Politechnika Wrocławska
Wydział Informatyki i Telekomunikacji

Regresja oparta na danych policyjnych dotyczących przestępczości w San Francisco

Sprawozdanie z laboratorium MSiD

Autor

Jakub Krupiński

nr albumu: **255356**

kierunek: **Informatyka Stosowana**

14 czerwca 2022

Streszczenie

Praca przedstawia program przewidujacy czas, który upływa między popełnieniem przestępstwa a złożeniem raportu policyjnego w jego sprawie. Opiera się on na danych pobranych ze strony: https://data.sfgov.org. Praca wykonywana jest na losowo wybranej próbce zawierającej 5000 wierszy danych. Są one oczyszczane przeze mnie ze zbędnych kolumn, odpowiednio formatowane, pozbawiane wierszy z niepoprawnymi lub nadmiernie odstającymi danymi oraz są wzbogacone o dodatkową kolumnę - "TimeDifference", która jest obiektem przeprowadzanych przeze mnie przewidywań. W programie wykorzystywany jest sporządzony przeze mnie niestandardowy model wykorzystujący metodę "curve_fit" z modułu "scipy", a także modele regresji liniowej oraz SVR z modułu "sklearn". Skuteczność poszczególnych metod jest oceniana za pomocą parametrów: błędu średniokwadratowego, średniego bezwzględnego błędu procentowego oraz współczynnika determinacji R^2.

1 Wstęp – sformułowanie problemu

Autor podejmuje próby oceny czasu, który upływa pomiędzy popełnieniem przestępstwa a sporządzeniem raportu policyjnego w jego sprawie, co pozwoli ocenić jak szybkie są reakcje policji w San Francisco na tego typu zdarzenia.

2 Opis danych

Wielkość datasetu to około 601000 wierszy. Pobierana przez mój program próbka na podstawie której budowane są modele jest wielkości 5000 wierszy, jednakże ostateczny jej rozmiar waha się, ze względu na to, że filtrowane są dane nie pasujące do wykonywanego przeze mnie zadania, a wstępna próbka 5000 wierszy dobierana jest losowo.

Kolumna "IncidentDayofWeek" - zmienna typu "string", kategoria posiadająca łącznie 7 różnych wartości odpowiadających dniom tygodnia

Kolumna "ReportTypeCode" - zmienna typu "string", kategoria posiadająca łącznie 4 różne wartości reprezentujące typ raportu

Kolumna "FiledOnline" - zmienna typu "int", posiadająca wartości 0 lub 1, odpowiadające kolejno przypadkom, w którym dany raport składany był odpowiednio przez internet lub ręcznie

Kolumna "IncidentCategory" - zmienna typu "string", kategoria posiadająca łącznie 49 różnych wartości opisujących rodzaj incydentu, który miał miejsce

Kolumna "Resolution" - zmienna typu "string", kategoria posiadająca łącznie 4 różne wartości opisujące w jaki sposób zakończyła się sprawa dotycząca danego incydentu

Kolumna "PoliceDistrict" - zmienna typu "string", kategoria posiadająca łącznie 11 różnych wartości opisujących w jakim okręgu policyjnym miał miejsce incydent

Kolumna "TimeDifference" - zmienna typu "int" opisująca w minutach czas, który upłynął od popełnienia przestępstwa do sporządzenia odpowiedniego raportu policyjnego w jego sprawie.

3 Opis rozwiązania

Dane do datasetu zostały pobrane ze strony: https://data.sfgov.org/Public-Safety/Police-Department-Incident-Reports-2018-to-Present/wg3w-h783. Baza została zapisana w postaci ramki danych biblioteki "Pandas". Zawiera ona dane na temat przestępstw, które zostały popełnione w obrębie San

Francisco w latach 2018 - 2022 w ilości nie większej niż 5000. Po dodaniu do ramki danych kolumny "TimeDifference" program odsiewa dane błędne, oraz odstające z obu stron 5% odpowiednio najmniejszych i największych danych, w celu wyeliminowania najbardziej skrajnych błędów. Po wykonaniu odpowiednich obliczeń, przy pomocy bibliotek "matplotlib" oraz "seaborn" dane prawdziwe oraz te wyliczone przez program są wyświetlane użytkownikowi na zestawie wykresów.

4 Rezultaty obliczeń

4.1 Plan badań

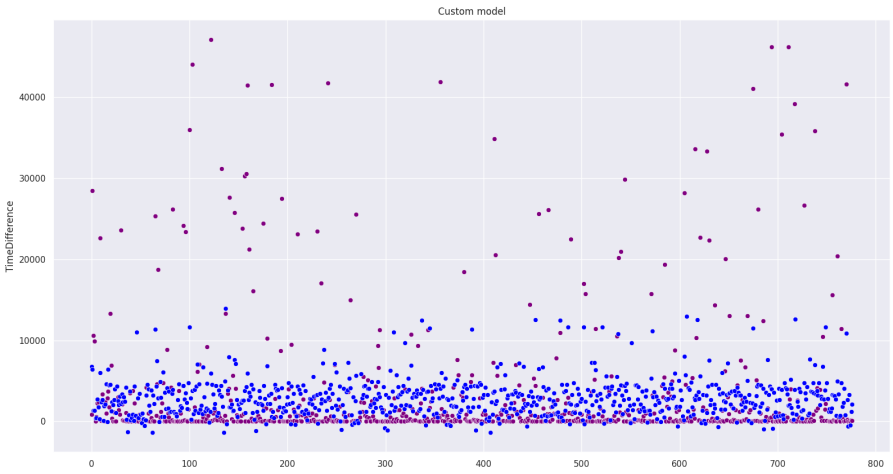
Zbiór danych zostanie podzielony na dwie części: treningową i testową w stosunku 80:20.

4.2 Wyniki obliczeń

Model oceny wina można przedstawić następującym wzorem:

$$\det TimeDifference = \alpha * get\_dummies(IncidentDayofWeek) + \beta * get\_dummies(ReportTypeCode) + \gamma * FiledOnline$$
$$+ \delta * get\_dummies(IncidentCategory) + \epsilon * get\_dummies(Resolution) + \zeta * get\_dummies(PoliceDistrict) \quad (1)$$

gdzie get_dummies() to funkcja mapująca dane kategoryczne na reprezentację one-hot.
Na rys. 1 pokazany jest przykładowy wykres.



Rysunek 1: Przewidywane wartości dla modelu niestandardowego (kolor niebieski) do prawdziwych danych (kolor fioletowy)

Dla danych, które zostały wygenerowane przy danym uruchomieniu programu, dla modelu niestandardowego, otrzymano parametry:
Błąd średniokwadratowy: 56524198.38275801
Średni bezwzględny błąd kwadratowy: 116.49666064992282
Współczynnik determinacji: 0.08562423484263215

Wartość błędu średniokwadratowego jest tu bardzo duża, co bez wątpienia jest spowodowane dużym odchyłem niektórych danych. Średni bezwzględny błąd kwadratowy okazał się być zdecydowanie mniejszy, lecz nie zawsze jest to prawdą - czasem, gdy dane układają się niefortunnie, staje się on niezwykle duży, co powoduje duże komplikacje dla implementowanych modeli, zwłaszcza dla modelu regresji liniowej. Współczynnik determinacji osiągnął wartość należącą do przedziału [0, 1], co jest bardzo dobrym znakiem, jako że jest on miarą stopnia, w jakim model pasuje do próby danych. Co prawda jest on bardzo blisko wartości 0, więc oznacza to, że model nie jest zadowalający, jednak co najważniejsze nie jest to wartość ujemna - taki przypadek informuje o tym, że model nie podąża poprawnie za trendem danych.

5 Wnioski

Przedstawione przeze mnie rozwiązanie pozwala na przygotowanie różnych modeli, które podejmują się przewidywań danych znajdujących się w datasecie. Modele te nie są idealne - błędy średniokwadratowe osiągają bardzo wysokie wartości dla każdej z metod, a średnie bezwzględne błędy procentowe, choć zazwyczaj są mniejsze, potrafią osiągać równie duże wartości przy modelu liniowym, co może sprawić, że jego rozwiązanie nie jest zdatne do analizy w danym przypadku. Współczynniki determinacji oscylują w okolicy zera dla każdego z tych modeli - model SVR zazwyczaj posiada wartości nieco ujemne, podczas gdy model liniowy oraz model niestandardowy (zazwyczaj) są nieznacznie dodatnie. Modele te nie sprawdzają się w przewidywaniu czasu upływającego między popełnionym przestępstwem a złożeniem z jego powodu raportu tak dobrze, jak bym sobie tego życzył. Bez wątpienia jednym z powodów jest bardzo duża liczba kategorii - kodowanie OneHot, choć niezwykle przydatne i skuteczne przy ograniczonych ilościach kolumn, potrafi wpłynąć negatywnie na wydajność rozwiązania, gdy tych kategorii jest za dużo, tak jak w przypadku chociażby kolumny "IncidentCategory". Rozwiązanie, które mogłoby poprawić wyniki sporządzonych przeze mnie modeli mogłoby oznaczać ograniczenie ilości kategorii dla tej kolumny i pogrupowanie wartości spokrewnionych ze sobą. Kolejnym powodem, który prawdopodobnie ma duży wpływ na wyniki programu są same dane, które znajdują się w datasecie - choć większość danych znajduje się w podobnych granicach wartości, w zestawie danych nie brakuje danych obarczonym gigantycznym błędem. Mimo odsiewania najbardziej ekstremalnych 5% przypadków, nie jest to wystarczające, by wartości funkcji można było skutecznie zamodelować. Kolejnym potencjalnym sposobem na poprawę skuteczności mojego rozwiązania byłoby więc bardziej surowe filtrowanie danych, na podstawie których tworzone są modele. Większy odsiew wartości obarczonych nieproporcjonalnie dużymi błędami oznaczałby bardziej spójny zestaw danych, który znacznie łatwiej byłoby opracować.

A Dodatek

Kody źródłowe(utrzymane w konwencji języka Python wraz z instrukcjmi uruchomienia) umieszczone zostały w repozytorium github:
`https://github.com/29379/MSiD_final_project`.