Projekt 3 - Metody klasyfikacji i redukcji wymiaru

Klaudia Weigel

1 Wstęp

Celem projektu jest klasyfikacja urządzeń domowych na podstawie danych dotyczących poboru prądu. Dane są fragmentem zbioru REDD `http://redd.csail.mit.edu/`. Format pliku z danymi jest następujący:

```
time,lighting2,lighting5,lighting4,refrigerator,microwave
1302930703,180,23,195,117,2
1302930721,181,23,195,119,2
1302930738,180,23,195,117,2
1302930765,181,23,195,117,2
1302930782,180,23,195,118,2
```

Do klasyfikacji poszczególnych urządzeń zostaną użyte ukryte modele Markowa (HMM, Hidden Markov Models).

2 Ukryte modele Markowa

Definicja 2.1. (Łańcuch Markowa). Niech $T = \{0, 1, 2, ..., N\}$. Proces stochastyczny $\{X_n, n \in T\}$ nazywamy *łańcuchem Markowa* jeśli

$$P(X_{n+1} = j | X_n = i, X_{n-1} = i_{n-1}, ..., X_0 = i_0) = P(X_{n+1} = j | X_n = i).$$

Zbiór wszystkich możliwych przyjmowanych stanów oznaczamy przez $\mathcal{S}$ $(j, i, i_{n-1}, ..., i_0 \in \mathcal{S})$.
Jeśli łańcuch Markowa jest *jednorodny w czasie* to dodatkowo

$$P(X_{n+1} = j | X_n = i) = P(X_1 = j | X_0 = i).$$

Łańcuch Markowa można opisać porzez tzw. macierz przejść $\boldsymbol{P}$, określoną

$$\boldsymbol{P} = (P_{ij}),\ \ P_{ij} = P(X_{n+1} = j | X_n = i).$$

Proces Markowa jest rozszerzeniem łańcucha Markowa do ciągłej przestrzeni czasowej, przestrzeń stanów dalej pozostaje dyskretna.

Ukryty model Markowa (HMM) jest szczególnym przypadkiem procesu/łańcucha Markowa, w którym proces mający własność Markowa jest nieznany. Zamiast tego znamy tylko wartości wyjściowe (obserwacje), których rozkład prawdopodobieństwa zależy od stanu w którym znajduje się ukryty proces Markowa.

Definicja 2.2. (HMM). Niech $\mathcal{S} = \{s_1, s_2, ..., s_N\}$ będzie skończoną przestrzenią (ukrytych) stanów, oraz niech $\mathcal{V} = \{v_1, v_2, ..., v_M\}$ będzie skończonym zbiorem obserwacji. Zdefiniujmy $\mathcal{Q} = q_1 q_2 ... q_T$ jako ustalony ciąg T stanów oraz niech $O = o_1 ... o_T$ będzie odpowiadającym $\mathcal{Q}$ ciągiem obserwacji.
$\lambda = (\boldsymbol{A}, \boldsymbol{B}, \mu)$ jest *ukrytym modelem Markowa* jeśli

1. $\boldsymbol{A}$ jest macierzą przejść pomiędzy ukrytymi stanami

$$\boldsymbol{A} = (A_{ij}), \quad A_{ij} = P(q_t = s_j | q_{t-1} = s_i).$$

2. $\boldsymbol{B}$ jest macierzą warunkowych prawdopodobieństw obserwacji pod warunkiem stanu

$$\boldsymbol{B} = [b_i(k)], \quad b_i(k) = P(o_t = v_k | q_t = s_i).$$

3. μ jest rozkładem początkowym ukrytych stanów

$$\mu = [\mu_i], \quad u_i = P(q_1 = s_i).$$

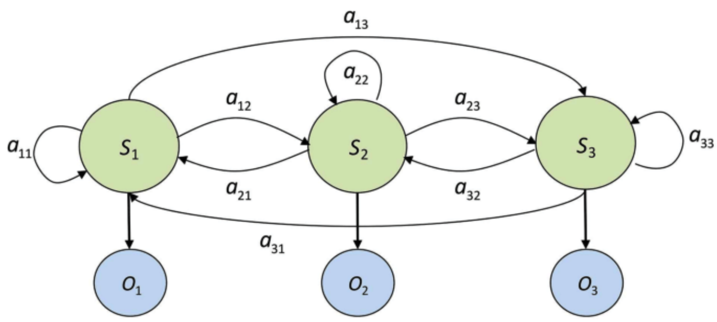
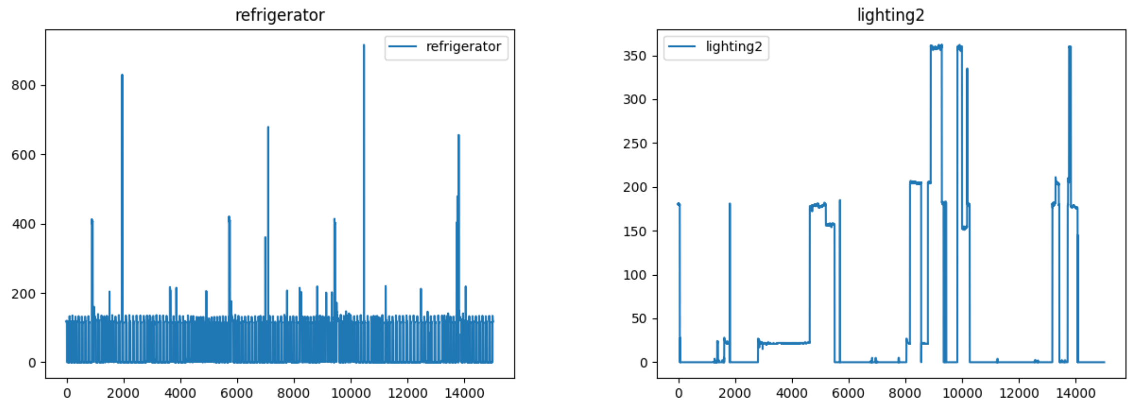


Figure 1: Diagram ilustrujący ukryty model Markowa. s_1, s_2, s_3 to ukryte stany modelu, o_1, o_2, o_3 to wyemitowane obserwacje a a_{ij} to prawdopodobieństwa przejśc między stanami.

Powyżej opisana definicja ukrytego modelu Markowa dotyczy przypadku, kiedy przestrzeń stanów jest dyskretna, ale może ona być rozszerzona, aby uwzględniać ciągły rozkład obserwacji. Czyli dalej mamy $\mathcal{S} = \{s_1, s_2, ..., s_N\}$, ale teraz $\mathcal{V}$ jest zbiorem ciągłym. Niech $\mathcal{V} = \mathbb{R}^d$. Zamiast macierzy $\boldsymbol{B}$ mamy pewne rozklady ciągłe $p_i(x), i = 1, ..., N, x \in \mathbb{R}^d$. Dla przykładu obserwacje mogą pochodzić z rozkładu gaussowskiego, wtedy $o_i \sim p_{q_i}(.) = N(\boldsymbol{\mu}_{q_i}, \boldsymbol{\Sigma}_{q_i})$.

Z ukrytymi modelami Markowa zwiazane są trzy fundamentalne problemy:

1. Mając obserwacje $O = o_1 ... o_T$ i model $\lambda = (\boldsymbol{A}, \boldsymbol{B}, \mu)$ jak obliczyć $P(O|\lambda)$ - prawdopodobieństwo obserwacji, pod warunkiem modelu (The forward-backward procedure).
2. Mając obserwacje $O = o_1 ... o_T$ i model $\lambda = (\boldsymbol{A}, \boldsymbol{B}, \mu)$, jak wybrać ciąg $\mathcal{Q} = q_1 q_2 ... q_T$, który najlepiej wyjaśnia obserwacje (Algorytm Viterbiego).
3. Jak dobrać model $\lambda = (\boldsymbol{A}, \boldsymbol{B}, \mu)$, aby zmaksymalizować $P(O|\lambda)$, dla danych obserwacji $O = o_1 ... o_T$ (Algorytm EM = algorytm Bauma-Welcha).



(a) Pobór prądu dla lodówki.

(b) Pobór prądu dla światła 2.

Figure 2: Przykładowe wykresy poborów prądu.

3 Rozwiązanie problemu

W naszym problemie mamy 5 możliwych urządzeń. Chcemy wyuczyć model na zbiorze uczącym, tak aby podając dane testowe zostały one sklasyfikowane jako jedno z urządzeń.

Aby dokonać klasyfikacji za pomocą ukrytych łańcuch Markowa postąpimy następująco

- Dla każdego z urządzeń 1=ligthing2, 2=lighting5, 3=lighting4, 4=refrigerator, 5=microwave wyuczamy się osobno pięciu róznych modeli HMM $\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5$.
- Dla każdego modelu λ_i próbujemy różne ilości stanów ukrytych. Jakość danego λ_i dla różnych ilości stanów ukrytych porównujemy za pomoca funckji wiarogodności. Ostatecznie wybieramy model, który najlepiej pasuje do zbioru uczącego.
- **Klasyfikacja**. Dla każdego z pięciu wyuczonych modeli $\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5$ sprawdzamy jak dobrze pasuje do niego dany zbior testowy. Klasyfikujemy zbiór testowy jako pochadzący od urządzenia i, jeśli największe likelihood zostało osiagnięte dla modelu, który odpowiada temu urządzeniu.

Ponieważ w naszym przypadku chcemy dobrać model $\lambda = (\boldsymbol{A}, \boldsymbol{B}, \mu)$, aby zmaksymalizować fukcję wiarogodności, to mamy do czynienia z problemem 3, opisanym w porzednim rozdziale. Przestrzeń stanów obserwacji (pobór mocy) jest ciągła.

4 Implementacja

4.1 Uruchomienie programu

Program należy uruchomić następującym poleceniem:

```
python3 NAZWA_PROGRAMU --train train_file.csv --test test_folder --output results.txt
```

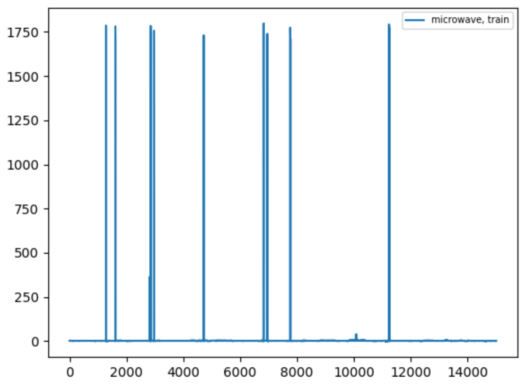
Gdzie `train_file.csv` to plik zawierajacym dane uczące, domyślnie jest to `house3_5devices_train.csv`, `test_folder` to nazwa folderu który zawiera pliki które należy sklasyfikować jako jedno z urządzeń, format plików testowych jest następujący:

```
time, dev
1303001413, 0
1303001430, 0
1303001487, 134
1303001509, 132
1303001526, 131
```

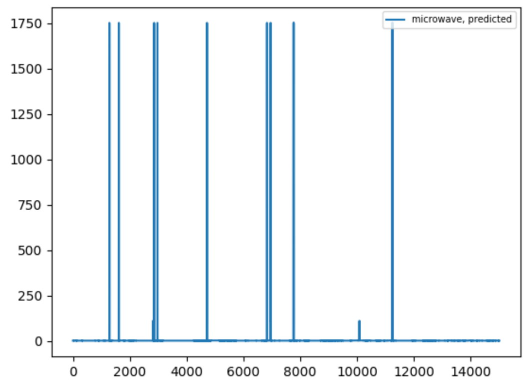
Do pliku `results.txt` zostaną zapisane wyniki w następujacym formacie:

```
file, dev_classified
dev1.csv, ligthing2
dev2.csv, ligthing2
dev3.csv, refrigerator
dev4.csv, microwave
dev5.csv, lighting5
dev6.csv, lighting4
```

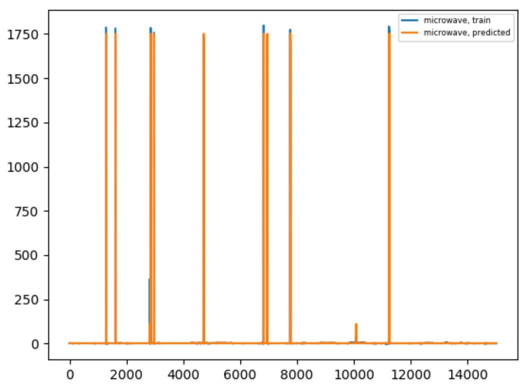
4.2 Szczegóły implementacji



(a) Pobór prądu dla lodówki, dane uczące.



(b) Pobór prądu lodówki, wartości przewidziane przez wytrenowany model.



(c) Pobór prądu lodówki, wartości przewidziane(predicted) i oryginalne.

Do implementacji użyjemy biblioteki `hmmlearn` `https://hmmlearn.readthedocs.io/en/latest/`. Dostępne w bibliotece ciągłe rozkłady obserwacji to rozkład gaussowski (`GaussianHMM`) oraz mieszanka rozkładów gaussowskich(`GMMHMM`). Aby obliczyć jak dobrze model pasuje do

danych możemy użyc wbudowanej funkcji `score`, która liczy log funkcji wiarogodności. Wśród innych metod znajduja się też funkcje `means_` i `covars_` które podają średnie i kowariancje rozkładów gaussowskich dopasowanych do każdego stanu. Macierz prawdopodobieństw przejścia można otrzymać przy użyciu `transmat_`.
Jednym z parametrów jakie trzeba podać przy tworzeniu modelu jest ilość ukrytych stanów. Ponieważ algorytm EM szuka lokalnego maksimum funkcji wiarogodności, aby otrzymać najbardziej optymalne rezultaty warto jest przetrenować model kilkakrotnie, zmieniając ilość ukrytych stanów.

Wyniki fukcji `score`, dla modelu z rozkładami gaussowskimi (`GaussianHMM`) dla światła 2:

```
n_components, score
2, -60152.68
3, -54789.37
4,  35426.73
5,  37117.89
6,  42926.5
7,  38633.94
8,  38882.54
9,  39756.24
10,  46768.46
```

Zamiast liczyć likelihood możemy także skorzystać z bardziej dokładnej metryki BIC (Bayesian information criterion), ktora uwzględnia fakt, że większa ilość parametrów zwiększa wartość fukcji wiarogodności, ale tym samym sprawia że model jest mocno dopasowany do danych uczących (overfitting). BIC liczy się następująco (im mniejszy BIC tym lepiej dopasowany model)

$$BIC = kln(n) - 2ln(\mathcal{L}(\boldsymbol{A}, \boldsymbol{B}, \mu))$$

gdzie

- $\mathcal{L}$ to wartość funkcji log wiarogodności dla parametrów $\boldsymbol{A}, \boldsymbol{B}, \mu$,
- k to ilość parametrów modelu,
- n to to ilość obserwacji.

W naszym przypadku obserwacje to pojedyncze punkty, więc mamy do czynienia z rozkładem gaussowskim jednowymiarowym, zatem ilość parametrów modelu to ilość średnich + ilość wariancji + ilość elementów macierzy przejścia(= #ukryte stany(#ukryte stany - 1)) + ilość początkowych prawdopodobieństw(= #ukryte stany - 1).
Wartości BIC dla światła 2

```
n_components, score
2 120372.66300460015
3 109713.34312655342
4 -70632.2982453539
5 -73908.82681186369
6 -85401.05598769839
7 -76671.71066336022
8 -77005.43910185911
9 -78570.13796371628
10 -92392.64538883281
```