An optimised ensemble for antibody-mediated rejection status prediction in kidney transplant patients

Mariel Barbachan e Silva
School of Mathematics, Statistics and Applied Mathematics
National University of Ireland, Galway
Galway, Ireland
m.barbachanesilva2@nuigalway.ie

Pilib Ó Broin
School of Mathematics, Statistics and Applied Mathematics
National University of Ireland, Galway
Galway, Ireland
pilib.obroin@nuigalway.ie

***Abstract*—Antibody-mediated rejection (AMR) is one of the primary mechanisms of graft loss following organ transplantation. A key difficulty with AMR diagnosis is that symptoms typically manifest when the graft is already damaged beyond repair. Diagnosis is also complicated by differing interpretations of histological data by pathologists, highlighting the urgent need for more quantitative approaches. In this paper we propose an ensemble classifier approach to predicting AMR status from gene expression data. We employ two random oversampling techniques - Synthetic Minority Oversampling Technique (SMOTE) and Adaptive Synthetic Oversampling (ADASYN) - to address the class imbalance in the original data set, and use particle swarm optimisation (PSO) for the selection of the ensemble hyperparameters. Our results demonstrate that applying the PSO-optimised ensemble to the balanced data set provides better predictive performance than the ensemble alone, and represents an important step towards more accurate sub-clinical prediction of AMR status and improved patient risk stratification.**

***Index Terms*—ensemble learning, particle swarm optimisation, risk prediction, gene expression**

I. Introduction

Antibody-mediated rejection (AMR) is a major mechanism of graft loss following organ transplantation. It can be stratified into three distinct subtypes – hyperacute AMR, acute AMR, and chronic AMR (CAMR). Hyperacute AMR can occur within minutes of transplantation and is associated with the presence of pre-existing anti-HLA (human leukocyte antigen) donor-specific antibodies (DSAs), which are generated by the recipient's immune system against donor cells [1], [2]. Improvements in pre-transplantation screening have resulted in a significantly reduced prevalence of hyperacute AMR. Long-term graft survival rates however, have not seen the same improvements; this is attributed mainly to acute AMR, which usually develops within the first six months following transplantation (often as a result of de novo DSAs) [3], [4], and chronic AMR (CAMR), which can develop anywhere from several months to several decades after transplantation [4]. A key difficulty with AMR is that clinical diagnosis typically occurs after the patient has presented with certain physiological symptoms, by which stage damage to the graft may already be too significant for it to be maintained. Diagnosis of AMR is also complicated by the fact that differing interpretations of histological data by different pathologists can result in contradictory diagnosis [5], [6] highlighting the urgent need for more objective quantitative approaches.

High-throughput gene expression profiling of graft biopsies can provide evidence of AMR before a clinical phenotype becomes apparent [7], [8] and Halloran et al. [9] have previously demonstrated promising results using a linear discriminant analysis (LDA) to predict AMR status in kidney transplant patients from such data. This has lead to increased interest in the application of machine learning approaches to this problem, with recent work focusing on the performance of various classification algorithms [9]–[11]. Any individual classifier however, will have its own inherent biases, so heterogeneous ensemble approaches are often employed to provide more stable predictive performance by combining decisions from multiple algorithms [12]. Finding an optimal combination of hyperparameters for such an ensemble can however be challenging, and metaheuristic approaches are often employed for efficient traversal of complex search spaces [13]–[15]. Swarm intelligence (SI) [16] refers to a set of metaheuristic optimisation techniques inspired by the collective behaviour of social animals. These algorithms have become very popular in recent years due to their efficiency in solving complex optimisation problems [17], [18]. Introduced by Kennedy and Eberhart, particle swarm optimisation (PSO) [19] is one such metaheuristic approach which is characterised by its socio-cognitive approach to combining exploration and exploitation in the search space.

In this work, we describe a PSO-optimised ensemble classifier comprising five supervised learning algorithms for AMR status prediction in kidney transplant patients. The remainder of the paper is organised as follows: the materials and methods section provides details on the data set used, the constituent algorithms of the ensemble classifier and the PSO algorithm implementation for hyperparameter selection; the experiments and results section describes the performance of the PSO and

MBS is supported by a College of Science fellowship from NUI Galway

978-1-7281-6929-3/20/$31.00 ©2020 IEEE

compares it to both the individual algorithms used and the ensemble classifier using a number of standard metrics, and the discussion section provides a summary of results and their significance, as well as an outline of study limitations and potential directions for future work.

II. Materials and Methods

A. The data set

We performed the experiments using log_2 quantile normalised microarray expression data from the Halloran et al. study [9] which is publicly available from the Gene Expression Omnibus (GEO) repository [20] under accession number GSE36059. This data set contains 409 samples with 54675 features for each sample. Sample numbers, grouped according to their rejection status, are shown in Table I below. In addition to the graft rejection samples – AMR, T-cell mediated rejection (TMR), and MIXED (a mixture of early AMR and TMR) – the data set is composed of a large number of samples from patients not showing graft rejection (non-rejecting), as well as a small number of samples that may or may not harbour physiological abnormalities related to the rejection process itself (Nephrectomy). When training our classifier, we follow the strategy of Halloran et al. [11], who have previously demonstrated that better results are obtained for AMR classification when TMR samples are included with the samples labelled as AMR-negative. This is likely due to TMR's stronger transcriptional signal as well as fundamental differences in the biological pathways activated by the two rejection mechanisms.

TABLE I
DATA SET GROUPED BY REJECTION STATUS

Group	Sample size
AMR	**65 (15.89%)**
TMR	35 (8.56%)
$MIXED$	22 (5.38%)
$Non-rejecting$	279 (68.22%)
$Nephrectomy$	8 (1.96%)
$Total$	409

From Table I, it is clear that the data set is highly imbalanced with AMR representing less than 16% of the total number of samples. As an imbalanced data set can have significant consequences for the learning process and ultimately generate inaccurate classification results (particularly for the minority class), we used two random over-sampling approaches to synthetically balance the data set – Synthetic Minority Oversampling Technique (SMOTE) [21] and Adaptive Synthetic (ADASYN) sampling approach [22]. SMOTE uses a k-nearest neighbours (KNN) approach to generate synthetic data for each feature of the minority class and has previously been applied to imbalanced biological data sets [23], [24]. Also using a KNN approach, ADASYN adaptively generates synthetic data based on the density distribution of the ratio between majority and minority classes. Both oversampling techniques were applied to the minority (AMR) class using the imbalanced-learn package [25] in Python. After the oversampling step, the balanced data sets were stratified as follows: SMOTE – 344 AMR samples, 344 Non-AMR samples; ADASYN – 353 AMR samples, 344 Non-AMR samples. The original and synthetically balanced data were divided into three subsets: i) training set containing 50% of the data, ii) validation set containing 25% of the data, and iii) test set containing the remaining 25% of data (see Figure 1).

B. Ensemble Learning

Given a training set in the form $\{(x_1, y_1), (x_2, y_2), ... ,(x_n, y_n)\}$ where x_i represents the feature vector for the i^{th} sample, y_i represents the class label assigned to that sample, and n is the number of samples, a supervised classification algorithm learns a class mapping function, $y = f(x)$, which allows it to predict class labels for new, unseen data, as well as to assign significance to individual features which are important for the prediction. As every classification algorithm will have a non-zero error value related to its predictive performance, and as this error can be uncorrelated between algorithms [26], heterogeneous ensemble learning is often used to combine decisions from a diverse set of algorithms (i.e. with uncorrelated predictive errors) in order to increase the overall performance of the model by learning a more complex class-mapping function. We produced such an ensemble comprised of five popular supervised classification approaches from the Python scikit-learn package [27]: i) support vector machine (SVM), ii) logistic regression (LR), iii) random forest (RF), iv) linear discriminant analysis (LDA), and v) k-nearest neighbours (KNN). The classification decisions of these algorithms were combined in the ensemble using soft voting on the predictive probabilities of AMR status.

Each of these algorithms has an associated set of hyperparameters that must be chosen before the learning process begins and these hyperparameters are known to have critical influence on the algorithm performance; hyperparameter optimisation (HPO) is therefore a crucial part of the machine learning process and in tailoring the algorithm to the specific problem being addressed [28]. Stochastic algorithms can be used to discover quality solutions for complex optimisation problems where often the exact solution is impossible to determine [29]. In this work we used a particle swarm optimisation algorithm to select hyperparameter values for the ensemble; this approach is described in more detail in the next section.

Table II shows the hyperparameters optimised for each individual algorithm. *C* is a regularisation parameter which controls the balance between producing a low training error and a low test error, *max_iter* is the maximum number of iterations of the LR algorithm, *n_estimators* is the number of trees generated in the random forest algorithm, *tol* is the tolerance used for rank estimation of the singular value decomposition used in LDA, and *n_neighbors* is the number of neighbours to be considered when using the KNN approach. For non-

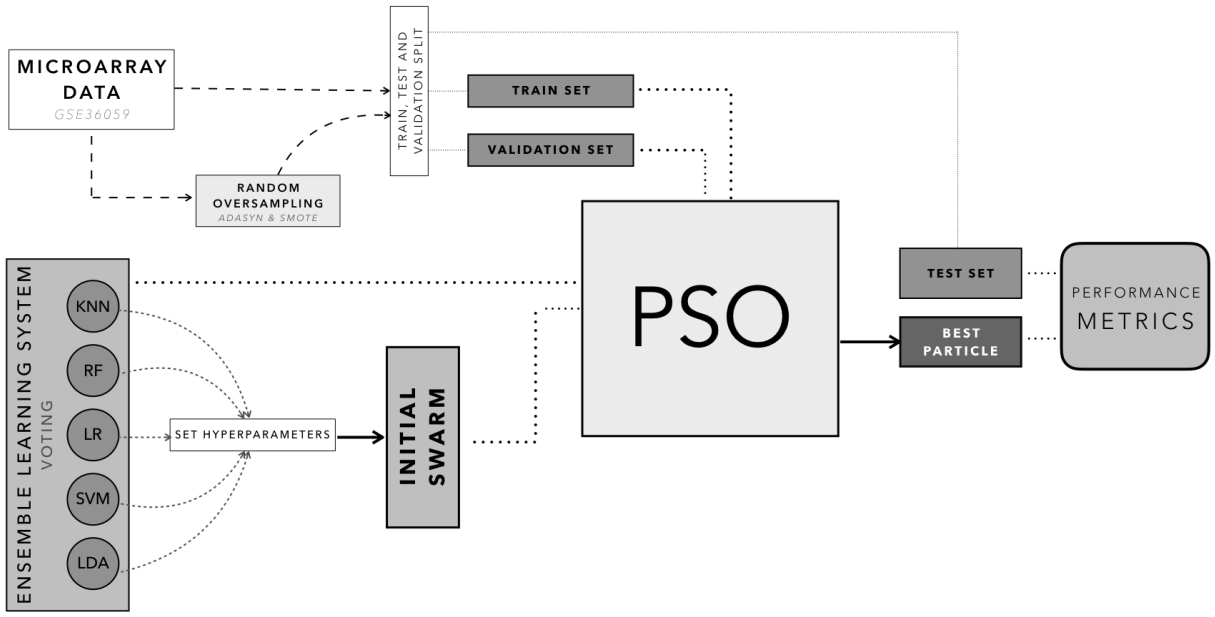

Fig. 1. Workflow for the proposed method. Original microarray data and randomly oversampled data using ADASYN and SMOTE were divided into train, validation, and test sets (ratio of 50:25:25). The training and validation data were used by the PSO. The initial swarm contained randomly generated sets of hyperparameters for each of the five supervised learning algorithms in the ensemble – k-nearest neighbours (KNN), random forest (RF), logistic regression (LR), support vector machine (SVM) and linear discriminant analysis (LDA). Performance metrics were calculated using 10-fold cross validation on the test set and the best particle from the final swarm.

TABLE II
OPTIMISED ENSEMBLE HYPERPARAMETERS

Supervised Learning Algorithm	Hyperparameters
SVM	C
LR	max_iter, C
RF	n_estimators
LDA	tol
KNN	n_neighbors

numerical hyperparameters, we conducted a separate analysis of each of the individual algorithms and chose hyperparameter values that provided the best predictive performance for use in the ensemble.

C. Particle Swarm Optimisation

The PSO algorithm uses a population (*swarm*) of candidate solutions (*particles*) that consist of elements representing the hyperparameters to be optimised. These particles move in the search space guided by a fitness function and their velocity is dependent on components with stochastic factors. Their movement is governed both by their own best position in the search space – cognitive component – as well as the best solution of the swarm – social component.

In our PSO implementation, we used a swarm of 60 particles, where each particle contained the following information: i) a set of hyperparameter values for each of the supervised learning algorithms (see Table II), ii) its current score, iii) its current velocity, and vi) the set of hyperparameter values associated with the particle's individual best score (i.e. memory). A global best topology was used for optimisation, which means that all particles were able to see the entire swarm, which was guided by a unique best particle.

For each iteration of the PSO, the j^{th} set of hyperparameters of particle i, the velocity $v_{i,j}$ (Eq. 1) and the position $x_{i,j}$ were updated (Eq. 2) taking into consideration: y, the particle's best position and g, the global best position. There is a probability of finding a better individual solution (tested on Algorithm 1, line 11) or a better global solution (tested on Algorithm 1, line 15), resulting in updates to the cognitive term of the velocity (Algorithm 1, line 18) and to the social term of the velocity (tested on Algorithm 1, line 19), respectively.

$$v_{i,j}(t+1) = wv_{i,j}(t) + c_1 r_{1,j}(t)(y_{i,j}(t) - x_{i,j}(t)) +$$

$$c2r_{2,j}(t)(g_{i,j}(t) - x_{i,j}(t)) \quad (1)$$

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1) \quad (2)$$

where w is the inertia weight, $c1$ and $c2$ are the social cognitive and social acceleration rates respectively, and $r_{1,j}$ and $r_{2,j}$ follow a $U(0,1)$ distribution.

Data: Normalised microarray expression data.
Result: The best particle

```
particle ← randomly generated set of
  hyperparameters;
c1 = 1.494;
c2 = 1.494;
w = 0.729;
stop criterion = max iterarions;
gbest ← random particle in the swarm;
while ! stop criterion do
    for p in Swarm do
        for j in set of hyperparameters do
            currentScore ← Score(p,j);
            if currentScore < p.score then
                p.fitness ← currentScore;
                p.bestPosition ← p.currentPosition;
            end
            if currentScore < gBest.Score then
                gBest ← i;
            end
            cognitive ←
              p, j.bestPosition − p, j.currentPosition;
            social ←
              gBest, j.position − p, j.currentPosition;
            C_term ← (c1 ∗ rand ∗ cognitive);
            S_term ← (c2 ∗ rand ∗ social);
            v_{t,i,j} ← w ∗ v_{t−1,i,j} + C_term + S_term;
            x_{t,i,j} = x_{t−1,i,j} + v_{t,i,j};
            i.actualPosition[j] ← position_{i,j};
        end
    end
end
return gBest
```

Algorithm 1: PSO approach for ensemble HPO

1) The fitness function: The performance of the PSO is intimately related to the selection of a suitable fitness function. Considering that the original data is imbalanced, using a function that evaluates performance based on binary class assignment may be misleading. Additionally, developing an algorithm for use with clinical data requires cognisance of a particular responsibility with regard to high confidence statements. Therefore, we used a log-loss function (equation 3) to evaluate the candidate solutions. The log-loss function is chosen because it more heavily penalises misclassification and also results in predictive probabilities rather than hard class assignments, providing a more natural framework for risk stratification. The log-loss is defined as follows:

$$logLoss = -\frac{1}{n}\sum_{i=1}^{n}[y_i log_e(p_i) + (1-y_i)log_e(1-p_i)] \quad (3)$$

where y_i is the true class label of observation i, p_i is the predictive probability assigned to observation i, and n is the number of samples.

From an information theory perspective, the log-loss function is the cross-entropy between the probability distribution of the predicted class labels and the actual class labels. The PSO particles traverse the search space guided by the minimisation of the log-loss values.

III. Experiments and Results

We compared the predictive performance of our PSO with both the individual classification algorithms and with the ensemble learning approach. We began by splitting the original data set and the balanced data set created using the SMOTE and ADASYN algorithms into the three data sets described in Section II-A. We then used the training and validation sets to minimise the log-loss function using the PSO with an initial swarm composed of randomly initiated hyperparameter values for each of the supervised learning algorithms described in Table II. We measured predictive performance using stratified 10-fold cross validation on the test data set, which was composed of data from the original and oversampled data that were not used in the optimisation process. We compared the hyperparameter values of the best particle in the swarm with the default hyperparameter values from scikit-learn – this latter approach will be referred to as *ensemble* hereafter. We evaluated the following metrics: log-loss, accuracy, F1-score, and area under the receiver operating characteristic curve (AUC-ROC).

TABLE III
Mean log-loss values and standard deviation from stratified 10-fold cross validation.

	Imbalanced	**SMOTE**	**ADASYN**
KNN	2.10 (2.66)	5.90 (3.39)	6.1 (3.68)
LDA	0.47 (0.18)	8.97 (3.69)	2.39 (1.81)
LR	0.95 (0.76)	0.43 (0.45)	0.39 (0.52)
RF	0.44 (0.11)	0.44 (0.12)	0.41 (0.07)
SVM	0.44 (0.09)	0.63 (0.05)	0.72 (0.02)
Ensemble	0.43 (0.15)	0.44 (0.08)	0.43 (0.07)
PSO	**0.43 (0.15)**	**0.35 (0.11)**	**0.33 (0.10)**

Table III shows the mean and standard deviation for log-loss values from the stratified 10-fold cross-validation for each of the individual classifiers, for the ensemble, and for the PSO-optimised ensemble. The results indicate that: i) there was a large diversity in performance amongst the individual classifiers, with KNN and LDA showing some of the worst log-loss values, ii) in the majority of cases, the log-loss for

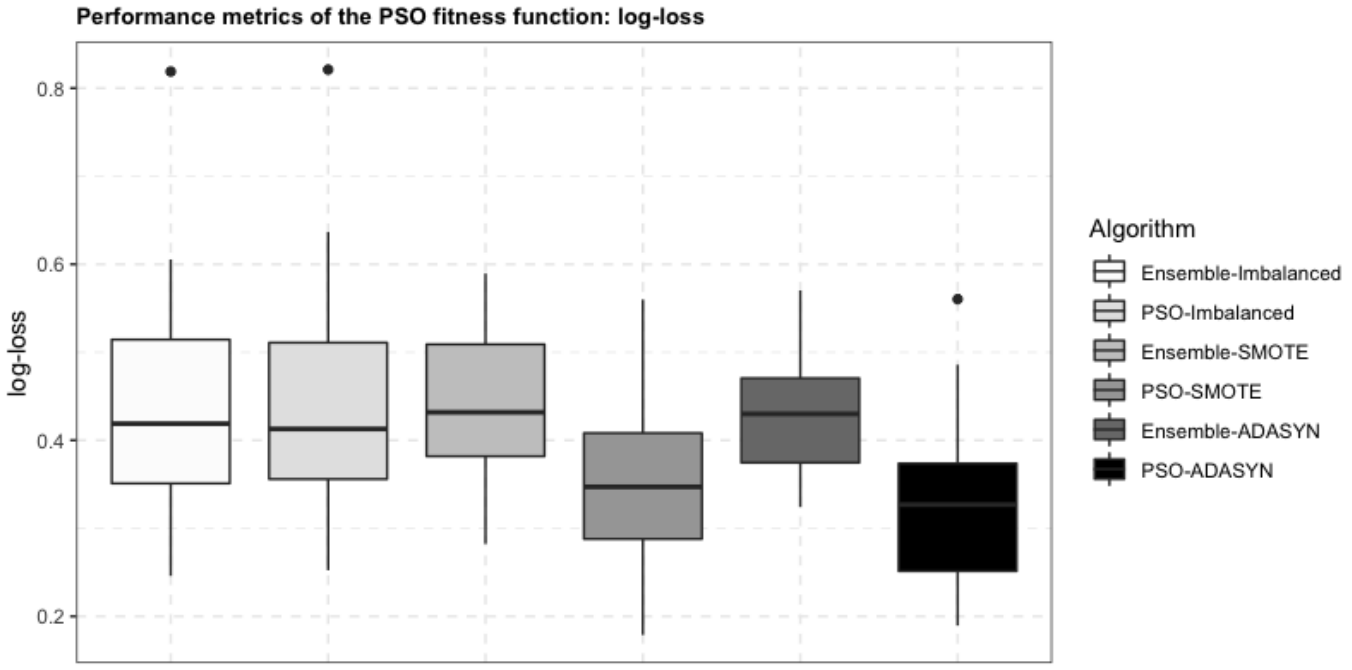

Fig. 2. Comparative performance results using 10-fold cross-validation. Box-plot representation of log-loss results shows lower values in the PSO approaches in balanced data.

TABLE IV
MEAN AND STANDARD DEVIATION ACCURACY, AUC-ROC, AND F1-SCORE VALUES FROM STRATIFIED 10-FOLD CROSS-VALIDATION.

	Imbalanced			**SMOTE**			**ADASYN**		
	Accuracy	AUC-ROC	F1-score	Accuracy	AUC-ROC	F1-score	Accuracy	AUC-ROC	F1-score
KNN	0.80 (0.13)	0.63 (0.29)	0.88 (0.08)	0.62 (0.13)	0.74 (0.16)	0.40 (0.29)	0.58 (0.05)	0.71 (0.12)	0.27 (0.14)
LDA	0.83 (0.03)	0.61 (0.32)	0.91 (0.02)	0.71 (0.10)	0.67 (0.19)	0.65 (0.13)	0.71 (0.11)	0.78 (0.13)	0.66 (0.18)
LR	0.81 (0.11)	0.64 (0.34)	0.89 (0.07)	0.88 (0.01)	0.94 (0.07)	0.87 (0.10)	0.9 (0.12)	0.95 (0.07)	0.89 (0.13)
RF	0.85 (0.04)	0.64 (0.37)	0.92 (0.02)	0.81 (0.14)	0.90 (0.11)	0.79 (0.17)	0.90 (0.10)	0.96 (0.06)	0.89 (0.11)
SVM	0.84 (0.01)	0.63 (0.38)	0.91 (0.01)	0.67 (0.11)	0.76 (0.15)	0.60 (0.16)	0.55 (0.06)	0.66 (0.17)	0.28 (0.13)
Ensemble	0.85 (0.04)	0.67 (0.30)	0.92 (0.02)	0.77 (0.12)	0.92 (0.07)	0.72 (0.16)	0.85 (0.09)	0.95 (0.06)	0.82 (0.12)
PSO	0.84 (0.01)	0.65 (0.29)	0.91 (0.01)	0.84 (0.12)	0.94 (0.07)	0.81 (0.15)	0.88 (0.12)	0.96 (0.05)	0.87 (0.14)

the combined ensemble was either lower than, or comparable to but more stable (shows reduced standard deviation) than the best individual classifier results, and iii) there was a marked improvement in the log-loss results for the PSO-optimised ensemble in the case of balanced data.

Figure 2 shows the mean and standard deviation in log-loss values for the stratified 10-fold cross validation between the ensemble and PSO approaches; statistical analysis of these results using the Wilcoxon rank test showed that the PSO approach achieved statistically significant decreases in log-loss values (p-values: SMOTE = 0.0189, ADASYN = 0.0022).

Table IV shows the mean and standard deviation values of accuracy, AUC-ROC and F1-score for the individual classifiers, for the ensemble and the PSO-optimised ensemble. These results indicate that: i) performance is similar for all approaches on the imbalanced data set (although it should be noted that, in the case of imbalanced data, higher accuracy and AUC-ROC scores can be achieved simply by classifying all samples as belonging to the majority class), ii) for the balanced data sets, the PSO approach provides better predictive performance than the ensemble in all cases, iii) using these metrics, the diversity in performance for poorly performing individual classifiers more negatively influences the results for the ensemble classifier than is apparent when using the log-loss values (Table III, where, in all but one case the ensemble showed a better predictive performance than all individual classifiers); the decrease in performance with these metrics however, is to be expected given that we are combining a

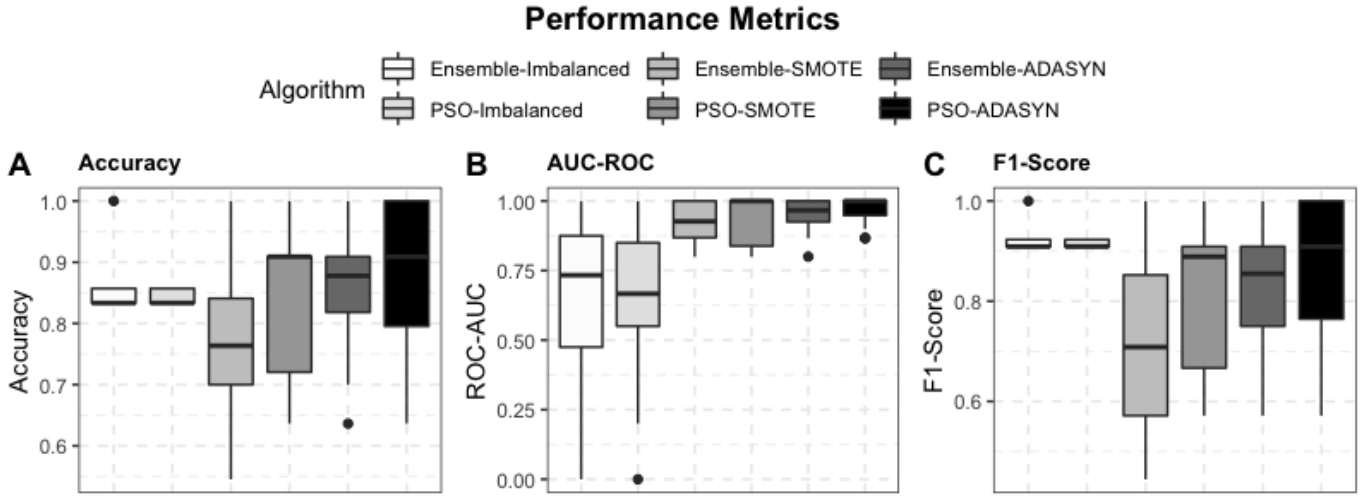

Fig. 3. Comparative performance using stratified 10-fold cross-validation. A) Boxplot representation of the accuracy results shows improvement in the performance with PSO on balanced data sets. B) AUC-ROC results indicate improvement when comparing imbalanced and balanced data, and the same can be seen in the comparison between ensemble and PSO in balanced data sets. C) F1-score results show overall better performance on the imbalanced data set and improved performance on PSO approaches when compared with the ensemble approach on both balanced data sets.

diverse set of algorithms – as indicated in the introduction section, with other data sets, individual classifier performance will likely be different and the diversity may prove more beneficial, and iv) while the fitness function for the PSO aims to minimise the log-loss value, the final results for the PSO on the other performance metrics is quite similar to the best performing individual classifiers.

Figure 3 shows accuracy, AUC-ROC, and F1-score results using the ensemble and PSO-optimised ensemble for both balanced and imbalanced data sets. It can be seen that: i) overall, the balanced data sets resulted in improved performance with the PSO approach, and ii) oversampling with ADASYN produced better results than SMOTE – this can be explained by the fact that the ADASYN algorithm is more sophisticated and adaptively balances the classes in each iteration. For the AUC-ROC, while the differences between the PSO and ensemble approaches with oversampling were not statistically significant, this is primarily due to the fact that the oversampled data results have median values close to the upper limit of the AUC.

Feature selection on the three data sets was carried out using Boruta [30] – a wrapper algorithm around the random forest approach – in order to identify genes whose expression values are important for the prediction of AMR status. Analysis of the imbalanced data resulted in 8 relevant features being selected, while analysis of the balanced data set resulted in 1180 and 1416 selected features for the SMOTE and ADASYN data sets respectively. Gene ontology (GO) analysis was carried out on the selected features to determine if these genes showed enrichment for any particular molecular function or biological pathway. The InteRmineR R package [31] was used for this analysis and returned 8 significant ontology terms which overlapped between the balanced data sets. As shown in table V, these GO terms included a number of leukocyte cell-cell adhesion terms, as well as terms related to cell surface receptor signalling – this indicates that the selected genes are involved in inflammation and immune response which is highly relevant to graft rejection conditions [32]. Features selected from the imbalanced data set did not result in any significantly enriched molecular functions or pathways.

IV. Discussion

In this work, we have developed a PSO-optimised heterogeneous ensemble of five learning algorithms for AMR status prediction based on gene expression data from kidney transplant patients. We applied particle swarm optimisation to the ensemble with the aim of improving the predictive performance through the selection of an optimal combination of hyperparameter values. The results indicated that the proposed algorithm was capable of minimising the log-loss values and increasing the predictive performance of the ensemble across all performance metrics when using balanced data sets. The gene ontology analysis of the selected features indicated that the expression values for genes involved in inflammation and immune response, cell signalling, and cell-cell adhesion are important for the classification of AMR status. Moreover, it suggests that the use of synthetic minority oversampling methods did not interfere with the fundamental signal in the data and represent an important tool for handling class imbalances in biological data sets.

The main limitations of the study are related to the data – aside from the fact that the original data set is not only small but also imbalanced, there is also inherent noise in biological data which is only amplified by the presence of five different biologically diverse groups. Future work will be carried out using a more balanced and larger AMR data set and will involve a more detailed analysis of the biological context of genes selected as important for determining class labels across

TABLE V
ENRICHED GO TERMS

Biological Process	GO Term	SMOTE (p-value)	ADASYN (p-value)
leukocyte cell-cell adhesion	GO:0007159	0.00029223	0.00268746
regulation of leukocyte cell-cell adhesion	GO:1903037	0.00024809	0.01204234
positive regulation of leukocyte cell-cell adhesion	GO:1903039	0.00275801	0.0218635
regulation of cell-cell adhesion	GO:0022407	0.0009136	0.0272042
regulation of signaling	GO:0023051	0.00327894	0.02810694
regulation of cell adhesion	GO:0030155	0.00528727	0.03137499
cellular response to chemical stimulus	GO:0070887	0.00035406	0.03145862
cell surface receptor signaling pathway	GO:0007166	0.00087517	0.04018634

all classification algorithms – this will help us to identify a robust 'core' rejection signature. As noted in the results for complementary performance metrics, the optimisation of log-loss values has not always been accompanied by improvements in the values of other metrics – future work will therefore also investigate a multi-objective PSO [33], in which the movement of the particles will be guided by more than one of the performance metrics. It is also important to recognise that the PSO itself is an algorithm with its own hyperparameters that influence the optimisation results, as such, we will also explore the use of the parameter-free PSO, as described in [34].

REFERENCES

[1] L. C. Racusena, R. B. Colvin, K. Solez, M. J. Mihatsch, P. F. Halloran, P. M. Campbell, M. J. Cecka, J. Cosyns, A. J. Demetris, M. C. Fishbein, A. Fogo, P. Furness, I. W. Gibson, D. Glotz, P. Hayry, L. Hunsickern, M. Kashgarian, R. Kerman, A. J. Magil, R. Montgomery, K. Morozumi, V. Nickeleit, P. Randhawa, H. Regele, D. Seron, S. Seshan, S. Sund, and K. Trpkov, "Antibody-Mediated Rejection Criteria – an Addition to the Banff 097 Classification of Renal Allograft Rejection," *American Journal of Transplantation*, vol. 3, pp. 708–714, 2003.

[2] R. B. Colvin and R. N. Smith, "Antibody-mediated organ-allograft rejection," *Nature Reviews Immunology*, vol. 5, no. 10, pp. 807–817, 2005.

[3] J. M. DeVos, A. O. Gaber, L. D. Teeter, E. A. Graviss, S. J. Patel, G. A. Land, L. W. Moore, and R. J. Knight, "Intermediate-term graft loss after renal transplantation is associated with both donor-specific antibody and acute rejection," *Transplantation*, vol. 97, no. 5, pp. 534–540, 2014.

[4] M. Pascual, T. Theruvath, T. Kawai, N. Tolkoff-Rubin, and A. B. Cosimi, "Strategies to improve long-term outcomes after renal transplantation," *New England Journal of Medicine*, vol. 346, no. 8, pp. 580–590, 2002.

[5] A. Loupy, C. Suberbielle-Boissel, G. Hill, C. Lefaucheur, D. Anglicheau, J. Zuber, F. Martinez, E. Thervet, A. Méjean, D. Charron *et al.*, "Outcome of subclinical antibody-mediated rejection in kidney transplant recipients with preformed donor-specific antibodies," *American Journal of Transplantation*, vol. 9, no. 11, pp. 2561–2570, 2009.

[6] C. Lefaucheur, C. Suberbielle-Boissel, G. S. Hill, D. Nochy, J. Andrade, C. Antoine, C. Gautreau, D. Charron, and D. Glotz, "Clinical relevance of preformed hla donor-specific antibodies in kidney transplantation," *American Journal of Transplantation*, vol. 8, no. 2, pp. 324–331, 2008.

[7] N. Hayde, Y. Bao, J. Pullman, B. Ye, R. Brent Calder, M. Chung, D. Schwartz, M. Lubetzky, M. Ajaimy, G. de Boccardo, and E. Akalin, "The clinical and genomic significance of donor-specific antibody-positive/C4d-negative and donor-specific antibody-negative/C4d-negative transplant glomerulopathy," *Clinical Journal of the American Society of Nephrology*, vol. 8, no. 12, pp. 2141–2148, 2013.

[8] N. Hayde, P. Ó. Broin, Y. Bao, G. De Boccardo, M. Lubetzky, M. Ajaimy, J. Pullman, A. Colovai, A. Golden, and E. Akalin, "Increased intragraft rejection-associated gene transcripts in patients with donor-specific antibodies and normal biopsies," *Kidney International*, vol. 86, no. 3, pp. 600–609, 2014. [Online]. Available: http://dx.doi.org/10.1038/ki.2014.75

[9] A. Loupy, C. Lefaucheur, D. Vernerey, J. Chang, G. Luis, T. Beuscart, J. Verine, O. Aubert, S. Dubleumortier, J.-p. D. V. Huyen, X. Jouven, D. Glotz, and C. Legendre, "Molecular Microscope Strategy to Improve Risk Strati fi cation in Early Antibody-Mediated Kidney Allograft Rejection," pp. 2267–2277, 2014.

[10] J. Reeve, G. A. Böhmig, F. Eskandary, G. Einecke, C. Lefaucheur, A. Loupy, and P. F. Halloran, "Assessing rejection-related disease in kidney transplant biopsies based on archetypal analysis of molecular phenotypes," no. 1, pp. 1–14.

[11] P. F. Halloran, "Comprehensive Analysis of Transcript Changes Associated With Allograft Rejection : Combining Universal and Selective Features."

[12] R. Polikar, "Ensemble learning," in *Ensemble machine learning*. Springer, 2012, pp. 1–34.

[13] S. Lessmann, M. Caserta, and I. M. Arango, "Tuning metaheuristics: A data mining based approach for particle swarm optimization," *Expert Systems with Applications*, vol. 38, no. 10, pp. 12 826–12 838, 2011.

[14] J.-S. Chou and A.-D. Pham, "Nature-inspired metaheuristic optimization in least squares support vector regression for obtaining bridge scour information," *Information Sciences*, vol. 399, pp. 64–80, 2017.

[15] L. Bianchi, M. Dorigo, L. M. Gambardella, and W. J. Gutjahr, "A survey on metaheuristics for stochastic combinatorial optimization," *Natural Computing*, vol. 8, no. 2, pp. 239–287, 2009.

[16] G. Beni and J. Wang, "Swarm intelligence in cellular robotic systems," in *NATO Advanced Workshop on Robotics and Biological Systems*, vol. 102, 1993, pp. 703–712.

[17] A. Garg and P. Gill, "An insight into swarm intelligence," *Int J Recent Trends Eng Tech*, vol. 2, no. 8, pp. 42–44, 2009.

[18] I. Fister Jr, X.-S. Yang, I. Fister, J. Brest, and D. Fister, "A brief review of nature-inspired algorithms for optimization," *arXiv preprint arXiv:1307.4186*, 2013.

[19] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Micro Machine and Human Science, 1995. MHS '95., Proceedings of the Sixth International Symposium on*, Oct 1995, pp. 39–43.

[20] R. Edgar, M. Domrachev, and A. E. Lash, "Gene expression omnibus: Ncbi gene expression and hybridization array data repository," *Nucleic acids research*, vol. 30, no. 1, pp. 207–210, 2002.

[21] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[22] H. He, Y. Bai, E. A. Garcia, and S. Li, "Adasyn: Adaptive synthetic sampling approach for imbalanced learning," in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*. IEEE, 2008, pp. 1322–1328.

[23] M.-F. Tsai and S.-S. Yu, "Data mining for bioinformatics: Design with oversampling and performance evaluation," *Journal of Medical and Biological Engineering*, vol. 35, no. 6, pp. 775–782, 2015.

[24] Q. Wei and R. L. Dunbrack Jr, "The role of balanced training and testing data sets for binary classifiers in bioinformatics," *PloS one*, vol. 8, no. 7, 2013.

[25] G. Lemaître, F. Nogueira, and C. K. Aridas, "Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning," *Journal of Machine Learning Research*, vol. 18, no. 17, pp. 1–5, 2017. [Online]. Available: http://jmlr.org/papers/v18/16-365.html

[26] T. G. Dietterich, "Ensemble methods in machine learning," in *International workshop on multiple classifier systems*. Springer, 2000, pp. 1–15.

[27] 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*, vol. 12, pp. 2825–2830, 2011.

[28] M. Feurer and F. Hutter, *Hyperparameter Optimization*. Cham: Springer International Publishing, 2019, pp. 3–33. [Online]. Available: https://doi.org/10.1007/978-3-030-05318-5_1

[29] X.-S. Yang, *Nature-Inspired Metaheuristic Algorithms: 2nd Edition*. Luniver Press, 2010.

[30] M. B. Kursa, W. R. Rudnicki *et al.*, "Feature selection with the boruta package," *J Stat Softw*, vol. 36, no. 11, pp. 1–13, 2010.

[31] K. A. Kyritsis, B. Wang, J. Sullivan, R. Lyne, and G. Micklem, "Interminer: an r package for intermine databases," *Bioinformatics*, vol. 35, no. 17, pp. 3206–3207, 2019.

[32] B. Sis, "Endothelial molecules decipher the mechanisms and functional pathways in antibody-mediated rejection," *Human immunology*, vol. 73, no. 12, pp. 1218–1225, 2012.

[33] C. A. Coello Coello and M. S. Lechuga, "Mopso: a proposal for multiple objective particle swarm optimization," in *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No.02TH8600)*, vol. 2, 2002, pp. 1051–1056 vol.2.

[34] A. Tangherloni, L. Rundo, and M. S. Nobile, "Proactive particles in swarm optimization: A settings-free algorithm for real-parameter single objective optimization problems," in *2017 IEEE Congress on Evolutionary Computation (CEC)*, 2017, pp. 1940–1947.