Homework 3 - Advanced Digital Signal Processing A digital marketing application by Google

Marco Ceran

A.A. 2020-2021

1 Introduction

In marketing applications an important problem is to evaluate whether a specific action or optimization has had an incremental effect over an outcome metric over time or not (this problem is often addressed in conjunction with A/B testing techniques). In this case we are taking as metrics three common KPIs (Key Performance Indicators): cost (how much an advertiser is spending on the Google properties), clicks (how many user clicks an ad receives), conversions (a specific action completed by the users relevant for the advertiser business) and we want to estimate the impact of the adoption of AB (automated bidding tool) on the time series of these metrics.

2 Execution and results

The first task of this activity deals with finding an estimator for the interest track from the 7 control tracks, evaluate the accuracy of such predictor and use it to estimate the impact of the adoption of AB on the KPI time series. The proposed method, that is a synthetic control method, aims at finding the track of interest as a weighted sum of the control tracks:

$$\hat{x}_0(t) = \sum_{k=1}^{7} a_k \cdot x_k(t)$$

The preperiod (the period before the implementation of AB) lasts from 2019-01-01 to 2020-04-15, i.e. it comprises the first 520 elements of the KPI data. The weights will be chosen with the aim of minimizing the RMSE between the true and estimated control track in the preperiod. In order to achieve this goal the method that was chosen can be thought of as a combination of Monte Carlo and Simulated Annealing procedure to find the minimum of a loss function (the RMSE) for which we haven't made convexity assumptions. The initial set of weight is a uniform one (all 7 weights are set to 1/7), then an iterative process tries to (randomly) find a better set of weights by generating gaussian values

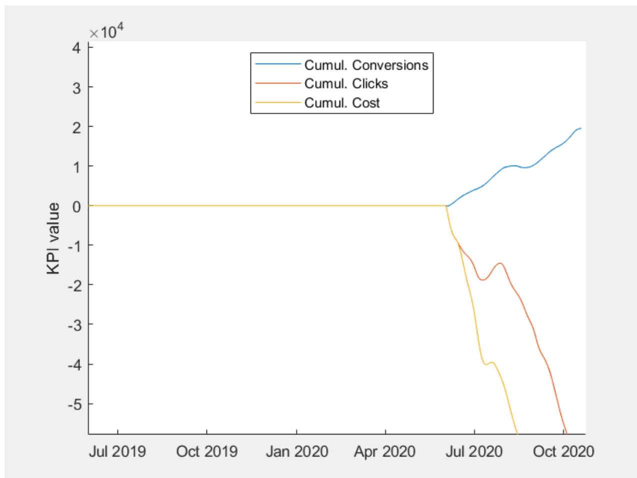
centered around the current weights, if this randomly generated set of weights outperforms (yields a lower RMSE than) the previous one it replaces it. In order to settle on an optimum set of weights and not roaming indefinitely around with the weights' values the standard deviation of the gaussianly generated weights is inversely proportional to the current iteration number (this idea is borrowed from the S.A. procedure).
In MATLAB code this procedure has been implemented as:

```
a = ones(7, 3)/7;
kmax = 8000;
RMSEpre = [0 0 0];
for i = 1:3
    for k = 0:kmax-1
        ak = 0.3*(kmax-k)/kmax*randn(length(a), 1);
        RMSE = mean(((a(:, i)+ak)'*squeeze(smoothstructdata(1:520,
            1:7, i)')-smoothstructdata(1:520, 8, i)').^2)^0.5;
        RMSEa = mean((a(:, i)'*squeeze(smoothstructdata(1:520, 1:7,
            i)')-smoothstructdata(1:520, 8, i)').^2)^0.5;
        if RMSE < RMSEa
            a(:, i)=a(:, i)+ak;
        end
    end
    RMSEpre(i) = min(RMSE, RMSEa);
end
```

We can see in this code that the number of iterations is set to 8000, and that the RMSE of the preperiod is also computed, and the values that have been found converge to [179.4 422.2 544.1]. This method has been preferred to more analytical/deterministic ones because it yields good results while being less computationally intensive than other methods. Other techniques that have been tried and discarded because considered inferior to the one that was chosen are: set of weights directly proportional to the correlation coefficients of the track of interest with the control tracks (very immediate to compute, but the results in term of RMSE are poor), stochastic gradient descent method (low RMSE, but if updated only in the preperiod it fails to be a good predictor in the postperiod due to KPI tracks nonstationarity: in general the set of weights is too dependant on the particular point in which updating is stopped).
Weights values are not bounded to be > 0 or to sum to one (or any other quantity), in order to allow the iterative algorithm to find whatever set of values minimizes the RMSE (negative weights allow the exploitation of inverse correlation between tracks). The set of weights that have been found, for conversions, clicks and cost tracks respectively is:

a_1	a_2	a_3	a_4	a_5	a_6	a_7
0.53	-2.01	0.43	0.50	-0.65	0.61	-0.065
-0.17	-0.143	0.822	0.147	0.996	0.65	0.095
-0.368	-0.095	0.085	0.845	1.884	0.07	0.269

In the coefficients computation the control tracks have been low-pass fitered with a binomial filter with length 28 in order to attenuate high frequency noise. The estimated track elements below zero have been set to 0 for consistency.
The RMSE values that have been found for the postperiod are [175.7 629 1141.1], that are significantly higher than those in the preperiod for clicks and cost, while the RMSE for conversions is very similar in the pre and postperiod.
Below are shown the cumulative difference between true and estimated KPI tracks time series:



As we can see there is in the post intervention period a clear increase in conversions and a decrease in clicks and costs with respect to the track of interest estimated from the other (unaffected by AB) tracks. This can be considered a consequence of the implementation of AB computed using the Causal impact methodology. The rise in conversions coupled with a decline in clicks and costs can be considered an increase in the efficiency of the advertisement campaign.

3 Conclusion

The main challenge in this difference-in-differences activity has been that of finding a good set of weights that estimate the track of interest from the other one. Changing the algorithm that finds the weights can change radically the results, so a lot of care has to be put in this choice.

4 Bibliography

- Spagnolini U., *Statistical Signal Processing in Engineering*, Wiley, 2018
- Brodersen K. H., Gallusser F., Koehler J., Remy N., Scott S. L., *Inferring*

Causal Impact using Bayesian structural time-series models, The Annals of Applied Statistics, 2015

- Luise M., Vitetta G. M., *Teoria dei Segnali*, Milano, McGraw-Hill, 2009