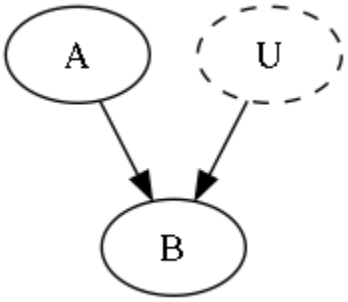
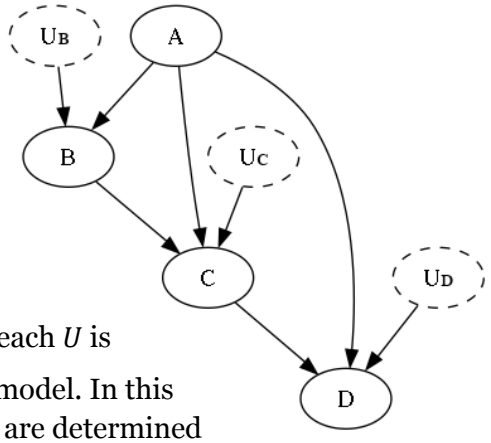
# 1: Simulated Data

```
# synthetic training data generation
method = 'linear'   # linear or nonlinear structural assignment
sem_type = 'gauss'  # noise distribution: gauss, exp, unif, gumbel (for linear)
n_nodes = 10
n_edges = 15
n = 2000  # number of observations in training data
weighted_random_dag = DAG.erdos_renyi(n_nodes=n_nodes, n_edges=n_edges,
                                      weight_range=(0.5, 2.0), seed=1)
dataset = IIDSimulation(W=weighted_random_dag, n=n, method=method,
                        sem_type=sem_type)
ground_truth_adj_matrix, training_data = dataset.B, dataset.X
```

gCastle uses the Erdős–Rényi model $G(n, p)$ for generating random directed acyclic graphs, and Structural Equation Modeling (SEM) to simulate training data for learning causal relationships. Structural Equation Models involve the modeling of multivariate causal relationships between both observable and latent (unobserved) variables, and are popular in many applied fields (Psychology, Economics, etc.). SEMs are similar but distinct from Pearl's [Structural] Causal Models (SCMs) in that SCMs are explicitly for causal inference (via interventions, accounting for confounders, etc.), while SEMs are focused more on model fit and exploring relationships among variables.




Structured Causal Models: A structural equation is denoted with the notation $:=$, and represents a directional relationship between either side of the equation. A structural equation for $X$ as a cause of $Y$ might be represented as $Y := f(X)$. Usually, structural equations will look similar to $B := f(A, U)$, where $U$ represents the unknown random effects (or noise) from outside of the model on $B$. The figure on the right graphically represents the structural equation for $B$.



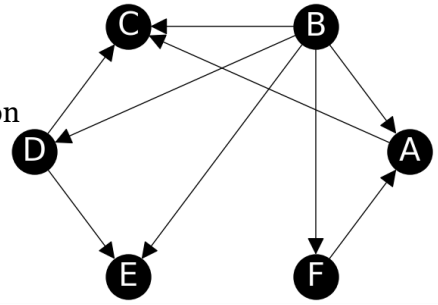

Structural causal models are collections of structural equations. We can loosely define $M$ as the following SCM:

$$M: \quad \begin{aligned} B &:= f_B(A, U_B) \\ C &:= f_C(A, B, U_C) \\ D &:= f_D(A, C, U_D) \end{aligned}$$

where $f_B$, $f_C$, and $f_D$ are some linear combination of their inputs, and each $U$ is some normal random variable representing external influence on the model. In this example, $B$, $C$, and $D$ are "endogenous" variables, because their values are determined

by factors within the model. The remaining variables ($A$, $U_B$, $U_C$, $U_D$) are "exogenous" variables, because their values are determined by factors outside of the model. Note that exogenous variables are parentless in the graphical representation of the SCM, and endogenous variables have at least one edge directed into them. Each of the endogenous variables are generated by a function of the other variables.

## 2. Model Training Example



Ground Truth

$n = 1000$; $N(\mu, \sigma, n)$ denotes $n$ samples of a normal distribution stored as a vector

$B := N(0, 1, n)$

$F := 0.7 * B + N(0.7, 2, n)$

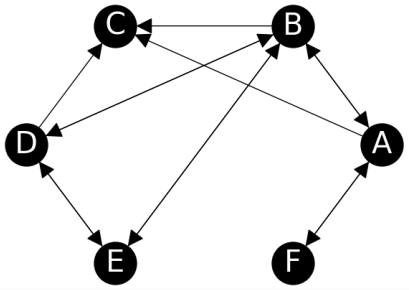
$A := 0.8 * B + F + N(1, 1.2, n)$

$D := B + N(0, 1, n)$

$C := A + B + 0.7 * D + N(0.2, 0.4, n)$

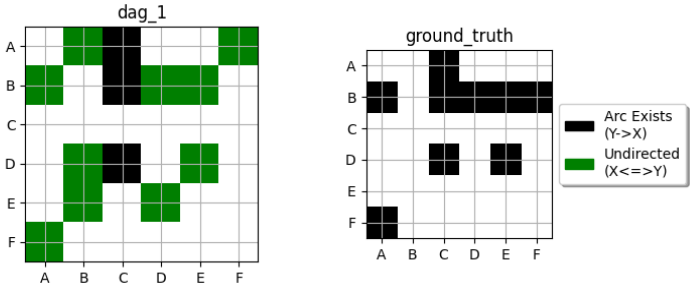
$E := B + 0.9 * D + N(1.3, 0.4, n)$

(Our training data is a 1000 x 6 matrix of random numbers generated using the process above)



PC Model Result

In the case of the PC algorithm, we have the option to include priori knowledge of what we might already know about the ground truth (in the form of known edges). The graph on the bottom is our causal graph learned from the simulated data using the PC-original algorithm.




## 3. Metrics

gCastle provides model metrics that compare the learned graph to the ground truth. gCastle also adds "reverse" (= the number of edges estimated that have opposite direction compared to the ground truth) to the $FDR$ and $FPR$ metrics.

| | |
|---|---|
| FDR | false discovery rate, expected proportion of rate of false positives/discoveries (incorrectly rejected null hypotheses), $(reverse + FP)/(FP + TP) = 1 - PPV$ |
| TPR | true positive rate, $TP/(TP + FN)$, P(edge in estimated graph\|that edge exists in |

| | |
|---|---|
| | ground truth with the same direction) |
| FPR | false positive rate, $(reverse + FP)/(FP + TN)$, P(edge in estimated graph\|that edge does not exist in ground truth). $N = FP + TN$ is the total number of ground truth negatives |
| SHD | Structural Hamming Distance, the number of edge insertions, deletions or flips in order to transform the predicted graph to the ground truth (or vice versa). = undirected extra + undirected missing + reverse |
| NNZ | number of non-negative entries/positives, $TP + FP$ |
| Precision/PPV | Positive Predictive value, How many learned facts are relevant? $TP/(TP + FP)$ $= 1 - FDR$ |
| Recall/TPR | true positive rate, $= TP/(TP + FN)$, P(predicting existing arc\|that arc exists in ground truth), "How many relevant facts are retrieved?" $= 1 - (\text{FNR}) = 1 - (FN)/(FN + TP)$ |
| F1 Score | measure of accuracy, $= 2(recall * precision)/(recall + precision)$ |
| gScore | $max(0, (TP - FP))/(TP + FN)$ |