WS 2016 Project 1

dpj482 Christian Edsberg MÃ¿llgaard

1. INTRODUCTION

In the previous study of Hansen et al.[4], we observed overall low error rate when predicting vaccination update using web mined and clinical data. This following study presents the same experiment, but only using web mined data from descriptions of the vaccines from ssi.dk[2] and sundhed.dk[3]. The work is done using 2 out of the 13 vaccines used in Hansen et al.[4]. The web mines data is presented by the amount of times people Googled each word.

I chose the vaccines MMR-2(12) and HPV-1.

The prediction is done using LASSO, and the web mined data originates from Google trends.[1]

2. METHODOLOGY

To accomplish this assignment I split it into several smaller pieces.

Find texts

For all vaccines I used the texts provided by the assignment text. These text may not have been the best descriptions of the vaccines, but since I have not extra knowledge of vaccines, I decided that I would not be able to find better texts.

A big plus about these text was, that they were made using layman terms. Those would be the same words, that potential buyers would use to search for vaccines, and thus would be what were were looking for.

Tokenize

Before the texts can be of any use, all the stop words has to be removed, and I should attempt to remove all words not describing the vaccine at all.

The methods to do both of these things are provided in the assignment. First I convert all punctuation to white space. Then i split all words based on that white spaces, to get every word for it self.

Now that I only use the two text provided, I can then extract all the words from on of the files, that is included in the other file.

I remove punctuation using the character provided by pythons string library. This may unintentionally split some words that were meant to be one.

Querys for Google

Now that I have each word, I get the data for each word. I then saved the result into one CSV file.

I had the problem, that some word would be downloaded in weeks instead of months which was the desired format. To handle this i made a naive split only looking at the first yeah and the first month, and combining all the weeks starting in the same month. This way the data was converted to months at the cost of some searches ending up in the wrong month. The effect is however spread out through 5 years of data, and would probably be spread out almost evenly.

Making the prediction

I know nothing of machine learning, so I decided to use some kind of linear model, and hope that the result is good enough. I decided on using lasso, because the sklearn library included cross validation for lasso. The function got run on almost default parameters. The only exception is, that I set cv to 5, to get 5 folder cross validation, and I set positive = True because it does not really make sense for coefficients to be negative. I do that because searches for one thing does not really negatively impact searches for other things.

2.1 Naive method

To compare the result, I need another method to compare with. For this the mean of the target data is chosen. It assumes that every month, the vaccines sold is the same always, so it's not

3. FINDINGS

I compare the result of the LASSO model with the naive mean result. My result is shown in table 1. It looks like the lasso function have done a decent job on both vaccines.

Table 1: Results

vaccine	rmse	naive RMSE
HPV-1	10.22	16.28
MMR-2(12)	20.55	29.58

4. CONCLUSIONS

Web Science 2016 DIKU, Denmark

There are a few problems with the target data. All target data hat zeroes in the last three months. This could be due to missing information, and the algorithm to train on missing data, and thus increase the error when there is data.

5. REFERENCES

[1] Google trends. https://www.google.com/trends/. "[Online; accessed 11-apr-2016]".

[2] Ssi.dk. www.ssi.dk. "[Online; accessed 20-feb-2016]".

[3] www.sundhed.dk. www.sundhed.dk. "[Online; accessed 20-feb-2016]".

[4] N. D. Hansen, C. Lioma, and K. Molbak. Predicting vaccination uptake using web search queries. *in press*, 2016.