Submitted to the Annals of Applied Statistics
arXiv: arXiv:0000.0000

FUNCTIONAL INSIGHTS INTO GOOGLE ADWORDS

By Dominik Liebl[*,†], Stefan Rameseder[*,‡] and Christoph Rust[*,‡]

University of Bonn[†] and University of Regensburg[‡]

Abstract In this work we analyze data from Google AdWords, today's most important platform for online advertisements. There, the success of an online ad campaign depends on (time-global) seasonal factors as well as on (time-local) events, such as the importance of Valentine's Day for an online flower shop. These components are, however, difficult to assess using existing key figures and methods. In order to reveal both components from the data, we build upon the recent advances in the literature on the functional linear regression model with points of impact. Our proposed model contributes a yearly perspective on the click-through rate—one of the most important measures for evaluating Google AdWords campaigns. For estimating the model parameters, we provide an adjusted estimation algorithm that leads to a significant improvement over the original estimation procedure.

1. Introduction. Online advertising is the most important branch of today's advertising industry, with an expected U.S. revenue of more than 60 billion USD in 2016 (Doty, Sruoginis and Silverman, 2016). In this work, we analyze data from Google AdWords, which is the most popular online advertising platform and of fundamental importance for Alphabet's (Google's parent company) economic success; in 2014, 90 percent of Alphabet's sales came from AdWords.

The main pricing mechanism at Google AdWords is the so-called Pay-Per-Click (PPC) mechanism. Here, advertisers (e.g., an online outdoor shop as in our application) can bid for a sponsored "impression" to be displayed along with Google's search results when a user conducts a search query related to a specific "keyword" (e.g., `outdoor jacket`)[1]. The bidding takes place with respect to "ad campaigns", i.e., samples of (hundreds, thousands, or

[*]The authors thank Crealytics (`www.crealytics.com`) for providing the data, stimulating and inspiring discussions, and posing the statistical problem considered in our application.

Keywords and phrases: functional data analysis, functional linear regression, points of impact, online advertising

[1]Sponsored impressions link to the advertised homepage—they are similar to, but distinguishable from ordinary Google search results.

ten-thousands, etc.) keywords related to the advertised product or online shop.

The limited number of sponsored impressions is allocated by an auction. Advertisers whose impression appears on the display are chosen according to their "ad-rank", which is basically their original bid, i.e., the maximum "costs-per-click" an advertiser is willing to pay times the "quality score", a discrete metric (from 1, the lowest, to 10, the best) determining the relevance of an advertiser's impression. An advertiser only pays if a user "clicks" on the displayed impression. The final cost for this click is the ad-rank of the advertiser with the next highest ad-rank divided by the own quality score—therefore, the auction is often referred to as a generalized second price auction (Edelman, Ostrovsky and Schwarz, 2007). Google AdWords auctions are time continuous, and advertisers can adjust their bids on a time continuous scale. (See Geddes, 2014, for an in-depth introduction to Google AdWords.)

Marketers typically exhibit a strategic bidding behavior and adjust their bids on a daily basis (Edelman and Ostrovsky, 2007). The bidding process is usually (semi-)automatic and based on bidding softwares that evaluate specific key-figures. One of the most important key-figures is the so-called "Click-Though Rate" (CTR), which is defined as the daily number of clicks per impression. The CTR estimates the current probability of receiving a click on a sponsored impression and therefore plays an important role in assisting the bidding process on a short-term basis (Geddes, 2014).

The economic success of ad campaigns, however, also depends on long-sighted bidding strategies taking into account product specific (time-global) seasonalities as well as (time-local) events, such as the importance of Valentine's Day for an online flower shop. Unfortunately, existing key-figures such as the CTR only provide a daily perspective and are not suitable for assisting in the implementation of long-sighted bidding strategies. Therefore, we provide a statistical model that allows us to identify the (global and local) functional relationship between the *yearly* clicks and the *yearly* trajectories of daily impressions—leading to a long-sighted version of the CTR.

As a yearly measure of clicks, we use the logarithmized yearly sums of clicks, i.e., $Y_i = \log(C_i)$ with $C_i := \sum_{t=1}^{365} \texttt{clicks}_{it}$, where i indexes the ith keyword of the considered ad campaign. As a yearly measure of impressions, we use the yearly trajectories of daily logarithmized numbers of impressions, i.e., $X_i(t) = \log(\mathcal{I}_i(t))$ with $\mathcal{I}_i(t) := \texttt{impressions}_{it}$, where $t = 1, \ldots, 365$ indexes the days of the considered year. Our application uses data from a real Google AdWords campaign run in the year from April 1st, 2012 to March 31st, 2013, by an online store selling outdoor equipment. The left plot in Figure 1 shows all trajectories $X_i(t)$ of the considered ad campaign. The

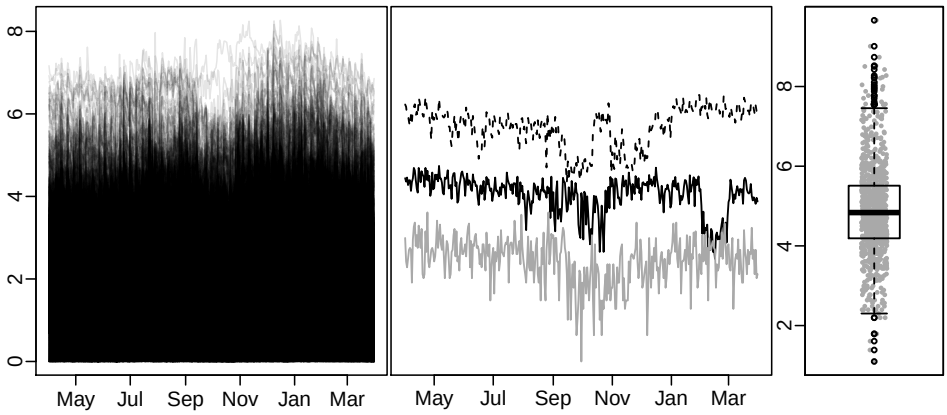

FIGURE 1. *Left: Yearly trajectories of daily logarithmized numbers of impressions. Middle: Three exemplary trajectories $X_i(t)$. Right: Logarithmized sums of yearly clicks, Y_i.*

middle plot shows three exemplary (logarithmized) impression trajectories $X_i(t)$. The right panel shows the (logarithmized) yearly sum of clicks Y_i, received on the impressions of the ith keyword.

We emphasize that considering keywords sampled from a (campaign-specific) keyword-population is essentially not different from considering individuals sampled from a certain population of humans. In both cases we are not interested in inference about single sampling units (keywords or individuals), but aim to make inference, for instance, about the conditional means in the considered population (of keywords or individuals).

Our aim is to identify the time-global and time-local components at which an increase in the number of impressions has a particularly strong (or weak) effect on the yearly number of clicks. However, extracting such valuable information from the scalar-valued dependent variable Y_i and the high-dimensional explanatory trajectories $X_i(t)$ is a non-trivial statistical problem. The slowly varying seasonal component could be estimated using the function-valued slope parameter of the classical functional linear regression model (see, e.g., Hall and Horowitz (2007)). The presence of time-local effects, however, harms such a simple estimation approach (see Figure 1 in Shin and Hsing, 2012, or the right plot in Figure 3 in this paper for notable examples). Therefore, we use the recent work of Kneip, Poss and Sarda (2016), which augments the classical functional linear regression model by incorporating so-called Points of Impact (PoI) that allow to identify and to control for time-local effects.

As demonstrated in our simulation study, the finite sample performance of the estimation procedure proposed by Kneip, Poss and Sarda (2016) is very

sensitive to the performance of the involved model selection. Therefore, we propose an adjusted estimation algorithm that leads to significantly improved and more robust estimation results by using a refined model selection procedure. Furthermore, deviating from Kneip, Poss and Sarda (2016), we use a penalized smoothing splines estimator instead of a Functional Principal Components (FPCA) based series estimator. Both estimators attain the minimax-optimal convergence rate (see Hall and Horowitz, 2007, and Crambes, Kneip and Sarda, 2009), but it turns out that the penalized smoothing splines estimator performs significantly better in finite samples. Although, motivated from our real data problem, we expect our adjusted estimation algorithm to be of a more general interest in applied statistics. Therefore, we provide the R-package `FunRegPoI` containing implementations of the proposed estimation procedure (see supplementary material Liebl, Rameseder and Rust (2017)).

Only a few empirical studies considering data from Google AdWords (or similar platforms) are published in refereed journals, and their contributions are very different to ours. Shan et al. (2016) use a tensor factorization model in order to predict the daily CTR. Robinson, Wysocka and Hand (2007) provide a more general evaluation of online advertising. Tutaj and van Reijmersdal (2012) and Yoo (2014) investigate questions related to the perception of different ad types and brand recognition. Mehta et al. (2007) consider profit maximization from the perspective of the search engine. Varian (2007) and Edelman, Ostrovsky and Schwarz (2007) provide insights from auction theory and mechanism design. To the best of our knowledge, we are the first to use methods from Functional Data Analysis (FDA) to analyze data of an online ad campaign; however, there are several contributions in FDA on related applications. Reddy and Dass (2006) use a classical functional linear regression model to analyze online art auctions, Liu and Müller (2008) analyze eBay auction prices using methods for sparse functional data, Wang, Jank and Shmueli (2008) forecast eBay auction prices, Wang et al. (2008) develop a model for the price dynamics at eBay using differential equation models, and Zhang, Jank and Shmueli (2010) consider real-time forecasting of eBay auctions using functional K-nearest neighbors.

The literature on functional linear regression models with PoIs is a mainly theory driven literature, and we are the first to show the usefulness of these models in a non-trivial real data application. Our estimation algorithm builds upon the work of Kneip, Poss and Sarda (2016). McKeague and Sen (2010) propose a similar regression model, but only allow for a single PoI. Ferraty, Hall and Vieu (2010) allow for multiple PoIs, however, do not provide a simultaneous estimation of the function-valued slope parameter. Matsui

and Konishi (2011) also consider extraction of local information within functional linear regressions using a LASSO-type approach, but do not estimate global components. Torrecilla, Berrendero and Cuevas (2016) focus on a classification context and Fraiman, Gimenez and Svarc (2016) consider feature selection for functional data at a more general level.

Our estimation algorithm uses the penalized smoothing splines estimator for functional linear regression models proposed by Crambes, Kneip and Sarda (2009). The related literature is extensive and the following is only an exemplary list of references. Cardot et al. (2007) consider functional linear regression with errors-in-variables, Crambes, Kneip and Sarda (2009) address optimality issues, Goldsmith et al. (2010) focus on penalized smoothing splines within a mixed model framework, and Maronna and Yohai (2013) propose a robust version of the penalized smoothing splines estimator.

Scalar-on-function regression models are successfully applied to solve important practical problems. Chiou (2012) propose a functional regression model for predicting traffic flows. Goldsmith et al. (2012) introduce a penalized functional regression model to explore the relationship between cerebral white matter tracts in multiple-sclerosis patients. Koeppe et al. (2014) consider regularized functional linear regression for brain image data. Gellar et al. (2014) and Gromenko, Kokoszka and Sojka (2017) propose functional regression models for incomplete curves. An overview article on methods for scalar-on-function regression is found in Reiss et al. (2016). Readers with a general interest in functional data analysis are referred to the textbooks of Ramsay and Silverman (2005), Ferraty and Vieu (2006), Horváth and Kokoszka (2012), and Hsing and Eubank (2015).

The rest of the paper is structured as follows. The next section introduces the statistical model and describes our estimation algorithm. Section 3 presents our simulation results and gives a detailed discussion of the advantage of our estimation algorithm over the original estimation procedure proposed by Kneip, Poss and Sarda (2016). Our application is found in Section 4. The supplementary material Liebl, Rameseder and Rust (2017) contains the R-codes to reproduce our simulation study and the real data application.

2. Methodology. We formally consider the following functional linear regression model with PoIs introduced by Kneip, Poss and Sarda (2016):

$$Y_i = \int_a^b \beta(t) X_i(t) dt + \sum_{s=1}^{S} \beta_s X_i(\tau_s) + \epsilon_i, \quad i = 1, \ldots, n. \tag{1}$$

Here, $(Y_1, X_1), \ldots, (Y_n, X_n)$ denote an i.i.d. sample of scalar response variables $Y_i \in \mathbb{R}$ and random predictor functions $X_i \in L^2([a, b])$, where $\mathbb{E}[Y_i] = 0$

and $\mathbb{E}[X_i(t)] = 0$ for all $t \in [a, b]$. Without loss of generality, we set $[a, b] = [0, 1]$. The i.i.d. error term ϵ_i has mean zero and homoscedastic variance $\mathbb{E}[\epsilon_i^2] = \sigma_\epsilon^2 < \infty$. The assumption that Y_i and X_i have mean zero is only for notational simplicity; for the estimation, however, we will explicitly denote the centering of the data.

The function-valued slope parameter $\beta \in L^2([0, 1])$ in Model (1) describes the time-global influences of X_i on Y_i. The scalar-valued slope parameters $\beta_s \in \mathbb{R}$ take into account the time-local influences where the corresponding (unknown) time-points τ_s denote the locations of the PoIs. Our estimation algorithm described below addresses the estimation of all unknown model parameters, namely, the "global" slope coefficient β, the "local" influences of the PoIs $\beta_1, \ldots, \beta_S$, and the set of PoI locations $\mathcal{T} = \{\tau_1, \ldots, \tau_S\}$.

2.1. *Estimation.* Estimating the model parameters in Model (1) bears the substantial risk of an omitted-variable-bias since not incorporating the (unknown) true PoI locations τ_s typically leads to a heavily biased estimator $\widehat{\beta}(t)$ (see Figure 1 in Shin and Hsing (2012), for a noteworthy example). This is a big issue in practice, and our simulation results show that the original estimation procedure of Kneip, Poss and Sarda (2016) suffers severely from such biases (see the right plot in Figure 3). Therefore, we contribute an adjusted estimation algorithm that decouples the estimation of the slope parameters from the selection of the PoI locations. Our estimation algorithm is built up from the following three basic Pre-select-Estimate-Sub-select (PES) steps:

1. **Pre-select** Pre-select "potential" PoIs $\widetilde{\mathcal{T}} = \{\widetilde{\tau}_1, \ldots, \widetilde{\tau}_{\tilde{S}}\}$. (See Section 2.1.1)
2. **Estimate** Estimate the function- and scalar-valued slope parameters $\beta, \beta_1, \ldots, \beta_{\tilde{S}}$ given the set of "potential" PoIs $\widetilde{\mathcal{T}}$. (See Section 2.1.2)
3. **Sub-select** Sub-select PoIs from the set of "potential" PoIs $\widetilde{\mathcal{T}}$. (See Section 2.1.3)

Typically, the estimation step (Step 2) leads to inefficient, but unbiased estimators $\widehat{\beta}(t)$—inefficient since $\widetilde{\mathcal{T}}$ tends to contain many redundant PoI locations, which reduces the number of degrees of freedom, and unbiased since the large set of "potential" PoIs $\widetilde{\mathcal{T}}$ has a high likelihood of containing the true PoI locations. Our final PES-ES algorithm, described in Section 2.2, uses a repetition of the latter two Estimate-Sub-select (ES) steps, which results in a further improvement of the estimation results by increasing the efficiency of the estimation procedure. A more detailed conceptual description of our estimation algorithm can be found in Section 3.2, where we also explain the

advantages of our estimation algorithm over the original estimation procedure of Kneip, Poss and Sarda (2016).

In the following, we introduce our basic notation. The functions $X_i(t)$ are observed at p equidistant grid points $t_1, \ldots, t_p$ with $t_j = (j-1)/(p-1)$. For non-equidistant designs, this can always be achieved by pre-smoothing the data. In $\mathbf{Y} = (Y_1, \ldots, Y_n)' \in \mathbb{R}^n$, we collect all observations of the response variable Y_i, and in $\mathbf{X} = (X_i(t_j))_{ij} \in \mathbb{R}^{n \times p}$, we collect all discretizations $X_i(t_j), i = 1, \ldots, n, j = 1, \ldots, p$. Let $\mathbf{Y}^c$, $\mathbf{X}^c$, $\mathbf{Y}^{st}$, and $\mathbf{X}^{st}$ define the centered and standardized versions of $\mathbf{Y}$ and $\mathbf{X}$, i.e., $\mathbf{Y}^c = (Y_1^c, \ldots, Y_n^c)'$, $\mathbf{X}^c = (X_i^c(t_j))_{ij}$, $\mathbf{Y}^{st} = (Y_1^{st}, \ldots, Y_n^{st})'$, and $\mathbf{X}^{st} = (X_i^{st}(t_j))_{ij}$, where $Y_i^c = Y_i - \overline{\mathbf{Y}}$, $X_i^c(t_j) = X_i(t_j) - \overline{\mathbf{X}}_j$, $Y_i^{st} = Y_i^c / \mathrm{sd}(\mathbf{Y})$, $X_i^{st}(t_j) = X_i^c(t_j)/\mathrm{sd}(\mathbf{X}_j)$, $\overline{\mathbf{Y}} = n^{-1}\sum_{i=1}^n Y_i$, $\overline{\mathbf{X}}_j = n^{-1}\sum_{i=1}^n X_i(t_j)$, $\mathrm{sd}(\mathbf{Y}) = (n^{-1}\sum_{i=1}^n (Y_i - \overline{\mathbf{Y}})^2)^{1/2}$, and $\mathrm{sd}(\mathbf{X}_j) = (n^{-1}\sum_{i=1}^n (X_i(t_j) - \overline{\mathbf{X}}_j)^2)^{1/2}$.

2.1.1. *Pre-Select PoIs.* To identify potential PoIs $\widetilde{\tau}_s, s = 1, \ldots, \widetilde{S}$, Kneip, Poss and Sarda (2016) propose a local maxima search (over t_j) based on the sample version $|n^{-1}\sum_{i=1}^n Z_{X_i}(t_j;\delta)Y_i|$ of the covariance $\mathbb{E}[Z_{X_i}(t;\delta)Y_i]$, where $Z_{X_i}(t;\delta) = X_i(t) - (X_i(t-\delta) + X_i(t+\delta))/2$ is the central second-order difference quotient of $X_i(t)$ given a parameter $\delta > 0$. This is motivated by the authors' Assumption 1 and Theorem 3, where they connect non-smooth behavior at the diagonal of the covariance function $\mathbb{E}[X_i(s)X_i(t)]$ with their property of *specific local variation*—a sufficient condition for identifying PoIs (cf. Kneip, Poss and Sarda, 2016, Theorem 1). Here, $Z_{X_i}(t;\delta)$ acts as a filter on $X_i(t)$ that uncovers local variation of the process $X_i(t)$ providing insights about PoIs at peaks of the pointwise absolute covariance with Y_i.

The pseudo code for pre-selecting PoIs is given in Algorithm 1 and corresponds to that of Kneip, Poss and Sarda (2016). The iterative algorithm selects in each iteration one PoI determined by the global maximum of the trajectory of $|n^{-1}\sum_{i=1}^n Z_{X_i^{st}}(t_j;\delta)Y_i^{st}|$ over $j \in \mathcal{J}_{s-1,\delta}$. Once a PoI is selected, the algorithm eliminates the grid points within a $\pm\sqrt{\delta}/2$-neighborhood around the selected PoI (see Line 8 of Algorithm 1) and continues with selecting the next potential PoI. The algorithm terminates when $\mathcal{J}_{s,\delta}$ is the empty set. The elimination step is necessary for providing a consistent estimation procedure. The selection of the first PoI is shown in the middle plot in Figure 2. The first elimination step is shown in the right plot of Figure 2, where the second PoI, $\widetilde{\tau}_2$, is determined by the global maximum of the remaining trajectory of $|n^{-1}\sum_{i=1}^n Z_{X_i^{st}}(t_j;\delta)Y_i^{st}|$ over $j \in \mathcal{J}_{1,\delta}$.

In contrast to Kneip, Poss and Sarda (2016), who use the original data $\mathbf{Y}^c$ and $\mathbf{X}^c$ in order to select potential PoIs, we recommend using the standardized data $\mathbf{Y}^{st}$ and $\mathbf{X}^{st}$, i.e., $|n^{-1}\sum_{i=1}^n Z_{X_i^{st}}(t_j;\delta)Y_i^{st}|$ instead of

Algorithm 1 Search Potential Points of Impact Algorithm

procedure SEARCHPOTPOI($\delta \in \mathcal{D} = (0, \delta_{\max}]$, $\mathbf{X} = \mathbf{X}^{st}$, $\mathbf{Y} = \mathbf{Y}^{st}$)
  Given δ, define the index $k_\delta \in \mathbb{N}$ such that $1 \leq k_\delta < (p-1)/2 \Longleftrightarrow \delta \approx k_\delta/(p-1)$.
  Restrict the set of possible grid indices, i.e., define $\mathcal{J}_{0,\delta} = \{k_\delta + 1, \ldots, p - k_\delta\}$.
  For each index $j \in \mathcal{J}_{0,\delta}$, calculate $Z_{X_i}(t_j; \delta) = X_i(t_j) - \frac{1}{2}(X_i(t_j - \delta) + X_i(t_j + \delta))$.
  while $\mathcal{J}_{s,\delta} \neq \emptyset$, iterate over $s = 1, 2, 3, \ldots,$ and **do**
    Determine the index $j_s \in \mathcal{J}_{s-1,\delta}$ of the empirical maximum of $Z_X(t; \delta)Y$, i.e.,

$$j_s = \underset{j \in \mathcal{J}_{s-1,\delta}}{\operatorname{argmax}} \left| \frac{1}{n} \sum_{i=1}^{n} Z_{X_i}(t_j; \delta) Y_i \right|.$$

    Define the s-th potential impact point $\widetilde{\tau}_s = t_{j_s}$ as grid point at index j_s.
    Eliminate all points in an environment of size $\sqrt{\delta}$ around $\widetilde{\tau}_s$, i.e., define

$$\mathcal{J}_{s,\delta} = \{j \in \mathcal{J}_{s-1,\delta} \mid |t_j - \widetilde{\tau}_s| \geq \sqrt{\delta}/2\}.$$

  end while
  return $\widetilde{\mathcal{T}} = \{\widetilde{\tau}_1, \ldots, \widetilde{\tau}_{\widetilde{S}}\}$
end procedure

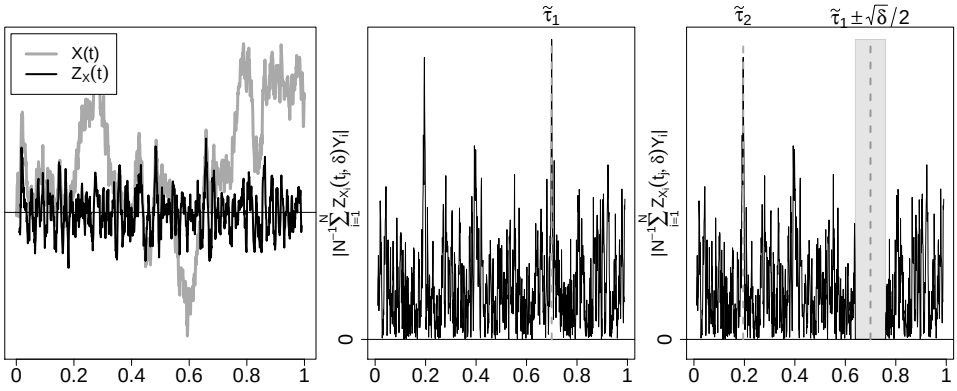

FIGURE 2. *Left: Trajectories of* $X_i^{st}(t_j)$ *and* $Z_{X_i}^{st}(t_j; \delta)$*, with* $\delta = 0.01$*. Middle: Trajectory of* $|n^{-1} \sum_{i=1}^{n} Z_{X_i^{st}}(t_j; \delta) Y_i^{st}|$ *with first choice* $\widetilde{\tau}_1$*. Right: Visualization of the second iteration of the* ***searchPotPoI****-Algorithm.*

$|n^{-1} \sum_{i=1}^{n} Z_{X_i^c}(t_j; \delta) Y_i^c|$. From an asymptotic perspective, this is an irrelevant adjustment; however, it offers a performance boost in finite samples since the unscaled data $\mathbf{Y}^c$ and $\mathbf{X}^c$ have a tendency to lead to artificial local maxima in $|n^{-1} \sum_{i=1}^{n} Z_{X_i^c}(t_j; \delta) Y_i^c|$.

2.1.2. *Estimate Slope Parameters.* To estimate the slope parameters—given the pre-selected PoIs $\widetilde{\mathcal{T}}$—we adapt the penalized smoothing splines estimator by Crambes, Kneip and Sarda (2009) in order to incorporate PoIs. Let us initially recap the situation of Model (1) without PoIs ($\mathcal{T} = \emptyset$), as considered in Crambes, Kneip and Sarda (2009). Their estimator of $\beta(t)$,

evaluated at the grid points $t_1, \ldots, t_p$, is given by

$$\left(\widehat{\beta}^{\rho}(t_1), \ldots, \widehat{\beta}^{\rho}(t_p)\right) = \frac{1}{n}\left(\frac{1}{np}\mathbf{X}^{c\prime}\mathbf{X}^{c} + \rho\mathbf{A}\right)^{-1}\mathbf{X}^{c\prime}\mathbf{Y}^{c}, \tag{2}$$

where the penalty matrix $\mathbf{A} = \mathbf{P} + p\mathbf{A}^{\star}$ is composed of a non-classical projection matrix $\mathbf{P}$ and a classical regularization matrix $\mathbf{A}^{\star}$. The non-classical $p \times p$ projection matrix $\mathbf{P} = \mathbf{W}(\mathbf{W}'\mathbf{W})^{-1}\mathbf{W}'$, with $\mathbf{W} = (t_j^l)_{j,l} \in \mathbb{R}^{p\times m}$, is introduced by Crambes, Kneip and Sarda (2009) in order to guarantee optimality and uniqueness of their estimator. Following the usual convention, we set $m = 2$, which results in the classical choice of *cubic* splines. The classical $p \times p$ regularization matrix $\mathbf{A}^{\star}$ is defined as usual by

$$\mathbf{A}^{\star} = \mathbf{B}(\mathbf{B}'\mathbf{B})^{-1}\left(\int_0^1 \mathbf{b}^{(2)}(t)\mathbf{b}^{(2)}(t)'dt\right)(\mathbf{B}'\mathbf{B})^{-1}\mathbf{B}',$$

where $\mathbf{b}(t) = (b_1(t), \ldots, b_p(t))'$ are natural cubic spline basis functions, $\mathbf{b}^{(2)}(t)$ denotes their second derivatives, and $\mathbf{B}$ is a $p \times p$ matrix with elements $b_i(t_j), i, j = 1, \ldots, p$. For the implementation of the natural cubic spline basis functions we use the `ns`-function contained in the R-package "splines", which is itself part of base R (R Core Team, 2017).

In order to incorporate the pre-selected PoIs, we need to extend the matrices $\mathbf{X}^c$ and $\mathbf{A}$. The extended data matrix is given by $\mathbf{X}^c_{\widetilde{\mathcal{T}}} = (\mathbf{X}^c, p\mathbf{X}^c(\widetilde{\tau}_1), \ldots, p\mathbf{X}^c(\widetilde{\tau}_{\widetilde{S}})) \in \mathbb{R}^{n\times(p+\widetilde{S})}$, where $\mathbf{X}^c(\widetilde{\tau}_s) = (X_1^c(\widetilde{\tau}_s), \ldots, X_n^c(\widetilde{\tau}_s))' \in \mathbb{R}^n$. The extended penalty matrix is given by

$$\mathbf{A}_{\widetilde{\mathcal{T}}} = \begin{pmatrix} \mathbf{A} & 0 \\ 0 & 0 \end{pmatrix} \in \mathbb{R}^{(p+\widetilde{S})\times(p+\widetilde{S})},$$

where all entries with respect to the PoIs are zero (see Goldsmith et al., 2010, for an equivalent extension of the penalty matrix). The augmented estimator of $\beta(t_1), \ldots, \beta(t_p)$ and $\beta_1, \ldots, \beta_S$,

$$\widehat{\beta}^{\rho}_{\widetilde{\mathcal{T}}} = \left(\widehat{\beta}^{\rho}_{\widetilde{\mathcal{T}}}(t_1), \ldots, \widehat{\beta}^{\rho}_{\widetilde{\mathcal{T}}}(t_p), \widehat{\beta}^{\rho}_{\widetilde{\mathcal{T}},1}, \ldots, \widehat{\beta}^{\rho}_{\widetilde{\mathcal{T}},\widetilde{S}}\right) = \frac{1}{n}\left(\frac{1}{np}\mathbf{X}^{c\prime}_{\widetilde{\mathcal{T}}}\mathbf{X}^{c}_{\widetilde{\mathcal{T}}} + \rho\mathbf{A}_{\widetilde{\mathcal{T}}}\right)^{-1}\mathbf{X}^{c\prime}_{\widetilde{\mathcal{T}}}\mathbf{Y}^{c}, \tag{3}$$

depends on the included set of PoIs $\widetilde{\mathcal{T}}$ and on the smoothing parameter ρ. In order to determine an optimal smoothing parameter, we use the following Generalized Cross-Validation (GCV) criterion, as proposed by Crambes, Kneip and Sarda (2009):

$$\mathrm{GCV}(\rho) = \frac{\frac{1}{n}\mathrm{RSS}(\widehat{\beta}^{\rho}_{\widetilde{\mathcal{T}}})}{\left(1 - \frac{1}{n}\mathrm{Tr}(\mathbf{H}^{c}_{\rho,\widetilde{\mathcal{T}}})\right)^2}. \tag{4}$$

Here, the Residual Sum of Squares (RSS) is defined as $\mathrm{RSS}(\widehat{\beta}^{\rho}_{\widetilde{\mathcal{T}}}) = ||\mathbf{Y}^c - \mathbf{H}^c_{\rho,\widetilde{\mathcal{T}}}\mathbf{Y}^c||^2$, where $||.||$ denotes the Euclidean norm, and the smoother matrix $\mathbf{H}^c_{\rho,\widetilde{\mathcal{T}}}$ is defined as $\mathbf{H}^c_{\rho,\widetilde{\mathcal{T}}} = n^{-1}\mathbf{X}^c_{\widetilde{\mathcal{T}}}((np)^{-1}\mathbf{X}^{c\prime}_{\widetilde{\mathcal{T}}}\mathbf{X}^c_{\widetilde{\mathcal{T}}} + \rho\mathbf{A}_{\widetilde{\mathcal{T}}})^{-1}\mathbf{X}^{c\prime}_{\widetilde{\mathcal{T}}}$. Our final estimator for the slope parameters is given by the GCV-optimized version of (3),

(5)
$$\widehat{\beta}_{\widetilde{\mathcal{T}}} = \big(\widehat{\beta}_{\widetilde{\mathcal{T}}}(t), \widehat{\beta}_{\widetilde{\mathcal{T}},1}, \ldots, \widehat{\beta}_{\widetilde{\mathcal{T}},\widetilde{S}}\big) = \big(\widehat{\beta}^{\rho_{\mathrm{GCV}}}_{\widetilde{\mathcal{T}}}(t), \widehat{\beta}^{\rho_{\mathrm{GCV}}}_{\widetilde{\mathcal{T}},1}, \ldots, \widehat{\beta}^{\rho_{\mathrm{GCV}}}_{\widetilde{\mathcal{T}},\widetilde{S}}\big), \quad t \in \{t_1, \ldots, t_p\},$$

where $\rho_{\mathrm{GCV}} = \mathrm{argmin}_{\rho\in(0,\rho_{\max}]}\mathrm{GCV}(\rho)$.

2.1.3. *Sub-Select PoIs.* This part of our estimation algorithm is aimed at selecting the true PoIs from the pre-selected set of potential PoIs $\widetilde{\mathcal{T}} = \widetilde{\mathcal{T}}(\delta)$ given the estimate $\widehat{\beta}_{\widetilde{\mathcal{T}}}$ in (5). This sub-selection is performed by minimizing the following BIC over subsets $\mathcal{R} \subseteq \widetilde{\mathcal{T}}(\delta)$:

$$\widehat{\mathcal{T}} = \underset{\mathcal{R}\subseteq\widetilde{\mathcal{T}}(\delta)}{\mathrm{argmin}}\ \mathrm{BIC}(\mathcal{R}), \quad \text{where}$$

$$\mathrm{BIC}(\mathcal{R}) = n\log\left(\frac{\mathrm{RSS}(\mathcal{R})}{n}\right) + \log(n)\cdot S_{\mathcal{R}}, \quad \text{with} \quad S_{\mathcal{R}} = |\mathcal{R}|. \tag{6}$$

Here, $\mathrm{RSS}(\mathcal{R})$ is made up of the residuals from regressing the "$\widehat{\beta}_{\widetilde{\mathcal{T}}}$-neutralized" and standardized $Y^{st}_{i,\widehat{\beta}_{\widetilde{\mathcal{T}}}} = (Y^c_i - \int_0^1 \widehat{\beta}_{\widetilde{\mathcal{T}}}(t)X^c_i(t)dt)^{st}$ onto $X^{st}_i(\widetilde{\tau}_s), \ldots, X^{st}_i(\widetilde{\tau}_{S_{\mathcal{R}}})$, with $\{\widetilde{\tau}_1, \ldots, \widetilde{\tau}_{S_{\mathcal{R}}}\} = \mathcal{R}$, where $\widehat{\beta}_{\widetilde{\mathcal{T}}}(t)$ is the estimate of $\beta(t)$ as given in (5).

For optimizing $\mathrm{BIC}(\mathcal{R})$ over $\mathcal{R} \subseteq \widetilde{\mathcal{T}}(\delta)$, we use a "directed" search strategy taking into account the information content in $\widetilde{\mathcal{T}} = \{\widetilde{\tau}_1, \ldots, \widetilde{\tau}_{\widetilde{S}}\}$. By construction, the order of the PoI locations $\widetilde{\tau}_1, \ldots, \widetilde{\tau}_{\widetilde{S}}$ reflects a decreasing signal-to-noise ratio and, therefore, a decreasing quality of the estimates. This suggests to minimize $\mathrm{BIC}(\mathcal{R})$ using a directed search strategy where $\mathrm{BIC}(\mathcal{R})$ is evaluated consecutively at the sets $\mathcal{R} = \{\widetilde{\tau}_1\}$, $\mathcal{R} = \{\widetilde{\tau}_1, \widetilde{\tau}_2\}, \ldots, \mathcal{R} = \{\widetilde{\tau}_1, \ldots, \widetilde{\tau}_{\widetilde{S}}\}$.

2.2. *PES-ES Estimation Algorithm.* Our estimation algorithm, PES-ES, consists of the above described Pre-select-Estimate-Sub-select (PES) steps and a uses a repetition of the latter two Estimate-Sub-select (ES) steps:

1. **P**re-Select $\widetilde{\mathcal{T}} = \widetilde{\mathcal{T}}(\delta)$ (Section 2.1.1)
2. **E**stimate $\widehat{\beta}_{\widetilde{\mathcal{T}}}$ (Section 2.1.2)
3. **S**ub-Select $\widehat{\mathcal{T}} \subseteq \widetilde{\mathcal{T}}$ (Section 2.1.3)
4. re**E**stimate $\widehat{\beta}_{\widehat{\mathcal{T}}}$ (Section 2.1.2, with $\widetilde{\mathcal{T}}$ replaced by $\widehat{\mathcal{T}}$)
5. re**S**ub-Select $\widehat{\mathcal{T}}_{\mathrm{re}} \subseteq \widehat{\mathcal{T}}$ (Section 2.1.3, with $\widetilde{\mathcal{T}}$ replaced by $\widehat{\mathcal{T}}$)

Note that the entire PES-ES algorithm depends on the initially preselected set of potential PoIs $\widetilde{\mathcal{T}}(\delta)$ and, therefore, on the choice of δ. Below we write $\widehat{\mathcal{T}}_{\mathrm{re}}(\delta)$ in order to emphasize this entire dependency on δ. We follow Kneip, Poss and Sarda (2016) and determine an optimal δ by minimizing the BIC. For each δ-value on a fine grid in $(0, \delta_{\max}]$, we run the entire PES-ES algorithm and select the optimal δ by,

$$\delta_{\mathrm{BIC}} = \underset{\delta \in (0,\delta_{\max}]}{\operatorname{argmin}} \ \mathrm{BIC}(\delta), \quad \text{with}$$

$$\mathrm{BIC}(\delta) = n \log\left(\frac{\mathrm{RSS}(\widehat{\mathcal{T}}_{\mathrm{re}}(\delta))}{n}\right) + \log(n) \cdot \mathrm{edf}(\widehat{\mathcal{T}}_{\mathrm{re}}(\delta)), \tag{7}$$

where $\mathrm{RSS}(\widehat{\mathcal{T}}_{\mathrm{re}}(\delta)) = ||\mathbf{Y}^c - \mathbf{H}^c_{\rho_{\mathrm{GCV}},\widehat{\mathcal{T}}_{\mathrm{re}}(\delta)} \mathbf{Y}^c||^2$ with smoother matrix $\mathbf{H}^c_{\rho_{\mathrm{GCV}},\widehat{\mathcal{T}}_{\mathrm{re}}(\delta)}$ defined as $\mathbf{H}^c_{\rho_{\mathrm{GCV}},\widehat{\mathcal{T}}_{\mathrm{re}}(\delta)} = n^{-1}\mathbf{X}^c_{\widehat{\mathcal{T}}_{\mathrm{re}}(\delta)}((np)^{-1}\mathbf{X}^{c\prime}_{\widehat{\mathcal{T}}_{\mathrm{re}}(\delta)}\mathbf{X}^c_{\widehat{\mathcal{T}}_{\mathrm{re}}(\delta)} + \rho_{\mathrm{GCV}}\mathbf{A})^{-1}\mathbf{X}^{c\prime}_{\widehat{\mathcal{T}}_{\mathrm{re}}(\delta)}$ and effective degrees of freedom $\mathrm{edf}(\widehat{\mathcal{T}}_{\mathrm{re}}(\delta)) = \mathrm{Tr}(\mathbf{H}^{c\prime}_{\rho_{\mathrm{GCV}},\widehat{\mathcal{T}}_{\mathrm{re}}(\delta)}\mathbf{H}^c_{\rho_{\mathrm{GCV}},\widehat{\mathcal{T}}_{\mathrm{re}}(\delta)})$. (See Hastie and Tibshirani, 1990, Ch. 3.5, for an overview of possible definitions of edf.)

3. Simulation and Discussion. In the following simulation study, we assess the finite sample properties of our PES-ES algorithm. We investigate estimation errors as well as the robustness against model misspecifications. The estimation procedure suggested by Kneip, Poss and Sarda (2016), abbreviated as KPS, serves as our main benchmark. The conceptual idea of our PES-ES algorithm is described in Section 3.2, where we also give a more detailed discussion of the advantage of our PES-ES estimation procedure over the original estimation procedure proposed by Kneip, Poss and Sarda (2016).

3.1. *Simulation Study.* Kneip, Poss and Sarda (2016) propose an FPCA based procedure using the "augmented" model $Y_i \approx \int_0^1 \beta_K(t) X_{i,K}(t)dt + \sum_{s=1}^{\widehat{S}} \beta_s X_i(\widehat{\tau}_s) + \epsilon_i$, where $\beta_K(t) \approx \beta(t)$ and $X_{i,K}(t) \approx X_i(t)$ are K-dimensional approximations based on the first K eigenfunctions of the empirical covariance operator of X_i (cf. Kneip, Poss and Sarda, 2016, Equation (6.1)). The parameter $K \in \{1, \ldots, K_{\max}\}$ acts as a smoothing parameter and corresponds to our smoothing parameter $\rho \in (0, \rho_{\max}]$. Besides the smoothing parameter K, one has to choose a subset $\widehat{\mathcal{T}} \subseteq \widetilde{\mathcal{T}}(\delta)$ and the δ parameter that determines the set of potential PoIs $\widetilde{\mathcal{T}}(\delta)$, where $\widetilde{\mathcal{T}}(\delta)$ is defined as in

Section 2.1.1. The model-selection parameters K, δ, and $\widehat{\mathcal{T}}$ of KPS are essentially equivalent to the model-selection parameters ρ, δ, and $\widehat{\mathcal{T}}_{\text{re}}$ of our PES-ES algorithm.

For selecting K, δ, and $\widehat{\mathcal{T}}$, Kneip, Poss and Sarda (2016) propose an infeasible and a feasible strategy. The infeasible strategy is used in their simulation study, where the authors perform a $\text{BIC}(K, \widehat{\mathcal{T}})$ based selection of K and $\widehat{\mathcal{T}}$, and set $\delta = 1/\sqrt{n}$. Setting $\delta = 1/\sqrt{n}$ is appropriate in their simulation study, but can be arbitrarily bad in practice. The feasible strategy is used in their real data application, where the authors additionally optimize the $\text{BIC}(K, \widehat{\mathcal{T}}, \delta)$ over $\delta \in (0, \delta_{\max}]$. In the following, we only consider their practically relevant feasible strategy. For selecting $\widehat{T}$, we use the "directed" search approach described in Section 2.1.3, which significantly improves the performance of KPS in comparison to the best-subset selection originally proposed by Kneip, Poss and Sarda (2016).

Kneip, Poss and Sarda (2016) arbitrarily set $K_{\max} = 6$, which is, however, too small for our simulation study where $K_{\max} = 6$ often becomes a binding upper optimization threshold. The choice of $K_{\max}$ is crucial since it constrains the magnitude of possible omitted-variable-biases in $\widehat{\beta}_K(t)$. The same issue, but reciprocally, applies to $\rho_{\min}$ when optimizing the GCV in (4) over $\rho \in [\rho_{\min}, \rho_{\max}]$ with $\rho_{\min} \approx 0$. Therefore, we choose very conservative optimization intervals $[K_{\min}, K_{\max}] = [1, 150]$ and $[\rho_{\min}, \rho_{\max}] = [10^{-6}, 200]$.

We consider four different Data Generating Processes (DGPs), as described in Table 1. The DGPs "Easy" and "Complicated" represent a simple and a more complex version of Model (1). The "Complicated" DGP is challenging due to the closeness of the PoI locations τ_1 and τ_2, which may trigger omitted-variable-biases in $\widehat{\beta}(t)$ when omitting either τ_1 or τ_2. The two further DGPs "NoPoI" ($\mathcal{T} = \emptyset$) and "OnlyPoI" ($\beta(t) \equiv 0$) are used to check the robustness of our PES-ES algorithm against model-misspecifications.

TABLE 1
Data Generating Processes.

DGP	$\beta(t)$	S	$\mathcal{T} = \{\tau_s\}_{s=1,\ldots,S}$	$\{(\beta_s)_{s=1,\ldots,S}\}$
Easy	$\beta(t) = -(t-1)^2 + 2$	2	$\{0.3, 0.6\}$	$\{-3, 3\}$
Complicated	$\beta(t) = -5(t-0.5)^3 - t + 1$	3	$\{0.3, 0.4, 0.6\}$	$\{-3, 3, 3\}$
NoPoI	$\beta(t) = -(t-1)^2 + 2$	0	$\emptyset$	$\emptyset$
OnlyPoI	$\beta(t) \equiv 0$	2	$\{0.3, 0.6\}$	$\{-3, 3\}$

As a benchmark for the "NoPoI" DGP, we use the (minimax-optimal) penalized smoothing splines estimator by Crambes, Kneip and Sarda (2009), abbreviated as CKS, as given in (2).

For each DGP, we generate 1000 replications of $n = 500$ functions $X_i(t)$

observed at $p = 300$ equidistant points $t_1, \ldots, t_p$ in $[0, 1]$. The functions $X_i(t)$ are standard Brownian Motions, and the dependent variables Y_i are generated according to Model (1) with $\epsilon_i \sim N(0, 0.125^2)$. We only report simulation results for $p = 300$ and $n = 500$ since these sample sizes are about of the order of magnitude as in our real data application. We also considered further sample sizes, but this aspect of the simulation study only replicates the empirical consistency results in Kneip, Poss and Sarda (2016) and are, therefore, omitted. Our simulation is implemented using the statistical language R (R Core Team, 2017), and the R-codes for reproducing the simulation results are part of our supplementary material Liebl, Rameseder and Rust (2017).

TABLE 2
Estimation errors.

	Easy		**Complicated**		**NoPoI**		**OnlyPoI**	
L^2 Distances $\int_0^1(\widehat{\beta}(t)-\beta(t))^2dt$								
	Mean	Median	Mean	Median	Mean	Median	Mean	Median
PES-ES	**0.073**	**0.009**	**0.594**	**0.071**	0.009	0.005	**0.005**	**0.000**
KPS	14.98	0.025	142.1	0.074	0.027	0.017	4.060	**0.000**
CKS	(46.53)	(46.56)	(54.09)	(53.98)	**0.004**	**0.003**	(46.47)	(46.44)
% of correctly found $\tau_1, \ldots, \tau_S$								
PES-ES	**99.6**		**93.9**		**100**		99.8	
KPS	99.1		84.2		**100**		**99.9**	
MSE of $\widehat{\beta}_s$ given correctly found τ_s								
	$\widehat{\beta}_1$	$\widehat{\beta}_2$	$\widehat{\beta}_1$	$\widehat{\beta}_2$	$\widehat{\beta}_3$	-	-	$\widehat{\beta}_1$	$\widehat{\beta}_2$
PES-ES	**0.007**	**0.002**	**0.026**	**0.012**	**0.006**	-	-	**0.002**	**0.000**
KPS	0.185	0.004	0.816	0.219	0.034	-	-	0.044	0.024

Table 2 contains our simulation results. For the estimators $\widehat{\beta}(t)$ of $\beta(t)$, we report the means and medians of the L^2-deviations $\int_0^1(\widehat{\beta}(t)-\beta(t))^2dt$. For the estimators $\widehat{\tau}_s$ of τ_s, we report the percentages (of 1000 replications) for which *all* PoI locations $\tau_1, \ldots, \tau_S$ are "correctly found", where a single τ_s is considered as "correctly found" if $|\widehat{\tau}_s - \tau_s| < 0.01$. This corresponds to requiring an estimation precision of only ± 3 grid points, which is substantially more challenging than the matching definition originally used by Kneip, Poss and Sarda (2016). Finally, for the estimators $\widehat{\beta}_s$ of β_s, we report the Mean Squared Errors (MSE), conditionally on the event that τ_s was correctly found. (It is impossible to compute reasonable estimation errors for $\widehat{\beta}_s$ with non-found τ_s.)

The simulation results in Table 2 clearly favor our PES-ES algorithm. In

terms of the L^2 distances $\int_0^1(\widehat{\beta}(t)-\beta(t))^2dt$, our algorithm shows significant improvements in comparison to KPS. In the case of the misspecified model "NoPoI", our PES-ES algorithm performs almost as well as the corresponding (minimax-optimal) benchmark-estimator CKS. For the misspecified model "OnlyPoI", our PES-ES algorithm estimates the zero-slope-function $\beta(t) \equiv 0$ with an essentially negligible estimation error. Considering the correctly found PoI locations $\tau_1, \ldots, \tau_S$, PES-ES finds at least 93.9 percent (84.2% KPS) of the true PoIs. Finally, the estimators $\widehat{\beta}_s$ of PES-ES have 50 to 96 percent smaller MSEs than those of KPS.

3.2. *Discussion.* The significant, partially drastic, improvements of our PES-ES estimation algorithm in comparison to KPS are due to a conceptual problem in the model-selection process of KPS since optimizing the $\mathrm{BIC}(K, \widehat{\mathcal{T}}, \delta)$ *simultaneously* over K and $\widehat{\mathcal{T}}$ leads to instable and ambiguous results. The reason for this can be clarified by considering the following two extreme situations. On the one hand, one can choose a very large smoothing parameter $K \gg 0$ and set $\widehat{T} = \emptyset$ since for very large K, $\widehat{\beta}_K(t)$ is flexible enough such that,

$$\int_0^1 \widehat{\beta}_K(t)X_{i,K}(t)dt \approx \int_0^1 \beta(t)X_i(t)dt + \sum_{s=1}^{S} \beta_s X_i(\tau_s).$$

In this case, $\widehat{\beta}_K(t)$ approximates $\beta(t)$, except for some very spiky components at $t = \tau_s$ such that $\int_{\tau_s-h}^{\tau_s+h} \widehat{\beta}_K(t)X_{i,K}(t)dt \approx \beta_s X_i(\tau_s)$ with, e.g., $h = 0.01$ (see the right plot in Figure 3 for examples of such estimates $\widehat{\beta}_K(t)$). On the other hand, one can set $K = 0$ and use a very large set of PoI locations $\widehat{\mathcal{T}} \approx \{\widehat{\tau}_1, \ldots, \widehat{\tau}_{\widehat{S}}\}$ with $\widehat{S} \gg 0$, such that

$$\sum_{s=1}^{\widehat{S}} \widehat{\beta}_s X_i(\widehat{\tau}_s) \approx \int_0^1 \beta(t)X_i(t)dt + \sum_{s=1}^{S} \beta_s X_i(\tau_s).$$

In this case, $\sum_{s=1}^{\widehat{S}} \widehat{\beta}_s X_i(\widehat{\tau}_s)$ acts like a Riemann sum for approximating $\int_0^1 \beta(t)X_i(t)dt$, except for the $\widehat{\beta}_s$-values at $\widehat{\tau}_s \approx \tau_s$, which approximate $\widehat{\beta}_s X_i(\widehat{\tau}_s) \approx \beta_s X_i(\tau_s)$.

That is, there is a certain ambiguity between the model-selection parameters K and $\widehat{\mathcal{T}}$ that allows the shifting of model-complexities between the integral-part and the PoI-part of the empirical model. This ambiguity leads to very unstable model selections when optimizing $\mathrm{BIC}(K, \widehat{\mathcal{T}}, \delta)$ simultaneously over K and $\widehat{\mathcal{T}}$, and the consequence are bad estimates of $\beta(t)$ due to severe omitted-variable-biases in $\widehat{\beta}(t)$, as shown in the right plot in Figure 3.

The conceptual idea of the PES-part in our PES-ES algorithm is to avoid this problem by decoupling the selection of ρ (equivalent to K) from the selection of $\widehat{\mathcal{T}}$. The estimator $\widehat{\beta}(t)$ in Step 2 (Estimation), which involves the optimization over ρ, is computed for a *given* set of potential PoIs $\widetilde{\mathcal{T}}$ selected in Step 1 (Pre-select). Typically, this estimation step leads to inefficient, but unbiased estimators $\widehat{\beta}(t)$—inefficient since $\widetilde{\mathcal{T}}$ tends to contain many redundant PoI locations, which reduces the number of degrees of freedom, and unbiased since the large set of "potential" PoIs $\widetilde{\mathcal{T}}$ has a high likelihood of containing the true (or at least almost true) PoI locations, which minimizes the risk (or at least the extent) of omitted-variable-biases in $\widehat{\beta}(t)$. The sub-selection of $\widehat{\mathcal{T}} \subseteq \widetilde{\mathcal{T}}$ in Step 3 (Sub-select) then takes place for a *given* estimate $\widehat{\beta}(t)$. The positive effect of decoupling the selection of ρ from that of $\widehat{\mathcal{T}}$ can also be seen in Figure 3, where we compare the 10 percent worst estimates $\widehat{\beta}(t)$ from PES-ES with those of KPS. While $\widehat{\beta}_{\mathrm{KPS}}(t)$ suffers from severe omitted-variable-biases, $\widehat{\beta}_{\mathrm{PESES}}(t)$ does not show any obvious biases.

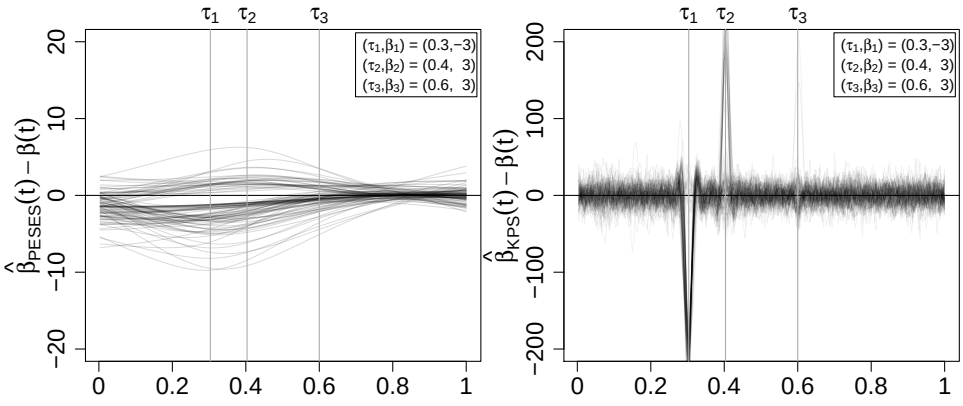

FIGURE 3. *Pointwise deviations* $\widehat{\beta}(t) - \beta(t)$ *of the 10% largest* L^2 *distances* $\int_0^1 (\widehat{\beta}(t) - \beta(t))^2 dt$ *for the "Complicated" DGP. Note that the scales of the two y-axes differ by a factor of 10.*

The repetition of the ES-part in our PES-ES algorithm is of secondary importance, but further improves the estimation results by reducing the variance in the estimator $\widehat{\beta}_{\mathrm{PESES}}(t)$. The use of a penalized smoothing splines estimator instead of an FPCA based estimator is also of secondary importance since both are minimax-optimal estimators of $\beta(t)$. Optimizing a continuous smoothing parameter ρ, instead of a discrete smoothing parameter K, however, improves and stabilizes the estimates.

4. Application. Despite its unquestionable economic relevance, research in applied statistics has not yet considered the analysis of data from Google AdWords. State of the art procedures consist of a mere compilation of simple quantitative key figures (see Geddes, 2014). The below described case study is a first approach towards a statistically innovative analysis and is motivated by the needs of the company that generously provided the data. Today this company uses the described method—with some further confidential enhancements—to support their daily business.

The tracking of the CTR ("clicks-per-impression") is of fundamental importance for evaluating the success of advertising strategies (see Geddes, 2014, for numerous examples). However, as already mentioned in the introduction, the daily perspective of the classical CTR does not provide any information that assists the formation of long-sighted bidding strategies. Therefore, we contribute a functional linear regression model that identifies the (global and local) functional relationship between the yearly clicks and the trajectories of daily impressions.

The data are provided by *Crealytics* (www.crealytics.com), an online advertising service provider with offices in Berlin (Germany), London (UK), and New York City (USA). The considered ad campaign belongs to an online store selling outdoor equipment. (For reasons of confidentiality, we cannot publish the company's name.) A lot of keywords received no impression during the considered time span of 365 days from April 1st, 2012, to March 31st, 2013. Therefore, we consider only the well established and relevant keywords that have been used on at least 320 days within the considered time span—leading to $n = 903$ trajectories observed at $p = 365$ grid points. The very few missing values in the logarithmized impression trajectories are imputed by zeros since a missing value means that the corresponding keyword did not receive an impression. This pre-processing of the data can be reproduced using the R-codes provided in our supplementary material Liebl, Rameseder and Rust (2017).

The considered functional linear regression model with PoIs in (1) is identifiable if the covariance function of the function-valued explanatory variable X_i is sufficiently non-smooth at the diagonal (see Theorem 3 in Kneip, Poss and Sarda (2016)). Kneip, Poss and Sarda (2016) propose the following consistent estimator $\widehat{\kappa}$ for their κ controlling the smoothness at the diagonal of the covariance function:

$$\widehat{\kappa} = \log_2 \left(\frac{(1/(p-2k_\delta)) \sum_{j \in \mathcal{J}_{0,\delta}} \sum_{i=1}^{n} Z_{\delta,X_i}(t_j)^2}{(1/(p-2k_\delta)) \sum_{j \in \mathcal{J}_{0,\delta}} \sum_{i=1}^{n} Z_{\delta/2,X_i}(t_j)^2} \right).$$

An estimate of $\widehat{\kappa} \ll 2$ indicates identifiability which is clearly fulfilled in our

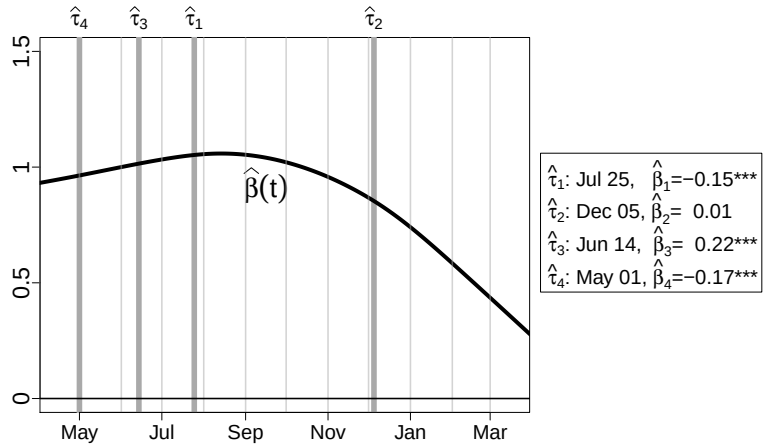

FIGURE 4. *Summary of the estimates* $\widehat{\beta}(t)$ *and* $(\widehat{\tau}_i, \widehat{\beta}_i), i = 1, 2, 3, 4$. *Three asterisks (***) indicate a 1 percent significance level.*

case where $\widehat{\kappa} = 0.03$.

The estimation result from applying our PES-ES estimation algorithm is summarized in Figure 4. The function-valued slope parameter $\widehat{\beta}(t)$ shows a "peak" in the late summer and a pronounced negative trend towards the end of the considered period. The positive peak is in line with our expectations since the demand for outdoor equipment is generally greater during the summer months than during the winter months. The negative trend towards the end of the considered period is due to the strongly increased competition for outdoor equipment ads in Google AdWords during the considered period. (We have not assumed stationary across years, but consider cross sectional data for only one year.)

Additionally, the estimation procedure favors four PoIs (in order of the magnitude of $|\widehat{\beta}_s|$): June 14th ($\widehat{\tau}_3$; $\widehat{\beta}_3 = 0.22$), May 1st ($\widehat{\tau}_4$; $\widehat{\beta}_4 = -0.17$), July 25th ($\widehat{\tau}_1$; $\widehat{\beta}_1 = -0.15$), and December 5th ($\widehat{\tau}_2$; $\widehat{\beta}_2 = 0.01$), where the slope estimate $\widehat{\beta}_2$ of the latter PoI is insignificant at any common significance level. (Critical values are obtained by using a bootstrap approximation based on 5000 bootstrap replications.)

The PoI $\widehat{\tau}_3$ on June 14th, with coefficient $\widehat{\beta}_3 = 0.22$, summarizes two positive effects. On the one hand, the store started a contest on May 23d, 2012, giving away outdoor gear. This contest ended on June 13th, i.e., one day before the PoI which resulted in an increased click-through ratio of contest participants looking for the winners. On the other hand, the closest competitor started his spring sale which led to a spillover bringing many

interested buyers onto the homepage to compare prices.

The two other significant PoIs are explained by effects specific to the German calendar (about 80 percent of the customers live in Germany). The PoI $\widehat{\tau}_4$ on May 1st, with coefficient $\widehat{\beta}_4 = -0.17$, marks the German national Labor Day (commemorating the Haymarket Riot in Chicago in 1886) which is typically an opportunity for family excursions. Similar in interpretation, the PoI $\widehat{\tau}_1$ on July 25th, with coefficient $\widehat{\beta}_1 = -0.15$, marks the beginning of the official summer holidays in Baden-Württemberg and Lower Saxony—two large German states. Both PoIs show a negative sign, which is due to a higher volume in search queries related to outdoor activities, however, the users do not click on the sponsored impressions since they do not intend to buy something—they only search for (free) information on hiking trails etc., which results in a lower CTR.

The log-transformations in $Y_i = \log(C_i)$ and $X_i(t) = \log(\mathcal{I}_i(t))$ allow us to interpret the estimated slope coefficients as elasticities. Taking derivatives with respect to $\mathcal{I}_i(t)$ at a single time point t leads to the following *time-local* elasticity:

$$\frac{\%\Delta C_i}{\%\Delta \mathcal{I}_i(t)} \approx \begin{cases} \widehat{\beta}_s & \text{if} \quad t = \widehat{\tau}_s \\ 0 & \text{else.} \end{cases}$$

That is, time-local changes in $\mathcal{I}_i(t)$ generally have no (i.e., practically negligible) effects on the yearly clicks C_i, except at PoIs, i.e., if $t = \widehat{\tau}_1, \ldots, \widehat{\tau}_{\widehat{S}}$. For instance, a 1% increase in the impressions at the time point of the after-contest PoI ($t = \widehat{\tau}_3$) causes (on average) a 0.22% ($\widehat{\beta}_3 = 0.22$) increase in the yearly clicks.

The function-valued slope parameter $\widehat{\beta}(t)$ does not contribute to the time-local elasticities; however, it determines the elasticities with respect to time-global changes in the impressions, for instance, over the course of a month. The following Riemann sum allows for a simple, approximative approach to interpret such *time-global* elasticities:

$$\widehat{\log(C_i)} \approx \frac{1}{365} \sum_{t=1}^{365} \widehat{\beta}(t) \log\big(\mathcal{I}_i(t)\big) + \sum_{s=1}^{\widehat{S}} \widehat{\beta}_s \log\big(\mathcal{I}_i(\widehat{\tau}_s)\big).$$

For instance, the total elasticity of C_i with respect to $\mathcal{I}_i(t)$ for *all* $t \in$ August is given by

$$\sum_{t \in \text{August}} \frac{\%\Delta C_i}{\%\Delta \mathcal{I}_i(t)} \approx \frac{1}{365} \sum_{t \in \text{August}} \widehat{\beta}(t) + \sum_{s=1}^{\widehat{S}} \widehat{\beta}_s \mathbf{1}_{(\widehat{\tau}_s \in \text{August})},$$

where $\mathbf{1}_{(\texttt{TRUE})} = 1$ and $\mathbf{1}_{(\texttt{FALSE})} = 0$. That is, a 1% increase in the impressions $\mathcal{I}_i(t)$, simultaneously for all $t \in$ August, causes a 0.1% increase in the

yearly clicks since $365^{-1}\sum_{t\in\text{August}}\widehat{\beta}(t)+\sum_{s=1}^{\widehat{S}}\widehat{\beta}_s\mathbf{1}_{(\widehat{\tau}_s\in\text{August})}\approx 0.1$. Hence, the time-global August-elasticity is half of the order of magnitude as the elasticity of the after-contest PoI. This is absolutely plausible since the superimposed influence of the contest and the spillover definitley outpferoms a high-season month such as August in terms of clicks-per-impressions.

Acknowledgements. The authors thank Prof. Alois Kneip (University of Bonn), Dominik Poss (University of Bonn), and Prof. Rolf Tschernig (University of Regensburg) for their valuable suggestions which helped to improve this research work.

SUPPLEMENTARY MATERIAL

Supplement: R-Codes
(doi: COMPLETED BY THE TYPESETTER; zip file). R-codes to replicate the simulation study and the real data application. The necessary data and implementations of the estimation procedures are contained in the R-package `FunRegPoI` which can be accessed from Dominik Liebl's GitHub account.

References.

Cardot, H., Crambes, C., Kneip, A. and Sarda, P. (2007). Smoothing splines estimators in functional linear regression with errors-in-variables. *Computational Statistics & Data Analysis* **51** 4832–4848.

Chiou, J.-M. (2012). Dynamical functional prediction and classification, with application to traffic flow prediction. *The Annals of Applied Statistics* **6** 1588–1614.

Crambes, C., Kneip, A. and Sarda, P. (2009). Smoothing splines estimators for functional linear regression. *The Annals of Statistics* **37** 35–72.

Doty, D., Sruoginis, K. and Silverman, D. (2016). IAB/PwC Internet Advertising Revenue Report. www.iab.com/adrevenuereport.

Edelman, B. and Ostrovsky, M. (2007). Strategic bidder behavior in sponsored search auctions. *Decision Support Systems* **43** 192–198.

Edelman, B., Ostrovsky, M. and Schwarz, M. (2007). Internet Advertising and the Generalized Second-Price Auction: Selling Billions of Dollars Worth of Keywords. *The American Economic Review* **97** 242–259.

Ferraty, F., Hall, P. and Vieu, P. (2010). Most-predictive design points for functional data predictors. *Biometrika* **97** 807–824.

Ferraty, F. and Vieu, P. (2006). *Nonparametric Functional Data Analysis - Theory and Practice*. Spinger.

Fraiman, R., Gimenez, Y. and Svarc, M. (2016). Feature selection for functional data. *Journal of Multivariate Analysis* **146** 191–208.

Geddes, B. (2014). *Advanced Google AdWords*. John Wiley & Sons.

Gellar, J. E., Colantuoni, E., Needham, D. M. and Crainiceanu, C. M. (2014). Variable-domain functional regression for modeling ICU data. *Journal of the American Statistical Association* **109** 1425-1439.

Goldsmith, J., Bobb, J., Crainiceanu, C. M., Caffo, B. and Reich, D. (2010). Penalized functional regression. *Journal of Computational and Graphical Statistics* **20** 830–851.

Goldsmith, J., Crainiceanu, C. M., Caffo, B. and Reich, D. (2012). Longitudinal penalized functional regression for cognitive outcomes on neuronal tract measurements. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* **61** 453–469.

Gromenko, O., Kokoszka, P. and Sojka, J. (2017). Evaluation of the cooling trend in the ionosphere using functional regression with incomplete curves. *The Annals of Applied Statistics* **11** 898–918.

Hall, P. and Horowitz, J. L. (2007). Methodology and convergence rates for functional linear regression. *The Annals of Statistics* **35** 70–91.

Hastie, T. J. and Tibshirani, R. J. (1990). *Generalized Additive Models* **43**. CRC Press.

Horváth, L. and Kokoszka, P. (2012). *Inference for Functional Data with Applications*. Springer.

Hsing, T. and Eubank, R. (2015). *Theoretical Foundations of Functional Data Analysis, with an Introduction to Linear Operators*. John Wiley & Sons.

Kneip, A., Poss, D. and Sarda, P. (2016). Functional Linear Regression with Points of Impact. *The Annals of Statistics* **44** 1-30.

Koeppe, R., Zhu, J., Nan, B. and Wang, X. (2014). Regularized 3D functional regression for brain image data via Haar wavelets. *The Annals of Applied Statistics* **8** 1045–1064.

Liebl, D., Rameseder, S. and Rust, C. (2017). Supplement to "Functional Insights into Google AdWords".

Liu, B. and Müller, H.-G. (2008). *Functional data analysis for sparse auction data* 269–290. Statistical Methods in eCommerce Research (eds W. Jank and G. Shmueli) John Wiley & Sons.

Maronna, R. A. and Yohai, V. J. (2013). Robust functional linear regression based on splines. *Computational Statistics & Data Analysis* **65** 46–55.

Matsui, H. and Konishi, S. (2011). Variable selection for functional regression models via the L1 regularization. *Computational Statistics & Data Analysis* **55** 3304–3310.

McKeague, I. W. and Sen, B. (2010). Fractals With Point Impact in Functional Linear Regression. *The Annals of Statistics* **38** 2559–2586.

Mehta, A., Saberi, A., Vazirani, U. and Vazirani, V. (2007). AdWords and generalized online matching. *Journal of the ACM* **54** 1–19.

Ramsay, J. O. and Silverman, B. W. (2005). *Functional Data Analysis*, 2. ed. Springer.

Reddy, S. K. and Dass, M. (2006). Modeling on-line art auction dynamics using functional data analysis. *Statistical Science* **21** 179–193.

Reiss, P. T., Goldsmith, J., Shang, H. L. and Ogden, R. T. (2016). Methods for scalar-on-function regression. *International Statistical Review* **0** 1–22.

Robinson, H., Wysocka, A. and Hand, C. (2007). Internet advertising effectiveness. *International Journal of Advertising* **26** 527–541.

Shan, L., Lin, L., Sun, C. and Wang, X. (2016). Predicting ad click-through rates via feature-based fully coupled interaction tensor factorization. *Electronic Commerce Research and Applications* **16** 30–42.

Shin, H. and Hsing, T. (2012). Linear prediction in functional data analysis. *Stochastic Processes and their Applications* **122** 3680–3700.

R Core Team (2017). R: A Language and Environment for Statistical Computing R Foundation for Statistical Computing, Vienna, Austria.

Torrecilla, J. L., Berrendero, J. R. and Cuevas, A. (2016). Variable selection in functional data classification: a maxima-hunting proposal. *Statistica Sinica* **26** 619–638.

Tutaj, K. and van Reijmersdal, E. A. (2012). Effects of online advertising format and persuasion knowledge on audience reactions. *Journal of Marketing Communications* **18**

5-18.

Varian, H. R. (2007). Position auctions. *International Journal of Industrial Organization* **25** 1163–1178.

Wang, S., Jank, W. and Shmueli, G. (2008). Explaining and forecasting online auction prices and their dynamics using functional data analysis. *Journal of Business & Economic Statistics* **26** 144-160.

Wang, S., Jank, W., Shmueli, G. and Smith, P. (2008). Modeling price dynamics in eBay auctions using differential equations. *Journal of the American Statistical Association* **103** 1100-1118.

Yoo, C. Y. (2014). Branding Potentials of Keyword Search Ads: The Effects of Ad Rankings on Brand Recognition and Evaluations. *Journal of Advertising* **43** 85–99.

Zhang, S., Jank, W. and Shmueli, G. (2010). Real-time forecasting of online auctions via functional k-nearest neighbors. *International Journal of Forecasting* **26** 666–683.

Dominik Liebl
Statistische Abteilung
University of Bonn
Adenauerallee 24-26
53113 Bonn, Germany
E-mail: dliebl@uni-bonn.de

Stefan Rameseder and Christoph Rust
Institute of Econometrics
University of Regensburg
Universitätsstrasse 31
93053 Regensburg, Germany
E-mail: stefan.rameseder@wiwi.uni-regensburg.de
christoph.rust@wiwi.uni-regensburg.de