РАЗРАБОТКА АЛГОРИТМА КЛАССИФИКАЦИИ НА МОДЕЛИ АКАДЕМИЧЕСКИХ ДАННЫХ С ПОМОЩЬЮ ГИПЕРПАРАМЕТРИЧЕСКОЙ НАСТРОЙКИ ДЛЯ НАЙМА ПРЕПОДАВАТЕЛЕЙ В ИРАКСКИХ УНИВЕРСИТЕТАХ

DOI: 10.36724/2072-8735-2022-16-9-33-38

Manuscript received 12 July 2022;
Accepted 20 August 2022

Ключевые слова: *контролируемая классификация, логистическая регрессия, support vector classifier (SVC), K Nearest Neighbors (KNN), Gaussian Naive Bayes, decision tree, random forest, повышение градиента, Linear Discriminant Analysis (LDA), оптимизация гиперпараметров*

Аль-Дулаими Омар Хатем Заидан,
Воронежский государственный технический университет, г. Воронеж, Россия, ***oomar1982@yandex.ru***

В этой статье исследователь предлагает метод классификации для трудоустройства в иракских университетах с использованием полных данных о размещении по категориям для набора данных о занятости в иракских учебных заведениях. Анализ научной литературы показывает, что данная проблематика исследована недостаточно и требует своего решения. Цель исследования – разработка метода классификации, основанного на генетических алгоритмах, для получения наилучших результатов для решения проблемы найма новых профессоров в иракских университетах, получения точных результатов и сравнения нескольких методов для решения этой проблемы. Представлены алгоритмы классификации обучения с учителем, такие как логистическая регрессия, Support Vector Classifier (SVC), Closest Neighbors KNN, Gaussian Naive Bayes алгоритм, decision tree, random forest, gradient boosting и linear factor analysis (LDA). Оптимизация гиперпараметров также используется для контролируемых алгоритмов для достижения лучших результатов. По предложенным алгоритмам достоверность и точность полученных результатов обеспечивалась посредством пятикратной проверки. Экспериментальные результаты показали, что с помощью настройки гиперпараметров в Linear Discriminant Analysis (LDA) можно повысить точность результатов, а также получить лучший результат по сравнению с другими алгоритмами. Во внимание принимались такие критерии анализа, как средняя точность, средний балл, G-Mean и Средняя ROC AUC. Именно по трем первым критериям линейный дискриминантный анализ демонстрирует наиболее высокую точность оценивания. Установлено, что среди всех применяемых алгоритмов классификации, классификация линейного дискриминантного анализа сравнительно более точна во всех других сценариях. Обнаружено, что добавление решателя параметров lsqr и усадки 0,81 в линейный дискриминантный анализ может повысить производительность. Обосновано, что предложенный алгоритм классификации на модели академических данных способен решать проблему найма преподавателей в университет.

Для цитирования:
Аль-Дулаими Омар Хатем Заидан. Разработка алгоритма классификации на модели академических данных с помощью гиперпараметрической настройки для найма преподавателей в иракских университетах // T-Comm: Телекоммуникации и транспорт. 2022. Том 16. №9. С. 33-38.

For citation:
Al-Dulaimi Omar Hatem Zaidan. (2022) Development of a classification algorithm based on an academic data model using hyperparametric tuning for hiring teachers at Iraqi universities. *T-Comm*, vol. 16, no. 9, pp. 33-38. (*in Russian*)

Введение

По мере того, как университеты смотрят в будущее, возникают новые проблемы в виде растущей конкуренции и общественного спроса на образование, что побуждает университеты разрабатывать более совершенную систему учебных программ для студентов. Кроме того, высшее образование признает, что развитие услуг оказывает значительное влияние на удовлетворение ожиданий и потребностей студентов и рынка труда [1]. Одной из важнейших бизнес-операций в университете является набор новых педагогических кадров, независимо от того, имеют ли они степень бакалавра (первого на кафедре), степень магистра или аспирантуру. Это означает, что в отсутствие новых преподавателей не будет новых бизнес-операций в университетах. Подбор новых педагогических кадров является одной из основных задач, которые должны быть решены в университете, поскольку обеспечивает преемственность основной работы, обучения и научных исследований. В результате университеты должны набирать новый преподавательский состав, чтобы продолжать работу в качестве существующего учреждения [2]. Глобальные университеты активно ищут новые способы поиска и приема на работу преподавательского состава. Широко распространено мнение, что университеты нанимают новых профессоров, потому что университет должен олицетворять плюрализм: обеспечивать равное образование для всех. Согласно одному исследованию, вербовка может быть использована для усиления аргументов подотчетности [3].

Было проведено несколько исследований с использованием различных процедур для использования данных об окружающей среде для прогнозирования занятости [4]. Модели вероятностной классификации позволяют определить степень неопределенности, связанной с прогнозом [5]. Хорошо известными методами классификации являются K-Nearest Neighborhood (KNN) [6], Logistic Regression [7], Support Vector Classifier (SVC) [8], Gaussian Naive Bayes [9], Decision Tree [10], Random Forest [11], Gradient Boosting [12], and Linear Discriminant Analysis (LDA) [13], Ensemble Voting Classifier [14]. В этой статье будет разработан метод классификации, основанный на генетических алгоритмах, для получения наилучших результатов для решения проблемы найма новых профессоров в иракских университетах, получения точных результатов и сравнения нескольких методов для решения этой проблемы.

Материалы и методы

Используемый набор данных представляет собой данные о занятости научных компетенций (бакалавриат, магистратура и аспирантура) в иракских университетских учреждениях, которые содержат 15 атрибутов с 216 строками данных. Включает в себя процентное соотношение (бакалавриат, магистратура и аспирантура). Специализация также включает в себя степень, пол, опыт работы и предложения по заработной плате для каждой категории. Атрибуты представляют собой процент выдающихся обладателей степени бакалавра (ssc_p), которые имеют процентные числа от 0 до 100, Мастер (ssc_b и hsc_b), который содержит процент магистров в (чистых науках) или гуманитарных и других дисциплинах (hsc_p), который содержит числа от 0 до 100, Степень_содержит относительные числа от 0 до 100, Аспирантура (степень_) содержит область образования с определенной степенью, а workex имеет строковый тип данных, который содержит опыт работы профессора и специализацию Аспирантура (чистой науки или гуманитарные науки и другие специализации) (hsc_s).

Предварительная обработка данных

На этапе предварительной обработки данных был проведен анализ полученного набора данных и обнаружено 67 пустых строк в атрибуте зарплата. Все столбцы зарплаты были пустыми, если в статусе стояло «Не размещено», поэтому этот атрибут был удален из набора данных. После этого были удалены неиспользуемые столбцы, а именно: пол, ssc_b, hsc_b и зарплата. Столбец «Пол» был удален, поскольку в анализе гендер не использовался. Столбцы ssc_b и hsc_b удалены, поскольку данные для обоих столбцов получены из столбцов ssc_p и hsc_p. При этом столбец зарплаты удален, поскольку он не используется в анализе в данной статье. Следующий этап – кластерный анализ с использованием нескольких методов, которые будут объяснены в подразделах ниже.

Logistic regression (Логистическая регрессия)

Логистическая регрессия аналогична линейной регрессии, если используется с биномиальной переменной отклика. По сравнению с результатом отношения Mantel Haenzel Odds Ratio (OR) неоспоримым фактом является то, что можно использовать непрерывную интерпретацию, и легче иметь дело с более чем двумя объясняющими переменными одновременно [15]. Но хотя эта последняя особенность может показаться незначительной, она важна, когда исследователей интересует влияние различных объясняющих переменных на переменную отклика. Когда несколько независимых переменных обрабатываются независимо, дисперсия между переменными игнорируется, и возникают неподдерживающие результаты. Модель логистической регрессии будет предсказывать вероятность исхода на основе индивидуальных характеристик. Поскольку шанс также является отношением, то будет проиллюстрирован логарифм вероятности, определяемый выражением:

$$\log\left(\frac{\pi}{1-\pi}\right) = \beta 0 + \beta 1X1 + \beta 2X2 + \cdots + \beta mXm, \qquad (1)$$

где π обозначает вероятность события, βi — коэффициенты регрессии, относящиеся к эталонному кластеру, xi – объясняющее переменное. Теперь должна быть представлена важная концепция. Эталонный кластер, обозначенный β0, состоит из лиц, представляющих референтный уровень каждой переменной *x1...m*. После этого исследователи могут изучить способ установки эталонного уровня [7].

Support Vector Classifier (SVM))(*Классификатор опорных векторов)*

В этом разделе представлены основные понятия классификатора опорных векторов (SVM). SVM является частью общей линейной классификации. Особенность SVM заключается в том, что он может свести к минимуму ошибки, связанные с эмпирической классификацией, и в то же время максимизировать геометрические поля. Таким образом, SVM также можно назвать классификатором максимальной маржи. SVM может отображать или подразделять входные векторы в пространстве более высокого измерения, где определена разде-

лительная гиперплоскость. Разделяющая гиперплоскость полезна для максимизации или увеличения расстояния между двумя параллельными гиперплоскостями. Таким образом, можно сделать вывод, что чем больше поле гиперплоскости, тем меньше вероятность ошибочной классификации (см. уравнение (2)):

$$\{(x1, y1), (x2, y2), (x3, y3), (x4, y4), \dots, (xn, yn)\}, \quad (2)$$

в этом уравнении y_n=1 / -1 – это константа, представляющая положение точки X_n в n со значением n, которое является количеством выборок. Каждый X_n представляет собой вектор действительных значений размерности p. Настройка масштаба становится очень важной, потому что она должна поддерживать атрибут или переменную с большим значением дисперсии. Чтобы этот процесс можно было реализовать, разделим гиперплоскость, для чего требуется:

$$w.\ x + b = 0, \quad (3)$$

где b – скалярное значение, а w – вектор размерности p. Если параметр b отсутствует, то гиперплоскость выйдет за границы, так что результирующее решение будет ограниченным [8].

K-Nearest Neighbor (KNN) (*K-ближайшие соседи)*

Алгоритм K-Nearest Neighbor (KNN) – это алгоритм, который использует все элементы данных для сравнения близости между точками сбора данных обучения и тестирования. K является фокусом набора данных элементов для обучения. Вес расчета расстояния умножается на частоту для расчета среднего веса. Средний вес представляет собой значение, указывающее местоположение набора данных, относящегося к точке фокусировки.

Данные обучения представлены многомерным пространством. Это пространство разделено на несколько разделов на основе классификации обучающих данных. Существует несколько способов измерения расстояния между набором данных для обучения и тестирования, например, Евклидово расстояние. См. следующее уравнение:

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (xi - yi)^2} \quad (4)$$

Переменное взвешивание реализуется путем нормализации значения наибольшего собственного вектора в матрице отношений. Эта деятельность проводится перед сравнением значений шкалы, влияющих на значимость переменных [6].

Gaussian Naive Bayes Method (Гауссовский наивный байесовский метод)

Naive Bayes метод классификации основан на теореме Bayes'. Метод классификации с использованием вероятностных и статистических методов, предложенный британским ученым Thomas Bayes, называется теоремой Bayes, поскольку он предсказывает будущие возможности на основе опыта. Пространство с наибольшей вероятностью принадлежности считается классом сфокусированных точек данных. Процесс классификации осуществляется путем определения категорий в новых обучающих данных. Этот классификатор является вероятностным классификатором, показанным в уравнении (5):

$$p(A / B) = \frac{p(B / A)p(B)}{p(B)} \quad (5)$$

Вероятность в *Naive Bayes* классификации – это уравнение атрибута $P(x_i|c_j)$, где x_i – i-й атрибут расстояния до захваченных данных. Поэтому уравнение расчета можно описать в виде (6):

$$(y|x1,\dots,xn)a\ P\ (y)\ \prod_{i=1}^{n} P(xi|y) \quad (6)$$

Точки данных классифицируются по пространствам, значение которых (6) является максимальным. Следовательно, уровень класса точек данных определяется формулой (6). Если в пространстве много переменных с непрерывными значениями, то используется уравнение Gaussian Naive Bayes, которое берется из распределения Gaussian. Поскольку в нем используется распределение Gaussian, уравнение корректируется до (7):

$$\rho(x_i / y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x_i - \mu y)^2}{2\sigma_y^2}\right) \quad (7)$$

Decision Tree (Древо решений)

Алгоритм Decision Tree выполняет интеллектуальный анализ данных с помощью модели прогнозирования с использованием дерева, или иерархической структуры. Концепция Decision Tree преобразует данные в иерархию и правила принятия Decision Tree решений представлено в виде древовидной структуры, которая имеет листья, корни и ветви, подобные дереву. Каждый узел имеет ровно одно ребро. Если есть узел с выходным краем, он называется тестовым узлом. В Decision Tree каждый тестовый узел разветвляется на два или более пространства в соответствии со значением входного атрибута. В некоторых случаях это состояние относится к диапазону [10].

Каждый лист нацелен на значение в каждом выбранном классе. Альтернативно лист может хранить вектор вероятности, который представляет вероятность того, что целевой атрибут имеет заданное значение. Маркировка каждого узла присваивается тестовому атрибуту и несет соответствующее значение. Дерево решений можно интерпретировать как набор гиперплоскостей, где каждая ветвь идет к одной из осей. Обычно сложность дерева можно измерить по общему количеству узлов, количеству листьев, глубине дерева и количеству используемых атрибутов.

Random Forest (Случайный лес)

Random Forest может быть предиктором, который включает в себя набор *M* рандомизированных деревьев регрессии. Цель этого сегмента — дать краткое, но математически уникальное представление алгоритмического приложения для построения Random Forest. Самая последняя схема — это статистическая регрессионная оценка, на протяжении которой определяется входной случайный вектор степени ассоциирования $X \in X \subset Rp$ и цель которой состоит в том, чтобы ожидать квадратную интегрируемую случайную реакцию $Y \in R$.

С помощью средств оценки выполняется регрессия $m(x) = E[Y|X = x]$. Данное допущение может быть образцом обучения $Dn = ((X1, Y1),\dots, (Xn, Yn))$ внештатных случайных величин, выделенных в связи с тем, что парадигма внештатного работника объединяет (X, Y).

Цель состоит в том, чтобы применить набор информации Dn для составления ассоциированной оценки степени mn: $X \rightarrow R$ выполнения m [11].

Для *j-th* дерева внутри семейства ожидаемая стоимость запроса по причине x обозначается через средние значения *mn(x; θj , Dn)*, где в *θl, ..., θM* – внештатные случайные величины, выделенные последовательно как широко распространенная переменная угроза, где переменная угроза – внештатный сотрудник *Dn*. Далее переменная *θ* нанимается для повторной выборки обучающего набора перед созданием отдельных деревьев и для выбора серийных рекомендаций по разбиению более уникальных определений, которые будут даны позже. С математической точки зрения оценка *j-th* дерева принимает уравнение:

$$m_n(x_i O_j, D_n) = \sum_{i \in D_n(oj)} \frac{1x_i \in A_n(x_i O_j, D_n)\gamma_i}{N_n(x_i O_j, D_n)}, \quad (8)$$

где *Dn*(θj)* – это множество информационных факторов, выбранных до построения дерева, An *(x; θj, Dn)* – это ячейка, содержащая *x, a Nn(x; θj, Dn)* – это множество (заранее выбранных) факторов, составляющих An *(x; θj, Dn)*. На этом этапе ученый говорит, что деревья смешиваются для получения (конечной) оценки леса:

$$m_{M,n}(x, \theta_i, ..., \theta_m, D_n) = \frac{1}{M} \sum_{j=1}^{M} m_n(x, \theta_j, D_n). \quad (9)$$

В пакете R random Forest значение по умолчанию M (разнообразие деревьев в лесу) равно n tree = 500 (5 сотен). Поскольку M также выбран без разбора массивным (ограниченным полностью с помощью доступных вычислительных ресурсов), было бы разумно, исходя из мотива моделирования чтения, позволить M иметь тенденцию к бесконечности и рассматривать его как альтернативу (1) (бесконечной) оценке леса:

$$m_{\infty,n}(x; D_n) = E_\theta[m_n(x; \theta, D_n)]. \quad (10)$$

В этом определении *Eθ* указывает ожидаемое значение, связанное со случайным параметром Θ, в соответствии с *Dn*. На самом деле, операция «$M \rightarrow \infty$ » доказывается законом больших чисел, который утверждает, что он почти наверняка информативен в отношении этого предельного вычисления. В дальнейшем для упрощения обозначений исследователи могут писать m ∞, *n (x)* вместо *mn, n (x; Dn)*:

$$\lim_{M \to \infty} m_{M,n}(x; \theta_1, ..., \theta_M, D_n) = m_{\infty,n}(x; D_n) \quad (11)$$

Случайность элементов явно не рассматривается, но неявно используется для введения другой случайности, и каждое дерево корректируется, чтобы соответствовать независимой, вводной, помеченной выборке исходной информации. Организации, участвующие в бутстрап-выборке, предоставляют частичный *θj*. Затем, когда узлы разорваны, каждый отдельный узел найдет простейшую долю выбранного набора из m предикторов вместо всех p предикторов. Организация выборки предикторов дает остаток *θj* [16].

В качестве первого шага целесообразно расширить их до тех пор, пока конечные узлы не станут чистыми (классификация) или не уменьшится диапазон точек знаний бусины для каждого конечного узла (регрессия). Самые последние советы доминируют в максимальном диапазоне терминальных узлов. Следующие деревья объединяются путем выбора невзвешенного, если ответ является категоричным (классификация), или среднего невзвешенного, если ответ является непрерывным (регрессия).

Gradient boost (Повышение градиента)

Каждая убыточная сделка назначается случайным образом, после чего следуют основные необходимые шаблоны обучения. При желании, используя конкретную производительность потерь *(y, f)* и / или пользовательскую поисковую систему *h (x),* ответ на оценки параметров может быть правильным. Чтобы опираться на это, рассчитываем определить совершенно новую характеристику *h(x;t)* как важнейшую параллель с отрицательным градиентом *{gt(xi)} iN=1* по установленным данным:

$$g_t(x) = E_y[\frac{\partial \Psi(y, f(x))}{\partial f(x)}|x]_{f(x)=f^{t-1}(x)} \quad (12)$$

Вместо того, чтобы искать окончательный ответ для приращения надбавки в пространстве производительности, можно просто выбрать новое приращение производительности, которое будет в первую очередь коррелировать с - gt (x). Это позволяет заменить, вероятно, ужасно трудоемкую оптимизационную задачу классической задачей уменьшения методом наименьших квадратов:

$$(\rho_t, \theta_t) = \arg\min_{\rho,\theta} \sum_{i=0}^{N} [-gt(x_i) + ph(x_i; \theta)]^2. \quad (13)$$

Точный вид производной формулы со всеми соответствующими формулами может сильно зависеть от выбора внешнего вида (y, f) и h (x, θ). Некоторые распространенные образцы этих алгоритмов можно найти у Milton Friedman (2001) [12].

Linear Discriminant Analysis (Линейный дискриминантный анализ)

Цель метода LDA состоит в том, чтобы спроецировать исходную матрицу знаний в низкоразмерное пространство. Для этого нужно сделать три шага. Первый шаг — вычислить разницу между совершенно разными классами (т. е. расстояние между автомобилями разных классов). Это называется межклассовым распределением, или межклассовой матрицей. Второй шаг заключается в вычислении среднего значения для каждой категории и отклонении между образцами, называемом дисперсией стекла или матрицей стекла. Третий шаг — построить низкоразмерное пространство, чтобы максимизировать дисперсию между классами и минимизировать дисперсию между классами.

Оптимизация результатов

Определение оптимальных параметров (гиперпараметров) для каждой модели важно для менее предвзятой оценки прогнозирующей силы модели. Случайный поиск имеет желаемые свойства в более высоких измерениях, чем поиск по сетке, и не имеет недостатков [17].

Проведенное сравнение позволяет оптимизировать результаты с использованием настройки гиперпараметров для повышения их точности.

Наилучшие параметры выбираются из результатов случайных экспериментов, или так называемых случайных алгоритмов поиска. После того, как лучший параметр найден, к нему можно применить гиперпараметр, чтобы получить результат с наибольшей точностью в эксперименте. Чем больше итераций экспериментов с алгоритмом случайного поиска выполняется для получения наилучших параметров, тем больше шансов получить более точные результаты.

Результаты

Наборы данных, которые были взяты для этого эксперимента, предназначены для набора новых научных кадров в иракских университетах – академические факторы и факторы занятости, влияющие на занятость в университетах [18]. Алгоритмы контрольной классификации, выбранные для экспериментов, включают логистическую регрессию, классификатор опорных векторов, K ближайших соседей, гауссовский наивный байесовский алгоритм, дерево решений, случайный лес, повышение градиента и линейный дискриминантный анализ. Используемые меры оценки производительности – Точность, Оценка F1, G-Mean и Оценка ROC AUC. Для сравнения различных алгоритмов классификации используется k-кратная перекрестная проверка. Для каждого алгоритма применяется 5-кратная перекрестная проверка. Данные по эксперименту представлены в таблице 1.

Таблица 1

Результаты эксперимента

	Средняя точность для 5 раз	*Средний балл F1 для 5 раз*	*Среднее (G-Mean) для 5 раз*	*Средняя ROC AUC Оценка за 5 раз*
Logistic Regression	0.852083	0.881472	0.833323	**0.940221**
Support Vector Classifier	0.820833	0.865375	0.762810	0.916292
K Nearest Neighbors	0.826894	0.865879	0.786156	0.860009
Gaussian Naive Bayes	0.832576	0.870851	0.816063	0.899945
Decision Tree	0.801326	0.847351	0.801294	0.808860
Random Forest	0.832765	0.871977	0.817492	0.931646
Gradient Boosting	0.857765	0.888133	0.831375	0.911607
Linear Discriminant Analysis	0.864394	0.889053	0.855183	0.938984
LDA	**0.876136**	**0.904648**	**0.868460**	0.926217

[*Источник: составлено автором*]

В таблице 1 показано сравнение результатов точности всех алгоритмов классификации, разработанных для наборов данных, использованных в нашем исследовании. Можно заметить, что среди всех применяемых алгоритмов классификации классификация линейного дискриминантного анализа сравнительно более точна во всех других сценариях алгоритмов классификации. Кроме того, в этом эксперименте с использованием настройки гиперпараметров исследователи пытаются точно настроить алгоритм линейного дискриминантного анализа для повышения производительности с помощью поиска по сетке. Исследователи обнаружили, что добавление решателя параметров lsqr и усадки 0,81 в линейный дискриминантный анализ может повысить производительность. Результаты, полученные с помощью настройки гиперпараметров в линейном дискриминантном анализе, имеют лучшую точность по сравнению с результатами других исследований того же набора данных с использованием классификатора ансамблевого голосования [14].

Заключение

В этой статье несколько методов контролируемого обучения используются для решения наборов данных о занятости в иракских университетах - академических факторах и факторах трудоустройства, которые влияют на создание новых возможностей трудоустройства. Классификация была выполнена с использованием методов Logistic Regression, Support Vector Classifier, K Nearest Neighbors, Gaussian Naive Bayes, Decision Tree, Random Forest, Gradient Boosting, and Linear Discriminant Analysis classification techniques. Настройка гиперпараметров используется на этих контролируемые алгоритмы для лучших результатов. Для оценки результатов используются показатели эффективности. Linear Discriminant Analysis classification techniques с настройкой параметров показал лучшие результаты по сравнению с другими алгоритмами.

Литература

1. *Cheong Cheng Y, Ming Tam W.* Multi-models of quality in education // Qual Assur Education. 1997. No. 5 (1), pp. 22-31.
2. *Nwedu C.N.* Strategies and Opportunities for Student Recruitment and Retention in African Universities // Lessons from Western Universities. 2019. No. 1(2), pp. 1-9.
3. *Frølich N., Stensaker B.* Student recruitment strategies in higher education: Promoting excellence and diversity? International Journal of Educational Management, 2010. No.24 (4), pp. 359-370.
4. *Chen D.-G., Ware D.M.* A neural network model for forecasting fish stock recruitment // Canadian Journal of Fisheries and Aquatic Sciences. 1999. No. 56(12), pp. 2385-2396.
5. *Friedman N., Goldszmidt M.* Bayesian Network Classifiers // Machine Learning. 1997. No. 29 (2-3), pp. 131-163.
6. *Surarso B., Gernowo R.* Implementation of the K-Nearest Neighbor Method to determine the Classification of the Study Program Operational Budget in Higher Education // International Conference on Healthcare, Science and Technology (ICOHETECH), 2019, pp. 201-204.
7. *Sperandei S.* Understanding logistic regression analysis // Biochemia Medica. 2014. No. 24 (1), pp. 12-8.
8. *Chen R.-C., Dewi C., Huang S.-W. Caraka R.E.* Selecting critical features for data classification based on machine learning methods // Journal of Big Data. 2020. No. 77(52).
9. *Agarwal S., Jha B., Kumar T., Kumar M., Ranjan P.* Hybrid of Naive Bayes and Gaussian Naive Bayes for Classification: A Map Reduce Approach // International Journal of Innovative Technology and Exploring Engineering. 2019. No. 8(6S3), pp. 266-268.
10. *Rokach L., Maimon O.* Decision Trees. In book: The Data Mining and Knowledge Discovery Handbook. 2005, pp. 165-192.
11. *Biau G., Scornet E.* A Random Forest Guided Tour. 2015. No. 25(2), pp. 1-42.
12. *Natekin A., Knoll A.* Gradient boosting machines, a tutorial // Frontiers in Neurorobotics. 2013. No. 7.
13. *Gaber T., Tharwat A., Ibrahim A., Hassanien A.E.* Linear Discriminant Analysis: a detailed tutorial // IOS Press. 2017, pp. 1-23.
14. *Dutta S., Bandyopadhyay S.* Forecasting of Campus Placement for Students Using Ensemble Voting Classifier // Asian Journal of Research in Computer Science. 2020. No. 5, pp. 1-12.
15. *Hailpern S.M., Visintainer P.F.* Odds ratios and logistic regression: further examples of their use and interpretation // Stata Journal, StataCorp LP. 2003. No. 3(3), pp. 213-225.
16. *Cutler A., Stevens J.* Random forests for microarrays // Methods in enzymology. 2006. No. 411, pp. 422-32.
17. *Schratz P., Muenchow J., Iturritxa E., Richter J., Brenning A.* Performance evaluation and hyperparameter tuning of statistical and machine-learning models using spatial data. arxiv:1803.11266v1 [stat.ML]. 2018, pp. 1-46.
18. *Roshan B.* Andrew Ng-Machine learning assignments in Python. Kaggle. URL: https://www.kaggle.com/benroshan/factors-affecting-campus (date of access: 10.08.2022).

DEVELOPMENT OF A CLASSIFICATION ALGORITHM BASED ON AN ACADEMIC DATA MODEL USING HYPERPARAMETRIC TUNING FOR HIRING TEACHERS AT IRAQI UNIVERSITIES

Al-Dulaimi Omar Hatem Zaidan, *Voronezh State Technical University, Voronezh, Russia,* ***oomar1982@yandex.ru***

Abstract

In this paper, the researcher proposes a classification method for any employment opportunity in Iraqi universities using complete placement data by category for a dataset of employment in Iraqi educational institutions. The analysis of the scientific literature shows that this problem has not been sufficiently investigated and requires its solution. Developing a classification method based on genetic algorithms to get the best results to solve the problem of hiring new professors in Iraqi universities, getting accurate results and comparing several methods to solve this problem. The researcher tries to study learning classification algorithms with a teacher, such as logistic regression, Support Vector Classifier (SVC), Closest Neighbors KNN, Gaussian Naive Bayes algorithm, decision tree, random forest, gradient boosting and linear factor analysis (LDA). Hyperparameter optimization is also used for controlled algorithms to achieve better results. According to the proposed algorithms, the reliability and accuracy of the results obtained was ensured by a five-fold check. Results. Experimental results have shown that by adjusting hyperparameters in Linear Discriminant Analysis (LDA), it is possible to increase the accuracy of the results, as well as get a better result compared to other algorithms. The analysis criteria such as average accuracy, average score, G-Mean and Average ROC AUC were taken into account. It is according to the first three criteria that linear discriminant analysis demonstrates the highest accuracy of evaluation. It is established that among all the classification algorithms used, the classification of linear discriminant analysis is comparatively more accurate in all other scenarios of classification algorithms. It was found that adding the lsqr parameter solver and 0.81 shrinkage to linear discriminant analysis can improve performance. The article proves that the proposed classification algorithm based on the academic data model is able to solve the problem of hiring teachers at the university.

Keywords: *controlled classification, logistic regression, support vector classifier (SVC), K Nearest Neighbors (KNN), Gaussian Naive Bayes, decision tree, random forest, gradient enhancement, Linear Discriminant Analysis (LDA), hyperparameter optimization.*

References

1. Cheong Cheng Y, Ming Tam W. (1997). Multi-models of quality in education. *Qual Assur Education*. No. 5 (1), pp. 22-31.
2. Nwedu C.N. (2019). Strategies and Opportunities for Student Recruitment and Retention in African Universities. *Lessons from Western Universities*. No.1(2), pp. 1-9.
3. Frolich N., Stensaker B. (2010). Student recruitment strategies in higher education: Promoting excellence and diversity? *International Journal of Educational Management*. No. 24 (4), pp. 359-370.
4. Chen D.-G., Ware D.M. (1999). A neural network model for forecasting fish stock recruitment. *Canadian Journal of Fisheries and Aquatic Sciences*. No. 56(12), pp. 2385-2396.
5. Friedman N., Goldszmidt M. (1997). Bayesian Network Classifiers Machine Learning. No. 29 (2-3), pp. 131-163.
6. Surarso B., Gernowo R. (2019). Implementation of the K-Nearest Neighbor Method to determine the Classification of the Study Program Operational Budget in Higher Education. *International Conference on Healthcare, Science and Technology (ICOHETECH)*, pp. 201-204.
7. Sperandei S. (2014). Understanding logistic regression analysis. Biochemia Medica. No. 24 (1), pp. 12-8.
8. Chen R.-C., Dewi C., Huang S.-W., Caraka R.E. (2020). Selecting critical features for data classification based on machine learning methods. *Journal of Big Data*. No. 77(52).
9. Agarwal S., Jha B., Kumar T., Kumar M., Ranjan P. (2019). Hybrid of Naive Bayes and Gaussian Naive Bayes for Classification: A Map Reduce Approach. *International Journal of Innovative Technology and Exploring Engineering*. No. 8(6S3), pp. 266-268.
10. Rokach L., Maimon O. (2005). Decision Trees. In book: The Data Mining and Knowledge Discovery Handbook, pp. 165-192.
11. Biau G., Scornet E. (2015). A Random Forest Guided Tour. No. 25(2), pp. 1-42.
12. Natekin A., Knoll A. (2013). Gradient boosting machines, a tutorial. *Frontiers in Neurorobotics*. No. 7.
13. Gaber T., Tharwat A., Ibrahim A., Hassanien A.E. (2017). Linear Discriminant Analysis: a detailed tutorial. *IOS Press*, pp. 1-23.
14. Dutta S., Bandyopadhyay S. (2020). Forecasting of Campus Placement for Students Using Ensemble Voting Classifier. *Asian Journal of Research in Computer Science*. No. 5, pp. 1-12.
15. Hailpern S.M., Visintainer P.F. (2003). Odds ratios and logistic regression: further examples of their use and interpretation. *Stata Journal, StataCorp LP*. 2No. 3(3), pp. 213-225.
16. Cutler A., Stevens J. (2006). Random forests for microarrays. *Methods in enzymology*. No. 411, pp. 422-32.
17. Schratz P., Muenchow J., Iturritxa E., Richter J., Brenning A. (2018). Performance evaluation and hyperparameter tuning of statistical and machine-learning models using spatial data. arxiv:1803.11266v1 [stat.ML], pp. 1-46.
18. Roshan B. Andrew (2022). Ng-Machine learning assignments in Python. Kaggle. URL: https://www.kaggle.com/benroshan/factors-affecting-campus (date of access: 10.08.2022).