Concept drift & IML

Mateusz Staniak

Warszawa, 15 X 2018

Plan

1. Concept drift

2. Wyjaśnienia przy pomocy EXPLAIN i IME

3. Zastosowanie wyjaśnień do wykrywania dryfu modelu

Concept drift

Problem

- Najogólniej: problemy obliczeniowe wywołane zmianami danych w czasie.
- Inne określenia: dataset shift / covariate shift (pattern recognition), niestacjonarność (przetwarzanie sygnałów).
- Róże możliwe przyczyny zmian: zmiana otoczenia (np. rynkowego), populacji (np. migracja), celowych działań (np. spam).
- Formalnie: zmiana w rozkładzie łącznym $\mathbb{P}(X, y)$.
- Wniosek: może to być zmiana w rozkładzie X, y lub $y|X$.
- Strumienie danych i uczenie przyrostowe (data streams & incremental learning).

Przykłady

1. Wykrywanie spamu: celowe zmiany treści spamu w celu oszukania systemów filtrujących spam.
2. Modelowanie zachowania użytkowników: ukryte zmienne - intencje użytkowników - zmieniają się z czasem (tak jak ich przejawy).

Typowe podejścia

Monitorowanie

- zmiany jakości modelu (performance),
- zmiany rozkładu zmiennych.

Zastosowanie

- pojedynczego modelu - poprzedni model zostaje zapomniany,
- grupy modeli (ensemble) - pamięć o poprzednich modelach - trenowanie nowego modelu, wybór najbardziej właściwego z grupy lub agregacja decyzji modeli.

Ważne techniki

- Test Page-Hinley
- Statistical Process Control

Wyjaśnienia przy pomocy EXPLAIN i IME

Tło

- Pierwsza publikacja: 2008 r. (Robnik-Sikonja, Kononenko).
- Metody dekompozycji predykcji klasyfikatorów.
- IME została ponownie odkryta przez Lundberga i in.

Metoda EXPLAIN

Dla każdej klasy y_k i każdej zmiennej x_i obliczamy

$$p(y_k|x) - p_{S\setminus\{x_i\}}(y_k|x),$$

gdzie S jest zbiorem wszystkich zmiennych w modelu.

Metoda EXPLAIN

Estymacja $p_{S\setminus\{x_i\}}(y_k|x)$:

- jeżeli x_i jest jakościowa,

$$\sum_{s=1}^{m_i} p(x_i = a_s)p(y_k|x, x_i = a_s)$$

- jeżeli x_i jest ilościowa, dyskretyzujemy ją na m_i przedziałów A_s i obliczamy

$$\sum_{s=1}^{m_i} p(x_i \in A_s)p(y_k|x, x_i = \overline{A_s}),$$

gdzie $\overline{A_s}$ oznacza środek przedziału A_s.

Metoda IME

- Aktualnie znana jako Shapley Values.
- Idea:
  - zmienne: gracze w grze koalicyjnej,
  - wypłata: różnica pomiędzy wyjaśnianą predykcją i średnią predykcją modelu,
  - cel: sprawiedliwe rozdzielenie wpływu zmiennych między nimi według ich wkładu do tej wypłaty
- Ta metoda ma dobre teoretyczne własności.
- Dokładne obliczenie wartości Shapley'a jest jest złożone obliczeniowo (2^p podzbiorów zmienych - koalicji), stosuje się przybliżenie MC.

Zastosowanie wyjaśnień do wykrywania dryfu modelu

Ogólnie

- Idea: zamiast obserwować rozkłady zmiennych, monitorować zmiany w wyjaśnieniach modelu (globalnych).

Trzy wersje algorytmu:

- ExStreamModel: porównanie średnich różnic między wyjaśnieniami (jednowymiarowy strumień).
- ExStreamAttr: porównanie różnic między wyjaśnieniami każdej zmiennej (p-wymiarowy strumień).
- ExStreamVal: porównanie wpływu każde wartości zmiennej.

Algorytmy

Algorithm 1: Outline of the *ExStreamModel* variant.

```
// Run incremental classification with model h and
   current example x_t
incremental_learn(h, x_t)
// Run explanation according to granularity g
if t mod g == 0          // time to compute explanation
then
    // Use IME to explain the current model h for class
       c
    φ_t ← (explain(h, c)) // φ_t - explanation vector at
       timestamp t
    // Calculate average dissimilarity d to other
       explanations
    dissimilarities.append(AVG(d(explanations, φ_t)))
    // Monitor the stream of dissimilarities for concept
       drift detection, retrain on change.
    δ(dissimilarities); δ ∈ {SPC, PH}
```

Rysunek 1: Algorytm 1: ExStreamModel

Algorytmy

Algorithm 2: Outline of the *ExStreamAttr* variant.

```
// Run incremental classification with model h and
   current example x_t
incremental_learn(h, x_t)
if t mod g == 0          // time to compute explanation
then
    φ_t ← (explain(h, c))
    // Calculate explanation dissimilarity for each of n
       attributes
    for i = 1 to n do
        dissimilarities[i].append(d(explanations, φ_t[i]))
    // Monitor the dissimilarities for each attribute
       separately, retrain on alert of any attribute
    for i = 1 to n do
        δ(dissimilarities[i])
```

Rysunek 2: Algorytm 2: ExStreamAttr

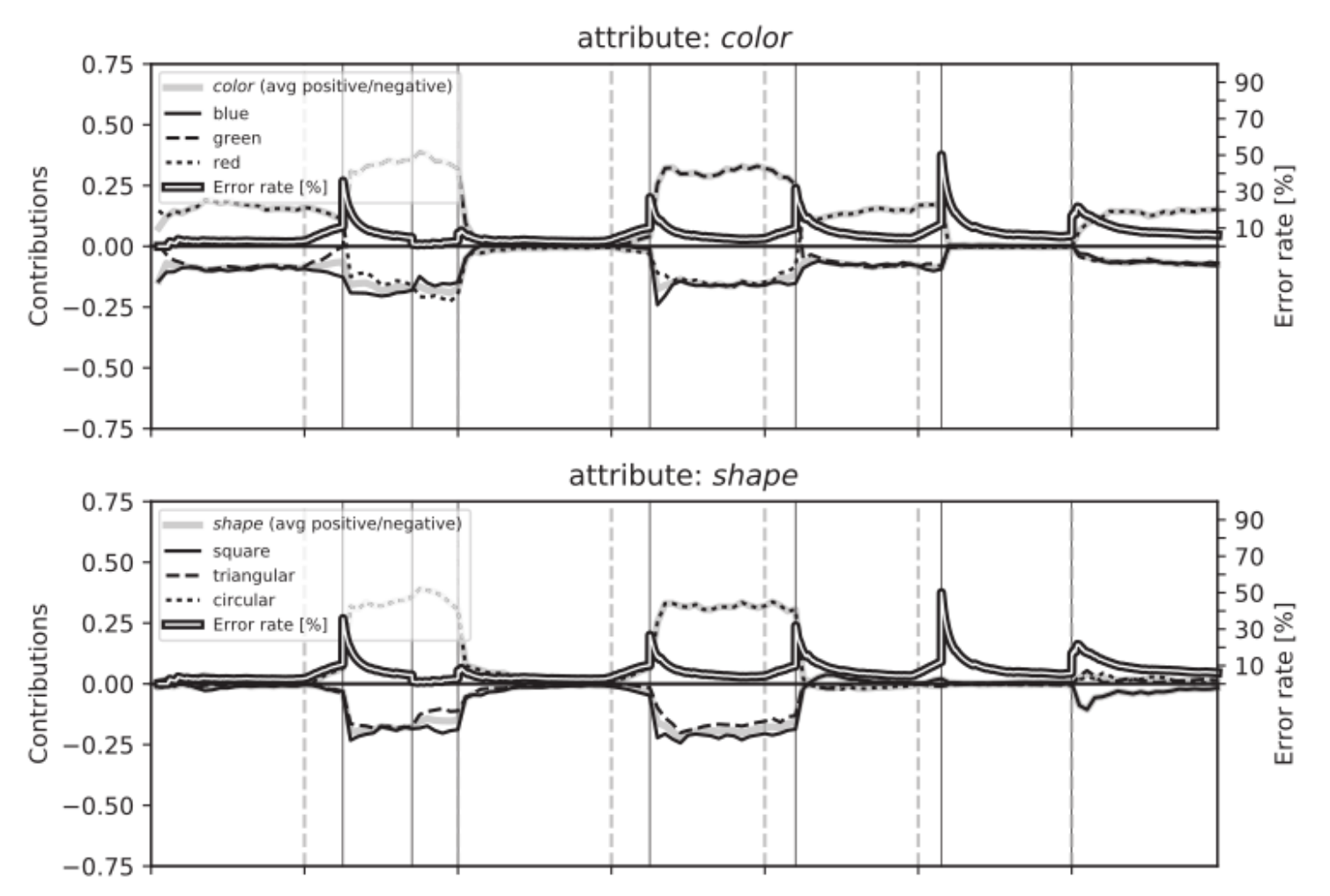
Algorytmy

Algorithm 3: Outline of the *ExStreamVal* variant.

```
// Run incremental classification with model h and
   current example x_t
incremental_learn(h, x_t)
if t mod g == 0          // time to compute explanation
then
    φ_t ← (explain(h, c))
    for atr ∈ attributes do
        for attr_value ∈ atr do
            // Monitor the stream of contributions for
               each attribute value, retrain on alert of
               any attribute value
            δ(φ[attr_value])
```

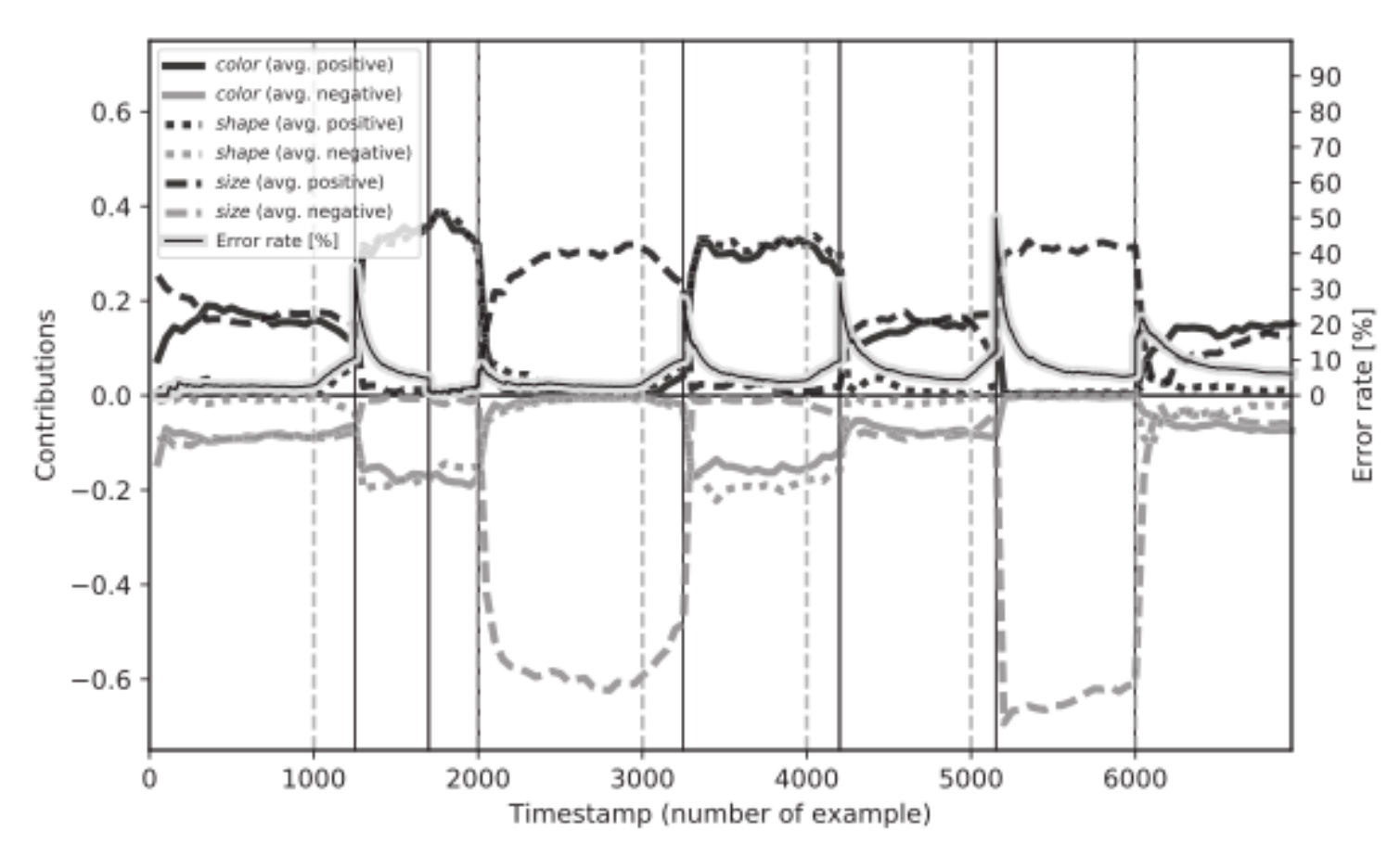
Rysunek 3: Algorytm 3: ExStreamVal

Wizualizacja



Rysunek 4: Wizualizacja zmian wyjaśnień w czasie dla pojedynczej zmiennej

Wizualizacja



Rysunek 5: Wizualizacja zmian wyjaśnień w czasie dla pojedynczej zmiennej

Porównanie z innymi metodami

Metody odniesienia:

- test P-H,
- SPC.

Miary:

- error rate,
- false alarm rate (odporność),
- mean time between false alarms,
- false alarm count,
- mean time to detection,
- missed detection rate.

Problemy:

- czułość na szerokość dryfu (drift width),
- wpływ szumu,
- wpływ anomalii sensorów,
- wpływ rozrzedzenia przyrostów.

Wnioski z symulacji

- Metody oparte na wyjaśnieniach są lepsze w sensie MTR, ale nie w sensie czasu wykrycia zmiany ani liczby fałszywych alarmów.
- Istotną zaletą metod opartych na wyjaśnieniach jest ich zrozumiałość (comprehensibility).
- Metody oparte na wyjaśnieniach są dość odporne na różnice w ”szerokości” dryfu.
- Wyjaśnienia dobrze reagują na szum przy anomaliach (kontrybucje równe 0).
- 50-200 obserwacji to najlepsze okno dla badanych metod.
- Algorytm ExStreamModel oparty na SPC okazał się najlepszą metodą.

Literatura i

J. Demar and Z. Bosni.
Detecting concept drift in data streams using model explanation.
Expert Syst. Appl., 92(C):546–559, Feb. 2018.

J. Demsar, Z. Bosnic, and I. Kononenko.
Visualization and concept drift detection using explanations of incremental models.
Informatica (Slovenia), 38(4), 2014.

M. Robnik-Sikonja.
Explanation of prediction models with explainprediction.
Informatica (Slovenia), 42(1), 2018.

Literatura ii


M. Robnik-Sikonja and I. Kononenko.
Explaining classifications for individual instances.
IEEE Trans. Knowl. Data Eng., 20(5):589–600, 2008.


E. Strumbelj and I. Kononenko.
An efficient explanation of individual classifications using game theory.
Journal of Machine Learning Research, 11:1–18, 2010.


I. Žliobaitė, M. Pechenizkiy, and J. Gama.
***An Overview of Concept Drift Applications*, pages 91–114.**
Springer International Publishing, Cham, 2016.