Predicting Vaccination Uptake using Web Search Queries
WS 2016 Project 1

1. INTRODUCTION

In the previous study of Hansen et al.[3], we observed overall low error when predicting vaccination uptake by means of web-mined data as well as clinical data. Following study presents the same experiment, but using only the web-mined data, keeping the clinical data as a ground truth. We could see in the previous study that the error increases only slightly when using the web-mined data alone instead of using both web-mined and clinical[3]. Further, we explore how significant this error is and how it changes with various data and models. We will work with 2 vaccines out of 13 that Hansen et al. originally worked with, namely Human papilloma virus (HPV) and a vaccine against measles, mumps, and rubella (MMR). The web-mined data are represented by the relative counts of how many times people Googled terms regarding specified vaccines. We focused on the period from January 2011 till December 2015.

2. METHODOLOGY

All computations are written in Python programming language using very helpful libraries such as scikit lear[4] and numpy[5]. The code consists of three modules (python files) where each of them corresponds to one of the following sections.

2.1 Query terms

First, it was necessary to collect terms that are most relevant to the vaccines. We gathered descriptions of the vaccines from the web pages of Statens Serum Institut[1] and the portal for the public Danish Healthcare Services[2]. We treated the descriptions as a bag of words and therefore order of the words did not play any role while creating queries. The process of creating queries from vaccine descriptions included removing punctuation, converting text to lower case, removing stopwords, keeping only unique words and finally filtering those words that appeared in the descriptions of both the institutions. We can emphasize some noteworthy tendencies among the resulting queries. Part of the words represents time expressions, for example, significant year or month for the disease. Other words relates to the affected population, symptoms, diagnosis or procedures of treatment. We decided not to filter the numbers, since people can also search number queries related to the vaccines and diseases.

[1] http://www.ssi.dk/
[2] https://www.sundhed.dk

Web Science 2016 DIKU, Denmark

2.2 Obtaining web data

The Google Trends engine[2] provides free access to the data regarding searching on the Google. One can specify many parameters of searching, such as location, time period or category. The url request for a query is following: http://www.google.com/trends/trendsReport?q=query&date=01/2011%2060m&geo=DK&hl=en-US&cmpt=q&content=1&export=1

where the important url parameters for us are:

q query term of the vaccine

date date to start from (01/2011) and number of months from that date (60)

geo two letter code of the country as a geographic specification (DK)

The process of scraping data from the Google Trends in Python language was simple, since we used PyTrends library[1], which provides suitable and easy to understand interface. The results contain one number for every week. The provided clinical data contains one entry for every month, thus we had to merge weeks into months first. One could think that the numbers in the response correspond to the count of searches of the specified term in the specified time period. However, Google normalizes values in the range of 0 and 100 where 100 is a rank of the month, when people searched the term most. Therefore, the numbers are prepared for the prediction and we do not have to normalize the data ourselves.

2.3 Prediction

We have collected data about searching queries for each of the 60 months as well as real vaccination uptake. Now we can predict uptake for specified month only with knowledge of the searched queries and compare it with the real vaccination uptake. A common aproach to predict continues scalar variable based on one or more independent variables is a linear regression. There are many models for linear regression, but we were working with two of them: Ordinary

Least Squares (OLS) and Least Absolute Shrinkage and Selection Operator (LASSO). All the models were created in the environment of the Python library scikit learn[4].

2.3.1 *Baseline*

Before we started to consider intelligent linear regression models, we had evaluated RMSE for a simple prediction model that always predicts the same uptake, namely the average value of the real uptakes across all the months.

2.3.2 *Ordinary Least Squares*

The OLS model tries to fit data points in the n-dimensional space with a 1-dimensional line. The fit is considered as best when the expression 1 is minimal for some vector w (coefficients of the OLS model).

$$||Xw - y||^2 \tag{1}$$

X is a matrix of input variables (our web-mined data) and y is a vector of output variable (real vaccination uptake) [4]. No parameters are necessary to train the OLS model, thus the prediction is straight forward. We tested the model by means of the cross validation, which means that all samples were divided into 5 folds of uniform length, from which 4 form the training set and one forms the testing set. We repeated this procedure five times, so that each fold appeared in the testing set. The final RMSE value is computed as an average of those five evaluations.

2.3.3 *LASSO*

The more advanced model is called LASSO, which is based on the OLS model, but tends to prefer solution with the fewest feature values and effectively reduces the number of variables upon which the given solution is dependent [4]. The fit is considered as best when the expression 2 is minimal for some coefficients vector w.

$$\frac{1}{2n}||Xw - y||^2 + \alpha||w|| \tag{2}$$

The parameters keep the same meaning as in the expression 1 and user defined parameter α defines weight of the coefficients vector regulation. Unlike the OLS model, LASSO effectively filters non-significant variables and sets them to zero. The OLS model sets those values close to zero, which creates a bias in the prediction.

Before the evaluation of the prediction, we need to find the most suitable parameter α. We ran the optimization for α values between 1 and 700, because the best fit never exceeded this value. The same procedure as for the OLS model (cross validation) was performed for every value α. Then we used α with the lowest RMSE value for the prediction.

3. FINDINGS

We show the RMSE value for both the training set and testing set, since the models try to generalize, and therefore zero RMSE on the training set is not guaranteed. The RMSE values may differ with each prediction, because the folds of the cross validation are picked randomly. We present the best achieved RMSE.

The table 1 shows errors for the baseline model as well as the applied average values. The RMSE values seem fairly high, but we can analyse them properly only after the comparison with the intelligent linear regression models.

Table 1: RMSE of the baseline prediction model.

Vaccine	Average value	RMSE
HPV	40.05	16.28
MMR	95.25	27.27

The table 2 reveals the results of the prediction for the OLS model. We can observe that the RMSE for both the vaccines is high for the testing set and even for the training set of the MMR vaccine. More surprisingly, the error rate is higher than the error rate of our baseline model. Obviously, the OLS model is not suitable for modeling our type of data.

Table 2: RMSE of the prediction using OLS.

Vaccine	RMSE	
	training set	testing set
HPV	3.15	25.94
MMR	13.21	31.96

The LASSO model predictions achieved lower RMSE value, as we can see in the table 3. The error rate for the training set is higher in comparison to the OLS model but it is lower on the testing data. We can assume, that LASSO model more generalizes the dataset and avoids overfitting. Also, the error rates are lower with comparison to the baseline model. However, the overall error is still high for the purposes of a reliable prediction, at least the predictions of MMR are not convincing. The predictions of the HPV vaccine uptake appears better. Low difference between its error of the training and the testing set (3.43) signs very good generalization of the model, which implies reliable predictions.

Table 3: RMSE of the prediction using LASSO.

Vaccine	RMSE		α
	training set	testing set	
HPV	7.04	10.47	20
MMR	14.02	23.30	12

Best RMSE accomplished by Hansen et al.[3] for HPV vaccine is 9.377, which is very close to our computations. On the other hand, the lowest possible error for the MMR vaccine is 14.928, which is far away from our value. The possible explanation could be the fact, that Google Trends reported zero search activity for a lot of the queries regarding MMR vaccines.

4. CONCLUSIONS

We introduced two linear regression models for prediction of the vaccination uptake based on the web-mined data. We colected the terms related to the vaccines from the web pages of the official registries of vaccines in Denmark. The data for prediction were obtained from the public engine Google Trends, which provided us with the relative number of seaches on the Google by month regarding specified vaccines.

Then we trained OLS and LASSO models on the web-mined data together with the clinical data about actual vaccine uptake. By means of the cross validation, we left out one fifth of all the months from the training procedure and applied a prediction for those months. We evaluated the

predictions with the RMSE metrics and reported the results in the tables for both models and a baseline.

The results show that the vaccine uptake is more difficult to predict for MMR than HPV. The OLS model appeared as inappropriate for modeling and predicting vaccine uptake. Its RMSE value was higher than RMSE of the baseline, which marks the OLS model as useless. The second model used, LASSO, was more successful in the predictions. The main reason was a good ability for generalization of the dataset. We compared the results with results of the study of Hansen et al.[3]. Depending on the type of vaccine and used models, the prediction of the vaccination uptake in the future appears possible even when only the web-mined data are taken in consideration.

5. REFERENCES

[1] GeneralMills. Pytrends. https://github.com/GeneralMills/pytrends. Accessed: 2016-02-06.

[2] Google.com. Google trends. https://support.google.com/trends/answer/www.google.com/trends. Accessed: 2016-02-06.

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

[4] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[5] S. C. C. StÃl'fan van der Walt and G. Varoquaux. The numpy array: A structure for efficient numerical computation. *Computing in Science & Engineering*, 13:22–23, 2011.