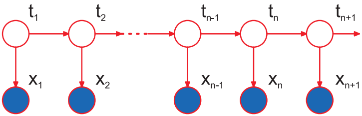
Лекция 5. Обучение без учителя скрытых марковских моделей и линейных динамических систем

Д. П. Ветров[1]

[1]МГУ, ВМиК, каф. ММП

Спецкурс «Графические модели»

Скрытая Марковская модель (СММ)



Скрытая Марковская модель [первого порядка] — это вероятностная модель последовательности, которая

- Состоит из набора наблюдаемых переменных $X = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\}$ и латентных (скрытых) переменных $T = \{\boldsymbol{t}_1, \dots, \boldsymbol{t}_N\}$.
- Латентные переменные T являются дискретными, поэтому их иногда называют переменными состояния.
- Значение наблюдаемого вектора в момент времени n $\boldsymbol{x}_n$ зависит только от скрытого состояния $\boldsymbol{t}_n$, которое в свою очередь зависит только от скрытого состояния в предыдущий момент времени $\boldsymbol{t}_{n-1}$.

Спецификация вероятностной модели I

Пусть имеется K возможных состояний. Закодируем состояние в каждый момент времени n бинарным вектором $\boldsymbol{t}_n = (t_{n1}, \dots, t_{nK})$, где

$$t_{nj} = \begin{cases} 1, & \text{если в момент } n \text{ модель находится в состоянии } j \\ 0, & \text{иначе} \end{cases}$$

Тогда распределение $p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1})$ можно задать матрицей перехода A размера $K \times K$, где $A_{ij} = p(t_{nj} = 1|t_{n-1,i} = 1)$, $\sum_j A_{ij} = 1$, т.е.

$$p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1}) = \prod_{i=1}^{K}\prod_{j=1}^{K} A_{ij}^{t_{n-1,i}t_{nj}}$$

Пусть в первый момент времени $p(t_{1j} = 1) = \pi_j$. Тогда

$$p(\boldsymbol{t}_1) = \prod_{j=1}^{K} \pi_j^{t_{1j}}$$

Спецификация вероятностной модели II

Пусть для каждого состояния $k \in \{1, \dots, K\}$ в момент времени n известна модель генерации наблюдаемых данных $\boldsymbol{x}_n$ $p(\boldsymbol{x}_n|\boldsymbol{\phi}_k)$, задаваемая с помощью набора параметров $\boldsymbol{\phi}_k$. Тогда

$$p(\boldsymbol{x}_n|\boldsymbol{t}_n) = \prod_{j=1}^{K} (p(\boldsymbol{x}_n|\boldsymbol{\phi}_k))^{t_{nk}}$$

Обозначим полный набор параметров СММ через $\Theta = \{\boldsymbol{\pi}, A, \boldsymbol{\phi}\}$. Тогда правдоподобие в СММ вычисляется как

$$p(X, T|\Theta) = p(\boldsymbol{t}_{1,\boldsymbol{\pi}}) \prod_{n=2}^{N} p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1,A}) \prod_{n=1}^{N} p(\boldsymbol{x}_n|\boldsymbol{t}_{n,\boldsymbol{\phi}}) =$$

$$= \prod_{j=1}^{K} \pi_j^{t_{1j}} \left(\prod_{n=2}^{N} \prod_{i=1}^{K} \prod_{j=1}^{K} A_{ij}^{t_{n-1,i} t_{nj}} \right) \left(\prod_{n=1}^{N} \prod_{k=1}^{K} (p(\boldsymbol{x}_n|\boldsymbol{\phi}_k))^{t_{nk}} \right)$$

Задачи в СММ

- Распознавание (предыдущая лекция). Известна некоторая последовательность X и набор параметров Θ. Задача состоит в получении наиболее правдоподобной последовательности состояний T как $\arg\max_T p(T|X,\Theta)$ (алгоритм Витерби).
- Обучение с учителем (предыдущая лекция). Известна некоторая последовательность X, для которой заданы T. Задача состоит в оценке по обучающей выборке набора параметров Θ.
- Обучение без учителя. Известна некоторая последовательность X и число состояний K. Задача состоит в оценке параметров Θ (EM-алгоритм).
  - Нахождение маргинального распределения $p(\boldsymbol{t}_n|X,\Theta)$ компоненты $\boldsymbol{t}_n$ по заданым X и Θ
- Прогнозирование. Известна некоторая последовательность X. Задача состоит в оценке наблюдаемого вектора в следующий момент времени $N+1$ — $p(\boldsymbol{x}_{N+1}|X)$.

План лекции

1. Метод максимального правдоподобия для СММ
2. Алгоритм «вперед-назад»
3. Устойчивые формулы для алгоритма «вперед-назад»
4. Модельный пример
5. Модификации стандартной СММ
6. Линейные динамические системы
7. Вывод в ЛДС
8. Расширенный фильтр Калмана
9. Обучение в ЛДС

План лекции

Метод максимального правдоподобия

Для оценки параметров СММ Θ воспользуемся методом максимального правдоподобия:

$$\Theta_* = \arg\max_{\Theta} p(X|\Theta) = \arg\max_{\Theta} \sum_{T} p(X, T|\Theta)$$

Прямая максимизация правдоподобия затруднительна, т.к. оптимизируемая функция не является выпуклой и, кроме того, для вычисления функции требуется суммирование N^K слагаемых. Можно воспользоваться итерационным EM-алгоритмом.

ЕМ-алгоритм в общем виде

Требуется найти максимум правдоподобия в вероятностной модели со скрытыми переменными:

$$p(X|\Theta) = \sum_T p(X,T|\Theta) \to \max_\Theta \Leftrightarrow \log\left(\sum_T p(X,T|\Theta)\right) \to \max_\Theta$$

- Е-шаг. Фиксируется значение параметров Θ_{old}. Оценивается апостериорное распределение на скрытые переменные $p(T|X,\Theta_{old})$, и полное правдоподобие усредняется по полученному распределению:

$$\mathbb{E}_{T|X,\Theta_{old}} \log p(X,T|\Theta) = \sum_T \log p(X,T|\Theta) p(T|X,\Theta_{old})$$

- М-шаг. Фиксируется апостериорное распределение $p(T|X,\Theta_{old})$, и производится поиск новых значений параметров Θ_{new}:

$$\Theta_{new} = \arg\max_\Theta \mathbb{E}_{T|X,\Theta_{old}} \log p(X,T|\Theta)$$

- Шаги Е и М повторяются до сходимости.

Е-шаг

Обозначим

$$\gamma(t_{nj}) = \mathbb{E}_{T|X,\Theta} t_{nj} = p(t_{nj} = 1|X, \Theta_{old}),$$
$$\xi(t_{n-1,i}, t_{nj}) = \mathbb{E}_{T|X,\Theta}(t_{n-1,i}t_{nj}) = p(t_{n-1,i} = 1, t_{nj} = 1|X, \Theta_{old}).$$

Тогда

$$\log p(X, T|\Theta) = \sum_{j=1}^{K} t_{1j} \log \pi_j + \sum_{n=2}^{N}\sum_{i=1}^{K}\sum_{j=1}^{K} t_{n-1,i}t_{nj} \log A_{ij} + \sum_{n=1}^{N}\sum_{j=1}^{K} t_{nj} \log p(\boldsymbol{x}_n|\phi_k)$$

$$\mathbb{E}_{T|X,\Theta_{old}} \log p(X, T|\Theta) = \sum_{j=1}^{K} \gamma(t_{1j}) \log \pi_j + \sum_{n=2}^{N}\sum_{i=1}^{K}\sum_{j=1}^{K} \xi(t_{n-1,i}, t_{nj}) \log A_{ij} + \sum_{n=1}^{N}\sum_{j=1}^{K} \gamma(t_{nj}) \log p(\boldsymbol{x}_n|\phi_k)$$

М-шаг

$$\boldsymbol{\pi}: \quad \sum_{j=1}^{K} \gamma(t_{1j}) \log \pi_j + \lambda \left(\sum_{j=1}^{K} \pi_j - 1 \right) \to \operatorname*{extr}_{\boldsymbol{\pi},\lambda}$$

$$\frac{\gamma(t_{1j})}{\pi_j} + \lambda = 0 \Rightarrow \pi_j = -\frac{\gamma(t_{1j})}{\lambda}$$

$$\sum_{j=1}^{K} \pi_j = 1 \Rightarrow \lambda = -\sum_{i=1}^{K} \gamma(t_{1i})$$

$$\pi_j^{new} = \frac{\gamma(t_{1j})}{\sum_{i=1}^{K} \gamma(t_{1i})}$$

Действуя аналогично для A, принимая во внимание, что $\sum_{j=1}^{K} A_{ij} = 1 \; \forall \; i$, получаем:

$$A_{ij}^{new} = \frac{\sum_{n=2}^{N} \xi(t_{n-1,i} t_{nj})}{\sum_{k=1}^{K} \sum_{n=2}^{N} \xi(t_{n-1,i} t_{nk})}$$

M-шаг для компонент ϕ

M-шаг для компонент генерации данных $p(\boldsymbol{x}_n|\phi_k)$ абсолютно аналогичен M-шагу для оценки параметров при восстановлении смесей распределений. В частности, если в качестве компонент выступают многомерные нормальные распределения

$$p(\boldsymbol{x}_n|\phi_k) = \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_k, \Sigma_k),$$

то задача оптимизации для параметров $\boldsymbol{\mu}_k, \Sigma_k$ может быть решена в явном виде:

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^N \gamma(t_{nk})\boldsymbol{x}_n}{\sum_{n=1}^N \gamma(t_{nk})}$$

$$\Sigma_k = \frac{\sum_{n=1}^N \gamma(t_{nk})(\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^T}{\sum_{n=1}^N \gamma(t_{nk})}$$

Инициализация параметров Θ

Для начала работы EM-алгоритма необходимо задать начальные значения параметров $\Theta = (\boldsymbol{\pi}, A, \boldsymbol{\phi})$. Заметим, что если какой-нибудь параметр инициализирован нулем, то в процессе итераций EM его значение не изменится.

- Значения параметров $\boldsymbol{\pi}$ и A обычно выбираются случайными при соблюдении ограничений $\sum_j \pi_j = 1$ и $\sum_j A_{ij} = 1\ \forall i$.
- Инициализация $\boldsymbol{\phi}$ зависит от формы распределений $p(\mathbf{x}|\boldsymbol{\phi})$. В случае нормальных распределений можно провести кластеризацию данных на K кластеров и выбрать в качестве $\boldsymbol{\mu}_k$ и Σ_k центр и разброс соответствующего кластера.

План лекции

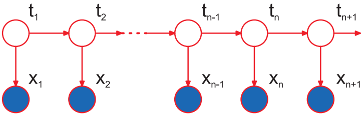
Вычисление апостериорных распределений на скрытые компоненты

На Е-шаге алгоритма обучения СММ требуется вычисление апостериорных распределений на скрытые компоненты

$$\gamma(t_{nj}) = p(t_{nj} = 1|X, \Theta), \quad \xi(t_{n-1,i}, t_{nj}) = p(t_{n-1,i} = 1, t_{nj} = 1|X, \Theta)$$

Алгоритм «вперед-назад» (Баума-Уэлша) позволяет эффективно вычислять эти величины для всех n, i, j за линейное по N время.
В дальнейшем для удобства будем опускать Θ во всех формулах, считая набор параметров фиксированным.

Свойства условной независимости для СММ



$$
\begin{aligned}
p(X|\boldsymbol{t}_n) &= p(\boldsymbol{x}_1,\dots,\boldsymbol{x}_n|\boldsymbol{t}_n)p(\boldsymbol{x}_{n+1},\dots,\boldsymbol{x}_N|\boldsymbol{t}_n)\\
p(\boldsymbol{x}_1,\dots,\boldsymbol{x}_{n-1}|\boldsymbol{x}_n,\boldsymbol{t}_n) &= p(\boldsymbol{x}_1,\dots,\boldsymbol{x}_{n-1}|\boldsymbol{t}_n)\\
p(\boldsymbol{x}_1,\dots,\boldsymbol{x}_{n-1}|\boldsymbol{t}_{n-1},\boldsymbol{t}_n) &= p(\boldsymbol{x}_1,\dots,\boldsymbol{x}_{n-1}|\boldsymbol{t}_{n-1})\\
p(\boldsymbol{x}_{n+1},\dots,\boldsymbol{x}_N|\boldsymbol{t}_n,\boldsymbol{t}_{n+1}) &= p(\boldsymbol{x}_{n+1},\dots,\boldsymbol{x}_N|\boldsymbol{t}_{n+1})\\
p(\boldsymbol{x}_{n+2},\dots,\boldsymbol{x}_N|\boldsymbol{t}_{n+1},\boldsymbol{x}_{n+1}) &= p(\boldsymbol{x}_{n+2},\dots,\boldsymbol{x}_N|\boldsymbol{t}_{n+1})\\
p(X|\boldsymbol{t}_{n-1},\boldsymbol{t}_n) &= p(\boldsymbol{x}_1,\dots,\boldsymbol{x}_{n-1}|\boldsymbol{t}_{n-1})p(\boldsymbol{x}_n|\boldsymbol{t}_n)\times\\
&\quad\times p(\boldsymbol{x}_{n+1},\dots,\boldsymbol{x}_N|\boldsymbol{t}_n)\\
p(\boldsymbol{x}_{N+1}|X,\boldsymbol{t}_{N+1}) &= p(\boldsymbol{x}_{N+1}|\boldsymbol{t}_{N+1})\\
p(\boldsymbol{t}_{N+1}|\boldsymbol{t}_N,X) &= p(\boldsymbol{t}_{N+1}|\boldsymbol{t}_N)
\end{aligned}
$$

Вычисление $\gamma(t_{nj})$

По формуле Байеса

$$\gamma(\boldsymbol{t}_n) = p(\boldsymbol{t}_n|X) = \frac{p(X|\boldsymbol{t}_n)p(\boldsymbol{t}_n)}{p(X)}$$

Пользуясь свойствами условной независимости для СММ, получаем

$$\gamma(\boldsymbol{t}_n) = \frac{p(\boldsymbol{x}_1, \dots, \boldsymbol{x}_n, \boldsymbol{t}_n)p(\boldsymbol{x}_{n+1}, \dots, \boldsymbol{x}_N|\boldsymbol{t}_n)}{p(X)} = \frac{\boldsymbol{\alpha}(\boldsymbol{t}_n)\boldsymbol{\beta}(\boldsymbol{t}_n)}{p(X)}$$

Здесь

$$\boldsymbol{\alpha}(\boldsymbol{t}_n) = p(\boldsymbol{x}_1, \dots, \boldsymbol{x}_n, \boldsymbol{t}_n)$$
$$\boldsymbol{\beta}(\boldsymbol{t}_n) = p(\boldsymbol{x}_{n+1}, \dots, \boldsymbol{x}_N|\boldsymbol{t}_n)$$

Алгоритм «вперед-назад» позволяет быстро рекуррентно вычислять значение $\boldsymbol{\alpha}(\boldsymbol{t}_n)$ через $\boldsymbol{\alpha}(\boldsymbol{t}_{n-1})$ (проход вперед) и $\boldsymbol{\beta}(\boldsymbol{t}_n)$ через $\boldsymbol{\beta}(\boldsymbol{t}_{n+1})$ (проход назад).

Рекуррентная формула для $\boldsymbol{\alpha}(\boldsymbol{t}_n)$

$$
\begin{aligned}
\boldsymbol{\alpha}(\boldsymbol{t}_n) =& p(\boldsymbol{x}_1,\dots,\boldsymbol{x}_n,\boldsymbol{t}_n) = \\
=& p(\boldsymbol{x}_1,\dots,\boldsymbol{x}_n|\boldsymbol{t}_n)p(\boldsymbol{t}_n) = \\
=& p(\boldsymbol{x}_n|\boldsymbol{t}_n)p(\boldsymbol{x}_1,\dots,\boldsymbol{x}_{n-1}|\boldsymbol{t}_n)p(\boldsymbol{t}_n) = \\
=& p(\boldsymbol{x}_n|\boldsymbol{t}_n)p(\boldsymbol{x}_1,\dots,\boldsymbol{x}_{n-1},\boldsymbol{t}_n) = \\
=& p(\boldsymbol{x}_n|\boldsymbol{t}_n)\sum_{\boldsymbol{t}_{n-1}} p(\boldsymbol{x}_1,\dots,\boldsymbol{x}_{n-1},\boldsymbol{t}_{n-1},\boldsymbol{t}_n) = \\
=& p(\boldsymbol{x}_n|\boldsymbol{t}_n)\sum_{\boldsymbol{t}_{n-1}} p(\boldsymbol{x}_1,\dots,\boldsymbol{x}_{n-1},\boldsymbol{t}_n|\boldsymbol{t}_{n-1})p(\boldsymbol{t}_{n-1}) = \\
=& p(\boldsymbol{x}_n|\boldsymbol{t}_n)\sum_{\boldsymbol{t}_{n-1}} p(\boldsymbol{x}_1,\dots,\boldsymbol{x}_{n-1}|\boldsymbol{t}_{n-1})p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1})p(\boldsymbol{t}_{n-1}) = \\
=& p(\boldsymbol{x}_n|\boldsymbol{t}_n)\sum_{\boldsymbol{t}_{n-1}} p(\boldsymbol{x}_1,\dots,\boldsymbol{x}_{n-1},\boldsymbol{t}_{n-1})p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1}) = \\
=& p(\boldsymbol{x}_n|\boldsymbol{t}_n)\sum_{\boldsymbol{t}_{n-1}} \boldsymbol{\alpha}(\boldsymbol{t}_{n-1})p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1})
\end{aligned}
$$

Вычисление $\boldsymbol{\alpha}(\boldsymbol{t}_n)$

Для запуска рекуррентного процесса необходимо вычислить

$$\boldsymbol{\alpha}(\boldsymbol{t}_1) = p(\boldsymbol{x}_1, \boldsymbol{t}_1) = p(\boldsymbol{t}_1)p(\boldsymbol{x}_1|\boldsymbol{t}_1) = \prod_{j=1}^{K}(\pi_j p(\boldsymbol{x}_1|\phi_k))^{t_{1j}}$$

На каждом шаге рекурсии вектор $\boldsymbol{\alpha}(\boldsymbol{t}_n)$ длины K умножается на матрицу $p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1})$ размера $K \times K$. Поэтому сложность всего рекуррентного процесса составляет $O(NK^2)$.

Рекуррентная формула для $\beta(\boldsymbol{t}_n)$

$$
\begin{aligned}
\beta(\boldsymbol{t}_n) =& p(\boldsymbol{x}_{n+1},\dots,\boldsymbol{x}_N|\boldsymbol{t}_n) = \\
=& \sum_{\boldsymbol{t}_{n+1}} p(\boldsymbol{x}_{n+1},\dots,\boldsymbol{x}_N,\boldsymbol{t}_{n+1}|\boldsymbol{t}_n) = \\
=& \sum_{\boldsymbol{t}_{n+1}} p(\boldsymbol{x}_{n+1},\dots,\boldsymbol{x}_N|\boldsymbol{t}_{n+1})p(\boldsymbol{t}_{n+1}|\boldsymbol{t}_n) = \\
=& \sum_{\boldsymbol{t}_{n+1}} p(\boldsymbol{x}_{n+2},\dots,\boldsymbol{x}_N|\boldsymbol{t}_{n+1})p(\boldsymbol{x}_{n+1}|\boldsymbol{t}_{n+1})p(\boldsymbol{t}_{n+1}|\boldsymbol{t}_n) = \\
=& \sum_{\boldsymbol{t}_{n+1}} \beta(\boldsymbol{t}_{n+1})p(\boldsymbol{x}_{n+1}|\boldsymbol{t}_{n+1})p(\boldsymbol{t}_{n+1}|\boldsymbol{t}_n)
\end{aligned}
$$

Инициализация рекуррентного процесса для $\beta(t_n)$

Вспомним формулу, связывающую значение $\gamma(t_n)$ с $\alpha(t_n)$ и $\beta(t_n)$:

$$\gamma(t_n) = \frac{p(x_1, \dots, x_n, t_n) p(x_{n+1}, \dots, x_N | t_n)}{p(X)} = \frac{\alpha(t_n)\beta(t_n)}{p(X)}$$

Подставляя в формулу значение $n = N$, получаем

$$\gamma(t_N) = p(t_N | X) = \frac{p(X, t_N)\beta(t_N)}{p(X)}$$

Таким образом, $\beta(t_N) = 1$.

Нормировочная константа $p(X)$

Значение $\boldsymbol{\gamma}(\boldsymbol{t}_n)$ определено с точностью до нормировочной константы $p(X)$. Однако, на М-шаге значение $\boldsymbol{\gamma}(\boldsymbol{t}_n)$, как правило, входит в числитель и в знаменатель. Таким образом, нормировочная константа сокращается. Например, при вычислении центра нормального распределения:

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^N \gamma(t_{nk})\boldsymbol{x}_n}{\sum_{n=1}^N \gamma(t_{nk})} = \frac{\sum_{n=1}^N \alpha(t_{nk})\beta(t_{nk})\boldsymbol{x}_k}{\sum_{n=1}^N \alpha(t_{nk})\beta(t_{nk})}$$

Тем не менее, сама нормировочная константа $p(X)$ — это значение правдоподобия, которое может представлять отдельный интерес (например, можно отслеживать возрастание правдоподобия при итерациях ЕМ-алгоритма). Зная $\boldsymbol{\alpha}(\boldsymbol{t}_n)$ и $\boldsymbol{\beta}(\boldsymbol{t}_n)$, правдоподобие может быть легко вычислено как

$$p(X) = \sum_{\boldsymbol{t}_n} \boldsymbol{\alpha}(\boldsymbol{t}_n)\boldsymbol{\beta}(\boldsymbol{t}_n), \ \forall n \ \Rightarrow \ p(X) = \sum_{\boldsymbol{t}_N} \boldsymbol{\alpha}(\boldsymbol{t}_N)$$

Формула для $\xi(\boldsymbol{t}_{n-1}, \boldsymbol{t}_n)$

$$
\begin{aligned}
\xi(\boldsymbol{t}_{n-1}, \boldsymbol{t}_n) &= p(\boldsymbol{t}_{n-1}, \boldsymbol{t}_n | X) = \\
&= \frac{p(X|\boldsymbol{t}_{n-1}, \boldsymbol{t}_n) p(\boldsymbol{t}_{n-1}, \boldsymbol{t}_n)}{p(X)} = \\
&= \frac{p(\boldsymbol{x}_1, \dots, \boldsymbol{x}_{n-1}|\boldsymbol{t}_{n-1}) p(\boldsymbol{x}_n|\boldsymbol{t}_n) p(\boldsymbol{x}_{n+1}, \dots, \boldsymbol{x}_N|\boldsymbol{t}_n) p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1}) p(\boldsymbol{t}_{n-1})}{p(X)} = \\
&= \frac{\boldsymbol{\alpha}(\boldsymbol{t}_{n-1}) p(\boldsymbol{x}_n|\boldsymbol{t}_n) p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1}) \boldsymbol{\beta}(\boldsymbol{t}_n)}{p(X)}
\end{aligned}
$$

Итоговый EM-алгоритм для случая, когда $p(\mathbf{x}_n|\mathbf{t}_n)$ — нормальные распределения

- Начальная инициализация параметров $\Theta = (\boldsymbol{\pi}, A, \boldsymbol{\phi})$. Параметры $\boldsymbol{\pi}$ и A можно инициализировать случайно, а $\boldsymbol{\phi}$ с помощью кластеризации данных.
- E-шаг. Вычисление $\boldsymbol{\alpha}(\mathbf{t}_n)$ и $\boldsymbol{\beta}(\mathbf{t}_n)$ с помощью рекуррентного алгоритма «вперед-назад». Вычисление величин $\gamma(t_{nj})$ и $\xi(t_{n-1,i}, t_{nj})$ и, возможно, правдоподобия $p(X)$.
- M-шаг. Вычисление новых значений параметров:

$$\pi_j^{new} = \frac{\gamma(t_{1j})}{\sum_{i=1}^K \gamma(t_{1i})}, \qquad A_{ij}^{new} = \frac{\sum_{n=2}^N \xi(t_{n-1,i}t_{nj})}{\sum_{k=1}^K \sum_{n=2}^N \xi(t_{n-1,i}t_{nk})}$$

$$\boldsymbol{\mu}_k^{new} = \frac{\sum_{n=1}^N \gamma(t_{nk})\mathbf{x}_n}{\sum_{n=1}^N \gamma(t_{nk})}, \quad \Sigma_k^{new} = \frac{\sum_{n=1}^N \gamma(t_{nk})(\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T}{\sum_{n=1}^N \gamma(t_{nk})}$$

- Повторять шаги E и M до сходимости (пока значение $p(X)$ или Θ не стабилизируется).

Задача прогнозирования

$$p(\boldsymbol{x}_{N+1}|X) = \sum_{\boldsymbol{t}_{N+1}} p(\boldsymbol{x}_{N+1}, \boldsymbol{t}_{N+1}|X) = \sum_{\boldsymbol{t}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{t}_{N+1}) p(\boldsymbol{t}_{N+1}|X) =$$

$$= \sum_{\boldsymbol{t}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{t}_{N+1}) \left(\sum_{\boldsymbol{t}_N} p(\boldsymbol{t}_{N+1}, \boldsymbol{t}_N|X) \right) =$$

$$= \sum_{\boldsymbol{t}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{t}_{N+1}) \left(\sum_{\boldsymbol{t}_N} p(\boldsymbol{t}_N|X) p(\boldsymbol{t}_{N+1}|\boldsymbol{t}_N) \right) =$$

$$= \sum_{\boldsymbol{t}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{t}_{N+1}) \left(\sum_{\boldsymbol{t}_N} p(\boldsymbol{t}_{N+1}|\boldsymbol{t}_N) \frac{p(\boldsymbol{t}_N, X)}{p(X)} \right) =$$

$$= \frac{1}{p(X)} \sum_{\boldsymbol{t}_{N+1}} p(\boldsymbol{x}_{N+1}|\boldsymbol{t}_{N+1}) \left(\sum_{\boldsymbol{t}_N} p(\boldsymbol{t}_{N+1}|\boldsymbol{t}_N) \boldsymbol{\alpha}(\boldsymbol{t}_N) \right)$$

Это фактически смесь распределений с компонентами $p(\boldsymbol{x}_{N+1}|\boldsymbol{t}_{N+1})$ и весами $w_k = \frac{1}{p(X)} \sum_{\boldsymbol{t}_N} p(\boldsymbol{t}_{N+1}|\boldsymbol{t}_N) \boldsymbol{\alpha}(\boldsymbol{t}_N)$. Для получения точечного прогноза $\boldsymbol{x}_{N+1}^*$ можно воспользоваться MCMC.

План лекции

Необходимость устойчивых вычислений для $\boldsymbol{\alpha}(\boldsymbol{t}_n)$ и $\boldsymbol{\beta}(\boldsymbol{t}_n)$

Формулы пересчета для $\boldsymbol{\alpha}(\boldsymbol{t}_n)$ и $\boldsymbol{\beta}(\boldsymbol{t}_n)$:

$$\boldsymbol{\alpha}(\boldsymbol{t}_n) = p(\boldsymbol{x}_n|\boldsymbol{t}_n)\sum_{\boldsymbol{t}_{n-1}}\boldsymbol{\alpha}(\boldsymbol{t}_{n-1})p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1})$$

$$\boldsymbol{\beta}(\boldsymbol{t}_n) = \sum_{\boldsymbol{t}_{n+1}}\boldsymbol{\beta}(\boldsymbol{t}_{n+1})p(\boldsymbol{t}_{n+1}|\boldsymbol{t}_n)p(\boldsymbol{x}_{n+1}|\boldsymbol{t}_{n+1})$$

На практике значения вероятностей $p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1})$ и $p(\boldsymbol{x}_n|\boldsymbol{t}_n)$ могут быть существенно меньше единицы. В процессе пересчета эти вероятности умножаются друг на друга, и получающиеся значения перестают укладываться в машинную точность.

Устойчивые формулы для $\boldsymbol{\alpha}(\boldsymbol{t}_n)$ I

Предлагается вместо $\boldsymbol{\alpha}(\boldsymbol{t}_n)$ рассмотреть следующую величину:

$$\hat{\boldsymbol{\alpha}}(\boldsymbol{t}_n) = p(\boldsymbol{t}_n|\boldsymbol{x}_1, \dots, \boldsymbol{x}_n) = \frac{\boldsymbol{\alpha}(\boldsymbol{t}_n)}{p(\boldsymbol{x}_1, \dots, \boldsymbol{x}_n)}$$

Можно надеяться, что значения $\hat{\boldsymbol{\alpha}}(\boldsymbol{t}_n)$ будут существенно отличны от нуля, т.к. $\sum_{\boldsymbol{t}_n} \hat{\boldsymbol{\alpha}}(\boldsymbol{t}_n) = 1$.
Рассмотрим также величины

$$c_n = p(\boldsymbol{x}_n|\boldsymbol{x}_{n-1}, \dots, \boldsymbol{x}_1).$$

Вычисление этих величин также будет устойчивым, т.к. они имеют смысл одномерных вероятностных распределений.

Устойчивые формулы для $\boldsymbol{\alpha}(\boldsymbol{t}_n)$ II

Очевидно, что

$$p(\boldsymbol{x}_1, \dots, \boldsymbol{x}_n) = \prod_{i=1}^{n} c_i$$

$$\boldsymbol{\alpha}(\boldsymbol{t}_n) = p(\boldsymbol{t}_n | \boldsymbol{x}_1, \dots, \boldsymbol{x}_n) p(\boldsymbol{x}_1, \dots, \boldsymbol{x}_n) = \hat{\boldsymbol{\alpha}}(\boldsymbol{t}_n) \left(\prod_{i=1}^{n} c_i \right)$$

Подставляя это выражение в формулу пересчета для $\boldsymbol{\alpha}(\boldsymbol{t}_n)$

$$\boldsymbol{\alpha}(\boldsymbol{t}_n) = p(\boldsymbol{x}_n | \boldsymbol{t}_n) \sum_{\boldsymbol{t}_{n-1}} \boldsymbol{\alpha}(\boldsymbol{t}_{n-1}) p(\boldsymbol{t}_n | \boldsymbol{t}_{n-1}),$$

получаем

$$c_n \hat{\boldsymbol{\alpha}}(\boldsymbol{t}_n) = p(\boldsymbol{x}_n | \boldsymbol{t}_n) \sum_{\boldsymbol{t}_{n-1}} \hat{\boldsymbol{\alpha}}(\boldsymbol{t}_{n-1}) p(\boldsymbol{t}_n | \boldsymbol{t}_{n-1})$$

Значение c_n определяется из условия нормировки для $\hat{\boldsymbol{\alpha}}(\boldsymbol{t}_n)$.

Устойчивые формулы для $\boldsymbol{\beta}(\boldsymbol{t}_n)$

Рассмотрим следующую величину

$$\hat{\beta}(\boldsymbol{t}_n) = \frac{p(\boldsymbol{x}_{n+1},\dots,\boldsymbol{x}_N|\boldsymbol{t}_n)}{p(\boldsymbol{x}_{n+1},\dots,\boldsymbol{x}_N|\boldsymbol{x}_1,\dots,\boldsymbol{x}_n)} = \frac{\beta(\boldsymbol{t}_n)}{\prod_{i=n+1}^{N} c_i}$$

Вычисление данной величины будет устойчивым, т.к. $\hat{\beta}(\boldsymbol{t}_n)$ является отношением двух распределений на $(\boldsymbol{x}_{n+1},\dots,\boldsymbol{x}_N)$. Подставляя выражение для $\hat{\beta}(\boldsymbol{t}_n)$ в формулу пересчета

$$\beta(\boldsymbol{t}_n) = \sum_{\boldsymbol{t}_{n+1}} \beta(\boldsymbol{t}_{n+1}) p(\boldsymbol{t}_{n+1}|\boldsymbol{t}_n) p(\boldsymbol{x}_{n+1}|\boldsymbol{t}_{n+1}),$$

получаем

$$c_{n+1}\hat{\beta}(\boldsymbol{t}_n) = \sum_{\boldsymbol{t}_{n+1}} \hat{\beta}(\boldsymbol{t}_{n+1}) p(\boldsymbol{t}_{n+1}|\boldsymbol{t}_n) p(\boldsymbol{x}_{n+1}|\boldsymbol{t}_{n+1})$$

Значения c_n определяются из формул пересчета для $\hat{\alpha}(\boldsymbol{t}_n)$.

Окончательные выражения для E-шага

Правдоподобие $p(X)$ вычисляется как

$$p(X) = \prod_{n=1}^{N} c_n$$

Другие необходимые величины вычисляются как

$$\begin{aligned}\gamma(\boldsymbol{t}_n) =& \hat{\alpha}(\boldsymbol{t}_n)\hat{\beta}(\boldsymbol{t}_n)\\ \xi(\boldsymbol{t}_{n-1}, \boldsymbol{t}_n) =& \frac{1}{c_n}\hat{\alpha}(\boldsymbol{t}_{n-1})p(\boldsymbol{x}_n|\boldsymbol{t}_n)p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1})\hat{\beta}(\boldsymbol{t}_n)\end{aligned}$$

План лекции

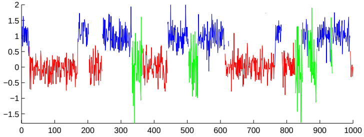
Модельный пример I

В качестве иллюстрации работы ЕМ-алгоритма обучения СММ рассмотрим простой модельный пример. $X \in \mathbb{R}$, $K = 3$, данные сгенерированы из нормальных распределений со следующими параметрами:

$$\mu_1 = 0, \quad \mu_2 = 0, \quad \mu_3 = 1$$
$$\sigma_1^2 = 0.1, \ \sigma_2^2 = 0.5, \ \sigma_3^2 = 0.1$$

Априорные вероятности и матрица перехода выбраны следующим образом:

$$\boldsymbol{\pi} = [0.3,\ 0.2,\ 0.5], \qquad A = \begin{bmatrix} 0.98 & 0.01 & 0.01 \\ 0.01 & 0.97 & 0.02 \\ 0.01 & 0.01 & 0.98 \end{bmatrix}$$



Метод максимального правдоподобия для СММ

Алгоритм «вперед-назад»

Устойчивые формулы для алгоритма «вперед-назад»

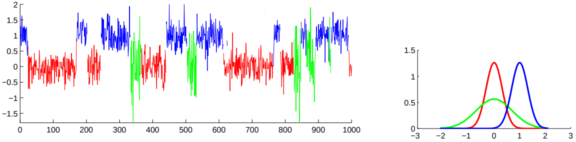
Модельный пример

Модификации стандартной СММ

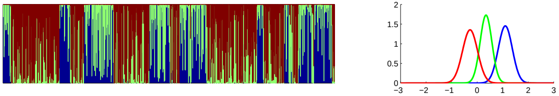
Линейные динамические системы

Вывод в ЛДС

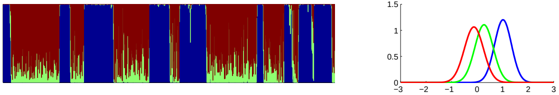
Модельный пример II



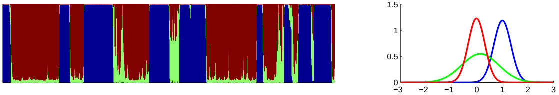
Ит. 1:



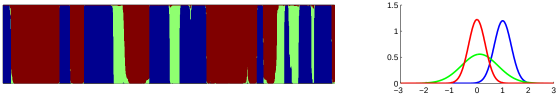
Ит. 5:



Ит. 20:



Ит. 54:



Модельный пример III

После 54-ой итерации EM-алгоритма значения параметров были следующие:

$$\boldsymbol{\pi} = [10^{-190},\ 10^{-125},\ 1], \qquad A = \begin{bmatrix} 0.984 & 0.004 & 0.012 \\ 0.013 & 0.949 & 0.038 \\ 0.011 & 0.011 & 0.978 \end{bmatrix}$$

$$\mu_1 = -0.01,\ \sigma_1^2 = 0.11$$
$$\mu_2 = 0.1,\ \sigma_2^2 = 0.51$$
$$\mu_3 = 1,\ \sigma_3^2 = 0.11$$

План лекции

Модификации стандартной СММ

TODO:

- Использование смесей распределений в качестве $p(\boldsymbol{x}_n|\boldsymbol{t}_n)$ невозможно ввиду вычислительной сложности
- Априорные вероятности на длину сегмента
- Авторегрессионная СММ
- Input-output СММ
- Factorial СММ
- Заключение

Лекция 5. Обучение без учителя скрытых марковских моделей и линейных динамических систем

Д. П. Ветров[1]

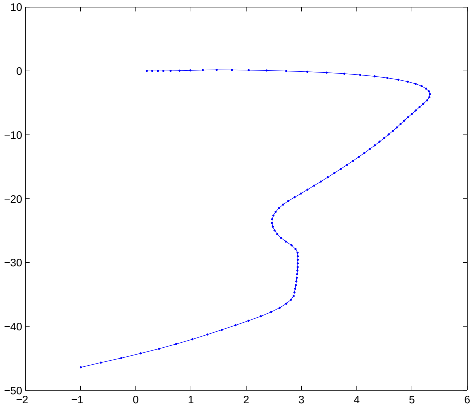
[1]МГУ, ВМиК, каф. ММП

Спецкурс «Графические модели»

План лекции

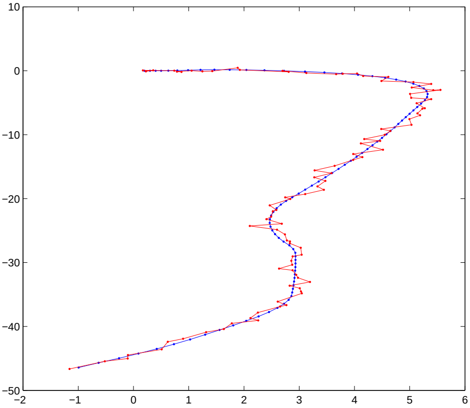
Пример - задача трекинга

Пусть имеется траектория движения объекта во времени



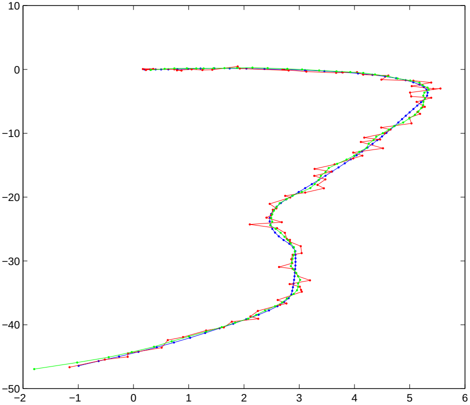
Пример - задача трекинга

Координаты объекты измерены с некоторой погрешностью

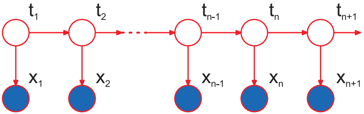


Пример - задача трекинга

Задача - оценить истинные координаты объекта с использованием координат объекта в различные моменты времени



Вероятностная модель



- $\boldsymbol{x}_1, \dots, \boldsymbol{x}_N$ — наблюдаемая последовательность
- $\boldsymbol{t}_1, \dots, \boldsymbol{t}_N$ — истинные параметры объекта
- $p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1})$ — модель движения объекта
- $p(\boldsymbol{x}_n|\boldsymbol{t}_n)$ — модель шума

Пример спецификации модели

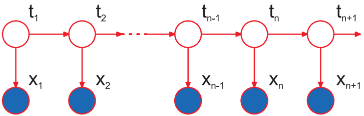
- $\boldsymbol{t}_n = \{\xi_1(n), \xi_2(n), \dot{\xi}_1(n), \dot{\xi}_2(n), \ddot{\xi}_1(n), \ddot{\xi}_2(n)\}$ - истинные координаты, скорости и ускорения объекта в момент времени n
- Модель движения объекта

$$\xi_i(n) = \xi_i(n-1) + \dot{\xi}_i(n-1)\Delta t + \frac{1}{2}\ddot{\xi}_i(n-1)(\Delta t)^2 + \varepsilon_1, \ \ i = 1, 2$$
$$\dot{\xi}_i(n) = \dot{\xi}_i(n-1) + \ddot{\xi}_i(n-1)\Delta t + \varepsilon_2, \ \ i = 1, 2$$
$$\ddot{\xi}_i(n) = \ddot{\xi}_i(n-1) + \varepsilon_3, \ \ i = 1, 2$$
$$\varepsilon_i \sim \mathcal{N}(0, \sigma_i), \ \ i = 1, 2, 3$$

- Модель сенсора (шума)

$$x_i(n) = \xi_i(n) + \nu_i, \quad i = 1, 2$$
$$\nu_i \sim \mathcal{N}(0, s_i), \ \ i = 1, 2$$

Скрытая марковская модель



- Переменные $\boldsymbol{x}_n$ — произвольные
- Переменные $\boldsymbol{t}_n$ — дискретные, принимают значения из $\{1, \ldots, K\}$
- Распределения $p(\boldsymbol{x}_n|\boldsymbol{t}_n)$ и $p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1})$ — произвольные.

Наша вероятностная модель имеют такую же структуру. Основное отличие - переменные $\boldsymbol{t}_n$ непрерывные.

Обучение и вывод в скрытой марковской модели

$$p(\boldsymbol{t}_n|X) = \frac{p(\boldsymbol{t}_n, X)}{p(X)} = \frac{p(\boldsymbol{t}_n, \boldsymbol{x}_1, \dots, \boldsymbol{x}_n)}{p(\boldsymbol{x}_1, \dots, \boldsymbol{x}_n)} \cdot \frac{p(\boldsymbol{x}_{n+1}, \dots, \boldsymbol{x}_N|\boldsymbol{t}_n)}{p(\boldsymbol{x}_{n+1}, \dots, \boldsymbol{x}_N|\boldsymbol{x}_1, \dots, \boldsymbol{x}_n)} = \hat{\alpha}(\boldsymbol{t}_n)\hat{\beta}(\boldsymbol{t}_n)$$

Формулы пересчета

$$c_n\hat{\alpha}(\boldsymbol{t}_n) = p(\boldsymbol{x}_n|\boldsymbol{t}_n)\sum_{\boldsymbol{t}_{n-1}}\hat{\alpha}(\boldsymbol{t}_{n-1})p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1})$$

$$c_{n+1}\hat{\beta}(\boldsymbol{t}_n) = \sum_{\boldsymbol{t}_{n+1}}\hat{\beta}(\boldsymbol{t}_{n+1})p(\boldsymbol{x}_{n+1}|\boldsymbol{t}_{n+1})p(\boldsymbol{t}_{n+1}|\boldsymbol{t}_n)$$

В случае непрерывных $\boldsymbol{t}_n$ сумма заменяется на интеграл.

Ограничения на вероятностную модель

- Алгоритм вывода должен иметь линейную по N сложность.
- Формулы пересчета

$$c_n\hat{\alpha}(\boldsymbol{t}_n) = p(\boldsymbol{x}_n|\boldsymbol{t}_n)\int\hat{\alpha}(\boldsymbol{t}_{n-1})p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1})d\boldsymbol{t}_{n-1}$$

$$c_{n+1}\hat{\beta}(\boldsymbol{t}_n) = \int\hat{\beta}(\boldsymbol{t}_{n+1})p(\boldsymbol{x}_{n+1}|\boldsymbol{t}_{n+1})p(\boldsymbol{t}_{n+1}|\boldsymbol{t}_n)d\boldsymbol{t}_{n+1}$$

  Эти интегралы должны вычисляться аналитически и модель не должна усложняться при переходе от $\hat{\alpha}(\boldsymbol{t}_{n-1})$ к $\hat{\alpha}(\boldsymbol{t}_n)$.
- Пример усложнения модели. Пусть $p(\boldsymbol{x}_n|\boldsymbol{t}_n)$ - смесь K гауссиан. Тогда если $\hat{\alpha}(\boldsymbol{t}_1)$ гауссиана, то $\hat{\alpha}(\boldsymbol{t}_2)$ - смесь из K гауссиан, $\hat{\alpha}(\boldsymbol{t}_3)$ - смесь из K^2 гауссиан и т.д.
- Для решения задачи сегментации функция Беллмана должна вычисляться аналитически.

Линейная динамическая система (ЛДС)

$$p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1}) = \mathcal{N}(\boldsymbol{t}_n|A\boldsymbol{t}_{n-1}, \Gamma)$$
$$p(\boldsymbol{x}_n|\boldsymbol{t}_n) = \mathcal{N}(\boldsymbol{x}_n|C\boldsymbol{t}_n, \Sigma)$$
$$p(\boldsymbol{t}_1) = \mathcal{N}(\boldsymbol{t}_1|\boldsymbol{\mu}_0, V_0)$$

Эквивалентная формулировка

$$\boldsymbol{t}_n = A\boldsymbol{t}_{n-1} + \boldsymbol{w}_n$$
$$\boldsymbol{x}_n = C\boldsymbol{t}_n + \boldsymbol{v}_n$$
$$\boldsymbol{t}_1 = \boldsymbol{\mu}_0 + \boldsymbol{u}$$
$$\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{w}|\boldsymbol{0}, \Gamma)$$
$$\boldsymbol{v} \sim \mathcal{N}(\boldsymbol{v}|\boldsymbol{0}, \Sigma)$$
$$\boldsymbol{u} \sim \mathcal{N}(\boldsymbol{u}|\boldsymbol{0}, V_0)$$

Параметры модели $\{A, \Gamma, C, \Sigma, \boldsymbol{\mu}_0, V_0\}$

Свойство ЛДС

$$p(X, T|\Theta) = p(\boldsymbol{t}_1)\prod_{n=2}^{N} p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1})\prod_{n=1}^{N} p(\boldsymbol{x}_n|\boldsymbol{t}_n)$$

Все атомарные распределения представляют собой линейную гауссовскую модель. Поэтому совместное распределение $p(X, T|\Theta)$, а также все его маргинальные и условные распределения будут также гауссовскими. Наиболее вероятная последовательность T_* определяется по индивидуально наиболее вероятным значениям $\boldsymbol{t}_n$:

$$\boldsymbol{t}_n^* = \arg\max_{\boldsymbol{t}_n} p(\boldsymbol{t}_n|X, \Theta)$$

Вывод: если известны $p(\boldsymbol{t}_n|X, \Theta)$, то аналог алгоритма Витерби для сегментации не требуется!

Многомерное нормальное распределение

$$\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \Sigma) = \frac{1}{(2\pi)^d \det \Sigma} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$$

Пусть $\boldsymbol{x}$ состоит из двух групп переменных $\boldsymbol{x}_a$ и $\boldsymbol{x}_b$, т.е.

$$\boldsymbol{x} = \begin{pmatrix} \boldsymbol{x}_a \\ \boldsymbol{x}_b \end{pmatrix}, \ \boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_a \\ \boldsymbol{\mu}_b \end{pmatrix}, \ \Sigma = \begin{pmatrix} \Sigma_{aa} & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_{bb} \end{pmatrix}, \ \Lambda = \Sigma^{-1} = \begin{pmatrix} \Lambda_{aa} & \Lambda_{ab} \\ \Lambda_{ba} & \Lambda_{bb} \end{pmatrix}$$

Тогда

$$p(\boldsymbol{x}_a) = \mathcal{N}(\boldsymbol{x}_a|\boldsymbol{\mu}_a, \Sigma_{aa})$$
$$p(\boldsymbol{x}_a|\boldsymbol{x}_b) = \mathcal{N}(\boldsymbol{x}_a|\boldsymbol{\mu}_a - \Lambda_{aa}^{-1}\Lambda_{ab}(\boldsymbol{x}_b - \boldsymbol{\mu}_b), \Lambda_{aa}^{-1})$$

План лекции

Вывод в ЛДС

Для решения задачи сегментации, а также для ЕМ-алгоритма обучения ЛДС важно уметь вычислять характеристики:

$$\hat{\alpha}(\boldsymbol{t}_n) = p(\boldsymbol{t}_n|\boldsymbol{x}_1, \dots, \boldsymbol{x}_n) = \mathcal{N}(\boldsymbol{t}_n|\boldsymbol{\mu}_n, V_n)$$

$$\hat{\beta}(\boldsymbol{t}_n) = \frac{p(\boldsymbol{x}_{n+1}, \dots, \boldsymbol{x}_N|\boldsymbol{t}_n)}{p(\boldsymbol{x}_{n+1}, \dots, \boldsymbol{x}_N|\boldsymbol{x}_1, \dots, \boldsymbol{x}_n)}$$

Для их вычисления можно применять алгоритм «вперед-назад». Проход вперед для ЛДС получил название фильтра Калмана, а проход назад - РТС уравнения (Rauch-Tung-Striebel).

Проход вперед (фильтр Калмана)

Формула пересчета

$$c_n\hat{\alpha}(\boldsymbol{t}_n) = p(\boldsymbol{x}_n|\boldsymbol{t}_n)\int\hat{\alpha}(\boldsymbol{t}_{n-1})p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1})d\boldsymbol{t}_{n-1}$$

Подставляем

$$c_n\mathcal{N}(\boldsymbol{t}_n|\boldsymbol{\mu}_n, V_n) = \mathcal{N}(\boldsymbol{x}_n|C\boldsymbol{t}_n, \Sigma) \int\mathcal{N}(\boldsymbol{t}_n|A\boldsymbol{t}_{n-1}, \Gamma)\mathcal{N}(\boldsymbol{t}_{n-1}|\boldsymbol{\mu}_{n-1}, V_{n-1})d\boldsymbol{t}_{n-1}$$

Результат

$$\begin{aligned}
\boldsymbol{\mu}_n &= A\boldsymbol{\mu}_{n-1} + K_n(\boldsymbol{x}_n - CA\boldsymbol{\mu}_{n-1})\\
V_n &= (I - K_nC)P_{n-1}\\
c_n &= \mathcal{N}(\boldsymbol{x}_n|CA\boldsymbol{\mu}_{n-1}, CP_{n-1}C^T + \Sigma)\\
P_{n-1} &= AV_{n-1}A^T + \Gamma\\
K_n &= P_{n-1}C^T(CP_{n-1}C^T + \Sigma)^{-1}
\end{aligned}$$

Фильтр Калмана

Традиционно в фильтре Калмана рассматривают два этапа:

Прогноз. Оценивается распределение

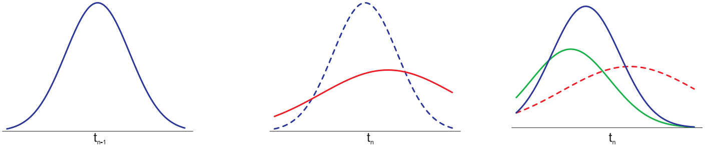
$$p(\boldsymbol{t}_n|\boldsymbol{x}_1, \dots, \boldsymbol{x}_{n-1}) \sim \mathcal{N}(\boldsymbol{t}_n|\tilde{\boldsymbol{\mu}}_n, \tilde{V}_n)$$

$$\tilde{\boldsymbol{\mu}}_n = A\boldsymbol{\mu}_{n-1}$$
$$\tilde{V}_n = P_{n-1}$$
$$\tilde{\boldsymbol{x}}_n = C\tilde{\boldsymbol{\mu}}_n$$

Коррекция.

$$\boldsymbol{\mu}_n = \tilde{\boldsymbol{\mu}}_n + K_n(\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n)$$
$$V_n = (I - K_n C)\tilde{V}_n$$
$$c_n = \mathcal{N}(\boldsymbol{x}_n|\tilde{\boldsymbol{x}}_n, C\tilde{V}_n C^T + \Sigma)$$

Иллюстрация фильтра Калмана



(a) (b) (c)

$p(\boldsymbol{t}_{n-1}|\boldsymbol{x}_1,\dots,\boldsymbol{x}_{n-1})$ $p(\boldsymbol{t}_n|\boldsymbol{x}_1,\dots,\boldsymbol{x}_{n-1})$ $p(\boldsymbol{t}_n|\boldsymbol{x}_1,\dots,\boldsymbol{x}_n)$

Фильтр Калмана. Начальное приближение.

Значение $\hat{\alpha}(\boldsymbol{t}_1)$ в начальный момент времени вычисляется из условия:

$$c_1\hat{\alpha}(\boldsymbol{t}_1) = p(\boldsymbol{t}_1)p(\boldsymbol{x}_1|\boldsymbol{t}_1)$$

Производя свертку двух гауссиан, получаем:

$$\boldsymbol{\mu}_1 = \boldsymbol{\mu}_0 + K_1(\boldsymbol{x}_1 - C\boldsymbol{\mu}_0)$$
$$V_1 = (I - K_1C)V_0$$
$$c_1 = \mathcal{N}(\boldsymbol{x}_1|C\boldsymbol{\mu}_0, CV_0C^T + \Sigma)$$
$$K_1 = V_0C^T(CV_0C^T + \Sigma)^{-1}$$

Проход назад

$$\gamma(\boldsymbol{t}_n) = p(\boldsymbol{t}_n|X,\Theta) = \hat{\alpha}(\boldsymbol{t}_n)\hat{\beta}(\boldsymbol{t}_n) = \mathcal{N}(\boldsymbol{t}_n|\hat{\boldsymbol{\mu}}_n, \hat{V}_n)$$

В отличие от $\gamma(\boldsymbol{t}_n)$ $\hat{\beta}(\boldsymbol{t}_n)$ не является маргинальным распределением:

$$\hat{\beta}(\boldsymbol{t}_n) = \frac{p(\boldsymbol{x}_{n+1},\dots,\boldsymbol{x}_N|\boldsymbol{t}_n)}{p(\boldsymbol{x}_{n+1},\dots,\boldsymbol{x}_N|\boldsymbol{x}_1,\dots,\boldsymbol{x}_n)}$$

Поэтому формулы для обратного прохода удобнее записывать в терминах $\gamma(\boldsymbol{t}_n)$

Формулы для обратного прохода

Формула пересчета

$$c_{n+1}\hat{\beta}(\boldsymbol{t}_n) = \int \hat{\beta}(\boldsymbol{t}_{n+1})p(\boldsymbol{x}_{n+1}|\boldsymbol{t}_{n+1})p(\boldsymbol{t}_{n+1}|\boldsymbol{t}_n)d\boldsymbol{t}_{n+1}$$

$$\hat{\boldsymbol{\mu}}_n = \boldsymbol{\mu}_n + J_n(\hat{\boldsymbol{\mu}}_{n+1} - A\boldsymbol{\mu}_n)$$

$$\hat{V}_n = V_n + J_n(\hat{V}_{n+1} - P_n)J_n^T$$

$$J_n = V_n A^T (P_n)^{-1}$$

Распределение для пары переменных

Для EM-алгоритма обучения понадобятся также величины

$$\xi(\boldsymbol{t}_{n-1},\boldsymbol{t}_n)=p(\boldsymbol{t}_{n-1},\boldsymbol{t}_n|X,\Theta)=$$

$$=(c_n)^{-1}\hat{\alpha}(\boldsymbol{t}_{n-1})p(\boldsymbol{x}_n|\boldsymbol{t}_n)p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1})\hat{\beta}(\boldsymbol{t}_n)=$$

$$\frac{\mathcal{N}(\boldsymbol{t}_{n-1}|\boldsymbol{\mu}_{n-1},V_{n-1})\mathcal{N}(\boldsymbol{t}_n|A\boldsymbol{t}_{n-1},\Gamma)\mathcal{N}(\boldsymbol{x}_n|C\boldsymbol{t}_n,\Sigma)\mathcal{N}(\boldsymbol{t}_n|\hat{\boldsymbol{\mu}}_n,\hat{V}_n)}{c_n\mathcal{N}(\boldsymbol{t}_n|\boldsymbol{\mu}_n,V_n)}=$$

$$=\mathcal{N}(\boldsymbol{t}_{n-1},\boldsymbol{t}_n|[\gamma(\boldsymbol{t}_{n-1}),\gamma(\boldsymbol{t}_n)]^T,J_{n-1}\hat{V}_n)$$

План лекции

1. Метод максимального правдоподобия для СММ
2. Алгоритм «вперед-назад»
3. Устойчивые формулы для алгоритма «вперед-назад»
4. Модельный пример
5. Модификации стандартной СММ
6. Линейные динамические системы
7. Вывод в ЛДС
8. **Расширенный фильтр Калмана**
9. Обучение в ЛДС

Нелинейная фильтрация

- Рассмотрим более сложную задачу фильтрации сигналов
- Пусть зависимости между соседними переменными нелинейные, но шум по-прежнему гауссовский

$$\boldsymbol{t}_n = f(\boldsymbol{t}_{n-1}) + \boldsymbol{w}_n, \quad \boldsymbol{w}_n \sim \mathcal{N}(\boldsymbol{w}|\boldsymbol{0}, \Gamma)$$

$$\boldsymbol{x}_n = h(\boldsymbol{t}_n) + \boldsymbol{v}_n, \quad \boldsymbol{v} \sim \mathcal{N}(\boldsymbol{v}|\boldsymbol{0}, \Sigma)$$

- Требуется по выборке $X_n = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ оценить распределение на текущую скрытую компоненту $\boldsymbol{t}_n$

Линеаризация нелинейных зависимостей

- Приблизим нелинейные зависимости линейными

$$\boldsymbol{t}_n = A_{n-1}\boldsymbol{t}_{n-1} + \boldsymbol{w}_n, \quad A_{n-1} = A(\boldsymbol{\mu}_{n-1}) = \left.\frac{\partial f}{\partial \boldsymbol{t}}\right|_{\boldsymbol{t}=\boldsymbol{\mu}_{n-1}}$$

$$\boldsymbol{x}_n = C_n\boldsymbol{t}_n + \boldsymbol{v}_n, \quad C_n = C(\tilde{\boldsymbol{\mu}}_n) = \left.\frac{\partial h}{\partial \boldsymbol{t}}\right|_{\boldsymbol{t}=\tilde{\boldsymbol{\mu}}_n}$$

- Обратите внимание, что производная функции $f(\boldsymbol{t})$ берется в точке $\boldsymbol{t} = \boldsymbol{\mu}_{n-1}$, а производная функции $h(\boldsymbol{t})$ — в точке $\boldsymbol{t} = \tilde{\boldsymbol{\mu}}_n$. Вопрос аудитории: почему?
- Линеаризация зависимостей позволяет использовать обычный фильтр Калмана, но с учетом того, что теперь матрицы A и C стали зависеть от времени

Расширенный фильтр Калмана

После того как мы линеаризовали задачу применяем фильтр Калмана

Прогноз. Оценивается распределение

$$p(\boldsymbol{t}_n|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}) \sim \mathcal{N}(\boldsymbol{t}_n|\tilde{\boldsymbol{\mu}}_n, \tilde{V}_n)$$

$$\begin{aligned}
\tilde{\boldsymbol{\mu}}_n &= f(\boldsymbol{\mu}_{n-1}) \\
\tilde{V}_n &= A_{n-1}V_{n-1}A_{n-1}^T + \Gamma \\
\tilde{\boldsymbol{x}}_n &= h(\tilde{\boldsymbol{\mu}}_n)
\end{aligned}$$

Коррекция.

$$K_n = \tilde{V}_n C_n^T (C_n \tilde{V}_n C_n^T + \Sigma)^{-1} \qquad \boldsymbol{\mu}_n = \tilde{\boldsymbol{\mu}}_n + K_n(\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n)$$

$$V_n = (I - K_n C_n^T)\tilde{V}_n$$

Обычный фильтр Калмана

Сравним формулы с обычным фильтром Калмана, описанным в предыдущем разделе

Прогноз. Оценивается распределение

$$p(\boldsymbol{t}_n|\boldsymbol{x}_1,\dots,\boldsymbol{x}_{n-1}) \sim \mathcal{N}(\boldsymbol{t}_n|\tilde{\boldsymbol{\mu}}_n, \tilde{V}_n)$$

$$\begin{aligned}
\tilde{\boldsymbol{\mu}}_n &= A\boldsymbol{\mu}_{n-1} \\
\tilde{V}_n &= AV_{n-1}A^T + \Gamma \\
\tilde{\boldsymbol{x}}_n &= C\tilde{\boldsymbol{\mu}}_n
\end{aligned}$$

Коррекция.

$$\begin{aligned}
K_n &= \tilde{V}_n C^T (C\tilde{V}_n C^T + \Sigma)^{-1} \qquad \boldsymbol{\mu}_n = \tilde{\boldsymbol{\mu}}_n + K_n(\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n) \\
V_n &= (I - K_n C^T)\tilde{V}_n
\end{aligned}$$

Заключительные замечания

- Если дисперсии шумов не слишком велики (т.е. мы не слишком сильно отклоняемся от точки, в которой выполнили линеаризацию) можно расчитывать на адекватное приближение и успешное решение задачи фильтрации
- Если шумы негауссовы, расширенный фильтр Калмана не подходит и нужно использовать другие методы, например, фильтр частиц (см. следующую лекцию)

План лекции

EM-алгоритм. Разложение логарифма правдоподобия

Требуется найти максимум правдоподобия в вероятностной модели со скрытыми переменными:

$$p(X|\Theta) = \int p(X,T|\Theta)dT \to \max_{\Theta} \Leftrightarrow \log p(X|\Theta) \to \max_{\Theta}$$

$$p(X,T|\Theta) = p(T|X,\Theta)p(X|\Theta) \Rightarrow$$
$$\log p(X,T|\Theta) = \log p(T|X,\Theta) + \log p(X|\Theta)$$

$q(T)$ — произвольное распределение.

$$\log p(X|\Theta) = \int \log p(X|\Theta)q(T)dT =$$
$$\int [\log p(X,T|\Theta) - \log p(T|X,\Theta)]\, q(T)dT =$$
$$\int \log p(X,T|\Theta)q(T)dT - \int \log p(T|X,\Theta)q(T)dT =$$
$$\int \log \frac{p(X,T|\Theta)}{q(T)}q(T)dT - \int \log \frac{p(T|X,\Theta)}{q(T)}q(T)dT$$

Нижняя оценка для логарифма правдоподобия

$$\log p(X|\Theta) = \underbrace{\int \log \frac{p(X,T|\Theta)}{q(T)} q(T)dT}_{l(q,\Theta)} - \underbrace{\int \log \frac{p(T|X,\Theta)}{q(T)} q(T)dT}_{KL(q||p)\geq 0}$$

Дивергенция Кульбака-Лейблера $KL(q||p)$ определяет расстояние между вероятностными распределениями

- $KL(q||p) = -\int q(x)\log(p(x)/q(x))dx$
- $KL(q||p) \geq 0$ и $KL(q||p) = 0 \Leftrightarrow q \equiv p$.
- $KL(q||p) \neq KL(p||q)$

Тогда $l(q,\Theta)$ является нижней оценкой правдоподобия $\log p(X|\Theta)$:

$$\log p(X|\Theta) \geq l(q,\Theta) \text{ и равенство } \Leftrightarrow q(T) = p(T|X,\Theta)$$

Идея EM-алгоритма

$$\log p(X|\Theta) = l(q,\Theta) + KL(q||p)$$

Итерационная схема. Фиксируем некоторое значение Θ_{old}. Приблизим в точке Θ_{old} правдоподобие с помощью его нижней оценки:

$$q(T) = p(T|X,\Theta_{old})$$

$$\log p(X|\Theta) \geq l(q,\Theta) = \int \log p(X,T|\Theta)p(T|X,\Theta_{old})dT - \int \log p(T|X,\Theta_{old})p(T|X,\Theta_{old})dT$$

Найдем новое значение Θ с помощью максимизации нижней оценки:

$$l(q,\Theta) \to \max_{\Theta} \Leftrightarrow \mathbb{E}_{T|X,\Theta_{old}} \log p(X,T|\Theta) \to \max_{\Theta}$$

Иллюстрация ЕМ-алгоритма

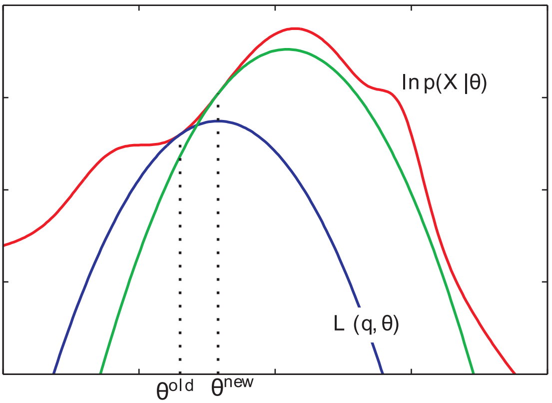


Схема ЕМ-алгоритма

- Е-шаг. Фиксируется значение параметров Θ_{old}. Оценивается апостериорное распределение на скрытые переменные $p(T|X,\Theta_{old})$, и полное правдоподобие усредняется по полученному распределению:

$$\mathbb{E}_{T|X,\Theta_{old}} \log p(X,T|\Theta) = \int \log p(X,T|\Theta) p(T|X,\Theta_{old}) dT$$

- М-шаг. Фиксируется апостериорное распределение $p(T|X,\Theta_{old})$, и производится поиск новых значений параметров Θ_{new}:

$$\Theta_{new} = \arg\max_{\Theta} \mathbb{E}_{T|X,\Theta_{old}} \log p(X,T|\Theta)$$

- Шаги Е и М повторяются до сходимости.

Максимизация апостериорного распределения

Задача

$$p(\Theta|X) \to \max_{\Theta} \Leftrightarrow F = \log p(X|\Theta) + \log p(\Theta) \to \max_{\Theta}$$

Справедливо разложение

$$F = L(q, \theta) + \log p(\Theta) + KL(q||p) \geq L(q, \Theta) + \log p(\Theta)$$

Е-шаг остается без изменений.
Модификация М-шага:

$$\mathbb{E}_{T|X,\Theta_{old}} \log p(X, T|\Theta) + \log p(\Theta) \to \max_{\Theta}$$

EM-алгоритм для ЛДС

Задача - поиск значений параметров $\{A, \Gamma, C, \Sigma, \boldsymbol{\mu}_0, V_0\}$ по методу максимального правдоподобия

Логарифм полного правдоподобия

$$\log p(X,T|\Theta) = \log p(\boldsymbol{t}_1|\boldsymbol{\mu}_0, V_0) + \sum_{n=2}^{N} \log p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1}, A, \Gamma) + \sum_{n=1}^{N} \log p(\boldsymbol{x}_n|\boldsymbol{t}_n, C, \Sigma)$$

Для вычисления нижней оценки

$$Q(\Theta, \Theta_{old}) = \mathbb{E}_{T|X,\Theta_{old}} \log p(X,T|\Theta)$$

достаточно знать следующие величины:

$$\begin{aligned}
\mathbb{E}\boldsymbol{t}_n &= \hat{\boldsymbol{\mu}}_n \\
\mathbb{E}\boldsymbol{t}_n\boldsymbol{t}_{n-1}^T &= J_{n-1}\hat{V}_n + \hat{\boldsymbol{\mu}}_n\hat{\boldsymbol{\mu}}_{n-1}^T \\
\mathbb{E}\boldsymbol{t}_n\boldsymbol{t}_n^T &= \hat{V}_n + \hat{\boldsymbol{\mu}}_n\hat{\boldsymbol{\mu}}_n^T
\end{aligned}$$

M-шаг. Формулы для $\boldsymbol{\mu}_0$ и V_0

$$Q(\Theta, \Theta_{old}) = -\frac{1}{2}\log\det V_0 - \mathbb{E}_{T|X,\Theta_{old}}\left[\frac{1}{2}(\boldsymbol{t}_1 - \boldsymbol{\mu}_0)^T V_0^{-1}(\boldsymbol{t}_1 - \boldsymbol{\mu}_0)\right] + \mathrm{const}$$

Здесь const не зависит от $\boldsymbol{\mu}_0$ и V_0.
Приравнивая производные по $\boldsymbol{\mu}_0$ и V_0 к нулю, получаем:

$$\boldsymbol{\mu}_0^{new} = \mathbb{E}\boldsymbol{t}_1$$
$$V_0^{new} = \mathbb{E}\boldsymbol{t}_1\boldsymbol{t}_1^T - \mathbb{E}\boldsymbol{t}_1\mathbb{E}\boldsymbol{t}_1^T$$

М-шаг. Формулы для A и Γ

$$Q(\Theta, \Theta_{old}) = -\frac{N-1}{2}\log\det\Gamma -$$
$$\mathbb{E}_{T|X,\Theta_{old}}\left[\frac{1}{2}\sum_{n=2}^{N}(\boldsymbol{t}_n - A\boldsymbol{t}_{n-1})^T\Gamma^{-1}(\boldsymbol{t}_n - A\boldsymbol{t}_{n-1})\right] + \mathrm{const}$$

Здесь const не зависит от A и Γ.

Приравнивая производные по A и Γ к нулю, получаем:

$$A^{new} = \left(\sum_{n=2}^{N}\mathbb{E}\boldsymbol{t}_n\boldsymbol{t}_{n-1}^T\right)\left(\sum_{n=2}^{N}\mathbb{E}\boldsymbol{t}_{n-1}\boldsymbol{t}_{n-1}^T\right)^{-1}$$

$$\Gamma^{new} = \frac{1}{N-1}\sum_{n=2}^{N}\Big\{\mathbb{E}\boldsymbol{t}_n\boldsymbol{t}_n^T - A^{new}\mathbb{E}\boldsymbol{t}_{n-1}\boldsymbol{t}_n^T -$$
$$\mathbb{E}\boldsymbol{t}_n\boldsymbol{t}_{n-1}^T(A^{new})^T + A^{new}\mathbb{E}\boldsymbol{t}_{n-1}\boldsymbol{t}_{n-1}^T(A^{new})^T\Big\}$$

M-шаг. Формулы для C и Σ

$$Q(\Theta, \Theta_{old}) = -\frac{N}{2}\log\det\Sigma -$$
$$\mathbb{E}_{T|X,\Theta_{old}}\left[\frac{1}{2}\sum_{n=1}^{N}(\boldsymbol{x}_n - C\boldsymbol{t}_n)^T\Sigma^{-1}(\boldsymbol{x}_n - C\boldsymbol{t}_n)\right] + \mathrm{const}$$

Здесь const не зависит от C и Σ.
Приравнивая производные по C и Σ к нулю, получаем:

$$C^{new} = \left(\sum_{n=1}^{N}\boldsymbol{x}_n\mathbb{E}\boldsymbol{t}_n^T\right)\left(\sum_{n=1}^{N}\mathbb{E}\boldsymbol{t}_n\boldsymbol{t}_n^T\right)^{-1}$$

$$\Sigma^{new} = \frac{1}{N}\sum_{n=1}^{N}\left\{\boldsymbol{x}_n\boldsymbol{x}_n^T - C^{new}\mathbb{E}\boldsymbol{t}_n\boldsymbol{x}_n^T - \right.$$
$$\left. \boldsymbol{x}_n\mathbb{E}\boldsymbol{t}_n^T(C^{new})^T + C^{new}\mathbb{E}\boldsymbol{t}_n\boldsymbol{t}_n^T(C^{new})^T\right\}$$