k$  union (30 proteins total) across cross-validation splits, for each value of  $k$  and the native selection, using mutual information (MI) as the feature ranking criterion. Higher frequency indicates more consistent selection across splits. The *final* column indicates whether the protein is included in the final cohort analysis.

| Protein | native | k=10 | k=25 | k=50 | k=100       | Final    |
|---------|--------|------|------|------|-------------|----------|
| CHCHD10 | 0.00   | 0.34 | 0.46 | 0.54 | 0.70        | $\times$ |
| KIR2DS2 | 0.00   | 0.24 | 0.46 | 0.54 | 0.60        | $\times$ |
| HMOX2   | 0.00   | 0.20 | 0.30 | 0.44 | 0.58        | $\times$ |
| CNTN4   | 0.00   | 0.20 | 0.32 | 0.46 | 0.50        | $\times$ |
| HDGFL1  | 0.00   | 0.18 | 0.40 | 0.70 | <b>0.80</b> | $\times$ |
| BDH1    | 0.02   | 0.18 | 0.26 | 0.38 | 0.50        | $\times$ |
| COPB1   | 0.00   | 0.18 | 0.24 | 0.44 | 0.46        | $\times$ |
| FKBPL   | 0.00   | 0.16 | 0.26 | 0.42 | 0.56        | $\times$ |
| OSTF1   | 0.00   | 0.16 | 0.26 | 0.36 | 0.46        | $\times$ |
| VCAN    | 0.00   | 0.16 | 0.22 | 0.34 | 0.48        | $\times$ |
| RNF24   | 0.00   | 0.14 | 0.16 | 0.26 | 0.36        | $\times$ |
| CBLN4   | 0.00   | 0.12 | 0.38 | 0.56 | 0.66        | $\times$ |
| CPNE3   | 0.00   | 0.12 | 0.22 | 0.34 | 0.46        | $\times$ |
| GOLPH3L | 0.00   | 0.12 | 0.14 | 0.28 | 0.40        | $\times$ |
| DCLK2   | 0.00   | 0.12 | 0.16 | 0.18 | 0.28        | $\times$ |
| ACSS3   | 0.00   | 0.10 | 0.18 | 0.30 | 0.50        | $\times$ |
| LCTL    | 0.00   | 0.10 | 0.14 | 0.26 | 0.42        | $\times$ |
| SCLY    | 0.00   | 0.10 | 0.24 | 0.26 | 0.46        | $\times$ |
| CADM1   | 0.00   | 0.10 | 0.16 | 0.24 | 0.30        | $\times$ |
| YME1L1  | 0.00   | 0.10 | 0.18 | 0.28 | 0.38        | $\times$ |
| ACAT1   | 0.00   | 0.08 | 0.26 | 0.42 | 0.56        | $\times$ |
| KYNU    | 0.00   | 0.10 | 0.24 | 0.30 | 0.44        | $\times$ |
| BCAP29  | 0.00   | 0.08 | 0.22 | 0.34 | 0.48        | $\times$ |
| PTK7    | 0.00   | 0.08 | 0.22 | 0.24 | 0.34        | $\times$ |
| WFDC8   | 0.00   | 0.08 | 0.20 | 0.30 | 0.34        | $\times$ |
| ZWINT   | 0.00   | 0.08 | 0.20 | 0.24 | 0.46        | $\times$ |
| DNAJC19 | 0.00   | 0.08 | 0.16 | 0.32 | 0.48        | $\times$ |
| ZNF788P | 0.00   | 0.08 | 0.18 | 0.28 | 0.46        | $\times$ |
| CAND1   | 0.00   | 0.04 | 0.12 | 0.24 | 0.36        | $\times$ |
| TMEM38B | 0.00   | 0.04 | 0.08 | 0.26 | 0.34        | $\times$ |



Figure 27: Heatmaps of mean AUC (left) and mean AUPR (right) across classifiers and values of  $k$ .

Resample frequencies are modest. CHCHD10 is the most consistently selected protein at 0.34, while the remainder fall between 0.08 and 0.20. This is consistent with the stability diagnostics in Section 8.4.3: the top of the MI ranking varies considerably across subsamples, and no single protein dominates.

Table 17: Top- $k$  selected proteins from the mutual information based selector at  $k = 10$ , used in place of a native selection as no proteins were natively selected. Proteins are ranked by  $q$ -values and mutual information score. Score denotes the estimated mutual information between protein expression and ARDS status. None of the proteins reach significance after BH correction ( $q = 0.764$  for all). Resample frequency reflects how often each protein was selected in the top- $k$  ( $k = 10$ ) across the 50 resampling splits.

| Rank | Symbol  | Gene name                                                | Score  | $q$ -value | Resample freq. |
|------|---------|----------------------------------------------------------|--------|------------|----------------|
| 1    | CHCHD10 | Coiled-coil-helix-coiled-coil-helix domain containing 10 | 0.0575 | 0.764      | 0.34           |
| 2    | BDH1    | 3-hydroxybutyrate dehydrogenase 1                        | 0.0574 | 0.764      | 0.18           |
| 3    | HDGFL1  | HDGF like 1                                              | 0.0570 | 0.764      | 0.18           |
| 4    | FKBP1   | FKBP prolyl isomerase like                               | 0.0532 | 0.764      | 0.16           |
| 5    | OSTF1   | Osteoclast stimulating factor 1                          | 0.0530 | 0.764      | 0.16           |
| 6    | ACAT1   | Acetyl-CoA acetyltransferase 1                           | 0.0530 | 0.764      | 0.08           |
| 7    | SCLY    | Selenocysteine lyase                                     | 0.0508 | 0.764      | 0.10           |
| 8    | ACSS3   | Acyl-CoA synthetase short chain family member 3          | 0.0504 | 0.764      | 0.10           |
| 9    | COPB1   | Coat protein complex I subunit beta 1                    | 0.0502 | 0.764      | 0.18           |
| 10   | HMOX2   | Heme oxygenase 2                                         | 0.0501 | 0.764      | 0.20           |

Table 18 summarises the enrichment analysis across the three pathway libraries. KEGG 2026 yields the strongest signal, with a BES  $z$ -score of 3.264 and an empirical  $p$ -value of 0.014, indicating that the enrichment is stronger than expected by chance at the global level. Reactome Pathways 2024 is borderline ( $p_{\text{emp}} = 0.056$ ), while GO Biological Process 2025 is not significant ( $p_{\text{emp}} = 0.337$ ) despite returning the largest number of individual significant terms (16). The contrast between the GO term count and the non-significant global BES illustrates that a large number of significant terms does not by itself imply enrichment beyond the permutation null.

The significant GO Biological Process terms (Table 19) are dominated by mitochondrial and metabolic processes. CHCHD10 drives several terms related to mitochondrial outer membrane permeabilization and protein localisation, HMOX2 contributes heme and porphyrin catabolism terms, and ACSS3 contributes ketone body and fatty acid derivative biosynthesis. SCLY and COPB1 each contribute a single term. All overlaps are 1-of-many, indicating that no term is supported by more than one selected protein.

The significant KEGG terms (Table 20) span selenocompound metabolism (SCLY), short-chain fatty acid metabolism (BDH1, ACSS3), and heme-related processes (HMOX2). As with the GO results, each term is driven by a single distinct protein.

The Reactome results (Table 21) show the strongest multi-protein enrichment in the analysis. Synthesis of Ketone Bodies and Ketone Body Metabolism are both highly significant ( $q < 0.0001$ ), with overlaps of 3/8 and 3/10 respectively, driven jointly by ACSS3, BDH1, and ACAT1. Mitochondrial Protein Degradation is also enriched ( $q = 0.0334$ ) via BDH1 and ACAT1.

Overall, the enrichment analysis points to mitochondrial dysfunction and altered energy metabolism, with ketone body metabolism emerging as the most robustly supported pathway. However, several caveats apply: the underlying protein selection does not clear the BH significance threshold, resample frequencies are modest, and most individual pathway terms are driven by a single protein. The findings should therefore be regarded as hypothesis-generating rather than as confirmed biology.

Table 18: Enrichment analysis results for the MI selected proteins ( $|G| = 10$ ). Here, we use the top- $k$  with  $k = 10$ , as no proteins were selected in the native selector. KEGG 2026 yielded the strongest enrichment signal. BES denotes the biological explainability score.

| Library                    | BES Raw | BES $z$ -score | $p_{\text{emp}}$ | Significant terms |
|----------------------------|---------|----------------|------------------|-------------------|
| GO Biological Process 2025 | 23.408  | 0.067          | 0.337            | 16                |
| KEGG 2026                  | 7.336   | 3.264          | 0.014            | 5                 |
| Reactome Pathways 2024     | 13.259  | 1.256          | 0.056            | 3                 |

Table 19: Significant GO Biological Process terms ( $q < 0.05$ ) for the MI selected proteins. Enrichment is driven primarily by CHCHD10 and HMOX2 (mitochondrial membrane permeabilization, heme and porphyrin catabolism) and ACSS3 (ketone body and fatty acid derivative biosynthesis), with single-protein contributions from SCLY and COPB1.

| Term                                                                                                | $q$ -value | $p$ -value | Overlap | Genes   |
|-----------------------------------------------------------------------------------------------------|------------|------------|---------|---------|
| Ketone Body Metabolic Process (GO:1902224)                                                          | 0.0297     | 0.0042     | 1/7     | ACSS3   |
| Positive Regulation of Mt Outer Membrane Permeabilization Inv in Apoptotic Sgnlng Pway (GO:1901030) | 0.0297     | 0.0052     | 1/7     | CHCHD10 |
| Heme Catabolic Process (GO:0042167)                                                                 | 0.0297     | 0.0063     | 1/10    | HMOX2   |
| Maintenance of Synapse Structure (GO:0099558)                                                       | 0.0297     | 0.0063     | 1/8     | CHCHD10 |
| Pigment Catabolic Process (GO:0046149)                                                              | 0.0297     | 0.0063     | 1/10    | HMOX2   |
| Positive Regulation of Mitochondrial Membrane Permeability (GO:0035794)                             | 0.0297     | 0.0063     | 1/9     | CHCHD10 |
| Small Molecule Biosynthetic Process (GO:0044283)                                                    | 0.0297     | 0.0063     | 1/8     | ACSS3   |
| Porphyrin-Containing Compound Catabolic Process (GO:0006787)                                        | 0.0346     | 0.0084     | 1/12    | HMOX2   |
| Regulation of Mt Outer Membrane Permeabilization Inv in Apoptotic Sgnlng Pway (GO:1901028)          | 0.0346     | 0.0094     | 1/20    | CHCHD10 |
| Cell Junction Maintenance (GO:0034331)                                                              | 0.0371     | 0.0115     | 1/17    | CHCHD10 |
| Fatty Acid Derivative Biosynthetic Process (GO:1901570)                                             | 0.0371     | 0.0126     | 1/26    | ACSS3   |
| Maintenance of Protein Location in Nucleus (GO:0051457)                                             | 0.0371     | 0.0136     | 1/20    | CHCHD10 |
| Maintenance of Protein Localization in Organelle (GO:0072595)                                       | 0.0371     | 0.0146     | 1/19    | CHCHD10 |
| Amino Acid Metabolic Process (GO:0006520)                                                           | 0.0418     | 0.0177     | 1/25    | SCLY    |
| Intra-Golgi Vesicle-Mediated Transport (GO:0006891)                                                 | 0.0451     | 0.0208     | 1/33    | COPB1   |
| Heme Metabolic Process (GO:0042168)                                                                 | 0.0451     | 0.0219     | 1/33    | HMOX2   |

Table 20: Significant KEGG 2026 terms ( $q < 0.05$ ) for the MI selected proteins. Enrichment spans selenocompound, short-chain fatty acid (butanoate, propanoate), porphyrin and mineral absorption metabolism, each driven by a single distinct protein.

| Term                      | $q$ -value | $p$ -value | Overlap | Genes |
|---------------------------|------------|------------|---------|-------|
| Selenocompound Metabolism | 0.0328     | 0.0115     | 1/17    | SCLY  |
| Butanoate Metabolism      | 0.0328     | 0.0146     | 1/27    | BDH1  |
| Mineral Absorption        | 0.0328     | 0.0208     | 1/58    | HMOX2 |
| Propanoate Metabolism     | 0.0328     | 0.0219     | 1/32    | ACSS3 |
| Porphyrin Metabolism      | 0.0398     | 0.0332     | 1/46    | HMOX2 |

Table 21: Significant Reactome Pathways 2024 terms ( $q < 0.05$ ) for the MI selected proteins. Ketone body synthesis and metabolism are strongly enriched, with full overlap (3/8 and 3/10) driven by ACSS3, BDH1, and ACAT1. Mitochondrial protein degradation is also enriched via BDH1 and ACAT1.

| Term                              | $q$ -value | $p$ -value | Overlap | Genes              |
|-----------------------------------|------------|------------|---------|--------------------|
| Synthesis of Ketone Bodies        | <0.0001    | <0.0001    | 3/8     | ACSS3, BDH1, ACAT1 |
| Ketone Body Metabolism            | <0.0001    | <0.0001    | 3/10    | ACSS3, BDH1, ACAT1 |
| Mitochondrial Protein Degradation | 0.0334     | 0.0018     | 2/97    | BDH1, ACAT1        |

## 9 Random Forest Selection through SHAP values

To address the limitations of linear and univariate analysis, a random forest (RF) classifier [Bre01] is trained on the proteomic data. Unlike the two-sample  $t$ -test, which evaluates each protein independently and is sensitive only to mean differences between groups, the random forest captures non-linear relationships and higher-order interactions between proteins during training. SHAP values are then used to extract and interpret the feature importance encoded in the fitted model, assigning each protein a contribution score for every individual prediction. This provides two complementary perspectives on protein importance: a global view, summarizing how strongly each protein contributes to the model’s predictions across the cohort, and a patient-level view, revealing how individual protein expression values contributed to each patient’s predicted ARDS probability. While the present work uses SHAP values to derive a global protein ranking, the patient-level decomposition makes it possible to extend the analysis to patient-specific interpretation in future work.

### 9.1 Random Forest and SHAP

Feature importance is extracted using SHAP (SHapley Additive exPlanations) [LL17]. While random forests natively provide importance measures such as Gini importance and permutation importance, these aggregate importance over the full dataset without decomposing contributions at the patient level. SHAP instead assigns each protein a contribution score for every individual prediction, which allows importance to be inspected per patient and per group. This patient-level decomposition provides a more nuanced and interpretable foundation enabling, for instance, group-specific analysis and future investigation of patient subgroup heterogeneity than aggregate importance measures alone.

The SHAP value  $\phi_{ij}$  for protein  $j$  and patient  $i$  is grounded in cooperative game theory: it measures the average marginal contribution of protein  $j$  to the model output, computed by considering all possible orderings in which proteins could be added to the model:

$$\phi_{ij} = \sum_{S \subseteq F \setminus \{j\}} \frac{|S|! (|F| - |S| - 1)!}{|F|!} [f(S \cup \{j\}) - f(S)] \quad (8)$$

where  $F$  is the full set of proteins and  $f(S)$  is the expected model output given only the proteins in  $S$ . The combinatorial weights ensure that each ordering is considered equally, giving protein  $j$  fair credit for its marginal contribution regardless of which other proteins are already present.

A fundamental property of SHAP values is their additivity: the sum of all SHAP values for a given patient exactly equals the difference between that patient’s predicted ARDS probability and the baseline prediction  $\mathbb{E}[f(x)]$ , which corresponds to the average predicted probability across the training cohort. `TreeExplainer` is used with default `model_output` settings, so  $f(x_i)$  corresponds to the predicted class probability for ARDS and the SHAP values are expressed on the probability scale:

$$f(x_i) = \mathbb{E}[f(x)] + \sum_{j=1}^p \phi_{ij} \quad (9)$$

This decomposition makes SHAP values directly interpretable: a positive value  $\phi_{ij} > 0$  means that protein  $j$  increased the predicted ARDS probability for patient  $i$  above the baseline, while a negative value  $\phi_{ij} < 0$  means it decreased it. The magnitude reflects the strength of the contribution.

#### 9.1.1 TreeSHAP and feature perturbation

For tree-based models, SHAP values can be computed exactly and efficiently using the TreeSHAP algorithm [LEC<sup>+</sup>20], which exploits the recursive structure of decision trees to avoid the exponential complexity of the exact Shapley computation. TreeSHAP is used here with `feature_perturbation="tree_path_dependent"` which uses the conditional distributions encoded in the tree structure rather than assuming proteins are independent when computing expected model outputs. This is preferred in our high-dimensional proteomics data where  $p \gg n$  and proteins are highly correlated, as the marginal distribution poorly approximates the true conditional. A consequence is that SHAP values may capture correlation-driven effects: a protein correlated with a true ARDS driver

may receive a non-zero SHAP value without direct causal involvement, as the model has learned to use it as a proxy.

## 9.2 Hyperparameters

The random forest is configured with 10,000 trees (`n_estimators`), following [WS18], substantially more than conventional defaults, to reduce variance in the importance estimates. This effect extends to SHAP-based importance scores, which are computed post-hoc from the fitted ensemble. An empirical stability analysis confirmed this for the present data: with 500 trees, the overlap in the top-25 ranked proteins between independent runs with different random seeds was approximately 43%, whereas with 10,000 trees this increased to approximately 80%, justifying the larger ensemble despite the additional computational cost, noting that 80 % overlap still implies that approximately one in five proteins may differ between independent runs. At each split, the number of candidate features considered is  $\sqrt{p} \approx 104$  (`max_features="sqrt"`), which is the standard default for classification and ensures that individual trees are sufficiently decorrelated. To handle the class imbalance between ARDS and non-ARDS patients, `class_weight="balanced"` is used as hyperparameter, which weights each class inversely proportional to its frequency in the training data. The random state is fixed to the split index to ensure reproducibility while allowing variation across splits.

## 9.3 Selection

### 9.3.1 Top- $k$ selection

For a fixed  $k$ , proteins are ranked by their mean absolute SHAP value across all patients [LL17], which summarizes each protein’s average contribution to the model’s predictions regardless of direction:

$$s_j = \frac{1}{n} \sum_{i=1}^n |\phi_{ij}|. \quad (10)$$

The top- $k$  proteins by  $s_j$  are selected. In this work,  $k \in \{10, 25, 50, 100\}$  is considered, following the evaluation protocol defined in Section 5.

Top- $k$  selection does not produce a significance mask: it returns a fixed-size set without any statistical claim about which proteins are genuinely associated with ARDS. It is applied independently on each training split for classifier and stability validation, but is not used to produce the final reported protein set; the latter is obtained via the stability-based native selection described below.

### 9.3.2 Native selection via stability selection

Unlike the  $t$ -test and mutual information, RF+SHAP does not produce a per-split significance mask: the SHAP score  $s_j$  is a continuous importance measure with no built-in cutoff, and permutation-based significance testing is computationally infeasible given the size of the ensemble (see Section 9.4). The native selection for RF+SHAP is therefore obtained via stability selection on the full cohort, following the procedure described in Section 4.5.1.

This differs from the  $t$ -test and mutual information selectors in two ways. First, those methods produce a native selection on every training split, whereas the RF+SHAP native set exists only at the full-cohort level. Second, the stability selection used here functions as the primary significance mechanism for RF+SHAP, rather than as a diagnostic of an existing native selector.

The procedure requires two parameters: a candidate pool size  $k_0$  that defines what counts as “selected” on each split (the top- $k_0$  proteins by  $s_j$ ), and a stability threshold  $\tau$  that defines what counts as “stable” across splits (a protein must appear in the top- $k_0$  in at least a fraction  $\tau$  of the  $B$  splits). A higher  $\tau$  requires more consistent selection and reduces false discoveries; a larger  $k_0$  admits more candidates and increases sensitivity at the cost of higher false discoveries. The values of  $k_0$  and  $\tau$  used here are reported with the results in Section 9.5.

## 9.4 Advantages and Limitations

The random forest and SHAP approach captures non-linear relationships between proteins and ARDS, and provides both global and patient-level importance scores in a theoretically principled way. The mean absolute SHAP value  $s_j$  is a well-established and interpretable summary of feature importance that is directly comparable across proteins and patients.

However, several limitations warrant consideration. Although the Random Forest internally exploits combinations of proteins at each split, SHAP attribution marginalises over these interactions and assigns each protein a single additive contribution to the prediction. A protein which is only informative in combination with another may therefore receive a low SHAP score even if the interaction is biologically meaningful. This is an inherent limitation of additive feature attribution methods and is shared by all SHAP-based approaches.

A further limitation concerns the tree path dependent perturbation mode, which means that SHAP values reflect the conditional dependencies encoded in the random forest rather than true causal effects. A protein that is highly correlated with a genuine ARDS driver may receive a high  $s_j$  score without being directly involved in the disease process. More broadly, SHAP values measure the importance of a protein to the model’s predictions, not its role in the underlying biology, a model is not necessarily a faithful representation of reality, and conclusions drawn from SHAP values should not extend beyond what the model has learned from the available data. Causal interpretation therefore requires external biological validation, as provided by the enrichment analysis described in Section 5.3.

Finally, the significance of the identified protein sets is not formally tested through permutation testing, as this would require retraining the random forest across a large number of permuted datasets, which is computationally prohibitive given the size of the ensemble. The protein sets are instead once again evaluated through the validation framework described in Section 5.

## 9.5 Results

### 9.5.1 Simulation validation

**Samples:**  $n = 500$

As shown in Figures 28 and 32, RF+SHAP reliably recovers monotonic signals: linear, bounded variance, and threshold at moderate to large effect sizes, with recall improving substantially with increasing  $k$ . At small effect sizes, recall is near zero across all signal types. U-shape signals are somewhat detected, while XOR-pairs are not recovered at any effect size or value of  $k$ , consistent with the limitations of additive SHAP attribution discussed in Section 9.4.

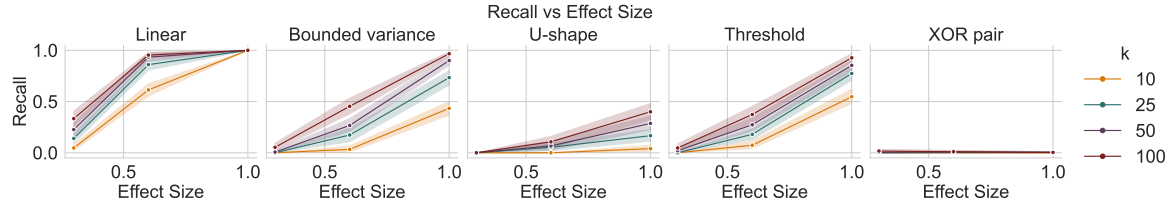


Figure 28: Recall as a function of effect size for each signal type and value of  $k$ , for the RF+SHAP selection method. Shaded regions denote 95% confidence intervals across  $R = 50$  independent simulation repeats. RF+SHAP recovers linear, bounded variance, and threshold signals at large effect sizes, but does not recover U-shape or XOR pair signals at any effect size.

Figure 30 shows that FDP increases monotonically with  $k$ , from approximately 0.18 at  $k = 10$  to 0.84 at  $k = 100$ , confirming the expected sensitivity–specificity trade-off. Results across all four values of  $k$  are therefore reported and assessed through classifier validation.

Table 22: FDP mean and standard deviation across values of  $k$  using  $n = 500$  for RF-SHAP feature selection.

| $k$ | FDP mean | FDP std |
|-----|----------|---------|
| 10  | 0.164    | 0.106   |
| 25  | 0.509    | 0.064   |
| 50  | 0.708    | 0.036   |
| 100 | 0.830    | 0.021   |

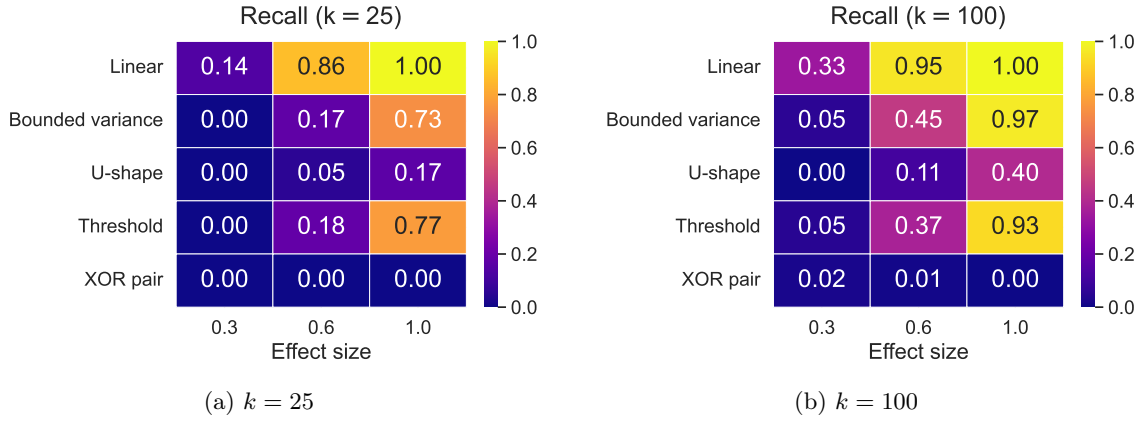


Figure 29: Mean recall per signal type and effect size for RF+SHAP, at  $k = 25$  (left) and  $k = 100$  (right). Larger  $k$  improves recall for linear, bounded variance, and threshold signals, particularly at intermediate effect sizes. U-shape and XOR pair signals remain unrecovered at both values of  $k$ .

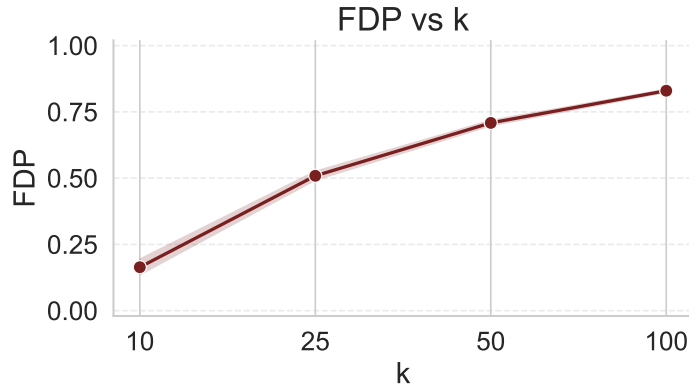


Figure 30: False discovery proportion (FDP) as a function of  $k$  for the RF+SHAP selection method, averaged across  $R = 50$  simulation repeats with 95% confidence intervals.

**Samples:**  $n = 2500$

The  $n = 2500$  results clarify which of the limitations observed at  $n = 500$  are structural and which are power-related. Linear, bounded variance, and threshold signals all reach near-perfect recall at moderate-to-large effect sizes across all values of  $k$ , extending the strong  $n = 500$  performance to weaker effects. U-shape recall also improves substantially with larger  $n$ , approaching the levels seen for the monotonic signal types at the largest effect size, indicating that the weak U-shape performance at  $n = 500$  was at least partly a power issue rather than a fundamental limitation. Direct comparison across signal types should however be made cautiously, as effect sizes are not strictly comparable between the different injection schemes.

XOR-pair signals remain at recall near zero across all  $k$  and effect sizes. As discussed in section 9.4, this is not a power limitation but a consequence of SHAP being additive by construction: even when the random forest exploits the interaction internally, the attribution step marginalizes over it and assigns each protein in the pair a near-zero importance score. The persistence of zero recall at  $n = 2500$  confirms that this is a structural ceiling of additive feature attribution rather than an artifact of sample size.

The FDP behavior, Figure 33 and Table 23, shows substantial improvement at small  $k$ , dropping to near zero for  $k \leq 25$ . With sufficient power, the top of the RF+SHAP ranking becomes essentially pure signal, while the monotonic increase in FDP at larger  $k$  persists as expected.

Taken together, the  $n = 2500$  results characterise RF+SHAP as able to detect any univariate signal type given sufficient sample and effect size, with the structural inability to detect higher-order interaction

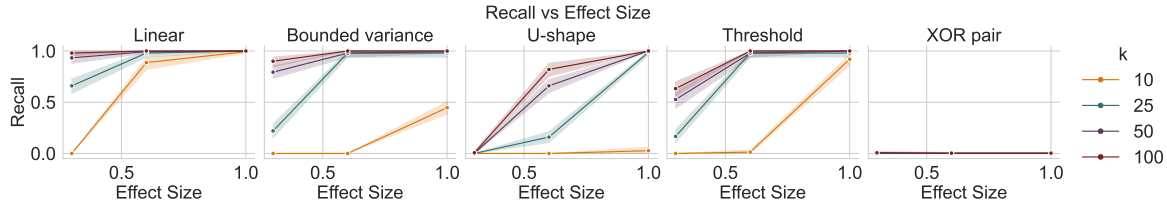


Figure 31: Recall as a function of effect size for each signal type and value of  $k$ , using  $n = 2500$ , for the RF+SHAP selection method. Shaded regions denote 95% confidence intervals across  $R = 50$  independent simulation repeats.

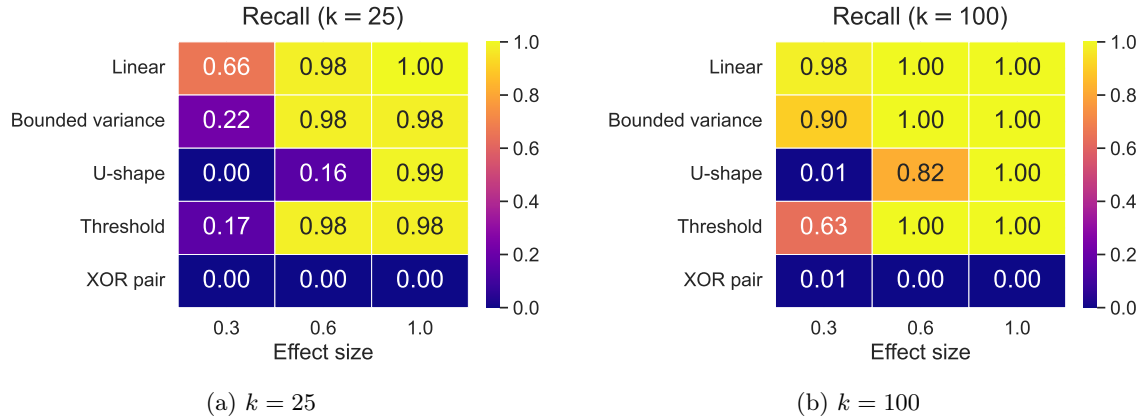


Figure 32: Mean recall per signal type and effect size for RF+SHAP with  $n = 2500$  samples, at  $k = 25$  (left) and  $k = 100$  (right).

effects confirmed as an inherent property of additive SHAP attribution.

### 9.5.2 Classifier validation

Table 24 and Figures 34 and 35 summarise classifier performance across all values of  $k$  for the RF+SHAP selected proteins. The random forest classifier consistently achieves the highest AUC and AUPR across all values of  $k$ . While this may suggest that the selected proteins carry non-linear signal that linear classifiers are less able to exploit, it is important to note that this result is not independent: the same model family is used for both selection and classification, and a random forest classifier may naturally perform better on proteins selected by a random forest-based importance measure. The performance advantage of the random forest classifier over logistic regression and XGBoost should therefore be interpreted with caution, and the absolute performance values across all classifiers are the more reliable indicator of the discriminative content of the selected proteins. Logistic regression performs comparably to XGBoost at small  $k$  but degrades more noticeably at  $k = 100$ , where the larger and noisier feature set appears to hurt its performance.

Performance peaks at  $k = 25$  and  $k = 50$  for most classifiers, with random forest reaching AUC of 0.666 and 0.667 respectively, and AUPR of 0.269 and 0.267. At  $k = 100$  performance remains stable for Random Forest and XGBoost but declines for Logistic Regression, consistent with the higher FDP observed at larger  $k$  in the simulation validation. The relatively modest improvement from  $k = 10$  to  $k = 25$  suggests that a small number of proteins carry most of the discriminative signal, with diminishing returns as the feature set grows.

### 9.5.3 Stability diagnostics

Table 25 shows the selection frequency of the top-20 proteins per  $k$  across the  $B = 50$  resampling splits. The final stable set is determined by stability selection with  $\tau = 0.7$  and  $k_0 = 50$ , meaning a protein must appear in the top-50 in at least 70% of splits to be included. This criterion decouples the final selection from any single ranking: a protein that appears important in most splits is included regardless of its exact rank, while a protein whose selection is sensitive to which patients are in the

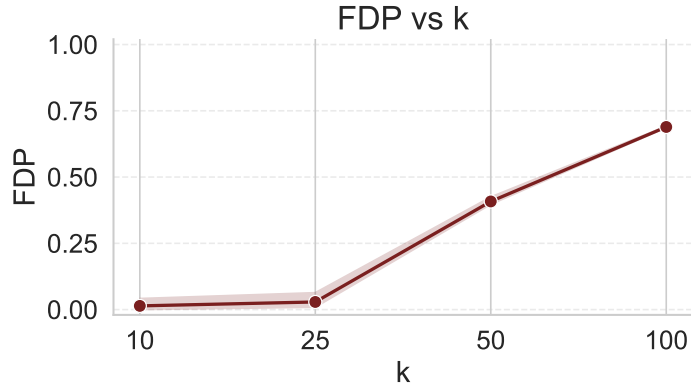


Figure 33: False discovery proportion (FDP) as a function of  $k$  for the RF+SHAP selection method using  $n = 2500$  samples, averaged across  $R = 50$  simulation repeats with 95% confidence intervals.

Table 23: Mean and standard deviation of the false discovery proportion (FDP) across top- $k$  selection sizes for RF+SHAP on simulated data with  $n = 2500$ , computed over  $R = 50$  simulation repeats.

| $k$ | FDP mean | FDP std |
|-----|----------|---------|
| 10  | 0.014    | 0.099   |
| 25  | 0.029    | 0.119   |
| 50  | 0.408    | 0.054   |
| 100 | 0.689    | 0.012   |

training fold is excluded.

This distinction is visible in the table. AMT, AGER, and SYNJ1 are selected with near-perfect frequency across splits and are robust inclusions in the final list. In contrast, proteins such as ALDH5A1 and CLIC3 reach reasonable frequency only at  $k = 100$ , suggesting their importance is more sensitive to the specific training sample. SDHAF2 illustrates the decoupling most clearly, it has low frequency at  $k = 10$  (0.08) but reaches 0.70 at  $k = 50$ , qualifying for the final list despite not being among the most consistently top-ranked proteins at small  $k$ . Proteins excluded from the final list despite appearing in the union, such as PHF11, TYK2, and STAT1, do not reach the  $\tau = 0.7$  threshold at  $k_0 = 50$ , indicating that their selection is too dependent on the specific training sample to be considered stable.

#### 9.5.4 Enrichment validation

Table 26 shows the 13 proteins in the final RF+SHAP selection, determined by stability selection with  $\tau = 0.7$  and  $k_0 = 50$ . GOT2 appears twice, reflecting two distinct SomaScan aptamers targeting the same protein, both passing the stability threshold independently.

Table 24: Classifier performance across top- $k$  selection sizes for RF+SHAP selected proteins on the SWECRIT cohort. Bold denotes the best classifier for each  $k$  per metric. Split std is the standard deviation of the metric across  $B = 50$  stratified train/test splits;  $SE_{NB}$  is the Nadeau–Bengio corrected standard error.

| Classifier          | k   | AUC Mean     | AUC Split Std | AUC $SE_{NB}$ | AUPR Mean    | AUPR Split Std | AUPR $SE_{NB}$ |
|---------------------|-----|--------------|---------------|---------------|--------------|----------------|----------------|
| LogisticRegression  | 10  | 0.605        | 0.072         | 0.037         | 0.248        | 0.073          | 0.038          |
| <b>RandomForest</b> | 10  | <b>0.640</b> | 0.073         | 0.038         | <b>0.268</b> | 0.075          | 0.039          |
| XGBoost             | 10  | 0.618        | 0.066         | 0.034         | 0.251        | 0.066          | 0.034          |
| LogisticRegression  | 25  | 0.631        | 0.071         | 0.037         | 0.259        | 0.077          | 0.040          |
| <b>RandomForest</b> | 25  | <b>0.666</b> | 0.069         | 0.036         | <b>0.269</b> | 0.076          | 0.039          |
| XGBoost             | 25  | 0.637        | 0.066         | 0.034         | 0.258        | 0.066          | 0.034          |
| LogisticRegression  | 50  | 0.609        | 0.075         | 0.039         | 0.237        | 0.064          | 0.033          |
| <b>RandomForest</b> | 50  | <b>0.667</b> | 0.065         | 0.034         | <b>0.267</b> | 0.070          | 0.036          |
| XGBoost             | 50  | 0.647        | 0.066         | 0.034         | 0.260        | 0.064          | 0.033          |
| LogisticRegression  | 100 | 0.589        | 0.067         | 0.035         | 0.219        | 0.054          | 0.028          |
| <b>RandomForest</b> | 100 | <b>0.662</b> | 0.066         | 0.034         | <b>0.255</b> | 0.067          | 0.035          |
| XGBoost             | 100 | 0.641        | 0.071         | 0.037         | 0.256        | 0.079          | 0.041          |

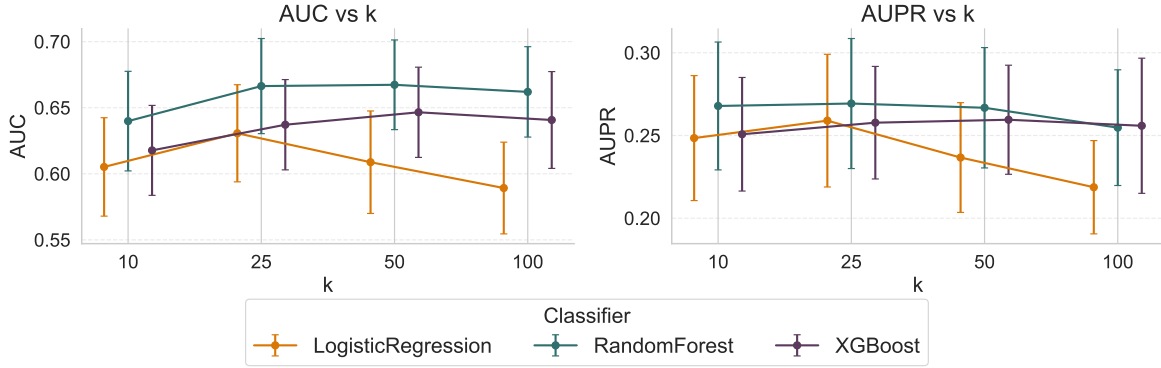


Figure 34: AUC (left) and AUPR (right) as a function of the top- $k$  selection size for each classifier (Logistic Regression, Random Forest, XGBoost), using RF+SHAP selected proteins on the SWECRIT cohort. Error bars denote Nadeau–Bengio corrected standard errors across  $B = 50$  stratified train/test splits. Random Forest achieves the highest performance across all values of  $k$ , while Logistic Regression degrades at larger  $k$ .

Table 26: Final selected proteins from RF+SHAP stability selection ( $n = 13$ , mapped symbols = 13). Proteins are ranked by mean absolute SHAP score  $s_j$ . Resample frequency reflects how often each protein appeared in the top- $k_0 = 50$  across the 50 resampling splits.

| Rank | Symbol   | Gene name                                            | Score  | Resample freq. |
|------|----------|------------------------------------------------------|--------|----------------|
| 1    | AMT      | Aminomethyltransferase                               | 0.0016 | 1.00           |
| 2    | AGER     | Advanced glycosylation end-product specific receptor | 0.0015 | 0.96           |
| 3    | SYNJ1    | Synaptojanin 1                                       | 0.0012 | 0.88           |
| 4    | RPA2     | Replication protein A2                               | 0.0012 | 0.82           |
| 5    | FKBP15   | FKBP prolyl isomerase family member 15               | 0.0011 | 0.86           |
| 6    | SIGLEC15 | Sialic acid binding Ig like lectin 15                | 0.0011 | 0.82           |
| 7    | MFAP5    | Microfibril associated protein 5                     | 0.0011 | 0.86           |
| 8    | RGN      | Regucalcin                                           | 0.0010 | 0.92           |
| 9    | KERA     | Keratocan                                            | 0.0010 | 0.70           |
| 10   | GOT2     | Glutamic-oxaloacetic transaminase 2                  | 0.0009 | 0.88           |
| 11   | GOT2     | Glutamic-oxaloacetic transaminase 2                  | 0.0009 | 0.72           |
| 12   | CROT     | Carnitine O-octanoyltransferase                      | 0.0008 | 0.76           |
| 13   | SDHAF2   | Succinate dehydrogenase complex assembly factor 2    | 0.0007 | 0.70           |

Table 25: Selection frequency of the top-20 proteins per  $k$  union (25 proteins total) across cross-validation splits, for each value of  $k$ , using RF+SHAP as the feature ranking criterion. Higher frequency indicates more consistent selection across splits. The *final* column indicates whether the protein is included in the final stable selection ( $\tau = 0.7$ ,  $k_0 = 50$ ).

| Protein      | k=10        | k=25        | k=50        | k=100       | Final |
|--------------|-------------|-------------|-------------|-------------|-------|
| AMT          | <b>0.92</b> | <b>1.00</b> | <b>1.00</b> | <b>1.00</b> | ✓     |
| AGER         | 0.68        | <b>0.90</b> | <b>0.96</b> | <b>1.00</b> | ✓     |
| SYNJ1        | 0.48        | <b>0.76</b> | <b>0.88</b> | <b>0.98</b> | ✓     |
| PHF11        | 0.38        | 0.56        | 0.62        | <b>0.88</b> | ✗     |
| RGN          | 0.34        | 0.64        | <b>0.92</b> | <b>0.96</b> | ✓     |
| MFAP5        | 0.34        | 0.60        | <b>0.86</b> | <b>0.94</b> | ✓     |
| RPA2         | 0.34        | 0.62        | <b>0.82</b> | <b>0.94</b> | ✓     |
| GOT2         | 0.32        | 0.70        | <b>0.88</b> | <b>0.96</b> | ✓     |
| SIGLEC15     | 0.32        | 0.66        | <b>0.82</b> | <b>0.90</b> | ✓     |
| FKBP15       | 0.30        | 0.56        | <b>0.86</b> | <b>1.00</b> | ✓     |
| seq.27613.39 | 0.28        | 0.44        | 0.66        | <b>0.94</b> | ✗     |
| SESN3        | 0.24        | 0.36        | 0.60        | <b>0.82</b> | ✗     |
| KERA         | 0.20        | 0.48        | 0.70        | <b>0.86</b> | ✓     |
| TYK2         | 0.20        | 0.30        | 0.44        | 0.60        | ✗     |
| SMAD9        | 0.18        | 0.38        | 0.64        | <b>0.86</b> | ✗     |
| CROT         | 0.16        | 0.46        | <b>0.76</b> | <b>0.98</b> | ✓     |
| PALMD        | 0.16        | 0.40        | 0.68        | <b>0.94</b> | ✗     |
| ARID3C       | 0.16        | 0.28        | 0.62        | <b>0.86</b> | ✗     |
| STAT1        | 0.16        | 0.36        | 0.58        | <b>0.78</b> | ✗     |
| KCTD6        | 0.16        | 0.30        | 0.46        | 0.70        | ✗     |
| SDHAF2       | 0.08        | 0.44        | 0.70        | <b>0.94</b> | ✓     |
| GOT2         | 0.14        | 0.40        | 0.72        | <b>0.84</b> | ✓     |
| SFTPB        | 0.12        | 0.36        | 0.62        | <b>0.82</b> | ✗     |
| ALDH5A1      | 0.00        | 0.18        | 0.58        | <b>0.88</b> | ✗     |
| CLIC3        | 0.08        | 0.28        | 0.54        | <b>0.84</b> | ✗     |



Figure 35: Mean AUC (left) and mean AUPR (right) for each classifier (Logistic Regression, Random Forest, XGBoost) and top- $k$  selection size, using RF+SHAP selected proteins on the SWECRIT cohort. Values are averaged across  $B = 50$  stratified train/test splits.

Table 27 shows the enrichment results for the RF+SHAP final protein list ( $|G| = 12$ ). The enrichment signal is weak overall, with no significant terms identified in GO Biological Process or KEGG, and a single significant term in Reactome Pathways 2024 (Table 28): Glyoxylate Metabolism and Glycine Degradation ( $q = 0.016$ ), driven by two proteins, GOT2 and AMT.

Table 27: Enrichment analysis results for the RF+SHAP selected proteins ( $|G| = 13$ ) across three pathway libraries. BES denotes the biological explainability score (raw value and  $z$ -score relative to the permutation null over random gene sets of matching size);  $p_{\text{emp}}$  is the empirical permutation  $p$ -value; “Significant terms” is the number of pathway terms with BH-adjusted  $q < 0.05$ .

| Library                    | BES Raw | BES $z$ -score | $p_{\text{emp}}$ | Significant terms |
|----------------------------|---------|----------------|------------------|-------------------|
| GO Biological Process 2025 | 0.000   | -0.314         | 1.000            | 0                 |
| KEGG 2026                  | 0.000   | -0.197         | 1.000            | 0                 |
| Reactome Pathways 2024     | 1.789   | 0.001          | 0.112            | 1                 |

Table 28: Significant enriched pathway terms ( $q < 0.05$ ) for the RF+SHAP selected proteins ( $|G| = 13$ ) in the Reactome Pathways 2024 library. Overlap is the number of selected proteins mapping to the term; term size is the total number of genes in the term in the library.

| Term                                          | $q$ -value | $p$ -value | Overlap | Term size | Genes     |
|-----------------------------------------------|------------|------------|---------|-----------|-----------|
| Glyoxylate Metabolism and Glycine Degradation | 0.016      | 0.0001     | 2       | 19        | GOT2, AMT |

The poor enrichment performance is likely attributable to several compounding factors. First, the final list is small ( $|G| = 13$ ), which reduces statistical power for over-representation analysis. Second, RF+SHAP selects proteins based on their contribution to a non-linear classification model rather than on differential expression, and the resulting proteins may not map cleanly onto the curated pathway structures that enrichment databases are built around. Third, as noted in Section 5, BES rewards proteins that are well-annotated in existing databases, which may penalise selection methods that identify less characterised proteins or novel associations. The single significant pathway should therefore be interpreted as a partial biological validation rather than a comprehensive summary of the biological relevance of the selected proteins.

## 10 Cross-Method Comparison

This section consolidates results across the three protein selection methods: random selection, differential expression via Welch’s  $t$ -test, mutual information, and RF+SHAP, evaluated under the unified framework described in Section 4. The comparison is structured around the three validation procedures: simulation validation, which characterises what types of signals each method is capable of detecting; classifier validation, which assesses whether the selected proteins are informative for ARDS classification on held-out data; and enrichment validation, which evaluates the biological coherence of the final protein sets. Since the same splits, classifiers, and evaluation protocol are used across all methods, differences in performance can be attributed to the selection methods themselves rather than to differences in experimental conditions. Native selection is generally excluded from the cross-method comparisons, as the native selection criteria differ between methods and do not produce directly comparable protein sets. The comparison is therefore restricted to top- $k$  selections across  $k \in \{10, 25, 50, 100\}$ .

### 10.1 Simulation Results

Cross-method comparisons are restricted to  $n = 500$  to match the SWECRIT cohort size; per-method results at  $n = 2500$ , which characterize sample-size dependence, are reported in the respective method sections.

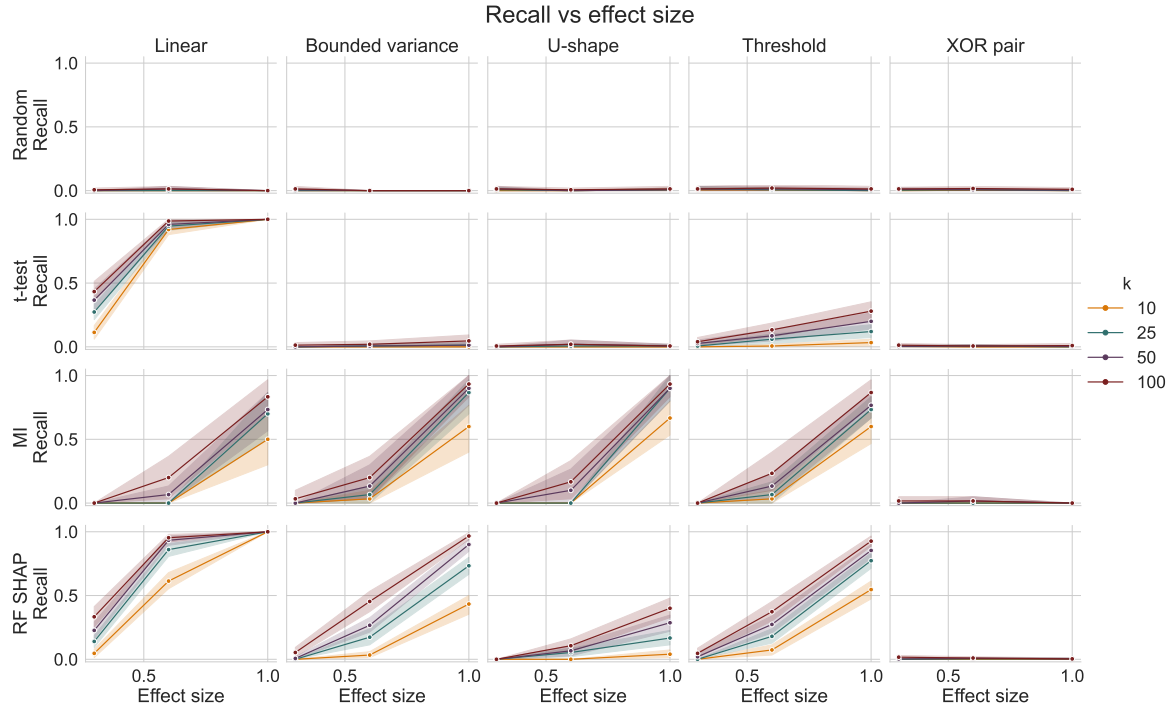
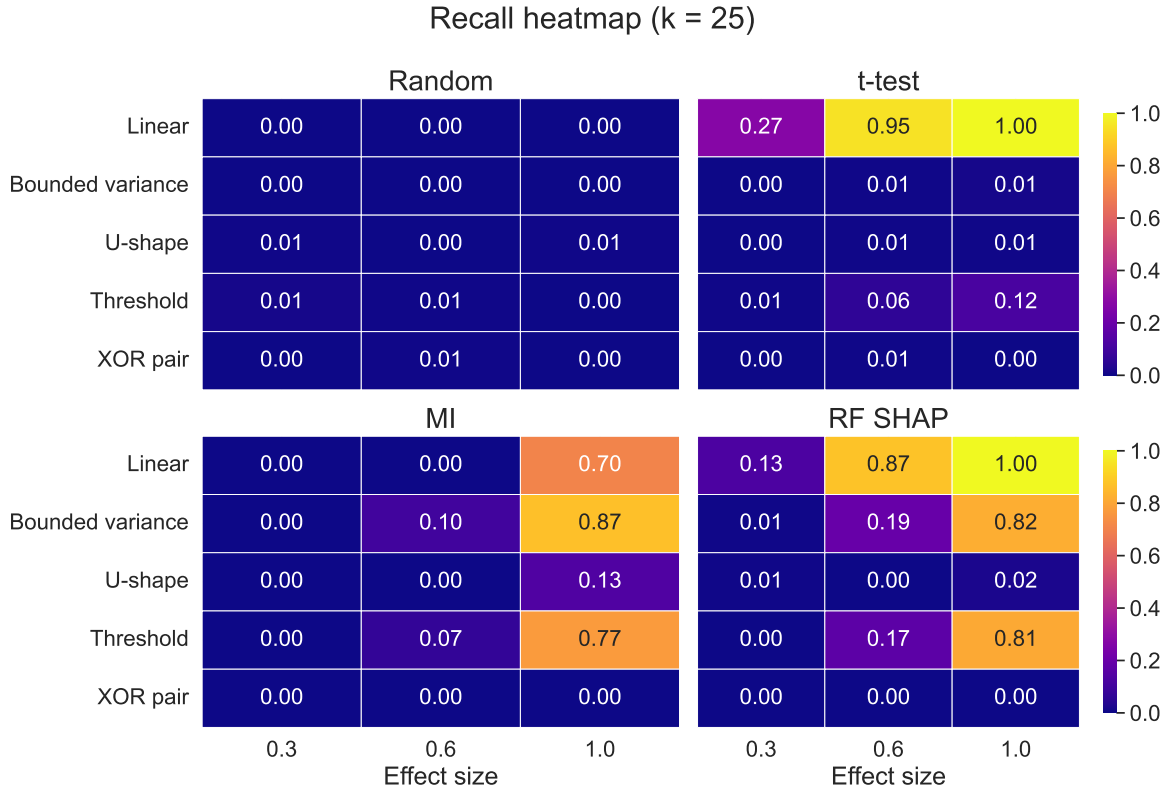
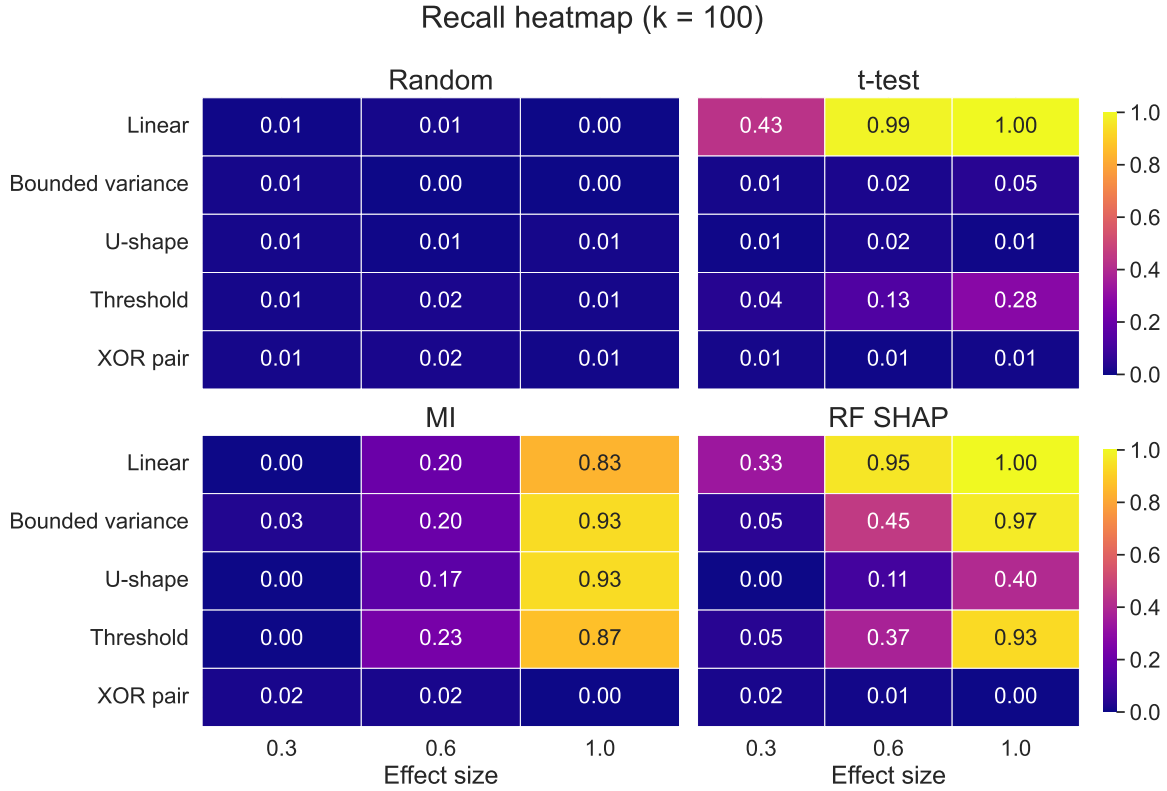


Figure 36: Recall as a function of effect size  $\delta$  for each signal type (columns: linear, bounded variance, U-shape, threshold, XOR pair) and selection method (rows: random,  $t$ -test, MI, RF+SHAP) on simulated data with  $n = 500$ . Lines show mean recall across  $R = 50$  simulation repeats for each top- $k$  selection size ( $k \in \{10, 25, 50, 100\}$ ); shaded regions denote 95% confidence intervals.

Figures 36 and 37 show recall across signal types, effect sizes, and values of  $k$  for all four methods, and Figure 38 shows the corresponding FDP. Linear signals are recoverable by all non-random methods. Threshold and bounded variance signals show a clear divide: the  $t$ -test recovers them only partially or not at all, as they produce little to no mean shift between classes, while MI and RF+SHAP achieve substantially higher recall by capturing non-linear associations. MI additionally recovers U-shape signals at large effect sizes while RF+SHAP does that to a small extent, reflecting a difference in sensitivity to distributional shape. No method recovers XOR pair signals at any effect size, confirming that all four methods operate at the univariate level and cannot detect pure interaction effects where neither protein has a marginal association with the outcome.



(a)  $k = 25$



(b)  $k = 100$

Figure 37: Recall heatmaps for each selection method (random,  $t$ -test, MI, RF+SHAP) at  $k = 25$  (top) and  $k = 100$  (bottom) on simulated data with  $n = 500$ . Each cell shows the mean recall across  $R = 50$  simulation repeats for the corresponding signal type (rows) and effect size  $\delta$  (columns). Recall is the fraction of planted features of a given signal type that appear in the method's top- $k$  selection.

The recall heatmaps in Figure 37 further illustrate the effect of increasing  $k$ . For the  $t$ -test, recall improves modestly between  $k = 25$  and  $k = 100$ , primarily for linear signals, and remains near zero for all non-linear signal types. Remember that we have a total of 9 linear signals, therefore, if  $k = 25$  detects almost all linear signals,  $k = 100$  is not likely to detect many more. For MI and RF+SHAP, the gain from  $k = 25$  to  $k = 100$  is considerably larger, particularly for bounded variance and threshold signals, suggesting that the relevant proteins are consistently ranked but may not always appear in the top-25.

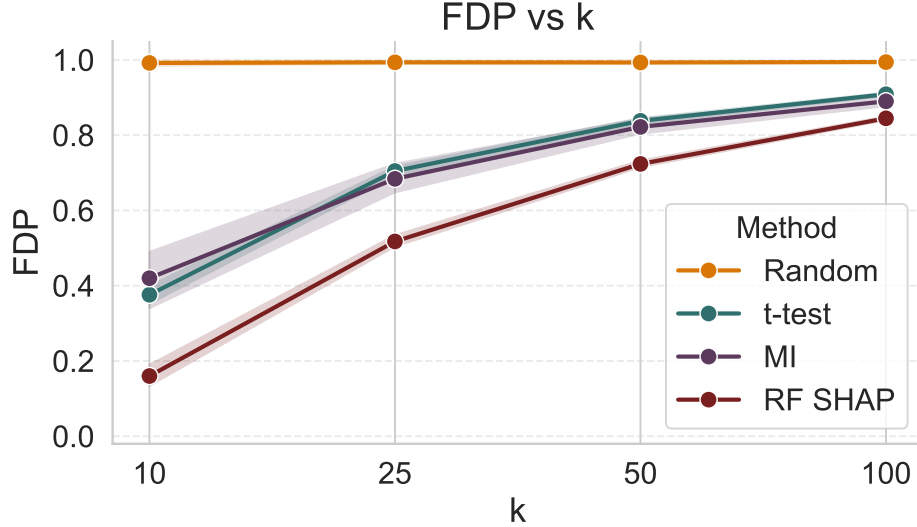


Figure 38: False discovery proportion (FDP) as a function of the top- $k$  selection size for each method on simulated data with  $n = 500$ . Shaded regions denote 95% confidence intervals across  $R = 50$  simulation repeats. Random selection maintains FDP near 1.0 across all  $k$ , while the  $t$ -test, MI, and RF+SHAP show monotonically increasing FDP with  $k$ .

The FDP increases with  $k$  for all methods, as Figure 38 shows, confirming that a larger candidate pool inevitably includes more noise proteins. Random selection maintains FDP near 1.0 across all  $k$ , confirming that it provides no discriminative power. The  $t$ -test and MI show similar FDP trajectories throughout. RF+SHAP consistently achieves lower FDP than the other non-random methods across all values of  $k$ , suggesting that the random forest importance criterion is more selective in ranking true signal proteins above noise.

## 10.2 Classifier Results

Tables 29 and 30 show AUC and AUPR across all classifiers, selection methods, and values of  $k$ , with significance assessed against the random baseline via paired permutation tests. Note here that significance is determined on unadjusted  $p$ -values, and should be read in light of this. Figure 39 illustrates the performance trajectories across  $k$ .

RF+SHAP is the only method that achieves consistently significant improvement over the random baseline across nearly all classifiers and values of  $k$ , for both AUC and AUPR. The  $t$ -test achieves significant improvement at small  $k$  for Logistic Regression and random forest, and at larger  $k$  for XGBoost. The latter is somewhat counterintuitive, as larger  $k$  implies a higher proportion of noise proteins. However, XGBoost’s tree-based structure may be better equipped to handle noisy features at larger  $k$  than at smaller  $k$ , where the signal is more concentrated but the feature set too small for XGBoost to exploit effectively. MI does not achieve significant improvement over random at any  $k$  or classifier combination, suggesting that the mutual information criterion does not consistently identify proteins with sufficient discriminative power for classification in this cohort.

The degradation of RF+SHAP performance at larger  $k$  for Logistic Regression, but not for random forest and XGBoost, is consistent with the higher FDP at large  $k$  observed in the simulation validation. Logistic regression assigns equal weight to all features and is more sensitive to noise proteins entering

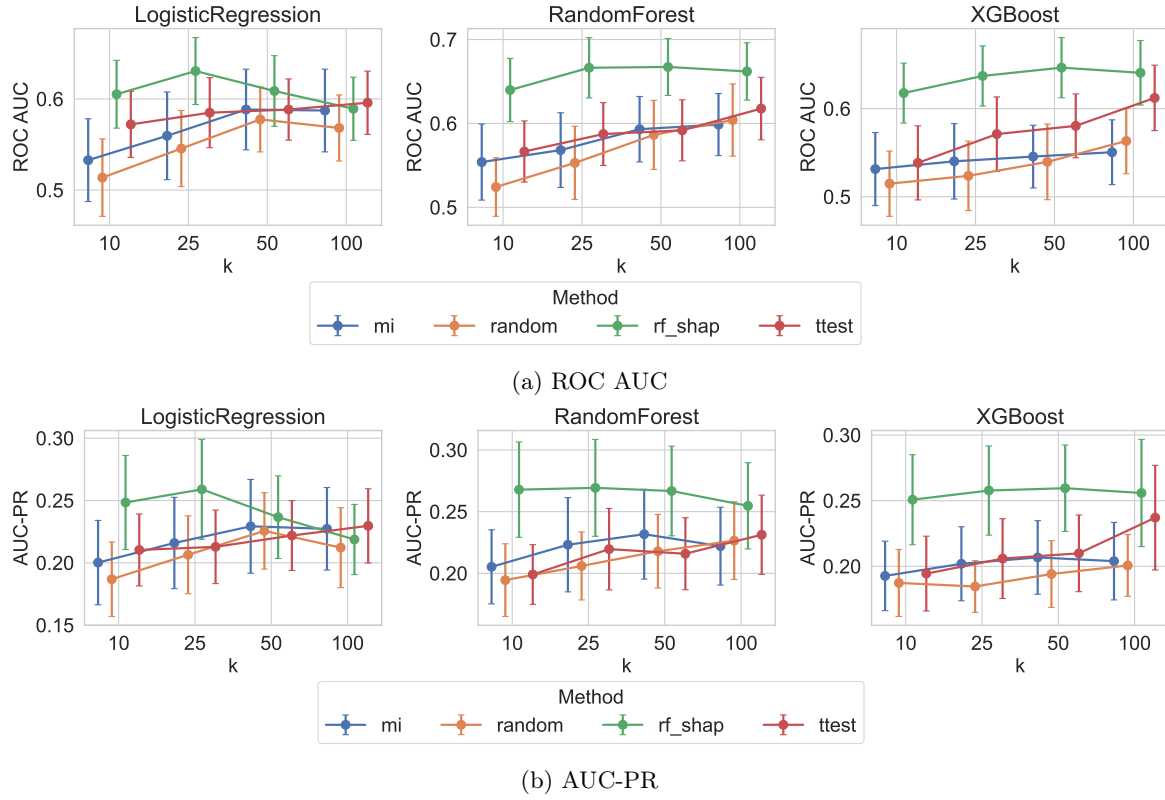


Figure 39: ROC AUC (top) and AUC-PR (bottom) as a function of the top- $k$  selection size for each classifier (Logistic Regression, Random Forest, XGBoost) and selection method (MI, random, RF+SHAP,  $t$ -test) on the SWECRIT cohort. Error bars denote Nadeau-Bengio corrected standard errors across  $B = 50$  stratified train/test splits.

the feature set, while tree-based classifiers can implicitly down-weight uninformative features through their splitting criterion.

Table 29: ROC AUC (mean  $\pm$  Nadeau-Bengio SE) per classifier and feature selection size  $k$ , across selection methods. Bold entries are significantly different from the random baseline (paired permutation test, unadjusted): \*  $p < 0.05$ , \*\*  $p < 0.01$ , \*\*\*  $p < 0.001$ .

| Classifier         | $k$ | Random            | $t$ -test                               | MI                | RF-SHAP                                 |
|--------------------|-----|-------------------|-----------------------------------------|-------------------|-----------------------------------------|
| LogisticRegression | 10  | 0.514 $\pm$ 0.042 | <b>0.572 <math>\pm</math> 0.036 ***</b> | 0.533 $\pm$ 0.045 | <b>0.605 <math>\pm</math> 0.037 ***</b> |
|                    | 25  | 0.546 $\pm$ 0.042 | <b>0.585 <math>\pm</math> 0.038 **</b>  | 0.560 $\pm$ 0.048 | <b>0.631 <math>\pm</math> 0.037 ***</b> |
|                    | 50  | 0.578 $\pm$ 0.036 | 0.588 $\pm$ 0.034                       | 0.588 $\pm$ 0.044 | <b>0.609 <math>\pm</math> 0.039 *</b>   |
|                    | 100 | 0.568 $\pm$ 0.036 | 0.596 $\pm$ 0.035                       | 0.587 $\pm$ 0.045 | 0.589 $\pm$ 0.035                       |
| RandomForest       | 10  | 0.524 $\pm$ 0.035 | <b>0.567 <math>\pm</math> 0.036 **</b>  | 0.554 $\pm$ 0.045 | <b>0.640 <math>\pm</math> 0.038 ***</b> |
|                    | 25  | 0.553 $\pm$ 0.044 | <b>0.587 <math>\pm</math> 0.037 *</b>   | 0.568 $\pm$ 0.045 | <b>0.666 <math>\pm</math> 0.036 ***</b> |
|                    | 50  | 0.586 $\pm$ 0.041 | 0.592 $\pm$ 0.036                       | 0.593 $\pm$ 0.039 | <b>0.667 <math>\pm</math> 0.034 ***</b> |
|                    | 100 | 0.604 $\pm$ 0.043 | 0.618 $\pm$ 0.037                       | 0.599 $\pm$ 0.037 | <b>0.662 <math>\pm</math> 0.034 ***</b> |
| XGBoost            | 10  | 0.515 $\pm$ 0.037 | 0.538 $\pm$ 0.042                       | 0.531 $\pm$ 0.041 | <b>0.618 <math>\pm</math> 0.034 ***</b> |
|                    | 25  | 0.524 $\pm$ 0.039 | <b>0.571 <math>\pm</math> 0.042 **</b>  | 0.540 $\pm$ 0.043 | <b>0.637 <math>\pm</math> 0.034 ***</b> |
|                    | 50  | 0.540 $\pm$ 0.043 | <b>0.580 <math>\pm</math> 0.036 **</b>  | 0.546 $\pm$ 0.036 | <b>0.647 <math>\pm</math> 0.034 ***</b> |
|                    | 100 | 0.563 $\pm$ 0.037 | <b>0.612 <math>\pm</math> 0.037 ***</b> | 0.551 $\pm$ 0.037 | <b>0.641 <math>\pm</math> 0.037 ***</b> |

Table 30: AUC-PR (mean  $\pm$  Nadeau–Bengio SE) per classifier and feature selection size  $k$ , across selection methods. Bold entries are significantly different from the random baseline (paired permutation test, unadjusted): \*  $p < 0.05$ , \*\*  $p < 0.01$ , \*\*\*  $p < 0.001$ .

| Classifier         | $k$ | Random            | $t$ -test                              | MI                | RF-SHAP                                 |
|--------------------|-----|-------------------|----------------------------------------|-------------------|-----------------------------------------|
| LogisticRegression | 10  | 0.187 $\pm$ 0.030 | <b>0.210 <math>\pm</math> 0.029 *</b>  | 0.200 $\pm$ 0.034 | <b>0.248 <math>\pm</math> 0.038 ***</b> |
|                    | 25  | 0.206 $\pm$ 0.031 | 0.213 $\pm$ 0.030                      | 0.216 $\pm$ 0.037 | <b>0.259 <math>\pm</math> 0.040 ***</b> |
|                    | 50  | 0.226 $\pm$ 0.031 | 0.222 $\pm$ 0.028                      | 0.229 $\pm$ 0.038 | 0.237 $\pm$ 0.033                       |
|                    | 100 | 0.212 $\pm$ 0.032 | 0.230 $\pm$ 0.030                      | 0.227 $\pm$ 0.033 | 0.219 $\pm$ 0.028                       |
| RandomForest       | 10  | 0.194 $\pm$ 0.029 | 0.199 $\pm$ 0.024                      | 0.205 $\pm$ 0.030 | <b>0.268 <math>\pm</math> 0.039 ***</b> |
|                    | 25  | 0.206 $\pm$ 0.028 | 0.220 $\pm$ 0.033                      | 0.223 $\pm$ 0.038 | <b>0.269 <math>\pm</math> 0.039 ***</b> |
|                    | 50  | 0.218 $\pm$ 0.030 | 0.216 $\pm$ 0.029                      | 0.232 $\pm$ 0.036 | <b>0.267 <math>\pm</math> 0.036 ***</b> |
|                    | 100 | 0.227 $\pm$ 0.031 | 0.231 $\pm$ 0.032                      | 0.222 $\pm$ 0.032 | <b>0.255 <math>\pm</math> 0.035 ***</b> |
| XGBoost            | 10  | 0.187 $\pm$ 0.026 | 0.194 $\pm$ 0.029                      | 0.193 $\pm$ 0.026 | <b>0.251 <math>\pm</math> 0.034 ***</b> |
|                    | 25  | 0.185 $\pm$ 0.028 | <b>0.206 <math>\pm</math> 0.030 *</b>  | 0.202 $\pm$ 0.028 | <b>0.258 <math>\pm</math> 0.034 ***</b> |
|                    | 50  | 0.194 $\pm$ 0.025 | 0.210 $\pm$ 0.029                      | 0.207 $\pm$ 0.028 | <b>0.260 <math>\pm</math> 0.033 ***</b> |
|                    | 100 | 0.201 $\pm$ 0.024 | <b>0.237 <math>\pm</math> 0.040 **</b> | 0.204 $\pm$ 0.030 | <b>0.256 <math>\pm</math> 0.041 ***</b> |

### 10.3 Enrichment Results

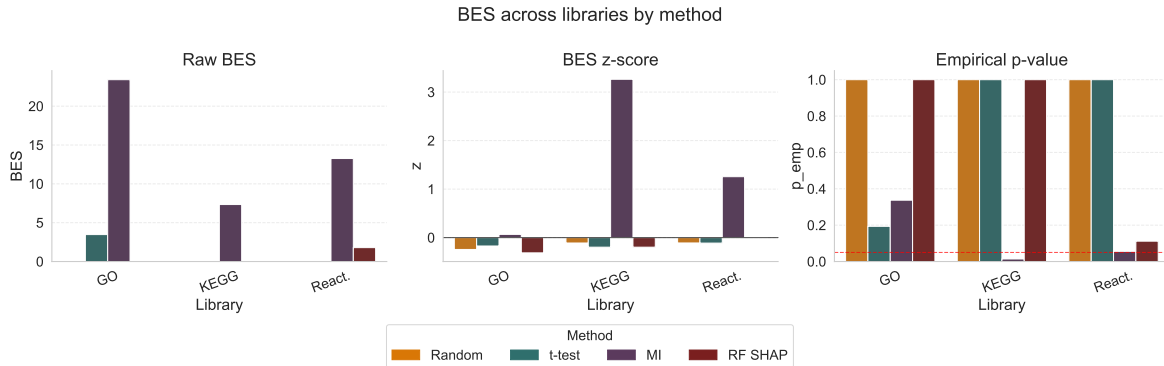


Figure 40: Biological Explainability Score (BES) across libraries and methods. Left: raw BES. Middle: BES  $z$ -score relative to a null distribution of randomly selected gene sets. Right: empirical  $p$ -value, with the dashed red line indicating  $p = 0.05$ . MI dominates across all three measures, with the strongest signal in KEGG. Random,  $t$ -test, and RF+SHAP show negligible enrichment signal across all libraries.

As shown in Figures 40 and 41, MI is the only method that produces meaningful enrichment signal. In terms of raw BES, MI achieves the highest scores across all three libraries, with the largest absolute value in GO Biological Process. However, the empirical  $p$ -value, which tests whether the observed BES exceeds what is expected from randomly selected gene sets of the same size, reveals that the GO result is not significant despite its large raw score. The BES  $z$ -score accounts for this by normalising against the null distribution, and here the picture changes: the strongest relative enrichment for MI is in KEGG ( $z \approx 3.3$ ,  $p < 0.05$ ), where the more discrete pathway structure makes enrichment a more specific signal. Reactome shows a moderate  $z$ -score for MI ( $z \approx 1.3$ ) but does not reach significance. Random,  $t$ -test, and RF+SHAP show BES  $z$ -scores near zero or negative across all libraries, indicating no enrichment beyond what is expected by chance. Reactome shows a moderate  $z$ -score for MI ( $z \approx 1.3$ ) but does not reach significance. Random,  $t$ -test, and RF+SHAP show BES  $z$ -scores near zero or negative across all libraries, indicating no enrichment beyond what is expected by chance.

The number of significant terms in Figure 41 confirms this pattern. MI identifies 16 significant GO terms, 5 KEGG pathways, and 3 Reactome pathways, making it the only method with broad biological support across multiple libraries. The  $t$ -test yields 2 significant GO terms but no significant pathways in

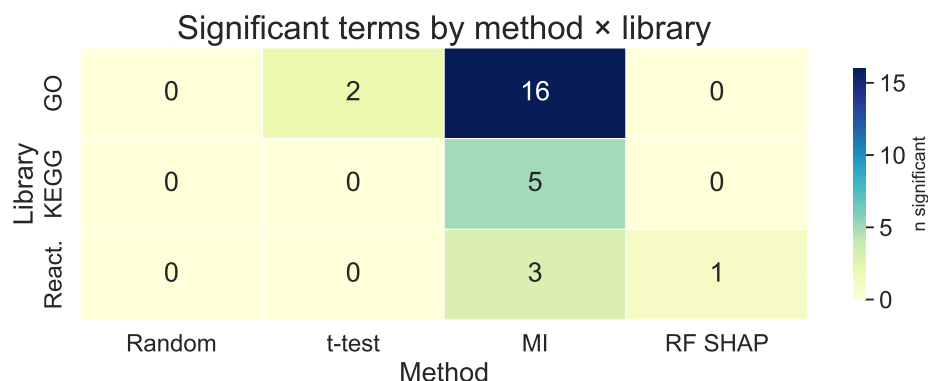


Figure 41: Number of significant enriched terms per method and library after multiple testing correction. MI yields the largest number of significant terms across all libraries, with 16 in GO Biological Process, 5 in KEGG, and 3 in Reactome. The *t*-test yields 2 significant terms in GO only. Random selection and RF+SHAP yield at most 1 significant term across all libraries.

KEGG or Reactome, suggesting that the differentially expressed proteins share some biological process annotations but do not form coherent pathway-level clusters. RF+SHAP yields a single significant Reactome term, and random selection yields none, as expected.

The dominance of MI in enrichment validation is notable given its relatively weak performance in classifier validation. This apparent contradiction may reflect the different objectives of the two validation approaches: enrichment rewards proteins that are biologically coherent and well-annotated, while classifier validation rewards proteins that carry discriminative signal for ARDS prediction. MI may identify proteins that participate in known biological pathways without necessarily being the strongest individual predictors of ARDS status, while RF+SHAP prioritises predictive utility at the cost of biological coherence in current databases.

## 11 Discussion

### 11.1 Method-Specific Findings and Cross-Method Patterns

#### *t*-test

Differential expression analysis via Welch’s *t*-test with BH-controlled FDR proved reliable for detecting linear signals, recovering both strongly and weakly regulated proteins in the simulation data, with stability diagnostics comparable to or better than the more complex methods evaluated here. The method is by construction blind to non-mean-shift regulation, and the partial recovery of threshold signals observed in simulation should not be interpreted as detection of genuinely non-linear structure, threshold signals a small mean shift that the *t*-test incidentally detects. It is also worth noting that the simulation data is generated from normal distributions, which inherently favours the *t*-test’s assumptions, and performance on real proteomic data, where relationships are likely non-linear and multivariate, may be more limited.

The *t*-test retains reasonable detection capability at smaller sample sizes precisely because of its strong distributional assumptions: regulated proteins are assumed to be mean-shifted and approximately normally distributed. It is therefore preferable when these assumptions hold, when sample sizes are limited, and when interpretability and FDR control are priorities. When the goal is to detect proteins regulated through non-linear or non-mean-shift mechanisms, however, the *t*-test is insufficient.

In the SWECRIT cohort, the proteins identified by the *t*-test, most notably SDHAF2 and COX5B and their associated mitochondrial pathway, should be interpreted as genuine mean shift. The enrichment results, however, did not yield BES *z*-scores or empirical *p*-values of interest beyond these mitochondrial findings, indicating that the selected proteins do not form a broadly enriched biological set.

#### Mutual Information

The mutual information based selector is, in principle, capable of detecting arbitrary univariate dependencies, and the use of permutation testing enables FDR control without assumptions about the form of regulation or the distribution of the data. In practice, however, the method is clearly power-limited at  $n = 500$ , as evidenced both by the simulation results and by the real data, where no proteins approached the significance threshold  $q = 0.05$ . The *k*-NN entropy estimators used here are sample-hungry, and the  $n = 2500$  simulation confirms that under-performance at small sample sizes reflects a power issue rather than a fundamental limitation of the approach. Even at larger sample sizes, the method struggles to detect weakly regulated proteins, and the biological signals of interest in ARDS may well fall in this weakly regulated regime.

A further practical consideration is computational cost. Larger sample sizes increase the time required to estimate mutual information, which is already the dominant bottleneck in the permutation testing procedure. In the present analysis, no proteins reached significance after BH correction, allowing the adaptive permutation test to terminate early for all proteins. If sample size increased and a non-trivial number of proteins reached significance, the required number of permutations, and thus total compute, would grow accordingly. Possible mitigations are discussed in Section 8.3.

Perhaps the most striking result of the cross-method comparison is the divergence between MI’s enrichment and classifier performance. Despite failing to identify any significant proteins in the real data, the top-ranked MI proteins yield by far the strongest enrichment signal across all methods and libraries, yet fail to produce protein sets that consistently outperform random selection in classifier validation. The MI final list is also unstable by the standards of this framework, only one protein appears in more than 70% of resampling splits at  $k = 100$ , suggesting that the selected proteins vary considerably depending on the training sample. One interpretation is that MI identifies proteins with genuine biological coherence, proteins that participate in known pathways and are therefore well-represented in enrichment databases, but whose individual associations with ARDS are too weak or too variable across patients to translate into reliable discriminative signal for classification. In other words, MI may capture biologically relevant but statistically fragile associations that are sufficient for pathway-level enrichment but insufficient for individual-level prediction. Mutual information is therefore best suited to settings with large sample sizes where non-linear regulation is suspected, and where interaction information can be used to detect higher-order dependencies.

## Random Forest with SHAP

Random forest selection via SHAP values produces the strongest classifier performance of any method evaluated, with all three classifiers improving substantially over the random baseline across nearly all values of  $k$ . The method also detects non-linear monotonic regulation, including threshold and bounded-variance signals, reflecting the random forest’s capacity to exploit non-linear structure during fitting. A methodological caveat applies, however: the result of an RF classifier evaluated on RF+SHAP selected features is partly self-validating, as the selection criterion favours proteins that are by construction well-suited to tree-based models. The more reliable signal is that all three classifiers, including the linear baseline, improve over random selection when using RF+SHAP features, and that the drop in performance at  $k = 100$  for logistic regression, but not for the tree-based classifiers, is consistent with a larger and noisier feature set being more damaging for a linear model that cannot implicitly down-weight uninformative proteins.

Despite the random forest’s capacity to exploit interactions during fitting, the method fails to detect XOR pair signals at any effect size. This is attributable to SHAP being additive by construction: it marginalises over interactions in the attribution step even when the underlying model used them during fitting, placing a fundamental ceiling on what RF+SHAP can detect in terms of higher-order protein interactions rather than a tuning issue that could be resolved with different hyperparameters. A natural extension would be to exploit TreeExplainer’s capacity to compute pairwise SHAP interaction values  $\phi_{ijk}$  [LEC<sup>+</sup>20], which decompose each prediction into individual and interaction contributions, allowing protein pairs with strong joint effects to be identified directly from the fitted model without requiring a separate combinatorial search.

A further limitation is the absence of a natural native selection criterion that controls FDR. In this work this is addressed through stability selection. The classifier results suggest that performance generally plateaus, or begins to degrade, beyond  $k \approx 25$ , which points to an alternative native criterion: defining the cutoff as the value of  $k$  at which classifier performance ceases to increase significantly. This would tie the selection more directly to the data, at the cost of introducing dependence on the choice of classifier.

The proteins identified by RF+SHAP, notably AGER, AMT, and SYNJ1, are predictively useful but show weak enrichment, with only a single significant Reactome term identified. The weak enrichment signal may reflect a genuine property of the selected proteins rather than a limitation of the method: proteins selected on the basis of non-linear predictive utility may not correspond to the linear and well-annotated pathway associations that current enrichment databases are built around. Whether this represents a limitation of current enrichment libraries or a genuine difference in the biological signal captured by RF+SHAP remains an open question.

## Cross-method protein overlap

The degree of overlap between the final protein lists across methods is another dimension worth considering. A low overlap would suggest that the methods capture genuinely different aspects of the proteomic signal, complementary rather than redundant, while a high overlap would indicate convergent evidence for a shared set of ARDS-associated proteins. The overlap between the final protein lists is minimal: only SDHAF2 and RGN appear in more than one list, both selected by the  $t$ -test and RF+SHAP. No proteins are shared between MI and either of the other two methods, consistent with the divergent validation profiles observed across methods.

## Univariate limitations and interaction effects

Finally, no method recovers XOR-pair signals, confirming that all selection criteria in this framework operate at the univariate level, regardless of whether the underlying model can exploit interactions during fitting, as discussed for RF+SHAP above. This is a fundamental constraint for the present framework as a whole, and developing selection methods that explicitly target interaction effects represents a natural direction for future work.

## 11.2 Framework Strengths and Limitations

### Strengths

This study proposes a complete framework and pipeline for evaluating protein selection methods that can be applied to any arbitrary selector. The shared infrastructure enables fair comparison across methods under a unified evaluation protocol, with paired tests made possible by reusing the same train/test splits across all methods.

The three complementary validation strategies proved valuable: each addresses a different question, and methods performed differently across them, providing a more nuanced characterisation of performance than any single strategy could yield in isolation. Simulation validation establishes what types of signals a method is capable of detecting under known ground truth; classifier validation assesses whether the selected proteins are informative for ARDS prediction on held-out data; and enrichment validation evaluates the biological coherence of the final protein set.

### Limitations

The framework nevertheless has several limitations. The simulation validation relies on simplifications that influence which methods appear most competitive. Features are drawn from a normal distribution, which favours the *t*-test through its distributional assumptions. The covariance structure is block-diagonal, which is unlikely to reflect the more complex correlation patterns of real proteomic data, including hub proteins that correlate with many others. Several of the injected signal types also alter the marginal variance of the affected features, meaning that a trivial variance filter could in principle recover them.

The enrichment validation also has an inherent bias: it rewards proteins that map onto well-annotated biological functions and pathways. This is in tension with one of the central aims of the thesis, which is to identify novel non-linear and higher-order effects that may involve poorly annotated proteins. A selection method that recovers genuinely novel mechanisms can therefore score lower in enrichment validation than one that re-identifies well-established sepsis or ARDS markers, simply because the former is penalised by the structure of existing annotation databases.

Beyond these validation-specific issues, the framework as implemented has further limitations worth noting. Classifier hyperparameters are fixed rather than tuned, which keeps comparisons across selectors clean but means absolute classifier performance should be interpreted as a lower bound. Stability selection is applied only to RF+SHAP rather than uniformly across methods, which limits cross-method comparability of the final selected sets. Permutation-based significance is computationally infeasible for RF+SHAP given the size of the ensemble, so its native selection rests on a different statistical foundation than the other methods. Finally, no clinical covariates are adjusted for in the selection step, which is a deliberate simplification but represents a clear limitation for any causal interpretation of the findings.

## 11.3 Clinical and Biological Implications

### Biologically plausible signals

The protein sets identified across methods contain several biologically plausible signals consistent with the metabolic and inflammatory dysregulation seen in sepsis-associated ARDS. The *t*-test identifies SDHAF2 and COX5B, both involved in the mitochondrial electron transport chain, pointing to impaired oxidative phosphorylation. MI enrichment analysis independently converges on a related theme: ketone body metabolism emerges as the strongest enrichment signal, with ACSS3, BDH1, and ACAT1 jointly driving significant terms in both GO and Reactome ( $q < 0.0001$ ). Together, these signals are consistent with the metabolic reprogramming observed in critical illness, where mitochondrial dysfunction and shifts toward alternative energy substrates are established features of sepsis-induced organ failure [Sin14]. AGER, also called RAGE, the top-ranked RF+SHAP protein, has been previously identified as a marker of alveolar epithelial injury in ARDS [USW<sup>+</sup>06], lending biological plausibility to its high stability across resampling splits in this cohort; its heterogeneous association pattern across individual ARDS patients is discussed further in Section 11.4.

## Data-driven limitations

However, all biological interpretations in this work are post-hoc, derived from database annotations rather than a priori biological hypotheses. The identified proteins should therefore be treated as candidates for future experimental validation rather than confirmed ARDS biomarkers, and the enrichment results as pointers toward plausible biological mechanisms rather than established findings.

## Confounders

Several unadjusted confounders limit the strength of any causal claims. ICU site is a notable source of heterogeneity, with ARDS prevalence ranging from 7.5% in Helsingborg to 22.4% in Lund, which may reflect differences in patient population or clinical practice rather than biology. Additionally, time from ICU admission to sampling and treatment differences between patients are not controlled for, and may influence protein expression independently of ARDS status.

## ARDS heterogeneity and endotypes

Finally, phenotypic heterogeneity within the ARDS category itself is a fundamental challenge [SP23]. ARDS encompasses distinct biological subtypes, or endotypes, characterised by different inflammatory profiles and clinical outcomes [CDP<sup>+</sup>14]. The presence of such endotypes within the ARDS group may partly explain why no single protein cleanly separates ARDS from non-ARDS patients across all individuals in this cohort. Methods that capture patient-level heterogeneity, such as the patient-level SHAP decomposition available from RF+SHAP, may be particularly valuable for future endotype-aware subgroup analyses, though this remains to be explored.

## 11.4 Future Work

### Larger and More Balanced Cohorts

The simulation results for mutual information illustrate the impact of sample size concretely: at  $n = 500$ , matching the size of the SWECRIT cohort, the native selector identifies on average only 4.7 proteins per simulation repeat with low recall across all signal types. At  $n = 2,500$ , the same method selects on average 16.1 proteins with substantially improved recall and near-zero FDP at  $k = 10$  (Table 14). No proteins reached significance in the real data, which is entirely consistent with this power limitation rather than an absence of biology. While MI is the most acutely affected given its reliance on sample-hungry  $k$ -NN entropy estimators, larger and more balanced cohorts would likely benefit all methods evaluated here: more samples reduce variance in SHAP estimates, improve stability of top- $k$  rankings, and increase the power of permutation tests. The class imbalance in the SWECRIT cohort, with only 73 ARDS patients out of 512, further compounds the power limitation, as the effective sample size for detecting ARDS-specific signals is constrained by the minority class. Future analyses drawing on larger subsets of the SWECRIT registry, or on multi-centre collaborations with similar ICU proteomics cohorts, would provide the statistical power needed to fully evaluate information-theoretic approaches and interaction-based selection methods.

### Investigating heterogeneous proteins

The patient-level SHAP decomposition available from RF+SHAP enables investigation of protein-level heterogeneity across patients, which is not possible with univariate methods. Preliminary exploration revealed that AGER, the top-ranked protein overall, exhibits a notably heterogeneous association pattern: approximately half of ARDS patients received large positive SHAP values for AGER, while the remaining half received negative values. The positive contributions were sufficiently strong that AGER ranked highly in the global mean absolute SHAP score despite this split, suggesting that its apparent importance reflects a subgroup-specific signal rather than a uniform association across all ARDS patients. This pattern is consistent with the existence of distinct ARDS endotypes [CDP<sup>+</sup>14, SP23], and raises the possibility that other top-ranked proteins exhibit similarly heterogeneous associations that are obscured by population-level averaging.

Addressing this requires both methodological and framework-level extensions. At the method level, patient-level SHAP profiles could be used directly for unsupervised clustering of ARDS patients, enabling data-driven endotype discovery without requiring a priori subgroup definitions. At the framework level, the current binary ARDS versus non-ARDS formulation could be extended to a multi-label or subgroup-aware setting, where selection methods are evaluated on their ability to identify

proteins that are informative for specific endotypes rather than for the heterogeneous ARDS category as a whole. Such an extension would also provide a more meaningful evaluation context for methods such as mutual information, whose sensitivity to complex distributional differences may be better suited to subgroup discrimination than to the coarse binary classification used here.

### SHAP Interaction Values

The TreeExplainer algorithm used for RF+SHAP can compute pairwise SHAP interaction values  $\phi_{ijk}$  [LEC<sup>+</sup>20], which decompose each prediction into additive individual contributions and pairwise interaction terms. In the present work, only the marginal SHAP values  $\phi_{ij}$  are used to rank proteins, which means that proteins whose importance is primarily interaction-driven may receive low scores even when the underlying model exploits their joint effect. Extending the selection criterion to incorporate interaction SHAP values would allow protein pairs with strong synergistic associations to be identified directly, and would provide a principled bridge between the additive attribution framework and the interaction-based selection methods motivated in the introduction. The main practical challenge is computational: interaction SHAP values scale quadratically in the number of features, making exhaustive computation over all 10,000 proteins infeasible. A targeted approach, computing interaction values only for the top- $k$  proteins identified by the marginal ranking, would reduce this cost substantially and represents a natural first step.

### Data-driven native cutoffs

A principled data-driven native selection criterion remains an open problem for model-based selectors more generally. As discussed in Section 9, stability selection addresses this practically but does not provide strict FDR guarantees in the 80/20 split setting used here. Alternative approaches worth exploring include classifier-based cutoffs, selecting the largest  $k$  at which classifier performance continues to improve.

A more statistically principled alternative is the knockoff filter [BC19], which provides exact FDR control without distributional assumptions on the response variable, and is agnostic to the choice of importance statistic, meaning the SHAP score  $s_j$  could in principle serve as the statistic directly. However, knockoff constructions rely on knowledge of, or accurate estimation of, the joint distribution of the proteins. At  $p \approx 10,000$  and  $n = 512$ , estimating this distribution reliably is challenging, and approximate constructions based on sparse covariance estimation may introduce errors that compromise the FDR guarantees. With larger cohorts, the distributional estimation becomes more reliable, making knockoff filters a natural and statistically rigorous complement to the framework developed here.

### Covariate-adjusted protein selection

None of the selection methods evaluated here adjust for clinical covariates such as ICU site, age, or comorbidities during the selection step. Covariate-adjusted selection, for example through conditional mutual information, covariate-adjusted regression, or propensity score matching prior to selection, would strengthen the causal interpretation of identified proteins by separating ARDS-specific associations from confounded ones. A natural starting point for differential expression specifically is *limma* [Smy04, RPW<sup>+</sup>15], a widely used linear-model framework with empirical Bayes moderation of variance that accommodates clinical covariates by construction through its design-matrix formulation. Benchmarking new methods against *limma* rather than the simpler  $t$ -test would provide a more demanding and clinically realistic baseline.

## 12 Conclusions

This thesis introduces a unified evaluation framework for proteomic feature selection, enabling controlled and reproducible comparison of arbitrary selection methods on the same cohort, splits, classifiers, and validation criteria. By decoupling the selection step from downstream evaluation, the framework makes method-specific strengths and blind spots directly comparable, a necessary condition for drawing meaningful conclusions from high-dimensional proteomics data in critical care.

The empirical results reveal a clear pattern of complementary limitations. The  $t$ -test reliably identifies linear (at least for normal distributed simulation data), mean-shift signals and provides stable, interpretable selections, but is blind to non-linear and non-mean-shift associations. RF+SHAP delivers the strongest classifier performance across all conditions, suggesting that ARDS-associated protein signals are at least partially non-linear in nature, but produces minimal enrichment signal, indicating that the selected proteins may not correspond to well-annotated pathway structures in current databases. The current RF+SHAP implementation also lacks a native selection criterion with FDR control, relying on stability selection as a practical substitute. Mutual information is the only method with broad biological coherence, identifying ketone body metabolism and mitochondrial processes consistent with sepsis pathophysiology, but is severely power-limited at  $n \approx 500$  and fails on two fronts: it identifies no significantly regulated proteins, and the protein sets it does select do not reliably outperform random selection in classification. No method recovers pure interaction signals, confirming that all evaluated approaches operate at the univariate level.

These findings do not immediately change clinical practice, nor are they intended to. ARDS remains a syndrome defined by clinical criteria rather than molecular mechanisms, and the proteomic signals identified here are early-stage hypotheses rather than validated biomarkers. What the results do demonstrate is that univariate differential expression analysis alone is insufficient for proteomic biomarker discovery in critical care: the choice of selection method materially changes which proteins are identified, and no single method captures the full complexity of the proteomic signal. As understanding of ARDS biology deepens and larger cohorts become available, the proteins and pathways identified here, particularly the convergent metabolic signals from mitochondrial dysfunction and ketone body metabolism, may inform more targeted hypotheses for experimental validation, and ultimately contribute to better stratification and treatment of patients in the ICU.

An important open question is whether higher-order protein interactions exist in ARDS that current methods are simply unable to detect. All methods evaluated here are fundamentally univariate in their selection criterion, and the XOR simulation confirms that purely interaction-driven signals remain invisible regardless of method sophistication. If ARDS is indeed driven by combinations of dysregulated proteins acting together, as the endotype literature suggests, then the field may need selection methods that explicitly target synergistic effects, larger cohorts to power them, and richer validation frameworks that go beyond binary classification. The proteome contains the molecular record of what happens to a critically ill patient; reading that record fully remains an open and important problem.

## References

- [Bat94] R. Battiti. Using mutual information for selecting features in supervised neural net learning. *IEEE Transactions on Neural Networks*, 5(4):537–550, 1994.
- [BBH<sup>+</sup>02] David Brealey, Michael Brand, Iain Hargreaves, Simon Heales, John Land, Ryszard Smolenski, Nathan A Davies, Chris E Cooper, and Mervyn Singer. Association between mitochondrial dysfunction and severity and outcome of septic shock. *The Lancet*, 360(9328):219–223, 2002.
- [BC91] Julian Besag and Peter Clifford. Sequential Monte Carlo  $p$ -values. *Biometrika*, 78(2):301–304, 1991.
- [BC19] Rina Foygel Barber and Emmanuel J Candès. A knockoff filter for high-dimensional selective inference. 2019.
- [BdH74] A. A. Balkema and L. de Haan. Residual Life Time at Great Age. *The Annals of Probability*, 2(5):792 – 804, 1974.
- [BGL11] Albert-László Barabási, Natali Gulbahce, and Joseph Loscalzo. Network medicine: a network-based approach to human disease. *Nature reviews genetics*, 12(1):56–68, 2011.
- [BH95] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)*, 57(1):289–300, 1995.
- [Bre73] Richard P. Brent. *Algorithms for Minimization Without Derivatives*. Prentice-Hall, Englewood Cliffs, NJ, 1973.
- [Bre01] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.
- [BW22] Lieuwe D. J. Bos and Lorraine B. Ware. Acute respiratory distress syndrome: causes, pathophysiology, and phenotypes. *The Lancet*, 400(10358):1145–1156, 2022.
- [CDP<sup>+</sup>14] Carolyn S Calfee, Kevin Delucchi, Polly E Parsons, B Taylor Thompson, Lorraine B Ware, and Michael A Matthay. Subphenotypes in acute respiratory distress syndrome: latent class analysis of data from two randomised controlled trials. *The Lancet Respiratory Medicine*, 2(8):611–620, 2014.
- [CG16] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.
- [CHB<sup>+</sup>26] Tianqi Chen, Tong He, Michael Benesty, Vadim Khotilovich, Yuan Tang, Hyunsu Cho, Kailong Chen, Rory Mitchell, Ignacio Cano, Tianyi Zhou, Mu Li, Junyuan Xie, Min Lin, Yifeng Geng, Yutian Li, Jiaming Yuan, and David Cortes. *xgboost: Extreme Gradient Boosting*, 2026. R package version 3.3.0.0.
- [CT06] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, Hoboken, NJ, 2nd edition, 2006.
- [CTK<sup>+</sup>13] Edward Y Chen, Christopher M Tan, Yan Kou, Qiaonan Duan, Zichen Wang, Gabriela Vaz Meirelles, Neil R Clark, and Avi Ma’ayan. Enrichr: interactive and collaborative html5 gene list enrichment analysis tool. *BMC bioinformatics*, 14(1):128, 2013.
- [Ern04] Michael D. Ernst. Permutation methods: A basis for exact inference. *Statistical Science*, 19(4):676–685, 2004.
- [FLP23] Zhuoqing Fang, Xinyuan Liu, and Gary Peltz. Gseapy: a comprehensive package for performing gene set enrichment analysis in python. *Bioinformatics*, 39(1):btac757, 2023.
- [GLL<sup>+</sup>23] Rui Gong, Hong Luo, Gangyu Long, Jiqian Xu, Chaolin Huang, Xin Zhou, You Shang, and Dingyu Zhang. Integrative proteomic profiling of lung tissues and blood in acute respiratory distress syndrome. *Frontiers in Immunology*, 14:1158951, 2023.

- [Goo05] Phillip I Good. *Permutation, Parametric, and Bootstrap Tests of Hypotheses*. Springer Series in Statistics. Springer-Verlag, New York, 3rd edition, 2005.
- [HG18] Jesse Hemerik and Jelle Goeman. Exact testing with random permutations. *TEST*, 27(4):811–825, 2018.
- [HSL09] Da Wei Huang, Brad T Sherman, and Richard A Lempicki. Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists. *Nucleic acids research*, 37(1):1–13, 2009.
- [III75] James Pickands III. Statistical Inference Using Extreme Order Statistics. *The Annals of Statistics*, 3(1):119 – 131, 1975.
- [JWH<sup>+</sup>23] Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, and Jonathan Taylor. *An Introduction to Statistical Learning: with Applications in Python*. Springer Nature, 2023.
- [KCTR15] Kai Kammers, Robert N Cole, Calvin Tiengwe, and Ingo Ruczinski. Detecting significant changes in protein abundance. *EuPA open proteomics*, 7:11–19, 2015.
- [KJR<sup>+</sup>16] Maxim V Kuleshov, Matthew R Jones, Andrew D Rouillard, Nicolas F Fernandez, Qiaonan Duan, Zichen Wang, Simon Koplev, Sherry L Jenkins, Kathleen M Jagodnik, Alexander Lachmann, et al. Enrichr: a comprehensive gene set enrichment analysis web server 2016 update. *Nucleic acids research*, 44(W1):W90–W97, 2016.
- [KL87] L. F. Kozachenko and N. N. Leonenko. Sample estimate of the entropy of a random vector. *Problemy Peredachi Informatsii*, 23(2):9–16, 1987.
- [KSG04] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. Estimating mutual information. *Phys. Rev. E*, 69:066138, Jun 2004.
- [LDEC02] Thomas Lumley, Paula Diehr, Scott Emerson, and Lu Chen. The importance of the normality assumption in large public health data sets. *Annual review of public health*, 23(1):151–169, 2002.
- [LEC<sup>+</sup>20] Scott M Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. From local explanations to global understanding with explainable ai for trees. *Nature machine intelligence*, 2(1):56–67, 2020.
- [LF19] Marta Lualdi and Mauro Fasano. Statistical analysis of proteomics data: A review on feature selection. *Journal of Proteomics*, 198:18–26, 2019. 10 Year Anniversary of Proteomics.
- [LK13] Jaesung Lee and Dae-Won Kim. Feature selection for multi-label classification using multivariate mutual information. *Pattern Recognition Letters*, 34(3):349–357, 2013.
- [LL17] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.
- [Ma’25] Ma’ayan Laboratory. Enrichr – background information. <https://maayanlab.cloud/Enrichr/>, 2025. Accessed: 2025.
- [MB10] Nicolai Meinshausen and Peter Bühlmann. Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(4):417–473, 2010.
- [McG54] William J. McGill. Multivariate information transmission. *Psychometrika*, 19(2):97–116, 1954.
- [MW47] Henry B Mann and Donald R Whitney. On a test of whether one of two random variables is stochastically larger than the other. *The annals of mathematical statistics*, pages 50–60, 1947.
- [NB03] Claude Nadeau and Yoshua Bengio. Inference for the generalization error. *Machine Learning*, 52(3):239–281, 2003.

- [Pit37] E. J. G. Pitman. Significance tests which may be applied to samples from any populations. *Supplement to the Journal of the Royal Statistical Society*, 4(1):119–130, 1937.
- [PLD05] Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, 2005.
- [PS10a] Fortunato Pesarin and Luigi Salmaso. *Permutation tests for complex data: theory, applications and software*. John Wiley & Sons, 2010.
- [PS10b] Belinda Phipson and Gordon K. Smyth. Permutation p-values should never be zero: Calculating exact p-values when permutations are randomly drawn. *Statistical Applications in Genetics and Molecular Biology*, 9(1):1–16, 2010.
- [PVG<sup>+</sup>11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [RIV<sup>+</sup>19] Jüri Reimand, Ruth Isserlin, Veronique Voisin, Mike Kucera, Christian Tannus-Lopes, Asha Rostamianfar, Lina Wadi, Mona Meyer, Jeff Wong, Changjiang Xu, et al. Pathway enrichment analysis and visualization of omics data using g: Profiler, gsea, cytoscape and enrichmentmap. *Nature protocols*, 14(2):482–517, 2019.
- [Ros14] Brian C. Ross. Mutual information between discrete and continuous data sets. *PLoS ONE*, 9(2):e87357, 2014.
- [RPW<sup>+</sup>15] Matthew E. Ritchie, Belinda Phipson, Di Wu, Yifang Hu, Charity W. Law, Wei Shi, and Gordon K. Smyth. limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Research*, 43(7):e47, 2015.
- [RRTT<sup>+</sup>12] V Marco Ranieri, Gordon D Rubenfeld, B Taylor Thompson, Niall D Ferguson, Ellen Caldwell, Eddy Fan, Luigi Camporota, and Arthur S Slutsky. Acute respiratory distress syndrome: the berlin definition. *JAMA: Journal of the American Medical Association*, 307(23), 2012.
- [SDS<sup>+</sup>16] Mervyn Singer, Clifford S. Deutschman, Christopher W. Seymour, Manu Shankar-Hari, Djillali Annane, Michael Bauer, Rinaldo Bellomo, Gordon R. Bernard, Jean-Daniel Chiche, Craig M. Coopersmith, Richard S. Hotchkiss, Mitchell M. Levy, John C. Marshall, Greg S. Martin, Steven M. Opal, Gordon D. Rubenfeld, Tom van der Poll, Jean-Louis Vincent, and Derek C. Angus. The Third International Consensus Definitions for sepsis and septic shock (Sepsis-3). *JAMA*, 315(8):801–810, 2016.
- [Sim07] Richard Simon. *Resampling Strategies for Model Assessment and Selection*, pages 173–186. Springer US, Boston, MA, 2007.
- [Sin14] Mervyn Singer. The role of mitochondrial dysfunction in sepsis-induced multi-organ failure. *Virulence*, 5(1):66–72, 2014.
- [Smy04] Gordon K. Smyth. Linear models and empirical Bayes methods for assessing differential expression in microarray experiments. *Statistical Applications in Genetics and Molecular Biology*, 3(1):Article 3, 2004.
- [Som23] SomaLogic. SomaScan 11K Assay v5.0 Technical Note. Technical Report SL00000919 Rev 1, SomaLogic Operating Co., Inc., Boulder, CO, 2023.
- [SP23] Calfee CS, Sinha P, Meyer NJ. Biological phenotyping in sepsis and ARDS. *Annual Review of Medicine*, 74:457–471, 2023.
- [SR15] Takaya Saito and Marc Rehmsmeier. The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. *PloS one*, 10(3):e0118432, 2015.
- [USW<sup>+</sup>06] Tokujiro Uchida, Madoka Shirasawa, Lorraine B Ware, Katsuo Kojima, Yutaka Hata, Koshi Makita, Gabe Mednick, Zachary A Matthay, and Michael A Matthay. Receptor

for advanced glycation end-products is a marker of type i cell injury in acute lung injury. *American journal of respiratory and critical care medicine*, 173(9):1008–1015, 2006.

- [VE13] Jorge R. Vergara and Pablo A. Estévez. A review of feature selection methods based on mutual information. *Neural Computing and Applications*, 24(1):175–186, March 2013.
- [VE14] Jorge Vergara and Pablo Estevez. A review of feature selection methods based on mutual information. *Neural Computing and Applications*, 24, 01 2014.
- [VGO<sup>+</sup>20] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.
- [Wel47] Bernard L Welch. The generalization of ‘student’s’ problem when several different population variances are involved. *Biometrika*, 34(1-2):28–35, 1947.
- [WS18] Stephane Wenric and Ruhollah Shemirani. Using supervised learning methods for gene selection in rna-seq case-control studies. *Frontiers in Genetics*, Volume 9 - 2018, 2018.
- [XL01] Qing-Song Xu and Yi-Zeng Liang. Monte carlo cross validation. *Chemometrics and Intelligent Laboratory Systems*, 56(1):1–11, 2001.
- [ZWL<sup>+</sup>22] Yan Zhang, Qiong Wang, Jingqi Liang, Liang Liu, Peilong Liu, and Hongmou Zhao. Serum proteomic analysis of differentially expressed proteins and pathways involved in the mechanism of endemic osteoarthritis. *Molecular Omics*, 18(8):745–753, 2022.

# Populärvetenskaplig Sammanfattning

## Matematiken bakom morgondagens medicin

**Tänk om vi en dag kunde läsa av vad som händer i kroppen på molekylnivå och utifrån det förstå hur sjukdomar som ARDS uppstår och kan behandlas. Det är drömscenariot för forskningen om livshotande tillstånd som ännu saknar riktade behandlingar. Men vägen dit går genom matematiken.**

Vi närmar oss tre år sedan covid-19-pandemin, och även om minnena av karantän och distansundervisning börjar blekna får vi inte glömma det som drabbade så många: lungtillståndet ARDS, akut andningssvikt. ARDS var den vanligaste dödsorsaken hos covid-patienter, och tillståndet drabbar än idag över tre miljoner människor globalt varje år, med en dödlighet på 30–40 procent. Trots decennier av forskning finns ingen riktad behandling. Vården kan ge stöd, men inte bota.

Idag kan vi mäta över 10 000 proteiner från ett enda blodprov, något som var helt otänkbart för 20 år sedan. Till skillnad från DNA, som är relativt oförändrat genom livet, är proteinerna en ögonblicksbild av vad kroppen håller på med just nu. Detta är vad fältet proteomik handlar om, och förhoppningen är att det ska kunna avslöja vilka molekylära mönster som ligger bakom sjukdomar som ARDS.

Men att hitta de avgörande proteinerna bland tiotusen är som att leta efter en nål i en höstack. Vi har analyserat proteinprofilen hos 512 sepsispatienter från intensivvårdsavdelningar i Skåne, varav 73 utvecklade ARDS, och frågat oss vilka proteiner som faktiskt skiljer grupperna åt.

Varför löser då inte AI bara problemet? Med 512 patienter och 10 000 variabler är risken stor att en modell hittar mönster som inte finns på riktigt. Hela vårt arbete handlar om att navigera det problemet på ett rigoröst sätt.

Vi har jämfört flera analysmetoder, både klassiska statistiska metoder och modernare maskininlärning, och visat att ingen enskild metod löser hela problemet. Varje metod har sina styrkor och blindas fläckar, men tillsammans bidrar de med olika ledtrådar. En av de stora frågorna är om man kan hitta kombinationer av proteiner som tillsammans avslöjar sjukdomen, även när inget enskilt protein gör det. Det är matematiskt svårare men biologiskt mer realistiskt, eftersom proteiner sällan arbetar ensamma utan i komplexa nätverk.

En central del av vårt arbete är ett ramverk för att rättvist utvärdera olika metoder. Det är en utmaning i sig: när man söker efter okända samband finns det inget facit att jämföra mot, så hur vet man att en metod faktiskt hittar rätt? Vi adresserar detta genom att kombinera simulerade data med kända svar, klassificering av verkliga patienter, och biologisk tolkning av resultaten. Ramverket är generellt och kan användas för att utvärdera vilken proteinurvalsmetod som helst.

Frågan kvarstår: finns det en förklaring gömd i den komplexa proteinvärlden? Våra resultat tyder på att det gör det. Vägen dit kräver större patientkohorter, smartare metoder och tålamod, men varje steg framåt tar oss närmare en bättre förståelse för vad som driver tillståndet, och därmed bättre behandlingsmöjligheter.

Master's Theses in Mathematical Sciences 2026:E61  
ISSN 1404-6342  
LUTFMS-3558-2026  
Mathematical Statistics  
Centre for Mathematical Sciences  
Lund University  
Box 118, SE-221 00 Lund, Sweden  
<http://www.maths.lu.se/>