Изучение и применение алгоритма Витерби

Кормщикова Юлия, 422 гр.

Санкт-Петербургский государственный университет
Кафедра статистического моделирования

Научный руководитель — профессор **Н.К. Кривулин**
Рецензент — доцент **А.Ю. Пономарева**

Санкт-Петербург
2015 г.

Постановка задачи

- Алгоритм Витерби — алгоритм восстановления последовательности состояний марковской цепи в скрытых марковских моделях (СММ).
- СММ — модель марковской цепи, состояния которой не могут наблюдаться непосредственно (скрыты), но могут быть определены со случайной ошибкой.

Постановка задачи

Задача:

- изучить теоретические основы и область применения алгоритма Витерби;
- изучить методы оценки параметров СММ (методы прямой оценки, алгоритм прямого-обратного хода, алгоритм Баума-Велша);
- разработать программные средства решения задач оценки параметров и восстановления состояний СММ;
- исследовать с помощью имитационного моделирования эффективность работы алгоритмов оценки и восстановления состояний в зависимости от параметров СММ.

Модель скрытой цепи маркова

Последовательности случайных величин:

- $\{\xi_k\}$ образуют скрытые состояния $i_1, \dots, i_K$;
- $\{\eta_k\}$ образуют наблюдаемые состояния $j_1, \dots, j_K$.

- $\{\xi_k\}$ образуют цепь маркова;
- $\{\eta_k\}$ связаны с цепью маркова условными вероятностями.

Параметры модели:

- $p = \{p_i\}$, где $p_i = P(\xi_0 = i)$,
- $P = \{p_{ij}\}$, где $p_{ij} = P(\xi_k = j | \xi_{k-1} = i); \quad i, j = 1, \dots, n;$
- $Q = \{q_i(j)\}$, где $q_i(j) = P(\eta_k = j | \xi_k = i); \quad k = 1, \dots, K.$

Восстановление состояний цепи маркова

Задача:

- Дано: наблюдаемые значения $j_1, \dots, j_K$ случайных величин $\eta_1, \dots, \eta_K$.
- Найти: наиболее вероятные значения $i_1, \dots, i_K$ случайных величин $\xi_1, \dots, \xi_K$.

Решение:

- Функция правдоподобия:

$$F_{j_1,\dots,j_K}(i_1, \dots, i_K) = \\ = P(\xi_1 = i_1, \dots, \xi_K = i_K | \eta_1 = j_1, \dots, \eta_K = j_K).$$

- Решением является последовательность $i_1, \dots, i_K$, на которой достигается

$$\max_{i_1,\dots,i_K} F_{j_1,\dots,j_K}(i_1, \dots, i_K).$$

Алгоритм Витерби

Вычислительная схема динамического программирования

$$f_1(i) = p_i,$$
$$f_k(i) = q_i(j_k) \max_{1 \leqslant l \leqslant n} p_{li} f_{k-1}(l), \qquad i = 1, \ldots, n, \qquad 1 < k < K.$$

определяет вероятность состояния i при каждом k.

- $j_1, \ldots, j_K$ — последовательность наблюдений.
- По найденным вероятностям можно определить наиболее вероятные состояния модели.

Алгоритм прямого-обратного хода

Вычисление апостериорных вероятностей последовательности наблюдений.

Введем обозначения: $X_k = \{\xi_k = i_k\}, \quad Y_k = \{\eta_k = j_k\}$.

Вычисление прямых вероятностей $\alpha_k(i) = P(Y_1 \cdots Y_k X_k)$.

$$\alpha_1(i) = q_i(j_1)p_i,$$

$$\alpha_k(i) = q_i(j_k)\sum_{j=1}^{n} p_{ji}\alpha_{k-1}(j), \qquad i = 1, \ldots, n,$$

$$k = 2, 3, \ldots K.$$

Алгоритм прямого-обратного хода

Вычисление обратных вероятностей

$$\beta_k(i) = P(Y_{k+1}Y_{k+2}\cdots Y_K|X_k):$$

$$\beta_k(i) = \sum_{j=1}^{n} p_{ij}\beta_{k+1}(j)q_j(j_{k+1}), \qquad k = K-1, K-2, \ldots, 1,$$

$$i = 1, \ldots, n,$$

$$\beta_K(i) = 1.$$

Алгоритм прямого-обратного хода

Вычисление сглаженных значений $\gamma_k(i) = P(X_k | Y_1 \cdots Y_K)$,

$$\delta_k(i,j) = P(X_k X_{k+1} | Y_1 \cdots Y_K):$$

$$\gamma_k(i) = \frac{\alpha_k(i)\beta_k(i)}{\sum\limits_{j=1}^{n} \alpha_k(j)\beta_k(j)},$$

$$\delta_k(i,j) = \frac{q_j(j_{k+1})p_{ij}\alpha_k(i)\beta_{k+1}(j)}{\sum\limits_{l=1}^{n}\sum\limits_{m=1}^{n} \alpha_k(m)q_l(j_{k+1})p_{ml}\beta_{k+1}(l)}; \qquad i = 1, \ldots, n;$$

$$k = 1, \ldots, K.$$

Оценка параметров скрытой цепи

Дано: наблюдения $j_1, \ldots, j_K$.

Определить: параметры модели $\Theta = (p, P, Q)$.

Функция правдоподобия:

$$L(j_1, \ldots, j_K; \Theta) = \\ = \sum_{1 \le i_1, \ldots, i_K \le n} p_{i_1} q_{i_1}(j_1) p_{i_1 i_2} \cdots q_{i_{K-1}}(j_{K-1}) p_{i_{K-1} i_K} q_{i_K}(j_K).$$

Оценка параметров скрытой цепи

Решение задачи:

$$\max_{\Theta} \quad L(j_1, \ldots, j_K; \Theta),$$

$$\sum_{i=1}^{n} p_i = 1, \qquad \sum_{j=1}^{n} p_{ij} = 1,$$

$$\sum_{j=1}^{n} q_i(j) = 1, \qquad i = 1, \ldots, n.$$

Алгоритм Баума-Велша

Исходные данные: параметры модели $\Theta = (p, P, Q)$;
последовательность наблюдений $(j_1, \ldots, j_K)$.

Уточненные значения вычисляются по формулам:

$$p_i = \gamma'_1(i), \qquad p_{ij} = \frac{\sum\limits_{k=1}^{K-1} \delta'_k(i,j)}{\sum\limits_{k=1}^{K-1} \gamma'_k(i)}, \qquad q_i(j) = \frac{\sum\limits_{l=1, j_l = j}^{K} \gamma'_l(i)}{\sum\limits_{l=1}^{K} \gamma'_l(i)}.$$

Алгоритм случайного поиска

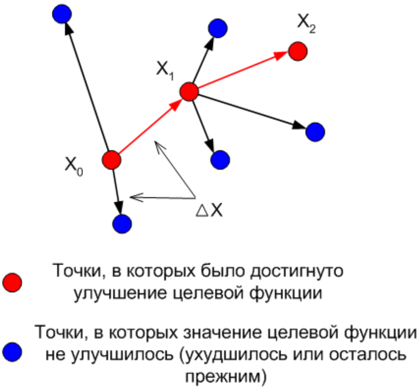


Рис.: Алгоритм случайного поиска

Программная реализация

- Произведено моделирование СММ методом Монте-Карло.
- Разработаны программные средства для алгоритмов восстановления скрытой последовательности и нахождения параметров модели.
- Проведены исследования эффективности, включая тесты работы указанных алгоритмов.
- Характеристики программы: $C\#$, 8 классов, $O(n^2K)$.

Программная реализация

Основные классы и методы:

- class HMM;
    - public HMM(HMM model);
- class Model;
    - void Modelling(HMM model, int numberObservations, int[ ] chainProb, int[ ] chainCond);
- class RandomSearch;
    - double AlgorithmRandomSearch(HMM model, int[ ] chain);

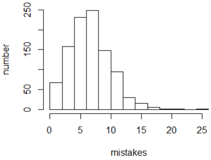
Программная реализация

Основные классы и методы:

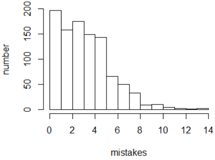
- class ForwardBack;
  - double[,] Alpha(HMM Lambda, int[ ] S, int tau);
  - double[,] Beta(HMM Lambda, int[ ] S, int tau);
  - double[,] double[][][] Gamma(HMM Lambda, int[ ] S, int tau);
  - double[, ,] Delta(HMM Lambda, int[ ] S, int tau);
- class Viterbi;
  - int[ ] AlgorithmViterbi(HMM Lambda, int[ ] S, int tau);
- class BaumWelsh;
  - void AlgorithmBaumWelsh(HMM Lambda, int[] S, int tau);

Результаты работы алгоритма Витерби

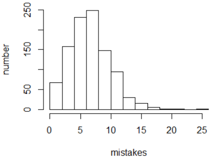
Исследование эффективности алгоритма в зависимости от числа наблюдений.



(a) Наблюдений: 10 (среднее число ошибок 0.7)



(b) Наблюдений: 50 (среднее число ошибок 3.6)



(c) Наблюдений: 100 (среднее число ошибок 7.1)

Рис.: Гистограмма ошибок

Результаты работы алгоритма Витерби

Исследование эффективности алгоритма в зависимости от числа наблюдений и параметров модели:

$$P = \begin{pmatrix} 0.9 & 0.1 \\ 0.2 & 0.8 \end{pmatrix}, \qquad Q = \begin{pmatrix} 0.9 & 0.1 \\ 0.9 & 0.1 \end{pmatrix}, \qquad p = \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix}.$$

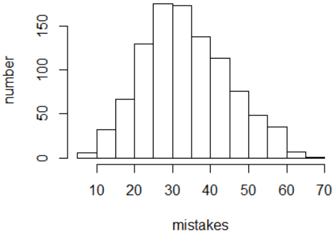


Рис.: Число наблюдений 100 (среднее число ошибок 34)

Результаты работы алгоритма Витерби

Исследование эффективности алгоритма в зависимости от местоположения ошибок.

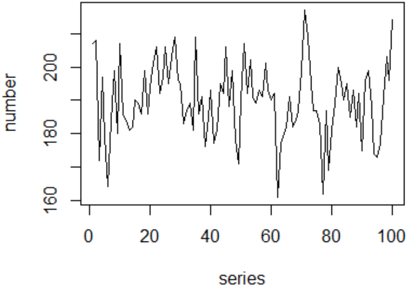


Рис.: График количества ошибок на i-ом месте, $n = 3$

Результаты работы алгоритма Баума-Велша

Исследование эффективности работы алгоритмов Баума-Велша и случайного поиска.

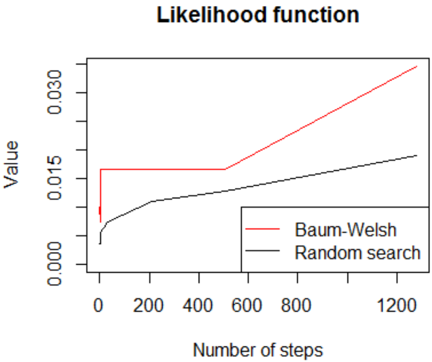


Рис.: График функции правдоподобия

Заключение

- Изучены вычислительные схемы и методы алгоритмов прямого-обратного хода, Витерби и Баума-Велша.
- Разработаны программные средства для моделирования СММ методом Монте-Карло.
- Разработаны программные средства для реализации указанных алгоритмов.
- Проведены исследования эффективности и анализ алгоритмов с помощью имитационного моделирования.
- Для экспериментально-практических исследований изучен и реализован алгоритм случайного поиска.