# SPADE Algorithm implementation

Data Mining (CS F415) Comprehensive assignment

Ravi Bharadwaj C 2016AAPS0244H

## **Data description and Pre-processing**

#### Data description:

- Sequential dataset with all sequences having one element per event
- Dataset contains information regarding order in which users visit webpages in a website

#### Pre-processing:

• The only major pre-processing step was to convert the data into vertical format suitable for the algorithm

#### Key points about implementation

Three steps involved in SPADE-

- 1. Generate frequent 1-sequences
- 2. Generate frequent 2-sequences
- 3. Generate frequent n-sequences recursively

The code was implemented by utilizing the pandas library and is faster than the naive loop based implementation of the other publically available python implementation

### Key points about implementation

The code takes too long to run on the entire dataset (greater than 5 hours). To address this issue, the dataset is sampled at random.

The algorithm was multiple times run on a different 20% sample of the data and correct results were obtained on every instance. Moreover, the results obtained on every sample were nearly the same.

#### Results

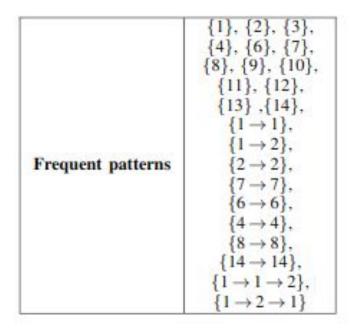
The frequent sequences generated are shown in Table I.

Parameters used:

- $min_sup = 0.05$
- 20% of the dataset sampled at random

Increasing the min\_sup reduced the number of frequent sequences greatly and a min\_sup of 0.1 gave only 1 frequent sequence of length 2

TABLE I FREQUENT PATTERNS GENERATED BY SPADE



#### **Drawbacks**

- The code takes too long to run on the entire dataset. It takes 5-6 hours for the entire dataset
- This code base can only run on sequential datasets in which all the sequences have single element events

## **Thank You**

GitHub repo Link to colab notebook