DEPARTMENT OF ECONOMICS
AND BUSINESS ECONOMICS
AARHUS UNIVERSITY

CREATES
Center for Research in Econometric Analysis of Time Series

In search of a job: Forecasting employment growth using Google Trends

Daniel Borup and Erik Christian Montes Schütte

CREATES Research Paper 2019-13

Department of Economics and Business Economics
Aarhus University
Fuglesangs Allé 4
DK-8210 Aarhus V
Denmark

Email: oekonomi@au.dk
Tel: +45 8716 5515

In search of a job: Forecasting employment growth using Google Trends*

Daniel Borup† Erik Christian Montes Schütte§

Abstract

We show that Google search activity on relevant terms is a strong out-of-sample predictor for future employment growth in the US over the period 2004-2018 at both short and long horizons. Using a subset of ten keywords associated with "*jobs*", we construct a large panel of 173 variables using Google's own algorithms to find related search queries. We find that the best Google Trends model achieves an out-of-sample R^2 between 26% and 59% at horizons spanning from one month to a year ahead, strongly outperforming benchmarks based on a large set of macroeconomic and financial predictors. This strong predictability extends to US state-level employment growth, using state-level specific Google search activity. Encompassing tests indicate that when the Google Trends panel is exploited using a non-linear model it fully encompasses the macroeconomic forecasts and provides significant information in excess of those.

Keywords: Google Trends, forecast comparison, US employment growth, targeting predictors, random forests, keyword search.

JEL Classification: C22, C53, C55, E17, E24.

This version: August 20, 2019.

*We are grateful to Maik Schmeling, Christian Hansen, Anders Rahbek, Stig Vinther Møller, Thomas Quistgaard Pedersen, Yunus Emre Ergemen, Jonas Nygaard Eriksen, Charlotte Christiansen, Nicolaj Nørgaard Mühlbach, and participants at the joint Econometrics-Finance seminar (2018) at the Center for Research in Econometric Analysis of Time Series (CREATES) at Aarhus University for useful comments, and to CREATES (funded by the Danish National Research Foundation, DNRF78) for research support. An earlier version of this paper has circulated as CREATES Research paper no. 2018-25 under the title "In Search of a Job: Forecasting Employment Growth in the US using Google Trends". Please cite this version and not the early version.

†CREATES, Department of Economics and Business Economics, Aarhus University. Email: dborup@econ.au.dk.

§CREATES, Department of Economics and Business Economics, Aarhus University. Email: christianms@econ.au.dk.

I. Introduction

Employment growth is a measure of economic expansion and regarded as a litmus test for US economic health. As such, it is a leading indicator that is important to policy makers, businesses and job seekers alike. It is one of the key macroeconomic series looked at by the Federal Open Market Committee when determining the path of the federal funds rate, which is the primary tool of monetary policy used by the Fed. Additionally, job growth figures are carefully scrutinized by the media every time they are released. Thus, it is no coincidence that the word "*jobs*" was mentioned a total of 42 times during the 90 minutes long first presidential debate between candidates Hillary Clinton and Donald Trump in September 2016. Despite its significance, employment growth has historically been a relatively difficult macroeconomic series to forecast. A case in point is the period that covered the recession of 2008-2009 and subsequent recovery, where it developed relatively different to projections made by the Bureau of Labor Statistics.[1]

Given the salience of jobs and job growth in the minds of the US working-age population, it should not come as a surprise that latent labor market sentiment leaves a heavy footprint on internet search behavior, particularly from job seekers. A survey made by the Pew Research Center in 2015 found that 80% of the US population uses the internet when searching for a job, and 34% say that it is the most important resource available to them during the job search process (Smith, 2015). In a recent contribution, D'Amuri and Marcucci (2017) show that search volume for the term "*jobs*" is a strong predictor of the unemployment rate in the US. This predictability is also present in international markets, as evidenced by Askitas and Zimmermann (2009), Francesco (2009), and Fondeur and Karamé (2013) who find predictability for the unemployment rate in Germany, Italy, and France, respectively.[2] Nonetheless,

[1]The employment projections can be found on the designated website https://www.bls.gov/emp/.

[2]The evidence for the predictive power of internet search volume for macroeconomic series is not limited to the unemployment rate. Other macroeconomic variables for which there is evidence of predictability are private consumption (Vosen and Schmidt, 2011), initial claims (Choi and Varian,

these studies have solely focused on search volume for a single query or, at best, a small group of queries as predictors, failing to account for the inherent benefits of data rich environments. The potential for high-dimensional models to bring about significant improvements over classical univariate or low-dimensional forecasting models has been documented by several studies, among others, Stock and Watson (2002a,b, 2006), De Mol, Giannone, and Reichlin (2008), Bai and Ng (2008), Buchen and Wohlrabe (2011), Fan, Lv, and Qi (2011), Elliott, Gargano, and Timmermann (2015), Kim and Swanson (2014), and Groen and Kapetanios (2016).

The aim of this paper is to forecast employment using a data rich environment formed by Google search activity and as such the paper has two main contributions. The first is to construct a real-time monitoring device for US employment growth using a broad spectrum of 173 internet search terms related to job-search activity and labor market sentiment. This index can be constructed instantaneously, is free from revisions, and displays much higher forecast accuracy than a large panel of (traditional) macroeconomic and financial variables. Our second contribution is to adapt state-of-the-art methods for forecasting with high-dimensional panels to the case of Google search activity and show that this results in much higher predictive power than models based on the single keyword, e.g. "*jobs*". By combining a large and heterogeneous set of Google search terms, we benefit in three important ways. First, each additional regressor has the potential of contributing with supplementary information. Second, the inclusion of different terms can possibly alleviate sample selection issues that arise due to variation in internet use across different groups by income and age since semantically related terms can potentially capture the same type of information but across distinctive demographical groups. Third, it minimizes the impact of noise in the data that arises due to changes in search terms or behavior across time.

Google Trends has several advantages over classical statistical measures used for

2012), and building permits (Coble and Pincheira, 2017).

macroeconomic forecasting. More specifically, official statistics are usually released with a lag and they are subject to substantial revisions. Household and business surveys can be more timely and they are relatively free from revisions but they are costly to obtain and might suffer from selection biases in response rates. Google Trends on the other hand, can be obtained in real time, be restricted to specific geographical areas, and can even be obtained at (intra-)daily frequencies. Moreover, the ease with which you can download additional Google Trends series makes it easy to expand the panel of predictors.

Starting with a set of ten keywords, we use Google's own algorithms to find semantically linked search queries and thereby expand the panel to a high-dimensional setting. By using text as data we face a sparse and non-linear structure (Kelly, Manela, and Moreira, 2018; Gentzkow, Kelly, and Taddy, 2019). Given these attributes of our text data set, we use soft thresholding variable selection based on regularization from Elastic Net, as proposed by Bai and Ng (2008), to choose the best ten predictors at each point in time within this large panel. We then employ off-the-shelf Random Forests to form predictions. This forecasting procedure yields consistently superior performance to benchmark models for horizons between one month and one year ahead, producing out-of-sample R^2 measures between 26% and 59%. This strongly outperforms benchmark models that employ a traditional high-dimensional panel of 128 macro, financial, and sentiment variables from McCracken and Ng (2016). The improvement is striking at horizons of 6-12 months ahead. The conclusions are robust to employing linear models such Bagging as in Rapach and Strauss (2012) or Complete Subset Regressions (Elliott, Gargano, and Timmermann, 2013; Elliott et al., 2015). In contrast to the benchmark models, Google Trends based models are particularly adept at forecasting employment growth during the latest recession and recovery that followed. In an attempt to further generalize our results and broaden their applicability, we construct state-level Google search activity panels and forecast the employment growth within each US state. We find an overall considerable degree

of predictability. Our forecasting methodology delivers out-of-sample R^2 measures that exceed 60% for some states at all forecast horizons analyzed. Even at a horizon of 12 months ahead, we obtain out-of-sample R^2 measures above 40% for 20 states. The states with strongest predictability tend to be highly populated. This might partly be attributed to less measurement error in search activity trends and a larger number of relevant keywords. In a forecast encompassing exercise, we show that the information embedded in Google Trends forecasts generally encompass that in the benchmark forecasts using the macro-financial data set and, for horizons above one month, they provide significant information in excess of that in the benchmark models. The general superior performance of the model appears to arise from the combination of heterogeneous search queries with its flexibility to let the selected keywords vary over time. Finally, we show that our results are robust to the choice of search terms used to build the data set and estimation window and scheme, and that they are very unlikely to be spurious using a placebo test in the spirit of Kelly and Pruitt (2013).

The rest of the paper is laid out as follows. In Section II we present the methodology used to construct the panel of predictors for both the Google Trends data and the benchmark data set. Section III introduces the main models we use to forecast employment growth as well as the methods we use to draw inference on predictability. In Section IV we present empirical findings, compare alternative models build, and discuss our results. We show the robustness of the results in Section V. Finally, in Section VI we present some concluding remarks.

II. Data

The sample that we use for this analysis spans from 2004:M1 to 2018:M12 and has a monthly frequency. The starting date is determined by the availability of Google Trends data. We obtain data for our target variable, seasonally adjusted employment growth in the US, from the Bureau of Labor Statistics.

Our set of search volume data predictors are obtained from Google Trends, which provides a time series index on the proportion of queries for a search term in a given geographical area.[3] The proportion of queries for a particular keyword is normalized by the total amount of Google searches in the selected geographic area and time range. The resulting number is then scaled on a range between 0 and 100 such that the maximum volume for the particular query in the selected time period takes the value 100. Due to privacy concerns, Google Trends does not explicitly provide its users with the actual number of queries made for each keyword. Nonetheless, for the purpose of forecasting, this does not represent a problem since we are only interested in the time series dynamics of relative search activity. A very useful feature of Google Trends is that, for each keyword, the user is provided with a list of up to 25 related terms (also referred to as related queries henceforth).[4] The final number of related queries depends on the search volume of the original query, i.e. relatively low volume series will have fewer than 25 related terms. According to Google, related terms are selected by looking at terms that are most frequently searched with the term you entered within the same search session. Although the precise algorithm that determines the related terms is proprietary, the output is generally intuitive. For example, querying for the term "*jobs*" in the US for the period of interest returns a list of 25 related terms of which the top five are: "*county jobs*", "*craigslist jobs*", "*jobs indeed*", "*indeed*", "*jobs hiring*".[5] From a forecasting perspective, this functionality is appealing for at least two reasons. First, each semantically related keyword can potentially provide additional information about the target variable and thereby truly harness the predictive power of "Big Data". Secondly, the algorithm performs a form of variable selection since it selects queries with high search volume that might be

[3] See http://www.google.com/Trends.

[4] Google divides related queries into two main categories, top and rising. We use the top related terms in our analysis.

[5] Indeed.com is an American worldwide employment-related search engine for job listings launched in November 2004. Craigslist.com is an American classified advertisements website with sections devoted to jobs, housing, for sale, items wanted, services, community service, gigs, résumés, and discussion forums.

unknown to the researcher. A clear example of this are the terms "*craigslist*" and "*indeed*" which are widely popular job search web pages, but this is not necessarily known nor exploited by the forecaster.

To construct the main set of predictors, which we denote $\boldsymbol{X_g}$, in a manner that is as objective as possible we rely on another service from Google called Google Keyword Planner which, for a given keyword, provides you with the most relevant keywords to include in your webpage to increase webtraffic. Starting with the search term "*jobs*" we use the Keyword Planner to obtain the top ten keywords associated with this term; "*jobs*", "*government jobs*", "*part time jobs*", "*online jobs*", "*career*", "*top jobs*", "*jobs hiring*", "*job*", "*job search*" and "*job vacancies*".[6] We take inspiration from Da, Engelberg, and Gao (2014) and call these words primitive queries (or alternatively primitive terms).[7] Figure 1 shows the Google Trends for our primitive queries over the period of interest. Figure 1 also shows how some of the queries, i.e. "*government jobs*" and "*online jobs*" clearly increase during the financial crisis as a result of the large drop in employment during this period which led an increasing amount people to look for job opportunities over the internet

For each of the ten primitive queries, we add their related terms and remove duplicates, low volume series and series that are clearly unrelated to the employment sentiment.[8] This methodology follows Da et al. (2014), who start with a set of primitive queries and then add related terms (removing duplicates, low volume series and

[6]Note that the keyword "*jobs*" is shown by D'Amuri and Marcucci (2017) to be a very good predictor of the unemployment rate in the U.S.

[7]We show in Section V results from the use of "*employment*" and "*unemployment*".

[8]We define low volume series as those for which more than than 95% of the observations are larger than 0. Da et al. (2014), working with data at a daily frequency, define low volume series as those for which there is less than 1,000 positive observations in their sample. Economically unrelated terms are those which are clearly unrelated to the main query from an economic or sentiment perspective. For example, "*nose job*" and "*Steve Jobs*" are among the related terms for the query "*jobs*" and we cannot expect these terms to have any predictive power for employment growth. The Elastic Net is generally successful at removing these terms, hence, the results presented here are not really sensitive to whether or not we manually remove these terms or not. Moreover, note that the spike in "*jobs*" in October 2011 coincides with the death of Steve Jobs. D'Amuri and Marcucci (2017) removes the effect of this observation. We leave the observation unaltered since we want to maintain generality of our methodology and rely on our predictor targeting (EN) to sort out the irrelevant information.

unrelated queries) to enrich the data set. Our raw data set (excluding duplicates) has 237 keywords that become 180 after removing low volume queries and 173 once economically unrelated terms are removed. As noted by D'Amuri and Marcucci (2017), Google Trends are created based on a sample of queries that change according to the time and IP address of the computer used to download the data. To account for the resulting sampling error, we compute the index for all Google Trends queries based on an average over 20 different days. The average correlation across different samples is always above 0.98. The results are, as a consequence, not sensitive to this precaution.

Following Da, Engelberg, and Gao (2011), Da et al. (2014), and Vozlyublennaia (2014), we start by converting the series to their natural logarithm. This is primarily done to account for the high volatility in some of the series. Considering Figure 1, there are two other things that stand out from Google Trends data. First, they contain a strong yearly seasonal component. Secondly, the series appear to be relatively heterogeneous in terms of their order of integration and whether they contain deterministic trends.[9] We account for the former by regressing each Google trends series on monthly dummies and taking the residuals of this regression. To address the latter, we adopt a sequential testing strategy in the spirit of Ayat and Burridge (2000). The idea is to successively test for stationarity, linear trend stationarity and quadratic trend stationarity using an augmented Dickey-Fuller (ADF) test. Hence, the first test is an ADF test with a constant term. If the null of non-stationarity is rejected, we stop and use the series without any transformation. Conversely, if the null is maintained, we use an ADF test that includes both a constant and a linear time trend. If the null of this second test is rejected, we linearly detrend the series by using the residuals of

[9]There is indeed no consensus on the literature as to whether or not Google Trends data is best characterized by stationarity, trend stationarity or a unit root since this appears to be completely dependent on the query in question. Choi and Varian (2012), Vozlyublennaia (2014), Bijl, Kringhaug, Molnár, and Sandvik (2016), and D'Amuri and Marcucci (2017) do not perform any differencing or detrending of the series, which posits that the Google Trends they use are stationary. Yu, Zhao, Tang, and Yang (2019) use an ADF test on three Google Trends queries "*oil inventory*", "*oil consumption*", and "*oil price*" and find evidence of stationarity at the 5% level (10% level) in "*oil inventory*" ("*oil consumption*") but are not able to reject the null of a unit root for "*oil price*". Da et al. (2014) take log-differences on the series.

a regression of the series on a constant and a time trend. Otherwise, we run a final ADF test that includes a constant, a linear trend and a quadratic trend. If we reject the null of this test, we detrend the series by a similar methodology as before but including a quadratic trend in the regression. Otherwise, we take first differences. All ADF tests are performed with a maximum lag length of 4 with the optimal number of lags selected by the BIC. We conduct each sequential test at the 1% level.[10] The list of all keywords and associated transformations (using the full sample) can be found in the Appendix. To avoid look-ahead bias, we deseasonalize and perform the sequential testing for unit roots on a recursively expanding window, where the smallest window used matches our estimation window for the forecasting model. Hence, only information available at time t is used in both procedures.[11] Figure 2 shows four log deseasonalized queries exemplifying all the possible transformations that each series can undergo when applying the Ayat and Burridge (2000) procedure on the full sample. For the series in the top panel, "*job fair*", we find evidence of stationarity (no transformation). For the series in the second panel, "*career opportunities*", we find evidence of linear-trend stationarity (linear detrending). For the series in the third panel, "*top jobs*", we find evidence of quadratic-trend stationarity (quadratic detrending). Finally, for the one in the bottom panel, "*government jobs*", we cannot reject the null of a unit root (first differences). Note that the latter series is not a related term but a primitive term. Hence, the effect of taking the log transform and deseasonalizing can also be seen by comparing the raw series data, shown in the upper right panel of Figure 1 with the lower left panel in Figure 2, which is log transformed, deseasonalized and standardized.

[10] Ayat and Burridge (2000) note that the procedure is able to retain relatively good size even though multiple tests are involved. We also note that using a 1% significance level on three consecutive tests will result, at most, at a nominal size of 3%, which is still fairly conservative.

[11] Note that this can result in some discordance (across time) about the presence of a unit root or deterministic Trends in some series. In particular, due to the low power of unit root tests in small samples, some of the series might be initially characterized as having a unit root and later on, as more information becomes available, they will be characterized as stationary or trend stationary.

A. Benchmark data set

Our benchmark data, which we denote $\boldsymbol{X_m}$, is comprised by a large panel of 128 macroeconomic and financial indicators. Data is obtained from McCracken and Ng (2016) and transformed according to authors' recommendations. In the Appendix we list all variable abbreviations and their associated descriptions. The data set represents broad categories of macroeconomic time series related to real activity, such as real output and income, employment and working hours, housing starts, and inventories, financial indicators such as bond and stock market indices, dividend yields, foreign exchange measures and stock market (implied) volatility, and sentiment indicators like the consumer sentiment index. This choice of benchmark data set is natural for at least two reasons. First, Rapach and Strauss (2008, 2010, 2012) consider a (pre-selected) set of macroeconomic and financial variables when forecasting employment growth. Secondly, the data set is high-dimensional similarly to the Google Trends panel, aiding benchmark results with a fair chance in the comparison.[12]

III. Forecasting methodology and inference

In this section, we outline our empirical methodology and briefly describe the methods we use to draw inference on the predictive performance of the models. We note in this regard three distinctive attributes of text as data. First, it is high-dimensional by definition as a large amount of unique phrases is available. Secondly, the structure of text data is typically sparse in the sense that several phrases have few searches and may be of little relevance to a given target variable. Thirdly, text is inherently non-linear and, as such, likely non-linearly associated with a given objective, see also Kelly et al. (2018) and Gentzkow et al. (2019). As such, our main forecasting model with Google Trends data will be Random Forests (Breiman, 2001) with a pre-

[12]Note that the benchmark data set contains employment growth itself, thus it also allows autoregressive effects and for a direct AR(1) model when we consider individual predictors.

selection step that targets relevant predictors using regularization techniques. We exploit the high-dimensional features of our data set, while acknowledging the sparse structure, in the targeting step, and allow for interaction and higher-order effects of the data by using RF. We elaborate in separate sections below. In an effort to make a methodologically fair comparison that can keep model effects constant, we also use RF on our benchmark data set, $\boldsymbol{X_m}$.

Given the different type of data characteristics in $\boldsymbol{X_m}$, it is possible that other forecasting methodologies are more appropriate. To make a fair comparison, we also include Bagging and the Complete Subset Regressions (CSR) method of Elliott et al. (2013). We include the former because Rapach and Strauss (2012) show that it can produce significant improvements in employment growth forecast accuracy over the autoregressive benchmark. CSR is included because Elliott et al. (2013) report that this forecast combination approach shows strong performance when compared to alternative forecasting techniques such as ridge regression, Bagging and LASSO. We also include an autoregressive model as additional benchmark, which is typically employed in the macroeconomic literature. The optimal lag order is determined recursively by BIC.[13]

Let our target variable, which is the h month ahead employment growth rate, be defined as

$$y_{t+h}^{h} = (1/h) \sum_{j=1}^{h} y_{t+j}, \tag{1}$$

where y_t is the log-difference of the seasonally adjusted employment growth at time t. Let us also define our $N \times 1$ vector of predictors at time t by $X_t = \left[X_{1,t,\ldots,}X_{N,t}\right]'$. Note that this should not be confused with the matrix of predictors, e.g. $\boldsymbol{X_g}$ or $\boldsymbol{X_m}$, which we denote with bold letters.

[13]Note that this differs from the direct AR(1) model that results from having employment growth in the panel of predictors, both because we allow for higher order models and the forecasts are made by an iterated model instead of a direct forecast.

A. Targeted predictors

To exploit our high-dimensional, data rich-environment as well as to maintain feasibility of the somewhat low-dimensional Bagging and CSR methodologies, we make use of targeted predictors.[14] Even though RF remain feasible in this high-dimensional setting, we aim to put all methods on equal footing and, as such, employ RF with targeted predictors as well. Moreover, the benefits of RF are often reduced when the input is high-dimensional, making targeted predictors natural using this methodology as well (Gentzkow et al., 2019).[15] Lastly, as noted above, it is likely that our predictors enjoy a sparse structure. Targeting predictors is due to Bai and Ng (2008) which take into account the fact that not necessarily all series in X_t are important when forecasting the target variable. The idea is to first pre-select a subset $X_t^* \in X_t$ of predictors that are targeted to the forecasting object and then subsequently estimate a forecasting model of interest. Bai and Ng (2008) propose both soft and hard thresholding regularization techniques for constructing X_t^*. In this section, we focus only on soft thresholding, which is based on dropping uninformative regressors using penalized regressions.[16] More specifically, we use the Elastic Net (EN) estimator of Zou and Hastie (2005) since it performs well when predictors are correlated.[17] If we let RSS be the residual sum of squares of a regression of y_{t+h}^h on X_t, EN solves the problem

$$\hat{\beta}^{\text{EN}} = \underset{\beta}{\text{argmin}} \left[RSS + \lambda \left((1-\alpha)\tfrac{1}{2} \parallel \beta \parallel_{\ell_2}^2 + \alpha \parallel \beta \parallel_{\ell_1} \right) \right] , \tag{2}$$

where $\alpha = (0,1]$ selects a weight between the LASSO and ridge regression, λ is a tuning parameter and $\| \cdot \|_{\ell_i}$ denotes the ℓ_i norm for $i = \{1,2\}$. Both the LASSO (Tibshirani,

[14] Strictly speaking, the Bagging method presented below can handle high dimensional data sets, however, hard thresholding runs the risk of including more predictors than there are observations in the estimation window. CSR can also be modified to handle large-dimensional sets as shown in Elliott et al. (2015), but soft thresholding variable pre-selection is much simpler to implement and puts all methods on a level playing field.

[15] The results of using RF without targeted predictors do not alter our conclusions. Targeted predictors improve the performance of RF by a small margin in our sample.

[16] We cover hard thresholding in the Bagging model below.

[17] We find that using the LASSO estimator of Tibshirani (1996) instead of the Elastic Net does not alter the results in any significant way.

1996) and ridge estimators work by regularizing the coefficients of unimportant or irrelevant predictors towards zero. The main difference is that ridge will only decrease the absolute size of the coefficients but it will never set them exactly equal to zero. In contrast, the LASSO is able to set the coefficients to zero and thus perform variable selection. We can then construct the soft threshold X_t^* by

$$X_t^* = \left\{ X_i \in X_t \mid \beta_i^{\text{EN}} \neq 0 \right\}, \tag{3}$$

with $i = 1, \ldots, N$. We follow Bai and Ng (2008) and tune λ such that ten predictors are selected. We set $\alpha = 0.5$ which means that ridge and LASSO regression get an equal weight. Hence, the idea is to use the EN estimator to remove uninformative predictors from X_t and thereby improve on the forecast of the target variable from a high-dimensional outset.

B. Bagging

Our implementation of Bagging follows the lines of Inoue and Kilian (2008). We start with the defining the hard-threshold multivariate forecast with N exogenous predictors

$$\begin{gathered} \hat{y}_{t+h}^{h,HT} = \hat{\alpha} + \sum_{i=1}^{N} \hat{\delta}_i X_{i,t}^{HT}, \\ X_t^{HT} = \left\{ X_i \in X_t^* \mid \left| t_{X_i} \right| > 2.58 \right\} \text{ with } i = 1, \ldots, N \end{gathered} \tag{4}$$

where t_{X_i} is the t-statistics formed on $\hat{\delta}_i$. Thus, from the variables in X_t^*, we select only those that are statistically significant at the 1% level.[18] The procedure is then augmented by using a moving block bootstrap to reduce variance coming from model uncertainty. More specifically, we generate B bootstrap samples by randomly drawing blocks of size m from the $\{y_{t+h}, X_t\}$ tuple. We then calculate (4) for each bootstrap sample using information only up to time t, and compute the hard-threshold bootstrap forecast, $y_{b,t+h}^h$, using bootstrap coefficients and original data X_t^{HT}. The Bagging

[18] Note that this is essentially combining a soft threshold pre-selection procedure (targeting predictors) with hard thresholding. Following Rapach and Strauss (2012) we use Newey and West (1987) standard errors to calculate the t-statistic. The lag truncation is set to $h - 1$.

forecast for $\hat{y}^h_{t+h}$ is then given as the average of the $B = 400$ hard threshold bootstrap forecasts

$$\hat{y}^h_{t+h} = \frac{1}{B}\sum_{b=1}^{B} \hat{y}^h_{b,t+h}. \tag{5}$$

We maintain the autocorrelation structure of the target variable by applying the circular block bootstrap of Politis and Romano (1992) with block size chosen optimally according to Politis and White (2004).[19]

C. Complete Subset Regressions

The Complete Subset Regressions (CSR) method of Elliott et al. (2013) is based on the idea of taking all combinations of models restricted to using a fixed number of regressors $k < N$. Specifically, if we let $X_{l,t}$ denote the matrix of predictors containing k variables for each model $l = 1, ..., M$ the l'th model forecasts is

$$\hat{y}^h_{l,t+h} = \hat{\alpha} + \hat{\beta} X_{l,t}, \tag{6}$$

such that the Complete Subset Regression forecast is given by

$$\hat{y}^h_{t+h} = \frac{1}{M}\sum_{l=1}^{M} \hat{y}^h_{l,t+h}. \tag{7}$$

We select model combinations that include a maximum of $k = 3$ predictors to maintain parsimony of the models, which has been shown by Elliott et al. (2013) to yield superior predictive accuracy.[20]

D. Random Forests

Both the Bagging and CSR methods above rely on linear prediction models. A means to allow for non-linearities is regression trees that nonparametrically estimates the

[19] For robustness we also used $m = h$ as Inoue and Kilian (2008), but the results are insensitive to this alteration since the Politis and White (2004) method tends to choose an optimal block size close to h.

[20] We find that results are similar, yet slightly weaker for $k = \{4, 5, 6\}$ across both $\boldsymbol{X_g}$ and $\boldsymbol{X_m}$.

function $f^h(X_t) = \mathbb{E}[y^h_{t+h}|X_t]$ to form forecasts using $\hat{f}^h(X_t)$. A regression tree is based on the principle of sequentially splitting the space of predictors into several regions, characterized by nodes, and model the response by the mean of y^h_{t+h} within each region. For instance, in the case of macroeconomic predictors, the tree might split according to expansionary and recessionary states, low stock-market volatility and high stock-market volatility, weak inflationary and strong inflationary states, etc., and provide forecasts within each of those regions. Moreover, the regression tree facilitates interactions by defining, for instance, a region as expansionary states together with weak inflation. Additional higher-order interactions can be entertained as well as higher-order terms of the single predictors. The tree regression forecast of y^h_{t+h} using X_t is, thus, the average of y_t within the region for which X_t falls into, that is,

$$\hat{f}^h(X_t) = \frac{1}{T_l} \sum_{l=1}^{M} y_t \mathbb{1}\{X_t \in R_l\}, \tag{8}$$

where $R_1, \ldots, R_M$ represents the region partition of the space of predictors and T_l the number of samples falling into region R_l. Random forests (RF) are ensembles of regressions trees proposed by Breiman (2001) and is based upon bagging of randomly constructed regression trees to control variance (overfitting). Each of the regression trees is specified on a bootstrapped sub-sample of the original data, which we denote by f^h_b, $b = 1, \ldots, B$. The final prediction by RF is then obtained by

$$\hat{y}^h_{t+h} = \frac{1}{B} \sum_{b=1}^{B} \hat{f}^h_b(X_t). \tag{9}$$

Similarly to the (linear) Bagging above, we take $B = 400$ in our application of RF. We use default parameters of the RF implementation in MATLAB, i.e. bagging from the full sample (with replacement) using a third of the total number of predictors used as input to split each node. The default number of predictors coincides with the number used in CSR. The regions are estimated using the CART algorithm based on least

squares criterion, following Breiman (2001) to which we refer for additional details.

E. Inference on predictability

We compare the performance of the competing models using the Campbell and Thompson (2007) out-of-sample R^2 defined as

$$R^2_{OoS} = 1 - \frac{\sum_t \left(y_t - \hat{y}_{m,t}\right)^2}{\sum_t (y_t - \bar{y}_t)^2}, \tag{10}$$

where $\bar{y}_t$ is the rolling-mean forecast, which is computed on a window that matches the model estimation window and $\hat{y}_{m,t}$ is the forecast of the model in question at time t. This measure lies in the range $(-\infty, 1]$, with negative numbers indicating that the model in question performs worse than the historical mean of the series. We conduct out-of-sample inference using the Diebold and Mariano (1995), t_{DM}, test statistic. The null hypothesis of the Diebold-Mariano test used in this paper is that the model in question does not beat the rolling mean of the series, while the alternative is that it does. Hence, it may be interpreted as the t-statistic of the R^2_{OoS}. When forecasting for horizons $h > 1$, we adjust for the moving average structure of the forecast errors by using Newey and West (1987) standard errors in the denominator of the test statistic with a bandwidth length equal to $h - 1$.

A positive R^2_{OoS} measure tells us that the model in question outperforms the rolling-mean benchmark by looking at the ratio of forecast errors over the whole out-of-sample period. However, it is possible that the model in question is only beating the rolling mean during a subset of the evaluation period and it is underperforming during others. To assess the stability of predictive accuracy, we follow Welch and Goyal (2008) and compute and plot the cumulative sum of squared error difference (CSSED) between the model of interest and the rolling mean model. The CSSED for a given

model, l, at time t is computed as

$$CSSED_{l,t} = \sum_{i=R}^{t} \left((y_t - \bar{y}_t)^2 - \left(y_t - \hat{y}_{l,t} \right)^2 \right), \tag{11}$$

where R and t are the beginning and the end of (the increasing) forecast evaluation period, respectively. For any t, a positive $CSSED_{l,t}$ means that model l is outperforming the rolling mean up to that point in time t. Increases in the $CSSED_{l,t}$ trajectory means that model l is improving against the rolling mean benchmark at that specific point in time t and vice versa for decreases.

Since we are dealing with several competing models, we also employ the model confidence set (MCS) approach developed by Hansen, Lunde, and Nason (2011) to compare the performance of the models. This approach returns a confidence set that includes the best model with probability $(1-\alpha)$, $\alpha = 20\%$ being the chosen significance level of the testing procedure. We use squared forecast errors as a loss function and set the bootstrap block size equal to h when applying the MCS. We rely on the range statistic to draw inference.

IV. Empirical results

This section presents the out-of-sample performance using our novel Google Trends data versus the benchmark macroeconomic data and investigates the source of any predictability. We also examine the informational content in either of the two forecasts by considering predictability from their combined data sets and a formal encompassing analysis. Lastly, we consider forecasting US state-level employment growth. In the following, we refer to RF using the data set $\boldsymbol{X}_d$, $d = \{g, m\}$, by $RF(\boldsymbol{X}_d)$ and analogously for Bagging and CSR.

A. Employment growth predictability

Table 1 shows the R^2_{OoS}, p-value of the Diebold-Mariano test statistic, and an inclusion indicator (shaded text) for the MCS of the competing models. All models are estimated using a rolling window scheme with 48 observations in the sample period 2004:M1-2018:M12. We forecast at horizons of $h = \{1,3,6,9,12\}$ months ahead. Thus, the first forecast for horizons of less than nine months ahead occurs during the recession of 2008-2009, allowing us to assess the performance of the models during a large contractionary period in the labor force. Panel A and Panel B show the RF, Bagging and CSR results for the Google Trends panel, $\boldsymbol{X_g}$, and the macroeconomic predictors, $\boldsymbol{X_m}$, respectively.[21] Panel C shows the results for the autoregressive model.

The results in Panel A show that $\boldsymbol{X_g}$ demonstrates a striking degree of forecasting power irrespective of the method used. However, it is clear that RF is best suited to utilize the predictive content of the Google Trends panel. By corollary, non-linearities in search activity is important and RF is successful in exploiting those. This is particularly the case for the one year ahead forecast where $RF(\boldsymbol{X_g})$ achieves a R^2_{OoS} of 59.15% which is more than double the one achieved by $Bagging(\boldsymbol{X_g})$ (27.41%). The performance of $CSR(\boldsymbol{X_g})$ is similar to its RF counterpart albeit lower for horizons above six months ahead. In terms of significance, the $RF(\boldsymbol{X_g})$ delivers results that are significant at conventional levels for across all forecast horizons. For $CSR(\boldsymbol{X_g})$ the results are similar although less statistically significant in general. Even though $Bagging(\boldsymbol{X_g})$ produces positive R^2_{OoS} measures, they are only significant at the 10% level for $h = \{1,3,6\}$ and insignificant otherwise.

Moving to Panel B, we find that none of the $\boldsymbol{X_m}$ models beat the $RF(\boldsymbol{X_g})$ at any forecast horizon and that they generally underperform their Google-based counter parts. The exception seems to be the $Bagging(\boldsymbol{X_m})$ which performs better than $Bagging(\boldsymbol{X_g})$ for $h = \{1,3\}$ with R^2_{OoS} measures that surpass its competitor by approximately 6 per-

[21] Plots of the actual employment growth versus the forecast of the best-performing model using $\boldsymbol{X_g}$ and $\boldsymbol{X_m}$ can be found in the Appendix.

centage points. Overall, it also seems that Bagging is the best performing methodology for macroeconomic predictors, the exception seems to be $h = 9$ where $RF(\boldsymbol{X_m})$ is the only macro-based model that delivers a positive (although statistically insignificant) R^2_{OoS} measure. It is also worth noting that none of the $\boldsymbol{X_m}$ models are able to beat the rolling mean benchmark for the one year ahead forecast. This stands in stark contrast to the $\boldsymbol{X_g}$ models which produce positive and large R^2_{OoS} measures for this forecast horizon. Finally, we note that the forecasting performance of the autoregressive model is relatively poor since this model obtains positive R^2_{OoS} measures only for $h = \{1, 3\}$ of which only the results for $h = 1$ are significant at the 10% level.

Considering the MCS results, it is worth noting that the only model that is included in the confidence set for all forecast horizons is the $RF(\boldsymbol{X_g})$ model. This is not surprising given its strong predictive performance. However, for $h = 1$ all models with the exception of $Bagging(\boldsymbol{X_g})$ are included in the model confidence set implying that we cannot statistically conclude that there is any difference in performance across them at this forecast horizon.[22] For $h = 3$ we have that all the Google-based models are included, but only $Bagging(\boldsymbol{X_g})$ is included from the three macro-based models. The confidence set for $h = 6$ includes all Google and macro models, but excludes the AR model. Finally, for horizons above six months ahead the only model included is $RF(\boldsymbol{X_g})$.

Figure 3 plots the CSSED for all models across the different forecast horizons. We see that most models have their greatest relative advantage over the rolling mean model during the early part of the forecast evaluation period, i.e. the period during the recession and subsequent recovery. This is particularly the case for horizons below six months ahead where the CSSED lines are increasing steeply during the recession period and then remain elevated subsequently. There is a second period of relative improvements for the majority of the models over the rolling mean in the

[22] We note however that the $RF(\boldsymbol{X_g})$ is always the best ranking model, in other words, the one with the highest model confidence probability.

period between 2012 and mid-2013 where the US economy experienced an accelerated expansion on the number of jobs created. After this period, most models seem to perform on par with the rolling mean benchmark. The figure also helps explaining the superior performance of Google-based models for long horizon forecasts ($h = \{9, 12\}$) since we can see that while some of the $\boldsymbol{X_m}$ initially outperform the Google-based models, they perform very poorly in the mid-2010 to mid-2011 period, and although there is some relative performance recovery in 2012 and 2013, it is not enough to make up for the lost ground. Overall, the CSSED lines for the Google Trends panel are mostly increasing or stable, indicating that the models mostly outperform or perform in par with the no-predictability benchmark of the rolling mean. In conjunction with the results above, we may conclude that the Google Trends panel delivers strong predictive accuracy compared to its benchmarks, in particular at long horizons, and that capturing inherent non-linearities (which may be achieved by RF) is important.

B. Where does predictive power come from?

In the preceding section, we showed that Google Trends have a high degree of predictive power for future employment growth. However, in Section III we argued for the necessity of variable pre-selection procedure given the sparsity of the data and a number of possibly uninformative predictors in $\boldsymbol{X_g}$. These results lead us to the critical question about where that predictive power is coming from. The first thing we do to answer this question is to asses the forecasting power of each individual search term, $X_i \in \boldsymbol{X_g}$, using a univariate regression forecasting model of the form

$$y_{t+h}^h = \alpha + \beta_i X_{i,t} + \varepsilon_{t+h}^h, \tag{12}$$

where the parameters of the model are estimated using OLS and forecasts are generated similarly to the procedures in the former sections. Figure 4 shows top twenty R^2_{OoS} among all the keywords in $\boldsymbol{X_g}$ for $h = \{1, 3, 6, 9, 12\}$ months ahead. The first thing we notice is that some individual predictors appear to be consistently good

across all forecast horizons. A case in point is "*part time job*" which achieves an average R^2_{OoS} of 40.4% across all horizons which makes it almost competitive with the $RF(\boldsymbol{X_g})$ with an average R^2_{OoS} (across horizons) of 48.5%. Nonetheless, the second best individual predictor which is "*job fair*" achieves an average R^2_{OoS} of 28.5%, which, although appreciable, lies substantially below the average R^2_{OoS} for $RF(\boldsymbol{X_g})$. Out of the 173 terms included in $\boldsymbol{X_g}$, seven attain an average $R^2_{OoS} > 20\%$ across all horizons analyzed and 52 have a positive average R^2_{OoS}. Thus, the Google Trends panel can be described as an environment with some strong predictors, several weak predictors, and a relatively large degree of noise. Queries related to almost all primitive keywords appear in the figure, which implies that each primitive keywords might contribute with additional predictive information to the data set. Despite this large amount of heterogeneity in the top search terms, some keywords re-appear relatively often. For example, queries that include "*government*", "*part time*" and "*career*" appear at all horizons, implying that "*government jobs*", "*part time jobs*" and "*career*" are important primitive keywords.

For the sake of comparison, we also show the R^2_{OoS} of the top twenty individual predictors in the benchmark data set, $\boldsymbol{X_m}$, in Figure 5. Not surprisingly, out of the top five variables from $\boldsymbol{X_m}$ in terms of average R^2_{OoS} four are related to the number of employees in the financial, wholesale trade, service, and trade/transportation/utilities industries, respectively, while the fifth one is the VXO stock market implied volatility index. These five predictors have an average R^2_{OoS} across forecast horizons between 42.3% and 35.15%, thus, they are relatively strong predictors on their own.[23] The next five predictors (in terms of average R^2_{OoS} across h) are all related to the unemployment rate, number of employees and business inventories and have average R^2_{OoS} between 34% and 29.3%. We also note that out of the total of 128 predictors in $\boldsymbol{X_m}$, 51 result in a positive R^2_{OoS}, which is proportionally higher than the share in $\boldsymbol{X_g}$.

[23]We note however, that strictly speaking, we test the null hypothesis as many times as we have covariates. A conventional Bonferroni correction would render most individual predictor's accuracy statistically insignificant, even though their predictability appear striking.

The evidence presented until now implies that, although there is predictive information in $\boldsymbol{X_g}$ about future employment growth, no individual predictor can account for the outstanding performance of the $RF(\boldsymbol{X_g})$ model since $\boldsymbol{X_m}$ has more predictors which are individually good. Thus, we can infer that soft thresholding is selecting a particularly good combination of Google Trends at each time period. Figure 6 shows the inclusion per period for the most often included predictors in $\boldsymbol{X_g}$ (ordered by inclusion frequency) as determined by our procedure of targeting predictors.[24] Several features of these figures are noteworthy. First, none of the series in $\boldsymbol{X_g}$ are included in the set for all periods at any forecast horizon. The most included predictors at each horizon, i.e. *"job search sites"* ($h = 1$)*, "job interview"* ($h = 3$), "*job fair*" ($h = 6$), "*interview questions*" ($h = 9$), *"part time job" (*$h = 12$*)* are included in the set between 40.1% and 55.9% of the time. Second, there appears to be heterogeneity across the most frequent predictors, which means that they are generally related to different primitive keywords and therefore each probably contributes with distinct and unique information. Finally, there is some overlap between the top individual predictors in Figure 4 and the most frequently included terms, but the overlap is far from complete, implying that the subset selected by targeting predictors is not necessarily composed of the best individual predictors but a combinations of keywords that produce an even better forecast.

Overall, it seems that Google-based methods outperform their macroeconomic benchmarks because of the heterogeneity of the search terms included, each one contributing to additional information. As such, constructing a high-dimensional panel of Google search activity is desirable from the outset. This information appears to be better exploited with a method that accounts for non-linearities between predictors, such as Random Forests. Interestingly, these non-linearities do not seem to play a big role in the case of the large macroeconomic data set since $RF(\boldsymbol{X_m})$ underperforms $Bagging(\boldsymbol{X_m})$.

[24] A similar plot for the inclusion of predictors from $\boldsymbol{X_m}$ can be found in the Appendix.

C. Combining data sets and encompassing tests

The former sections established an overall superiority of methods based on the Google Trends panel relative to the benchmark data set. However, it is possible that $\boldsymbol{X_m}$ embodies useful information that is not contained in $\boldsymbol{X_g}$. To investigate this, we create a data set that combines $\boldsymbol{X_g}$ and $\boldsymbol{X_m}$, and denote it by $\boldsymbol{X_c} = \left[\boldsymbol{X_g}\, \boldsymbol{X_m}\right]$. We then run the out-of-sample forecast using all three methods using this data, $RF(\boldsymbol{X_c})$, $Bagging(\boldsymbol{X_c})$, and $CSR(\boldsymbol{X_c})$. The results are presented in Table 2. For most forecast horizons the combined data set does not lead to improvements in predictability. For long horizons the performance deteriorates, yet at $h = 1$ (and $h = \{3,6\}$ for $Bagging(\boldsymbol{X_c})$) the predictive accuracy improves, but in most cases only slightly. Considering the inclusion frequency for this combined data set we find that several macroeconomic variables are included in the set of targeted predictors across all horizons.[25] Thus, even though those variables appear as desirable predictors ex-ante, they do not seem to provide additional predictive information.

We also use forecast encompassing tests to formally compare the informational content in the forecasts using Google Trends and macroeconomic data across all methods. To that end, consider the case of forming a combination forecast of employment growth y^h_{t+h} by a convex combination of the two forecasts

$$y^h_{t+h} = \lambda_g \hat{y}^h_{g,t+h} + \lambda_m \hat{y}^h_{m,t+h} \tag{13}$$

where $\hat{y}^h_{g,t+h}$ and $\hat{y}^h_{g,t+h}$ are the forecasts using Google Trends and macroeconomic data, respectively, and $\lambda_g + \lambda_m = 1$. If $\lambda_m = 0$, Google Trends forecasts encompass the macroeconomic forecasts in the sense that the latter do not contribute any relevant information to forecasting employment growth in addition to that contained in the Google Trends forecasts. The alternative $\lambda_m > 0$ suggest the Google Trends forecast do not encompass the macroeconomic forecasts and, as such, the latter contains relevant

[25] Predictor inclusions for the combined data set, $\boldsymbol{X_c}$, can be found in the Appendix.

information in addition to the Google Trends forecasts in predicting employment growth. We follow Rapach and Strauss (2010) and employ the statistic of Harvey, Leybourne, and Newbold (1998) to test the null hypothesis of $\lambda_m = 0$ against the one-sided alternative hypothesis of $\lambda_m > 0$. Accordingly, let $\hat{\varepsilon}^h_{l,t+h}$ denote the forecast error associated with the l'th forecasts, $l = \{g, m\}$. We then define

$$\hat{d}^h_{t+h} = (\hat{\varepsilon}^h_{g,t+h} - \hat{\varepsilon}^h_{m,t+h})\hat{\varepsilon}^h_{g,t+h}, \tag{14}$$

to obtain the test statistic

$$HLN^h = \bar{d}^h (\hat{V}^h)^{1/2}, \tag{15}$$

where $\bar{d}^h = P_h^{-1} \sum_{t=R}^{T} \hat{d}^h_{t+h}$ is the average of $\hat{d}^h_{t+h}$ over the P_h out-of-sample observations and $\hat{V}^h$ is a consistent estimate for the variance of $\bar{d}^h$. We employ a HAC estimator with Bartlett kernel and bandwidth length of $h-1$. Harvey et al. (1998) show that $HLN^h \xrightarrow{d} N(0,1)$ under the null of $\lambda_m = 0$. Similarly to Rapach and Strauss (2010), we employ the modified HLN^h test

$$MHLN^h = \frac{P_h + 1 - 2h + P_h^{-1} h(h-1)}{P_h} HLN^h,$$

which enjoys a asymptotic Student's t-distribution with degrees of freedom $P_h - 1$. The estimated weight $\hat{\lambda}_m$ is obtained as

$$\hat{\lambda}_m = \sum_{t=R}^{T} \hat{d}^h_{t+h} / \sum_{t=R}^{T} (\hat{\varepsilon}^h_{g,t+h} - \hat{\varepsilon}^h_{m,t+h})^2, \tag{16}$$

Analogously, we may test the converse null hypothesis of $\lambda_g = 0$ that the macroeconomic forecasts encompass Google Trends forecasts against the alternative $\lambda_g > 0$ that they do not. This requires redefining $\hat{d}^h_{t+h} = (\hat{\varepsilon}^h_{m,t+h} - \hat{\varepsilon}^h_{g,t+h})\hat{\varepsilon}^h_{m,t+h}$, and the weight $\hat{\lambda}_g$ is obtained using this definition of $\hat{d}^h_{t+h}$ together with (16).

Table 3 reports the estimated weights and associated p-values of the encompassing

analysis. The pattern generally echo that of the analysis using $\boldsymbol{X_c}$. At all horizons and at conventional significance levels, the $RF(\boldsymbol{X_m})$ provides no significant additional information relative to $RF(\boldsymbol{X_g})$, whereas $RF(\boldsymbol{X_g})$ provides significant predictive information above that of $RF(\boldsymbol{X_m})$. A similar picture is obtained using the CSR methods, whereas Bagging suggests both data sets provide significant information. Since Bagging tends to be the worst-performing method and RF the best, we may conclude on this basis that when the Google Trends panel is exploited in a non-linear setting it fully encompasses the macroeconomic forecasts and provides significant information in excess of those.

D. State-level predictability

State-level employment growth is of interest to both local and national policy-makers as well as for business location and investment planning. Figure 7 shows a large degree of cross-sectional heterogeniety in the average employment growth levels as well as their time series variability. Louisiana (LA) exhibits large changes over time with relatively weak employment growth, whereas Texas (TX) has relativeliy little time series variability, yet strong employment growth. Pennsylvania (PA), for instance, has little variability and little employment growth. Given this heterogenity in the dynamics of employment growth, if one identifies a robust forecasting methodology across most states, one can be reasonably confident that the methodology will deliver predictability in other applications as well (Rapach and Strauss, 2012). As an attempt to enhance the generality of the success of the $RF(\boldsymbol{X_g})$ and investigate its applicability on the state level, we construct a Google Trends state-level panel for each of the US states. We follow the same procedures outlined in Section II but restrict the obtained Google search activity of state i to that occuring only in state i itself. We then follow Rapach and Strauss (2012) and use both state-level and national predictors. We also attach the Google Trends panel for the states that have borders next to the state in

question.[26]

In Figure 8 we depict the distribution of R^2_{OoS} across states sorted in a descending manner. It stands out that the $RF(\boldsymbol{X_g})$ is highly successful in predicting state-level employment growth at all forecast horizons. This is also summarized in Table 4. It can deliver an R^2_{OoS} of above 60% for several states, and only a few states at each forecast horizon have a negative R^2_{OoS}. Interestingly, the states with strongest predictability tends to be large and vice versa. We gather 2018:Q4 state-level population data from the Bureau of Labor Statistics and compute each state's share of total population. As reported in Table 4 the R^2_{OoS} when weighted by those population shares is notably higher than the equally weighted one. Moreover, it also stands out that the R^2_{OoS} correlates positively (and significantly for $h = \{1,3\}$) with state population share. This may be explained by three, not necessarily mutually exclusive, reasons. First, smaller states may be less predictable simply due to a larger share of unpredictable, idiosyncratic variation in employment trends. Second, since Google trends is based on a sample of searches which is a function of total search volume, the larger population, the less measurement error is present in Google Trends. Third, since we eliminate keywords with insufficient search volume less populated states have less predictors in $\boldsymbol{X_g}$. This can also be seen from Table 5. For instance, Alaska (AK), which is less populated and has no bordering states, uses a total of 203 keywords, 173 of which comes from the national Google Trends panel. Accordingly, cf. Figure 8, AK is generally one of the least predictable states across all forecast horizons. On the other hand, California (CA), which has a large number of state-level keywords, yet not as many highly populated bordering states, has much stronger predictability with R^2_{OoS} consistently above 50% at all forecast horizons. Utah (UT) to the contrary, has relatively few state-level keywords but a large number of keywords coming from neighbouring states. It shows R^2_{OoS} at 60% or above. North Carolina (NC) has a large number of both state-level keywords and from neightbouring states, totalling 741

[26] Predictability generally remains using state-only data or by including just those from bordering states, yet with less significance. This occurs at especially longer horizons.

keywords in use before targeting predictors. As such, its predictability is generally high at 50% or above.

V. Robustness checks

In this section we show that the forecasting power of $RF(\boldsymbol{X_g})$, which is our main model, is not sensitive to the words we use to construct the data or alternative estimation windows. Finally, we also show that the methodology we use does not result in spurious out-of-sample predictive power by running a placebo test.

A. Alternative keywords

The primitive set of words we use to build $\boldsymbol{X_g}$ are based on the keyword "*jobs*". This raises the possibility that the results depend on that particular keyword. To address this issue we construct two alternative Google trends data sets by selecting primitive keywords using the Google Keyword Planner but taking the top ten terms linked to "*employment*" and "*unemployment*", respectively. The primitive keywords for the former are *"employment", "jobs", "job search", "job", "government jobs", "jobs hirinig", "job vacancies", "online jobs", "career"* and *"part time jobs".* , whereas the primitive keywords for the latter are *"unemployment", "unemployment rate", "unemployment office", "ui online", "unemployment insurance", "unemployment benefits", "unemployment number", "file for unemployment", "apply for unemployment"* and *"unemployment claim".* Note that the overlap between the primitive keywords for $\boldsymbol{X_g}$, and the alternative data set based on *"employment"* is almost complete and the only difference is that $\boldsymbol{X_g}$ includes "*top jobs*" instead of "*employment*", thus we can expect similar predictive performance.[27] The data set based on "*unemployment*" has no overlap in primitive keywords with the other two. Once we have the set of primitive keywords, we follow the same procedure as with $\boldsymbol{X_g}$ and end up with 176 terms for the data set based on "*employment*" and 120 for the data set based on "*unemployment*". As before,

[27] By the same token, using primitive keywords derived from "*careers*" leads to almost identical results.

we remove seasonality and detrend recursively when estimating the RF forecasting model. The R^2_{OoS} and p-values from the Diebold- Mariano test using a rolling window of 48 observations are shown in Table 6. The results for the alternative data set based on ""*employment*" are very similar to $\boldsymbol{X_g}$, albeit slightly lower, except for $h = 6$. This is to be expected given the large degree of overlap in primitive keywords. The forecasting performance for the alternative data set based on "unemployment" is worse than the other two, particularly at horizons above $h = 1$. We note, however, that for $h = 9$ the R^2_{OoS} of the model is still on par with $RF(\boldsymbol{X_m})$, and strongly outperforms $Bagging(\boldsymbol{X_m})$ and $CSR(\boldsymbol{X_m})$. At $h = 12$ it outperforms all methods using macroeconomic data notably. We believe the decrease in predictive accuracy is natural as we find it more likely that individuals search for words related to "*jobs*", "*employment*", and "*careers*" rather than "*unemployment*" when looking for a job.

B. Alternative estimation windows

The CSSED analysis shows that the predictive performance of the $RF(\boldsymbol{X_g})$ has strong predictive ability during the recent crisis and continues to improve, albeit less pronounced, subsequently. To further check the robustness and stability of the results, we also perform the $RF(\boldsymbol{X_g})$ forecast with three alternative estimation windows; a rolling window of 36 observations, a rolling window of 60 observations and an expanding window with 48 initial observations. Table 7 shows the R^2_{OoS} and p-values of the Diebold-Mariano test of these alternative estimation windows. Decreasing the size of the rolling estimation window results in R^2_{OoS} measures that are slightly smaller (3-6 percentage points) for $h = \{1, 3, 6\}$, and more pronounced (12-24 percentage points) for $h = \{9, 12\}$. This decrease in performance is possibly arising for two reasons. First, a smaller estimation window will inevitably lead to noisier parameter estimates. Second, the sample evaluation period is longer, which means that it covers most of the recession of 2008-2009, a period that is inherently more difficult to forecast at primary longer horizons. The results for the models that are estimated using a longer rolling window of 60 observations are similar to the ones we obtain for our main

model (with an estimation window of 48 observations), implying that the results are not sensitive to these changes. This also indicates the the predictability obtained from $RF(\boldsymbol{X_g})$ is not purely driven by the recent economic crisis. The expanding window scheme results in values of R^2_{OoS} similar to that for the short rolling window scheme. Overall, while decreasing the estimation window has a negative effect on predictability for $RF(\boldsymbol{X_g})$, the R^2_{OoS} measures remains statistically significant and generally larger than those using the macroeconomic data set, especially at longer horizons. In unreported results, we find that decreasing the estimation window to 36 observations has an even larger (negative) effect on models based on $\boldsymbol{X_m}$ with forecasts at longer horizons deteriorating.

C. Placebo test

To show that the our RF methodology does not result in spurious out-of-sample predictive ability, we construct a placebo test in a spirit similar to Kelly and Pruitt (2013). If the methodology results in a mechanical bias, simulated placebo data that is similar to the data in $\boldsymbol{X_g}$ or $\boldsymbol{X_m}$, but unrelated to our target variable, will also display out-of-sample predictability.[28] As such, for each time period, we generate thirty AR(1) series that have the same mean, variance and autoregressive coefficient as the series selected by soft thresholding at each time period in the out-of-sample window. Innovations in the series are generated using an i.i.d. normal distribution that has zero covariance with our target variable. Thus, they are independent of y_t. Table 8 shows summary statistics of the empirical distribution of the R^2_{OoS} under the null of no predictability obtained from 10,000 placebo replications. It is clear that the predictability obtained from the combination of RF and Google Trends data is very unlikely spurious at all forecast horizons since they are notably greater than the 99% percentile. It even exceeds the maximum across the simulations. That is, the placebo test shows that the probability of actually getting a positive R^2_{OoS} by chance

[28] As the Google Trends panel and the macroeconomic data set likely exhibit difference time series characteristics, we run the placebo test twice, using either $\boldsymbol{X_g}$ or $\boldsymbol{X_m}$.

is virtually null. Note also that although we can expect a result of zero asymptotically, in finite samples the results at longer horizons are mostly negative due to small sample bias. When looking at $RF(\boldsymbol{X_m})$, we find that this model only has empirical probabilities below conventional significance levels at horizons between one and nine months ahead. However, even though the $RF(\boldsymbol{X_m})$ has a negative R^2_{OoS} at $h = 12$, it is actually better than the median of the placebo test distribution.

VI. Concluding remarks

Employment growth is a leading indicator that has important implications for both policy makers and the private sector. Therefore, the need for accurate and timely predictions is relatively self-evident. In this paper, we show that there is plenty of relevant information about future employment growth in internet search volume. Our findings imply that Google-based forecasting models can be a particularly valuable tool for obtaining accurate real-time information on future employment growth and labor market conditions. We also show that individual Google Trends series do not appear to embed enough information to be better predictors than the classical macroeconomic and financial series. However, the combination of many Google Trends series, preferably in a non-linear manner, can substantially increase the forecasting power and considerably improve upon models based on classical macroeconomic and financial series. This is obtained by targeting predictors using a variable pre-selection procedure such as soft thresholding. Moreover, the Google Trends panel fully encompasses benchmark forecasts and provides significant information in excess of those. Overall, our contributions show that the high predictive power of Google Trends implies that it should be added to the toolbox of practitioners and policy makers interested in forecasting employment growth. Our results also suggest that internet search volume should be further investigated to forecast other macroeconomic variables.

References

ASKITAS, N. AND K. F. ZIMMERMANN (2009): "Google econometrics and unemployment forecasting." *Applied Economics Quarterly*, 55, 107–120.

AYAT, L. AND P. BURRIDGE (2000): "Unit root tests in the presence of uncertainty about the non-stochastic trend." *Journal of Econometrics*, 95, 71–96.

BAI, J. AND S. NG (2008): "Forecasting economic time series using targeted predictors," *Journal of Econometrics*, 146, 304–317.

BIJL, L., G. KRINGHAUG, P. MOLNÁR, AND E. SANDVIK (2016): "Google searches and stock returns," *International Review of Financial Analysis*, 45, 150–156.

BREIMAN, L. (2001): "Random forests," *Machine learning*, 45, 5–32.

BUCHEN, T. AND K. WOHLRABE (2011): "Forecasting with many predictors: Is boosting a viable alternative?" *Economics Letters*, 113, 16–18.

CAMPBELL, J. Y. AND S. B. THOMPSON (2007): "Predicting excess stock returns out of sample: Can anything beat the historical average?" *The Review of Financial Studies*, 21, 1509–1531.

CHOI, H. AND H. VARIAN (2012): "Predicting the present with Google Trends," *Economic Record*, 88, 2–9.

COBLE, D. AND P. PINCHEIRA (2017): "Nowcasting Building Permits with Google Trends." *MPRA Working Paper No. 76514, University Library of Munich, Germany*.

DA, Z., J. ENGELBERG, AND P. GAO (2011): "In search of attention." *Journal of Finance*, 66, 1461–1499.

——— (2014): "The sum of All FEARS: Investor sentiment and asset prices," *Review of Financial Studies*, 28, 1–32.

D'AMURI, F. AND J. MARCUCCI (2017): "The predictive power of Google searches in forecasting US unemployment," *International Journal of Forecasting*, 33, 801–816.

DE MOL, C., D. GIANNONE, AND L. REICHLIN (2008): "Forecasting using a large number of predictors: Is Bayesian shrinkage a valid alternative to principal components?" *Journal of Econometrics*, 146, 318–328.

DIEBOLD, F. X. AND R. S. MARIANO (1995): "Comparing predictive accuracy." *Journal of Business & Economic Statistics*, 13, 253–263.

ELLIOTT, G., A. GARGANO, AND A. TIMMERMANN (2013): "Complete subset regressions." *Journal of Econometrics*, 177, 357–373.

——— (2015): "Complete subset regressions with large-dimensional sets of predictors," *Journal of Economic Dynamics and Control*, 54, 86–110.

FAN, J., J. LV, AND L. QI (2011): "Sparse high-dimensional models in economics," *Annu. Rev. Econ.*, 3, 291–317.

FONDEUR, Y. AND F. KARAMÉ (2013): "Can Google data help predict French youth unemployment?" *Economic Modelling*, 30, 117–125.

FRANCESCO, D. (2009): "Predicting unemployment in short samples with internet job search query data." *MPRA Working Paper No. 18403. University Library of Munich, Germany*.

GENTZKOW, M., B. T. KELLY, AND M. TADDY (2019): "Text as data," *Journal of Economic Literature*, Forthcoming.

GROEN, J. J. AND G. KAPETANIOS (2016): "Revisiting useful approaches to data-rich macroeconomic forecasting." *Computational Statistics & Data Analysis*, 100, 221–239.

HANSEN, P. R., A. LUNDE, AND J. M. NASON (2011): "The model confidence set," *Econometrica*, 79, 453–497.

HARVEY, D. I., S. J. LEYBOURNE, AND P. NEWBOLD (1998): "Tests for forecast encompassing," *Journal of Business & Economic Statistics*, 16, 254–259.

INOUE, A. AND L. KILIAN (2008): "How useful is bagging in forecasting economic time series? A case study of US consumer price inflation," *Journal of the American Statistical Association*, 103, 511–522.

KELLY, B., A. MANELA, AND A. MOREIRA (2018): "Text selection," *Working paper*, 14.

KELLY, B. AND S. PRUITT (2013): "Market expectations in the cross-section of present values," *The Journal of Finance*, 68, 1721–1756.

KIM, H. H. AND N. R. SWANSON (2014): "Forecasting financial and macroeconomic variables using data reduction methods: New empirical evidence," *Journal of Econometrics*, 178, 352–367.

MCCRACKEN, M. W. AND S. NG (2016): "FRED-MD: A monthly database for macroeconomic research," *Journal of Business & Economic Statistics*, 34, 574–589.

NEWEY, W. K. AND K. D. WEST (1987): "A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix." *Econometrica*, 55, 703–708.

POLITIS, D. N. AND J. P. ROMANO (1992): *A circular block-resampling procedure for stationary data.*, Wiley, 263–270.

POLITIS, D. N. AND H. WHITE (2004): "Automatic block-length selection for the dependent bootstrap." *Econometric Reviews*, 23, 53–70.

RAPACH, D. E. AND J. K. STRAUSS (2008): "Forecasting US employment growth using forecast combining methods." *Journal of Forecasting*, 27, 75–93.

——— (2010): "Bagging or combining (or both)? An analysis based on forecasting US employment growth," *Econometric Reviews*, 29, 511–533.

——— (2012): "Forecasting US state-level employment growth: An amalgamation approach," *International Journal of Forecasting*, 28, 315–327.

SMITH, A. (2015): "Searching for work in the digital era." *Pew Research Center: Internet, Science & Tech*, 19.

STOCK, J. H. AND M. W. WATSON (2002a): "Forecasting using principal components from a large number of predictors." *Journal of the American Statistical Association*, 97, 1167–1179.

——— (2002b): "Macroeconomic forecasting using diffusion indexes." *Journal of Business & Economic Statistics*, 20, 147–162.

——— (2006): "Forecasting with many predictors." *In G. Elliott, C. Granger, & A. Timmermann (Eds.) Handbook of economic forecasting*, 1, 515–554.

TIBSHIRANI, R. (1996): "Regression shrinkage and selection via the lasso." *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 58, 267–288.

VOSEN, S. AND T. SCHMIDT (2011): "Forecasting private consumption: Survey-based indicators vs. Google trends," *Journal of Forecasting*, 30, 565–578.

VOZLYUBLENNAIA, N. (2014): "Investor attention, index performance, and return predictability," *Journal of Banking & Finance*, 41, 17–35.

WELCH, I. AND A. GOYAL (2008): "A comprehensive look at the empirical performance of equity premium prediction." *The Review of Financial Studies*, 21, 1455–1508.

YU, L., Y. ZHAO, L. TANG, AND Z. YANG (2019): "Online big data-driven oil consumption forecasting with Google trends," *International Journal of Forecasting*, 35, 213–223.

ZOU, H. AND T. HASTIE (2005): "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67, 301–320.

VII. Tables

Table 1: Out-of-sample predictability for employment growth
This table shows the R^2_{OoS} (in percentage) and p-value from the Diebold-Mariano test (in parenthesis) for all models using a rolling window of 48 observations. Grey shading indicates that the model is included in the 80% model confidence set ($\alpha = 20\%$).

Method (predictors)	$h = 1$	$h = 3$	$h = 6$	$h = 9$	$h = 12$
Panel A: Google Trends predictors, $\boldsymbol{X_g}$					
RF ($\boldsymbol{X_g}$)	26.24	48.73	51.81	56.73	59.15
	(0.006)	(0.014)	(0.034)	(0.046)	(0.076)
Bagging ($\boldsymbol{X_g}$)	16.64	38.60	48.30	39.60	27.41
	(0.098)	(0.075)	(0.060)	(0.105)	(0.203)
CSR ($\boldsymbol{X_g}$)	23.67	47.23	50.49	50.65	42.68
	(0.009)	(0.016)	(0.037)	(0.055)	(0.107)
Panel B: Macroeconomic predictors, $\boldsymbol{X_m}$					
RF ($\boldsymbol{X_m}$)	20.03	38.92	37.74	18.05	−9.57
	(0.011)	(0.035)	(0.130)	(0.339)	(0.563)
Bagging ($\boldsymbol{X_m}$)	22.31	44.44	45.34	−0.13	−65.44
	(0.014)	(0.050)	(0.126)	(0.565)	(0.709)
CSR ($\boldsymbol{X_m}$)	21.23	35.89	35.72	−2.14	−63.22
	(0.011)	(0.031)	(0.113)	(0.515)	(0.730)
Panel C: Autoregressive model					
AR (−)	17.16	9.40	−12.44	−67.41	−108.13
	(0.061)	(0.298)	(0.724)	(0.813)	(0.810)

Table 2: Out-of-sample predictability with combined data set

This table shows the R^2_{OoS} (in percentage) and p-value from the Diebold-Mariano test (in parenthesis) for all models using a rolling window of 48 observations that combines the Google Trends panel $\boldsymbol{X_g}$ and the macroeconomic data set $\boldsymbol{X_m}$ into $\boldsymbol{X_c} = [\boldsymbol{X_g}, \boldsymbol{X_m}]$.

Method (predictors)	$h = 1$	$h = 3$	$h = 6$	$h = 9$	$h = 12$
RF ($\boldsymbol{X_c}$)	27.15	46.18	42.77	33.56	23.37
	(0.001)	(0.020)	(0.099)	(0.191)	(0.315)
Bagging ($\boldsymbol{X_c}$)	21.05	46.84	59.22	35.76	−0.08
	(0.035)	(0.062)	(0.071)	(0.220)	(0.501)
CSR ($\boldsymbol{X_c}$)	29.58	41.82	37.34	29.00	−32.91
	(0.003)	(0.041)	(0.144)	(0.254)	(0.653)

Table 3: Forecast encompassing tests

This table shows estimated weights in (13) according to (16) as well as the p-value (in parenthesis) associated with the $MHLN^h$ test statistic in (15).

Weight	$h=1$	$h=3$	$h=6$	$h=9$	$h=12$
			Panel A: Random forests		
$\hat{\lambda}_g$	0.755	1.127	0.891	1.068	1.074
	(0.000)	(0.000)	(0.051)	(0.094)	(0.096)
$\hat{\lambda}_m$	0.245	−0.127	0.109	−0.068	−0.074
	(0.101)	(0.725)	(0.321)	(0.624)	(0.649)
			Panel B: Bagging		
$\hat{\lambda}_g$	0.399	0.402	0.528	0.701	0.808
	(0.004)	(0.007)	(0.017)	(0.092)	(0.138)
$\hat{\lambda}_m$	0.601	0.598	0.472	0.299	0.192
	(0.000)	(0.000)	(0.016)	(0.044)	(0.134)
			Panel C: Complete subset regressions		
$\hat{\lambda}_g$	0.620	0.948	0.765	0.847	0.952
	(0.013)	(0.001)	(0.021)	(0.076)	(0.117)
$\hat{\lambda}_m$	0.380	0.052	0.235	0.153	0.048
	(0.051)	(0.410)	(0.164)	(0.255)	(0.399)

Table 4: Summary of state-level predictability

This table shows equal- or population-share-weighted average R^2_{OoS} (in percentage) across all 50 US states using a rolling window of 48 observations and the Google Trends panel for state, its neighbouring states and at the national level. The last row reports a pairwise correlation coefficient (in percentage) among R^2_{OoS} at each horizon and the population share. We report in parenthesis below each number the associated two-sided p-value obtained from Fisher-z transformations of estimated correlation coefficients.

	$h=1$	$h=3$	$h=6$	$h=9$	$h=12$
Equal-weighted R^2_{OoS}	21.93	39.46	42.14	43.59	34.48
Population-share-weighted R^2_{OoS}	30.78	46.19	46.24	46.28	36.08
Correlation coefficient	39.73	37.37	22.31	15.24	7.19
	(0.004)	(0.008)	(0.119)	(0.291)	(0.620)

Table 5: State-level number of Google Trends keywords

This table shows the number of keywords for each state coming from its own Google Trends panel (the column labelled "State only"), including those from its neighbouring states' Google Trends panels (the column labelled "Incl. neighbours") and including the national Google Trends panel (the column labelled "Total").

	Number of keywords				Number of keywords		
State	State only	Incl. neighbours	Total	State	State only	Incl. neighbours	Total
AK	30	30	203	MT	33	141	314
AL	61	471	644	NC	140	568	741
AR	30	575	748	ND	24	184	357
AZ	90	546	719	NE	54	441	614
CA	166	424	597	NH	36	208	381
CO	113	513	686	NJ	124	458	631
CT	73	421	594	NM	42	532	705
DE	21	385	558	NV	66	538	711
FL	141	332	505	NY	189	651	824
GA	130	642	815	OH	125	591	764
HI	52	52	225	OK	67	575	748
IA	63	601	774	OR	102	491	664
ID	45	444	617	PA	124	734	907
IL	145	571	744	RI	37	232	405
IN	82	570	750	SC	76	346	519
KS	61	406	579	SD	22	318	491
KY	69	789	962	TN	94	808	981
LA	77	303	476	TX	151	367	540
MA	122	476	649	UT	69	442	615
MD	116	424	597	VA	128	582	755
ME	31	67	240	VT	19	366	539
MI	156	464	637	WA	112	259	432
MN	105	315	488	WI	101	570	743
MO	111	694	867	WW	35	597	770
MS	45	307	480	WY	17	353	526

Table 6: Out-of-sample predictability with alternative keywords
This table shows the R^2_{OoS} (in percentage) and p-value from the Diebold-Mariano test (in parenthesis) for the random forests using a rolling window of 48 observations and alternative keywords for generating the Google Trends panel.

Keyword	$h=1$	$h=3$	$h=6$	$h=9$	$h=12$
Employment	24.43	45.39	52.70	55.99	56.16
	(0.007)	(0.020)	(0.040)	(0.059)	(0.095)
Unemployment	26.43	38.10	27.44	16.21	11.99
	(0.003)	(0.027)	(0.074)	(0.230)	(0.341)

Table 7:
Out-of-sample predictability with alternative estimation windows
This table shows the R^2_{OoS} (in percentage) and p-value from the Diebold-Mariano test (in parenthesis) for the random forests with $\boldsymbol{X_g}$ using alternative estimation windows.

Method (predictors)	$h=1$	$h=3$	$h=6$	$h=9$	$h=12$
Panel A: Rolling window of 36 observations					
RF ($\boldsymbol{X_g}$)	21.51	42.75	48.10	44.79	35.35
	(0.004)	(0.016)	(0.043)	(0.049)	(0.062)
Panel B: Rolling window of 60 observations					
RF ($\boldsymbol{X_g}$)	17.28	40.82	57.74	57.85	54.19
	(0.040)	(0.020)	(0.016)	(0.066)	(0.097)
Panel C: Expanding window with initial size of 48 observations					
RF ($\boldsymbol{X_g}$)	20.83	40.34	40.94	39.21	31.85
	(0.024)	(0.033)	(0.088)	(0.109)	(0.073)

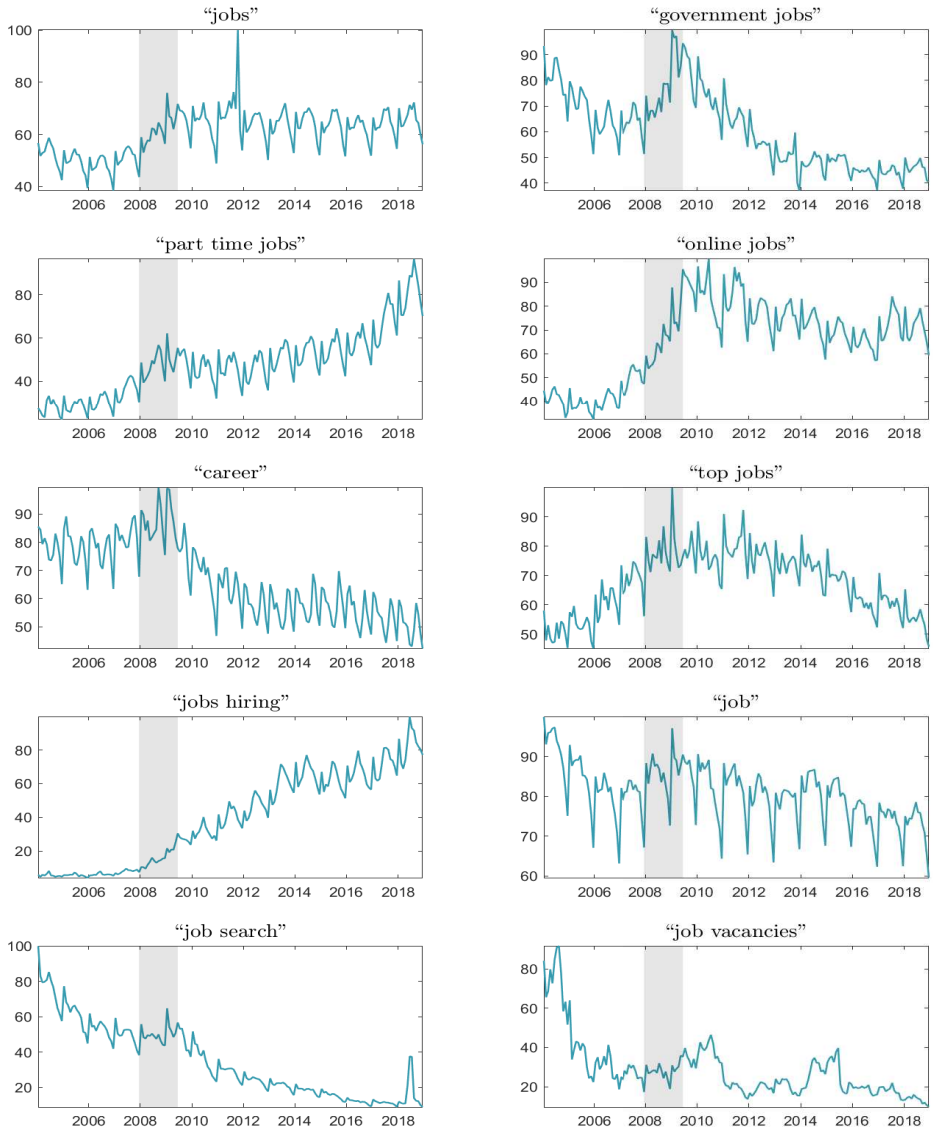
Table 8: Distribution of placebo R^2_{OoS}

The table shows a summary of the place distribution of R^2_{OoS} for the random forests using Google Trends and macroeconomic data. The placebo data $\boldsymbol{X}_{\text{placebo}}$ is constructed to have the same mean, variance and AR(1) coefficient as the series selected by soft thresholding but with no true predictive power for employment growth. The models are estimated using a rolling window of 48 observations. We conduct 10,000 placebo replications.

	$h=1$	$h=3$	$h=6$	$h=9$	$h=12$
Panel A: Random forests with Google Trends, $\boldsymbol{X_g}$					
Median	6.85	11.55	2.98	−3.14	−11.42
95th percentile	14.19	20.37	12.67	9.31	4.83
99th percentile	17.35	23.73	16.48	14.36	11.09
Minimum	−13.04	−10.33	−23.98	−34.13	−51.45
Maximum	23.32	29.72	29.96	23.66	25.44
Panel B: Random forests with macroeconomic data, $\boldsymbol{X_m}$					
Median	8.50	11.92	8.93	0.30	−13.55
95th percentile	15.45	21.48	17.92	13.20	3.31
99th percentile	18.92	24.44	21.65	18.00	10.57
Minimum	−8.93	−8.86	−11.45	−40.76	−55.44
Maximum	24.72	27.91	27.05	27.22	24.89

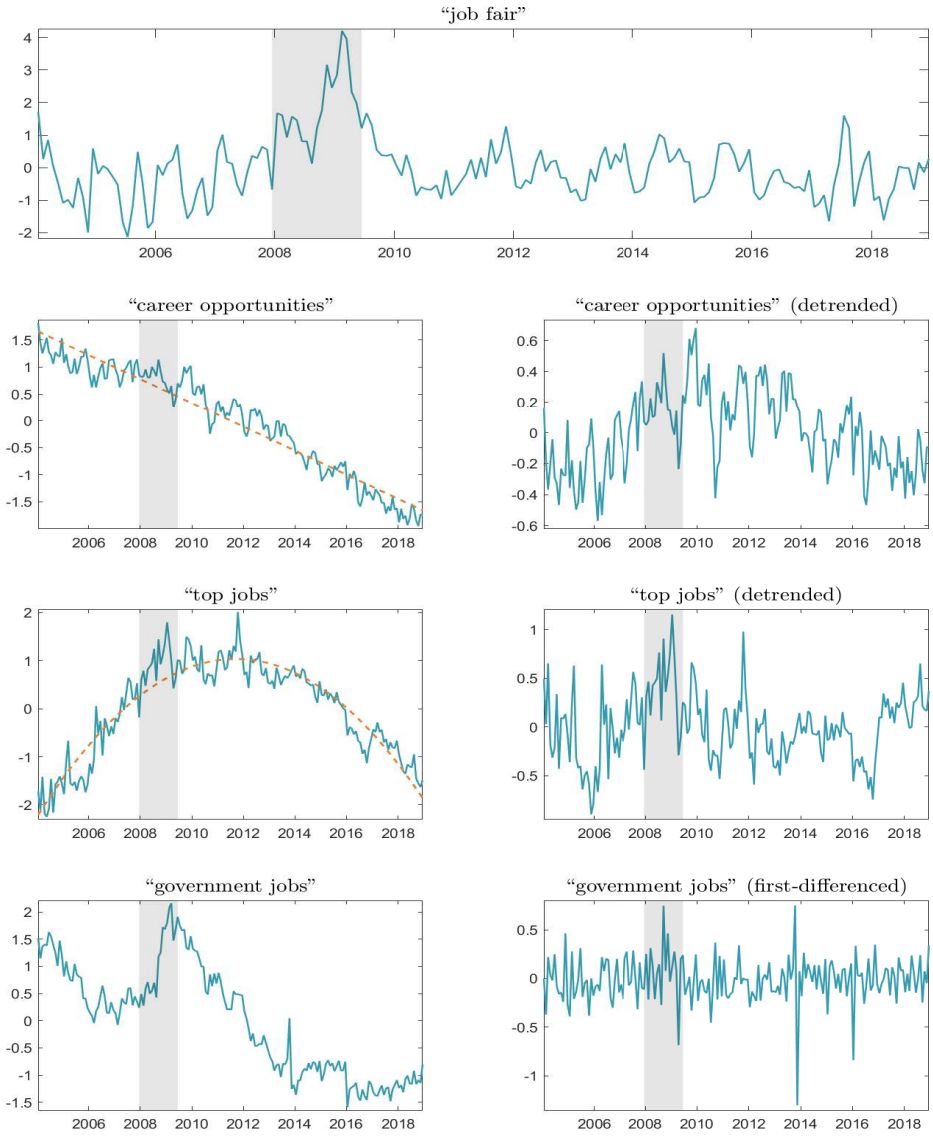
VIII. Figures

Figure 1: Primitive queries



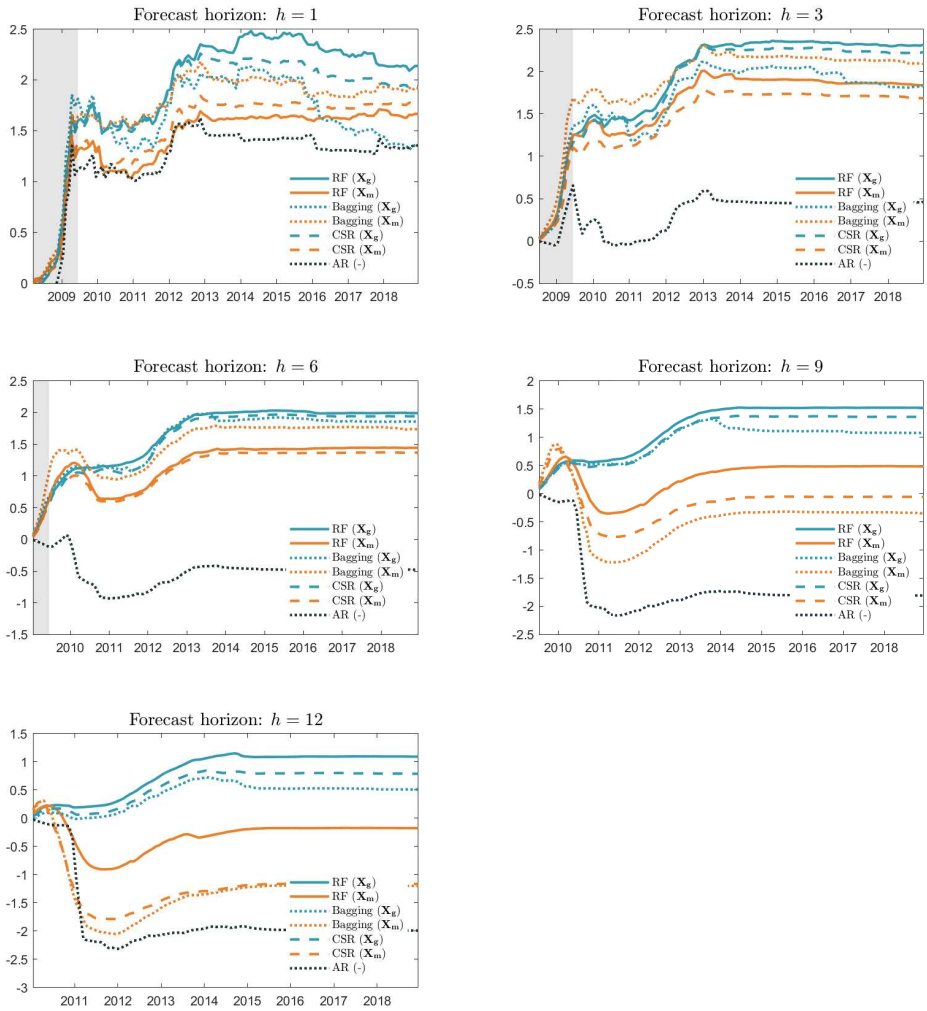
This figure shows the ten primitive Google Trends queries in the period 2004:M1-2018:M12. The index is calculated as a simple of average of the index for each word over twenty different days. Grey shaded areas indicate NBER recessions.

Figure 2: Data transformation to construct X_g



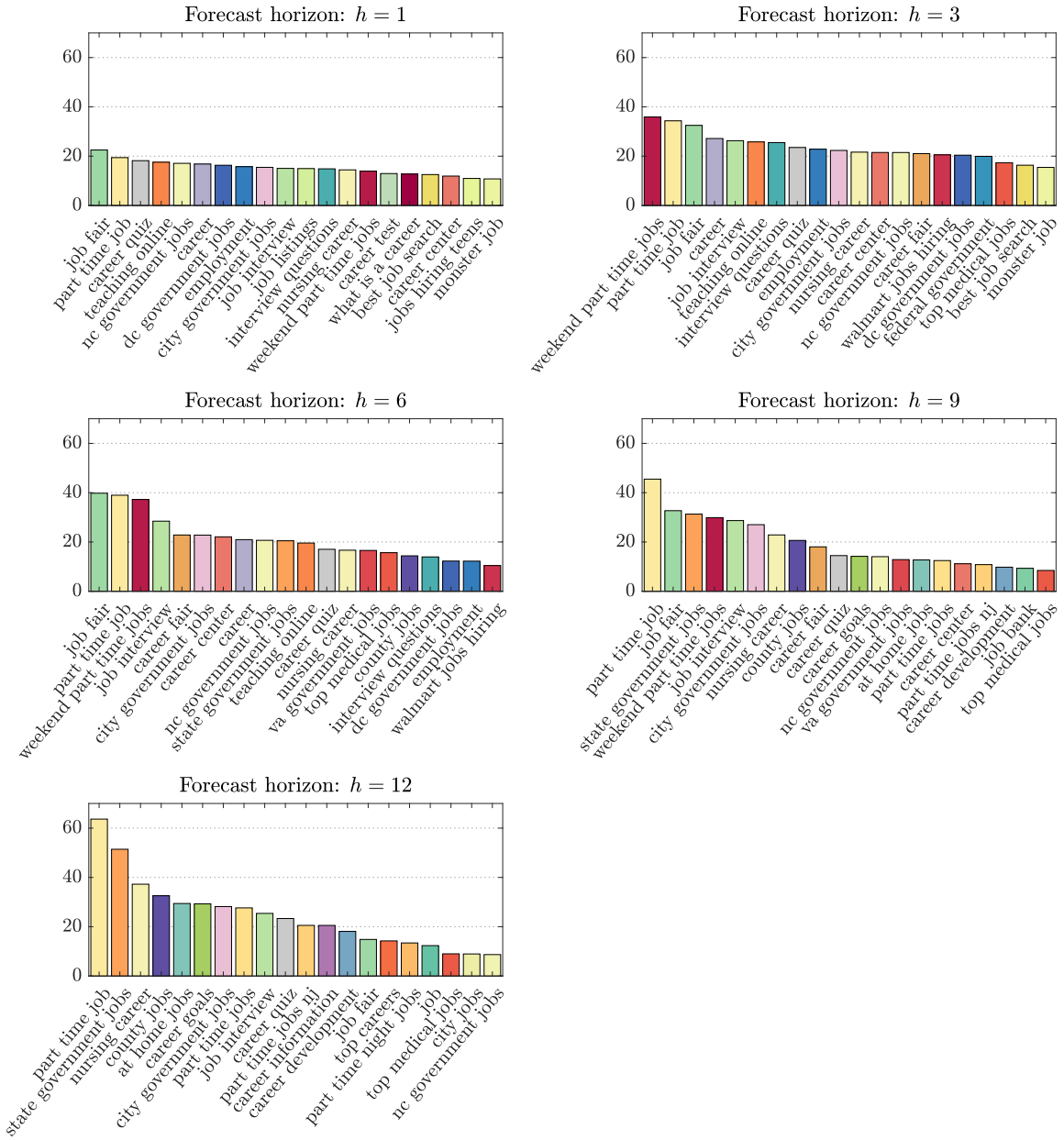
This figure shows the natural logarithm of four deseasonalized Google Trends queries in the period 2004:M1-2018:M12. The panel on top shows an example of a stationary query: "*job fair*", for which we do not perform any detrending or diferencing. The second panel on the left shows a linear trend-stationary query "*career opportunities*" (solid line) and its linear trend estimate (orange dashed line) while the panel on the right shows deviations from this trend. The third panel on the left shows shows a quadratic trend stationary query "*top jobs*" (solid line) and its trend estimate (orange dashed line) while the panel on the right shows deviations from this trend. The panel at the bottom right shows a series for which we could not reject the null of a unit root "*government jobs*" and the panel on the bottom right shows the same series in differences. For ease of comparison the series have been standardized to have mean zero and standard deviation of one. Grey shaded areas indicate NBER recessions.

Figure 3: Cumulative sum of squared error difference (CSSED)



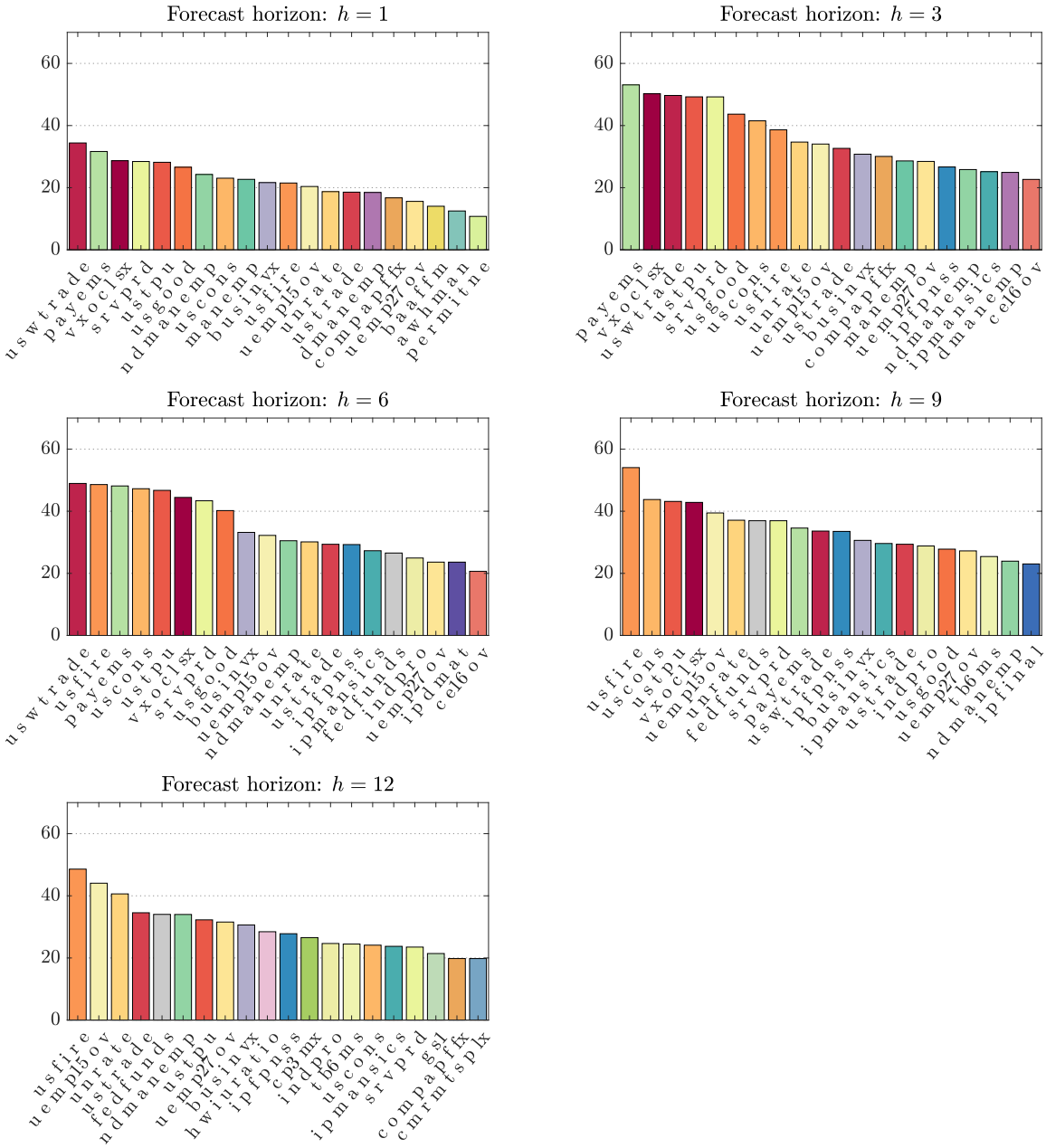
This figure shows the cumulative sum of squared errors difference ($CSSED_{p,t}$) as per (11) for all models using a rolling window of 48 observations. Grey shaded areas indicate NBER recessions.

Figure 4: R^2_{OoS} for the best 20 individual predictors in X_g



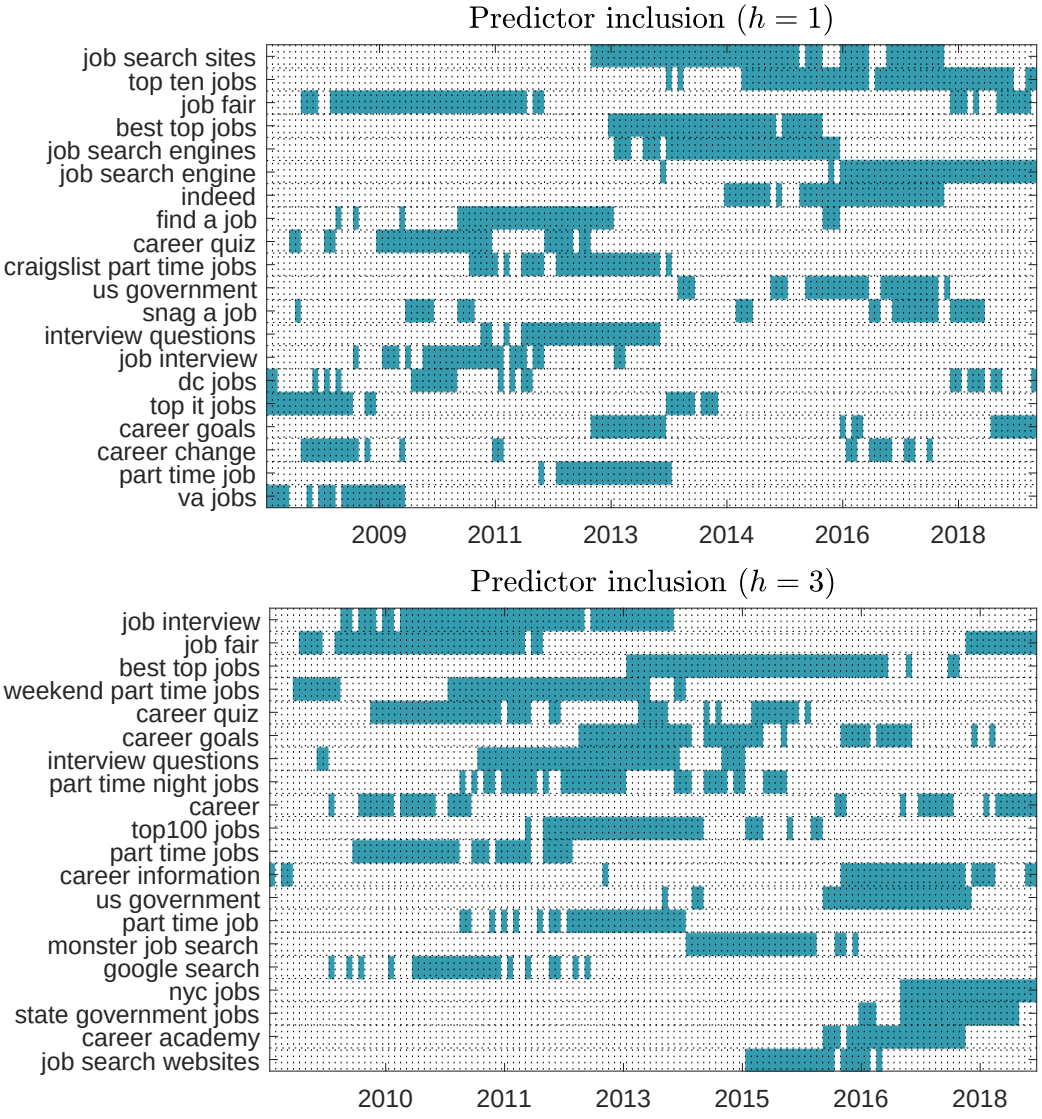
This figure shows the R^2_{OoS} for univariate forecasts, $\hat{y}^h_{t+h} = \hat{\alpha} + \hat{\beta} X_{i,t}$, for the top 20 predictors in the Google Trends panel $\boldsymbol{X_g}$, using a rolling window of 48 observations.

Figure 5: R^2_{OoS} for the best 20 individual predictors in X_m



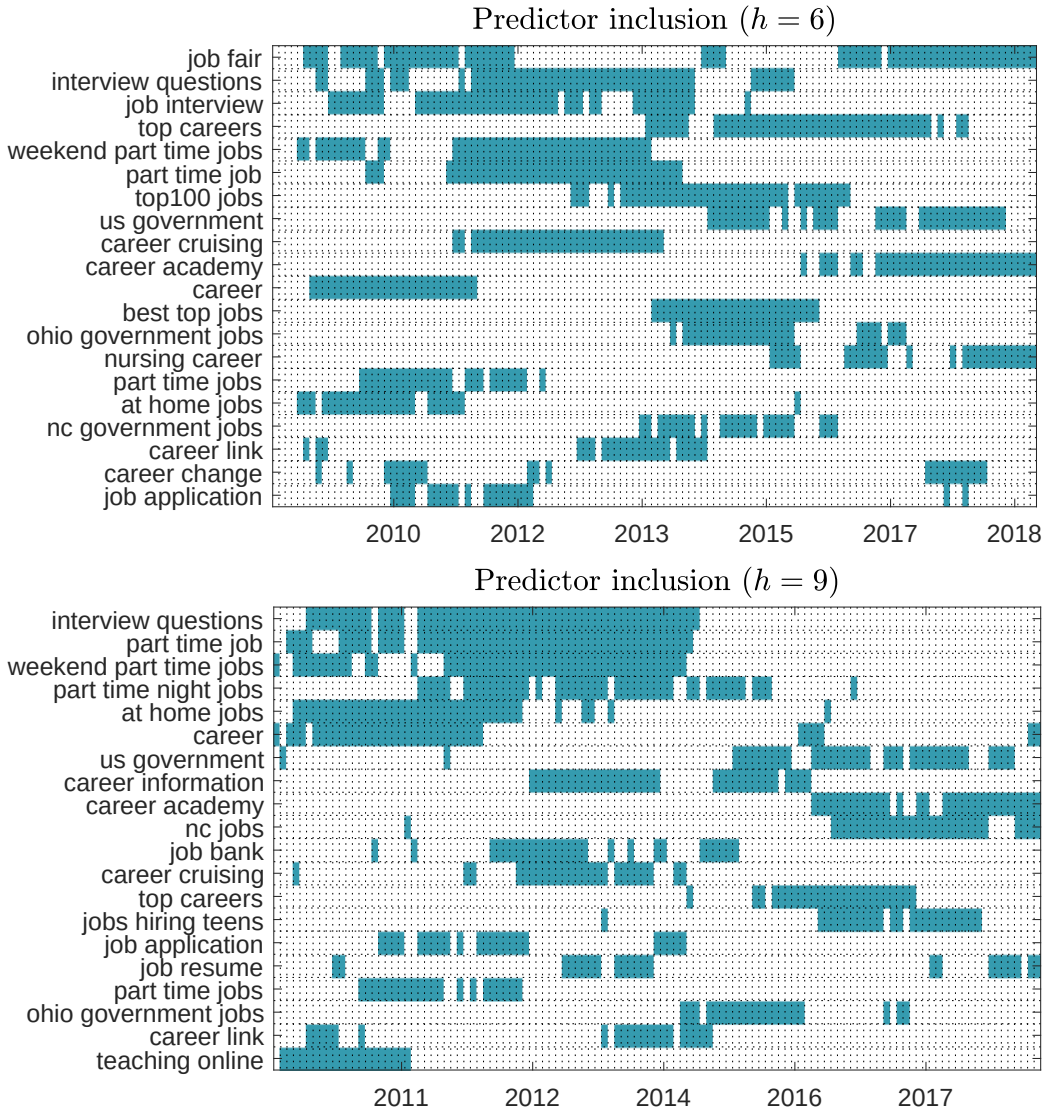
This figure shows the R^2_{OoS} for univariate forecasts, $\hat{y}^h_{t+h} = \hat{\alpha} + \hat{\beta} X_{i,t}$, for the top 20 predictors in the macroeconomic panel $\boldsymbol{X_m}$, using a rolling window of 48 observations.

Figure 6: Soft thresholding inclusion for individual predictors in X_g



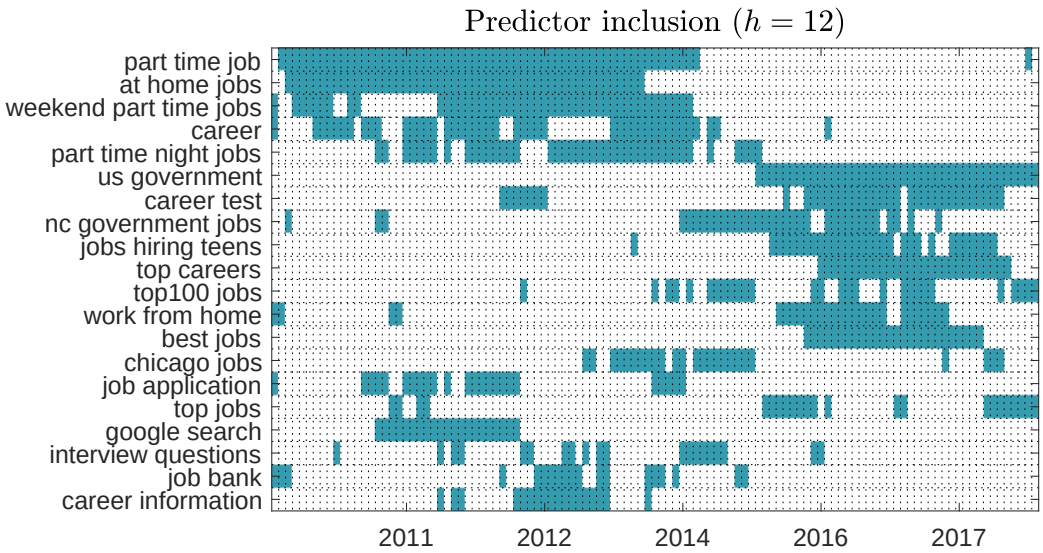
This figure shows the soft thresholding inclusion for the predictors in $\boldsymbol{X_g}$ during the out-of-sample evaluation period for the top 20 predictors. The predictors are ordered from top to bottom according to their inclusion frequency.

Figure 6 (Cont.): Soft thresholding inclusion for individual predictors in X_g



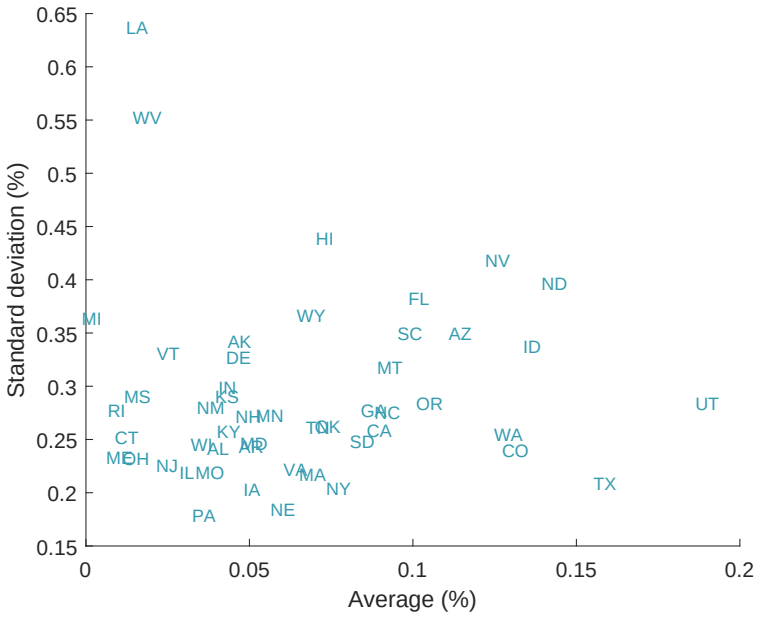
This figure shows the soft thresholding inclusion for the predictors in $\boldsymbol{X_g}$ during the out-of-sample evaluation period for the top 20 predictors. The predictors are ordered from top to bottom according to their inclusion frequency.

Figure 6 (Cont.): Soft thresholding inclusion for individual predictors in X_g



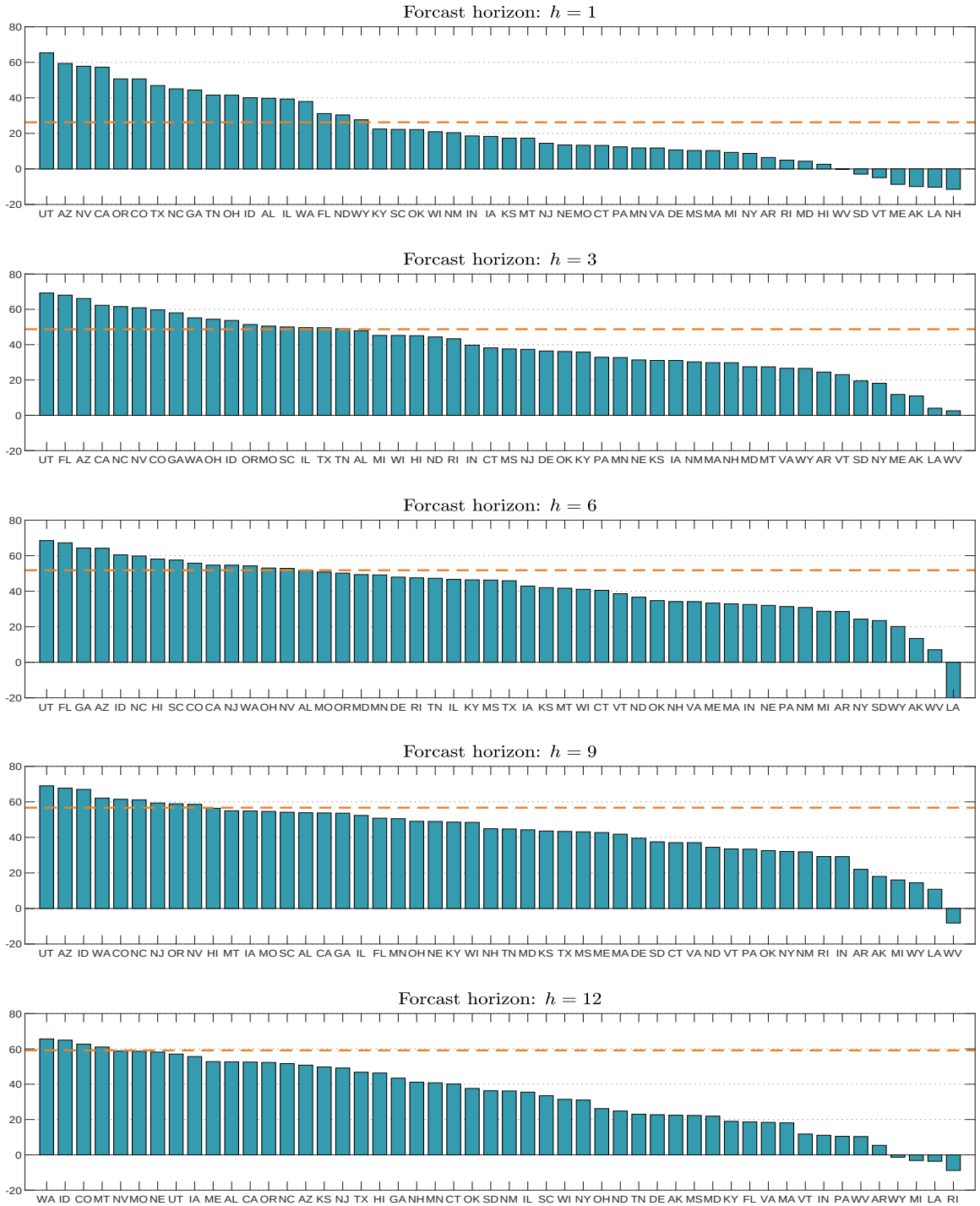
This figure shows the soft thresholding inclusion for the predictors in X_g during the out-of-sample evaluation period for the top 20 predictors. The predictors are ordered from top to bottom according to their inclusion frequency.

Figure 7: State-level employment growth



This figure shows a scatterplot of the state-level standard deviation (y-axis) and average (x-axis), in percentages, one-period employment growth over the 2004:M1 to 2018:M12 period.

Figure 8: State-level out-of-sample predictability for employment growth



This figure shows R^2_{OoS} (in percentage) for all 50 US states using a rolling window of 48 observations of the Google Trends panel for the state, its neighbouring states and at the national level. For each forecast horizon, the states are sorted in descending order according to their predictive ability. The orange dashed line indicates the R^2_{OoS} from predicting national level employment growth using the Google Trends Panel as reported in Table 1.

Research Papers 2019

2018-34: James G. MacKinnon, Morten Ørregaard Nielsen, David Roodman and Matthew D. Webb: Fast and Wild: Bootstrap Inference in Stata Using boottest

2018-35: Sepideh Dolatabadim, Paresh Kumar Narayan, Morten Ørregaard Nielsen and Ke Xu: Economic significance of commodity return forecasts from the fractionally cointegrated VAR model

2018-36: Charlotte Christiansen, Niels S. Grønborg and Ole L. Nielsen: Mutual Fund Selection for Realistically Short Samples

2018-37: Niels S. Grønborg, Asger Lunde, Kasper V. Olesen and Harry Vander Elst: Realizing Correlations Across Asset Classes

2018-38: Riccardo Borghi, Eric Hillebrand, Jakob Mikkelsen and Giovanni Urga: The dynamics of factor loadings in the cross-section of returns

2019-01: Andrea Gatto and Francesco Busato: Defining, measuring and ranking energy vulnerability

2019-02: Federico Carlini and Paolo Santucci de Magistris: Resuscitating the co-fractional model of Granger (1986)

2019-03: Martin M. Andreasen and Mads Dang: Estimating the Price Markup in the New Keynesian Model

2019-04: Daniel Borup, Bent Jesper Christensen and Yunus Emre Ergemen: Assessing predictive accuracy in panel data models with long-range dependence

2019-05: Antoine A. Djogbenou, James G. MacKinnon and Morten Ørregaard Nielsen: Asymptotic Theory and Wild Bootstrap Inference with Clustered Errors

2019-06: Vanessa Berenguer-Rico, Søren Johansen and Bent Nielsen: The analysis of marked and weighted empirical processes of estimated residuals

2019-07: Søren Kjærgaard, Yunus Emre Ergemen, Kallestrup-Lamb, Jim Oeppen and Rune Lindahl-Jacobsen: Forecasting Causes of Death using Compositional Data Analysis: the Case of Cancer Deaths

2019-08: Søren Kjærgaard, Yunus Emre Ergemen, Marie-Pier Bergeron Boucher, Jim Oeppen and Malene Kallestrup-Lamb: Longevity forecasting by socio-economic groups using compositional data analysis

2019-09: Debopam Bhattacharya, Pascaline Dupas and Shin Kanaya: Demand and Welfare Analysis in Discrete Choice Models with Social Interactions

2019-10: Martin Møller Andreasen, Kasper Jørgensen and Andrew Meldrum: Bond Risk Premiums at the Zero Lower Bound

2019-11: Martin Møller Andrasen: Explaining Bond Return Predictability in an Estimated New Keynesian Model

2019-12: Vanessa Berenguer-Rico, Søren Johansen and Bent Nielsen: Uniform Consistency of Marked and Weighted Empirical Distributions of Residuals

2019-13: Daniel Borup and Erik Christian Montes Schütte: In search of a job: Forecasting employment growth using Google Trends