[UMA] – Dokumentacja wstępna projektu

Uczenie aktywne (temat nr 1)

Michał Szwejk Damian D'Souza

1 Opis projektu

Celem projektu jest przygotowanie programu realizującego schemat uczenia aktywnego. Jest to technika, w której model sam wybiera przykłady uczące, które są najbardziej wartościowe dla jego treningu. Takie podejście pozwala minimalizować liczbę etykietowanych danych przy zachowaniu wysokiej skuteczności modelu. Aktywne uczenie jest szczególnie przydatne, gdy dostępnych jest niewiele danych, są one niezbalansowane lub mają zróżnicowaną dystrybucję.

W ramach projektu zostanie przygotowany interaktywny program przedstawiający użytkownikowi próbki w kolejnych seriach (np. o rozmiarze 10), które musi on manualnie poetykietkować zgodnie ze swoją oceną. Po każdej rundzie przypisania klas model ponownie uruchomi algorytm uczący, tym razem na zwiększonej liczbie przykładów w zbiorze trenującym. Proces jest iteracyjny, powtarzany tak długo, aż wszystkie dane nie zostaną poetykietowane.

2 Opis algorytmów

2.1 Random forest

Las losowy jest algorytmem, który generuje wiele różnych drzew w procesie uczenia i przypisuje klasy, będące dominantą wyników poszczególnych drzew. W odróżnieniu od pojedynczego drzewa tendencja modelu do nadmiernego dopasowania jest dużo mniejsza. Każde drzewo trenowane jest na tylu przykładach ile jest w zbiorze trenującym, jednakże losowane są one ze zwracaniem. Ponadto ograniczony jest zestaw atrybutów, wybierane jest ich wyłącznie $\sqrt{|A|}$ (bez zwracania), gdzie A – to zbiór atrybutów.

Każde drzewo wchodzące w skład lasu losowego jest drzewem typu *CART*. Reguły podziału w poszczególnych węzłach są konstruowane w taki sposób, aby po podziale minimalizować wartość **zanieczyszczenia Gini'ego** (w przypadku zadania klasyfikacji). Miara ta jest dana wzorem:

$$Gini(x) = \sum_{i=0}^{|C|} 1 - p_i^2$$

gdzie p_i – prawdopodobieństwo i-tej klasy w węźle, a C – zbiór klas.

2.2 SVM

Maszyna wektorów nośnych jest algorytmem klasyfikacji, który wyznacza optymalną hiperpłaszczyznę rozdzielającą klasy w przestrzeni cech. Dla problemu binarnego hiperpłaszczyzna jest definiowana przez wektor wag $\mathbf{w}$ i wyraz wolny b, a klasyfikacja nowej próbki $\mathbf{x}$ odbywa się na podstawie znaku funkcji decyzyjnej $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b$.

Kluczową ideą SVM jest maksymalizacja marginesu, czyli odległości między hiperpłaszczyzną a najbliższymi punktami każdej z klas (tzw. wektorami nośnymi). W przypadku danych nieliniowo rozdzielnych wprowadza się parametr kary C, który kontroluje kompromis między maksymalizacją marginesu a minimalizacją błędu klasyfikacji. Proces optymalizacji wag może zostać przeprowadzony m.in. metodą spadku gradientu, poprzez minimalizację funkcji celu zawierającej **funkcję straty zawiasowej**:

$$L(\mathbf{w}, b) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\max(0, 1 - y_i(\mathbf{w}^T\mathbf{x}_i + b))$$

gdzie $y_i \in \{-1, 1\}$ – etykieta i-tej próbki, a C – parametr kary.

W sytuacjach wymagających probabilistycznego wyjścia modelu (co jest kluczowe np. w uczeniu aktywnym), możliwe jest zastosowanie metody **Platta**. Pozwala ona na dopasowanie funkcji sigmoidalnej do wartości funkcji decyzyjnej SVM, co umożliwia przekształcenie ich w oszacowania prawdopodobieństwa:

$$P(y = 1|\mathbf{x}) = \frac{1}{1 + \exp(A \cdot f(\mathbf{x}) + B)}$$

gdzie parametry A i B wyznaczane są poprzez minimalizację ujemnej logarytmicznej funkcji wiarygodności na zbiorze treningowym.

3 Plan eksperymentów

Do ewaluacji jakości modelu wykorzystane zostaną dwa wskaźniki – precyzja (ang. *precision*) i czułość (ang. *recall*). Podane miary wyliczane są następująco:

$$precision = \frac{TP}{TP + FP} \qquad recall = \frac{TP}{TP + FN}$$

gdzie oznaczenia są zgodne ze standardowymi oznaczeniami w macierzy pomyłek (tzn. TP – *true positive* itd.).

Zależności między między precyzją a czułością można uchwycić na krzywej *precision-recall*, która powstaje poprzez odcinanie kolejnych wartości progów klasyfikatora. Krzywa dobrze pokazuje kompromis: zwiększając czułość (wykrywamy więcej pozytywnych przypadków), zwykle

obniżamy precyzję (więcej fałszywych alarmów). Jest ona szczególnie użyteczna przy niezrównoważonych zbiorach danych, ponieważ skupia się na jakości predykcji klasy pozytywnej, a nie na licznych przykładach klasy większościowej.

W ramach eksperymentów, na różnych etapach uczenia (poetykietowane jest: 25%, 30%, 40%, 50% lub 100% danych) odpowiednie krzywe zostaną wyznaczone i zobrazowane na wykresach. Liczony będzie także wskaźnik **PR AUC**, który ocenia jakość modelu.

Dodatkowo proces uczenia aktywnego zostanie zrealizowany w dwóch warianatach. W każdym z nich model będzie wybierał próbki przeznaczone do anotacji na podstawie następujących kryteriów:

- największa niepewność predykcji (ang. *uncertainty sampling*):

$$\max_{x \in P} \left(1 - \max_{\hat{y} \in C} P(\hat{y}|x)\right)$$

- największa różnorodność (ang. *diversity sampling*):

$$\max_{x \in P} \left(\min_{x' \in T} \delta(x, x')\right)$$

gdzie P – zbiór danych niepoetykietowanych, T – zbiór trenujący, $\delta(x, x')$ – miara podobieństwa między dwoma próbkami (np. euklidesowa).

4 Zbiór danych

Jako zbiór danych wykorzystany zostanie zestaw zawierający zdjęcia kwiatów. Przed przystąpieniem do przeprowadzenia eksperymentów zostanie on wstępnie przetworzony w następujący sposób:

- Pozostawione zostaną wyłącznie próbki zklasyfikowane jako mniszki lekarskie lub słoneczniki. W ten sposób klasyfikacja zostanie ograniczona do binarnej;
- Usunięte zostanie większość próbek jednej z klas, dzięki czemu zbiór stanie się mocno niezrównoważony (pożądany stosunek powinien być większy niż 1:10);
- Dla 60% początkowego zbioru danych początkowe etykiety zostaną usunięte. Pozostałe próbki zostaną podzielone w proporcji 1:1 na startowy zbiór treningowy i zbiór testowy;
- Zdjęcia zostaną sprowadzone do postaci wektora cech dzięki wcześniej wytrenowanej konwolucyjnej sieci neuronowej (np. *RestNet*, *VGG16*), a następnie wprowadzone do przygotowanych modeli. Takie rozwiązanie umożliwi sprawniejsze etykietowanie danych. Użytkownik nie będzie musiał patrzeć w dane numeryczne, wystarczy, że spojrzy na zdjęcie i przypisze mu odpowiednią klasę.