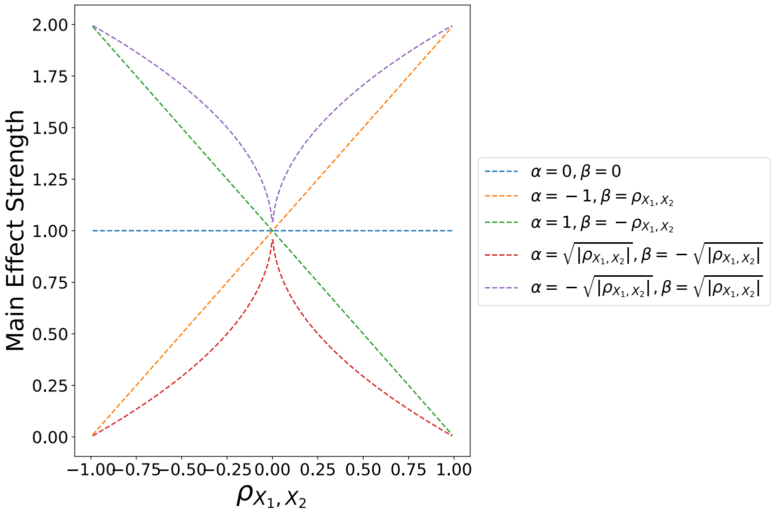


Figure 7: Strength of the main effect of X_1 implied by the coefficients of the multiplication model (3) for various settings of α, β. The setting $\alpha = \beta = 0$ mean-centers the main effects, but does not mean-center the interaction effect. In contrast, the four settings with $\alpha\beta = -\rho_{X_1,X_2}$ also mean-center $(X_1 - \alpha)(X_2 - \beta)$. Strength of the main effect of X_1 is measured as $b + d\beta$. In all experiment settings, $a = 0$, $b = c = d = 1$.

A Multiplication Model

The challenge of selecting α, β in the multiplication model (3) is not overcome by simply mean-centering all of the effects. For correlated X_1, X_2 with $\mathbb{E}[X_1] = 0, \mathbb{E}[X_2] = 0$, we have $\mathbb{E}[(X_1 - \alpha)(X_2 - \beta)] = \rho_{X_1,X_2} + \alpha\beta$. Thus, if we would want the intercept a to represent $\mathbb{E}[Y]$, we must carefully select α, β such that $\alpha\beta = -\rho_{X_1,X_2}$. This selection process has a degree of freedom. As shown in Fig. 7, different choices of α, β lead to very different conclusions about the strengths of the main effects. Thus, we need rules to pick particular coefficients from this equivalence class of models.

B Algorithm Intuition

Fig. 8 is an illustration of applying the mass-moving algorithm (Alg. 2) to purify an “OR” model into the canonical “XOR” representation. Each row in this figure represents a step in the mass-moving process, beginning with the representation in Fig. 1a and finishing with the representation in Fig. 1d.

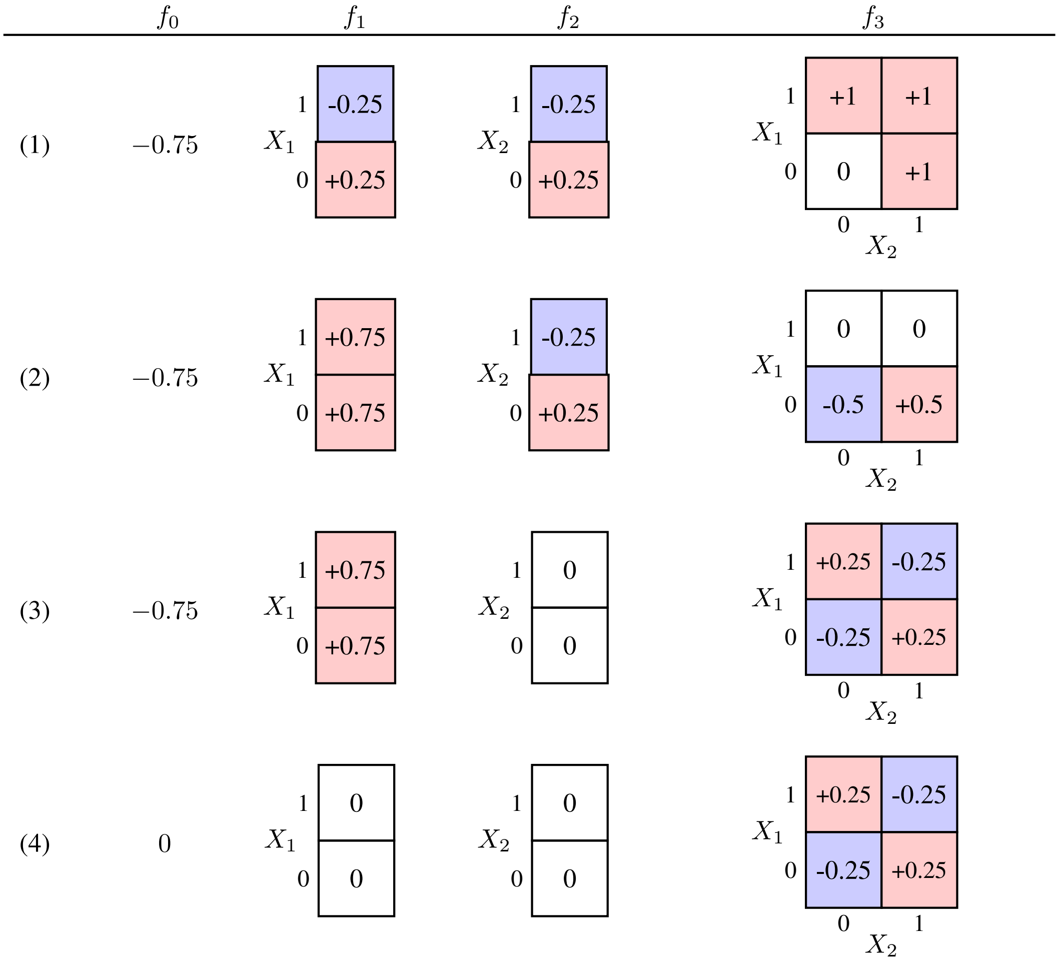


Figure 8: An illustration of applying Alg. 2 to transform the representation in Fig. 1b into the canonical form in Fig. 1d. In each row, we have an overall intercept f_0, main effects f_1 and f_2, and an interaction effect f_3. Red indicates a positive and blue a negative effect. To move from representation (1) to representation (2), we move the row-means of f_3 into the corresponding entries in f_1. To move from representation (2) to representation (3), we move the column-means of f_3 into the corresponding entries in f_2. In representation (3), there is no more mass to move from f_3, so turn our attention to f_1 and f_2. We move these means into the overall intercept f_0, resulting in representation (4). At this point, there is no mass to move in any function, so we have achieved the canonical form with pure interaction effects.

C Analysis

We will prove the rate of convergence by setting an upper-bound on M^t:

Lemma 3. *For any $T_{a,b}, w, \Omega$, if iteration t set the column means to be zero, then*

$$M^{t+1} \leq = \sum_{j \in \Omega_b} \min_{\phi_j} \left| \sum_{i \in \Omega_a} (\frac{w_{i,j}}{w_{i,\cdot}} - \phi_j)\big(\min_{\psi_i} \sum_{k \in \Omega_b} (w_{i,k} - \psi_i w_{\cdot,k}) c_k^{t-1} \big) \right| \tag{16}$$

C.1 Proof of Lemma 3

Proof. Let us consider a matrix T representing the interaction effect of variables X_a and X_b. Let X_a take on values from the set Ω_a,, and X_b take on values from the set Ω_b, with $m = |\Omega_a|$, $n = |\Omega_b|$. Let w be a density defined on Ω_a and Ω_b, normalized such that $\sum_{i \in \Omega_a} \sum_{j \in \Omega_b} w_{i,j} = 1$. Without loss of generality, we assume that T is mean-centered such that $\sum_{i \in \Omega_a} \sum_{j \in \Omega_b} w_{i,j} T_{i,j} = 0$. For clarity, we also use the shorthand:

$$w_{\cdot,j} = \sum_{i \in \Omega_a} w_{i,j} \tag{17a}$$

$$w_{i,\cdot} = \sum_{j \in \Omega_b} w_{i,j} \tag{17b}$$

Without loss of generality, we can assume that iteration t set all of the column-means to zero. Then iteration $t+1$ will set the row means to zero, and:

$$M^{t+1} = \sum_{j \in \Omega_b} w_{\cdot,j} \left| c_j^{t+1} \right| \tag{18a}$$

$$= \sum_{j \in \Omega_b} w_{\cdot,j} \left| \frac{1}{w_{\cdot,j}} \sum_{i \in \Omega_a} w_{i,j} r_i^t \right| \tag{18b}$$

$$= \sum_{j \in \Omega_b} \left| \sum_{i \in \Omega_a} w_{i,j} r_i^t \right| \tag{18c}$$

$$= \sum_{j \in \Omega_b} \left| \sum_{i \in \Omega_a} w_{i,j} \frac{1}{w_{i,\cdot}} \big(\sum_{k \in \Omega_b} w_{i,k} c_k^{t-1} \big) \right| \tag{18d}$$

$$= \sum_{j \in \Omega_b} \min_{\phi_j} \left| \sum_{i \in \Omega_a} (w_{i,j} - \phi_j w_{i,\cdot}) \frac{1}{w_{i,\cdot}} \big(\sum_{k \in \Omega_b} w_{i,k} c_k^{t-1} \big) \right| \tag{18e}$$

$$= \sum_{j \in \Omega_b} \min_{\phi_j} \left| \sum_{i \in \Omega_a} (\frac{w_{i,j}}{w_{i,\cdot}} - \phi_j) \big(\sum_{k \in \Omega_b} w_{i,k} c_k^{t-1} \big) \right| \tag{18f}$$

$$= \sum_{j \in \Omega_b} \min_{\phi_j} \left| \sum_{i \in \Omega_a} (\frac{w_{i,j}}{w_{i,\cdot}} - \phi_j) \big(\min_{\psi_i} \sum_{k \in \Omega_b} (w_{i,k} - \psi_i w_{\cdot,k}) c_k^{t-1} \big) \right| \tag{18g}$$

where (18e) holds because $\sum_{i \in \Omega_a} \sum_{k \in \Omega_b} w_{i,k} c_k^{t-1}$ is the overall mean of the matrix, which is zero. □

C.2 Proof of Theorem 1

Proof. Under the assumptions of Lemma 3,

$$
\begin{aligned}
M^{t+1} &\leq \sum_{j\in\Omega_b} \min_{\phi_j} \left| \sum_{i\in\Omega_a} (\frac{w_{i,j}}{w_{i,\cdot}} - \phi_j)\big(\min_{\psi_i} \sum_{k\in\Omega_b} (w_{i,k} - \psi_i w_{\cdot,k}) c_k^{t-1}\big) \right| && \text{by Lemma 3} \quad (19a)\\
&= \sum_{j\in\Omega_b} \left| \sum_{i\in\Omega_a} (\frac{w_{i,j}}{w_{i,\cdot}} - \frac{1}{n})\big(\min_{\psi_i} \sum_{k\in\Omega_b} (w_{i,k} - \psi_i w_{\cdot,k}) c_k^{t-1}\big) \right| && (19b)\\
&= \sum_{j\in\Omega_b} \left| \sum_{i\in\Omega_a} (0)\big(\min_{\psi_i} \sum_{k\in\Omega_b} (w_{i,k} - \psi_i w_{\cdot,k}) c_k^{t-1}\big) \right| && \text{for uniform } w \quad (19c)\\
&= 0 && (19d)
\end{aligned}
$$

□

C.3 Proof of Theorem 2

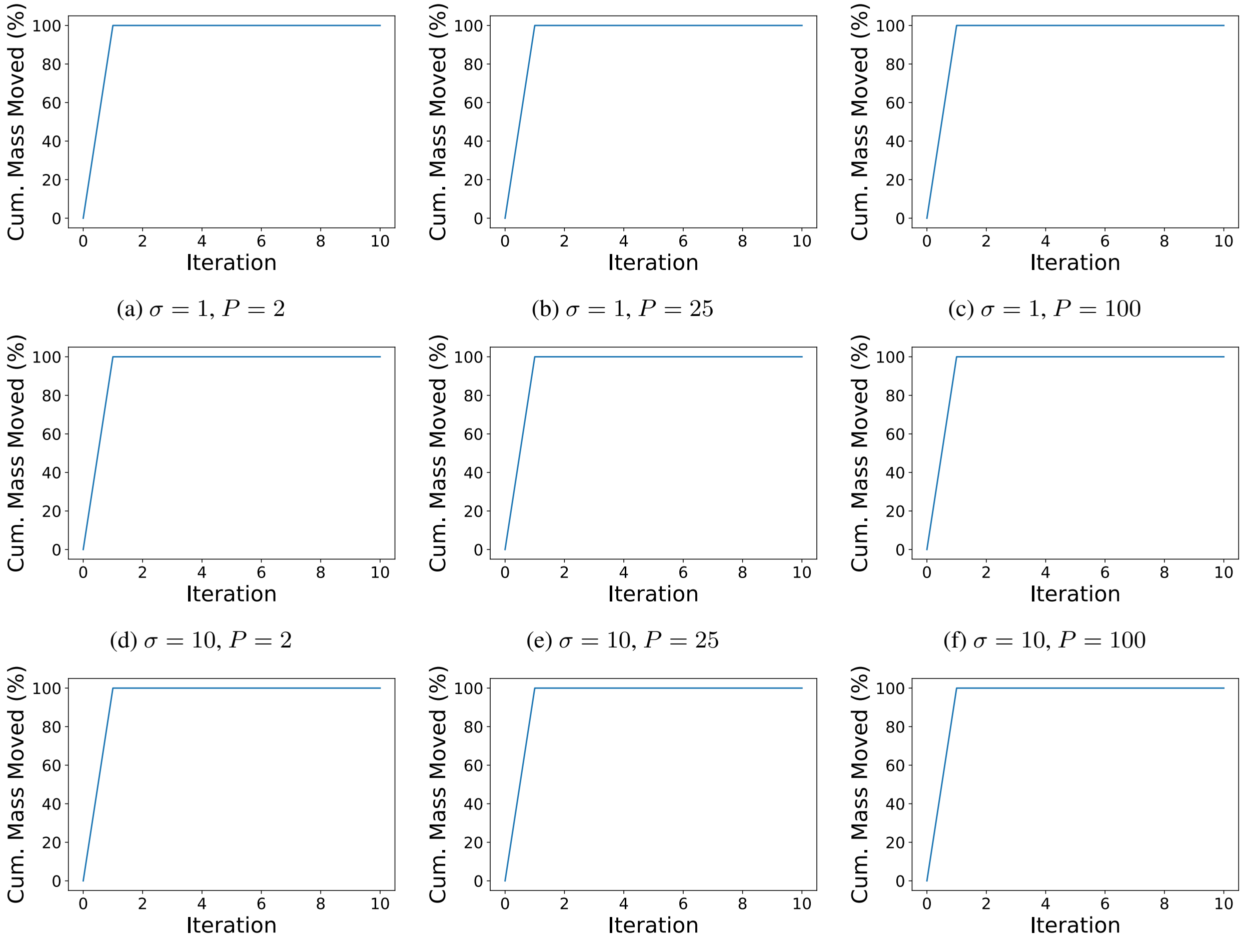
Proof. For any normalized w,

$$
\begin{aligned}
M^{t+1} &= \sum_{j\in\Omega_b} \min_{\phi_j} \left| \sum_{i\in\Omega_a} (\frac{w_{i,j}}{w_{i,\cdot}} - \phi_j)\big(\min_{\psi_{i,j}} \sum_{k\in\Omega_b} (w_{i,k} - \psi_{i,j} w_{\cdot,k}) c_k^{t-1}\big) \right| && \text{by Lemma 3} \quad (20a)\\
&= \sum_{j\in\Omega_b} \min_{\phi_j} \left| \sum_{i\in\Omega_a} (\frac{w_{i,j}}{w_{i,\cdot}} - \phi_j) \min_{\psi_{i,j}} w_{i,\cdot} \sum_{k\in\Omega_b} (\frac{w_{i,k}}{w_{i,\cdot}} - \frac{\psi_{i,j}}{w_{i,\cdot}} w_{\cdot,k}) c_k^{t-1} \right| && (20b)\\
&= \sum_{j\in\Omega_b} \min_{\phi_j} \left| \sum_{i\in\Omega_a} (\frac{w_{i,j}}{w_{i,\cdot}} - \phi_j) \sum_{k\in\Omega_b} \sum_{l\neq i} w_{l,k} c_k^{t-1} \right| && (20c)\\
&\leq \sum_{j\in\Omega_b} \min_{\phi_j} \left| \sum_{i\in\Omega_a} (\frac{w_{i,j}}{w_{i,\cdot}} - \phi_j) \sum_{k\in\Omega_b} w_{\cdot,k} c_k^{t-1} \right| && (20d)\\
&\leq \sum_{j\in\Omega_b} \min_{\phi_j} \left| \sum_{i\in\Omega_a} (\frac{w_{i,j}}{w_{i,\cdot}} - \phi_j) \sum_{k\in K^+} w_{\cdot,k} c_k^{t-1} \right| && K^+ = \{k \in \Omega_b : c_k^{t-1} > 0\} \quad (20e)\\
&\leq \sum_{j\in\Omega_b} \min_{\phi_j} \left| \sum_{i\in\Omega_a} (\frac{w_{i,j}}{w_{i,\cdot}} - \phi_j) \frac{1}{2} M^{t-1} \right| && (20f)\\
&\leq \frac{1}{2} M^{t-1} \sum_{j\in\Omega_b} \min_{\phi_j} \left| \sum_{i\in\Omega_a} (\frac{w_{i,j}}{w_{i,\cdot}} - \phi_j) \right| && (20g)\\
&\leq \frac{1}{2} M^{t-1} \sum_{i\in\Omega_a} \sum_{j\in\Omega_b} \min_{\phi_j} \left| \frac{w_{i,j}}{w_{i,\cdot}} - \phi_j \right| && (20h)\\
&\leq \frac{1}{2} M^{t-1} \sum_{i\in\Omega_a} w_{i,\cdot} && (20i)\\
&= \frac{1}{2} M^{t-1} && (20j)
\end{aligned}
$$

So the divergence from the fANOVA decomposition is cut in half each iteration. This is a loose bound which could be tightened by examining the dispersion of the density. □

D Empirically Measuring Purification Convergence

Purification by the mass-moving algorithm converges in a very small number of iterations. We examine this behavior empirically by generating tensors $T \sim N(0, \sigma I)$ with weight values either: (1) uniform distribution: $w \propto 1$ or (2) drawn from a multivariate normal distribution: $w \sim N(0, \sigma I)$ of dimension P. Results for a variety of settings of σ and P are shown in Figures 9 and 10. In all cases, we see that the mass-moving algorithm moves almost all of the mass in the first iteration. In the case of the uniform weight distribution, we confirm that the mass-moving algorithm takes only a single iteration (per row/column) to convergence. These results show that the mass-moving algorithm can scale to purify large models.



(a) $\sigma = 1$, $P = 2$

(b) $\sigma = 1$, $P = 25$

(c) $\sigma = 1$, $P = 100$

(d) $\sigma = 10$, $P = 2$

(e) $\sigma = 10$, $P = 25$

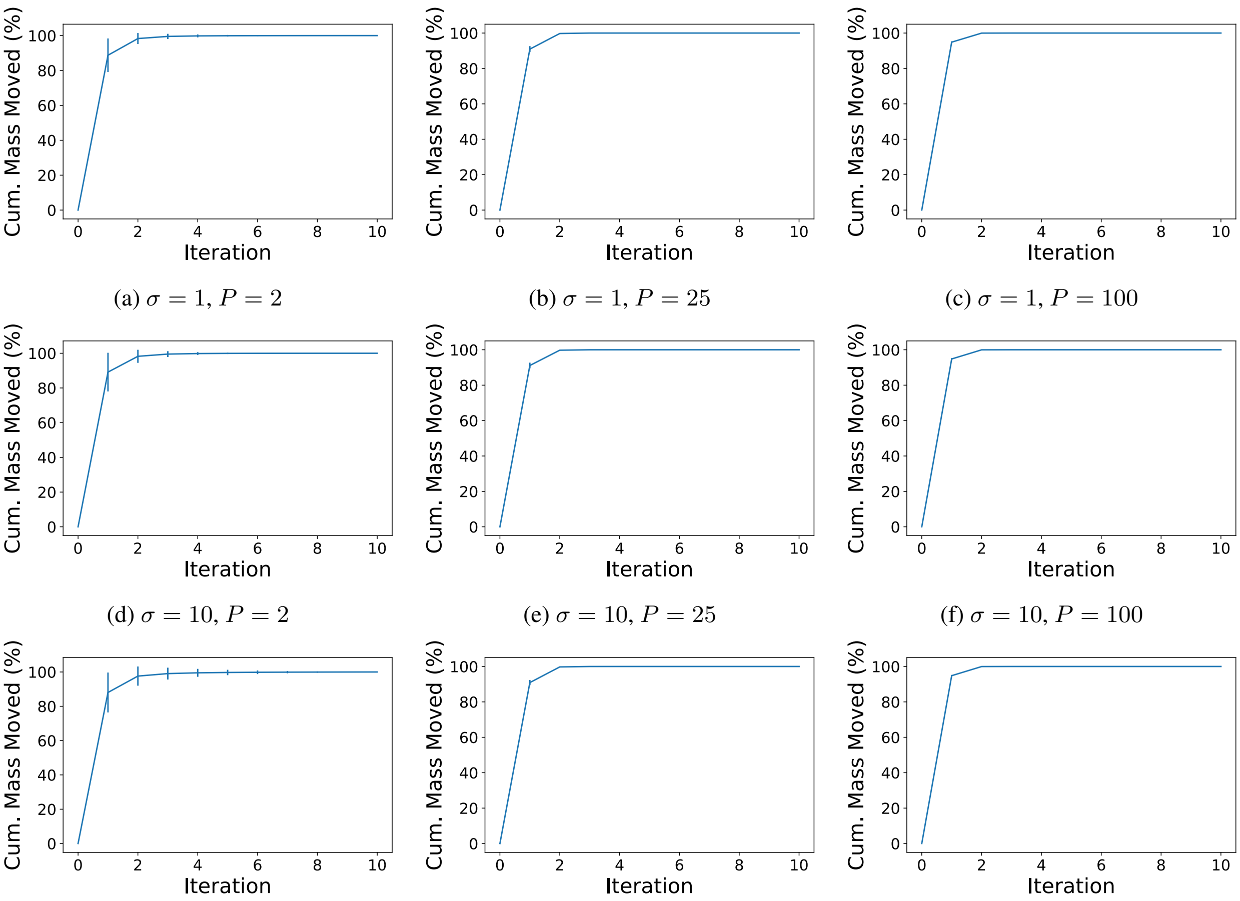
(f) $\sigma = 10$, $P = 100$

(g) $\sigma = 100$, $P = 2$

(h) $\sigma = 100$, $P = 25$

(i) $\sigma = 100$, $P = 100$

Figure 9: Convergence of the mass-moving algorithm under uniform weight distributions. In all settings, the unpurified effect matrix is drawn from $N(0, \sigma I)$ of dimension P while the weighting is uniform. In all settings, the algorithm converges in a single iteration.



(a) $\sigma = 1$, $P = 2$

(b) $\sigma = 1$, $P = 25$

(c) $\sigma = 1$, $P = 100$

(d) $\sigma = 10$, $P = 2$

(e) $\sigma = 10$, $P = 25$

(f) $\sigma = 10$, $P = 100$

(g) $\sigma = 100$, $P = 2$

(h) $\sigma = 100$, $P = 25$

(i) $\sigma = 100$, $P = 100$

Figure 10: Convergence of the mass-moving algorithm under random weight distributions. In all settings, In all settings, the unpurified effect matrix is drawn from $N(0, \sigma I)$ of dimension P while the weighting is drawn from a P-dimensional normal distribution $N(0, \sigma I)$. Errorbars indicate the mean ± stddev. over 100 experiments. In all settings, the algorithm converges in a small number of iterations.

Method	XOR	OR	AND	Mean	% Improvement
GAM	60 ± 4	72 ± 4	$\mathbf{60 \pm 5}$	64 ± 4	–
GA2M-Purified	36 ± 5	$\mathbf{64 \pm 5}$	63 ± 10	54 ± 7	15.6
XGB	15 ± 9	132 ± 10	256 ± 6	134 ± 8	–
XGB2-Purified	$\mathbf{1 \pm 1}$	116 ± 8	114 ± 22	77 ± 10	42.5

Table 3: Total squared error of the recovery of pure main effects from simulation data, where the outcome is generated according to the column title. Values displayed are the (mean ± std) over five runs.

E Recovery of Main Effects from Simulation Data

We evaluate the effectiveness of our purification algorithm to generate the fANOVA decomposition by comparing recovery error of main effects from simulation data. We test the following continuous versions of the bitwise logical operators:

- $AND(X_1, X_2) = \min(X_1, X_2)$
- $OR(X_1, X_2) = \max(X_1, X_2)$
- $XOR(X_1, X_2) = -AND(X_1, X_2) + 0.5X_1 + 0.5X_2$

where X_1, X_2 are $N(0, 1)$. In addition to these two variables, we generate P nuisance variables with linear main effects. Mean squared error of the implied main effects are shown in Table 3 for $P = 5$ and 100 training samples and purification with respect to the Laplace-smoothed empirical distribution. The reduced squared error of the purified models indicates that this algorithm recovers the correct decomposition, even surpassing additive models without interactions in some cases.

F COMPAS

Shown in Fig. 11 are main effects effects implied by purification of models to predict the ground-truth recidivism label in the COMPAS dataset.

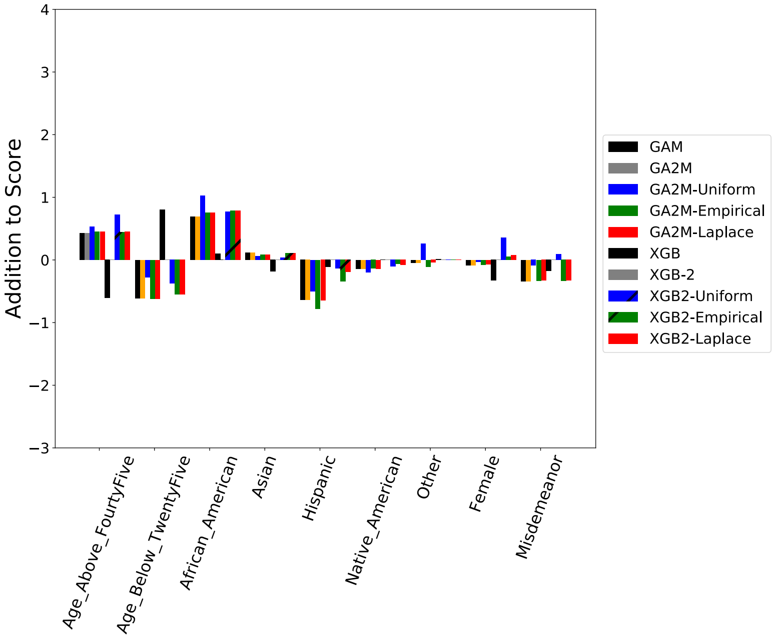
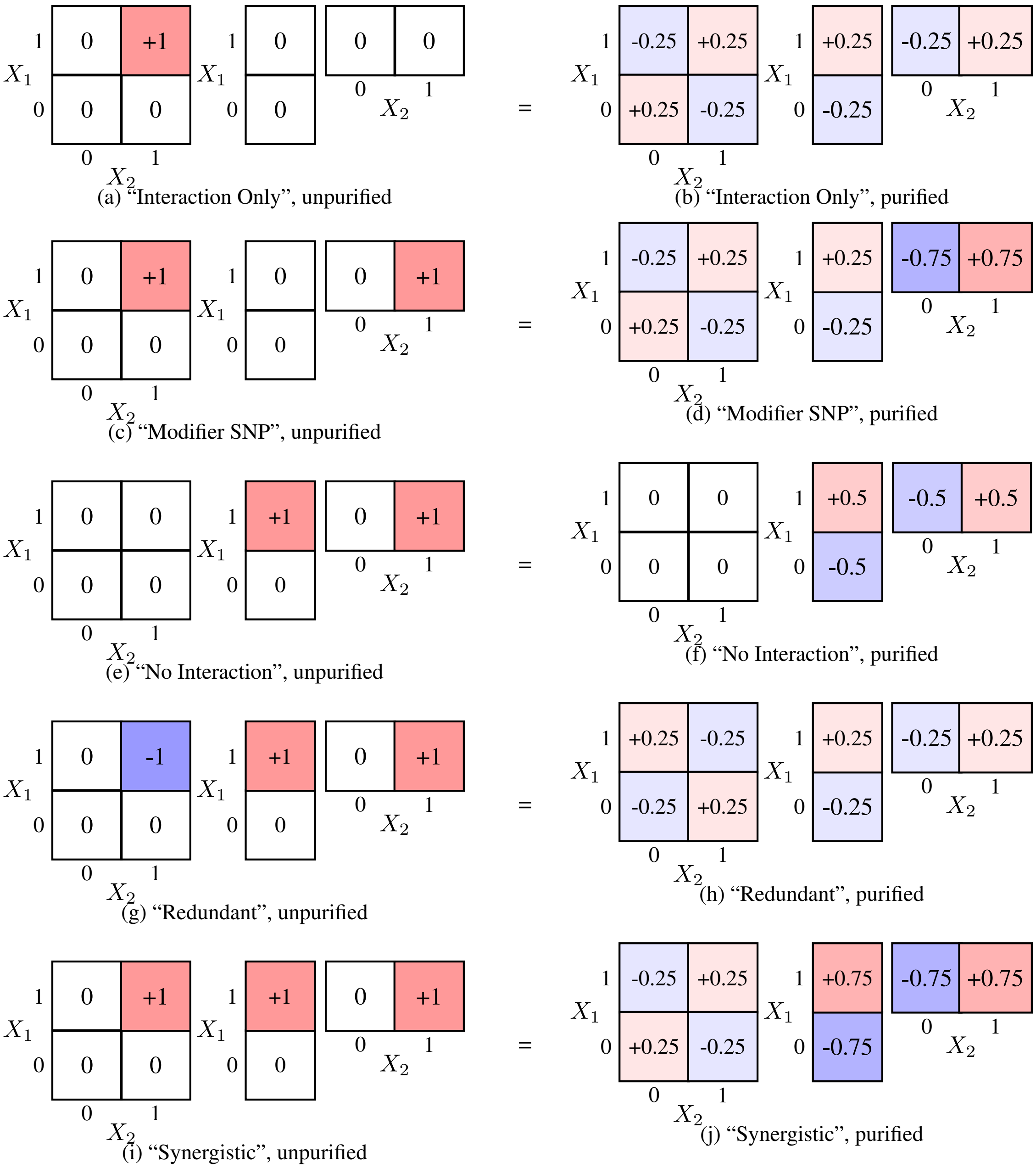


Figure 11: Main effects of additive models with interactions trained on the COMPAS dataset for ground-truth recidivism. We see that the implications of the main effects depends on both the model trained and the distribution used for purification.

G Interactions Studied in Wright, Ziegler, and König 2016

The data generators studied in Wright, Ziegler, and König 2016 are depicted in Fig. 12. Please refer to Section 7.1 of the main text for a discussion of the implications of these models.



(a) “Interaction Only”, unpurified
(b) “Interaction Only”, purified
(c) “Modifier SNP”, unpurified
(d) “Modifier SNP”, purified
(e) “No Interaction”, unpurified
(f) “No Interaction”, purified
(g) “Redundant”, unpurified
(h) “Redundant”, purified
(i) “Synergistic”, unpurified
(j) “Synergistic”, purified

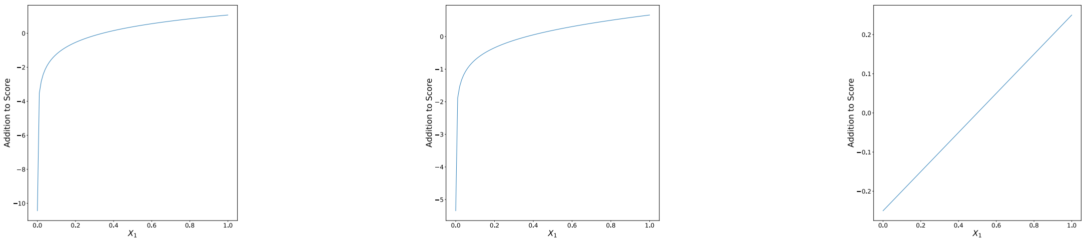
Figure 12: Several models of genetic interaction effects. (Left) Representations presented in Wright, Ziegler, and König (2016) (Right) Representations purified according to a uniform distribution. For visual clarity, we have removed the overall intercept.

H The Mystery of $\log(X_1, X_2)$

To test that our mass-moving procedure recovers the correct decomposition, we generate data according to the model:

$$Y = (1 - \lambda)\log(X_1 X_2) + \lambda(X_1 X_2), \tag{21}$$

which allows us to control how much the function applied to X_1X_2 behaves like $\log()$ for $\lambda \approx 0$ vs. multiplicative interaction for $\lambda \approx 1$. By varying $\lambda \in [0, 1]$, we can examine the ability of Alg. 2 to distinguish these effects.



(a) $\lambda = 0.0$ (b) $\lambda = 0.5$ (c) $\lambda = 1.0$

Figure 13: Pure Main Effects of the log model (21). The purification algorithm recovers the transition from logarithmic to linear main effects.

As shown in Fig. 13, the main effects recovered at $\lambda = 0$ are logarithmic, and the interaction effect completely disappears. As λ increases, the main effects transition toward linearity, and the interaction effect returns, becoming the continuous variant of XOR. In fact, for any $\lambda \neq 0$, the pattern of the pure interaction effect is the same, and the overall strength of the interaction effect increases as $\lambda \to 1$.[4]

[4] The base of the logarithm does not alter the shape of the interaction, but does affect how rapidly C varies with λ.