ПРОГРАММНАЯ РЕАЛИЗАЦИЯ СИСТЕМЫ ПРИНЯТИЯ РЕШЕНИЙ ДЛЯ ДОГОВОРНОЙ РАБОТЫ ПРЕДПРИЯТИЯ

Построены два дерева решений для договорной и претензионной работы предприятия, реализованных на языке программирования высокого уровня Python. Рассмотрено влияние разных наборов показателей на результат построения дерева решений. Представлены рекомендации по определению целевых наборов показателей для эффективного применения интеллектуальной системы поддержки принятия решений.

Ключевые слова: дерево решений, Data Mining, договорная и претензионная работа предприятия, выборка, классификация, энтропия

И.А. Журавлев

Е.А. Журавлева

Внедрение реальных приложений искусственного интеллекта в корпоративной среде становится все более популярным направлением. Умные системы проникают во все сферы общественной и деловой жизни. В основе таких систем лежат передовые инструменты анализа и сбора данных, задачи прогнозирования и задачи принятия решений. Интеллектуальные информационные системы обладают способностью быстро анализировать поступающую в них информацию и действовать в динамично меняющихся условиях, что приводит к повышению эффективности и качества операций. Рассмотрим применение метода дерева решений в договорной и претензионной работе предприятия.

Дерево решений (decision tree) является одним из самых мощных и популярных инструментов Data Mining. Оно позволяет эффективно решать задачи регрессии, задачи классификации и задачи прогнозирования основных социальных, экологических и экономических показателей.

При построении дерева решений для договорной и претензионной работы предприятия необходимо рассмотреть договоры, которые уже были исполнены в компании. На основании анализа этих договоров выделим их основные признаки, которые характеризуют каждый договор и оформим их в виде таблицы. Ниже представлен фрагмент таблицы с данными по договорам (табл. 1).

Закодируем результат успешной реализации договора в виде «0» и «1», где «0» – данный договор для компании принес убытки, а «1» – при выполнении договора компания получила прибыль.

В классической постановке задачи машинного обучения выборка состоит из объектов и признаков

Журавлев Илья Александрович, кандидат технических наук, доцент кафедры «Системы управления транспортной инфраструктурой» Российской открытой академии транспорта Российского университета транспорта (РОАТ РУТ (МИИТ)). Область научных интересов: надежность и эффективность функционирования систем железнодорожной автоматики и телемеханики, моделирование систем и процессов, эволюционные вычисления, интеллектуальный анализ данных. Автор 61 научных работ, в том числе трех учебных пособий.

Журавлева Елена Анатольевна, магистр, специалист тендерного отдела ООО «АСБК софт». Область научных интересов: искусственный интеллект, эволюционные вычисления, интеллектуальный анализ данных. Автор четырех научных работ.

Таблица 1

Фрагмент таблицы с данными по договорам, используемыми для реализации метода интеллектуального анализа данных

Наименование контрагента	Предмет договора	Цена договора, тыс. руб.	Срок выполнения работ/ услуг/ поставки, мес.	Наличие по договору штрафных санкций, %	Способ оплаты выполненных работ	Процентное соотношение прибыли от цены контракта, %	Результат успешной реализации договора
Контрагент 1	Поставка товара	138	12	4,3	20	37	1
Контрагент 2	Поставка товара	783	6	7,5	40	21,11	0
Контрагент 3	Поставка товара	426	3	2,1	40	50,94	1
Контрагент 4	Поставка товара	2371	1	8,2	40	51.69	1
Контрагент 5	Поставка товара	4620	3	3,7	30	60.97	1
Контрагент 6	Оказание услуг	118	6	6,3	30	26.75	0
Контрагент 7	Оказание услуг	752	1	9,4	40	31.91	0
Контрагент 8	Оказание услуг	485	3	8,1	40	74,64	1
Контрагент 9	Оказание услуг	2637	3	2,9	20	37.92	0
Контрагент 10	Техническое обслуживание	915	3	2,4	40	7.94	0
Контрагент 11	Техническое обслуживание	4218	12	6,8	20	78.03	1

(рис. 1): X – типичная матрица объектов признаков (design matrix); l – длина выборки (length) или число объектов; d – число признаков (dimensions). В работе применяется способ машинного обучения – обучение с учителем (Supervised learning), т.е. в обучающей выборке есть метки: X – типичная матрица объектов признаков; y – вектор ответов. Следовательно, табл. 1 представлена в виде размеченной выборки, где объектом служит договор, т.е. строчка в матрице, с X-признаками (Nazvanie kontagenta, Predmet dogovora, Cena dogovora tis.rub, Srok mes., Shtraf %, Sposob oplati, Pribil %) и соответствующая метка y (Resultat). Данная задача имеет вид классификации (classification). Классификация – это отнесение объекта к одной из категорий на основании его признаков, т.е. число уникальных значений ограничено, метка y принимает значение «True» или «False», где «False» – реализация договора нанесла ущерб компании; «True» – договор выполнен успешно.

Перед нами стоит задача отображения зависимости X - конкретной матрицы (NumPy) от y в виде функции 1.

$$f: X \to y. \qquad (1)$$

Функция 1 строится методом «Дерево решений». На каждом шаге выбирается лучший признак, который в конкретный момент времени выигрывает по сравнению с остальными. В нашей постановке дерево решения опирается на бинарный признак y - Resultat, а числовые признаки: Nazvanie kontagenta, Predmet dogovora, Cena dogovora tis.rub, Srok mes., Shtraf %, Sposob oplati, Pribil % – используются для сравнения с заданным порогом. Из этого следует, что нам необходимо найти такой признак, который лучше всего разделяет выборку по целевому классу. Эвристикой сформулированной цели является критерий информативности, который покажет, насколько заданный признак «хороший» для того, чтобы разбить выборку на два класса. Формализация представлена с помощью понятия энтропии.

Энтропия Шеннона определяется для системы с N возможными состояниями по формуле:

$$S = -\sum_{i=1}^{N} p_i \log_2 p_i, \qquad (2)$$

где p_i – вероятности нахождения системы в i-м состоянии.

Полный порядок в системе, если энтропия принимает значение ноль. Так как энтропия – это, по сути, степень неопределенности в системе, уменьшение энтропии называют приростом информации. Прирост информации (information gain, IG) формализует то, насколько хорош заданный критерий (Q) для деления выборки на две группы по определенному признаку:

$$IG(Q) = S_0 - \sum_{i=1}^{q} \frac{N_i}{N} S_i, \qquad (3)$$

где q – число групп после разбиения;

N_i – число элементов выборки, у которых признак Q имеет i-е значение.

Выборка делится на две подгруппы. Сравнивая признаки и считая для каждого прирост информации, определяется, какое разбиение будет лучшим. Данный алгоритм работает рекурсивно, т.е. процедура поиска признака с лучшим приростом информации продолжается в каждой подгруппе до тех пор, пока не построится дерево решений, классифицирующее нужный итог. В работе дерево решения прогнозирует: заключать договор или нет.

В нашем случае будем пользоваться оценкой Джини.

Дерево решений обучается на 70% выборки, а на оставшихся 30% проверяются прогнозы. Другими словами, когда обученное дерево сделает прогноз для отложенной выборки, появится вектор прогнозов (prediction). Нам известен вектор реальных ответов, поэтому достаточно сравнить вектор реальных ответов и вектор прогнозов. Описанный метод называется оценкой модели по отложенной выборке. Проблема метода заключается в том, что мы можем наладить работу на отложенной выборке, но иметь плохой результат на тестовой.

Более универсальным методом является метод кросс-валидации (k-fold cross-validation). Этот метод

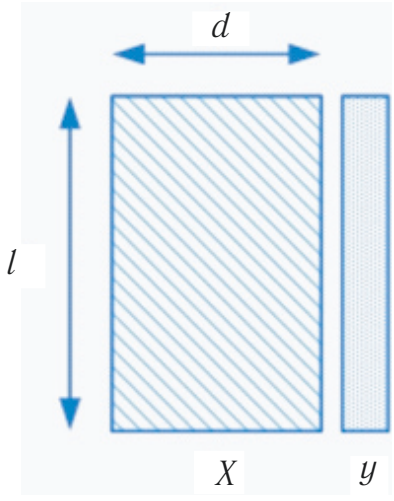

Рис. 1. Представление выборки

случайным образом разбивает данные на k непересекающихся блоков примерно одинакового размера. Поочередно каждый блок рассматривается как валидационная выборка, а остальные $k-1$ блоков – как обучающая выборка. Модель обучается на $k-1$ блоках и прогнозирует валидационный блок. Процесс повторяется k раз и мы получаем k оценок, для которых рассчитывается среднее значение, являющееся итоговой оценкой модели. Обычно k выбирают равным 10, 7 или 5. Если k равен количеству элементов в исходном наборе данных, этот метод называется кросс-валидацией по отдельным элементам (leave-one-out cross-validation), но вычислительно его очень сложно осуществить и поэтому он не используется. Важным аспектом является то, что при проведении такого разбиения нужно учитывать соотношение целевого класса.

Подкласс кросс-валидации – Stratified гарантирует, что распределение целевого класса в каждой подвыборке (fold) будет одним и тем же. Метод StratifiedkFold разбивает выборку на n-частей, учитывая соотношение целевого класса в каждой подгруппе, т.е. все подвыборки похожи между собой и на исходную выборку по соотношению целевого класса. Кросс-валидация дает ответ на то, как сравнить две модели и как гарантировать, что наша модель будет работать на новых данных.

В машинном обучении комбинируют метод оценки по отложенной выборке и метод кросс-валидации. То есть мы отщепляем 30% исходной выборки, отложенная часть, а на оставшихся 70% выборки проводим кросс-валидацию.

Главной проблемой в машинном обучении является переобучение, когда модель, обученная на своей выборке, будет лучше работать, чем на другой. Переобучение появляется при слишком долгом настраивании на имеющую выборку. В дереве решений это происходит при построении до максимальной глубины. Ограничить построение дерева поможет гиперпараметр – max_depth. Этот параметр необходимо настроить так, чтобы одновременно дерево не переобучалось и не было слишком мелким. Мелкое дерево – это недообученное дерево, т.е. не выявлены необходимые закономерности для данных, чтобы хорошо прогнозировать. Для достижения баланса, подбирая параметр max_depth, на 70% выборки надо сравнивать деревья разных глубин по параметру кросс-валидации.

При добавлении новых параметров модели проверка происходит также с помощью кросс-валидации. Отложенная выборка требуется только в конце, она дает самую последнюю оценку, т.е. показывает, как модель работает на новых данных.

Построение дерева решений для договорной и претензионной работы предприятия реализовано на языке программирования высокого уровня Python. На рис. 2 изображено, в каком виде первоначально выводится часть данных для анализа.

Очевидно, что успешность выполнения договора зависит от прибыли, которая получена при его реализации; строя дерево решения с данным показателем, есть возможность потерять другие важные критерии. Поэтому для сравнения построим два дерева решения, отличие которых будет в наличии показателя {Pribil %} при вводе данных.

Первое дерево решений строим со следующими признаками {Nazvanie kontragenta, Predmet dogovora, Cena dogovora tis.rub, Srok mes., Shtraf %, Sposob oplati, Resultat}, исключая {Pribil %}. Итог реализации кода построения дерева решений изображен на рис. 3.

Показатели, рассматриваемые для построения дерева решений, представленного на рис. 3, изображены на рис. 4.

Результат реализации программного кода для построения дерева решения со всеми признаками { Nazvanie kontragenta,Predmet dogovora,Cena dogovora tis.rub,-Srok mes.,Shtraf %,Sposob oplati ,Pribil %,Resultat } приведен на рис. 5. Данное дерево решения применимо в организации, которая нацелена только на получение прибыли, не обращая внимания на сопутствующие факторы при выполнении договора.

	Nazvanie kontragenta	Predmet dogovora	Cena dogovora tis.rub	Srok mes.	Shtraf %	Sposob oplati	Pribil %	Resultat
0	1	100	138	12	4.3	20	37.00	True
1	2	100	783	6	7.5	40	21.11	False
2	3	100	426	3	2.1	40	50.94	True
3	4	100	2371	1	8.2	40	51.69	True
4	5	100	4620	3	3.7	30	60.97	True

Рис. 2. Фрагмент таблицы с данными для построения дерева решений

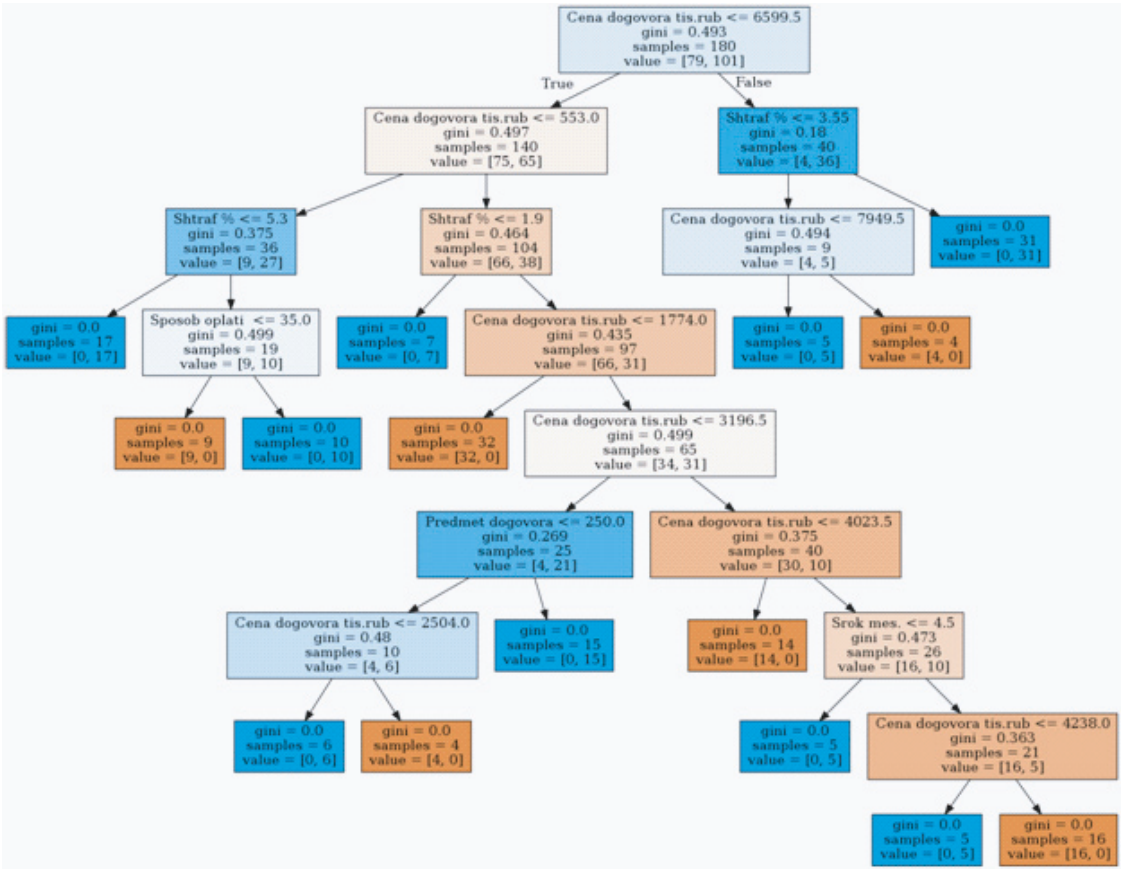

Рис. 3. Дерево решений для договорной и претензионой работы предприятия со следующими признаками {Predmet dogovora, Cena dogovora tis.rub, Srok mes., Shtraf %, Sposob oplati, Resultat}

	Predmet dogovora	Cena dogovora tis.rub	Srok mes.	Shtraf %	Sposob oplati	Resultat
0	100	138	12	4.3	20	True
1	100	783	6	7.5	40	False
2	100	426	3	2.1	40	True
3	100	2371	1	8.2	40	True
4	100	4620	3	3.7	30	True

Рис. 4. Фрагмент таблицы с данными для построения дерева решений со следующими признаками {Nazvanie kontragenta, Predmet dogovora, Cena dogovora tis.rub, Srok mes.,Shtraf %, Sposob oplati, Resultat }

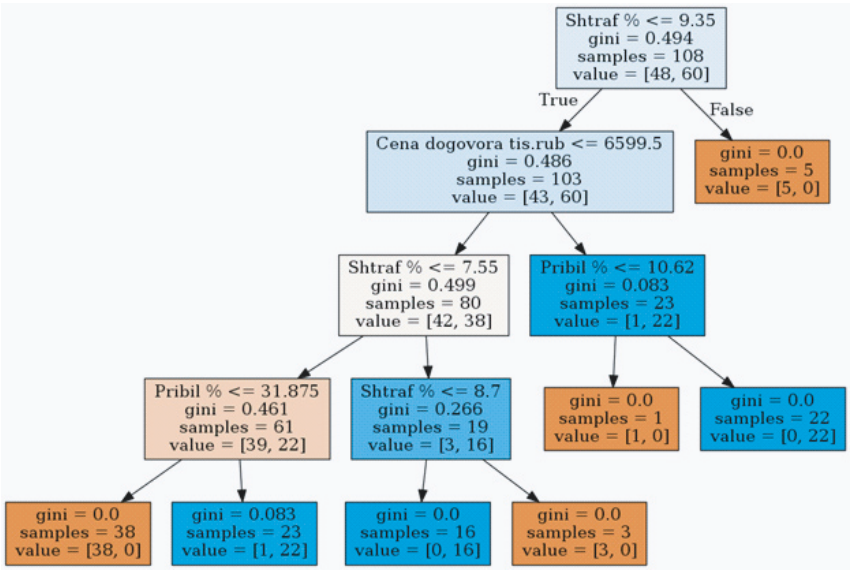

Рис. 5. Дерево решений для договорной и претензионой работы предприятия со следующими признаками {Predmet dogovora, Cena dogovora tis.rub, Srok mes., Shtraf %, Sposob oplati, Pribil %, Resultat}

Литература

1. Мусаев, А.А. Алгоритмы Data Mining в задачах управления динамическими процессами / А.А. Мусаев, И.А. Барласов. —Текст: непосредственный // Труды СПИИРАН. —Вып. 5. —Санкт-Петербург: Наука, 2007. —С. 300—313.

2. Де Янг, К. Эволюционные вычисления: новейшие достижения и нерешенные проблемы / К.Де Янг. —Текст: непосредственный // Обозрение прикладной и промышленной математики. —1996. —Т. 3, вып. 5.

3. Скобцов, Ю.А. Эволюционные вычисления: учебное пособие / Ю.А. Скобцов, Д.В. Сперанский; Нац. открытый ун-т (ИНТУИТ). —Москва: Нац. открытый ун-т (ИНТУИТ), 2015. —326 с. —ISBN 978-5-9556-0171-7. —Текст: непосредственный.

4. Маслова, В.М. Система рекрутинга с элементами искусственного интеллекта / В.М. Маслова. —Текст: непосредственный // Экономические системы, 2018. —Т.11. —№1 (40). —С. 56—59.