

در این تمرین، قصد داریم به پیاده‌سازی الگوریتم PC و مقایسه‌ی آن با سایر الگوریتم‌های یادگیری ساختار گراف بپردازیم.

## ۱ الگوریتم PC

یک دسته از الگوریتم‌های یادگیری ساختار گرافی، الگوریتم‌های مبتنی بر استقلال شرطی هستند. یک ایده‌ی ابتدایی این است که برای هر جفت متغیر تصادفی  $(X, Y)$ ، مجموعه‌ی  $S_{XY}$  را پیدا کنیم به شرطی که  $X$  و  $Y$  به شرط  $S_{XY}$  مستقل شوند. در نتیجه‌ی این کار می‌توان گفت رئوس  $X$  و  $Y$ ، به شرط  $S_{XY}$  d-separate هستند. سپس می‌توان به کمک قواعد Meek به کلاس هم‌ارزی مارکوف رسید. یک روش کارآمد برای انجام این کار، بدون جست و جوی کامل، الگوریتم PC است که در ادامه، شبه‌کد آن آمده است. در صورتی که فرض کنیم متغیرها توزیع گوسی دارند، می‌توان از Partial

---

### Algorithm 1 The $PC_{pop}$ -algorithm

---

- 1: **INPUT:** Vertex Set  $V$ , Conditional Independence Information
  - 2: **OUTPUT:** Estimated skeleton  $C$ , separation sets  $S$  (only needed when directing the skeleton afterwards)
  - 3: Form the complete undirected graph  $\tilde{C}$  on the vertex set  $V$ .
  - 4:  $\ell = -1$ ;  $C = \tilde{C}$
  - 5: **repeat**
  - 6:    $\ell = \ell + 1$
  - 7:   **repeat**
  - 8:     Select a (new) ordered pair of nodes  $i, j$  that are adjacent in  $C$  such that  $|adj(C, i) \setminus \{j\}| \geq \ell$
  - 9:     **repeat**
  - 10:       Choose (new)  $\mathbf{k} \subseteq adj(C, i) \setminus \{j\}$  with  $|\mathbf{k}| = \ell$ .
  - 11:       **if**  $i$  and  $j$  are conditionally independent given  $\mathbf{k}$  **then**
  - 12:         Delete edge  $i, j$
  - 13:         Denote this new graph by  $C$
  - 14:         Save  $\mathbf{k}$  in  $S(i, j)$  and  $S(j, i)$
  - 15:       **end if**
  - 16:     **until** edge  $i, j$  is deleted or all  $\mathbf{k} \subseteq adj(C, i) \setminus \{j\}$  with  $|\mathbf{k}| = \ell$  have been chosen
  - 17:   **until** all ordered pairs of adjacent variables  $i$  and  $j$  such that  $|adj(C, i) \setminus \{j\}| \geq \ell$  and  $\mathbf{k} \subseteq adj(C, i) \setminus \{j\}$  with  $|\mathbf{k}| = \ell$  have been tested for conditional independence
  - 18: **until** for each ordered pair of adjacent nodes  $i, j$ :  $|adj(C, i) \setminus \{j\}| < \ell$ .
- 

Correlation به عنوان تست استقلال شرطی استفاده کرد. الگوریتم PC را پیاده‌سازی کنید. از Partial Correlation به عنوان آزمون استقلال شرطی خود بهره ببرید. برای مطالعه‌ی بیشتر در مورد Partial Correlation می‌توانید به مدخل مربوطه در سایت ویکیپدیا مراجعه کنید. برای Significance Level هم می‌توانید از Fisher's Z-transform استفاده کنید که توضیحات آن در همان مدخل سایت ویکی‌پدیا، تحت بخش As conditional independence test آمده است.

## ۲ قواعد Meek

در درس با قواعد Meek آشنا شدید. به کمک این قواعد، کدی بنویسید که با دریافت خروجی الگوریتم فوق، کلاس هم‌ارزی مارکوف را به دست آورد. یعنی  $C$  و  $S$  خروجی الگوریتم فوق را دریافت کرده و یک CPDAG به عنوان خروجی تحویل

### ۳ بررسی عملکرد الگوریتم

(الف) یک بار صد و یک بار هزار نمونه با SCM زیر تولید کنید. فرض کنید نویزهای  $\epsilon_i$  گوسی استاندارد و مستقل باشند.

$$\begin{cases} X_1 = 1.2\epsilon_1 \\ X_2 = \epsilon_2 \\ X_3 = 2X_1 - 0.5X_2 + \epsilon_3 \\ X_4 = 0.4X_3 + \epsilon_4 \\ X_5 = 0.8X_3 - X_4 + 0.6\epsilon_5 \\ X_6 = X_2 + X_5 + \epsilon_6 \end{cases}$$

با انتخاب Significance Level مناسب، CPDAG را به کمک داده‌ها به دست آورده و با CPDAG واقعی مقایسه کنید.

(ب) به کمک توابع randomDAG و rmvDAG در پکیج pcalg، در ابتدا یک گراف تصادفی با بیست راس و با پارامترهای  $uB = 1$  و  $lB = 0.1$ ،  $p = 0.2$  بسازید. توزیع نویزهای برونی را نرمال استاندارد گرفته و برای هر نمونه هزار متغیر بسازید. حال به کمک الگوریتم PC پیاده‌سازی شده، نمودار Hamming Distance بین DAG واقعی و CPDAG به دست آمده را بر حسب سطح اطمینان رسم کنید.