MPUM

Miniprojekt 2 – Sprawozdanie

Michał Horodecki

1 Kod

Projekt zawiera cztery pliki:

- `common.py` – definicje stałych
- `naive_bayes.py` – obliczanie naiwnego klasyfikatora bayesowskiego;
- `logistic_regression.py` – obliczanie wektora θ za pomocą regresji logistycznej
- `main.py` – rysowanie wykresów z tego sprawozdania

2 Przetwarzanie danych

Do wczytania i operacji na danych używamy biblioteki `pandas` aby nie pisać od zera trywialnych operacji.

Danych do tego zadania nie trzeba było specjalnie przetwarzać – naiwny bayes nie bierze pod uwagę wielkości cech a jedynie ich liczbę i liczbę przyjmowanych wartości, a dla regresji logistycznej dane są wystarczająco małe.

Trzeba jedynie podzielić dane na zbiór treningowy i testowy tak aby oba zawierały obserwacje pozytywne i negatywne w tym samym stosunku. W tym celu najpierw dzielimy dane na obserwacje pozytywne i negatywne a następnie z każdej części wybieramy 70% (477) obserwacji do zbioru treningowego i 30% (206) do testowego.

3 Naiwny klasyfikator bayesowski

3.1 Wyznaczanie

Mamy 9 cech, każda przyjmuje 10 wartości, zatem dostajemy $2 \cdot 9 \cdot 10 + 1 = 181$ parametrów. Formalnie wystarczyłoby trzymać dla każdej cechy tylko $2 \cdot 9$ zamiast $2 \cdot 10$, ale dodajemy parametr na ostatnią cechę dla uproszczenia obliczeń.

Wartości parametrów ϕ obliczane są wzorami:

$$\phi_y = \frac{1 + \sum_{i=1}^{m} \mathbf{1}\left[y^{(i)} = 4\right]}{2 + m}$$

$$\phi_{x_j = a \mid y = c} = \frac{1 + \sum_{i=1}^{m} \mathbf{1}\left[y^{(i)} = c \wedge x_j^{(i)} = a\right]}{2 + \sum_{i=1}^{m} \mathbf{1}\left[y^{(i)} = c\right]}$$

W związku z wygładzeniem Laplace'a dla każdego j mamy

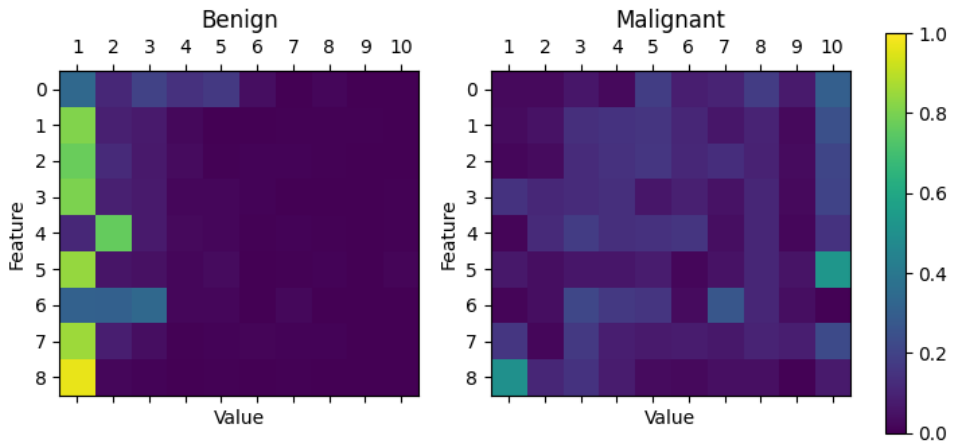
$$\sum_{i=1}^{9} \phi_{x_j = i \mid y = c} = 1.026$$

wartość ta jest na tyle blisko 1, że nie powinno stanowić to wielkiego problemu; model dobrze przewiduje dane.

3.2 Wyniki

$\phi_y = 0.35$ – czyli w ok. 1/3 przypadków rak był złośliwy.

Wartości $\phi_{x_j = a \mid y = c}$ najlepiej jest przedstawić na wykresie.



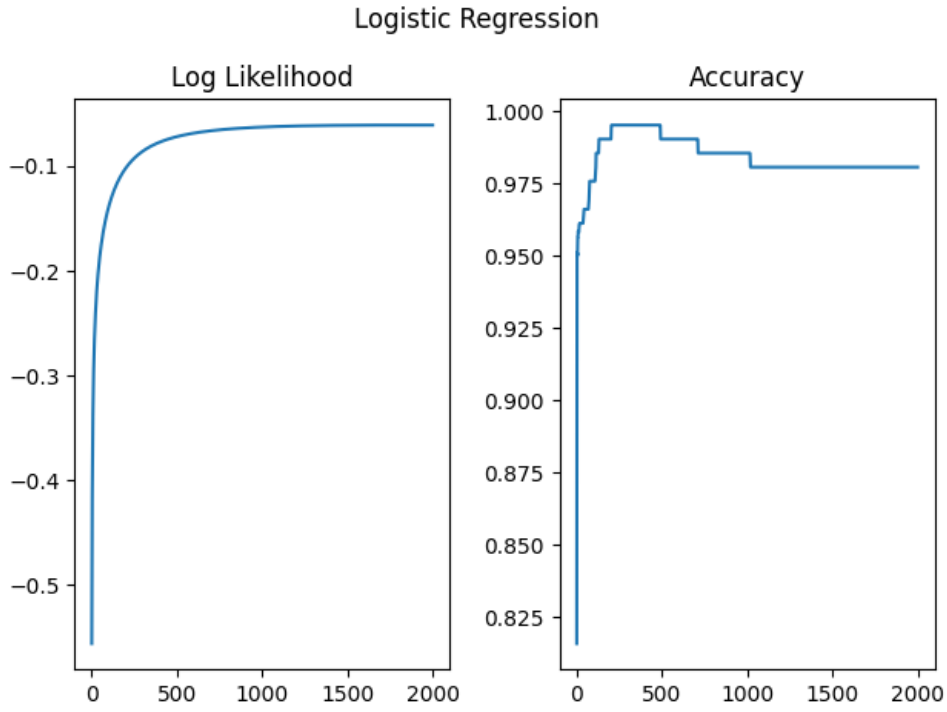
Widzimy że łagodny rak zachowuje się dużo bardziej przewidywalnie – wartości cech w większości przypadków wynoszą 1 lub 2, natomiast w przypadku złośliwego nowotworu są one dużo bardziej porozrzucane.

4 Regresja logistyczna

Zaczynamy od ustalenia learning rate $\alpha = 0.0005$ i znalezienia optymalnej długości treningu.

Ponadto zamiast liczyć zwykłej wiarygodności liczymy średnią log-wiarygodność tj.

$$L(\theta) = \frac{1}{m} \ln\left(\prod_{i=1}^{m} p(y^{(i)} \mid x^{(i)}, \theta)\right)$$



Rysunek 1: Zależność średniej wiarygodności i trafności względem liczby epok uśredniona na 20 przebiegach

Widzimy, że (dla zbioru testowego złożonego z 477 obserwacji) optymalnie jest skończyć po ok. 500 epokach – taka wartość została użyta przy późniejszych obliczeniach.

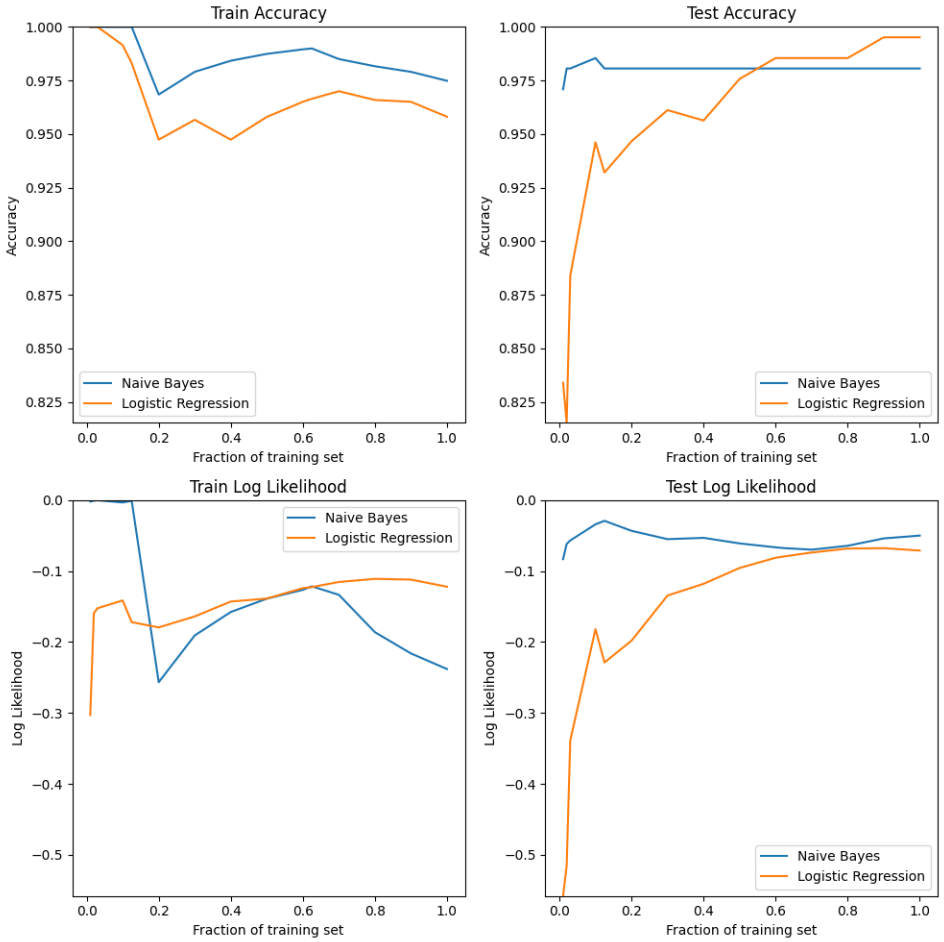
5 Porównanie

5.1 Końcowe wyniki

	train acc	train ll	test acc	test ll
NB	97.5%	-0.24	98.1%	-0.05
LR	95.6%	-0.12	99.5%	-0.07

Tabela 1: Trafność i średnia log-wiarygodność, uśrednione na 20 przebiegach

5.2 Rozmiar zbioru treningowego



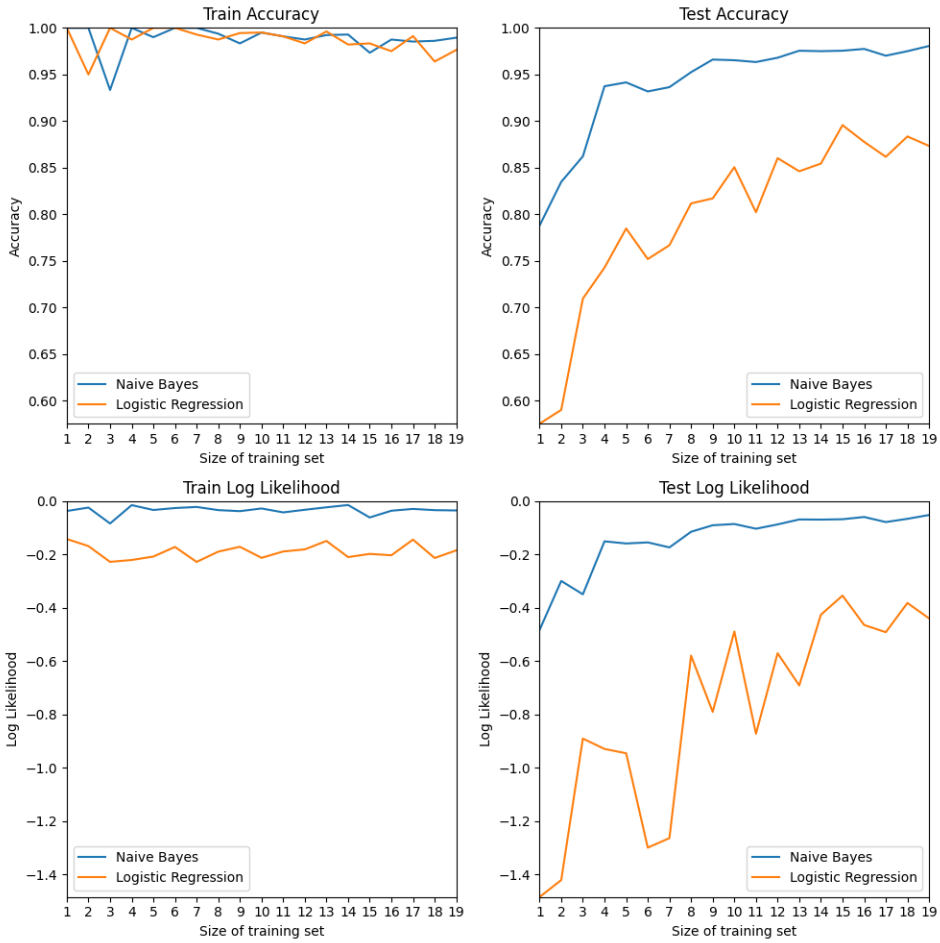
Rysunek 2: Zależność średniej wiarygodności i trafności względem rozmiaru zbioru treningowego, uśredniona na 20 przebiegach

5.3 Ekstremalnie mały zbiór treningowy

Widzimy, że nawet na 1% (tj. 4 elementach) zbioru treningowego naiwny Bayes radzi sobie zaskakująco dobrze, a dalsze zwiększanie zbioru treningowego nie poprawia trafności.

Nasuwa się więc pytanie – jak zachowują się obie metody na bardzo małych zbiorach treningowych?

Okazuje się, że już na zbiorze złożonym z pięciu elementów naiwny bayes osiąga skuteczność 95%, podczas gdy regresja osiąga tę samą skuteczność dopiero na zbiorze stuelementowym.



Rysunek 3: Zależność średniej wiarygodności i trafności względem rozmiaru małego zbioru treningowego, uśredniona na 20 przebiegach

6 Podsumowanie

Tak jak pisali Ng i Jordan – regresja logistyczna radzi sobie lepiej na większych zbiorach danych, podczas gdy na mniejszych naiwny klasyfikator bayesowski wygrywa.

Nasz zbiór danych jest mały, a ponadto cechy łagodnego nowotworu wynoszą w większości 1 lub 2, co sprawia, że znając ten rozkład bardzo łatwo jest rozróżnić czy rak jest złośliwy czy nie, dlatego naiwny klasyfikator bayesowski radzi sobie dużo lepiej.

Regresja logistyczna nie zna tych rozkładów i nadgania dopiero gdy rozważy większy zbiór danych.