

Минобрнауки России
Санкт-Петербургский политехнический университет Петра Великого
Институт компьютерных наук и технологий
Высшая школа интеллектуальных систем и суперкомпьютерных
технологий

Раздаточный материал на тему: "Вербальный анализ решений Шкала
Нормализованных Упорядоченных Различий (ШНУР)"

Руководители:
доцент, к.т.н. А. В. Щукин
ассистент В. А. Пархоменко

Выполнил студенты группы 3540203/90101
Зимин Юрий Геннадьевич

Санкт-Петербург
2020

Оглавление

1	Обзор метода вербального анализа ШНУР	3
1.1	Постановка задачи	3
1.2	Алгоритм поиска лучшей альтернативы	4
2	Программная реализация алгоритма	6
2.1	Обзор реализации алгоритма	6
2.2	Установка и запуск программы	7
2.3	Псевдокод	8
3	Заключение	12

Глава 1

Обзор метода вербального анализа ШНУР

1.1 Постановка задачи

ШНУР решает задачу выбора наилучшей альтернативы.

$K = K_1, K_2, \dots, K_n$ - множество критериев оценки альтернатив.

$K = K^\uparrow \cup K^\downarrow$, где K^\uparrow - подмножество критериев, оценку по которым нужно максимизировать, K^\downarrow - подмножество критериев, оценку по которым нужно минимизировать.

ЛПР – лицо, принимающее решение (эксперт осуществляющий выбор лучшего варианта по заданному алгоритму).

$X_q = x_q^k$ - множество оценок q -го критерия.

$A = A_1, A_2, \dots, A_n$ – множество реальных объектов.

$V = V(A_i)$ - ценность альтернативы A_i для ЛПР.

Необходимо на основе предпочтений ЛПР выделить из множества A лучший объект, соответствующий наибольшему значению априорно неизвестной функции ценности $V(A_i)$.

Сравнивая две альтернативы A_1 и A_2 ЛПР может дать один из трех возможных ответов:

- Альтернатива A_1 предпочтительнее альтернативы A_2
- Альтернатива A_2 предпочтительнее альтернативы A_1
- Альтернативы A_1 и A_2 одинаково предпочтительны

При этом не предусматривается возможность ответа «Не знаю» или «Альтернативы не сравнимы», поскольку считается, что ЛПР обладает сформулированными выше возможностями и всегда может ответить на простой вопрос.

1.2 Алгоритм поиска лучшей альтернативы

1. Объединение близких оценок по критериям. Если оценки одинаково предпочтительны для ЛПР, их можно объединить и заменить средними значениями.
2. Исключение альтернатив с низкими некомпенсируемыми оценками. Если по мнению ЛПР оценка оказалась недопустимо низкой, то вариант A_i можно исключить из рассмотрения.
3. Парные сравнения альтернатив. На данном шаге для каждой пары строится шкала нормализованных упорядоченных различий (рисунок 1). Слева на шкале указывается достоинства варианта A_1 по мере убывания, а справа достоинства варианта A_2 по мере возрастания. ЛПР предлагается сравнивать достоинства начиная с предположительно больших достоинств и недостатков, т.е. двигаясь от концов шкалы по направлению к ее центру. Если одного достоинства не хватает для компенсации достоинства, то добавляется следующее достоинство с меньшим значением.

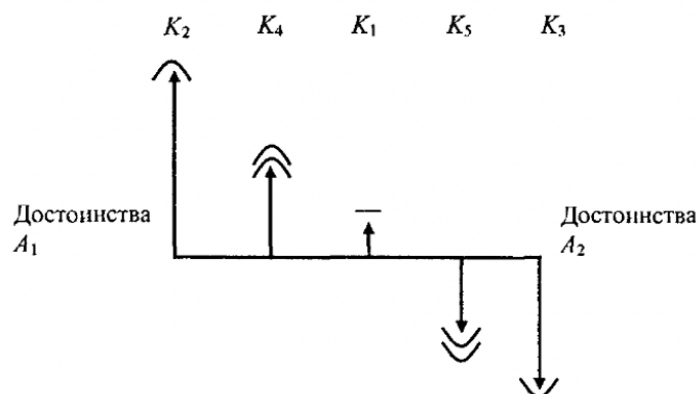


Рисунок 1 – Шкала Нормализованных Упорядоченных различий

Если в выборе лучшей альтернативы имеются качественные критерии, то их необходимо одинаково шкалировать:

- 1 – плохая (низкая, мало);
- 3 – средняя;
- 5 – хорошая (высокая, много)
- 7 – очень хорошая (высокая, много)

В отличие от многокритериального анализа, где выбор лучшей альтернативы без участия ЛПР, а на основе исходных оценок, результат может другим, так как какой-то критерий для ЛПР может быть необходим и иметь первостепенное значение, тогда как в многокритериальном анализе все значимость критериев одинаково.

Глава 2

Программная реализация алгоритма

2.1 Обзор реализации алгоритма

Система предполагает процесс авторизации/регистрации. После авторизации пользователь может создать модель (набор оценок альтернатив по каждому критерию) в следующих форматах:

- Демо-данные (набор готовых данных);
- Загрузить данные с помощью .csv файла;

После проверки корректности исходных данных создаются пары альтернатив и их оценки по критериям шкалируются в соответствии со шкалой нормализованных упорядоченных различий. После пользователю (ЛПР) предлагается ответить на вопросы в соответствии с алгоритмом, описанным ранее. После проведения опроса ЛПР предлагается найденная лучшая альтернатива. Пользователь может посмотреть ранее созданные модели и результаты по ним: лучшая альтернатива по методу ШНУР, лучшая альтернатива по многокритериальному подходу и результаты ответов пользователя (ЛПР).

Критерий	Направление	A1	A2	A3	A4
K1 Количество мест для парковки автомашин	max	400	300	250	150
K2: Наличие поблизости конкурентов	min	1	5	3	5
K3 Плотность населения в радиусе 1км.	max	200	4500	6000	7000
K4 Стоимость участка	min	6	16	12	20
K5 Поток общественного транспорта	max	1	3	5	7
K6 Видимость магазина с главной улицы	max	5	5	3	1
K7 Инфраструктура связей	max	3	3	5	7

Таблица 1 – Демо-данные: варианты постройки магазина

Средства программирования алгоритма:

- Backend – Python 3.9 (Django)
- Frontend – ReactJS
- DB – Sqlite3

Репозиторий проекта: Github

2.2 Установка и запуск программы

Для устновки приложения необходимо установить интерпретатор языка программирования Python (>3.7) (для работы прилоежни] на стороне сервера) и NodeJS (latest) для корректной работы ReactJs на стороне клиента.

Далее необходимо выполнить в терминале (UNIX) следующий набор команд:

- `git clone https://github.com/ziminyuri/verbal_decision_analytics.git`
- `cd backend`
- `pip install -r requirements.txt`
- `python manage.py runserver`
- `cd ..`
- `cd frontend`
- `npm start`

2.3 Псевдокод

Алгоритм 1 Main

```
1:  $M := (K, A)$ 
2:  $n := \text{len}(A)$ 
3:  $k := 1$ 
4: for each  $n$  in  $n$  do
5:   for each  $b$  in  $(k, n)$  do
6:     if  $a$  then Not  $b$ 
7:        $Pairs := \text{createPairs}(a, b, Pairs)$ 
8:     end if
9:   end for
10: end for
11: for each pair in Pairs do
12:    $M := \text{Normalisation}(\text{pair}.A1, \text{pair}.A2, K, M)$ 
13:    $M := \text{Sort}(M)$ 
14:    $W = \text{FindWinner}(\text{pair}.A1, \text{pair}.A2, M, K, W)$ 
15: end for
```

Алгоритм 2 function Normalisation(A1, A2, K, M)

```
1: for each k in K do
2:   a12 := (A1[k] + A2[k]) / 24
3:   if k.direction is max then then
4:     a1 := A1[k]/a12
5:     a2 := A2[k]/a12
6:   else
7:     a1 := 2 - (A1[k] /a12)
8:     a2 := 2- (A2[k]/a12)
9:   end if
10:  M.append(a1 - a2)
11: end for
12: return M
```

Алгоритм 3 function Sort(M)

```
1: s := 1
2: for each k in (length (rows) -s) do
3:   for each j in (lenght (rows) -1) do
4:     if M then[j]<M[j+1]
5:       temp := M[j]
6:       M[j] := M[j+1]
7:       M[j+1] := temp
8:     end if
9:   end for
10: end for
11: return M
```

Алгоритм 4 function FindWinner(M)

```
1:  $n := \text{length}(M)$ 
2:  $param_1, param_2 := []$ 
3:  $flag_1, flag_2 := True$ 
4: for each  $i$  in  $n$  do
5:   for each  $j$  in  $n$  do
6:     if  $flag_2$  is True then
7:        $param_1.append(K[i])$ 
8:        $flag_2 := False$ 
9:     end if
10:    if  $flag_1$  is True then
11:       $param_2.append(K[j])$ 
12:       $flag_1 := False$ 
13:    end if
14:    if  $i \neq 0$  and  $j \neq 0$  and  $i \neq j$  then
15:       $answer := AskQuestion(param_1, param_2)$ 
16:      if (  $answer$  is 1)
17:        if  $length(param_1) == 1$ 
18:           $win1 += 1$ 
19:        else
20:           $param_1, param_2 := []$ 
21:        end if
22:      else if  $answer$  is 2 then
23:        if  $length(param_2) == 1$ 
24:           $win2 += 1$ 
25:        else
26:           $param_1, param_2 := []$ 
27:        end if
28:      else
29:         $param_1, param_2 := []$ 
30:      end if
31:    else if  $i \neq 0$  then
32:       $win1 += 1$ 
33:    else if  $j \neq 0$  then
34:       $win2 += 1$ 
```

Алгоритм 5 function FindWinner(M)

```
35:      else
36:          if ( thenwin1 > win2)
37:              winner = pair.A1
38:          else
39:              winner = pair.A2
40:          end if
41:      end if
42:  end for
43: end for
```

Глава 3

Заключение

Метод ШНУР выбора лучшей альтернативы имеет следующие особенности:

В методе используются достаточно простые процедуры выявления предпочтений ЛПР.

Диалог ведется на понятном для ЛПР языке, причем рассматриваются как качественные, так и количественные оценки альтернатив по критериям.

Метод позволяет сравнить большое число альтернатив при минимальном числе вопросов к ЛПР.

Метод позволяет приспособиться к конкретной задаче (набору альтернатив) и выделить лучшую или предположительно лучшую альтернативу.

Метод позволяет ЛПР получить объяснения сделанному выбору путем предъявления тех ответов, которые привели к полученному результату.

Во многих случаях метод помогает ЛПР в выборе лучшей альтернативы. Однако метод не всегда приводит к нужному результату, т. к. альтернативы могут быть несравнимы.

Литература

- [1] О.И.Ларичев Вербальный анализ решений; [отв. ред. А.Б.Петровский];
Ин-т системы анализа РАН. -М.: Наука, 2006. 181 с.