

УДК 61:004.651(075.8)

І.Є.Андрушак

Луцький національний технічний університет

## ПРОГРАМНА РЕАЛІЗАЦІЯ МЕТОДУ DECISION CURVE ANALISYS (DCA) В СИСТЕМІ ПІДТРИМКИ РІШЕНЬ ДЛЯ ЗАДАЧ КЛІНІЧНОЇ МЕДИЦИНИ

**І.Є.Андрушак. Програмна реалізація методу decision curve analisys (dca) в системі підтримки рішень для задач клінічної медицини.** Робота присвячена адаптації до задач клінічної медицини та програмній реалізації методу DCA, що полягає у виборі оптимальної стратегії лікування виходячи із значень сукупної функції корисності. Метод має доступну графічну інтерпретацію, що робить можливим його використання на практиці.

**Ключові слова:** прийняття рішень, дерево рішень, Java, SQL.

**И.Е.Андрушак Програмная реализация метода decision curve analisys (dca) в системе поддержки решений для задач клинической медицины.** Работа посвящена адаптации к задачам клинической медицины и программной реализации метода DCA, который заключается в выборе оптимальной стратегии лечения исходя из значений совокупной функции полезности. Метод имеет доступную графическую интерпретацию, что делает возможным его использование на практике.

**Ключевые слова:** принятие решений, дерево решений, Java, SQL.

**I.Ye.Andruschak. Program implementation of decision curve analisys (dca) in support system solutions for problems clinical medicine.** The work is devoted to the problems of adaptation of clinical medicine and program implementation of DCA, which is to select the optimal treatment strategy based on the values of the aggregate utility function. The method has Graphical interpretation that makes possible its use in practice.

**Keywords:** decision making, decision tree, Java, SQL.

Клінічна медицина традиційно поділяється на діагностику, лікування та профілактику. З точки зору прийняття рішень діагностика та профілактика мають схоже призначення: клініцист має деяку інформацію і хоче знати, як вона пов'язана з реальним станом пацієнта – теперішнім (діагностика) або майбутнім (профілактика).

Ця інформація може бути у вигляді результатів обстежень (наприклад, ЕКГ, УЗД і ін.) або статистичної прогностичної моделі, що включає ряд змінних. Останні досягнення в галузі медичної біології, особливо геноміки, біоінформатики пропонують для діагностики та профілактики молекулярні маркери.

Традиційні біостатистичні методи для оцінки результатів обстежень, моделей та молекулярних маркерів є обмеженими. Традиційний біостатистичний підхід концентрується на точності, калібруванні та дискримінативності використовуючи такі метрики як чутливість, специфічність та площа під кривою (AUC, AUROC). Такі методи є математично простими, можуть застосовуватися як для неперервних, так і бінарних вхідних даних та в загальному мають інтуїтивну інтерпретацію.

Однак вони мають недостатню клінічну релевантність. Наприклад, наскільки великою повинна бути площа AUC, щоб обґрунтувати клінічне застосування прогностичної моделі? Або, наприклад, розглянемо випадок, коли результати обстеження збільшили специфічність на 10%, але зменшили чутливість на 5% порівняно із стандартною діагностичною методикою – яку ж діагностичну методику використовувати?

Відповідь на такі запитання залежить від наслідків конкретних клінічних рішень, прийнятих в результаті обстеження, моделі або маркеру. Так, коли методика обстеження була специфічнішою, але менш чутливою порівняно із стандартною, її значимість залежить від шкоди випадку пропуску захворювання порівняно із шкодою від лікування здорового пацієнта.

Традиційні методи прийняття рішень мають ряд суттєвих недоліків, що обмежують їх застосування в клінічних дослідженнях [Vickers, 2008]. По-перше, математичні методи можуть бути складними для пояснення клінічній аудиторії [4-6]. По-друге, багато прогностичних показників в медицині є неперервними, як наприклад, ймовірність на основі прогностичної моделі для рівня сироватки крові – такі прогностичні показники складно впровадити в процес прийняття рішень. По-третє, і найбільш критично, раціональний аналіз прийняття рішень переважно вимагає інформацію, яка є недоступною в реальних наборах даних.

Тому метою роботи є адаптувати для прийняття клінічних рішень та програмно реалізувати метод DCA, який володіє доступною графічною візуалізацією.

**Метод DCA.** Метод ґрунтується на принципі, що відносні шкоди від помилкових позитивів та помилкових негативів можуть бути виражені в термінах порогової ймовірності. Наприклад, пацієнту буде призначено біопсію, якщо ризик раку простати складає більше 20% і навпаки – не буде призначено, якщо ризик менше 20%. У цьому випадку можна показати, що шкода від пропущеного раку простати в чотири рази є більшою від шкоди, завданої від непотрібної біопсії. Тобто відношення шкод є відношенням невизначеностей для порогової ймовірності. Дана порогова ймовірність використовується для моделювання клінічних наслідків від вірних та помилкових позитивів на основі сукупної функції корисності (Net benefit):

$$B_{net} = \frac{TP}{n} - \frac{FP}{n} \left( \frac{p_t}{1-p_t} \right). \quad (1)$$

Тут  $B_{net}$  – сукупна функція корисності,  $TP$ ,  $FP$  – вірно та помилково позитивні,  $n$  – загальна кількість пацієнтів,  $p_t$  – порогова ймовірність.

Далі шляхом зміни значення порогової ймовірності можемо побудувати «криву рішень» для будь-якої статистичної моделі, методики обстежень або молекулярних маркерів.

Слід зазначити, що модель, результат обстеження або маркер, що вивчаються, попередньо повинні бути конвертовані у ймовірність (наприклад раку при біопсії), що позначається  $\hat{p}$ : для бінарних обстежень ці ймовірності становлять 1 або 0 для позитивних та негативних результатів відповідно; для молекулярних маркерів рівні маркерів конвертуються у ймовірності на основі логістичної регресії.

Метод DCA може бути далі представлений у вигляді такого алгоритму [Vickers, 2006].

**Вхідні дані:** інформація про пацієнтів: кількість  $n$ , вектор діагнозів:  $d = (d_1, \dots, d_n)^T$  та матриця результатів обстежень:  $I = (I_1, \dots, I_n) \in R^{m \times n}$ .

Тут  $d_i = \begin{cases} 1, & \text{захворювання є} \\ 0, & \text{захворювання немає} \end{cases}$ ,  $I_i \in R^m$  – діагностичні дані для кожного пацієнта, що

можуть бути перетворені у ймовірнісне значення, наприклад на основі логістичної регресії.

**Вихідні дані:** функції сукупної корисності для усіх моделей та стратегій: лікувати всіх, лікувати нікого як функції від порогової ймовірності.

**Метод:**

1. Вибираємо порогову ймовірність  $p_t$ .
2. Вважаємо, що пацієнт є позитивним, якщо  $\hat{p} \geq p_t$ .
3. Обчислюємо кількості вірно та помилково позитивних.
4. Обчислюємо сукупну функцію корисності.
5. Повторюємо кроки 1-4 на інтервалі значень  $p_t$ .
6. Повторюємо кроки 1-5 для усіх моделей та для стратегії лікувати всіх пацієнтів (тобто  $\hat{p}=1$ ).
7. Для кожного значення порогової ймовірності вибираємо стратегію або модель з найбільшим значенням сукупної корисності.

Зауважимо, що метод може бути застосований не лише до чисельних вхідних даних про пацієнта, але й мультимедійних, наприклад, ЕКГ, ЕЕГ, УЗД, КТ, МРТ і ін., які повинні попередньо бути виражені в ймовірнісних значеннях.

**Програмна реалізація.** Метод реалізовано в програмному середовищі Netbeans в складі проекту HealthInsurance. З цією метою в пакеті healthinsurance створено клас Decision Curve Analysis, що походить з абстрактного класу medbioinvestigations.graph.GraphConstruction [7]. Вхідні дані зберігаються у вигляді хеш-таблиці m\_htDCAdata, що має таку структуру:

Тип ключа	Int
Тип об'єкта	Node
Структура об'єкта	Перший нащадок – діагноз $d_i$ ; другий нащадок – діагностичні дані $l_i$ .

Дані для побудови кривих рішень зберігаються у двовимірному масиві `m_aNetBenefit` типу `double[][]`. У методі `init()` здійснено розрахунок сукупної корисності для стратегій лікувати всіх та нікого не лікувати, а також для моделі на основі формули (1). При реалізації абстрактного методу `public double f(int i, double x)`

повертається значення:

```
return m_aNetBenefit[strategy-
1][(int)Math.round(thresholdProbability*m_nPoints)];
```

де `strategy` – номер стратегії або моделі типу `int`, `thresholdProbability` – порогова ймовірність типу `double`, `m_nPoints` – кількість точок на графіку типу `int`.

Вхідні дані зберігаються у вигляді xml-файлу, що має таку dtd-структуру:

```
<!DOCTYPE medical_publication.body [
<!ELEMENT medical_publication.body (patient) *>
<!ELEMENT patient (тег_наявності_захворювання, тег_результату_обстеження) *>
<!ELEMENT тег_наявності_захворювання (#PCDATA) *>
<!ELEMENT тег_результату_обстеження (#PCDATA) *>
]>
```

Наприклад, дані про різницю систолічного та діастолічного артеріальних тисків для пацієнтів, яких буде використано для побудови кривої рішень при лікуванні серцевої недостатності, зберігаються у вигляді xml-файлу `dataPulseDifference.xml`:

```
<?xml version="1.0" encoding="UTF-8"?>
<medical_publication.body>
<patient><heart_disease>0</heart_disease> <pulse_difference>40</pulse_difference></patient>
.....
<patient><heart_disease>1</heart_disease> <pulse_difference>60</pulse_difference></patient>
</medical_publication.body>
```

Програма-парсер здійснює синтаксичний аналіз xml-файлу з даними і зберігає їх у форматі Document Object Model (DOM) в об'єкті типу `Document` за допомогою такого методу.

```
public Document constructDOM(String url) {
    Document document = null;
    try {
        DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
        DocumentBuilder builder = factory.newDocumentBuilder();
        document = builder.parse( new File(url));
    } catch (SAXException sxe) {
        // Error generated during parsing)
        Exception x = sxe;
        if (sxe.getException() != null)
            x = sxe.getException();
        x.printStackTrace(System.err);
    } catch (ParserConfigurationException pce) {
        // Parser with specified options can't be built
        pce.printStackTrace();
    } catch (IOException ioe) {
        // I/O error
```

```
ioe.printStackTrace(System.err);  
}  
return document;  
}
```

Далі структура DOM перетворюється в об'єкт типу NodeList шляхом створення вузлів з тегів patient. Для зручності використання в програмі із списку вузлів створюється хеш-таблиця:

```
for (int i=0; i<nl.getLength(); i++)  
    nodeDiagnosis = nl.item(i).getFirstChild();  
    nodeTest = nl.item(i).getLastChild();  
    m_htDCAdata.put(i, nl.item(i));  
}
```

**Приклад.** Метод апробовано на тестових даних 63 пацієнтів навчально-практичного центру первинної медико-санітарної допомоги Тернопільського державного медичного університету. Дані вводяться через xml-файл dataPulseDifference.xml. Графіки з отриманими сукупними функціями корисності наведено на рис.1:

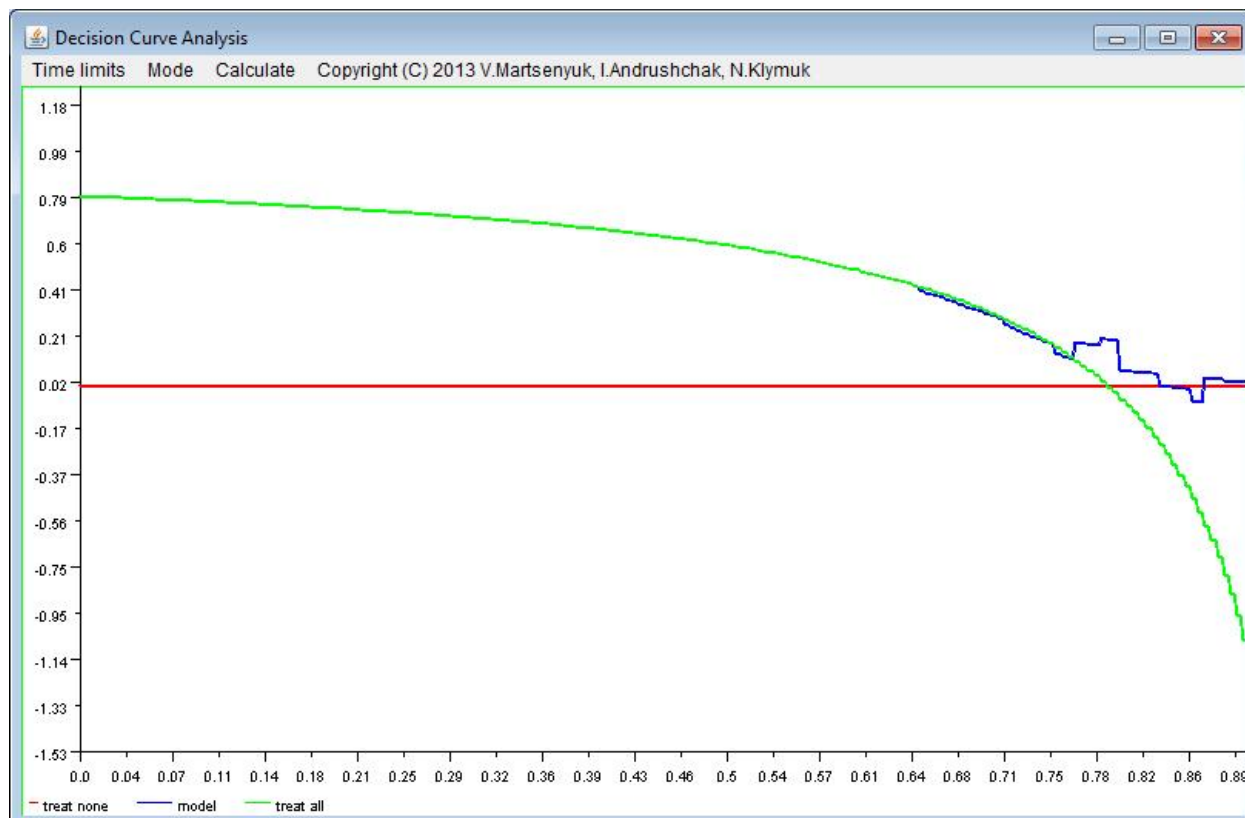
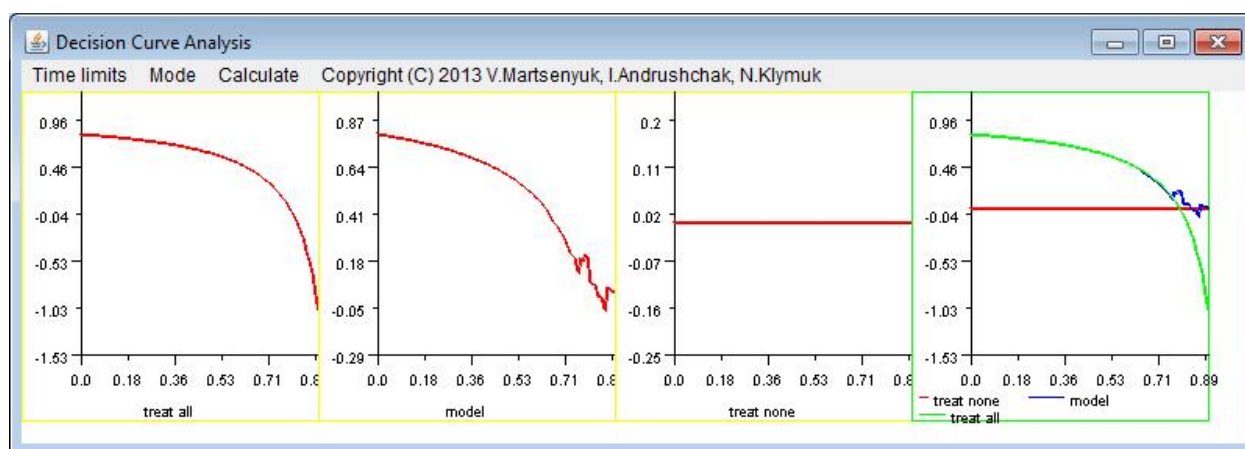


Рис.1. Сукупні функції корисності

Звідси робимо висновок, що при порогових ймовірностях для призначення лікування серцевої недостатності менших 0,75 лікування варто призначати всім. При порогових ймовірностях в межах від 0,75 до 0,84 та від 0,87 і вище лікування варто призначати лише тим пацієнтам, в яких ймовірність захворювання є вищою від порогової. При порогових ймовірностях від 0,84 до 0,87 не варто призначати лікування.

**Висновки.** В роботі представлено програмну реалізацію методу DCA з метою підтримки прийняття рішень щодо вибору оптимальної стратегії лікування. Порівняно із традиційними біостатистичними підходами такий метод має зрозумілу графічну візуалізацію та приводить до гнучкої стратегії, яка інтегрує в собі різні стратегії та моделі прийняття рішення залежно від значення порогової ймовірності.

Метод реалізовано у вигляді Netbeans-додатку з вхідними даними у вигляді xml-файлу, що робить можливим його використання в різноманітних Інтернет-проектах, пов'язаних з прийняттям рішень. Перспективним бачиться також застосування запропонованого програмного забезпечення для реалізації методу Regret DCA, що використовує regret-теорію [Tsalatsanis, 2010]

#### **Список використаних джерел.**

1. A.J. Vickers, A.M.Cronin, E.B.Elkin, M.Gonen Extensions to decision curve analysis, a novel method for evaluating diagnostic tests, prediction models and molecular markers, BMC Medical Informatics and Decision Making 2008, 8:53 doi: 10.1186/1472-6947-8-53
2. Vickers AJ, Elkin EB: Decision curve analysis: a novel method for evaluating prediction models. Med Decis Making 2006, 26(6):565-574.
3. Tsalatsanis et al.: A regret theory approach to decision curve analysis: A novel method for eliciting decision makers' preferences and decision making. BMC Medical Informatics and Decision Making 2010 10:51.
4. Марценюк В.П. Математичні моделі в системі підтримки прийняття рішень страхового забезпечення лікування онкологічних захворювань: підхід на основі динаміки Гомперца / В.П. Марценюк, І.Є. Андрущак, І.С. Гвоздецька, Н.Я. Климук // Доповіді Національної академії наук України. –2012. – №10. – С. 34-39.
5. Марценюк В.П. Підхід на основі актуарних математичних моделей до задач страхової медицини / В. П. Марценюк, І.Є. Андрущак, Н.Я. Климук // Медична інформатика та інженерія. Науково-практичний журнал. – 2010. – №4. – С. 85-87.
6. Марценюк В.П. О модели онкологического заболевания со временем пребывания на стадии в соответствии с распределением Гомперца / В.П. Марценюк, Н.Я. Климук // Проблемы управления и информатики. Международный научно-технический журнал . – 2012. – № 6. – С. 137-143.
7. Марценюк В.П., Семенець А.В. Медична інформатика. Інструментальні та експертні системи. – Тернопіль: Укрмедкнига, 2004. – 222 с.