Learning Classifier Systems in an Automated Machine Learning Framework

Ryan Urbanowicz, PhD

Department of Biostatistics, Epidemiology, and Informatics

University of Pennsylvania, Philadelphia PA

Twitter: @DocUrbs

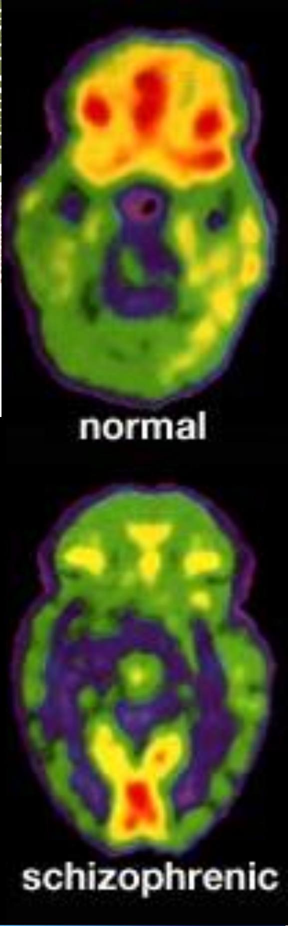
Lab Website: www.med.upenn.edu/urbslab/

Perelman School of Medicine University of Pennsylvania

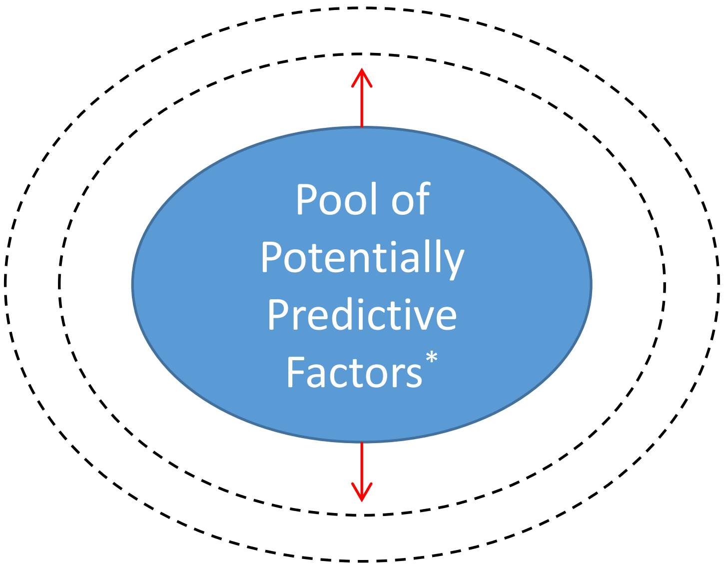
Institute for Biomedical Informatics


THE COMING JOB BOOM ■ PAKISTAN'S SPY AGENCY ■ REUNION TV
TIME
INSIDE THE WORLD OF AUTISM
More than one million Americans may have it, and the number of new cases is exploding. What scientists have discovered. What families should know.
Tommy Barrett, 11, of San Jose, California, has Asperger syndrome

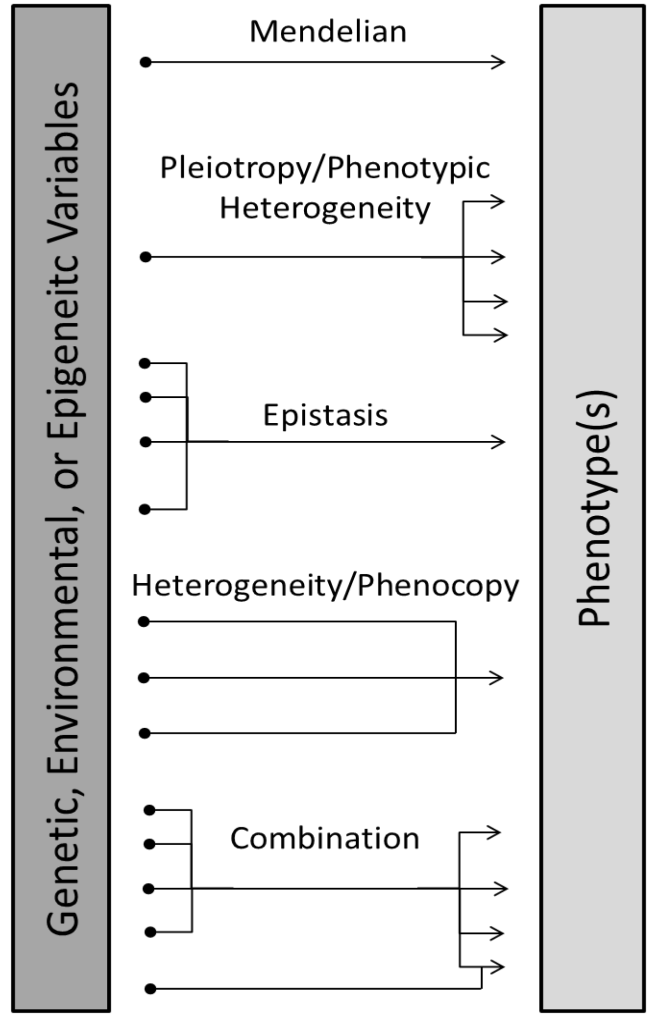

CANCER
GENERAL
LEUKEMIA
PROGNOSIS
HOPE
REPLICATION
CARCINOGENS
GENETIC
TYPE
TREATED
LOCATION
TISSUES
CANCERS
SEVERE
GENES
SYMPTOMS
METASTASIZE
PATHOLOGIST
ACQUIRED
MICRORNAS
CONNECTIVE
HYPERACTIVE
PROTECTIVE
PREVENTION
INACTIVATED
ENVIRONMENTS
SUPPRESSOR
INCREASINGLY
METASTASIS
DIFFERENTIATE
LOCATIONS
FUNCTIONS
ADHESION
STAGE
PROTECTION
NEW
HEMATOPOIETIC
ESTABLISHING
MOLECULAR
HOST'S
BODY
TREATMENTS
COMPLICATIONS
PROPERTIES
LOSS
PATHOGENESIS
TWO
CELL
MUTATION
CELLS
ONCOGENES
HISTOLOGY
THEREFORE
BLOOD-FORMING
BRANCH
REQUIRES
CERTAIN
MARKERS


normal
schizophrenic

Challenge: Biomedical Data Analysis

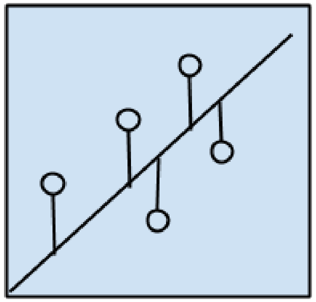


Do we have [all] the relevant features collected/available?

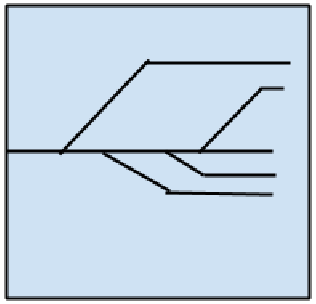


Disease Phenotype(s)

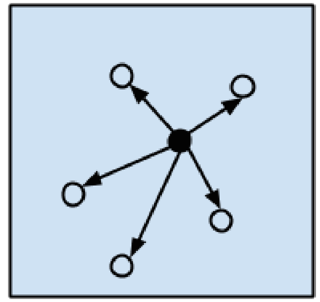
Machine Learning Algorithm Families



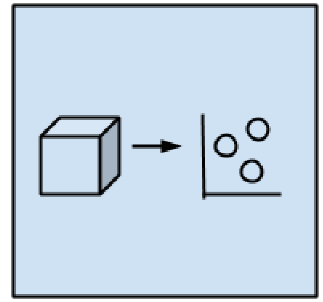
Regression Algorithms



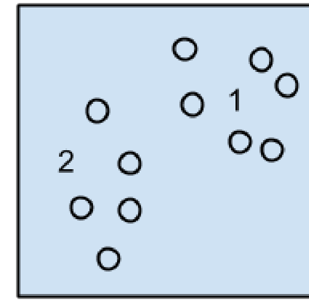
Regularization Algorithms



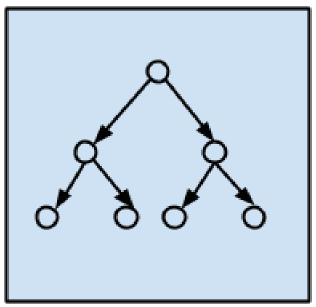
Instance-based Algorithms



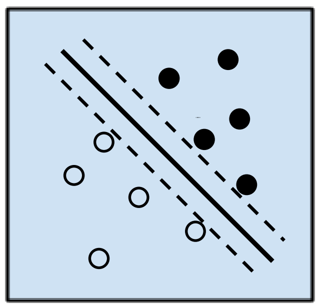
Dimensional Reduction Algorithms



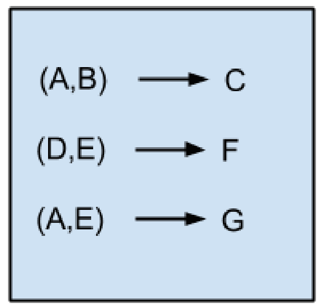
Clustering Algorithms



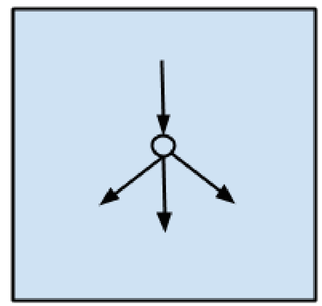
Decision Tree Algorithms



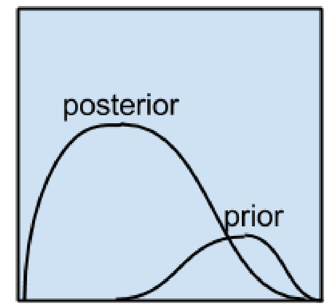
Support Vector Machines



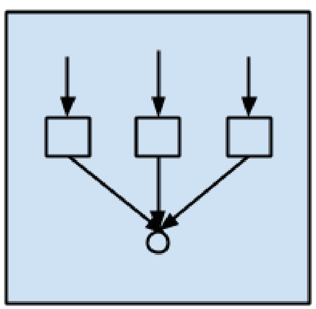
Association Rule Learning Algorithms



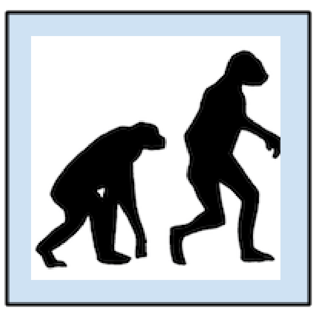
Artificial Neural Network Algorithms



Bayesian Algorithms

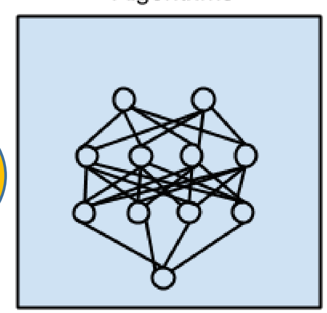


Ensemble Algorithms

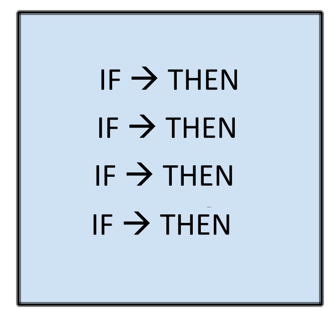


Evolutionary Algorithms

Non-exhaustive list of ML families



Deep Learning Algorithms



Rule-Based Machine Learning

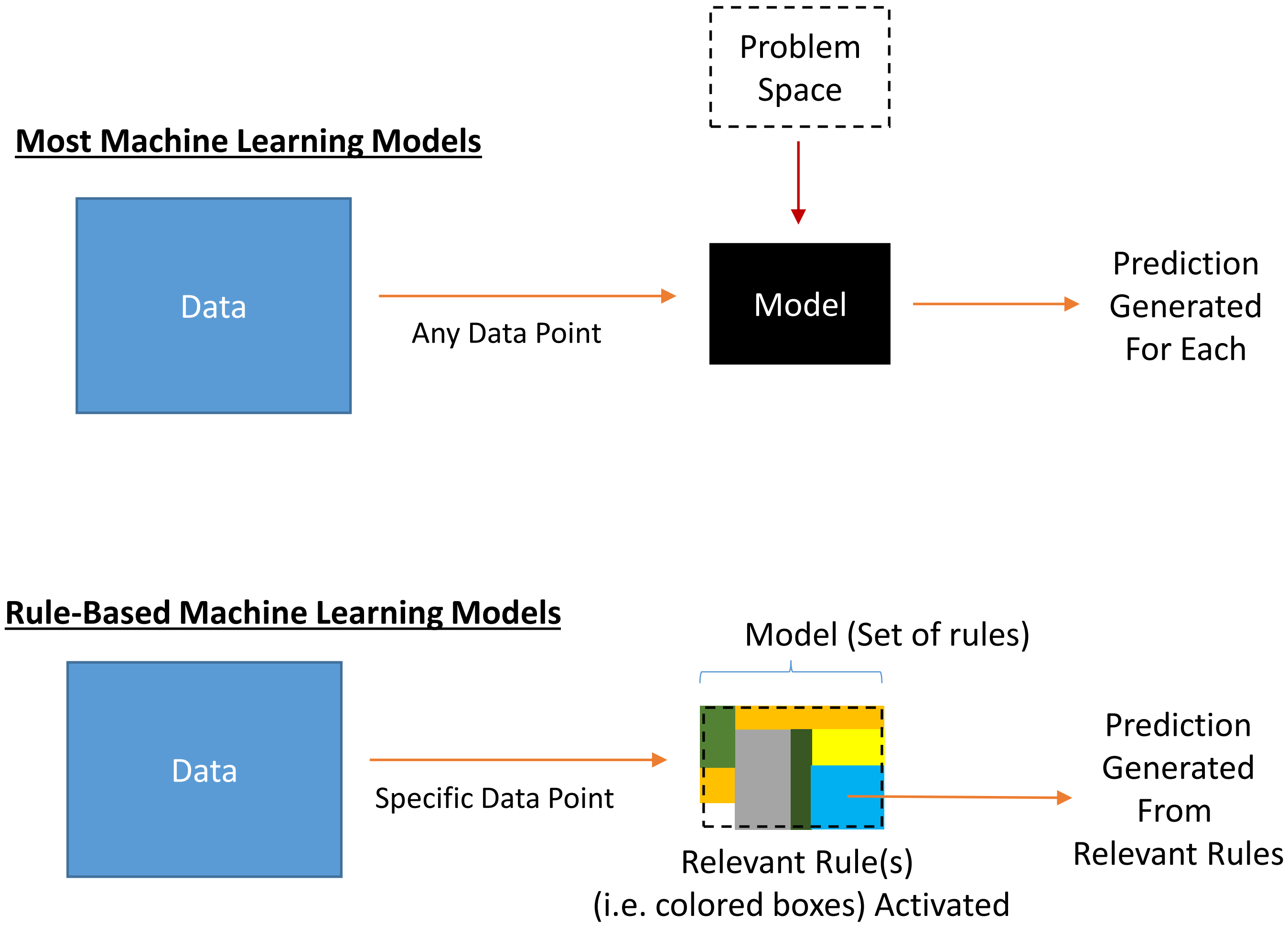
Learning Classifier Systems

- LCS as an RBML
  - The intuition and interpretability of a rule-based system
  - Data-driven improvement of performance of ML
  - Open ended discovery of evolutionary algorithms
  - Focused primarily on classification tasks
- 'Piece-wise' Learning
  - Ideal for directly modeling heterogeneous associations
- A different modeling perspective than most ML algorithms
  - Part of a valuable ML toolkit

John H. Holland

- Set of IF:THEN rules
  - IF SNP_A = 0 AND BMI < 20 THEN Control
  - IF SNP_X = 1 THEN Case
  - IF 30 < BMI > 40 THEN Case
  - IF 20 < BMI > 30 THEN Control
  - IF 20 < BMI > 30 AND SNP_Y = 2 THEN Case
  - …

Challenging the Paradigm of Modeling



Rule Representation

Quaternary Rule Representation

Rule Condition	#	2	#	#	0	#	1	2	#	#

Classification	1

Mixed Discrete-Continuous Attribute-List Knowledge Representation

Attribute Reference	2	5	7	8
Rule Condition	2	0	0.4 - 0.7	'high'

Classification	1

KEY: Continuous Discrete

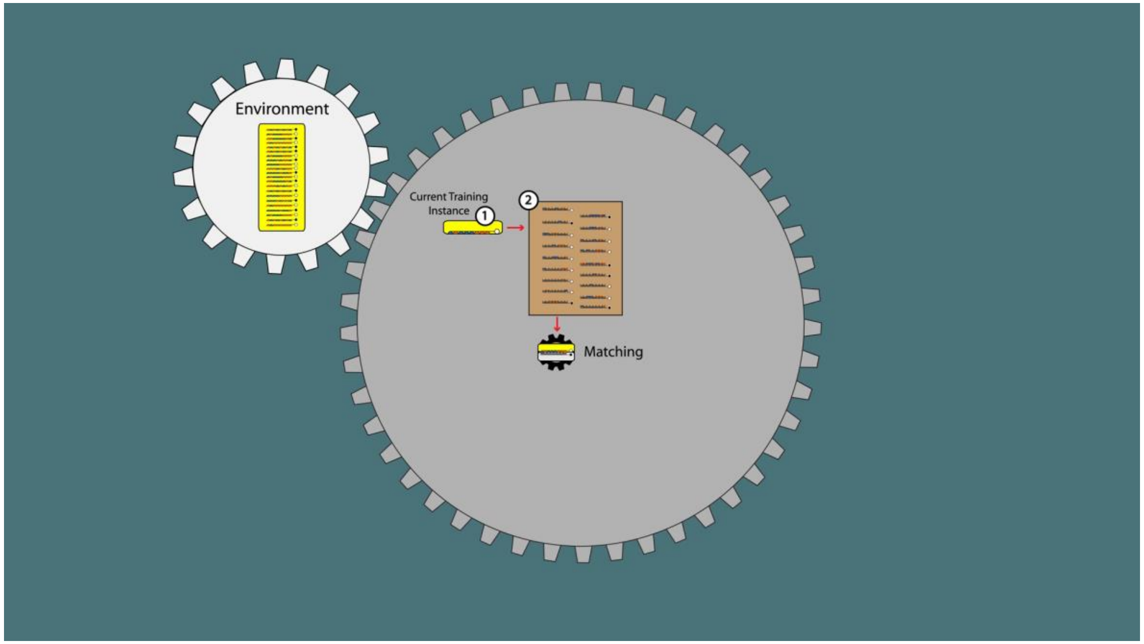
Very similar to ALKR(Attribute List Knowledge Representation): [Bacardit et al. 09]

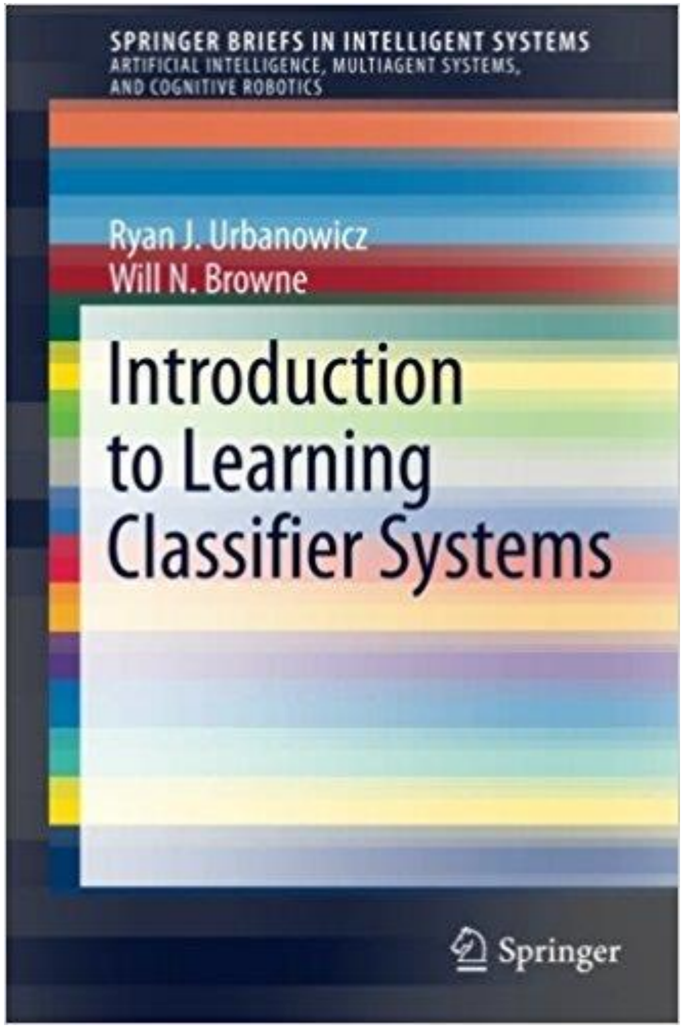
Urbanowicz et. al 2015 – ExSTraCS 2.0: Description and Evaluation of a Scalable LCS

Learning Classifier Systems - Resources

<u>Review Paper</u>: Urbanowicz & Moore 2009 – Learning Classifier Systems: A complete introduction, review, and roadmap

Learning Classifier Systems in a Nutshell

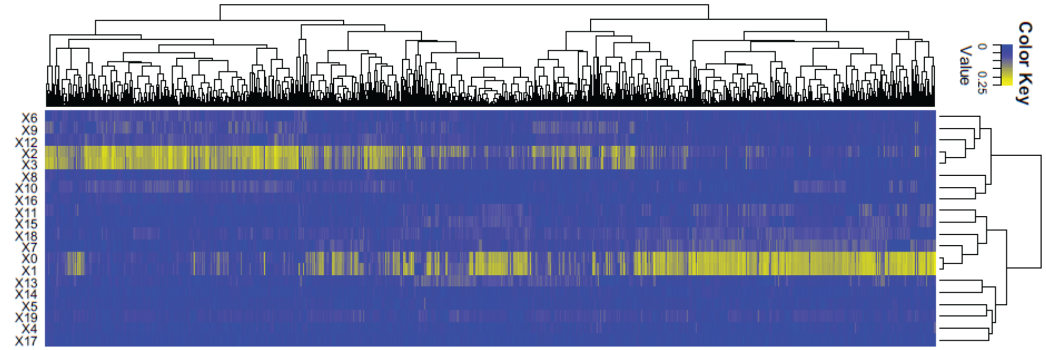




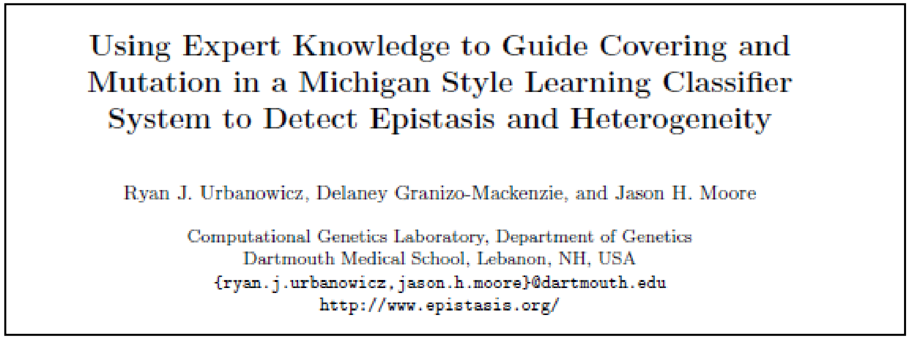
eLCS

Improving Performance and Interpretability

- Attribute Tracking/Feedback
  - Characterize heterogeneity
  - Speed up learning with useful building blocks.



- Expert Knowledge Guided Rule Discovery
  - EK probabilistically weighting mutation, crossover, and covering.



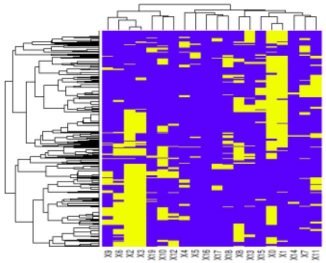
Using Expert Knowledge to Guide Covering and Mutation in a Michigan Style Learning Classifier System to Detect Epistasis and Heterogeneity

Ryan J. Urbanowicz, Delaney Granizo-Mackenzie, and Jason H. Moore

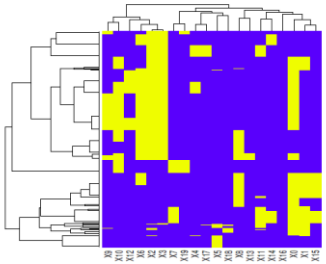
Computational Genetics Laboratory, Department of Genetics
Dartmouth Medical School, Lebanon, NH, USA
{ryan.j.urbanowicz,jason.h.moore}@dartmouth.edu
http://www.epistasis.org/

- Rule Compaction

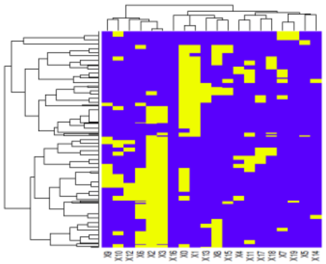
NONE



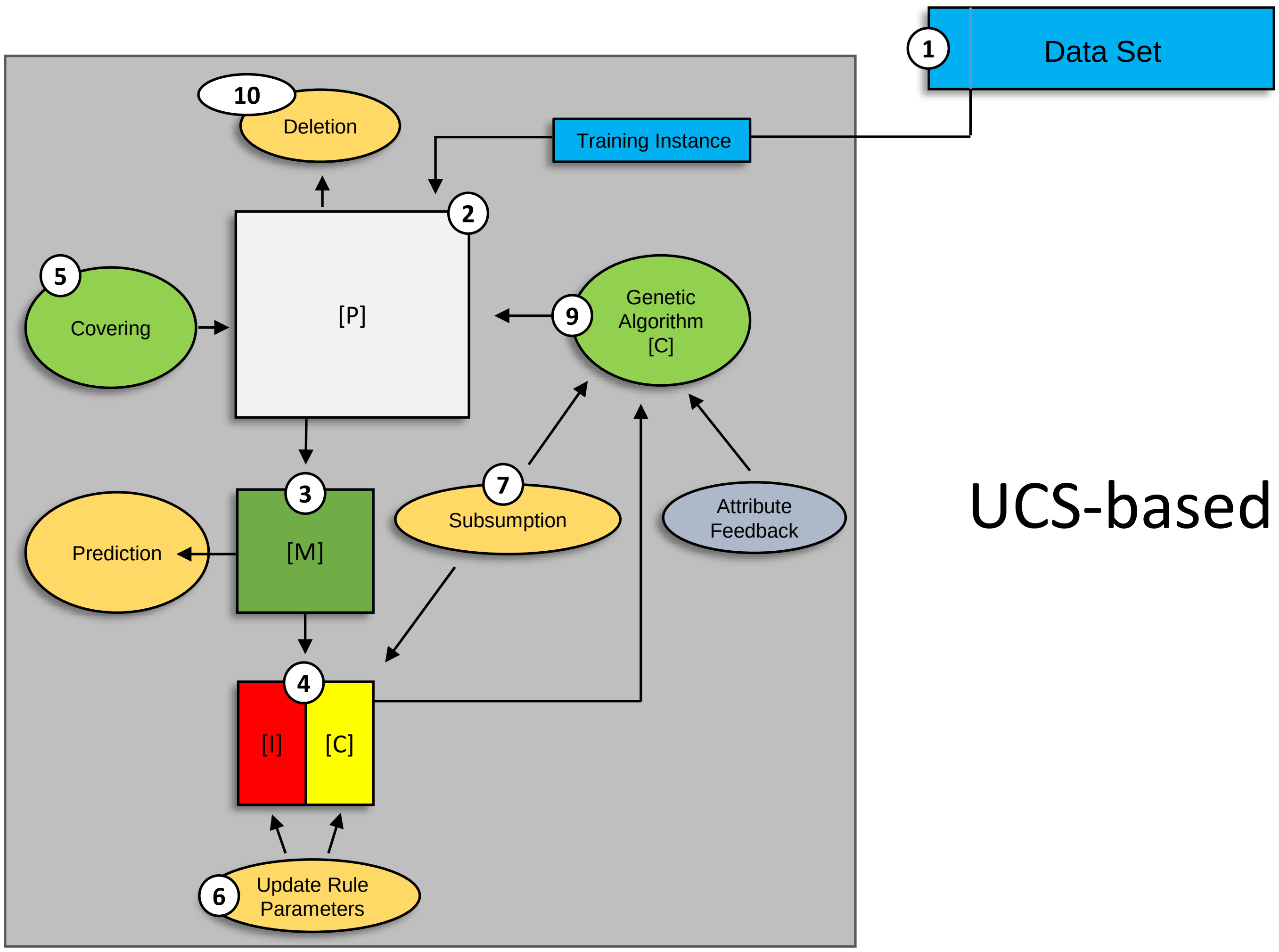
Fu2



PDRC

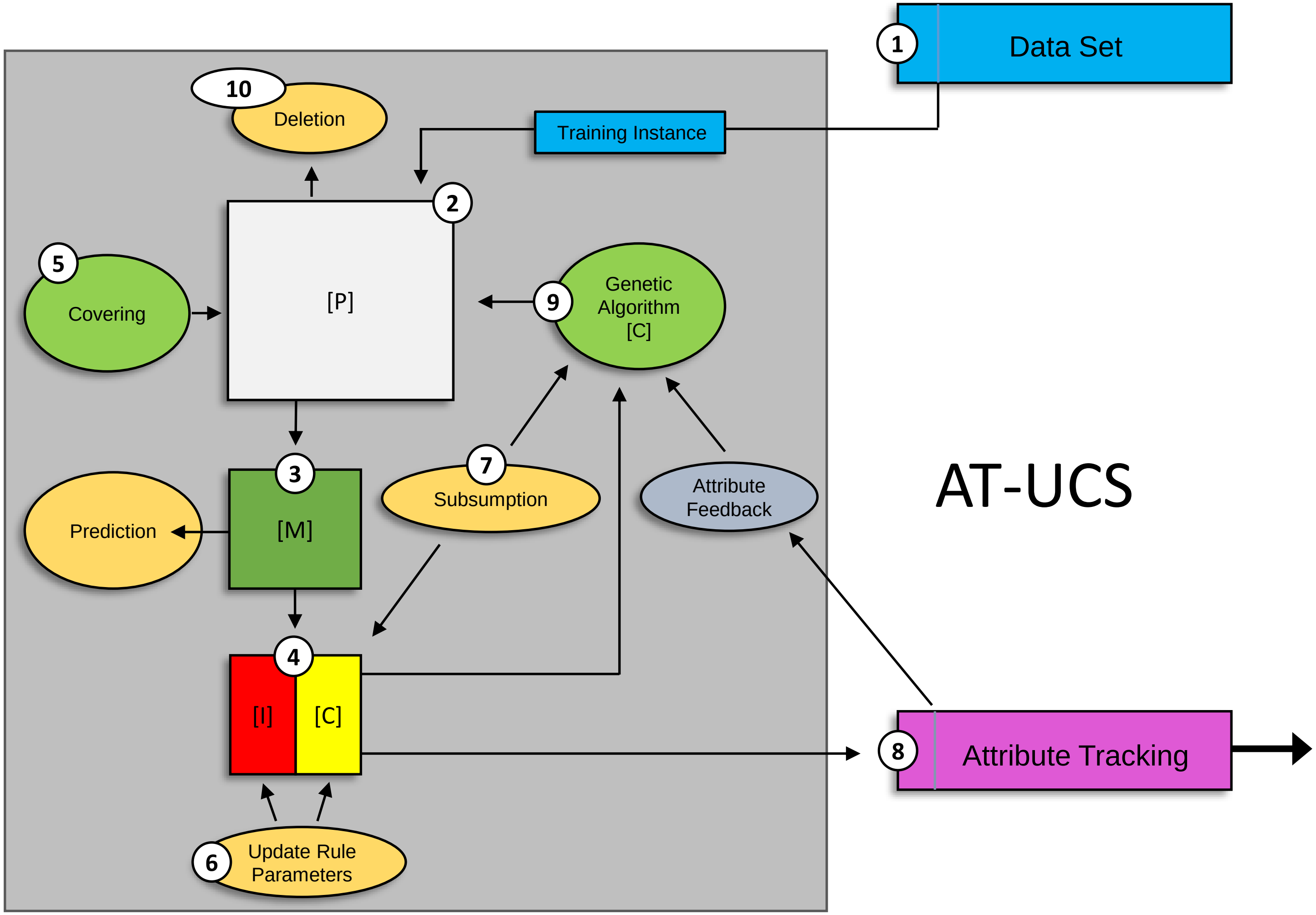


Stepping back: combining the heuristics

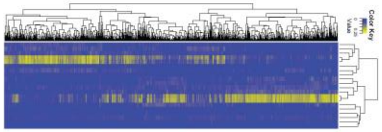


UCS-based

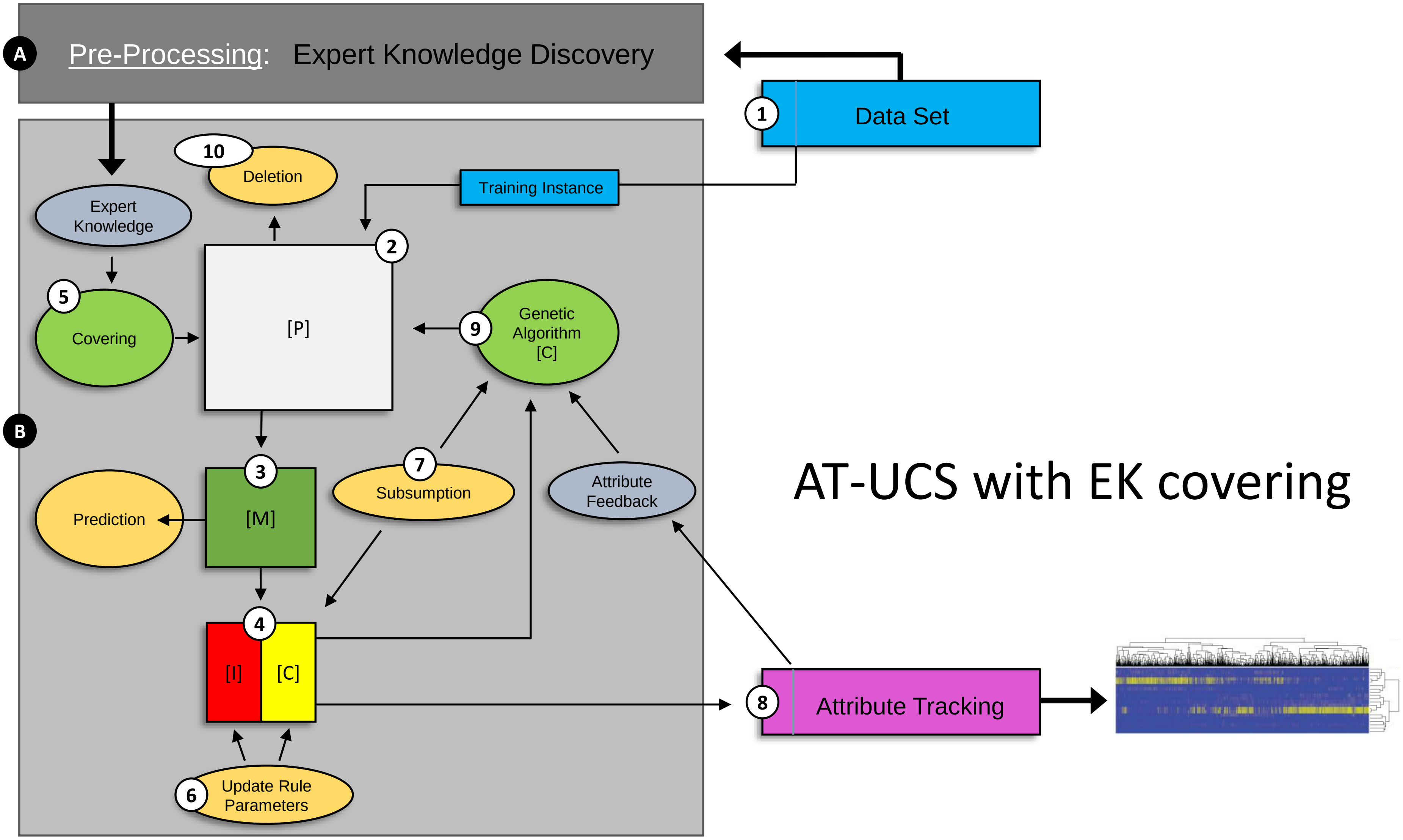
Stepping back: combining the heuristics



AT-UCS

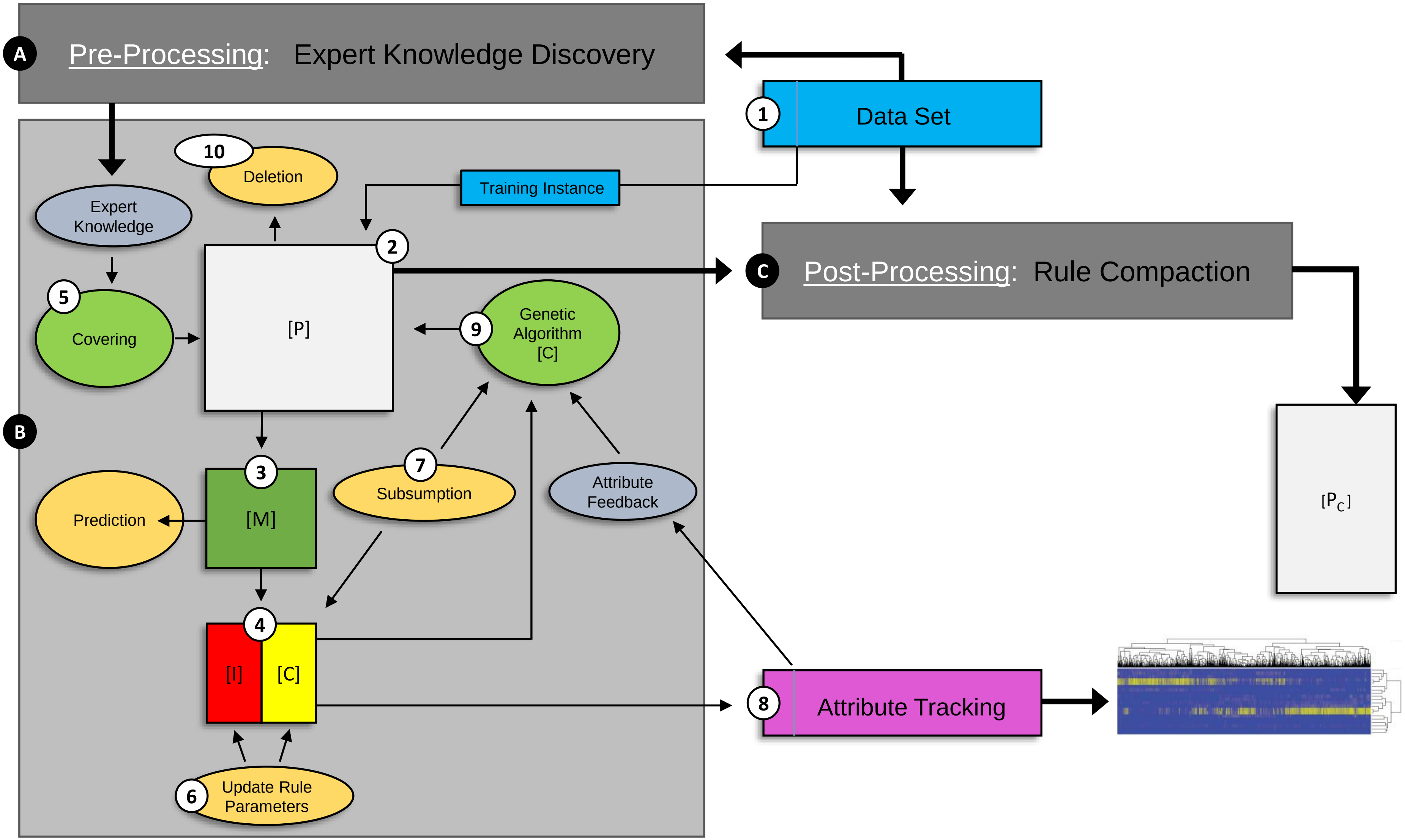


Stepping back: combining the heuristics

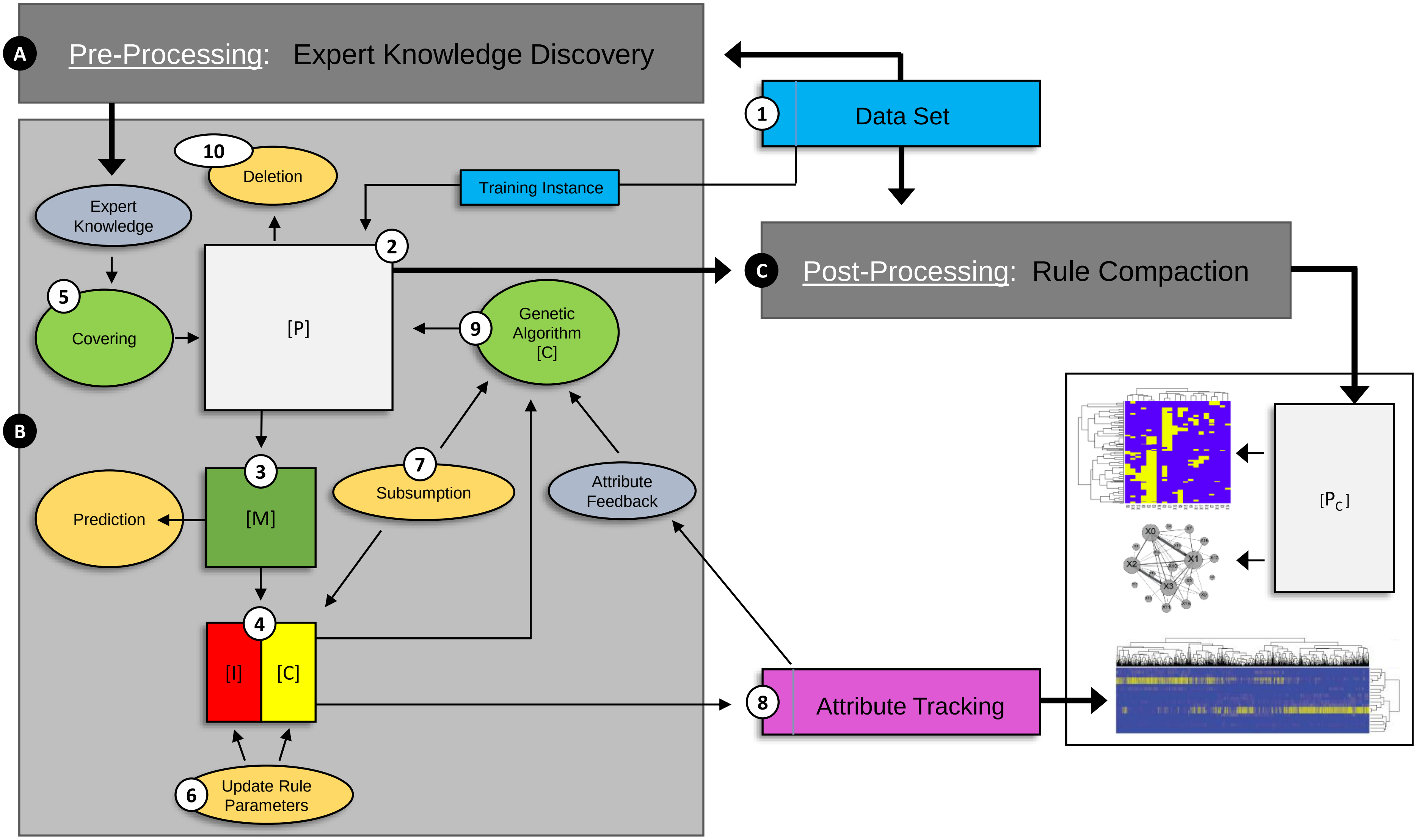


AT-UCS with EK covering

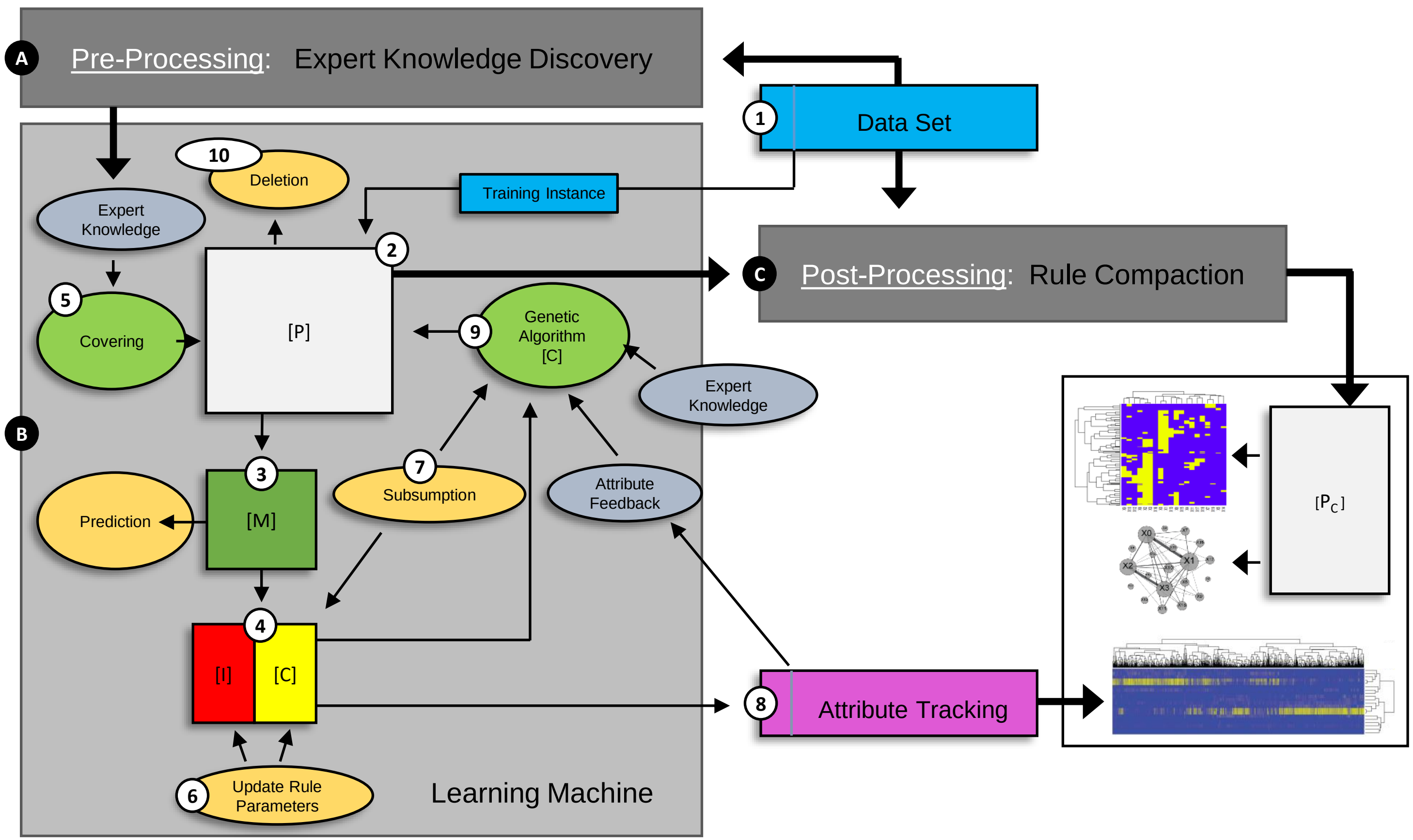
Stepping back: combining the heuristics



Stepping back: combining the heuristics



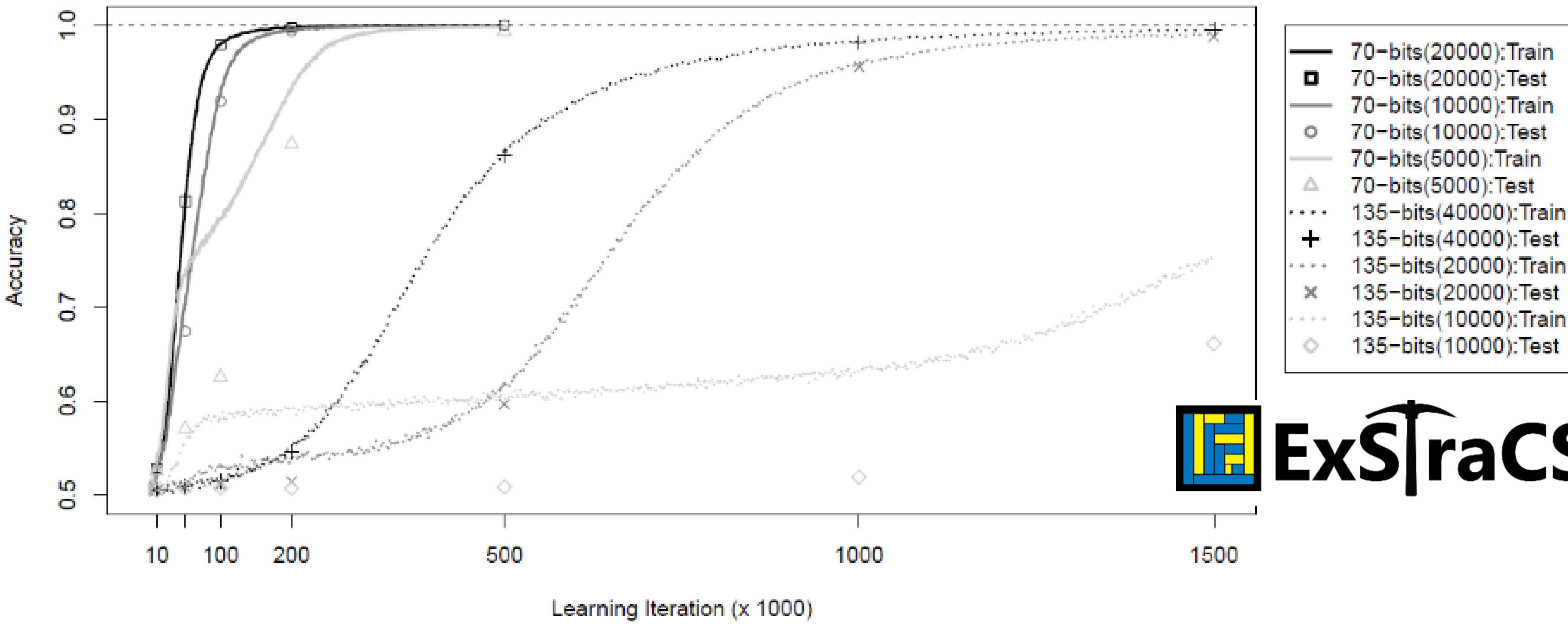
ExSTraCS



Urbanowicz, Ryan J., and Jason H. Moore. "ExSTraCS 2.0: description and evaluation of a scalable learning classifier system." *Evolutionary intelligence* 8.2-3 (2015): 89-116.

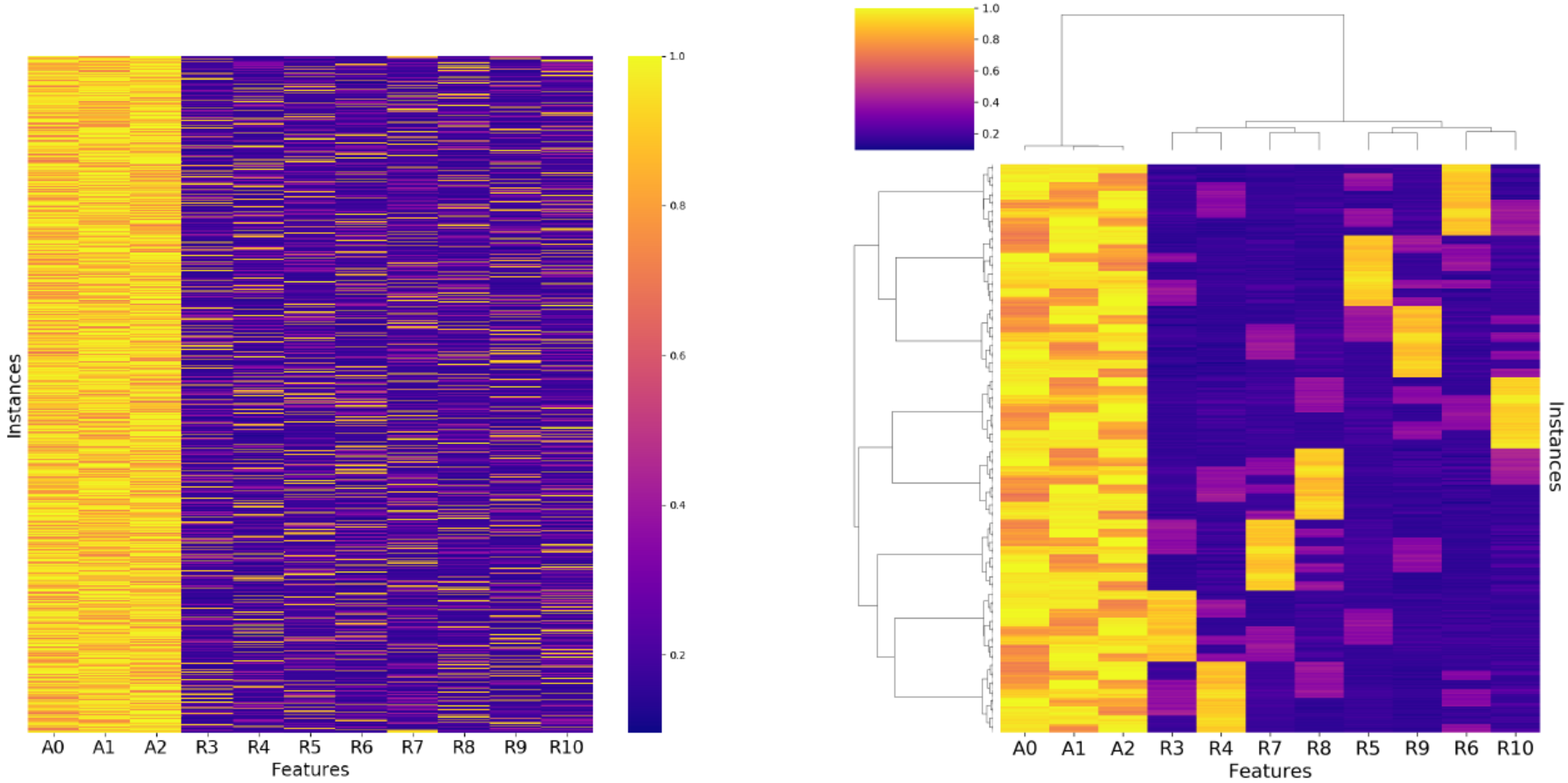
Directly solving the 135-bit Multiplexer

- TO SOLVE: 135-bit Multiplexer
  - Find 128 heterogeneous 8-way interactions
  - All 135 features are predictive in at least some subset of the dataset.
  - 256 optimal rules (in a traditional XCS-based algorithm)



ExSTraCS

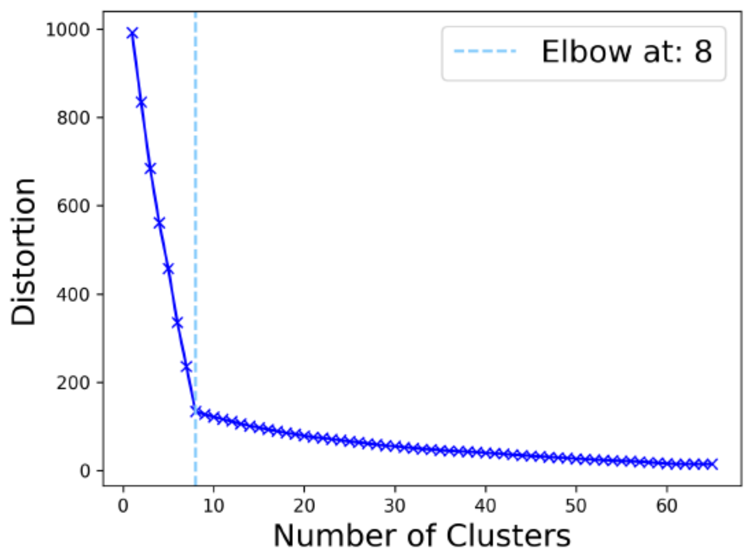
LCS-DIVE: AT Clustering (11-bit MUX)



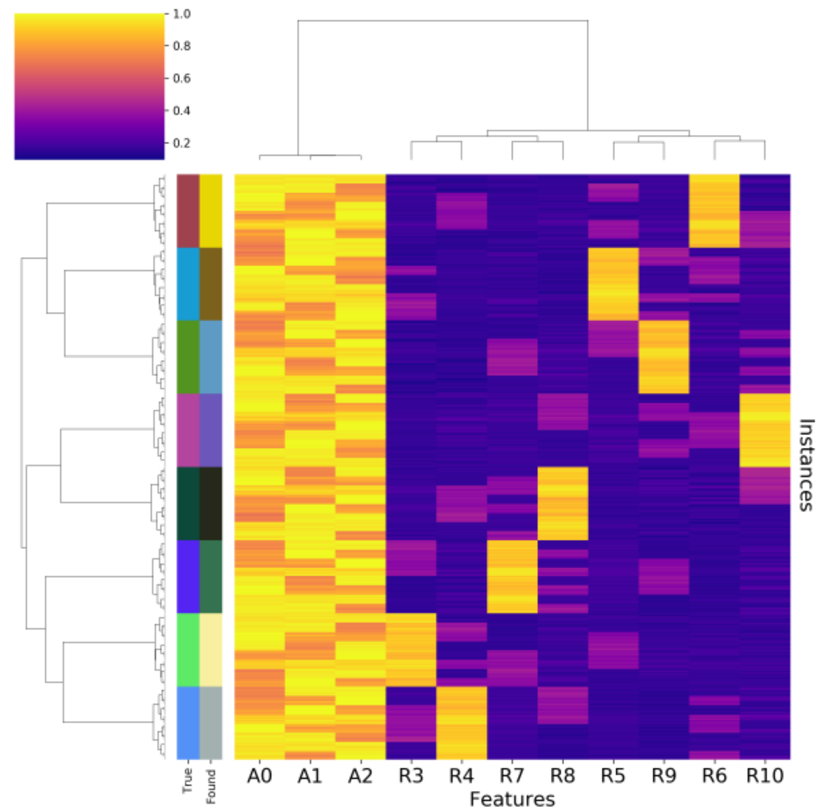
(a) Unclustered Heatmap

(b) Clustered Heatmap

LCS-DIVE: AT Clustering (11-bit MUX)

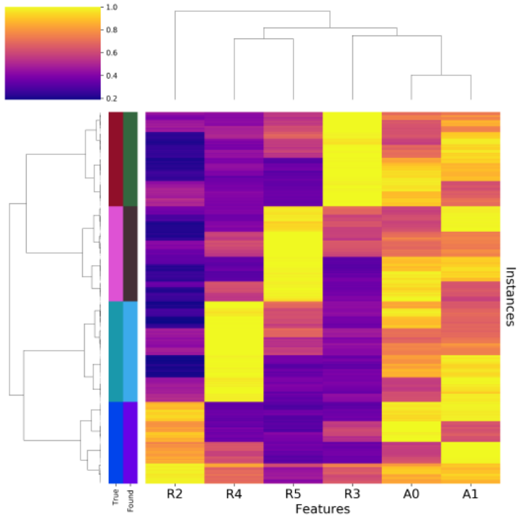


(b) Elbow Plot

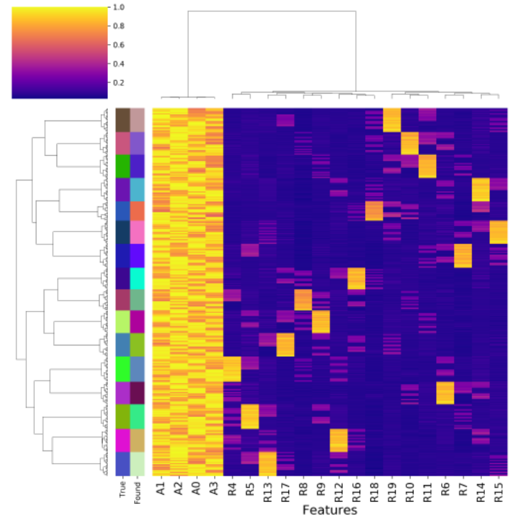


(a) Clustermap

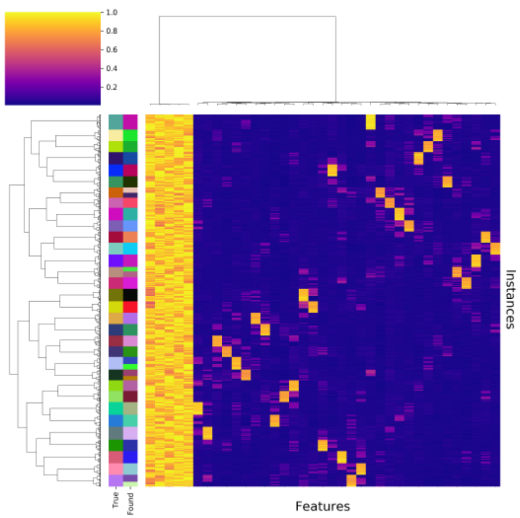
LCS-DIVE: MUX Problems



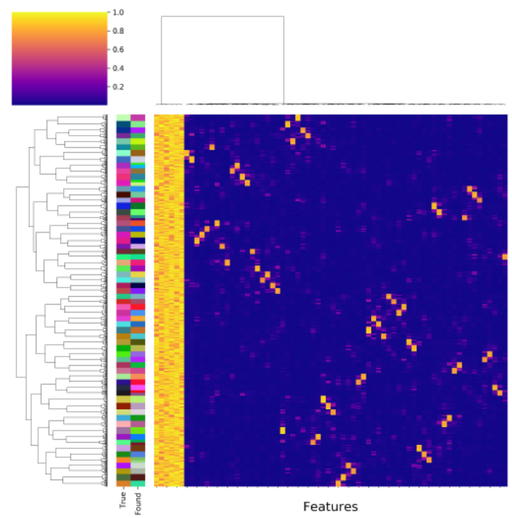
(a) 6-bit MUX



(b) 20-bit MUX



(c) 37-bit MUX

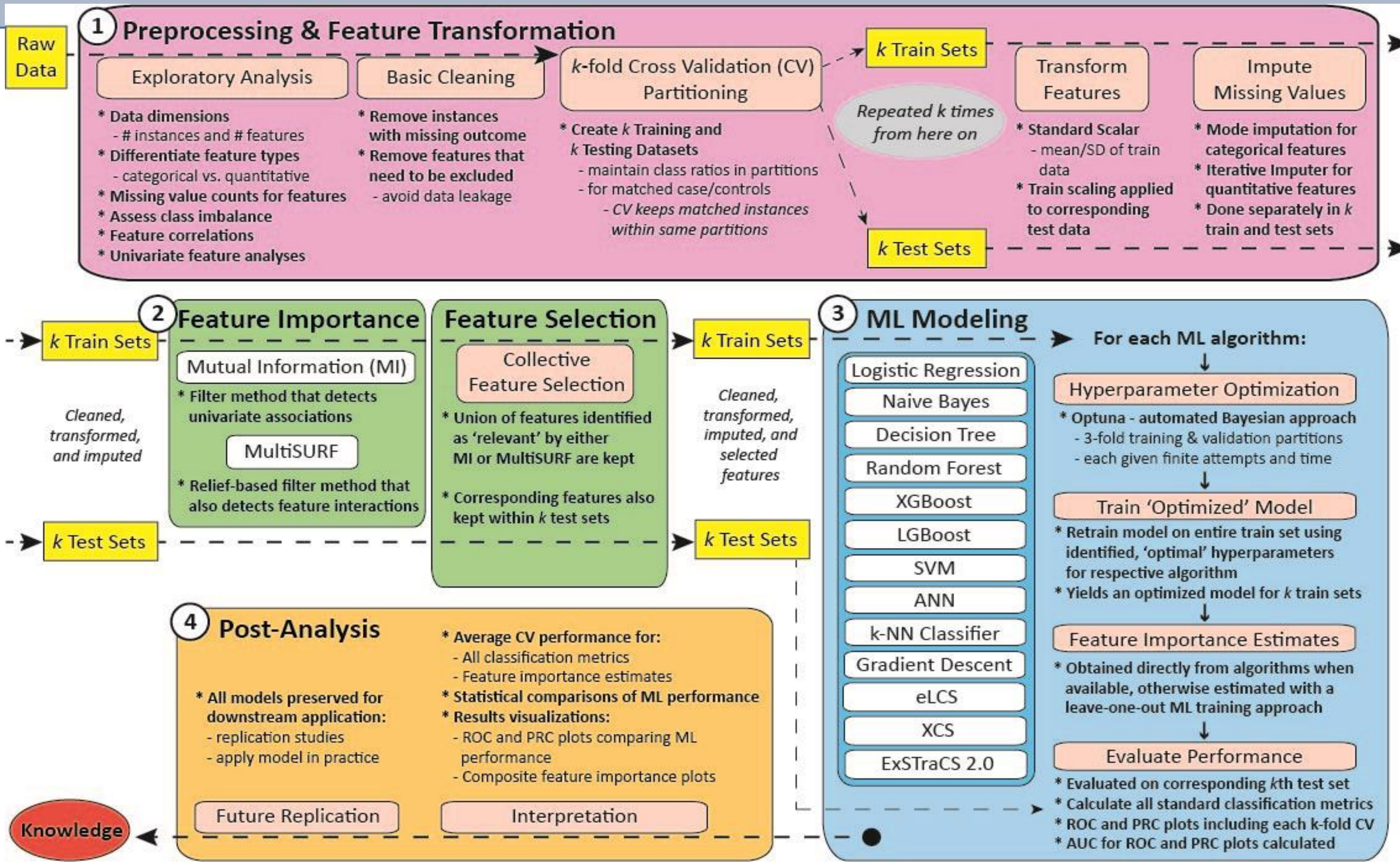


(d) 70-bit MUX

Scikit-Learn LCS Implementations

- Scikit-learn – is a very popular Python ML algorithm library
- Scikit-eLCS
  - https://github.com/UrbsLab/scikit-eLCS
- Scikit-XCS
  - https://github.com/UrbsLab/scikit-XCS
- Scikit-ExSTraCS
  - https://github.com/UrbsLab/scikit-ExSTraCS

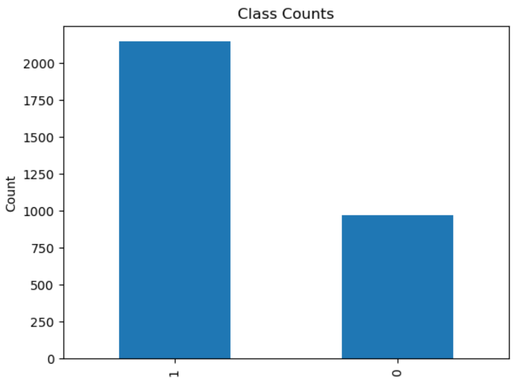
Automated ML Analysis Pipeline



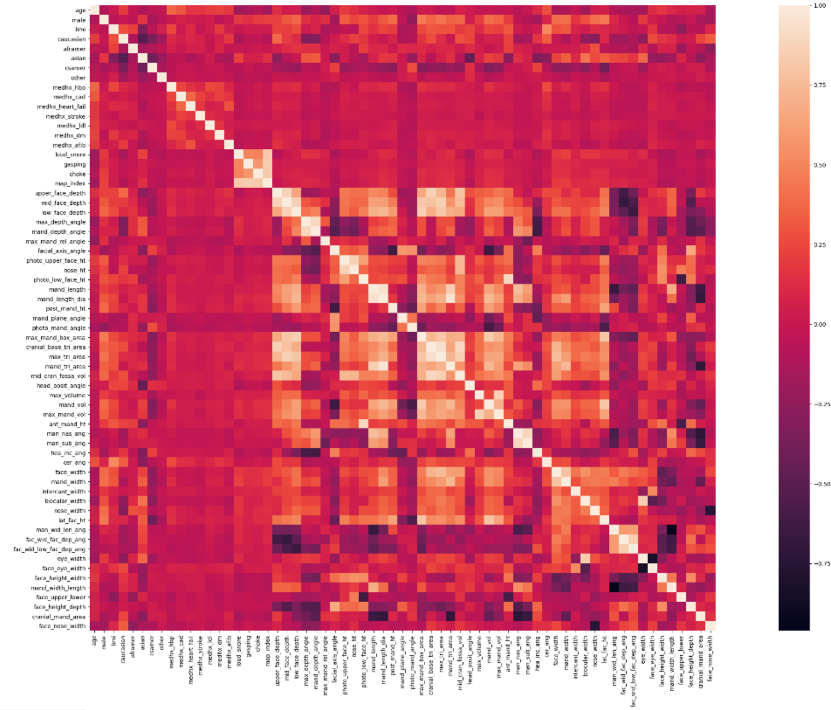
Preprint: Urbanowicz et al. A Rigorous Machine Learning Analysis Pipeline for Biomedical Binary Classification: Application in Pancreatic Cancer Nested Case-control Studies with Implications for Bias Assessments

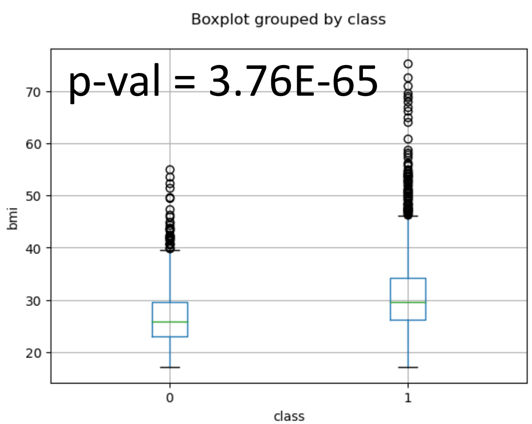
AutoMLPipe-BC: Exploratory Analysis

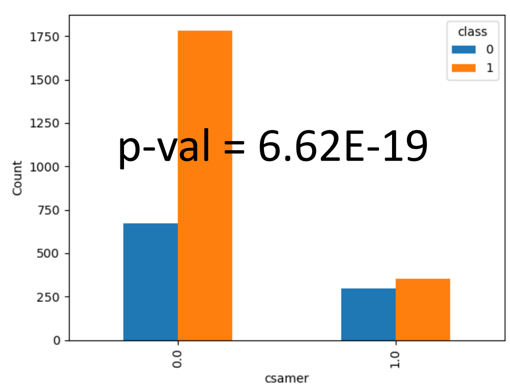
AHI>5: DEM+DX+SYM+CF



Variable	Count
instances	3111
features	65
categorical_features	16
quantitative_features	49







Categorical → Chi Square
Quantitative → MannWhitney U-Test

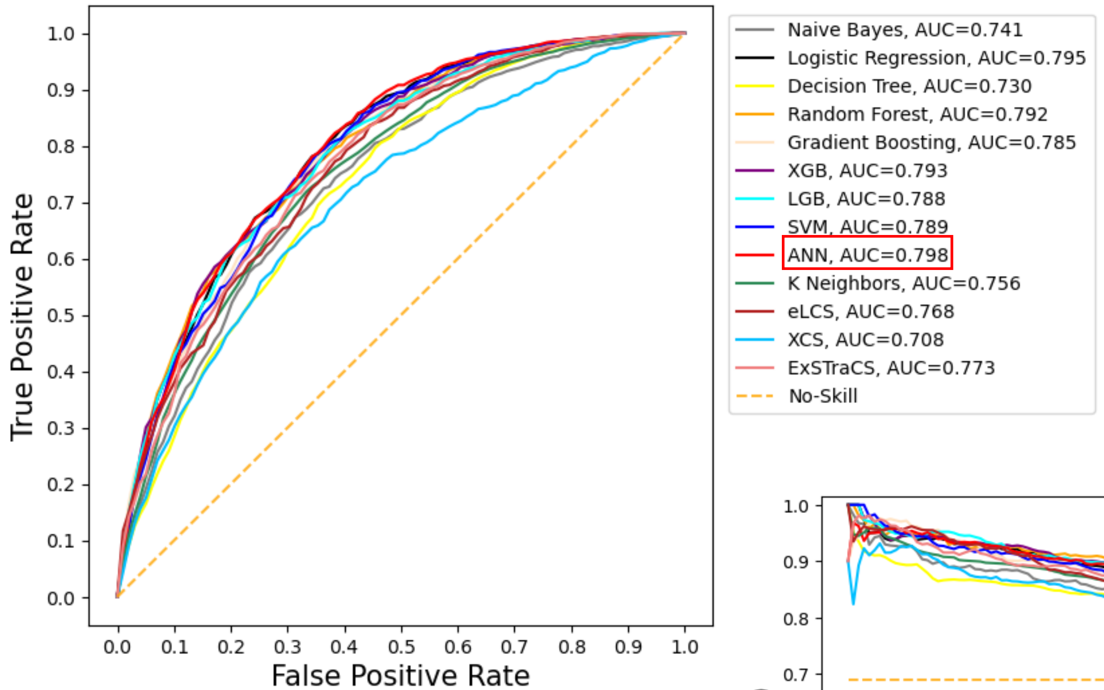
Modeling: Classification Evaluation Metrics

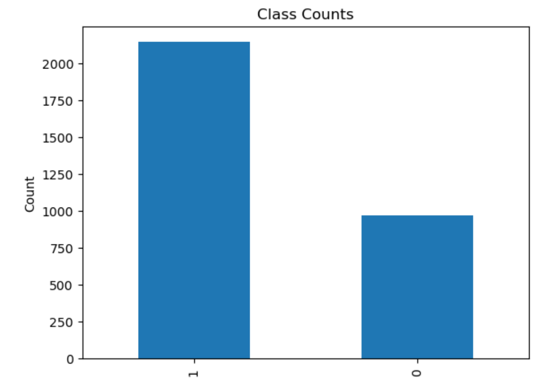
Average Model Prediction Statistics: D7 = SAGIC_GE5_DEMDXSYMCF_2021-04-28

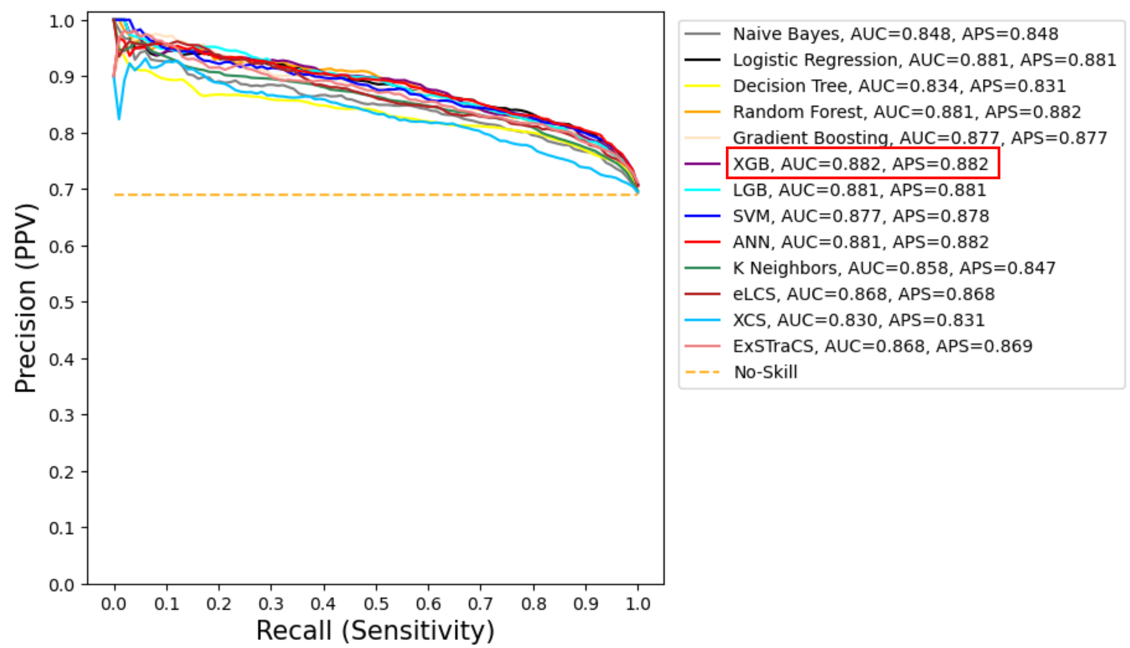
index	Balanced Accuracy	Accuracy	F1_Score	Sensitivity (Recall)	Specificity	Precision (PPV)	TP	TN	FP
Naive Bayes	0.679	0.6805	0.7464	0.683	0.6749	0.8234	146.3	65.4	31.5
Logistic Regression	0.7186	0.721	0.7812	0.725	0.7122	0.8478	155.3	69.0	27.9
Decision Tree	0.6684	0.6815	0.7519	0.7031	0.6337	0.8092	150.6	61.4	35.5
Random Forest	0.7026	0.7445	0.8143	0.8137	0.5914	0.8152	174.3	57.3	39.6
Gradient Boosting	0.685	0.7631	0.8383	0.8922	0.4778	0.7909	191.1	46.3	50.6
XGB	0.7059	0.7448	0.8134	0.8091	0.6027	0.8185	173.3	58.4	38.5
LGB	0.7063	0.7274	0.7935	0.7624	0.6502	0.8286	163.3	63.0	33.9
SVM	0.7122	0.7258	0.7894	0.7484	0.676	0.8361	160.3	65.5	31.4
ANN	0.703	0.774	0.8442	0.8912	0.5148	0.8031	190.9	49.9	47.0
K Neighbors	0.6737	0.7351	0.8123	0.8366	0.5107	0.7926	179.2	49.5	47.4
eLCS	0.692	0.7563	0.8298	0.8627	0.5212	0.7997	184.8	50.5	46.4
XCS	0.5729	0.7078	0.8136	0.9308	0.215	0.7265	199.4	20.8	76.1
ExSTraCS	0.6939	0.7268	0.7972	0.781	0.6068	0.815	167.3	58.8	38.1

index	FN	NPV	LR+	LR-	ROC_AUC	PRC_AUC	PRC_APS
Naive Bayes	67.9	0.4905	2.1464	0.4717	0.741	0.8477	0.8479
Logistic Regression	58.9	0.541	2.5892	0.3878	0.7953	0.8808	0.8814
Decision Tree	63.6	0.4931	1.9659	0.4739	0.7302	0.834	0.8313
Random Forest	39.9	0.5893	2.0209	0.3177	0.7915	0.8815	0.882
Gradient Boosting	23.1	0.6692	1.7202	0.2265	0.785	0.8766	0.8766
XGB	40.9	0.5905	2.0615	0.3175	0.7929	0.8819	0.8824
LGB	50.9	0.5549	2.2172	0.3657	0.7883	0.8806	0.8811
SVM	53.9	0.551	2.3251	0.3727	0.7888	0.8775	0.878
ANN	23.3	0.6921	1.8611	0.2075	0.7981	0.8814	0.8822
K Neighbors	35.0	0.5963	1.7544	0.3116	0.7563	0.8578	0.8466
eLCS	29.4	0.6326	1.8282	0.2669	0.7677	0.8676	0.8683
XCS	14.8	0.5754	1.221	0.2482	0.7081	0.8299	0.8312
ExSTraCS	46.9	0.558	2.0161	0.3614	0.7728	0.868	0.8688

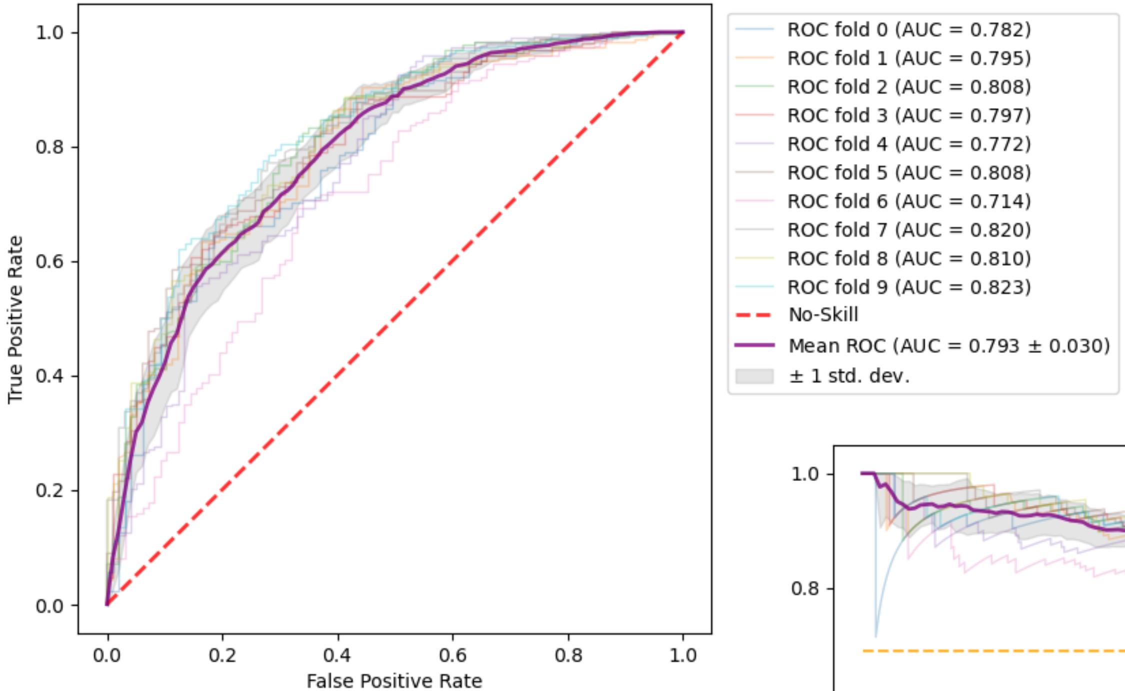
Modeling: Average - ROC/PRC

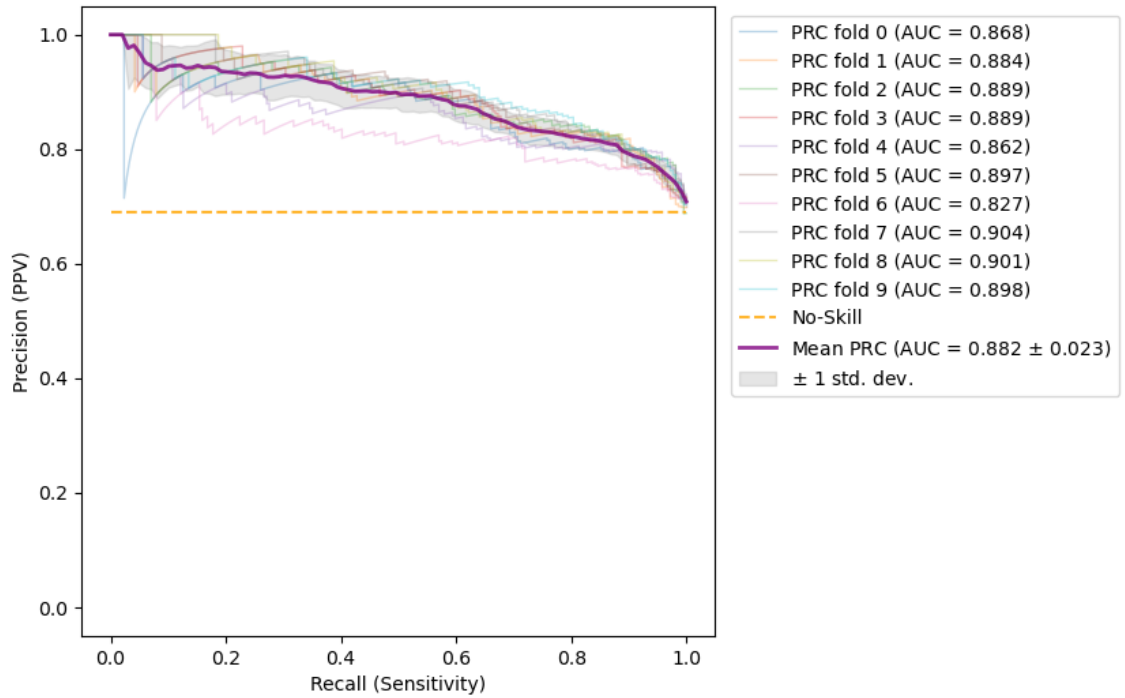




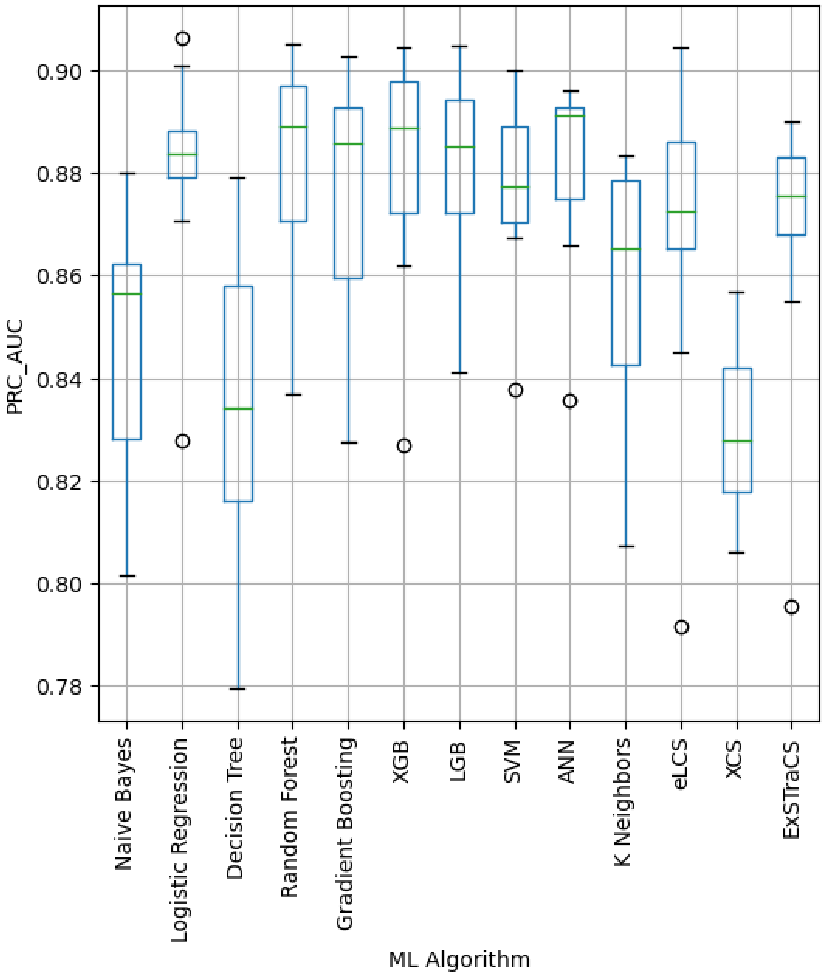


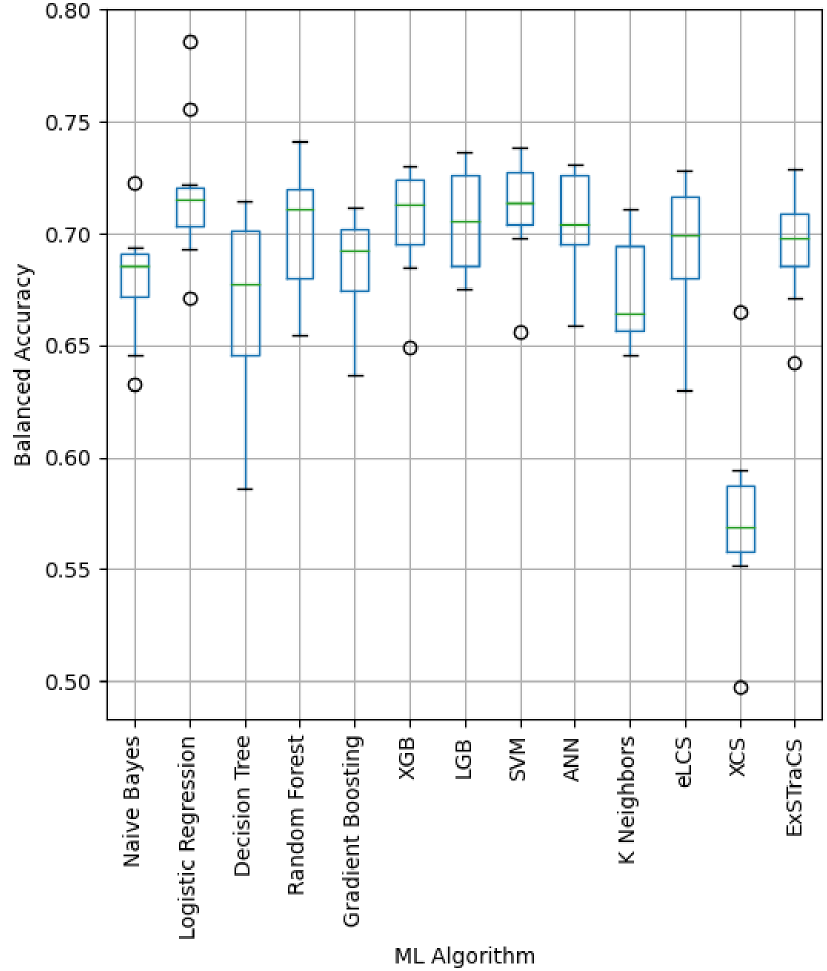
Modeling: XGB – ROC/PRC





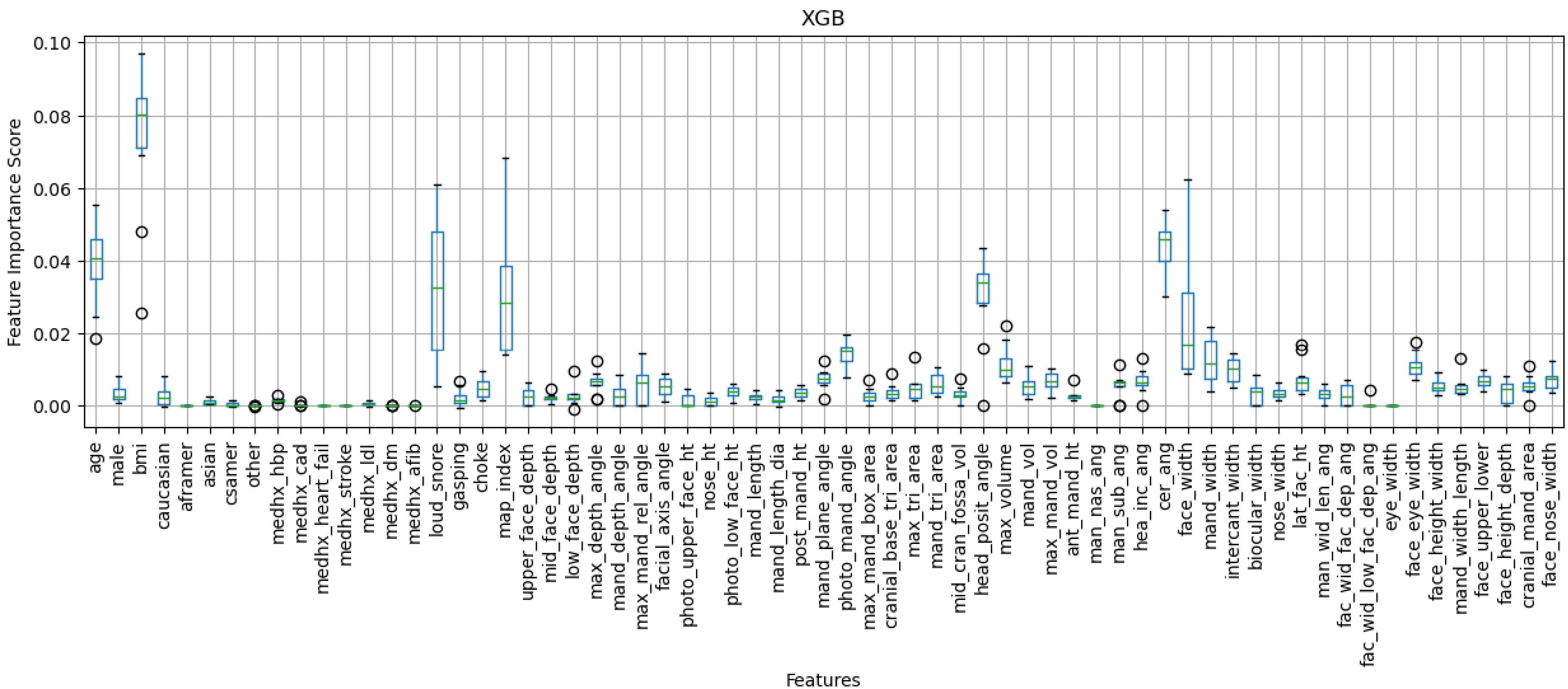
Modeling: Comparing ML Statistics



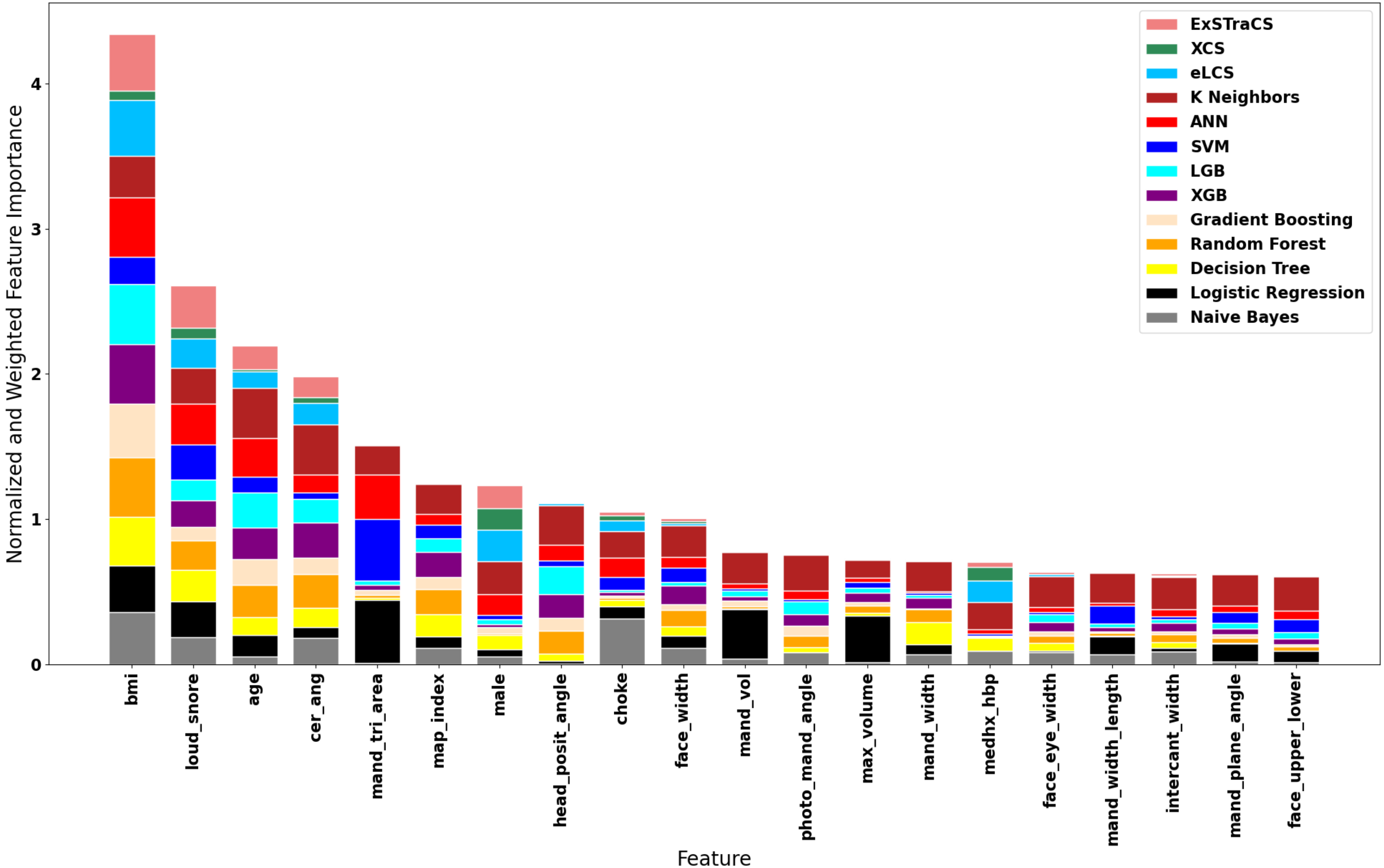


Modeling: Feature Importance

- Internal Feature Importance Estimators (if available)
- Permutation Feature Importance Estimation
  - Applied for all ML algorithms or whenever internal estimators unavailable



Modeling: Composite FI Plot

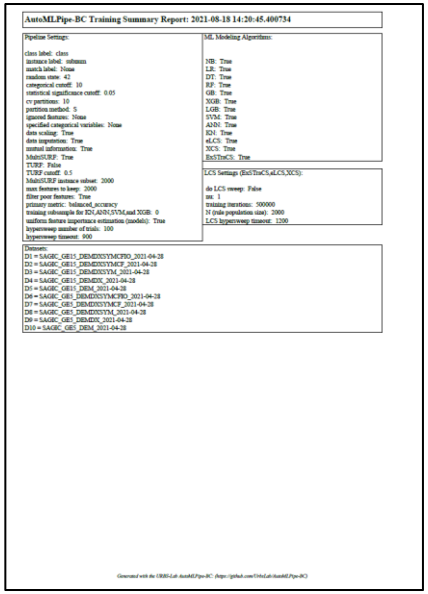


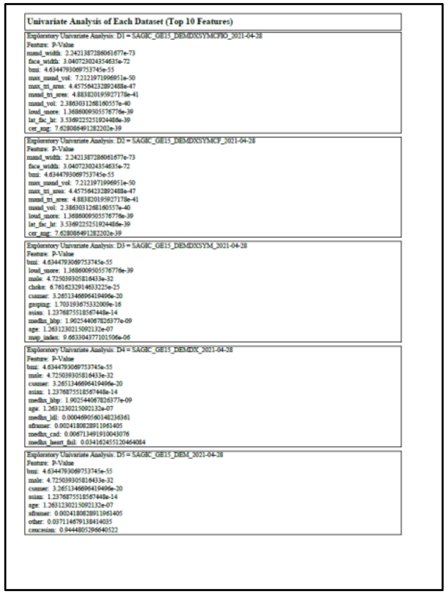
Statistical Comparisons

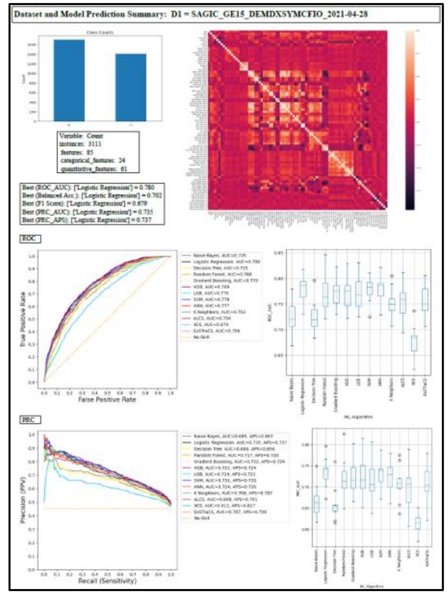
- Nonparametric statistical comparisons
  - Kruskall Wallis: one-way ANOVA on ranks is a non-parametric method for testing whether samples originate from the same distribution
  - Wilcoxon Rank Sum: test whether two samples (pairwise) are likely to derive from the same population

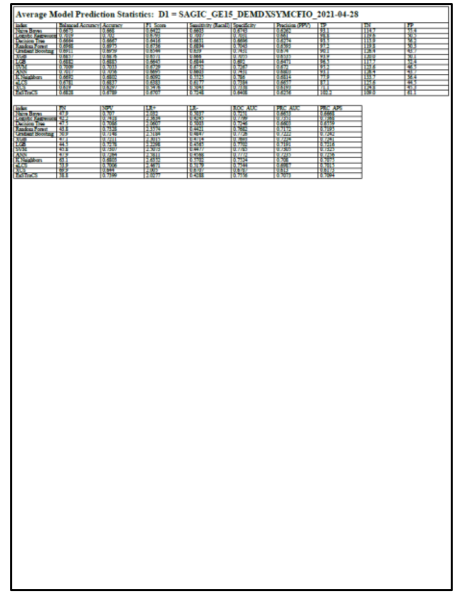
- Performed for
  - For each metric (across all algorithms)
  - For each metric (between datasets)
  - For each metric (comparing best performing algorithms)

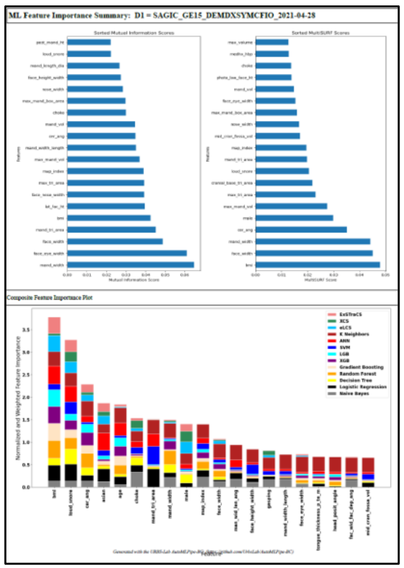
PDF Pipeline Summary Report

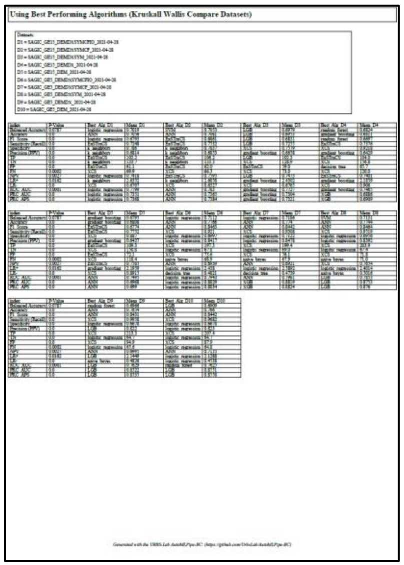

AutoMLPipe-BC Training Summary Report: 2021-08-18 14:20:45.400734

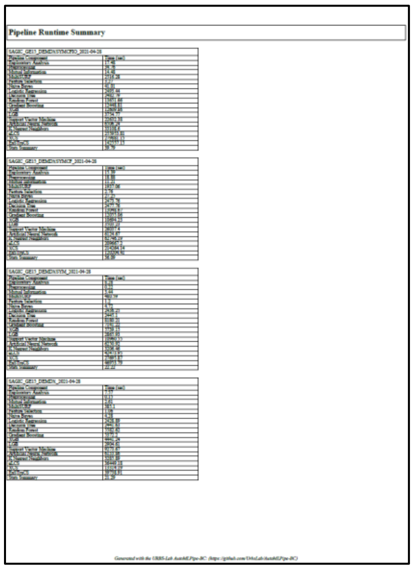

Univariate Analysis of Each Dataset (Top 10 Features)


Dataset and Model Prediction Summary: D1 = SAGIC_GE15_DEMDXSYMCFIO_2021-04-28


Average Model Prediction Statistics: D1 = SAGIC_GE15_DEMDXSYMCFIO_2021-04-28


ML Feature Importance Summary: D1 = SAGIC_GE15_DEMDXSYMCFIO_2021-04-28


Using Best Performing Algorithms (Kruskall Wallis Compare Datasets)


Pipeline Runtime Summary

Implementation

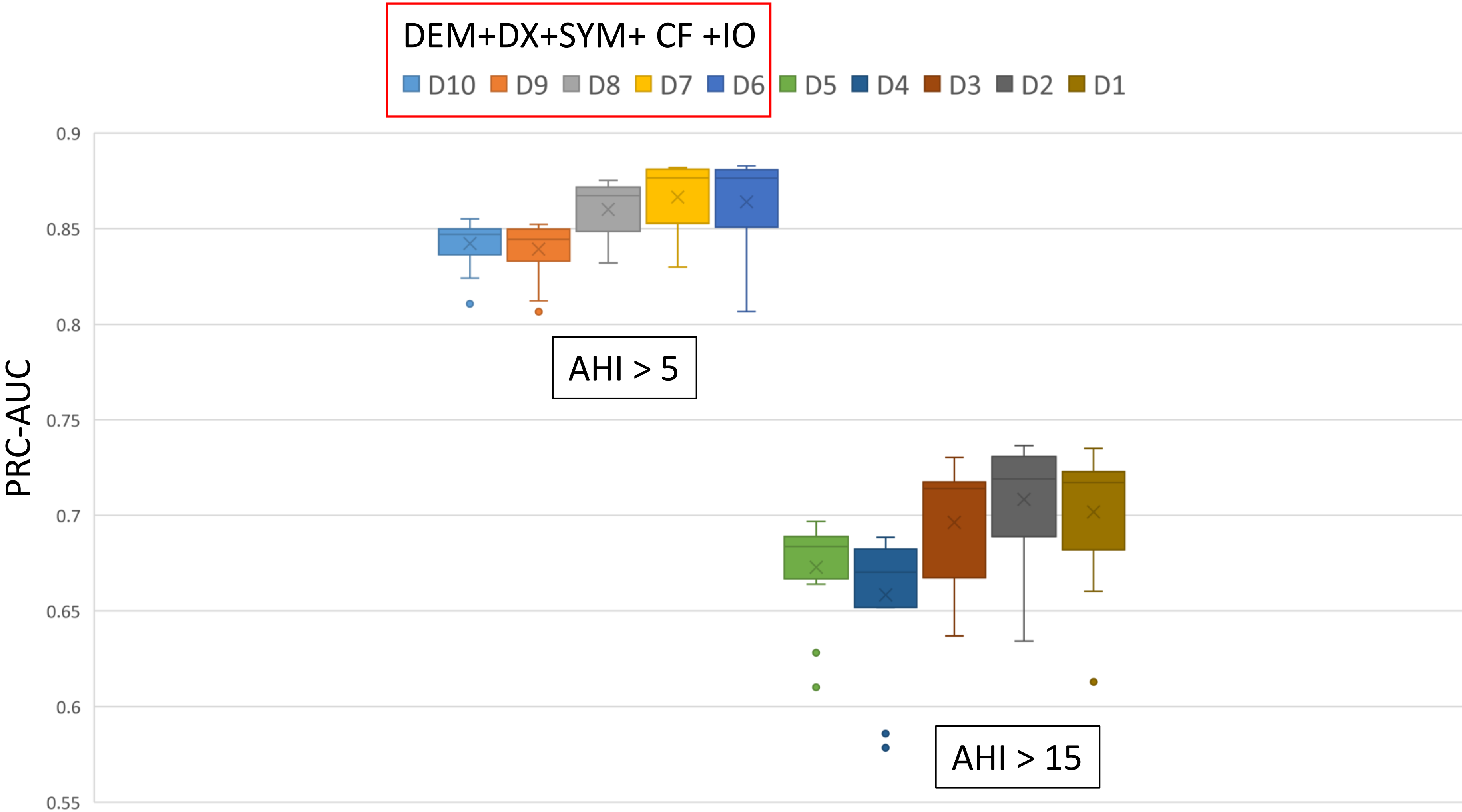
https://github.com/UrbsLab/AutoMLPipe-BC

- Detailed README
  - Accessible to use for those with very basic Python experience
- Local (serially)
  - Jupyter notebook
  - Command line
- Computing Cluster (In Parallel)
- PostHoc Jupyter Notebooks:
  - Regenerate figures
  - Evaluate models using alternative decision thresholds
  - Show decision tree models

Phase 1: Exploratory Analysis
Phase 2: Data Preprocessing
Phase 3: Feature Importance Evaluation
Phase 4: Feature Selection
Phase 5: Machine Learning Modeling
Phase 6: Statistics Summary
Phase 7: [Optional] Compare Datasets
Phase 8: [Optional] Copy Key Files
Phase 9: [Optional] Generate PDF Training Summary Report
Phase 10: [Optional] Apply Models to Replication Data
Phase 11: [Optional] Generate PDF 'Apply Replication' Summary Report

Average PRC-AUC Across All Datasets

Each box includes average PRC-AUC for all 13 ML Algorithms



Acknowledgements

Current Collaborators:

Loren Gragert
John Holmes
Jason Moore
Malek Kamoun
Elizabeth Goldmuntz
Deanne Taylor
Alan Pack
Ulysses Magalang
Shannon Lynch
Rachael Stolzenberg-Solomon
Danielle Mowery
Diego Mazzotti
Philip Gehrman
Susan Yoon
Blanca Himes
Steven Kawut

Students:

Robert Zhang
Satvik Desariraju
Vivek Sriram
Alexa Woodward
Sy Hwang

Penn
University of Pennsylvania

Department of Biostatistics Epidemiology & Informatics

Institute for Biomedical Informatics

Lab GitHub: https://github.com/UrbsLab
YouTube: https://www.youtube.com/channel/UCHIKWNLhglKHJpkxw993sKQ

Funding: (NIH) R01 HL134015
(DOD) W81XWH-17-1-0276
UPenn award for Faculty Mentoring of Undergraduate Research

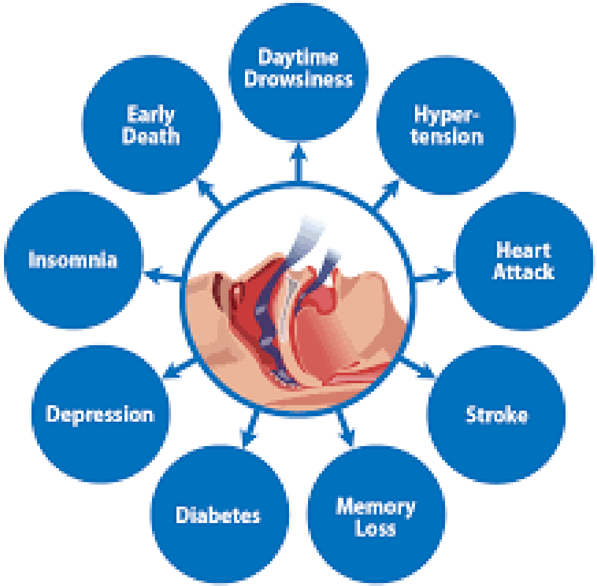
Application to Obstructive Sleep Apnea (OSA)

- Sleep Apnea Global Interdisciplinary Consortium (SAGIC)
  - 3111 de-identified subjects who underwent polysomnography (PSG)

- Target Outcome: Apnea-hypopnea index (AHI)
  - Events/hour
  - OSA diagnosis:
    - Liberal Cutoff: AHI > 5 (Controls: 969, Cases: 2142)
    - Conservative Cutoff: AHI > 15 (Controls: 1410 , Cases: 1701)

- Features (5 categories, 85 total):
  - Demographics (DEM) - 8
  - Comorbidities (DX) -7
  - Symptoms (SYM) - 4
  - Craniofacial Photos (CF) - 46
  - Intraoral Photos (IO) - 39

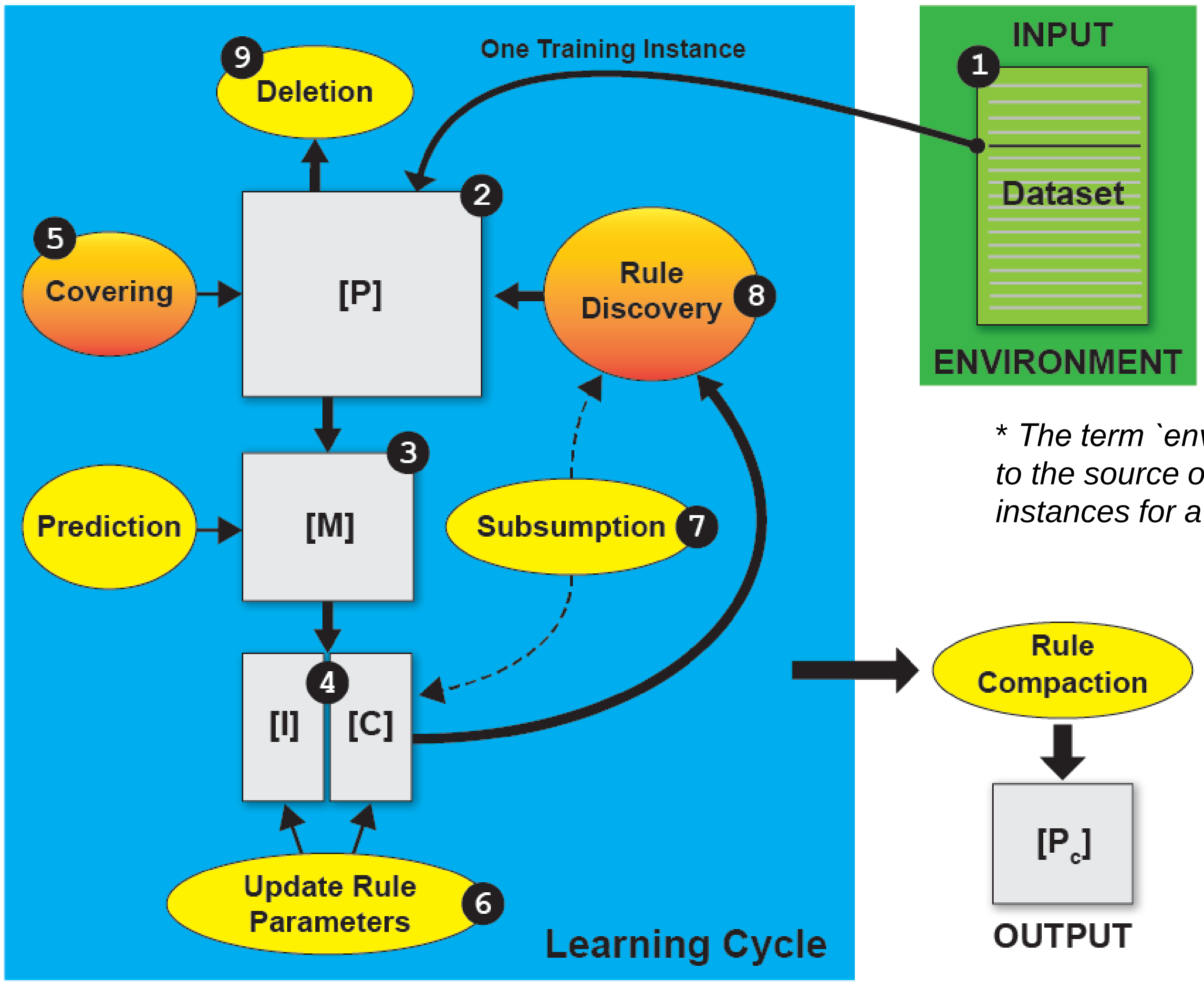
- Comparing sequential addition of feature categories
  - DEM + DX + SYM + CF + IO



AutoMLPipe-BC Goals

- Automate and parallelize phases of ML analysis pipeline for binary classification (BC)
  - Obtain a rigorous and thorough look at performance
- Capability to handle big data
  - Feature selection is key
- Capability to deal with typical data challenges
  - Missing data
  - Imbalanced data
  - Data normalization for ML efficacy and interpretability
- Capability to take complex associations into account
- Provide standard of comparison for other AutoML tools
- Compare different ML modeling algorithm performance
- Compare performance between target datasets
  - Addition of different feature types
  - Adjusting (or not) for covariates
- Interpret models
  - Feature importance (within and across models/algorithms)
  - Characterizing associations
- Apply models to future data

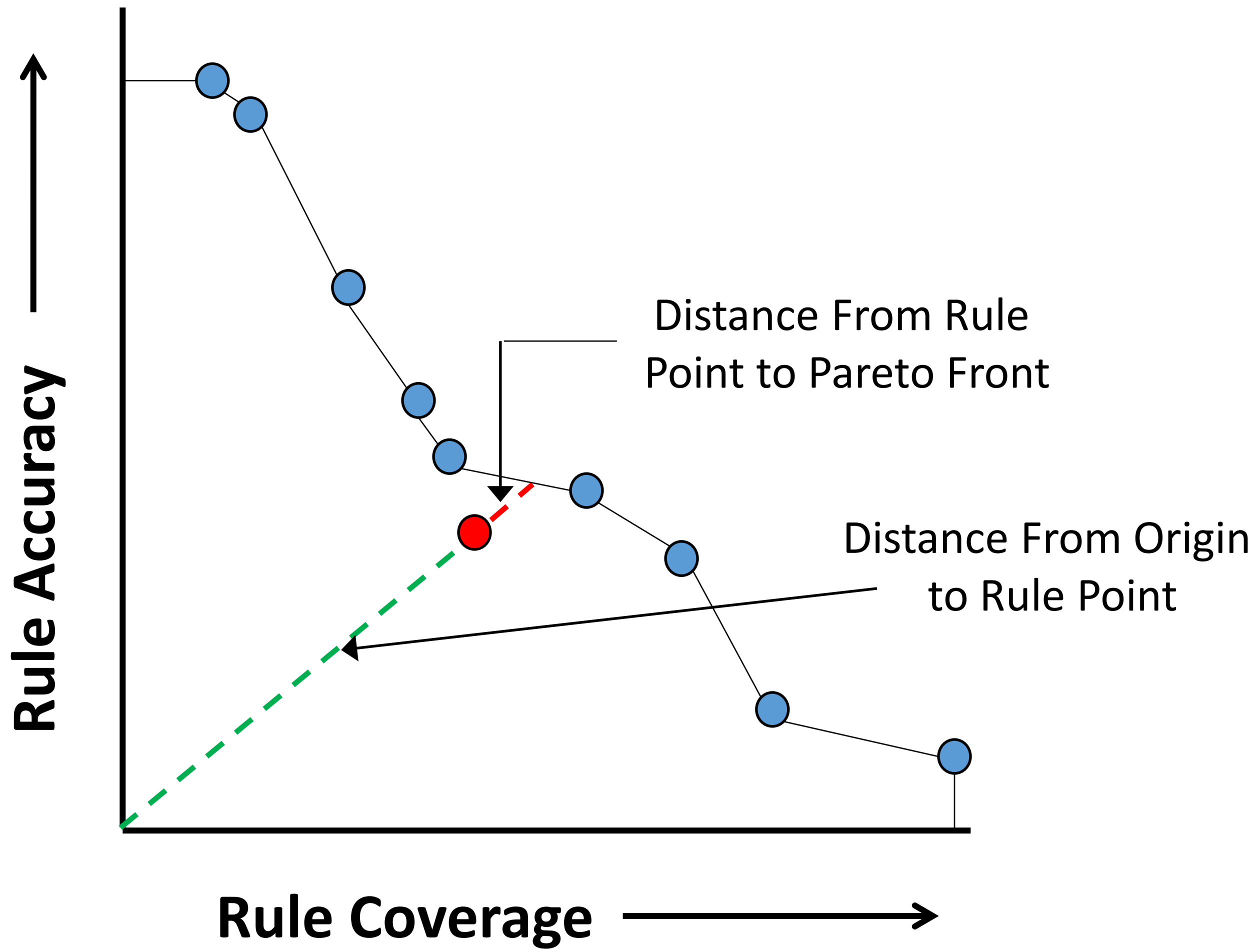
LCS – A General Schematic (Supervised)



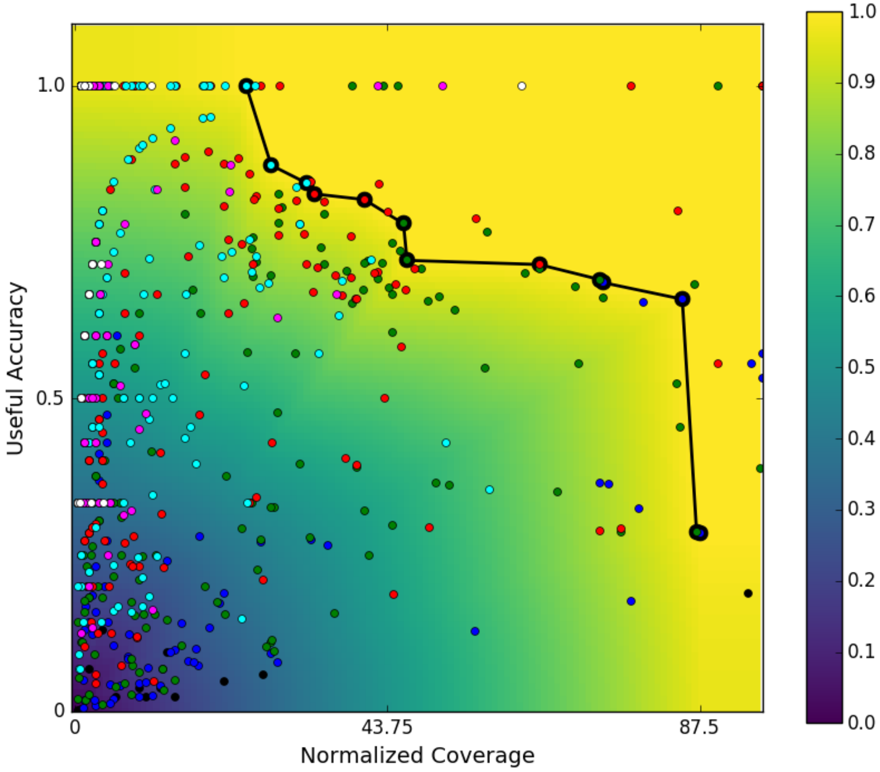
** The term `environment' refers to the source of training instances for a problem/task.*

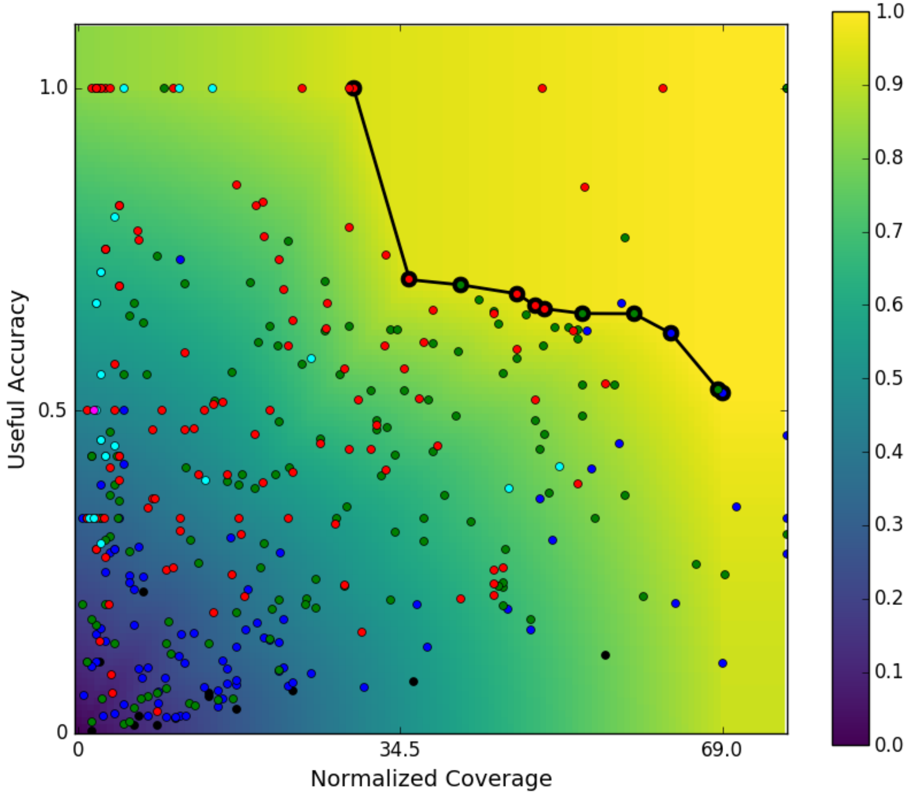
Urbanowicz, Ryan J., and Will N. Browne. Introduction to learning classifier systems. Springer, 2017.

LCS Rule-fitness ‘front’ – Calculate fitness?

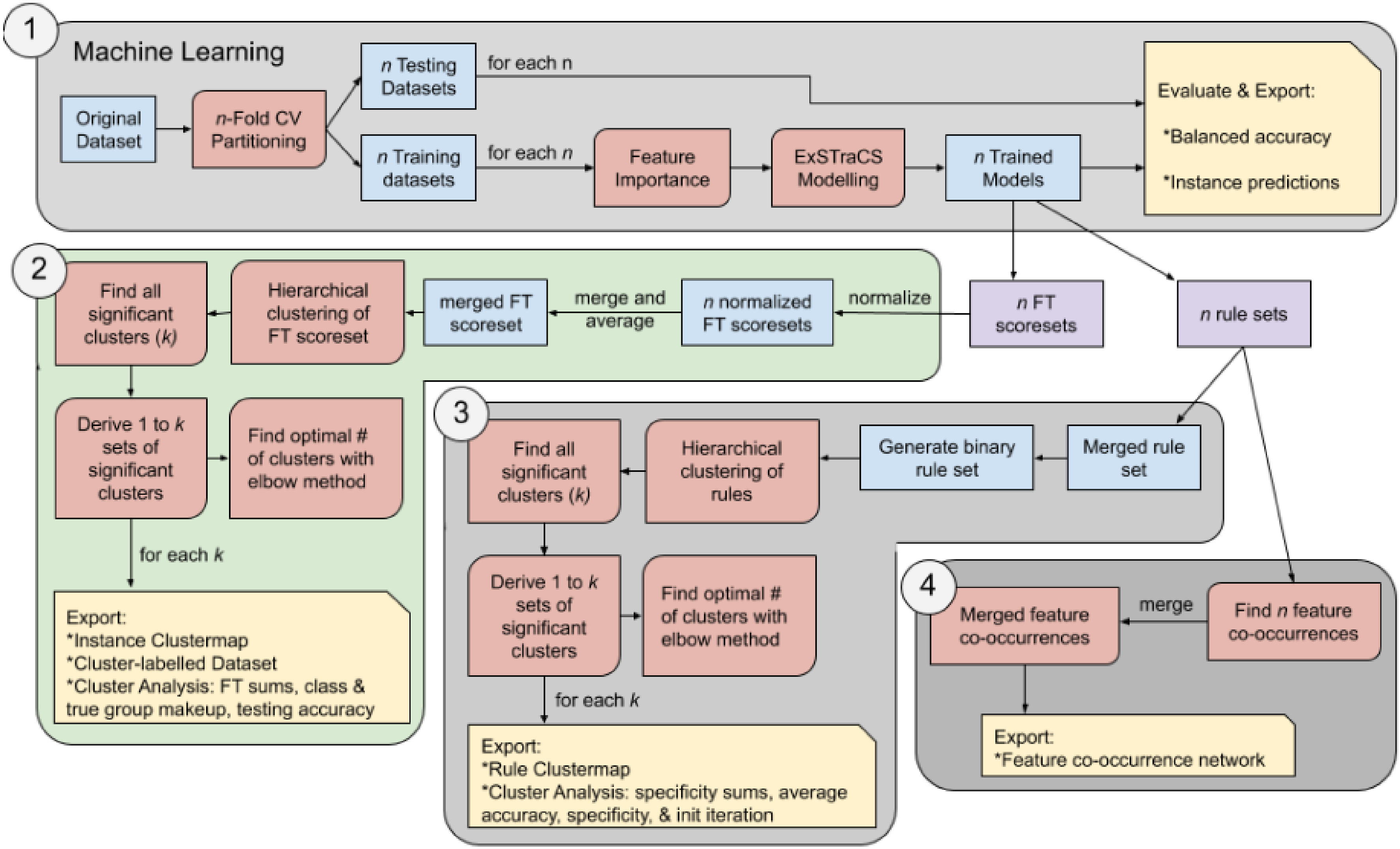


Data-set specific rule-fitness gradient





LCS-DIVE: Automated Rule-based Machine Learning Visualization Pipeline

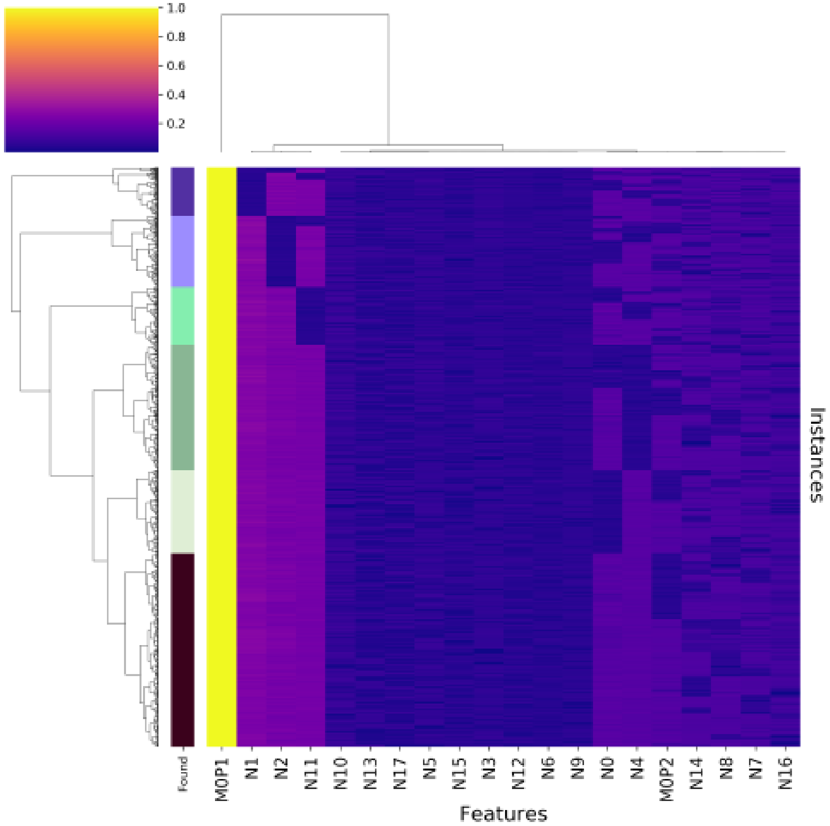


LCS-DIVE: Association Signatures

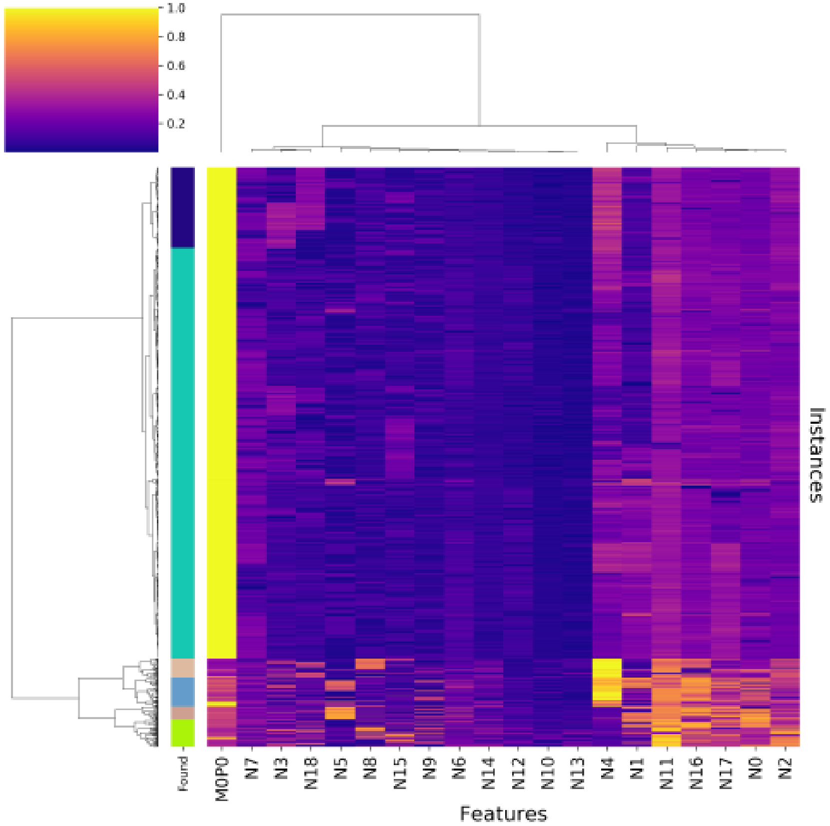
Table 2: Characteristics of the 21 GAMETES Simulated Datasets

Dataset ID	Underlying Association	Predictive Features	# of Models	Model Ratio	Model Heritability	Instances
1 & 2	univariate	1	1	NA	1 & 0.4	1600
3 & 4	additive	2	2	equal	1 & 0.4	1600
5 & 6	additive	4	4	equal	1 & 0.4	1600
7 & 8	2-way pure epistasis	2	1	NA	1 & 0.4	1600
9 & 10	3-way pure epistasis	3	1	NA	1 & 0.2	3200
11 & 12	additive (2-way epistasis + univariate)	4	3	equal	1 & 0.4	1600
13 & 14	additive (2, 2-way epistasis)	4	2	equal	1 & 0.4	1600
15 & 16	heterogeneity (univariate)	2	2	equal	1 & 0.4	1600
17 & 18	heterogeneity (univariate)	4	4	equal	1 & 0.4	1600
19 & 20	heterogeneity (2, 2-way epistasis)	4	2	equal	1 & 0.4	1600
21	heterogeneity (2, 2-way epistasis)	4	2	75 : 25	0.4	1600

LCS-DIVE: Univariate (1 Feature)

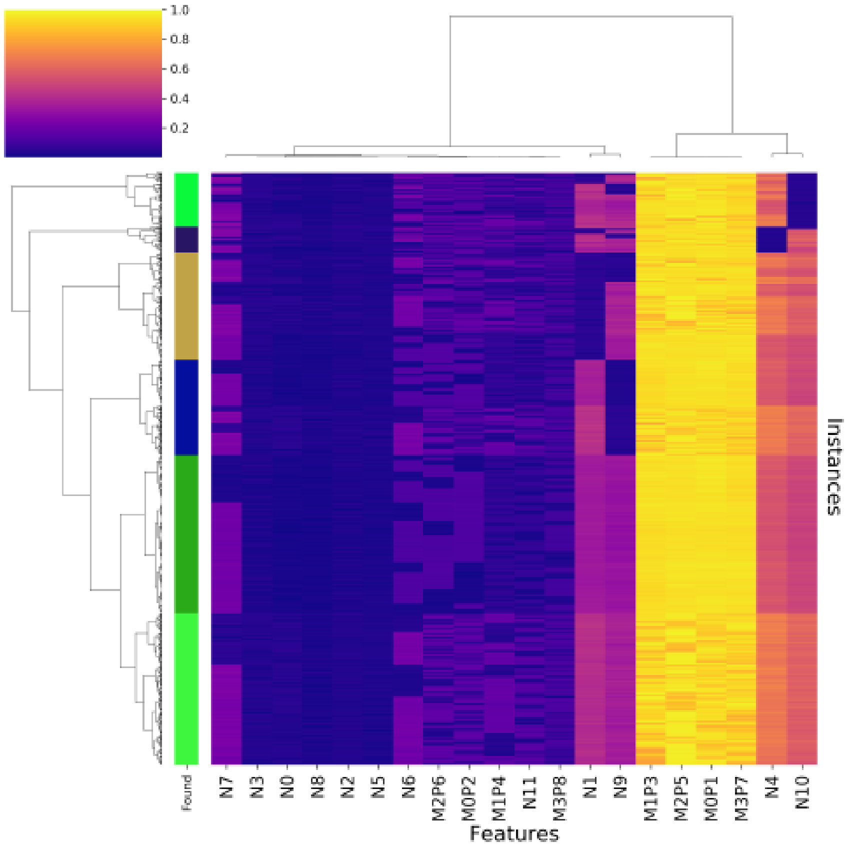


(a) D1: Clean Univariate

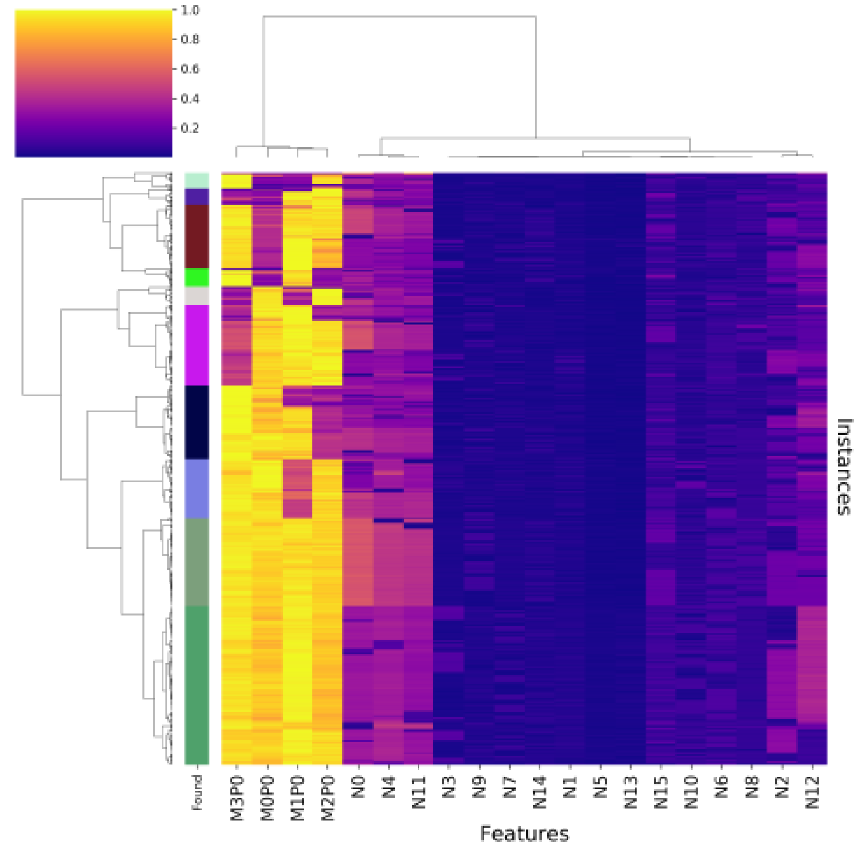


(b) D2: Noisy Univariate

LCS-DIVE: Univariate-Additive (4 Features)

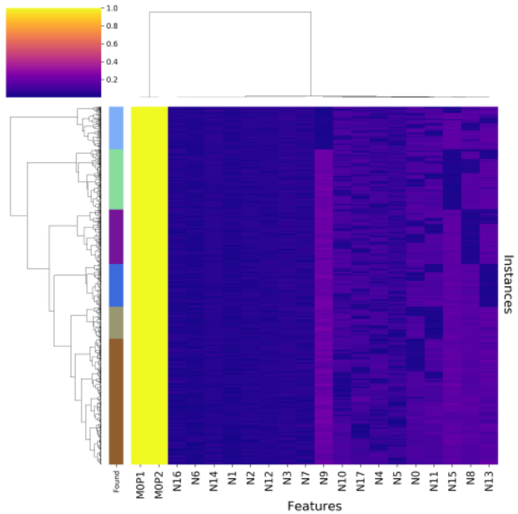


(e) D5: Clean 4-feature additive

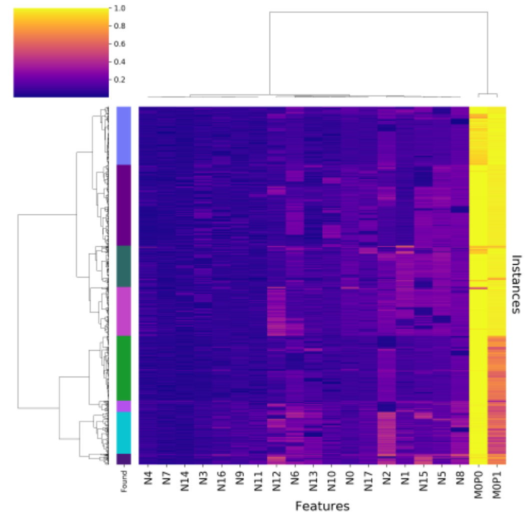


(f) D6: Noisy 4-feature additive

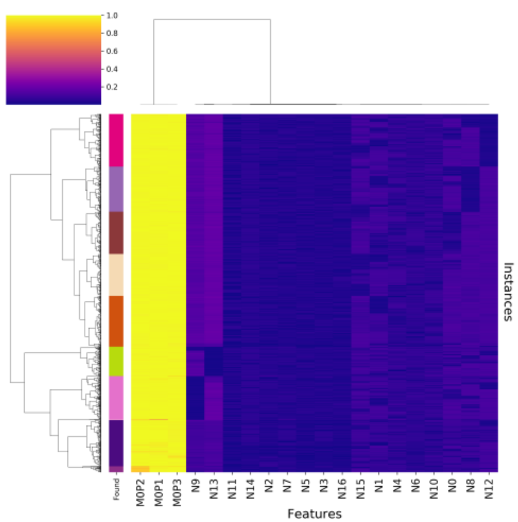
LCS-DIVE: Epistasis



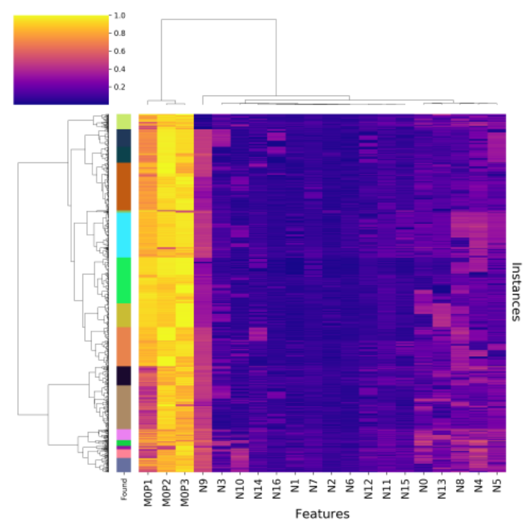
(a) D7: Clean 2-way pure epistasis



(b) D8: Noisy 2-way pure epistasis

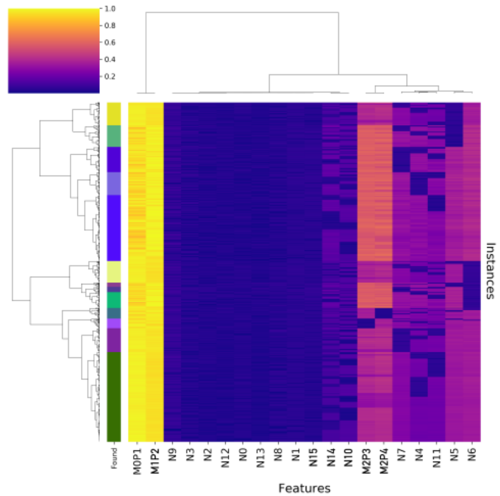


(c) D9: Clean 3-way pure pistasis

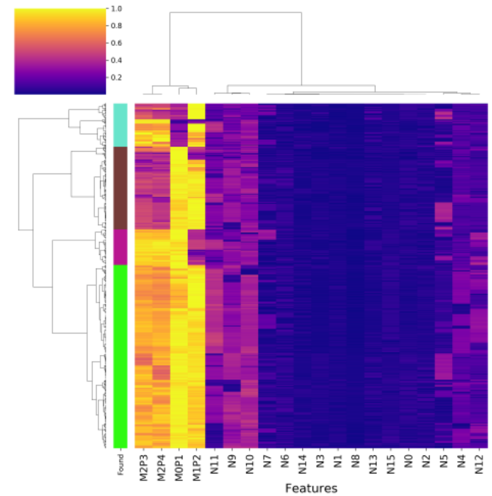


(d) D10: Noisy 3-way pure epistasis

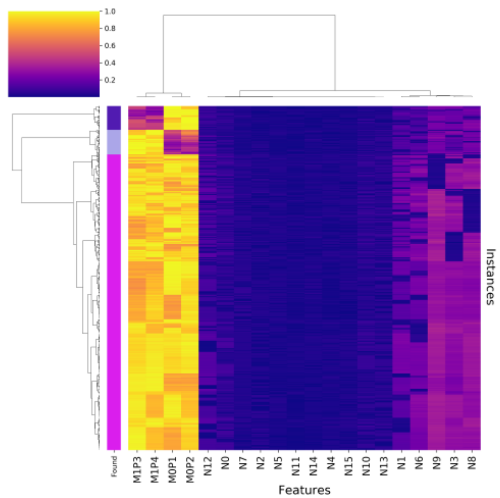
LCS-DIVE: Epistasis+Additive



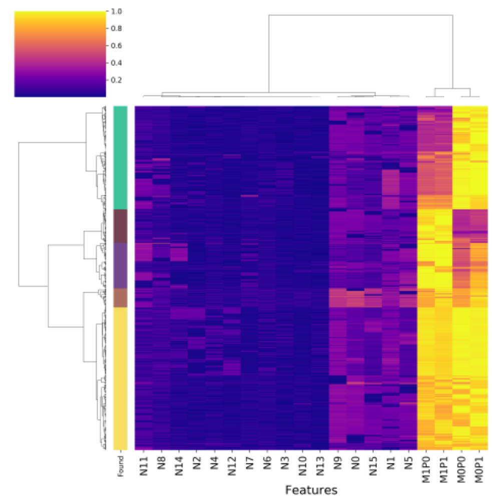
(a) D11:Clean additive (2-way epistasis+univariate)



(b) D12: Noisy additive (2-way epistasis+univariate)

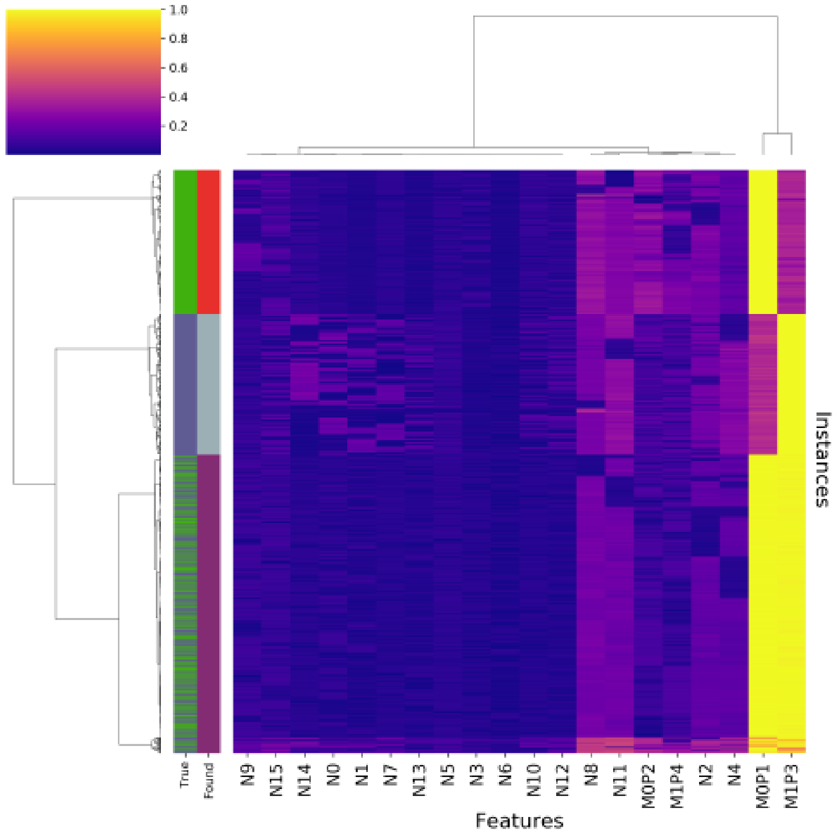


(c) D13: Clean additive (2, 2-way epistasis)

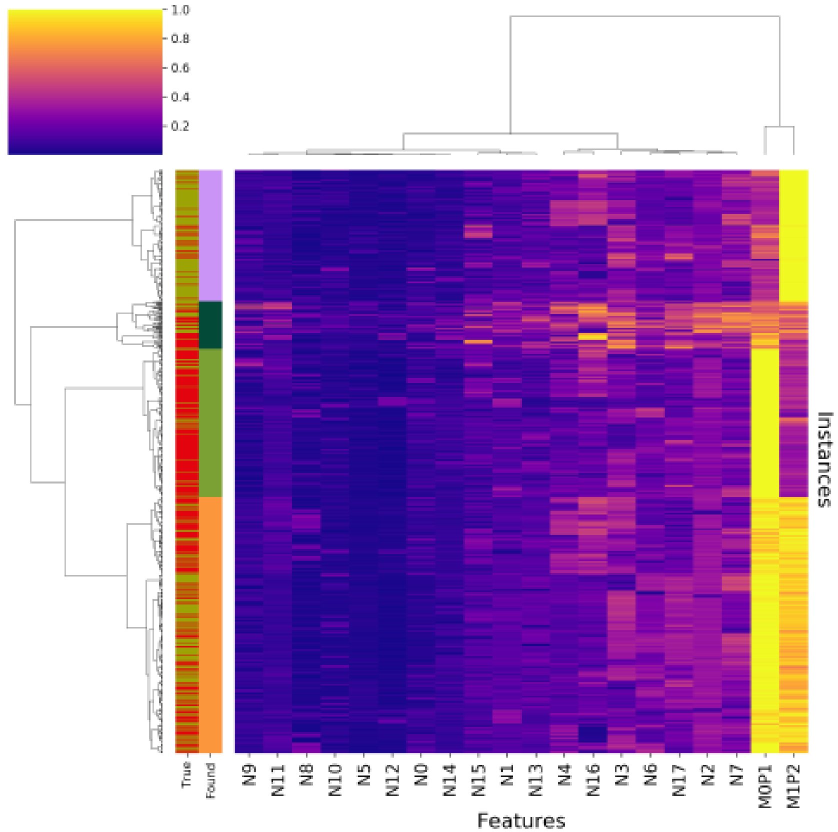


(d) D14: Noisy additive (2, 2-way epistasis)

LCS-DIVE: Heterogeneous-Univariate

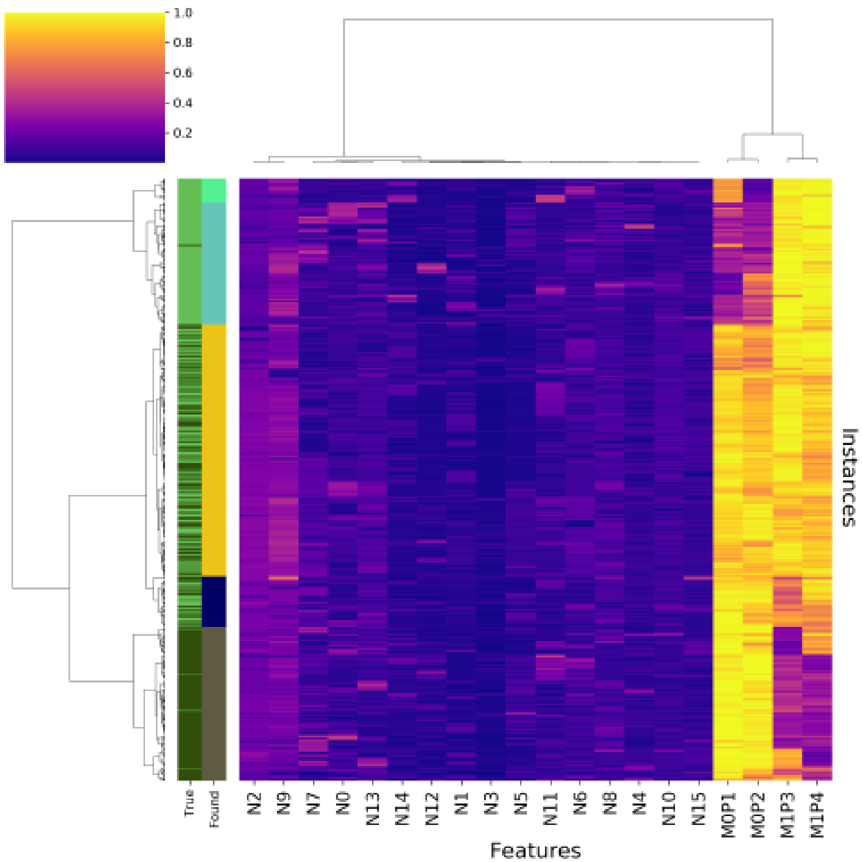


(a) D15: Clean 2-feature het. univariate

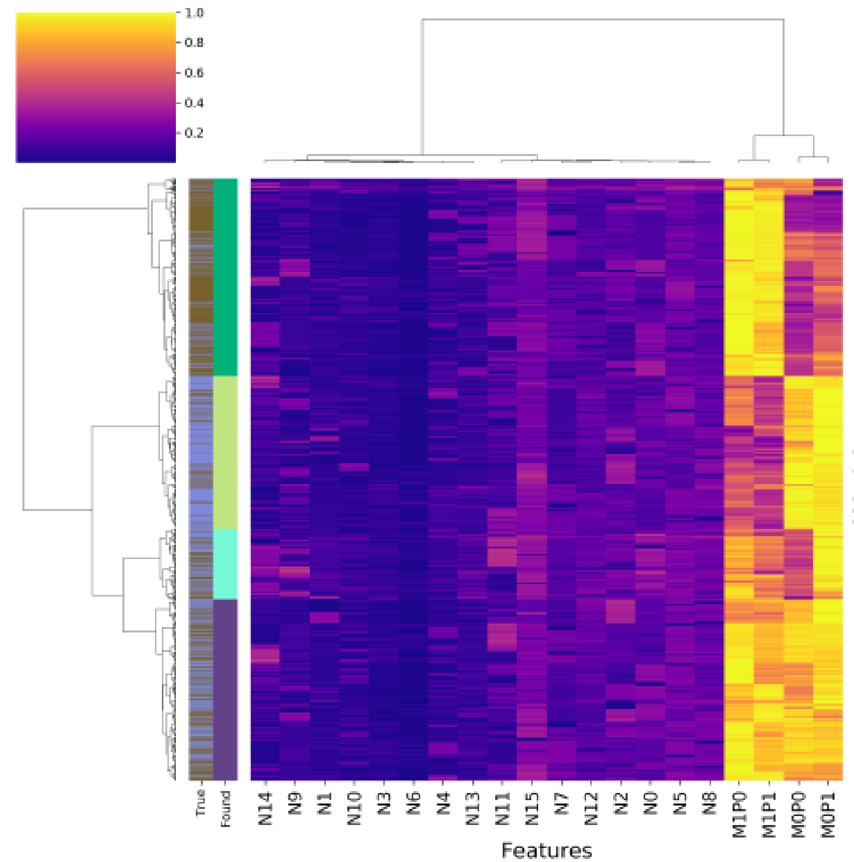


(b) D16: Noisy 2-feature het. univariate

LCS-DIVE: Heterogeneous-Epistasis



(e) D19: Clean 2, 2-way Epistasis, het.



(f) D20: Noisy 2, 2-way Epistasis, het.