NETTAB 2017
Methods, tools & platforms for Personalized Medicine in the Big Data Era

Parameters tuning boosts hyperSMURF predictions of rare deleterious non-coding genetic variants

A. Petrini, M. Schubach, M. Re, M. Frasca, M. Mesiti, G. Grossi, T. Castrignanò, P. Robinson, G. Valentini

The Jackson Laboratory

CINECA SCAI
SuperComputing Applications and Innovation

BERLIN INSTITUTE OF HEALTH
Charité & Max Delbrück Center

Computer Science Department
UNIVERSITÀ DEGLI STUDI DI MILANO

Anacleto Lab
Computational Biology and Bioinformatics

Outline

- The detection of *deleterious* genetic variants in human genome: a key problem in Personalized and Precision Medicine
- *HyperSMURF*, an imbalance-aware ML method for detecting pathogenic variants in non-coding genome
- Tuning its learning parameters can boost *hyperSMURF* predictions
- An ongoing *HPC massively parallel implementation* of the method to automatically fit different genomic problems characterized by imbalanced big data

Prediction of pathogenic variants in non-coding genome: a challenging machine learning problem

Issues:

- How to find pathogenic variants in the sea of background (neutral) genetic variation in human genome?
- A huge imbalance between deleterious (positive examples) and neutral (negative examples) variants (e.g. 1/36000 ratio in Mendelian diseases, *Smedley et al*., 2016)
- Which features should be used to train learning machines for the prediction of pathogenic variants?

Classical ML algorithms fail:

they are biased toward the majority class

State-of-the-art ML methods for the prediction of deleterious variants

- CADD (Kircher, et al. 2014)
- GWAVA (Ritchie et al 2014)
- DeepSEA (Zhou & Troyanskaya, 2015)
- FATHMM-MKL (Shibab et al. 2015)
- Eigen (Ionita-Laza et al. 2016)

Quite surprisingly none of the above methods (apart from GWAVA) use imbalance-aware learning strategies

A first version of hyperSMURF

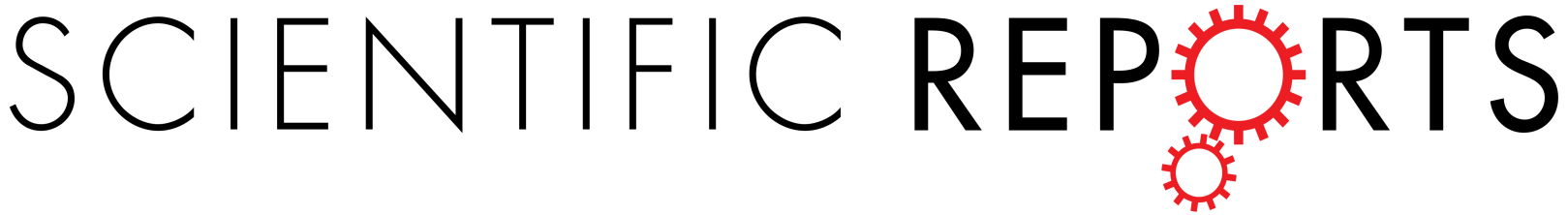
Am J Hum Genet. 2016 Sep 1; 99(3): 595–606. PMCID: PMC5011059
Published online 2016 Aug 25. doi: 10.1016/j.ajhg.2016.07.005

A Whole-Genome Analysis Framework for Effective Identification of Pathogenic Regulatory Variants in Mendelian Disease

Damian Smedley,[1,2,15] Max Schubach,[3,15] Julius O.B. Jacobsen,[4,15] Sebastian Köhler,[3] Tomasz Zemojtel,[3,5] Malte Spielmann,[3,6] Marten Jäger,[3,7] Harry Hochheiser,[8] Nicole L. Washington,[9] Julie A. McMurry,[10] Melissa A. Haendel,[10] Christopher J. Mungall,[9] Suzanna E. Lewis,[9] Tudor Groza,[11,12] Giorgio Valentini,[13] and Peter N. Robinson[3,6,7,14,16,*]

- REMM (Regulatory Mendelian Mutation Score) a first version of hyperSMURF is part of the Genomiser tool for the identification of pathogenic regulatory variants in Mendelian disease (Smedley et al. AJHG, 2016)

www.nature.com/scientificreports



OPEN

Imbalance-Aware Machine Learning for Predicting Rare and Common Disease-Associated Non-Coding Variants

Received: 17 October 2016
Accepted: 21 April 2017
Published online: 07 June 2017

Max Schubach[1], Matteo Re[2], Peter N. Robinson[1,3,4] & Giorgio Valentini[2]

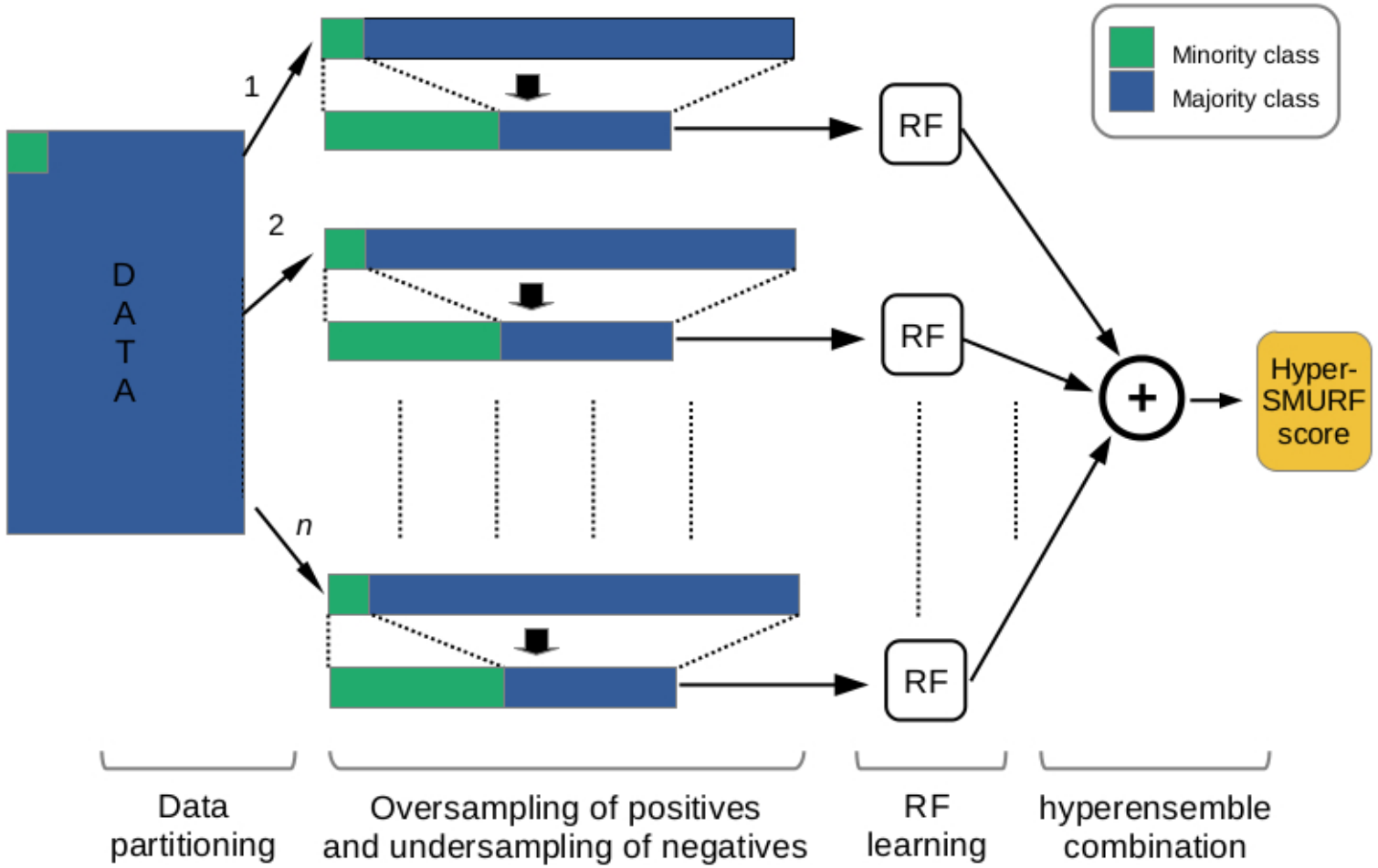
- HyperSMURF - Hyper-ensemble SMote Undersampled Random Forest: a novel multi-parametric version of the method able to fit different problems in the context of the prediction of deleterious variants

HyperSMURF - 1

A ML approach to deleterious variants detection
Hyper-ensemble of Smote Undersampled Random Forests (*HyperSMURF*)

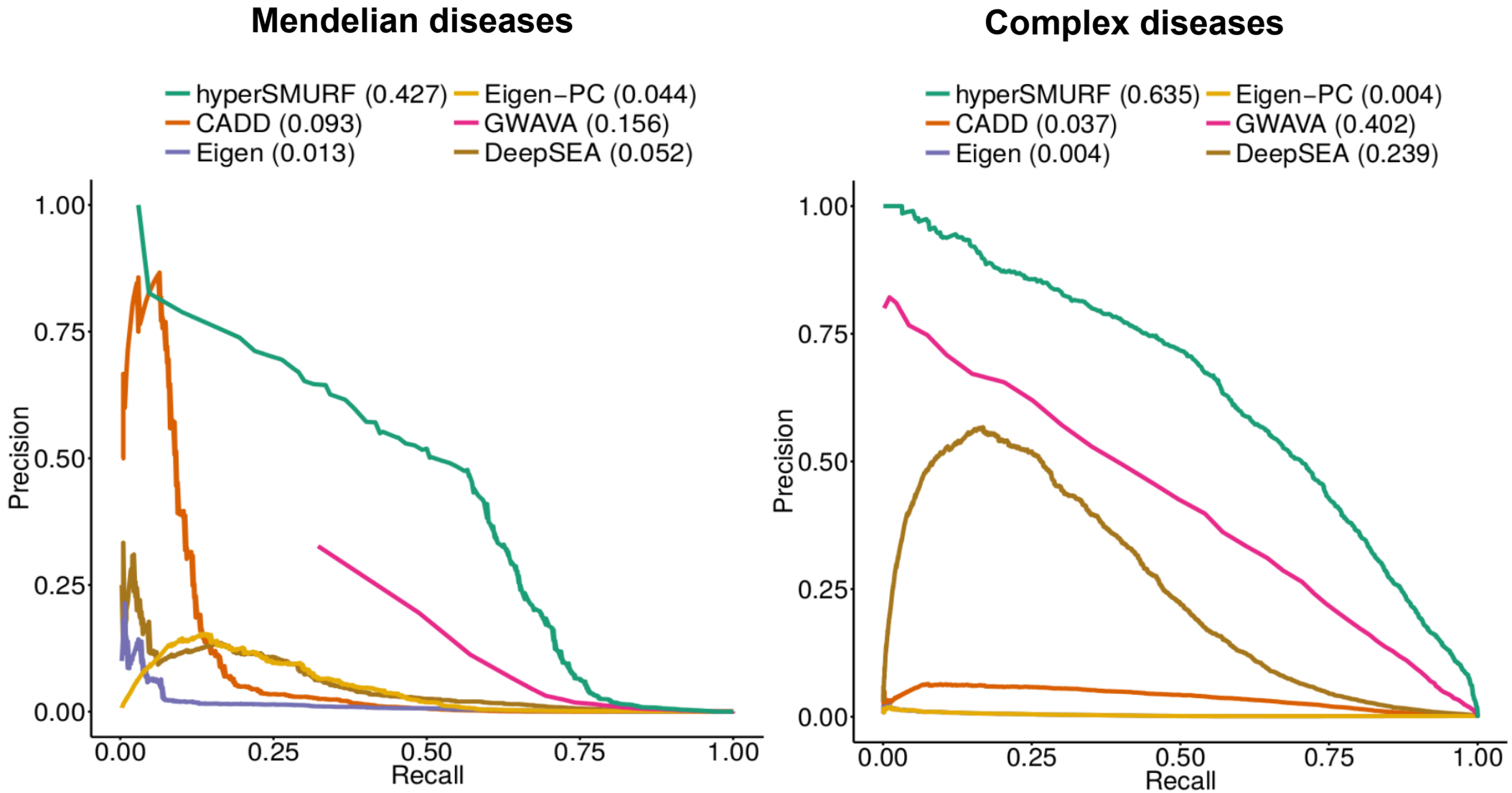
- Balancing training data through differential sampling:
  - Oversampling of the minority class
  - Partitioning and undersampling of the majority class
- Data coverage improvement and variance reduction through ensembling techniques
- Enhancing accuracy and diversity of the base learners through Hyper-ensembling

HyperSMURF:
Hyper-ensemble of SMote Undersampled Random Forests



HyperSMURF is very competitive with state-of-the-art methods:

AUPRC comparative results with state-of-the-art methods (*Schubach et al*. 2017)



10-fold “cytoband-aware” cross-validation: precision/recall curves

Drawbacks of HyperSMURF

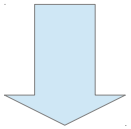
HyperSMURF learning depends on several parameters:

- the number *n* of partitions/ensembles
- the oversampling factor *f*
- the undersampling factor *u*
- other “minor” parameters

Hyper-ensemble parameters

- the number *t* of decision trees of the RF
- the number *m* of randomly selected features

Ensemble parameters



Fitting different prediction problems requires proper tuning of the learning parameters

Learning parameters strongly affect HyperSMURF performances

High impact of the hyperSMURF learning parameters on:

- Coverage of the data
- Balancing between deleterious and neutral variants
- Informativeness of the positive (deleterious) examples
- Effectiveness of the representation of the learning space
- Runtime and learning process
- Accuracy and diversity of the base learners

Results highly depend on the correct selection of the parameters for the specific problem under study

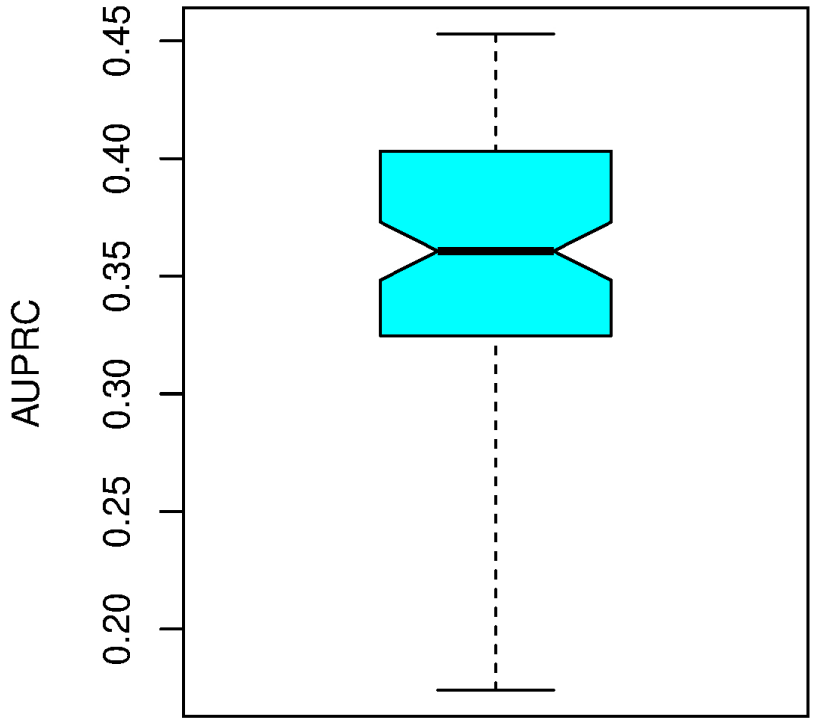
An experimental study of the impact of learning parameters for the prediction of non-coding deleterious variants in Mendelian diseases

- Same data used in *Schubach et al*, 2017:
  - 406 SNV mutations manually curated (positives)
  - 14M neutral variants (negatives)
  - 26 genomic features indicators of variant functionality (e.g. GC-content, conservation, histone modifications, DNase I accessibility, overlap withTFB sites and enhancers, overlapping CNVs)
- Hold-out setting for performance evaluation and internal cytoband-aware cross-validation (Smedley et al. 2016) for parameter tuning.
- 100 hyperSMURF models trained considering different combinations of *n*, *f* and *u* parameters
- Results obtained using a serial implementation and an arrays of jobs on the CINECA Marconi cluster.

Results

Cross-validation results on the training set across the 100 models

Best model:
n=300, *f*=1, *u*=10



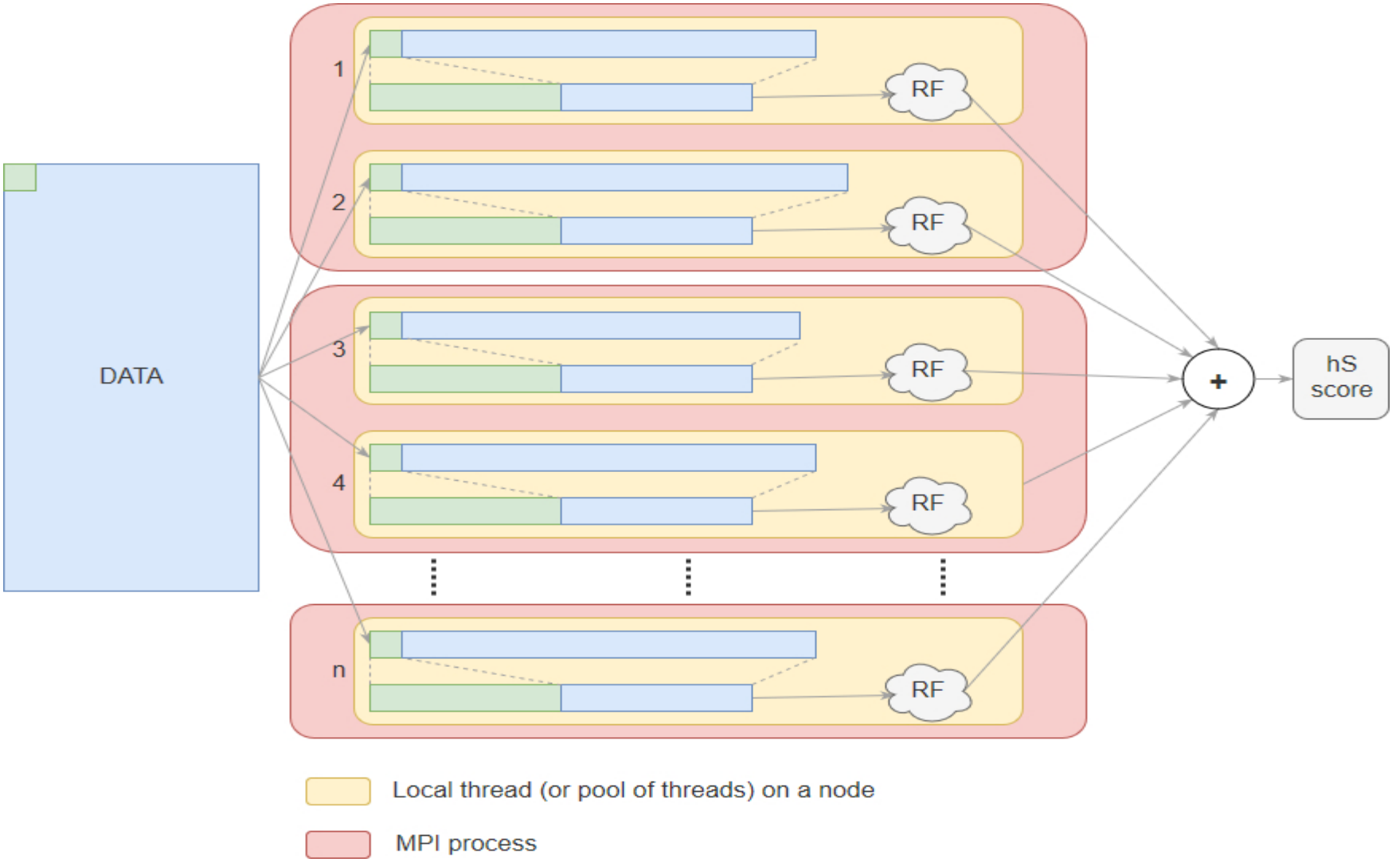
	$AUPRC$	$AUROC_{50}$	$AUROC_{100}$	$AUROC_{500}$	$AUROC_{1000}$
hyperSMURF default par.	0.3568	0.8600	0.9300	0.9091	0.8868
hyperSMURF best par.	0.4156	0.9220	0.9610	0.9407	0.9460

Results on an independent test set.
Default parameters: *n*=100, *f*=2, *u*=3 (*Schubach et al.*, 2017)

Advantages and drawbacks

- Results show that parameter tuning can boost hyperSMURF results
- Drawbacks: training and testing require from 2 to about 20 hours of computation for each model using Intel Xeon processors E5, 2.30 GHz and 128 GB RAM
- The situation can be even worse if we use e.g. thousands of features extracted from DNA with deep convolutional networks (Zhou and Troyanskaya, 2015)
- Serial implementations, even with a cluster and arrays of jobs is not enough

Par-hyperSMURF: HPC version of hyperSMURF through a mixed MPI/OpenMP parallel implementation



A very flexible HPC architecture by which we can apply hyperSMURF not only to the prediction of pathogenic variants, but more in general to genomic problems characterized by big data and very small a priori available knowledge

Conclusions

- Data imbalance in genome-wide studies motivates *hyperSMURF*
- Drawbacks of *hyperSMURF:* many learning parameters that significantly affect prediction performance
- Parameter tuning can significantly boost *hyperSMURF* results

Par-hyperSMURF - ongoing HPC parallel version of *hyperSMURF:*

- Automatic tuning of learning parameters
- Application of *Par-hyperSMURF* to:
  - Whole genome ranking and detection of mutations in genetic diseases
  - Ranking and detection of cancer driver mutations
  - Personalized Medicine problems characterized by small a priori available knowledge and big data

References:

- *M. Schubach, M. Re, P.N. Robinson and G. Valentini*. Imbalance-Aware Machine Learning for Predicting Rare and Common Disease-Associated Non-Coding Variants.
  Scientific Reports - Nature Publishing, 7:2959, 2017.

- *D. Smedley, M. Schubach, J.O.B. Jacobsen, S. Köhler, T. Zemojtel, M. Spielmann, M. Jäger, H. Hochheiser, N.L. Washington, J.A. Mcmurry, M.A. Haendel, C.J. Mungall, S.E. Lewis, T. Groza, G. Valentini, P.N. Robinson*. A Whole-Genome Analysis Framework for Effective Identification of Pathogenic Regulatory Variants in Mendelian Disease.
  American Journal of Human Genetics – Cell Press, 99:3, pp. 595-606, 2016.

Thank you for your attention!

http://anacletolab.di.unimi.it

Appendices

Detection of genetic variants – disease associations

Whole Genome Sequencing (WGS) enables the investigation of genomic variation in coding as well in non-coding regions across the entire human genome

Application to the detection of mutations associated with Mendelian (e.g. Cystic fibrosis or Huntington disease) and *complex* (e. Alzheimer's and Parkinson's) genetic disease.

Two main problems:
1) Most of genetic variation in human genome is “neutral”: how to find “possible deleterious” variants?
2) Most studies focused on coding regions, but what about non coding regions?

HyperSMURF

Pseudocode of the HyperSMURF algorithm

Input:
- $\mathcal{P}$: set of positive examples (Deleterious variants)
- $\mathcal{N}$: set of negative examples (Non-deleterious variants)
- n: number of partitions
- k: number of nearest neighbors for *SMOTE* oversampling
- f: oversampling factor

begin algorithm

(i) Initialization and partitioning of $\mathcal{N}$:
  $n_{ex} := (f+1)|\mathcal{P}|$
  $\{\mathcal{N}_1, \mathcal{N}_2, \ldots, \mathcal{N}_n\}$:= Do.partition $(\mathcal{N}, n)$
  $i := 1$
while $(i \leq n)$ do
  (ii) *SMOTE* oversampling:
    $\mathcal{P}_S$:= SMOTE $(\mathcal{P}, k, f)$
  (iii) Undersampling of non-deleterious variants:
    $\mathcal{N}'$:= Undersample $(\mathcal{N}_i, n_{ex})$
  (iv) Training set assembly:
    $\mathcal{T} := \mathcal{P} \cup \mathcal{P}_S \cup \mathcal{N}'$
  (v) Random Forest training:
    M_i := RF $(\mathcal{T})$
  $i := i + 1$
end while

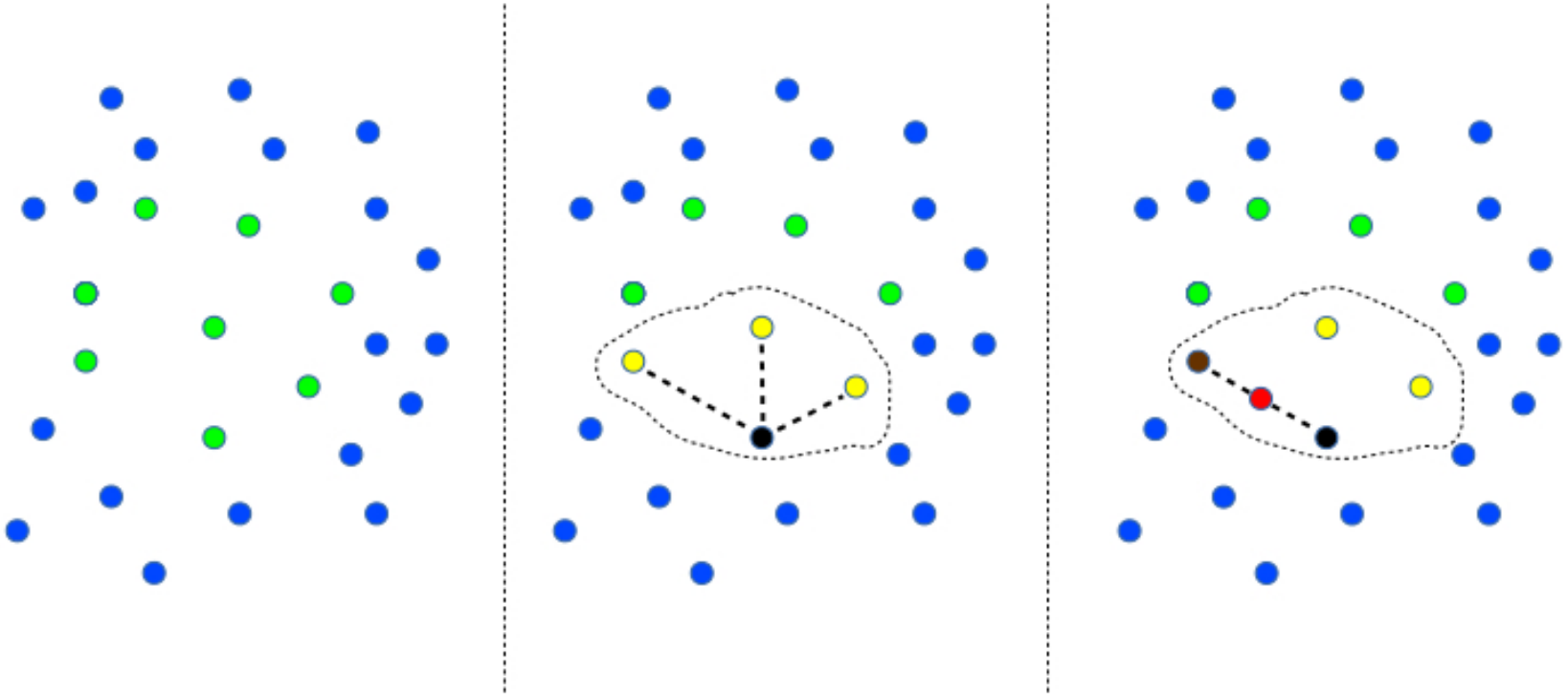
end algorithm

Output:
$M = \{M_1, M_2, \ldots, M_n\}$: a set of RF models

Output on a test variant $\boldsymbol{x}$:
- $Hy_{score}(\boldsymbol{x}) := \frac{1}{n} \sum_{i=1}^{n} P(\boldsymbol{x}$ is positive $|M_i)$

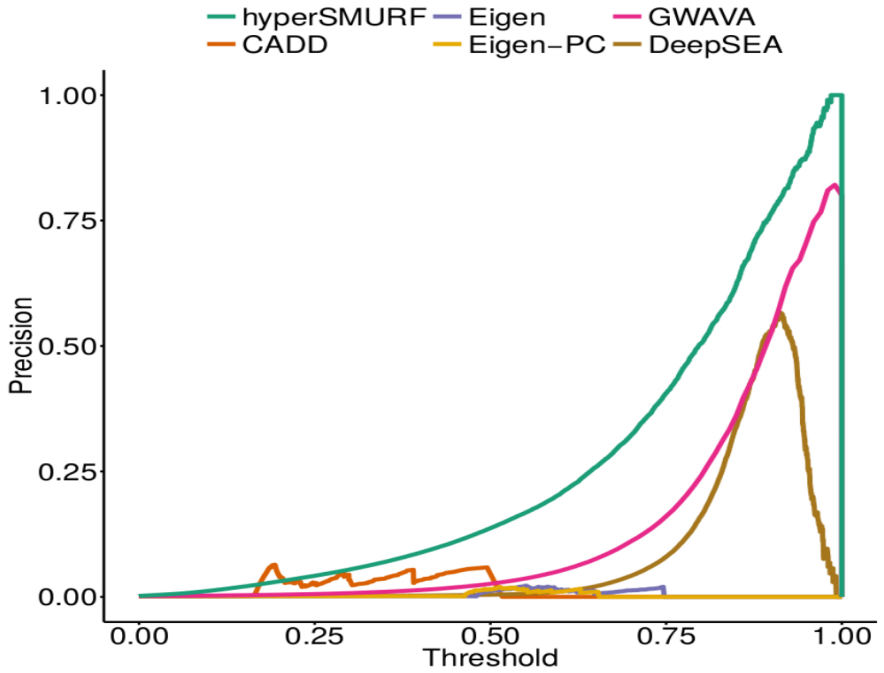
SMOTE :

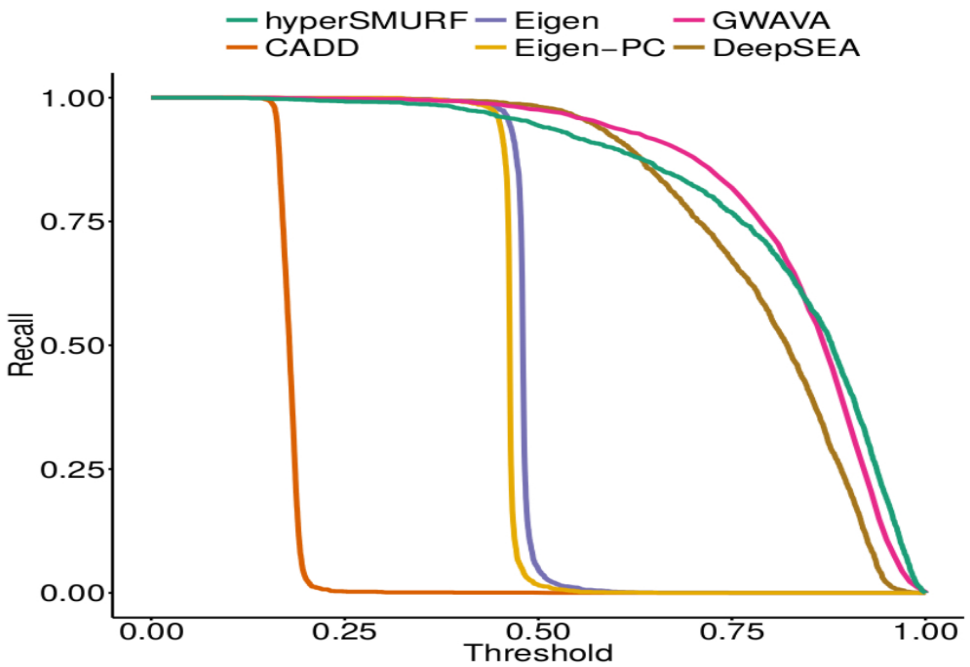
Synthetic Minority Oversampling Technique (*Hall et al*. 2002)

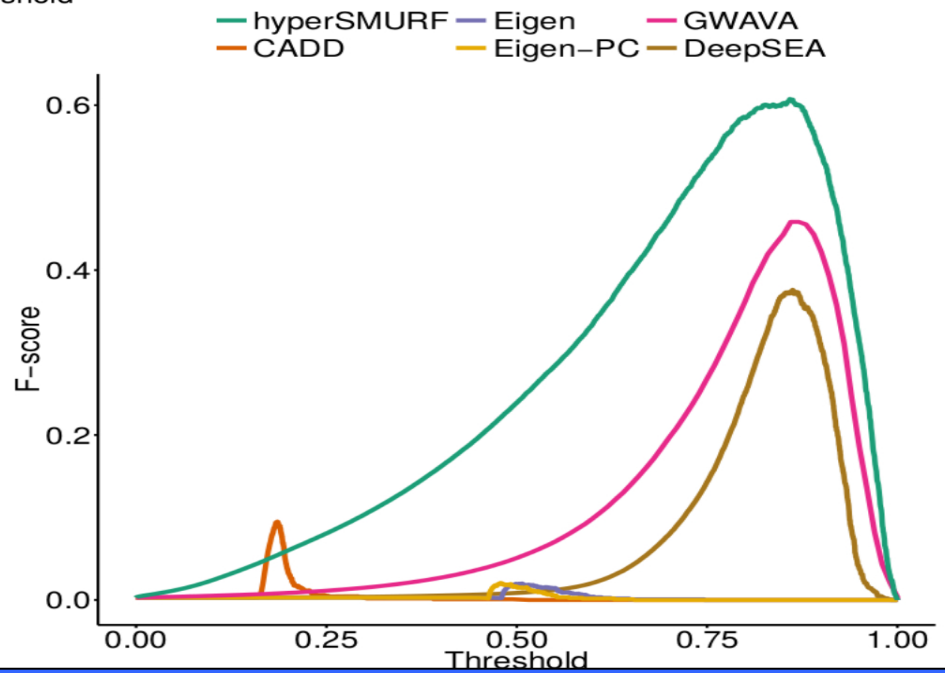


Results

Compared precision, recall and F-score (complex diseases)

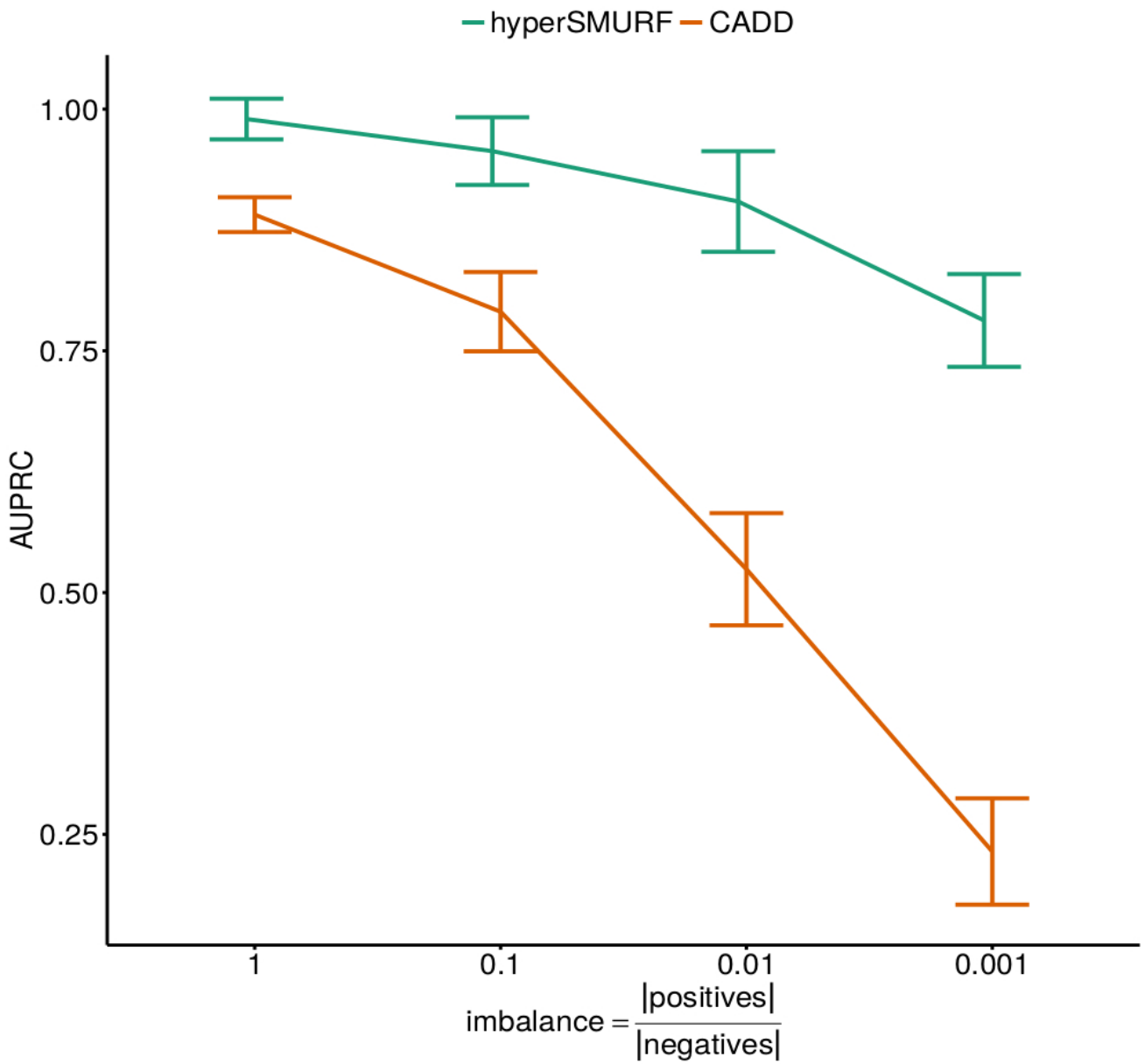






HyperSMURF strength is evident with imbalanced data

AUPRC results of HyperSMURF and CADD at different imbalance levels



Experimental set-up

Genomic experiments

Genome-wide prediction of deleterious variants in non coding region

1) *Mendelian diseases*:
406 SNV mutations manually curated (positive examples)

14M neutral variants (negatives)

2) *Complex diseases*:
2115 regulatory GWAS hits from the GWAS catalog (National Human Genome Research Institute)

1.4M neutral variants (negatives)

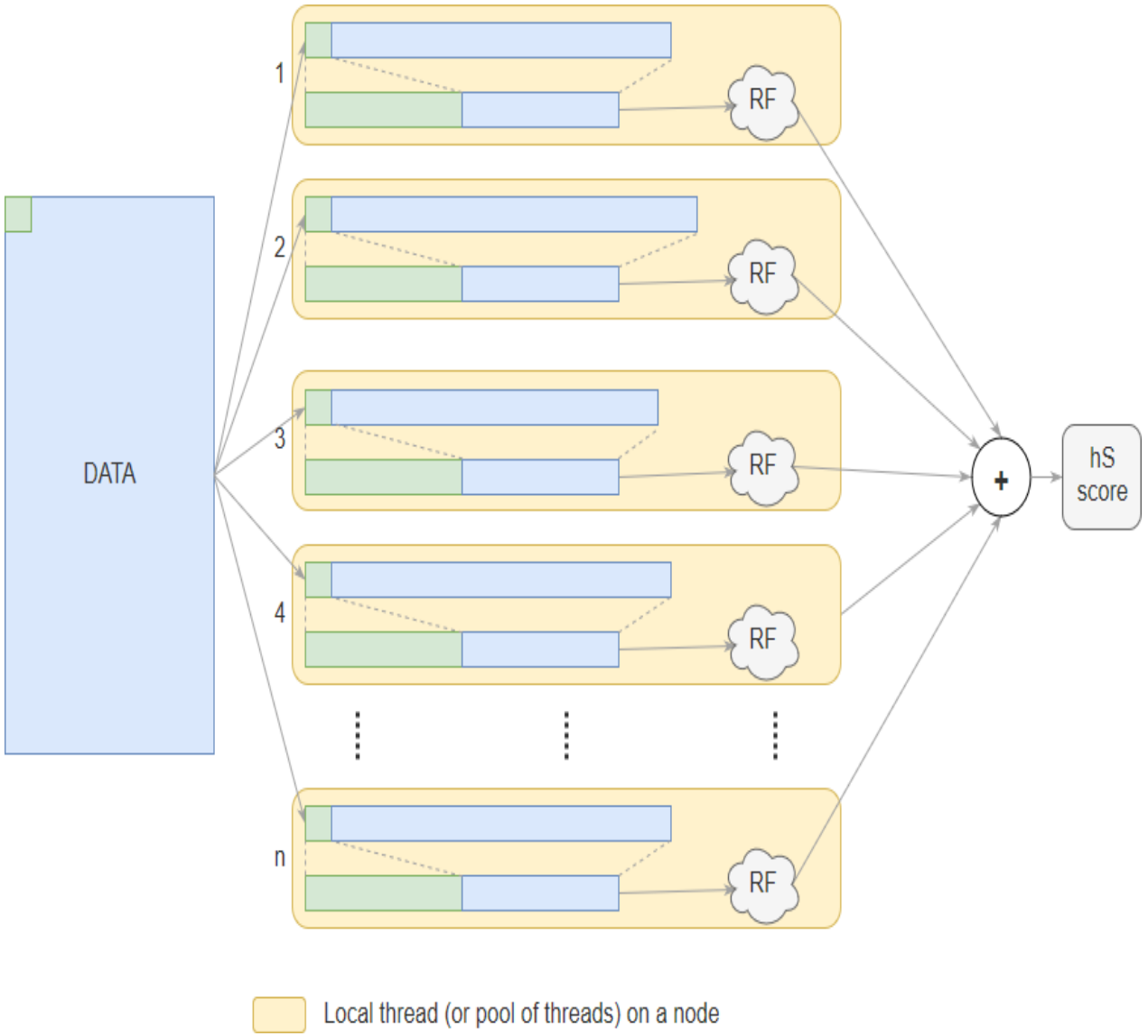
Genomic attributes

1) Mendelian data: 26 genomic attributes downloaded from public data bases (UCSC, Stanford, NCBI and others):

- Conservation scores
- Transcriptional features
- Regulation features
- Overlapping CNVs
- GC content
- Epigenomic features

2) GWAS data: 1842 genomic attributes directly extracted from DNA sequence through deep convolutional networks (Zhou & Troyanskaya, 2015)

- DNAse features
- Transcription factor features
- Histone features
- Conservation scores

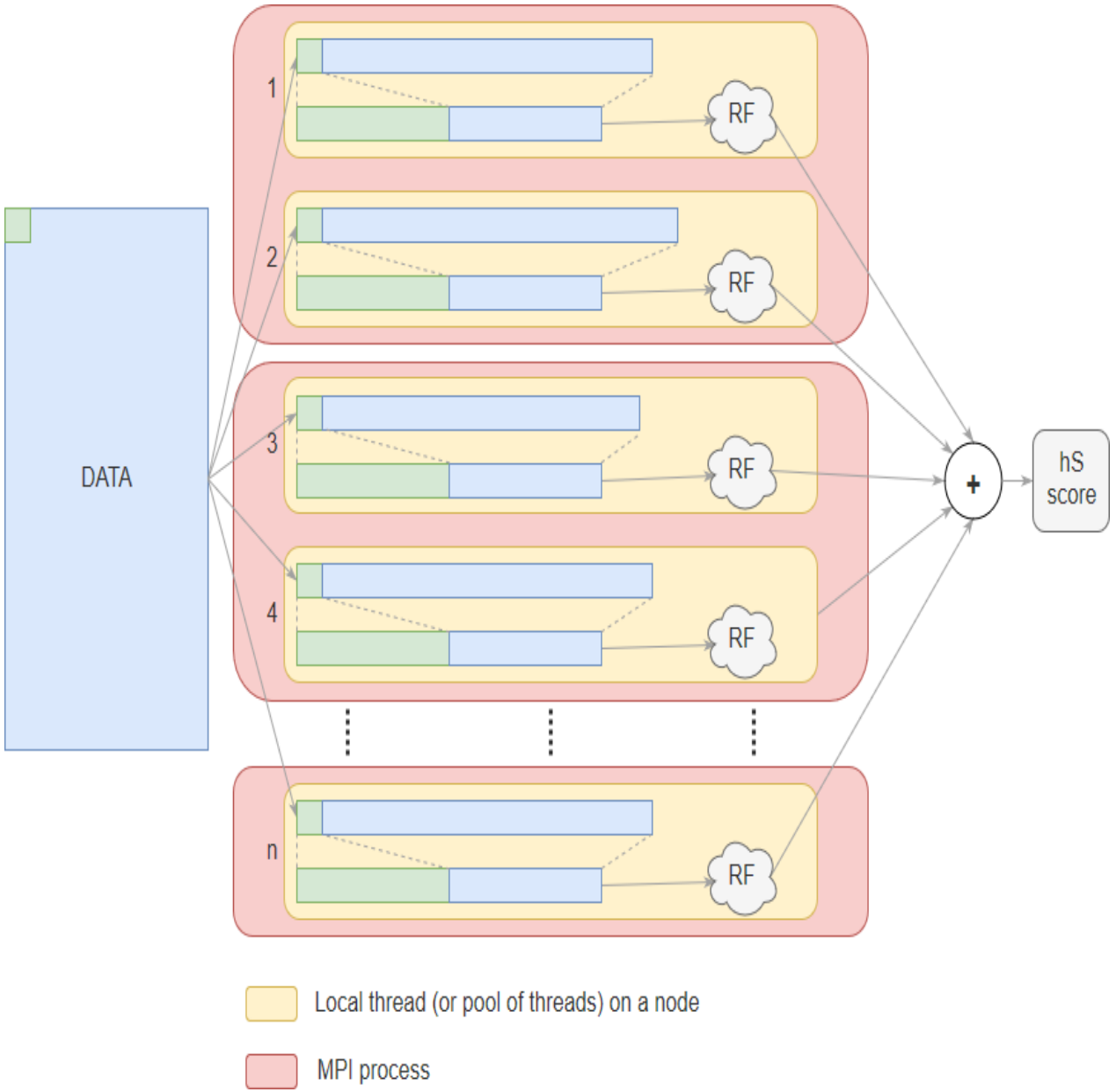


First level of parallelization: effective multi-threading via OpenMP

Each partition is assigned to a thread (or a pool of threads) in a single node.

Tasks of over/undersampling and random forest training for each partition are executed in parallel.

This approach is very scalable, since the only sequential operation lies in the final score accumulation (performed atomically by each thread)



Second level of parallelization: multi-node computing via MPI

Partitions are divided into chunks, and each chunk is assigned to a MPI process.

In each MPI process the same first level parallelization is applied.

Each MPI process can be handled by a different node in a cluster.

Intercommunication between MPI processes is required only in the initial dataset transfer and in the final accumulation.