Mining massive Datasets WS 2017/18

Problem Set 11

Rudolf Chrispens, Marvin Klaus, Daniela Schacherer

January 22, 2018

Exercise 01

a) Let w be the size of the sliding window. One can then compute the arithmetic mean (remember the sum over the stream elements p and w) over the first w sized window. As the window slides for one position one can subtract the first element (which is not in the window any more) from the sum, add the new element and again divide by w. This will result in 3 operations per new stream element.

b)
1. If $p_{new} = p - o + n < p \Rightarrow n < o$
2. $p_{new} = \frac{p*w-o^2+n^2}{w}$

c) Stream 1:

$$var_{t=0} = 20$$
$$var_{t=1} = 88$$

Stream 2:

$$var_{t=0} = 20$$
$$var_{t=1} = 130$$

To compute the variance over a sliding window we have to keep track of w, the sum of the values, and the sum of the squares of the values like explained in a). Then one can compute the variance by $\sum_{i=1}^{n}(x_i - \vec{x})^2 = [\sum_{i=1}^{n} x_i^2] - 1/n[\sum_{i=1}^{n} x_i]^2$

d)

Exercise 02

Exercise 03

$$M = \begin{bmatrix} 1/3 & 1/2 & 0 \\ 1/3 & 0 & 1/2 \\ 1/3 & 1/2 & 1/2 \end{bmatrix}$$

$$r^{(0)} = \begin{pmatrix} 1/3 \\ 1/3 \\ 1/3 \end{pmatrix}, r^{(1)} = \begin{pmatrix} 0.2777 \\ 0.2777 \\ 0.444 \end{pmatrix}, r^{(2)} = \begin{pmatrix} 0.231 \\ 0.315 \\ 0.454 \end{pmatrix}, r^{(3)} = \begin{pmatrix} 0.235 \\ 0.304 \\ 0.461 \end{pmatrix}$$

At this point $\|r^{(3)}\|_1$ is 0.0216 which is smaller than $\frac{1}{12} = \epsilon$.

Exercise 04

MapReduce:

```
map(inputTuple):
        word = inputTuple[1]
        numberLocals = word.count("a", "e", "i", "o", "u") # count and return the
            ↪ number of vowels in a word
        return(numberLocals, word)

reduce(tuple, list):
        insertTupleInSortedList(tuple, list)

main(input):
        list = []
        m = map(input)
        reduce(m, list)
        print(list[:10])
```

Spark:

```
def countVowels(word):
        vowels = 0
        vowels = word.count("a") + word.count("e") + word.count("i") + word.count("o
            ↪ ") + word.count("u")
        return (vowels, word)

topElements = rdd.map(countVowels)\
```

```
                    .sortByKey(ascending=False)
                    .top(10)
```

Exercise 05

It may happen (but it is very unlikely) that in the phase when the k initialized clusters are recomputed with the points assigned to it, two cluster centers collapse. Thus fewer than k clusters can be the output.

Exercise 06

Exercise 07

a) PCA is a method for compressing a dataset with a lot of dimensions into new data that captures the essence o the original data. The resulting dataset is typically shown in under 4 dimensions to preserve comprehensibility. PCA looks into the dimensions and if the variations are not as "strong" as in the other dimensions the get "reduced". In the end PCA delivers Principle Components ordered with the "importance". (PC1 most important, PCn n important)

b)

c)