1 SUPPLEMENTAL MATERIAL

1.1 Proofs

Theorem 1. *Under C1, object conditioning on P produces exchangeability among values of X and among values of Y.*

Proof. For a given conditioning object P' with k child objects, $p(x_1, y_1, x_2, y_2, \ldots, x_k, y_k)$

$$= \int_z \prod_{i=1}^{k} p_{x,y|z}(x_i, y_i|z) p_z(z)\, dz$$
$$= \int_z \prod_{i=1}^{k} p_{x|z}(x_i|z) p_{y|z}(y_i|z) p_z(z)\, dz$$
$$= \int_z \left(\prod_{i=1}^{k} p_{x|z}(x_i|z)\right) \left(\prod_{i=1}^{k} p_{y|z}(y_i|z)\right) p_z(z)\, dz$$
$$= \int_z \left(\prod_{i=1}^{k} p_{x|z}(x_i|z)\right) \left(\prod_{i=1}^{k} p_{y|z}(y_{\pi_y(i)}|z)\right) p(z)\, dz$$
$$= \int_z \left(\prod_{i=1}^{k} p_{x|z}(x_{\pi_x(i)}|z)\right) \left(\prod_{i=1}^{k} p_{y|z}(y_{\pi_y(i)}|z)\right) p(z)\, dz$$

In which the last statement proves exchangeability. That is, no x_i provides special information about the value of the corresponding y_i, thus values of X are exchangeable and values of Y are exchangeable. However, any value x_i provides information about z which, in turn, provides information about y_j, thus X and Y are not conditionally independent given the object P. In contrast, X and Y *are* conditionally independent given the variable Z. ☐

Next, consider a version of the generative process in which Z is a *mediator* (abbreviated as C2 and illustrated in Figure **??**b). We prove that object conditioning on P produces exchangeability among values of X and among values of Y within the set of child objects of each instance of P.

Theorem 2. *Under C2, object conditioning on P produces exchangeability among values of X and among values of Y.*

Proof. For a given conditioning object P' with k child objects, the k values of X are i.i.d. by definition, and thus exchangeable. To prove that values of Y are exchangeable, we can proceed as in Theorem 1. This time, we assume that the x_i are i.i.d. and we have a permutation $\pi_y(\cdot)$. Then, $p(x_1, y_1, x_2, y_2, \ldots, x_k, y_k)$

$$= \left(\prod_{i=1}^{k} p_x(x_i)\right) \int_z \prod_{i=1}^{k} p_{y|z}(y_i|z) p_{z|\vec{x}}(z|x_1, \ldots, x_k)\, dz$$
$$= \left(\prod_{i=1}^{k} p_x(x_i)\right) \int_z \prod_{i=1}^{k} p_{y|z}(y_{\pi_y(i)}|z) p_{z|\vec{x}}(z|x_1, \ldots, x_k)\, dz$$

Which again shows the exchangeability property. Again, any value x_i provides information about z which, in turn, provides information about y_j, thus X and Y are not conditionally independent given the object P. In contrast, X and Y *are* conditionally independent given the variable Z. ☐

Finally, we consider a version of the generative process in which Z is a *collider* (abbreviated as C3 and illustrated in Figure **??**c). Once again, we prove that object conditioning on P produces exchangeability among values of X and among values of Y within the set of child objects of each instance of P.

Theorem 3. *Under C3, object conditioning on P produces exchangeability among values of X and among values of Y.*

Proof. Given the generative model, $p(x_1, y_1, x_2, y_2, \ldots, x_k, y_k)$

$$= \left(\prod_{i=1}^{k} p_{x,y}(x_i, y_i)\right)$$
$$= \left(\prod_{i=1}^{k} p_x(x_i) p_y(y_i)\right)$$
$$= \left(\prod_{i=1}^{k} p_x(x_i)\right) \left(\prod_{i=1}^{k} p_y(y_i)\right)$$
$$= \left(\prod_{i=1}^{k} p_x(x_{\pi_x(i)})\right) \left(\prod_{i=1}^{k} p_y(y_{\pi_y(i)})\right)$$

Which again shows exchangeability and is a direct consequence of both X and Y being i.i.d. Here, X and Y are marginally independent and remain independent even when conditioned on the object P. In contrast, X and Y are *not* conditionally independent given the variable Z. ☐

1.2 Relative prevalence of literature on different methods

This analysis used Google Scholar to estimate the approximate number of articles corresponding to specific boolean queries. The queries and number of results are shown below.

Table 1: Number of returned query results in Google Scholar for different causal modeling methods

Query	Number of results
”within subjects design”	40,300
”multilevel model” AND ”regression”	33,500
”interrupted time series”	27,100
”difference-in-differences”	32,400
(DAG OR ”directed graphical model”) AND (causal OR causality)	54,700

1.3 Grounding example

For readers unfamiliar with groundings of plate models, Figure 1 provides three examples of how a plate model is “grounded” (produces instances corresponding to its specified structure).

1.4 Pseudo-code for generative processes

Below are pseudo-code versions of example generative processes to further clarify the cases examined in section **??**, including a case in which X and Y are causally unrelated (C4).

```
#C1
def generate_subgraphs_with_confounder(N):
    subgraphs = make_vector(N)
    for i in range(N):
            k = uniform(5,15)
            s = make_child(k)
            s.a.z = poisson(θ_z)
            for j in range(k):
                    s.b[j].x = poisson(s.a.z)
                    s.b[j].y = poisson(s.a.z)
            subgraphs[i] = s
    return subgraphs
```

```
#C2
def generate_subgraphs_with_mediator(N):
    subgraphs = make_vector(N)
    for i in range(N):
            k = uniform(5,15)
            s = make_struct(k)
            for j in range(k):
                    s.b[j].x = poisson(θ_x)
            s.a.z = poisson(2 * sum(s.b.x))
            for j in range(k):
                    s.b[j].y = poisson(s.a.z)
            subgraphs[i] = s
    return subgraphs
```

```
#C3
def generate_subgraphs_with_collider(N):
    subgraphs = make_vector(N)
    for i in range(N):
            k = uniform(5,15)
            s = make_struct(k)
            for j in range(k):
                    s.b[j].x = poisson(θ_x)
                    s.b[j].y = poisson(θ_y)
            s.a.z = poisson(sum(s.b.x * s.b.y))
            subgraphs[i] = s
    return subgraphs
```

```
#C4
def generate_subgraphs_with_independence(N):
    subgraphs = make_vector(N)
    for i in range(N):
            k = uniform(5,15)
            s = make_struct(k)
            for j in range(k):
                    s.b[j].x = poisson(θ_x)
                    s.b[j].y = poisson(θ_y)
            subgraphs[i] = s
    return subgraphs
```

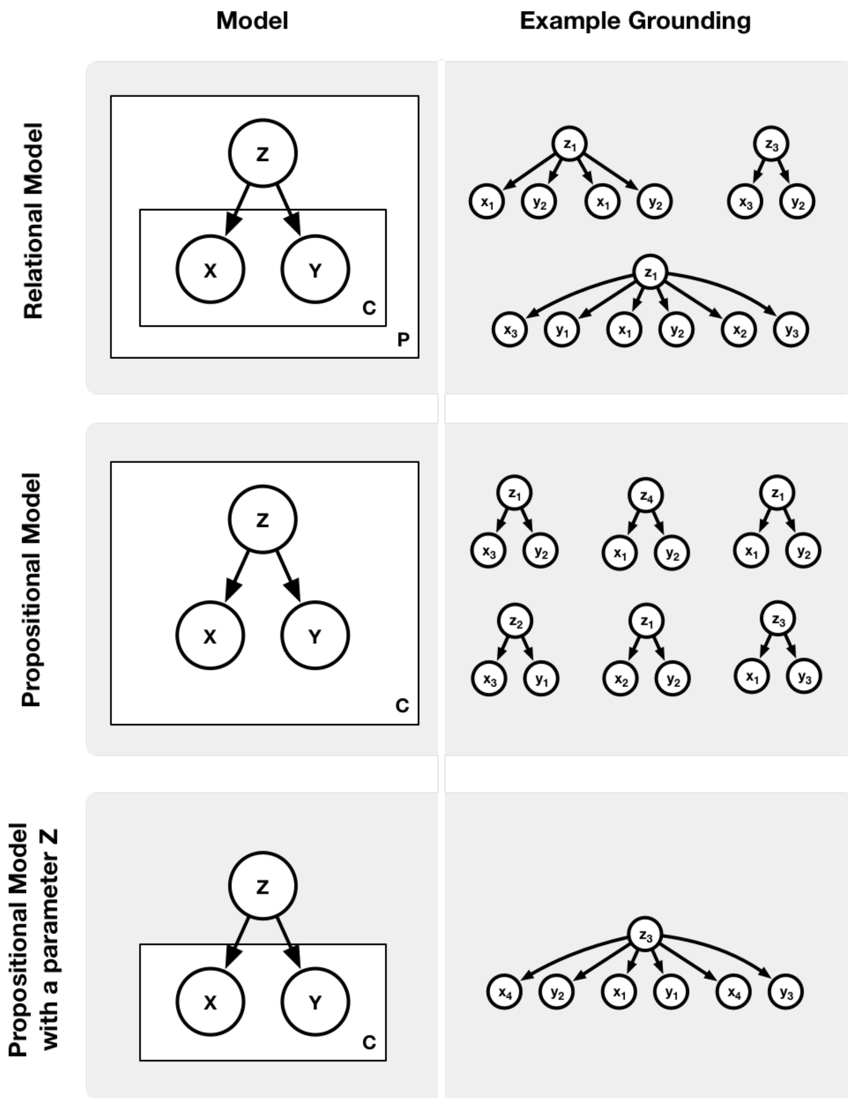

Figure 1: Example groundings of three different plate models