AutoPrognosis

Automating the design of predictive models for clinical risk and prognosis

Mihaela van der Schaar

University of Oxford

The Alan Turing Institute

Acknowledgements

- Ahmed Alaa
- Jinsung Yoon
- Prof. William Zame
- Many clinical collaborators

Research Goal

Improve Quality and Safety of Healthcare while Managing Costs using Machine Learning

The Policy Perspective

- Population-serving
- Cross-sectional

The Clinical Perspective

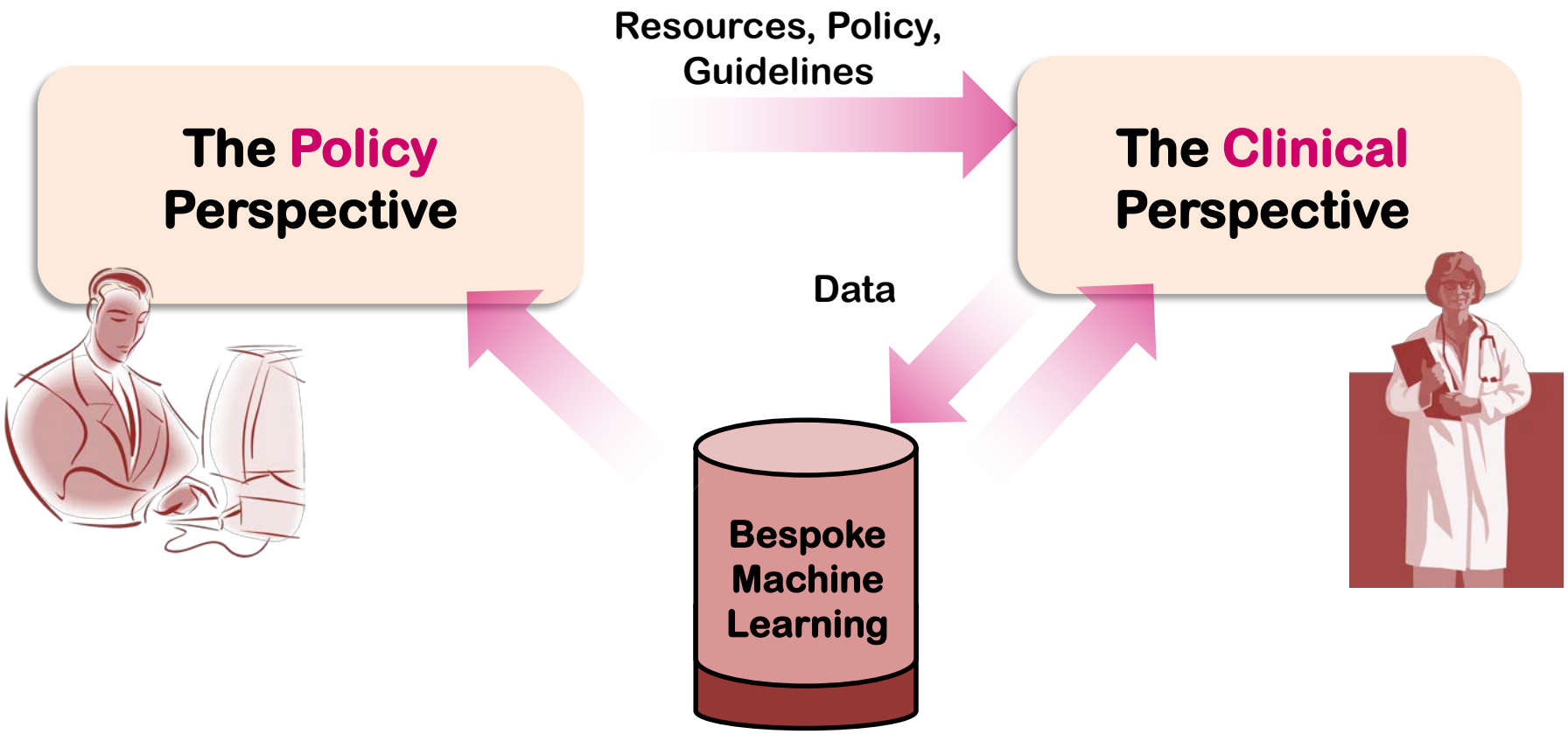
- Personalized

Guidelines, Policies, Standards

Clinical Decision Support
Actionable Intelligence

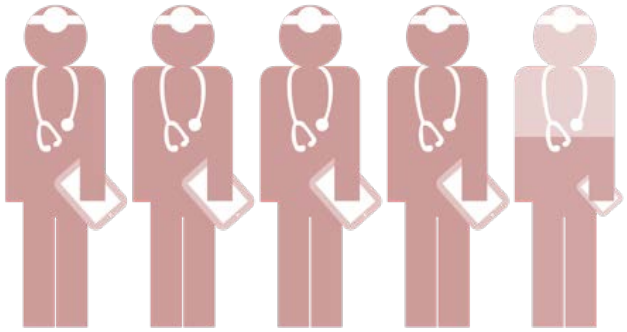
Research Goal

Learning, Co-Evolving, Improving Health Systems



The Clinical Perspective: Decision Support Systems to Improve Patient Care

- **Goal:** develop machine learning algorithms to extract actionable intelligence in order to improve clinical practice



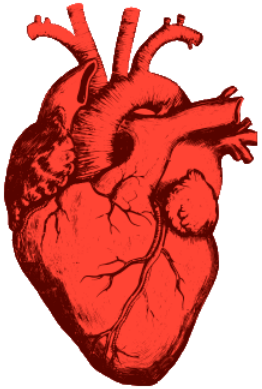
Observational data

Actionable intelligence
(Predictions, recommendations, practice guidelines, treatment effects, etc)

Diagnosis and Prognosis → **Screening and testing** → **Treatments and interventions**

Who should get a heart?

Ann



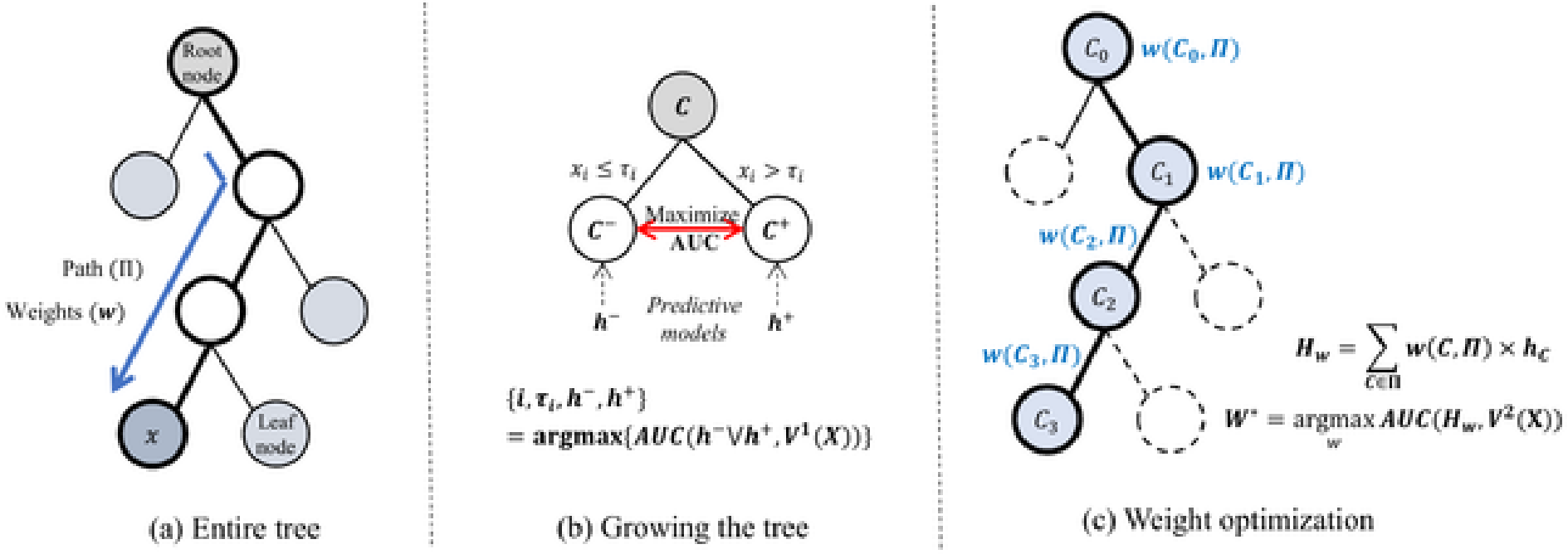
Bob

- **Urgency**
  - How long will Ann/Bob survive while waiting?
- **Benefit**
  - How much will Ann/Bob benefit from this heart?

Clinical Risk Scores

- **Urgency: Survival on Wait List**
  - HFSS
  - MAGGIC
  - SHFM
- **Benefit: Survival after Transplantation**
  - DRI
  - IMPACT
  - RSS

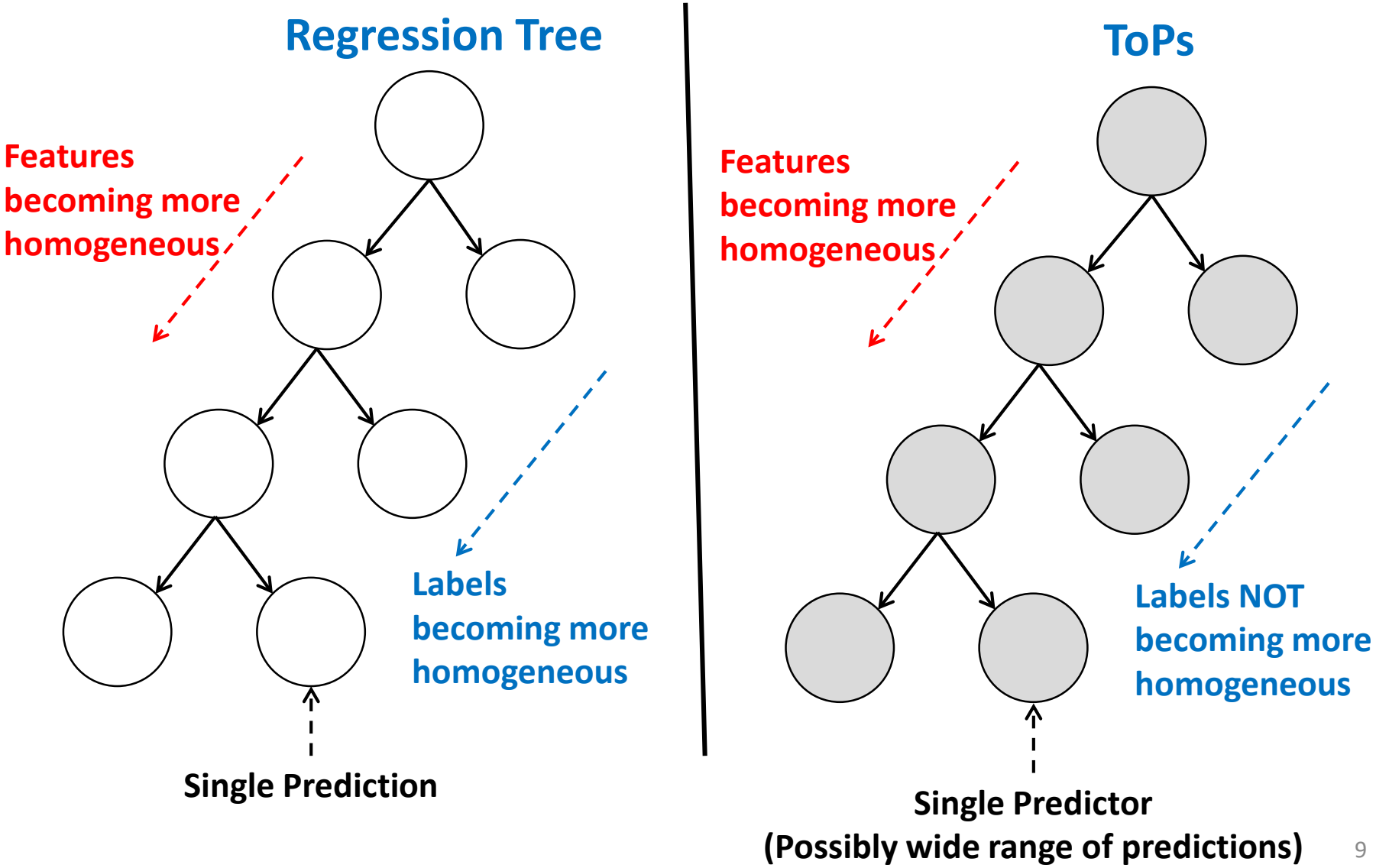
Personalized survival predictions via Trees of Predictors (ToPs)



Yoon J, Zame WR, van der Schaar, M. (2018) ToPs: Ensemble Learning with Trees of Predictors. *Trans. on Signal Processing*

Yoon J, Zame WR, Banerjee A, Cadeiras M, Alaa AM, van der Schaar, M. (2018) Personalized survival predictions via Trees of Predictors: An application to cardiac transplantation. *PLOS ONE* 13(3): e0194985. https://doi.org/10.1371/journal.pone.0194985

ToPs is NOT a regression tree!



Dataset

United Network for Organ Transplantation (UNOS)

- ALL patients registered for heart transplantation in US in 1985-2015
- 35,000+ patients wait-listed but did not receive heart transplant
  - Date of waitlisting + survival
  - 33 features of patients
- 60,000+ patients received heart transplant
  - Date of transplantation + survival
  - 53 features of patients/donors

Performance

Wait-List

	3 months	1 year	3 years	10 years
ToPs/R	0.8467	0.8130	0.7921	0.7897
MAGGIC	0.6298	0.6413	0.6425	0.6290

Post-Transplant

	3 months	1 year	3 years	10 years
ToPs/R	0.6763	0.6637	0.6538	0.6562
IMPACT	0.5808	0.5700	0.5524	0.5308

Survival/Mortality at 3 Months

	Actual Survival	Correctly Predicted (Specificity = .80)	Actual Mortality	Correctly Predicted (Sensitivity =.80)
MAGGIC	4,723	1,984 (37.8%)	2,542	915 (36%)
ToPs/R	4,723	3,212 (68.0%)	2,542	1,754 (69.0%)
Additional Correct Predictions		1,228		839

Newsweek

AFTER A HEART TRANSPLANT WITH SCARY ACCURACY

By Dana Dovey On Friday, May 18, 2018 - 12:18

The algorithm may help us make better use of limited available hearts.The first heart used in a heart donation.

IN THE JOURNALS

Algorithm predicts life expectancy in advanced HF

Yoon J, et al. *PLoS One*. 2018;doi:10.1371/journal.pone.0194985.

June 21, 2018

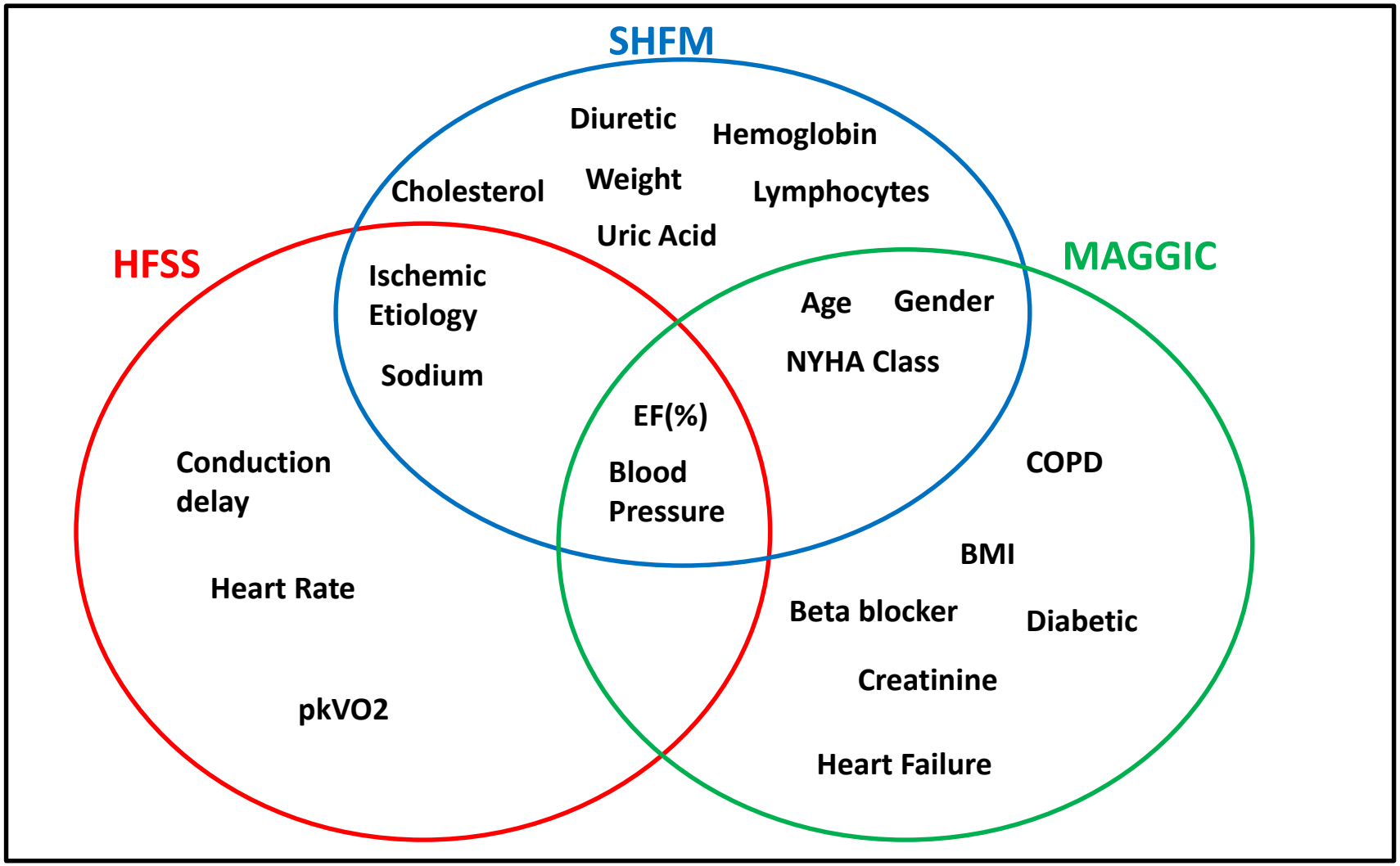
Researchers reported that they developed a new algorithm that more accurately predicts how long patients with advanced HF will survive, regardless of whether they receive a transplant.

"Our work suggests that more lives could be saved with the application of this new machine learning-based algorithm," **Mihaela van der Schaar, PhD,** Chancellor's

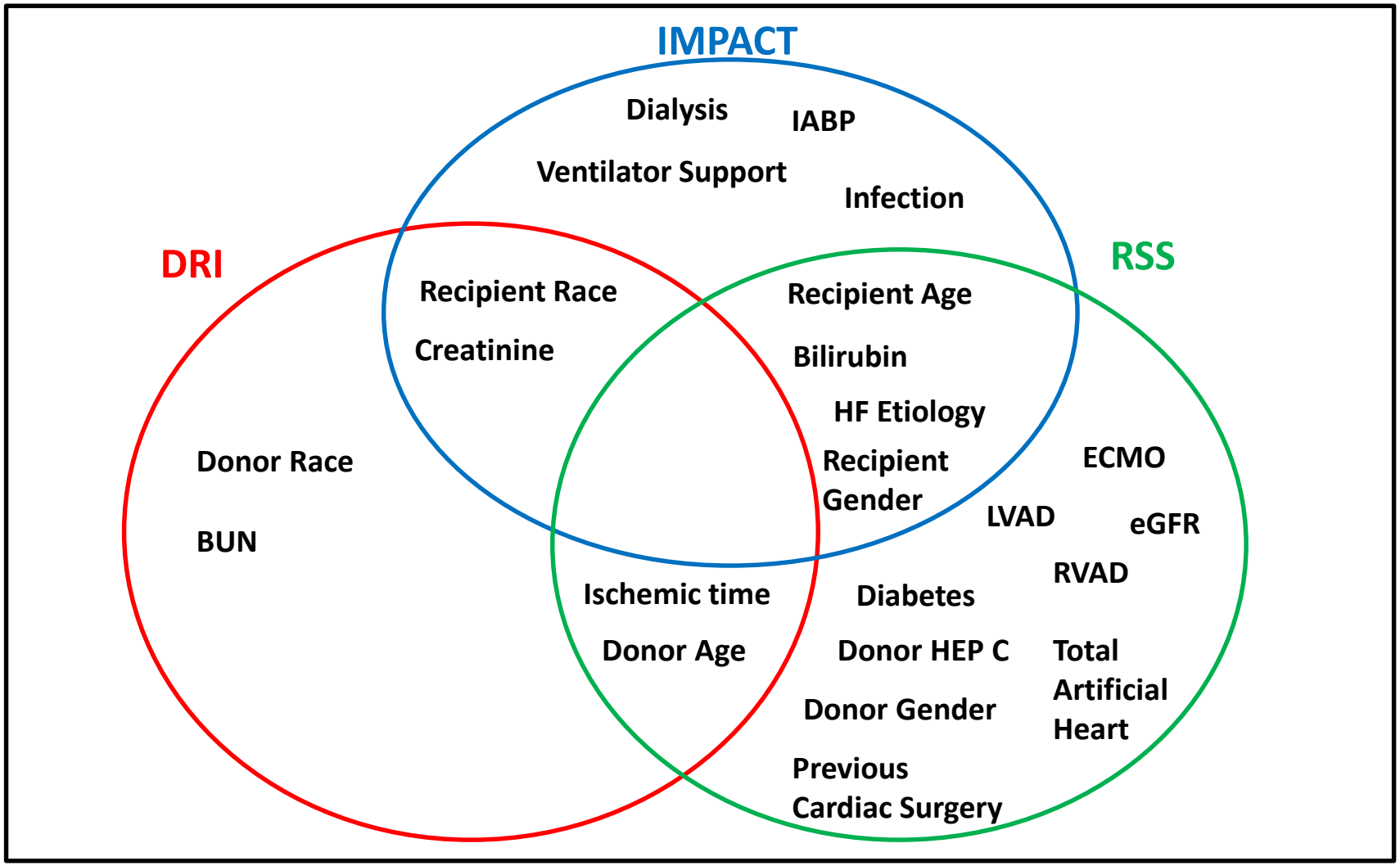
Sources of gain

- Information gain
- Modeling gain

Features used by Clinical Risk Scores (Wait-list)



Features used by Clinical Risk Score (Post-transplantation)



What are the Problems with Clinical Scores?

1. Models are one size-fits-all
   - but … population(s) are very heterogeneous
2. Models are linear
   - but … survival is non-linear: features interact
3. Models are horizon-independent
   - but … long-term survival is different from short-term survival; different features matter for different time horizons

Our Method ToPs – Designed to Solve Problems

1. **Model is individualized**
   - addresses heterogeneous population(s)
2. **Model is non-linear (where needed)**
   - addresses interaction of features
3. **Model is horizon-dependent**
   - addresses differences between long-term survival and short-term survival; different features matter for different time horizons

Interpretability? Tops/R (Regressions as Base Learners)

- Built on
  - Cox Regression
  - Linear Regression
  - Logistic Regression
- Choice of regression model represents interaction of features
- Choice of coefficients represents importance of features
- *Data tells us*
  - how to group/cluster patients
  - which regression model to use for each group/cluster
  - which coefficients to use for each group/cluster
  - how to aggregate predictions

Clinical Decision Support System

University of California Los Angeles

Input Variables

Recipient Variables

Clinical and demographic features

Age (years)	50
Gender	Male
Height (cm)	178
Weight (kg)	80
Diabetes	Yes
Infection	No
Transfusion	No
Number of Previous Heart Transplants	0
Creatinine (µmol/l)	1.2
Total Bilirubin (µmol/l)	1.3
Mean PRA (%)	5.1
Blood Type	A

Life support

Ventilator Assist	No
ECMO Assist	No
LVAD Assist	No
Dialysis	No
IABP Assist	Yes
Total Artificial Heart	No
Inotropic Assist	No
Other Circulatory Support	No

Waiting status and time

Days in Status 1A (days)	3
Days in Status 1B (days)	9
Days in Status 2 (days)	25

Donor Variables

Age (years)	28
Gender	Male
Height (cm)	181
Weight (kg)	83
Donor blood group	A
Hep C Antigen	Yes
Diabetes	Yes

Compatibility Variables

Ischemic Time (hour)	3
HLA-A Mismatch	1
HLA-B Mismatch	1
HLA-DR Mismatch	1

ML Performance Comparisons (Wait-list)

	Algorithms	3-month	1-year	3-year	10-year
	ToPs/R	**0.8467**	**0.8130**	**0.7921**	**0.7897**
Boosting Methods	AdaBoost	0.8180	0.7865	0.7773	0.7452
	Deep Boost	0.8211	0.7898	0.7731	0.7392
	Logit Boost	0.7449	0.7371	0.7232	0.6776
	XGBoost	0.8243	0.7935	0.7779	0.7456
Tree-based Methods	Decision Tree	0.8188	0.7833	0.7642	0.7440
	Random Forest	0.8239	0.7926	0.7744	0.7280
Other	Neural Nets	0.7881	0.7811	0.7705	0.7412

ML Performance Comparison (Post-transplantation)

	Algorithms	3-month	1-year	3-year	10-year
	ToPs/R	**0.6763**	**0.6637**	**0.6538**	**0.6562**
Boosting Methods	AdaBoost	0.6506	0.6302	0.6034	0.6155
	Deep Boost	0.6464	0.6347	0.6100	0.6133
	Logit Boost	0.6370	0.6216	0.5961	0.6130
	XGBoost	0.6183	0.6083	0.5877	0.6152
Tree-based Methods	Decision Tree	0.6296	0.6107	0.5895	0.5990
	Random Forest	0.6529	0.6413	0.6113	0.6194
Other	Neural Nets	0.6415	0.6387	0.6101	0.6150

Previous Machine Learning in Prognostic Research

+ High predictive accuracy (for some datasets)
+ Data-driven, few assumptions
- Many algorithms: Which one to choose?
- Many hyper-parameters: Need expertise in data science

AUROC	MAGGIC	UK Biobank	UNOS-I	UNOS-II
Best predictor	0.80 ± 0.004	0.76 ± 0.002	0.78 ± 0.002	0.65 ± 0.001
	NN	GradientBoost	ToPs	ToPs
Best Clinical Score	0.70 ± 0.007	0.70 ± 0.003	0.62 ± 0.001	0.56 ± 0.001
Cox PH	0.75 ± 0.005	0.74 ± 0.002	0.70 ± 0.001	0.59 ± 0.001

Previous Machine Learning in Prognostic Research

+ High predictive accuracy (for some datasets)
+ Data-driven, few assumptions
- Many algorithms: Which one to choose?
- Many hyper-parameters: Need expertise in data science

AUROC	MAGGIC	UK Biobank	UNOS-I	UNOS-II
Best predictor	0.80 ± 0.004	0.76 ± 0.002	0.78 ± 0.002	0.65 ± 0.001
	NN	GradientBoost	ToPs	ToPs
Best Clinical Score	0.70 ± 0.007	0.70 ± 0.003	0.62 ± 0.001	0.56 ± 0.001
Cox PH	0.75 ± 0.005	0.74 ± 0.002	0.70 ± 0.001	0.59 ± 0.001

- Can we predict in advance which method is best?
- Can we do better?
- Many metrics of performance (AUROC, AUPRC, C-index, quality of well-being)

How to do this?

Many diseases, many variables, various needs! All is changing!

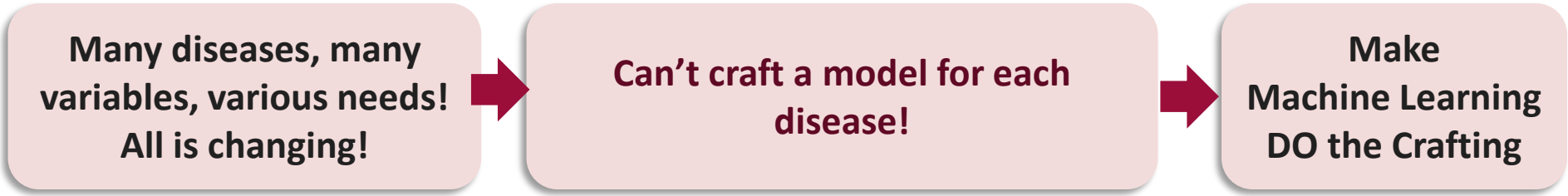
→

Can't craft a model for each disease!

→

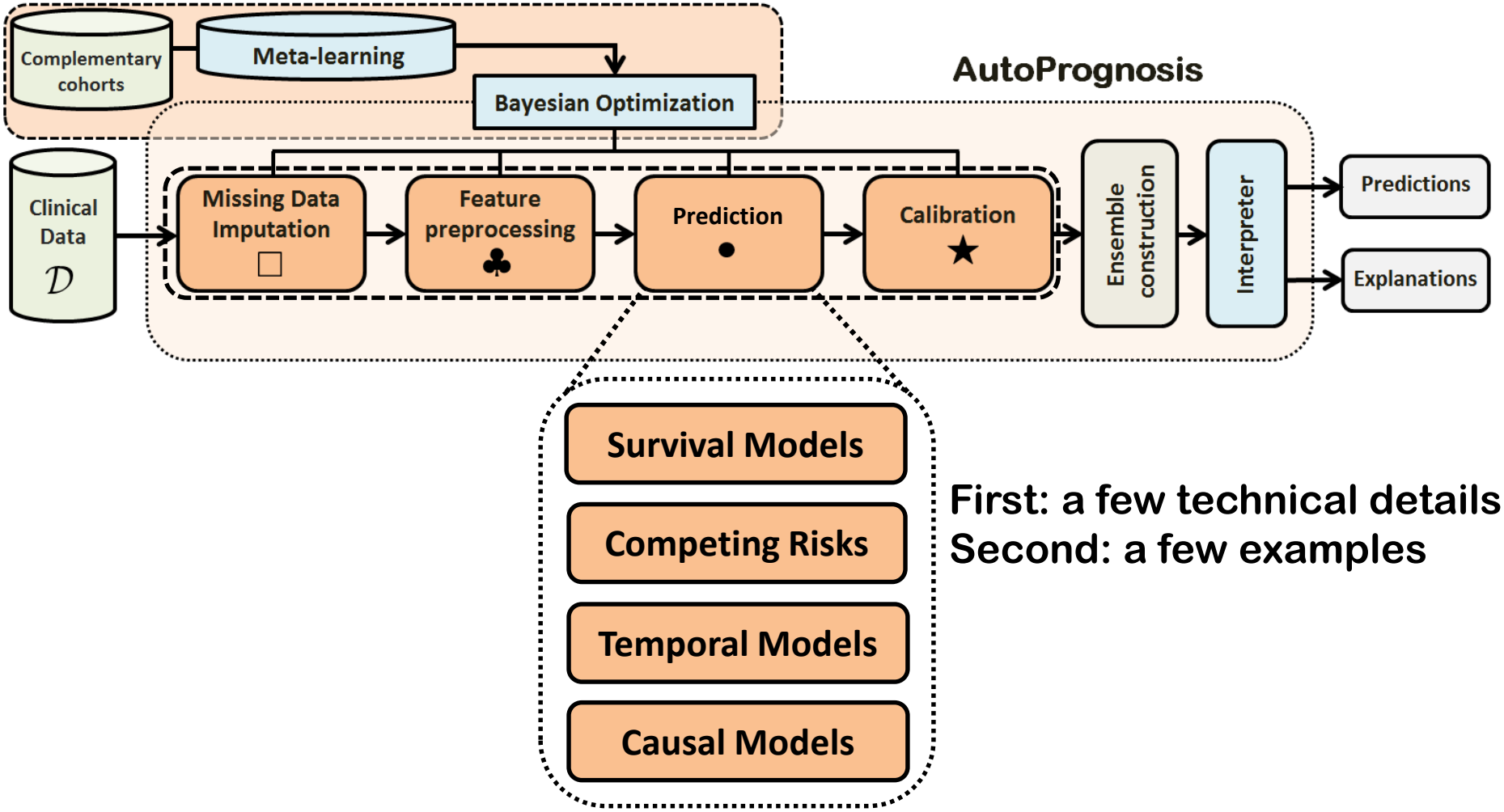
Make Machine Learning DO the Crafting

How to do this?



- **Previous AutoML? Auto-WEKA and Auto-Sklearn**
  - Limited performance gains
  - Ad-hoc optimization and ad-hoc meta-learning
  - Simplistic handling of missing data
  - Do not capture uncertainty
  - Limited to classification problems **(survival, competing risks etc.)**

AutoPrognosis: A tool for crafting Prognostic Scores for *Many Diseases*

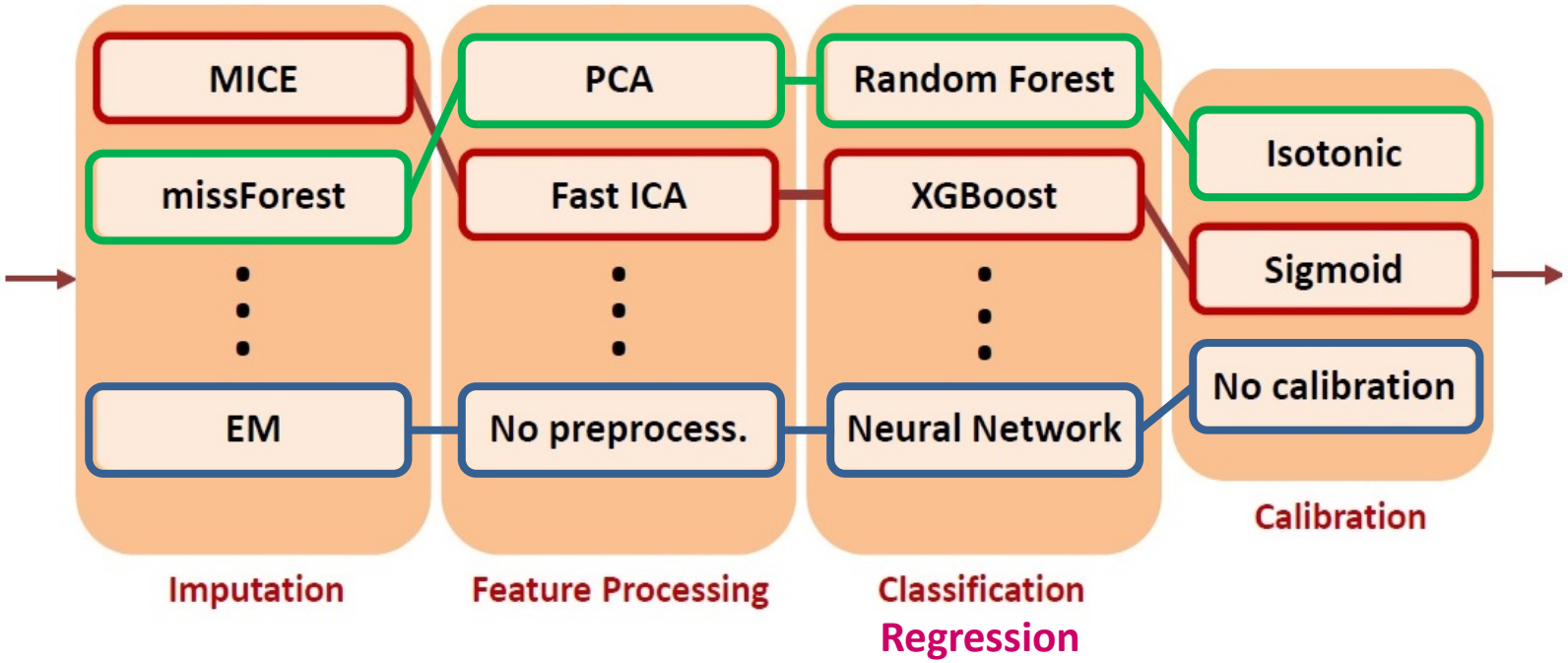


First: a few technical details
Second: a few examples

Principled Bayesian Optimization

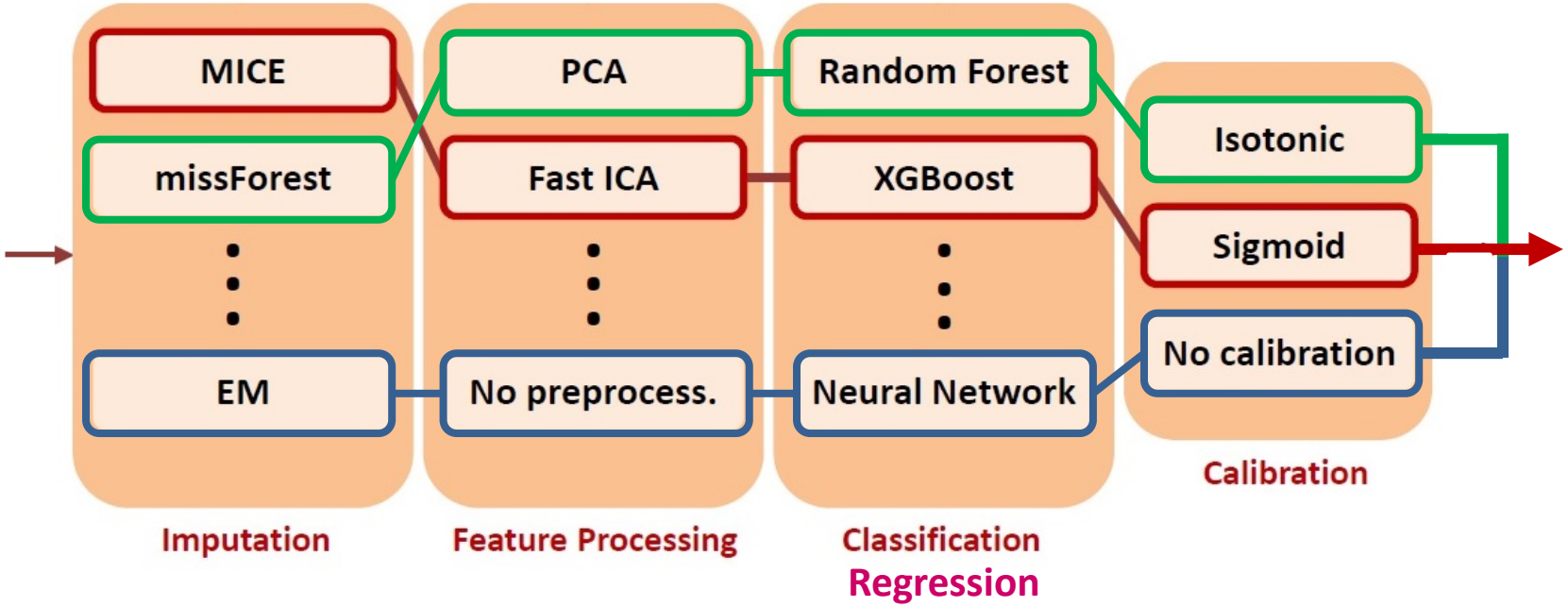
We need an entire pipeline!

- Each pipeline is a path of algorithms!
- Find the best paths and tune parameters:
  A hard optimization problem!



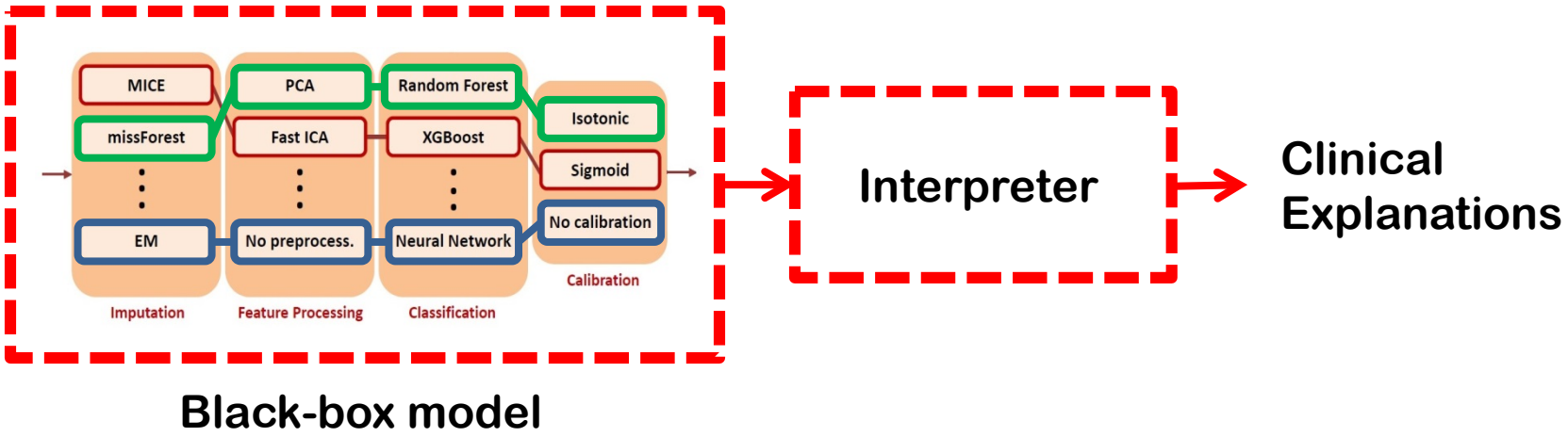
Ensembles

- **Instead of the single best pipeline we use an ensemble**
- **Why?**
  - ***Uncertainty*: finite data set to learn from, so we are not sure which pipeline is "best"**
  - ***Information loss:* using a single pipeline discards useful information from other pipelines**



Interpretability

- We don't want simply a **black-box**, we want explanations that users can **interpret**



- Interpreter provides **logical associations**

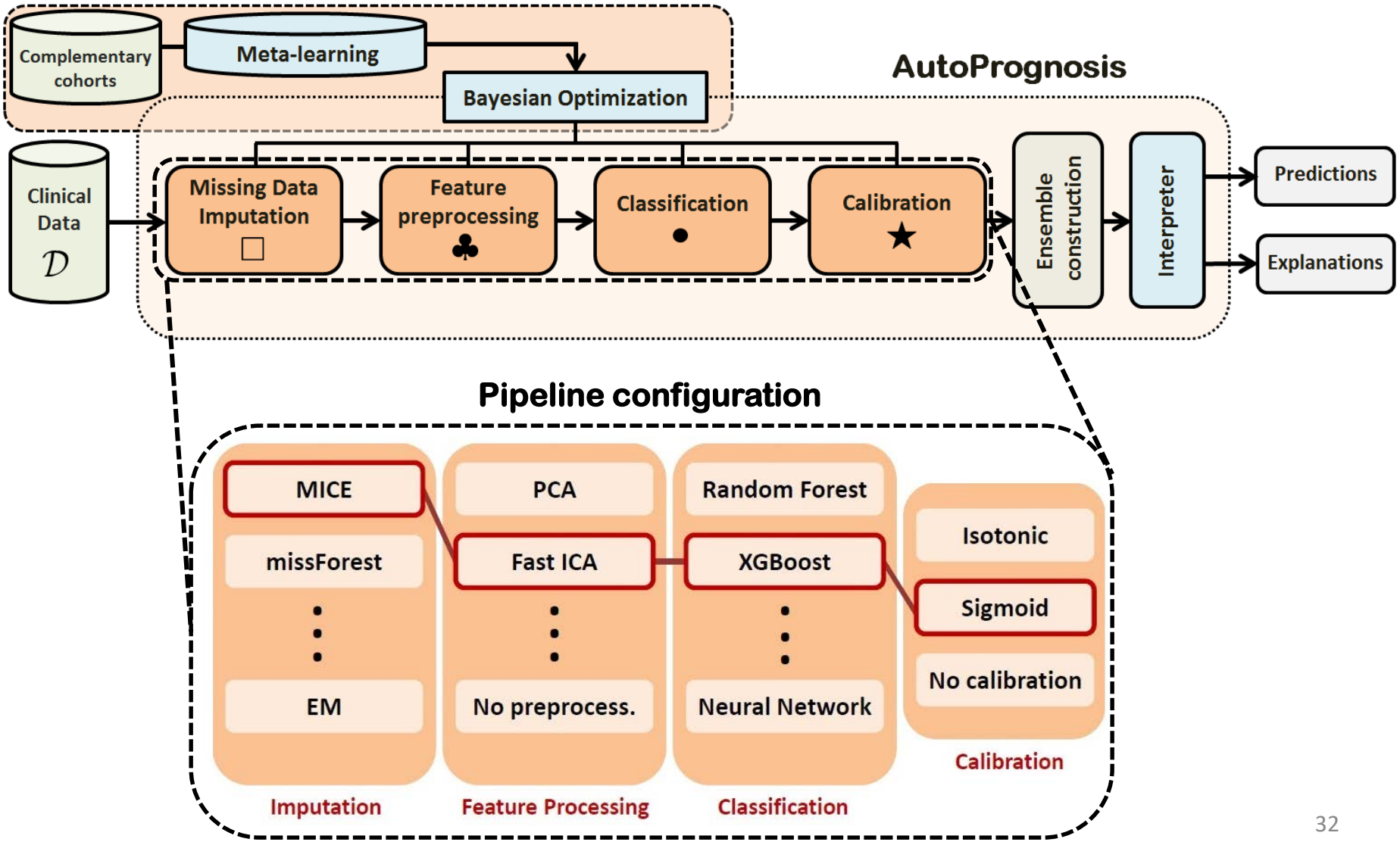
$$C_1 \wedge C_2 \wedge \ldots \wedge C_{l(r)} \implies r, \forall r \in \mathcal{R}$$

Clinical conditions — Risk stratum

$\mathcal{R}$: **risk strata**

- **Example:** Diabetic $\wedge$ Smoking $\implies$ High risk for ischemic stroke

AutoPrognosis: System Overview



AutoPrognosis: Pipeline Components

- 8 imputation algorithms, 10 feature preprocessing algorithms, 20 classifiers, 3 calibration methods
- MANY hyperparameters in each algorithm
- Total number of hyperparameters = 110

Pipeline Stage	Algorithms				
□ Data Imputation	□ missForest (2) □ Matrix completion (2)	□ Median (0) □ MICE (1)	□ Most-frequent (0) □ GAIN	□ Mean (0) □ None (0)	□ EM (1)
♣ Feature process.	♣ Feature agglo. (4) ♣ R. kitchen sinks (2)	♣ Kernel PCA (5) ♣ Nystroem (5)	♣ Polynomial (3) ♣ Linear SVM (3)	♣ Fast ICA (4) ♣ Select Rates (3)	♣ PCA (2) ♣ None (0)
• Prediction	• Bernoulli NB (2) • Gaussian NB (0) • Multinomial NB (2) • Ridge Class. (1) • DMGP	• AdaBoost (4) • XGBoost (5) • R. Forest (5) • Bagging (4) • CMGP	• Decision Tree (4) • Extr. R. Trees (5) • Neural Net. (5) • k-NN (1) • DeepHit	• Grad. Boost. (6) • Light GBM (5) • Log. Reg. (0) • Surv. Forest (5) • HBM	• LDA (4) • L. SVM (4) • GP (3) • Cox Reg. (0) • TOPs
★ Calibration	★ Sigmoid (0)	★ Isotonic (0)	★ None (0)		

Automated Pipeline Configuration (I)

- Imputation algorithms $\mathcal{A}_d$
  Hyperparameters Θ_d
- Feature process. algorithms $\mathcal{A}_f$
  Hyperparameters Θ_f
- Classification algorithms $\mathcal{A}_c$
  Hyperparameters Θ_c
- Calibration algorithms $\mathcal{A}_a$
  Hyperparameters Θ_a

- Set of all pipelines $\mathcal{P} = \mathcal{A}_d \times \mathcal{A}_f \times \mathcal{A}_c \times \mathcal{A}_a$
- Set of all hyperparameters $\Theta = \Theta_d \times \Theta_f \times \Theta_c \times \Theta_a$
- Set of all pipeline configurations $\mathcal{P}_\Theta$
- Combined Pipeline Selection and Hyperparameter optimization problem (CPSH)

$$P^*_{\theta^*} \in \arg\max_{P_\theta \in \mathcal{P}_\Theta} \frac{1}{K} \sum_{i=1}^{K} \mathcal{L}(P_\theta; \mathcal{D}^{(i)}_{\text{train}}, \mathcal{D}^{(i)}_{\text{valid}})$$

Automated Pipeline Configuration (II)

- **The CPSH problem** $\arg\max_{P_\theta \in \mathcal{P}_\Theta} \frac{1}{K} \sum_{i=1}^{K} \mathcal{L}(P_\theta; \mathcal{D}_{\text{train}}^{(i)}, \mathcal{D}_{\text{valid}}^{(i)})$
- **Bayesian optimization**

Gaussian process prior

$$f \sim \mathcal{GP}(\mu(P_\theta), k(P_\theta, P'_\theta))$$

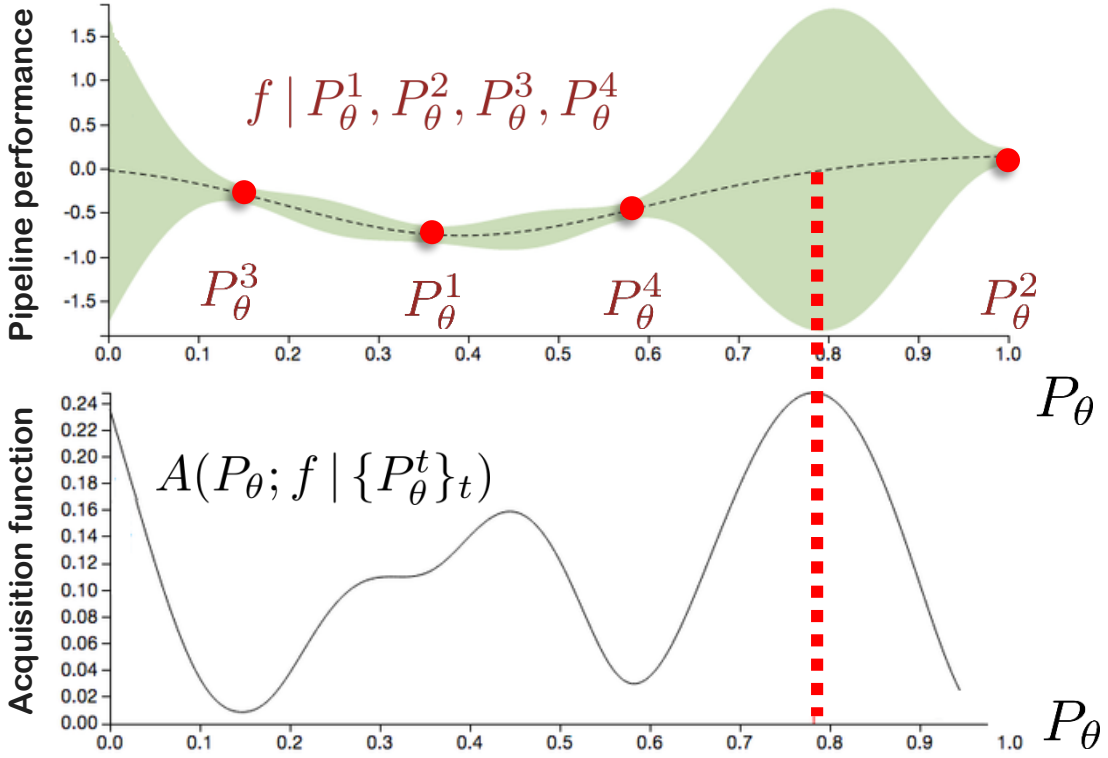
Gaussian process posterior

$$f \mid \{P_\theta^t\}_t$$

Select new pipeline via acquisition function

$$P_\theta^{t+1} = \arg\max_{P_\theta} A(P_\theta; f \mid \{P_\theta^t\}_t)$$

$$f(P_\theta) = \frac{1}{K} \sum_{i=1}^{K} \mathcal{L}(P_\theta; \mathcal{D}_{\text{train}}^{(i)}, \mathcal{D}_{\text{valid}}^{(i)}) + \varepsilon$$



The Curse of Dimensionality

- **Statistical** and **computational** **complexity** of the CPSH problem
- **GP BO does not work well for *D* > 10** [Wang, 2013]

Gaussian process prior

$$f \sim \mathcal{GP}(\mu(P_\theta), k(P_\theta, P'_\theta))$$

Sample complexity for non-parametric estimation of α-smooth functions [Stone, 1982]

$$\Theta\left(t^{-\frac{\alpha}{2\alpha+D}}\right)$$

Exponentially many iterations!

Gaussian process posterior

$$f \mid \{P_\theta^t\}_t$$

Computational complexity of GP posterior After ***t*** iterations [Rasmussen & Williams, 2006]

$$\mathcal{O}(t^3)$$

Select new pipeline via acquisition function

$$P_\theta^{t+1} = \arg\max_{P_\theta} A(P_\theta; f \mid \{P_\theta^t\}_t)$$

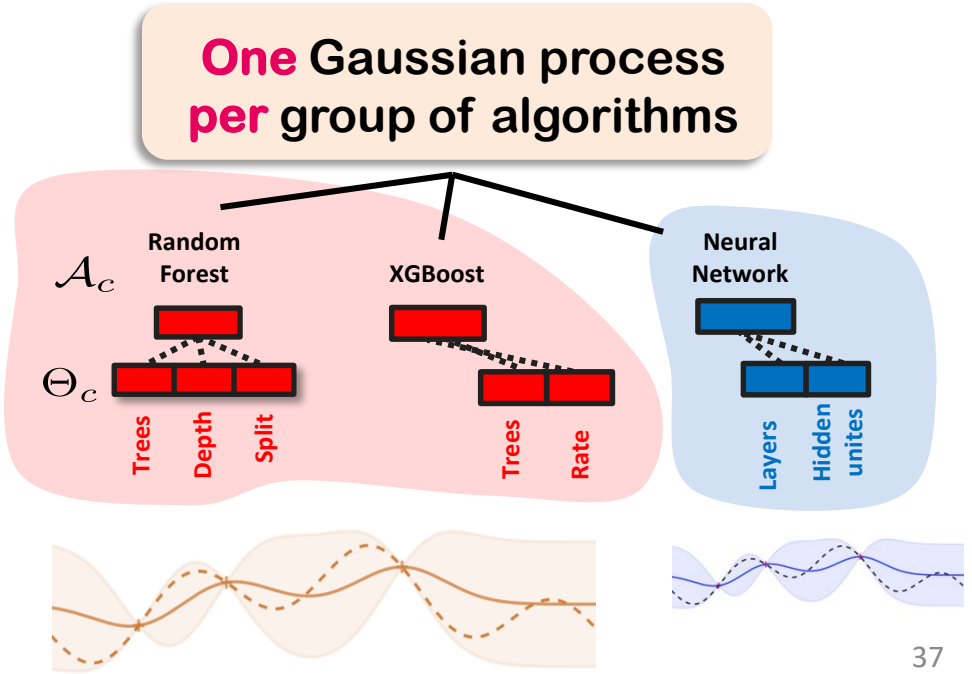
Computational complexity of maximizing acquisition [Snoek, 2015]

$$\mathcal{O}(n^D)$$

Bayesian Optimization with Structured Kernel Learning

- **Main idea: Some** algorithms are "correlated" and **some** are not => **Correlated** algorithms should be made to **share information**
- **Correlation is not known in advance, so must be learned**
- **Learn a structured kernel that clusters correlated algorithms:**

  - Low dimensionality for every cluster
  - Relevant information sharing within a cluster



Sparse Additive Gaussian Processes

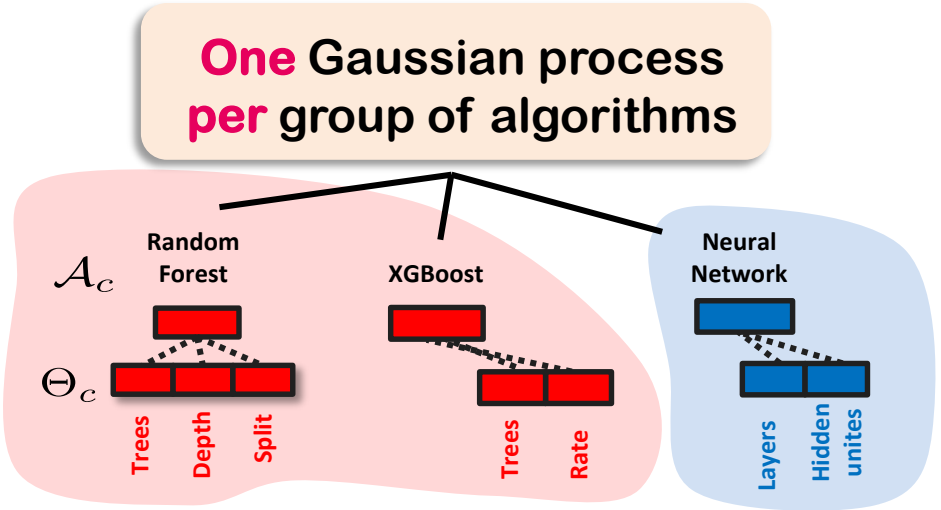
- **Decompose high-dimensional GP into sum of low-dimensional components**

Λ

Space of all pipelines

$\{\Lambda^{(m)}\}_m$

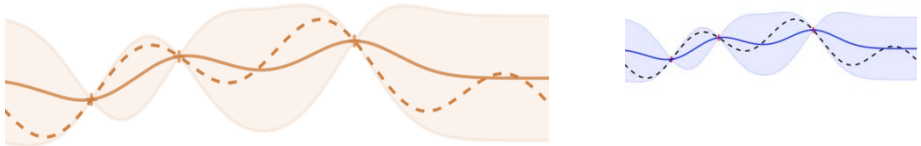
Partitions of Λ



Sparse additive GPs:

$$f(\Lambda) = \sum_{m=1}^{M} f_m(\Lambda^{(m)})$$

D-dimensional GP — **Low-dimensional GPs**



$k_1(\Lambda^{(1)}, \Lambda'^{(1)})$ $k_2(\Lambda^{(2)}, \Lambda'^{(2)})$

Structured kernel: $k(\Lambda, \Lambda') = \sum_{m=1}^{M} k_m(\Lambda^{(m)}, \Lambda'^{(m)})$

Structured Kernel Learning

- **Define the variable** $z_{v,i} \in \{1, \ldots, M\}$ **:** indicator for the subspace allocation for algorithm i in $\mathcal{A}_v$

Prior on $z_{v,i}$ **= Prior on** $\{\Lambda^{(m)}\}_m$

- **Bayesian inference:**

Prior on decompositions

$$\alpha \sim \text{Dirichlet}(M, \gamma)$$
$$z_{v,i} \sim \text{Multinomial}(\alpha)$$

Compute posterior in concurrence with BO

$$\mathbb{P}(z, \alpha \,|\, \{f(P_\theta^t)\}_t, \gamma) \propto \mathbb{P}(\{f(P_\theta^t)\}_t \,|\, z)\, \mathbb{P}(z \,|\alpha)\, \mathbb{P}(\alpha, \gamma)$$

Gibbs Sampling

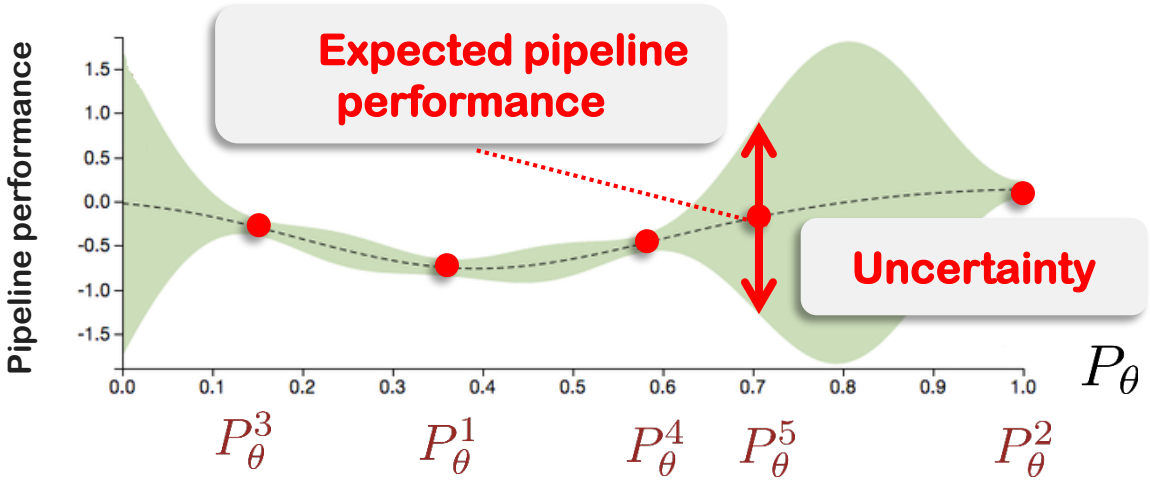
$$\mathbb{P}(z_{v,i} = m \,|\, z/\{z_{v,i}\}, \mathcal{H}_t) \propto \mathbb{P}(\mathcal{H}_t \,|\, z)\, (|\mathcal{A}_v^{(m)}| + \gamma_m)$$

Gumbel-Max Sampler

$$\omega_m \overset{\text{i.i.d}}{\sim} \text{Gumbel}(0, 1),\ m \in \{1, \ldots, M\},$$
$$z_{v,i} \sim \arg\max_m \mathbb{P}(\mathcal{H}_t \,|\, z, z_{v,i} = m)(|\mathcal{A}_v^{(m)}| + \gamma_m) + \omega_m.$$

Post-hoc Ensemble Construction

- **Create an ensemble using the posterior distribution of performances**

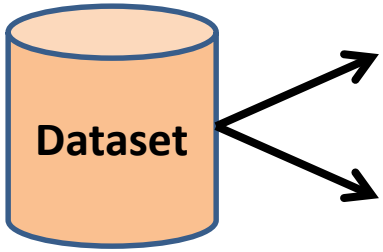


- **Bayesian model averaging**
- **Create a linear combination of pipelines** $\sum_i w_i P_\theta^i$
- **Weight of every pipeline = empirical probability of it being the best!**

$$
\begin{aligned}
w_i &= \mathbb{P}(P_\theta^{i^*} = P_\theta^i \mid \mathcal{H}_t) \\
&= \prod_{j \neq i} \Phi\left((\mu_i - \mu_j) \cdot (\sigma_i^2 + \sigma_j^2)^{-\frac{1}{2}}\right),
\end{aligned}
$$

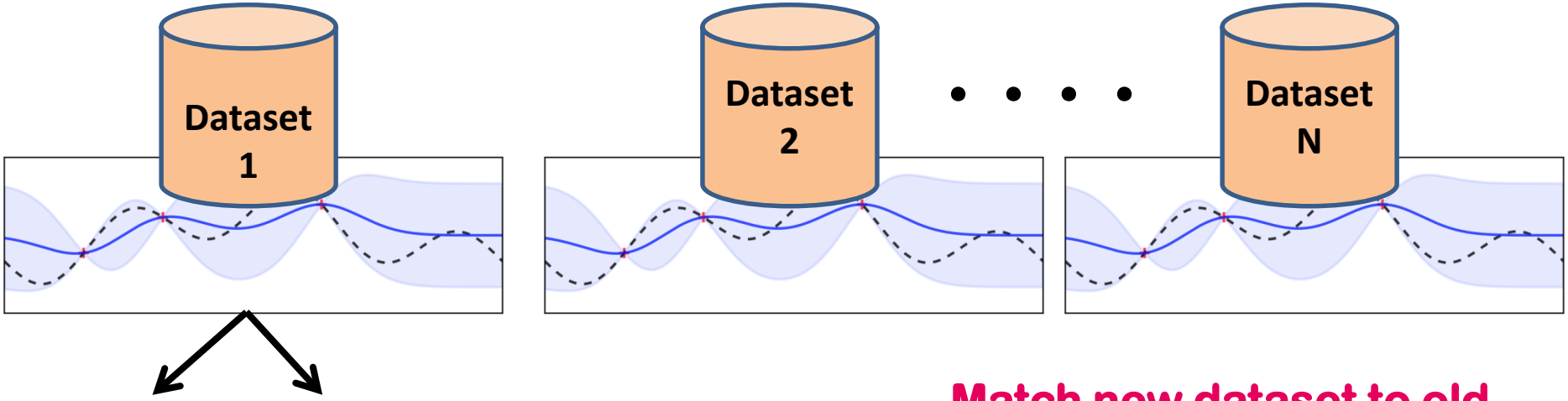
Meta-learning via Empirical Bayes

- For every dataset seen by the system, create meta-features



Statistical meta-features: entropy, size of dataset, number of features, class imbalance, etc.

Clinical meta-features: ICD-10 codes, lab tests, etc



Meta-features

Tuned hyperparameters via empirical Bayes

Match new dataset to old ones using meta-features…

Example 1: Cystic Fibrosis (Scientific Reports, 2018)

- **Collaboration with the UK CF trust**

  Using cross-sectional observational data for 99% of CF patients in the UK

- **Scarce resources: donated lungs, surgical resources**

- **Questions:**

Who should be referred to a lung transplant?

What guidelines should be used for referral to a lung transplant?

AutoPrognosis: Better Predictions (Out-of-sample)

Prognostic Model	AUC-PR
AutoPrognosis	**0.59 ± 0.03**
Clinical Practice	0.49 ± 0.02

AUC-PR (Sensitivity-Precision) is the metric of interest!

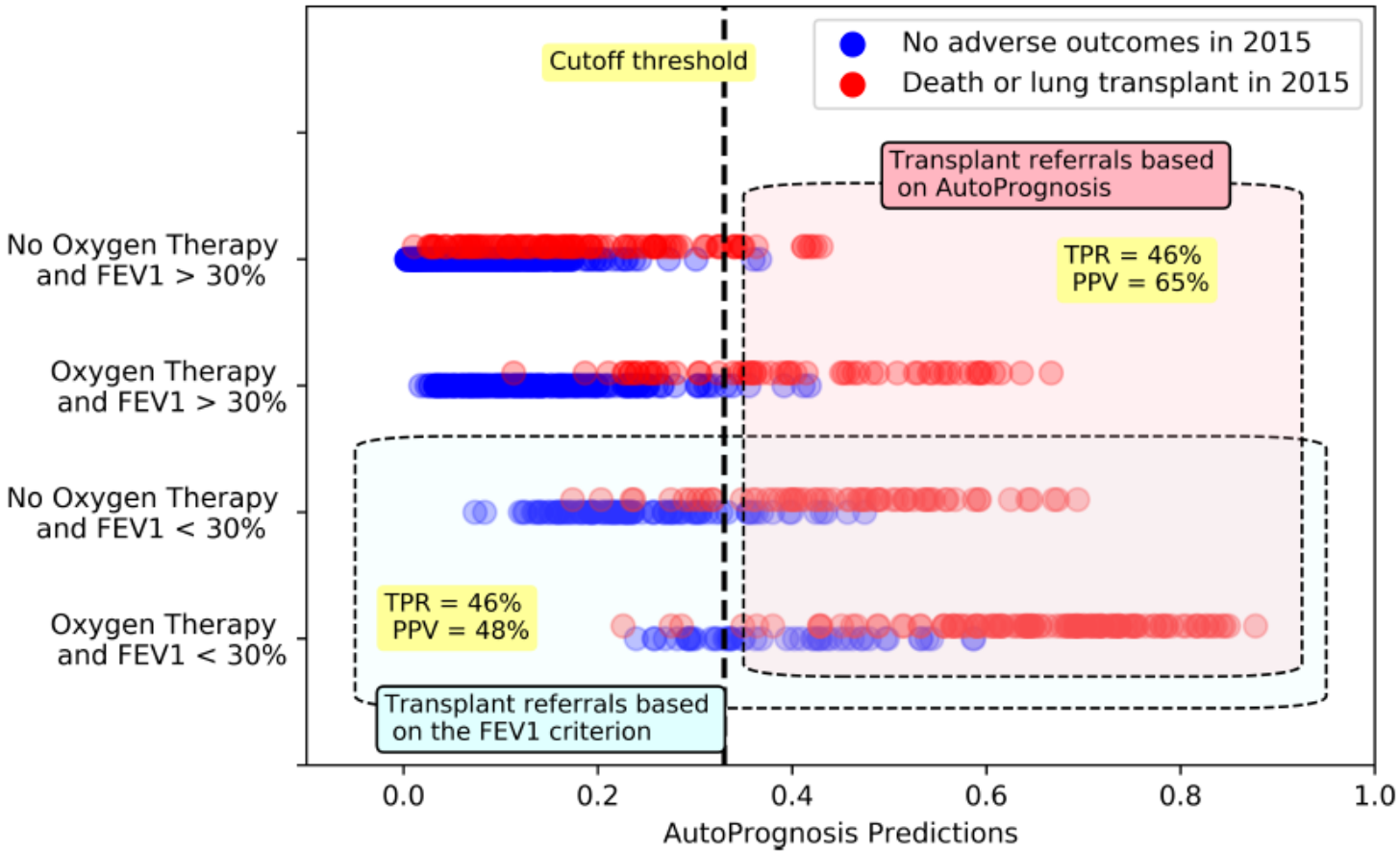
AutoPrognosis: Better Predictions (Out-of-sample)

Prognostic Model	AUC-PR
AutoPrognosis	**0.59 ± 0.03**
Clinical Practice	0.49 ± 0.02
Auto-WEKA	0.50 ± 0.03
Auto-sklearn	0.51 ± 0.02
Nkam et al., 2017	0.49 ± 0.02
CF-ABLE-UK	0.28 ± 0.04

AUC-PR (Sensitivity-Precision) is the metric of interest!

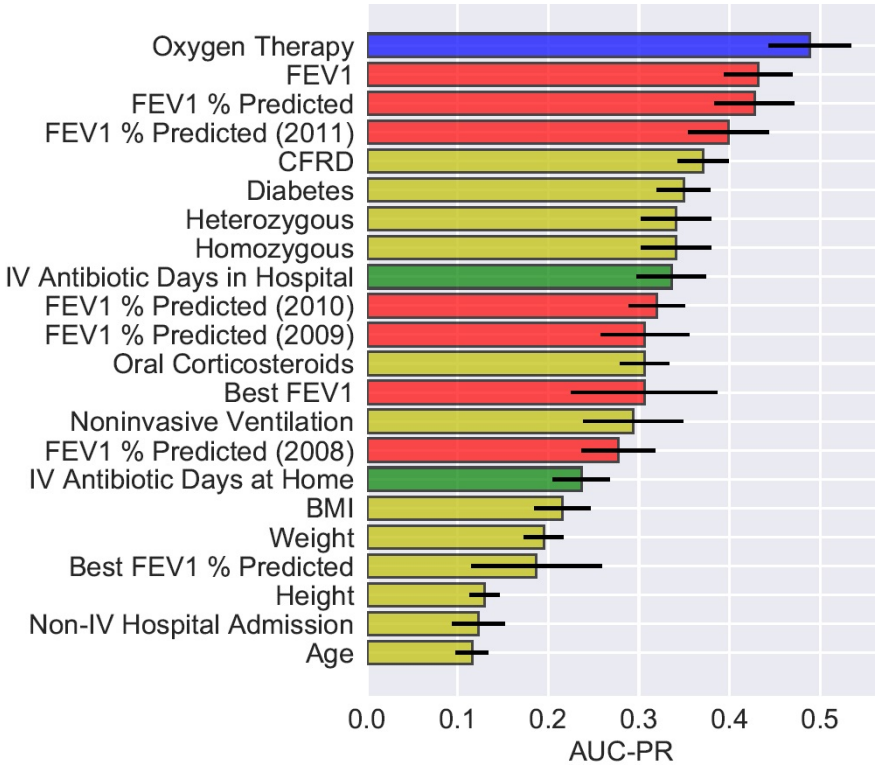
AutoPrognosis at work

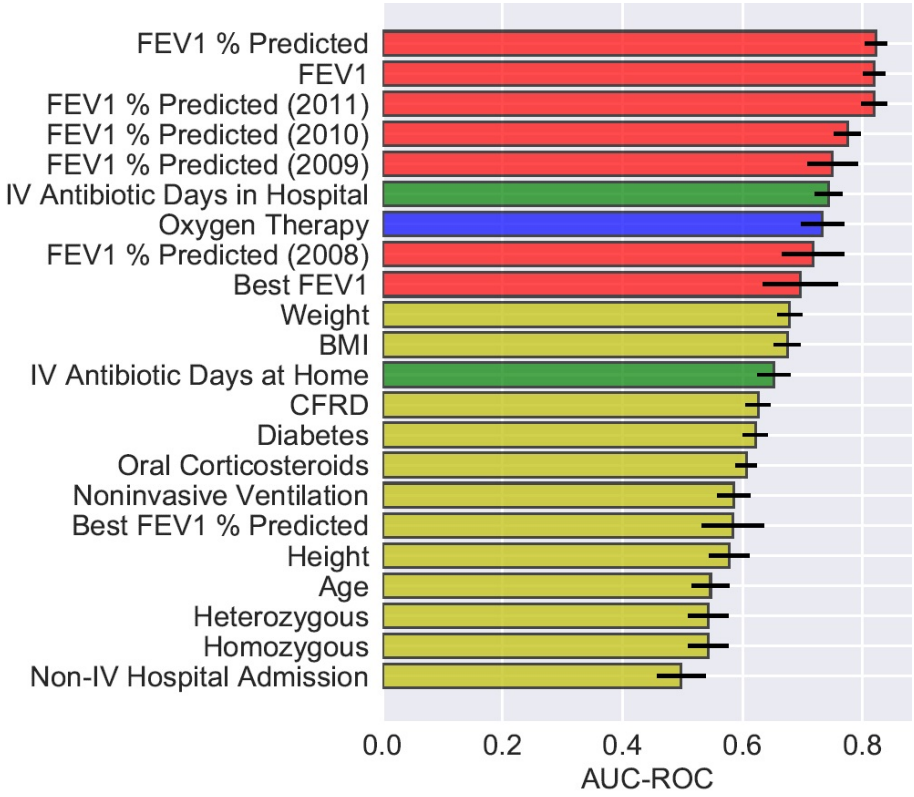
- AutoPrognosis learned to stratify the UK CF population in a way that led to a much more efficient allocation of donor lungs



AutoPrognosis: Identifying risk factors

- **Contribution of different features to predictive accuracy**





AutoPrognosis: Risk Stratification

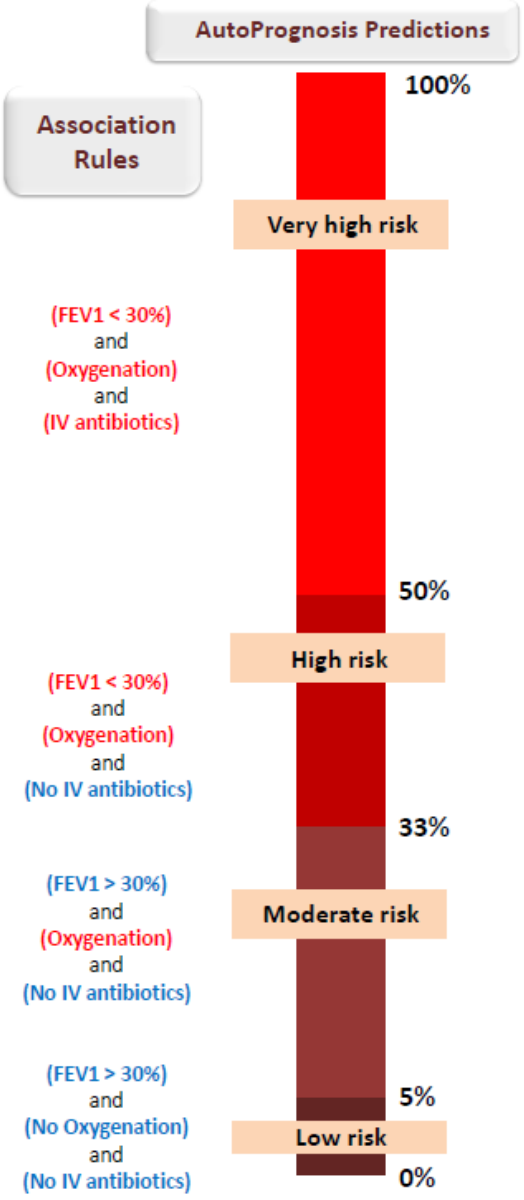
Lung function

Spirometry

Gas exchange

Current practice focuses on **spirometric** variables (FEV1) to make decisions

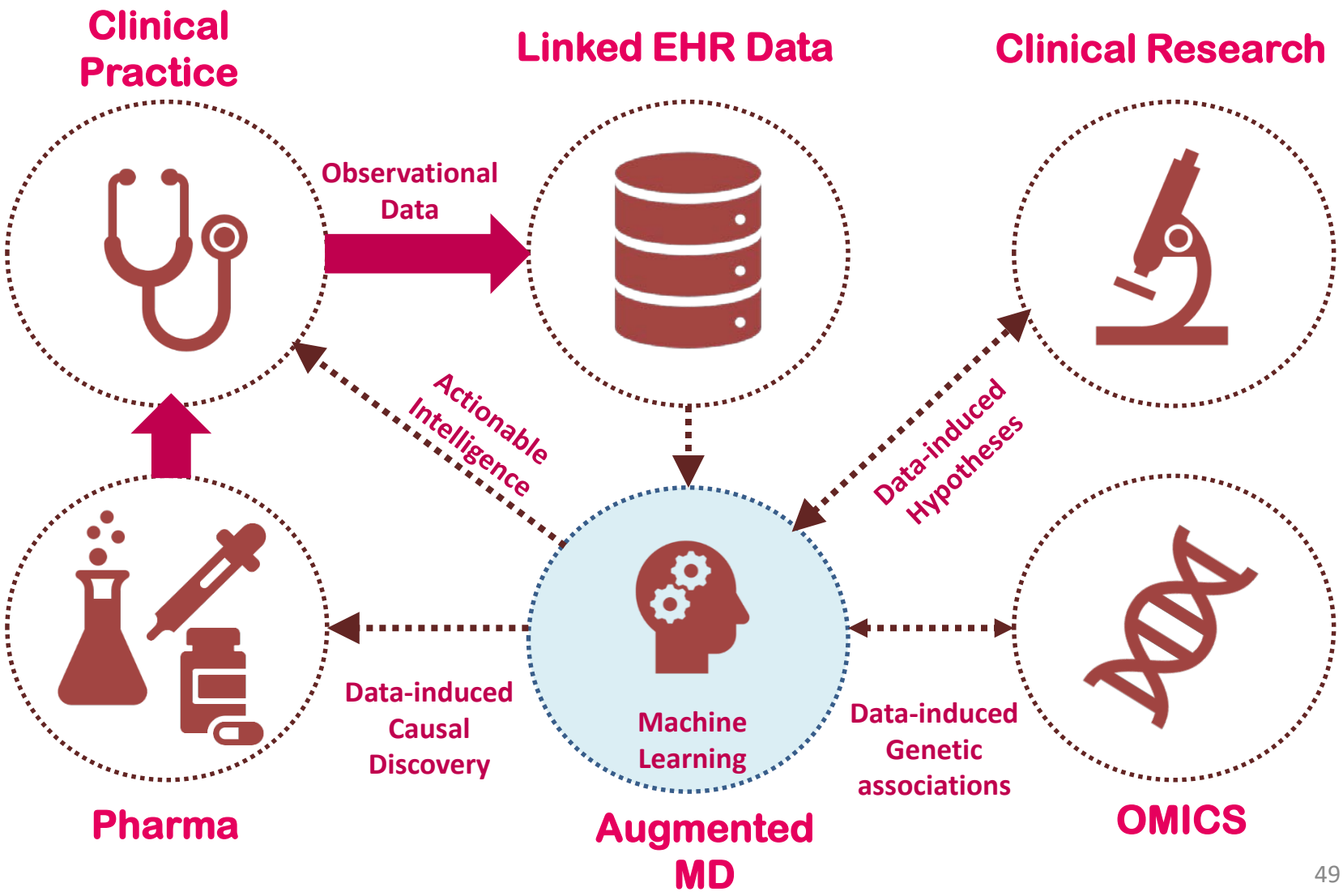
AutoPrognosis discovers that more refined decisions can be achieved by incorporating variables related to **gas exchange**



AutoPrognosis: Other examples

Cardiovascular Disease
Breast Cancer
Dementia

ML+AI:
Enormous potential for transformative impact in medicine



How fast does AutoPrognosis learn?

Theorem

Let the number of algorithms in every subspace be bounded by d. For a Matérn kernel with length-scale parameter ℓ, then the cumulative regret of AutoPrognosis is given by

$$R(T) = 2^d D T^{\frac{\ell + d(d+1)}{2\ell + d(d+1)}} \log(T)$$

- **Conventional GP-based BO:** $R(T) \sim T$
- **AutoPrognosis** $R(T) \sim T^{\frac{2}{3}}$ → **10-fold improvement For T = 1000!!**

(common scenario: $d = 5, \ell = 25$ **)**

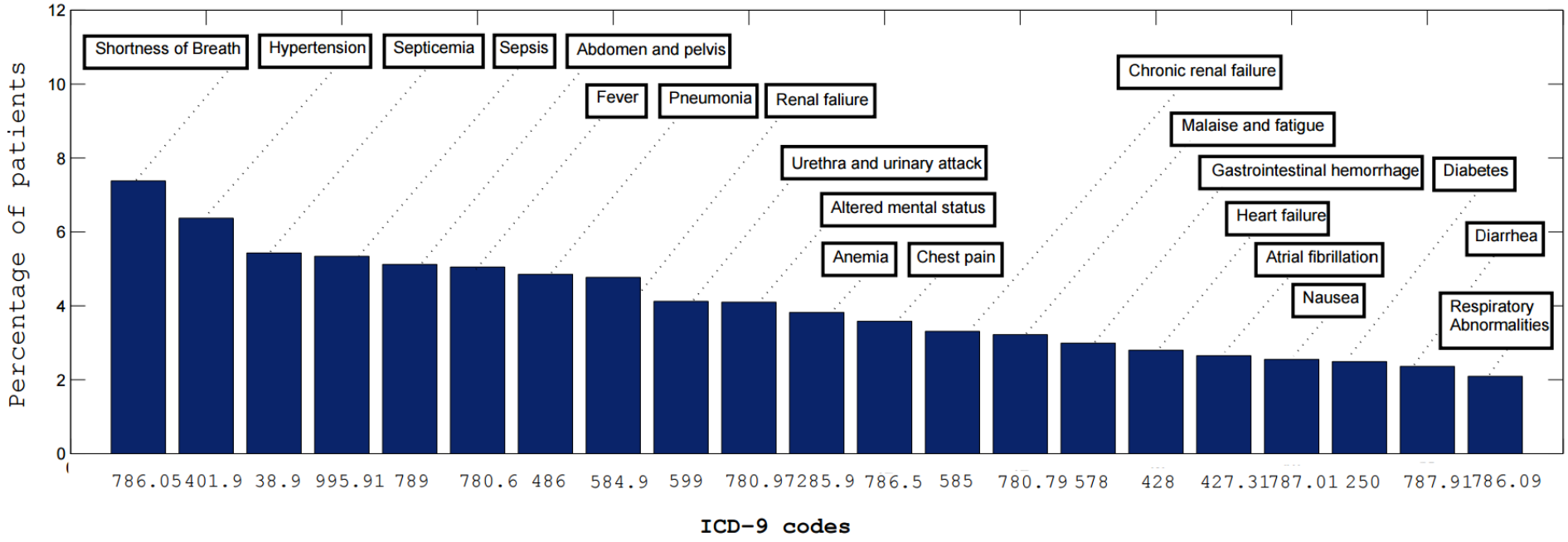
Forecast ICU in practice

- Hospital: UCLA Ronald Reagan Medical Center

- **Cohort of 6,094 patients**
- Period: March 2013 ~ June 2015 (tested July 2015 – July 2016)
- Age: 18 ~ 100+ years
- Gender:
  - Male (3,018 patients, 49.5%)
  - Female (3,076 patients, 50.5%)
- Length of stay: 1.5 hours ~ 159 days

Wide Variety of Diagnoses

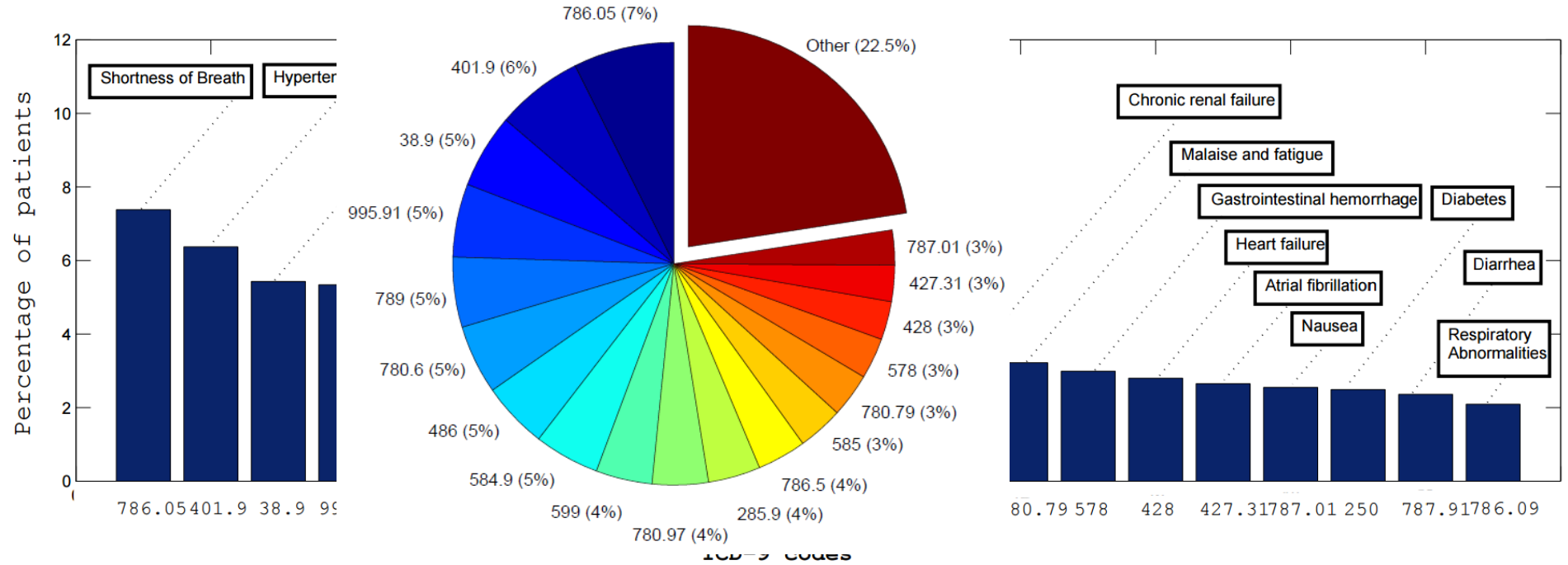
Percentage of patients in top 20 ICD 9 codes



Among 6,094 patients, 306 patients (5.0%) admitted to ICU unexpectedly; 5,788 patients (95.0%) discharged

Wide Variety of Diagnoses

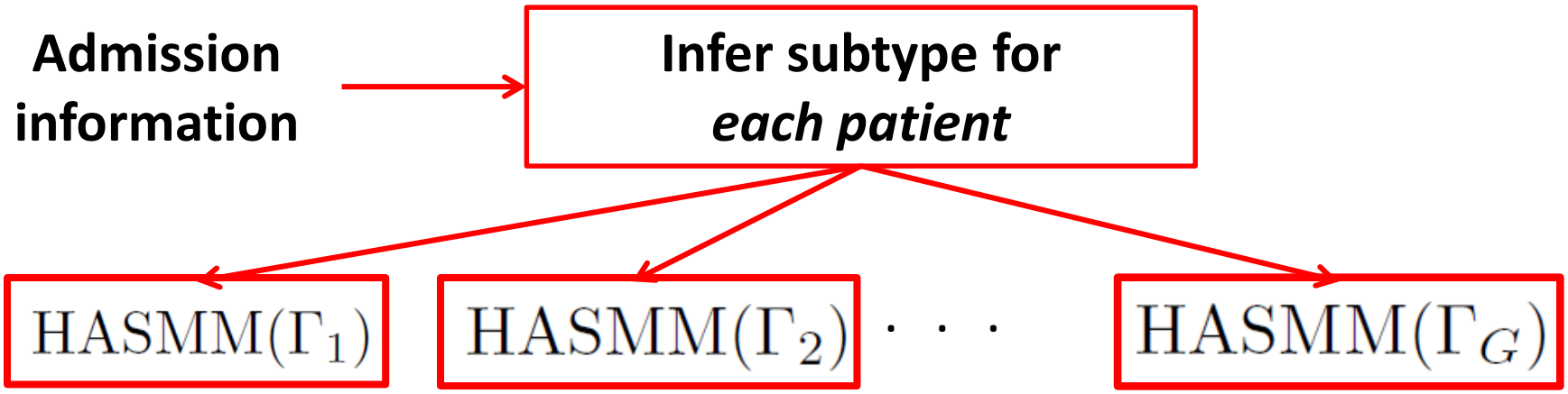
Percentage of patients in top 20 ICD 9 codes



Among 6,094 patients, 306 patients (5.0%) admitted to ICU unexpectedly; 5,788 patients (95.0%) discharged

Subtyping (Phenotyping)

- Discovering the different ways in which a disease manifests in different patients
- Key approach for **personalized medicine**



ICML-W 2016

Performance Metrics

- **TPR** (True Positive Rate, i.e. **Sensitivity**) = True Positive/True ICU Patients
- **TNR** (True Negative Rate, i.e. **Specificity**) = True Negative/True Discharge patients
- **PPV** (Positive Predictive Value, i.e. **Precision**) = True Positive/Predicted ICU Patients
- **NPV** (Negative Predictive Value) = True Negative/Predicted Discharge patients

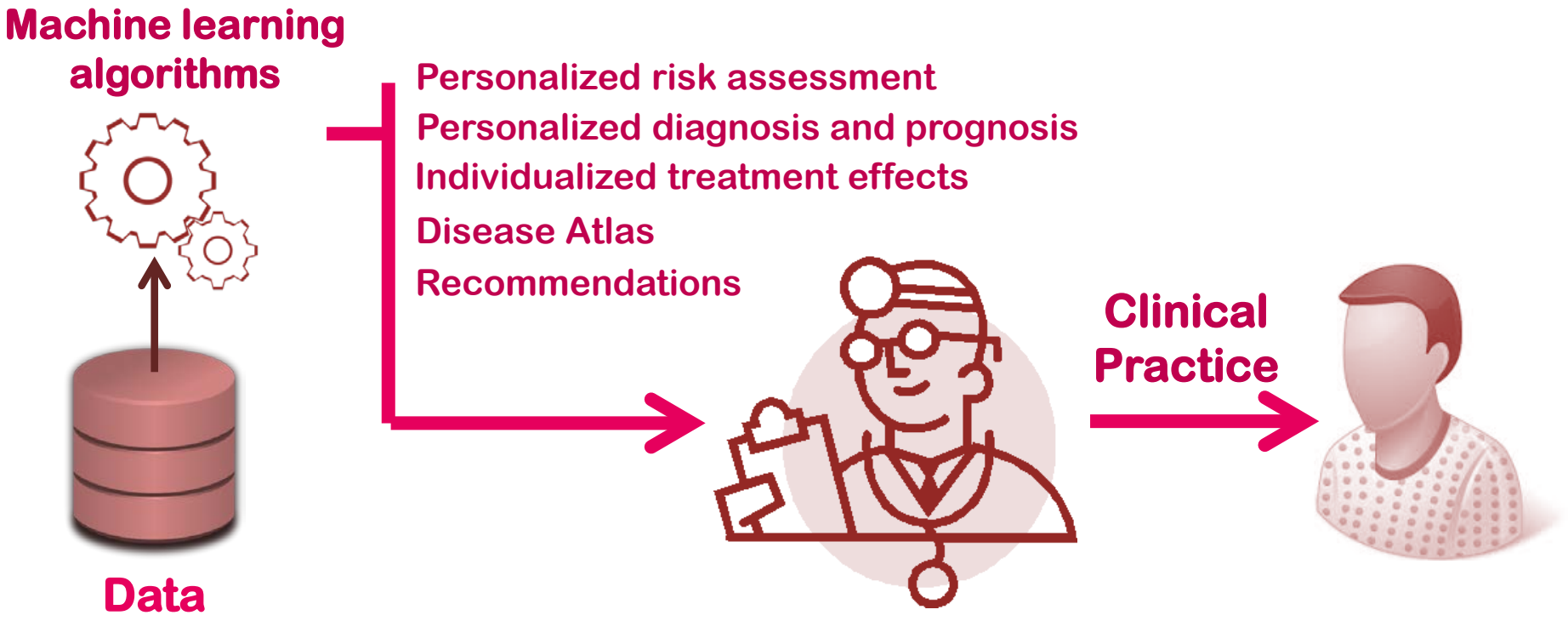
	Predicted ICU patients	**Predicted Discharge patients**
True ICU patients	True Positive	False Negative
True Discharge patients	False Positive	True Negative

The "Augmented" MD

- **Machine learning**

  …*can't* do medicine!

  …*can* provide doctors with actionable information!



Application to Cardiovascular Patient Care

- **Preventive care:**
  - **Meta-analysis Global Group in Chronic Heart Failure (MAGGIC)**
  - **UK bio-bank.**
- **Heart-transplant wait-list management:**
  - **United Network for Organ Sharing**
- **Post-transplant care:**
  - **United Network for Organ Sharing**