DOI: https://doi.org/10.54422/1994-439X.2021.2-50.61-70

УДК 614.841.31:311.4::004.942::004.62

д-р техн. наук, проф. Татур М.М., Проровский В.М.
канд. техн. наук, доц. Иваницкий А.Г.*, Ходин М.В.**

Сравнение точности алгоритмов автоматического машинного обучения при прогнозировании обстановки с пожарами на объектах жилого сектора

Учреждение образования «Белорусский государственный университет информатики и радиоэлектроники», г. Минск

**Государственное учреждение образования «Университет гражданской защиты Министерства по чрезвычайным ситуациям Республики Беларусь», г. Минск*

***Учреждение «Научно-исследовательский институт пожарной безопасности и проблем чрезвычайных ситуаций» МЧС Республики Беларусь, г. Минск*

Целью работы является исследование применения предсказательных моделей и мета-моделей, в том числе с использованием технологий машинного обучения, для прогнозирования обстановки с пожарами. Отдельно рассматривается сравнительный анализ их точности при прогнозировании обстановки с пожарами на объектах жилого сектора на долгосрочный период.

Ключевые слова: пожар, интеллектуальный анализ, системный анализ, временные ряды, машинное обучение, статистика пожаров, прогнозирование

Grand Ph.D. (Tech.), Prof. M.M. Tatur, V.M. Prorovsky,
Ph.D. (Tech.), Assoc. Prof. A.G. Ivanitski*, M.V. Hodin**

Comparison of the accuracy of automatic machine learning algorithms in forecasting the situation with fire in the residential sector

The Educational Establishment «Belarusian State University of Informatics and Radioelectronics», Minsk

**State Educational Establishment «University of Civil Protection of the Ministry for Emergency Situations of the Republic of Belarus», Minsk*

***The institution "Scientific and Research Institute of Fire Safety and Emergency Situations" of the Ministry for Emergency Situations of the Republic of Belarus, Minsk*

The aim of the work is to study the using of predictive models and meta-models, including using machine learning technologies, to predict the situation with fires. A comparative analysis of their accuracy in predicting the situation with fires at the objects of the residential sector for a long-term period is considered.

Keywords: fire, data mining, systems analysis, time series, machine learning, fire statistics, forecasting

Введение

Большое количество различных разработанных прогнозных моделей и методов, которые постоянно совершенствуются, для исследователя создает сложность постоянного выбора. При этом точность прогноза зависит от множества факторов, таких как: предметная область и качество набора данных, конкретная реализация алгоритма модели, методика верификации результатов. Таким образом, существует необходимость автоматизации рутинной части вычислений, которая может быть реализована с помощью методов автоматического машинного анализа.

Основной алгоритм настоящего исследования соответствует этапам 4 и 5 методологии SEMMA (Sample, Explore, Modify, Model и Assess) [1], которая определяет этапы проведения интеллектуального анализа данных (Data Mining).

Стандарт SEMMA представляет унифицированный межотраслевой подход к итеративному процессу интеллектуального анализа данных. При его использовании исследователь располагает научными методами построения концепции проекта, его реализации и оценки результатов.

Предметной областью послужила обстановка с пожарами в населенных пунктах республики, выраженная в их суммарном количестве за сутки.

Пожаром, относящимся к категории техногенных чрезвычайных ситуаций, считается неконтролируемое горение вне специального очага, приводящее к ущербу. В данной работе не рассматриваются различные малозначительные загорания, не отнесенные к пожарам (не нанесшие материального ущерба), и природные пожары.

В Республике Беларусь государственный статистический учет пожаров осуществляет Национальный статистический комитет. В рамках этого учета сведения о пожарах и их последствиях на объектах жилого сектора собирает и предоставляет Министерство по чрезвычайным ситуациям [2].

Основная часть

Вопросы, касающиеся разведочного анализа и исследования аномалий с помощью модели на базе временных рядов, представлены в [3]. Кроме выявления скрытых закономерностей, эта модель используется для прогнозирования значений исследуемого показателя в будущем.

Вместе с тем методы прогнозирования, построенные на использовании различных модификаций модели авторегрессии и скользящего среднего, разработанной Боксом и Дженкинсом [4], требуют для использования привлечения специалистов высокого уровня. Настройка параметров таких моделей требует глубокого понимания математической составляющей этих подходов.

В настоящее время разработаны алгоритмы, позволяющие за счет применения методов машинного обучения оптимизировать участие исследователя при подборе моделей и их параметров. Очевидно, что на различных видах исходных данных модели будут давать различный результат, и что не существует идеальной модели для всех случаев.

Одним из вариантов примене-

ния методов машинного обучения является их автоматизация.

Автоматическое машинное обучение (AutoML) – это процесс автоматизации сквозного процесса применения машинного обучения при решении задач в различных предметных областях.

В классическом приложении машинного обучения исследователь должен применить подходящие методы предварительной обработки данных, конструирования признаков и выбора признаков. После этого он выбирает алгоритмы и оптимизирует гиперпараметры для снижения ошибки прогнозирования для конкретной модели. Так как многие из этих операций могут быть выполнены только экспертами, были предложены подходы, основанные на автоматизации сквозного процесса применения машинного обучения. Они дают возможность быстрого создания простых решений и моделей, которые по своим характеристикам превосходят модели, построенные вручную.

Для оценки работы различных моделей использован модуль AutoModel коммерческой платформы анализа данных RapidMiner, позволяющей создавать прогнозные модели машинного обучения для решения аналитических задач. RapidMiner объединяет весь жизненный цикл науки о данных, включая их подготовку, моделирование, визуализацию результатов, валидацию моделей, развертывание и оптимизацию.

При невозможности достичь требуемой точности или для ее улучшения в машинном обучении часто используют ансамбли методов. При их использовании алгоритмы учатся одновременно и могут исправлять ошибки друг друга. Модель, построенную на основе ансамбля, часто называют «метамоделью». Типичные методы объединения моделей в ансамбль представлены ниже.

Стекинг. Могут использоваться разнородные модели, которые образуют мета-модель, на вход которой подаются базовые модели, а выходом является итоговый прогноз.

Бэггинг. Используются однородные модели, которые параллельно обучаются независимо на различных исходных данных, а затем их результаты просто усредняются. Известным представителем данного метода является «случайный лес».

Бустинг. Однородные модели, которые обучаются последовательно, при этом последующая модель должна исправлять ошибки предыдущей. Один из наиболее популярных методов – градиентный бустинг.

Для оценки возможности применения прогнозных моделей использованы следующие методы:

1. Обобщенная линейная модель (Generalized Linear Model) представляет собой гибкое обобщение классической линейной регрессии, которое позволяет использовать переменные реакции, имеющие модели распределения ошибок, отличные от нормального распределения. Обобщает линейную регрессию, позволяя линейной модели быть связанной с переменной реакции через функцию [5].

Традиционная линейная модель часто неэффективна из-за того, что

в реальной жизни зависимости чаще всего не являются линейными. Поэтому разработаны более гибкие автоматические статистические методы, которые можно использовать для выявления и характеристики эффектов нелинейной регрессии. Эти методы называются «обобщенными аддитивными моделями».

В настройке регрессии обобщенная аддитивная модель имеет следующий вид (1):

$$E(Y \mid X_1, X_2, ..., X_p) = \alpha + f_1(X_1) + f_2(X_2) + \cdots + f_p(X_p), \quad (1)$$

где Y – результат; $X_1, X_2, ..., X_p$ – предикторы; f_j – неопределенные гладкие («непараметрические») функции.

Если моделировать каждую функцию, используя расширение базисных функций, то полученная модель может быть подогнана методом наименьших квадратов. В данном случае каждая функция подбирается с помощью более сглаживающей диаграммы рассеяния (например, кубического сглаживающего сплайна или сглаживания ядра) и предоставляется алгоритм для одновременной оценки всех p-функций.

2. Деревья решений (Decision Tree) – это иерархическая древовидная структура, состоящая из правил принятия решений вида «если ..., то ...». Правила генерируются автоматически в процессе обучения на обучающем множестве.

Существенным преимуществом деревьев решений является то, что они легко интерпретируются и понятны людям. В связи с этим и сходством с моделью принятия решений человеком деревья решений получили широкое распространение.

Процесс построения деревьев решений состоит в последовательном, рекурсивном разбиении обучающего набора на подмножества с применением решающих правил в узлах. Процесс разбиения продолжается до тех пор, пока все узлы в конце всех ветвей не будут объявлены листьями. Объявление узла как листа может произойти естественным образом (если он будет содержать единственный объект, или объекты только одного класса) или по достижении некоторого условия остановки, задаваемого пользователем (например, минимально допустимое число примеров в узле или максимальная глубина дерева) [6].

В настоящее время разработано значительное число алгоритмов обучения деревьев решений, но наибольшее распространение и популярность получили следующие:

ID3 (Iterative Dichotomizer 3) – алгоритм позволяет работать только с дискретной целевой переменной, поэтому деревья решений, построенные с помощью данного алгоритма, являются классифицирующими. Число потомков в узле дерева не ограничено. Не может работать с пропущенными данными.

C4.5 – усовершенствованная версия алгоритма ID3, в которую добавлена возможность работы с пропущенными значениями атрибутов.

CART – алгоритм обучения деревьев решений, позволяющий использовать как дискретную, так и непрерывную целевую перемен-

ную, то есть решать как задачи классификации, так и регрессии. Алгоритм строит деревья, которые в каждом узле имеют только два потомка.

Построение дерева решений состоит из нескольких этапов:

– Выбор признака, по которому будет производиться разбиение в данном узле (атрибута разбиения).

– Выбор критерия остановки обучения.

– Выбор метода отсечения ветвей (упрощения).

– Оценка точности построенного дерева.

3. Случайный лес (Random Forest) – алгоритм машинного обучения, заключающийся в использовании ансамбля решающих деревьев. Сочетает в себе две основные идеи: метод бэггинга Бреймана и метод случайных подпространств, предложенный Тин Кам Хо. Основная идея заключается в использовании большого ансамбля решающих деревьев, каждое из которых само по себе дает очень невысокое качество классификации, но за счет их большого количества результат получается хорошим.

Решающие деревья являются хорошим семейством базовых классификаторов для бэггинга, поскольку они достаточно сложны и могут достигать нулевой ошибки на любой выборке. Метод случайных подпространств позволяет снизить коррелированность между деревьями и избежать переобучения. Базовые алгоритмы обучаются на различных подмножествах признакового описания, которые также выделяются случайным образом [7].

Алгоритм построения случайного леса, состоящего из N деревьев, следующий:

Для каждого $n = 1,\ldots, N$:

Методом бутстрапа создается выборка X_n.

По выборке X_n строится решающее дерево b_n:

– по заданному критерию выбирается лучший признак, делается разбиение в дереве по нему и так до исчерпания выборки;

– дерево строится, пока в каждом листе не более $n_{\min}$ объектов или пока не достигается определенная глубина дерева;

– при каждом разбиении сначала выбирается m случайных признаков из n исходных, и оптимальное разделение выборки ищется только среди них.

Итоговый классификатор $a(x)= \frac{1}{N}\sum_{n=1}^{N} b_n(x)$ для задачи классификации выбирается решение голосованием по большинству, в задаче регрессии – средним.

Рекомендуется в задачах классификации брать $m=\sqrt{n}$, а в задачах регрессии – $m=\frac{n}{3}$, где n – число признаков. Также рекомендуется в задачах классификации строить каждое дерево до тех пор, пока в каждом листе не окажется по одному объекту, а в задачах регрессии – пока в каждом листе не окажется по пять объектов [7].

4. Градиентный бустинг (Gradient Boosted Trees) – техника машинного обучения для задач классификации и регрессии, которая строит модель предсказания в форме ансамбля слабых предсказывающих моделей, обычно деревьев решений.

Бустинг представляет собой жадный алгоритм[1] построения композиции алгоритмов. Основная идея заключается в том, чтобы, имея множество относительно слабых алгоритмов обучения, построить их хорошую линейную комбинацию. Сходство с бэггингом в том, что базовый алгоритм обучения фиксирован. Отличие состоит в том, что обучение базовых алгоритмов для композиции происходит итеративно, и каждый следующий алгоритм стремится компенсировать недостатки композиции всех предыдущих алгоритмов.

На примере бустинга стало ясно, что хорошим качеством могут обладать сколь угодно сложные композиции классификаторов, при условии, что они правильно настраиваются. Это развеяло существовавшие долгое время представления о том, что для повышения обобщающей способности необходимо ограничивать сложность алгоритмов.

Впоследствии этот феномен бустинга получил теоретическое обоснование. Оказалось, что взвешенное голосование не увеличивает эффективную сложность алгоритма, а лишь сглаживает ответы базовых алгоритмов. Эффективность бустинга объясняется тем, что по мере добавления базовых алгоритмов увеличиваются отступы обучающих объектов. Причем бустинг продолжает раздвигать классы даже после достижения безошибочной классификации обучающей выборки [8].

5. Теория опорных векторов была разработана В.Н. Вапником в 1990 году. Метод опорных векторов (Support Vector Machine) – семейство алгоритмов бинарной классификации, основанных на обучении с учителем, использующих линейное разделение пространства признаков с помощью гиперплоскости.

Основная идея метода заключается в отображение векторов пространства признаков, представляющих классифицируемые объекты, в пространство более высокой размерности. Это связано с тем, что в пространстве большей размерности линейная разделимость множества оказывается выше, чем в пространстве меньшей размерности. Причины этого интуитивно понятны: чем больше признаков используется для распознавания объектов, тем выше ожидаемое качество распознавания.

После перевода в пространство большей размерности в нем строится разделяющая гиперплоскость. При этом все векторы, расположенные с одной «стороны» гиперплоскости, относятся к одному классу, а расположенные с другой – ко второму. Также, по обе стороны основной разделяющей гиперплоскости, параллельно ей и на равном расстоянии от нее строятся две вспомогательные гиперплоскости, расстояние между которыми называют зазор.

Задача заключается в построении разделяющей гиперплоскости таким образом, чтобы максимизировать зазор – область пространства признаков между вспомогательными гиперплоскостями, в которой не должно быть векторов. Предполагается, что разделяющая гиперплос-

[1] Жадный алгоритм (Greedy algorithm) — алгоритм, заключающийся в принятии локально оптимальных решений на каждом этапе, допуская, что конечное решение также окажется оптимальным.

кость, построенная по данному правилу, обеспечит наиболее уверенное разделение классов и минимизирует среднюю ошибку распознавания.

Векторы, которые попадают на границы зазора (т.е. будут лежать на вспомогательных гиперплоскостях), называют опорными векторами [9].

В качестве метрики оценки ошибки прогнозирования используем корень среднеквадратичной ошибки (2):

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(Z(t) - \hat{Z}(t))^2}, \quad (2)$$

где N – количество периодов; $Z(t)$ – фактическое значение временного ряда; $\hat{Z}(t)$ – прогнозное.

Для расчета RMSE исходный набор данных разделяется на две части: обучающую и тестовую. Обучающая используется для тренировки модели, а тестовая – для вычисления величины ошибки на основании прогноза. В рамках данного исследования набор данных разбит соответственно на 60 % и 40 %, что частично определяется спецификой работы с модулем AutoModel.

Модуль AutoModel позволяет выполнить оценку точности прогноза для нескольких основных видов моделей, результаты оценки которых представлены в таблице 1.

Набор данных за период с 2011 по 2020 год извлечен из базы данных программного комплекса «Учет ЧС» [10] и представлен в виде временного ряда. Информация сгруппирована по суткам.

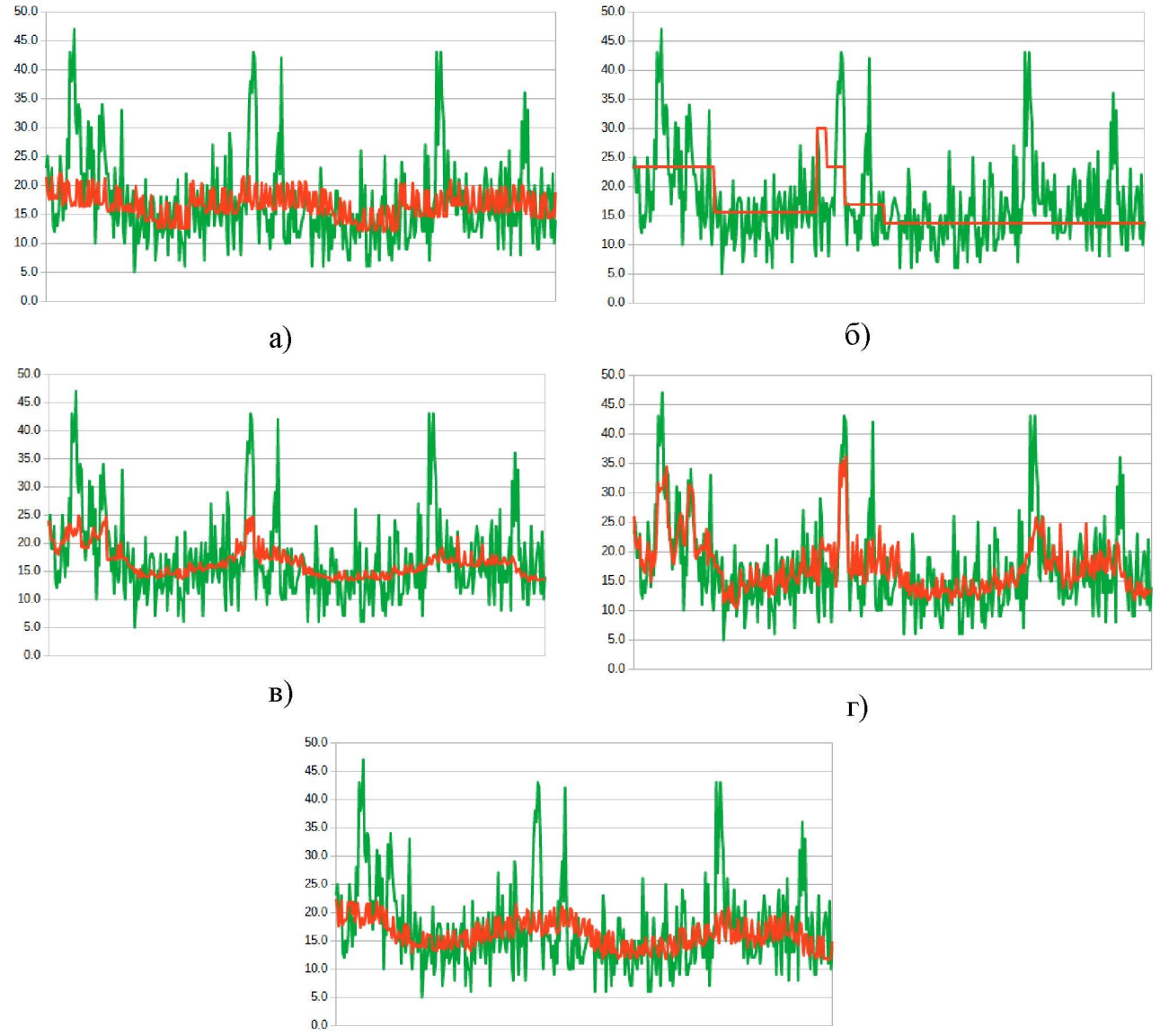
Таблица 1. – Оценка точности методов

№ п/п	Модель, описание	RMSE
1	Обобщенная линейная модель (GLM)	6,1
2	Дерево решений (DT)	6,37
3	Случайный лес (RF)	5,79
4	**Градиентный бустинг (GBT)**	**5,26**
5	Метод опорных векторов (SVM)	5,96

Графическое представление исторических данных в сравнении с результатами прогнозирования представлено на рисунке.

На основании величины квадратного корня среднеквадратичной ошибки прогнозирования наилучшими методами для использованного набора данных являются градиентный бустинг и случайный лес.

Анализ графического представления прогноза показал, что пиковые всплески были предсказаны наиболее точно также методом градиентного бустинга.



а) б)

в) г)

д)

а) обобщенная линейная модель; б) деревья решений; в) случайный лес;
г) градиентный бустинг; д) метод опорных векторов
Рисунок. – Ежедневный прогноз на год (оранжевый) в сравнении
с реальными данными за 2020 год

Заключение

1. В статье рассмотрен подход к выбору прогнозных моделей на базе алгоритмов машинного обучения для прогнозирования обстановки с пожарами в жилом секторе республики на основании исторического набора данных за 10 лет (2011–2020 годы).

2. Проведена оценка моделей и алгоритмов, входящих в модуль автоматического машинного обучения платформы анализа данных RapidMiner.

3. На основании величины ошибки прогнозирования определены наилучшими методами прогнозирования обстановки с пожарами в жилом секторе: градиентный бустинг и случайный лес.

ЛИТЕРАТУРА

1. Azevedo, A. KDD, SEMMA and CRISP-DM: A parallel overview [Electronic resource] /

A. Azevedo, M. Santos // IADIS Multi Conf. on Computer Science and Information Systems, Amsterdam, 22-27 July 2008 / Intern. Assoc. for Development if the Inform. Soc.; Associate Ed.: Luís Rodrigues and Patrícia Barbosa. – Amsterdam, 2008. – P. 182–185.

2. Об утверждении формы государственной статистической отчетности 1-ос (пожары) «Отчет о пожарах (кроме лесных) и последствиях от них» и указаний по ее заполнению : постановление Национ. статист. ком. Респ. Беларусь от 27 июня 2017 г. № 49 // Национ. правовой Интернет-портал Респ. Беларусь [Электронный ресурс]. – Минск, 2021 – Режим доступа : https://pravo.by/upload/docs/op/T21703807p_1501016400.pdf. – Дата доступа : 20.04.2021.

3. Татур, М.М. Анализ временных рядов как элемент процесса интеллектуального анализа данных обстановки с техногенными пожарами / М.М. Татур, А.Г. Иваницкий, В.М. Проровский // Чрезвычайные ситуации: предупреждение и ликвидация. – 2021. – № 21(49). – С. 56–68.

4. Бокс, Дж. Анализ временных рядов, прогноз и управление / Дж. Бокс, Г. Дженкинс: Пер. с англ. // Под ред. В.Ф. Писаренко. – М.: Мир, 1974 кн. 1. – 406 с.

5. Dunn, Peter K. Generalized Linear Models With Examples in R. / Peter K. Dunn, Gordon K. Smyth // Springer, New York, NY. – 2018. – P. 562. https://doi.org/10.1007/978-1-4419-0118-7.

6. Classification, Decision Trees and k Nearest Neighbors [Electronic resource]. – Mode of access: https://medium.com/open-machine-learning-course/open-machine-learning-course-topic-3-classification-decision-trees-and-k-nearest-neighbors-8613c6b6d2cd. – Date of access: 20.10.2021.

7. Bagging and Random Forest [Electronic resource]. – Mode of access: https://medium.com/open-machine-learning-course/open-machine-learning-course-topic-5-ensembles-of-algorithms-and-random-forest-8e05246cbba7. – Date of access: 20.10.2021.

8. Gradient Boosting Forest [Electronic resource]. – Mode of access: https://medium.com/open-machine-learning-course/open-machine-learning-course-topic-10-gradient-boosting-c751538131ac. – Date of access: 20.10.2021.

9. Cristianini, N. An Introduction to Support Vector Machines and Other Kernel-based Learning Methods / N. Cristianini, J. Shawe-Taylor — Cambridge University Press. – 2000. – 189 P.

10. Разработать программный комплекс сбора и анализа информации о чрезвычайных ситуациях и их последствиях: отчет о НИР (заключ.) / В.М. Проровский [и др.]. Науч.-исслед. ин-т пожар. безопасности и проблем чрезвычайн. ситуаций МЧС Респ. Беларусь. – Минск, 2017. – 54 с. – № ГР 20163551. – Деп. в БелИСА 04.07.2018, № Д201828.

REFERENCES

1. Azevedo, A. KDD, SEMMA and CRISP-DM: A parallel overview [Electronic resource] / A. Azevedo, M. Santos // IADIS Multi Conference on Computer Science and Information Systems, Amsterdam, 22-27 July 2008 / Intern. Assoc. for Develop-

ment if the Inform. Soc.; Associate Ed.: Luís Rodrigues and Patrícia Barbosa. – Amsterdam, 2008. – P. 182–185

2. Ob utverzhdenii formy gosudarstvennoj statisticheskoj otchetnosti 1-os (pozhary) «Otchet o pozharah (krome lesnyh) i posledstviyah ot nih» i ukazanij po ee zapolneniyu. : postanovlenie Nacion. statist. kom. Resp. Belarus' ot 27 iyunya 2017 g. № 49 // Nacion. pravovoj Internet-portal Resp. Belarus' [Elektronnyj resurs]. – Minsk, 2021. – Rezhim dostupa: https://pravo.by/upload/docs/op/T21703807p_1501016400.pdf. – Data dostupa : 20.04.2021.

3. Tatur, M.M. Analiz vremennyh ryadov kak element processa intellektual'nogo analiza dannyh obstanovki s tekhnogennymi pozharami / M.M. Tatur, A.G. Ivanickij, V.M. Prorovskij // Chrezvychajnye situacii: preduprezhdenie i likvidaciya. – 2021. – № 21(49). – S. 56–68.

4. Boks, Dzh. Analiz vremennyh ryadov, prognoz i upravlenie / Dzh. Boks, G. Dzhenkins: Per. s angl. // Pod red. V.F. Pisarenko. – M.: Mir, 1974 kn. 1. – 406 s.

5. Dunn, Peter K. Generalized Linear Models With Examples in R. / Peter K. Dunn, Gordon K. Smyth // Springer, New York, NY. – 2018. – P. 562. https://doi.org/10.1007/978-1-4419-0118-7.

6. Classification, Decision Trees and k Nearest Neighbors [Electronic resource]. – Mode of access: https://medium.com/open-machine-learning-course/open-machine-learning-course-topic-3-classification-decision-trees-and-k-nearest-neighbors-8613c6b6d2cd. – Date of access: 20.10.2021.

7. Bagging and Random Forest [Electronic resource]. – Mode of access: https://medium.com/open-machine-learning-course/open-machine-learning-course-topic-5-ensembles-of-algorithms-and-random-forest-8e05246cbba7. – Date of access: 20.10.2021.

8. Gradient Boosting Forest [Electronic resource]. – Mode of access: https://medium.com/open-machine-learning-course/open-machine-learning-course-topic-10-gradient-boosting-c751538131ac. – Date of access: 20.10.2021.

9. Cristianini, N. An Introduction to Support Vector Machines and Other Kernel-based Learning Methods / N. Cristianini, J. Shawe-Taylor – Cambridge University Press. – 2000. – 189 P.

10. Razrabotat' programmnyj kompleks sbora i analiza informacii o chrezvychajnyh situaciyah i ih posledstviyah: otchet o NIR (zaklyuch.) / V.M. Prorovskij [i dr.]. Nauch.-issled. in-t pozhar. bezopasnosti i problem chrezvychajn. situacij MCHS Resp. Belarus'. – Minsk, 2017. – 54 s. – № GR 20163551. – Dep. v BelISA 04.07.2018, № D201828.