CSCM603130: Sistem Cerdas
Probabilistic Reasoning

Fariz Darari, Aruni Yasmin Azizah

Fakultas Ilmu Komputer
Universitas Indonesia

2019/2020 • Semester Ganjil

Outline

1. Bertindak dalam Ketidakpastian (*Uncertainty*)
2. Mengukur Ketidakpastian
   - Notasi Dasar Probabilitas
   - Inference
   - Kompleksitas
3. *Efficient* probabilistic reasoning
   - (Absolute) Independence
   - Bayes' Rule & Conditional Independence
4. Bayesian Networks
   - Knowledge Representation dalam Ketidakpastian
   - Semantic Bayesian Network
   - Efficient representation of conditional distributions
   - Exact vs Approximate Inference

Outline

Knowledge engineering dalam bidang Kedokteran Gigi - 1

- Anda diminta membuat agent untuk membantu diagnosis dokter gigi.

Knowledge engineering dalam bidang Kedokteran Gigi - 1

- Anda diminta membuat agent untuk membantu diagnosis dokter gigi.
- Dengan menggunakan aturan diagnosis—menyimpulkan sebab dari akibat:

Knowledge engineering dalam bidang Kedokteran Gigi - 1

- Anda diminta membuat agent untuk membantu diagnosis dokter gigi.
- Dengan menggunakan aturan diagnosis—menyimpulkan sebab dari akibat:
  $Toothache \implies Cavity$

Knowledge engineering dalam bidang Kedokteran Gigi - 1

- Anda diminta membuat agent untuk membantu diagnosis dokter gigi.
- Dengan menggunakan aturan diagnosis—menyimpulkan sebab dari akibat: $Toothache \implies Cavity$
- Tidak semua pasien sakit gigi karena berlubang, beberapa disebabkan karena gusi bermasalah, nanah, dll:

Knowledge engineering dalam bidang Kedokteran Gigi - 1

- Anda diminta membuat agent untuk membantu diagnosis dokter gigi.
- Dengan menggunakan aturan diagnosis—menyimpulkan sebab dari akibat:
  $Toothache \implies Cavity$
- Tidak semua pasien sakit gigi karena berlubang, beberapa disebabkan karena gusi bermasalah, nanah, dll:
  $Toothache \implies Cavity \lor GumProblem \lor Abscess \ldots$

Knowledge engineering dalam bidang Kedokteran Gigi - 1

- Anda diminta membuat agent untuk membantu diagnosis dokter gigi.
- Dengan menggunakan aturan diagnosis—menyimpulkan sebab dari akibat:
  $Toothache \implies Cavity$
- Tidak semua pasien sakit gigi karena berlubang, beberapa disebabkan karena gusi bermasalah, nanah, dll:
  $Toothache \implies Cavity \vee GumProblem \vee Abscess \ldots$
- Masalah: agar aturan tersebut benar, perlu menyatakan banyak sekali (mungkin tak hingga!) penyebab sakit gigi.

Knowledge engineering dalam bidang Kedokteran Gigi - 1

- Anda diminta membuat agent untuk membantu diagnosis dokter gigi.
- Dengan menggunakan aturan diagnosis—menyimpulkan sebab dari akibat:
  $Toothache \implies Cavity$
- Tidak semua pasien sakit gigi karena berlubang, beberapa disebabkan karena gusi bermasalah, nanah, dll:
  $Toothache \implies Cavity \lor GumProblem \lor Abscess \ldots$
- Masalah: agar aturan tersebut benar, perlu menyatakan banyak sekali (mungkin tak hingga!) penyebab sakit gigi.
- Alternatif: ubah menjadi aturan causal—menyimpulkan akibat dari sebab:

Knowledge engineering dalam bidang Kedokteran Gigi - 1

- Anda diminta membuat agent untuk membantu diagnosis dokter gigi.
- Dengan menggunakan aturan diagnosis—menyimpulkan sebab dari akibat:
  $Toothache \implies Cavity$
- Tidak semua pasien sakit gigi karena berlubang, beberapa disebabkan karena gusi bermasalah, nanah, dll:
  $Toothache \implies Cavity \lor GumProblem \lor Abscess \ldots$
- Masalah: agar aturan tersebut benar, perlu menyatakan banyak sekali (mungkin tak hingga!) penyebab sakit gigi.
- Alternatif: ubah menjadi aturan causal—menyimpulkan akibat dari sebab: $Cavity \implies Toothache$

Knowledge engineering dalam bidang Kedokteran Gigi - 1

- Anda diminta membuat agent untuk membantu diagnosis dokter gigi.
- Dengan menggunakan aturan diagnosis—menyimpulkan sebab dari akibat:
  $Toothache \implies Cavity$
- Tidak semua pasien sakit gigi karena berlubang, beberapa disebabkan karena gusi bermasalah, nanah, dll:
  $Toothache \implies Cavity \lor GumProblem \lor Abscess \ldots$
- Masalah: agar aturan tersebut benar, perlu menyatakan banyak sekali (mungkin tak hingga!) penyebab sakit gigi.
- Alternatif: ubah menjadi aturan causal—menyimpulkan akibat dari sebab: $Cavity \implies Toothache$
- Masalah (lagi!): tidak semua gigi berlubang menyebabkan sakit! Anteseden perlu diperkuat: gigi berlubang bagaimana yang menyebabkan sakit gigi?—qualification problem

Knowledge engineering dalam bidang Kedokteran Gigi - 2

Pendekatan logika secara murni sulit karena:

- Laziness: perlu menyatakan secara lengkap anteseden maupun konsekuen dari suatu *rule*.
- Theoretical ignorance: perlu memiliki teori yang lengkap dalam kedokteran gigi untuk problem ini.
- Practical ignorance: bahkan jika teorinya sudah lengkap pun, ada ketidakpastian tentang kondisi seorang pasien karena tidak semua tes dilakukan (butuh banyak biaya, waktu, dst).

Dunia penuh ketidakpastian (*uncertainty*)

- Sebuah taxi agent T mengantarkan penumpang ke bandara untuk terbang ke luar negeri.
  Mis. action A_t = pergi ke bandara t menit sebelum pesawat berangkat.
  Apakah T dengan action A_t *pasti* berhasil mengantarkan penumpang tiba di bandara tepat waktu?

Dunia penuh ketidakpastian (*uncertainty*)

- Sebuah taxi agent T mengantarkan penumpang ke bandara untuk terbang ke luar negeri.
  Mis. action A_t = pergi ke bandara t menit sebelum pesawat berangkat.
  Apakah T dengan action A_t *pasti* berhasil mengantarkan penumpang tiba di bandara tepat waktu?
- Ada banyak masalah:

Dunia penuh ketidakpastian (*uncertainty*)

- Sebuah taxi agent T mengantarkan penumpang ke bandara untuk terbang ke luar negeri.
  Mis. action A_t = pergi ke bandara t menit sebelum pesawat berangkat.
  Apakah T dengan action A_t *pasti* berhasil mengantarkan penumpang tiba di bandara tepat waktu?
- Ada banyak masalah:
  - Tidak bisa memastikan keadaan jalan, kemacetan, dll. (partially observable).

Dunia penuh ketidakpastian (*uncertainty*)

- Sebuah taxi agent T mengantarkan penumpang ke bandara untuk terbang ke luar negeri.
  Mis. action A_t = pergi ke bandara t menit sebelum pesawat berangkat.
  Apakah T dengan action A_t *pasti* berhasil mengantarkan penumpang tiba di bandara tepat waktu?
- Ada banyak masalah:
  - Tidak bisa memastikan keadaan jalan, kemacetan, dll. (partially observable).
  - Kebenaran informasi tidak bisa dijamin (noisy sensor).

Dunia penuh ketidakpastian (*uncertainty*)

- Sebuah taxi agent T mengantarkan penumpang ke bandara untuk terbang ke luar negeri.
  Mis. action A_t = pergi ke bandara t menit sebelum pesawat berangkat.
  Apakah T dengan action A_t *pasti* berhasil mengantarkan penumpang tiba di bandara tepat waktu?
- Ada banyak masalah:
  - Tidak bisa memastikan keadaan jalan, kemacetan, dll. (partially observable).
  - Kebenaran informasi tidak bisa dijamin (noisy sensor).
  - Ketidakpastian dalam melakukan tindakan, mis. ban kempes (nondeterministic).

Dunia penuh ketidakpastian (*uncertainty*)

- Sebuah taxi agent T mengantarkan penumpang ke bandara untuk terbang ke luar negeri.
  Mis. action A_t = pergi ke bandara t menit sebelum pesawat berangkat.
  Apakah T dengan action A_t *pasti* berhasil mengantarkan penumpang tiba di bandara tepat waktu?
- Ada banyak masalah:
  - Tidak bisa memastikan keadaan jalan, kemacetan, dll. (partially observable).
  - Kebenaran informasi tidak bisa dijamin (noisy sensor).
  - Ketidakpastian dalam melakukan tindakan, mis. ban kempes (nondeterministic).
  - **Kalaupun** semua hal di atas bisa dinyatakan, *representation* dan *reasoning* pada logical taxi agent akan luar biasa repot.

Keterbatasan pendekatan logika murni

Sebuah pendekatan yang murni secara logika...

Keterbatasan pendekatan logika murni

Sebuah pendekatan yang murni secara logika...

- beresiko menyimpulkan dengan salah, mis: "*A_{60} berhasil mengantarkan penumpang dengan tepat waktu*", atau

Keterbatasan pendekatan logika murni

Sebuah pendekatan yang murni secara logika...

- beresiko menyimpulkan dengan salah, mis: *"A_{60} berhasil mengantarkan penumpang dengan tepat waktu"*, atau
- kesimpulan terlalu lemah, mis: *"A_{60} berhasil mengantarkan penumpang dengan tepat waktu asal ... nggak ada kecelakaan di tol, dan nggak hujan, dan ban nggak kempes, ..."*

Keterbatasan pendekatan logika murni

Sebuah pendekatan yang murni secara logika...

- beresiko menyimpulkan dengan salah, mis: "*A_{60} berhasil mengantarkan penumpang dengan tepat waktu*", atau
- kesimpulan terlalu lemah, mis: "*A_{60} berhasil mengantarkan penumpang dengan tepat waktu asal ... nggak ada kecelakaan di tol, dan nggak hujan, dan ban nggak kempes, ...*"
- kesimpulan tidak rational, mis: kesimpulannya A_{1440}—berhasil mengantarkan tepat waktu, tetapi terpaksa menunggu semalam di bandara...

Keterbatasan pendekatan logika murni

Sebuah pendekatan yang murni secara logika...

- beresiko menyimpulkan dengan salah, mis: *"A_{60} berhasil mengantarkan penumpang dengan tepat waktu"*, atau
- kesimpulan terlalu lemah, mis: *"A_{60} berhasil mengantarkan penumpang dengan tepat waktu asal ... nggak ada kecelakaan di tol, dan nggak hujan, dan ban nggak kempes, ..."*
- kesimpulan tidak rational, mis: kesimpulannya A_{1440}—berhasil mengantarkan tepat waktu, tetapi terpaksa menunggu semalam di bandara...

Masalah ini bisa diselesaikan dengan probabilistic reasoning

"Berdasarkan info yang ada, A_{60} akan berhasil dengan probabilitas 0.04".

Menangani ketidakpastian

- Kalimat seperti "A_{60} akan berhasil dengan probabilitas 0.04" disebut probabilistic assertion.

Menangani ketidakpastian

- Kalimat seperti "A_{60} akan berhasil dengan probabilitas 0.04" disebut probabilistic assertion.
- Nilai probabilitas pada probabilistic assertion merangkum efek ketidakpastian (info tak lengkap, tak bisa dijamin kebenarannya, action nondeterministic, dst.) dan menyatakannya sebagai sebuah bilangan.

Menangani ketidakpastian

- Kalimat seperti "A_{60} akan berhasil dengan probabilitas 0.04" disebut probabilistic assertion.
- Nilai probabilitas pada probabilistic assertion merangkum efek ketidakpastian (info tak lengkap, tak bisa dijamin kebenarannya, action nondeterministic, dst.) dan menyatakannya sebagai sebuah bilangan.
- Logical agent menyatakan suatu pernyataan salah, benar, atau tidak ada jawaban. Sedangkan, probabilistic agent menyatakan suatu pernyataan dengan *degree of belief* (derajat keyakinan) antara 0 (pasti salah) sampai dengan 1 (pasti benar).

Menangani ketidakpastian

Memaknai *probabilistic assertion*:

"Kalimat X bernilai *true* dengan probabilitas N, $0 \leq N \leq 1$".
Pernyataan tentang knowledge atau belief state dari agent tentang X, BUKAN pernyataan tentang X itu sendiri dalam dunia nyata

Menangani ketidakpastian

Memaknai *probabilistic assertion*:

"Kalimat X bernilai *true* dengan probabilitas N, $0 \leq N \leq 1$".
Pernyataan tentang knowledge atau belief state dari agent tentang X, BUKAN pernyataan tentang X itu sendiri dalam dunia nyata

- Probabilitas pasien memiliki gigi berlubang, jika diketahui (*given*) ia menderita sakit gigi, adalah 0.8. (berdasarkan *knowledge state*, ada 8 dari 10 pasien punya gigi berlubang, jika diketahui mengalami sakit gigi).

Menangani ketidakpastian

Memaknai *probabilistic assertion*:

"Kalimat X bernilai *true* dengan probabilitas N, $0 \leq N \leq 1$".
Pernyataan tentang knowledge atau belief state dari agent tentang X, BUKAN pernyataan tentang X itu sendiri dalam dunia nyata

- Probabilitas pasien memiliki gigi berlubang, jika diketahui (*given*) ia menderita sakit gigi, adalah 0.8. (berdasarkan *knowledge state*, ada 8 dari 10 pasien punya gigi berlubang, jika diketahui mengalami sakit gigi).
- $P(A_{60} |$ tidak ada laporan kecelakaan$) = 0.06$

Menangani ketidakpastian

Memaknai *probabilistic assertion*:

"Kalimat X bernilai *true* dengan probabilitas N, $0 \leq N \leq 1$".
Pernyataan tentang knowledge atau belief state dari agent tentang X, BUKAN pernyataan tentang X itu sendiri dalam dunia nyata

- Probabilitas pasien memiliki gigi berlubang, jika diketahui (*given*) ia menderita sakit gigi, adalah 0.8. (berdasarkan *knowledge state*, ada 8 dari 10 pasien punya gigi berlubang, jika diketahui mengalami sakit gigi).
- $P(A_{60}|$ tidak ada laporan kecelakaan$) = 0.06$
- Nilai probabilitas sebuah proposition bisa berubah dengan informasi baru.

Menangani ketidakpastian

Memaknai *probabilistic assertion*:

"Kalimat X bernilai *true* dengan probabilitas N, $0 \leq N \leq 1$".
Pernyataan tentang knowledge atau belief state dari agent tentang X, BUKAN pernyataan tentang X itu sendiri dalam dunia nyata

- Probabilitas pasien memiliki gigi berlubang, jika diketahui (*given*) ia menderita sakit gigi, adalah 0.8. (berdasarkan *knowledge state*, ada 8 dari 10 pasien punya gigi berlubang, jika diketahui mengalami sakit gigi).
- $P(A_{60}|$ tidak ada laporan kecelakaan$) = 0.06$
- Nilai probabilitas sebuah proposition bisa berubah dengan informasi baru.
- Probabilitas pasien memiliki gigi berlubang, jika diketahui (*given*) ia menderita sakit gigi dan adanya pengalaman masalah kesehatan pada gusi, adalah 0.4.

Menangani ketidakpastian

Memaknai *probabilistic assertion*:

"Kalimat X bernilai *true* dengan probabilitas N, $0 \leq N \leq 1$".
Pernyataan tentang knowledge atau belief state dari agent tentang X, BUKAN pernyataan tentang X itu sendiri dalam dunia nyata

- Probabilitas pasien memiliki gigi berlubang, jika diketahui (*given*) ia menderita sakit gigi, adalah 0.8. (berdasarkan *knowledge state*, ada 8 dari 10 pasien punya gigi berlubang, jika diketahui mengalami sakit gigi).
- $P(A_{60}|$ tidak ada laporan kecelakaan$) = 0.06$
- Nilai probabilitas sebuah proposition bisa berubah dengan informasi baru.
- Probabilitas pasien memiliki gigi berlubang, jika diketahui (*given*) ia menderita sakit gigi dan adanya pengalaman masalah kesehatan pada gusi, adalah 0.4.
- $P(A_{60}|$ tidak ada laporan kecelakaan, jam 4 pagi$) = 0.15$

Mengambil keputusan rasional dlm ketidakpastian

Andaikan agent mempercayai nilai-nilai sbb.:

$$\begin{aligned} P(A_{60}|\ldots) &= 0.04 \\ P(A_{120}|\ldots) &= 0.7 \\ P(A_{150}|\ldots) &= 0.9 \\ P(A_{1440}|\ldots) &= 0.999 \end{aligned}$$

Tindakan mana yang dipilih?

Mengambil keputusan rasional dlm ketidakpastian

Andaikan agent mempercayai nilai-nilai sbb.:

$$\begin{aligned} P(A_{60}|\ldots) &= 0.04 \\ P(A_{120}|\ldots) &= 0.7 \\ P(A_{150}|\ldots) &= 0.9 \\ P(A_{1440}|\ldots) &= 0.999 \end{aligned}$$

Tindakan mana yang dipilih?

- Tergantung prioritas, mis. ketinggalan pesawat vs. begadang di lobby bandara, dst.
- Utility theory digunakan untuk representation dan reasoning terhadap preferensi (memberi nilai utility pada hasil tindakan suatu action).
- Decision theory = utility theory + probability theory

Outline

Logical Assertion vs. Probabilistic Assertion

Possible worlds:

- Pada *logical assertion*, kita bisa memastikan dunia mana yang membuat assertion tersebut bernilai benar.
- Pada *probability assertion*, kita **tidak tahu pasti** yang mana, tetapi satu dunia bisa lebih mungkin dari dunia yang lain.

Outline

Sample space & probability model

Himpunan semua *possible worlds* disebut sample space (Ω). Masing-masing dunia alternatif disebut sample point, atau atomic event (ω).

Contoh

Jika kita melempar sebuah dadu, maka Ω berisi 6 sample point: $\omega_1 \dots \omega_6$.

Sebuah probability model dibangun dari *sample space* di mana tiap *sample point* ω diberi nilai $P(\omega)$ sehingga:

- Setiap nilai antara 0 s/d 1: $0 \leq P(\omega) \leq 1$
- Jumlah nilai seluruh *sample point* $= 1$: $\Sigma_{\omega \in \Omega} P(\omega) = 1$

Contohnya, untuk "dunia" dengan lemparan 1 dadu:

$P(\omega_1) = P(\omega_2) = P(\omega_3) = P(\omega_4) = P(\omega_5) = P(\omega_6) = \frac{1}{6}$

Sample space & probability model

Himpunan semua *possible worlds* disebut sample space (Ω). Masing-masing dunia alternatif disebut sample point, atau atomic event (ω).

Contoh

Jika kita melempar sebuah dadu, maka Ω berisi 6 sample point: $\omega_1 \ldots \omega_6$.

Sebuah probability model dibangun dari *sample space* di mana tiap *sample point* ω diberi nilai $P(\omega)$ sehingga:

- Setiap nilai antara 0 s/d 1: $0 \leq P(\omega) \leq 1$
- Jumlah nilai seluruh *sample point* = 1: $\Sigma_{\omega \in \Omega} P(\omega) = 1$

Contohnya, untuk "dunia" dengan lemparan 1 dadu:

$P(\omega_1) = P(\omega_2) = P(\omega_3) = P(\omega_4) = P(\omega_5) = P(\omega_6) = \frac{1}{6}$

Contohnya, untuk "dunia" dengan lemparan 2 dadu:

$P(\omega_{1,1}) = P(\omega_{1,2}) = \ldots = P(\omega_{6,5}) = P(\omega_{6,6}) = \frac{1}{36}$

Event

- Sebuah event A adalah sembarang subset dari sample space Ω.
- Event A mendeskripsikan sebuah proposition ϕ : himpunan sample points (*possible worlds*) yang menyebabkan proposition ϕ bernilai benar.
- Probabilitas event A (proposition ϕ) adalah jumlah probabilitas *sample point* anggotanya.

$$P(A) = \sum_{\omega \in A} P(\omega) \text{ atau } P(\phi) = \sum_{\omega \in \phi} P(\omega)$$

- Contoh: Pada lemparan satu dadu, $P(dadu \geq 4) = 3 \times \frac{1}{6} = \frac{1}{2}$
- Contoh: Pada lemparan dua dadu, $P(dadu_1 = 5) = 6 \times \frac{1}{36} = \frac{1}{6}$

Prior vs. posterior probability

Suatu event juga dapat dilihat dari ada atau tidaknya informasi tambahan.

Prior probability:

Nilai probability tanpa informasi spesifik (unconditional).
Mis. $P(cavity)$, $P(toothache)$, dst.

Prior vs. posterior probability

Suatu event juga dapat dilihat dari ada atau tidaknya informasi tambahan.

Prior probability:

Nilai probability tanpa informasi spesifik (unconditional).
Mis. $P(cavity)$, $P(toothache)$, dst.

Seringkali, suatu *event* memiliki beberapa informasi tambahan, yang disebut sebagai **evidence**.

Prior vs. posterior probability

Suatu event juga dapat dilihat dari ada atau tidaknya informasi tambahan.

Prior probability:

Nilai probability tanpa informasi spesifik (unconditional).
Mis. $P(cavity)$, $P(toothache)$, dst.

Seringkali, suatu *event* memiliki beberapa informasi tambahan, yang disebut sebagai **evidence**.

Posterior probability:

Nilai probability jika sesuatu informasi spesifik diketahui (conditional).
Mis.: $P(cavity|toothache)$
Baca: "*probabilitas gigi pasien berlubang jika diketahui ia sakit gigi*"

Definisi *conditional probability*: $P(a|b) = \frac{P(a \wedge b)}{P(b)}$ untuk $P(b) > 0$
Perumusan alternatif (product rule): $P(a \wedge b) = P(a|b)P(b) = P(b|a)P(a)$

Prior vs. posterior probability

Contoh: Sampel pasien

Toothache	+	+	+	+	+	-	+	+	+	+	+	-	+	+	-
Cavity	+	+	-	-	+	+	+	+	+	+	+	+	-	-	-
GumProblem	+	-	-	+	+	+	-	-	-	-	-	+	-	-	-

- $P(t|c) =$

Prior vs. posterior probability

Contoh: Sampel pasien

Toothache	+	+	+	+	+	-	+	+	+	+	+	-	+	+	-
Cavity	+	+	-	-	+	+	+	+	+	+	+	+	-	-	-
GumProblem	+	-	-	+	+	+	-	-	-	-	-	+	-	-	-

- $P(t|c) = \frac{P(t \wedge c)}{P(c)} = \frac{8/15}{10/15} = 8/10$
- $P(t \wedge c) =$

Prior vs. posterior probability

Contoh: Sampel pasien

Toothache	+	+	+	+	+	-	+	+	+	+	+	-	+	+	-
Cavity	+	+	-	-	+	+	+	+	+	+	+	+	-	-	-
GumProblem	+	-	-	+	+	+	-	-	-	-	-	+	-	-	-

- $P(t|c) = \frac{P(t \wedge c)}{P(c)} = \frac{8/15}{10/15} = 8/10$
- $P(t \wedge c) = P(t|c)P(c) = (8/10)(10/15) = 8/15$

Prior vs. posterior probability

Contoh: Sampel pasien

Toothache	+	+	+	+	+	-	+	+	+	+	+	-	+	+	-
Cavity	+	+	-	-	+	+	+	+	+	+	+	+	-	-	-
GumProblem	+	-	-	+	+	+	-	-	-	-	-	+	-	-	-

- $P(t|c) = \frac{P(t \wedge c)}{P(c)} = \frac{8/15}{10/15} = 8/10$
- $P(t \wedge c) = P(t|c)P(c) = (8/10)(10/15) = 8/15$
- $P(c|t)$?
- $P(t \wedge c)$?
- $P(t|g)$?
- $P(t|c \wedge g)$?

Random variable

- Secara formal, random variable adalah fungsi yang memetakan setiap *sample point* ke dalam suatu domain, mis. boolean, integer, real.

Contoh dgn. random variable *Ganjil* pada dadu:

$Ganjil(\omega_1) = true$,
$Ganjil(\omega_2) = false$,
dst...

Random variable & probability distribution

- Menghitung probabilitas suatu proposisi:

$$P(X = x_i) = \sum_{\omega : X(\omega) = x_i} P(\omega)$$

Contoh dgn. dadu:

$P(Ganjil = false) = P(\neg ganjil) = \frac{1}{6} + \frac{1}{6} + \frac{1}{6} = \frac{1}{2}$

- Sebuah *probability model P* menghasilkan probability distribution **P**: probabilitas dari *semua* nilai pada suatu *random variable*

Contoh dgn cuaca:

$P(Weather = sunny) = 0.7$
$P(Weather = rain) = 0.2$
$P(Weather = cloudy) = 0.08$
$P(Weather = snow) = 0.02$
atau disingkat $\mathbf{P}(Weather) = \langle 0.7, 0.2, 0.08, 0.02 \rangle$, dengan asumsi urutan sbb: $\langle sunny, rain, cloudy, snow \rangle$

Joint probability distribution

- Dalam AI, seringkali *sample point didefinisikan* oleh nilai sekumpulan *random variable*.
- Jadi, *sample space* berisi semua kemungkinan kombinasi nilai semua variable.
- Joint probability distribution dari sekumpulan random variable memberikan nilai probability untuk setiap *sample point* tersebut.

Contoh:

Andaikan kita tertarik mengamati hubungan cuaca dengan sakit gigi, contoh joint probability distribution-nya, $\mathbf{P}(Toothache, Weather)$:

	Weather			
	sunny	*rain*	*cloudy*	*snow*
$Toothache = true$	0.144	0.02	0.016	0.02
$Toothache = false$	0.576	0.08	0.064	0.08

Contoh yang memilukan

Bayangkan masalah dokter gigi, di mana ada 3 *random variable*:

- Cavity: apakah pasien memiliki gigi berlubang atau tidak?
- Toothache: apakah pasien merasa sakit gigi atau tidak?
- Catch: apakah pisau dokter nyangkut di gigi pasien atau tidak?

Full joint probability distribution sbb.:

	$toothache$		$\neg\, toothache$	
	$catch$	$\neg\, catch$	$catch$	$\neg\, catch$
$cavity$	**.108**	**.012**	**.072**	**.008**
$\neg\, cavity$	**.016**	**.064**	**.144**	**.576**

Full joint probability distribution: joint probability distribution untuk *semua* random variable.

Outline

Inference dengan full joint distribution probability

Dengan *full joint probability distribution*, probability sembarang proposition bisa dihitung sbg. jumlah probability sample point di mana proposition tersebut bernilai true.

	toothache		$\neg$ *toothache*	
	catch	$\neg$ *catch*	*catch*	$\neg$ *catch*
cavity	**.108**	**.012**	**.072**	**.008**
$\neg$ *cavity*	**.016**	**.064**	**.144**	**.576**

Inference dengan full joint distribution probability

Dengan *full joint probability distribution*, probability sembarang proposition bisa dihitung sbg. jumlah probability sample point di mana proposition tersebut bernilai true.

	toothache		¬ *toothache*	
	catch	¬ *catch*	*catch*	¬ *catch*
cavity	**.108**	**.012**	**.072**	**.008**
¬ *cavity*	**.016**	**.064**	**.144**	**.576**

$P(toothache) = 0.108 + 0.012 + 0.016 + 0.064 = 0.2$

Inference dengan full joint distribution probability

Dengan *full joint probability distribution*, probability sembarang proposition bisa dihitung sbg. jumlah probability sample point di mana proposition tersebut bernilai true.

	toothache		$\neg$ *toothache*	
	catch	$\neg$ *catch*	*catch*	$\neg$ *catch*
cavity	**.108**	**.012**	**.072**	**.008**
$\neg$ *cavity*	**.016**	**.064**	**.144**	**.576**

$$
\begin{aligned}
P(cavity \vee toothache) &= 0.108 + 0.012 + 0.072 + 0.008 + \\
&\quad 0.016 + 0.064 \\
&= 0.28
\end{aligned}
$$

Inference dengan full joint distribution probability

Dengan *full joint probability distribution*, probability sembarang proposition bisa dihitung sbg. jumlah probability sample point di mana proposition tersebut bernilai true.

	toothache		¬ *toothache*	
	catch	¬ *catch*	*catch*	¬ *catch*
cavity	**.108**	**.012**	**.072**	**.008**
¬ *cavity*	**.016**	**.064**	**.144**	**.576**

Bisa juga menghitung conditional probability:

$$P(\neg cavity | toothache) = \frac{P(\neg cavity \land toothache)}{P(toothache)}$$

$$= \frac{0.016 + 0.064}{0.108 + 0.012 + 0.016 + 0.064} = 0.4$$

Normalisasi pada Inference Conditional Probability

$P(\neg cavity | toothache) = P(\neg cavity \wedge toothache) / P(toothache)$
$P(cavity | toothache) = P(cavity \wedge toothache) / P(toothache)$
Perhatikan bahwa pembaginya sama!
Disebut konstanta normalisasi $\alpha \rightarrow$ supaya jumlahnya 1.

$$\mathbf{P}(Cavity | toothache) = \alpha\, \mathbf{P}(Cavity, toothache)$$
$$\begin{aligned} &= \alpha\, [\mathbf{P}(Cavity, toothache, catch) + \mathbf{P}(Cavity, toothache, \neg catch)] \\ &= \alpha\, [\langle 0.108, 0.016 \rangle + \langle 0.012, 0.064 \rangle] \\ &= \alpha\, \langle 0.12, 0.08 \rangle = \langle 0.6, 0.4 \rangle \end{aligned}$$

Ide utama:

α dapat diperoleh ketika menghitung **distribusi** utk. *query variable* (mis. *Cavity*) $\rightarrow$ semua kemungkinan nilai *Cavity*.

Inference procedure untuk conditional probability

Contoh sebuah "query"

- $P(\neg cavity | toothache)$
- "Berapakah probabilitas gigi Anto tidak berlubang jika diketahui ia menderita sakit gigi?"

- Query variable: variable yang ingin dihitung probabilitasnya (*Cavity*)
- Evidence variable: variable yang kita tahu nilainya (*Toothache*)
- Hidden variable: variable yang tidak kita tahu nilainya (*Catch*)

Inference dengan (full) joint probability distribution:

Diberikan kumpulan *evidence variable* $\mathbf{E}$, di mana $\mathbf{e}$ adalah kumpulan nilai dari variabel tsb, dan *hidden variable* $\mathbf{Y}$. $\mathbf{P}(X|\mathbf{e})$ adalah penjumlahan probabilitas *query variable* X untuk semua kemungkinan nilai $\mathbf{y}$:

$$\mathbf{P}(X|\mathbf{e}) = \alpha \mathbf{P}(X,\mathbf{e}) = \alpha \Sigma_{\mathbf{y}} \mathbf{P}(X, \mathbf{e}, \mathbf{y})$$

Outline

Complexity inference dgn full joint distribution

- Jika ada n random variable Boolean, tabel full joint distribution berisi $O(2^n)$ nilai.
- **Space complexity** eksponensial dalam n. Untuk disimpan saja tidak feasible, apalagi **menghitung semua nilainya dari data empiris!**
- **Time complexity** juga eksponensial dalam n: $O(2^n)$.

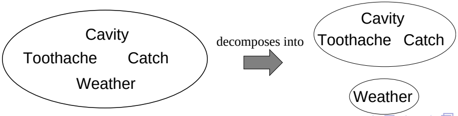
Dalam kenyataan, masalah probabilistic reasoning melibatkan ribuan *random variable*. Kita butuh metode yang lebih efisien!

Outline

Contoh: Gigi dan Cuaca

Mis. ditambahkan random variable $Weather = \langle sunny, rain, cloudy, snow \rangle$ ke masalah dokter gigi ($Cavity, Toothache, Catch$)

- Dibutuhkan 32 nilai $(4 \times 2 \times 2 \times 2)$ untuk tabel full joint distribution-nya.
- Untuk menghitung probabilitas *proposition* mengenai cuaca dan gigi menggunakan product rule:
  $P(cloudy \wedge toothache) = P(cloudy|toothache) \times P(toothache)$
- Tapi cuaca dan gigi tidak berhubungan...sehingga bisa kita nyatakan:
  $P(cloudy|toothache) = P(cloudy)$
- Dari sini, dapat kita hitung probabilitasnya:
  $P(cloudy \wedge toothache) = P(cloudy) \times P(toothache)$
- Tabel full joint distribution dengan 4 variabel (32 nilai) dapat didekomposisi: tabel dengan 3 variabel (8 nilai) dan tabel dengan 1 variabel (4 nilai).



Outline

Independence

- Independence menyatakan bahwa ada random variable yang tidak saling mempengaruhi (**independence/absolute independence**).
- Sebagai contoh, gigi dan cuaca tidak saling mempengaruhi, sehingga *Toothache* dan *Weather* dapat dikatakan saling independent.

Dua proposition a dan b independent jhj

$P(a|b) = P(a)$ atau $P(b|a) = P(b)$ atau $P(a \wedge b) = P(a) \times P(b)$

Dua variable A dan B independent jhj

$\mathbf{P}(A|B) = \mathbf{P}(A)$ atau $\mathbf{P}(B|A) = \mathbf{P}(B)$ atau $\mathbf{P}(A, B) = \mathbf{P}(A)\mathbf{P}(B)$

- Independence menyederhanakan kompleksitas inferensi (lebih efisien): mengurangi jumlah informasi yang dibutuhkan dalam menyatakan *full joint probability distribution*.
  - Kasus "Gigi dan Cuaca": 32 vs. 12 nilai *probability distribution*!
- Dalam kenyataan, *absolute independence* jarang terjadi.

Outline

Bayes' Rule

Product Rule:

$$P(a \wedge b) = P(a|b)P(b) \quad \text{dan} \quad P(a \wedge b) = P(b|a)P(a)$$

Dengan memanipulasi *product rule* di atas diperoleh Bayes' Rule.

Bayes' Rule

$$P(b|a) = \frac{P(a|b)P(b)}{P(a)}$$

Penerapan Bayes' Rule

Seringkali *evidence* yang didapatkan berupa *akibat* dari suatu *sebab* yang belum diketahui dan kita ingin menentukan sebab tersebut. Dalam Bayes' rule:

$$P(Sebab|Akibat) = \frac{P(Akibat|Sebab)P(Sebab)}{P(Akibat)}$$

$P(Akibat|Sebab)$ menyatakan hubungan kausal, sedangkan $P(Sebab|Akibat)$ menyatakan hubungan diagnostik.

Contoh: Dalam diagnosis medis, info ttg hubungan kausal $P(gejala|penyakit)$ biasanya diketahui, dan diagnosis dilakukan untuk menentukan $P(penyakit|gejala)$. Untuk menggunakan Bayes' rule, dokter juga memiliki informasi *prior probability* dari gejala yang diobservasi dan *prior probability* penyakit tersebut.

Penerapan Bayes' Rule

Contoh:
Dokter mengetahui penyakit *meningitis* menyebabkan 70% pasien mengalami leher kaku (*stiff neck*). Selain itu, dokter tahu fakta *unconditional*; probabilitas pasien menderita meningitis adalah 1/50,000, dan probabilitas pasien mengalami leher kaku adalah 1%. (s sebagai proposisi pasien mengalami leher kaku, m sebagai proposisi pasien menderita meningitis)

$$\begin{aligned} P(s|m) &= 0.7 \\ P(m) &= 1/50000 \\ P(s) &= 0.01 \\ P(m|s) &= \frac{P(s|m)P(m)}{P(s)} = \frac{0.7 \times 1/50000}{0.01} = 0.0014 \end{aligned}$$

Untuk mendapat distribusi $\mathbf{P}(M|s)$, yaitu $\langle P(m,s), P(\neg m, s)\rangle$, perhitungan $P(s)$ dapat digantikan dengan menghitung *conditional probability* untuk setiap nilai dari *query variable* m dan $\neg m$ lalu normalisasi hasilnya.
$\mathbf{P}(M|s) = \alpha\langle P(s|m)P(m), P(s|\neg m)P(\neg m)\rangle$

Bayes' rule dengan kombinasi evidence

- Seringkali ada banyak gejala (akibat) dari suatu penyakit (sebab). Mis. dokter ingin menghitung: $\mathbf{P}(Cavity|toothache \wedge catch)$
- Dengan tabel full joint probability distribution:
  $\mathbf{P}(Cavity|toothache \wedge catch) = \alpha\langle 0.108, 0.016\rangle \approx \langle 0.871, 0129\rangle$
  Masalah: eksponensial terhadap banyaknya variabel evidence.
- Dengan Bayes' rule:
  $\mathbf{P}(Cavity|toothache \wedge catch) = \alpha\mathbf{P}(toothache \wedge catch|Cavity)\mathbf{P}(Cavity)$
  Masalah: dibutuhkan *conditional probability* untuk kombinasi nilai variabel-variabel evidence (eksponensial terhadap banyaknya variabel).
- **Solusi**: conditional independence

Conditional Independence

- Mis. dalam kedokteran gigi, ratusan *variable* biasanya berhubungan $\rightarrow$ absolute independence jarang terjadi.
- Tapi hubungan itu bisa langsung atau tidak langsung!

Conditional independence

cavity mempengaruhi probabilitas *toothache*
cavity mempengaruhi probabilitas *catch*
Tetapi ini adalah dua efek yang terpisah!

- Nilai *Toothache* memberi kita info tentang nilai *Catch*, *secara tidak langsung* (lewat *Cavity*).
- Dkl, diberikan nilai *Cavity*, kedua variabel *Toothache* dan *Catch* independent:

  $$\mathbf{P}(toothache \land catch|Cavity) = \mathbf{P}(toothache|Cavity)\mathbf{P}(catch|Cavity)$$

Conditional independence - 1

Conditional Independence

Conditional independence dua variabel X dan Y, jika diketahui Z:
$\mathbf{P}(X, Y|Z) = \mathbf{P}(X|Z)\mathbf{P}(Y|Z)$

Conditional independence - 2

Perhatikan, *full joint distribution* bisa didekomposisi:

$$\mathbf{P}(Toothache, Catch, Cavity)$$
$$= \mathbf{P}(Toothache, Catch|Cavity)\ \mathbf{P}(Cavity)$$
$$= \mathbf{P}(Toothache|Cavity)\ \mathbf{P}(Catch|Cavity)\ \mathbf{P}(Cavity)$$

Pengaruh efisiensi

Scr. umum, *conditional independence* mengubah kompleksitas *joint probability distribution* dari **eksponensial** menjadi **linier** dalam jumlah variable.

Outline

Review

- Teori probabilitas: merangkum ketidakpastian sebagai suatu bilangan.
- Inference: menggunakan full joint probability distribution untuk menjawab berbagai masalah dalam domain.
  Tidak efisien! Kompleksitas untuk n variable: $O(2^n)$!
- Efisiensi: absolute independence & conditional independence. Mengurangi banyaknya nilai probabilitas yang perlu dihitung saat inference.
- Bayes' Rule: inference dengan conditional probability.

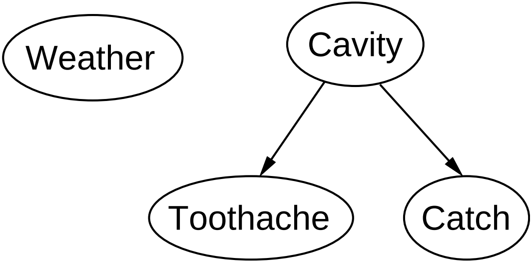
Outline

Bayesian Network

- Notasi graf yang menyatakan *conditional independence* dalam suatu domain.
- Node menyatakan sebuah *random variable*.
- Himpunan directed edge menghubungkan dua buah nodes untuk menyatakan hubungan pengaruh langsung (direct influence).
    - Jika A adalah parent dari B ($A \rightarrow B$), maka B dipengaruhi langsung oleh A (A has *direct influence* on B).
    - Node sibling menyatakan variable yang *conditionally independent* jika diketahui parent-nya.
- Conditional distribution untuk setiap node terhadap parent-nya: $\mathbf{P}(X_i | Parents(X_i))$
- Tidak ada *cycle* di dalam Bayesian Network (d.k.l. directed acyclic graph atau DAG)

Contoh kedokteran gigi

Topologi sebuah Bayesian Network menyatakan hubungan conditional independence:



- *Weather* absolute independent dari semua variable lain.
- *Toothache* dan *Catch* conditionally independent jika diketahui *Cavity*.

Contoh lain

Anto sedang di kantor. Tetangganya, John, menelepon mengatakan alarm anti-perampoknya bunyi. Tetangganya, Mary, tidak menelepon. Kadang-kadang alarmnya nyala karena gempa bumi. Apakah ada perampok di rumah Anto?

Contoh lain

Anto sedang di kantor. Tetangganya, John, menelepon mengatakan alarm anti-perampoknya bunyi. Tetangganya, Mary, tidak menelepon. Kadang-kadang alarmnya nyala karena gempa bumi. Apakah ada perampok di rumah Anto?

Variable dalam domain:

$Burglar$, $Earthquake$, $Alarm$, $JohnCalls$, $MaryCalls$

Contoh lain

Anto sedang di kantor. Tetangganya, John, menelepon mengatakan alarm anti-perampoknya bunyi. Tetangganya, Mary, tidak menelepon. Kadang-kadang alarmnya nyala karena gempa bumi. Apakah ada perampok di rumah Anto?

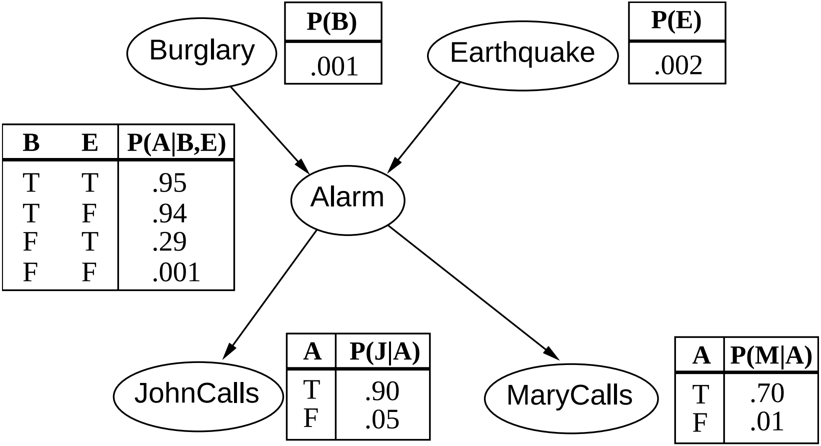
Variable dalam domain:

$Burglar$, $Earthquake$, $Alarm$, $JohnCalls$, $MaryCalls$

Hubungan sebab akibat:

- Perampok bisa membuat alarm nyala.
- Gempa bumi bisa membuat alarm nyala.
- Alarm bisa membuat John menelepon.
- Alarm bisa membuat Mary menelepon.

Contoh Bayesian Network



P(B)
.001

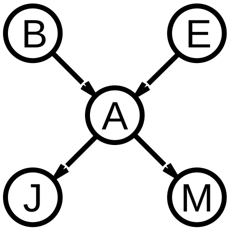
P(E)
.002

B	E	P(A|B,E)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J|A)
T	.90
F	.05

A	P(M|A)
T	.70
F	.01

Bayesian Network = ringkas



- Conditional probability sebuah node dgn. k parent $= O(2^k)$.
- Untuk n buah variable: $O(n \times 2^k)$
- Bandingkan dengan $O(2^n)$ untuk full joint distribution.

Outline

Memaknai Bayesian Network

Ada dua cara dalam memahami semantik Bayesian Network, yaitu:

1. sebagai representasi dari *full joint probability distribution*
   $\rightarrow$ membantu dalam membangun Bayesian Network
2. sebagai sekumpulan pernyataan *conditional independence*
   $\rightarrow$ membantu dalam merancang inference procedure

Merepresentasikan full joint distribution

- Bayesian Network adalah deskripsi lengkap sebuah domain.
- Full joint distribution bisa diperoleh dari local conditional distribution:
- $\mathbf{P}(X_1, \ldots, X_n) = \prod_{i=1}^{n} \mathbf{P}(X_i | Parents(X_i))$
- Contoh: hitung probabilitas John menelepon, Mary menelepon, alarm nyala, tidak ada perampok, tidak ada gempa bumi.
  $P(j \land m \land a \land \neg b \land \neg e) =$
  $P(j|a)P(m|a)P(a|\neg b, \neg e)P(\neg b)P(\neg e) =$
  $0.9 \times 0.7 \times 0.001 \times 0.999 \times 0.998 = 0.00062$

Chain rule & conditional independence

- Aturan chain rule:
  Product rule dlm conditional probability untuk himpunan random variables:
  $$\begin{aligned} P(x_1, \ldots, x_n) &= P(x_n | x_{n-1}, \ldots, x_1) P(x_{n-1}, \ldots, x_1) \\ &= P(x_n | x_{n-1}, \ldots, x_1) P(x_{n-1} | x_{n-2}, \ldots, x_1) \ldots P(x_2 | x_1) P(x_1) \\ &= \prod_{i=1}^{n} P(x_i | x_{i-1}, \ldots, x_1) \end{aligned}$$
- Pada Bayesian Network, untuk setiap variable X_i berlaku:
  $\mathbf{P}(X_i | X_{i-1}, \ldots, X_1) = \mathbf{P}(X_i | Parents(X_i))$
  dimana $Parents(X_i) \subseteq \{X_{i-1}, \ldots, X_1\}$
- Mis., berdasarkan chain rule:
  $\mathbf{P}(A, B, C, D) = \mathbf{P}(A | B, C, D) \mathbf{P}(B | C, D) \mathbf{P}(C | D) \mathbf{P}(D)$
  $\rightarrow$ Bayesian Network dengan urutan D, C, B, A tanpa conditional independence.
- Bagaimana jika, mis:
  A conditionally independent thd. B karena C dan D
  B conditionally independent thd. C karena D:
  $\mathbf{P}(A, B, C, D) = \mathbf{P}(A | C, D) \mathbf{P}(B | D) \mathbf{P}(C | D) \mathbf{P}(D)$

Membangun Bayesian Network

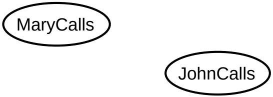
Membangun Bayesian Network dengan aturan chain rule:

Sebuah algoritma:

1. Pilih ordering variable $X_1, \ldots, X_n$, idealnya mulai dari *sebab* kemudian *akibat*
2. For $i = 1$ to n
   - Tambahkan X_i ke network
   - Pilih (himpunan minimal) parent X_i dari $X_1, \ldots, X_{i-1}$ shg. $\mathbf{P}(X_i | Parents(X_i)) = \mathbf{P}(X_i | X_1, \ldots, X_{i-1})$

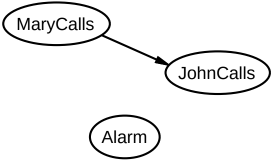
Contoh membangun Bayesian Network

Mis. kita pilih urutan: *MaryCalls*, *JohnCalls*, *Alarm*, *Burglary*, *Earthquake*.



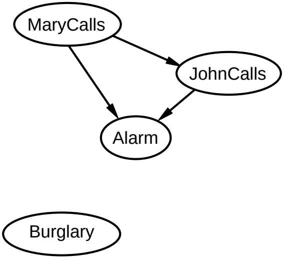
Contoh membangun Bayesian Network

Mis. kita pilih urutan: *MaryCalls*, *JohnCalls*, *Alarm*, *Burglary*, *Earthquake*.



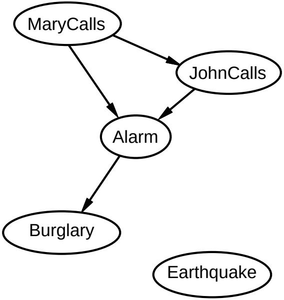
Contoh membangun Bayesian Network

Mis. kita pilih urutan: *MaryCalls*, *JohnCalls*, *Alarm*, *Burglary*, *Earthquake*.



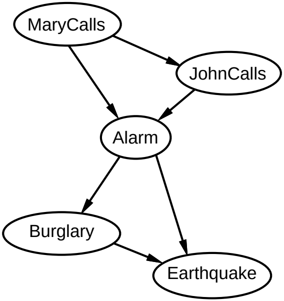
Contoh membangun Bayesian Network

Mis. kita pilih urutan: *MaryCalls*, *JohnCalls*, *Alarm*, *Burglary*, *Earthquake*.



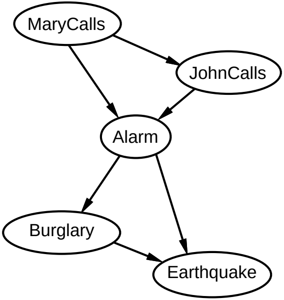
Contoh membangun Bayesian Network

Mis. kita pilih urutan: *MaryCalls*, *JohnCalls*, *Alarm*, *Burglary*, *Earthquake*.



Contoh membangun Bayesian Network

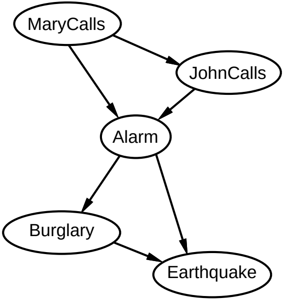
Mis. kita pilih urutan: *MaryCalls*, *JohnCalls*, *Alarm*, *Burglary*, *Earthquake*.



$P(J|M) = P(J)$?

Contoh membangun Bayesian Network

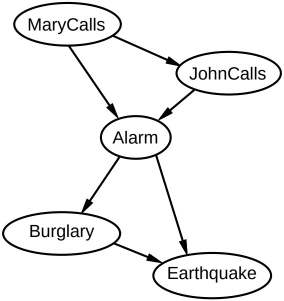
Mis. kita pilih urutan: *MaryCalls*, *JohnCalls*, *Alarm*, *Burglary*, *Earthquake*.



$P(J|M) = P(J)$? Tidak

Contoh membangun Bayesian Network

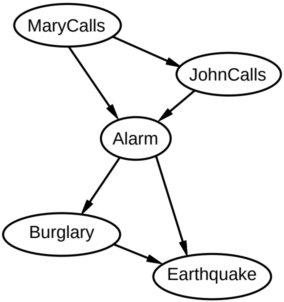
Mis. kita pilih urutan: *MaryCalls*, *JohnCalls*, *Alarm*, *Burglary*, *Earthquake*.



$P(J|M) = P(J)$? Tidak
$P(A|J,M) = P(A|J)$? $P(A|J,M) = P(A)$?

Contoh membangun Bayesian Network

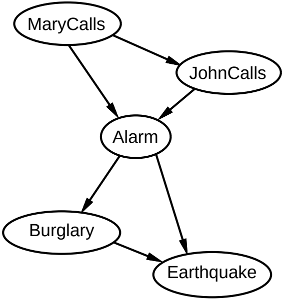
Mis. kita pilih urutan: *MaryCalls*, *JohnCalls*, *Alarm*, *Burglary*, *Earthquake*.



$P(J|M) = P(J)$? Tidak
$P(A|J,M) = P(A|J)$? $P(A|J,M) = P(A)$? Tidak

Contoh membangun Bayesian Network

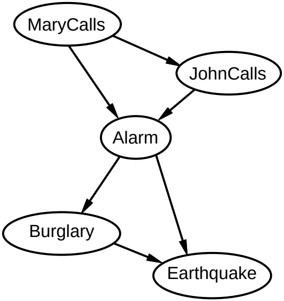
Mis. kita pilih urutan: *MaryCalls*, *JohnCalls*, *Alarm*, *Burglary*, *Earthquake*.



$P(J|M) = P(J)$? Tidak
$P(A|J, M) = P(A|J)$? $P(A|J, M) = P(A)$? Tidak
$P(B|A, J, M) = P(B|A)$?
$P(B|A, J, M) = P(B)$?

Contoh membangun Bayesian Network

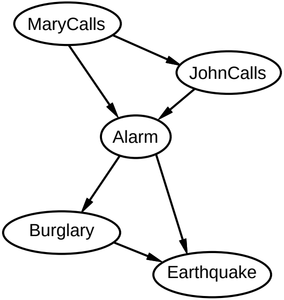
Mis. kita pilih urutan: *MaryCalls*, *JohnCalls*, *Alarm*, *Burglary*, *Earthquake*.



$P(J|M) = P(J)$? Tidak
$P(A|J,M) = P(A|J)$? $P(A|J,M) = P(A)$? Tidak
$P(B|A,J,M) = P(B|A)$? Ya
$P(B|A,J,M) = P(B)$? Tidak

Contoh membangun Bayesian Network

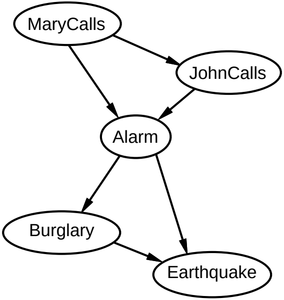
Mis. kita pilih urutan: *MaryCalls*, *JohnCalls*, *Alarm*, *Burglary*, *Earthquake*.



$P(J|M) = P(J)$? Tidak
$P(A|J,M) = P(A|J)$? $P(A|J,M) = P(A)$? Tidak
$P(B|A,J,M) = P(B|A)$? Ya
$P(B|A,J,M) = P(B)$? Tidak
$P(E|B,A,J,M) = P(E|A)$?

Contoh membangun Bayesian Network

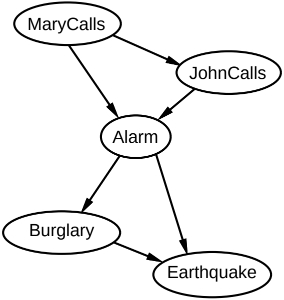
Mis. kita pilih urutan: *MaryCalls*, *JohnCalls*, *Alarm*, *Burglary*, *Earthquake*.



$P(J|M) = P(J)$? Tidak
$P(A|J,M) = P(A|J)$? $P(A|J,M) = P(A)$? Tidak
$P(B|A,J,M) = P(B|A)$? Ya
$P(B|A,J,M) = P(B)$? Tidak
$P(E|B,A,J,M) = P(E|A)$? Tidak

Contoh membangun Bayesian Network

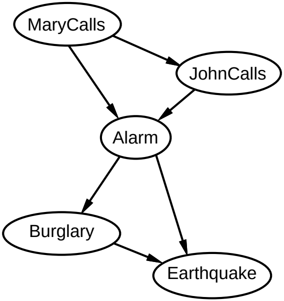
Mis. kita pilih urutan: *MaryCalls*, *JohnCalls*, *Alarm*, *Burglary*, *Earthquake*.



$P(J|M) = P(J)$? Tidak
$P(A|J, M) = P(A|J)$? $P(A|J, M) = P(A)$? Tidak
$P(B|A, J, M) = P(B|A)$? Ya
$P(B|A, J, M) = P(B)$? Tidak
$P(E|B, A, J, M) = P(E|A)$? Tidak
$P(E|B, A, J, M) = P(E|A, B)$?

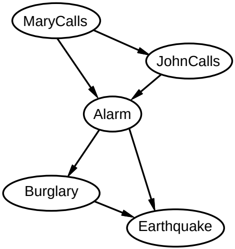
Contoh membangun Bayesian Network

Mis. kita pilih urutan: *MaryCalls*, *JohnCalls*, *Alarm*, *Burglary*, *Earthquake*.



$P(J|M) = P(J)$? Tidak
$P(A|J,M) = P(A|J)$? $P(A|J,M) = P(A)$? Tidak
$P(B|A,J,M) = P(B|A)$? Ya
$P(B|A,J,M) = P(B)$? Tidak
$P(E|B,A,J,M) = P(E|A)$? Tidak
$P(E|B,A,J,M) = P(E|A,B)$? Ya

Mulailah dari penyebab



- Menentukan conditional independence sulit untuk arah non-causal.
  - Akibatnya, ada penambahan *dependency* antara penyebab yang saling independent
- Menghitung conditional probability sulit untuk arah non-causal.
  - Lebih mudah menghitung nilai probability dari aturan kausal ketimbang diagnostik!
- *Conditional independence* lebih intuitif pada *causal model*!
- Kesimpulan: mulai dari variable "sebab" dulu.

Naive vs. paranoid...

Naive Bayes model

Semua variable akibat dianggap saling *conditionally independent* karena variable sebab.

$$\mathbf{P}(Sebab, Akibat_1, \ldots, Akibat_n) = \mathbf{P}(Sebab)\Pi_i\mathbf{P}(Akibat_i|Sebab)$$

Full joint distribution (paranoid?)

Semua random variable dianggap saling mempengaruhi.

Yang kita cari: analisis domain-specific yang menghasilkan informasi conditional independence yang benar!

Outline

Deterministic nodes

- *Conditional distribution* sebuah node dgn. k parent exponential dlm. k.
- Ada beberapa representasi yang lebih efisien $\rightarrow$ canonical distribution
- Conditional distribution dari suatu deterministic node bisa dihitung sepenuhnya dari nilai *parent*-nya. Contoh
    - Relasi logik: Sekumpulan *parent nodes*: *Canadian*, *US*, *Mexican* dan sebuah *child node*: *NorthAmerican*, maka *child node* adalah disjungsi semua *parent nodes*.
    - Relasi numerik: Sekumpulan *parent nodes*: harga suatu mobil jenis tertentu pada beberapa dealer dan sebuah *child node*: batas harga yang dapat ditawar pembeli, maka *child node* adalah minimum value dari *parent nodes*.

Deterministic nodes

- Representasi dari conditional distribution dapat dinyatakan dalam relasi **Noisy-OR**, yaitu generalisasi dari logical-OR.
- Sebagai contoh, dalam propositional logic, kita dapat nyatakan: $\mathit{Fever} \Leftrightarrow \mathit{Cold} \vee \mathit{Flu} \vee \mathit{Malaria}$
- Noisy-OR mengijinkan *ketidakpastian* pengaruh parent untuk menentukan nilai kebenaran child node. Contoh: Pasien mengalami *Cold*, tapi tidak berakibat ia mengalami *Fever*.
- Dalam model yang menggunakan Noisy-OR, terdapat dua asumsi:
  - semua kemungkinan penyebab diketahui (jika tidak, cukup tambahkan sebuah node berlabel "Lain-lain" disebut sebagai leak node)
  - pengaruh suatu parent terhadap child node bersifat independent terhadap parent lain dari child tersebut.

Noisy-OR Distribution

Contoh:

$P(\neg fever | cold, \neg flu, \neg malaria) = 0.6$
$P(\neg fever | \neg cold, flu, \neg malaria) = 0.2$
$P(\neg fever | \neg cold, \neg flu, malaria) = 0.1$

$$P(x_i | parents(X_i)) = 1 - \prod_{j:X_j = true} q_j$$

Cold	*Flu*	*Malaria*	$P(Fever)$	$P(\neg Fever)$
F	F	F	**0.0**	1.0
F	F	T	0.9	**0.1**
F	T	F	0.8	**0.2**
F	T	T	0.98	$0.02 = 0.2 \times 0.1$
T	F	F	0.4	**0.6**
T	F	T	0.94	$0.06 = 0.6 \times 0.1$
T	T	F	0.88	$0.12 = 0.6 \times 0.2$
T	T	T	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

Jumlah parameter linier thd. jumlah *parent*

Outline

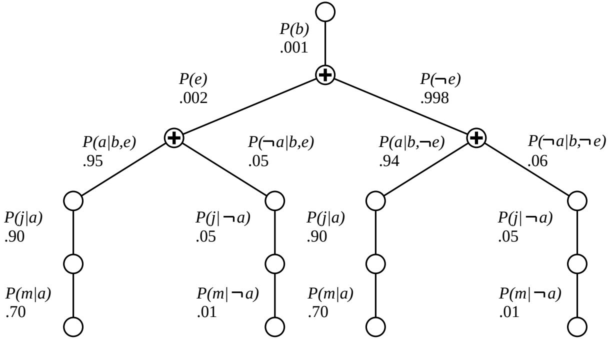
Inference by enumeration (1)

- Suatu *conditional probability* dapat dihitung dengan menjumlahkan semua variable dari *full joint distribution*.
- X menyatakan *query variable*; $\mathbf{E}$ menyatakan himpunan *evidence variable*, $\mathbf{e}$ merupakan event yang diobservasi; $\mathbf{Y}$ adalah *hidden variables*
$\mathbf{P}(X|\mathbf{e}) = \alpha\mathbf{P}(X, \mathbf{e}) = \alpha \sum_{\mathbf{y}} \mathbf{P}(X, \mathbf{e}, \mathbf{y})$
- Contoh: hitung probabilitas ada perampok jika John dan Mary menelepon.

$$\begin{aligned} P(b|j,m) &= \alpha \textstyle\sum_e \sum_a P(b)P(e)P(a|b,e)P(j|a)P(m|a) \\ &= \alpha P(b) \textstyle\sum_e P(e) \sum_a P(a|b,e)P(j|a)P(m|a) \end{aligned}$$

Inference by enumeration (2)

$$P(b|j,m) = \alpha P(b) \sum_e P(e) \sum_a P(a|b,e) P(j|a) P(m|a)$$



Perhatikan bahwa $P(j|a)P(m|a)$ dihitung untuk setiap nilai e. Gunakan dynamic programming: hitung sekali, simpan hasilnya!

Approximate inference

Pendekatan lain: jangan hitung nilai persis, tapi cukup di-*sample* (Monte Carlo).

Direct sampling methods

function PRIOR-SAMPLE(bn) **returns** an event sampled from bn
  inputs: bn, a Bayesian network specifying joint distribution $\mathbf{P}(X_1, \ldots, X_n)$

  $\mathbf{x} \leftarrow$ an event with n elements
  foreach variable X_i **in** $X_1, \ldots, X_n$ **do**
    $x[i] \leftarrow$ a random sample from $\mathbf{P}(X_i \mid Parents(X_i))$
  return x

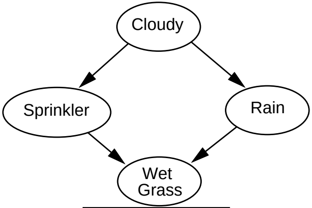
function REJECTION-SAMPLING($X, \mathbf{e}, bn, N$) **returns** an estimate of $P(X|\mathbf{e})$
  local variables: **N**, a vector of counts over X, initially zero

  for $j = 1$ to N **do**
    $\mathbf{x} \leftarrow$ PRIOR-SAMPLE(bn)
    if x is consistent with **e then**
      $\mathbf{N}[x] \leftarrow \mathbf{N}[x]+1$ where x is the value of X in **x**
  return NORMALIZE(**N**)

Contoh sampling dengan PRIOR-SAMPLE

Asumsi urutan variable yang diproses: [*Cloudy*, *Sprinkler*, *Rain*, *WetGrass*]

P(C)
.50



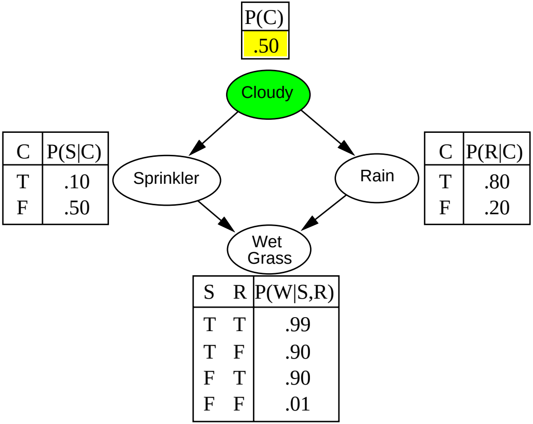
C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

Contoh sampling dengan PRIOR-SAMPLE

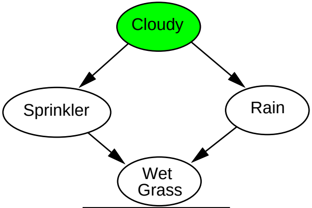
Asumsi urutan variable yang diproses: [*Cloudy*, *Sprinkler*, *Rain*, *WetGrass*]



Contoh sampling dengan PRIOR-SAMPLE

Asumsi urutan variable yang diproses: [*Cloudy*, *Sprinkler*, *Rain*, *WetGrass*]

P(C)
.50



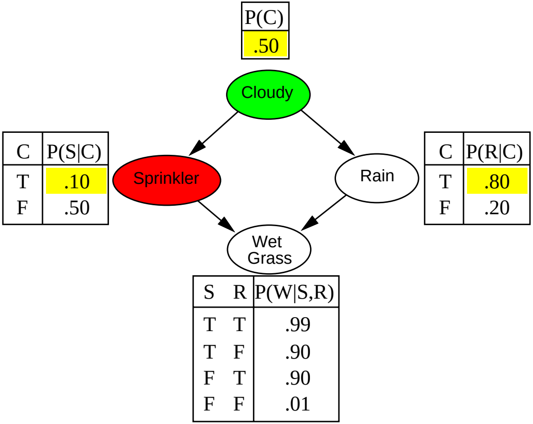
C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

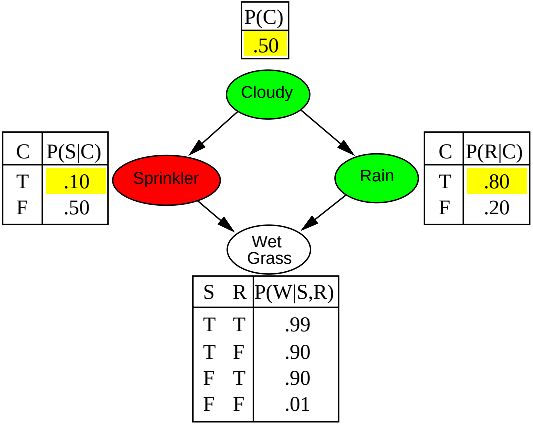
Contoh sampling dengan PRIOR-SAMPLE

Asumsi urutan variable yang diproses: [*Cloudy*, *Sprinkler*, *Rain*, *WetGrass*]



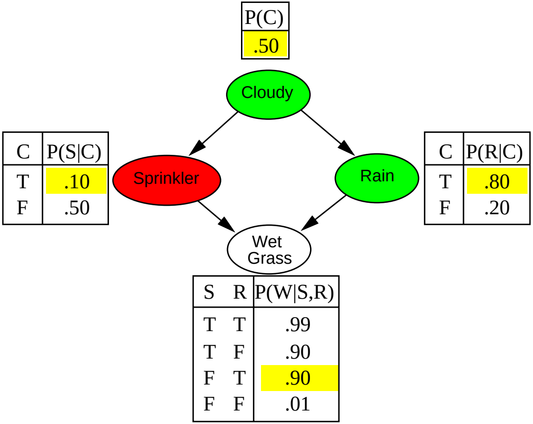
Contoh sampling dengan PRIOR-SAMPLE

Asumsi urutan variable yang diproses: [*Cloudy*, *Sprinkler*, *Rain*, *WetGrass*]



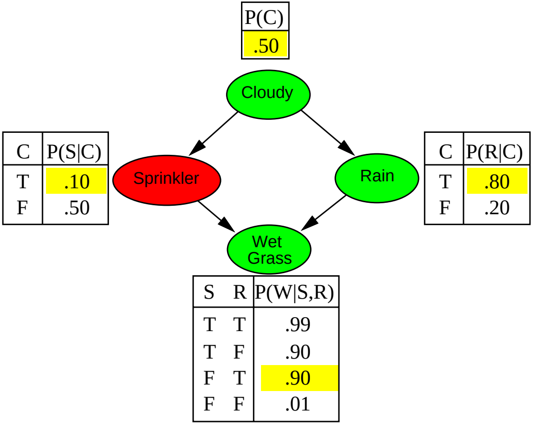
Contoh sampling dengan PRIOR-SAMPLE

Asumsi urutan variable yang diproses: [*Cloudy*, *Sprinkler*, *Rain*, *WetGrass*]



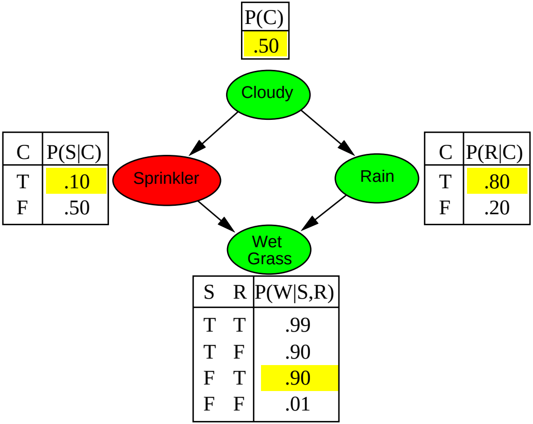
Contoh sampling dengan PRIOR-SAMPLE

Asumsi urutan variable yang diproses: [*Cloudy*, *Sprinkler*, *Rain*, *WetGrass*]



Contoh sampling dengan PRIOR-SAMPLE

Asumsi urutan variable yang diproses: [*Cloudy*, *Sprinkler*, *Rain*, *WetGrass*]



PRIOR-SAMPLE menghasilkan event: [*true*, *false*, *true*, *true*].

Contoh sampling dengan Rejection-Sampling

- Diasumsikan kita ingin mengestimasi distribusi $\mathbf{P}(Rain|Sprinkler = true)$ dalam 100 sample.
- Dari 100 kali sample, ternyata 73 kali menghasilkan $Sprinkler = false$, dan 27-nya $Sprinkler = true$.
- Hasil sample $Sprinkler = false \Rightarrow$ tidak konsisten. Rejected!
- Dari 27 sample yang menghasilkan $Sprinkler = true$ didapat 8 $Rain = true$ dan 19 $Rain = false$.
- $\mathbf{P}(Rain|Sprinkler = true) \approx Normalize(\langle 8, 19 \rangle) = \langle 0.296, 0.704 \rangle$

Ringkasan (1): Probability sebagai KRL

- Teori probabilitas adalah bahasa formal yang dapat merepresentasikan pengetahuan tidak pasti (uncertain knowledge).
- Nilai probabilitas menyatakan keadaan knowledge/belief sebuah agent.
- Sebuah joint probability distribution mendefinisikan prior probability untuk setiap atomic event.
- Inference dicapai dengan menjumlahkan nilai probabilitas.

Ringkasan (2): Probabilistic inference secara efisien

- Inference dengan *full joint distribution* konsepnya sangat mudah dimengerti, tetapi dalam kenyataan tidak feasible (*exponential time & space complexity*)
- Agar inference bisa *tractable*, kita mengambil asumsi **independence**.
- Dalam kenyataan, kita hanya bisa mengambil asumsi **conditional independence**.
- Bayes' Rule, ditambah dengan *conditional independence*, adalah mekanisme yang *sangat* berguna.

Ringkasan (3): Bayesian networks

- Bayes Net cocok utk. representasi (causal) conditional independence
- Mulai dari variable *penyebab*, tambahkan variable *akibat*
- Compact conditional distribution dengan representasi canonical
- Exact inference bisa dipercepat dengan dynamic programming
- Approximate inference bisa dilakukan dengan sampling