МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**
Кафедра математической кибернетики и компьютерных наук

СОЗДАНИЕ ПРИЛОЖЕНИЯ ДЛЯ ОЦЕНКИ ШАНСОВ АБИТУРИЕНТОВ НА ПОСТУПЛЕНИЕ В СГУ С ПОМОЩЬЮ ЯЗЫКА R.

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 411 группы
направления 02.03.02 — Фундаментальная информатика и информационные технологии
факультета КНиИТ
Трапезникова Андрияна Андреевича

Научный руководитель
Старший преподаватель _____ М. И. Сафрончик

Заведующий кафедрой
к. ф.-м. н., доцент _____ С. В. Миронов

Саратов 2021

ВВЕДЕНИЕ

Накопление огромного объема информации в различных сферах деятельности привело к пониманию важности задач, связанных с анализом этой информации, с целью получения новых знаний и выработки эффективной стратегии дальнейших действий.

Каждый абитуриент сталкивается с проблемой выбора университета для обучения. Он не только должен определиться с местом поступления, но еще и понять хватит ли ему для этого баллов. Чтобы это сделать абитуриенту необходимо найти информацию о ранее зачисленных студентах, сравнить их баллы со своими и исходя из этого принять решение. Данные часто бывают не слишком удобно структурированы для подобного анализа. В этом случае абитуриенту можно помочь, представив данные в наглядном виде, с возможность фильтровать их. В таком случае он сможет без труда оценить свои возможности зачисления в ВУЗ.

Целью данной работы является создание веб - приложения помогающего абитуриентам оценить свои шансы на поступление. Уже много лет механизм поступления в университеты не изменяется, что подтверждает актуальность работы. Для реализации приложения был выбран язык статистической обработки данных R и его пакет для разработки веб - приложений Shiny.

В работе ставятся следующие задачи:

1. Подготовить данные для анализа.
2. Создать удобный интерфейс, позволяющий абитуриентам проанализировать количественный состав и диапазон баллов зачисленных студентов за предыдущие годы, а также вероятность зачисления на то или иное направление.
3. Реализовать анализ баллов поступивших студентов и количества отчисленных, переведшихся и оставшихся студентов по различным с помощью средств языка R.
4. Связать интерфейс и серверную часть программы.
5. Установить готовый проект на сайт «shinyapps.io» для удобного доступа.

Дипломная работа состоит из двух глав:

1. Теоретические аспекты.
2. Практическая часть.

1 Основные теоретические выкладки

1.1 Что такое анализ данных

Не существует одного единственно верного определения термина анализа данных. Согласно области исследования данной работы более подходящем будет следующее определение: "Анализ данных — это область математики и информатики, занимающаяся построением и исследованием наиболее общих математических методов и вычислительных алгоритмов извлечения знаний из экспериментальных (в широком смысле) данных; процесс исследования, фильтрации, преобразования и моделирования данных с целью извлечения полезной информации и принятия решений". [1]

1.2 Область применения

Анализ данных широко применяется в огромном количестве сфер деятельности как научных, так и прикладных. Например, в ЦЕРНе физики используют этот инструмент для обработки информации о соударении частиц. Только за один день там накапливается объем информации, который сравним с информацией во всем интернете. Очевидно, что невозможно хранить такой объем данных, поэтому физикам приходиться отбирать события, которые являются наиболее интересными для изучения.

Одной из прикладных задач, которая решается с помощью анализа данных является задача медицинской диагностики. У каждого пациента берутся необходимые анализы, собирается информация о наличии симптомов, производятся измерения температуры тела, массы, давления, а затем эти данные анализируются и ставиться диагноз.

1.3 Подходы к решению задач анализа данных

Существуют два основных подхода к решению задач анализа данных. Первый подход, который в настоящем времени считается устаревшим — это самостоятельное создание математических подходов для решения определенной задачи. Например, для распознавания цифр можно самостоятельно подбирать функции реализующие соответствующие каждой конкретной цифре отображения. Однако, для решения этой задачи возможно использовать другой, более оптимальный подход — машинное обучение. Наглядным примером сравнения эффективности является использование машинного обучения в ЦЕРНе. До его внедрения определенные алгоритмы анализа данных отлавливали лишь

47 процентов нужных событий, а после внедрения машинного обучения этот показатель составил 77 процентов.

1.4 Машинное обучение

1.4.1 Постановка задачи машинного обучения по прецендентам

Задача заключается в построении функции, которая бы аппроксимировала неизвестную зависимость.

1.4.2 Как задаются объекты

При задании объектов чаще всего используется способ — признаковое описание. По сути признаки — это функции, которые какие-то значения, обычно числовые, ставят в соответствие объектам. Существуют также типы признаков, которые зависят от способа измерения над объектами:

- бинарный признак в данном случае это какой-то ответ "нет"или "да"про исследуемый нами объект;
- номинальный признак;
- упорядочено - порядковый признак — другими словами признак считается порядковым, если задано некое отношение порядка на множестве значений;
- количественный признак являющийся числовыми измерениями над объектами.

1.4.3 Как задаются ответы

Типы задач:

- Задачи классификации:
  - $Y = [-1, +1]$ — классификация на 2 класса;
  - $Y = [1, ..., M]$ — на M непересекающихся классов, например распознавание текста, написанного от руки;
  - $Y = [0, 1]^M$ — на M классов, которые могут пересекаться, например задача медицинской диагностики, когда один пациент может быть болен несколькими болезнями.
- Задачи восстановления регрессии
  - $Y = \mathbb{R}$ или $Y = \mathbb{R}^m$ — когда ответ является действительным числом. К эту типу задач относятся, например, задачи прогнозирования.
- Задачи ранжирования

– Y — конечное упорядоченное множество. Такие задачи решаются поисковыми системами при ранжировании поисковой выдачи.

1.4.4 Предсказательная модель

Предсказательная модель обычно выбирается из семейства параметрических функций $A = a(x) = g(x, \theta)|\theta \subset \Theta$, где $g : X * \Theta \to Y$ — фиксированная функция, Θ - множество допустимых значений параметра θ. Такое семейство должно содержать функцию, которая хорошо аппроксимирует неизвестную зависимость.

1.4.5 Проблема переобучения

Переобучение — это нежелательное явление, возникающее при решении задач обучения по прецедентам, когда вероятность ошибки обученного алгоритма на объектах тестовой выборки оказывается существенно выше, чем средняя ошибка на обучающей выборке. Переобучение возникает при использовании избыточно сложных моделей.

1.4.6 Эмпирические оценки обобщающей способности

Самый простой способ измерения эмпирического риска это деление на обучающую и контрольную выборки. Пусть X^l — это обучение, X^k — это контроль, тогда $(\mu, X^l, X^k) = Q(\mu(X^l), X^k) \to min$. Получается, что изначально имелось $(l+k)$ точек, затем они делятся на 2 части — одна для обучения, другая для проверки.

1.4.7 Кластеризация

Кластеризация — это обучение без учителя. В обучении без учителя выходные данные создаются самостоятельно, а алгоритм работает только на основании полученных входных данных и производных сигналов от них .

Формальное описание метода полной связи «complete linkage clustering». Этот метод будет использоваться для кластеризации студентов.

Исходно каждый элемент выборки считается отдельным кластером. После чего кластеры последовательно объединяются, пока все элементы не попадут в один кластер. На каждом шаге алгоритма объединяются два кластера, расстояние между которыми минимальное. Формализация понятия «минимальное расстояние» может зависеть от модификаций алгоритма, в методе

полной связи минимальное расстояние определяется как максимум из множества расстояний между элементом первого кластера и элементом второго кластера. То есть, расстояние $D(X, Y)$ между кластерами X и Y считается по формуле 1:

$$D(X, Y) = \max_{x \in X, y \in Y} d(x, y) \tag{1}$$

Где $d(x, y)$ — расстояние между $x \in X$ и $y \in Y$; X и Y — различные кластеры.

1.5 Метод главных компонент

Метод главных компонент — это линейный метод, который позволяет сократить размерность анализируемых данных. В практической части данный метод используется для сведения пяти переменных в две, что позволит расположить компоненты на плоскости и узнать как наблюдения располагаются в координатах главных компонент.

Линейная модель метода главных компонент представлена в формуле 2:

$$z_{ij} = \sum_{r=1}^{n} (a_{ir} * f_{ir}), j = 1, 2, ..., N, i = 1, 2, ..., n \tag{2}$$

Где z_{ij} — нормированное значение i-го признака на j-ом объекте исследования; a_{ir} — весовой коэффициент r-го фактора на i-ом признаке; f_{rj} — значение r-го фактора j-ом объекте исследования.

Идея метода главных компонент состоит в том, что после нахождения всех n главных компонент из их числа отбирают $m (m < n)$ наиболее весомых, иными словами вносящих наибольший вклад в объясняемую часть общей дисперсии. Для определения весовых компонент коэффициентов a_{ij} имеется система линейных уравнений [2]:

$$\begin{cases} A^T A = \Lambda \\ A A^T = \Sigma \end{cases}$$

1.6 Коэффициент корреляции Спирмена

Коэффициент корреляции Спирмена — представляет собой меру линейной связи случайных величин. Данная корреляция является ранговой, в силу того что вместо использования числовых значений для оценки силы связи,

применяются соответствующие им ранги. Коэффициент инвариантен по отношению к любому монотонному преобразованию шкалы измерения. В практической части данный коэффициент применяется для интерпретации смыслового значения главных компонент.

Пусть заданы выборки: $x = (x_1, ..., x_n)$ и $y = (y_1, ..., y_n)$.

Коэффициент корреляции Спирмена вычисляется по формуле 3:

$$\rho = 1 - \frac{6}{n(n-1)(n+1)} \sum_{i=1}^{n} (R_i - S_i)^2 \tag{3}$$

1.7 Реактивное программирование

Реактивное программирование — программирование с асинхронными потоками данных. Асинхронность в программировании — выполнение процесса в неблокирующем режиме системного вызова, что позволяет потоку программы продолжить обработку.

Реактивный подход повышает уровень абстракции кода и дает возможность сконцентрироваться на взаимосвязи событий, которые определяют бизнес-логику, вместо того, чтобы постоянно поддерживать код с большим количеством деталей реализации.

Преимущество более заметно в современных веб и мобильных приложениях, которые работают с большим количеством разнообразных UI-событий. Несколько лет назад все взаимодействие с веб-страницей сводилось к отправке больших форм на сервер и выполнении простого рендеринга в клиентской части. Сейчас приложения более сложны: изменение одного поля может повлечь за собой автоматическое сохранение данных на сервере, информация о новом «лайк» должна отправиться другим подключенным пользователям и так далее.

Реактивное программирование очень хорошо подходит для обработки большого количества разнообразных событий.

2 Основные выкладки практической части

2.1 Данные

Исходные данные имеют формат «csv» и содержат информацию о студентах, зачисленных на факультет КНиИТ за 2018, 2019 и 2020 год. Они содержат 5 столбцов:

1. year — год, в который студент был зачислен в ВУЗ;
2. specialty — направление подготовки, на которое был зачислен студент;
3. full name — имя фамилия и отчество студента
4. scores — сумма баллов за три предмета и дополнительные успехи;
5. phase — номер этапа в который студент был зачислен.

2.2 Интерфейс

За пользовательский интерфейс в приложении Shiny отвечает файл ui.R. Он обеспечивает интерактивность для приложения Shiny, принимая данные от пользователя и динамически отображая сгенерированный вывод на экране.

Главная панель включает в себя заголовок «SSU», и две вкладки для навигации: «Main» «Analitycs». Название заголовка устанавливается в функции titlePanel, параметром windowTitle. За создание вкладок навигации отвечает функция tabsetPanel с параметром типа type имеющим значение tabs и вспомогательные функции tabPanel с установленными параметрами Summary и Analytics — названия вкладок.

Основная вкладка «Main» имеет вкладку более низкого уровня «Summary», которая задается функцией tabPanel. «Summary» имеет боковую панель Data by Year с возможностью выбора даты для графика. Макет боковой панели задается функцией sidebarPanel. Возможность выбора года предоставляется функция selectInput, параметрами которой являются:

- checkYear — имя для входных данных для связи с серверной частью
- Select Year — название панели выбора
- choices — параметр, содержащий список значений, которые можно выбрать
- selected — выбранное значение по умолчанию

Связанная с боковой центральная панель Diagram budget places, на которой представлена диаграмма с информацией о количестве бюджетных мест на каждом направлении, а также список соответствий цветов диаграммы с на-

правлениями. Макет центральной панели задается функцией mainPanel, внутри которой макет вкладки — функция tabsetPanel, содержащая tabPanel("Diagram budget places plotOutput(outputId = "piePlot")) — функцию вывода диаграммы.

Ниже на вкладке «Summary» находится боковая панель Data Overview, которая служит для фильтрации списка студентов, имеющая возможность с помощью слайдера выбирать диапазон суммы баллов и выбор определенных лет. Сумма баллов установлена в диапозоне от 0 — минимально возможное значение, до 300 — максимально возможное значение. На этой панели возможен выбор не только одного конкретного года, но и нескольких. Это возможно благодаря дополнительному параметру multiple в значении TRUE. Ниже находится кнопка «Filter», при нажатии на которую данные обновляются согласно выбранным значениям. Кнопку создает функция actionButton.

Правее этой боковой панели располагается таблица списка студентов, с возможностью поиска по ключевым словам. За вывод данных отвечает функция dataTableOutput.

На другой основной вкладке «Analytics» боковая панель с возможностью выбора направлений, которые будет отображать график оценок студентов по направлениям, на котором представлена информация о минимальных, средних и максимальных значениях сумм вступительных баллов по направлениям. Ниже на этой вкладке располагается боковая панель с полем ввода, в которое пользователь вводит сумму своих баллов, а справа от боковой панели есть поля вывода, на которых отображается посчитанная вероятность поступления на каждое отдельное направление, основанная на 2020, 2019, 2018 годах.

Слева от названия каждого факультета находятся поля в виде маленьких квадратов с галочкой внутри. Галочка означает, что данное направление выбрано для отображения на графике. При необходимости можно убрать галочку, кликнув по ней. После этого график автоматически обновится уже без информации об этом направлении.

Возможность ввода суммы баллов с клавиатуры предоставляет функция textInput, с параметром txt1, для того, чтобы можно введенные данные можно было обработать в серверной части приложения, и с параметром «Score:», который представляет из себя имя поля ввода.

2.3 Серверная часть

За серверную часть в приложении Shiny отвечает файл server.R. Этот файл содержит последовательность шагов для преобразования введенных пользователем данных в желаемый вывод для отображения.

Перед тем как начать работу с данными, необходимо их считать. Для этого подойдет функция readRDS, на вход которой поступает путь к файлу формата «rds», относительно установленной директории. Затем считанные данные присваиваются переменной data.

Для вывода на экран диаграммы Diagram budget places используется функция renderPlot, которая обращается идентификационному коду вывода piePlot для связи с интерфейсом.

Сам график строится с помощью функции ggplot. Графики этой функции многослойные, то есть строятся они поэтапно, по слоям. Первый слой — функция aes, в качестве ее аргументов задаются переменные, которые будут отражаться на графике. За визуализацию отвечает следующий слой, а именно функция geombar, имеющая следующие параметры:

- stat = "identity— чтобы высота столбца соответствовала значению
- width = 1 — ширина
- color = "black— цвет границ диаграммы
- size = 2 — размер

Затем идет themevoid() — она убирает оси и метки.

Функция theme() - оформление графика.

По умолчанию geombar диаграмма имеет столбчатый вид. Чтобы она отображалась в форме круга используются полярные координаты.

Таблица отображающая список студентов может реагировать на изменение входных данных. Например фильтрацию или поиск по ключевым словам. Для этого используется data reactive. В пользовательском интерфейсе фильтрация данных происходит с помощью кнопки Filter. Чтобы реакция на нажатие отражалась в серверной части используется input$actionDT. Для фильтрации в диапазоне значений задаются переменные minScore и maxScore. Затем они сравниваются с фактическими данными при помощи функции filter, в которую в качестве аргументов передаются булевы значения: scores > minScore и scores < maxScore. Функция select отвечает за выбранные столбы данных.

Диаграмма для анализа баллов имеет вид горизонтальный гистограм-

мы. Это достигается за счет использования функции coord_flip(). Аргументы поступающие на вход theme() основные заголовки и метки оси.

Реализация главной функции приложения начинается с конвертации данных в формат data.table. Это делается для расширения синтаксиса data.frame. Затем создаётся фрейм, в который попадут только те данные с минимальной суммой баллов за каждый фиксированный год и направление. Он будет иметь только три столбца: год, направление, баллы. Далее введенное значение сравнивается поочередно с баллами и если оно больше, то прибавляется единица, иначе ноль.

2.3.1 Оценка сложности направления при помощи средств анализа языка R

В первую очередь необходимо стереть прежние переменные, подключить нужные библиотеки.

Далее загружаются исследуемые данные, проверяются пропущенные значения, удаляется переменная phase, так как в дальнейшем она не будет использоваться в ходе анализа:

Подключается библиотека dummis, содержащая команды позволяющие создать из номинально качественных переменных, одиночно-бинарные переменные. Есть три градации:

- df_statement — студент остался на направлении: «1», студент ушел с направления: «0»;
- df_zabrali — студент не забрал документы: «0», забрал: «1»;
- df_perevel — студент перевелся: «1», не перевелся: «0».

Строится гистограмма распределения баллов по всем студентам при использовании библиотеки ggplot2.

Следующим шагом центрируются и шкалируются переменные. От каждого элемента выборки отнимается медиана и производится деление на межквартильный размах. Проводится для того, чтобы метод главных компонент одинаково учитывался влияние каждой переменной. Таким образом каждому столбцу соответствует свое значение медианы и свой межквартильный размах [3].

Далее происходит стандартизация сгрупированных данных и расчет методом главных компонент. Затем происходит интерпретация их смыслового значения.

Заключительным этапом является кластерный анализ.

Первый кластер — направления «CS» и «SE». На этих направлениях тяжелее всего учиться. Студенты чаще забирают документы по сравнению с другими направлениями. На «SE» учатся студенты с самым высоким средним баллом ЕГЭ. Второй кластер — SAIS, FIIT, ICE, PE. В него попали направления со сравнительно небольшим оттоком студентов. Третий кластер — включает в себя только направление SAC. Туда идут меньше всего студентов и с него практически не уходят. Среднему балл ЕГЭ у студентов, обучающихся SAC — практически самый низкий, ниже только у тех, кто на PE.

ЗАКЛЮЧЕНИЕ

В результате выпускной квалификационной работы было реализовано приложение, позволяющее абитуриентам оценить свои шансы на поступление, сравнить свои баллы с баллами студентов, поступавших в прошедшие годы и оценить сложность того или иного направления. Приложение выполнено на базе языка R, с использованием библиотеки Shiny. В ходе выполнения был создан простой и понятный интерфейс, предоставляющий возможность ввести свои баллы и узнать шанс на поступление в процентах на каждое из существующих направлений факультета КНиИТ, просмотреть таблицу с информацией о студентах, поступавших в предыдущие годы и возможностью поиска и фильтрации, кроме того построен график который отражает рейтинг переводов студентов по направлениям. Также были построены две гистограммы, первая показывает количество бюджетных мест по направлениям, вторая баллы в удобной для анализа форме.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1 *Барсегян, М. А.* Анализ данных и процессов / М. А. Барсегян, М. С. Куприянов, И. И. Холод, М. Д. Тесс, С. И. Елизаров. — Санкт-Петербург: БВХ-Петербург, 2009.

2 *Бирюкова, Л. Г.* Теория вероятностей и математическая статистика / Л. Г. Бирюкова, Г. И. Бобрик, Р. В. Сагитов, Е. В. Швед, В. И. Матвеев. — Москва: ИНФРА-М, 2020.

3 *Джеймс, Г.* Введение в статистическое обучение с примерами на языка R / Г. Джеймс, Д. Уиттон, Т. Хасти, Р. Типширани. — Москва: ДМК Пресс, 2016.