Raport

Przewidywanie Skali Przestępczości

Alisa Krsek s19542, Vasyl Korzh s18846

Własne komentarze, wnioski.

Rekomendacja:

Dla pełnego zrozumienia projektu zaleca się go otworzyć, ponieważ kod został napisany w formie JupyterNoteBook i jest naprawdę dobrze napisany wraz ze wszystkimi komentarzami i odpowiada na wszystkie pytania postawione do napisania w dokumentacji oraz można go traktować jak dokumentację Kod z dokumentacją (plik project-prod1.html - cały uruchomiony kod w formacie html).

Kod obejmuje wszystkie procesy badania zbioru danych, grafiki, budowy modeli pośrednich, a także ostatecznego modelu, który służy jako predyktor wyników. Pełne wykonanie kodu zajmuje około **22 minut**.

Kod został napisany z użyciem platformy Kaggle, dlatego po uruchomieniu na własnym komputerze mogą wystąpić błędy związane z brakiem zainstalowanych modułów na nim (np. XGBRegressor, LGBMRegressor).

Cel badania :

Celem badania jest opracowanie modelu do predykcji skali przestępczości nowego regionu, znając jego dane demograficzne.

Opis zbioru danych

Dane zawierają opisy demograficzne oraz stopień przestępczości w miastach w USA. Razem jest ponad 120 cech (średnia pensja, średnia wielkość rodziny, warunki zamieszkania,...). Etykietami są liczby interwencji policji (w procentach w damym mieście, w danym roku - 1995).

1. Liczba kolumn = 128
2. Liczba wierszy = 1994
3. dtypes: float64(125), int64(2), object(1)
4. Pierwsze pięć kolumn nie są używane w budowaniu modelu
5. Liczba wartości brakujących:
    a. w 22 kolumnach dotyczących policji = 1675
    b. Community = 1177 (non-predictive)
    c. county = 1174 (non-predictive)
    d. OtherPerCap = 1
6. Wszystkie dane liczbowe zostały znormalizowane do zakresu dziesiętnego 0,00-1,00
7. Atrybuty zachowują swoją dystrybucję i pochylenie.
8. Wszystkie wartości powyżej 3 SD powyżej średniej są znormalizowane do 1,00; wszystkie wartości powyżej 3 SD poniżej średniej są znormalizowane do 0,00

Inne można przeczytać na stronie z dokumentacja

Metodologia i rozwiązanie

Dla rozwiązywania problemu były wykorzystane modele regresyjne.

Etapy rozwiązywania problemu:

1. Import danych
2. Badanie zmiennej docelowej
    - Macierz korelacji
    - Wykresy Zmienna Docelowa vs Top Skorelowane zmienne
3. Badanie wartości brakujących
    - Usunięcie non-predictive zmiennych

    - Zmiana wartości brakujących na 0
4. Budowanie modelów z top Skorelowanymi Kolumnami (linia bazowa)
5. Badanie Skośności (Skewness)
    - Zmienna docelowa
    - Inne zmienne
6. Budowanie modelów na tabele ze wszystkimi kolumnami
7. Selekcja cech
8. Feature Engineering (nie działa - nie poprawia wyników modelu)
    - Dokumentacja Jednak normalizacja nie zachowuje relacji między wartościami atrybutów (np. Nie miałoby sensu porównywanie wartości whitePerCap z wartością blackPerCap dla społeczności)
9. Optymalizacja Hiperparametrów modelów
    - Sprawdzanie RMSE Modelów z wybranymi Hiperparametrami
10. Predykcja końcowa na zbiorze treningowym
11. Predykcja końcowa na zbiorze testowym
12. Rezultat

Wstepne przetwarzanie danych.

Zostało wykonano 3 etapy przetwarzania danych

1. Na początku wszystkie wartości brakujące zostały zmienione z '?' na null oraz typy niektórych kolumn zostały zmienione na 'float64' ponieważ python zdefiniował ich typy jako 'obiekt', a z dokumentacji wiadomo jest, że wszystkie dane oprócz jednej kolumny są numeryczne.
2. Następnym etapem było rozwiązanie problemu związanego z wartościami brakującymi. Większość kolumn z wartościami brakującymi były związane z danymi o policji dla tego wszystkie brakujące wartości zostały uzupełnione jako 0. Ponieważ sugerowano, że komisariat po prostu nie istnieje na tym obszarze.
3. Ostatecznym etapem w przekształceniu danych, który został zrealizowany, było usunięcie w zmiennych. Został pobrany pierwiastek kubiczny z wartości zmiennej docelowej i wszystkich innych.

Metoda oceniania jakości modelu.

Główną miarą oceny modelu był RMSE, tak jak rozwiązujemy regresyjną problem.

Wyniki eksperymentalne + wykresy w razie potrzeby

Ostateczna wartość RMSE na zbiorze testowym stanowi 0,1167820864486548.

Eksperymenty

Wszystkie wyniki pokazane poniżej oraz w pliku excel są RMSE liczonym na danych treningowych z użyciem cross validation =10.

1. **RMSE (Tabela z top skorelowanymi zmiennymi):** - wyniki modeli przyjęto i wykorzystano jako linie bazową

Model Name	**RMSE (std)**
LR	0.1457 (0.0160)
LSO	0.1460 (0.0165)
RIDGE	0.1461 (0.0166)
ELNT	0.1495 (0.0156)

KR	0.1471 (0.0167)
DT	0.2046 (0.0169)
SVM	0.1441 (0.0177)
KNN	0.1546 (0.0189)
RF	0.1470 (0.0188)
ET	0.1467 (0.0182)
AB	0.1815 (0.0111)
GB	0.1462 (0.0171)
XGB	0.1557 (0.0168)
LGB	0.1496 (0.0178)

2. **RMSE (tabela ze wszystkimi kolumnami po transformacji (kolumny LEMAS z brakującymi danymi zostały usunięte))**
   podczas pracy z brakującymi danymi rozważano dwie opcje
   najpierw - usuwamy kolumny Lemas, a resztę wypełniamy 0
   drugi - zostawiamy kolumny Lemasa i wypełniamy 0

Model Name	**RMSE (std)**
LR	0.1114 (0.0069)
LSO	0.1088 (0.0061)
RIDGE	0.1091 (0.0064)
ELNT	0.1171 (0.0059)
KR	0.1093 (0.0065)
DT	0.1585 (0.0091)
SVM	0.1123 (0.0074)
KNN	0.1211 (0.0083)
RF	0.1119 (0.0064)
ET	0.1115 (0.0062)
AB	0.1201 (0.0073)
GB	0.1120 (0.0066)
XGB	0.1198 (0.0068)
LGB	0.1140 (0.0082)

Naprawdę było zrobiono i przeanalizowano dużo eksperymentów, i umieszczać tu wszystkie wyniki nie jest fajnym pomysłem dla tego w pliku zadanka jest umieszczony plik excel(Eksperementy.xlsx) ze wszystkimi wynikami testów.

Zostały rozważono opcje:

- wyboru najlepszych cech za pomocy różnych modelów.
- usunięcie asymetrii poprzez podzielenie zmiennych na dwie grupy, prawostronne i lewostronne oraz usunięcie w każdej grupie na różne sposoby poprzez podniesienie ich do potęgi, wyprowadzenie pierwiastków w różnych stopniach.
- podczas radzenia sobie z brakującymi danymi, usuwając kolumny Lemas lub nie, a następnie wybierając najlepsze zmienne pry pomocy różnych modelów.
- nadal podejmowano próby tworzenia nowych zmiennych poprzez analizę i łączenie istniejących (nieudane), ponieważ wszystkie zmienne zostały znormalizowane i wartości mieściły się w przedziale od 0 do 1. (Jednak normalizacja nie zachowuje

relacji między wartościami atrybutów (np. Nie miałoby sensu porównywanie wartości whitePerCap z wartością blackPerCap dla społeczności) – z dokumentacji)

- próbowano usunąć współliniowość w tabelach z już wybranymi zmiennymi przez pewien model
- najpierw jeden model wybierał zmienne, a następnie spośród nich wybierał drugi
- znalezienie najlepszych parametrów modelu

Najlepszą opcją okazało się uczenie modelów na różnych zbiorach danych (na których model wykazywał najlepsze wyniki) a następnie połączyć je i spróbować przewidzieć wynik na danych testowych.
Ostateczny wynik uzyskano poprzez połączenie wyników stack_model* 0,8 + LGB * 0,2
Stack_Model, który składa się z:
KernelRidge, ExtraTreesRegressor, RandomForestRegressor jako estymatory bazowe
oraz **LassoRegression** przy koniecznym ocenianiu.
LGB - LGBMRegressor.

Jeżeli Pani będzie chciała zobaczyć wyniki wszystkich eksperymentów zostawiamy link do pliku excel.