Лекция 4
Введение в теорию индуктивных функций на последовательностях

(с) Д.Н. Лавров
2017

Обозначения

- X – произвольный алфавит, конечное множество.
- $\Omega(X) = \{x_1\, x_2 \ldots x_n : n \in Z^+,\ x_i \in X,\ \forall i = 1,\ldots,n\}$ – множество всех конечных последовательностей над X . Для упрощения записи будем писать просто Ω.
- Δ – пустая последовательность, не содержащая ни одного элемента. Для этой последовательности верно $\Delta \in \Omega$.
- $\Omega_k\ (X) = \{x_1\, x_2 \ldots x_n : n \in Z^+,\ n > k,\ x_i \in X\}$ — множество конечных последовательностей дины не менее k. Заметим, что

  $\Omega = \Omega_0 \supset \Omega_1 \supset \Omega_2 \ldots$
- $* : \Omega \times X \rightarrow \Omega$ – операция добавления дописывания элемента в конец последовательности ($\times$ – операция декартового произведения).
- Таким образом, $x_1\, x_2 \ldots\ x_n * x = x_1\, x_2 \ldots\ x_n x$. В частности $\Delta * x = x$. Операция * естественно сужается на $\Omega_k\ \forall k$

Обозначения

- N – множество натуральных чисел.
- Z^+ – множество неотрицательных целых чисел.
- Z – множество целых чисел.
- R^+ – множество неотрицательных действительных чисел.
- R – множество действительных чисел.
- $\overline{N}$ – множество натуральных чисел с $+\infty$ и $-\infty$.
- Аналогичные определения даются и для других множеств.

Определение ИФ

- ***Определение***. *Индуктивная функция* (ИФ) – это отображение $F : \Omega \to Y$ такое, что, зная x и $F(\omega)$ можно вычислить $F(\omega * x)$.
- Строго говоря, если $\exists G : Y \times X \to Y : \forall \omega \in \Omega,\ x \in X,$ $F(\omega * x) = G(F(\omega), x).$

Примеры ИФ

- F =«Число элементов последовательности».
- Если $\omega_n = x_1\ x_2\ \dots\ x_n$ — последовательность длины n, то $F(\omega_n) = n$.
- Действительно $\forall \omega \in \Omega,\ x \in X$ верно, что $F(\omega * x) = F(\omega) + 1$.
- То есть действительно существует функция $G(y,x) = y + 1$.
- Эта функция удовлетворяет условию определения индуктивности, следовательно, F – индуктивна.

Примеры ИФ

- Пример. F = «Сумма элементов последовательности». В этом случае
- $G(y,x) = y + x$ и $F(\omega * x) = F(\omega) + x$.

Реализация

- Если найдена функция перехода (действие) *G*, то вычисления F легко организовать. Для этого необходимо определить начальное значение F.
- Как правило, его определяют либо на пустой последовательности
  $F_0 = F(\omega_0) = F(\Delta)$,
  либо на последовательности из одного элемента
  $F_1 = F(\omega_1) = F(x_1)$.
- Далее вычисление производиться по правилу
  $F(\omega_k) = G(F(\omega_{k-1}), x_k)$.
- Реализовать алгоритм можно следующим Python-кодом, если ω представлен списком или кортежем:

```
F=F0 # стоим в начале последовательности
for x in omega:
        F=G(F,x)
```

Обсуждение

- Фактически данное понятие – это адаптация метода построения цикла с помощью инварианта для последовательности
- Оно еще больше похоже на реализацию метода математической индукции в программировании.
- Что будет в этом случае инвариантом цикла?

Обсуждение

- Шаг итерации $(F(\omega_k),x_{k+1})=(G(F(\omega_{k-1})),x_{k+1})=T(F(\omega_{k-1}),x_k)$
- $I(x)$ = «В переменной F *на шаге* k *хранится искомое значение задачи для* ω_k»
- Q(x)=(k=len(ω)) – условие окончания итерации.
- $I(x)\&Q(x)$ => F решение задачи для ω.
- Начальная инициализация тоже корректна $I(x_0)=True$, так как $F(\Delta)$- *решение для пустой последовательности, то* $I(F(\Delta),x_1)=True$.

Критерий индуктивности

- **Теорема**. *Критерий индуктивности*.
  F – индуктивна $\Leftrightarrow \forall a,b \in \Omega : F(a) = F(b), \forall x \in X$ верно, что $F(a*x) = F(b*x)$.
- Другими словами F – индуктивна тогда и только тогда, когда из равенства значений F на двух возможно разных последовательностях, следует, что равны их значения F после удлинения последовательностей a и b одними и тем же элементом x.

!!! Будете плохо учиться, буду требовать доказательства этой теоремы на экзамене !!!

Доказательство

- **Доказательство**. Необходимость (=>) немедленно следует из определения индуктивности. Если F – индуктивна, то $\forall a,b \in \Omega\ \forall x \in X$
  $F(a*x) = G(F(a),x) = G(F(b),x) = F(b*x)$.

Достаточность

- Для доказательства достаточности (<=) построим требуемое отображение
  $G: Y \times X \to Y$ такое, что $\forall \omega \in \Omega, \forall x \in X \;\; F(\omega * x) = G(F(\omega), x)$. Зададим это отображение формулой
-
- $$G(y,x) = \begin{cases} F(\omega * x), & \text{если существует } \omega \in \Omega \text{ такая, что } F(\omega) = y, \\ y, & \text{иначе.} \end{cases}$$
-
- Корректность этого определения вытекает из заданного в условии теоремы свойства функции *F*.
- В самом деле, пусть найдутся две различные цепочки *a* и *b* такие, что *F(a) = F(b)*. Тогда можно гарантировать, что $F(a * x) = F(b * x)$, что и доказывает корректность определения отображения *G*, ибо *G(y,x)* действительно не зависит от выбора конкретного прообраза элемента *y*.
- Так как $\forall \omega \in \Omega \; \forall x \in X \; F(\omega * x) = G(F(\omega), x)$ для построенного отображения *G,* то теорема доказана.

Использование критерия

- Критерий индуктивности часто используют для доказательства неиндуктивности функций.
- Пример F = «Среднеарифметическое элементов последовательности». Можно показать, что данная функция неиндуктивна. Действительно, знаний значения среднего $F(\omega_{k-1})$ для предыдущей последовательности и нового x недостаточно, чтобы вычислить $F(\omega_{k-1}*x)$:

$$F(\omega_{k-1}) = \frac{1}{k-1}\sum_{i=1}^{k-1} x_i$$

$$F(\omega_{k-1} * x) = F(\omega_k) = \frac{1}{k}\sum_{i=1}^{k} x_i = \frac{1}{k}\sum_{i=1}^{k-1} x_i + \frac{1}{k}x_k = \frac{k-1}{k}F(\omega_{k-1}) + \frac{1}{k}x_k,$$

  но k – неизвестно, а значить вычислить $F(\omega_k)$, зная лишь $F(\omega_{k-1})$ и x, кажется невозможным.

- Пользуясь критерием индуктивности, легко построить контрпример для индуктивности.

  Пусть a = [1,2,3] , a b = [2,2] . В обоих случаях среднее равно 2: то есть

  F(a) = F(b) = 2 . Пусть x = 5 , тогда $F(a * x) = 2{,}75$, а $F(b * x) = 3$, следовательно, $F(a * x) \neq F(b * x)$. Критерий индуктивности не выполнен.

Стационарные значения ИФ

- **Определение**. *Стационарным значением* индуктивной функции $F : \Omega \to Y$, где её функция перехода $G(\omega,x)$, называется значение $s = F(\omega)$ такое, что $G(s,x) = s$ или более строго, но другими словами $\forall \omega \in \Omega, \forall x \in X, F(\omega) = s \Rightarrow F(\omega * x) = s$.

Применение

- Если у индуктивной функции есть стационарное значение, то дальнейшие вычисления можно остановить, так как дальше значение индуктивной функции изменяться не будет.
- Алгоритм на Python может выглядеть следующим образом:

```
F=F0 # стоим в начале последовательности
for x in omega :
    if F!=s :
        F=G(F,x)
    else: break
```

- Или даже лучше так (для тех, кто принципиально не использует break):

```
F=F0
i=0
while (i<len(omega)) and (F!=s):
    F=G(F,omega[i])
    i+=1
```

Пример

- F=«Все элементы последовательности равны 5»
- Определим функцию перехода
  $F(\omega * x) = G(F(\omega), x) = (F(\omega)$, если $x = 5$, иначе False)
- Очевидно, что значение False является стационарным.
- Доопределим F на пустой последовательности. Нам необходимо, чтобы
  $F(\Delta * 5) =$ True и $F(\Delta * x) =$ False , если $x != 5$.
  Тогда True $= F(\Delta * 5) = G(F(\Delta), 5) = F(\Delta)$, то есть $F(\Delta) =$ True.
- После небольшого упрощения получаем алгоритм на Python

```
omega=[5,5,1,5,5] # Для примера
F=True; # инициализация на пустой последовательности
i=0 # в начало последовательности
while (i<len(omega)) and F:
    if (omega[i]==5): F=F
    else: F=False
i+=1
print(F)
```

- Избавимся от if внутри цикла

```
omega=[5,5,1,5,5] # Для примера
F=True; # инициализация на пустой последовательности
i=0 # в начало последовательности
while (i<len(omega)) and F:
    F=(omega[i]==5)
    i+=1
print(F)
```

Продолжение примера

- Сравните последнюю версию алгоритма с менее эффективной, но более короткой и понятной предварительно упрощённой версией, не учитывающей наличие стационарных точек:

```
omega=[5,5,1,5,5]  # для примера

F=True;

for x in omega:

    F=F and (x==5)

print(F)
```

- Такая версия будет пробегать список до конца, несмотря на то, что ответ уже понятен.

Вопросы