

Курсовой проект

Мегафон.

Вероятность подключения услуги

Табунов Миша

07.06.2020

Задача

У нас появился запрос из отдела продаж и маркетинга. Как вы знаете «МегаФон» предлагает обширный набор различных услуг своим абонентам. При этом разным пользователям интересны разные услуги. Поэтому необходимо построить алгоритм, который для каждой пары пользователь-услуга определит вероятность подключения услуги.

Данные

В качестве исходных данных вам будет доступна информация об отклике абонентов на предложение подключения одной из услуг. Каждому пользователю может быть сделано несколько предложений в разное время, каждое из которых он может или принять, или отклонить.

Отдельным набором данных будет являться нормализованный анонимизированный набор признаков, характеризующий профиль потребления абонента. Эти данные привязаны к определенному времени, поскольку профиль абонента может меняться с течением времени.

Данные train и test разбиты по периодам – на train доступно 4 месяцев, а на test отложен последующий месяц.

Предобработка

Обработка основном даты и была попытка
“подшумить данные” : `data['diff1_l']`

`(np.log(data.iloc[1:,3].values))`

Признаки отбирал на основе их IV
(информационной важности)

Признаки с 10 уникальными значения
перевел в категориальные. Остальные ушли в
числовые. Признаки потом
трансформировал.

DATA PREPROCESSING

Getting Started with Machine Learning





Step 1: Importing the required Libraries

These Two are essential libraries which we will import every time.
NumPy is a Library which contains Mathematical functions.
Pandas is the library used to import and manage the data sets.



Step 2: Importing the Data Set

Data sets are generally available in .csv format. A CSV file stores tabular data in plain text. Each line of the file is a data record. We use the read_csv method of the pandas library to read a local CSV file as a dataframe. Then we make separate Matrix and Vector of independent and dependent variables from the dataframe.

Step 3: Handling the Missing Data

The data we get is rarely homogeneous. Data can be missing due to various reasons and needs to be handled so that it does not reduce the performance of our machine learning model. We can replace the missing data by the Mean or Median of the entire column. We use Imputer class of sklearn.preprocessing for this task.

NaN



Step 4: Encoding Categorical Data

Categorical data are variables that contain label values rather than numeric values. The number of possible values is often limited to a fixed set. Example values such as 'Yes' and 'No' cannot be used in mathematical equations of the model so we need to encode these variables into numbers. To achieve this we import LabelEncoder class from sklearn.preprocessing library.



Step 5: Splitting the dataset into test set and training set

We make two partitions of dataset one for training the model called training set and other for testing the performance of the trained model called test set. The split is generally 80/20. We import train_test_split() method of sklearn.crossvalidation library.



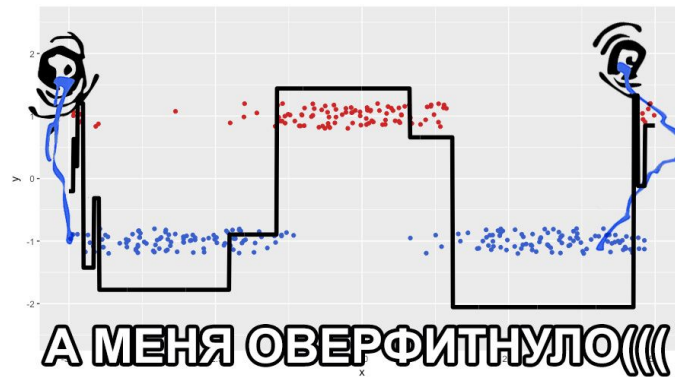
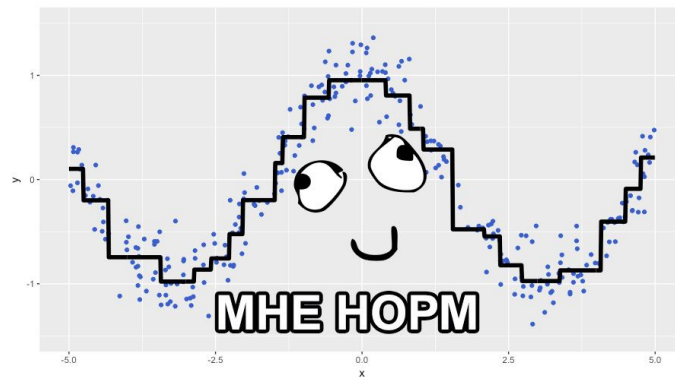
Step 6: Feature Scaling

Most of the machine learning algorithms use the Euclidean distance between two data points in their computations. Features highly varying in magnitudes, units and range pose problems. High magnitudes features will weigh more in the distance calculations than features with low magnitudes. Done by Feature standardization or Z-score normalization. StandardScaler of sklearn.preprocessing is imported.

Модели

Обкатывал три модели:

1. LogisticRegression (большие ошибки)
2. XGBoostClassifier ("0" находит отлично, а вот "1" пропускает)
3. BernoulliNB ("1" находит отлично, а вот "0" пропускает)



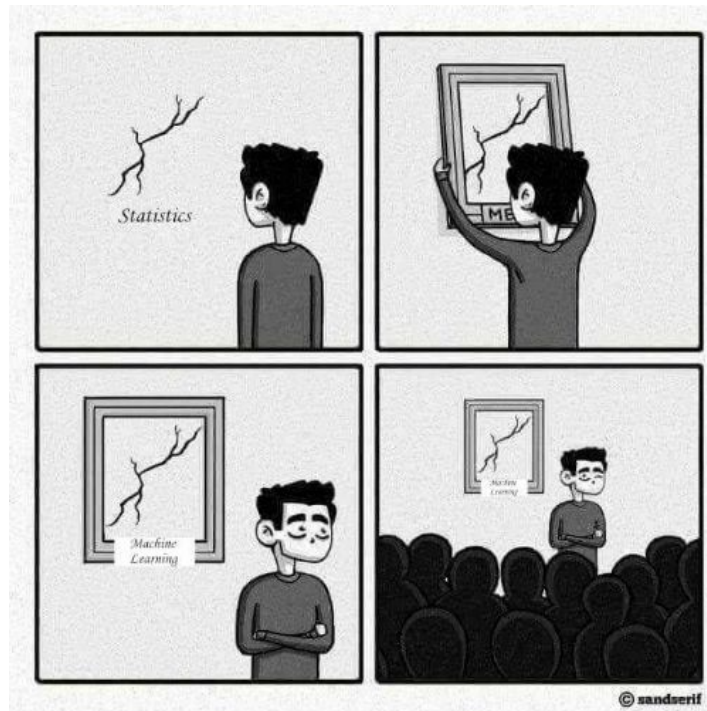
Оценка резу^льтатов

Работу модели оценивал по матрице смещений.

Модель подгонял к наибольшей правильности TP/TN

CONFUSION MATRIX

col_0	0.0	1.0
target		
0.0	1026	15
1.0	73	5



Machine Learning

Data Analyst



What my friends think I do



What my Mom thinks I do



What my boss thinks I do



What my customers think I do



What I think I do



What I really do