Patient Similarity with Multiple Kernel Learning

Conroy et. al, MLHC 2017

MEDG Reading Group

Rahul G. Krishnan

[Images taken from paper and from lecture notes on Kernel methods]

Motivation

- Why do we care about patient similarity?
  - A doctor within a clinical setting might be interested in asking “who else is in the hospital is similar to the patient that I am seeing next”
  - Matching for causal inference
  - Euclidian distance in feature space is insufficient
- What are good criteria for similarity?
  - Similarity should depend on clinical context (not just age and gender alone)
  - Should be modulated by frequency and specificity of individual feature values
- **Key Idea:**
  - Two patients with heart rates in the normal range of $70 - 75$ should receive a lower similarity score than two patients with elevated heart rates in the range $120 - 125$.
  - Interesting and most relevant aspects of patient state typically lie in the abnormal (tails of the distribution)

Approach

- Propose the use of population level features within a kernel
- Multiple kernel learning (MKL) framework (Gonen and Alpaydin (2011))
- Goal:
  - Learn an ensemble kernel over the individual population feature kernels described above that is capable of predicting one or more clinical contextual targets of interest.
  - Ensemble kernel is comprised of many base kernels, each of which is tuned to emphasize distribution tails
  - Ensemble weights assigned to the base kernels are determined by how discriminative each is in predicting a clinical context.

Why Kernels - ML101

$$\hat{y} = \operatorname{sgn} \sum_{i=1}^{n} w_i y_i k(\mathbf{x}_i, \mathbf{x}'),$$

where

- $\hat{y} \in \{-1, +1\}$ is the kernelized binary classifier's predicted label for the unlabeled input $\mathbf{x}'$ whose hidden true label y is of interest;
- $k: \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is the kernel function that measures similarity between any pair of inputs $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$;
- the sum ranges over the n labeled examples $\{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$ in the classifier's training set, with $y_i \in \{-1, +1\}$;
- the $w_i \in \mathbb{R}$ are the weights for the training examples, as determined by the learning algorithm;
- the sign function sgn determines whether the predicted classification $\hat{y}$ comes out positive or negative.

SVM Optimization

- Only depends on the dot product
- Replace the dot product with a nonlinear function of the inputs we can do classification in a projection of the input space

$$\max_{\alpha} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,k=1}^{m} \alpha_i \alpha_j y_i y_k \, \mathbf{x}_i^T \mathbf{x}_k \leftarrow \text{inner product}$$

$$\text{s.t. } 0 \leq \alpha_i \leq C, i = 1, ..., m \text{ and } \sum_{i=1}^{m} \alpha_i y_i = 0$$

Mapping to a different dimension

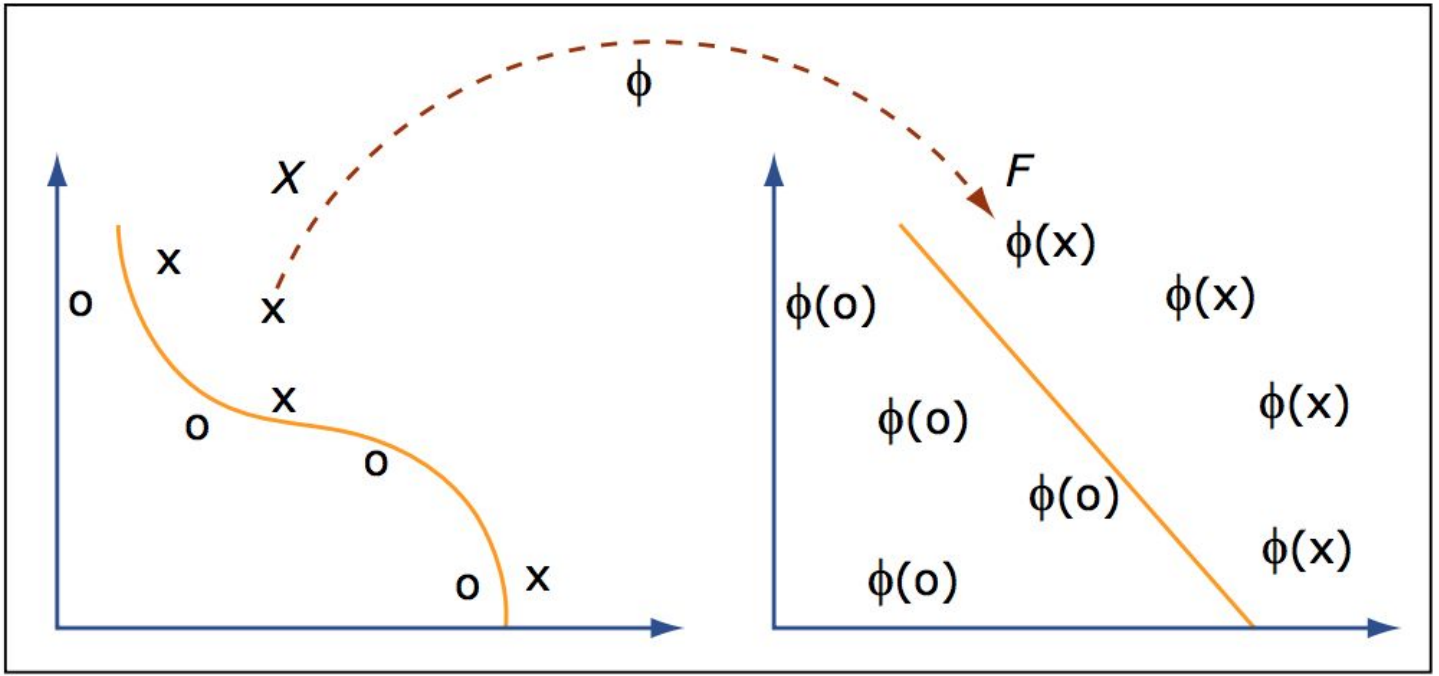


Image by MIT OpenCourseWare.

Boundaries in a feature space

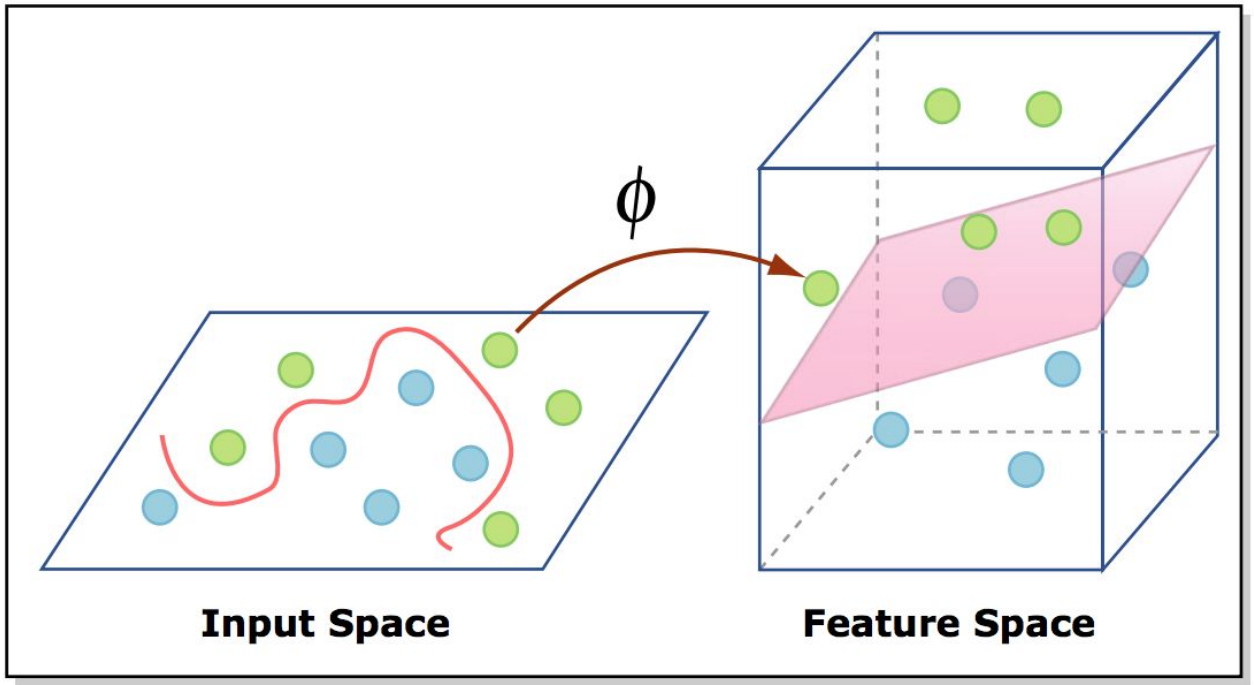


Image by MIT OpenCourseWare.

Patient Features

- Consider X_1,...,X_p as patient features
- For a single feature, let x_j, and z_j be the corresponding feature value between two patients

$$k_{j,c}(x, z) = (1 - P(\min(x_j, z_j) \leq X_j \leq \max(x_j, z_j)))^c$$

- c controls speed of decay
- Expected number of patients that lie between the values taken by x_j and z_j
- P(X_j) is the population distribution for feature j

Kernel for continuous and binary random variables

$$\text{Continuous: } k_{j,c}(x,z) = (1 - |F_j(z_j) - F_j(x_j)|)^c$$

$$\text{Binary: } \quad k_{j,c}(x,z) = \begin{cases} (1 - P(X_j = 1))^c & , x_j = z_j = 1 \\ (1 - P(X_j = 0))^c & , x_j = z_j = 0 \\ 0 & , x_j \neq z_j \end{cases}$$

Visualizing the Kernel for Three Features [ICU popn]

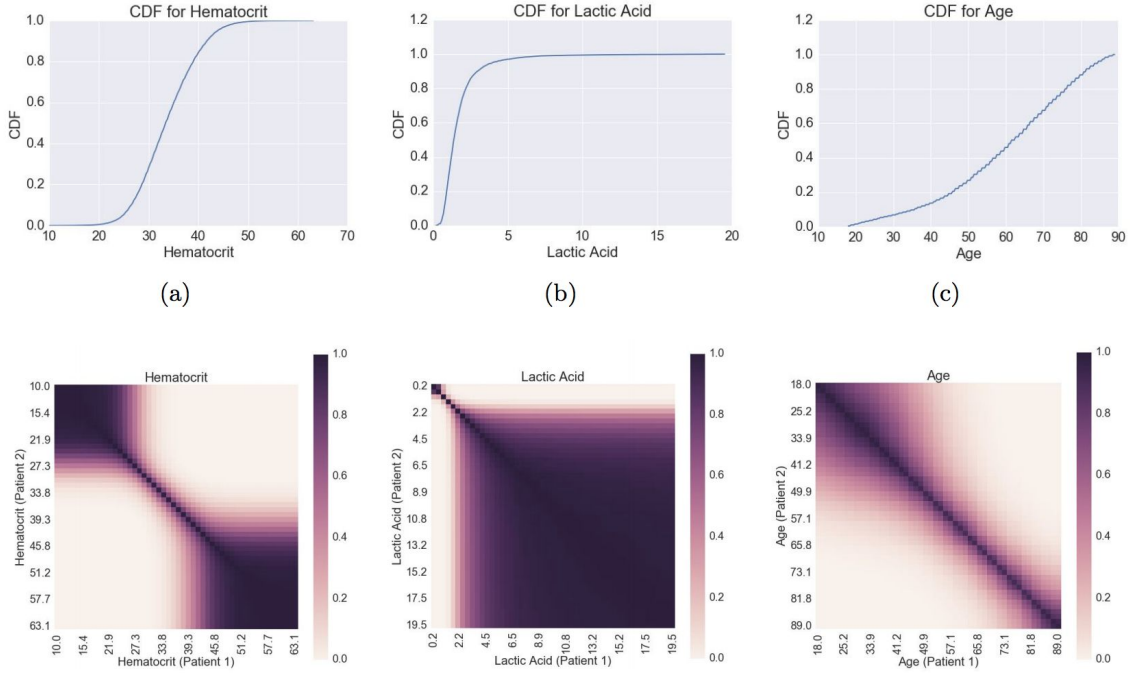


Figure 1: Examples of the kernel $k_{j,c}(x, z)$ in (1) with $c = 5$ on three features evaluated on adult ICU population: Hematocrit, Lactic Acid, and Patient Age

Reparameterizing the kernel

- This is a intuitive kernel but before we go forward, lets formulate it differently

For each kernel $k_{j,c}$, define a $2D$ transformation $x \to (F_j(x), R_j(x))$ defined by:

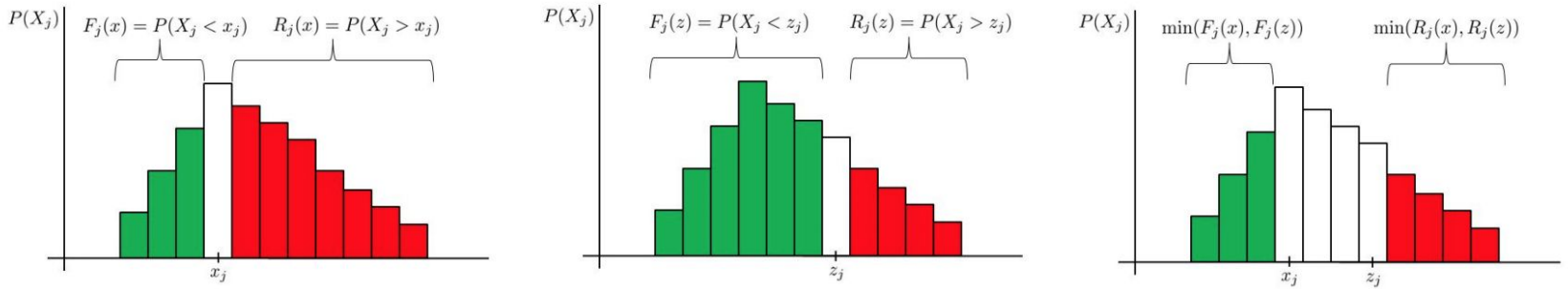
$$F_j(x) = P(X_j < x_j) \quad , \quad R_j(x) = P(X_j > x_j) \tag{2}$$

Sum of Intersection Kernels

Given this transformation, the kernel in (1) for $c = 1$ can be equivalently expressed as:

$$k_{j,1}(x, z) = \min(F_j(x), F_j(z)) + \min(R_j(x), R_j(z)) \tag{3}$$

Thus, $k_{j,1}(x, z)$ is a sum of two intersection kernels applied in a two-dimensional space $x \to (F_j(x), R_j(x))$. The equivalence is shown visually in Figure 2.



(a) $x \to (F_j(x), R_j(x))$ (b) $z \to (F_j(z), R_j(z))$ (c) $k_{j,1}(x, z)$

Figure 2: Expressing $k_{j,1}(x, z)$ on X_j as a sum of intersection kernels in a transformed space.

Adding c back in

- Uses Binomial Expansion for (a+b)^c

For $c \geq 1$, we can use the fact that $k_{j,c}(x,z) = k_{j,1}(x,z)^c$ and apply the binomial expansion on (3) to obtain:

$$k_{j,c}(x,z) = \sum_{i=0}^{c} \binom{c}{i} \min(F_j(x), F_j(z))^i \min(R_j(x), R_j(z))^{(c-i)} \tag{4}$$

$$\tilde{\Psi}_i(x) \text{ for kernels } \min(x,z)^i,\ i = 0, 1, \ldots, c,$$

$$\Psi_{j,c}(x) = \bigoplus_{i=0}^{c} \binom{c}{i} \left[\tilde{\Psi}_i(F_j(x)) \otimes \tilde{\Psi}_{c-i}(R_j(x))\right] \tag{5}$$

where $\oplus$ is the direct sum of feature spaces and $\otimes$ is the Kronecker product.

Population Based Representation

- Dimensionality of the explicit feature map may exceed the number of distinct values -- unroll categorical features
- At a high level, what we've achieved thus far is to take a patient's representation and map it to a feature representation
- For each feature and pair of patients, we've come up with a kernel function to tell us how similar they are
- If we wanted to know how similar patients are, as is, we can just evaluate the kernel pairwise

Supervision

- Often we wish to find similar patients towards a certain task
- That “task” may be represented as labels
- How can we make use of these labels?
- Multiple kernel learning framework [Gonen et. al]
  - Compute the explicit kernel representation for each patient or kernel trick
  - Train a linear function of the form: $$E(y|x) = g^{-1}\left(\sum_{j=1}^{p} f_j(x)\right) \quad (6)$$
  - This is nonlinear in X (the original feature space) -- the nonlinearity is an explicit function of population parameters
- Missing Features: Set the kernel to 0 if missing, each feature’s missingness label is also incorporated (often informative)
- Supervised learning yields weights corresponding to each feature

Ensemble Kernels

- Supervised learning gives us a set of weights w_1….w_p that represent how predictive the transformed feature is
- Task specific-ensemble kernel:
  - Previously, we had a kernel for *every* feature
  - Weigh those *feature* kernels by the predictive weights under the GLM to get the ensemble kernel

$$k(x,z) = \sum_j \alpha_j k_{j,c}(x,z) \qquad \alpha_j = ||w_j||^2 / \sum_j ||w_j||^2$$

- Patient specific-ensemble kernel: $k(x,z) = \sum_j \alpha_j(x) k_{j,c}(x,z)$
  - No longer symmetric

$$\alpha_j(x) = |f_j(x)|$$

Putting it all together

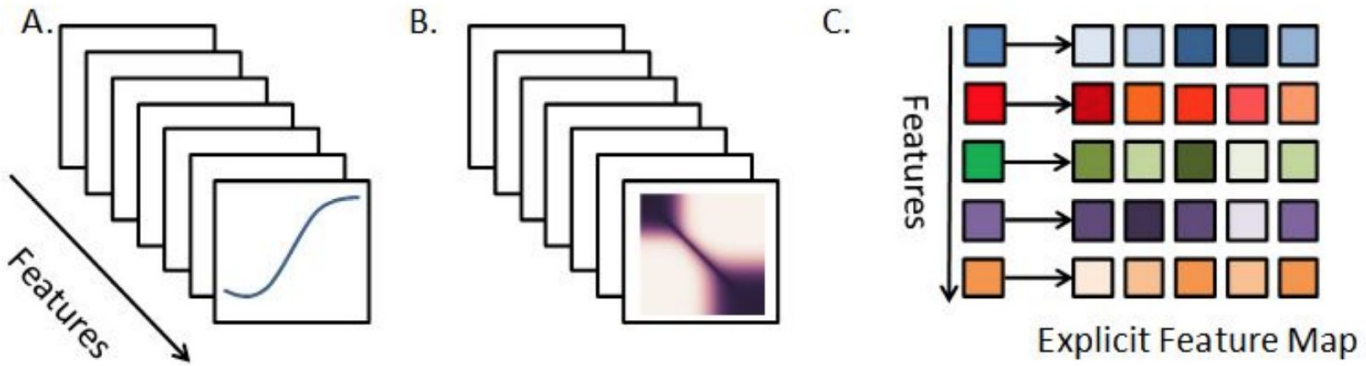


Figure 3: Learning framework block diagram. (A) For each feature, a cumulative distribution function (CDF) is estimated via training data; (B) The CDF for each feature induces a CDF kernel (Section 2.2); (C) Each feature is then transformed into a higher-dimensional space via its kernels explicit feature map (Section 2.3). These explicit maps are concatenated to form the high-dimensional feature space used by the multiple kernel learning algorithm (Section 2.4).

Data & Tasks

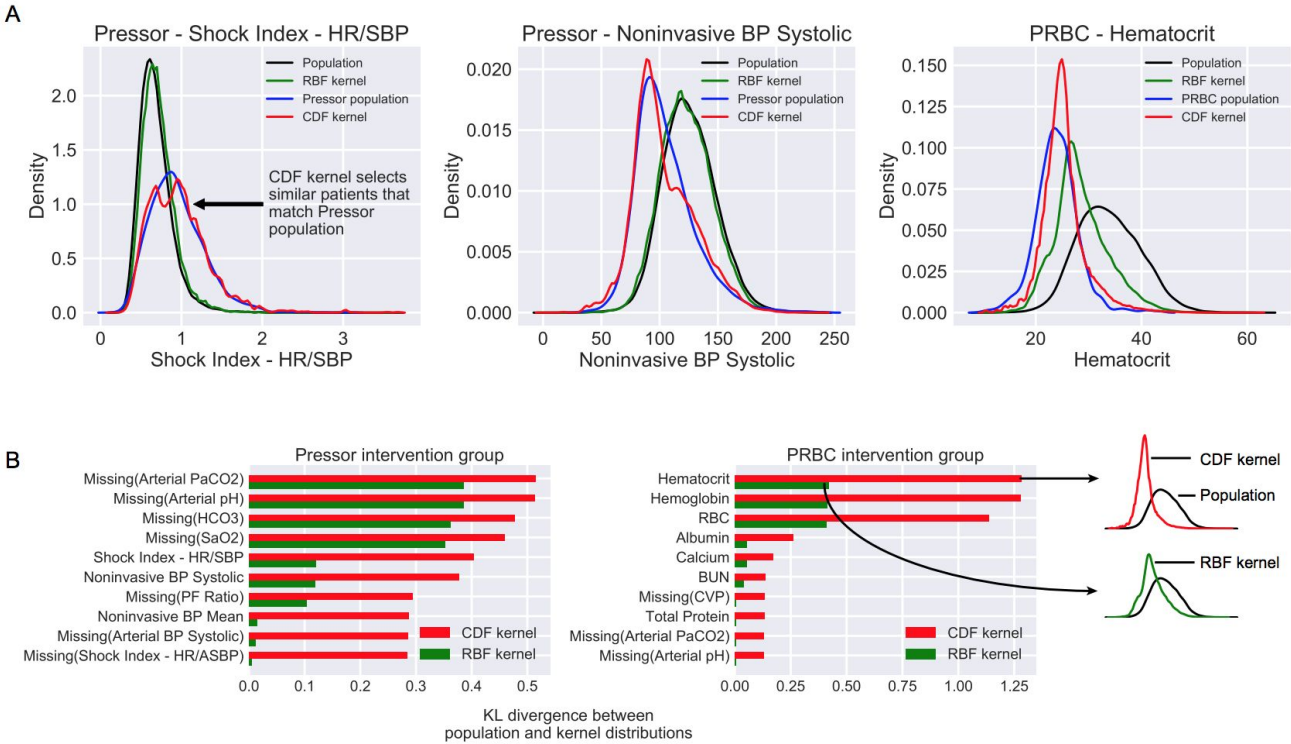
- eICU
- Hemodynamic Instability
  - Administration of inotropic or vasopressor medications
  - Administration of at least 2.4L of fluid (colloid or crystalloid) over 8 hours,
  - Administration of packed red blood cells (PRBC's)
- Patients ICU stays were divided into 6 h segments
  - labeled as either stable or unstable
  - Unstable: 6h period before any of the above intervention [As above]
  - Stable: None of the above interventions, ended stay with at least 18 h without an intervention (pick random 6 hour)
- Predict instability: AUC = 0.881 ± 0.004
- Baseline [RBF Kernel] : $\text{cross-validated AUC} = 0.874 \pm 0.007$

$$k_j(x, z) = \exp(-\gamma(x_j - z_j)^2)$$

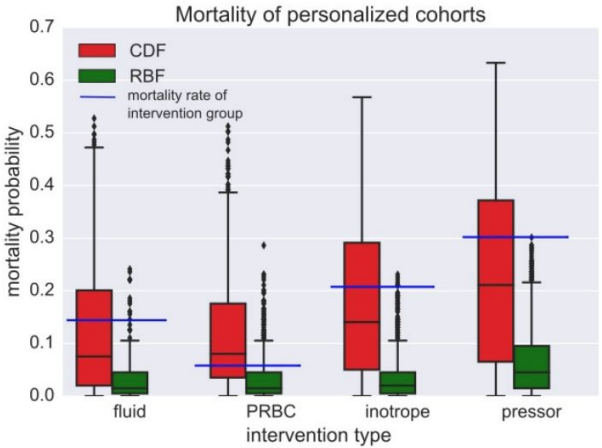
Visualizing Learned Model

- Top 3 predictive features:
  - Noninvasive Systolic Blood Pressure
  - Hematocrit
  - Shock Index
- Next up, using the kernel for evaluating similarity:
  - First grouped hemodynamically unstable patients by the intervention they eventually received (PRBC, fluid, inotrope, or pressor)
  - For a new patient within each group, they ask, can you get a personalized cohort (similar to this) patient
    - Don’t specify this but likely done by evaluating the weighted kernel, ranking and picking by some threshold

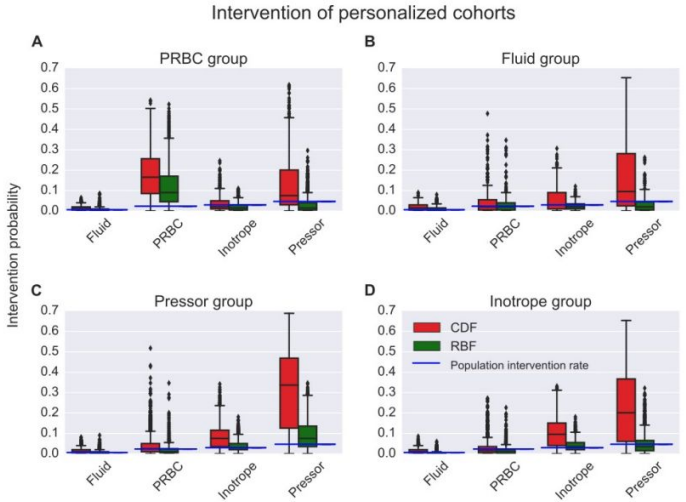
Results -- Evaluating ranked performance



Similarity by Mortality



(a)



(b)

Figure 5: (a) Mortality rate of personalized cohort for each intervention group. Compared to RBF (green), CDF-based cohorts (red) have mortality rates that are closer to the true mortality rate observed in a given intervention group. (b) Interventions given to personalized cohort, grouped by intervention.

Overview

- Patient features have different distributions that are often very difficult to reason about:
  - They lack the statistical redundancy across pixels that characterizes images
- This kernel is intuitive -- captures interesting facets of patient similarity
- Limitations:
  - Currently, features are still assumed to have been independent
  - Thought exercise: how might one incorporate correlations between features
  - Could be a useful building block for more interesting non-linear representations [left for future work by the authors]
- Does this break iid in small sample data?