

## 1. APPLICATION OF THE LAW OF MINIMUM FOR TCP

# Закон минимума и анализ различий для приоритизации регрессионных тестов

**Annotation.** We present the classical VIKOR multi-criteria decision-making method together with two important extensions: interval VIKOR and fuzzy VIKOR. We discuss their theoretical foundations, applicability, advantages and limitations, and provide comparative insight into decision stability under uncertainty. A computational example illustrates interval and fuzzy modifications of VIKOR evaluation and ranking.

**Keywords.** VIKOR, MCDM, interval analysis, fuzzy sets.

**Аннотация.** В работе рассматривается классический метод многокритериального принятия решений VIKOR, а также две его ключевые модификации: интервальный и нечеткий VIKOR. Излагаются теоретические основы методов, области применимости, особенности и ограничения, а также проводится сравнение устойчивости решений в условиях неопределенности. На иллюстративном примере демонстрируется выполнение и результаты интервальной и нечеткой модификации VIKOR.

**Ключевые слова.** VIKOR, многокритериальное принятие решений, интервальный анализ, нечеткие множества.

### 1.1. Введение

Метод VIKOR (*VIseKriterijumska Optimizacija I Kompromisno Resenje*) был предложен Сергием Опричевичем как подход к многокритериальному принятию решений, ориентированный на нахождение компромиссного решения, максимально близкого к идеальной точке. В отличие от методов, стремящихся к парето-оптимальным решениям, VIKOR фокусируется на ситуациях, где компромисс между критериями является предпочтительным.

Однако классическая версия метода предполагает, что значения критериев заданы в виде точных чисел. В реальных задачах часто присутствуют различные

формы неопределённости: интервальные оценки показателей, экспертные оценки в форме языковых термов, нечеткие треугольные числа и др. Это приводило к необходимости расширить VIKOR.

Существуют две важные модификации:

- **интервальный VIKOR**, предложенный в работе [4], адаптирующий алгоритм к критериям, представленным через интервалы;
- **fuzzy VIKOR**, описанный, например, в работах [2; 5], позволяющий работать с лингвистическими и нечеткими данными.

Сравнительный анализ различных версий VIKOR выполнен в статье [1], где показано, что интервальная и нечеткая модификации позволяют повысить устойчивость решений при наличии неопределенности.

Цель данной работы — систематизировать основные идеи интервальной и нечеткой модификаций VIKOR, показать отличие от классического алгоритма и продемонстрировать работу интервального и нечеткого VIKOR на примере, написанном на языке программирования python.

## 1.2. Обзор метода VIKOR

Метод VIKOR относится к классу многокритериальных подходов принятия решений и был предложен Опричевичем как способ получения компромиссного решения в ситуациях, где необходимо учитывать несколько конфликтующих критериев. Первоначально метод создавался для инженерных задач, требующих выбора оптимального варианта среди множества альтернатив с разнородными характеристиками. Основная идея метода заключается в нахождении решения, минимизирующего расстояние до идеальной точки и одновременно обеспечивающего баланс между улучшением средних показателей и уменьшением максимального отклонения. Метод основан на компромиссном программировании, но отличается тем, что явно использует процедуру поиска решения, приемлемого для большинства заинтересованных сторон, что делает его особенно полезным в задачах, предполагающих наличие противоречивых критериев.

VIKOR применяется в ситуациях, где требуется учитывать не только максимальную близость альтернативы к идеальному варианту, но и степень её доминирования по самому неблагоприятному критерию. Поэтому метод получил

широкое применение в инженерии, инвестиционном анализе, экологии, управлении проектами и других областях, где невозможно достичь одновременной оптимизации всех целей. Преимуществом метода является способность сочетать стратегии минимизации наихудшего отклонения и максимизации интегральной пользы, что позволяет получать взвешенное компромиссное решение.

### **1.2.1. Математическая постановка задачи**

Пусть задано множество альтернатив  $A = \{A_1, A_2, \dots, A_m\}$ , каждая из которых оценивается по набору критериев  $C = \{c_1, c_2, \dots, c_n\}$ . Определена матрица оценок  $f_{ij}$ , где  $f_{ij}$  — значение альтернативы  $A_i$  по критерию  $c_j$ . Для каждого критерия задан тип: выгодный (benefit), для которого большие значения предпочтительны, или затратный (cost), для которого предпочтительны меньшие значения. Также каждому критерию назначен весовой коэффициент  $w_j$ , удовлетворяющий условию

$$\sum_{j=1}^n w_j = 1.$$

Для каждого критерия определяются лучшие и худшие значения среди всех альтернатив. Для выгодных критериев применяются выражения

$$f_j^* = \max_i f_{ij}, \quad f_j^- = \min_i f_{ij},$$

а для затратных критериев

$$f_j^* = \min_i f_{ij}, \quad f_j^- = \max_i f_{ij}.$$

Затем вычисляются две меры отклонения альтернативы от идеального решения. Первая мера представляет собой взвешенную сумму отклонений:

$$S_i = \sum_{j=1}^n w_j \frac{f_j^* - f_{ij}}{f_j^* - f_j^-}.$$

Вторая мера характеризует максимальное взвешенное отклонение по одному критерию:

$$R_i = \max_{1 \leq j \leq n} \left\{ w_j \frac{f_j^* - f_{ij}}{f_j^* - f_j^-} \right\}.$$

Для получения компромиссного решения определяется интегральный индекс  $Q_i$ :

$$Q_i = v \cdot \frac{S_i - S^*}{S^- - S^*} + (1 - v) \cdot \frac{R_i - R^*}{R^- - R^*},$$

где

$$S^* = \min_i S_i, \quad S^- = \max_i S_i, \quad R^* = \min_i R_i, \quad R^- = \max_i R_i,$$

а параметр  $v \in [0,1]$  отражает предпочтения относительно стратегий компромисса. Значение  $v = 0.5$  соответствует равному учёту обеих мер, приближение  $v \rightarrow 1$  подчеркивает ориентацию на минимизацию суммарных отклонений, а  $v \rightarrow 0$  — на минимизацию наибольшего отклонения.

Итоговое решение определяется альтернативой с минимальным значением  $Q_i$ , однако процедура выбора включает дополнительные условия устойчивости. Если различие между лучшими альтернативами недостаточно велико или ранжирование по метрикам  $S$  и  $R$  противоречит индексу  $Q$ , формируется не одна альтернатива, а набор компромиссных решений.

### 1.3. Интервальная модификация метода VIKOR

Как уже упоминалось в 1.2, классический метод VIKOR предполагает, что оценки всех альтернатив по каждому критерию представлены точечными значениями. Однако в реальных задачах многокритериального анализа нередко встречаются ситуации, когда значения критериев заданы неточно, являются приближёнными или варьируются в некотором допустимом диапазоне. Такие ситуации типичны для инженерных, экономических и экологических задач, в которых невозможно получить точные числа из-за экспертной неопределённости, вариативности данных и измерительных ошибок.

Для учёта неопределённости авторы [4] предложили интервальную модификацию метода VIKOR, в которой вместо точечных значений используются интервальные оценки

$$f_{ij} = [\underline{f}_{ij}, \bar{f}_{ij}],$$

где  $\underline{f}_{ij}$  и  $\bar{f}_{ij}$  обозначают нижнюю и верхнюю границы возможного значения критерия. Такой подход позволяет работать с более гибкой моделью данных и принимать решения, устойчивые к колебаниям входной информации.

### 1.3.1. Математическая постановка интервального VIKOR

Пусть задано множество альтернатив  $A = \{A_1, \dots, A_m\}$ , множество критериев  $C = \{c_1, \dots, c_n\}$  и интервальная матрица оценок

$$f_{ij} = [\underline{f}_{ij}, \bar{f}_{ij}].$$

Каждый критерий относится к типу *выгодный* или *затратный*, а веса  $w_j$  удовлетворяют условию  $\sum_{j=1}^n w_j = 1$ .

Для интервальных данных определяются *интервальные идеальные и антиидеальные точки*. Для выгодного критерия:

$$f_j^* = \left[ \max_i \underline{f}_{ij}, \max_i \bar{f}_{ij} \right], \quad f_j^- = \left[ \min_i \underline{f}_{ij}, \min_i \bar{f}_{ij} \right].$$

Для затратного критерия:

$$f_j^* = \left[ \min_i \underline{f}_{ij}, \min_i \bar{f}_{ij} \right], \quad f_j^- = \left[ \max_i \underline{f}_{ij}, \max_i \bar{f}_{ij} \right].$$

Далее необходимо определить интервальные меры отклонения. Интервальный нормализованный разрыв альтернативы  $A_i$  по критерию  $c_j$  определяется как

$$D_{ij} = \left[ \frac{f_j^{*(\text{low})} - \bar{f}_{ij}}{f_j^{*(\text{low})} - f_j^{-(\text{high})}}, \quad \frac{f_j^{*(\text{high})} - \underline{f}_{ij}}{f_j^{*(\text{high})} - f_j^{-(\text{low})}} \right],$$

где использование противоположных концов интервалов обеспечивает *наиболее пессимистичную* и *наиболее оптимистичную* оценки, согласуясь с подходом Sayadi et al.

Интервальные показатели  $S_i$  и  $R_i$  вычисляются как:

$$S_i = \left[ \sum_{j=1}^n w_j D_{ij}^{(\text{low})}, \quad \sum_{j=1}^n w_j D_{ij}^{(\text{high})} \right],$$

$$R_i = \left[ \max_j w_j D_{ij}^{(\text{low})}, \quad \max_j w_j D_{ij}^{(\text{high})} \right].$$

### 1.3.2. Особенности метода

Интервальный VIKOR сохраняет структуру исходного метода, но адаптирует все вычисления к интервальной арифметике. Основные особенности:

- вместо точек используются интервалы значений, что делает метод чувствительным к неопределённости;
- нормализация использует “наиболее широкое” сочетание концов интервалов, что гарантирует корректную оценку в условиях неполных данных;
- результаты ранжирования также представляются интервалами и требуют процедуры сравнения интервальных чисел;
- метод может порождать частичные или нестрогие порядки, когда интервалы  $Q_i$  альтернатив пересекаются.

### ***1.3.3. Недостатки и ограничения***

Несмотря на преимущества, интервальная модификация имеет ряд недостатков:

- усложнение вычислений из-за необходимости поддерживать интервальную арифметику;
- увеличение числа случаев, когда альтернативы оказываются несравнимыми из-за пересечения интервалов;
- зависимость качества результата от корректного задания интервалов экспертами.

Тем не менее модификация доказала свою эффективность в задачах с высокой неопределенностью и стала основой для дальнейших расширений метода, включая fuzzy модификацию [2; 5], о которой пойдет речь в 1.4.

### ***1.3.4. Реализация интервальной модификации на Python***

Для практической демонстрации интервальной модификации метода VIKOR был использован Python-код, реализующий шаги, описанные в работе [4]. В частности, реализация учитывает следующие ключевые моменты:

- каждая альтернатива  $A_i$  имеет интервальные оценки по каждому критерию  $c_j$ :  $f_{ij} = [\underline{f}_{ij}, \overline{f}_{ij}]$ ;
- вычисляются интервальные идеальные и антиидеальные точки  $f_j^*$  и  $f_j^-$ , учитывая тип критерия (выгода или затраты);
- рассчитываются интервальные меры отклонения  $D_{ij}$ , нормализованные с использованием противоположных концов интервалов для пессимистичных и оптимистичных оценок;

D. агрегируются интервальные показатели  $S_i$  и  $R_i$  по формулам:

$$S_i = \sum_{j=1}^n w_j D_{ij}, \quad R_i = \max_j w_j D_{ij};$$

E. определяется интервальная величина  $Q_i$  как взвешенная комбинация  $S_i$  и  $R_i$ , с учётом коэффициента стратегии  $v$ ;

F. ранжирование альтернатив производится по центрам интервалов  $Q_i$ , при необходимости формируется компромиссный набор лучших альтернатив.

Данная реализация полностью согласуется с формулами, представленными в пункте 1.3.1, и позволяет на практике оценивать устойчивость решений к колебаниям входных данных.

На приведённом примере, включающем четыре альтернативы и три критерия, были получены следующие результаты:

*Таблица 1.1*

**Идеальная и надирная точки интервалов**

Критерий	$f_j^*$	$f_j^-$
1	[0.800, 1.000]	[0.500, 0.700]
2	[0.800, 1.000]	[0.500, 0.700]
3	[0.500, 0.700]	[0.800, 1.000]

*Таблица 1.2*

**Интервальные метрики S, R и Q для альтернатив**

Альтернатива	S-интервал	R-интервал	Q-интервал	Q_center
A1	[0.6667, 0.6667]	[0.3000, 0.3000]	[0.7500, 0.7500]	0.7500
A2	[0.4000, 0.4000]	[0.3000, 0.3000]	[0.3500, 0.3500]	0.3500
A3	[0.6000, 0.6000]	[0.4000, 0.4000]	[0.9000, 0.9000]	0.9000
A4	[0.3333, 0.3333]	[0.2000, 0.2000]	[0.0000, 0.0000]	0.0000

*Таблица 1.3*

**Ранжирование альтернатив по интервальной метрике Q**

Ранжирование	Лучшая альтернатива
A4 → A2 → A1 → A3	A4

Анализ полученных результатов показывает, что интервальная модификация позволяет:

- выявить компромиссные альтернативы даже при колебаниях значений критериев;
- формировать устойчивое ранжирование, при этом лучшая альтернатива (A4) сохраняется при различных значениях коэффициента стратегии  $v$ ;
- настраивать степень влияния предпочтений через параметр  $v$ , что отражено в анализе чувствительности (лучшие альтернативы изменяются при  $v = 0.7$  и 1.0, формируя компромиссные наборы A4+A2).

## 1.4. Нечёткая (fuzzy) модификация метода VIKOR

В ряде практических задач исходные данные содержат неопределённость, которую невозможно корректно описать только интервалами. В таких случаях применяется нечеткая (fuzzy) модификация метода VIKOR, предложенная в [3]. В этой версии критерии и веса допускают неопределенность, выражаемую *треугольными нечеткими числами* (TFN). Нечеткие оценки позволяют моделировать неточность экспертических суждений, а также различать нижнюю, наиболее вероятную и верхнюю оценку каждого параметра.

Треугольное нечеткое число определяется как

$$\tilde{x} = (x^L, x^M, x^U),$$

где  $x^L$ ,  $x^M$  и  $x^U$  соответствуют нижней, модальной и верхней оценкам соответственно. Все операции в алгоритме выполняются покомпонентно, что обеспечивает корректность результата в нечеткой среде.

### 1.4.0.1. Определение нечетких идеального и антиидеального решений

Аналогично классическому VIKOR для каждого критерия определяются лучшие и худшие значения. Отличие заключается в том, что каждая оценка является TFN, а операции  $\max$  и  $\min$  применяются к соответствующим компонентам.

Для выгодных критериев:

$$\tilde{f}_j^* = \left( \max_i f_{ij}^L, \max_i f_{ij}^M, \max_i f_{ij}^U \right), \quad \tilde{f}_j^- = \left( \min_i f_{ij}^L, \min_i f_{ij}^M, \min_i f_{ij}^U \right).$$

Для затратных критериев:

$$\tilde{f}_j^* = \left( \min_i f_{ij}^L, \min_i f_{ij}^M, \min_i f_{ij}^U \right), \quad \tilde{f}_j^- = \left( \max_i f_{ij}^L, \max_i f_{ij}^M, \max_i f_{ij}^U \right).$$

#### 1.4.0.2. Нормирование нечетких расстояний

Нечеткое расстояние альтернативы от идеального решения также является треугольным числом. В работе [3] предложена нормировка вида

$$\tilde{d}_{ij} = \left( \frac{f_j^L - f_{ij}^U}{f_j^U - f_j^-}, \frac{f_j^M - f_{ij}^M}{f_j^M - f_j^-}, \frac{f_j^U - f_{ij}^L}{f_j^L - f_j^-} \right),$$

что гарантирует сохранение структуры TFN.

#### 1.4.0.3. Нечеткие показатели $S_i$ и $R_i$

Аналоги интегральной и максимальной меры отклонения определяются как

$$\tilde{S}_i = \sum_{j=1}^n (\tilde{w}_j \otimes \tilde{d}_{ij}),$$

$$\tilde{R}_i = \max_j (\tilde{w}_j \otimes \tilde{d}_{ij}),$$

где  $\otimes$  обозначает покомпонентное умножение TFN, а максимум по критериям также определяется покомпонентно:

$$\max((a^L, a^M, a^U), (b^L, b^M, b^U)) = (\max(a^L, b^L), \max(a^M, b^M), \max(a^U, b^U)).$$

#### 1.4.0.4. Дефазификация и вычисление итогового показателя

Поскольку результатом являются нечеткие числа, необходимо дефазифицировать  $\tilde{S}_i$  и  $\tilde{R}_i$ . В [3] применяется классическая дефазификация треугольного числа:

$$\text{Defuzz}(\tilde{x}) = \frac{x^L + x^M + x^U}{3}.$$

Таким образом,

$$S_i = \text{Defuzz}(\tilde{S}_i), \quad R_i = \text{Defuzz}(\tilde{R}_i).$$

После этого итоговый компромиссный показатель вычисляется по стандартной формуле VIKOR:

$$Q_i = \nu \frac{S_i - S^*}{S^- - S^*} + (1 - \nu) \frac{R_i - R^*}{R^- - R^*},$$

где

$$S^* = \min_i S_i, \quad S^- = \max_i S_i, \quad R^* = \min_i R_i, \quad R^- = \max_i R_i,$$

а  $\nu \in [0,1]$  — параметр компромисса.

#### 1.4.0.5. Итоговое ранжирование

Окончательные решения выбираются по тем же критериям приемлемости преимущества и стабильности, что и в оригинальном методе VIKOR. Отличие заключается только в том, что все сравнения проводятся над дефазифицированными значениями.

#### 1.4.1. Реализация fuzzy модификации на Python

Для практической демонстрации нечеткой модификации метода VIKOR реализован класс FuzzyVIKOR, описывающий процедуру, предложенную в работе Opricovic (2011) [3]. Ниже перечислены ключевые шаги реализации и их соответствие формальному описанию метода.

- A. Представление данных: каждая альтернатива  $A_i$  задаётся набором треугольных нечетких чисел (TFN) по каждому критерию  $c_j$ :

$$\tilde{f}_{ij} = (f_{ij}^L, f_{ij}^M, f_{ij}^U).$$

В коде — трёхмерный массив `matrix` формы  $(m,n,3)$ .

- B. Идеальная и антиидеальная точки: для каждого критерия покомпонентно вычисляются  $\tilde{f}_j^*$  и  $\tilde{f}_j^-$  (L/M/U):

$$\tilde{f}_j^* = \begin{cases} (\max_i f_{ij}^L, \max_i f_{ij}^M, \max_i f_{ij}^U), & \text{если } c_j \text{ — выгодный} \\ (\min_i f_{ij}^L, \min_i f_{ij}^M, \min_i f_{ij}^U), & \text{если } c_j \text{ — затратный} \end{cases}$$

(в коде — массивы `ideal_point` и `nadir_point`).

- C. Нормализация нечетких расстояний: для каждой альтернативы и критерия формируется тройка компонент  $\tilde{d}_{ij} = (d_{ij}^L, d_{ij}^M, d_{ij}^U)$  по формулам,

предложенным в [3]:

$$\tilde{d}_{ij} = \left( \frac{f_j^{*L} - f_{ij}^U}{f_j^{*U} - f_j^{-L}}, \frac{f_j^{*M} - f_{ij}^M}{f_j^{*M} - f_j^{-M}}, \frac{f_j^{*U} - f_{ij}^L}{f_j^{*L} - f_j^{-U}} \right).$$

В коде нормализованные значения сохраняются в массиве `normalized`.

- D. Вычисление нечетких  $S$  и  $R$ : покомпонентное умножение нечетких весов  $\tilde{w}_j$  и нормализованных расстояний, агрегирование даёт:

$$\tilde{S}_i = \sum_{j=1}^n \tilde{w}_j \otimes \tilde{d}_{ij}, \quad \tilde{R}_i = \max_j (\tilde{w}_j \otimes \tilde{d}_{ij}).$$

В реализации это массивы `S_values` и `R_values` размерности  $(m,3)$ .

- E. Дефазификация: треугольные числа переводятся в скаляры; в классическом варианте Opricovic используется  $\text{Defuzz}(\tilde{x}) = (x^L + x^M + x^U)/3$ . В коде применяется дефазификация (в коде производится усреднение компонент — в реализации присутствует вариант  $(L + 2M + U)/4$ ; для строгого соответствия статье рекомендуется использовать  $(L + M + U)/3$ ). Дефазифицированные  $S_i$  и  $R_i$  обозначаются как скаляры.
- F. Вычисление скалярного  $Q_i$  и ранжирование: после дефазификации применяются стандартные формулы VIKOR:

$$Q_i = v \frac{S_i - S^*}{S^- - S^*} + (1 - v) \frac{R_i - R^*}{R^- - R^*},$$

где  $S^* = \min_i S_i$ ,  $S^- = \max_i S_i$ ,  $R^* = \min_i R_i$ ,  $R^- = \max_i R_i$ .

- G. Постпроцесс: в коде реализована проверка условий *acceptable advantage* и *stability* и механизм *trade-off* (уступки) с пересчётом весов и повторным выполнением шагов агрегации и ранжирования.

#### 1.4.1.1. Замечания по параметрам реализации

- `weights` в реализации ожидаются в виде TFN весов формы  $(n,3)$ . Если исходные веса заданы как скаляры, их можно расширить до TFN, положив все три компоненты равными.
- Параметр  $v \in [0,1]$  задаёт стратегию компромисса (в коде — аргумент `v`).
- Механизм уступки (*trade-off*) в коде позволяет задать либо вектор `tr_custom_values`, либо вычислить его автоматически, после чего пересчитываются веса и алгоритм прогоняется повторно.

Алгоритм был запущен на тестовом наборе (6 альтернатив, 4 критерия, детали см. исходный .ipynb). Для данного прогона получены следующие истинные идеальные и надирные точки (покомпонентно, L/M/U):

*Таблица 1.4*

**Идеальная и надирная точки (TFN) для примера**

Критерий $j$	$\tilde{f}_j^*$	$\tilde{f}_j^-$
1 (затратный)	(20.00, 21.06, 24.00)	(44.54, 46.89, 56.27)
2 (выгодный)	(3.26, 4.08, 4.08)	(2.25, 2.50, 2.62)
3 (затратный)	(6.00, 6.00, 6.00)	(60.00, 62.00, 68.00)
4 (затратный)	(0.00, 0.00, 0.00)	(10.00, 10.00, 10.00)

Из прогона выведены итоговые показатели (дефазифицированные) и решение:

- Допустимое превосходство (acceptable\_advantage): **True**.
- Допустимая стабильность (acceptable\_stability): **True**.
- Компромиссное решение: **альтернатива А3**.

(Полный детализированный вывод — в файле .ipynb; при необходимости можно вставить таблицы с полными векторами  $\tilde{S}_i$ ,  $\tilde{R}_i$ ,  $\tilde{Q}_i$  и их дефазифицированными значениями.)

#### *1.4.1.2. Краткий анализ результатов*

- Модификация корректно обрабатывает неопределённость оценок (TFN) — итоговое ранжирование учитывает и средние, и крайние оценки альтернатив.
- Возможность задания `tr_custom_values` и перерасчёта весов позволяет моделировать различные сценарии trade-off и смотреть, как они влияют на итоговый компромисс.
- Важно учитывать метод дефазификации: переход от TFN к скалярам существенно влияет на  $S_i, R_i, Q_i$ . По умолчанию в литературе используется  $(L + M + U)/3$ ; если в реализации применяется иная формула — это необходимо явно указать и обосновать.

#### *1.4.1.3. Псевдокод реализованного алгоритма (кратко)*

- A. Ввести  $\tilde{f}_{ij}$  и  $\tilde{w}_j$  (TFN).
- B. Для каждого  $j$  вычислить  $\tilde{f}_j^*$  и  $\tilde{f}_j^-$ .
- C. Для всех  $i, j$  получить  $\tilde{d}_{ij}$  по формулам (L,M,U).
- D. Вычислить  $\tilde{S}_i$  и  $\tilde{R}_i$  покомпонентно.
- E. Дефазифицировать  $\tilde{S}_i, \tilde{R}_i$  в скаляры  $S_i, R_i$ .
- F. Посчитать  $S, S^-, R, R^-$  и скалярный  $Q_i$ ; ранжировать.
- G. Проверить условия acceptable advantage / stability и, при необходимости, применить trade-off (пересчитать веса и повторить шаги 3–6).

#### *1.4.2. Сравнение с базовыми методами*

В среднем по рассматриваемым проектам значения APFD для различных методов приоритизации распределились следующим образом (по данным статьи):

*Таблица 1.5*

**Средние значения APFD для разных методов**

Метод	Средний APFD
Additional-Diff	0,9805
Dis-LoM	0,9727
Total-Diff	0,9625
LoM	0,9336
Additional	0,8656
Random (avg)	0,8236
Total	0,8062

Из таблицы 1.5 видно, что методы, учитывающие diff (Additional-Diff, Total-Diff), демонстрируют более высокие значения APFD по сравнению с классическими Total и Additional. Предложенный метод Dis-LoM оказывается близок к лучшему baseline Additional-Diff и превосходит остальные методы, в том числе стандартные coverage-based и случайный порядок.

#### *1.4.3. Краткий анализ результатов*

Результаты показывают, что:

- интеграция информации о распространении изменений (через цепь Маркова) и закона минимума позволяет более точно оценивать важность тестов по отношению к изменённому коду;
- учёт различий между тестами (Dis-LoM) уменьшает дублирование и ускоряет обнаружение дефектов в разных частях системы;
- Dis-LoM демонстрирует стабильное улучшение по сравнению с LoM и конкурентоспособен с лучшими diff-ориентированными baseline.

## 1.5. Выводы

В работе рассмотрен подход к приоритизации регрессионных тестов, основанный на сочетании анализа различий, закона минимума и модели цепей Маркова. В отличие от классических coverage-based методов, данный подход:

- моделирует распространение влияния изменений по графу вызовов;
- учитывает чувствительность методов к параметрам через forward slicing;
- опирается на закон минимума для выявления “узких мест” в покрытии;
- снижает избыточность тестов за счёт анализа непохожести покрытия.

Экспериментальные результаты, полученные авторами исходной статьи на наборе Defects4J, показывают, что метрика Dis-LoM обеспечивает высокие значения APFD и сопоставима с лучшими diff-ориентированными методами, превосходя при этом классические Total и Additional, а также случайный порядок.

Перспективными направлениями развития подхода являются:

- расширение анализа на другие языки программирования и типы проектов;
- интеграция исторической информации о сбоях тестов;
- оптимизация вычислительной стоимости построения графов и модели цепи Маркова для очень больших систем.

## Библиографический список

1. Chatterjee P., Chakraborty S. A comparative analysis of VIKOR method and its variants // Decision Science Letters. — 2016. — Т. 5, № 4. — С. 469—486. — DOI 10.5267/j.dsl.2016.5.004.

2. *Liu P., Qin X.* An Extended VIKOR Method for Decision Making Problem with Interval-Valued Linguistic Intuitionistic Fuzzy Numbers Based on Entropy // *Informatica*. — 2017. — T. 28, № 4. — C. 665—685. — DOI 10.15388/Informatica. 2017.151.
3. *Opricovic S.* Fuzzy VIKOR with an application to water resources planning // *Expert Systems with Applications*. — 2011. — T. 38, № 10. — C. 12983—12990. — DOI 10.1016/j.eswa.2011.04.097. — URL: <https://www.sciencedirect.com/science/article/pii/S0957417411006245>.
4. *Sayadi M. K., Heydari M., Shahanaghi K.* Extension of VIKOR method for decision making problem with interval numbers // *Applied Mathematical Modelling*. — 2009. — T. 33, № 5. — C. 2257—2262. — DOI 10.1016/j.apm.2008.06.002.
5. *Wan S.-P.* The extended VIKOR method for multi-attribute group decision making with triangular intuitionistic fuzzy numbers // *Knowledge-Based Systems*. — 2013. — T. 52. — C. 65—77. — DOI 10.1016/j.knosys.2013.06.019.