

Разработка скоринговой карты для прогнозирования клинического исхода в оториноларингологии

А. А. Корнеев¹

¹ Санкт-Петербургский научно-исследовательский институт уха, горла, носа и речи
Минздрава России, Санкт-Петербург, 190013, Россия
(Директор – засл. врач РФ, акад. РАН, проф. Ю. К. Янов)

Development of a scorecard for predicting clinical outcome in otorhinolaryngology

А. А. Korneenkov¹

¹ Saint Petersburg Research Institute of Ear, Throat, Nose and Speech,
the Ministry of Healthcare of Russia, Saint Petersburg, 190013, Russia

Методика позволяет создавать простые и эффективные оценочные или скоринговые карты предварительной оценки риска определенного клинического исхода у пациентов с оториноларингологической патологией. Для иллюстрации применяемых методов использованы виртуальные данные, сгенерированные на основе опубликованных результатов клинических исследований факторов риска осложнений при оперативном лечении пациентов, страдающих болезнью Меньера. Генерация виртуальной выборки обеспечивает воспроизводимость решения поставленных в исследовании задач. На основе логистической регрессионной модели клинического исхода с помощью процедуры шкалирования была разработана скоринговая карта для стратификации риска развития осложнения оперативных вмешательств. Для решения статистических задач была использована программная среда R-языка, которая в настоящее время считается наиболее мощным и гибким инструментом для проведения статистического анализа медицинских данных и визуализации его результатов. Описание задач сопровождается приведением полного программного кода R-языка, который может быть использован при воспроизведении результатов этого исследования и решении подобных задач на других исходных данных. Использованные пакеты программной среды R показали хорошую эффективность для решения подобных задач, визуализации ее результатов и валидации полученной скрининговой системы.

Ключевые слова: прогнозирование клинического исхода, оценка риска, скоринговая карта, стратификация риска, программная среда R, R-язык, модель логистической регрессии, валидация скоринговой карты.

Для цитирования: Корнеев А. А. Разработка скоринговой карты для прогнозирования клинического исхода в оториноларингологии. *Российская оториноларингология*. 2019;18(2):25–35. <https://doi.org/10.18692/1810-4800-2019-2-25-35>

The method makes it possible to create simple and effective assessment cards or scorecards of a preliminary assessment of risks of a certain clinical outcome in patients with otorhinolaryngological pathology. To illustrate the methods, we used virtual data generated on the basis of the published results of clinical studies of complications risk factors in the surgical treatment of patients with Meniere's disease. The virtual fetching generation provides reproducibility of the solution of the problems set in the study. Based on a clinical outcome logistic regression model, using a scaling procedure, we developed a scorecard to stratify the risk of complications of surgical interventions. To solve statistical problems, we used the R-language software environment, which is currently considered the most powerful and flexible tool for statistical analysis of medical data and visualization of its results. The description of the tasks is accompanied by the introduction of R-language full program code which can be used to reproduce the results of this study and solve similar problems with other source data. The used R software packages have presented high efficacy in the solution of such problems, visualization of its results and validation of the obtained screening system.

Keywords: clinical outcome prediction, risk assessment, scorecard, risk stratification, R software environment, R-language, logistic regression model, scorecard validation.

For citation: Korneenkov A. A. Development of a scorecard for predicting clinical outcome in otorhinolaryngology. *Rossiiskaya otorinolaringologiya*. 2019;18(2):25–35. <https://doi.org/10.18692/1810-4800-2019-2-25-35>

Введение

Скоринг (от англ. *scoring* – подсчет очков в игре) в медицине – это модель классификации пациентов на группы по диагнозу или риску клинического исхода на основе скоринговых баллов. По каждому отобранному в модель и имеющемуся у пациента симптому присваиваются очки, суммируются в общий скоринговый балл, который сравнивается с заданным пороговым значением. Если полученный итоговый скоринговый балл больше установленного порогового значения, то пациент классифицируется в одну группу, если меньше – в другую.

В клинической практике хорошо известны простые и достаточно эффективные в использовании оценочные или скоринговые карты предварительной оценки риска неблагоприятного исхода или определенной болезни у обследуемых пациентов [5]: QRISK2, SAPS (simplified acute physiology score), APS (acute physiological score), LODS (Logistic Organ Dysfunction system), SOFA (Sequential Organ Failure Assessment), OASIS (Oxford Acute Severity of Illness Score) и qSOFA и др. Однако, несмотря на визуальную простоту скоринговой карты, ее разработка представляет собой сложный многоэтапный процесс, который в отечественной медицинской литературе не описан. В настоящем исследовании для создания скоринговой карты использован R – язык программирования для статистической обработки данных и работы с графикой, свободно доступный по адресу <https://cran.r-project.org>. Синтаксис R для общего процесса управления данными и статистического анализа подробно документирован, что играет решающую роль во время пересмотра статьи и перекрестной проверки результатов исследования.

Цель исследования

Разработка пошагового руководства для создания скоринговой карты для оценки риска определенного клинического исхода в оториноларингологии средствами программной среды R.

Материалы и методы исследования

Для иллюстрации применяемых методов использованы виртуальные данные, сгенерированные на основе опубликованных результатов клинических исследований факторов риска осложнений при оперативном лечении пациентов, страдающих болезнью Меньера [1]. Генерация виртуальной выборки обеспечивает воспроизводимость данных и решения всех поставленных задач.

Результаты и обсуждение

Генерация исходных данных предусматривает использование программного кода R-языка

(табл. 1), который создает таблицу с данными о пациентах, содержащую пять переменных, включая бинарный (дихотомичный) результат операции – «Исход» (*outcome*), две числовые переменные («Возраст», «СКУ») и две категориальные переменные («Рвота», «Спонтанный нистагм»). Переменная «Возраст» (*age*) отражает возраст пациента в годах и переменная «СКУ» (*sku*) (сводный коэффициент устойчивости – СКУ) представляет собой суммарный показатель отклонения центра тяжести в ответ на различные динамические стимулы, дающий комплексное представление о состоянии статического и динамического равновесия человека (%). Переменная «Рвота» (*vomitus*) имеет два уровня: «по», «yes» (т. е. отсутствие признака и его наличие). Переменная «Спонтанный нистагм» (*nystagmus*) также имеет два уровня: «по», «yes». Значение переменной «Исход» равное 1 соответствует наличию осложнения операции, 0 – его отсутствию.

Описание программы следующее. Строка 1 вызывает функцию «library» для загрузки пакета «dummies» в программу R. В строке 2 с помощью функции «set.seed» устанавливается произвольное значение переменной генератора случайных чисел, чтобы сделать результаты полностью воспроизводимыми. Строка 3 устанавливает размер генерируемой выборки (150 наблюдений). В строке 4 присваивается значение свободному члену регрессионного уравнения (*b0*). Строки 5 и 6 генерируют количественные переменные: «Возраст» и «СКУ», которые распределены по нормальному закону распределения со средним значением 46,3 года и 65,6%, а также стандартным отклонением 10 лет и 2,7% соответственно. В строке 7 создается факторная переменная – «Спонтанный нистагм» с двумя уровнями «по», «yes», количеством наблюдений *n* (строка 8), с заменой имеющихся значений вставляемыми (строка 9) и соответствующими пропорциями каждого из двух уровней (строка 10). В строках 11–14 таким же образом создается факторная переменная «Рвота». В строке 15 уровни этой переменной переопределяются, базовым уровнем назначался «yes». Строка 16 создает линейный предиктор «lp» уравнения логистической регрессии. Функция «dummy» используется для преобразования факторной переменной в фиктивные или «дамми» (*dummy*) переменные. Символом «%*%» обозначается матричный оператор умножения. Для каждой переменной задается линейный предиктор. Например, коэффициент 2,5, соответствующий уровню «yes» (строка 16) переменной «Спонтанный нистагм», интерпретируется как увеличение линейного предиктора на 2,5 единицы для значения «yes», по сравнению с «по». Строка 17 преобразует линейный предиктор в вероятность интересующего исхода посредством

Таблица 1

Программный код создания виртуальной выборки

Table 1

R codes for data simulation

Строка	Программный код R-языка
1	library(dummies)
2	set.seed(30)
3	n <- 150
4	b0 <- 22
5	age <- round(rnorm(n, mean = 46.3, sd = 10))
6	sku <- round(rnorm(n, mean = 65.6, sd = 2.7), 1)
7	NYSTAGMUS <- as.factor(sample(x = c("no", "yes"),
8	size = n,
9	replace = TRUE,
10	prob = c(0.42, 0.58)))
11	VOMITUS <- as.factor(sample(x = c("no", "yes"),
12	size = n,
13	replace = TRUE,
14	prob = c(0.47, 0.53)))
15	VOMITUS <- relevel(VOMITUS, ref = "yes")
16	lp <- b0 + 0.12*age - 0.41*sku + cbind(1, dummy(NYSTAGMUS)[, -1]) %*% c(0.001,
17	2.5) - cbind(1, dummy(VOMITUS)[, -1]) %*% c(0, 2.2) + 0.001*age^2 - 0.001*sku^2
18	pi.x <- exp(lp)/(1 + exp(lp))
19	OUTCOME <- rbinom(n = n, size = 1, prob = pi.x)
20	df <- data.frame(OUTCOME, age, sku, NYSTAGMUS, VOMITUS)
21	df\$dataset <- sample(x = c("train", "validate"),
22	size = n,
23	replace = TRUE,
24	prob = c(0.75, 0.25))
25	f.y <- glm(OUTCOME~age + sku + NYSTAGMUS + VOMITUS, data = df, family = "binomial")
26	summary(f.y)

логит-преобразования, а строка 18 генерирует переменную исхода, которая распределена по биномиальному распределению. Строка 19 объединяет все переменные в фрейм данных (data frame). Весь набор данных с помощью функции «sample» разделяется на подмножества данных для обучения и для валидации (строки 20–23). Три четверти всего набора сгенерированных данных используется для обучения модели (строка 23), а оставшаяся четверть используется для ее валидации. С помощью кода в строке 24 формируется обобщенная линейная модель «glm» под именем «f.m», по которой для общего контроля процесса с помощью функции «summary» (строка 25) выводятся коэффициенты и показатели созданной модели.

В основе скоринговой карты чаще всего лежат модель логистической регрессии и ее коэффициенты. Логистическая регрессия формирует коэффициенты регрессии (и их стандартные ошибки и уровни значимости) по формуле логит-преобразования вероятности интересующего значения переменной исхода:

$$\text{logit}(p) = b_0 + b_1 X_1 + b_2 X_2 + b_3 X_3 + \dots + b_k X_k,$$

где p – вероятность наличия признака, представляющего интерес; b – коэффициенты модели; X – значения предикторов. Логит-преобразование определяется так же, как логарифм шансов:

$$\text{logit}(p) = \ln \left(\frac{p}{1-p} \right).$$

Модель может использоваться для расчета вероятности интересующего исхода (p) для заданных значений предсказывающих переменных – предикторов.

$$p = \frac{e^{\text{logit}(p)}}{1 + e^{\text{logit}(p)}}.$$

Далее определенным значениям вероятности интересующего исхода назначается также определенная скоринговая оценка (балл). Кроме того, определяется пороговое значение, по которому те пациенты, у кого скоринговая оценка ниже порогового значения, в нашей задаче классифицируются как пациенты с осложнением, а пациенты

с оценкой выше порога классифицируются как пациенты без осложнения оперативного вмешательства.

Создавая скоринговую карту, необходимо решить две специфические задачи: преобразования количественных предикторов в категориальные и масштабирования скоринговой карты. Масштабирование или шкалирование не влияет на прогностическую силу скоринговой карты [4], это косметическая процедура, выполняемая для улучшения удобства, понимания и интерпретации скоринговой карты. Масштабирование может быть реализовано с использованием различных подходов, которые невозможно привести в рамках одной статьи. Мы коснемся масштабирования только одного коэффициента скоринговой карты – базового количества баллов (basepoints), к которому добавляются скоринговые баллы, соответствующие значениям предикторов. Для расчета этого коэффициента используются рассчитанный по логистической модели свободный член b_0 и заранее заданные константы: 1) количество баллов, соответствующее удвоению шансов интересующего исхода, – «pdo» («points to double the odds») (обычно, pdo = 50); 2) «points0» – значение баллов (обычно 600), при котором наблюдаются определенные целевые шансы исхода «odds_0» (например, шанс 1/19 означает, что на один интересующий нас исход приходится 19 альтернативных исходов). Кроме того, ниже приведено уравнение для расчета базового количества баллов:

$$\text{basepoints} = \text{points}_0 - \text{pdo} \times \frac{(b_0 - \ln(\text{odds}_0))}{\ln(2)}.$$

Например, при $b_0 = 0,56$, $\text{points}_0 = 600$, $\text{pdo} = 50/\ln(2) = 72,1$, $\ln(\text{odds}_0) = \ln(1/19) = -2,9$

базовое количество баллов составит 427,7 (или округленно до целого – 428 баллов).

Именно это значение используется в скоринговой карте, созданной с помощью нижеприведенного программного кода (табл. 2).

В представленном выше программном коде используется несколько функций, специально созданных в пакете «scorecard» [3]. Функция «scorecard» создает скоринговую карту на основе результатов выполнения функций «woebin» и «glm». Функция «split_df» случайным образом разделяет набор данных на основе переменной исхода с соотношением 0,6 : 0,4 (или, что то же самое, в процентах – 60 : 40). Аргумент «seed» обеспечивает воспроизводимость результатов разделений при следующем запуске функции. Если в качестве «seed» при запуске кода будет задано то же число (в нашем случае 30), результаты деления будут идентичны.

Функция «woebin» генерирует оптимальное разделение количественных переменных на уровни, т. е. производит категоризацию количественной переменной (используется также термин «биннинг», от англ. binning). Функция «woebin_ply» конвертирует исходные входные данные в значение «woe» на основе категорий «bin», генерируемых «woebin». Функция «glm» используется для создания логрегрессионной модели с указанием формулы линейного предиктора (OUTCOME~.), семейства моделей (binomial) и источника данных для этой модели (train_woe). Функция «step» отбирает наиболее подходящую модель на основе информационного критерия Акаике (AIC, от англ. an information criterion). В строке 12 функция «scorecard» создает скоринговую карту, основанную на результатах вы-

Таблица 2

Программный код создания скоринговой карты

Table 2

R codes for creating scorecards

Строка	Программный код R-языка
1	library(scorecard)
2	df.sc <- subset(df, select = c(OUTCOME, age, sku, NYSTAGMUS, VOMITUS))
3	mod <- glm(OUTCOME ~., df.sc, family = "binomial")
4	dt_list = split_df(df.sc, y = "OUTCOME", ratio = 0.6, seed = 30)
5	train = dt_list\$train; test = dt_list\$test;
6	bins = woebin(df.sc, y = "OUTCOME")
7	train_woe = woebin_ply(train, bins)
8	test_woe = woebin_ply(test, bins)
9	m1 = glm(OUTCOME ~., family = binomial(), data = train_woe)
10	m_step = step(m1, direction="both", trace = FALSE)
11	m2 = eval(m_step\$call)
12	card = scorecard(bins, m2)

Программный код для печати коэффициентов скоринговой карты

Таблица 3

Table 3

R codes for printing scorecards

Строка	Программный код R-языка
1	<code>printList <- function(list) {</code>
2	<code> for (item in 1:length(list)) {</code>
3	<code> print(head(list[[item]]))</code>
4	<code> }</code>
5	<code>}</code>
6	<code>printList(card)</code>

полнения функций «woebin» и «glm». Затем на основе полученной скоринговой карты функция «scorecard_ply» вычисляет скоринговый балл для каждого пациента в выборке.

С помощью приведенной ниже функции (табл. 3) получены коэффициенты для скоринговой таблицы.

На рис. 1 представлен возможный вид скоринговой карты с примером для расчета баллов и классификации пациента по риску осложнения операции (пациент 46 лет, SKU – 60%, рвота и спонтанный нистагм отсутствуют, результат – риск осложнения операции высокий).

Для того чтобы представить связь количества скоринговых баллов с вероятностью интересующего нас исхода в виде визуально удобной диа-

граммы, был выполнен программный R-код по подготовке данных (табл. 4). Функция «scorecard_ply» рассчитывает скоринговые баллы по скоринговой карте «card», полученной ранее функцией «scorecard» (строка 1) и вставляет их в таблицу «score_tbl». Аргумент «only_total_score» отвечает за представление только итоговых баллов из расчета. Если этот аргумент установлен как TRUE, то вывод включает только общую сумму баллов. Если FALSE, вывод включает как итоговую сумму баллов, так и рассчитанные баллы для каждой переменной. В строке 2 функцией «cbind» объединяются данные из таблиц «score_tbl» и «df.sc». В строках 3–6 в таблицу вводится новая переменная «dataset» со значениями «train» и «validate», распределенными случайным обра-

ОЦЕНОЧНАЯ КАРТА РИСКА ОСЛОЖНЕНИЙ ПРИ ОПЕРАЦИОННОМ ВМЕШАТЕЛЬСТВЕ				
БАЗОВЫЙ БАЛЛ - 428				
Признак	Атрибут	Балл	Значение атрибута для пациента	Баллы для пациента
Возраст, лет	<41	261		
	41-54	-6	46 лет	-6
	54-58	-125		
	>58	-368		
СКУ, %	<61,8	-291	60%	-291
	61,8 – 65,4	-27		
	65,4 - 66	-187		
	>66	163		
Рвота	есть	-59		
	нет	69	нет	69
Спонтанный нистагм	нет	150	нет	150
	есть	-99		
Общий балл			428-78=350	

Результат: 350<600 – Риск осложнения операции высокий

Рис. 1. Форма скоринговой карты.
Fig. 1. Scorecard.

зом с вероятностью 0,75 и 0,25 соответственно. Таким образом, после выполнения этой процедуры имеется набор данных с исходными переменными, рассчитанными скоринговыми баллами и, переменной «dataset», которая случайным образом разделяет этот набор данных на две выборки: «train» и «validate», используемые в последующих расчетах для валидации скоринговой карты.

Модель логистической регрессии подгоняется функцией «glm», с аргументом «binomial», определяющим используемое семейство биномиальных моделей (строки 7–9). Объект «newx» (строка 10) представляет собой вектор значений скоринговых баллов от минимального до максимального с возрастающим шагом, равным 1. Функция «pred» используется для оценки вероятности интересующего исхода при каждом значении скорингового балла (строка 12). Тип предсказания, по умолчанию, находится в шкале или масштабе линейного предиктора. Для вывода стандартной ошибки (SE) аргументу «se.ft» присваивается TRUE (сокращенно «Т») (строка 15).

Объект «count» – это матрица, содержащая количество хороших и плохих исходов для каждого диапазона значений итоговых скоринговых баллов (строки 10–14). Эти диапазоны были получены разбивкой диапазона значений скоринговой оценки с помощью функции «seq», по 50 скорин-

говых баллов в каждом диапазоне (by = 50) (строка 19).

Для визуализации ассоциации скоринговых баллов и вероятности исхода с помощью кода R-языка (табл. 5) была построена столбиковая диаграмма (barplot), на которой отображалось число хороших (good) и плохих (bad) исходов, стратифицированных по количеству скоринговых баллов. На этой же диаграмме в виде линейного графика показана вероятность интересующего исхода.

Функция «par» используется для установки графических параметров, в частности полей (или отступов) с четырех сторон графика в виде числа строк (чем больше строк, тем больше ширина поля) (строка 1). Первый аргумент «barplot» – это матрица значений для столбиковой диаграммы (строка 2). Значения в столбцах матрицы соответствуют высоте столбиков диаграммы. Аргумент «col» задает вектор цветов для столбцов или их составных частей (строка 7). Функция «legend» – добавляет легенду к диаграмме (строки 8–13). Легенда добавляется в левый верхний угол, а меткам исходов («Non-complication» и «Complication») присваиваются желтый и светло-голубой цвета соответственно. Аргументы функции «legend» дают описания столбиков и линий графика, представленных на диаграмме. Параметр «border» определяет цвета границ прямоугольников с цветовым

Программный код подготовки данных для визуализации

Таблица 4

Table 4

Data preparation for visualization

Строка	Программный код R-языка
1	score_tbl = scorecard_ply(df.sc, card, only_total_score = F)
2	df.f <- cbind(score_tbl, df.sc)
3	df.f\$dataset <- sample(x = c("train", "validate"),
4	size = nrow(df.f),
5	replace = TRUE,
6	prob = c(0.75, 0.25))
7	glmod <- glm(formula = OUTCOME ~ score,
8	df.f[df.f\$dataset == "train",],
9	family = "binomial")
10	newx <- seq(min(df.f[df.f\$dataset == "train",]\$score),
11	max(df.f[df.f\$dataset == "train",]\$score))
12	prd <- predict(glmod,
13	newdata = data.frame(score = newx),
14	type = "response",
15	se.fit = T)
16	count <- as.matrix(table(cut(df.f[df.f\$dataset == "train",]\$score,
17	breaks = seq(min(df.f[df.f\$dataset == "train",]\$score),
18	max(df.f[df.f\$dataset == "train",]\$score),
19	by = 50),
20	include.lowest = T),
21	df.f[df.f\$dataset == "train",]\$OUTCOME))

наполнением столбиков (строка 13). Аргументу «new» в функции «par» присваивается TRUE, чтобы результаты выполнения последующих команд не приводили к очищению предыдущих результатов на диаграмме (строка 14). Это важно, если необходимо построить два графика на одной и той же диаграмме. Значения предсказанной вероятности построены по отношению к значениям скоринговых баллов (строки 15–18). Оси x и y на диаграмме не выводятся (строка 17), и названия обеих осей не указываются (строка 18). Функция «polygon» рисует доверительный интервал предсказанной вероятности исхода (строка 19). Нижняя и верхняя границы доверительного интервала вероятности исхода обозначаются штриховыми линиями (строки 24–27). Функция «axis» добавляет к текущему графику с правой стороны ось, которая соответствует шкале предсказанной вероятности (строка 28). Функция «mtext» добавляет текст к текущему графику (строка 29).

Аргумент «line = 3» указывает, что текст добавляется к третьей строке этого поля. Счет маргинальных строк (т. е. по краям, на границах графика) начинается с 0, снаружи (строка 30).

Результаты показаны на рис. 2.

Оценка валидности разработанной модели является обязательным этапом любого моделирования. В пакет scorecard встроены некоторые из этих методов: статистика Колмогорова-Смирнова (Kolmogorov-Smirnov statistic (KS), AUC (area under curve), PSI (population stability index). Ниже представлены программные коды для проверки модели встроенными в этот пакет функциями (табл. 6)

В строках 1–2 выполняется расчет прогностических значений моделей с помощью функции «predict». Функция «perf_eva» вычисляет статистику Колмогорова-Смирнова (Kolmogorov-Smirnow (KS), AUC и другие показатели. Функция «perf_psi» вычисляет индекс стабильности популяции

Программный код для построения диаграммы

Таблица 5

Table 5

R codes for creating graphs

Строка	Программный код R-языка
1	par(mar = c(5,4,4,5) + .1)
2	barplot(t(count),
3	main = "Scores versus probability of complication",
4	xlab = "Scores",
5	ylab = "Observed number of patients",
6	space = 0,
7	col = c("yellow", "lightblue"))
8	legend("topleft", fill = c("yellow", "lightblue", NA),
9	lty = c(NA, NA, 1), lwd = c(NA, NA, 2),
10	legend = c("Non-complication", "Complication",
11	"Predicted Prob."),
12	col = c("black"),
13	border = c("black", "black", NA))
14	par(new = TRUE)
15	plot(prd\$fit~newx,
16	type = "l", col = "black",
17	lwd = 2, xaxt = "n", yaxt = "n",
18	xlab = "", ylab = "")
19	polygon(c(rev(newx), newx),
20	c(rev(prd\$fit + 1.96*prd\$se.fit),
21	prd\$fit - 1.96*prd\$se.fit),
22	col = adjustcolor('grey80', alpha = 0.5),
23	border = NA)
24	lines(newx, prd\$fit + 1.96*prd\$se.fit,
25	lty = 'dashed', col = 'red')
26	lines(newx, prd\$fit - 1.96*prd\$se.fit,
27	lty = 'dashed', col = 'red')
28	axis(4)
29	mtext("Predicted probability of complication",
30	side = 4, line = 3)

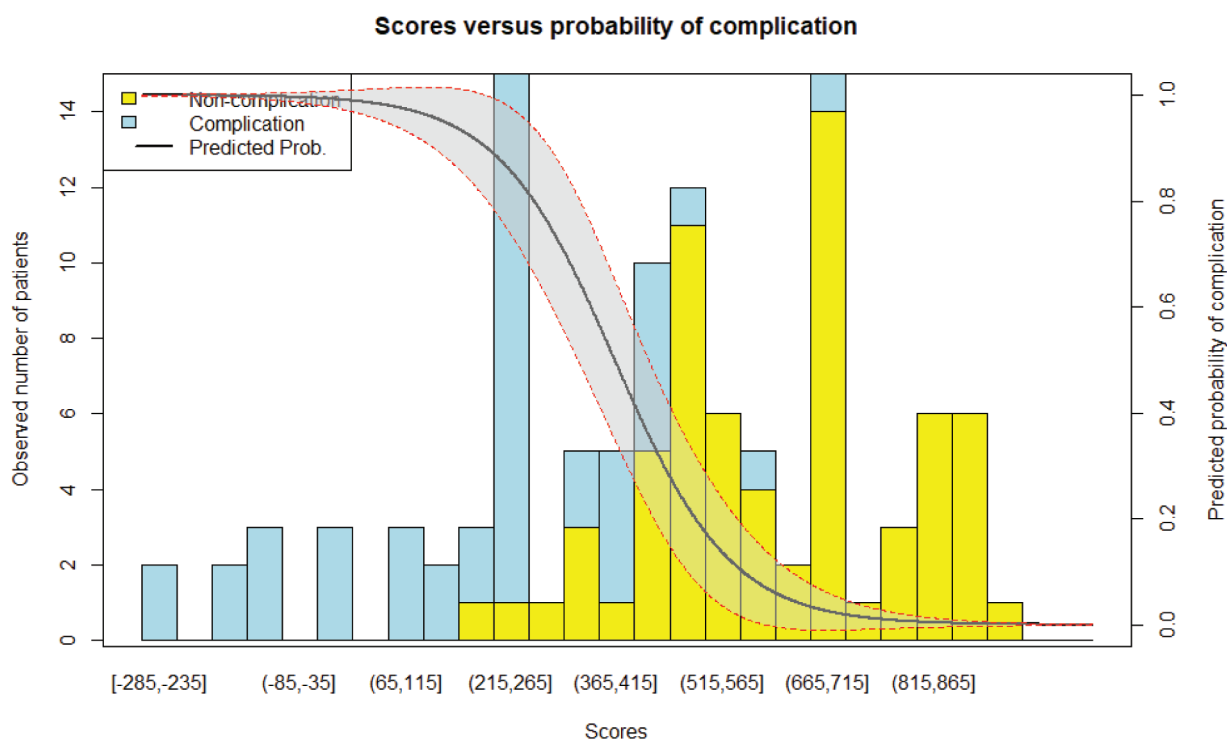


Рис. 2. Соотношение скоринговых баллов и вероятности интересующего исхода (ось x – интервалы значений скоринговых баллов, ось y слева – количество пациентов, ось y справа – предсказанная вероятность исхода).

Fig. 2. Predicted probability of complication versus the number of observed complication and non-complication.

Таблица 6

Программный код для расчетов KS и ROC

Table 6

R codes for calculating ROC and KS statistics

Строка	Программный код R-языка
1	<code>train_pred = predict(m2, train_woe, type = 'response')</code>
2	<code>test_pred = predict(m2, test_woe, type = 'response')</code>
3	<code>train_perf = perf_eva(train\$OUTCOME, train_pred, title = "train")</code>
4	<code>test_perf = perf_eva(test\$OUTCOME, test_pred, title = "test")</code>
5	<code>train_score = scorecard_ply(train, card, print_step = 0)</code>
6	<code>test_score = scorecard_ply(test, card, print_step = 0)</code>
7	<code>perf_psi(score = list(train = train_score, test = test_score),</code>
8	<code>label = list(train = train\$OUTCOME, test = test\$OUTCOME))</code>

PSI (population stability index). На рис. 3 представлены диаграммы функций распределения «хороших» (good) и «плохих» (bad) исходов, на основе статистики Колмогорова–Смирнова (KS) и ROC-анализа с расчетом AUC (area under curve) для обучающей (train) и тестовой (test) выборки. По оси ординат откладывается кумулятивная пропорция для «плохих» и «хороших» исходов, по оси абсцисс – пропорция от популяции. На диаграмме представлено несколько линий: 1) кумулятивная пропорция «хороших» исходов; 2) кумулятивная пропорция «плохих» исходов; 3) KS-статистика. KS-статистика – это максимальная разница между кумулятивными пропорциями для «плохих» и «хороших» исходов. Чем дальше друг от друга

находятся две линии, тем больше степень дифференциации между накопленным (кумулятивным) процентом «плохих» и «хороших» исходов и, следовательно, лучше (точнее) модель.

Например, можно увидеть, что по обучающей выборке KS-статистика составляет около 0,74, а для тестовой – 0,88. Один из показателей, характеризующих дискриминационную способность модели, AUC (area under curve) для обучающей и тестовой выборки составляет значительные 0,94 и 0,95 соответственно.

Для более серьезной оценки валидности модели недостаточно методов, встроенных в пакет «scorecard». Для этой цели можно рекомендовать пакет «rms» [2], который предлагает множество

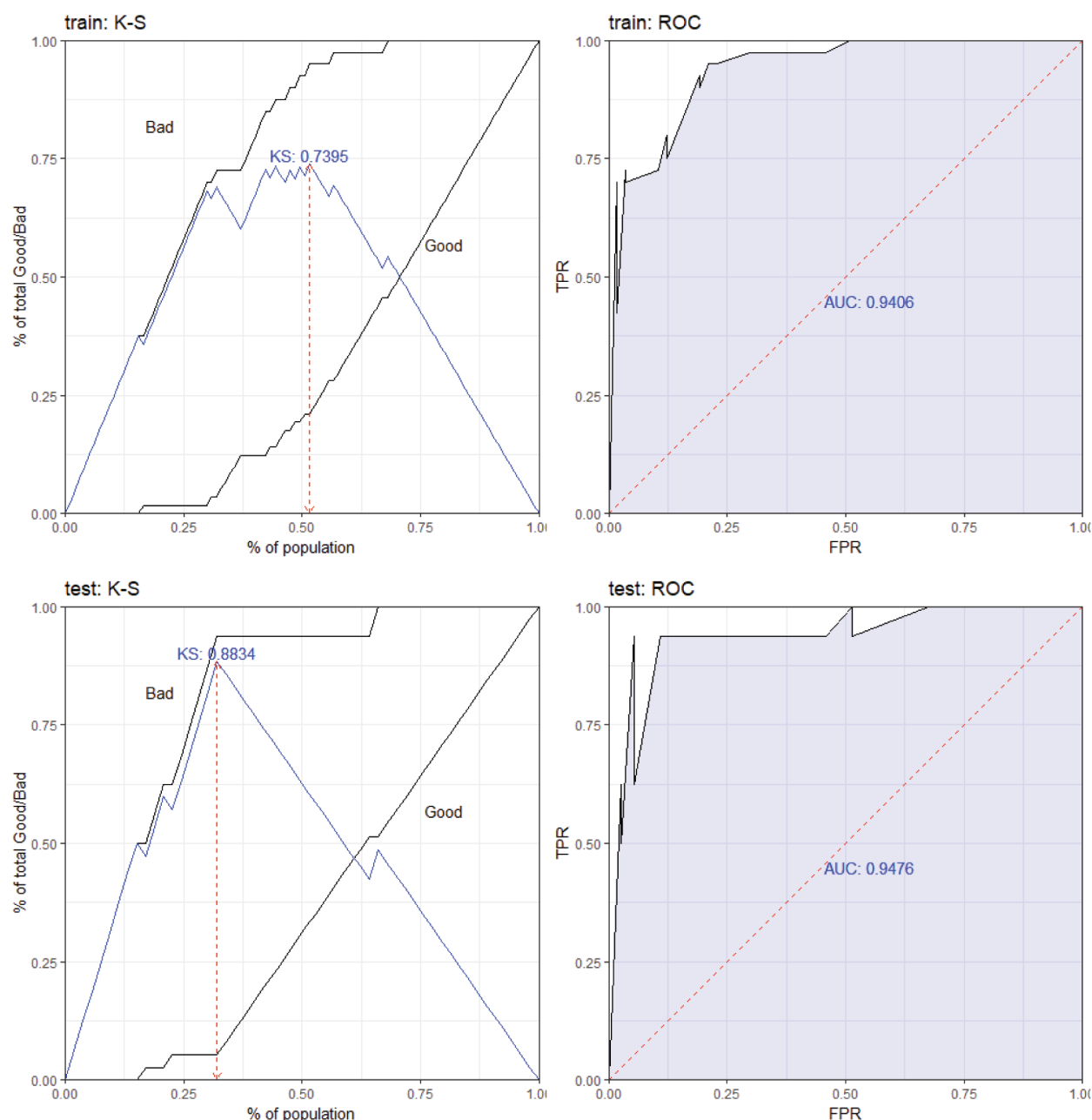


Рис. 3. Оценка функций распределения хороших и плохих исходов на основе статистики Колмогорова–Смирнова (KS) и ROC анализа с расчетом AUC (area under curve) для обучающей (train) и тестовой (test) выборки.

Fig. 3. Evaluation of the distribution of good and bad outcomes, based on Kolmogorov–Smirnov statistics (KS) and ROC analysis with the calculation of AUC (area under curve) for training (train) and test (test) samples.

инструментов для создания и оценки регрессионных моделей в R. Он сопровождает книгу Фрэнка Харрелла, которая является великолепным изданием для всех, кто работает в области «data science».

Валидация модели включает оценку дискриминационной способности модели и калибровку. Для получения этих показателей исходные данные специальным образом подготавливаются с помощью метода бутстрапа (bootstrap methods), который в настоящее время является методом выбора для решения подобных задач.

Применительно к нашей задаче в начале моделирования исходный набор данных (original

sample) был случайным образом в соотношении 75 : 25 разделен на две выборки: обучающую (модель) выборку (training sample) и тестовую выборку (test sample). Считается, что модель по обучающей выборке имеет лучшие значения показателей производительности, поэтому она обладает определенным оптимизмом по сравнению с моделью, полученной на тестовой выборке. Разница показателей производительности для обучающей и тестовых выборок называется оптимизмом (optimism) модели, и для получения откорректированного показателя (corrected index) значение оптимизма вычитается из показателя,

Таблица 7

Программный код для валидации скоринговой модели

Table 7

R codes for scoring model validation

Строка	Программный код R-языка
1	library(scorecard)
2	glmod <- lrm(formula = OUTCOME ~ score,
3	data = df.f[df.f\$dataset == "train",])
4	cal <- calibrate(glmod, B = 200)
5	plot(cal)
6	options(prType = 'latex')
7	v <- validate(glmod, B = 200)
8	print(v, digits = 3)

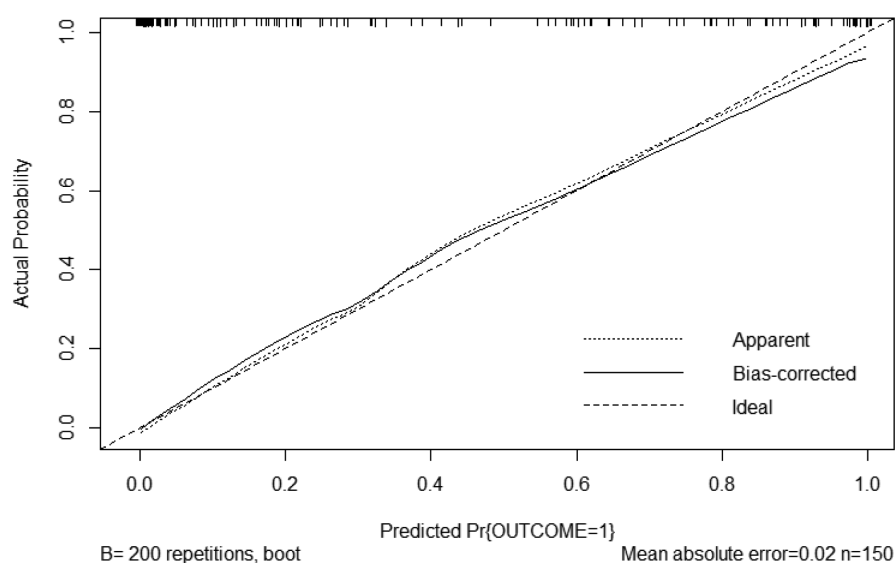


Рис. 4. Калибровочная кривая (КК) для модели исхода
(Apparent – КК по оригинальному набору данных, Bias-corrected – скорректированная на оптимизм КК, Ideal – идеальная КК, линия под углом 45°).

Fig. 4. Calibration Curve for model.

Таблица 8

Статистика валидации модели

Table 8

Validation scorecard

Показатель	Для исходного набора данных	Для обучающей выборки	Для пробной выборки	Значение «оптимизма»	Скорректированный показатель	n
Dxy	0,878	0,890	0,866	0,024	0,855	200
R ²	0,697	0,719	0,680	0,039	0,658	200
Intercept	0,000	0,000	-0,003	0,003	-0,003	200
Slope	1,000	1,000	0,867	0,133	0,867	200
E _{max}	0,000 0	0,000	0,032	0,032	0,032	200
D	0,709	0,743	0,683	0,059	0,650	200
U	-0,013	-0,013	0,009	-0,023	0,009	200
Q	0,723	0,756	0,674	0,082	0,641	200
B	0,098	0,090	0,104	-0,014	0,112	200
g	4,045	4,539	3,824	0,715	3,330	200
gp	0,413	0,416	0,408	0,008	0,405	200

полученного на оригинальной выборке. В табл. 7 приведен программный код для валидации скоринговой модели.

Результаты показаны на рис. 4.

Прогнозируемая вероятность исхода (Predicted Pr{OUTCOME=1}) строится на фоне наблюдаемой вероятности (Actual Probability), а отклонение от идеальной линии указывает на разницу между прогнозируемыми и наблюдаемыми рисками.

Близость калибровочной кривой к диагональной линии 45° демонстрирует приемлемую валидацию по шкале абсолютной вероятности. В табл. 8 представлены показатели дискриминационной способности полученной модели.

Например, Dху – ранговая корреляция Сомерса (Somers' D) между прогнозируемой вероятностью осложнения (OUTCOME = 1) и наблюдаемым исходом; D – показатель дискриминации – отношение правдоподобия модели χ^2 , деленное на размер выборки; R^2 – псевдоR-квадрат (R^2 Nagelkerke-Cox-Snell-Maddala-Magee), коэффициент детерминации модели. Более подробную информацию о пред-

ставленных валидационных показателях можно почерпнуть из описания пакета rms (<https://cran.r-project.org/web/packages/scorecard/scorecard.pdf>). Основные показатели демонстрируют приемлемую оценку валидации модели.

Заключение

Приведенная методика создания и оценки производительности скоринговой карты требует определенных статистических знаний и навыков в использовании R-языка. Однако, как это часто бывает, разработка простых в использовании инструментов требует большой подготовительной работы и сложных методик. Внедрение клинических скоринговых карт дает возможность переложить сложную часть математического моделирования на статистиков и математиков, оставив для конечного пользователя только то, что он и должен делать – удобным для него образом собирать, анализировать, оценивать клиническую информацию и принимать врачебные решения.

Автор заявляет об отсутствии конфликта интересов.

ЛИТЕРАТУРА/REFERENCES

1. Корнеев А. А., Лиленко С. В., Лиленко А. С., Вяземская Е. Э., Бахилин В. М. Использование модифицированной процедуры последовательного распознавания Вальда для определения исхода оперативного лечения у пациентов с болезнью Меньера. *Российская оториноларингология*. 2018;3(94):54–59. doi: 10.18692/1810-4800-2018-3-54-59 [Korneev A. A., Lilenko S. V., Lilenko A. S., Vyazemskaya E. E., Bakhilin V. M. Use of the modified procedure of Wald's sequential probability ratio test for determining the outcome of a surgical treatment in patients with Meniere's disease. *Rossiiskaya otorinolaringologiya*. 2018;3(94):54–59]. (In Russ.) doi: 10.18692/1810-4800-2018-3-54-59
2. Package 'rms' January 7, 2018. Version 5.1–2. Date 2018-01-06 Title Regression Modeling Strategies Author Frank E Harrell Jr. – Режим доступа: <https://cran.r-project.org/web/packages/rms/rms.pdf>. – Загл. с экрана.
3. Package 'scorecard' September 11, 2018 Version 0.1.9 Title Credit Risk Scorecard. – Режим доступа: <https://cran.r-project.org/web/packages/scorecard/scorecard.pdf> – Загл. с экрана.
4. Siddiqi N. Credit Risk Scorecards: Developing and Implementing Intelligent Credit Scoring. John Wiley & Sons, 2005. 208 p.
5. Zhang Z., Zhang H., Khanal M. K. Development of scoring system for risk stratification in clinical medicine: a step-by-step tutorial. *Annals of Translational Medicine*. 2017;5;21:436. (doi: 10.21037/atm.2017.08.22)

Информация об авторе

✉ Корнеев Алексей Александрович – доктор медицинских наук, профессор, заведующий лабораторией информатики и статистики, Санкт-Петербургский научно-исследовательский институт уха, горла, носа и речи Минздрава России (190013, Россия, Санкт-Петербург, Бронницкая ул., д. 9); тел. +7(904)554-07-40, e-mail: korneyenkov@gmail.com
ORCID: <http://orcid.org/0000-0001-5870-8042>

Information about the author

✉ Alexei A. Korneev – MD, Professor, Head of the Laboratory of Informatics and Statistics, Saint Petersburg Research Institute of Ear, Throat, Nose and Speech Ministry of Healthcare of Russia (190013, Russia, St. Petersburg, Bronnitskaya ul., 9); tel. +7 (904) 554-07-40, e-mail: korneyenkov@gmail.com
ORCID: <http://orcid.org/0000-0001-5870-8042>