方差可排序性研究背景与意义

□ MSE最小化损失函数导致估计的因果方向受数据测量单位和数据缩放（标准化）的影响：

True SCM：

$$\begin{cases} \text{causal graph } A \to B \\ A = N_A \\ B = wA + N_B \quad w \neq 0 \\ \text{independent zero-centered noise variables } N_A, N_B \end{cases}$$

双变量情况

MSE loss最小化推断出的因果方向为：小方差变量→大方差变量

$$\mathrm{MSE}\,(A \to B) < \mathrm{MSE}\,(B \to A) \iff \mathrm{Var}(A) < \mathrm{Var}(B)$$

只需要调整变量方差（改变数据测量单位和对数据缩放），就能控制因果发现结果，严重影响可微框架在真实数据上的CD结果。

数据底层因果方向与变量边际方差递增顺序之间的一致性会决定可微因果发现性能

[Reisach A, Seiler C, Weichwald S. Beware of the simulated dag! causal discovery benchmarks may be easy to game[J]. Advances in Neural Information Processing Systems, 2021, 34: 27772-27784.]

方差可排序性

可变排序性（$varsortability, v$）定义：

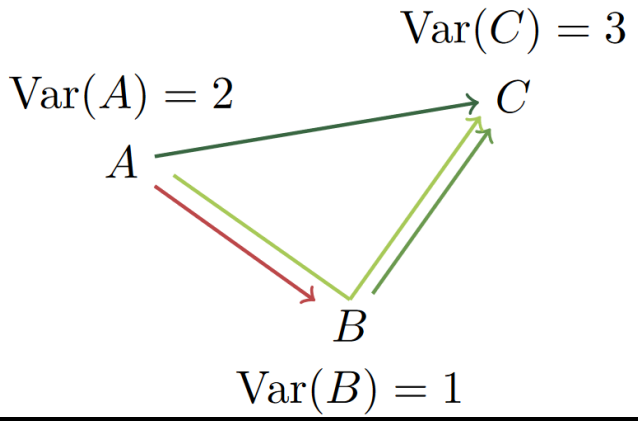
$$v := \frac{\sum_{k=1}^{d-1} \sum_{i \to j \in E^k} \text{increasing}(\text{Var}(X_i), \text{Var}(X_j))}{\sum_{k=1}^{d-1} \sum_{i \to j \in E^k} 1} \in [0, 1]$$

$$\text{where increasing}(a, b) = \begin{cases} 1 & a < b \\ 1/2 & a = b \\ 0 & a > b \end{cases}$$

可变排序性大小用于衡量因果结构顺序与节点边际方差递增顺序之间的一致性。

直觉上，数据的可变排序性越大（小），可微框架效果越好（差）。

- **例子：**



$$v = \frac{1+1+1}{1+1+1+1} = \frac{3}{4}$$

[Reisach A, Seiler C, Weichwald S. Beware of the simulated dag! causal discovery benchmarks may be easy to game[J]. Advances in Neural Information Processing Systems, 2021, 34: 27772-27784.]

合成数据上的可变排序性

线性加性噪声模型：

- **实验配置：**每种图和噪声类型的组合均在50个节点的图上进行10次实验，每次采样1000个样本，统计10次实验的min、mean和max可变排序性。
- **实验结论：**常见的线性ANM实验配置均有高可排序性，这是可微框架性能好的原因。

graph	noise	varsortability min	varsortability mean	varsortability max
ER-1	Gauss-EV	0.94	0.97	0.99
	exponential	0.94	0.97	0.99
	gumbel	0.94	0.97	1.00
ER-2	Gauss-EV	0.97	0.99	1.00
	exponential	0.97	0.99	1.00
	gumbel	0.98	0.99	0.99
ER-4	Gauss-EV	0.98	0.99	0.99
	exponential	0.98	0.99	0.99
	gumbel	0.98	0.99	0.99
SF-4	Gauss-EV	0.98	1.00	1.00
	exponential	0.98	1.00	1.00
	gumbel	0.98	1.00	1.00

[Reisach A, Seiler C, Weichwald S. Beware of the simulated dag! causal discovery benchmarks may be easy to game[J]. Advances in Neural Information Processing Systems, 2021, 34: 27772-27784.]

合成数据上的可变排序性

□ 非线性加性噪声模型：

- **实验配置：** 每种图和噪声类型的组合均在20个节点的图上进行10次实验，每次采样1000个样本，统计10次实验的min、mean和max可变排序性。
- **实验结论：** 常见的非线性ANM实验配置均有高可排序性，这是可微框架性能好的原因。

graph	ANM-type	varsortability min	varsortability mean	varsortability max
ER-1	Additive GP	0.81	0.91	1.00
	GP	0.72	0.86	0.96
	MLP	0.55	0.79	0.96
	Multi Index Model	0.62	0.82	1.00
ER-2	Additive GP	0.79	0.91	0.98
	GP	0.82	0.89	0.97
	MLP	0.46	0.71	0.87
	Multi Index Model	0.65	0.79	0.89
ER-4	Additive GP	0.90	0.95	0.98
	GP	0.74	0.88	0.93
	MLP	0.59	0.72	0.85
	Multi Index Model	0.57	0.73	0.85
SF-4	Additive GP	0.95	0.97	0.99
	GP	0.88	0.94	0.97
	MLP	0.75	0.83	0.93
	Multi Index Model	0.77	0.84	0.97

[Reisach A, Seiler C, Weichwald S. Beware of the simulated dag! causal discovery benchmarks may be easy to game[J]. Advances in Neural Information Processing Systems, 2021, 34: 27772-27784.]

排序回归算法

排序回归算法：只利用数据中的方差来进行因果发现

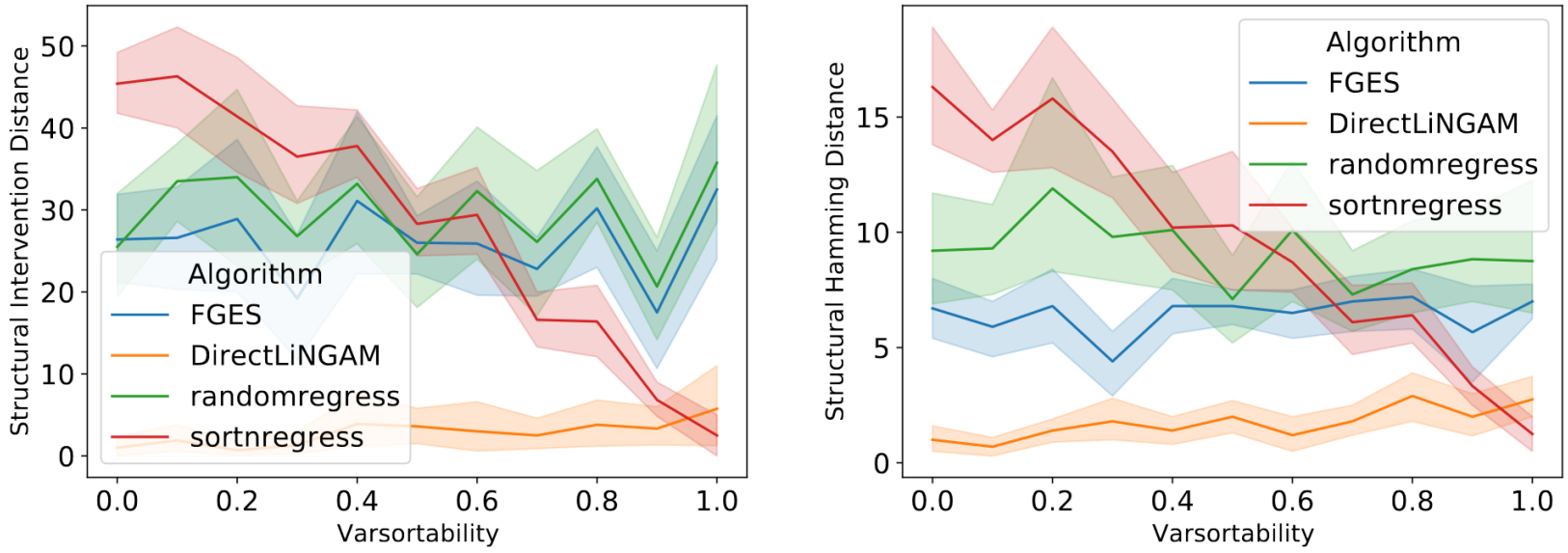
- **第一步，排序搜索：**按照节点边际方差递增的顺序对节点排序
- **第二步，父节点搜索：**使用线性回归+Lasso回归（BIC信息准则进行模型选择）

```
def sortnregress(X):
    """ Take n x d data, order nodes by marginal variance and
    regresses each node onto those with lower variance, using
    edge coefficients as structure estimates. """
    LR = LinearRegression()
    LL = LassoLarsIC(criterion='bic')

    d = X.shape[1]
    W = np.zeros((d, d))
    increasing = np.argsort(np.var(X, axis=0))

    for k in range(1, d):
        covariates = increasing[:k]      #每次取的自变量
        target = increasing[k]           #每次取的因变量

        LR.fit(X[:, covariates], X[:, target].ravel())
        weight = np.abs(LR.coef_)
        LL.fit(X[:, covariates] * weight, X[:, target].ravel())
        W[covariates, target] = LL.coef_ * weight
    return W
```

[Reisach A, Seiler C, Weichwald S. Beware of the simulated dag! causal discovery benchmarks may be easy to game[J]. Advances in Neural Information Processing Systems, 2021, 34: 27772-27784.]

合成数据实验结果

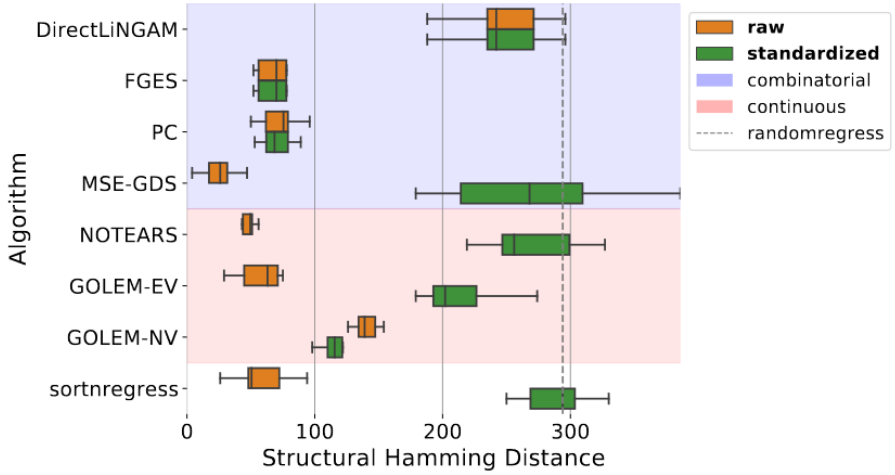
- **实验配置：** 合成数据为线性，10节点ER-1图，边权范围为$(-0.5,-0.1) \cup (0.1,0.5)$，进行10次实验。
- **Baseline方法：** DirectLiNGAM(Shimizu et al., JMLR'11)适用于线性非高斯下的尺度不变因果发现；randomregress随机选取节点顺序，其余步骤与排序回归相同。
- **实验结论：** 可变排序性越大，排序回归算法性能越好。



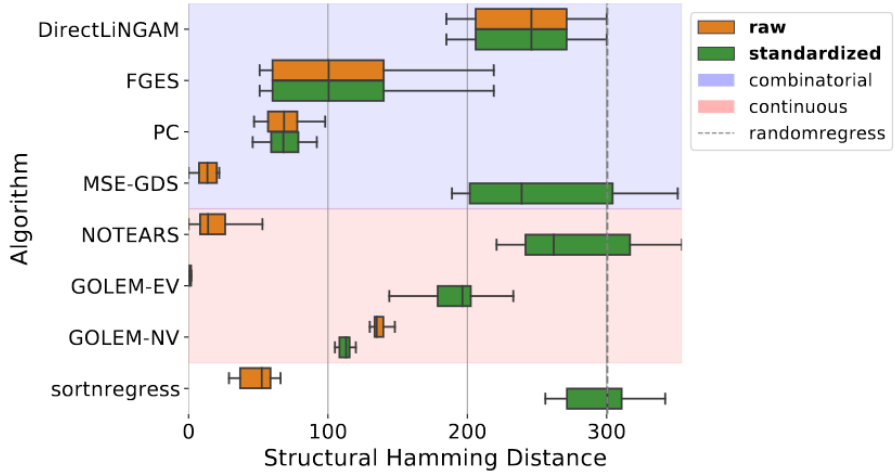
[Reisach A, Seiler C, Weichwald S. Beware of the simulated dag! causal discovery benchmarks may be easy to game[J]. Advances in Neural Information Processing Systems, 2021, 34: 27772-27784.]

合成数据实验结果

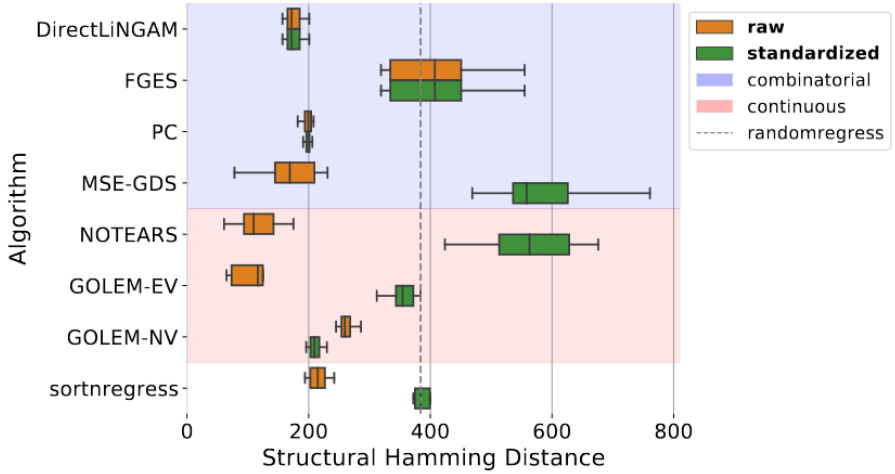
- **实验结论：原始数据与标准化数据的性能差异在各种噪声分布上相似 。**



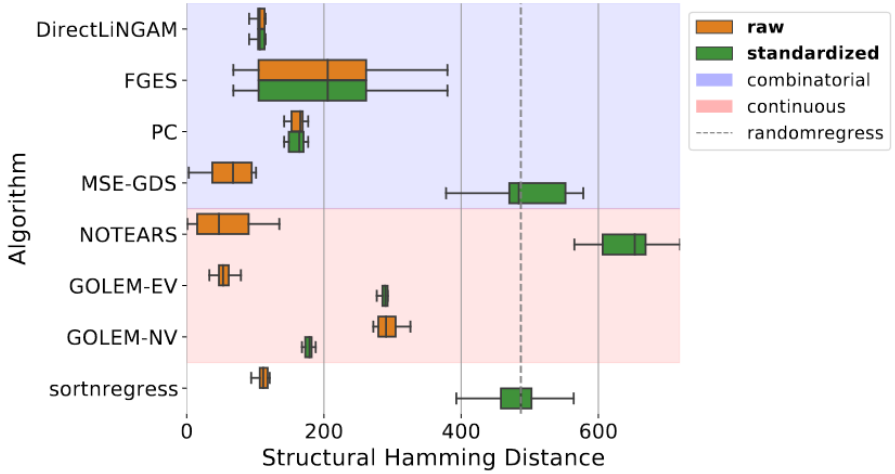
(a) SHD, Gaussian-NV noise, ER-2, 50 nodes



(b) SHD, Gaussian-EV noise, ER-2, 50 nodes



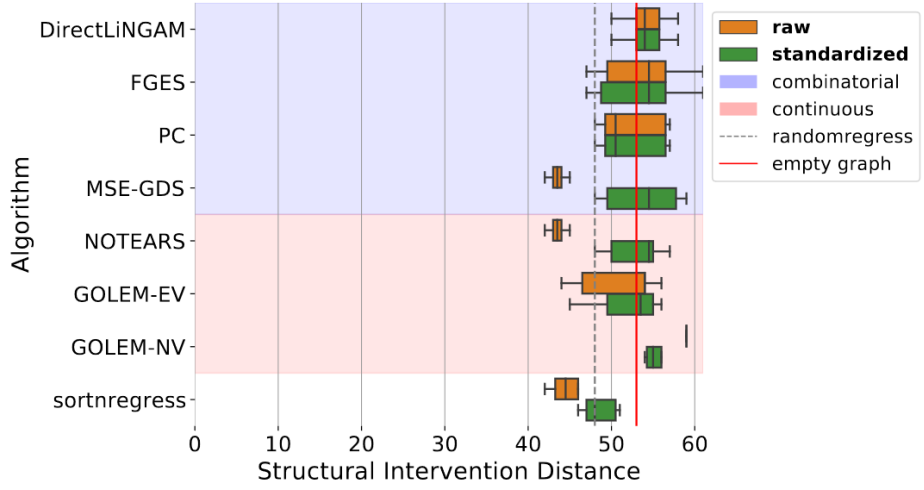
(c) SHD, Exponential Noise, ER-4, 50 nodes

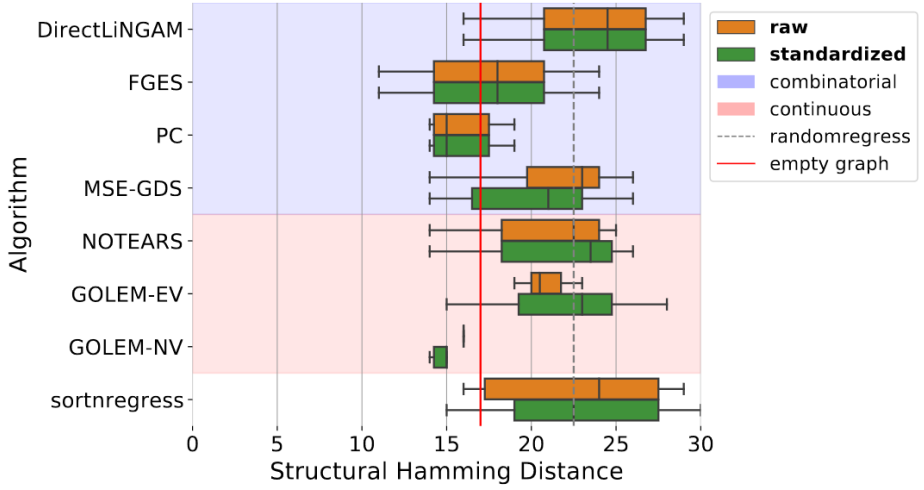


(d) SHD, Gumbel Noise, SF-4, 50 nodes

真实数据实验结果

- **实验配置：** Sachs数据（853个样本，11节点，17条边），可变排序性为0.57。
- **实验结论：** 在可变排序性低的Sachs数据上，各种算法在原始数据与标准化数据上的性能差异较小。





[Reisach A, Seiler C, Weichwald S. Beware of the simulated dag! causal discovery benchmarks may be easy to game[J]. Advances in Neural Information Processing Systems, 2021, 34: 27772-27784.]