# The Single-Point Crossover Process

**Anonymous Author(s)**
Affiliation
Address
email

## Abstract

The last decade has seen a burst of data optimisation techniques, mainly via noisification mechanisms, in two separate contexts: (i) improve supervised learning (as in dropout) and (ii) enhance individual privacy (as in differential privacy). In this paper, we tackle both objectives for a *process* level of protection on data: we want to keep the utility of data for supervised learning while being able to control — or fool — *causal inference attacks* on subsets of its features. It is well known that privacy protection and learnability are usually two conflicting objectives. Usually, but not always: we show how to jointly control causal inference attacks *and* learnability by a noise-free process that mixes observations in the training sample, the Single-Point Crossover Process (SPCP). One key point is that the SPCP is typically able to alter joint distributions *without* touching on marginals, nor altering the sufficient statistics for the class — in other words, it saves (and sometimes improves) generalization abilities for learning, but can fool causal inference attacks into misleading *ad-hoc* conclusions. Extensive experiments validate the theory and display the competitive advantages of SPCPs.

## 1 Introduction

— "Even when individuals are not 'identifiable', they may still be 'reachable', may still be comprehensibly represented in records that detail their attributes and activities, and may be subject to *consequential inferences* and predictions taken on that basis" [5].

There are at least too good reasons to alter the training sample of a supervised learning problem. One is privacy, a growing concern in the public sphere [5, 15, 17]. Two leading mechanisms for the private release of data are differential privacy and $k$-anonymity [14, 15, 29, 38]. They guarantee the *individual (identifiability) level* protection mentioned above [5] and rely on low-level modifications, typically touching marginals, mostly with noise. The other good reason to alter a training sample is to optimize learning: feature bagging [9, 25, 37], importance weighting in features, boosting [23, 30], denoising autoencoders [40], independent noisification, dropout [3, 36, 39], all are methods that optimise the presentation of observation variables to a learner with the objective to improve its generalization abilities.

In 2014, a report to the US president by the Presidents Council of Advisors on Science and Technology on big data and privacy judged, in its first recommendation, that "policies focused on the regulation of data collection [...] and analysis (*absent identifiable actual uses of the data or products of analysis*) are unlikely to yield effective strategies" [17]. True causality is deemed very sensitive but its imputation in Big Data is judged "a research field in its infancy" [17]. It is however hard to exagerate the recent burst in causal inference techniques [8, 12, 13, 18, 19, 20, 21, 26, 27, 33], as well as the threats this may pose on privacy issues [4, 5, 15, 24]. In this framework, rather than consequential inference [5], we shall name them causal inference *attacks*, and we argue that they deserve an adequate *process level* protection for data [4, 5], more than an all-purpose individual level protection. For example, in a medical diagnosis data which gives a sickness state as a function of genetic, behavioral, habits

Submitted to 29th Conference on Neural Information Processing Systems (NIPS 2016). Do not distribute.

and infection history, we may want to make causal inference between specific traits and behaviour harder, or we may want to hide gender-prone infections [34], *while* making sure that the utility of the dataset for predicting the sickness state remains unaltered. Rather than making it harder, ultimately, we may just look to *reverse* the observable causal direction between specific observation variables, and thereby fool causal inference or causal rule mining. In short, the dataset's utility for the black-box supervised prediction task remains within control, but it is surgically altered against fined-grained specific causal inference attacks among features.

Is such a task within technical reach ? It is well known that individual protection mechanisms do not get on well with optimizing learning [10, 29, 41]. Here comes an eventually surprising observation: coping with our protection level may require wrangling the complete data, but this may be done with a tight control of its utility for supervised learning, and it may even yield *better* models for prediction.

— This is our main contribution: to achieve these two goals, we introduce the *Single Point Crossover Process*, SPCP. An analogy may be done with the biological crossover: a population of DNA strands gets mixed with a crossover, but there is a single zone for chiasma (*i.e.* contact point) for the whole population. In the same way as DNA strands exchange genetic material during recombination, feature values get mixed between observations during a SPCP (in a more general way than the exchange in genetic recombination). The key to learning is that the SPCP may be done without changing the sufficient statistics for the class, nor touching marginals.

**Organisation of the paper** — Section §2 gives general definitions. Section §3 presents the Single-Point Crossover Process and its relationships with learnability. Section §5 presents the applications of the SPCP and details related experiments. A last Section discusses and concludes. A Supplementary Material (SM) [1] provides all proofs, additional results and the complete experiments performed. A movie is also included in SM, showing the effects of the SPCP on a popular domain for causal discovery [20].

## 2 General notations and definitions

**Learning setting** — We let $[m] \doteq \{1, 2, ..., m\}$ and $\Sigma_m \doteq \{\boldsymbol{\sigma} \in \{-1, 1\}^m\}$. $\mathcal{X} \subseteq \mathbb{R}^d$ is a domain of observations. Examples are couples (observation, label) $\in \mathcal{X} \times \Sigma_1$, sampled i.i.d. according to some unknown but fixed distribution $\mathcal{D}$. We denote $\mathcal{F} \doteq [d]$ the set of observation attributes (or features). $\mathcal{S} \doteq \{(\boldsymbol{x}_i, y_i), i \in [m]\} \sim \mathcal{D}_m$ is a training sample of $|\mathcal{S}| = m$ examples. For any vector $\boldsymbol{z} \in \mathbb{R}^d$, $z_j$ denotes its coordinate $j$. Finally, notation $x \sim X$ for $X$ a set denotes uniform sampling in $X$, and the mean operator is $\boldsymbol{\mu}_{\mathcal{S}} \doteq \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{S}}[y \cdot \boldsymbol{x}]$ [32].

In supervised learning, the task is to learn classifier $\mathcal{H} \ni h : \mathcal{X} \to \mathbb{R}$ from $\mathcal{S}$ with good generalisation properties, that is, having a small *true risk* $\mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}[L_{0/1}(y, h(\boldsymbol{x}))]$, with $L_{0/1}(z, z') \doteq 1_{zz' \leq 0}$ the 0/1 loss ($1_{\cdot}$ is the indicator variable). In general, this is achieved by minimising over $\mathcal{S}$ a $\varphi$*-risk* $\mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{S}}[\varphi(yh(\boldsymbol{x}))] = (1/m) \cdot \sum_i \varphi(y_i h(\boldsymbol{x}_i))$, where $\varphi(z) \geq 1_{z \leq 0}$ is a *surrogate* of the 0/1 loss. In this paper, $\varphi$ is any differentiable proper symmetric (PS) loss [28] (symmetric meaning that there is no class-dependent misclassification cost). The logistic, square and Matsushita losses are examples of PS losses. Set $\mathcal{H}$ is a predefined set of classifiers, such as linear separators, decision trees, etc. .

**Matrix quantities** — The set of unnormalised column stochastic (UCS) matrices, $\mathcal{M}_n \subset \mathbb{R}^{n \times n}$, is the superset of column stochastic matrices for which we drop the non-negativity constraint, thus keeping the sole constraint of unit per-*column* sums. We let $S_n \subset \mathcal{M}_n$ denote the symmetric group of order $n$. For any $\mathrm{A}, \mathrm{B} \in \mathbb{R}^{n \times n}$ and $\mathrm{M} \in \mathcal{M}_n$, we let $\langle \mathrm{A}, \mathrm{B} \rangle_{\mathrm{M}} \doteq \mathrm{tr}\left((\mathrm{I}_n - \mathrm{M})^\top \mathrm{A} (\mathrm{I}_n - \mathrm{M}) \mathrm{B}\right)$ denote the *centered inner product* of A and B with respect to M. It is a generalisation of the centered inner product used in kernel statistical tests of independence [18], for which $\mathrm{M} = (1/n)\mathbf{1}\mathbf{1}^\top$.

Without loss of generality, we shall assume that indexes in $\mathcal{S}$ cover first the positive class: $(y_i = +1 \wedge y_{i'} = -1) \Rightarrow i < i'$. A key subset of matrices of $\mathbb{R}^{m \times m}$ consists of block matrices whose coordinates on indexes corresponding to different classes in $\mathcal{S}$ are zero: block-class matrices.

**Definition 1** $\mathrm{A} \in \mathbb{R}^{m \times m}$ *is a* ***block-class matrix*** *iff* $(y_i \cdot y_{i'} = -1) \Rightarrow \mathrm{A}_{ii'} = 0, \forall i, i'$.

An asterisk exponent in a subset of matrices indicates the intersection of the set with block class matrices, such as for $\mathcal{M}_n^* \subset \mathcal{M}_n$ and $S_n^* \subset S_n$. Finally, matrix entries are noted with double indices like $\mathrm{M}_{ii'}$, and replacing an index by a dot, "·", indicates a sum over the index, like $\mathrm{M}_{i\cdot} \doteq \sum_{i'} \mathrm{M}_{ii'}$.

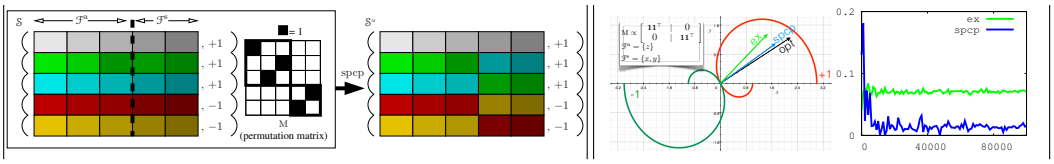


Figure 1: *Left*: example of SPCP with M a block-class permutation matrix (the two blocks are in bold). *Center+Right*: toy domain where $d = 3$, but all examples have zero $z$-coordinate (not shown). The SPCP uniformly mixes examples by class. The domain consists of two spirals (red for positive, green for negative examples) with $\mathcal{D}$ = uniform distribution. Arrows depict respectively the optimal direction (black), and the directions learned by minimizing $\varphi$ = square loss over $\mathcal{S}$ (light green) and $\mathcal{S}^{\mathscr{T}}$ (cyan). The rightmost plot displays test errors ($y$-scale) on uniform sampling of datasets of different sizes ($x$-scale). The effect of the SPCP is to produce in $\mathcal{S}^{\mathscr{T}}$ *two* distinct examples that *average* the positive / negative examples, and yield a better approximation of the optimum.

## 3 The Single-Point Crossover Process

The single-point crossover process (SPCP) transforms $\mathcal{S}$ in two steps: the split and the shuffle step. In the split step, a bi-partition of the features set $\mathcal{F}$ is computed: $\mathcal{F} = \mathcal{F}_{\rm a} \cup \mathcal{F}_{\rm s}$. $\mathcal{F}_{\rm a}$ is the *a*nchor set and $\mathcal{F}_{\rm s}$ is the *s*huffle set. To perform the shuffle step, we need some additional notations. Without loss of generality, we assume $\mathcal{F}_{\rm a} \doteq [d_{\rm a}]$ and $\mathcal{F}_{\rm s} \doteq \{d_{\rm a} + j, j \in [d_{\rm s}]\}$, $d_{\rm a} > 0, d_{\rm s} > 0, d_{\rm a} + d_{\rm s} = d$. So, $\mathcal{F}_{\rm a}$ contains the first $d_{\rm a}$ features and $\mathcal{F}_{\rm s}$ contains the last $d_{\rm s}$ features. Let $\mathrm{I}_d$ be the identity matrix, and $[\mathrm{F}^{\rm a}|\mathrm{F}^{\rm s}] = \mathrm{I}_d$ be a vertical block partition where $\mathrm{F}^{\pi} \in \mathbb{R}^{d \times d_\pi}$ has columns representing the features of $\mathcal{F}_\pi$ (for $\pi \in \{\mathrm{a}, \mathrm{s}\}$). Finally, we define the (*row*-wise) observation matrix $\mathrm{S} \in \mathbb{R}^{m \times d}$ with $(\mathrm{S})_{ij} \doteq x_{ij}$. Let $\mathbf{1}_i$ be the $i^{th}$ canonical basis vector.

**Definition 2** *For any block partition* $[\mathrm{F}^{\rm a}|\mathrm{F}^{\rm s}] = \mathrm{I}_d$ *and any* ***shuffle matrix*** $\mathrm{M} \in \mathcal{M}_n$*, the single-point crossover process* $\mathscr{T} \doteq \textsc{spcp}(\mathcal{S}; \mathrm{F}^{\rm a}, \mathrm{F}^{\rm s}, \mathrm{M})$ *returns* $m$*-sample* $\mathcal{S}^{\mathscr{T}}$ *such that its observation matrix is* $\mathrm{S}^{\rm M} \doteq [\mathrm{SF}^{\rm a}|\mathrm{MSF}^{\rm s}]$*, and each example* $\mathcal{S}^{\mathscr{T}} \ni (\boldsymbol{x}_i^{\rm M}, y_i) \doteq ((\mathrm{S}^{\rm M})^\top \mathbf{1}_i, y_i)$.

M can be fixed beforehand or learned with data. Figure 1 (left) presents the SPCP on a toy data with M a permutation matrix. Figure 1 (center + right) presents another example with M block-uniform.

**The Single-Point Crossover Process and learnability** — We now explore the effect of the SPCP on generalisation. We need two assumptions on $\mathcal{H}$ and $\varphi$. The first is a weak linearity condition on $\mathcal{H}$:

**(i)** $\forall h \in \mathcal{H}$, $\exists$ classifiers $h_\pi$ over features of $\mathcal{F}_\pi$ ($\pi \in \{\mathrm{a}, \mathrm{s}\}$) s. t. $h(\boldsymbol{x}) = \sum_\pi h_\pi((\mathrm{F}^\pi)^\top \boldsymbol{x})$.

Such an assumption is also made in the feature bagging model [37]. We let $\mathcal{H}_{\rm s}$ denote the set of all $h_{\rm s}$. Any linear classifier satisfies **(i)**, but also any linear combination of arbitrary classifiers, each learnt over one of $\mathcal{F}_\pi$ for $\pi \in \{\mathrm{a}, \mathrm{s}\}$. The second postulates that key quantities are bounded [6]:

**(ii)** $0 \leq \varphi(z) \leq K_\varphi, \forall z$ and $|h_{\rm s}((\mathrm{F}^{\rm s})^\top \boldsymbol{x})| \leq K_{\rm s}, \forall \boldsymbol{x} \in \mathcal{X}, \forall h_{\rm s} \in \mathcal{H}_{\rm s}$.

$(\mathrm{F}^{\rm s})^\top \boldsymbol{x}$ picks the features of $\boldsymbol{x}$ in $\mathcal{F}_{\rm s}$. Let $\mathrm{R}_{\mathcal{S}}(\mathcal{H}) \doteq \mathbb{E}_{\boldsymbol{\sigma} \sim \Sigma_m}\left[\sup_{h \in \mathcal{H}} |(1/m) \cdot \sum_i \sigma_i h\left(\boldsymbol{x}_i\right)|\right]$ be the empirical Rademacher complexity of $\mathcal{H}$.

**Definition 3** *The Rademacher discrepancy* (D) *of* $\mathcal{H}$ *with respect to* $\mathscr{T} \doteq \textsc{spcp}(\mathcal{S}; \mathrm{F}^{\rm a}, \mathrm{F}^{\rm s}, \mathrm{M})$ *is:*

$$\mathrm{D}_{\mathscr{T}}(\mathcal{H}) \quad \doteq \quad \mathbb{E}_{\boldsymbol{\sigma} \sim \Sigma_m}\left[\sup_{h \in \mathcal{H}_{\rm s}} \left| \frac{1}{m} \sum_i \sigma_i \left( h((\mathrm{SF}^{\rm s})^\top \mathbf{1}_i) - h((\mathrm{MSF}^{\rm s})^\top \mathbf{1}_i) \right) \right| \right] \quad . \tag{1}$$

Notice that the Rademacher discrepancy is computed over the shuffle set of features only, and $(\mathrm{SF}^{\rm s})^\top \mathbf{1}_i = (\mathrm{F}^{\rm s})^\top \boldsymbol{x}_i$. We investigate generalization under two different settings on $\mathcal{H}$ / $\mathscr{T}$:

Setting (A) Classifiers $h^{\rm s}$ in **(i)** above are linear;

Setting (B) $\mathrm{M} \in S_m^*$.

**Theorem 4** *Consider any $\mathcal{H}$, $\varphi$ and split $\mathcal{F} = \mathcal{F}_{\rm a} \cup \mathcal{F}_{\rm s}$ such that **(i)** and **(ii)** hold. For any $m$ and any $\delta > 0$, with probability $\geq 1 - \delta$ over i.i.d. $m$-sample $\mathcal{S}$, we have:*

$$\mathbb{E}_{\mathcal{D}}\left[L_{0/1}(y, h(\boldsymbol{x}))\right] \quad \leq \quad \mathbb{E}_{\mathcal{S}^{\mathscr{T}}}\left[\varphi(yh(\boldsymbol{x}))\right] + \mathrm{D}_{\mathscr{T}}(\mathcal{H}) + \frac{4}{b_\varphi} \cdot \mathrm{R}_{\mathcal{S}}(\mathcal{H}) + (2K_\varphi + K_{\rm s}) \cdot \sqrt{\frac{2}{m}\log\frac{3}{\delta}} \ ,$$

*for every classifier $h$ and every $\mathscr{T} \doteq \textsc{spcp}(\mathcal{S}; \mathrm{F}^{\rm a}, \mathrm{F}^{\rm s}, \mathrm{M})$ such that $\mathrm{M} \in \mathcal{M}_m^*$ and whichever of (A) or (B) holds. Here, $b_\varphi > 0$ is a constant depending on $\varphi$.*

([1], Subsection 2.1) Notice that Theorem 4 requires that M is a block-class matrix. A key to the proof is the invariance of the mean operator: $\boldsymbol{\mu}_{\mathcal{S}} = \boldsymbol{\mu}_{\mathcal{S}^{\mathscr{T}}}$. The roles of $\mathcal{F}_{\rm a}$ and $\mathcal{F}_{\rm s}$ may be interchanged for Setting (B). This allows to improve at no cost the result, which thus holds over the *min* over the permutation of the roles.

Theorem 4 gives a perhaps counterintuitive rationale for the SPCP that goes beyond our framework to machine learning at large: *learning over a* SPCP*'ed $\mathcal{S}$ may improve generalization over $\mathcal{D}$ as well*. By means of words, learning over transformed data may improve generalization over the initial domain. Figure 1 (right) gives a toy example for which this holds. It is also not hard to exhibit domains for which we even have:

$$\min_h \mathbb{E}_{\mathcal{S}^{\mathscr{T}}}\left[\varphi(yh(\boldsymbol{x}))\right] + \mathrm{D}_{\mathscr{T}}(\mathcal{H}) \quad < \quad \min_h \mathbb{E}_{\mathcal{S}}\left[\varphi(yh(\boldsymbol{x}))\right] \ . \tag{2}$$

To save space, we present such an example in the Supplementary Material ([1], Subsection 2.2). Theorem 4 says that a key to good generalization is the "complexity" of the process encapsulated in $\mathrm{D}_{\mathscr{T}}(\mathcal{H})$. We would typically want it to be small compared to the Rademacher complexity. The rest of this Section shows that (and when) this is indeed achievable.

**Upperbounds on $\mathbf{D}_{\mathscr{T}}(\mathcal{H})$** — The following Lemma establishes a first bound on $\mathrm{D}_{\mathscr{T}}(\mathcal{H})$.

**Lemma 5** *if $\mathscr{T}$ satisfies the conditions of Theorem 4, then $\mathrm{D}_{\mathscr{T}}(\mathcal{H}) \leq 2 \cdot \mathrm{R}_{\mathcal{S}'}(\mathcal{H}_{\rm s})$, for $\mathrm{S}' \doteq (\mathrm{I}_m - \mathrm{M})\mathrm{SF}^{\rm s}$ in Setting (A), and $\mathrm{S}' \doteq \mathrm{SF}^{\rm s}$ in Setting (B). $\mathrm{S}'$ is the row-wise observation matrix of $\mathcal{S}'$.*

**Proof** (Sketch) Consider for example Setting (B). In this case, recalling that $(\mathrm{SF}^{\rm s})^\top \mathbf{1}_i = (\mathrm{F}^{\rm s})^\top \boldsymbol{x}_i$ and letting $\varsigma : [m] \to [m]$ denote the permutation that M represents, we have:

$$\begin{aligned}
\mathrm{D}_{\mathscr{T}}(\mathcal{H}) \quad &= \quad \mathbb{E}_{\boldsymbol{\sigma}\sim\Sigma_m}\left[\sup_{h\in\mathcal{H}_{\rm s}}\left|\frac{1}{m}\sum_i \sigma_i\left(h((\mathrm{F}^{\rm s})^\top\boldsymbol{x}_i) - h((\mathrm{F}^{\rm s})^\top\boldsymbol{x}_{\varsigma(i)})\right)\right|\right] \qquad (3)\\
&\leq \quad \mathbb{E}_{\boldsymbol{\sigma}\sim\Sigma_m}\left[\sup_{h\in\mathcal{H}_{\rm s}}\left|\frac{1}{m}\sum_i \sigma_i h((\mathrm{F}^{\rm s})^\top\boldsymbol{x}_i)\right|\right] + \mathbb{E}_{\boldsymbol{\sigma}\sim\Sigma_m}\left[\sup_{h\in\mathcal{H}_{\rm s}}\left|\frac{1}{m}\sum_i \sigma_i h((\mathrm{F}^{\rm s})^\top\boldsymbol{x}_{\varsigma(i)})\right|\right]\\
&\quad = 2 \cdot \mathrm{R}_{\mathcal{S}'}(\mathcal{H}_{\rm s}) \ ,
\end{aligned}$$

as claimed. The case of Setting (A) follows the same path. ■

Lemma 5 says that $\mathrm{D}_{\mathscr{T}}(\mathcal{H})$ is at most twice a Rademacher complexity over the *shuffle set*. This bound is however loose since many terms can cancel in the sum of eq. (3), and the inequality does not take this into account. We now exploit this to show bounds that can be much smaller.

**Theorem 6** *Under Setting (A), suppose any $h_{\rm s}$ is of the form $h_{\rm s}(\boldsymbol{x}) = \boldsymbol{\theta}^\top\boldsymbol{x}$ with $\|\boldsymbol{\theta}\|_2 \leq r_{\rm s}$, for some $r_{\rm s} > 0$. Let $\mathrm{K}^{\rm s} \doteq \mathrm{SF}^{\rm s}(\mathrm{SF}^{\rm s})^\top$. Then $\exists u \in (0,1)$ depending only in $\mathcal{S}$ such that for any $\mathrm{M} \in \mathcal{M}_m$,*

$$\mathrm{D}_{\mathscr{T}}(\mathcal{H}) \quad \leq \quad (ur_{\rm s}/m) \cdot \sqrt{\langle \mathrm{I}_m, \mathrm{K}^{\rm s}\rangle_{\mathrm{M}}} \ . \tag{4}$$

Notice that $\mathrm{K}^{\rm s}$ is a Gram matrix in the shuffle feature space. The proof technique [1] (Subsection 2.3) relies on a data-dependent expression for $u$ which depends on the cosines of angles between the observations in $\mathcal{S}$. It can be used to refine and improve a popular bound on the empirical Rademacher complexity of linear classifiers [22] (we give the proof in [1] (Theorem 4)). We now investigate an upperbound on Setting (B) in which classifiers in $\mathcal{H}^{\rm s}$ are (rooted) directed acyclic graph (DAG), like decision trees, with bounded real valued predictions (say, $K_{\rm s} > 0$) at the leaves. Each classifier $h_{\rm s}$ defines a partition over $\mathcal{X}$. We let $\mathcal{H}^{\rm s}_+$ be the subset of $\mathcal{H}^{\rm s}$ in which all leaves have in absolute value the largest magnitude, *i.e.*, $K_{\rm s}$. Remark that we may have $|\mathcal{H}^{\rm s}_+| \ll \infty$ while $|\mathcal{H}^{\rm s}| = \infty$ in general.

**Theorem 7** *Under Setting (B), suppose* $\mathcal{H}^{\mathrm{s}}$ *is* DAG *and assumption* ***(ii)*** *is satisfied. Suppose that* $\log|\mathcal{H}^{\mathrm{s}}_{+}| \geq (4\varepsilon/3)\cdot m$ *for some* $\varepsilon > 0$*. Then, letting* $\mathrm{odd\_cycle}(\mathrm{M})$ *denote the set of odd cycles (excluding fixed points) of* M*, we have:*

$$\mathrm{D}_{\mathscr{T}}(\mathcal{H}) \quad \leq \quad K_{\mathrm{s}} \cdot \sqrt{\frac{2}{m} \cdot \log \frac{|\mathcal{H}^{\mathrm{s}}_{+}|}{(1+\varepsilon)^{|\mathrm{odd\_cycle}(\mathrm{M})|}}} \ . \tag{5}$$

([1], Subsection 2.4) The assumption on $\mathcal{H}^{\mathrm{s}}_{+}$ is not restrictive and would be met by decision trees, branching programs, etc. (and subsets). The Rademacher complexity of decision trees would roughly be the right-hand side of (5) *without* the denominator in the $\log$. Hence, the Rademacher discrepance may be much smaller than the Rademacher complexity for more "involved" SPCPs. The number of cycles is not the only relevant parameter of the SPCP on which relies non-trivial bounds on $\mathrm{D}_{\mathscr{T}}(\mathcal{H})$: the Supplementary Material presents, for the interested reader, a proof that the number of fixed points is another parameter which can decrease significantly the *expected* Rademacher discrepancy (by a factor $\sqrt{1-|\mathrm{fixed\_points}|/m}$), when SPCPs are picked at random.

# 4 The SPCP, learnability, inference and causality

**The Single-Point Crossover Process and measures of independence** — In this section, we assume that $\mathcal{S}$ is subject to quantitative tests of independence, that is, assessing $\mathcal{U} \perp\!\!\!\perp \mathcal{V}$ for some $\mathcal{U}, \mathcal{V} \subset \mathcal{X}$. We therefore compute SPCPs such that $\mathcal{U} \subseteq \mathcal{F}_{\mathrm{a}}$ and $\mathcal{V} \subseteq \mathcal{F}_{\mathrm{s}}$, so that the SPCP alters the measure of independence. The problem is thus essentially the design of the shuffle matrix M. One popular criterion to determine (conditional) (in)dependence is Hilbert-Schmidt Independence Criterion [13, 18, 19].

**Definition 8** *Let* $\mathcal{U} \subset [d]$ *and* $\mathcal{V} \subset [d]$ *be non-empty and disjoint. Let* $\mathrm{K}^u$ *and* $\mathrm{K}^v$ *be two kernel functions over* $\mathcal{U}$ *and* $\mathcal{V}$ *computed using* $\mathcal{S}$*. The (unnormalised) Hilbert-Schmidt Independence Criterion (*HSIC*) between* $\mathcal{U}$ *and* $\mathcal{V}$ *is defined as* $\mathrm{HSIC}(\mathrm{K}^u, \mathrm{K}^v) \doteq \langle \mathrm{K}^u, \mathrm{K}^v \rangle_{(1/m)\mathbf{1}\mathbf{1}^\top}$.

We choose not to normalise the HSIC: various exist but they mainly rely on a multiplicative factor depending on $m$ only, so they do not affect the results to come. Our first result shows more than just how to pick M in the SPCP so as to alter HSIC: (a) while storing kernels requires $O(m^2)$ space, controlling the evolution of the HSIC requires only *linear*-space information about kernels, and (b) this information can be computed beforehand, and can be efficiently approximated from low-rank approximations of the kernels [2]. Hereafter, we restrict ourselves to the scenario of the *decrease* of the HSIC, yet all our results would apply to the opposite polarity for the modification. We use for sake of readability the shorthand HSIC for $\mathrm{HSIC}(\mathrm{K}^u, \mathrm{K}^v)$.

**Theorem 9** *Let* HSIC *and* $\mathrm{HSIC}_{\mathscr{T}}$ *denote the* HSIC *before and after applying the* SPCP $\mathscr{T}$ *to* $\mathcal{S}$ *(*$\mathrm{K}^u$ *and* $\mathrm{K}^v$ *implicit). Let* $\tilde{\boldsymbol{u}} \doteq (1/m)\sum_i \lambda_i(\mathbf{1}^\top \boldsymbol{u}_i)\boldsymbol{u}_i$*,* $\tilde{\boldsymbol{v}} \doteq (1/m)\sum_i \mu_i(\mathbf{1}^\top \boldsymbol{v}_i)\boldsymbol{v}_i$*, where* $\{\lambda_i, \boldsymbol{u}_i\}_{i\in[d]}$*,* $\{\mu_i, \boldsymbol{v}_i\}_{i\in[d]}$ *are respective eigensystems of* $\mathrm{K}^u$ *and* $\mathrm{K}^v$*. Then* $\mathrm{HSIC}_{\mathscr{T}} < \mathrm{HSIC}$ *iff* $\tilde{\boldsymbol{u}}^\top(\mathrm{I}_m - \mathrm{M})\tilde{\boldsymbol{v}} < 0$.

([1], Subsection 2.5) In the following Theorem, we compose SPCP processes with $T$ different elementary permutation shuffling matrices. Notice that since the composition of permutation matrices is a permutation matrix, when the matrix of the final process $\mathrm{HSIC}_{\mathscr{T}_T}$ is block class, Theorem 4 can be applied *directly* to $\mathrm{HSIC}_{\mathscr{T}_T}$. We also let $\mathscr{R}^{u,v} \doteq m\left(1 - (\mathrm{K}^u_{\cdot\cdot} + \mathrm{K}^v_{\cdot\cdot})/(2m^2)\right)$.

**Theorem 10** *Suppose* $\mathcal{S}$ *is shuffled by a sequence of* $T = \epsilon m$ *elementary permutation (*$\epsilon > 0$*) and the kernels* $\mathrm{K}^u$ *and* $\mathrm{K}^v$ *have unit diagonal. Suppose that the initial* $\mathrm{HSIC} > \mathscr{R}^{u,v}$*. Then there exists such a sequence such that* $\mathscr{T}_T$ *satisfies* $\mathrm{HSIC}_{\mathscr{T}_T} \leq \mathscr{R}^{u,v} + \alpha \cdot (\mathrm{HSIC} - \mathscr{R}^{u,v})$*, where* $\alpha \doteq \exp(-8\epsilon)$.

The proof ([1], Subsection 2.6) states a more general result, not restricted to unit diagonal kernels. Theorem 10 is a worst-case result: some sequences of permutations may be much more efficient in decreasing HSIC. If we compare this bound to Theorem 3 in [19], then $\mathscr{R}^{u,v}$ may be *below* the expectation of the HSIC, and so Theorem 10 guarantees efficient jamming of dependence.

We finish with independence on a rather specific causality model, but that we feel is enlightening on one important potential of SPCP with respect to independence: trick statistical tests into keeping

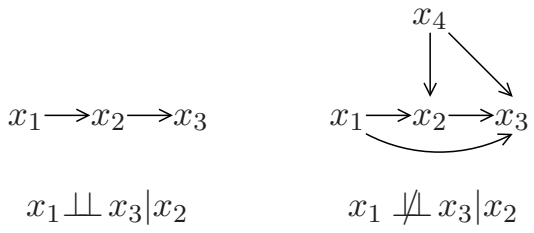


Figure 2: The Cornia-Mooij model [12]. Left: belief, right: true model.

**Algorithm 1** SPCP-Train DT($\mathcal{S}, T, \boldsymbol{\theta}_0, \mathrm{M}_0, \mathrm{F}^{\mathrm{a}}, \mathrm{F}^{\mathrm{s}}; \mathscr{G}_1, [\mathscr{G}_2]$)

**Input** Sample $\mathcal{S}$, iterations $T$, classifier $\boldsymbol{\theta}_0$, initial SPCP $\mathscr{T}$ (matrices $\mathrm{M}_0 \in S_m^*$, $[\mathrm{F}^{\mathrm{a}}|\mathrm{F}^{\mathrm{s}}] = \mathrm{I}_m$);
Step 1 : **for** $t = 1, 2, ..., T$
  Step 1.1 : let $\mathrm{M} \leftarrow \arg\min_{\mathrm{M}' \in S_m^{\mathrm{e}*}} \mathscr{G}_1(\mathrm{M}' \circ \mathrm{M}_{t-1}[|\boldsymbol{\theta}_{t-1}])$;
  Step 1.2 : let $\mathrm{M}_t \leftarrow \mathrm{M} \circ \mathrm{M}_{t-1}$;
  [Step 1.3 : let $\boldsymbol{\theta}_t \leftarrow \arg\min_{\boldsymbol{\theta} \in \mathbb{R}^d} \mathscr{G}_2(\boldsymbol{\theta}|\mathrm{M}_t)$;]
**Return** classifier $\boldsymbol{\theta}_T$ and / or dataset $\mathcal{S}^{\mathscr{T}(\mathrm{M}_T)}$

independence *and* then incur arbitrarily large errors in estimating causal effects. Also, its causal graph is so simple that it may be found as subgraph of real-world domains, thus for which the results we give would directly transfer. The model we refer to is the Cornia-Mooij (CM) model [1, 12]. In the CM model, there are $d = 3$ observation variables, and a true model which relies on a weak conditional dependence $x_1 \not\perp\!\!\!\perp x_3 | x_2$. [12] show that *if* one keeps the independence assumption $H_0$ that $x_1 \perp\!\!\!\perp x_3 | x_2$, this can lead to very high causal estimation errors, as measured by $|\mathbb{E}[x_3|x_2] - \mathbb{E}[x_3|\mathrm{do}(x_2)]|/|x_2|$ [12]. We show that the SPCP is precisely able to trick statistics into keeping $H_0$. The Supplementary Material (Subsection 2.7) presents in detail the model as well as how the SPCP may be chosen.

**The Single-Point Crossover Process and causal calculus** — Rather than the shuffle matrix M, the problem in this section is rather the design of $\mathcal{F}_{\mathrm{a}}$ and $\mathcal{F}_{\mathrm{s}}$.

# 5 Experiments

Our applications use the same meta-level algorithm (Algorithm 1), which operates in Setting (A) ∩ Setting (B) (M in $S_m^*$, linear classifiers), iteratively composing block-class elementary permutations. Here, $S_m^{\mathrm{e}*} \subset S_m$ is the set of block-class elementary permutations. The iteration step minimises a criterion $\mathscr{G}_1$ over $S_m^{\mathrm{e}*}$, and eventually after having updated the SPCP matrix, a criterion $\mathscr{G}_2$ over $\mathcal{H}$. The experimental setup and the results are provided *in extenso* in [1] (Section 3.4); Table 1 summarises them. Due to the lack of space, we comment experiments alongside the applications. The split step and the choice of $\mathcal{F}_{\mathrm{a}}$ are highly domain and task dependent: to keep experiments of reasonable length, we chose to put in $\mathcal{F}_{\mathrm{a}}$ the first half of features for each domain. In Abalone2D and Digoxin, this amounts to jam a particular ground truth (see below). Also, $\varphi$=logistic loss.

**Data optimisation for efficient learning** — We perform Algorithm 1 with $\mathrm{M}_0 = \mathrm{I}_m$, $\mathscr{G}_1(\mathrm{M}|\boldsymbol{\theta}) = \mathbb{E}_{\mathcal{S}^{\mathscr{T}}(\mathrm{M})}\left[\varphi(y\boldsymbol{\theta}^\top \boldsymbol{x})\right]$ (Theorem 4) and $\mathscr{G}_2(\boldsymbol{\theta}|\mathrm{M}) = \mathbb{E}_{\mathcal{S}^{\mathscr{T}}}\left[\varphi(y\boldsymbol{\theta}^\top \boldsymbol{x})\right] + \lambda\|\boldsymbol{\theta}\|_2^2$ where $\lambda$ is learnt through cross-validation. The algorithm returns classifier $\boldsymbol{\theta}_T$. The top row in Table 1, and [1] display that DT almost always find some permutations that reduce the test error compared to the initial data, validating this part of the theory, even when a specific data optimisation should care for a risk of over-fitting, which seems to occur on Ionosphere and Abalone2D [1] (Subsection 3.5) also compares DT as in Algorithm 1 to the one where we relax the constraint that permutations must be block-class (implying the invariance of the mean operator). The results are a clear advocacy for the constraint, as relaxing it brings poor results, from both the $\varphi$-risk and test error standpoints.

**Disrupting dependence and causality without touching marginals** — In our experiments, we run DT without step 1.3, and fixing $\mathscr{G}_1$ to be HSIC. We use two Gaussian kernels for $\mathrm{L}^u$ and $\mathrm{L}^v$, each computed over its full subset of features [1]. Experimental results display that DT achieves in general

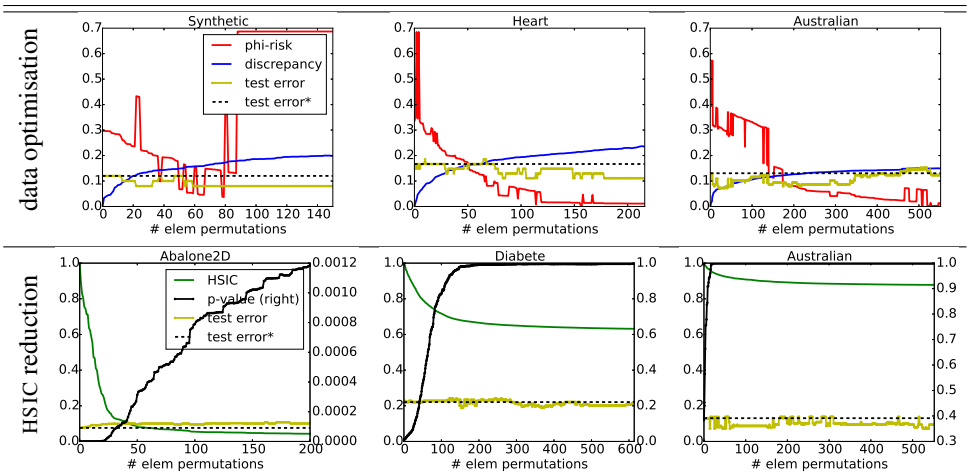


Table 1: Experiments performed with SPCP. From top row to bottom row: data optimisation, reduction in HSIC. References to domain names are provided in [1]. "test-error*" is test error over initial, non shuffled data.

a better control of the Rademacher discrepancy (than *e.g.* in data optimisation), and in several cases, manages to decrease the true error as well through the process. It also manages in general to relatively blow-up the $p$-value (computed as in [19]), even for domains for which the ground truth *implies the alternative hypothesis* $H_1$. In Abalone2D for example, a gold standard for dependence [20], while the initial $p$-value is zero up to *thirteen* digits, after shuffling, the final $p$-value exceeds $1\%$■. For the Digoxin domain, another popular domain [13] with ground truth, $p$ is very small at the beginning (which corresponds to the ground truth $D \not\perp\!\!\!\perp U$; $D$ = digoxin clearance, $U$ = urin flow). After shuffling, we obtain $p > 0.4$, which easily brings $D \perp\!\!\!\perp U$, while the ground truth is $D \perp\!\!\!\perp U | C$ ($C$ = creatinine clearance). In both cases, the effect on test error is minimal, considering that Abalone2D and Digoxin have $d = 2$ attributes only.

## 6 Discussion and conclusion

This paper introduces the Single-Point Crossover Process (SPCP), a process that cross-modifies data using a generalisation of stochastic matrices. This process can be used to cope with data optimisation for supervised learning, as well as for the problem of handling a process-level protection on data: causal inference attacks on a supervised learning dataset. In this case, the SPCP allows to release data with spotless low-level description (variable names, observed values), substantial utility (learnability), but disclosing dependences and causal effects under control, and thus that could even be crafted to be conflicting with a ground truth to protect[1]. Note that causality in big data may still be in its infancy [17], it is however rapidly growing with a variety of techniques and lots of promises. We have chosen to focus here on two major components of the actual trends (stastical measures of causal inference and causal calculus), but our technique has more applications in the field of causal discovery: suppose for example that description features denote transactions. Since we jam joint distributions without touching on marginals, our technique has direct applications in causal rule mining, with the potential to fool any spawn of Apriori — that is, level-wise association rule mining algorithms [26].

The theorey we develop fo SPCP introduces a new complexity measure of the process, the Rademacher discrepancy. We do believe that the SPCP is a good candidate in the pool of methods optimising data for learning, and may provide new metrics, algorithms and tools to devise improved solutions that fit to challenging domains not just restricted to optimizing learning or data privacy. One example is learning without entity resolution [31].

[1]Note that the initial data may not be lost, as opposed to differential privacy: knowing the noise parameters does not allow to revert differential privacy protection, while a SPCP protection is reversible when M is invertible.

# References


[1] Anonymous. Supplementary material to this paper, 2016.

[2] F. Bach. Sharp analysis of low-rank kernel matrix approximations. In *26 $^{th}$ COLT*, pages 185–209, 2013.

[3] P. Bachman, O. Alsharif, and D. Precup. Learning with pseudo-ensembles. In *NIPS*27*, pages 3365–3373, 2014.

[4] S. Barocas and H. Nissenbaum. Big data's end run around procedural privacy protection. *Communications of the ACM*, 57:31–33, 2014.

[5] S. Barocas and H. Nissenbaum. Big data's end run around anonymity and consent. In J. Lane, V. Stodden, S. Bender, and H. Nissenbaum, editors, *Privacy, Big Data, and the Public Good*, pages 44–75. Cambridge University Press, 2014.

[6] P.-L. Bartlett and S. Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *JMLR*, 3:463–482, 2002.

[7] O. Benjelloun, H. Garcia-Molina, D. Menestrina, Q. Su, S.-E. Whang, and J. Widom. Swoosh: a generic approach to entity resolution. *VLDB J.*, 18(1):255–276, 2009.

[8] L. Bottou, J. Peters, J. Quiñonero Candela, D.-X. Charles, M. Chickering, E. Portugaly, D. Ray, P.-Y. Simard, and E. Snelson. Counterfactual reasoning and learning systems: the example of computational advertising. *JMLR*, 14:3207–3260, 2013.

[9] R.-K. Bryll, R. Gutierrez-Osuna, and F.-K.-H. Quek. Attribute bagging: improving accuracy of classifier ensembles by using random feature subsets. *Pattern Recognition*, 36:1291–1302, 2003.

[10] K. Chaudhuri, C. Monteleoni, and A.-D. Sarwate. Differentially private empirical risk minimization. *JMLR*, 12:1069–1109, 2011.

[11] P. Christen. *Data Matching Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection*. Springer Data-Centric Systems and Applications, 2012.

[12] N. Cornia and J.-M. Mooij. Type-II errors of independence tests can lead to arbitrarily large errors in estimated causal effects: An illustrative example. In *30$^{th}$ UAI Workshops*, pages 35–42, 2014.

[13] G. Doran, K. Muandet, K. Zhang, and B. Schölkopf. A permutation-based kernel conditional independence test. In *30 $^{th}$ UAI*, 2014.

[14] C. Dwork and A. Roth. The algorithmic foudations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9:211–407, 2014.

[15] M. Enserink and G. Chin. The end of privacy. *Science*, 347:490–491, 2015.

[16] L. Getoor and A. Machanavajjhala. Entity resolution: theory, practice & open challenges. *Proceedings of the VLDB Endowment*, 5(12):2018–2019, 2012.

[17] S. Graham, W. Press, S.-J. Gates, M. Gorenberg, J.-P. Holden, E. Lander, C. Mundie, M. Savitz, and E. Schmidt. Big data and privacy: a technological perspective. CreateSpace Independent Publishing Platform, 2015. — President's Council of Advisors on Science and Technology.

[18] A. Gretton, O. Bousquet, A.-J. Smola, and B. Schölkopf. Measuring statistical dependence with Hilbert-Schmidt norms. In *16$^{th}$ ALT*, pages 63–77, 2005.

[19] A. Gretton, K. Fukumizu, C.-H. Teo, L. Song, B. Schölkopf, and A. Smola. A kernel statistical test of independence. In *NIPS*20*, pages 585–592, 2007.

[20] P.-O. Hoyer, D. Janzing, J.-M. Mooij, J. Peters, and B. Schölkopf. Nonlinear causal discovery with additive noise models. In *NIPS*21*, pages 689–696, 2008.

[21] D. Janzing, J.-M. Mooij, K. Zhang, J. Lemeire, J. Zscheischler, P. Daniusis, B. Steudel, and B. Schölkopf. Information-geometric approach to inferring causal directions. *Artificial Intelligence*, 182-183:1–31, 2012.

[22] S. Kakade, K. Sridharan, and A. Tewari. On the complexity of linear prediction: Risk bounds, margin bounds, and regularization. In *NIPS*21*, pages 793–800, 2008.

[23] A. Kolcz and C.-H. Teo. Feature weighting for improved classifier robustness. In $6^{th}$ *CEAS*, 2009.

[24] M.-J. Kusner, Y. Sun, K. Sridharan, and K.-Q. Weinberger. Inferring the causal direction privately. In *19$^{th}$ AISTATS*, 2016.

[25] A. Lazarevic and V. Kumar. Feature bagging for outlier detection. In *11$^{th}$ KDD*, pages 157–166, 2005.

[26] J. Li, T.-D. Le, L. Liu, J. Liu, Z. Jin, B. Sun, and S. Ma. From observational studies to causal rule mining. *ACM Trans. IST*, 7:1–27, 2016.

[27] J.-M. Mooij, J. Peters, D. Janzing, J. Zscheischler, and B. Schölkopf. Distinguishing cause from effect using observational data: methods and benchmarks. *JMLR*, 2016.

[28] R. Nock and F. Nielsen. On the efficient minimization of classification-calibrated surrogates. In *NIPS*21*, pages 1201–1208, 2008.

[29] R. Nock, G. Patrini, and A. Friedman. Rademacher observations, private data, and boosting. *32 $^{nd}$ ICML*, 2015.

[30] J. O'Sullivan, J. Langford, R. Caruana, and A. Blum. Featureboost: A meta-learning algorithm that improves model robustness. In *17 $^{th}$ ICML*, pages 703–710, 2000.

[31] G. Patrini, R. Nock, S. Hardy, and T. Caetano. Fast learning from distributed datasets without entity matching. In *26 $^{th}$ IJCAI*, 2016.

[32] G. Patrini, R. Nock, P. Rivera, and T. Caetano. (Almost) no label no cry. In *NIPS*27*, 2014.

[33] J. Pearl and E. Bareinboim. External validity: from do-calculus to transportability across populations. *Statistical Science*, 29:579–595, 2014.

[34] S.-J. Rizvi and J.-R. Haritsa. Maintaining data privacy in association rule mining. In *Proc. of the 28 $^{th}$ VLDB*, pages 682–693, 2002.

[35] L. Song, A. Smola, A. Gretton, J. Bedo, and K.-M. Borgwardt. Feature selection via dependence maximization. *JMLR*, 13:1393–1434, 2012.

[36] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *JMLR*, 15:1929–1958, 2014.

[37] C.-A. Sutton, M. Sindelar, and A. McCallum. Reducing weight undertraining in structured discriminative learning. In *10$^{th}$ HLT-NAACL*, 2006.

[38] L. Sweeney. Achieving $k$-anonymity privacy protection using generalization and suppression. *Int. J. Uncertainty, Fuzziness and Knowledge-based systems*, 10:571–588, 2002.

[39] L. van der Maaten, M. Chen, S. Tyree, and K.-Q. Weinberger. Learning with marginalized corrupted features. In *30 $^{th}$ ICML*, pages 410–418, 2013.

[40] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol. Extracting and composing robust features with denoising autoencoders. In *25 $^{th}$ ICML*, pages 1096–1103, 2008.

[41] Y.-X. Wang, S.-E. Fienberg, and A.-J. Smola. Privacy for free: Posterior sampling and stochastic gradient monte carlo. In *32 $^{nd}$ ICML*, pages 2493–2502, 2015.