

# Simulating the Impact of Interventions

By simulating the impact of interventions, we answer the questions such as:

What will happen to the variable Z if I intervene on Y?

## How to use it

To see how the method works, let's generate some data:

```
>>> import numpy as np, pandas as pd
```

```
>>> X = np.random.normal(loc=0, scale=1, size=1000)
>>> Y = 2*X + np.random.normal(loc=0, scale=1, size=1000)
>>> Z = 3*Y + np.random.normal(loc=0, scale=1, size=1000)
>>> training_data = pd.DataFrame(data=dict(X=X, Y=Y, Z=Z))
```

Next, we'll model cause-effect relationships as a probabilistic causal model and fit it to the data:

```
>>> import networkx as nx
>>> from dowhy import gcm
```

```
>>> causal_model = gcm.ProbabilisticCausalModel(nx.DiGraph([('X', 'Y'), ('Y', 'Z')])
>>> gcm.auto.assign_causal_mechanisms(causal_model, training_data)
```

```
>>> gcm.fit(causal_model, training_data)
```

Finally, let's perform an intervention on X. Here, we explicitly perform the intervention  $do(X := 1)$

[Skip to main content](#)

```
>>> samples = gcm.interventional_samples(causal_model,
>>>                                     {'X': lambda x: 1},
>>>                                     num_samples_to_draw=1000)
>>> samples.head()
```

	X	Y	Z
0	1	3.481467	12.475105
1	1	1.282945	3.279435
2	1	2.508717	7.907412
3	1	2.077061	5.506252
4	1	1.400568	6.097633

As we can see, X is now fixed at a constant value of 1. This is known as an atomic intervention. We can also perform shift interventions where we shift the random variable X by some value:

```
>>> samples = gcm.interventional_samples(causal_model,
>>>                                     {'X': lambda x: x + 0.5},
>>>                                     num_samples_to_draw=1000)
>>> samples.head()
```

	X	Y	Z
0	-0.542813	0.031771	1.195391
1	1.615089	2.156833	6.704683
2	1.340949	1.910316	5.882468
3	1.837919	4.360685	12.565738
4	3.791410	8.361918	25.477725

## Related example notebooks

- [Basic Example for Graphical Causal Models](#)
- [Impact of 401\(k\) eligibility on net financial assets](#)
- [Finding the Root Cause of Elevated Latencies in a Microservice Architecture](#)

Previous

< [Asking and Answering What-If Questions](#)

Next

[Computing Counterfactuals](#) >

© Copyright 2022, PyWhy contributors.

Created using [Sphinx](#) 7.1.2.

Built with the [PyData Sphinx Theme](#) 0.14.4.