Министерство науки и высшего образования Российской Федерации

Томский государственный университет
систем управления и радиоэлектроники

А. А. Захарова

АНАЛИЗ ДАННЫХ В EXCEL И CALC

Учебно-методическое пособие по выполнению лабораторных работ и самостоятельной работе по дисциплине «Анализ больших данных»
для студентов технических направлений подготовки

Томск 2024

УДК 004.48
ББК 16.333я22
З-38

Рецензент:
Мицель А.А., профессор кафедры АСУ, докт. техн. наук

З-38 Захарова, Александра Александровна
Анализ данных в Excel и Calc: учебно-методическое пособие по выполнению лабораторных работ и самостоятельной работе по дисциплине «Анализ больших данных» для студентов технических направлений подготовки / А. А. Захарова. – Томск: Томск. гос. ун-т систем упр. и радиоэлектроники, 2024. – 61 с.

Пособие содержит задания и требования по выполнению лабораторных работ, самостоятельной работе студентов по дисциплине «Анализ больших данных». Лабораторные работы направлены на закрепление теоретических знаний, а также формирование навыков применения методов машинного обучения для анализа больших наборов экономических данных, используя возможности табличных процессоров Excel и Calc.

Одобрено на заседании каф. АСУ протокол № 11 от 23.11.2023

УДК 004.48
ББК 16.333я22

Оглавление

ВВЕДЕНИЕ

Целью курса является освоение основных концепций и методов аналитики данных, особенностей областей применения и использования их как готового инструмента принятия решений при работе со структурированными и неструктурированными данными больших объемов.

Лабораторные работы направлены на закрепление теоретических знаний, а также формирование навыков применения методов машинного обучения для анализа больших наборов экономических данных, используя возможности табличных процессоров Excel и Calc.

Самостоятельная работа связана с изучением теоретического материала, выполнением лабораторных работ, написанием отчетов, а также подготовкой к устной защите по контрольным вопросам

В пособии предложены задания по следующие темам:

- шкалы измерений;
- регрессионные модели;
- наивный байесовский классификатор;
- кластерный анализ;
- сетевые модели;
- ансамбли моделей (бэггинг и бустинг).

Каждая лабораторная работа содержит подробный пример ее выполнения, при этом использованы задачи машинного обучения из [4].

1 ИНДЕКСА ЭКОНОМИКИ ЗНАНИЙ (ШКАЛЫ ИЗМЕРЕНИЙ)

1.1. Теоретические сведения

Индекс экономики знаний (Knowledge Economy Index, **KEI**) был разработан в 2004 году исследовательской группой Всемирного банка (The World Bank) для оценки способности стран создавать, принимать и распространять знания [1].

Индекс отражает состояние основных слагаемых экономики знаний: экономических стимулов и институционального режима, инновационной активности страны, уровня образования населения и развития ИКТ. Индекс используется для выявления «уязвимых мест» в научно-технической и инновационной политике, а также для измерения готовности страны перейти к экономике, основанной на знаниях.

В основе расчета Индекса лежит предложенная Всемирным банком «Методология оценки знаний» (The Knowledge Assessment Methodology – KAM), которая включает комплекс из 109 структурных и качественных показателей, объединенных в четыре основные группы: образования, инноваций, ИКТ, экономического и институционального режима. Индекс может рассчитываться по набору из 12 основных показателей, часть которых является составными (объединяющими в себе несколько других). Структура подындексов и основных показателей представлена в таблице 1.1. Веса подындексов одинаковые, веса показателей внутри каждого из подындексов одинаковые.

Также на основе подындексов 2, 3, 4 (образования, инноваций, ИКТ) в соответствии с Методологией оценки знаний KAM рассчитывается еще один индекс – **Индекс знаний (KnowledgeIndex – KI).**

Таблица 1.1 – Структура индекса экономики знаний KEI

Наименование индекса / подындекса / показателя	**Единица измерения**
Индекс экономики знаний (Knowledge Economy Index, KEI)	**Пункт**
Подындекс 1. Экономические стимулы и институциональная среда (Economic Incentive and Institutional Regime)	**Пункт**
Уровень тарифных и нетарифных импортных барьеров (Tariff & Nontariff Barriers)	Балл
Интегральный показатель качества системы регулирования рынков (Regulatory Quality)	Пункт
Интегральный показатель соблюдения правовых норм в стране (Rule of Law)	Пункт
Подындекс 2. Инновационный потенциал и технологическое развитие (Innovation and Technological Adoption)	**Пункт**
Сумма выплат и доходов по роялти и лицензионным платежам на душу населения (Royalty Payments and receipts(US$/pop.))	Доллар
Среднегодовое количество патентов, выданных российским заявителям Американским бюро патентов и торговых знаков, на 1 млн человек населения (Patents Granted by USPTO / Mil. People)	Единица
Число публикаций в научных журналах в области естественно-научных и технических дисциплин на 1 млн человек населения (S&E Journal Articles / Mil. People)	Единица

Окончание табл. 1.1

Подындекс 3. Система образования и подготовки кадров (Education and Training)	**Пункт**
Средняя продолжительность обучения населения в возрасте 15 лет и старше (Average Years of Schooling)	Год
Удельный вес обучающихся по программам основного и среднего общего образования, среднего профессионального (программам подготовки квалифицированных рабочих, служащих) образования в общей численности населения в возрасте 11-17 лет (Gross Secondary Enrollment rate)	Процент
Удельный вес учащихся по программам среднего профессионального (программам подготовки среднего звена) и высшего образования в общей численности населения в возрасте 18-22 лет (Gross Tertiary Enrollment rate)	Процент
Подындекс 4. ИКТ-инфраструктура (Information and Communications Technologies (ICT) Infrastructure)	**Пункт**
Совокупное число подключенных терминалов подвижной радиотелефонной связи и телефонных аппаратов на 1000 человек населения (Total Telephones per 1000 People)	Единица
Число персональных компьютеров на 100 человек населения (Computers per 1000 People)	Единица
Число пользователей Интернета на 1000 человек (Internet Users per 1000 People)	Единица

В рамках подындексов исходные показатели нормируются таким образом, что максимально возможное значение каждого подындекса (и, следовательно, интегрального индекса) равно 10 пунктам, минимально возможное — 0.

Для этого выполняются следующие шаги:

1. По каждому из m показателей ранжируются N_c обследуемых стран и получают первоначальный ранг ($u = (1..n)$), при этом ранг $u = 1$ получает страна с наилучшим значением показателя; страны с одинаковыми значениями показателя получают одинаковый ранг.

2. Для каждой страны по каждому показателю подсчитывается число стран, имеющих более высокий ранг N_h.

3. Для каждой страны рассчитывается нормированное значение показателя по формуле (1.1)

$$Norm(u) = 10 * \left(1 - \frac{N_h}{N_c}\right). \quad (1.1)$$

4. После расчета нормированных значений для каждого из показателей, рассчитываются значения подындексов как среднее арифметическое нормированных значений показателей, входящих в данную группу.

5. Рассчитывается индекс экономики знаний KEI как среднее арифметическое значений четырех подындексов.

1.2. Лабораторная работа. Задание и порядок выполнения работы

Цель работы: научиться рассчитывать индекс экономики знаний (по методологии КАМ Всемирного банка) на основе основных показателей и ограниченного количества стран.

Задачи:

1. Изучить методику Всемирного банка по расчету индекса экономики знаний и индекса знаний.

2. Рассчитать индексы экономики знаний по набору из 12 основных показателей для 15 стран, используя инструменты Excel (Calc).

Задание: используя рассчитать Индекс экономики знаний для 15 стран на основе значений показателей, представленных в таблице А.1 Приложения А.

Номера стран выбрать по вариантам (таблица 1.2). Номер варианта – по номеру студента в журнале преподавателя.

Таблица 1.2 – Варианты заданий

№ варианта	Номера стран	№ варианта	Номера стран	№ варианта	Номера стран
1	1-15	5	5-19	9	3-12, 16-20
2	2-16	6	6-20	10	2-11, 15-19
3	3-17	7	1-5, 10-19	11	1-8, 14-20
4	4-18	8	1,2, 5-14, 18-20	12	2-9, 13-19

Порядок выполнения работы.

Для выполнения работы выполните следующие этапы.

1. Средствами Excel (Calc) проранжировать страны по каждому показателю. Для этого нужно расположить значения показателей в порядке убывания. Максимальному значению присваивается первый ранг, следующему значению – второй и т.п. Если имеются два одинаковых значения, то им присваивается одинаковый ранг (причем наивысший). Например, если третья и четвертая позиция имеют одинаковое значение, то им присваивается ранг 3, а пятой позиции – ранг 5. Следует использовать функцию РАНГ.РВ. Результаты ранжирования представить в виде таблицы 1.3.

Таблица 1.3 – Результаты ранжировки

	Страны / Ранг страны по показателям									
Показатели	**1**	**2**	**….**			**8**			**….**	**15**
Показатель 1										
…….										
Показатель 12										

2. Средствами Excel (Calc) для каждой страны по каждому показателю подсчитать количество стран N_h, превосходящих данную страну по данному показателю, то есть имеющей более высокий ранг (меньший по значению). Следует использовать функцию СЧЁТЕСЛИ. Результаты представить в виде таблицы 1.4.

Таблица 1.4 – Количество стран, превосходящих исследуемую страну по значению показателя

	Страны / Количество стран, превосходящих исследуемую страну по значению показателя									
Показатели	**1**	**2**	**….**			**8**			**….**	**15**
Показатель 1										
…….										
Показатель 12										

3. Средствами Excel (Calc) для каждой страны и показателя рассчитать нормированное значение показателя по формуле (1.1). Результаты расчетов представить в виде таблицы 1.5.

Таблица 1.5 – Нормированные значения показателей

	Страны / Нормированные значения показателей									
Показатели	**1**	**2**	**…..**			**8**			**….**	**15**
Показатель 1										
…….										
Показатель 12										

4. Рассчитать значения подындксов по группам показателей для всех стран. Следует использовать функцию СРЗНАЧ. Результаты представить в таблице 1.6.

Таблица 1.6 – Значения индекса экономики знаний

	Страны / Значения Индекса/подындекса									
Индекс/подындекс	**1**	**2**	**…..**			**8**			**….**	**15**
Индекс экономики знаний										
Подындекс 1. Экономические стимулы и институциональная среда										
Подындекс 2. Инновационный потенциал и технологическое развитие										
Подындекс 3. Система образования и подготовки кадров										
Подындекс 4. ИКТ-инфраструктура										

5. Средствами Excel рассчитать значения Индекса экономики знаний по всем странам. Результаты представить в таблице 1.6.

6. Сделать выводы по результатам расчетов.

1.3 Контрольные вопросы

1. Понятие Индекса экономики знаний KEI, назначение, разработчик.
2. Структура Индекса экономики знаний KEI (подындексы и набор основных показателей, общее количество показателей)
3. Методика расчета Индекса экономики знаний
4. Что понимается под ранжированием?
5. Для чего осуществляется нормирование?
6. Метод нормировки, применяемый для расчета KEI.
7. Какая шкалы измерений используются в показателях, промежуточных расчетах и самом индексе экономики знаний?

2 РЕГРЕССИОННЫЕ МОДЕЛИ

2.1 Теоретические сведения

Модель множественной линейной регрессии (или коротко - множественная линейная регрессия) предназначена для проверки и изучения связи (объяснения поведения) между одной зависимой переменной (эндогенной) и несколькими независимыми (экзогенными) переменными. Предполагается, что такая связь теоретически может быть описана (специфицирована) линейной зависимостью (функцией) вида:

$$Y = \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + U$$

где Y - зависимая (объясняемая, эндогенная) переменная – регрессанд, U - случайная составляющая модели, x_j - независимые (объясняющие, экзогенные) переменные - регрессоры.

Будем рассматривать классическую модель множественной линейной регрессии. Это означает, что независимые переменные (регрессоры) предполагаются неслучайными (детерминированными) величинами.

Коэффициент детерминации.

Коэффициент детерминации – это доля объясненной дисперсии в общей, в случае линейной регрессии с константой определяется по формуле:

$$R^2 = \frac{ESS}{TSS},$$

где ESS – объясненная сумма квадратов отклонений;

TSS – общая дисперсия.

Коэффициент детерминации для модели с константой принимает значения от 0 до 1. Чем ближе значение коэффициента к 1, тем сильнее зависимость. При оценке регрессионных моделей это интерпретируется как соответствие модели данным. Для приемлемых моделей предполагается, что коэффициент детерминации должен быть хотя бы не меньше 50% (в этом случае коэффициент множественной корреляции превышает по модулю 70%). Модели с коэффициентом детерминации выше 80% можно признать достаточно хорошими (коэффициент корреляции превышает 90%). Равенство коэффициента детерминации единице означает, что объясняемая переменная в точности описывается рассматриваемой моделью [2, 3].

Статистика Фишера

Статистика Фишера используется для проверки гипотезы о связи между объясняемым рядом и регрессорами. Используется нулевая гипотеза: коэффициенты при всех регрессорах равны нулю.

Статистическая значимость уравнения множественной регрессии в целом оценивается с помощью общего F-критерия Фишера/

Формула для расчёта статистики Фишера для модели с константой:

$$F = (R^2/(k-1))/((1-R^2)/(N-k)), \quad (2.1)$$

формула для расчёта статистики Фишера для модели без константы:

$$F = (R^2/k)/((1-R^2)/(N-k)), \quad (2.2)$$

где R^2 – коэффициент детерминации;

k – количество факторов, включенных в модель (включая константу);

N – количество наблюдений.

Для нецентрированного коэффициента детерминации может быть рассчитана соответствующая статистика Фишера.

Вероятность статистики Фишера.

Статистика Фишера имеет распределение Фишера:

- для модели с константой: *F(k - 1, N - k);*
- для модели без константы*: F(k, N - k).*

Нулевая гипотеза о равенстве нулю коэффициентов при всех регрессорах отклоняется, если вероятность меньше, чем уровень значимости. Рассматривают один из стандартных уровней значимости 0,1; 0,05 или 0,01.

t-статистика.

Оценка значимости коэффициентов регрессии (кроме свободного члена) осуществляется сравнением t-статистики (рассчитывается по формуле 2.3) с табличным значением t-статистики Стьюдента.

$$t_j = a_j/(SE\sqrt{(b_{ij}\,)}), \tag{2.3}$$

где $(b_{ij}\,)$– диагональный элемент матрицы $(X^tX)^{-1}$;

SE – среднеквадратическое отклонение ошибки.

Если значение превосходит табличное значение t-статистики Стьюдента, то j-й коэффициент считается значимым, в противном случае фактор, соответствующий данному коэффициенту, следует исключить из модели.

Линейная регрессия используется для прогнозирования значений зависимой переменной, при этом выходные значения являются непрерывными и неограниченными по области определения. А в задачах классификации необходимо разделить на основе выходного значения объекты по классам, т.е. предсказать дискретное значение. В линейной регрессии можно ввести пороговые значения для отнесения объектов к тому или иному классу. Логистическая регрессия применяется для прогнозирования вероятности возникновения некоторого события по значениям множества признаков. Метод логистической регрессии основан на применении логистической или сигмоидной функции. В случае бинарной классификации, например, результат работы метода может интерпретироваться как вероятность отнесения объекта к положительному (целевому) классу, для четкой классификации выбирают некоторое пороговое значение вероятности, например, 0,5.

Оценка качества модели

При оценке точности модели мы сопоставляем прогнозные результаты модели с фактическими данными. Возможны 4 ситуации:

– истинно положительные (ИП), верно классифицированные положительные примеры (больному человеку поставлен диагноз);

– истинно отрицательные (ИО), верно классифицированные отрицательные примеры (здоровый человек отнесен к классу здоровых);

– ложно положительные (ЛП) отрицательные примеры, классифицированные как положительные (ошибка II рода). Это ложное обнаружение, т.к. при отсутствии события ошибочно выносится решение о его присутствии (здоровый человек отнесен к классу больных);

– ложно отрицательные (ЛО) положительные примеры, классифицированы как отрицательные (ошибки 2 рода). Это так называемый «ложный пропуск» — когда интересующее нас событие ошибочно не обнаруживается (больной классифицирован как здоровый).

Что является положительным событием, а что – отрицательным, зависит от конкретной задачи. Например, если мы прогнозируем вероятность наличия заболевания (как в примере выше), то положительным исходом будет класс «Больной пациент», отрицательным – «Здоровый пациент». И наоборот, если мы хотим определить вероятность того, что человек здоров, то положительным исходом будет класс «Здоровый пациент».

При анализе чаще оперируют не абсолютными показателями, а относительными – долями (rates), возможно выраженными в процентах.

Основные метрики.

Доля ложно положительных примеров (ДЛП) (False Positives Rate) рассчитывается по формуле:

$$\text{ДЛП} = \text{ЛП}/((\text{ИО} + \text{ЛП})).$$

Чувствительность (sensitivity), также известная, как **полнота** (recall), вычисляется по следующей формуле:

$$\text{Чувствительность} = \text{ИП}/((\text{ИП} + \text{ЛО})).$$

То есть чувствительность – это и есть доля истинно положительных случаев от общего числа положительных. Поскольку формула не учитывает ЛП и ИО, чувствительность может дать нам смещенную оценку, особенно в случае несбалансированных классов.

Специфичность (specificity) вычисляется по следующей формуле:

$$\text{Специфичность} = \text{ИО}/((\text{ИО} + \text{ЛП})).$$

То есть специфичность – это доля истинно отрицательных случаев, которые были правильно идентифицированы моделью. Поскольку формула не учитывает ЛО и ИП, специфичность может дать нам смещенную оценку, особенно в случае несбалансированных классов (заметим, что $\text{ДЛП} = 1 - \text{Специфичность}$).

Модель с высокой чувствительностью часто дает истинный результат при наличии положительного исхода (обнаруживает положительные примеры). Наоборот, модель с высокой специфичностью чаще дает истинный результат при наличии отрицательного исхода (обнаруживает отрицательные примеры).

Точность (accuracy) вычисляется в общем случае по следующей формуле:

$$\text{Точность} = (\text{ИП} + \text{ИО})/(\text{ИП} + \text{ИО} + \text{ЛП} + \text{ЛО}).$$

То есть точность – это доля всех правильно классифицированных примеров (по всем классам).

Но в случае, если положительный класс – это событие относительно редкое, то больший вклад в показатель Точности будет вносить показатель ИО, хотя для прогнозирования более важным является класс ИП. Поэтому в случае асимметрии классов, можно использовать метрики, которые не учитывают ИО и ориентируются на ИП. Если рассмотрим долю правильно предсказанных положительных объектов среди всех объектов, предсказанных положительным классом, то мы получим метрику, которая называется точностью (precision).

$$\text{Точность}\ (P) = \text{ИП}/(\text{ИП} + \text{ЛП}).$$

ROC-анализ.

ROC-кривая (Receiver Operator Characteristic) — кривая, которая наиболее часто используется для представления результатов бинарной классификации в машинном обучении. Название пришло из систем обработки сигналов. ROC-кривая показывает зависимость количества верно классифицированных положительных примеров от количества неверно классифицированных отрицательных примеров. ROC-кривая получается следующим образом.

Для каждого значения порога отсечения, с некоторым шагом (например, 0,01) рассчитываются значения чувствительности и специфичности. В качестве альтернативы порогом может являться каждое последующее значение примера в выборке.

Строится график зависимости: по оси Y откладывается чувствительность, по оси X — – доля ложно положительных примеров (ДЛП).

Для идеального классификатора график ROC-кривой проходит через верхний левый угол, где доля истинно положительных случаев составляет 100% или 1,0 (идеальная чувствительность), а доля ложно положительных примеров равна нулю. Поэтому чем ближе кривая к верхнему левому углу, тем выше предсказательная способность модели. Наоборот,

чем меньше изгиб кривой и чем ближе она расположена к диагональной прямой, тем менее эффективна модель.

2.2 Лабораторная работа. Задание и порядок выполнения работы

Цель лабораторной работы: научится строить регрессионную модель для классификации покупателей и оценивать её качество.

Задание. Используя Excel или Calc постройте модель множественной линейной регрессии для классификации покупателей, осуществите классификацию на тестовом наборе данных, оцените качество модели.

Далее будет описан порядок выполнения работы, пример выполнения приводится на основе данных из [4].

Исходные данные находятся в файле на листе «Training Data» (файлы выдаются по вариантам).

В файле представлены данные покупателях магазина и сведения о их покупках в течение года, например: пол владельца учетной записи (покупателя): мужской, женский, не указан; адрес (частный дом, квартира или абонентский ящик; недавно заказывал книги о кулинарии; недавно заказывал вино; недавно заказывал одежду определенного типа и другие.

В столбце S файла содержатся сведения о положительном событии (событие, прогнозирование которого нужно осуществить в лабораторной работе). Например, в примере [4] таким событием является факт ожидания ребенка в семье покупателя). Всего имеется выборка по 1000 покупателям (500 положительных примеров 500 отрицательных) (рис.2.1. [4]).

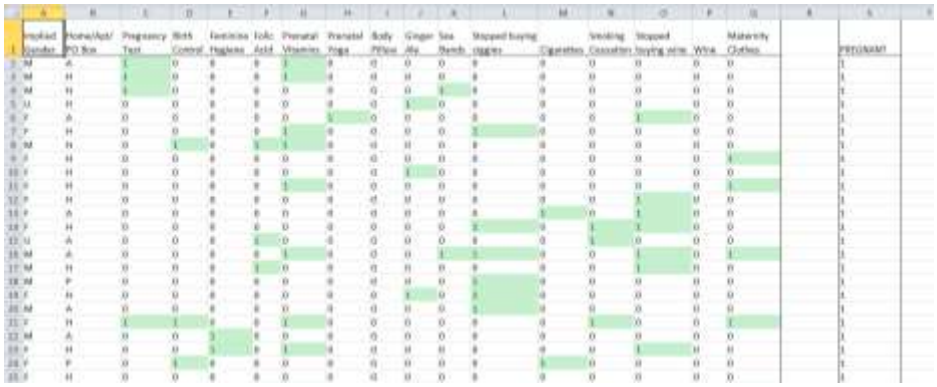


Рисунок 2.1 – Данные о покупателях

Необходимо создать регрессионную модель для предсказания факта положительного события на основании имеющихся данных о покупках и проверить модель на тестовых данных (лист TestSet).

Для выполнения работы выполните следующие этапы.

1. На листе «Training Data w Dummy Vars» переведите категорийные данные в числовые. Для этого введите «фиктивные переменные». Информацию о поле представьте в виде столбцов Male, Female (вариант «пол не указан» будет выражаться в значениях 0 в обоих столбцах). Информацию об адресе представьте в виде двух столбцов Home и Apt (вариант «а/я» будет выражаться через 0 в этих столбцах). Используйте функцию ЕСЛИ (рис.2.2).

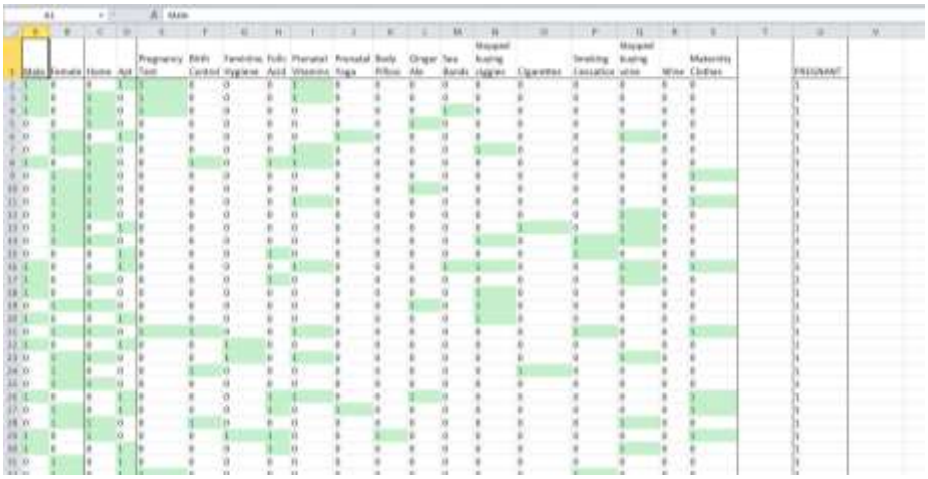


Рисунок 2.2 – Данные о покупателях после преобразования категорийных данных в числовые

2. Создайте линейную регрессионную модель.

2.1. На листе «Linear Model» начиная со столбца B и строки 8 вставьте данные с листа «Training Data w Dummy Vars». Вставьте строку с названиями столбцов в строку 1. Столбец U назовите «Intercept» – в нем будет отражаться свободный член линейной регрессии, заполните его с 8 по 1007 строку единицами. Строку 2 озаглавьте «Model Coefficients» и запишите стартовые значения равные 1 в каждом столбце с данными (с B по U).

2.2. Проведите обучение модели

2.2.1. В столбец W добавьте (ячейка W7) добавьте название «Linear Combination (Prediction)», а ниже для каждого покупателя поместите линейную комбинацию коэффициентов и данных покупателей (свободный член включен) (используйте функцию СУММПРОИЗВ коэффициентов на значения переменных). Отформатируйте столбец до двух знаков после запятой. Проанализируйте полученные расчетные данные по прогнозируемому признаку.

2.2.2. Осуществите расчет отклонений в столбце X (название в ячейке X7 - «Squared Error»). Значения в столбце X – это квадрат отклонения фактического значения в столбце V от прогнозного значения в столбце W.

2.2.3. В ячейке W1 впишите название «Sum Squared Error», а в ячейке X1 найдите сумму квадратов отклонений.

2.2.4. Осуществите настройку модели оптимизации в Поиске решений:

– необходимо минимизировать целевую функцию (сумма квадратов отклонений);

– изменять нужно значения переменных (коэффициентов модели) в ячейках от B2 до U2;

– установите ограничения на значения коэффициентов модели «>=-1» И «<=1»;

– используйте эволюционный алгоритм, так как целевая функция нелинейна.

2.2.5. Представьте результаты Поиска решений в отчете – значения полученных коэффициентов, свободного члена и целевой функции. Сделайте вывод.

2.3. Проверьте полученную регрессионную модель на адекватность.

2.3.1. Рассчитайте коэффициент детерминации (R^2). Для этого в ячейке W2 введите название «Total Sum of Squares», а в ячейке X2 рассчитайте дисперсию значений зависимой переменной (функция КВАДРОТКЛ). В ячейке W3 введите название «Explained Sum of Squares» и в ячейке X3 рассчитайте объясненную дисперсию (X2-X1). В ячейке W4 введите название «R squared » и в ячейке X3 рассчитайте коэффициент детерминации (R2) (X3/X2). В отчете приведите результаты расчета и сделайте вывод.

2.3.2. Проверьте статистическую значимость уравнения множественной регрессии в целом с помощью общего F-критерия Фишера. В ячейках Y1:Z5 введите названия показателей

и рассчитайте значения, представленные в таблице 2.1. Приведете результаты расчета, сделайте вывод.

Таблица 2.1 – Проверка по F-критерию

Столбец Y	**Пояснения**	**Столбец Z (подсказка функции Excel)**
Observation Count	Количество наблюдений	СЧЕТ
Model Coefficient Count	Количество факторов в уравнении регрессии	СЧЕТ
Degrees of Freedom	Степень свободы	Формула: N-k
F statistic	Значение F-статистики	Формула (2.1)
F Test P Value	P-значение F-статистики	FРАСП

2.3.3. Проверьте значимость отдельных переменных, проведите t-тест.

2.3.3.1. Рассчитайте среднеквадратическое отклонение прогноза как квадратный корень из суммы квадрата отклонений (X1), разделенный на количество степеней свободы (Z3).

2.3.3.2. Рассчитайте среднеквадратические отклонения коэффициентов. Для этого:

2.3.3.2.1. Создайте лист «ModelCoefficientStdError». Для формирования матрицы суммы квадратов и векторных произведений (СКВП) умножьте транспонированную матрицу плана на саму себя. Для этого вставьте строку заголовков обучающих данных на листе «ModelCoefficientStdError» в ячейки B1:U1 и транспонированную строку заголовков в A2:A21 (вместе с заголовком Intercept). Выделите B2:U21 и далее с помощью функции МУМНОЖ и ТРАНС, умножьте транспонированную матрицу 'Linear Model'!B8:U1007 на её саму же. Не забудьте использовать формулу для массива. Пример результата расчетов представлен на рис.2.3.

Проанализируйте значения в матрице СКВП. По диагонали считаются совпадения переменной самой с собой (то есть значения равны сумме по каждому столбцу матрицы плана), свободный член равен 1000. В ячейках, не входящих в диагональ проанализируйте число совпадений признаков. Матрица СКВП дает нам представление о величине переменных и о том, насколько они пересекаются и соотносятся между собой.

2.3.3.2.2. Преобразуйте матрицу СКВП в обратную. Для этого вставьте заголовки ниже матрицы в ячейки B24:U24 и A25:A44. Выделите диапазон B24:U44 и с помощью функции МОБР как формулы массива получите обратную матрицу из матрицы СКВП.

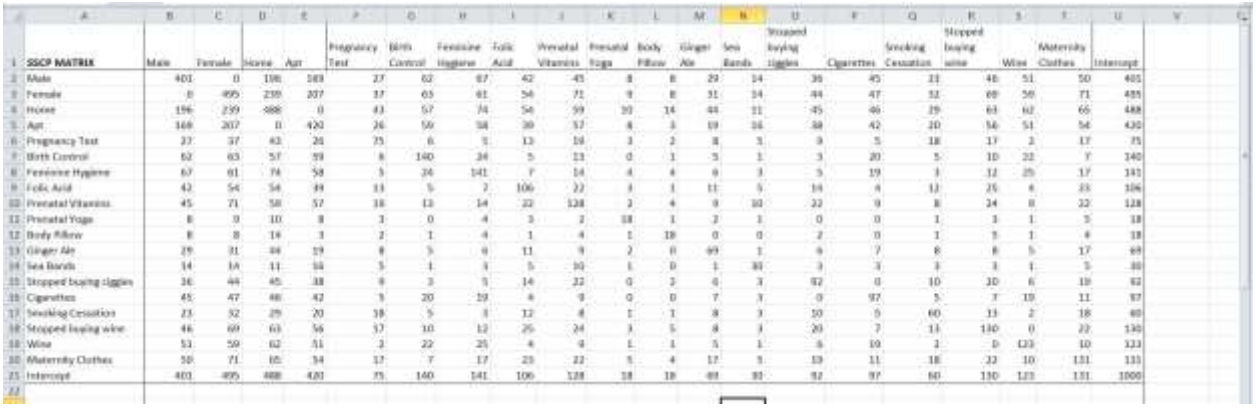


Рисунок 2.3 – Матрица СКВП

2.3.3.2.3. Рассчитайте среднеквадратическое отклонение коэффициента как произведение среднеквадратического значения прогноза модели (ячейка X5 на листе «Linear

Model») на квадратный корень из соответствующего значения диагонального элемента обратной матрицы СКВП.

Для облегчения расчётов пронумеруйте переменные, начиная с 1 в B46 до 20 в U46. Затем используйте формулу ИНДЕКС, чтобы найти соответствующее значение диагонального элемента.

Например, ИНДЕКС(ModelCoefficientStdError!B25:B44;ModelCoefficientStdError!B46) выдает значение пересечения строки Male со столбцом Male. Пример представлен на рис. 2.4.

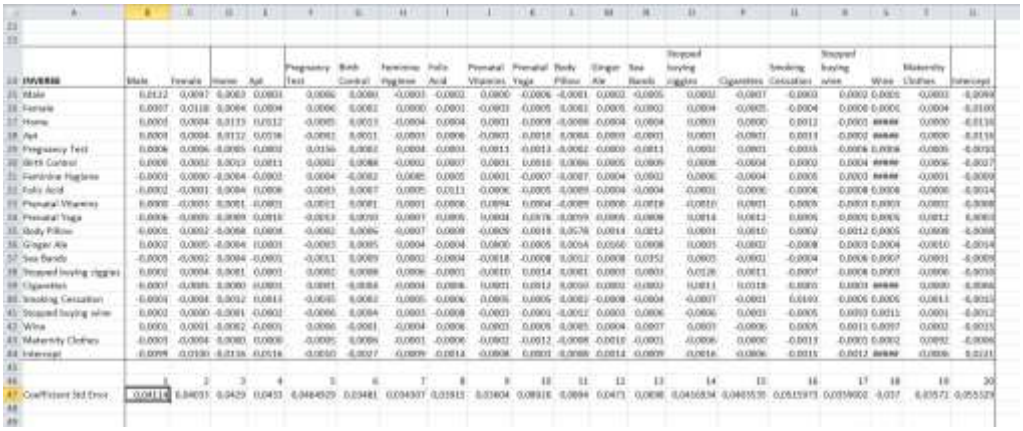


Рисунок 2.4 – Результаты расчета обратной матрицы СКВП и среднеквадратического отклонения коэффициентов

2.3.3.2.4. На листе «Linear Model» поместите в ячейку A3 заголовок «Coefficient Standard Error» и скопируйте в строку A3 значения рассчитанных отклонений из листа «ModelCoefficientStdError» как значения.

2.3.3.3. Рассчитайте t-статистику для каждого коэффициента по двустороннему критерию. Для этого нужно разделить значение коэффициента (по модулю) на соответствующее среднеквадратическое отклонение коэффициента. В ячейке A4 напишите наименование «t Statistic» и рассчитайте в строке 3 значения t-статистики для каждого коэффициента.

2.3.3.4. Рассчитайте оценку распределения Стьюдента относительно значения t-статистики и количества степеней свободы (Z3). Ячейку A5 назовите «t Test p Value», и далее в строке с помощью формулы СТЬЮДРАСП рассчитайте вероятность того, что коэффициент будет по меньшей мере таким, как при нулевой гипотезе. В формуле СТЬЮДРАСП используйте параметр «двусторонний критерий». Пример представлен на рис.2.5.

2.3.3.5. Представьте перечень статистически незначимых критериев (значение вероятности больше либо равно 0,05). Для обучения в дальнейшие эти критерии можно будет удалить. Сделайте вывод.

3. Тестирование модели.

3.1. На вкладке «Test Set» содержится тестовый набор данных о покупателях, которые не участвовали в обучении модели (тестовая выборка). Всего 1000 покупателей, 6% из которых относятся к прогнозируемому классу.

Ячейку V1 назовите «Linear Prediction» и в ячейках ниже осуществите прогноз по полученной модели для всех покупателей. Представьте результат в отчете. Проанализируйте, сделайте вывод.

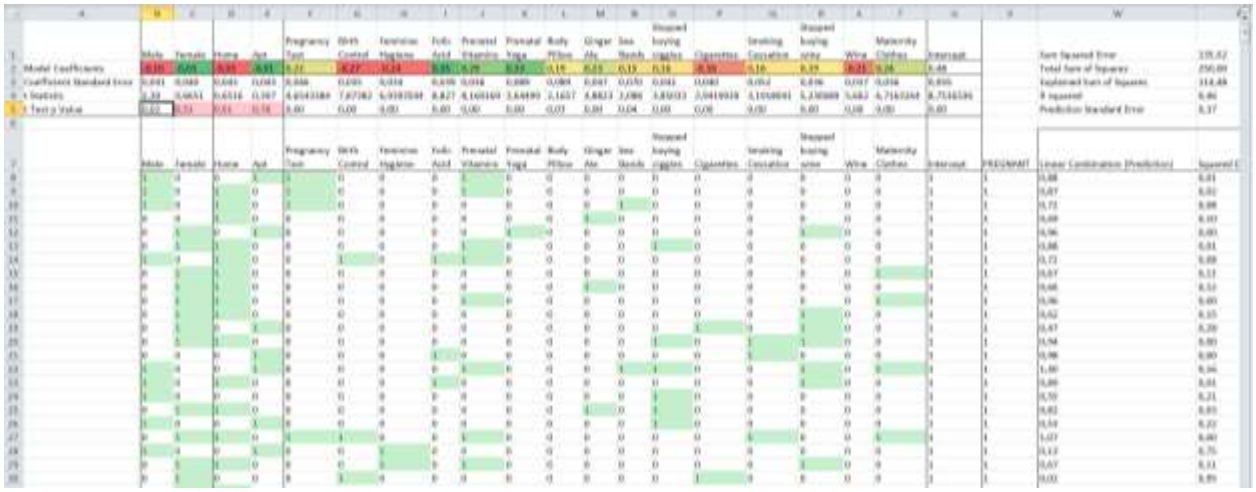


Рисунок 2.5 – Результаты проверки по t-критерию

3.2. Оцените параметры качества модели.

3.2.1. Установите граничные значения.

Для этого создайте лист «Performance». На листе введите названия столбцов как в таблице 2.2.

Таблица 2.2 – Показатели для оценки качество модели

A	B	C	D	E	F
Min Prediction (нижняя граница)	Probability Cutoff Classification (значения для классификации)	Precision (точность класса)	Specificity (Специфичность)	False Positive Rate (Доля ложно положительных примеров)	True Positive Rate / Recall / Sensitivity (Чувствительность)

В ячейке A2 найдите возможную нижнюю границу отсечения положительного (прогнозируемого) класса от другого (минимальное значение из всех полученных прогнозных значений на листе «Test Set»). В ячейке A4 введите «Max Prediction» и в A5 найдите возможную верхнюю границу отсечения одного класса от другого (максимальное значение из всех полученных прогнозных значений на листе «Test Set»).

3.2.2. В столбце B разместите возможные значения границы в диапазоне от минимального к максимальному с шагом 0,05. Например, -0,35; -0,3; - 0,25 и т.д.

3.2.3. В столбце C найдите точность (прогностическую положительную ценность полученного результата) для каждого граничного значения в столбце B. Для этого нужно посчитать сколько покупателей получили прогноз положительного класса больше либо равный данному граничному значению и разделить на общее количество строк с прогнозом, больше либо равным данному значению, например:

=СЧЁТЕСЛИМН('Test Set'!$V$2:$V$1001;">=" & B2;'Test Set'!$U$2:$U$1001;"=1")/СЧЁТЕСЛИ('Test Set'!$V$2:$V$1001;">=" & B2).

Проанализируйте как изменяется точность модели, сделайте вывод.

3.2.4. В столбце D оцените специфичность модели (доля действительно отрицательных результатов). То есть нужно рассчитать какая доля покупателей отрицательного класса для данной границы были верно отнесены к общему числу покупателей отрицательного класса.

Например:

=СЧЁТЕСЛИМН('Test Set'!$V$2:$V$1001;"<" & B2;'Test Set'!$U$2:$U$1001;"=0")/СЧЁТЕСЛИ('Test Set'!$U$2:$U$1001;"=0").

Проанализируйте как изменяется специфичность модели, сделайте вывод.

3.2.5.В столбце E рассчитайте долю ложноположительных результатов. Можно вычислить как (1 – Специфичность). Проанализируйте как изменяется показатель, сделайте вывод.

3.2.6. В столбце F вычислите долю действительно положительных результатов (чувствительность). Это доля правильно определенных покупателей положительного класса в общем количестве таких покупателей по всему набору данных. Например:

=СЧЁТЕСЛИМН('Test Set'!$V$2:$V$1001;">=" & B2;'Test Set'!$U$2:$U$1001;"=1")/СЧЁТЕСЛИ('Test Set'!$U$2:$U$1001;"=1")

Проанализируйте, как изменяется показатель модели, сделайте вывод.

Пример расчет метрик модели представлен на рис.2.6.

	A	B	C	D	E	F	G
1	Min Prediction	Probability Cutoff for Pregnant Classification	Precision	Specificity / True Negative Rate	False Positive Rate (1 - Specificity)	True Positive Rate / Recall / Sensitivity	
2	-0,35	-0,35	0,06	0,00	1,00	1,00	
3		-0,30	0,06	0,00	1,00	1,00	
4	Max Prediction	-0,25	0,06	0,01	0,99	1,00	
5	1,25	-0,20	0,06	0,02	0,98	1,00	
6		-0,15	0,06	0,03	0,97	1,00	
7		-0,10	0,06	0,05	0,95	1,00	
8		-0,05	0,06	0,08	0,92	1,00	
9		0,00	0,07	0,11	0,89	1,00	
10		0,05	0,07	0,13	0,87	0,98	
11		0,10	0,07	0,18	0,82	0,98	
12		0,15	0,08	0,23	0,77	0,98	
13		0,20	0,09	0,34	0,66	0,97	
14		0,25	0,10	0,44	0,56	0,95	
15		0,30	0,11	0,50	0,50	0,95	
16		0,35	0,11	0,53	0,47	0,95	
17		0,40	0,16	0,69	0,31	0,90	
18		0,45	0,19	0,76	0,24	0,87	
19		0,50	0,31	0,89	0,11	0,78	
20		0,55	0,34	0,91	0,09	0,75	
21		0,60	0,38	0,93	0,07	0,72	

Рисунок 2.6 – Результаты оценки качества модели

3.3. Постройте кривую ошибок (ROC). Это график зависимости действительно положительных значений от ложноположительных (столбцы E и F). Пример представлен на рис.2.7. По кривой на рисунках 2.7, 2.8, например, можно понять, что модель позволяет идентифицировать 40% покупателей положительного класса при граничном значении прогноза 0,85 без единого ложного срабатывания. А доли действительно положительных значений 75% можно достигнуть всего лишь при 9 ложных срабатываниях, граничное значение 0,55 (рис.2.8).

3.4. Выберите и обоснуйте граничные значения для вашей модели.

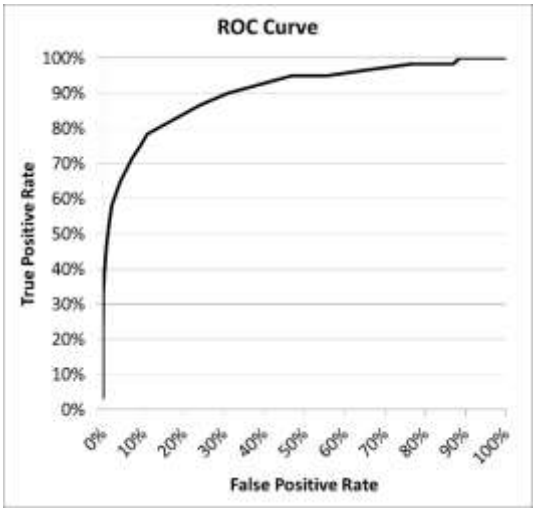


Рисунок 2.7 – Кривая ошибок (ROC-кривая)

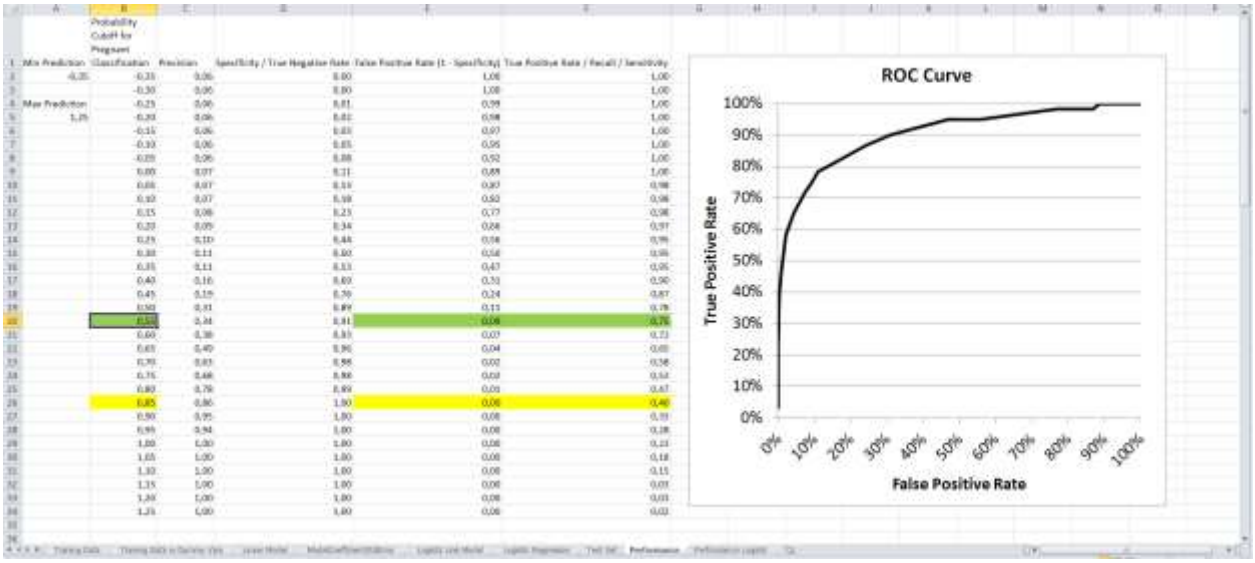


Рисунок 2.8 – Анализ кривой ошибок

2.3 Контрольные вопросы

1. Понятие множественной регрессии.
2. Опишите процесс обучения регрессионной модели.
3. Показатели значимости регрессионной модели: коэффициент детерминации, критерии Фишера и Стьюдента.
4. Метрики качества регрессионной модели.
5. Кривая ошибок.

3 КЛАСТЕРНЫЙ АНАЛИЗ

3.1. Теоретические сведения

Кластерный анализ (Data clustering) — задача разбиения заданной выборки объектов (ситуаций) на непересекающиеся подмножества, называемые кластерами, так, чтобы каждый кластер состоял из схожих объектов, а объекты разных кластеров существенно отличались [5-7].

Задача кластеризации относится к широкому классу задач обучения без учителя.

Типы входных данных:

1. Признаковое описание объектов. Каждый объект описывается набором своих характеристик, называемых признаками. Признаки могут быть числовыми или нечисловыми.
2. Матрица расстояний между объектами. Каждый объект описывается расстояниями до всех остальных объектов обучающей выборки.

Матрица расстояний может быть вычислена по матрице признаковых описаний объектов многими способами, в зависимости от того, как ввести функцию расстояния (метрику) между признаковыми описаниями. Часто используется евклидова метрика, однако этот выбор в большинстве случаев является эвристикой и обусловлен лишь соображениями удобства.

Алгоритмов кластерного анализа достаточно много. Все их можно подразделить на иерархические и неиерархические. Иерархические (древовидные) процедуры – наиболее распространённые алгоритмы кластерного анализа по их реализации на ЭВМ. Различают агломеративные (от слова agglomerate – собирать) и итеративные дивизимные (от слова division – разделять) алгоритмы.

Принцип работы иерархических агломеративных процедур состоит в последовательном объединении групп элементов сначала самых близких, а затем всё более отдалённых друг от друга. Принцип работы иерархических дивиизмных процедур, наоборот, состоит в последовательном разделении групп элементов сначала самых далёких, а затем всё более близких друг от друга. Большинство этих алгоритмов исходит из матрицы расстояний (сходства).

Меры расстояния.

Евклидово расстояние. Это наиболее общий тип расстояния. Оно является геометрическим расстоянием в многомерном пространстве и вычисляется пj формуле:

$$\rho(x, x') = \sqrt{\sum_{i=1}^{n} (x_i - x_i')^2}.$$

Заметим, что евклидово расстояние (и его квадрат) вычисляется по исходным, а не по стандартизованным данным.

Квадрат евклидова расстояния. Может применяться, чтобы придать большие веса более отдаленным друг от друга объектам. Это расстояние вычисляется следующим образом:

$$\rho(x, x') = \sum_{i=1}^{n} (x_i - x_i')^2.$$

Расстояние городских кварталов (манхэттенское расстояние). Это расстояние является просто средним разностей по координатам. В большинстве случаев эта мера расстояния приводит к таким же результатам, как и для обычного расстояния Евклида. Однако отметим, что для этой меры влияние отдельных больших разностей (выбросов) уменьшается (так как они не возводятся в квадрат).

Манхэттенское расстояние вычисляется по формуле:

$$\rho(x, x') = |x_i - x_i'|.$$

Расстояние Чебышева. Это расстояние может оказаться полезным, когда желают определить два объекта как «различные», если они различаются по какой-либо одной координате (каким-либо одним измерением). Расстояние Чебышева вычисляется по формуле:

$$\rho(x, x') = \max(|x_i - x_i'|).$$

Степенное расстояние. Иногда желают прогрессивно увеличить или уменьшить вес, относящийся к размерности, для которой соответствующие объекты сильно отличаются. Это может быть достигнуто с использованием степенного расстояния. Степенное расстояние вычисляется по формуле:

$$\rho(x, x') = \sqrt[r]{\sum_{i=1}^{n} (x_i - x_i')^p}.$$

где *r* и *p* - параметры, определяемые пользователем. Параметр *p* ответственен за постепенное взвешивание разностей по отдельным координатам, параметр *r* ответственен за прогрессивное взвешивание больших расстояний между объектами. Если оба параметра - *r* и *p*, равны двум, то это расстояние совпадает с расстоянием Евклида.

Процент несогласия. Эта мера используется в тех случаях, когда данные являются категориальными. Это расстояние вычисляется по формуле:

$$\rho(x, x') = (\text{Количество } x_i \neq x_i') \,/\, i.$$

Расстояние по косинусу. Широко используемый метод подсчета асимметричного расстояния для бинарных данных (формат 0–1) называется расстоянием по косинусу. Рассмотрим пару двумерных бинарных векторов (1,1) и (1,0), характеризующих заказы покупателей. В первом векторе были заказаны оба товара, в то время как во втором только первый. Косинус угла между двумя бинарными заказами — это число совпадений заказов в двух векторах, разделенное на произведение квадратных корней количества заказов первого и второго векторов (рис.3.1).

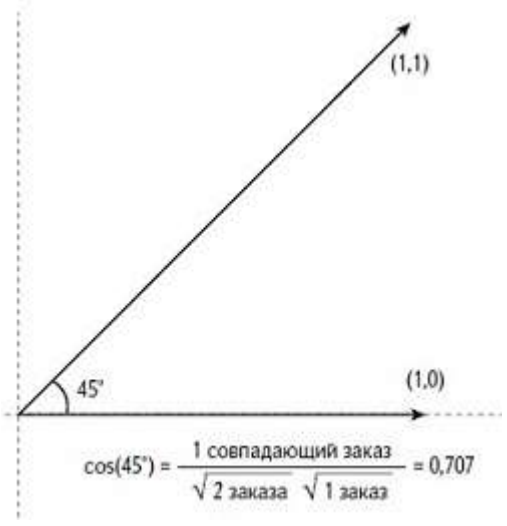


Рисунок 3.1 – Близость по косинусу на примере бинарных векторов

Для бинарных данных эта близость находится в промежутке от 0 до 1, причем у двух векторов не получается 1, пока все их заказы не совпадут. Это означает, что величина равная (1 – близость по косинусу) может использоваться как мера расстояния, называемая расстоянием по косинусу, которое также варьируется от 0 до 1.

Метод k-средних.

Предположим, есть гипотезы относительно числа m кластеров (по переменным или наблюдениям). Тогда можно задать программе создать ровно m кластеров так, чтобы они были настолько различны, насколько это возможно. Именно для решения задач этого типа предназначен метод k-means (k-средних). Гипотеза может основываться на теоретических соображениях, результатах предшествующих исследований или догадке. Выполняя последовательное разбиение на различное число кластеров, можно сравнивать качество получаемых решений. Программа начинает с m случайно выбранных кластеров, а затем изменяет принадлежность объектов к ним, чтобы минимизировать изменчивость внутри кластеров и максимизировать изменчивость между кластерами. Алгоритм случайным образом в пространстве назначает центры будущих кластеров. Затем вычисляет расстояние между центрами кластеров и каждым объектом, и объект приписывается к тому кластеру, к которому он ближе всего. Завершив приписывание, алгоритм вычисляет средние значения для каждого кластера. Этих средних будет столько, сколько используется переменных для проведения анализа, – k штук. Набор средних представляет собой координаты нового положения центра кластера.

Алгоритм вновь вычисляет расстояние от каждого объекта до центров кластеров и приписывает объекты к ближайшему кластеру. Вновь вычисляются центры тяжести кластеров, и этот процесс повторяется до тех пор, пока центры тяжести не перестанут «мигрировать» в пространстве. Если в древовидной кластеризации можно использовать категориальные переменные, то так как в методе k-средних в качестве метрики используют евклидову метрику, то перед проведением кластеризации необходимо стандартизовать переменные. По этой же причине в методе предполагается, что переменные непрерывные и измерены как минимум в интервальной шкале.

Метод k-медиан – применяемая в статистике и машинном обучении вариация метода k-средних для задач кластеризации, где для определения центроида кластера вместо среднего вычисляется медиана. Задача определения k-медиан состоит в поиске таких k центров, что сформированные по ним кластеры будут наиболее «компактными». Формально, при заданных точках данных x_i, k центров c_j должны быть выбраны так, чтобы минимизировать сумму расстояний от каждой x_i до ближайшего c_j. Метод k-медиан иногда работает лучше, чем метод k-средних, где минимизируется сумма квадратов расстояний. Критерий суммы расстояний широко используется для транспортных задач.

Метрики качества кластеризации.

Задача оценки качества кластеризации является более сложной по сравнению с оценкой качества классификации. Во-первых, такие оценки не должны зависеть от самих значений меток, а только от самого разбиения выборки. Во-вторых, не всегда известны истинные метки объектов, поэтому также нужны оценки, позволяющие оценить качество кластеризации, используя только неразмеченную выборку.

Выделяют внешние и внутренние метрики качества. Внешние используют информацию об истинном разбиении на кластеры, в то время как внутренние метрики не используют никакой внешней информации и оценивают качество кластеризации, основываясь только на наборе данных. Оптимальное число кластеров обычно определяют с использованием внутренних метрик.

Силуэт.

Данный коэффициент не предполагает знания истинных меток объектов, и позволяет оценить качество кластеризации, используя только саму (неразмеченную) выборку и результат кластеризации. Сначала силуэт определяется отдельно для каждого объекта. Обозначим через a — среднее расстояние от данного объекта до объектов из того же кластера, через b – среднее расстояние от данного объекта до объектов из ближайшего кластера (отличного от того, в котором лежит сам объект). Тогда силуэтом данного объекта называется величина:

$$s = ((b - a))/(\max(a, b)).$$

Силуэтом выборки называется средняя величина силуэта объектов данной выборки. Таким образом, силуэт показывает, насколько среднее расстояние до объектов своего кластера отличается от среднего расстояния до объектов других кластеров. Данная величина лежит в диапазоне [−1, 1]:

- значения, близкие к -1, соответствуют плохим (разрозненным) кластеризациям;
- значения, близкие к нулю, говорят о том, что кластеры пересекаются и накладываются друг на друга;
- значения, близкие к 1, соответствуют «плотным» четко выделенным кластерам.

Таким образом, чем больше силуэт, тем более четко выделены кластеры, и они представляют собой компактные, плотно сгруппированные облака точек. С помощью силуэта можно выбирать оптимальное число кластеров k (если оно заранее неизвестно) – выбирается число кластеров, максимизирующее значение силуэта. Силуэт зависит от формы кластеров, и достигает больших значений на более выпуклых кластерах, получаемых с помощью алгоритмов, основанных на восстановлении плотности распределения.

3.2. Лабораторная работа. Задание и порядок выполнения работы

Цель лабораторной работы: научиться проводить кластерный анализ методами k-средних и k-медиан.

Задание. Используя Excel или Calc проведите сегментирование клиентской базы компании (кластеризацию методом k-средних и k-медиан) для проведения таргетированных рассылок о предложениях компании.

Исходные данные находятся в файле (файлы выдаются по вариантам).

В нем содержатся данные о сделках некоторой компании за год:

- метаданные по каждому предложению сохранены в электронной таблице, включая вид продукции, минимальное количество продукции в заказе, скидку на розничную продажу, информацию о том, пройден ли ценовой максимум, информацию о стране происхождения. Эти данные размещены во вкладке под названием OfferInformation, как показано на рис. 3.1;
- данные о заказах клиентов представлены на листе Transactions в формате имя-№ заказа (рис.3.2).

	A	B	C	D	E	F	G
1	Offer #	Campaign	Varietal	Minimum Qty (kg)	Discount (%)	Origin	Past Peak
2	1	January	Malbec	72	56	France	FALSE
3	2	January	Pinot Noir	72	17	France	FALSE
4	3	February	Espumante	144	32	Oregon	TRUE
5	4	February	Champagne	72	48	France	TRUE
6	5	February	Cabernet Sauvignon	144	44	New Zealand	TRUE
7	6	March	Prosecco	144	86	Chile	FALSE
8	7	March	Prosecco	6	40	Australia	TRUE
9	8	March	Espumante	6	45	South Africa	FALSE

Рисунок 3.1 – Метаданные предложений компании

	A	B
1	Customer Last Name	Offer #
2	Smith	2
3	Smith	24
4	Johnson	17
5	Johnson	24
6	Johnson	26
7	Williams	18
8	Williams	22
9	Williams	31
10	Brown	7
11	Brown	29
12	Brown	30

Рисунок 3.2 – Список заказов по покупателям

Порядок выполнения работы.

Далее будет описан порядок выполнения работы, пример выполнения приводится на основе данных из [4] (в нем анализируются сделки винной компании).

Для выполнения работы выполните следующие этапы.

1. Создайте матрицу сделок по покупателям (лист Pivot), используя Мастер создания сводных таблиц, В результате получается таблица, показанная на рис.3.3.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
3	Count of Offer #	Colum													
4	Row Labels	Adams	Allen	Anderson	Bailey	Baker	Barnes	Bell	Bennett	Brooks	Brown	Butler	Campbell	Carter	Clark
5	1											1			
6	2							1					1		
7	3									1					
8	4											1			1
9	5														
10	6														
11	7				1		1				1			1	
12	8								1	1					
13	9		1												
14	10					1		1							
15	11									1					1

Рисунок 3.3 – Сводная таблица «клиент-сделка»

2. Скопируйте лист OfferInformation и назовите его Matrix. В этот новый лист вставьте значения из сводной таблицы (не нужно копировать и вставлять номер сделки, потому что он уже содержится в информации о заказе), начиная со столбца H. В итоге у вас должна получиться расширенная версия матрицы, дополненная информацией о заказах, как на рис. 3.4.

	A	B	C	D	E	F	G	H	I	J	K
1	Offer #	Campaign	Varietal	Minimum Qty (kg)	Discount (%)	Origin	Past Peak	Adams	Allen	Anderson	Baile
4	3	February	Espumante	144	32	Oregon	TRUE				
5	4	February	Champagne	72	48	France	TRUE				
6	5	February	Cabernet Sauvignon	144	44	New Zealand	TRUE				
7	6	March	Prosecco	144	86	Chile	FALSE				
8	7	March	Prosecco	6	40	Australia	TRUE				1
9	8	March	Espumante	6	45	South Africa	FALSE				
10	9	April	Chardonnay	144	57	Chile	FALSE		1		
11	10	April	Prosecco	72	52	California	FALSE				
12	11	May	Champagne	72	85	France	FALSE				
13	12	May	Prosecco	72	83	Australia	FALSE				
14	13	May	Merlot	6	43	Chile	FALSE				
15	14	June	Merlot	72	64	Chile	FALSE				

Рисунок 3.4 – Описание сделок и данные о заказах, объединенные в единую матрицу

3. Проведите кластерный анализ для 4-х кластеров (значение k=4).

3.1. Скопируйте данные из листа Matrix, в новый лист и назовите его 4MC. Вставьте 4 столбца после ценового максимума в столбцы от H до K, которые будут кластерными центрами. Назовите эти кластеры от Cluster 1 до Cluster 4. Лист 4MC появится будет выглядеть, как показано на рис.3.5.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	
1	Offer #	Campaign	Varietal	Minimum Qty	Discount (%)	Origin	Past Peak	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Adams	Allen	Anderson	Bailey	B
2	1	January	Malbec	72	56	France	FALSE									
3	2	January	Pinot Noir	72	17	France	FALSE									
4	3	February	Espumante	144	32	Oregon	TRUE									
5	4	February	Champagn	72	48	France	TRUE									
6	5	February	Cabernet S	144	44	New Zeala	TRUE									
7	6	March	Prosecco	144	86	Chile	FALSE									
8	7	March	Prosecco	6	40	Australia	TRUE								1	
9	8	March	Espumante	6	45	South Afric	FALSE									
10	9	April	Chardonna	144	57	Chile	FALSE						1			
11	10	April	Prosecco	72	52	California	FALSE									
12	11	May	Champagn	72	85	France	FALSE									

Рисунок 3.5 – Подготовленные столбцы кластерных центров

3.2. Рассчитайте евклидовы расстояния для каждого клиента до каждого кластерного центра. В ячейках G34:G37 впишите названия «Distance to Cluster 1»….. «Distance to Cluster 4».

Например, в ячейке L34, под заказами Адамса, можно вычислить разницу между вектором Адамса и кластерным центром, возвести ее в квадрат, сложить и затем извлечь корень, используя следующую формулу для массивов:

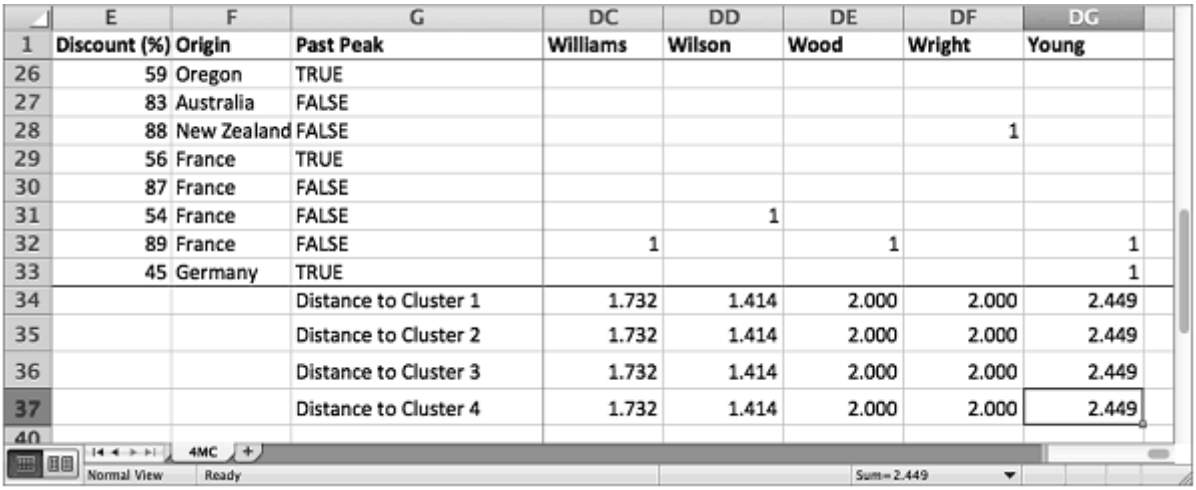
{=КОРЕНЬ(СУММА(L$2:L$33-$H$2:$H$33)A2))}.

Формулу для массивов (введите формулу и нажмите Ctrl+Shift+Enter) нужно использовать, потому что ее часть (L2:L33-H2:H33)^2 должна «знать», куда обращаться для вычисления разниц и возведения их в квадрат, шаг за шагом.

Аналогично посчитайте расстояния для каждого из кластеров, а затем растяните получившиеся формулы на всех клиентов. Рассчитанные расстояния показаны на рис. 3.6.

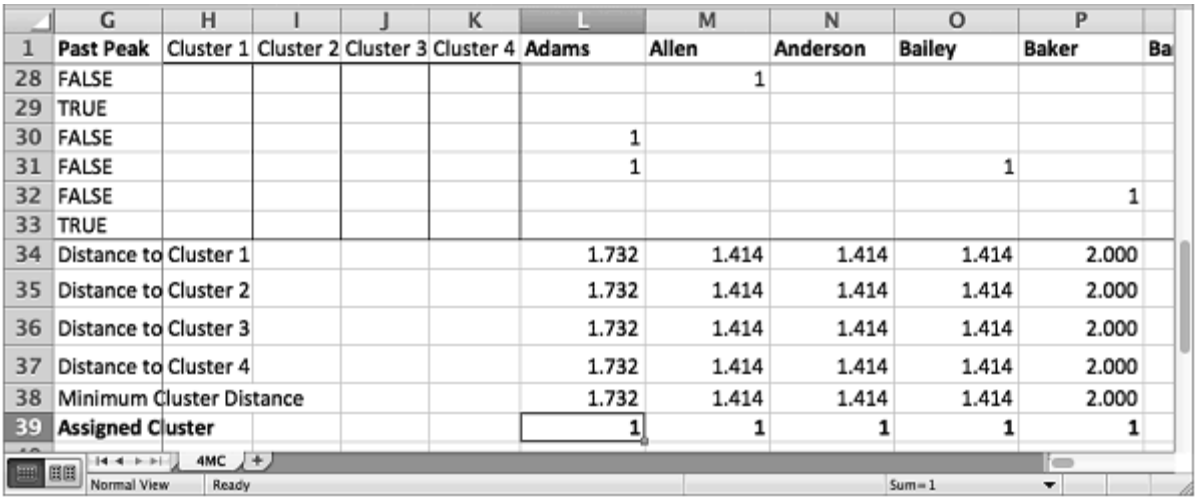
3.3. Проведите распределение по кластерам по кратчайшему расстоянию следующим образом:

- добавьте названия строк 38 и 39 в ячейки 38 и 39 столбца G «Minimum Cluster Distance» и «Assigned Cluster»;
- рассчитайте минимальные расстояния до кластерного центра по каждому клиенту (функция МИН);
- выведите номер кластера с минимальным расстоянием для каждого клиента (используем формулу ПОИСКПОЗ). Изначально для всех клиентов это будет кластер 1 (рис.3.7).



	E	F	G	DC	DD	DE	DF	DG
1	Discount (%)	Origin	Past Peak	Williams	Wilson	Wood	Wright	Young
26	59	Oregon	TRUE					
27	83	Australia	FALSE					
28	88	New Zealand	FALSE				1	
29	56	France	TRUE					
30	87	France	FALSE					
31	54	France	FALSE		1			
32	89	France	FALSE	1		1		1
33	45	Germany	TRUE					1
34			Distance to Cluster 1	1.732	1.414	2.000	2.000	2.449
35			Distance to Cluster 2	1.732	1.414	2.000	2.000	2.449
36			Distance to Cluster 3	1.732	1.414	2.000	2.000	2.449
37			Distance to Cluster 4	1.732	1.414	2.000	2.000	2.449
40								

Рисунок 3.6 – Расчет расстояний от каждого покупателя до кластерных центров



	G	H	I	J	K	L	M	N	O	P	
1	Past Peak	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Adams	Allen	Anderson	Bailey	Baker	Ba
28	FALSE						1				
29	TRUE										
30	FALSE					1					
31	FALSE					1			1		
32	FALSE									1	
33	TRUE										
34	Distance to	Cluster 1				1.732	1.414	1.414	1.414	2.000	
35	Distance to	Cluster 2				1.732	1.414	1.414	1.414	2.000	
36	Distance to	Cluster 3				1.732	1.414	1.414	1.414	2.000	
37	Distance to	Cluster 4				1.732	1.414	1.414	1.414	2.000	
38	Minimum C	luster Distance				1.732	1.414	1.414	1.414	2.000	
39	Assigned C	luster				1	1	1	1	1	

Рисунок 3.7 – Добавление на лист привязки к кластерам

3.4. Осуществите поиск решений для кластерных центров. Чтобы установить наилучшее положение кластерных центров, нужно найти такие значения в столбцах от H до K, которые минимизируют общее расстояние между покупателями и кластерными центрами, к которым они привязаны. Оптимизация в Excel производится с помощью надстройки «Поиск решения» (вкладка «Данные-Анализ). В Calc оптимизация проводится с помощью встроенной функции «Решатель» в меню «Сервис».

Для этого:

3.4.1. Задайте целевую функцию в ячейке A36 (сумма минимальных расстояний от клиентов до кластеров);

3.4.2. Осуществите постановку задачи для Поиска решения (рис.3.8, 3.9):

- цель: минимизировать общие расстояния от покупателей к их кластерным центрам (A36);
- переменные: вектор каждой сделки относительно кластерного центра (H2:K33);
- условия: кластерные центры должны иметь значения в пределах от 0 до 1.

Поскольку евклидово расстояние – нелинейная функция, используйте эволюционный алгоритм. Также установите параметры Эволюционного алгоритма (рекомендуемое время ожидания 600 с).

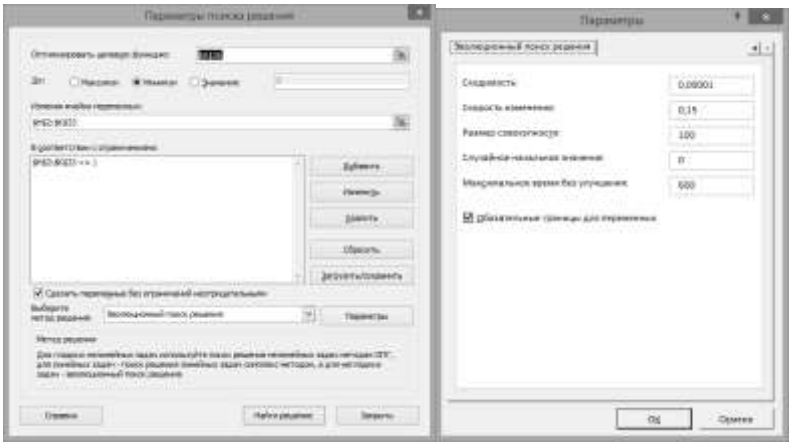


Рисунок 3.8 – Установки «Поиска решения» для 4-центровой кластеризации в Excel

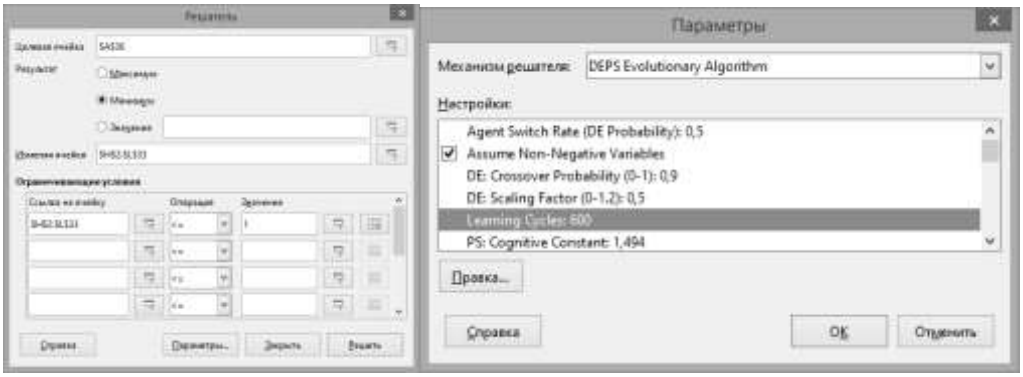


Рисунок 3.9 – Установки «Поиска решения» для 4-центровой кластеризации в Calc

3.4. Осуществите поиск решения (кнопка «Найти решение»). Проанализируйте результат и сделайте выводы.

3.5. Проведите рейтингование сделок по результатам кластеризации.

3.5.1. Скопируйте лист Offerlnformation, копию назовите 4MC — TopDealsByCluster. Пронумеруйте столбцы от H до K на этом новом листе от 1 до 4 (как на рис. 3.10), примените к ним условное форматирование (для удобства визуального анализа данных).

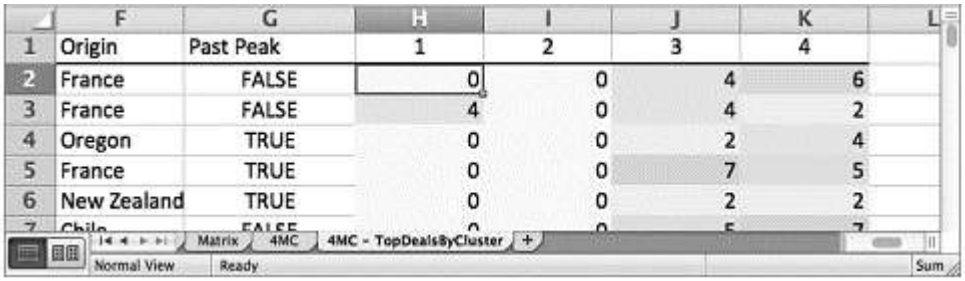
	A	B	C	D	E	F	G	H	I	J	K
1	Offer #	Offer date	Product	Minimum Qt	Discount	Origin	Past Peak	1	2	3	4
2	1	January	Malbec	72	56	France	FALSE				
3	2	January	Pinot Noir	72	17	France	FALSE				
4	3	February	Espumante	144	32	Oregon	TRUE				
5	4	February	Champagne	72	48	France	TRUE				

Matrix | 4MC | 4MC – TopDealsByCluster

Normal View | Ready | Sum = 1

Рисунок 3.10 – Создание листа таблицы для подсчета популярности сделок с помощью кластеров

3.5.2. Посчитайте количество сделок по кластерам. Для этого, исходя из привязок клиентов по кластерам на листе 4MC, нужно сравнить названия столбцов от H до K на листе 4MC – TopDealsByCluster, с кластером на листе 4MC и затем сложить количество сделок в каждой строке. Используйте функцию СУММЕСЛИ (рис.3.11).

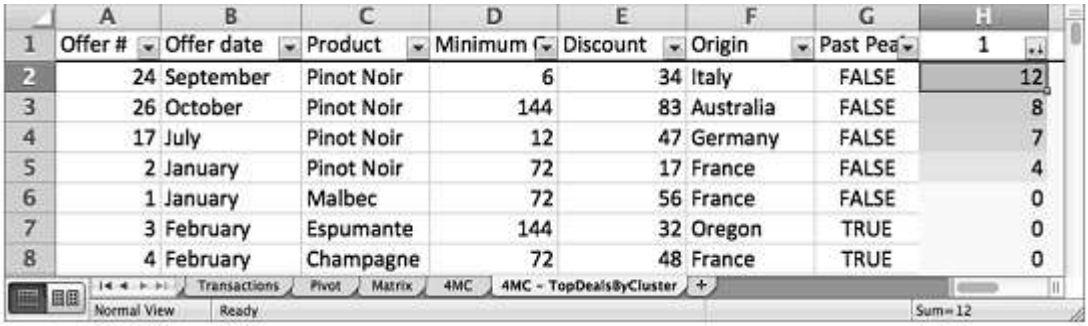


	F	G	H	I	J	K	L
1	Origin	Past Peak	1	2	3	4	
2	France	FALSE	0	0	4	6	
3	France	FALSE	4	0	4	2	
4	Oregon	TRUE	0	0	2	4	
5	France	TRUE	0	0	7	5	
6	New Zealand	TRUE	0	0	2	2	

Рисунок 3.11 – Общее количество сделок по каждому предложению в кластерах

3.5.3. Проанализируйте полученные данные по сделкам и сделайте предположения об особенностях клиентах каждого кластера.

Выделяя столбцы от A до K и применяя автофильтрацию, вы можете сортировать полученные данные. Например, отсортировав от наибольшего к наименьшему столбец H, мы видим, какие сделки наиболее популярны в кластере 1 (рис. 3.12). Можно сделать предположение, что клиенты 1 кластера предпочитают сорт Pinot Noir.



	A	B	C	D	E	F	G	H
1	Offer #	Offer date	Product	Minimum	Discount	Origin	Past Pea	1
2	24	September	Pinot Noir	6	34	Italy	FALSE	12
3	26	October	Pinot Noir	144	83	Australia	FALSE	8
4	17	July	Pinot Noir	12	47	Germany	FALSE	7
5	2	January	Pinot Noir	72	17	France	FALSE	4
6	1	January	Malbec	72	56	France	FALSE	0
7	3	February	Espumante	144	32	Oregon	TRUE	0
8	4	February	Champagne	72	48	France	TRUE	0

Рисунок 3.12 – Сортировка кластера 1 (любители *Pinot Noir)*

3.6. Оцените качество кластеризации по четырем кластерам. Рассчитайте силуэт.

3.6.1. Рассчитайте матрицу расстояний, для этого:

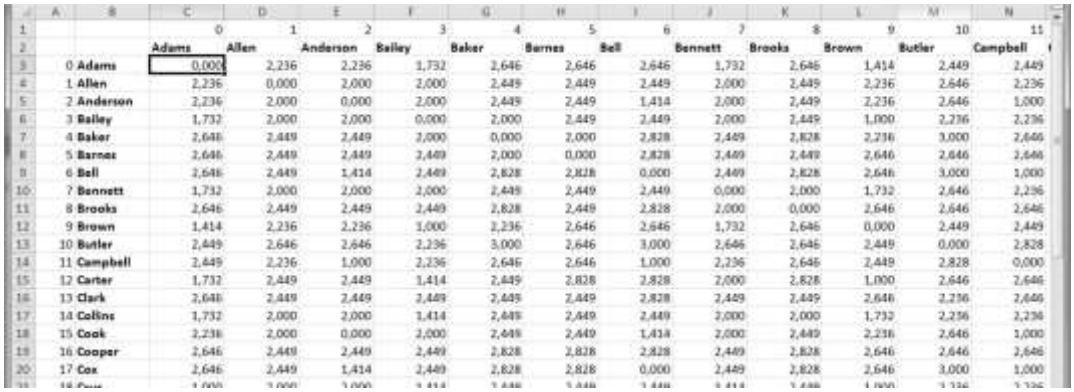
- создайте новый лист «Distances»;
- вставьте список клиентов по горизонтали и вертикали, пронумеруйте клиентов по строкам и столбцам от 0 до 99 (расположите нумерацию, соответственно, в первой строке и первом столбце);
- рассчитайте евклидовы расстояния между всеми клиентами (рис.3.13). Для удобства используйте функцию СМЕЩ. Не забывайте про формулы массивов!

Пример формулы для клетки C3:

={КОРЕНЬ(СУММ((СМЕЩ(Matrix!$H$2:$H$33;0;Distances!C$1)-СМЕЩ(Matrix!$H$2:$H$33;0;Distances!$A3))^2))}.

Пример формулы для клетки E9:

={КОРЕНЬ(СУММ((СМЕЩ(Matrix!$H$2:$H$33;0;Distances!E$1)-СМЕЩ(Matrix!$H$2:$H$33;0;Distances!$A9))^2))}.



		0	1	2	3	4	5	6	7	8	9	10	11
		Adams	Allen	Anderson	Bailey	Baker	Barnes	Bell	Bennett	Brooks	Brown	Butler	Campbell
0	Adams	0,000	2,236	2,236	1,732	2,646	2,646	2,646	1,732	2,646	1,414	2,449	2,449
1	Allen	2,236	0,000	2,000	2,000	2,449	2,449	2,449	2,000	2,449	2,236	2,646	2,236
2	Anderson	2,236	2,000	0,000	2,000	2,449	2,449	1,414	2,000	2,449	2,236	2,646	1,000
3	Bailey	1,732	2,000	2,000	0,000	2,000	2,449	2,449	2,000	2,449	1,000	2,236	2,236
4	Baker	2,646	2,449	2,449	2,000	0,000	2,000	2,828	2,449	2,828	2,236	3,000	2,646
5	Barnes	2,646	2,449	2,449	2,449	2,000	0,000	2,828	2,449	2,449	2,646	2,646	2,646
6	Bell	2,646	2,449	1,414	2,449	2,828	2,828	0,000	2,449	2,828	2,646	3,000	1,000
7	Bennett	1,732	2,000	2,000	2,000	2,449	2,449	2,449	0,000	2,000	1,732	2,646	2,236
8	Brooks	2,646	2,449	2,449	2,449	2,828	2,449	2,828	2,000	0,000	2,646	2,646	2,646
9	Brown	1,414	2,236	2,236	1,000	2,236	2,646	2,646	1,732	2,646	0,000	2,449	2,449
10	Butler	2,449	2,646	2,646	2,236	3,000	2,646	3,000	2,646	2,646	2,449	0,000	2,828
11	Campbell	2,449	2,236	1,000	2,236	2,646	2,646	1,000	2,236	2,646	2,449	2,828	0,000
12	Carter	1,732	2,449	2,449	1,414	2,449	2,828	2,828	2,000	2,828	1,000	2,646	2,646
13	Clark	2,646	2,449	2,449	2,449	2,449	2,449	2,828	2,449	2,449	2,646	2,236	2,646
14	Collins	1,732	2,000	2,000	1,414	2,449	2,449	2,449	2,000	2,000	1,732	2,236	2,236
15	Cook	2,236	2,000	0,000	2,000	2,449	2,449	1,414	2,000	2,449	2,236	2,646	1,000
16	Cooper	2,646	2,449	2,449	2,449	2,828	2,828	2,828	2,449	2,828	2,646	2,646	2,646
17	Cox	2,646	2,449	1,414	2,449	2,828	2,828	0,000	2,449	2,828	2,646	3,000	1,000

Рисунок 3.13 – Матрица расстояний

3.6.2. Рассчитайте силуэт, для этого:

3.6.2.1. Создайте новый лист **4MC Silhouette.** Скопируйте с листа 4MC имена клиентов в столбец A, а привязки к кластерам (значения) в столбец B. Озаглавьте столбцы с C до F «Distance from people in 1» … «Distance from people in 4»

3.6.2.2. Рассчитайте средние расстояния для каждого клиента между ним и другими клиентами, входящими в конкретный кластер. Используйте функцию СРЗНАЧЕСЛИ.

Например, для Адамса, формулы будут иметь вид:

=СРЗНАЧЕСЛИ('4MC'!$L$39:$DG$39;1;Distances!$C3:$CX3);

=СРЗНАЧЕСЛИ('4MC'!$L$39:$DG$39;2;Distances!$C3:$CX3);

=СРЗНАЧЕСЛИ('4MC'!$L$39:$DG$39;3;Distances!$C3:$CX3);

=СРЗНАЧЕСЛИ('4MC'!$L$39:$DG$39;4;Distances!$C3:$CX3).

3.6.2.3. В столбце G (Заголовок столбца - Closest) найдите ближайшую группу покупателей. Используйте функцию МИН.

3.6.2.4. В столбце H (Заголовок столбца - Second Closest) найдите вторую по близости группу покупателей. Используйте функцию НАИМЕНЬШИЙ.

3.6.2.5. В столбце I (Заголовок столбца - My Cluster) найдите расстояние до членов собственного кластера. Используйте функцию ИНДЕКС.

3.6.2.6. В столбце J (Заголовок столбца - Neighboring Cluster) найдите расстояние до ближайшего группы покупателей, находящихся не в вашем кластере. Если собственное кластерное расстояние равно расстоянию ближайшего кластера, то ответ в столбце H (Second Closest), если нет – в столбце G (Closest). Используйте функцию ЕСЛИ.

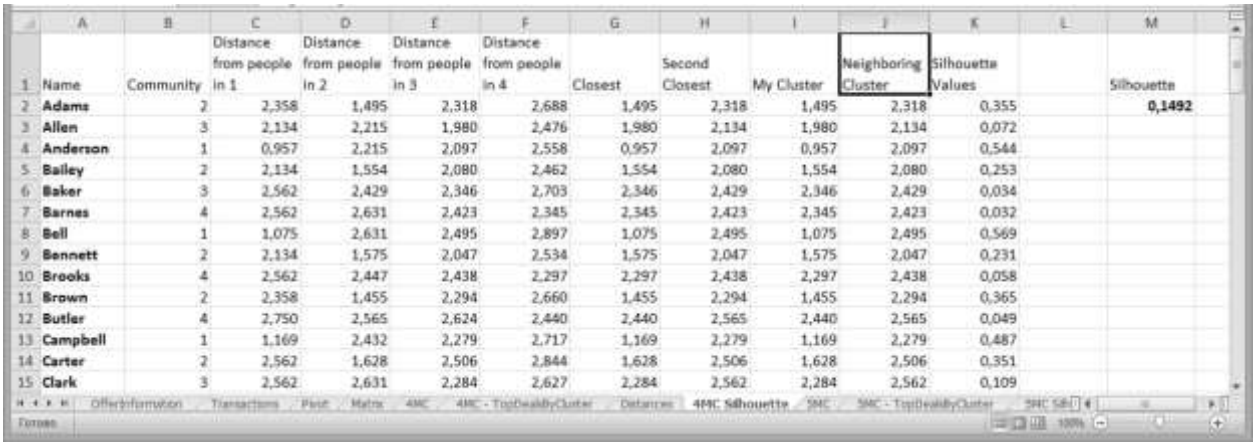
3.6.3.7. Рассчитайте силуэты по каждому клиенту по формуле:

$$s = ((b - a))/(\max(a, b)).$$

где a – My Cluster, B –Neighboring Cluster.

Проанализируйте получившиеся значения.

3.6.3.8. Рассчитайте Итоговый силуэт как среднее значение всех силуэтов по клиентам (рис.3.14). Проанализируйте получившееся значение.



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Name	Community	Distance from people in 1	Distance from people in 2	Distance from people in 3	Distance from people in 4	Closest	Second Closest	My Cluster	Neighboring Cluster	Silhouette Values		Silhouette
2	Adams	2	2,358	1,495	2,318	2,688	1,495	2,318	1,495	2,318	0,355		0,1492
3	Allen	3	2,134	2,215	1,980	2,476	1,980	2,134	1,980	2,134	0,072		
4	Anderson	1	0,957	2,215	2,097	2,558	0,957	2,097	0,957	2,097	0,544		
5	Bailey	2	2,134	1,554	2,080	2,462	1,554	2,080	1,554	2,080	0,253		
6	Baker	3	2,562	2,429	2,346	2,703	2,346	2,429	2,346	2,429	0,034		
7	Barnes	4	2,562	2,631	2,423	2,345	2,345	2,423	2,345	2,423	0,032		
8	Bell	1	1,075	2,631	2,495	2,897	1,075	2,495	1,075	2,495	0,569		
9	Bennett	2	2,134	1,575	2,047	2,534	1,575	2,047	1,575	2,047	0,231		
10	Brooks	4	2,562	2,447	2,438	2,297	2,297	2,438	2,297	2,438	0,058		
11	Brown	2	2,358	1,455	2,294	2,660	1,455	2,294	1,455	2,294	0,365		
12	Butler	4	2,750	2,565	2,624	2,440	2,440	2,565	2,440	2,565	0,049		
13	Campbell	1	1,169	2,432	2,279	2,717	1,169	2,279	1,169	2,279	0,487		
14	Carter	2	2,562	1,628	2,506	2,844	1,628	2,506	1,628	2,506	0,351		
15	Clark	3	2,562	2,631	2,284	2,627	2,284	2,562	2,284	2,562	0,109		

Рисунок 3.14 – Результаты расчеты Силуэта

4. Проведите кластерный анализ для 5 кластеров (значение k=5). Этапы выполнения аналогичны пункту 3. Результаты должны быть представлены в листах, соответственно, 5MC, 5MC — TopDealsByCluster, 5MC Silhouette.

Сделайте вывод о наилучшем количестве кластеров.

5. Проведите кластерный анализ методом k-медиан, используя расстояние по косинусу.

5.1. Скопируйте лис 5MC и переименуйте его в 5MedC и удалите данные, рассчитанные с помощью Поиска решения (кластерные центры).

5.2. Рассчитайте расстояние по косинусу в строках 34-38. Близость по косинусу: косинус угла между двумя бинарными заказами — это число совпадений заказов в двух векторах, разделенное на произведение квадратных корней количества заказов первого и второго векторов. Расстояние по косинусу = 1–Близость по косинусу.

Чтобы найти значение совпадающих заказов у клиента и кластера, используйте функцию СУММПРОИЗВ.

Вставьте в формулу проверку на ошибку, если кластерный центр окажется равным 0. В этом случае присвойте ему значение 1.

Таким образом, формула для расчёта расстояния по косинусу от Адамса до кластера 1 имеет вид:

=ЕСЛИОШИБКА(1-СУММПРОИЗВ(M$2:M$33;$H$2:$H$33)/

/(КОРЕНЬ(СУММ(M$2:M$33))*КОРЕНЬ(СУММ($H$2:$H$33)));1).

5.3. Осуществите поиск решения, для этого замените условие <=1 на бинарное (рис.3.15). Проанализируйте результат и сделайте выводы.

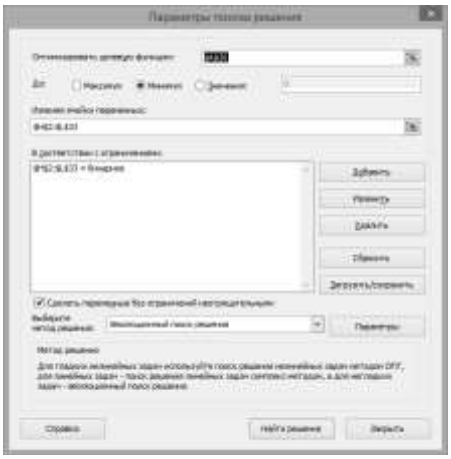


Рисунок 3.15 – Постановка задачи поиска решения для метода k-медиан

5.4. Рассчитайте рейтинг сделок для 5-медианных кластеров. Для этого скопируйте лист 5C – TopDealsByCluster и переименуйте его в 5MedC – TopDealsByCluster. Измените ссылки на листы в формулах.

5.5. Проанализируйте полученные данные по сделкам и сделайте предположения о клиентах каждого кластера.

6. Сравните результаты кластеризации по методам k-средних и k-медиан.

3.3. Контрольные вопросы:

1. Понятие кластеризации.
2. Классификация методов кластеризации.
3. Перечислите известные вам меры расстояния.
4. Метрики качества кластеризации. Силуэт.
5. Сущность метода k-средних.
6. Сущность метода k-медиан.

4 НАИВНЫЙ БАЙЕСОВСКИЙ КЛАССИФИКАТОР

4.1 Теоретические сведения

В основе NBC (Naive Bayes Classifier) лежит теорема Байеса:

$$P(c/d) = \frac{P(d/c)P(c)}{P(d)}$$

где $P(c/d)$ – вероятность что документ d принадлежит классу c, именно её нам надо рассчитать;

$P(d/c)$ – вероятность встретить документ d среди всех документов класса c;

$P(c)$ – безусловная вероятность встретить документ класса c в корпусе документов;

$P(d)$ – безусловная вероятность документа d в корпусе документов [8].

Её смысл можно выразить следующим образом. Теорема Байеса позволяет переставить местами причину и следствие. Зная с какой вероятностью, причина приводит к некоему событию, эта теорема позволяет рассчитать вероятность того, что именно эта причина привела к наблюдаемому событию.

Существует два разных подхода к NBC, которые дают разные результаты: мультиномиальный (multinomial) и многомерный (multivariate). Разница особенно отчётливо проявляется в классификации текстов. Она заключается в том, как именно порождается документ (это называется генеративной моделью).

В **многомерной** модели документ – это вектор бинарных атрибутов, показывающих, встретилось ли в документе то или иное слово. Когда мы подсчитываем правдоподобие документа, мы перемножаем вероятности того, что встретилось каждое слово из документа и вероятности того, что не встретилось каждое (словарное) слово, которое не встретилось. Получается модель многомерных испытаний Бернулли. Наивное предположение в том, что события «встретилось ли слово» предполагаются независимыми. Для применения требуется зафиксировать словарь, а количество повторений каждого слова теряется.

В **мультиномиальной** модели документ – это последовательность событий. Каждое событие – это случайный выбор одного слова из того самого «bag of words». Когда мы подсчитываем правдоподобие документа, мы перемножаем вероятности того, что мы достали из мешка те самые слова, которые встретились в документе. Наивное предположение в том, что мы достаём из мешка разные слова независимо друг от друга. Получается мультиномиальная генеративная модель, которая учитывает количество повторений каждого слова, но не учитывает, каких слов нет в документе.

Цель классификации состоит в том, чтобы понять к какому классу принадлежит документ, поэтому нам нужна не сама вероятность, а наиболее вероятный класс. Байесовский классификатор использует оценку апостериорного максимума (Maximum a posteriori estimation) для определения наиболее вероятного класса. Можно сказать, что это класс с максимальной вероятностью:

$$c_{map} = \arg\max_{c \in C} \frac{P(d/c)P(c)}{P(d)}$$

То есть надо рассчитать вероятность для всех классов и выбрать тот класс, который обладает максимальной вероятностью. Обратите внимание, знаменатель (вероятность документа) является константой и никак не может повлиять на ранжирование классов, поэтому в нашей задаче мы можем его игнорировать.

Соответственно:

$$c_{map} = \arg\max_{c \in C}[\, P(d/c)P(c)] \tag{4.1}$$

Далее делается предположение условной независимости. Байесовский классификатор представляет документ как набор слов, вероятности которых условно не зависят друг от друга. Этот подход иногда еще называется **bag of words model.** Исходя из этого предположения условная вероятность документа аппроксимируется произведением условных вероятностей всех слов входящих в документ.

$$P(d/c) \approx P(w_1/c)P(w_2/c)...P(w_n/c) \approx \prod_{i=1}^{n} P(w_i/c) \tag{4.2}$$

Этот подход также называется Unigram Language Model.

Подставив полученное выражение (4.2) в формулу (4.1) мы получим:

$$c_{map} = \arg\max_{c \in C}[\prod_{i=1}^{n} P(w_i/c)P(c)] \tag{4.3}$$

Проблема арифметического переполнения.

При достаточно большой длине документа придется перемножать большое количество очень маленьких чисел. Для того чтобы при этом избежать арифметического переполнения снизу зачастую пользуются свойством логарифма произведения: Логарифм произведения двух положительных чисел равен сумме логарифмов этих чисел. Переписываем формулу (4.3) с использованием свойства логарифма:

$$c_{map} = \arg\max_{c \in C}[\sum_{i=1}^{n} \log P(w_i/c) + \log P(c)] \tag{4.4}$$

Основание логарифма в данном случае не имеет значения. Вы можете использовать как натуральный, так и любой другой логарифм.

Оценка параметров Байесовской модели.

Оценка вероятностей Pc и $P(w_i/c)$ осуществляется на обучающей выборке. Вероятность класса мы можем оценить как:

$$P(c) = \frac{D_c}{D} \tag{4.5}$$

где D_c – количество документов, принадлежащих классу c, а D – общее количество документов в обучающей выборке.

Оценка вероятности слова в классе может делаться несколькими путями. Например, по multinomial bayes model (4.6).

$$P(w_i/c) = \frac{W_{ic}}{\sum_{i'' \in V} W_{i'c}}, \tag{4.6}$$

где W_{ic} количество раз сколько i-ое слово встречается в документах класса c;

V – словарь корпуса документов (список всех уникальных слов).

Другими словами, числитель описывает, сколько раз слово встречается в документах класса (включая повторы), а знаменатель – это суммарное количество слов во всех документах этого класса.

Проблема неизвестных слов.

По формуле (4.6), если на этапе классификации вам встретится слово, которого вы не видели на этапе обучения, то значения W_{ic}, а следственно и $P(w_i / c)$ будут равны нулю. Это приведет к тому, что документ с этим словом нельзя будет классифицировать, так как он будет иметь нулевую вероятность по всем классам. Типичным решением проблемы неизвестных слов является аддитивное сглаживание (сглаживание Лапласа). Идея заключается в том, что мы предполагаем, будто видели каждое слово на один раз больше, то есть прибавляем единицу к частоте каждого слова.

$$P(w_i / c) = \frac{W_{ic} + 1}{\sum_{i'' \in V} (W_{i'c} + 1)} = \frac{W_{ic} + 1}{|V| + \sum_{i'' \in V} W_{i'c}} \tag{4.7}$$

Подставив выбранные нами оценки в формулу (4.4), получаем окончательную формулу (4.8), по которой происходит байесовская классификация.

$$c_{map} = \arg\max_{c \in C} [\sum_{i=1}^{n} \log \frac{W_{ic} + 1}{|V| + \sum_{i'' \in V} W_{i'c}} + \log \frac{D_c}{D}] \tag{4.8}$$

Реализация классификатора.

Для реализации Байесовского классификатора необходима обучающая выборка, в которой проставлены соответствия между текстовыми документами и их классами. Затем нам необходимо собрать следующую статистику из выборки, которая будет использоваться на этапе классификации:

– относительные частоты классов в корпусе документов, то есть, как часто встречаются документы того или иного класса;

– суммарное количество слов в документах каждого класса;

– относительные частоты слов в пределах каждого класса;

– размер словаря выборки. Количество уникальных слов в выборке.

Совокупность этой информации мы будем называть моделью классификатора. Затем на этапе классификации необходимо для каждого класса рассчитать значение выражения (4.9) и выбрать класс с максимальным значением.

$$\sum_{i \in Q}^{n} \log \frac{W_{ic} + 1}{|V| + L_c} + \log \frac{D_c}{D}] \tag{4.9}$$

где D_c – количество документов в обучающей выборке принадлежащих классу c;

D – общее количество документов в обучающей выборке;

$|V|$ – количество уникальных слов во всех документах обучающей выборки;

L_c – суммарное количество слов в документах класса c в обучающей выборке;

W_{ic} — сколько раз i-ое слово встречалось в документах класса c в обучающей выборке;

Q – множество слов классифицируемого документа (включая повторы).

Формирование вероятностного пространства.

В простейшем случае выбирается класс, который получил максимальную оценку. Но если, например, необходимо, например, помечать сообщение как спам только если соответствующая вероятность больше 80%, то сравнение логарифмических оценок ничего не даст. Оценки, которые выдает данный алгоритм не удовлетворяют двум формальным свойствам, которым должны удовлетворять все вероятностные оценки:

– они все должны быть в диапазоне от нуля до единицы;

– их сумма должна быть равна единице.

Для того чтобы решить эту задачу, необходимо из логарифмических оценок сформировать вероятностное пространство. А именно: избавиться от логарифмов и нормировать сумму по единице по формуле:

$$P(c/d) = \frac{e^{q_c}}{\sum_{c' \in C} e^{q_{c'}}} \tag{4.10}$$

Здесь q_c — это логарифмическая оценка алгоритма для класса c, а возведение e (основания натурального логарифма) в степень оценки используется для того чтобы избавиться от логарифма ($a^{\log_a x} = x$). Таким образом, если вы в расчётах использовали не натуральный логарифм, а десятичный, необходимо использовать не e, а число 10.

4.2 Лабораторная работа. Задание и порядок выполнения работы

Цель лабораторной работы: научиться создавать модель наивного байесовского классификатора

Задание. Используя Excel или Calc, проведите классификацию твитов по принадлежности к определенному классу.

Исходные данные находятся в файле (файлы выдаются по вариантам). На первом листе содержатся твиты, относящиеся относится к первому классу (AboutApp), на втором листе – ко второму классу (AboutOther). Третий лист (TestTweets) содержит тестовый набор твитов.

Порядок выполнения работы.

Далее будет описан порядок выполнения работы, пример выполнения приводится на основе данных из [4]. В примере приведены данные о твитах, содержащих упоминание слова Mandrill. На первом листе (AboutMandrillApp) твиты относятся к сервису Mandrill, на втором листе (AboutOther) – нет. Третий лист (TestTweets) содержит тестовый набор твитов. Mandrill – это «дочерний» сервис MailChimp, предназначенный для отправки транзакционных писем, т.е. писем, уведомляющих пользователя об определённых событиях: регистрации, смене пароля, оформлении заказа, оплате счетов и т.п.

Для выполнения работы выполните следующие этапы.

1. Осуществите токенизацию твитов на листах AboutApp и AboutOther. Для этого.

1.1. Преобразуйте все буквы в строчные в столбце B2. (функция СТРОЧН).

1.2. Уберите лишнюю пунктуацию, для этого последовательно в ячейках C2-H2 замените знаки «. », «: », «?», «!», «;» и «,» на пробелы. (функция ПОДСТАВИТЬ). При этом обратите внимание точку и двоеточие нужно искать с пробелом после них.

1.3. Разделите каждый твит на токены.

1.3.1. Создайте два новых листа AppTokens и OtherTokens.

1.3.2. Предполагаем, что каждый твит содержит не более 30 слов. Соответственно вам нужно 30*150=4500 строк для того чтобы записать все слова. Озаглавьте ячейку A1 Tweet, выделите диапазон A2:A4501 и с помощью специальной вставки вставьте значения твитов из соответствующего столбца H.

1.3.3. В столбце B нужно найти позиции пробелов в твитах. Назовите стобец Space Position. Введите для первых 150 твитов начальное значение 0. Далее (начиная с повтора набора твитов, строка 152) рассчитайте положение следующего пробела, при этом предусмотрите проверку на ошибку для случая, если твит содержкит менее 30 слов. Например, для B152 формула будет иметь вид:

=ЕСЛИОШИБКА(НАЙТИ(" ";A152;B2+1);ДЛСТР(A152)+1)

1.3.4. В столбце C (Token) извлеките токены с помощью фукнции ПСТР. Предусмотрите проверку на ошибку для коротких твитов. Если ошибка есть, замените этот токен, например на «.». Например, для C2 формула будет иметь вид:

=ЕСЛИОШИБКА(ПСТР(A2;B2+1;B152-B2-1);".")

1.3.5. В стоблце D (Length) посчитайте длину токена (функция ДЛСТР).

Вид листа после выполнения этой части задания представлен на рис.4.1.

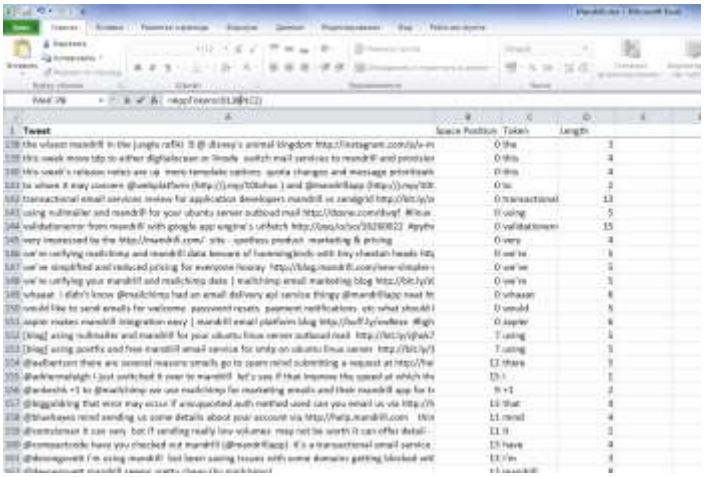


Рисунок 4.1 – Вид листа AppToken

2. Рассчитайте условную вероятность каждого токена.

2.1. Выделите на листе AppTokens область с токенами длиной (C1:D4501) и создайте сводную таблицу на лист AppTokensProbabality. В конструкторе сводных таблиц отфильтруйте токены по длине более 3 (чтобы избавиться от коротких несмысловых токенов) и расположите их по горизонтали. Затем в окне значений установите значение для подсчета количества каждого токена. В результате вы получите перечень всех токенов в твитах с указанием их количества.

2.2. Осуществите дополнительное сглаживание. Для этого в столбце C (Заголовок Add One To Everything) прибавьте к количеству каждого токена единицу. Найдите сумму числа токенов после сглаживания (после последнего числа токенов).

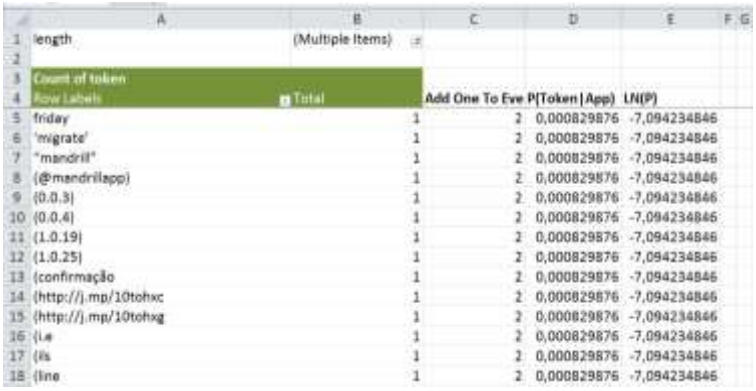
2.3. В столбце D (название P(Token|App) рассчитайте вероятность токена по формуле:

$$P_i = \frac{N_i}{\sum_i N_i}$$

где N_i – количество i-того токена;

P_i – вероятность i-того токена.

2.4. Найдите в столбце E (название — LN(P)) натуральные логарифмы вероятностей. Вид листа AppTokensProbabality после расчета вероятностей представлен на рис.4.2.



	A	B	C	D	E	F	G
1	length	(Multiple Items)					
2							
3	Count of token						
4	Row Labels	Total	Add One To Eve	P(Token|App)	LN(P)		
5	friday	1	2	0,000829876	-7,094234846		
6	'migrate'	1	2	0,000829876	-7,094234846		
7	"mandrill"	1	2	0,000829876	-7,094234846		
8	(@mandrillapp)	1	2	0,000829876	-7,094234846		
9	(0.0.3)	1	2	0,000829876	-7,094234846		
10	(0.0.4)	1	2	0,000829876	-7,094234846		
11	(1.0.19)	1	2	0,000829876	-7,094234846		
12	(1.0.25)	1	2	0,000829876	-7,094234846		
13	(confirmação	1	2	0,000829876	-7,094234846		
14	(http://).mp/10tohxc	1	2	0,000829876	-7,094234846		
15	(http://).mp/10tohxg	1	2	0,000829876	-7,094234846		
16	(i.e	1	2	0,000829876	-7,094234846		
17	(is	1	2	0,000829876	-7,094234846		
18	(line	1	2	0,000829876	-7,094234846		

Рисунок 4.2 – Вид листа AppTokensProbabality

2.5. Аналогично получаем лист OtherTokensProbability, по твитам, относящимся ко второму классу.

Таким образом вы получите модель наивного байесовского классификатора, содержащую две таблицы с условными вероятностями.

3. Тестирование модели.

3.1. В столбцах D – J осуществите обработку текста твитов (аналогично п.1.1, 1.2)

3.2. Для этого создайте лист TestPredictions и вставьте в него столбцы Number и Class из TestTweets. Столбец C назовите Prediction. В нем будете размещать предполагаемые значения классов. В столбец D скопируйте обработанные твиты с листа TestTweets.

3.3. Извлеките токены. Так как, в отличие от таблиц вероятностей, нет необходимости комбинировать токены по всем твитам, токенизацию можно осуществить боле простым способом.

Выделите твиты D2:D21 и выберите «Текст по столбцам» в меню «Данные». В Мастере текстов выберите «С разделителями», в качестве разделителей выберите знаки табуляции и пробела, «Считать последовательные разделители одним», ограничитель строк установите на «нет». В результате твиты будут разбросаны по столбцам в виде отдельных токенов.

3.4. Рассчитайте оценку отношения токенов к обоим классам.

3.4.1. Начиная со столбца D, строка 25 рассчитайте оценку принадлежности токенов к первому классу. Используйте функцию ВПР. Вы должны найти соответствующий токен с листа TestPredictions на листе AppTokensProbability в столбце A и взять значение соответствующей вероятности из столбца E.

Но важно обработать следующие условия:

– вероятность редких слов (отсутствующих в классификаторе) необходимо принять равной LN(1/общее число токенов на листе AppTokensProbability), используйте функцию ЕНД;

– вероятность коротких токенов (число знаков меньше либо равно 3) следует принять равной 0.

Например, для ячейки D25 формула будет иметь вид:

=ЕСЛИ(ДЛСТР(D2)<=3;0;ЕСЛИ(ЕНД(ВПР(D2;$AppTokensProbability.$A$5:$E$827;5;0));LN(1/$AppTokensProbability.$C$828);ВПР(D2;$AppTokensProbability.$A$5:$E$827;5;0)))

3.4.2. Просуммируйте в столбце C оценки вероятности по каждой строке.

Вид листа TestPredictions после выполнения этих операций представлен на рис. 4.3.

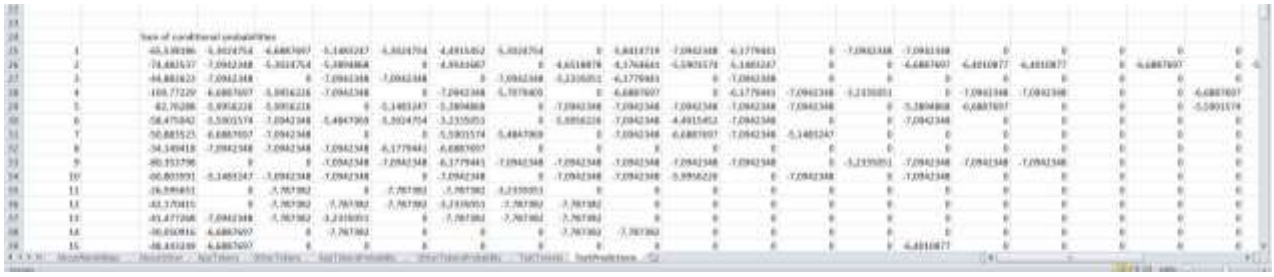


Рисунок 4.3– Вид листа TestPredictions после расчета оценок принадлежности твитов к первому классу

3.4.3. Начиная с ячейки D48 рассчитайте оценки вероятности отношения токенов к другим твитам.

3.5. Проклассифицируйте твиты в столбце C (диапазон C1:C21). Нужно сравнить полученные оценки принадлежности к первому и второму классу. Наибольшее значение вероятности относит твит к соответствующему классу. Используйте фукнцию ЕСЛИ. Пример результата представлен на рис.4.4.

Сделайте выводы.

	A	B	C	D	E	F	G	H	I	J	K	
1	Number	Class	Prediction	Tokens								
2	1	APP	APP	just	love	@mandrillap	transactional	email	service	-	http://mand	sorry
3	2	APP	APP	@rossdeane	mind	submitting	a	request	at	http://help.m	with	accou
4	3	APP	APP	@veroapp	any	chance	you'll	be	adding	mandrill	support	to
5	4	APP	APP	@elie_	@camj59	jparle	de	relai	smtp		1 million	de
6	5	APP	APP	would	like	to	send	emails	for	welcome	password	reset
7	6	APP	APP	from	coworker	about	using	mandrill	"i	would	entrust	email
8	7	APP	APP	@mandrill	realised	i	did	that	about		5 seconds	after
9	8	APP	APP	holy	shit	it's	here	http://www.mandrill.com/				
10	9	APP	APP	our	new	subscriber	profile	page	activity	timeline	aggregate	enga
11	10	APP	APP	@mandrillap	increases	scalability	(	http://bit.ly/*)		then	decreases	pricir
12	11	OTHER	OTHER	the	beets	rt	@missmya	#nameanam	mandrill			
13	12	OTHER	OTHER	rt	@luissandov	fernando	vargas	mandrill	mexican	pride	mma	
14	13	OTHER	OTHER	photo	oculi-ds	mandrill	by	natalie	manuel	http://tmblr.co/zjqanxhdswlr		
15	14	OTHER	OTHER	@mandrill	me	neither	we	can	be	:sadpanda	together	:(
16	15	OTHER	OTHER	@mandrill	n	/	(	k	*	(	n	-
17	16	OTHER	OTHER	megaman	x	-	spark	mandrill	acapella	http://youtu	@youtube さんか	
18	17	OTHER	OTHER	@angeluser	storm	eagle	ftw	nomás	no	dejes	que	se
19	18	OTHER	OTHER	gostei	de	um	video	@youtube	http://youtu	aspark	–	manc
20	19	OTHER	APP	what	is	2-year-old	mandrill	jj	thinking	in	this	pic
21	20	OTHER	OTHER	120	years	of	moscow	zoo	-	mandrill	-	nocta
22												

Рисунок 4.4 – Вид листа TestPredictions после классификации

4.3. Контрольные вопросы

1. Смысл теоремы Байеса.
2. Многомерная и мультиноминальная модель наивного Байесовского классификатора.
3. Для чего используется оценка апостериорного максимума.
4. Предположение условной независимости.
5. Проблема арифметического переполнения.
6. Оценка параметров Байесовской модели.
7. Проблема неизвестных слов.
8. Методика реализации наивного Байесовского классификатора.
9. Формирование вероятностного пространства.

5 КЛАСТЕРИЗАЦИЯ НА ОСНОВЕ СЕТЕВЫХ ГРАФОВ

5.1 Теоретические сведения

Сетевой граф — это набор объектов (предметов), называемых вершинами графа, которые соединены друг с другом ребрами (или связями). Социальные сети, такие как Facebook, содержат много данных, которые легко можно объединить в сетевой граф [9].

Кроме визуализации граф можно представить в виде матрицы смежности. Матрица смежности – это таблица вершин, заполненная 0 и 1, где 1 в конкретной ячейке означает «ребро», а 0 — «эти вершины не связаны» (рис. 5.1 [4]). Ребрам можно добавить числовые значения, например, вместимость самолета. Матрица смежности со значениями также называется матрицей мер конвергенции.

Степень вершины – количество ребер, связанных с вершиной.

	A	B	C	D	E	F	G
1		Ross	Rachel	Chandler	Monica	Joey	Phoebe
2	Ross		1				
3	Rachel	1		1		1	
4	Chandler		1		1		1
5	Monica			1			
6	Joey		1				
7	Phoebe			1			

Рисунок 5.1 – Матрица смежности для героев сериала «Друзья»

Задачей сетевого анализа является «разделить» всю сеть на некие кластеры, группы и определить принадлежность каждой вершины (объекты) к тому или иному кластеру. Заметим, что мы будем искать только один кластер для каждого объекта, хотя в сети могут существовать и вероятнее всего существуют объекты, которые принадлежат одновременно к разным группам, сообществам.

Сообщество – с содержательной точки зрения это группа вершин сети, участники которой связаны друг с другом значительно теснее, чем с остальными вершинами сети. На изображении сети такие группы выглядят как «сгустки» вершин, соединённые друг с другом множеством связей. С формальной точки зрения можно говорить, что «структура графа образована сообществами, если он отличается от случайного графа».

В случайном графе связи между вершинами распределены относительно равномерно, поэтому если применительно к изучаемому графу обнаруживается, что это не так, то можно говорить о существовании в данной сети сообществ. Случайный граф выступает здесь нулевой моделью, с которой сравнивают изучаемый граф.

Модульность (modularity) – это функция, позволяющая оценить «качество разделения графа на кластеры». Согласно этой функции «субграф представляет собой сообщество, если число ребер внутри субграфа превышает ожидаемое число внутренних ребер, которое этот субграф имел бы в нулевой модели». Нулевой моделью здесь предполагается, что ожидаемая последовательность степеней соответствует реальной последовательности степеней графа.

Модульность является «одновременно глобальным критерием определения сообщества, функцией качества и ключевым ингредиентом наиболее популярных методов кластеризации графа». Алгоритмов выделения сообществ, в том числе использующих функцию модульности, в настоящее время разработано много. Изучим один из них.

Алгоритм «edge.betweenness.community» [9].

В первую очередь надо отметить, что данный алгоритм был первым, который базируется на идее модульности. Как пишут сами авторы – М. Ньюман и М. Гирван, особенности их алгоритма заключаются в том, что происходит итеративное исключение ребер с наибольшей промежуточностью, и после каждого исключенного ребра значения промежуточности пересчитываются. Кроме того, они ввели функцию модульности для оценки качества выделения сообществ. Авторы утверждают, что ребра с наибольшей промежуточностью находятся именно на границах между сообществами, тогда как ребра с малой промежуточностью – внутри сообществ. Промежуточность авторами определяется на основе геодезического (то есть кратчайшего) пути между вершинами, они рассматривают и возможности вычисления промежуточности случайного блуждания (random walk betweenness) и промежуточности тока (current flow betweenness), однако для практических целей рекомендуют первый, наименее требовательный с точки зрения вычислительной мощности вариант, дающий результаты, незначительно отличающиеся от других. Что касается модульности, то Ньюман и Гирван говорят о ней как о «мере качества определённого деления сети». Она измеряет фракции ребер в сети, которые соединяют вершины того же типа (т.е. ребра внутри сообщества) минус ожидаемый уровень того же количественного показателя в сети с таким же делением на сообщества, но случайно соединёнными вершинами. Если количество внутренних ребер не более чем случайно, то Q=0. Величины, близкие к Q=1, которая является максимумом, отражают сильную структуру сообществ». На практике, как пишут авторы, величина модульности обычно располагается в интервале между 0,3 и 0,7. Модульность подсчитывается для каждой итерации, т.е. при каждом удалении ребра, и отслеживается момент, когда она достигает наивысшей точки. Эта точка свидетельствует о наилучшем делении сети на сообщества.

Основные этапы кластеризации

1. Построение матрицы смежности графа.

2. Определение близости по косинусу вершин графа.

3. Удаление малозначимых ребер.

Для придания данным осмысленности существуют две популярные техники удаления малозначимых ребер из сетевых графов:

− граф r-окрестности: оставляем только ребра определенной толщины; например, при r = 0,5 останутся только ребра со значениями больше или равно 0,5;

− граф k ближайших соседей: определяем максимальное число ребер, исходящих из одной вершины; например, при k = 5, при каждой вершине останется не более 5 ребер с наибольшими значениями.

4. Модульная максимизация.

Алгоритм использует отношения между вершинами на графе, чтобы делать предположения об их принадлежности к той или иной группе. Применяя модульную максимизацию, вы даете себе одно «очко» каждый раз, когда группируете в один кластер две вершины, имеющие общее ребро в матрице. И получаете ноль очков, если группируете две несвязанные вершины. А также начисляется штраф. Модульная максимизация основывает свои штрафы на следующем утверждении: если убрать все ребра графа, а затем соединить вершины случайным образом (учитывая степени вершин), сколько бы в этом случае ребер получилось между двумя вершинами? Вот это предполагаемое число и есть «штрафное».

Модульность определения групп — это просто сумма всех очков для пар вершин, помещенных в одну и ту же группу, разделенная на сумму всех степеней вершин графа. Деление на сумму степеней сохраняет максимальную модульность в пределах 1, вне зависимости от размеров графа, что облегчает сравнение разных графов между собой.

5.2. Лабораторная работа. Задание и порядок выполнения работы

Цель лабораторной работы: научиться проводить кластерный анализ на основе сетевых графов.

Используя Excel или Calc проведите кластеризацию клиентской базы компании, использую модульную максимизацию сетевых графов.

Исходные данные находятся в файле для лабораторной работы 3 (файлы выдаются по вариантам). В ходе лабораторной работы № 3 те же исходные данные были обработаны кластерным анализом с использованием метода k-средних и k-медиан.

Ход работы.

Далее будет описан порядок выполнения работы, пример выполнения приводится на основе данных из [4]. В примере в файле представлена сводная таблица, содержащая данные о разосланных предложениях по продаже вина и о совершенных сделках по каждому клиенту. Если покупатель откликнулся на предложение, в соответствующей ячейке матрицы проставляется единица. Для наглядности эта область матрицы подсвечена условным форматированием (рис.5.2).

	A	B	C	D	E	F	G	H	I	J	K
1	№ предложения	Период	Сорт	Минимальное количество, кг	Скидка, %	Происхождение	После пика сезона	Adams	Allen	Anders	Bailey
2	1	January	Malbec	72	56	France	FALSE				
3	2	January	Pinot Noir	72	17	France	FALSE				
4	3	February	Espumante	144	32	Oregon	TRUE				
5	4	February	Champagne	72	48	France	TRUE				
6	5	February	Cabernet Sauvignon	144	44	New Zealand	TRUE				
7	6	March	Prosecco	144	86	Chile	FALSE				
8	7	March	Prosecco	6	40	Australia	TRUE				1
9	8	March	Espumante	6	45	South Africa	FALSE				
10	9	April	Chardonnay	144	57	Chile	FALSE		1		
11	10	April	Prosecco	72	52	California	FALSE				
12	11	May	Champagne	72	85	France	FALSE				
13	12	May	Prosecco	72	83	Australia	FALSE				
14	13	May	Merlot	6	43	Chile	FALSE				

Рисунок 5.2 – Матрица отклика покупателей на предложения

Для выполнения работы выполните следующие этапы.

1. На листе Similarity создайте матрицу близости различных покупателей между собой (рис.5.3). Используйте для этого близость по косинусу. Процесс создания матрицы подробно описан в методических указаниях для лабораторной работы № 3.

	A	B	C	D	E	F	G	H
1			0	1	2	3	4	
2			Adams	Allen	Anderson	Bailey	Baker	Barnes
3	0	Adams						
4	1	Allen						
5	2	Anderson						
6	3	Bailey						
7	4	Baker						
8	5	Barnes						
9	6	Bell						

Рисунок 5.3 – Пустая таблица для матрицы близости косинусов

Близость косинусов между бинарными заказами двух покупателей определяется по формуле:

$$S_{\cos} = \frac{N_s}{\sqrt{N_1}\sqrt{N_2}}$$

где: N_s- количество совпадающих заказов по двум векторам,

N_1 - количество заказов по первому вектору,

N_2 - количество заказов по второму вектору.

Не забывайте про функции СМЕЩ. Так, для расчета близости косинусов в ячейке С3 (Adams–Adams) используется формула:

=СУММПРОИЗВ(СМЕЩ(Matrix!$H$2:$H$33;0;Similarity!C$1);СМЕЩ(Matrix!$H$2:$H$33;0;Similarity!$A3))/(КОРЕНЬ(СУММ(СМЕЩ(Matrix!$H$2:$H$33;0;Similarity!C$1)))*КОРЕНЬ(СУММ(СМЕЩ(Matrix!$H$2:$H$33;0;Similarity!$A3)))).

2. Уберите единицы в диагонали матрицы. Добавьте соответствующее условие ЕСЛИ ТО. Пример рассчитанной матрицы с условным форматированием представлен на рис.5.4.

	A	B	C	D	E	F	G	H
1			0	1	2	3	4	5
2			Adams	Allen	Anderson	Bailey	Baker	Barnes
3	0	Adams	0,000000	0,000000	0,000000	0,408248	0,000000	0,000000
4	1	Allen	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000
5	2	Anderson	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000
6	3	Bailey	0,408248	0,000000	0,000000	0,000000	0,353553	0,000000
7	4	Baker	0,000000	0,000000	0,000000	0,353553	0,000000	0,500000
8	5	Barnes	0,000000	0,000000	0,000000	0,000000	0,500000	0,000000

Рисунок 5.4 – Заполненная матрица близости косинусов покупателей

3. Используя граф r-окрестности исключите малозначимые ребра.

3.1. Для начала нужно определить значение r. Воспользуемся принципом Парето: 20% связей определяют 80% сути графа. В ячейке С104 рассчитайте **Edge Count -** общее число ребер с близостью больше 0 (функция СЧЁТЕСЛИ).

Используя в ячейке С105, используя формулу НАИБОЛЬШИЙ определите мощность ребра, соответствующую 80-ому перцентилю (**80th Ptile)**. Если 80-й прецентиль составит, например 0,4, то 20% самых мощных ребер будут попадать в диапазон 0,4–1.

3.2. На листе r-NeighborhoodAdj создайте новую матрицу смежности, используя условие: если значение близости из листа Similarity больше либо равно рассчитанному значению 80-го перцентиля (**80th Ptile)**, то присваиваем соответствующему ребру на листе r-NeighborhoodAdj значение 1, в противном случае – 0 (рис.5.5).

	A	B	C	D	E	F	G	H
1		Adams	Allen	Anderson	Bailey	Baker	Barnes	Bell
2	Adams	0	0	0	0	0	0	0
3	Allen	0	0	0	0	0	0	0
4	Anderson	0	0	0	0	0	0	1
5	Bailey	0	0	0	0	0	0	0
6	Baker	0	0	0	0	0	1	0
7	Barnes	0	0	0	0	1	0	0
8	Bell	0	0	1	0	0	0	0
9	Bennett	0	0	0	0	0	0	0
10	Brooks	0	0	0	0	0	0	0

Рисунок 5.5 – Матрица смежности (граф r-окрестности)

4. Постройте таблицу модульности.

4.1. Рассчитайте степени вершин графов по столбцам и строкам (количество возможных ребер, исходящих из каждой вершины), а также общее количество ребер (рис.5.6).

	A	CT	CU	CV	CW	CX
1		Wilson	Wood	Wright	Young	Пеньков
93	Walker	0	0	0	0	14
94	Ward	0	0	1	0	5
95	Watson	0	0	0	0	16
96	White	0	0	0	0	3
97	Williams	0	0	0	0	3
98	Wilson	0	0	0	0	18
99	Wood	0	0	0	0	5
100	Wright	0	0	0	0	4
101	Young	0	0	0	0	4
102	Пеньков	18	5	4	4	858

Рисунок 5.6 – Подсчет степеней вершин на графе r-окрестности

4.2. Создайте лист Scores, куда вставьте имена клиентов в строку 1 и столбец A.

4.3. Рассчитайте модульность для каждой пары клиентов (для каждого возможного ребра) по формуле:

$$M = P - F = P - \frac{D_1 D_2}{N_s}$$

где M – значение модульности;

P – фактическое наличие ребра (значение равно 1 или 0 по таблице смежности из листа r-NeighborhoodAdj);

F – штраф;

D_1 – степень вершины 1 (клиента 1);

D_2 – степень вершины 2 (клиента 2);

Ns – общее количество ребер.

Пример рассчитанной таблицы представлен на рис.5.7.

	A	B	C	D	E	F	G	H	I
1		Adams	Allen	Anderson	Bailey	Baker	Barnes	Bell	Bennett
2	Adams	-0,2284382	-0,0979021	-0,2121212	-0,2773893	-0,1142191	-0,0979021	-0,2121212	-0,2447
3	Allen	-0,0979021	-0,041958	-0,0909091	-0,1188811	-0,048951	-0,041958	-0,0909091	-0,1048
4	Anderson	-0,2121212	-0,0909091	-0,1969697	-0,2575758	-0,1060606	-0,0909091	0,8030303	-0,2272
5	Bailey	-0,2773893	-0,1188811	-0,2575758	-0,3368298	-0,1386946	-0,1188811	-0,2575758	-0,2972
6	Baker	-0,1142191	-0,048951	-0,1060606	-0,1386946	-0,0571096	0,95104895	-0,1060606	-0,1223
7	Barnes	-0,0979021	-0,041958	-0,0909091	-0,1188811	0,95104895	-0,041958	-0,0909091	-0,1048
8	Bell	-0,2121212	-0,0909091	0,8030303	-0,2575758	-0,1060606	-0,0909091	-0,1969697	-0,2272
9	Bennett	-0,2447552	-0,1048951	-0,2272727	-0,2972028	-0,1223776	-0,1048951	-0,2272727	-0,2622
10	Brooks	-0,1305361	-0,0559441	-0,1212121	-0,1585082	-0,0652681	-0,0559441	-0,1212121	-0,1398
11	Brown	0,75524476	-0,1048951	-0,2272727	0,7027972	-0,1223776	-0,1048951	-0,2272727	-0,2622
12	Butler	-0,032634	-0,013986	-0,030303	-0,039627	-0,016317	-0,013986	-0,030303	-0,034
13	Campbell	-0,1631702	-0,0699301	0,84848485	-0,1981352	-0,0815851	-0,0699301	0,84848485	-0,1748
14	Carter	0,73892774	-0,1118881	-0,2424242	0,68298368	-0,1305361	-0,1118881	-0,2424242	-0,2797
15	Clark	-0,0979021	-0,041958	-0,0909091	-0,1188811	-0,048951	-0,041958	-0,0909091	-0,1048
16	Collins	-0,1142191	-0,048951	-0,1060606	0,86130536	-0,0571096	-0,048951	-0,1060606	-0,1223
17	Cook	-0,2121212	-0,0909091	0,8030303	-0,2575758	-0,1060606	-0,0909091	0,8030303	-0,2272

Рисунок 5.7 – Таблица модульности

5. Проведите кластеризацию, используя иерархический дивизимный алгоритм.

5.1. Начните процесс кластеризации с разбиения графа на 2 группы. Создайте лист Split1 и вставьте клиентов в столбец A. Столбец B назовите Community. В нем будем обозначать принадлежность клиента к группе, поскольку их всего две используйте бинарную переменную 0/1 (0 – означает принадлежность к одной группт, 1 – к другой). Столбцы C, D, E назовите соответственно Score, UB1, UB2.

5.2. Посчитайте принадлежность каждого покупателя к группе. В столбцах D и E вычислите модульность для каждого покупателя при отнесении его к соответствующей группе. Например, если вы поместите Адамса в группу 1, то вам нужно сложить все значения ячеек из строки 2 листа «r-NeighborhoodAdj», которые относятся к другим покупателям, также

попавшим в группу 1. Так как мы присваиваем только 0/1, можно использовать СУММПРОИЗВ для умножения вектора группы на вектор модульности и сложения результата.

Поскольку значения модульностей на листе «r-NeighborhoodAdj» расположены слева направо, а в оптимизационной модели они идут сверху вниз, придется использовать ТРАНСП (что, в свою очередь, означает использование формулы массива): {=СУММПРОИЗВ(B$2:B$101,ТРАНСП(' Scores!B2:CW2))}. Формула умножает значения модульности на значения принадлежности к группе. Остаются только те, которые относятся к членам группы 1, в то время как остальные обращаются в 0. СУММПРОИЗВ складывает их все.

Если Адамс был отнесен к группе 0, нужно всего лишь полученное значение вычесть из 1: {=СУММПРОИЗВ(1-(B$2:B$101),ТРАНСП(Scores!B2:CW2))}. Мы бы могли соединить эти две формулы с помощью функции ЕСЛИ, которая бы проверила принадлежность Адамса к той или иной группе, а затем использовала бы соответствующую формулу для сложения модульностей тех или иных соседей. Но при использовании ЕСЛИ пришлось бы задействовать нелинейную модель в *Поиске решения*). В данном случае максимизация модульности слишком тяжела для нелинейного *Поиска решения*, и он становится неэффективен. Следует сделать задачу линейной.

Обе предыдущие формулы линейны, поэтому можно установить переменную модульности такую, чтобы она была меньше их обеих. Чтобы максимизировать общую модульность, например, модульность Адамса будет стремиться вверх, пока не наткнется на меньшую из ограничивающих формул. Добавим в формулу вторую часть: {=СУММПРОИЗВ(B$2:B$101;ТРАНСП('Scores!B2:CW2))+**(1- B2)*СУММ(ABS(**'Scores!B2:CW2**))**}.

Если Адамс отнесен в группу 1, то эта часть формулы обращается в 0 (из-за умножения на 1-B2). В этом случае формула становится идентичной первой из рассмотренных выше формул. Но если Адамса отнесли в группу 0, то эта формула больше не подходит и ее нужно отключить. Поэтому часть формулы (1-B2)*СУММ(ABS('$Scores.'B2:CW2)) добавляет 1, умноженную на сумму всех абсолютных значений модульности, которые Адамс может получить. Это действие гарантирует, что результат окажется выше, чем у ее перевернутой версии из соседней колонки.

Таким образом для ячейки D2 формула имеет вид: =СУММПРОИЗВ(B$2:B$101;ТРАНСП(Scores!B2:CW2))+(1- B2)*СУММ(ABS(Scores!B2:CW2)).

Для ячейки E2 формула имеет вид:

=СУММПРОИЗВ(1- (B$2:B$101);ТРАНСП(Scores!B3:CW3))+B3*СУММ(ABS(Scores!B3:CW3)).

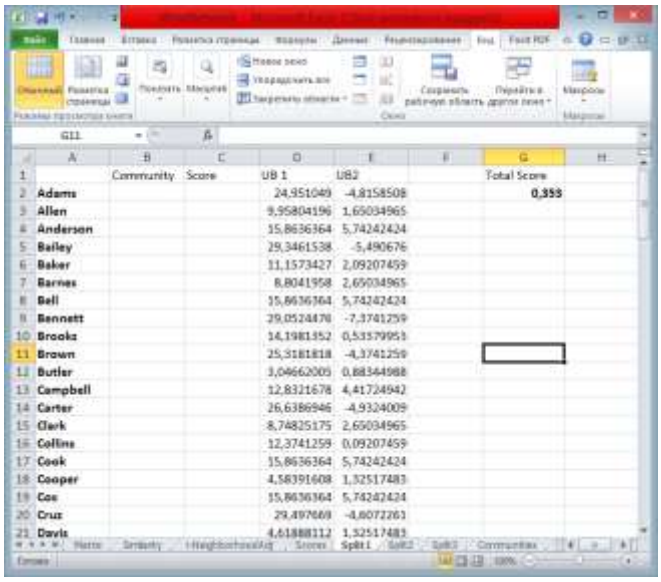
Это заставляет модульность Адамса быть меньше или равной правильному расчету и удаляет другую формулу из рассуждения, завышая ее значение.

Таким образом, столбец C – это столбец модульности, которая будет переменной решения, а в столбцы D и E содержат две формулы в качестве верхних границ модульности. Складывая значения модульности в столбце G2, получаем целевую функцию для максимизации: =СУММ(C2:C101)/'r-NeighborhoodAdj'!CX102.

Лист, подготовленный к оптимизации представлен на рис.5.8.

5.3. Запустите оптимизацию. Откройте *Поиск решения* и отметьте, что вы максимизируете значение модульности графа в ячейке G2 (рис.5.9). Переменные решения — это значения принадлежности к группе в B2:B101 и значения модульности в C2:C101. К значениям принадлежности к группе в B2:B101 нужно добавить условие бинарности. Также необходимо сделать переменные модульности покупателей в столбце C меньше, чем обе верхние границы в столбцах D и E. Также следует сделать все переменные неотрицательными, отметив галочкой эту опцию и выбрать *Поиск решения линейных задач симплекс-методом*.

Кликните кнопку *Параметры* и установите *Максимальное число подзадач* на 15 000. Это гарантирует нам, что *Поиск решения* остановится минут через 20 после начала работы. Нажмите *Ok*, а затем *Найти решение*.



	A	B	C	D	E	F	G	H
1		Community	Score	UB 1	UB2		Total Score	
2	Adams			24,951049	-4,8158508		0,353	
3	Allen			9,95804196	1,65034965			
4	Anderson			15,8636364	5,74242424			
5	Bailey			29,3461538	-5,490676			
6	Baker			11,1573427	2,09207459			
7	Barnes			8,8041958	2,65034965			
8	Bell			15,8636364	5,74242424			
9	Bennett			29,0524476	-7,3741259			
10	Brooks			14,1981352	0,53379953			
11	Brown			25,3181818	-4,3741259			
12	Butler			3,04662005	0,88344988			
13	Campbell			12,8321678	4,41724942			
14	Carter			26,6386946	-4,9324009			
15	Clark			8,74825175	2,65034965			
16	Collins			12,3741259	0,09207459			
17	Cook			15,8636364	5,74242424			
18	Cooper			4,58391608	1,32517483			
19	Cox			15,8636364	5,74242424			
20	Cruz			29,497669	-4,8072261			
21	Davis			4,61888112	1,32517483			

Рисунок 5.8 – Лист, подготовленный к оптимизации

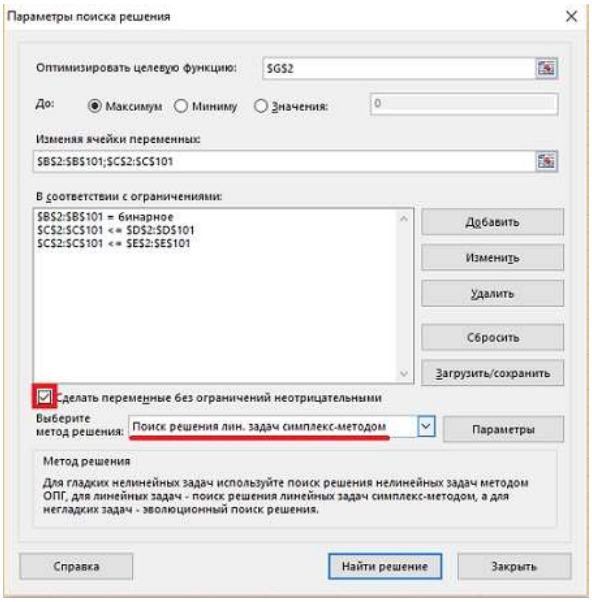
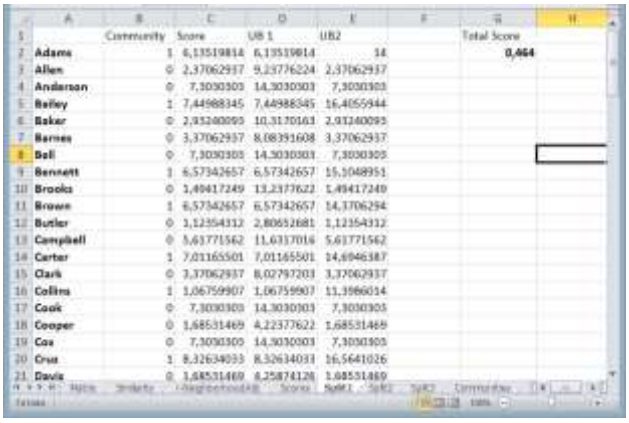


Рисунок 5.9 – Настройка Поиска решения для первого деления графа на два кластера

При работе в Excel 2010 и 2013 то встроенный Поиск решения может не справиться с поставленной задачей и выдавать сообщения, что модель слишком велика. В этом случае нужно установить надстройку OpenSolver (скачайте с сайта https://opensolver.org/ при необходимости запустите её, во вкладке данные появится соответствующей раздел). Ограничьте работу OpenSolver 300 секундами.

Пример решения представлен на рис.5.10. Просматривая столбец B, вы видите, кто оказался в группе 0, а кто — в группе 1. Теперь встает вопрос о том, окончательное ли это решение. Чтобы ответить на этот вопрос, попытайтесь разделить эти группы пополам. Если дальнейшее разделение увеличит модульность, то вы получите четыре кластера.



	A	B	C	D	E	F	G	H
1		Community	Score	UB 1	UB2		Total Score	
2	Adams	1	6,13519814	6,13519814	14		0,464	
3	Allen	0	2,37062937	9,23776224	2,37062937			
4	Anderson	0	7,3030303	14,3030303	7,3030303			
5	Bailey	1	7,44988345	7,44988345	16,4055944			
6	Baker	0	2,93240093	10,3170163	2,93240093			
7	Barnes	0	3,37062937	8,08391608	3,37062937			
8	Bell	0	7,3030303	14,3030303	7,3030303			
9	Bennett	1	6,57342657	6,57342657	15,5048951			
10	Brooks	0	1,49417249	13,2377622	1,49417249			
11	Brown	1	6,57342657	6,57342657	14,3706294			
12	Butler	0	1,12354312	2,80652681	1,12354312			
13	Campbell	0	5,61771562	11,6317016	5,61771562			
14	Carter	1	7,01165501	7,01165501	14,6946387			
15	Clark	0	3,37062937	8,02797203	3,37062937			
16	Collins	1	1,06759907	1,06759907	11,3966014			
17	Cook	0	7,3030303	14,3030303	7,3030303			
18	Cooper	0	1,68531469	4,23377622	1,68531469			
19	Cox	0	7,3030303	14,3030303	7,3030303			
20	Cruz	1	8,32634033	8,32634033	16,5641026			
21	Davis	0	1,68531469	4,25874126	1,68531469			

Рисунок 5.10 – Оптимальное решение для первого деления графа на два кластера

5.4. Скопируйте лист Split1 и переименуйте в Split2. Вставьте новый столбец после B и назовите его Last Run. Скопируйте в него значения из столбца B. Теперь принадлежность к одной группе определяется парой значений в столбцах B и C. Полученные в результате кластеризации новые группы 0 и 1 должны учитывать результаты предыдущей кластеризации. Соответственно, теперь кластеров может быть четыре: 0–0, 0–1, 1–0 и 1–1.

5.5. Измените верхние границы расчета модульности. Например, в E2 используется формула:
=СУММПРОИЗВ(B$2:B$101;ЕСЛИ(C$2:C$101=C2;1;0);ТРАНСП(Scores!B2:CW2))+(1- B2)*
СУММПРОИЗВ(ЕСЛИ(C$2:C$101=C2;1;0);ТРАНСП(ABS(Scores!B2:CW2))).

Логическое выражение ЕСЛИ(C$2:C$101=C2,1,0) предотвращает начисление «очков» Адамсу, пока его соседи не окажутся с ним в одном кластере после первого деления. Здесь можно использовать ЕСЛИ, потому что значения в столбце C больше не являются переменными.

Аналогично, для столбца F пример формулы:
=СУММПРОИЗВ(1- (B$2:B$101);ЕСЛИ(C$2:C$101=C2;1;0);ТРАНСП(Scores!B2:CW2))+B2*
СУММПРОИЗВ(ЕСЛИ(C$2:C$101=C2;1;0);ТРАНСП(ABS(Scores!B2:CW2)))

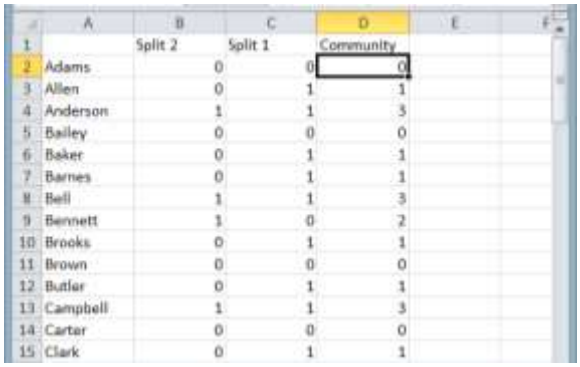
5.6. Запустите оптимизацию. Сделайте вывод об изменении модульности.

5.7. Осуществите дальнейшее деление на группы, чтобы проверить изменение модульности. Если значение не увеличится, значит число кластеров 4, если существенно увеличится – продолжайте алгоритм. Скопируйте лист Split2 и переименуйте в Split3. Вставьте новый столбец после C и назовите его Last Run2, остальные действия аналогичны п.5.4-5.6.

6. Проанализируйте принадлежности к группам подробнее. Последовательность действий далее описана для случая, если максимальная модульность получилась для 4-х кластеров.

6.1. Создайте новый лист Communities, вставьте туда имена покупателей, и номера групп, преобразовав два бинарных столбца (B и C с листа Split2) в один десятеричный с помощью формулы: = =ДВ.В.ДЕС(СЦЕПИТЬ(B2;C2)) (рис.5.11). Получилось четыре кластера с ярлыками от 0 до 3.

6.2. Проанализируйте эти кластеры, выделив наиболее характерные для них предложения (из 32). Создайте лист TopDealsByCluster, возьмите часть таблицы Matrix (диапазон A1:G33), добавьте столбцы H:K, озаглавьте их по номерам кластеров – 0, 1, 2 и 3 (рис.5.12). Найдите количество покупателей для каждого предложения по кластерам (аналогичную таблицу вы уже создавали в лабораторной работе 3). Но нужно использовать формулу СУММПРОИЗВ. Формула для ячейки I2:
=СУММПРОИЗВ(ЕСЛИ($CommunitiesGephi.$B$2:$B$101=$TopDealsByClusterGephi.I$1;1;0)
;ТРАНСП($Matrix.$H2:$DC2)).



	A	B	C	D
1		Split 2	Split 1	Community
2	Adams	0	0	0
3	Allen	0	1	1
4	Anderson	1	1	3
5	Bailey	0	0	0
6	Baker	0	1	1
7	Barnes	0	1	1
8	Bell	1	1	3
9	Bennett	1	0	2
10	Brooks	0	1	1
11	Brown	0	0	0
12	Butler	0	1	1
13	Campbell	1	1	3
14	Carter	0	0	0
15	Clark	0	1	1

Рисунок 5.11 – Итоговые номера кластеров для модульной максимизации

	A	B	C	D	E	F	G	H	I	J	K
1	№ предложения	Период	Сорт	Мин. кол-во, кг	Скидка, %	Происхождение	После пива сезона	0	1	2	3
2	1	January	Malbec	72	56	France	FALSE	0	8	0	2
3	2	January	Pinot Noir	72	17	France	FALSE	0	3	0	7
4	3	February	Espumante	144	32	Oregon	TRUE	0	6	0	0
5	4	February	Champagne	72	48	France	TRUE	0	12	0	0
6	5	February	Cabernet Sauvi	144	44	New Zealanc	TRUE	0	4	0	0
7	6	March	Prosecco	144	86	Chile	FALSE	1	11	0	0
8	7	March	Prosecco	6	40	Australia	TRUE	14	5	0	0
9	8	March	Espumante	6	45	South Africa	FALSE	16	4	0	0
10	9	April	Chardonnay	144	57	Chile	FALSE	0	10	0	0
11	10	April	Prosecco	72	52	California	FALSE	1	5	0	1
12	11	May	Champagne	72	85	France	FALSE	2	11	0	0
13	12	May	Prosecco	72	83	Australia	FALSE	1	3	0	1
14	13	May	Merlot	6	43	Chile	FALSE	6	0	0	0
15	14	June	Merlot	72	64	Chile	FALSE	0	9	0	0
16	15	June	Cabernet Sauvi	144	19	Italy	FALSE	0	6	0	0
17	16	June	Merlot	72	88	California	FALSE	1	3	0	1
18	17	July	Pinot Noir	12	47	Germany	FALSE	0	0	0	7
19	18	July	Espumante	6	50	Oregon	FALSE	13	1	0	0
20	19	July	Champagne	12	66	Germany	FALSE	0	5	0	0
21	20	August	Cabernet Sauvi	72	82	Italy	FALSE	1	5	0	0

Рисунок 5.12 – Распределение предложений по кластерам

6.3. С помощью автофильтрации и сортировки по кластерам, установите наиболее популярные предложения для каждого кластера, охарактеризуйте клиентов каждого кластера.

6.4. Сравните полученное решение с решениями, полученными в ходе лабораторной работы 3.

5.3. Контрольные вопросы

1. Понятие сетевого графа, вершины, ребра.
2. Матрица смежности.
3. Степень вершины графа.
4. Задача сетевого анализа.
5. Понятие сообщества в сетевом анализе.
6. Понятие случайного графа.
7. Агломеративная и дивизимнная кластеризация.
8. Понятие модульности.
9. Алгоритм «edge.betweenness.community».

6 АНСАМБЛИ МОДЕЛЕЙ (БЭГГИНГ И БУСТИНГ)

6.1.Теоретические сведения

Бэггинг – от англ. ***B**ootstrap **agg**regat**ing***, это технология классификации, использующая композиции алгоритмов, каждый из которых обучается независимо. Результат классификации определяется путем голосования. Бэггинг позволяет снизить процент ошибки классификации в случае, когда высока дисперсия ошибки базового метода [10].

Бэггинг – технология классификации, где в отличие от бустинга все элементарные классификаторы обучаются и работают параллельно (независимо друг от друга). Идея заключается в том, что классификаторы не исправляют ошибки друг друга, а компенсируют их при голосовании. Базовые классификаторы должны быть независимыми, это могут быть классификаторы, основанные на разных группах методов или же обученные на независимых наборах данных. Во втором случае можно использовать один и тот же метод.

Бэггинг на подпространствах. Этот алгоритм применяется для классификации многомерных объектов. Рассматриваемый алгоритм помогает добиться качественной классификации в условиях, когда разделить объекты на группы на всем пространстве параметров не представляется возможным. Предлагается разделить пространство характеристик на подмножества объединенных по смыслу параметров. Классификация на каждом подпространстве производится отдельно, затем результаты учитываются в голосовании. В этом случае будет учтен вклад каждой смысловой группы и много повысится вероятность того, что итоговые результаты классификации окажутся более качественными нежели без деления на подпространства, так как параметры, по которым представители разных классов неотличимы, попадут, почти наверняка, не во все группы.

Постановка задачи.

Существует матрица характеристик объекта $X: x_1,...x_n$ m-мерные столбцы с характеристиками n объектов. Необходимо сопоставить каждому вектору параметров метку класса (т.е. существует некоторое отображение $X \to Y$, где $Y = (y_1...y_k)$ y_j – метки классов), на основании известных пар (x_i, y_j) для объектов обучающей выборки.

Алгоритм классификации в технологии бэггинг на подпространствах.

Необходимо разделить пространство параметров на подмножества, то есть каждый объект будет характеризоваться уже не одним m-мерным вектором параметров, а несколькими векторами $(x_{i1}...x_{il})$, причем сумма размерностей этих векторов не может превышать m, то есть подпространства не могут пересекаться. Для этого прибегают к экспертному мнению, эксперт выделяет смысловые подпространства на основании своего опыта.

Производится независимое обучение каждого элементарного классификатора (каждого алгоритма, определенного на своем подпространстве).

Производится классификация основной выборки на каждом из подпространств (также независимо).

Принимается окончательное решение о принадлежности объекта одному из классов. Это можно сделать несколькими разными способами, подробнее описано ниже.

Методы принятия решений.

Окончательное решение о принадлежности объекта классу может приниматься, например, одним из следующих методов:

1. Консенсус: если все элементарные классификаторы присвоили объекту одну и ту же метку, то относим объект к выбранному классу.

2. Простое большинство: консенсус достижим очень редко, поэтому чаще всего используют метод простого большинства. Здесь объекту присваивается метка того класса, который определило для него большинство элементарных классификаторов.

3. Взвешивание классификаторов: если классификаторов четное количество, то голосов может получиться поровну, еще возможно, что для экспертов одна из групп параметров важна в большей степени, тогда прибегают к взвешиванию классификаторов. То есть при голосовании голос классификатора умножается на его вес.

Бустинг (англ. boosting – улучшение) – это процедура последовательного построения композиции алгоритмов машинного обучения, когда каждый следующий алгоритм стремится компенсировать недостатки композиции всех предыдущих алгоритмов. Бустинг представляет собой жадный алгоритм построения композиции алгоритмов. Бустинг является одним из наиболее популярных методов машинного обучения, наряду с нейронными сетями и машинами опорных векторов. Основные причины — простота, универсальность, гибкость (возможность построения различных модификаций), и, главное, высокая обобщающая способность. Бустинг над решающими деревьями считается одним из наиболее эффективных методов с точки зрения качества классификации.

Алгоритм «Случайный лес»

Random forest (с англ. – «случайный лес») [11] – алгоритм машинного обучения, предложенный Лео Брейманоми Адель Катлер, заключающийся в использовании комитета (ансамбля) решающих деревьев. Алгоритм сочетает в себе две основные идеи: метод бэггинга Бреймана, и метод случайных подпространств, предложенный Tin Kam Ho. Алгоритм применяется для задач классификации, регрессии и кластеризации. Основная идея заключается в использовании большого ансамбля решающих деревьев, каждое из которых само по себе даёт очень невысокое качество классификации, но за счёт их большого количества результат получается хорошим.

Алгоритм обучения классификатора.

Пусть обучающая выборка состоит из N образцов, размерность пространства признаков равна M, и задан параметр m (в задачах классификации обычно $m = \sqrt{M}$) как неполное количество признаков для обучения.

Наиболее распространённый способ построения деревьев комитета следующий.

Сгенерируем случайную подвыборку с повторениями размером N из обучающей выборки.

Построим решающее дерево, классифицирующее образцы данной подвыборки, причём в ходе создания очередного узла дерева будем выбирать набор признаков, на основе которых производится разбиение (не из всех M признаков, а лишь из m случайно выбранных). Выбор наилучшего из этих m признаков может осуществляться различными способами. В оригинальном коде Бреймана используется критерий Джини. В некоторых реализациях алгоритма вместо него используется критерий прироста информации.

Классификация объектов проводится путём голосования: каждое дерево комитета относит классифицируемый объект к одному из классов, и побеждает класс, за который проголосовало наибольшее число деревьев.

Оптимальное число деревьев подбирается таким образом, чтобы минимизировать ошибку классификатора на тестовой выборке. В случае её отсутствия, минимизируется оценка ошибки out-of-bag: тех образцов, которые не попали в обучающую подвыборку за счёт повторений (их примерно N/e).

Загрязнение Джини — вероятность неверной маркировки в узле случайно выбранного образца.

Если набор данных T содержит данные n классов, тогда индекс Gini определяется следующим образом.

$$Gini(T) = 1 - \sum_{i=1}^{n} (p_i)^2,$$

где p_i - вероятность (относительная частота) класса i в T.

Если набор *T* разбивается на две части *T1* и *T2* с числом примеров в каждом *N1* и *N2* соответственно, тогда показатель качества разбиения будет равен:

$$Gini_{split}(T) = \frac{N_1}{N} Gini(T_1) + \frac{N_2}{N} Gini(T_2)$$

Наилучшим считается то разбиение, для которого $Gini_{split}(T)$ минимально.

При классификации по бэггинговой модели в качестве вероятности класса принимается среднее значений всех классификаторов.

В бустинговых моделях для выигравшего дерева рассчитывается величина α по формуле

$$\alpha = 0{,}5 \ln \frac{1 - \text{Winning Error}}{\text{Winning Error}}.$$

Значение α используется в бустинговых моделях для прогнозирования: каждый классификатор (дерева) выдает значение α, если зависимый признак имеет требуемое значение и $-\alpha$, если нет. Итоговый прогноз ансамбля моделей – это сумма всех отрицательных и положительных значений. В качестве классификационного значения обычно используется 0, но это значение можно менять по своему усмотрению**.**

6.2. Задание и порядок выполнения работы.

Цель лабораторной работы: научится строить комплексные модели (ансамбли моделей) для классификации покупателей и оценивать параметры качества.

Задание. Используя Excel или Calc с помощью бэггинга и бустинга осуществите классификацию покупателей по заданному признаку.

Исходные данные находятся в файле для лабораторной работы из раздела 2 (файлы выдаются по вариантам). Данные аналогичны данным в лабораторной работе 2.

Постройте бэггинговую и бустинговую модели для классификации покупателей, осуществите классификацию на тестовом наборе данных, оцените качество модели. Сравните результаты моделей бэггинга и бустинга с результатами между собой и с результатами, полученными с помощью регрессионной модели (раздел 2).

Далее будет описан порядок выполнения работы, пример выполнения приводится на основе данных из [4].

Ход работы.

Для выполнения работы выполните следующие этапы.

1. Бэггинговая модель. Формирование набора деревьев решений на основе случайных наборов данных.

1.1. Скопируйте A2:U1002 на новый лист «TD_BAG». Названия признаков копировать не нужно. Скопируйте только их номера из строки 2 (рис.6.1).

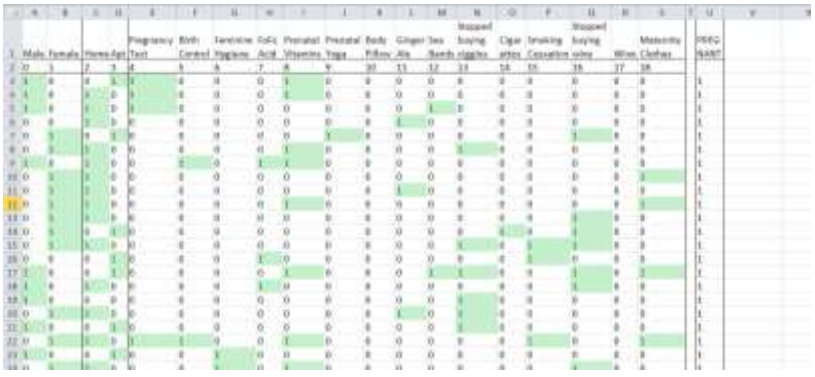
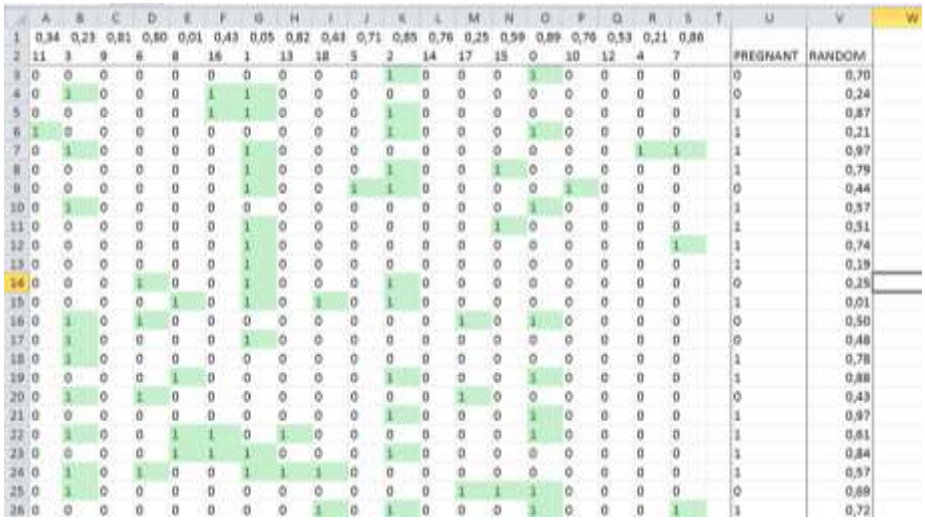


Рисунок 6.1 – Данные о покупателях

1.2. Вставьте пустую строку над номерами признаков, столбец V назовите Random. С помощью функции СЛЧИС() заполните первую строку и столбец V случайными числами (рис. 6.2).



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	0,34	0,23	0,81	0,80	0,01	0,43	0,05	0,82	0,43	0,71	0,85	0,76	0,25	0,59	0,89	0,76	0,53	0,21	0,86				
2	11	3	9	6	8	16	1	13	18	5	2	14	17	15	0	10	12	4	7		PREGNANT	RANDOM	
3	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0		0	0,70	
4	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0		0	0,24	
5	0	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0		1	0,87	
6	1	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0		1	0,21	
7	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1		1	0,97	
8	0	0	0	0	0	0	1	0	0	0	1	0	0	1	0	0	0	0	0		1	0,79	
9	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	1	0	0	0		0	0,44	
10	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0		1	0,57	
11	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0		1	0,51	
12	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1		1	0,74	
13	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0		1	0,19	
14	0	0	0	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0		0	0,25	
15	0	0	0	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0		1	0,01	
16	0	1	0	1	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0		0	0,50	
17	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0		0	0,48	
18	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		1	0,78	
19	0	0	0	0	1	0	0	0	0	0	1	0	0	0	1	0	0	0	0		1	0,88	
20	0	1	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0		0	0,43	
21	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0		1	0,97	
22	0	1	0	0	1	1	0	1	0	0	0	0	0	0	1	0	0	0	0		1	0,61	
23	0	0	0	0	1	1	1	0	0	0	1	0	0	0	0	0	0	0	0		1	0,84	
24	0	1	0	1	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0		1	0,57	
25	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0		0	0,69	
26	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	1		1	0,72	

Рисунок 6.2 – Лист «TD_BAG»

1.3. Создайте случайную выборку переменных и покупателей.

1.3.1. Перемешайте столбцы с помощью окна настраиваемой сортировки. Для этого выделите столбцы, в окне сортировки выберите строку 1 для сортировки слева направо, уберите галку «Мои данные содержат заголовки», в Параметрах выберите «столбцы диапазона».

1.3.2. Перемешайте строки, не включаю строку 1 со случайными числами, но включая столбец с классификационным признаком и столбец со случайными числами. Сортировка сверху вниз. Данные содержат заголовки.

1.3.3. Выберите прямоугольник, образованный первыми четырьмя столбцами и 666 строками. Создайте новый лист «RandomSelection» и настройте его для заполнения данными из выбранного прямоугольника автоматически.

В A1 введите формулу =TD_BAG!A2, а затем скопируйте формулу до D667. В столбец E введите ссылки на соответствующие значения из столбца U листа TD_BAG. Таким образом, вы получите случайную выборку признаков и покупателей для обучения первого классификатора (дерева решений).

1.6. Осуществите обучение

1.6.1. Озаглавьте столбец G “PREDICTOR”, столбец H “PREGNANT”. Ниже в строках напишите четыре возможные комбинации значений независимой и зависимой переменных: 0-1, 0-0, 1-1, 1-0. Диапазон I1:L1 настройте, чтобы он автоматически заполнялся значениями из A1:D1 (рис.6.3).

G	H
PREDICTOR	PREGNANT
0	1
0	0
1	1
1	0

Рисунок 6.3 – Варианты значений предиктора и класса

1.6.2. Заполните таблицу (диапазон I2:L5) количеством обучающих строк, значения которых совпадают с комбинацией прогноза (значением предиктора) и значением класса. Например, для ячейки I2 формула имеет вид =СЧЁТЕСЛИМН(A$2:A$667;$G2;$E$2:$E$667;$H2).

1.6.3. Определите, какое значение каждого из признаков является индикатором положительного класса. В ячейке G6 напишите «Какое значение определяет положительный класс?» и в ячейках I6:L6 определите классифицирующее значение каждого признака. Для этого с помощью функции ЕСЛИ проверьте, какое из соотношений больше:

1) отношение числа обучающих примеров при значении признака равном 0 и значением класса равном 1 к общему числу случаев примеров со значением признака равным 0;

2) отношение числа обучающих примеров при значении признака равном 1 и значением класса равном 1 к общему числу случаев со значением признака равным 1.

Большее из этих значений соответствует значению определяющего признака (0 или 1).

1.6.4. Рассчитайте загрязненность классов для каждого дерева решений (по классам и общую). Например, для ячейки I8 формула имеет вид: =1-(I2/(I2+I3))^2-(I3/(I2+I3))^2; для ячейки I9 формула имеет вид: =1-(I4/(I4+I5))^2-(I5/(I4+I5))^2; для ячейки I10 формула имеет вид: **=(I8*(I2+I3)+I9*(I4+I5))/666.**

1.6.5. Определите номер признака-«победителя» из данной выборки. Это признак, имеющий минимальное комбинированное значение загрязненности. В ячейку O1 выведите значение номера признака с минимальной загрязненностью (используйте функции ИНДЕКС, ПОИСКПОЗ, МИН).

1.6.6. В ячейке O2 выведите, какое значение данного признака (0 или 1) ассоциируется с положительным классом (используйте функции ИНДЕКС, ПОИСКПОЗ, МИН).

Таким образом вы получите информацию о первом выигравшем обучающем дереве (пример на рис.6.4.).

	G	H	I	J	K	L	N	O
1	PREDICTOR	PREGNANT	11	3	9	6	Winner:	6
2	0	1	299	187	319	314	Pregnant is:	0
3	0	0	325	188	334	257		
4	1	1	33	145	13	18		
5	1	0	9	146	0	77		
6	Which value indicates pregnancy?		1	0	1	0		
7								
8	Impurity	0	0,499	0,500	0,500	0,495		
9		1	0,337	0,500	0,000	0,307		
10		Combined	0,489	0,500	0,490	0,468		
11								

Рисунок 6.4 – Победитель из четырех деревьев

1.6.7. Скопируйте и вставьте значения из O1:O2 в P1:P2.

1.6.8. Теперь создайте второе дерево решений. Для этого вернитесь на лист TD_BAG и снова перемешайте строки и столбцы. Вернувшись на лист RandomSelection, вы увидите, что изменилась выборка набора признаков и клиентов, а, соответственно, у вас появился новый признак- «победитель» и новое дерево решений. Теперь вставьте справа от столбца. О новый столбец и скопируйте в него полученные данные о втором дереве решений (первое дерево будет смещено вправо).

1.6.9. Запишите Макрос для формирования еще 198 таких деревьев. Всего у вас должно быть 200 деревьев. Вы получили комплексную бэггинговую модель.

2. Оценка бэггинговой модели

2.1. Прогнозирование на тестовой выборке.

2.1.1. Создайте копию листа Test Set и назовите TestBag. На листе TestBag вставьте две пустые строки вверху для деревьев решений. В них, начиная со столбца W, вставьте значения деревьев из листа RandomSelection (P1:HG2). Заголовки строк 1 и 2 поместите в ячейках V1:V2.

2.1.2. Запустите каждую строку тестовых данных в каждое дерево. Вам нужно для каждой строки тестовых данных для каждого дерева сравнить значение соответствующего признака, ассоциирующееся в дереве решений с положительным классом, с тестовым его значением. Если эти значения равны, то прогнозируется принадлежность покупателя к положительному класса. Используйте функцию СМЕЩ для получения возможности растягивания формул. Например, в ячейке W4 формула будет иметь вид =ЕСЛИ(СМЕЩ($A4;0;W$1)=W$2;1;0). Получится лист, как на рис.6.5.

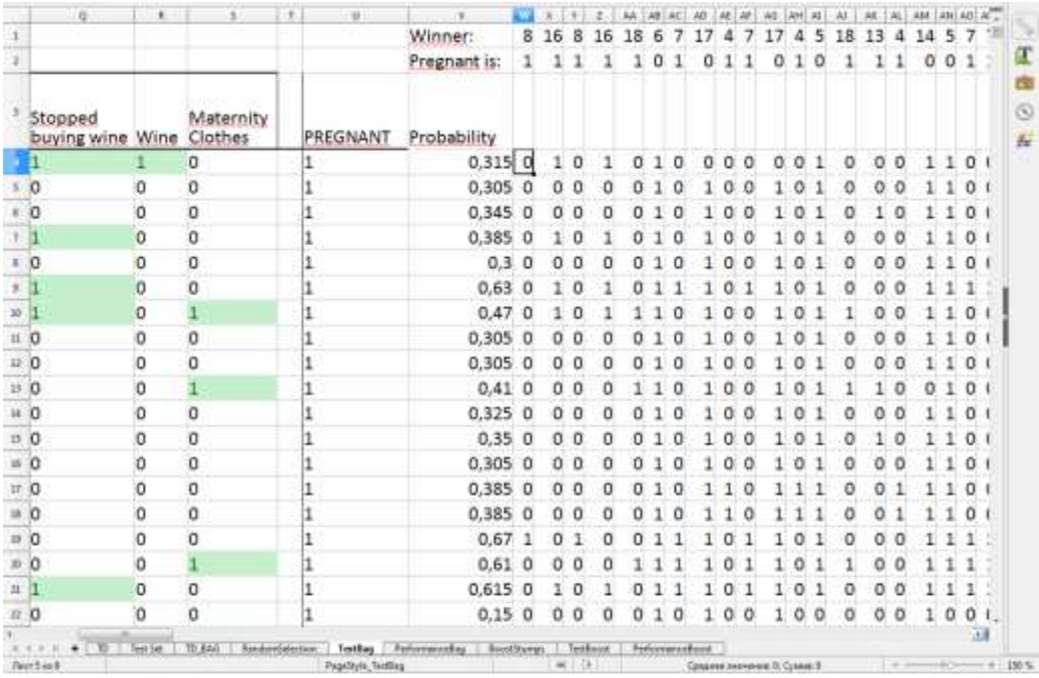


Рисунок 6.5 – Результаты прогнозирования по деревьям решений

2.1.3. В столбце V вычислите классовую вероятность, для этого найдите средне значение всех прогнозов. Вы получили прогнозную вероятность класса для каждого покупателя

2.2. Оцените качество бэггинговой модели (как в лабораторной работе в разделе 2). Для этого создайте лист PerformanceBag. Вычислите точность, специфичность, долю ложноположительных результатов и чувствительность модели. Постройте кривую ошибок (рис.6.6). Сделайте выводы.

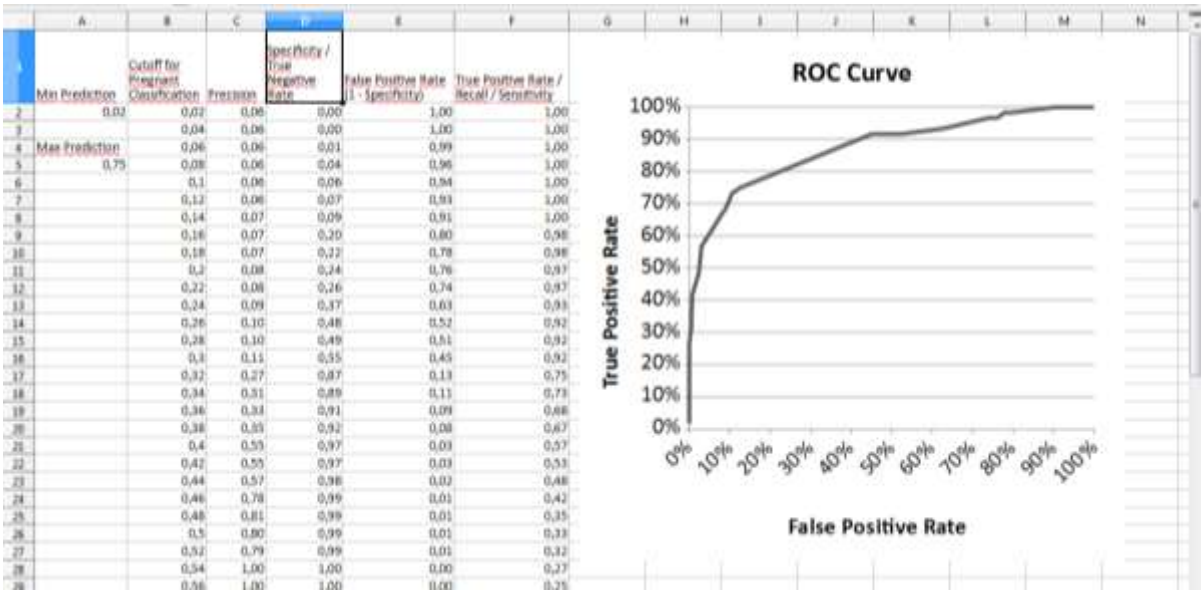


Рисунок 6.6 – Результаты оценки качества бэггинговой модели

3. Создайте и обучите бустинговую модель.

3.1. Создайте лист «BoostStumps». Вставьте в ячейки A1:И5 возможные комбинации признак/результат из промежутка G1:H5 листа «RandomSelection». Начиная с C1 до U1 вставьте номера признаков от 0 до 18.

3.2. Под каждым признаком подсчитайте количество строк обучающего набора, попадающих в каждую из четырех комбинаций признака и зависимой переменной (результат). Например, в C2 формула будет иметь вид =СЧЁТЕСЛИМН(TD!A$3:A$1002;$A2;TD!$U$3:$U$1002;$B2)

3.3. Так же, как и в бэггинге, в C6:U6 найдите значение признака, ассоциированное с положительным классом.

3.4. В столбец B запишите веса для каждого значения данных. Для этого B9 назовите «Current Weights», а под ней с B10 до B1009 введите значение 0,001 для каждой из обучающих строк (первоначально все веса одинаковые и равны 1/количество примеров).

3.5. В строку 9 вставьте названия признаков из листа TD/.

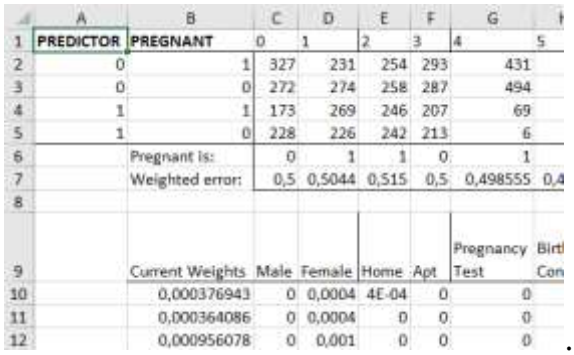
3.6. В диапазоне C6:U1009 оцените взвешенную погрешность для каждого из возможных деревьев решений. Для этого необходимо определить положения обучающих строк, отнесенных к неправильному классу по данному признаку и начислить штраф в размере их весов. Так, например для ячейки C10 штраф начисляется в двух случаях:

1) при совпадении индикатора положительного класса в C6 (лист BoostStumps) и значения соответствующего признака в A3 (лист TD) штраф начисляется, если покупатель не принадлежит к положительному класса (см. значение в столбце U Pregnant на листе TD);

2) при несовпадении индикатора положительного класса в C6 (лист BoostStumps) и значения соответствующего признака в A3 (лист TD) штраф начисляется, если покупатель относится к положительному классу (см. значение в столбце U Pregnant на листе TD).

Формула имеет вид =ЕСЛИ(И(TD!A3=C$6;TD!$U3=0);$B10;0)+ЕСЛИ(И(TD!A3<>C$6;TD!$U3=1);$B10;0)

3.7. В строке 7 рассчитайте взвешенную погрешность для каждого возможного дерева решений как сумму погрешностей по каждому признаку (название ячейки B7 «Weighted error») (рис.6.7).



	A	B	C	D	E	F	G
1	PREDICTOR	PREGNANT	0	1	2	3	4 5
2	0	1	327	231	254	293	431
3	0	0	272	274	258	287	494
4	1	1	173	269	246	207	69
5	1	0	228	226	242	213	6
6		Pregnant is:	0	1	1	0	1
7		Weighted error:	0,5	0,5044	0,515	0,5	0,498555 0,4
8							
9		Current Weights	Male	Female	Home	Apt	Pregnancy Test Birt Con
10		0,000376943	0	0,0004	4E-04	0	0
11		0,000364086	0	0,0004	0	0	0
12		0,000956078	0	0,001	0	0	0

Рисунок 6.7 – Расчет взвешенной погрешности для каждого дерева решений

3.8. Определите дерево-победитель.

В X1 найдите минимальное значение взвешенной погрешности (=МИН(C7:U7)).

В X2 найдите номер выигрышного дерева (с помощью функций ИНДЕКС и ПОИСКПОЗ).

В X3 найдите значение, ассоциирующееся с положительным классом для этого дерева (с помощью функций ИНДЕКС и ПОИСКПОЗ). (пример на рис.**6.**8).

Укажите в отчете название победившего признака.

S	T	U	V	W	X
16	17	18		Winning Error	0,49731981
394	480	389		Column	5
476	397	480		Pregnant is	0
106	20	111		Alpha	0,005
24	103	20			
1	0	1			
0,49825	0,5	0,4985357			

Рисунок 6.8 – Результат определения дерева-победителя

3.9. Рассчитайте величину α для выигравшего дерева по формуле:

$$\alpha = 0{,}5 \ln \frac{1 - \text{Winning Error}}{\text{Winning Error}}$$

Значение α используется в бустинговых моделях для прогнозирования: в нашем случае каждый классификатор (дерева) выдает значение α, если покупатель относится к положительному классу и -α, если нет. Итоговый прогноз ансамбля моделей – это сумма всех отрицательных и положительных значений. В качестве классификационного значения обычно используется 0, но это значение можно менять по своему усмотрению.

3.10. Проведите повторное взвешивание обучающих данных.

3.10.1. В столбце V V9 «Wrong». Теперь по каждой строке для выигравшего признака в столбце Wrong выведите значение 1, если погрешность признака в этой строке больше 0, и значение 0 – если значение погрешности 0. Используйте формулу СМЕЩ, чтобы автоматизировать процесс расчета для других деревьев решений.

3.10.2. В столбце W обозначьте ячейку W9 «Scale by Alpha» и в последующих строках пересчитайте значение весов по формуле:

$$w_2 = w_1 * \exp(\alpha * Wrong)$$

где w_2 - новое ненормированное значение веса для каждого значения данных (строки);

w_1 – исходное значение весов;

Wrong – взвешенная погрешность строки по выигравшему признаку.

3.10.3. Нормируйте новые веса (в результате их сумма должна быть равна 1). Для этого в столбце X ячейку X9 назовите «Normalize» и ниже рассчитайте нормированные значения по формуле:

$$w_N = \frac{w_i}{\sum_i w_i}$$

3.11. Постройте второе дерево.

3.11.1. Скопируйте данные первого дерева из X1:X4 в Y1:Y4.

3.11.2. Скопируйте новые нормированные значения весов из столбца X в столбец B. Лист обновится, и вы получите новое дерево-победитель.

3.12. Запишите макрос для построения деревьев. Не забывайте вставлять новый столбец Y и переносить в него полученные данные по каждому новому дереву. Всего вы должны обучить 200 деревьев (рис.**6.**9).

Приведите фрагмент результатов в отчете, сделайте вывод по изменению взвешенной погрешности и значению α, а следовательно «силе» деревьев при голосовании.

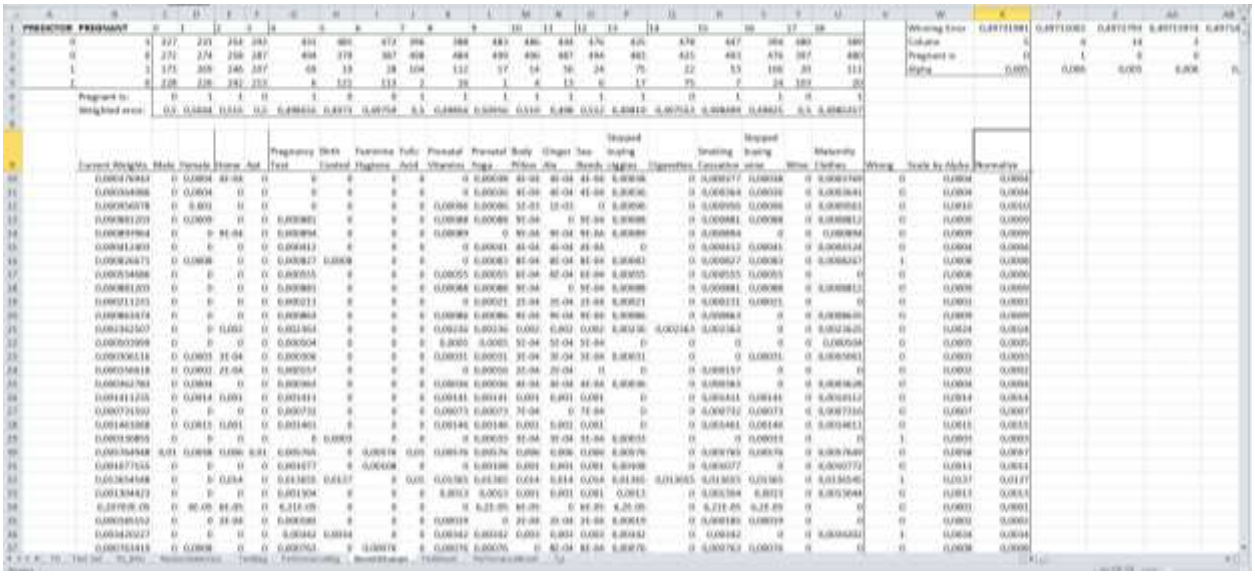


Рисунок 6.9 – Результат обучения бустинговой модели

4. Проведите оценку качества полученной бустинговой модели.

4.1. Осуществите прогнозирование по тестовой выборке.

Сделайте копию листа с тестовой выборкой и назовите его TestBoost. Вставьте 4 пустые строки сверху. И, начиная со столбца W, вставьте ваши деревья с листа BoostStumps

4.2. Рассчитайте прогноз для всех деревьев по каждой строке тестовых данных. Как уже говорилось выше, прогноз имеет значение α, если тестовое значение данного признака равно ассоциированному в дереве с положительным классом значению данного признака. Прогноз имеет значение -α обратном случае. Используйте функции ЕСЛИ И СМЕЩ. Так, для ячейки W6, формула имеет вид =ЕСЛИ(СМЕЩ($A6;0;W$2)=W$3;W$4;-W$4).

4.3. Рассчитайте в столбце V прогноз для каждой строки тестовых данных (покупателей) как сумму прогнозов этой строки по всем деревьев (рис.6.10).

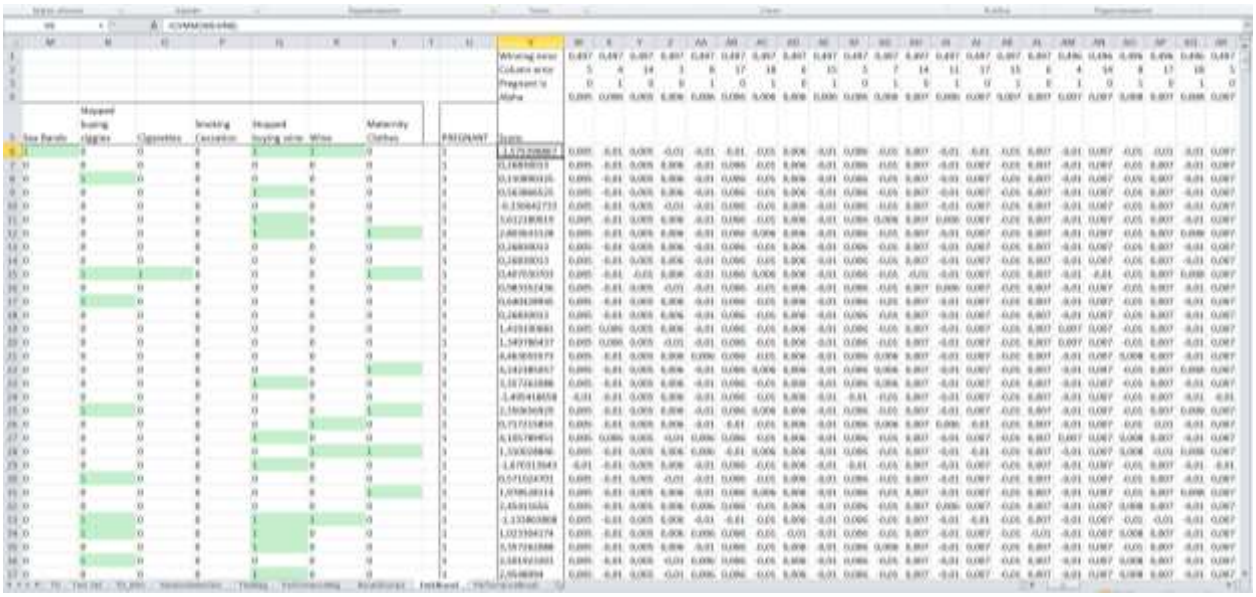


Рисунок 6.10 – Результаты прогнозирования по бустинговой модели

4.4. Создайте лист PerformanceBoost. Вычислите точность, избирательность, долю ложноположительных результатов и чувствительность модели. Постройте кривую ошибок (рис.6.11). Сделайте выводы.

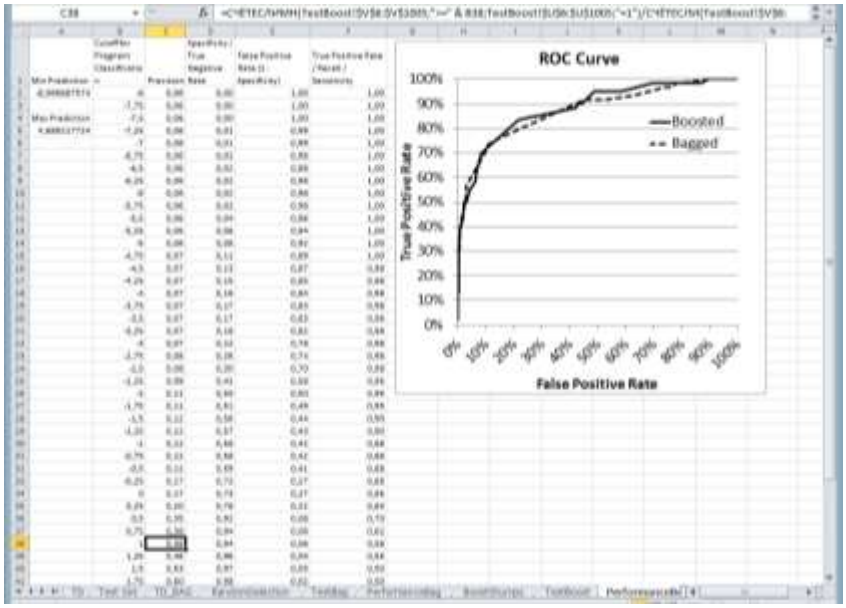


Рисунок 6.11 – Оценка качества бустинговой модели

5. Сравните результаты прогнозирования по бэггинговой и бустингговой моделям.
6. Сравните результаты прогнозирования с регрессионной моделью.

6.3. Контрольные вопросы

1. Понятие ансамбля моделей.
2. Понятие бэггинга. Назначение, основные этапы
3. Понятие бустинга. Назначение, основные этапы
4. Как рассчитывается прогнозное классификационное значение ансамбля моделей при бэггинге?
5. Как рассчитывается прогнозное классификационное значение ансамбля моделей при бустинге? Для чего используется параметр α?
6. Алгоритм «Случайный лес».
7. Загрязнение Джини.

СПИСОК ЛИТЕРАТУРЫ

1. Индекс экономики знаний (Knowledge Economy Index) : сайт Минкомсвязи России. – URL: https://digital.gov.ru/ru/activity/statistic/rating/indeks-ekonomiki-znanij/#tabs|Compare:Place (дата обращения: 10.11.2023).
2. Мицель, А.А. Прикладная математическая статистика: Практические работы / А. А. Мицель. – Томск : ТУСУР, 2019. – 81 с.
3. Мицель, А.А. Прикладная математическая статистика: Учебное пособие / А.А. Мицель. – Томск : ТУСУР, 2019. – 113 с. – URL: https://edu.tusur.ru/publications/9151 (дата обращения: 10.11.2023).
4. Форман, Д. Много цифр: Анализ больших данных при помощи Excel / Д. Форман; перевод А. Соколовой. – Москва : Альпина Паблишер, 2016. – 461 с.
5. Поручиков, М.А. Анализ данных: учеб. пособие / М.А. Поручиков. – Самара : Изд-во Самарского университета, 2016. – 88 с.
6. Агалаков, С.А. Статистические методы в экономике: лабораторный практикум / С.А. Агалаков. – Омск : Изд-во Ом. гос. ун-та, 2010. – 116 с.
7. Салмин, А.А. Анализ данных. Конспект лекций / А.А. Салмин. – Самара: ФГОБУ ВПО «ПГУТИ», 2013. – 111 с.
8. Наивный байесовский классификатор : сайт Bazhenov.me. – URL: http://bazhenov.me/blog/2012/06/11/naive-bayes.html (дата обращения: 10.11.2023).
9. Кинчарова, А. Применение алгоритмов выявления сообществ для социологического исследования блогов: результаты пилотного исследования : сайт ВШЭ. – URL: https://www.hse.ru/data/2012/12/20/1303709367/dzh.pdf (дата обращения: 10.11.2023).
10. Бэггинг : сайт Машинное обучение. – URL: http://www.machinelearning.ru/wiki/index.php?title=%D0%91%D1%8D%D0%B3%D0%B3%D0%B8%D0%BD%D0%B3 (дата обращения: 10.11.2023).
11. Лимановская, О.В. Основы машинного обучения : учебное пособие / О.В. Лимановская, Т.И. Алферьева ; Мин-во науки и высш. образования РФ. – Екатеринбург : Изд-во Урал. ун-та, 2020. – 88 с

Приложение А
(справочное)
Данные для расчета KEI

Таблица А.1 – Значения показателей для расчета KEI для 20 стран

	Ед.изм.	Значения показателей по странам (номера стран)									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Уровень тарифных и нетарифных импортных барьеров	Балл	68,2	86,4	71,6	69,5	84,4	69,8	88,1	77,1	82,6	64,2
Интегральный показатель качества системы регулирования рынков	Пункт	-0,46	1,36	-0,2	-0,9	1,74	0,18	1,64	-0,28	1,07	-0,28
Интегральный показатель соблюдения правовых норм в стране	Пункт	-0,77	1,53	-0,35	-0,66	1,73	-0,18	1,78	-0,81	1,31	0,05
Сумма выплат и доходов по роялти и лицензионным платежам на душу населения	Доллар	32,43	374,65	8,63	35,66	174,02	15,21	324,15	2,39	302,08	1,78
Среднегодовое количество патентов, выданных на 1 млн человек населения	Единица	98,2	695,99	43,11	85,19	846,31	62,54	844,07	11,28	414,09	16,18
Число публикаций в научных журналах в области естественно-научных и технических дисциплин на 1 млн чел. населения	Единица	1,28	308,84	1,05	1,12	68,88	0,68	119,63	0,12	284,91	0,51
Средняя продолжительность обучения населения в возрасте 15 лет и старше	Год	9,69	12,2	8,17	9,35	12,12	7,54	11,37	10,2	11,58	5,12
Удельный вес обучающихся по программам основного и среднего общего образования, среднего профессионального образования в общей численности населения в возрасте 11-17 лет	Процент	84,81	93,57	78,19	85,86	132,69	100,79	102,21	99,36	101,02	60,02
Удельный вес учащихся по программам среднего профессионального (программам подготовки среднего звена) и высшего образования в общей численности населения в возрасте 18-22 лет	процент	77,19	85,93	24,53	69,38	82,33	34,44	62,36	19,06	58,62	13,48
Совокупное число подключенных терминалов подвижной радиотелефонной связи и телефонных аппаратов на 1000 человек	Единица	1940	1470	800	1530	1520	1110	1220	1040	1250	480
Число персональных компьютеров на 100 человек населения	Единица	130	810	60	260	680	350	940	80	690	30
Число пользователей Интернета на 1000 человек	Единица	420	780	290	300	720	390	780	420	780	50

Окончание табл. А.1

Показатели	Ед.изм.	Значения показателей по странам (номера стран)									
		11	12	13	14	15	16	17	18	19	20
Уровень тарифных и нетарифных импортных барьеров	Балл	85,3	63,8	71,1	91,2	69,2	75,3	81,2	83,3	67,5	64,2
Интегральный показатель качества системы регулирования рынков	Пункт	0,78	-0,15	-0,4	1,3	0,7	0,13	-0,2	1,34	-0,45	0,12
Интегральный показатель соблюдения правовых норм в стране	Пункт	1,1	0,04	-0,1	1,22	0,6	0,95	0,07	0,5	-0,2	0,34
Сумма выплат и доходов по роялти и лицензионным платежам на душу населения	Доллар	289,3	1,56	45,33	298,05	16,7	78,3	15,6	112,8	54,6	116,2
Среднегодовое количество патентов, выданных на 1 млн человек населения	Единица	345,2	14,2	98,12	412,78	15,5	289,3	45,6	105,6	87,3	65,3
Число публикаций в научных журналах в области естественно-научных и технических дисциплин на 1 млн чел. населения	Единица	75,3	1,1	7,56	175,25	112,5	5,6	87,6	48,6	13,5	12,8
Средняя продолжительность обучения населения в возрасте 15 лет и старше	Год	10,1	6,7	8,87	10,2	9,5	10,1	10,3	9,8	10,9	8,9
Удельный вес обучающихся по программам основного и среднего общего образования, среднего профессионального образования в общей численности населения в возрасте 11-17 лет	Процент	86,7	65	83,84	98,7	88,9	88,6	94,5	86,5	96,3	85,9
Удельный вес учащихся по программам среднего профессионального (программам подготовки среднего звена) и высшего образования в общей численности населения в возрасте 18-22 лет	процент	75,3	15,54	71,22	67,3	67,5	77,8	75,6	65,8	34,8	37,8
Совокупное число подключенных терминалов подвижной радиотелефонной связи и телефонных аппаратов на 1000 человек	Единица	953	497	1212	1175	876	875	678	1010	650	880
Число персональных компьютеров на 100 человек населения	Единица	750	121	310	980	452	756	897	987	756	645
Число пользователей Интернета на 1000 человек	Единица	578	51	298	760	432	650	678	545	384	412