The London School of Economics and Political Science

Essays in economics and machine learning

Friedrich Christian Geiecke

A thesis submitted to the Department of Economics of the London School of Economics for the degree of Doctor of Philosophy, London, September 2019

Declaration

I certify that the thesis I have presented for examination for the PhD degree of the London School of Economics and Political Science is solely my own work other than where I have clearly indicated that it is the work of others (in which case the extent of any work carried out jointly by me and any other person is clearly identified in it).

I declare that my thesis consists of approximately 28,000 words.

Statement of conjoint work

I confirm that Chapter 1 was jointly co-authored with Adrien Bussy and I contributed 50% of this work.

I confirm that Chapter 2 was jointly co-authored with Karun Adusumilli and Claudio Schilter and I contributed 33% of this Chapter. It is the empirical application with additional explanations of the more extensive and theoretical paper Adusumilli, Geiecke, Schilter (2019).

Statement of inclusion of previous work

I can confirm that a version of Chapter 3 was used in 2016 for progression to PhD.

Abstract

This thesis studies questions in innovation, optimal policy, and aggregate fluctuations, partially with the help of methods from machine learning and artificial intelligence.

Chapter 1 is concerned with innovation in patents. With tools from natural language processing, we represent around 4.6 million patents as high dimensional numerical vectors and find a rich underlying geometry. We measure economy wide and field specific trends and detect patents which anticipated such trends or widened existing ideas. These patents have on average higher citations and their firms tend to make higher profits.

Chapter 2 discusses an application of reinforcement learning to study outcomes from causal experiments. We model individuals who lost their jobs and arrive sequentially at a policy maker's office to register as unemployed. After paying a cost to provide job training to an individual, the policy maker observes an individual treatment effect estimate which we obtain from RCT data. Due to a limited budget, she cannot provide training to all individuals. We use reinforcement learning to solve for the optimal policy in this dynamic problem.

Chapter 3 turns to the analysis of macroeconomic fluctuations. It introduces a mechanism through which perpetual cycles in aggregate output can result endogenously. Individuals share sentiments similarly to diseases in models of disease transmission. Consumption of optimistic consumers is biased upwards and consumption of pessimistic consumers downwards. In a behavioural New Keynesian model, recurring waves of optimism and pessimism lead to cyclical aggregate output as an inherent feature of this economy.

Chapter 4 concludes with a brief empirical investigation of newspaper sentiments and their co-movement with aggregate variables. Here the focus is not on contagion, but on the measurement of historical business and economic sentiment since around 1850. Using the archive of the New York Times, I build an indicator and discuss its properties.

Acknowledgements

When I first enrolled into university to study business administration, I did not know what exactly economics was. During undergraduate studies I was lucky to cross the path of Mark Trede and later during master's studies that of Tony Atkinson. Without their infectious enthusiasm, support, and time I would not have done this PhD programme. I am extremely grateful to them.

My time during the PhD at the LSE was strongly influenced by my supervisor Wouter den Haan and advisor Tim Besley and I owe a great debt of gratitude to them. Wouter allowed me large freedom to study and work on the topics I was truly interested in, which was one of the principal reasons this journey has been so enjoyable. The many discussions with him taught me much about reasoning in research and about which conclusions I could draw and defend and which ones not. Tim's deep insight into the social sciences has been invaluable and I will try to remember his note that, rather than getting lost in technicalities, the best papers usually answer a precise question and leave the reader with a clear feeling of having learned something. I would also like to thank Ricardo Reis for the very insightful conversations about macroeconomics and this thesis, the faculty at the CfM and LSE for research seminars and discussions, and the team of the LSE's high-performance computing service. Furthermore, I am thankful for scholarships from the Economic and Social Research Council during the MRes/PhD and from the Foundation of German Business earlier in my studies.

The daily life as a PhD student was most shaped by my peers. I have learned and am learning an incredible amount through the discussions with them, through the work on joint projects, and I am grateful for their friendship. The time would not have been the same without Karun Adusumilli, Andrea Alati, Miguel Bandeira, Andres Barrios Fernandez, Matthias Breuer, Adrien Bussy, Laura Castillo-Martínez, Weihan Ding, Thomas Drechsel, Dita Eckardt, Jesús Gorrín, Felix König, Lukas Kremens, Jonathan Pinder, Bernardo Ricca, Wolfgang Ridinger, Claudio Schilter, and all the other students in the LSE PhD programmes.

Mathieu Frelet, Tobias Odendahl, and Nina Tholuck have helped me with much advice over the last years and I am glad we continued to stay in close contact despite the increased distance. Thank you for being such great friends.

Most of all I would like to thank my parents, brother, grandparents, and wider family. Particularly my parents and grandparents have greatly supported my education for as long as I can remember and none of this would have been possible without them. This thesis is dedicated to them.

Contents

List of Tables

List of Figures

Chapter 1

A Geometry of Innovation

1.1 Introduction

In this paper we represent millions of US patents as numerical vectors in high dimensional space and find a rich underlying geometry. We explore this geometry of innovation to think about three economic questions through one common approach. First, can we identify macro and field-specific trends in innovation and are patents anticipating such trends more successful? Second, can we use the space of patents to define and identify patents that seem to *widen* the knowledge base by venturing into new areas? Third, can the spatial representation of patent content be helpful to think about patents that relate to diverse technological fields, akin to general purpose technologies?

Using methods from machine learning and natural language processing (NLP) we transform each patent's text into a numerical vector representation, also called embedding. The structures in this space of vector representations allow us to characterise the technological landscape. Patents of similar textual content are close or similar to each other in vector space. Armed with these vector representations, we can identify trends in innovation at the macro and technological field levels. Specifically, we analyse the similarity of a patent's content to existing patents and topics (i) at the time when it is filed, and (ii) 10 years later. We then test whether patents dissimilar to existing patents when they are filed yet similar to subsequent patents 10 years later are successful innovations, where success is measured in terms of citations, private economic value and effects on firm-level measures of output and profitability. After a refinement of our patent score, we call patents whose content is novel at the time of filing and popular subsequently *widening* patents. This label appeals to the intuitive sense that such patents

widened the scope of knowledge by successfully venturing into unexplored areas of the space. Last, we outline a method to measure heterogeneity in content of neighbouring patents closest to a given patent. We discuss these findings in the light of General Purpose Technologies (GPT), yet, conclude that our concept of neighbourhood heterogeneity is not able to identify GTPs.

Data and methodology Our data consists of approximately 4.6 million utility patents granted by the United States Patent and Trademark Office (USPTO) over the years 1975-2017, a subset of which is linked to firms from the Compustat database. After building a document term matrix from their texts, we employ truncated singular value decomposition on it (also often referred to as LSA, Deerwester et al., 1990) to obtain vector representations better manageable in terms of computing power. This allows us to compute three key metrics based on the location of each patent within the technological space over time. The first score is based on what we name *centroid patents*, which are imaginary *mean* patents in a given technological space. Specifically, a centroid score is computed for each patent by taking a ratio of its distance to the centroid 10 years after the filing date and that to the centroid at the time of filing. A high score implies that a patent was dissimilar to the mean patent when filed, yet similar to subsequent patents. It anticipated a trend in the economy or a specific industry. The second metric — that we call no-centroid score — is a refinement of the idea above, where the similarity to centroids is replaced by the average similarity to a patent's closest neighbours. The third metric, forward neighbourhood heterogeneity, is a measure of how diverse — in terms of technology fields — close neighbours of a patents are, 10 years after it has been filed. To visualise the spatial geometry of patents, we additionally use a recent t-Distributed Stochastic Neighbor Embedding (t-SNE) variant by Linderman et al. (2019) from RNA sequencing visualisation in genetics (see also Maaten and Hinton, 2008, for the first paper on this method). t-SNE is a tool for visualisation which preserves structures from high dimensional space also in lower dimensional representations, e.g. in 2D or 3D. Applied to our patent representations, it allows us to see clusters and neighborhoods that relate to different technological fields and it visually guides our subsequent analysis.

Findings In our analysis, we find that patents which anticipated economy-wide and field-specific shifts are cited more, and have higher private economic value. At the firm level, firms which were granted patents that anticipated shifts in their field tend to grow faster and to be more profitable, although our analysis does not allow us to claim any causal links between the specific patent we identify and

firm outcomes. However, we show that it is possible to identify innovative firms, which also seem to make higher profits, based purely on the language content of patents. Due to the persistence of R&D quality in firms, these firms also have been performing better than their peers some years before we identified some of their patents.

Patents which we identify as *widening* innovations also tend to be cited more and have a higher private value. Similarly to patents anticipating more systemic shifts, firms that patent these inventions tend to have higher growth in profits and output, as well as capital. Again, firms that we identify as innovative based on their patents' spatial properties also grew faster than other firms a few years prior to filing the identified patents. It it reassuring to see that the inventions our methods pick among close to 3 million patents are the output of fast-growing, successful firms. We also analyse how heterogeneous the neighbourhood of a patent is in space and find that patents which have neighbourhoods consisting of patents from many different International Patent Classification (IPC) codes have significantly lower citations than more specialised patents.

In a subsequent discussion, we argue that methods like ours largely pick up the evolution of technology over the past 30 years and in particular the information technology (IT) revolution, which we know has significantly shaped the economy. The IT revolution is identified by two of the scores: It resulted in an economy-wide and field-specific shift in the content of innovation, and early IT patents in the 1990s appeared in a completely empty part of our representation of the technological space. We find that IT is responsible for a considerable share of the results we present in this paper, yet the results qualitatively survive in a sub-sample dropping the most IT-related fields.

Relation to the literature This paper relates to several strands of the literature. First, it contributes to the extensive literature that applies NLP methods to patents to characterise technological progress. Balsmeier et al. (2018) and Packalen and Bhattacharya (2015) identify novel inventions based on the first appearance on words in the patent corpus. Bowen, Frésard and Hoberg (2018) construct a measure of ex-ante technological disruptive potential of patents based on their use of new or fast-growing words across contemporaneous patent applications. The closest paper to the present work is a very recent working paper by Kelly, Papanikolaou, Seru and Taddy (2018), who develop a conceptually identical method to ours to identify significant inventions, based on their similarity to previous and subsequent patents. Our paper's no-centroid scores which we

use to identify widening patents are almost the same as their score, however, we develop the notion of a centroid patent and introduce scores based on centroids or on another concept we call forward neighbourhood heterogeneity. Centroids also allow us to visualise patent content and trends in words usage. In general, our approach is a very visual one and we are able to discuss many findings graphically through different dimension reduction techniques such Linderman et al. (2019) or through word frequencies.

This work also relates to the literature on innovation and economic growth. First, it contributes to attempts at evaluating the economic importance of inventions. This is a difficult task only imperfectly measured by citations, which tend to reflect scientific importance — which is however related to economic value (Hall, Jaffe and Trajtenberg, 2005; Nicholas, 2008). Kogan, Papanikolaou, Seru and Stoffman (2017) provide a measure of private economic value based on the stock market response to news about patent publications.[1] Our measure of the importance of patents is correlated to both citations and private economic values, but provides additional insights as to what features of inventions create value.

First, our concept of *widening* inventions relates to the notion that an innovation can deepen the specialised knowledge of a given narrowly defined technological field, or widen the scope of knowledge by creating a yet nonexistent technology. This notion is related to the Schumpeterian patterns of innovation, whereby innovation by new firms has been called *widening* and innovation by existing innovators is referred to as *deepening* (Breschi, Malerba and Orsenigo, 2000; Malerba and Orsenigo, 1995). It also relates to endogenous growth models in the tradition of Klette and Kortum (2004), where innovations can either replace existing products or create new ones. Our measure of *widening* patents tentatively provides a tangible quantification of the concepts underlying these theories.[2] Second, by looking at the distribution of technological fields of the patents most closely related to an invention, our work relates to the identification of GPTs, landmark technologies that are pervasive — they are used across most sectors — and spawn innovation, i.e. trigger further innovations (Bresnahan, 2010; Helpman, 1998). Examples of notable GPTs include the steam engine (Crafts, 2004; Rosenberg and Trajtenberg, 2004) and electricity and computers (Jovanovic and Rousseau, 2005).[3]

[1] This measure is extended by Kline et al. (2019) to a larger sample of US firms.

[2] Other work measuring the technological breadth and depth of sectors, firms or patents include Katila and Ahuja (2002); Ozman (2007); Moorthy and Polley (2010); Lodh and Battaggion (2014), who base their measures on the diversity and intensity with which technological sub-fields are related to each other. Our measure is purely based on the text of patents.

[3] The printer may also have been a GPT, although it is difficult to assess given how early it was invented (Dittmar, 2011).

Structure of the paper The paper is structured as follows. Section 1.2 describes the data. Details of the methodology are given in section 1.3. Section 1.4 presents the results, followed by a discussion in Section 1.5. Section 1.6 concludes. Additional tables and figures can be found in the Appendix.

1.2 Data

This section contains details on the data used in this project. Data on patents is from the United States Patent and Trademark Office (USPTO). Firm-level data is from the CRSP/Compustat Merged Database, and data on the economic importance of patents is from Kogan et al. (2017).

1.2.1 Patents

The USPTO provides data on all patents that have been granted in the US, both by US and foreign entities. We restrict the sample to utility patents granted by the USPTO over the years 1976-2017, for which detailed information in machine-readable format is available. There are some cases where two or more patents have the exact same abstract. This happens when an original patent is followed by continuing applications that either change claims or focus on a subset of claims from the original application.[4] Since these applications refer to the same invention, we only keep the patent that was first granted.

For each patent, the following information is available: the full texts of the abstract, the description of the invention and the claims; the filing and grant dates; the IPC classification, which indicates to which areas of technology a patent pertains; information on the filing and beneficiary entities (which can differ); and data on forward citations by subsequent patents. The final sample consists of about 4.6 million patents granted over the years 1976-2017. Out of these, a subset of around 1.2 million patents in years 1976-2010 is matched to firms from Compustat, using the matches publicly provided by Kogan et al. (2017).[5] An estimate of the economic value as calculated by Kogan et al. (2017) is also available for most of these patents.

The final sample of patents for which a score can be computed consists of around 2.8 million patents over the years 1985-2008 — as the score can only be calculated

[4]See section 201 of the USPTO manual, for exact definitions.
[5]The data can be found here. Last accessed on 30/07/2019.

Table 1.1: Descriptive statistics of patent citations

	N	Mean	Sd	Min	p25	Median	p75	p90	Max
5-year citations	4,633,305	3	7.1	0	0	1	3	7	1431
10-year citations	4,633,305	6.4	16	0	0	2	6	15	2801
15-year citations	4,633,305	8.9	24	0	0	3	8	21	2801
Citations as of 2017	4,633,305	12	32	0	0	3	11	28	3997

Note: Descriptive statistics of the number of citations of patents at different horizons after their grant dates.

for patents with 10 years of data before and after their filing date, see Section 1.3 for details on the methodology. Out of these 2.8 million patents, around 1 million can be matched to Compustat firms. Figure 1.17 plots the number of patents filed and granted in the sample over time, and the coverage in terms of scores and linkage to firms. In Panel (a), note that the number of filed patents decreases dramatically in the later years. This is because the data only contains patents that have been granted and most patents filed in recent years have not been granted yet. On average, a patent is granted 2.5 years after it was filed. In the whole sample, patents are cited 9 times on average in the 10 years following their publication. The distributions of forward citations is skewed to the left: 50% of patents get 2 or less citations over 10 years — descriptive statistics of the number of citations at different time horizons can be found in Table 1.1.

1.2.2 Firms

Firm-level data comes from the Compustat database. The sample covers listed firms in the US and provides financial and accounting data at the firm level. Following Kogan et al. (2017), we restrict the sample to observations where values of SIC (*Standard industrial classification of economic activities*) codes and book assets are not missing. We omit industries that never patent, as well firms from the financial and utilities sectors (SIC codes 6000 to 6799 and 1900 to 1949, respectively). The final sample consists of around 120,000 firm-year observations over the years 1985-2008. Around 26% of firm-year observations filed at least one patent over the sample period. The variables of interest at the firm level are profits — defined as sales minus costs of goods solds — output — defined as sales plus inventories — employment and capital (property, plants and equipment). All variables are expressed in real terms — profits and output are deflated using the CPI, whereas capital is deflated using the equipment implicit price deflator from the National Income and Product Accounts (NIPA). All firm-level variables are winsorised at the 1% percent level yearly.

Table 1.2: Descriptive statistics of the firm sample

	N	Mean	Sd	Min	p25	Median	p75	p90	Max
# patents granted	119,836	8.3	77	0	0	0	1	5	4207
# patents filed	119,836	9.1	86	0	0	0	1	6	4422
1-y gr. rate: capital	107,436	.13	.46	-12	.024	.097	.22	.45	10
1-y gr. rate: profits	98,192	.072	.5	-9.7	-.083	.061	.22	.48	7.8
1-y gr. rate: output	106,319	.082	.51	-10	-.06	.056	.2	.44	9.4
1-y gr. rate: employment	102,799	.05	.41	-8.8	-.062	.029	.15	.35	8.6

Note: # patents granted and filed: number of patents granted and filed for a firm in a given year; 1-y gr. rates: 1-year growth rates of firm outcomes as defined in the text.

Table 1.2 provides summary statistics for the main variables. The distributions of number of patents filed and granted are very skewed to the left: While most firms do not file a patent, few firms innovate massively, filing in excess of 4000 patents in a given year. The big innovators — in terms of patent filed — are large firms like IMB, Microsoft or Sony, to name a few. Firms in this sample tend to be large (they are all listed companies) and also tend to grow fast. For instance, the average yearly growth rate of ouput is 8.2%.

1.3 Methodology

The following sections introduce the methodology of this paper. In Section 1.3.1.1 we describe how we build patent representations, illustrate their matching quality and their visualisation. We then continue with describing with illustrative examples how to compute the scores per patent that this paper uses. Section 1.3.2 then proceeds with introducing the regression frameswork for citation as well as firm level outcomes.

1.3.1 Patent representations

1.3.1.1 Derivation

Document term matrix This section explains how we obtain vector representations of documents. Suppose the entire corpus consists of four patents. Patent one's full text reads "Invention!", patent two reads "A good invention.", patent three "A better invention.", and patent four "A last good invention." When analysing the USTPO database, we concatenate the text of the abstract, brief description, and claims. After deleting punctuation, numbers, and converting

everything to lowercase letters, the resulting document term matrix (dtm) would have the following form storing word counts for $m = 4$ documents and $n = 5$ terms:

$$\underset{4\times5}{\tilde{A}} = \begin{array}{|ccccc|}\hline a & invention & better & good & last \\ \hline 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ \hline \end{array}$$

Each row vector is a 5 dimensional patent representation. As already visible in this example, dtms are usually sparse, i.e. contain large numbers of zero elements. In our application, we create vector representations for $m = 4,633,363$ unique patents from 1976 to 2017. To reduce noise and irrelevant information, we delete those words which are not at least contained in 5 documents. Furthermore, we also delete all words which are contained in 15% or more of the documents.[6] Without this, each row vector/document would seem similar because of sharing very common words ('the', 'invention', etc.) which have little signal for the technical content of a patent. Note that we abstract from this step in the exemplary dtm discussed here (otherwise we would e.g. delete the column for 'invention'). Both steps very substantially decrease the number of columns, however, we are still left with $n = 765,149$ columns/words in our final dtm which is therefore of dimension $4,633,363 \times 765,149$. To make documents comparable despite different lengths, we divide each row by its total word count. In the example above, we would divide the respective rows by $(1, 3, 3, 4)'$. Unlike related work such as Kelly et al. (2018), we do not employ additional inverse document frequency (idf) steps but rather delete too common words with the measures discussed above across all years. They also introduce a concept they call "backward-IDF" which makes the idf term weighting based only on patents filed previously (see Kelly et al., 2018, for full details). In contrast, we build the patent dtm using all years of the dataset in one go, and later select those rows which correspond to certain years whenever needed. In our example above, normalising by word counts would imply a dtm of:

[6] We furthermore delete stopwords such as 'the', 'a', etc., however, this should already be taken care of through the step of deleting words which are contained in 15% or more of the documents.

$$
\underset{4\times5}{A} = \begin{array}{|ccccc|}
\hline
a & invention & better & good & last \\
\hline
0 & 1 & 0 & 0 & 0 \\
1/3 & 1/3 & 0 & 1/3 & 0 \\
1/3 & 1/3 & 1/3 & 0 & 0 \\
1/4 & 1/4 & 0 & 1/4 & 1/4 \\
\hline
\end{array}
$$

Centroids This matrix now also allows to illustrate a concept which we use to direct our thinking in this paper. It is the mean patent vector of (potentially a subset of) the dtm or its counter parts in reduced dimensions (see next paragraph). We call this imaginary mean patent *centroid*. In the above example the centroid patent vector would be $centroid_A = (11/48,\ 23/48,\ 1/12,\ 7/48,\ 1/16)$. Different times and different industries (i.e. subsets of rows) have different centroids.

Dimension reduction with truncated SVD In a next step, we reduce the dimensionality (columns) of the dtm A. On the one hand this yields a more manageable patent dimension for computing millions of similarities between vectors, on the other it allows parts of the graphical analysis of Section 1.4 which is based on low dimensional representations. We use truncated singular value decomposition (tSVD) on the dtm to obtain these representations and store them in a matrix Z. To continue with our example, we run tSVD on our exemplary 4×5 dtm A with three components. It yields:

$$
\underset{4\times3}{Z} = \begin{array}{|ccc|}
\hline
z_1 & z_2 & z_3 \\
\hline
0.9377 & 0.3453 & -0.0387 \\
0.4621 & -0.3002 & -0.1103 \\
0.4390 & -0.1909 & 0.3223 \\
0.3626 & -0.2791 & -0.1496 \\
\hline
\end{array}
$$

Each column of Z is an eigenvector of AA^T multiplied by the associated singular value σ_i of A, starting with the largest. For a discussion of the technique, see for example Strang (2016). If we had subtracted column means in A before applying the SVD, the vectors in Z would be the often used Principal Components. The truncated version of the SVD computes only those vectors associated with the largest N singular values. This makes running this method feasible when considering very large matrices such as our true dtm which is of dimension $4,633,363 \times 765,149$, i.e. has trillions of cells. We choose the first 300 components

from the tSVD to represent our documents in reduced dimensions. The matrix Z is therefore of dimension $4,633,363 \times 300$. Each component/column is a linear combination of words. Each row is a depiction of a patent's textual content in a common space. Vectors of patents with similar content should have a similar angle. Running such a dimension reduction on a dtm and using it for analysis is often called Latent Semantic Analysis (Deerwester et al., 1990). In fact, a system using it for information retrieval was actually patented in the USTPO under patent number *4839853* from 1988 but it is now expired![7]

Dimension reduction with t-SNE We use the vectors obtained from the tSVD for computations. These computations should be faster with 300 dimensional vectors than with the 765149 dimensional counterparts in the dtm. Yet, to visualise the patent geometry, we would need representations in two or three dimensions. Taking only the first 2 or 3 columns from the SVD, however, would not preserve enough meaningful geometric information. This leaves the question whether we could still have a glimpse into the structures of high dimensional space. The method of t-distributed stochastic neighbour embedding (t-SNE) by Maaten and Hinton (2008) tries to achieve exactly this. It preserves cluster and structures from high dimensional space and separates clusters of vectors also in low dimensional space. This makes groups and structures of observations visible. We employ a very recent faster variant of t-SNE developed by Linderman et al. (2019) that the authors used for visualising single-cell RNA sequences in genetics. This fast interpolation-based t-SNE (FIt-SNE) creates a good approximation and is feasible even for matrices of very large dimensions like our patent representation matrix containing around 4.6 million patents/rows. In detail, we use their method to reduce the pre-reduced dimension of the patent representations contained in Z further to only two columns:

$$\underset{4,633,363\times 300}{Z} \rightarrow \underset{4,633,363\times 2}{Z^{\text{t-SNE}}}$$

Again, each row in this new matrix is a patent, just that every patent can now (for visualisation only) be represented by only two values. We can therefore plot its position in a plane. In terms of parametrisation, we use $20,000$ iteration and a learning rate 386113 which depends on the amount of rows in our matrix.[8]

[7] See `https://en.wikipedia.org/wiki/Latent_semantic_analysis`

[8] We are grateful to the author George Linderman for his explanations and the recommendation of the paper Kobak and Berens (2019) which, among others, discusses hyper-parameter settings for t-SNE. We use their recommendation for the learning rate as $M/12$, however, leave all other hyper-parameters at FIt-SNE's default except for using $20,000$ iterations.

Computing similarities between patents The main purpose of our patent representation is to compute similarities between patents. Patents of similar content should have similar patent vectors. Instead of the row vectors contained in A, we use those in Z to compute these similarities. We could not, however, use row vectors contained in $Z^{\text{t-SNE}}$. These vectors are only used for graphical illustrations as, in particular, the distance between clusters in 2D are distorted. When thinking about similar patents it might first seem most natural to think of Euclidean distance: Vectors which are close in space should have similar content. This works, however, only if we standardise all vectors by the L2 norm to force them have unit length and lie on a unit N-sphere. The reason is that similar documents, i.e. rows in Z, might otherwise still have different lengths. A method which does not need the standardisation (albeit it can still sometimes benefit from it a bit) is cosine similarity. Instead of the distances between between vectors, it looks at the angle between them. The angle of vectors that point into the same direction is zero, even if these vectors have different length. Cosine similarity is usually the standard used in the NLP literature as rows in the dtm can be distorted by different document lengths if not normalised. We use cosine similarity on the Z matrix and do not standardise vectors. Cosine similarity is given by:

$$\text{cosine similarity}(x, y) = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}} \in [-1, 1] \tag{1.1}$$

Going back to our example, patent number 2 "A good invention" has the following cosine similarities with the other patents:

$$\text{cosine similarity}(z_2, z_i) : (0.5941,\ 1,\ 0.69251,\ 0.9900).$$

The patent's similarity with itself is 1. And despite using the reduced dimension in Z, the most similar patent is indeed patent 4 "A good last invention.". Vector representations of similar documents also have similar angles.

1.3.1.2 Illustrations

Illustrating match quality Once we have computed Z, each row vector is the representation of a patent. As discussed above, picking one row, computing its similarity with each other row, and then sorting the array allows to find the most similar patents. Table 1.3 illustrates a few examples of the most similar patent

found. Often, the very closest match is a patent which seems to be a small perturbation of the original (see e.g. Example 3 in the table). Yet, we also show cases where our tools returns a most similar patent which is a less close match. Match quality seems very good given that such a method only has information on the bag of words that a document contains. In fact, Latent Semantic Analysis has also been proposed for patent examination support (Elman, 2007). Only Example 4 in the table seems to be a somewhat odd match. Yet, we add the full abstracts, descriptions, and claims text into Appendix 1.7.1. In fact, both technologies are much more similar than visible from the abstract and can e.g. be used for earthquake detection. This emphasises why we concatenate the texts of abstract, brief description, and claims instead of just e.g. using one of them. The exemplary text also shows that, despite their overall very good match quality, methods like ours will also still be biased by things such as common headers, names contained in the patents, etc. To alleviate this problem, we delete large parts of words (e.g. those that are in 15% of documents) or one could alternatively do weighting such in an idf step. Yet, headers or other common structures contained in only small subsets of patents from e.g. similar firms will also always make them, everything else equal, more likely to be matched. This is a limitation of these methods. As it is not trivial to distinguish common content in a small set of patents that is e.g. a common header versus content that is about innovations, this is an area where more sophisticated methods could be helpful in future research.

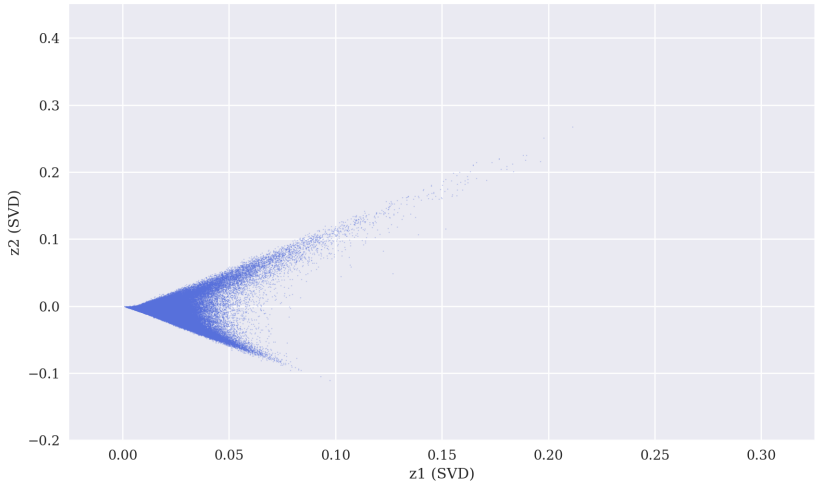
Illustrating t-SNE How do the two dimensional patent representations in the t-SNE matrix look like when plotted? Could we not just use the two first components from the tSVD or the two first principal components also for our visual analysis? 300 components are an excellent representation of a document, but just two of those unfortunately do not contain enough visual information for seeing any interesting structures from high dimensional space in just two dimensions. Figure 1.1 illustrates this. It contains all patents (each being a point in the plane) from 1976 to 1985 represented by their first two vectors from the tSVD. Most vectors sit in very similar areas.

Figure 1.2 depicts the same patents as the previous one, however, now it uses two t-SNE dimensions. The difference is very stark as t-SNE was built to separate cluster structures from higher dimensions also in lower dimensional representations. We can immediately see that our high dimensional space must contain very rich structures. This rich information is the graphical intuition behind the good match quality we saw before.

Table 1.3: Exemplary patent matches

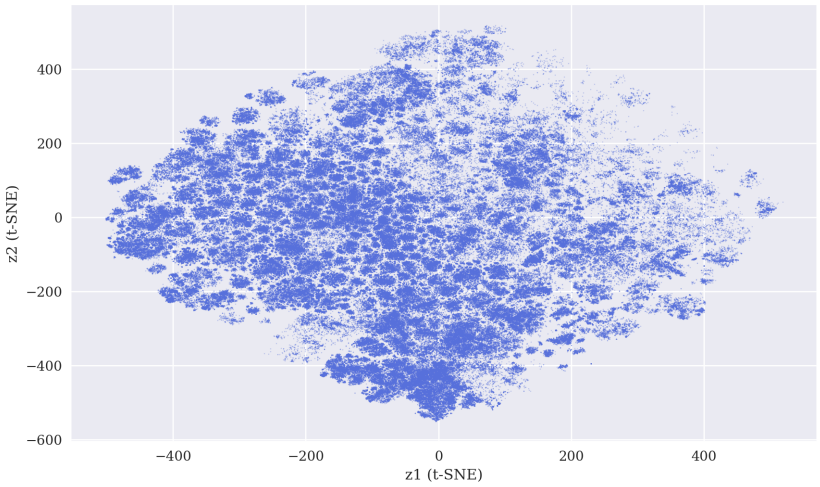
Reference patent	Beginning of abstract	Closest match	Beginning of abstract
9475668 (2013)	A modular element for a creel includes a structure having at least one support for supporting a package or bobbin of yarn; the structure being modularly couplable with other similar structures to allow the feeding of multiple yarns to a textile machine; ...	4753064 (1986)	The spinning or twisting machine comprises a plurality of drafting rolls, a plurality of associated spindles and positioned between them a plurality of yarn breaking devices pivotable between an upright spinning machine operating position into a yarn breaking position. ...
4900994 (1988)	An automatic window glass elevating apparatus for moving a motor normally or reversely by drive control by a switch actuation to move a window glass in a closing or opening direction which has a resistor connected in series with the motor, ...	4746845 (1986)	An automatic window regulator for automobiles includes a drive motor for lifting and lowering the window glass, and a detector for detecting when there is a foreign object jammed between the window glass as it is lifted or lowered and a window frame. ...
4640147 (1987)	A gear assembly comprising two gears and a spring in the form of a C-shaped clip interconnecting the two gears. Two pins are provided on one of the two gears and the spring has two holes, one in each end thereof, whereby the spring can be carried by one gear in a pretressed state by means of the pin-and-hole connection. ...	4745823 (1988)	A gear assembly comprising two gears and a spring in the form of a C-shaped clip for applying a resilient force between the gears. Two pins are provided on the respective side surfaces of the gears to receive the spring which has concave end surfaces to be received by the pins. ...
8919201 (2012)	An acceleration measuring apparatus that can easily detect acceleration with high accuracy is provided. In the apparatus, positional displacement of a swingable pendulum member is detected, feedback control is performed to maintain the pendulum member in a stationary state using an actuator, ...	8677828 (2012)	Provided is a device capable of easily and accurately detecting a vibration period when, for example, an earthquake occurs. When a quartz-crystal plate bends upon application of a force, capacitance between a movable electrode provided at its tip portion and a fixed electrode provided on a vessel to face the movable electrode changes, ...

Figure 1.1: The two first components obtained from truncated SVD



Each dot represents a patent filed from 1976 to 1985.

Figure 1.2: The two first components obtained from t-SNE



Each dot represents a patent filed from 1976 to 1985.

1.3.1.3 Computing scores

The empirical analysis of this paper uses a range of different scores which try to proxy the inventiveness of a patent. Continuing the example from before, this section describes how they are computed.

Before we can compute the three scores of the paper, we first have to introduce a concept we refer to as backward and forward spaces. Their key feature is that they are subsets of our matrix Z. Hence, we first obtained the patent representations in Z containing all year and all patents from 1976 to 2017 in one computation based on the full dtm. Afterwards, we repeatedly use different subsets of rows from this matrix Z when we compute different forward and backward spaces. Both our backward and forward time interval consists of 10 years. Backward and forward intervals thereby always include the year of the patent currently considered. Say a patent is from 1995, then its backward interval would be $\{1986, 1987, \ldots, 1995\}$ and its forward interval would be $\{1995, 1996, \ldots, 2004\}$. The intersection of the two sets is $\{1995\}$. Going back to our example, say the 4 patents in Z are from 1976, 1989, 1995, and 2004:

$$
\underset{4\times 3}{Z} = \begin{array}{ccc|c} z_1 & z_2 & z_3 & \\ \hline 0.9377 & 0.3453 & -0.0387 & (1976) \\ 0.4621 & -0.3002 & -0.1103 & (1989) \\ 0.4390 & -0.1909 & 0.3223 & (1995) \\ 0.3626 & -0.2791 & -0.1496 & (2004) \end{array}
$$

Then for patent number three from 1995 we have the following backward and forward space:

$$
\underset{2\times 3}{Z}^{\text{1995 backward}} = \begin{array}{ccc|c} z_1 & z_2 & z_3 & \\ \hline 0.4621 & -0.3002 & -0.1103 & (1989) \\ 0.4390 & -0.1909 & 0.3223 & (1995) \end{array}
$$

$$
\underset{2\times 3}{Z}^{\text{1995 forward}} = \begin{array}{ccc|c} z_1 & z_2 & z_3 & \\ \hline 0.4390 & -0.1909 & 0.3223 & (1995) \\ 0.3626 & -0.2791 & -0.1496 & (2004) \end{array}
$$

Backward and forward spaces always include the base year and patents from that year. If we have many patents from 1995 like in our actual data, note that each

patent from a given year has the same forward and backward spaces. These two spaces can easily contain hundred thousands of patents - all patents filed in a 10 year interval around and including 1995. Taking a constant 10 year interval around a given year makes computations comparable across patents. Yet, the 10 year interval prevents us from computing scores for any patent filed before 1985 and after 2008 since the available data ranges from 1976 to 2017.

Next we discuss how we use these concepts to compute scores for each patent that try to proxy its innovativeness. One type of score is based on centroids, another type is independent of these mean vectors.

1. Centroid based scores For centroid based scores we first compute the centroid (i.e. the mean vector) of the backward and forward space respectively. We use centroids to proxy both macro and micro trends in innovation, however, they can also be used to assign innovativeness scores to individual patents. In the following we will show how. Continuing with the example, the backward and forward centroids are given by:

$$z_{\text{centroid}}^{\text{1995 backward}} = (0.4506, -0.2456, 0.1060)$$

$$z_{\text{centroid}}^{\text{1995 forward}} = (0.4008, -0.2350, 0.0864)$$

Now the idea of centroid based scores is to see whether a patent, in our example patent three, anticipated the move in the centroid in the economy or an industry. For this we compute its similarity to the backward centroid and its similarity to the forward centroid. Here these are given by:

$$\text{backward similarity patent 3}^{\text{ centroid based}} = \text{cosine similarity}(z_3, z_{\text{centroid}}^{\text{1995 backward}}) = 0.9221$$

$$\text{forward similarity patent 3}^{\text{ centroid based}} = \text{cosine similarity}(z_3, z_{\text{centroid}}^{\text{1995 forward}}) = 0.9117$$

Following Kelly et al. (2018), we name similarities to patents in the past and future backward and forward similarities. If a patent is dissimilar to its backward centroid and similar to its forward centroid, we conjecture it is innovative and we would like it to have a high score. Like Kelly et al. (2018), we use the fraction of two similarities as the final score (albeit our scores in this paragraph are based

on centroids):

$$\text{final score patent 3}^{\text{ centroid based}} = \frac{\text{forward similarity patent 3}^{\text{ centroid based}}}{\text{backward similarity patent 3}^{\text{ centroid based}}} = \frac{0.9117}{0.9221} = 0.9887$$

Lastly, note that depending on which subsets we take from Z, the backward and forward spaces can contain all patents of the whole economy or only patents from certain IPC codes, e.g. at the IPC3 level. We call centroids of the first kind *macro centroids* and centroids of the second kind *micro centroids*. Each can be used as a reference point when computing different scores for patents: How well did a patent anticipate economy wide trends, how well did it anticipate field specific trends.

2. No-centroid scores We can also compute scores for each patent without the use of centroids. Rather than computing the patent's similarity to only a centroid vector, we now compute its similarity to each individual patent in its full backward and forward spaces. Afterwards we sort the similarities, and compute its mean similarity to the closest 100 patents. This no centroid score now is almost the same as used in Kelly et al. (2018), however, they take the sum of all similarities in the spaces. By focusing on the mean similarity to the top 100 closest patents (excluding itself), we try to emphasise whether a patent went into an empty part of space. Again with our example:

$$\text{backward similarities patent 3}^{\text{ no-centroid}} = \text{cosine similarity}(z_3, Z^{1995 \text{ backward}}) = (0.6925, 1)$$

$$\text{forward similarities patent 3}^{\text{ no-centroid}} = \text{cosine similarity}(z_3, Z^{1995 \text{ forward}}) = (1, 0.5912)$$

Strictly speaking, the cosine similarity was now applied repeatedly between z_3 and each row of the Z matrices. Next each list will be sorted: $(1, 0.6925)$ and $(1, 0.5912)$. Here our toy example reaches its limit as the backward and forward space only contain two patents each. The backward and forward spaces in our application consist of very large amounts of patents, and so these two lists can contain several hundred thousand similarities. We sort them, pick the top 100 most similar values and exclude the first (which is 1 always as the patent itself is contained in both backward and forward space), then we compute their mean. Here these two means are given by:

$$\text{backward similarity patent 3}^{\text{ no-centroid}} = 0.6925$$

$$\text{forward similarity patent 3}^{\text{ no-centroid}} = 0.5912$$

Again the hypothesis is that a patent that was dissimilar to patents in the pasts and similar to patents in the future should be innovative and have a high score. We compute the final no centroid score as:

$$\text{final score patent 3}^{\text{ no-centroid}} = \frac{\text{forward similarity patent 3}^{\text{ no-centroid}}}{\text{backward similarity patent 3}^{\text{ no-centroid}}} = \frac{0.5912}{0.6925} = 0.8537$$

The difference of this score to the centroid based scores is that it compares patent three to all patents contained in the backward and in the forward spaces. The centroid based scores proxy the state of the backward and forward space just by their respective mean patent and take only one similarity to each of them. Centroid based scores are therefore also much less computationally costly. Yet, no-centroid scores should have a better chance to find out which parts of space are still relatively empty as the patent is compared to all other individual patents in these spaces.

3. Forward neighbourhood heterogeneity For the last score we introduce, we take a patent and identify the 100 most similar patents in the forward space only. Different to the no-centroid score, we then look at their composition instead of taking a mean over their similarities. This is easily explained, however, we have to move away from our illustrative example used so far. In the data, there are 8 unique IPC 1 codes (see Table 1.9 in the Appendix for details) which classify the technological area of a patent. We then store how many of the most similar 100 patents belong to which IPC code. Say we consider a patent with a patent neighbourhood of $p = (60/100, 10, 0, 0, 0, 0, 0, 30/100)$ over the IPC codes (A, B, C, D, E, F, G, H). Then we take an Euclidean distance $d(p, q) = \sqrt{\sum(p_i - q_i)^2}$ to a uniform vector $q = (\frac{1}{8}, \ldots, \frac{1}{8})'$. The uniform vector $(\frac{1}{8}, \ldots, \frac{1}{8})'$ represents an imaginary reference neighbourhood with all IPC codes in equal proportion, i.e. a neighbourhood that is as heterogenous as possible in this setting. Next, note that the maximum Euclidean distance in our setting is given when a patent has a neighbourhood with 100 patents from a single IPC code, e.g. $p = (100/100, 0, 0, 0, 0, 0, 0, 0)'$. In that case, we have an Euclidean distance of $d\Big(p = (1, 0, 0, 0, 0, 0, 0, 0)',\ q = (\frac{1}{8}, \ldots, \frac{1}{8})'\Big) = 0.9354$. As our score is meant to measure heterogeneity not specialisation, however, we then define it as:

$$\text{forward neighbourhood heterogeneity of p } = 0.9354 - d(p,\ q = (\frac{1}{8}, \ldots, \frac{1}{8})')$$

A patent's value for forward neighbourhood heterogeneity is highest if its neighbourhood in the next 10 years is most heterogenous, i.e. if contains an equal mix of all 8 IPC 1 codes.

The next section continues with introducing how we use these scores in regressions on citations and firm level outcomes.

1.3.2 Regression frameworks

This section details the regressions that are estimated at the patent and firm levels. The regressions will be run with both centroid and no-centroid scores. In this section, we therefore us a generic term *score* in the regressions and the descriptions for exposition purposes.

1.3.2.1 Citations

The aim is to check whether our scores are associated with forward citations. Strictly speaking, the exercise is not a prediction exercise, as information available several years after the filing date is used to compute the scores. Regressions of the following form will be estimated:

$$\text{citations}_{pjf,t+h} = \beta \text{ score standardised}_{pjf,t} + \mathbf{FE} + \epsilon_{pjf,t}, \tag{1.2}$$

where the dependent variable is the number of times a patent p from IPC code j owned by firm f — when this information is available — is cited by other patents h years after its grant year t, where $h \in \{5, 10\}$. The variable *score standardised* is the score of the patent, as defined above, standardised to unit standard deviation to make the interpretation of regression coefficients easier. **FE** is a set of fixed effects that varies depending on the exact specification. Combinations of filing year, IPC codes at different levels and firm fixed effects are included. Standard errors are clustered at the filing year level. An alternative specification in logs will also be estimated for robustness, as follows

$$\log(1+\text{citations}_{pjf,t+h}) = \tilde{\beta} \ \log(\text{score}_{pjf,t}) + \mathbf{FE} + \epsilon_{pjf,t}, \tag{1.3}$$

where the score variable is not standardised to unit standard deviation, and the remaining variables are defined above. $\tilde{\beta}$ can therefore be interpreted as an elasticity. The coefficients of interest are β and $\tilde{\beta}$, which we expect to be positive, i.e. a high score is predicted to be associated with more forward citations. This is the hypothesis that will be tested. The regressions above will also be estimated using the private economic value of patents as dependent variable, another measure of importance of a patent. Note that the citations are counted from the grant date of a patent, yet the scores are computed using the filing year

of a patent as the time of reference for determining the technological landscape at the time of innovation.

1.3.2.2 Firm level outcomes

At the firm level, the aim is to estimate the effect of filing patents with a high score on firm-level outcomes in the following years. We choose the filing date as opposed to the grant date to define the event, because a firm may start using a patented invention before the patent is granted. The firm-level outcomes of interest are output, profits, capital and employment. Patent-level information must be aggregated at the firm level. One firm may file several patents in a given year — especially in this sample of large firms. Patents with a high score are typically rare and filed only by few firm-year observations. We define *high-score patents* as patents whose scores are in the top 5% of the overall score distribution net of year fixed effects.[9] The variable of interest at the firm level is a dummy denoted $D_{fi,t}$ that takes value 1 if firm f from industry i — defined as the 3-digit SIC code — filed such a patent in year t. We consider a dummy as opposed to the number of top patents since firm-year observations that file a top patent typically represent a small fraction of the observations, and observations with more than one top patent are even rarer. See Kelly et al. (2018, p. 29) for a discussion of this choice.

The generic specification of interest at the firm level closely follows that in Kelly et al. (2018, Equation 19) and reads as follows:

$$\log\left[\frac{1}{\mid h \mid}\sum_{\tau=1}^{h} Y_{fi,t+\tau}\right] - \log Y_{fi,t} = \beta_h D_{fi,t} + \boldsymbol{\gamma}\boldsymbol{Z}_{fi,t} + \mathbf{FE}_{it} + \epsilon_{fi,t+h}, \qquad (1.4)$$

where $Y_{fi,t}$ is either output, profits, capital or employment of firm f from industry i in year t, $\boldsymbol{Z}_{fi,t}$ is a vector of controls and $D_{fi,t}$ is defined above.[10] The left-hand side of the regression is the growth in the average of outcome variable Y between t and $t+h$ relative to year t.[11] Following Kelly et al. (2018), we consider horizons between -5 and +10, i.e. from 5 years before and up to 10 years after the filing date.[12] In all these regressions, we control for the log of total assets, the age of the firm since its entry into the data, a dummy for whether the firm filed a patent

[9] The results are also estimated for a threshold of 1% for robustness, and are available upon request.

[10] This specification is also similar to Equation (12) in Kogan et al. (2017).

[11] The coefficients for $h < 0$ are therefore the growth rate between $t + h$ and h, taking t as the base year. A negative β_h therefore means a positive growth rate if $h < 0$.

[12] Kelly et al. (2018) also use the filing year as the event date.

in that year, the log of (1+) the number of filed patents in that year, a dummy for whether the firm is in the top percentile(s) in terms of number of patents filed in that year, the share of top patents among filed patents in that year (for that firm), as well as in the stock of patents up to $t-1$ (for that firm). The lag of the level of the outcome variable — whose growth rate is the dependent variable — is also included in all regressions. By including year-industry fixed effects, the aim is to compare how a firm that files a top patent fares relative to other firms in that industry, at that time. Standard errors are clustered at the firm-year level.[13]

The coefficients of interest are the sequence of β_h. This specification allows to test for the absence of pre-trends: $\beta_h \, \forall h < 0$ should not be significantly different from 0, otherwise that firm may already be on a different trend before the filing date, and this may be unrelated to the invention. This dynamic specification also allows to study the effect of filing a top patent over time, as it is not obvious when — if at all — the effects should be apparent.

It should be noted that the dependent variable in Equation (1.4) is very smooth over time for a given firm, i.e. it will vary little and especially so in the later years. This is because the average profits over longer horizons is not very sensitive to the last data point added to the average. Furthermore, a stance needs to be taken relative to missing data points as it is unclear what should the value of the average growth rate be if a value (or more) is missing at some point of time within the horizon of interest, which is not uncommon given the unbalanced nature of the panel. For robustness and transparency, we drop any observation which has a missing or more at any given horizon — for a given firm. This changes the composition of the sample over h whereby the effects for large h are estimated based on firms that survive at least for that long. However, firms with missing data are not kept in the sample when they are actually not observed.[14] As a last robustness check, we also estimate Equation (1.4) with the growth rate of firms outcomes over the whole horizon as dependent variable, namely $\log Y_{fi,t+h} - \log Y_{fi,t}$ as used by Kogan et al. (2017).

[13] Autocorrelation-consistent standard errors were also used (in conjunction with clustering) for robustness as the dependent variable is likely to be auto-correlated over time. The resulting standard errors are usually much smaller. We chose to err on the side of caution and report the main results with the regular clustered standard errors.

[14] The results generally look stronger and the sequence of β_h is smoother when keeping observations with missing data points, but we feel it is partly artificially driven.

1.4 Results

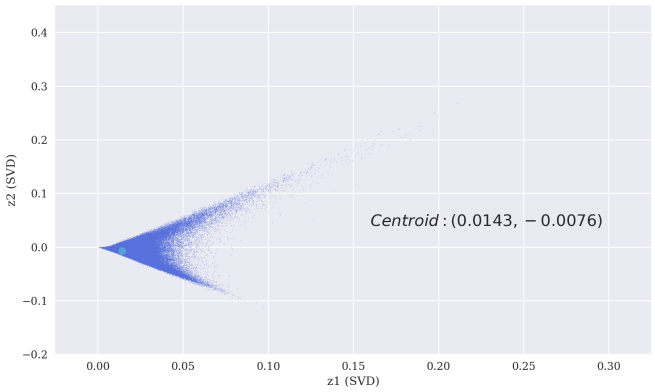
1.4.1 Micro and macro trends in innovation

The first question we think about is whether we can visualise economy wide macro and IPC-specific micro trends in innovation and see whether patents anticipating these trends have higher citations and benefit the filing firms. For this we use the concept of centroids which was introduced and discussed in Sections 1.3.1.1 and 1.3.1.3.

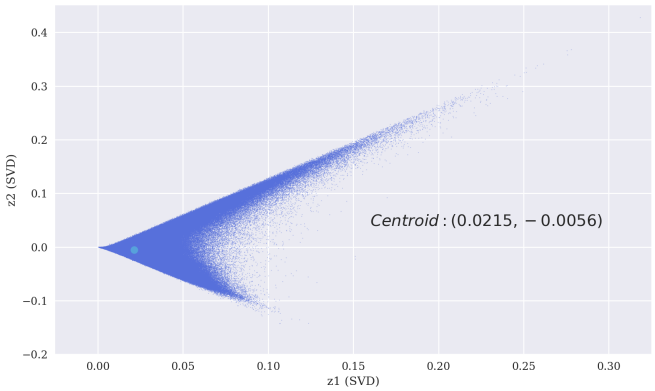
1.4.1.1 Geometry

First recall that a centroid is the mean vector of a set of patents. This set of patents is some subset of Z. For a macro centroid, it is the subset of all patents in the economy filed in a range of years. For a micro centroid, it is the subset of all patents with a given IPC code filed in a range of years. Put differently, centroids are imaginary mean patents which can be thought of as an approximate centre of gravity at a given time, either of the whole economy or of some field. Centroids also move over time. Their movement is substantial and meaningful if we consider the change in the mean of the full 300 dimensional patent representations. Yet, to illustrate the concept intuitively, let us look at the movement of the first two components of the tSVD. This is depicted in Figure 1.3. As already noted in Section 1.3.1.2 the spatial information contained in the two first component is very limited. We could not use the t-SNE plots to compute centroids, however, as t-SNE is a tool only for visualisation and the distances between clusters are distorted. Hence computing some mean point in between t-SNE clusters is infeasible. For all computations we therefore use the components obtained via the tSVD. In the graphical analysis of this section, we always compare the same two time frames: All patents filed 1976 to 1985 vs. all patents filed 2006 to 2015. In all our computations, however, we analyse smaller movements with a 10 year backward and a 10 year forward interval (see Section 1.3.1.3 for all details). This works very well for computations, but to make the visual analysis more pronounced we visualise the 30 year time span between the edges of our sample in the figures of this section. Figure 1.3 shows a move in centroid vectors. The actual distance moved is moderate visually if we only consider the two components (although the x dimension changes by roughly 50 percent), but it hopefully carries the intuition of what such changes in centroids mean.

Figure 1.3: Economy wide macro centroids (first two SVD components)

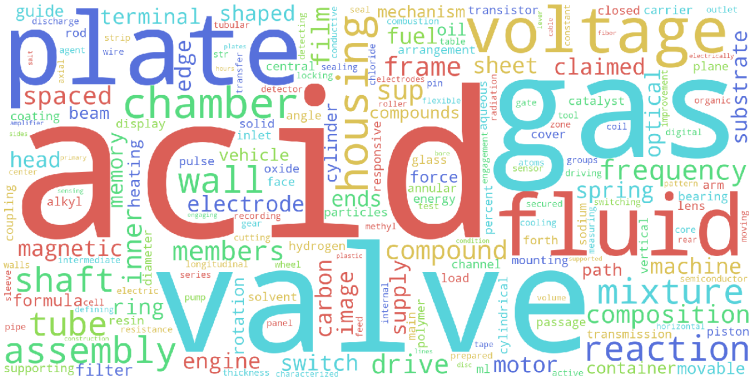


(a) All patents filed from 1976 to 1985

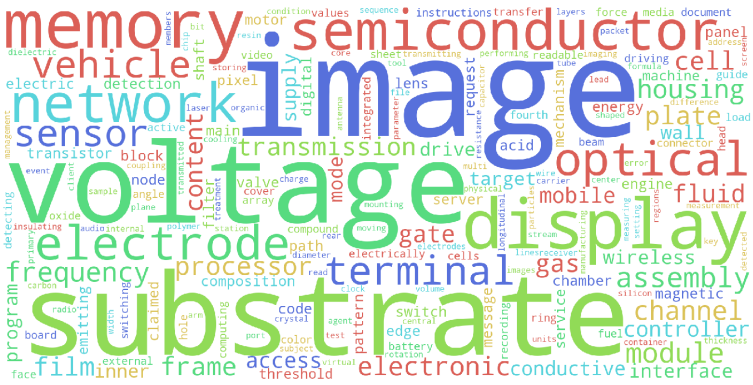


(b) All patents filed from 2006 to 2015

Figure 1.4: Economy wide macro centroids



(a) Words in 1976 to 1985 macro centroid



(b) Words in 2006 to 2015 macro centroid

Can we visualise the changes in centroids also differently or are all meaningful changes hidden in high dimensions? A great particularity of centroids is that we can look "into" their language content. For this we compute the same centroid, but instead of using the Z matrix we use the corresponding rows in the dtm (in which columns are words) and for each column take the mean over rows which yields mean word frequencies. Hence we can change between numerical points in our 300 dimensional space and the words approximately contained in these points (subject to some approximation error). Recomputing the same two centroids but now with the dtm makes clear that the centroid movements have been indeed very substantial over this time. Figure 1.4 illustrates that whereas the economy wide or macro centroid 1976-1985 prominently features words from chemistry such as *acid* or *gas*, the more current macro centroid from 2006-2015 shows words such as *memory*, *semiconductor*, or *network*. The macro centroid movement, i.e. the movement of the imaginary mean patent for the whole economy, intuitively depicts the development of the computing age.

The same methodology also allows to look into trends in innovation at an almost arbitrarily small IPC code level. First note that for many IPC 3 codes the move-

ment in word is surprisingly stable - at least at the approximate visual level via word clouds.[15] We therefore illustrate the movement in centroids for two IPC 3 levels with relatively clear trends. The first movement of IPC 3 centroids is that of "H04: Electric communication technique" depicted in Figure 1.5. The movement in centroid content immediately reveals the emergence of wireless communication technologies. The word *network* was hardly mentioned before, *telephone* became *mobile*. The second IPC 3 level micro centroid movement we show as an illustration is "C01: Inorganic chemistry" depicted in Figure 1.6. Besides others, the trend depicted indicates that chemistry patents with *acids* lost in importance whereas *carbon* and *gas* rose in popularity within the inorganic chemistry area. Equipped with the knowledge of this quick view into the centroid, one could now easily analyse the contents of patents that contain these specific words in depth. Note also that IPC 3 codes are still relatively aggregated. To be consistent with the subsequent regression analysis which is based on scores computed with macro centroids and IPC 3 centroids, we depicted word clouds at an economy wide level and at an IPC 3 level of aggregation. Yet, the same concept of centroid visualisation could still be used in further research to make developments in innovation visible in an automated way at say an IPC 8 level or a specific granularity of interest.

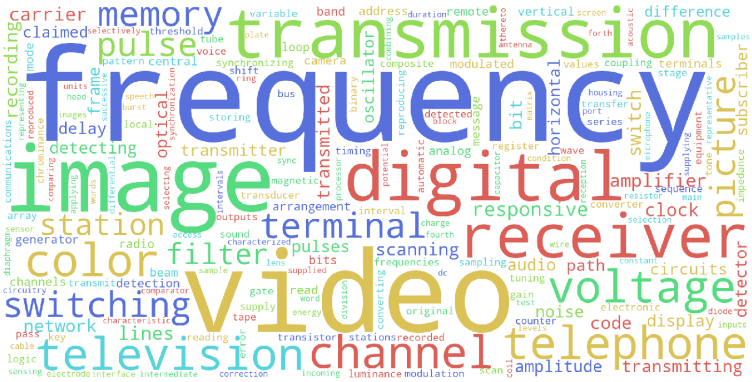
Centroids are very handy for a quick visualisation of macro as well as potentially very small grain micro trends in innovation. But the geometric intuition of their movement through space turns out to also have strong links with citations and with some firm level outcomes. Patents and firms which anticipated these movements in macro as well as micro trends, have higher citations and firms that own these patents tend to maker higher profits. Section 1.4.1.2 illustrates these findings.

1.4.1.2 Regressions

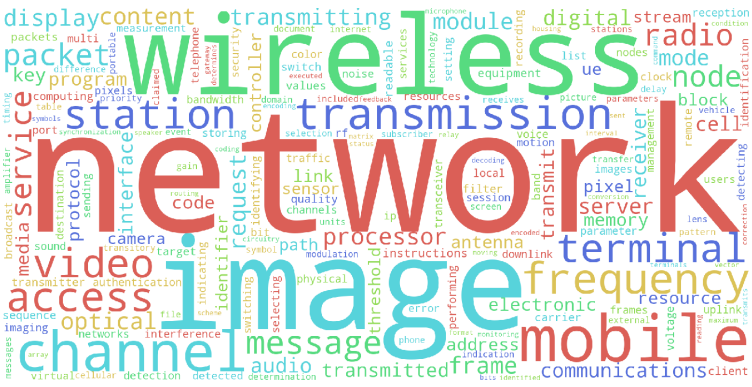
The existence of macro and micro trends suggest that the timing and content of firms' innovative output may matter for how successful their patents are, and ultimately for their economic performance. Specifically, a firm that anticipates or starts a major shift in the technological landscape may reap the benefits of being the first to do so. This idea applies both at the macro level — an invention in the IT sector in the 1990s may be very influential — and at the micro level

[15] In general, world cloud visualisations provides a good first idea idea, however, font size does not only relate to frequency but also e.g. to the method trying to position large amount of words in a rectangular shape. Depending on the random seed chosen they can give somewhat different impressions.

Figure 1.5: Industry specific centroids for "H04: Electric communication technique"

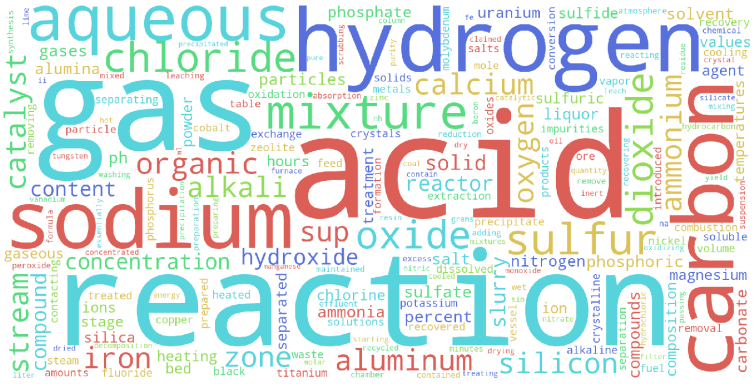


(a) Words in 1976 to 1985 *H*04 micro centroid

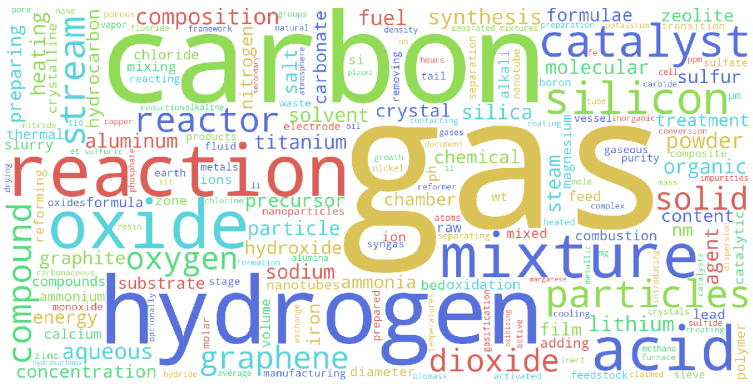


(b) Words in 2006 to 2015 *H*04 micro centroid

Figure 1.6: Industry specific centroids for "C01: Inorganic chemistry"

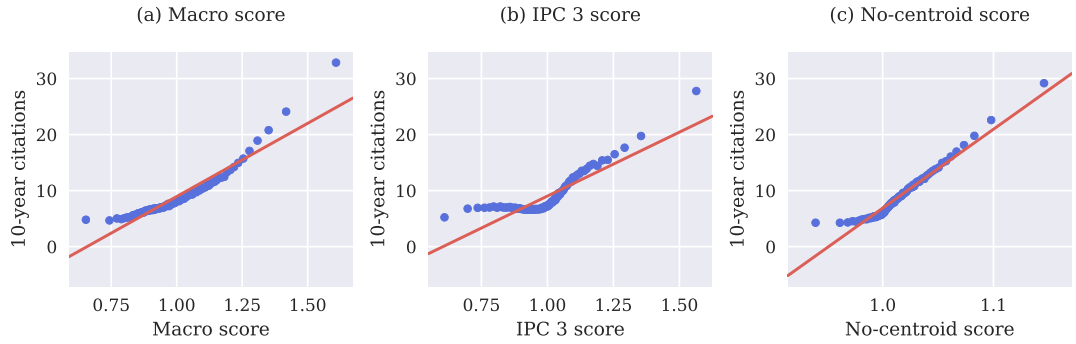


(a) Words in 1976 to 1985 *C*01 micro centroid



(b) Words in 2006 to 2015 *C*01 micro centroid

Figure 1.7: Scores versus citations



Note: Binned scatter of scores against 10-year forward citations. Number of bins in each plot is 100.

within a narrowly defined technological field. Analysing how far a patent is to the relevant centroid when it is filed and how close it is to the new centroid 10 year later may be indicative of how topical a patent is. In the following sections, we study whether such patents have higher citations, and whether they are associated with better performance at the firm level. The scores in this section are centroid based scores as described in detail in Section 1.3.1.3. The *macro centroid scores* is computed with backward and forward spaces that contain all patents (their mean patent is a macro centroid) and the *IPC 3 centroid score* is computed with backward and forward spaces that only contain patents of the same IPC 3 level as those for which the score is computed (the mean of this set of patents is the micro centroid).

Citations Both macro and IPC 3 scores are positively associated with patent citations, as depicted in Figure 1.7.[16] Patenting an invention far from the centroid at the time of filing yet close to the new centroid 10 years later is associated with higher citations. To formalise the results, Equation (1.2) is estimated using both the macro and IPC 3 scores. The results can be found in columns (1), (2) and (3) of Tables 1.4 and 1.5, respectively. The relation between scores and citations is statistically significant and economically large. In the specification including filing year and IPC 3 fixed effects, a one standard deviation increase in the macro score is associated with an increase of 4 citations, more than half of the mean citations (see Table 1.1). Similarly, a one standard deviation increase in the IPC 3 score is associated with an increase of 3 citations — half of the mean citations

[16]Since binned scatter plots sometimes hide relevant heterogeneity in the raw data, binned scatter plots with varying number of bins are provided in Figures 1.20 and 1.21 for the macro and IPC 3 scores, and scatter plots of the raw data are provided in Figure 1.23.

Table 1.4: 10-year citations and macro score

	Dependent variable: 10-year citations					
	Whole sample			*Until 2000*		
	(1)	(2)	(3)	(4)	(5)	(6)
Macro score, standardised	3.976***	4.425***	3.870***	4.517***	4.474***	4.002***
	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
Constant	-17.16***			-19.60***		
	(0.000)			(0.000)		
Year FE		✓	✓		✓	✓
IPC 3 FE		✓			✓	
Firm FE			✓			✓
Adjusted R^2	0.043	0.079	0.163	0.086	0.122	0.222
Within R^2		0.028	0.026		0.042	0.042
Observations	2896300	2896013	1025243	1545952	1545675	602117

Notes: *: p<0.1, **: p<0.05, ***: p<0.01. P-values from standard errors clustered at the filing year level in parenthesis.

in the sample.[17] Higher IPC 3 scores are also associated with a higher economic value as measured by Kogan et al. (2017), whereas it is not the case for macro scores — see Tables 1.14 and 1.15 in the Appendix. All the results above remain qualitatively unchanged using citations 5 or 15 years after grant date.

It should be noted that the association between citations and scores is somewhat weaker in the last decade of our sample. Columns (4), (5) and (6) of Tables 1.4 and 1.5 present the results of the same regressions as above, limiting the sample to the years up to 2000. Although the coefficients have similar magnitudes, the R-squared are significantly higher. It is unclear why this is the case. Tentatively, it could be due to the increase in the number of patents filed in the later years, which results in a more crowded space in the vector representation and prevents our method from identifying successful patents. This issue will be further discussed in Section 1.5.

It is interesting that the relationship with citations is strong both at the macro and the IPC 3 levels. At the macro level, it is likely to reflect the advantage of being an early innovator in a field that will become important for the economy as a whole. Given the illustrations of the previous section, it seems that the macro scores may actually capture the IT revolution, a field that was near nonexistent in the 1980s and that came to dominate the technological landscape in the 1990s and 2000s. Indeed, most of the patents with high macro scores are from IT-related sectors. This is also true for IPC 3 scores, which perhaps indicates that

[17] The results using specification (1.3) can be found in the Appendix, in Tables 1.12 and 1.13. The results are similar.

Table 1.5: 10-year citations and IPC 3 score

	Dependent variable: 10-year citations					
	Whole sample			*Until 2000*		
	(1)	(2)	(3)	(4)	(5)	(6)
IPC 3 score, standardised	3.071***	2.999***	2.704***	3.584***	3.417***	2.842***
	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
Constant	-13.78***			-16.27***		
	(0.000)			(0.000)		
Year FE		✓	✓		✓	✓
IPC 3 FE		✓			✓	
Firm FE			✓			✓
Adjusted R^2	0.025	0.075	0.157	0.048	0.124	0.212
Within R^2		0.024	0.019		0.047	0.030
Observations	2745260	2745260	993767	1448555	1448555	580505

Notes: *: p<0.1, **: p<0.05, ***: p<0.01. P-values from standard errors clustered at the filing year level in parenthesis.

IT is the sector whose centroid moved the most over time.

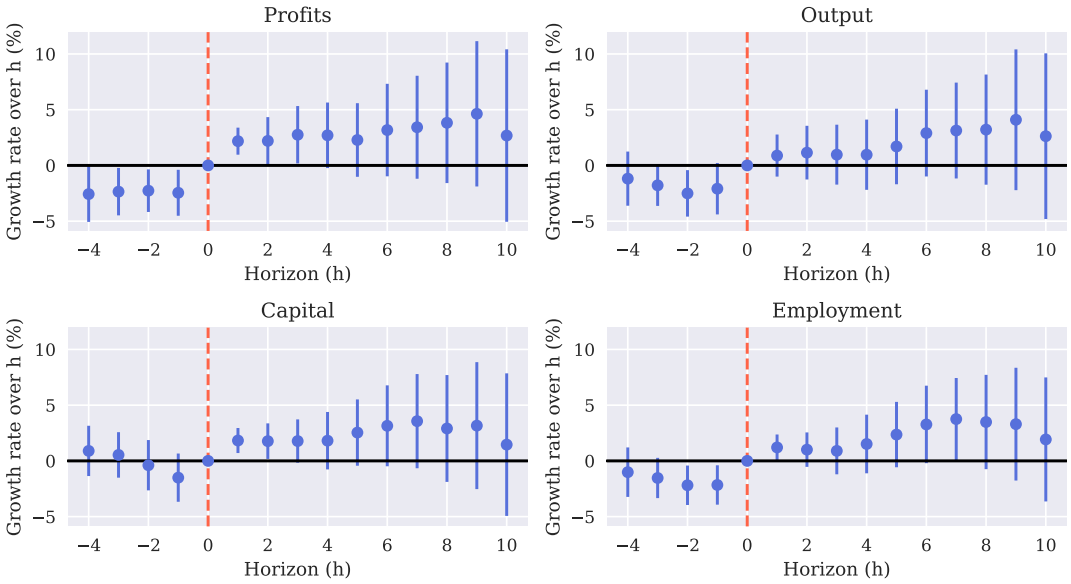
Firm level performance A firm that files a patent with a high score may benefit from such an invention. In order to test whether this is the case, Equation (1.4) is estimated for both the macro and the IPC 3 scores. The treatment variable is a dummy that indicates whether a firm filed a patent in the top 5% of the chosen score distribution, which we call a high-score patent.[18] The results are shown graphically in Figures 1.8 and 1.9, which depict the sequence of estimated β_h for profits, output, capital and employment.[19]

The results suggest that filing a top patent in terms of IPC 3 scores is generally associated with higher subsequent growth in output and capital for the filing firm relative to other firms in that industry and year, although the effects are not strongly significant. Profits seem to be on an upward trajectory even before the event date. These pre-trends suggest that those firms that file a high-score patent are *already* growing faster than other firms *before* the innovation happens. It is therefore impossible to identify the causal effect of patenting a high-score invention on profits. Instead, our method, despite being based only on language, seems to identify some firms which are generally innovative. Since innovative

[18] As mentioned in Section 1.3.2, we consider top patents in the distribution of scores removing year fixed effects, which means that high score patents are those with the highest scores among the patents filed in the same year. We also ran the regressions using the top patents in the overall distribution without removing year fixed effects, and the effects are similar. The results are also similar when classifying the top 1% patents as high-score patents.

[19] The results estimated using the alternative specification of (1.4) with $\log Y_{fi,t+h} - \log Y_{fi,t}$ as dependent variable can be found in Figures 1.24 and 1.25.

Figure 1.8: Top macro patents and firms dynamics



Note: Estimates from Equation (1.4) using the macro score to qualify top patents. 95% confidence intervals are depicted.

output is highly persistent, it is perhaps not surprising that high-score patents are produced by firms that have a consistently high level of innovation and that grow faster than other firms on average. Filing a top patent in terms of the macro score does not result in any statistically significant effect on profits, output, capital or employment for the filing firm.

1.4.2 Widening existing ideas

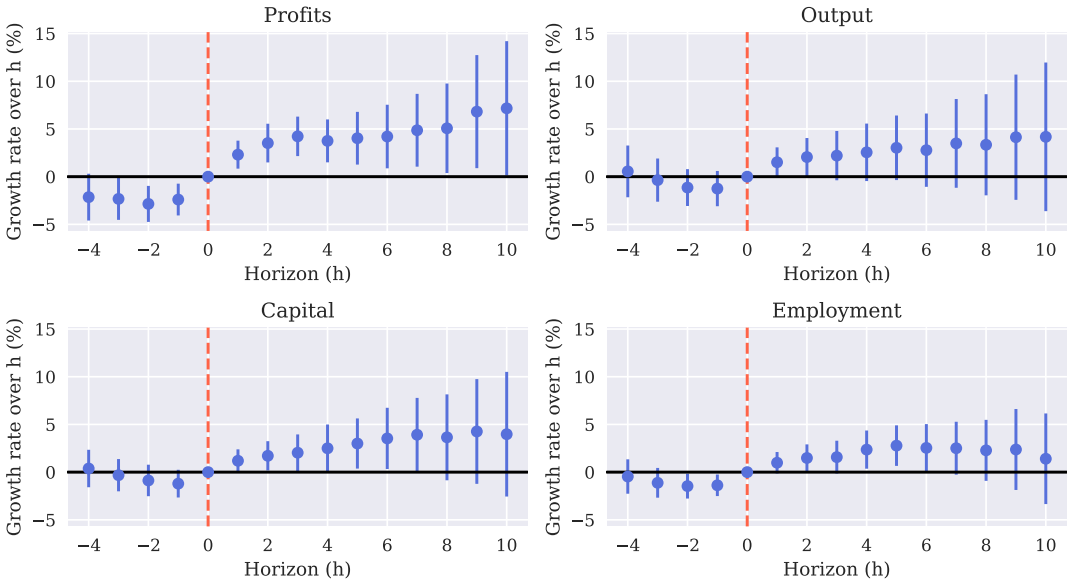
The second economic question we explore is whether we can use our geometrical approach to define a group of patents that widened knowledge by venturing into unexplored parts of the technological space.

1.4.2.1 Geometry

As widening innovation can mean very different things in different contexts, let us begin with a definition of what we mean when we refer to widening patents in our context:

Widening patent When filed, a patent which we call *widening* was dissimilar to its closest neighbours of the past 10 years, but its closest neighbours of the then

Figure 1.9: Top IPC 3 patents and firms dynamics



Note: Estimates from Equation (1.4) using the IPC 3 score to qualify top patents. 95% confidence intervals are depicted.

following 10 years became very similar to it. In other word, a widening patents entered a region with few similar patents in the backward backward space and lies in a region with many similar patents in the forward space.

Can we just stick to our previously computed centroid based scores with some IPC level and then say that patents with a high score widened their field? For this question, the t-SNE visualisations come very handy. They allow us to have a look into our high dimensional patent space. In particular, they shows how mixed some topics seem to be across IPC classes. Consider Figure 1.10 which shows the universe of patents which have been filed from 1976 to 1985, each dot being a patent. Importantly, when we created our patent vectors, the method had no information about which IPC code a patent belonged to. Only afterwards we colour-code each dot according to its IPC 1 level (see Table 1.9 in the Appendix for a description of each IPC 1 code). Now we use t-SNE to visualise in two dimensions how our 300 dimensional tSVD representation space looks like. And in fact, the tSVD must have autonomously arranged large relatively separated clouds of chemistry patents (C) and of electronics and physics (G and H) without knowledge they belonged to what we call the same IPC codes. Other groups, however, such as Human Necessities (A), Textiles (D), or Fixed Constructions (E) are much less clustered and seem to spread across many topic areas of the language space. We later find the same computationally in Section 1.4.3. There seem to be distinct areas in the space of innovation, e.g. around coordinates

(-150, 50), which seem to share content from a lot of different IPC classes.

This is very helpful information. If we were to rely on IPC classes for centroids, we could in fact not accurately identify whether a patent satisfies our definition of widening: a patent might be surrounded by many patents from other fields than its own. The t-SNE visualisation indicated that if we want to talk about widening innovation in the way we defined it, centroid scores may not be accurate.

We therefore move to the *no-centroid* score described in Section 1.3.1.3. For this score, we compare a patent to every patent in its backward space (i.e. patents of the past 10 years) and its forward space (i.e. patents of the next 10 years), not just a single IPC-based centroid or mean vector. No-centroid scores lack the convenient notion of a trend or centre of gravity of innovation, but have a better chance at giving good proxies of empty parts of space irrespective of IPC classes. We say that patents with a high no-centroid score are on average widening topics: They should have been dissimilar to their neighbourhood in the backward space and similar to their neighbourhood in the forward space. Note hereby that our score is very similar to the one proposed very recently in Kelly et al. (2018), with the difference that we take means of the closest patents' similarities only whereas they use the sum of all individual similarities in the backward and forward space.

Besides emphasising the need for a different score which we analyse in the following section, there are also a few other interesting points to see in the t-SNE visualisation. These become clear particularly when looking at Figure 1.10 and Figure 1.11 together. The areas of space for Physics (G) and Electricity (H) in 2006 to 2015 are very crowded, in line with what we would expect from the IT revolution. This part of space was also largely empty before, as visible in Figure 1.10. Of course this is exaggerated by our large illustration time interval between the two plots. But also with the 10 year intervals in the score computations this will most likely be exactly the area where we will find a lot of patents that have high scores and meet our definition of widening. We confirm this later in the discussion in Section 1.5, but the dominance of classes G & H as widening patents can already be seen intuitively in the plots here. They are in the regions with most empty areas early and crowded areas later. It remains a key question for which technology this will be true next. As of now patents from fields such quantum computing and artificial intelligence might still be more similar to existing patents than was the case for the IT revolution.

We can also use the two t-SNE plots to see the birth and death of certain areas of ideas. For example a subfield of chemistry strongly linked with human necessities was born around the coordinates (-200,-300): It is not visible in Figure 1.10, but

appears in Figure 1.11. Another subfield of chemistry at the southern border around coordinates (0, - 300) in contrast disappeared over the last decades.

A key question is whether these geometric observations also translate into economically more tangible quantities like citations and firm outcomes. The next sections shows that this seems to be the case.

1.4.2.2 Regressions

Armed with the no-centroid scores computed as described in Section 1.3.1.3, we analyse whether high-score patents — which we call widening patents — are successful patents in terms of citations and private value, and whether the firms from which the inventions originate benefit and perform better relative to their peers. The no-centroid scores allow for a neater narrative and a finer analysis than the centroid scores, where the score of a patent was obtained by comparing its content to economy-wide or sector-specific *average* content at different points in time. A patent with a high no-centroid score is a patent that was distant to its closest neighbours when filed and that got new close neighbours in the subsequent 10 years. The neighbours need not be from the same sector or field of technology.

Citations High no-centroid scores are associated with more citations: a one standard deviation increase in the score is associated with slightly more than 3 additional citations over 10 years — an increase representing about 50% of the mean citations over 10 years in the sample. The results can be found in Table 1.6 — and Table 1.16 for the specification in logs. Both the effects and goodness of fit are similar to the results obtained using macro and IPC 3 scores, which is somewhat surprising since the definitions of the scores differ. Widening patents are also associated with higher private values — as reported in Table 1.17 — and the correlation is higher than in the case of IPC 3 scores. All the results above remain qualitatively unchanged using citations 5 or 15 years after grant date. Note again the decrease in the goodness-of-fit of these regressions in the years after 2000 — columns (1)-(3) versus (4)-(6) in Tables 1.6 and 1.16.

A high score could in principle be the result of a high forward similarity and a low backward similarity — our definition of widening patents — but could also be driven by a very low backward similarity and a low forward similarity. In that case, the patent would be dissimilar at the time of filing, and yet remain fairly dissimilar 10 years later, although less so. The case that this type of patents are wideners would be less strong. To confirm that the positive association between

Figure 1.10: IPC 1 coloured t-SNE representation of patents from 1976 to 1985

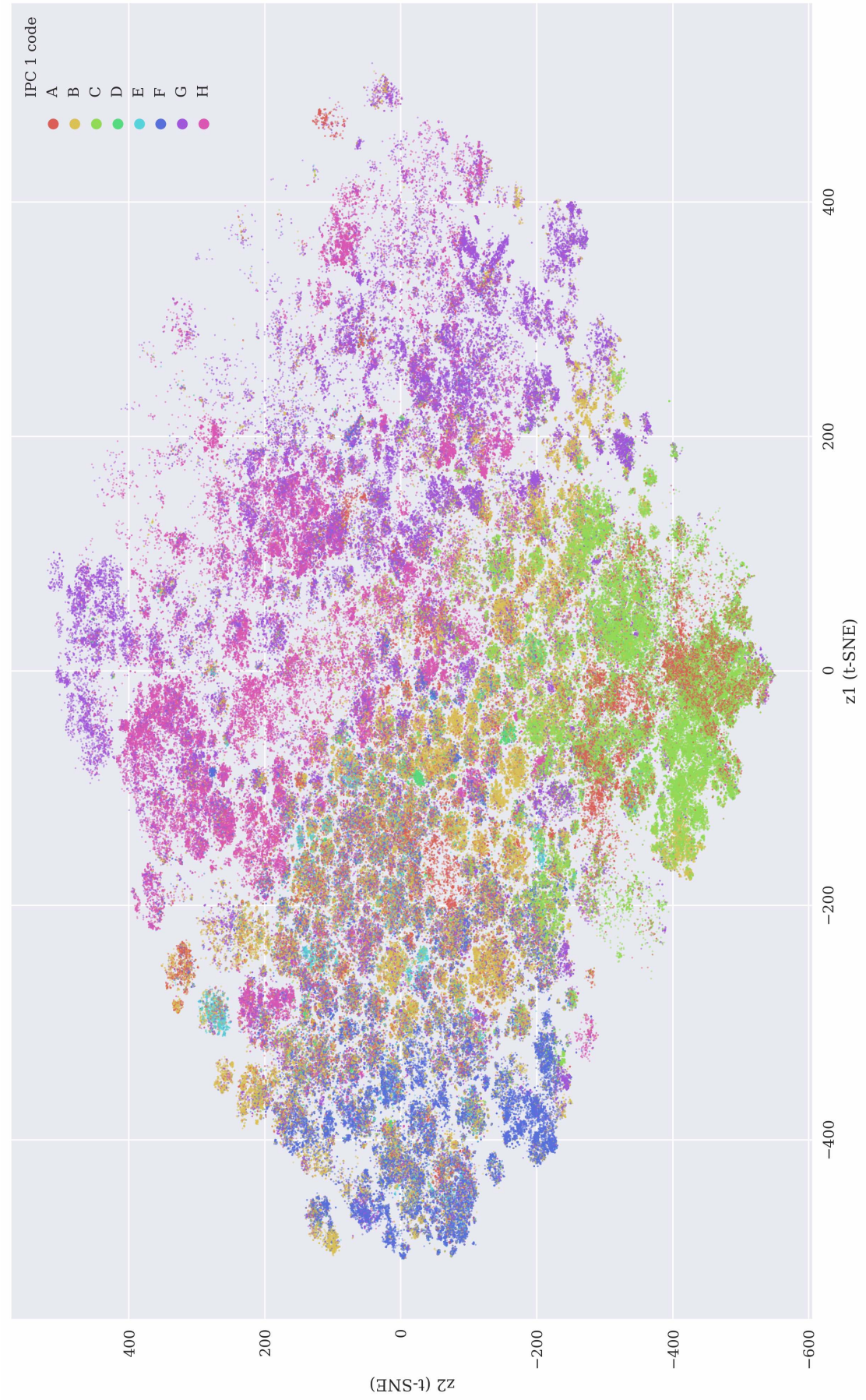


Figure 1.11: IPC 1 coloured t-SNE representation of patents from 2006 to 2015

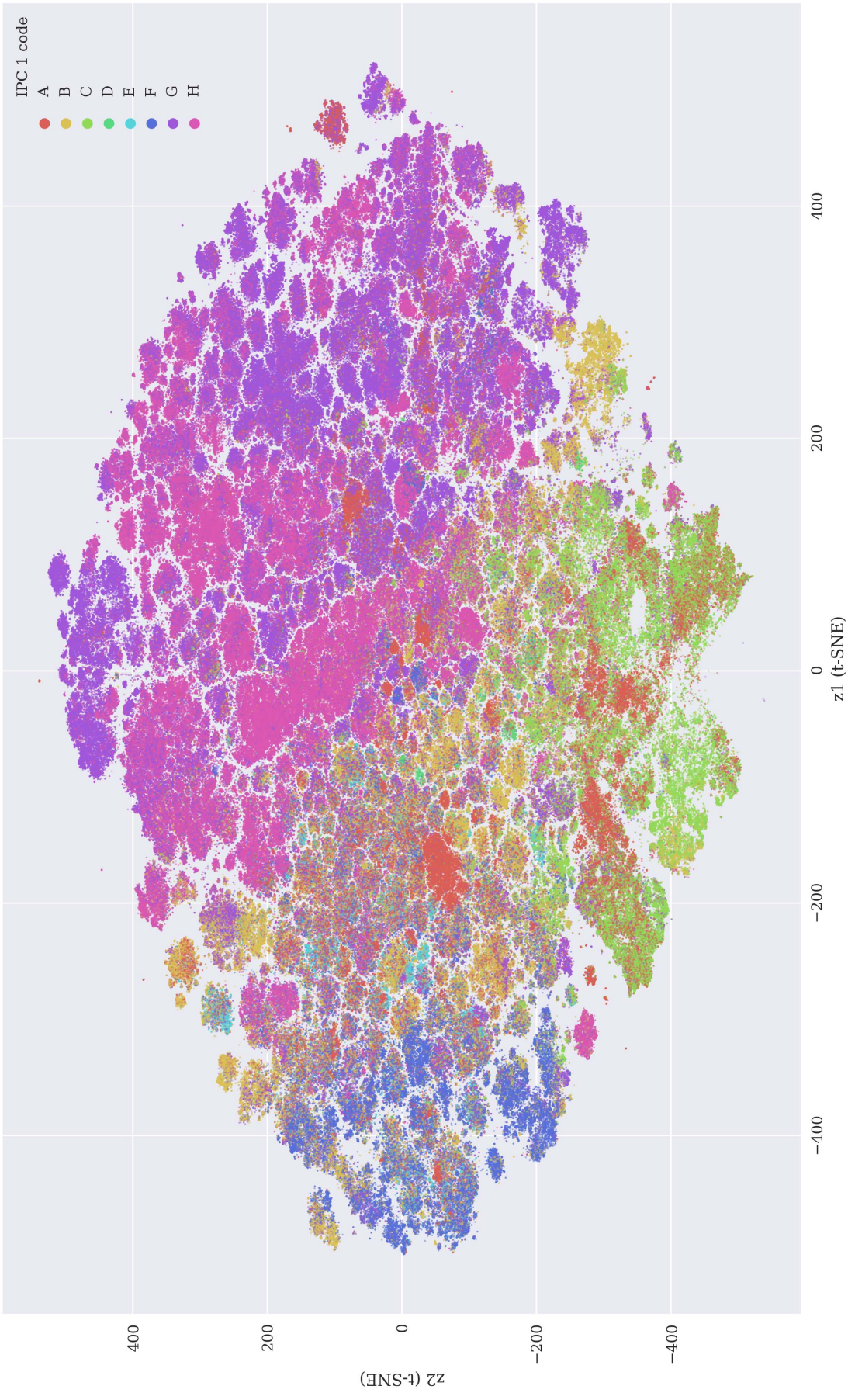


Table 1.6: 10-year citations and no-centroid score

	Dependent variable: 10-year citations					
	Whole sample			*Until 2000*		
	(1)	(2)	(3)	(4)	(5)	(6)
No-centroid core, standardised	4.026***	3.395***	3.429***	4.334***	3.488***	3.368***
	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
Constant	-133.9***			-145.1***		
	(0.000)			(0.000)		
Year FE		✓	✓		✓	✓
IPC 3 FE		✓			✓	
Firm FE			✓			✓
Adjusted R^2	0.044	0.075	0.163	0.067	0.119	0.219
Within R^2		0.024	0.026		0.038	0.039
Observations	2896300	2896013	1025243	1545952	1545675	602117

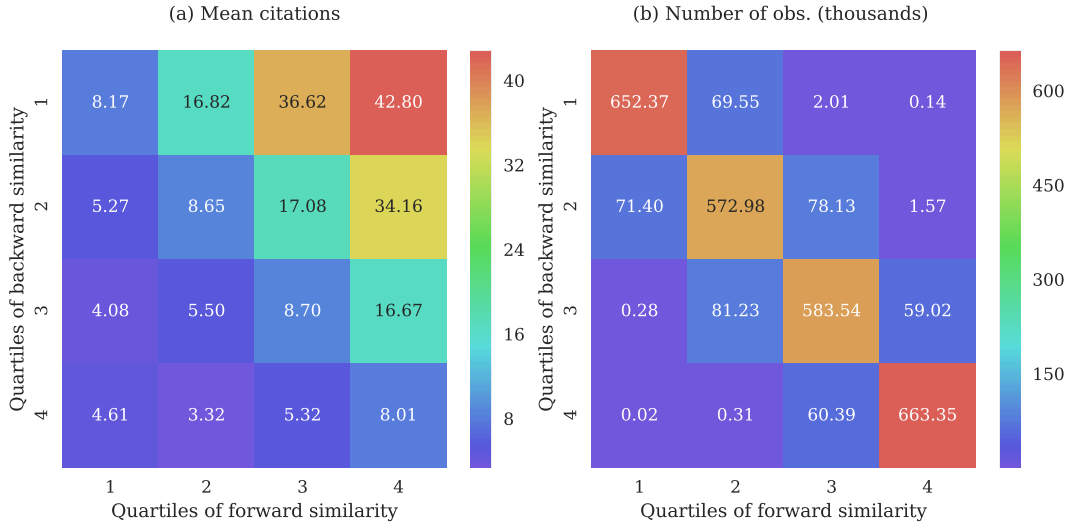
Notes: *: p<0.1, **: p<0.05, ***: p<0.01. P-values from standard errors clustered at the filing year level in parenthesis.

citations and no-centroid scores is driven by wideners, Figure 1.12 reports the average citations for patents in different quintiles of the distributions of backward and forward similarities. Citations increase as a patent is dissimilar in the backward space, and similar in the forward space.

Firm level performance The results can be found in Figure 1.13.[20] Firms that file a high-score patent see their profits, output and capital grow faster than their peers within the same industry and years. There is however very strong evidence of existing pre-trends, as suggested by significantly negative coefficients before the event date (the filing of the high-score patent). This means that those firms were on a differential trend before the invention came out. As mentioned earlier, this renders causal statements about the effect of filing a high-score patent impossible. It seems that we identify fast-growing firms that innovate and continue to grow quickly relative to other firms after the patent has been filed. It is however reassuring that the effects seem to be stronger than when high-score patents are identified using macro and IPC 3 scores, since the no-centroid scores are identifying *widening* patents, which may be rarer inventions, whereas the other two scores merely identify patents that anticipated a general trend in innovation. However, the firms with high no-centroid scores are often IT firms. Table 1.11 lists the firms with most top patents in the overall distribution of no-centroid scores. They are all IT-related.

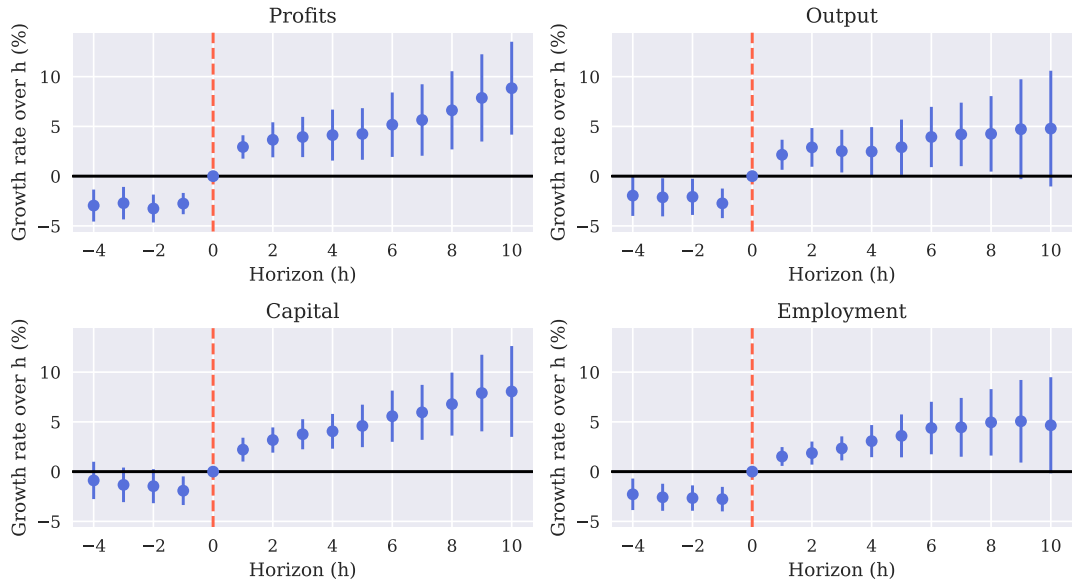
[20]The results estimated using the alternative specification of (1.4) with $\log Y_{fi,t+h} - \log Y_{fi,t}$ as dependent variable can be found in Figure 1.26

Figure 1.12: Mean citations per decile of backward and forward no-centroid similarity



Note: Panel (a) displays the mean citations of patents whose backward and forward similarity scores fall in the corresponding combinations of quartiles of the distributions of scores, and panel (b) reports the underlying number of observations. A high value of the quartile means higher similarity. The values on the y-axis are swapped (i.e. they range from 4 to 1).

Figure 1.13: No-centroid patents and firms dynamics



Note: Estimates from Equation (1.4) using the no-centroid score to qualify top patents. 95% confidence intervals are depicted.

It is interesting to note that differential pre-trends can also be observed when considering the filing of top patents in terms of citations and private value. Specifically, estimating Equation (1.4) classifying top patents as being in the top 5% of the private value distribution or in the top 0.1% of the citations distributions yield pre-trends in both cases (especially marked in the case of private value) as reported in Figures 1.30 and 1.31. This problem in estimating the causal effect of top patents — irrespective of the criteria used to qualify them — seems to be a recurrent issue.

1.4.3 Heterogenous areas of innovation and general purpose technologies

Given that we have seen how mixed patent content is in some areas of the space, it seems natural to try using such a methodology to think about general purpose technologies (GPTs). Do GPT-type of patents lie in very diverse areas of language spaces, surrounded by patents from many different IPC codes?

1.4.3.1 Geometry

The t-SNE visualisations in Figures 1.10 and 1.11 indicate that there exist some areas in high dimensional space which are shared by many patent classes. Patents in these areas therefore very likely use similar language. Building on the visual intuition, we now try to compute numerical counterparts of this observation. For a given patent, we again look at the top 100 most similar patents 0-9 years after its publication and we measure from how many different IPC classes these most similar patents come (see the description in Section 1.3.1.3 for full details). To remain consistent with our two exemplary t-SNE plots, we look at patents published in 1985 and in 2006. These are years contained in the two plots, however, they are also years for which we have full 10 year forward and backward intervals of data and hence can compute scores as well (our patent data range from 1976 to 2017).[21]

The following two heatmap tables in Figure 1.14 show the key results of our computation. Each row is an IPC 1 class of a patent for which we compute the top 100 most similar patents. The columns indicate the average amount of patents from different IPC classes among these top 100 most similar patents. For

[21] In principle, the *forward neighbourhood heterogeneity scores* can be computed for every patent in our sample. We plan to do so in the future.

example, the cell in row 3 column 7 of the 1985 patents indicates that on average, 3.48 patents in the most similar 100 patents around a chemistry patent came from electronics. A few outcomes are particularly noticeable. First, all but D & E have maximum values on the diagonal. Hence, on average the majority of their 100 most similar patents comes from their own IPC 1 code. Second, however, the top 100 neighbourhoods are much more diverse than what one might intuitively expect. Yet, they also resemble what the t-SNE suggested: A, B, D, E, and F have more diverse patent classes in their surrounding. Chemistry (C) and technology patents (H and G) are more insular. The time dimension from 1985 to 2008 is also interesting. IPC classes which contain technology patents in fact became more specialised over time - the distributions of H and G have more weight on their own class in 2006 than in 1985.

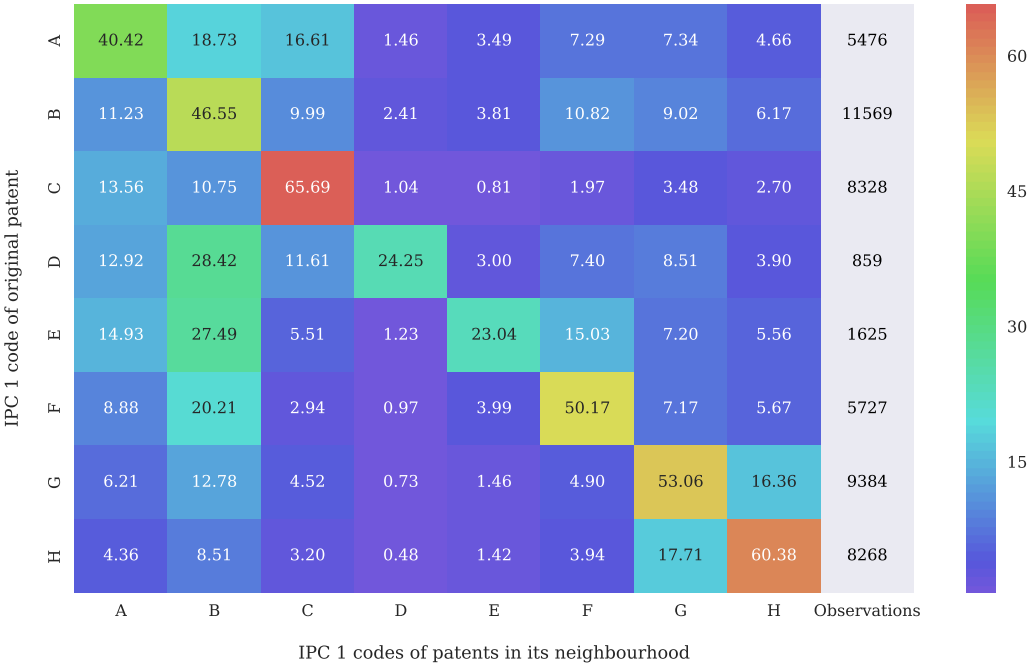
Equipped with the findings from describing neighbourhoods for average patents, we include this information into our regressions on citations as well — although we only have two years of data as of now. For this we first need to compute one value summarising the neighbourhood of each patent, because citations are at the patent level too. We use the approach introduced in Section 1.3.1.3, i.e. *forward neighbourhood heterogeneity*. A score of 0 means that the patents neighbourhood only consists of 100 patents from a single IPC 1 code - the patent is likely about a very specialised topic. A score of 0.9354 means that the patent's neighbourhood includes patents from all 8 IPC 1 codes in exactly the same proportions. We compute this value for every patent in 1985 and 2006. Table 1.10 in the Appendix shows this score averaged for patents in the two years. Again the outcomes are supporting the graphical intuition. Chemistry and electronics are more specialised than the other IPC 1 codes (their *forward neighbourhood heterogeneity* is lower) and electronics even became more specialised over the years contrary to some other classes.

The next section adds this new variable to our regressions. Now that we can measure the heterogeneity in the neighbourhood of a patent, we are interested in whether a heterogeneous language neighbourhood correlates positively or negatively with citations.

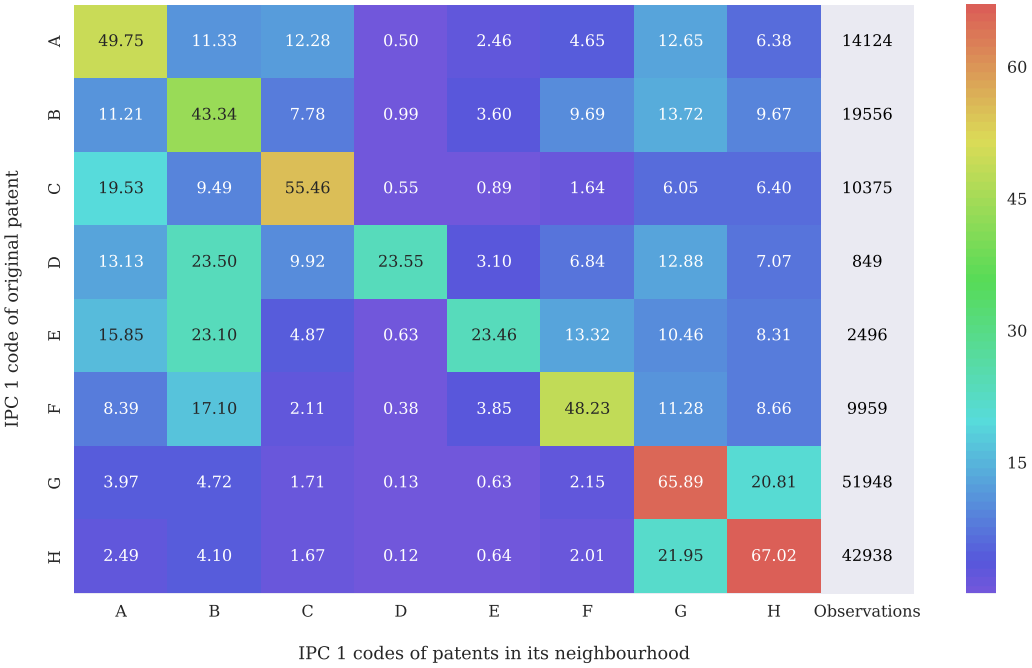
1.4.3.2 Regressions

This section shows regression results on citations including the forward neighbourhood heterogeneity of a patent. In summary, a patent's neighbourhood heterogeneity is consistently negatively associated with citations, not just in more

Figure 1.14: Average patent neighbourhood heterogeneity for each IPC 1 code



(a) IPC distributions in patent neighbourhoods 1976 - 1985



(b) IPC distributions in patent neighbourhoods 2006 - 2015

Table 1.7: Forward neighbourhood heterogeneity score and citations, 1985 and 2006

	(1) 10-year citations	(2) Log(1+10-year citations)
No-cent. score, std	1.781***	
	(0.000)	
For. Het. score, std	-0.607***	
	(0.000)	
Log(no-cent. score)		6.934***
		(0.000)
Log(For. Het. score)		-0.0576***
		(0.000)
Year FE	✓	✓
IPC 3 FE	✓	✓
Adjusted R^2	0.021	0.064
Within R^2	0.009	0.025
Observations	203432	199422

Notes: *: p<0.1, **: p<0.05, ***: p<0.01. P-values from robust standard errors in parenthesis. Standard errors can not be clustered at the filing year level since there are only two years in the data underlying this regression.

specialised areas such as Chemistry and Electronics, but across all IPC codes. Patents in more heterogeneous neighbourhoods are cited significantly less. See Tables 1.7 for the results across IPCs, and Tables 1.18 and 1.19 for IPC 1-specific regressions. On average, one standard deviation increase in neighbourhood heterogeneity yields 0.6 less citations. On average, it seems to pay off to be in a highly specialised area of space, although the effect is not large — for instance compared to the coefficient on the no-centroid scores.[22] The insight that areas of space which use words shared by all 8 fields tend to have lower citations is interesting in itself and not obvious ex ante. With lower citations, however, patents in heterogenous neighbourhoods can hardly be key technologies such as GPTs. Being in a part of space with words shared by many IPC codes might rather mean that topics can be imprecise or irrelevant. In any case, it is not a sufficient condition to identify GPTs.

Ideas for future research Language and word linkages applied in the right way might nonetheless offer some interesting clues towards identifying GPTs: Hardly anyone would cite Nash's original paper, but the Nash equilibrium is arguably as close to a GPT in economics as it gets and the words are mentioned in many papers. This could complement methods that use citations to qualify

[22]The regressions have also been run without including the no-centroid score, and the estimated coefficients on the forward neighbourhood heterogeneity score are unchanged.

the generality of a patent.[23] Yet, giving patents the highest scores which have neighbourhoods consisting of all 8 IPC codes in equal proportions might not be the most accurate measure. In future research, it could be interesting to identify areas consisting of only e.g. 3 out of 8 IPC codes and then to look into the patents in those areas that have the highest amounts of citations. Other avenues for research into this direction would be to look at smaller grain neighbourhoods. For example, if we were to look into neighbourhoods only within electronics patents and detect patents which have neighbours from many electronics sub-fieds, could these inventions in fact serve a more general purpose?[24] One could also study trends in the specialization of inventions over time and check whether modern inventions are more specialised than in the past.

1.4.4 Relationships between different scores

With Section 1.4.1 using centroid-based scores (global and IPC3) and Sections 1.4.2 and 1.4.3 using no centroid scores, we conclude the results part with some comparisons between scores (we leave out the heterogeneity scores because we only computed them for 1985 and 2006). Recall that macro, IPC and no-centroid scores are calculated for every patent in our sample between 1985 and 2008. As explained in Section 1.3.1.3, the scores are the ratio of a forward similarity and a backward similarity, which are essentially similarities to a reference point capturing the existing technological states at different point in time. Scores are obtained by dividing the forward similarity by the backward similarity. A patent dissimilar to existing patents at the time of filing and similar 10 years later will have a high score.

Table 1.8 reports some descriptive statistics. First note that both the mean and the median scores are close to 1, which means that on average the position of a patent in the backward and forward spaces is very similar. Forward and backward similarities are indeed very highly correlated — see Figure 1.18. The scores are positively correlated with each other with correlations ranging from 0.33 to 0.46. A correlogram of the scores containing the histograms of each score distribution and scatter plots of the scores against each other can be found in Figure 1.19. Even though all these scores capture similar notions, they differ markedly from

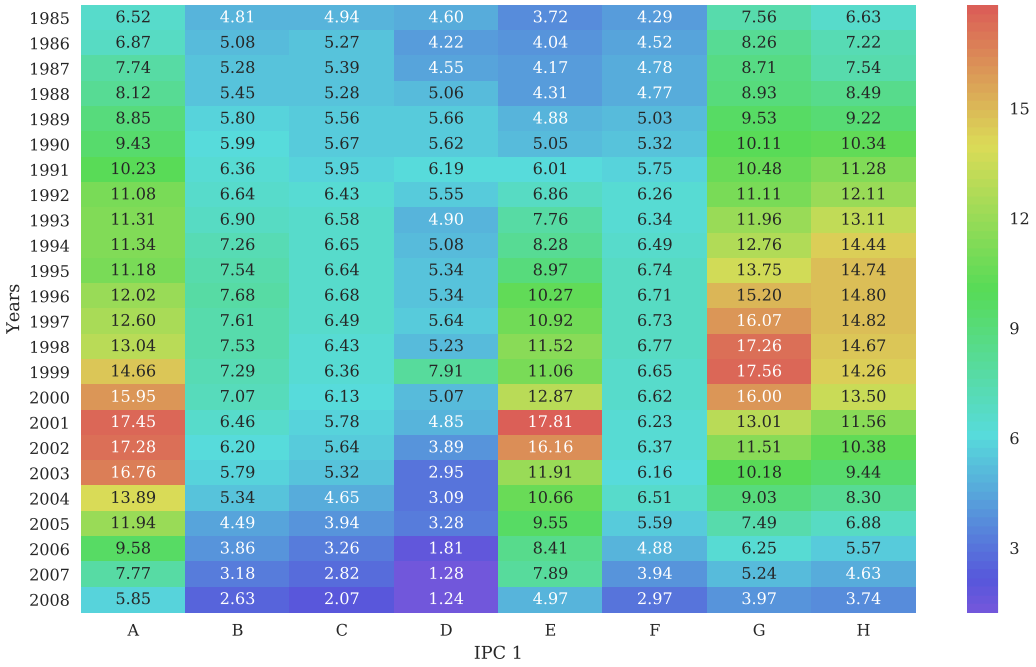
[23]For instance, Moser and Nicholas (2004) use the diversity of the technological fields of citing patents as a measure of generality and find that electricity was not a GPT according to this definition.

[24]Those would not be General Purpose Technologies as defined in the literature however, since GPTs must spread and spawn innovation from most or all broadly defined sectors of the economy.

Table 1.8: Descriptive statistics of innovation scores

	N	Mean	Sd	Min	p25	Median	p75	p90	Max
Macro score	2896300	.994	.152	.093	.887	.966	1.08	1.19	3.06
IPC 3 score	2745260	.997	.135	.0899	.927	.993	1.06	1.14	4.74
No-centroid score	2896300	1.01	.0286	.551	.998	1.01	1.02	1.05	2.36

Figure 1.15: Citations



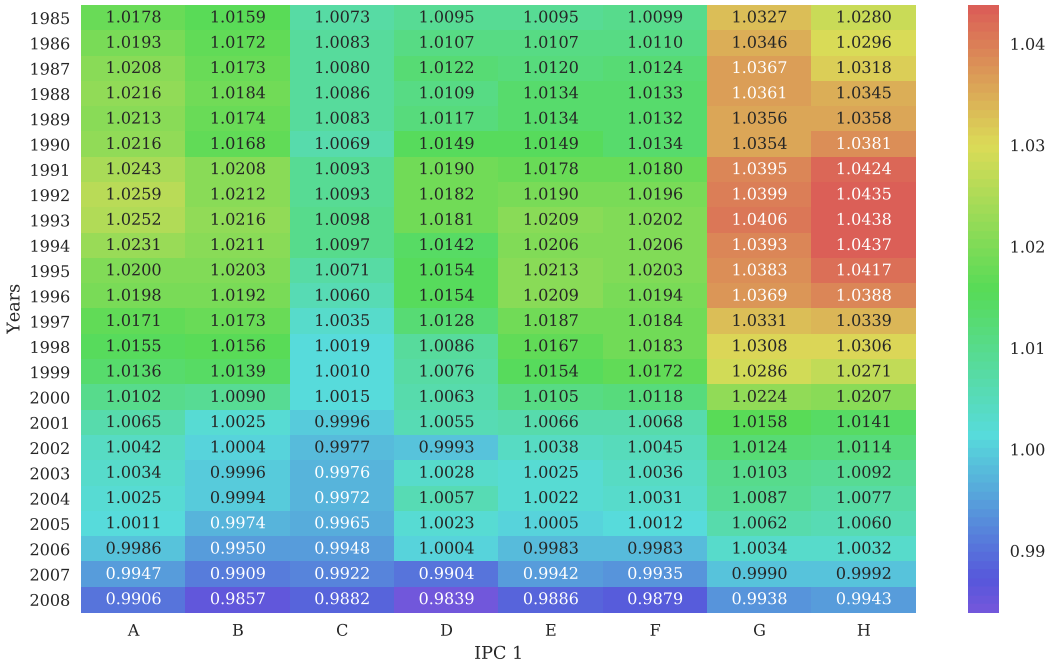
each other.

1.5 Discussion

An approach which identifies patents as innovative that are dissimilar to past language but similar to future language will tend to give highest scores to those patents which were in very empty parts of the space when they got released. A key question is therefore which industries we identify as being particularly innovative. Our t-SNE driven intuition suggests that patents with high scores will come in large proportions from electronics and physics fields as these areas have been the emptiest in the early space of innovation. The heatmaps in Figures 1.15 and 1.16 confirm this intuition. The former shows citations per year-IPC 1 combination and the latter displays their no-centroid scores. As expected, our no-centroid score is highest in IT-related areas, which is where the space must have been the emptiest.

Figures 1.32, 1.33, and 1.34 in the Appendix show the same for macro and IPC 3

Figure 1.16: No centroid scores



scores, as well as depict the number of patent per cell. For the macro centroids, IT patents have even more pronounced scores reflecting that the economy moved to exactly this macro technology over the time span we considered. For IPC 3 centroids on the other side, scores are more balanced as each IPC code now has its own reference point. To conclude, it seems that methods such as the ones discussed in this — as well as other related papers, in all likelihood — identify the IT revolution. We see a decline in how well we fit citations in the more recent years from 2000 onwards. This might be driven by the fact that as of now also the IT space is crowded and a method like ours has less bite than before.

The importance of IT One possible worry is that most of the results are driven by patents relating to the IT sector. At the macro level, IT is one important sector that was nascent in the 1980s, and became central in the subsequent decades. At the IPC level, IPC G and H — Physics and Electricity, the IPC seemingly most related to IT — account for over 50% of granted patents in our sample. Regarding *widening* patents, we also find that large IT firms disproportionately file high-score patents — see Table 1.11. It is therefore natural to wonder whether the results are robust to excluding patents from these technological areas. We do so by dropping all patents from IPC codes G and H once the scores have been computed, and re-estimate all our results.[25] In general, the

[25]We do not, however, re-estimate the vector representation of patents dropping IPC codes G and H, i.e. we use the same score as in the main text, simply omitting patents from those

results are qualitatively unchanged, although in most cases the magnitudes of the effects are smaller — they sometimes as much as halve. The patent-level regression results can be found in Tables 1.20, 1.21 and 1.22 for macro, IPC 3 and no-centroid scores, respectively. The decrease in magnitude is strongest for the macro scores: in the citations regressions, both coefficients and r-squared decrease by around 50%. For results based on IPC 3 and no-centroid scores, the decrease in magnitude is smaller. At the firm level, the estimated effects of filing a top patent are generally smaller and less significant, but qualitatively similar as can be seen in Figures 1.27, 1.28 and 1.29.[26] One marked difference is the relationship between private value and scores, which is most cases becomes negative. It is unclear why that is so: It seems that the positive relationship between scores and private value was solely driven by patents from IPC codes G and H — it is not the case that these patents have systematically higher values than those in other fields, however. Overall, it seems that IT alone is indeed responsible for some, yet not all of the results presented.

Construction of the scores The no-centroid scores are built using the proximity of patents to their 100 most similar neighbours in space. This cutoff is arbitrary and entails a trade-off. On the one hand, taking the distance to more patents, e.g. 500, might dilute the information that we attempt to capture about the proximity of the closest inventions by taking the average distance over too many patents. On the other hand, choosing too few patents, e.g. 10, may also give a misleading idea of the position of a patent relative to its neighbours — e.g. if the 10 are very close and filed by the same firm, and the 11th is really far. We chose 100 as it seemed to have the best fit with citations, but the results are similar when using alternative thresholds.[27] Furthermore, it is not obvious that this threshold must remain constant over time. Since the space becomes increasingly crowded over time as the volume of patents increases, the closest 100 to a patent in 1985 may be further away than the closest 100 to a patent in 2008. An alternative could be to let the threshold change over time proportionally to the size of the corpus in each year. Yet, in turn this would give many degrees of freedom and introduce some problems of over-fitting the data. Irrespective of which amount of neighbours is chosen, however, it seems likely that the increased crowdedness of space due to many more patents being published in more recent years mechanically influences the results.

IPC codes.

[26] The differential pre-trend issue is also less severe.

[27] We tried 50, 100, 200 and 500.

1.6 Conclusion

In this paper, we apply methods from natural language processing and machine learning to analyse the innovative content of patents based on their text. Numerical representations of patents as high dimensional vectors allow us to compute similarities between documents at a large scale. We structure our thoughts around the idea that a patent that is dissimilar to the state of knowledge when it is filed, yet similar to it subsequently, could be successful. Our measures of success are a patent's citations, its private value and its effect on the filing firm's performance indicators. We find that patents which anticipate trends in the economy or within a narrow technological field are more successful, as are those that *widen* areas in the knowledge space. The causal interpretation of the effect of filing such patents on firm's outcomes is rendered difficult due to the existence of differential pre-trends: Based only on the language of patents, we identify successful firms, and these firms also have been successful some years before. Furthermore, we find that patents which are surrounded by other inventions from diverse technological fields tend to have less citations than specialised patents. We conclude by arguing that methods such as ours capture the IT revolution.

References

Balsmeier, Benjamin, Mohamad Assaf, Tyler Chesebro, Gabe Fierro, Kevin Johnson, Scott Johnson, Guan-Cheng Li, Sonja Lück, Doug O'Reagan, Bill Yeh et al., "Machine learning and natural language processing on the patent corpus: Data, tools, and new measures," *Journal of Economics & Management Strategy*, 2018, *27* (3), 535–553.

Bowen, Donald E, Laurent Frésard, and Gerard Hoberg, "Technological disruptiveness and the evolution of IPOs and sell-outs," 2018.

Breschi, Stefano, Franco Malerba, and Luigi Orsenigo, "Technological regimes and Schumpeterian patterns of innovation," *The economic journal*, 2000, *110* (463), 388–410.

Bresnahan, Timothy, "General purpose technologies," in "Handbook of the Economics of Innovation," Vol. 2, Elsevier, 2010, pp. 761–791.

Crafts, Nicholas, "Steam as a general purpose technology: a growth accounting perspective," *The Economic Journal*, 2004, *114* (495), 338–351.

Deerwester, Scott, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman, "Indexing by latent semantic analysis," *Journal of the American society for information science*, 1990, *41* (6), 391–407.

Dittmar, Jeremiah E, "Information technology and economic change: the impact of the printing press," *The Quarterly Journal of Economics*, 2011, *126* (3), 1133–1172.

Elman, Gerry J, "Automated Patent Examination Support—A Proposal," *Biotechnology Law Report*, 2007, *26* (5), 435–436.

Hall, Bronwyn H, Adam Jaffe, and Manuel Trajtenberg, "Market value and patent citations," *RAND Journal of economics*, 2005, pp. 16–38.

Helpman, Elhanan, *General purpose technologies and economic growth*, MIT press, 1998.

Jovanovic, Boyan and Peter L Rousseau, "General purpose technologies," in "Handbook of economic growth," Vol. 1, Elsevier, 2005, pp. 1181–1224.

Katila, Riitta and Gautam Ahuja, "Something old, something new: A longitudinal study of search behavior and new product introduction," *Academy of management journal*, 2002, *45* (6), 1183–1194.

Kelly, Bryan, Dimitris Papanikolaou, Amit Seru, and Matt Taddy, "Measuring technological innovation over the long run," Technical Report, National Bureau of Economic Research 2018.

Klette, Tor Jakob and Samuel Kortum, "Innovating firms and aggregate innovation," *Journal of political economy*, 2004, *112* (5), 986–1018.

Kline, Patrick, Neviana Petkova, Heidi Williams, and Owen Zidar, "Who profits from patents? rent-sharing at innovative firms," *The Quarterly Journal of Economics*, 2019, *134* (3), 1343–1404.

Kobak, Dmitry and Philipp Berens, "The art of using t-SNE for single-cell transcriptomics," *bioRxiv*, 2019.

Kogan, Leonid, Dimitris Papanikolaou, Amit Seru, and Noah Stoffman, "Technological innovation, resource allocation, and growth," *The Quarterly Journal of Economics*, 2017, *132* (2), 665–712.

Linderman, George C., Manas Rachh, Jeremy G. Hoskins, Stefan Steinerberger, and Yuval Kluger, "Fast interpolation-based t-SNE for improved visualization of single-cell RNA-seq data," *Nature Methods*, 2019, *16* (3), 243–245.

Lodh, Suman and Maria Rosa Battaggion, "Technological breadth and depth of knowledge in innovation: the role of mergers and acquisitions in biotech," *Industrial and Corporate Change*, 2014, *24* (2), 383–415.

Malerba, Franco and Luigi Orsenigo, "Schumpeterian patterns of innovation," *Cambridge journal of Economics*, 1995, *19* (1), 47–65.

Moorthy, Subba and Douglas E Polley, "Technological knowledge breadth and depth: performance impacts," *Journal of Knowledge Management*, 2010, *14* (3), 359–377.

Moser, Petra and Tom Nicholas, "Was electricity a general purpose technology? Evidence from historical patent citations," *American Economic Review*, 2004, *94* (2), 388–394.

Nicholas, Tom, "Does innovation cause stock market runups? Evidence from the great crash," *American Economic Review*, 2008, *98* (4), 1370–96.

Ozman, Müge, "Breadth and depth of main technology fields: an empirical investigation using patent data," *Science and Technology Policies Research Centre, Working Paper Series*, 2007, *7* (01).

Packalen, Mikko and Jay Bhattacharya, "New ideas in invention," Technical Report, National Bureau of Economic Research 2015.

Rosenberg, Nathan and Manuel Trajtenberg, "A general-purpose technology at work: The Corliss steam engine in the late-nineteenth-century United States," *The Journal of Economic History*, 2004, *64* (1), 61–99.

Strang, Gilbert, *Introduction to Linear Algebra*, 5 ed. 2016.

van der Maaten, Laurens and Geoffrey Hinton, "Visualizing data using t-SNE," *Journal of machine learning research*, 2008, *9* (Nov), 2579–2605.

1.7 Appendix

1.7.1 Exemplary full patent text

```
############################################################
##  Reference patent: 8919201 from 2012
############################################################

## Abstact

An acceleration measuring apparatus that can easily detect
    acceleration with high accuracy is provided. In the
    apparatus, positional displacement of a swingable
    pendulum member is detected, feedback control is
    performed to maintain the pendulum member in a
    stationary state using an actuator, and acceleration is
    measured by measuring the output of the actuator at
    this time. A movable electrode is provided to the
    pendulum member, and a loop is formed in which a fixed
    electrode provided to oppose the movable electrode, and
    an oscillating circuit, a crystal unit, and the
    movable electrode are electrically connected in series.
    By measuring an oscillating frequency of the
    oscillating circuit at this time, a change in the size
    of a variable capacitance formed between the movable
    electrode and the fixed electrode is detected, and
    thereby the positional displacement of the pendulum
    member is detected.


## Brief description

CROSS-REFERENCE TO RELATED APPLICATION
```

This application claims the priority benefit of Japanese application serial no. 2011−127644, filed Jun. 7, 2011. The entirety of the above−mentioned patent applications is hereby incorporated by reference herein and made a part of specification.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to a technology for detecting acceleration based on an oscillating frequency using a piezoelectric plate such as a crystal plate.

2. Description of Related Art

In order to measure an earthquake or the like, one important issue is detecting weak and low frequency acceleration. It is desirable to measure with high accuracy using a structure that is as simple as possible when carrying out this kind of measurement. As a sensor for detecting weak and low frequency acceleration, a servo−type acceleration measuring apparatus is often used.

In general, a servo−type acceleration measuring apparatus is constituted by a pendulum, a pendulum position detector, an actuator that applies a force to the pendulum, and a regulating unit that controls the actuator based on a detection result by the pendulum position detector. The pendulum is constituted by a spindle and a spring. One end of the spring is fixed to a container of the acceleration measuring apparatus, and the position of the spindle is displaced relative to the container by the action of an inertial force when acceleration is exerted on the acceleration measuring apparatus. A resonance frequency of the pendulum is set extremely low, and even a slight acceleration largely displaces the pendulum. The displacement of the pendulum relative to the container is proportional to the exerted acceleration in a frequency range that is lower than the resonance frequency of the pendulum. The pendulum position detector is a sensor that detects positional

displacement of the pendulum relative to the container. The actuator consists of a coil provided to the pendulum and a magnetic circuit provided to the container. The position of the pendulum can be displaced by an electromagnetic force that is generated when a current is applied to the coil. The regulating unit is for applying a current to the coil of the actuator based on positional displacement data of the pendulum obtained by the pendulum position detector.

If acceleration from an outside force is exerted on the acceleration measuring apparatus, the position of the pendulum is displaced by an inertial force. At this time, a current is applied to the actuator from the regulating unit, and by exerting an electromagnetic force that is the same size as the inertial force but is in the opposite direction on the pendulum, the pendulum can be maintained in a stationary state. Therefore, by detecting the positional displacement of the pendulum with the pendulum position detector and then operating the actuator so that the positional displacement is zero, the positional displacement of the pendulum can be feedback controlled. By measuring the output of the actuator at this time, for example by measuring the current value applied to the coil, the acceleration of the outside force can be measured. This kind of servo−type acceleration measuring apparatus has characteristics of high accuracy and high resolution and is capable of measuring frequencies of about 0 to 400 Hz.

As a method for detecting the position of the pendulum in a servo mechanism of a servo−type acceleration measuring apparatus, an optical method and a capacitor method are mainly used. Among these, an optical−type pendulum position detector uses a laser diode, a two−element segmented photodiode, and a lens. As a method for detection using an optical−type pendulum position detector, a type of differential method is employed. However, an optical−type pendulum position detector presents problems in that it has a complex structure

and the life of the photodiode is short. A capacitor-type pendulum position detector has a structure in which a capacitance is formed such that it changes by the positional displacement of the pendulum, and the pendulum position is detected by detecting the change in capacitance. However, the influence of noise cannot be eliminated, and thus it is difficult to detect the acceleration with high accuracy.

Patent Document 1 discloses an acceleration sensor in which a constant current is made to flow through a movable electrode, and acceleration is detected by detecting the number of pulses of an induced current generated in a fixed electrode that opposes the movable electrode. However, this acceleration sensor is different from the present invention. Patent Document 2 discloses a capacitance change detection-type acceleration sensor, in which two variable capacitances are formed between a movable center plate and fixed plates provided on both sides of the movable center plate, and antiphase pulse voltages are respectively applied to the two fixed plates. Both variable capacitances change when the position of the center plate is displaced due to the generation of acceleration, and the acceleration is detected at this time by detecting a shift in the pulse phase of the voltage applied from the fixed plates to the center plate. However, this capacitance change detection-type acceleration sensor is different from the present invention.

[Patent Document 1] Japanese Patent Application Laid-Open (JP-A) No. H7-167885

[Patent Document 2] Japanese Patent Application Laid-Open (JP-A) No. 2004-198310

SUMMARY OF THE INVENTION

The present invention was created in light of the above-described background, and an objective thereof is to

provide an acceleration measuring apparatus that can easily detect acceleration with high accuracy.

The acceleration measuring apparatus of the present invention detects a displacement from a reference position of a pendulum member that is about to swing due to an inertial force, applies an external force to the pendulum member by an operating unit based on a detection result of the displacement to immobilize the pendulum member in the reference position, and evaluates an acceleration acting on the pendulum member based on a size of the external force at this time, the apparatus comprising:

a piezoelectric plate;

a first drive electrode and a second drive electrode provided respectively on a first surface side and a second surface side of the piezoelectric plate in order to vibrate the piezoelectric plate;

an oscillating circuit that is electrically connected to the first drive electrode;

a movable electrode for forming a variable capacitance that is provided on the pendulum member and the movable electrode being electrically connected to the second drive electrode;

a fixed electrode separated from the pendulum member, provided so as to oppose the movable electrode, and connected to the oscillating circuit, the fixed electrode forming a variable capacitance upon a change in capacitance between the fixed electrode and the movable electrode due to swinging of the pendulum member; and

a frequency information detecting unit for detecting a signal that is frequency information corresponding to an oscillating frequency of the oscillating circuit,

wherein an oscillation loop is formed beginning from the oscillating circuit, passing through the first drive electrode, the second drive electrode, the movable electrode, and the fixed electrode, and then returning to the oscillating circuit, and

the frequency information detected by the frequency information detecting unit is for evaluating the displacement from the displacement position of the pendulum member.

Further, in the acceleration measuring apparatus of the present invention,

the movable electrode includes a first movable electrode and a second movable electrode provided so as to sandwich the pendulum member and oppose each other in a direction in which the inertial force acts;

the fixed electrode includes a first fixed electrode and a second fixed electrode provided so as to be separated from the pendulum member and respectively oppose the first movable electrode and the second movable electrode;

a switching unit is further provided, wherein the switching unit is capable of switching an electrical connection destination of the oscillating circuit between a first variable capacitance between the first movable electrode and the first fixed electrode and a second variable capacitance between the second movable electrode and the second fixed electrode; and

the frequency information detecting unit calculates information corresponding to a difference between an oscillating frequency corresponding to the first variable capacitance and another oscillating frequency corresponding to the second variable capacitance that are time-divided by the switching unit.

In addition, the pendulum member can be cantilevered at one end thereof by a supporting unit. Also, the pendulum member can be the piezoelectric plate, or can include the piezoelectric plate in a portion thereof.

The present invention captures the displacement when the pendulum member is displaced from a reference position upon swinging due to acceleration as a change in the oscillating frequency of the piezoelectric plate via a change in a capacitance between the movable electrode of the pendulum member and the fixed electrode that opposes the movable electrode. Therefore, the

acceleration can be easily detected with high accuracy. Further, by forming a variable capacitance on both sides relative to the direction in which the pendulum swings, a differential method can be applied when measuring a change in the oscillating frequency. Thus, the influence of noise and temperature characteristics can be suppressed, and the acceleration can be detected with even higher accuracy.

Claims text

1. An acceleration measuring apparatus comprising:
a pendulum member, extending in a vertical direction, wherein an upper end of the pendulum member is supported;
a piezoelectric resonator comprising a piezoelectric plate and a first drive electrode and a second drive electrode provided respectively on a first surface side and a second surface side of the piezoelectric plate in order to vibrate the piezoelectric plate;
an oscillating circuit, configured to oscillate the piezoelectric resonator;
a first movable electrode and a second movable electrode each respectively disposed on opposite surfaces of the pendulum member in a horizontal direction, configured to form variable capacitances;
a first fixed electrode separated from the pendulum member, provided so as to oppose the first movable electrode, configured to form a first variable capacitance upon a change in capacitance between the first fixed electrode and the first movable electrode due to swinging of the pendulum member;
a second fixed electrode separated from the pendulum member, provided so as to oppose the second movable electrode, configured to form a second variable capacitance upon a change in capacitance between the second fixed electrode and the second movable electrode

due to swinging of the pendulum member;

a switching unit, configured to alternately switch the connection of the first variable capacitance and the second variable capacitance with an oscillation loop comprising the oscillating circuit and the piezoelectric resonator;

a frequency information detecting unit, configured to calculates frequency information corresponding to a difference between an oscillating frequency corresponding to the first variable capacitance and another oscillating frequency corresponding to the second variable capacitance that are time-divided by the switching unit; and

an operating unit, configured to apply an external force to the pendulum member to immobilize the pendulum member in a reference position based on the frequency information detected by the frequency information detecting unit, the reference position is a stationary state in a vertical posture, and wherein an acceleration acting on a pendulum member is evaluated based on a size of the external force.

2. The acceleration measuring apparatus according to claim 1, wherein the pendulum member is the piezoelectric plate, or includes the piezoelectric plate in a portion thereof.

##

Clostest match: 8677828 from 2012

##

Abstact

Provided is a device capable of easily and accurately detecting a vibration period when, for example, an earthquake occurs. When a quartz-crystal plate bends upon application of a force, capacitance between a movable electrode provided at its tip portion and a fixed electrode provided on a vessel to face the movable electrode changes, so that an oscillation frequency of the quartz-crystal plate changes according to this capacitance. Therefore, when the vessel is vibrated, there appear a first state where the quartz-crystal plate ends to approach the fixed electrode and a second state where the quartz-crystal plate is in an original state or bends to be apart from the fixed electrode. Accordingly, an oscillation frequency corresponding to the first state and corresponding to the second state alternately appear, and therefore, it is possible to find the period (frequency) of the vibration.

Brief description

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to a technical field for detecting a vibration period by using a quartz-crystal resonator.

2. Description of the Related Art

There sometimes arises a need for detecting a period (frequency) of vibration applied to an object. For example, quick warning is necessary, for example, when an earthquake occurs. When the scale of the earthquake is large, its vibration frequency is about 0.01 Hz to about 30 Hz and is lower than vibration caused by daily life vibration, and therefore, if its vibration frequency can be detected, it is possible to discriminate the earthquake from the daily life

vibration. However, detecting such a low frequency is difficult.

An object of the formation of capacitance in Patent Document 1 is to stabilize an oscillation frequency of a quartz-crystal resonator and is different from an object of the present invention.

Patent Document 1: Japanese Patent Application Laid-open No. Hei 07-131279

SUMMARY OF THE INVENTION

The present invention was made under such circumstances and has an object to provide a device capable of easily and accurately detecting a vibration period (vibration frequency).

The present invention is a device detecting a period of vibration of an object and an external force, including:

a piezoelectric plate;

a first excitation electrode and a second excitation electrode provided on one surface and another surface of the piezoelectric plate respectively to vibrate the piezoelectric plate;

an oscillator circuit electrically connected to the first excitation electrode;

a plate-shaped member provided in a vessel and having one end supported in a cantilever manner;

a movable electrode for variable capacitance formation provided at another end of the plate-shaped member and electrically connected to the second excitation electrode;

a fixed electrode provided in the vessel to face the movable electrode, connected to the oscillator circuit, and forming variable capacitance when capacitance between the fixed electrode and the movable electrode is varied due to bending of the plate-shaped member; and

a frequency information detecting part detecting a signal being frequency information corresponding to an oscillation frequency of the oscillator circuit,

wherein an oscillation loop is formed from the oscillator circuit back to the oscillator circuit through the first excitation electrode, the second excitation electrode, the movable electrode, and the fixed electrode, and

wherein vibration of the vessel produces a first state where the plate-shaped member bends toward the fixed electrode to approach the fixed electrode and a second state where the plate-shaped member is more apart from the fixed electrode than in the first state, and the frequency information is used for finding a period of the vibration by utilizing a fact that an oscillation frequency corresponding to the first state and an oscillation frequency corresponding to the second state alternately appear.

One form of this invention may be a structure where the plate-shaped member also serves as the piezoelectric plate.

Further, another form may be

a structure where, in the plate-shaped member, a portion where the movable electrode is provided is larger in thickness than a portion sandwiched by the first excitation electrode and the second excitation electrode, or

a structure where, in the plate-shaped member, a portion between a portion that the first excitation electrode and the second excitation electrode sandwich and the movable electrode is smaller in thickness than the sandwiched portion.

In the present invention, the vibration of the vessel produces the first state where the quartz-crystal plate bends toward the fixed electrode to approach the fixed electrode and the second state where the quartz-crystal plate is more apart from the fixed electrode than in the first state, and the oscillation frequency corresponding to the first state and the oscillation

frequency corresponding to the second state alternately appear. Therefore, it is possible to find the period (frequency) of the vibration based on a change between these oscillation frequencies.

Claims text

1. A vibration detecting device detecting a period of vibration of an object and an external force, comprising:

a piezoelectric plate;

a first excitation electrode and a second excitation electrode provided on one surface and another surface of the piezoelectric plate respectively to vibrate the piezoelectric plate;

an oscillator circuit electrically connected to the first excitation electrode;

a plate−shaped member provided in a vessel and having one end supported in a cantilever manner;

a movable electrode for variable capacitance formation provided at another end of the plate−shaped member and electrically connected to the is second excitation electrode;

a fixed electrode provided in the vessel to face the movable electrode, connected to the oscillator circuit, and forming variable capacitance when capacitance between the fixed electrode and the movable electrode is varied due to bending of the plate−shaped member; and

a frequency information detecting part detecting a signal being frequency information corresponding to an oscillation frequency of the oscillator circuit,

wherein an oscillation loop is formed from the oscillator circuit back to the oscillator circuit through the first excitation electrode, the second excitation electrode, the movable electrode, and the fixed electrode, and

wherein vibration of the vessel produces a first state where the plate-shaped member bends toward the fixed electrode to approach the fixed electrode and a second state where the plate-shaped member is more apart from the fixed electrode than in the first state, and the frequency information is used for finding a period of the vibration by utilizing a fact that an oscillation frequency corresponding to the first state and an oscillation frequency corresponding to the second state alternately appear.

2. The vibration detecting device according to claim 1, wherein the plate-shaped member also serves as the piezoelectric plate.

3. The vibration detecting device according to claim 1, wherein, in the plate-shaped member, a portion where the movable electrode is provided is larger in thickness than a portion sandwiched by the first excitation electrode and the second excitation electrode.

4. The vibration detecting device according to claim 1, wherein, in the plate-shaped member, a portion between a portion that the first excitation electrode and the second excitation electrode sandwich and the movable electrode is smaller in thickness than the sandwiched portion.

5. The vibration detecting device according to claim 1, further comprising

on an internal wall part on a side where the fixed electrode is provided in the container, a projecting part allowing a contact of a portion shifted toward one

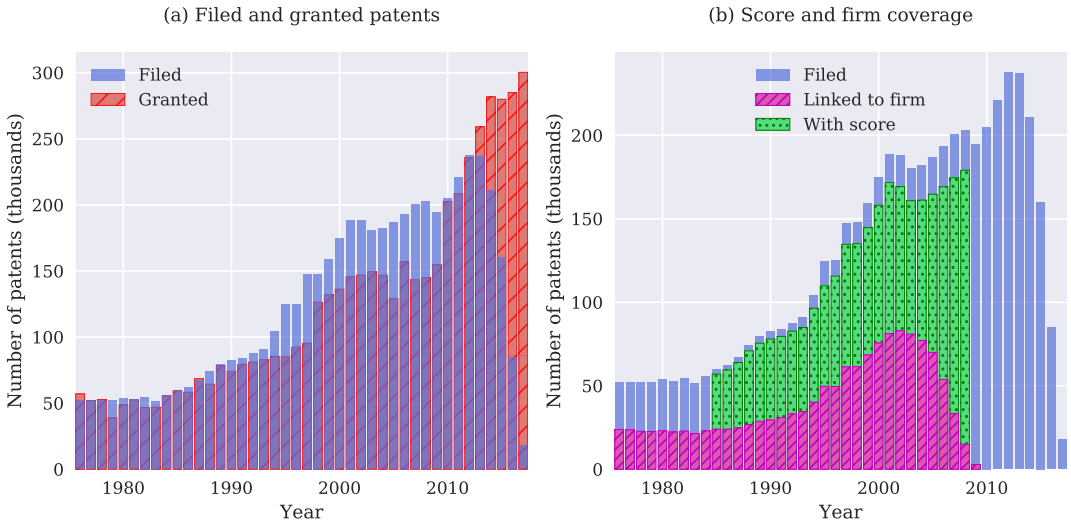
end side from the other end side of the plate-shaped member to restrict bending of this portion when the plate-shaped member bends excessively, thereby avoiding collision of the other end of the plate-shaped member with the inner wall part of the container.

6. The vibration detecting device according to claim 5, wherein

with respect to a face of the projecting part which faces the plate-shaped member, a vertical cross-sectional shape in a length direction of the piezoelectric piece is a mound shape.

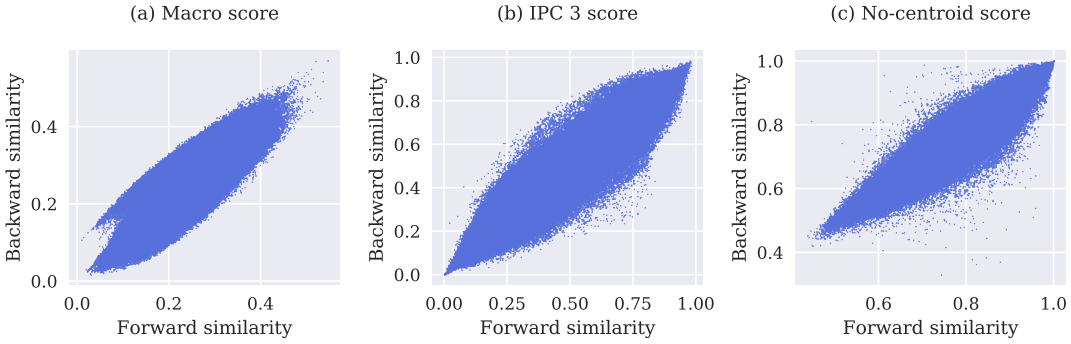
1.7.2 Figures

Figure 1.17: Number of patents over time



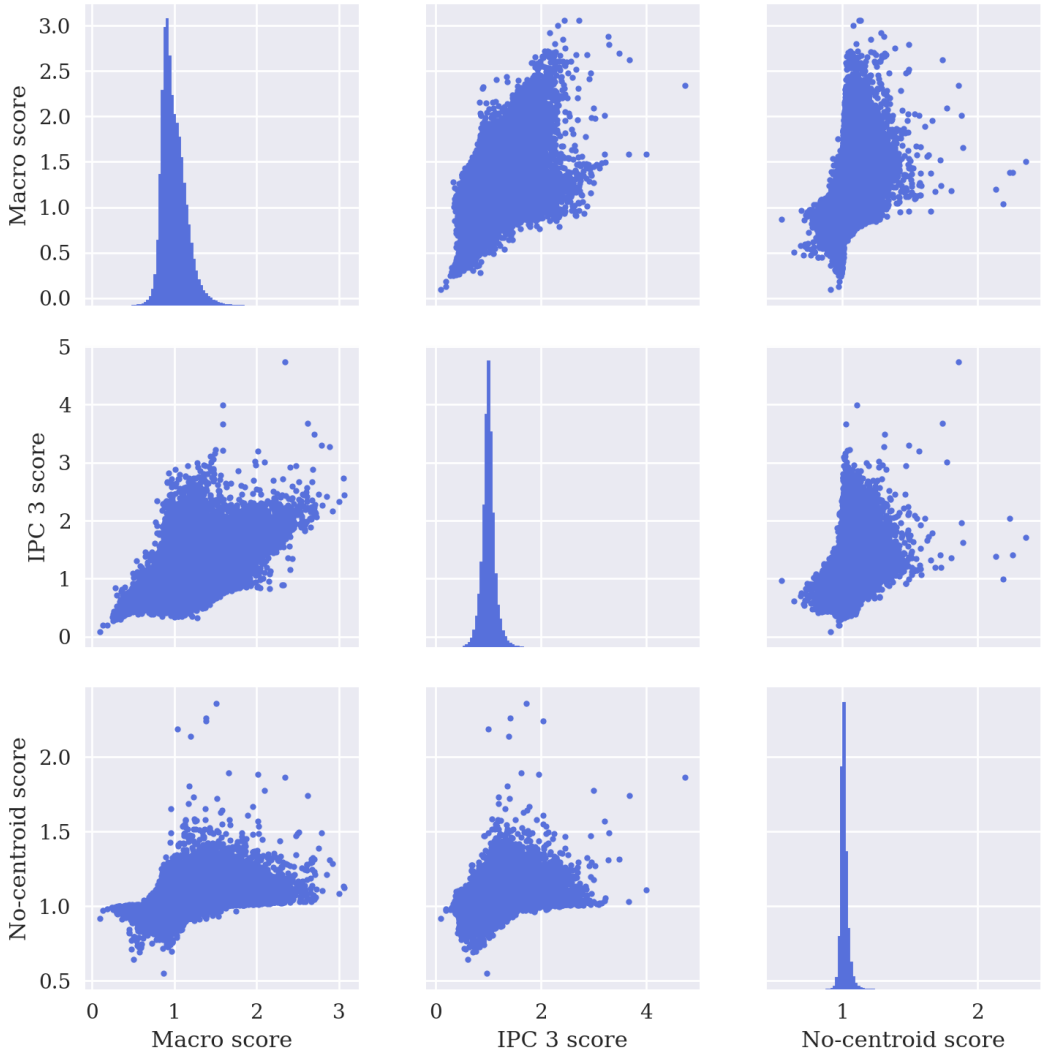
Note: Panel (a): number of patents filed and granted per year. Panel (b) number of patents for which a score is available, and that can be linked to firms in Compustat.

Figure 1.18: Backward versus forward similarities



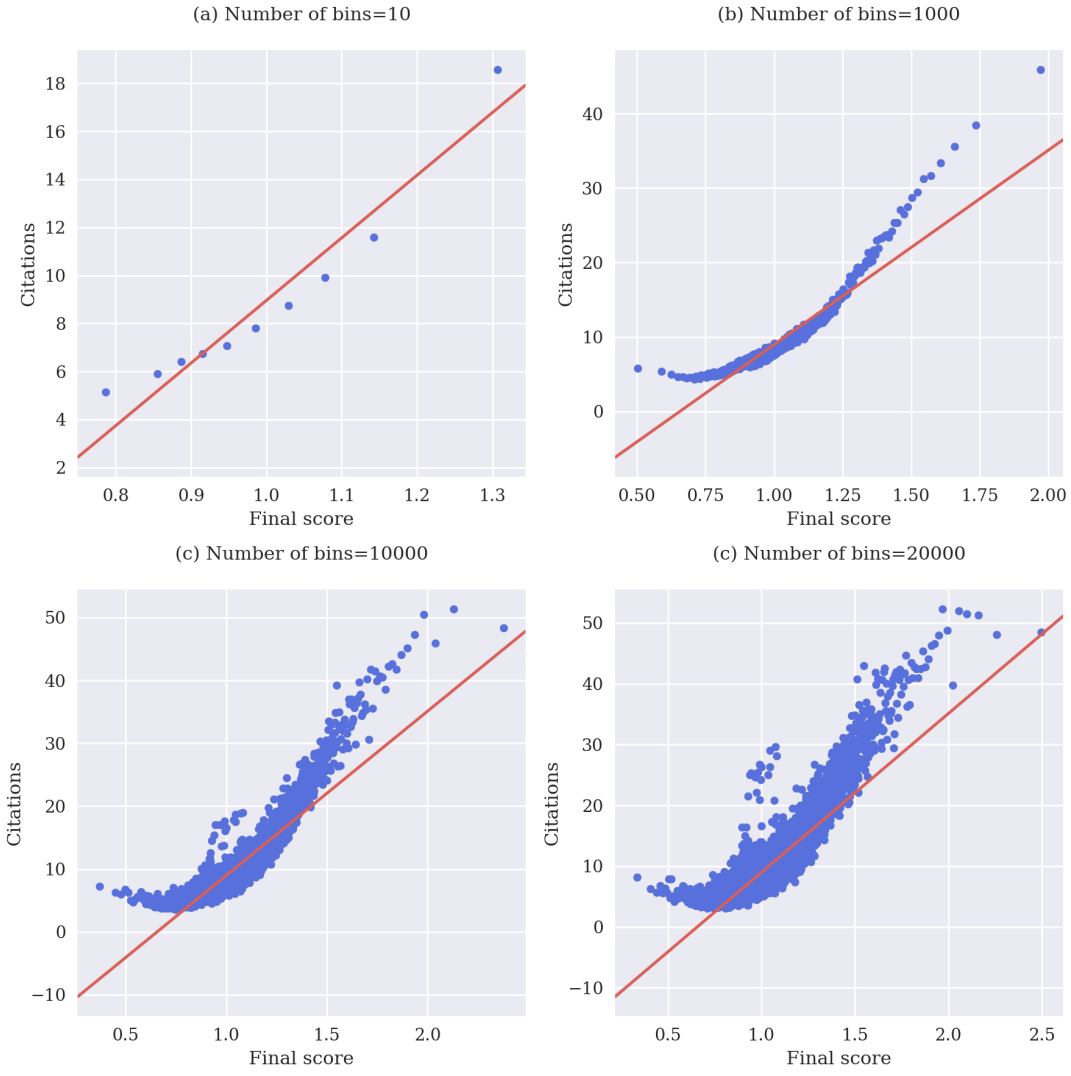
Note: Scatter plots of backward and forward similarities for each score type: macro, IPC3 and no-centroid. Each dot is a patent.

Figure 1.19: Correlogram of scores



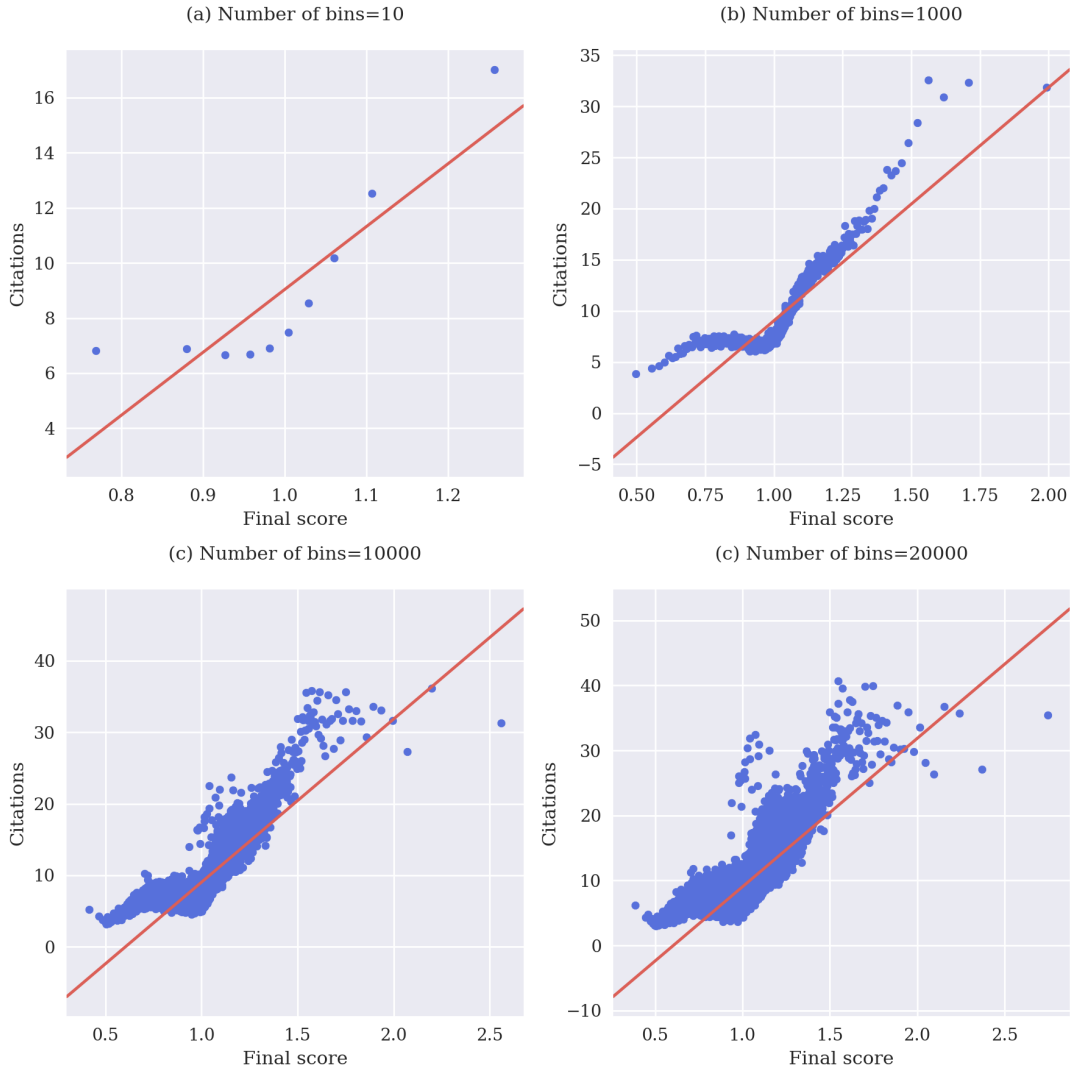
Note: The histograms of the scores are displayed on the diagonal, and scatter plots of scores against each other are displayed off the diagronal. Each dot is a patent.

Figure 1.20: Macro scores versus citations



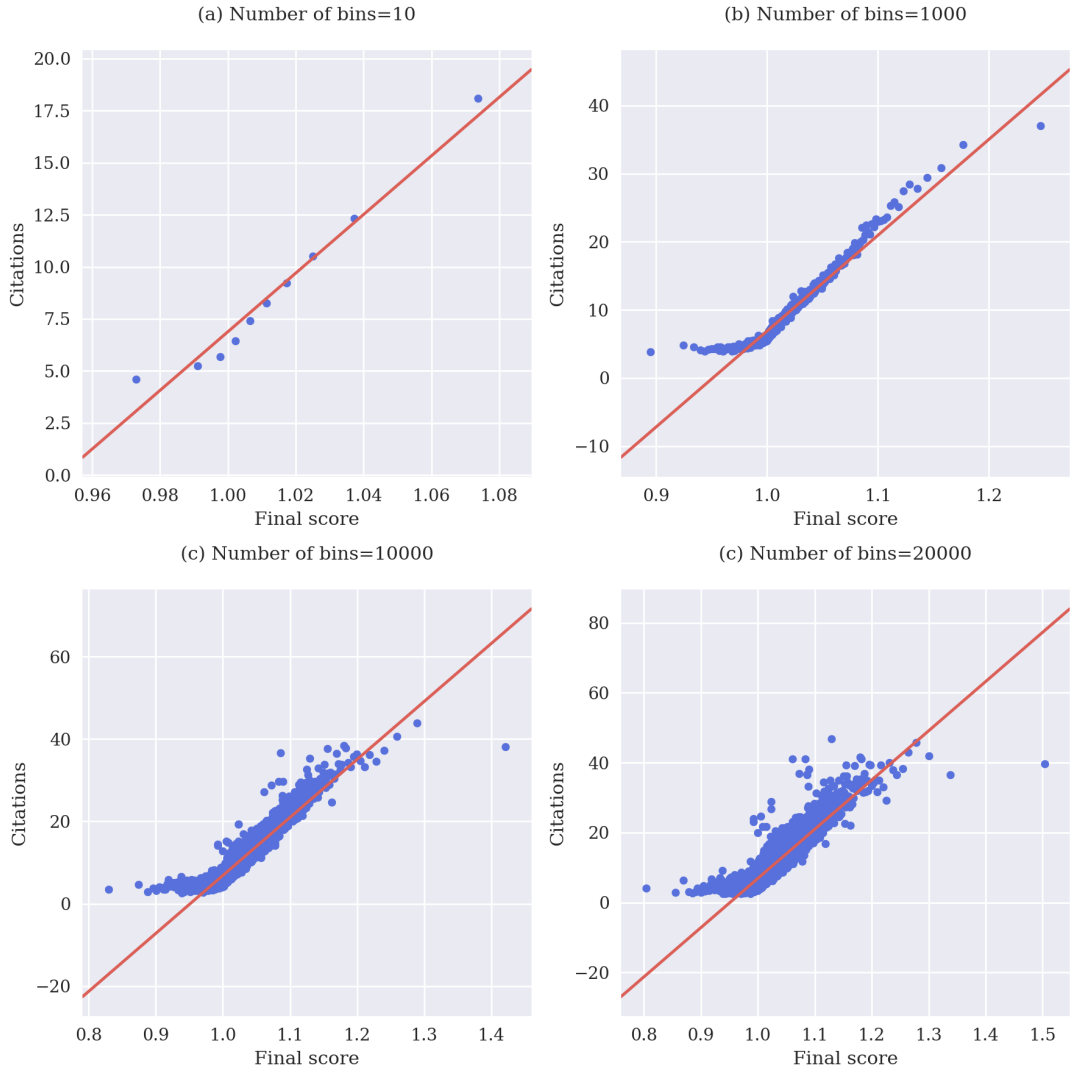
Note: Binned scatter of scores against 10-year forward citations, with varying number of bins.

Figure 1.21: IPC 3 scores versus citations



Note: Binned scatter of scores against 10-year forward citations, with varying number of bins.

Figure 1.22: No-centroid scores versus citations



Note: Binned scatter of scores against 10-year forward citations, with varying number of bins.

Figure 1.23: Scatter plots of raw data: macro, IPC 3 and no-centroid scores against citations

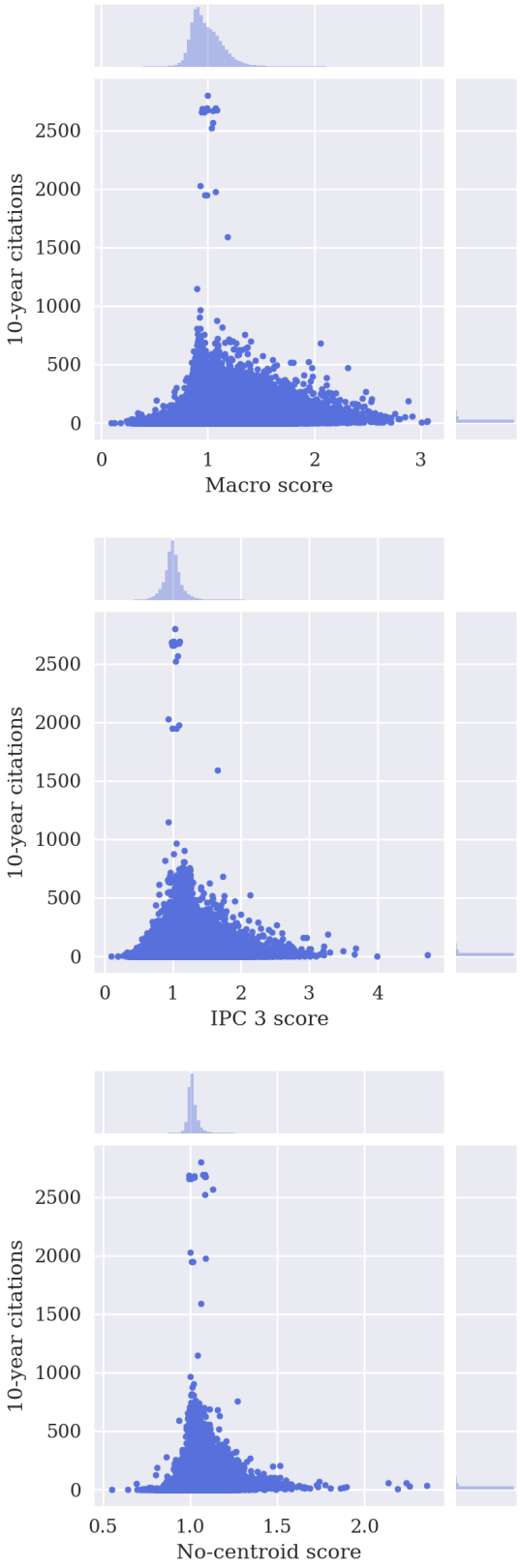
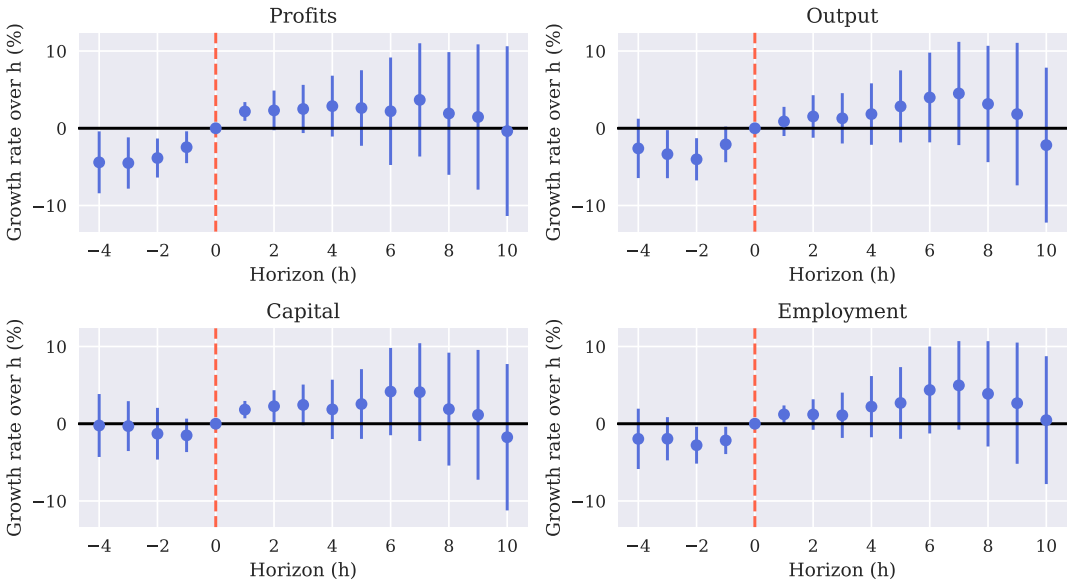
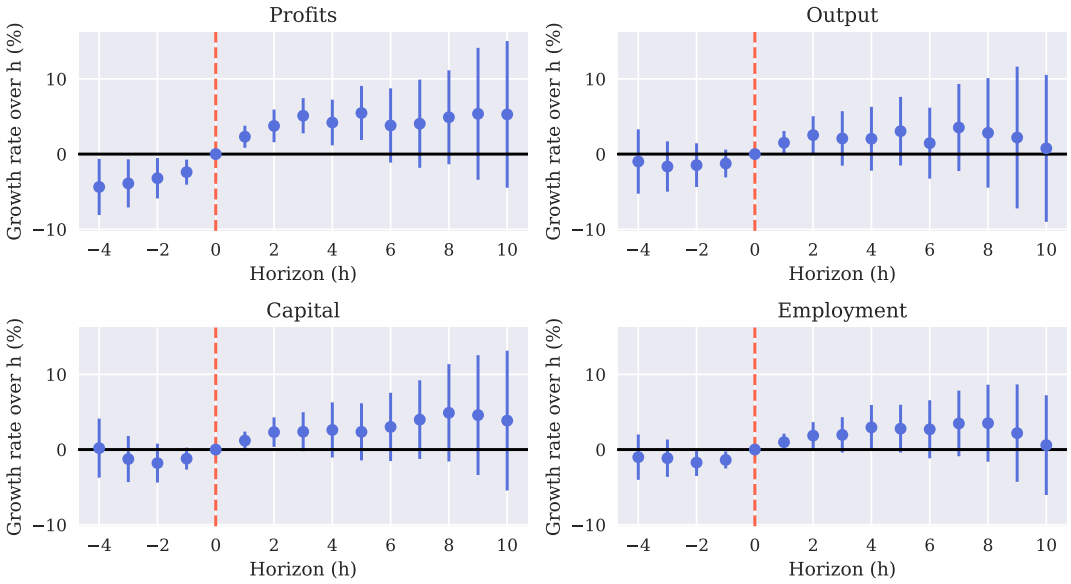


Figure 1.24: Top macro patents and firms dynamics (2)



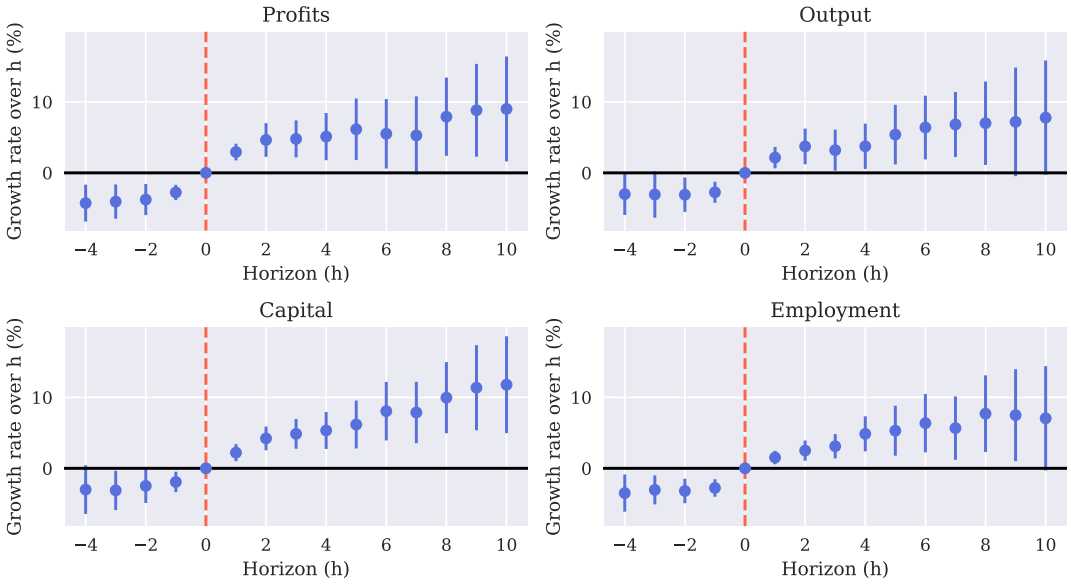
Note: Estimates from Equation (1.4) with alternative definition of growth rate using the macro score to qualify top patents. Dependent variable is $\log Y_{fi,t+h} - \log Y_{fi,t}$, i.e. the growth rate of the outcome value between time 0 and h. 95% confidence intervals are depicted.

Figure 1.25: Top IPC 3 patents and firms dynamics (2)



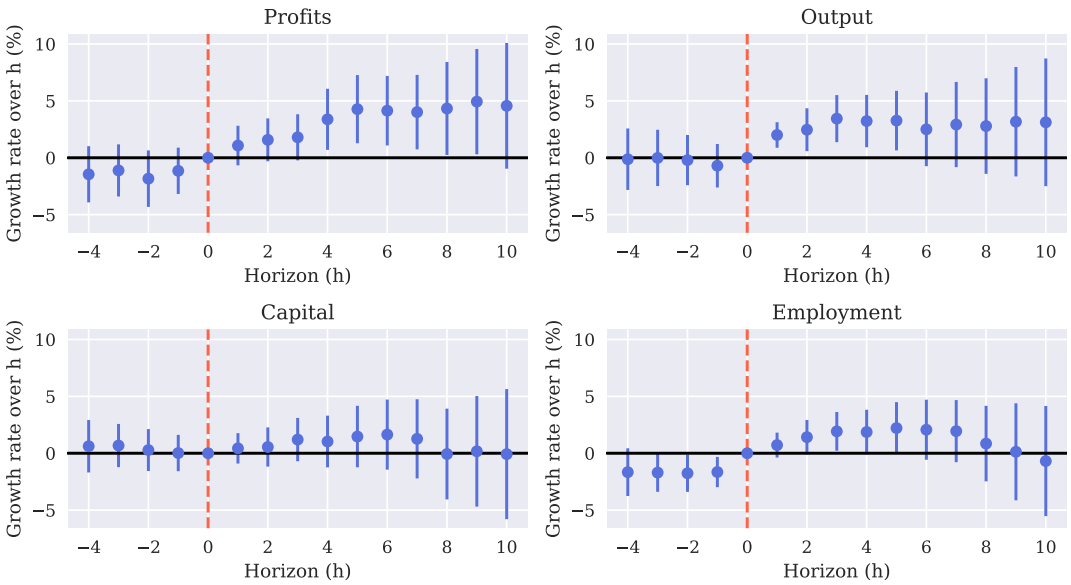
Note: Estimates from Equation (1.4) with alternative definition of growth rate using the IPC 3 score to qualify top patents. Dependent variable is $\log Y_{fi,t+h} - \log Y_{fi,t}$, i.e. the growth rate of the outcome value between time 0 and h. 95% confidence intervals are depicted.

Figure 1.26: No-centroid patents and firms dynamics (2)



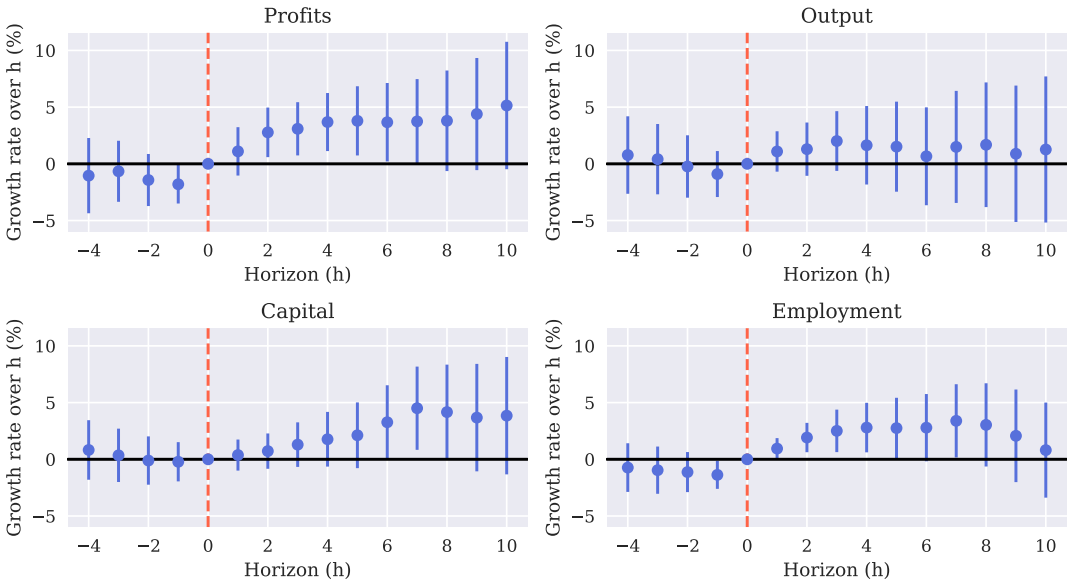
Note: Estimates from Equation (1.4) with alternative definition of growth rate using the no-centroid score to qualify top patents. Dependent variable is $\log Y_{fi,t+h} - \log Y_{fi,t}$, i.e. the growth rate of the outcome value between time 0 and h. 95% confidence intervals are depicted.

Figure 1.27: Top macro patents and firms dynamics: without IT



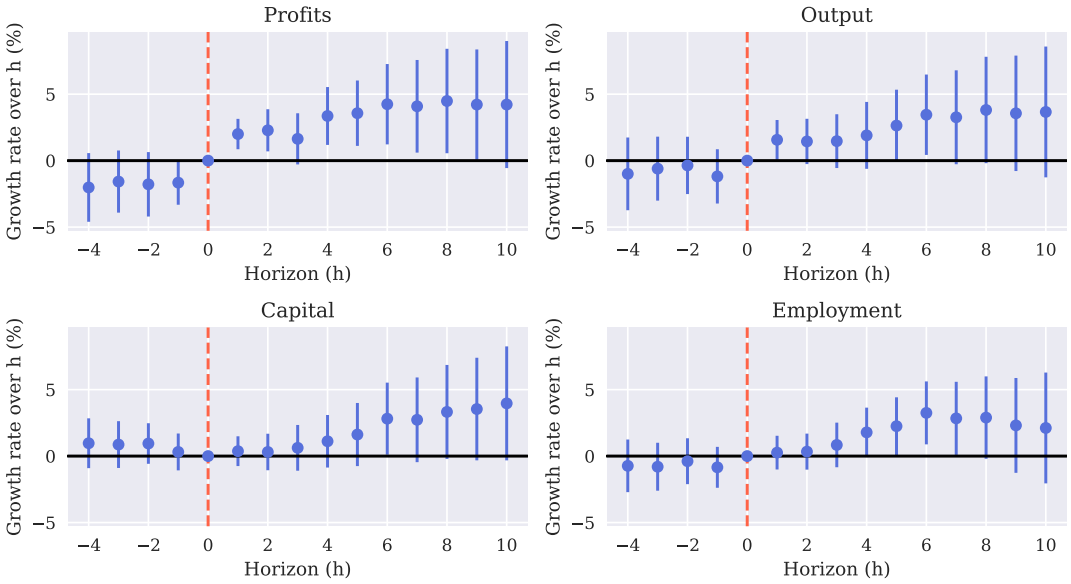
Note: Estimates from Equation (1.4) using the macro score to qualify top patents. 95% confidence intervals are depicted. The sample excludes patents from IPC 1 codes G and H.

Figure 1.28: Top IPC 3 patents and firms dynamics: without IT



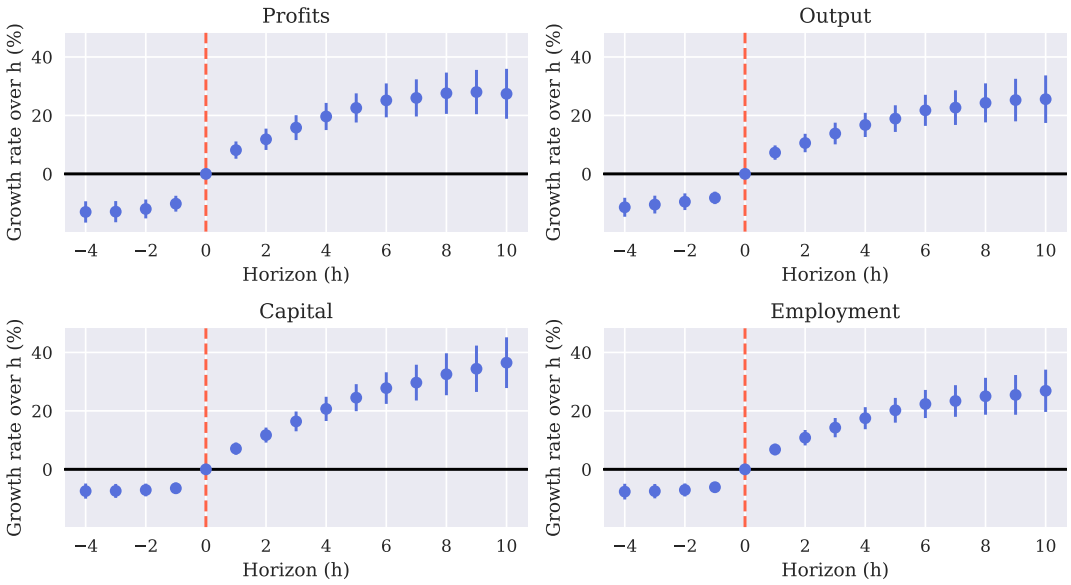
Note: Estimates from Equation (1.4) using the IPC 3 score to qualify top patents. 95% confidence intervals are depicted. The sample excludes patents from IPC 1 codes G and H.

Figure 1.29: Top no-centroid patents and firms dynamics: without IT



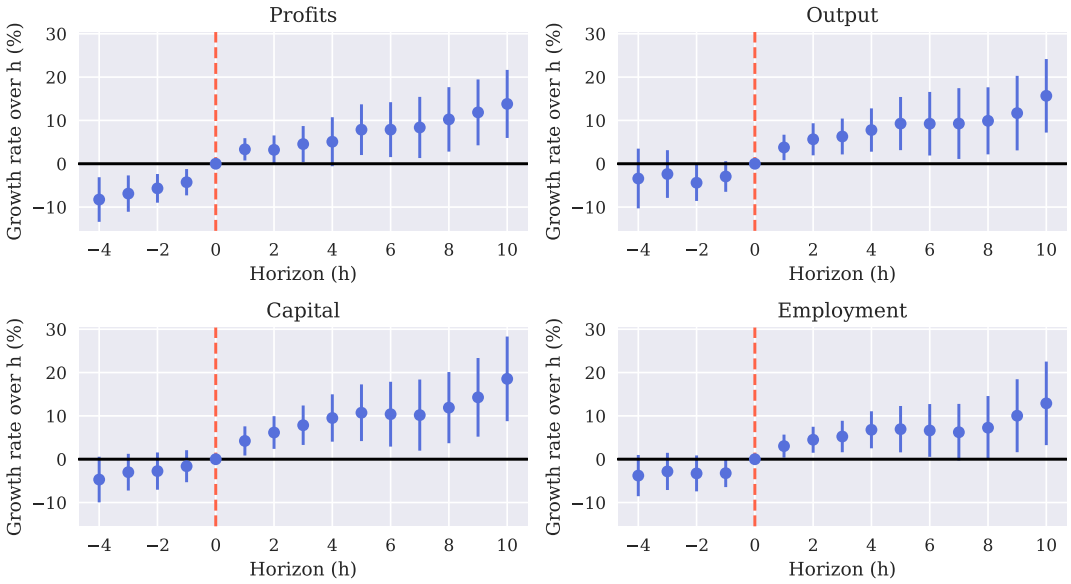
Note: Estimates from Equation (1.4) using the no-centroid score to qualify top patents. 95% confidence intervals are depicted. The sample excludes patents from IPC 1 codes G and H.

Figure 1.30: Top private value patents and firms dynamics



Note: Estimates from Equation (1.4) using private value from Kogan et al. (2017) to qualify top patents. A top patent is one in the top 5% of the private value distribution (controlling for year fixed effects). 95% confidence intervals are depicted.

Figure 1.31: Top patents in terms of citations and firms dynamics



Note: Estimates from Equation (1.4) using citations at the 10-year horizon to qualify top patents. A top patent is one in the top 0.1% of the citations distribution (controlling for year fixed effects). 95% confidence intervals are depicted.

Figure 1.32: Macro scores

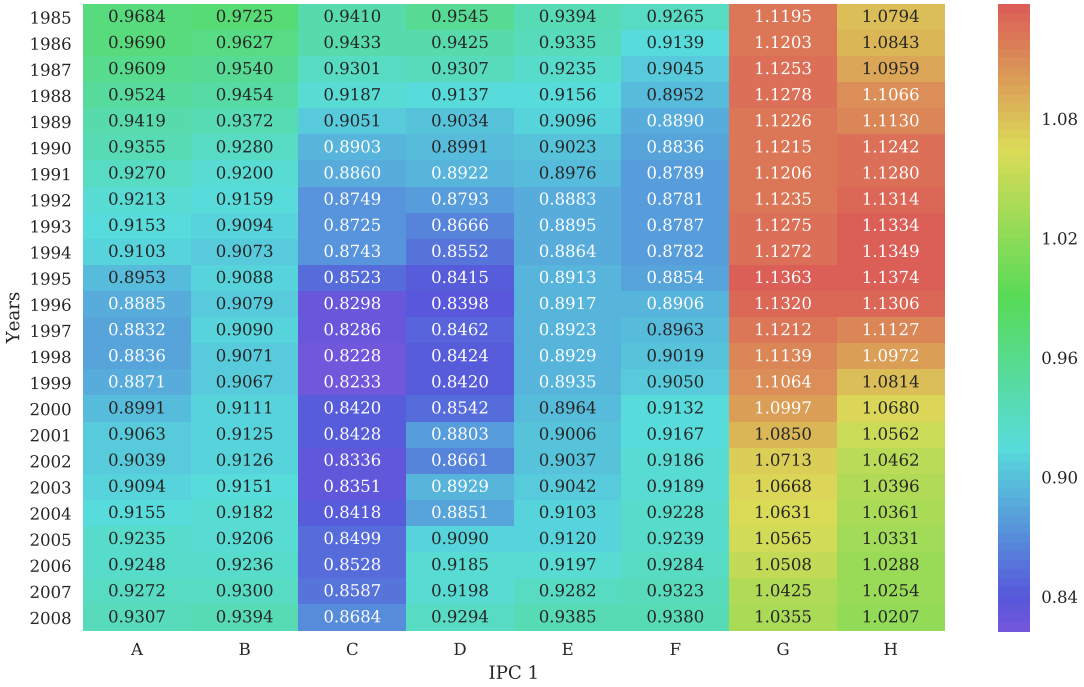


Figure 1.33: IPC 3 scores

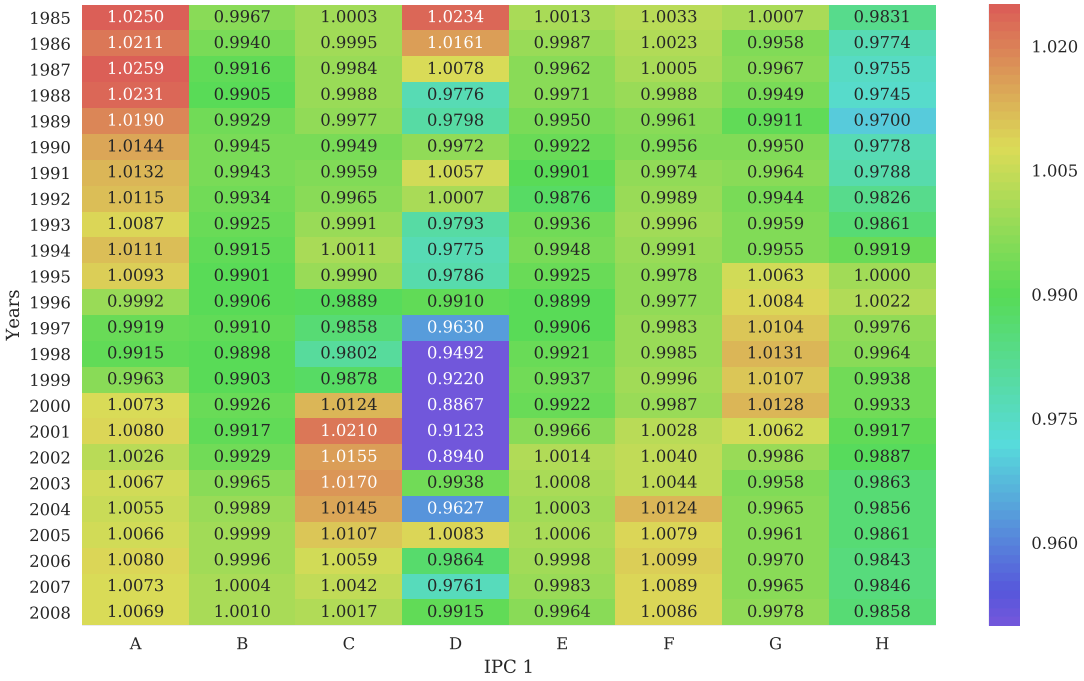
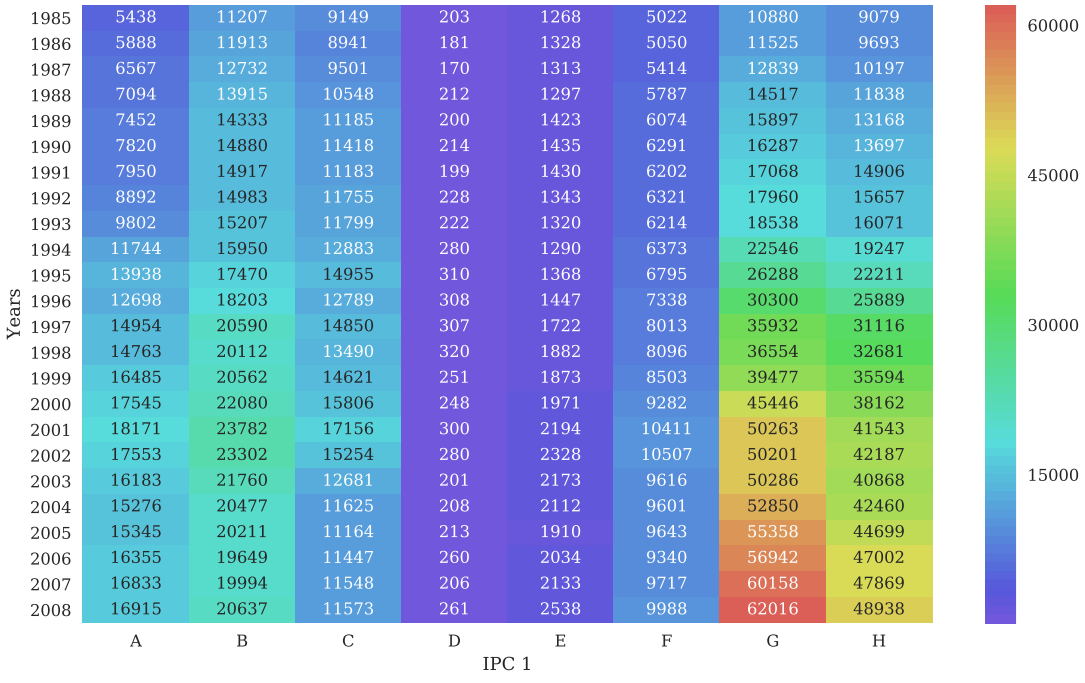


Figure 1.34: Observations (Total: $2,745,260$)



1.7.3 Tables

Table 1.9: One digit IPC codes

Code	Name
A	Human necessities
B	Performing operations; transporting
C	Chemistry; metallurgy
D	Textiles; paper
E	Fixed constructions
F	Mechanical engineering; lighting; heating; weapons; blasting
G	Physics
H	Electricity

Table 1.10: One digit IPC codes and forward space heterogeneity scores

Year	A	B	C	D	E	F	G	H
1985	0.385218	0.385403	0.255063	0.438129	0.483908	0.379408	0.311440	0.293972
2006	0.328542	0.385600	0.301386	0.448159	0.485887	0.380327	0.231544	0.230153

Larger values imply a higher heterogeneity in the forward space.

Table 1.11: Firms with most top patents, no-centroid score

Company name	# of patents	# of top patents	top in % of all top patents	top in % of all own patents
INTL BUSINESS MACHINES CORP	62612	7751	5.4	12
MOTOROLA SOLUTIONS INC	18888	3400	2.3	18
HITACHI LTD	51050	3034	2.1	5.9
NEC CORP	27621	2780	1.9	10
CANON INC	40927	2712	1.9	6.6
PANASONIC CORP	37779	2131	1.5	5.6
SONY CORP	31085	2097	1.4	6.7
HP INC	21909	2011	1.4	9.2
AT&T CORP	12316	1864	1.3	15
INTEL CORP	19925	1725	1.2	8.7
MICROSOFT CORP	14967	1511	1	10
LUCENT TECHNOLOGIES INC	9373	1405	.97	15
GENERAL ELECTRIC CO	31605	1313	.91	4.2
TEXAS INSTRUMENTS INC	16239	1257	.87	7.7
SUN MICROSYSTEMS INC	7503	1080	.75	14
EASTMAN KODAK CO	19728	1058	.73	5.4
XEROX HOLDINGS CORP	16064	987	.68	6.1
MICRON TECHNOLOGY INC	18466	878	.61	4.8
NORTEL NETWORKS CORP	6291	860	.59	14
TELEFONAKTIEBOLAGET LM ERICS	5527	843	.58	15

Notes: List of firms owning the highest number of top patents, ranked using the no-centroid score. The second column indicates the number of patent belonging to that firm, the third column indicates that of top patents belonging to that firm (patents whose score ranks in the top 5% of the overall score distribution, not controlling for year fixed effects); the forth and fifth columns contain the fractions of top patents accruing to that firm (i) out of all the top patents and (ii) out of all the patents of that firm.

Table 1.12: 10-year citations and macro score: log specification

	Dependent variable: Log(1+10-year citations)					
	Whole sample			*Until 2000*		
	(1)	(2)	(3)	(4)	(5)	(6)
Log(macro score)	1.991***	2.032***	1.790***	2.194***	1.976***	1.776***
	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
Constant	1.623***			1.828***		
	(0.000)			(0.000)		
Year FE		✓	✓		✓	✓
IPC 3 FE		✓			✓	
Firm FE			✓			✓
Adjusted R^2	0.069	0.167	0.214	0.117	0.163	0.223
Within R^2		0.040	0.043		0.048	0.053
Observations	2896300	2896013	1025243	1545952	1545675	602117

Notes: *: p<0.1, **: p<0.05, ***: p<0.01. P-values from standard errors clustered at the filing year level in parenthesis.

Table 1.13: 10-year citations and IPC 3 score: log specification

	Dependent variable: Log(1+10-year citations)					
	Whole sample			*Until 2000*		
	(1)	(2)	(3)	(4)	(5)	(6)
Log(IPC 3 score)	1.378***	1.434***	1.171***	1.572***	1.589***	1.171***
	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
Constant	1.618***			1.837***		
	(0.000)			(0.000)		
Year FE		✓	✓		✓	✓
IPC 3 FE		✓			✓	
Firm FE			✓			✓
Adjusted R^2	0.027	0.162	0.200	0.046	0.163	0.204
Within R^2		0.033	0.025		0.052	0.030
Observations	2745260	2745260	993767	1448555	1448555	580505

Notes: *: p<0.1, **: p<0.05, ***: p<0.01. P-values from standard errors clustered at the filing year level in parenthesis.

Table 1.14: Private value and macro score

	Dependent variable: Log(private value)					
	Whole sample			*Until 2000*		
	(1)	(2)	(3)	(4)	(5)	(6)
Log(macro score)	-0.717**	0.217	0.0851	-1.123***	0.324	-0.188
	(0.018)	(0.392)	(0.510)	(0.006)	(0.334)	(0.231)
Constant	0.743***			0.942***		
	(0.000)			(0.000)		
Year FE		✓	✓		✓	✓
IPC 3 FE		✓			✓	
Firm FE			✓			✓
Adjusted R^2	0.002	0.107	0.893	0.007	0.105	0.906
Within R^2		0.000	0.000		0.000	0.001
Observations	948577	948384	947558	557831	557657	556925

Notes: *: p<0.1, **: p<0.05, ***: p<0.01. P-values from standard errors clustered at the filing year level in parenthesis.

Table 1.15: Private value and IPC 3 score

	Dependent variable: Log(private value)					
	Whole sample			*Until 2000*		
	(1)	(2)	(3)	(4)	(5)	(6)
Log(IPC 3 score)	1.056***	0.681***	0.177***	1.149***	0.933***	0.0510
	(0.000)	(0.000)	(0.001)	(0.000)	(0.000)	(0.207)
Constant	0.724***			0.904***		
	(0.000)			(0.000)		
Year FE		✓	✓		✓	✓
IPC 3 FE		✓			✓	
Firm FE			✓			✓
Adjusted R^2	0.004	0.109	0.893	0.005	0.108	0.907
Within R^2		0.002	0.001		0.004	0.000
Observations	919686	919686	918708	538095	538095	537221

Notes: *: p<0.1, **: p<0.05, ***: p<0.01. P-values from standard errors clustered at the filing year level in parenthesis.

Table 1.16: 10-year citations and no-centroid score: log specification

	Dependent variable: Log(1+10-year citations)					
	Whole sample			*Until 2000*		
	(1)	(2)	(3)	(4)	(5)	(6)
Log(no-centroid score)	12.51***	8.176***	7.853***	11.27***	8.324***	7.628***
	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
Constant	1.425***			1.536***		
	(0.000)			(0.000)		
Year FE		✓	✓		✓	✓
IPC 3 FE		✓			✓	
Firm FE			✓			✓
Adjusted R^2	0.097	0.162	0.212	0.092	0.162	0.220
Within R^2		0.035	0.041		0.046	0.049
Observations	2896300	2896013	1025243	1545952	1545675	602117

Notes: *: p<0.1, **: p<0.05, ***: p<0.01. P-values from standard errors clustered at the filing year level in parenthesis.

Table 1.17: Private value and no-centroid score

	Dependent variable: Log(private value)					
	Whole sample			*Until 2000*		
	(1)	(2)	(3)	(4)	(5)	(6)
Log(no-centroid score)	3.518**	3.346***	0.991***	0.0969	4.230***	0.243
	(0.016)	(0.000)	(0.003)	(0.941)	(0.000)	(0.380)
Constant	0.668***			0.909***		
	(0.000)			(0.000)		
Year FE		✓	✓		✓	✓
IPC 3 FE		✓			✓	
Firm FE			✓			✓
Adjusted R^2	0.002	0.109	0.893	-0.000	0.107	0.906
Within R^2		0.002	0.001		0.003	0.000
Observations	948577	948384	947558	557831	557657	556925

Notes: *: p<0.1, **: p<0.05, ***: p<0.01. P-values from standard errors clustered at the filing year level in parenthesis.

Table 1.18: Forward neighbourhood heterogeneity score and citations: by IPC 1 code

	Dependent variable: 10-year citations							
IPC 1	(A)	(B)	(C)	(D)	(E)	(F)	(G)	(H)
No-cent. score, std	3.925***	1.097***	1.408***	0.727***	2.099***	1.386***	1.634***	1.795***
	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
For. Het. score, std	-1.538***	-0.346***	-0.369***	-0.182	-1.019***	-0.455***	-0.581***	-0.388***
	(0.000)	(0.000)	(0.000)	(0.232)	(0.002)	(0.000)	(0.000)	(0.001)
Year FE	✓	✓	✓	✓	✓	✓	✓	✓
IPC 3 FE	✓	✓	✓	✓	✓	✓	✓	✓
Adjusted R^2	0.025	0.024	0.029	0.020	0.069	0.036	0.023	0.006
Within R^2	0.012	0.013	0.015	0.008	0.016	0.017	0.014	0.006
Observations	19595	31115	18699	1706	4115	15680	61324	51198

Notes: *: p<0.1, **: p<0.05, ***: p<0.01. P-values from robust standard errors in parenthesis. Standard errors can not be clustered at the filing year level since there are only two years in the data underlying this regression.

Table 1.19: Forward neighbourhood heterogeneity score and log citations: by IPC 1 code

	Dependent variable: Log(1+10-year citations)							
IPC 1	(A)	(B)	(C)	(D)	(E)	(F)	(G)	(H)
Log(no-cent. score)	8.267***	6.230***	7.270***	3.814***	6.152***	7.658***	6.622***	6.661***
	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
Log(For. Het. score)	-0.134***	-0.0487***	-0.0261***	-0.0752**	-0.149***	-0.0829***	-0.0616***	-0.0379***
	(0.000)	(0.000)	(0.006)	(0.041)	(0.000)	(0.000)	(0.000)	(0.000)
Year FE	✓	✓	✓	✓	✓	✓	✓	✓
IPC 3 FE	✓	✓	✓	✓	✓	✓	✓	✓
Adjusted R^2	0.057	0.064	0.097	0.063	0.091	0.048	0.061	0.047
Within R^2	0.028	0.020	0.016	0.009	0.017	0.029	0.026	0.026
Observations	19345	30996	18639	1705	4115	15625	59256	49741

Notes: *: p<0.1, **: p<0.05, ***: p<0.01. P-values from robust standard errors in parenthesis. Standard errors can not be clustered at the filing year level since there are only two years in the data underlying this regression.

Table 1.20: Citations, private value and macro score: without IT

	(1) 10-year citations	(2) Log(1+10-year citations)	(3) Private value	(4) Log(private value)
Macro score, std.	2.277***		-3.224***	
	(0.000)		(0.000)	
Log(macro score)		1.799***		-1.430***
		(0.000)		(0.000)
Year FE	✓	✓	✓	✓
IPC 3 FE	✓	✓	✓	✓
Adjusted R^2	0.067	0.120	0.086	0.107
Within R^2	0.015	0.027	0.006	0.004
Observations	1402076	1402076	350304	350304

Notes: *: p<0.1, **: p<0.05, ***: p<0.01. P-values from standard errors clustered at the filing year level in parenthesis.

Table 1.21: Citations, private value and IPC 3 score: without IT

	(1) 10-year citations	(2) Log(1+10-year citations)	(3) Private value	(4) Log(private value)
IPC 3 score, std.	2.499***		-1.886***	
	(0.000)		(0.002)	
Log(IPC 3 score)		1.595***		-0.954***
		(0.000)		(0.000)
Year FE	✓	✓	✓	✓
IPC 3 FE	✓	✓	✓	✓
Adjusted R^2	0.072	0.125	0.081	0.107
Within R^2	0.022	0.031	0.002	0.003
Observations	1260360	1260360	323789	323789

Notes: *: p<0.1, **: p<0.05, ***: p<0.01. P-values from standard errors clustered at the filing year level in parenthesis.

Table 1.22: Citations, private value and no-centroid score: without IT

	(1) 10-year citations	(2) Log(1+10-year citations)	(3) Private value	(4) Log(private value)
No-centroid score, std.	2.553***		-0.962**	
	(0.000)		(0.026)	
Log(no-centroid score)		8.388***		-1.733***
		(0.000)		(0.001)
Year FE	✓	✓	✓	✓
IPC 3 FE	✓	✓	✓	✓
Adjusted R^2	0.071	0.120	0.081	0.103
Within R^2	0.020	0.027	0.001	0.000
Observations	1402076	1402076	350304	350304

Notes: *: p<0.1, **: p<0.05, ***: p<0.01. P-values from standard errors clustered at the filing year level in parenthesis.

Chapter 2

Dynamically Optimal Treatment Allocation using Reinforcement Learning - Empirical Application

2.1 Introduction

Consider a situation where a stream of individuals arrive sequentially - e.g. when they get unemployed - to a social planner. Once each individual arrives, our planner needs to decide on an action, i.e a treatment assignment for the individual - e.g. whether or not to offer free job training - taking into account the individual's characteristics and various institutional constraints such as limited budget. The decision on the treatment is to be taken instantaneously. It is taken without knowledge of the characteristics of future individuals, though the planner can, and should, form expectations over these future characteristics. Once an action is taken, the individual is assigned a specific treatment, leading to a reward, i.e a change in the utility for that individual. The planner does not observe these rewards directly since they may be only realized much later, but she can observe an estimate of them using data for example from randomised control trials (RCTs). At the same time, the action of the planner generates an observed change to the institutional variable, such as a reduced budget. The planner takes note of these changes, and waits for the next individual to arrive. This process ends e.g. when her budget is depleted. Expanding Kitagawa and Tetenov (2018) to a dynamic setting, we model such a problem based on the Job Training Partnership Act (JTPA) RCT dataset. We then propose a reinforcement learning algorithm to try to obtain the welfare maximizing treatment allocation rule for this dynamic

setting.

Reinforcement learning is the tool behind many of the most noticed recent advances in artificial intelligence, from applications like autonomously learning to play Atari games from their screen input (Mnih et al., 2015), beating the best humans at the game of Go (Silver et al., 2017), learning and mastering chess, shogi, and Go without human knowledge (Silver et al., 2018), or mastering robotic tasks such as learning to fly artistic manoeuvres with a model helicopter (Abbeel et al., 2007).[1] All these problems share a common structure: There is a dynamic environment in which a point in time can be summarised as a state. An agent observes this states, takes an action, then observes an associated reward, finds itself in the new state that resulted from its action and other forces, and so on. Many dynamic problems in economics have the same structure and reinforcement learning can in principle be applied to them. While the descriptions of these current methods from artificial intelligence are sometimes vague, reinforcement learning is in essence a powerful solution technique to obtain good policy functions for very complicated dynamic problems. Other than some more frequently used methods in economics such as policy iteration, it obtains this policy function by solving dynamic problems in a "forward" manner: An agent wanders through the state space in episodes, initially taking entirely random actions, observing resulting rewards, and then reinforcing behaviour which led to high rewards.

In some cases, reinforcement learning can become very interesting as a solution technique for problems in economics, e.g. if state spaces are so large that traditional methods become infeasible or if policy classes have to be restricted. This is the motivation for our application: Many if not most real world dynamic problems that economic policy makers could realistically face involve taking into account very large amounts of state variables from covariates of individuals, to institutional constraints, etc. This application discusses one such policy problem. The provision of job training to those unemployed individuals that can benefit most from it if resources of the state are limited and individuals arrive sequentially.[2] To stay comparable to Kitagawa and Tetenov (2018), we focus on the JTPA and model sequential arrival of unemployed individuals as well as their treatment effects with this dataset. Yet, as the methods scale to problems with much larger state spaces, they could offer ways to improve welfare in other economic policy problems when scarce resources need to be allocated in very complicated dynamic

[1] Many of such examples today also additionally make use of deep neural networks for approximation of policy and value functions. We stick to a logistic policy function and a linear value function with additional nonlinear transformation of state variables.

[2] This chapter is the empirical section of the paper Adusumilli et al. (2019) with further explanations.

settings. Their availability might also create an incentive to run larger observational studies with more covariates to generate data to train such algorithms. Lastly, as the rewards in our environment are estimated individual treatment effects, we also give an example from economics of how to apply reinforcement learning to outcomes from causal inference.

Importantly, the paper follows a two step approach: In a first step, we use the full RCT dataset to estimate treatment effects (which we use as rewards for the agent) and arrival rates. Afterwards we take these estimates as given, construct our environment with them, and then let our reinforcement learning agent solve for the optimal policy given these estimates. This optimal policy function maps the current state variables of observed characteristics and institutional constraints to probabilities over the set of actions the agent or policy maker. We treat the class of policy functions as given. Then for any policy from that class, we can write down a dynamic programming problem (or in more general contexts, a Partial Differential Equation, see Adusumilli et al., 2019), that characterizes the expected value function under that policy, where the expectation is taken over the distribution of the individual covariates. Using the data, we can similarly write down a sample version of the dynamic programming problem that provides estimates of these value functions. The estimated policy rule is the one that tries to maximize the estimated value function at the start of the program.

One particularity of the algorithm we use is that it solves the optimum within a pre-specified policy class, e.g. for a softmax function as policy function. As explained by Kitagawa and Tetenov (2018), one may wish to restrict the policy class for computational or legal reasons. Another reason is incentive compatibility, e.g. the planner may want the policy to change slowly with time to prevent individuals from manipulating arrival times. The key assumption that we do impose is that the environment, i.e the arrival rates and distribution of individuals, is not affected by the policy. This is a reasonable assumption in many contexts, especially in settings like unemployment, arrivals to emergency rooms, childbirth (e.g. for provision of daycare) etc., where either the time of arrival is not in complete control of the individual, or where it is determined by factors exogenous to the provision of treatment. In addition, even where this assumption is suspect, we could try to build a model of responses to the agent's actions and include it into the environment.

If the dynamic aspect can be ignored, there exist a number of methods to estimate an optimal policy function that maximizes social welfare, starting from the seminal contribution of Manski (2004), and further extended by Hirano and Porter

(2009), Stoye (2009) and Stoye (2012), Chamberlain (2011), Bhattacharya and Dupas (2012) and Tetenov (2012), among others. More recently, Kitagawa and Tetenov (2018), and Athey and Wager (2018) proposed using Empirical Welfare Maximization (EWM) in this context. While these papers address the question of optimal treatment allocation under co-variate heterogeneity, the resulting treatment rule is static in that it is determined ex-ante, before observing the data on which it will be applied. It does not change with time, nor with current values of institutional constraints. In fact, in our application the EWM is not even applicable. This is so even if we restricted ourselves to using a static policy. For instance, with budget constraints, the EWM rule requires one to specify the fraction of population that can be treated, but in our dynamic environment the number of individuals the planner faces is endogenous to the policy.

There also exist a number of methods for estimating the optimal treatment assignment policy using 'online' data. This is known as the contextual bandit problem, and there is a large literature on this, see e.g. Dudik et al. (2011), Agarwal et al. (2014), Russo and Van Roy (2016) and Dimakopoulou et al. (2017). Yet, bandit algorithms do not have a forward looking nature; the eventual policy function that is learnt is still static in that it does not take into account the effect of current actions on future states or rewards. By contrast, our primary goal in this paper is to use 'offline', i.e historical data, to estimate a policy rule that is optimal under such inter-temporal trade-offs. This policy function could then be applied to new data from the same distribution. Yet, methods like algorithms can also be applied in a completely offline manner in infinite-horizon Markov Decision Process settings, where the usual bandit algorithms do not apply. See Sutton and Barto (2018) Chapter 3 on the difference between Markov Decision and bandit problems.

Another set of results close to our work is from the literature on Dynamic Treatment Regimes (DTRs). We refer to Laber et al. (2014) and Chakraborty and Murphy (2014) for an overview. Some of the papers reviewed use tools from reinforcement learning such as e.g. Murphy (2005) which uses a variant of Q learning. DTRs consist of a sequence of individualized treatment decisions for health related interventions. These are typically estimated from sequential randomized trials, e.g. Murphy (2005) and Lei et al. (2012), where participants move through different stages of treatment, which is randomized in each stage. By contrast, we only make use of a single set of observational data, and this data itself does not come in a dynamic form. Each individual in our setup is only exposed to treatment once. The dynamics are faced not by the individual, but by the social planner. Additionally, in DTRs the number of stages or decision points is quite

small, typically between 1 and 3. By contrast, the number of decision points, i.e the rate of arrivals, in our setting is high.

Previous work in economics on dynamic programming has often used Generalised Policy Iteration, e.g. Benitez-Silva et al. (2000). While this method works well with discrete states, there are three main drawbacks: First, it might not allow for restricting the solution to a pre-specified class of policy rules. Second, the algorithm can become cumbersome even with a few continuous states and a few thousand decision points. Continuous states may be handled through discretisation or parametric policy iteration. The former is typically slower and suffers from a strong curse of dimensionality, see Section 2.5 of Benitez-Silva et al. (2000); while the latter requires numerical integration which is also very demanding with more than a few states. Third, it cannot be directly applied to our setup without incorporating a regularization parameter to avoid over-fitting the value function (and it is not obvious how such a regularization may be employed). This is because standard reward estimates (inverse propensity weighting, doubly robust, etc.) are direct functions of the outcome variables from the observational data. Hence the usual policy iteration algorithm could overfit the estimate of the value function to this data. In this paper, we discuss a modified reinforcement learning (RL) algorithm that tries to improve on these problems. We refer to Sutton and Barto (2018) for a detailed comparison of recent RL algorithms with policy iteration. We adapt the Actor-Critic algorithm, e.g. Sutton et al. (2000) and Bhatnagar et al. (2009), that has been applied recently to great effect in applications as diverse as playing Atari games Mnih et al. (2016), image classification Mnih et al. (2014) and machine translation Bahdanau et al. (2016). Our algorithm improves the over-fitting issue by working with the expected value function that integrates over the rewards at each step. The integration is implicit since we use stochastic gradient descent, so the computational complexity is not affected.

The paper is structured as follows: Section 2.2 briefly summaries the optimisation problem with discrete arrivals, Section 2.3 describes how we build the world from RCT data in which our reinforcement learning agent lives, Section 2.4 illustrates the algorithm and pseudocode, Section 2.5 describe how we parametrise environment and algorithm, Section 2.6 discusses the results, and Section 3.4 concludes.

2.2 The problem

Recall that a social planner or government employee faces the following problem: Individuals who lost their jobs sequentially arrive at her office to register as unemployed. She can provide job training to individuals which is free for them but costly for her. Due to a limited budget, however, she cannot provide training to all individuals that arrive. To maximise welfare, she therefore has to solve a challenging dynamic optimisation problem. She has to understand the likely welfare benefit of treatment (here: earnings after job training) from some information about an individual (e.g. their previous earnings, age, etc.). Furthermore, she has to form expectations over future arrivals of newly unemployed to avoid depleting her budget too quickly when she could have used her resources more efficiently.

True problem We begin by stating the problem assuming we knew everything about the world such as distributions, rewards, and arrival rates.

Let x denote the vector of characteristics of an individual, based on which the planner makes a decision on whether to provide training ($a = 1$) or not ($a = 0$). The current budget is denoted by z. Once an action, a, has been chosen, the planner affects an increase in social welfare by the quantity $Y(a)$ that is equivalent to the potential outcome of the individual under action a. We shall assume that $Y(a)$ is affected by the covariates x but not the budget. Define $r(x, a) = E[Y(a)|x]$ as the expected (instantaneous) reward for the social planner when the planner chooses action a for an individual with characteristics x. Since we only consider additive welfare criteria in this paper, we may normalize $r(x, 0) = 0$, and set $r(x, 1) = E[Y(1) - Y(0)|x]$. Note that we can accommodate various welfare criteria, as long as they are utilitarian, by redefining the potential outcomes.

If the planner takes action a=1, her budget is decreased by c, otherwise it stays the same. The next individual arrives after a waiting time Δt given by an exponential distribution $\text{Exp}(\lambda(t))$. Each time a new individual arrives, the covariates for the individual are assumed to be drawn from a distribution F that is fixed but unknown. The utility from treating successive individuals is discounted exponentially by $e^{-\beta \Delta t}$.

To simplify matters, we shall assume for this section that time t is not a state variable. This is reasonable if we assume $\lambda(t)$ is independent of t. The inclusion of time as a state variable creates some technical difficulties, which are taken care of in our companion paper Adusumilli et al. (2019).

The planner chooses a policy function $\pi_\theta(a|x,z;\theta)$, indexed by θ that maps the current state variables x, z to a probabilistic choice over the set of actions:

$$\pi(.|x,z;\theta) : (x,z) \longrightarrow [0,1].$$

The aim of the social planner is to determine a policy rule that maximizes expected welfare after discounting. In other words, our agent would solve the following stochastic dynamic optimisation problem:

$$\max_{\theta} \quad \mathbb{E}\left[\sum_{i=1}^{\infty} e^{-\beta T_i}\, \pi(a=1|x_i,z_i;\theta)\, r(x_i,a=1)\, \mathbb{I}\{z_i \geq c\}\right] \tag{2.1}$$

$$s.t. \quad z_{i+1} = z_i - ac \tag{2.2}$$

$$z_1 = \bar{z} \tag{2.3}$$

where the expectation is joint over the times of arrival T_i of each individual, covariates $x \sim F$ and z_i which evolves according to the distribution of x and the randomization of the policy $\pi(.)$. The solution to this problem would yield optimal parameters θ^*.

From the point of view of estimation, it is more convenient to write the above in a value function form, which gives another way to characterize θ^*. Let $v_\theta(x,z)$ denote the value function under parameter θ, defined as the expected welfare from implementing policy $\pi(\cdot|x,z;\theta)$ starting from the state (x,z). This can be represented in a recursive form as the fixed point to the equations:

$$v_\theta(x,z) = r(x,1)\pi(1|x,z;\theta) + \left(1-\tilde{\beta}\right) E_{x'\sim F}\left[v_\theta\left(x', z-c\right)\pi(1|x,z;\theta) + v_\theta(x',z)\pi(0|x,z;\theta)\right],$$

in conjunction with the terminal condition $v_\theta(0,t) = 0$, and where $\tilde{\beta} = 1 - E[e^{-\beta\Delta t}]$. To obtain a more insightful expression, we can integrate out x. This motivates the integrated value function:

$$h_\theta(z) := E_{x\sim F}[v_\theta(x,z)].$$

Define $\bar{\pi}(a|z;\theta) = E_{x\sim F}[\pi_\theta(a|x,z;\theta)]$ and $\bar{r}(z;\theta) = E_{x\sim F}\left[r(x,1)\pi(1|x,z;\theta)\right]$. We can then characterize $h_\theta(.)$ as the solution to the recursive equations

$$h_\theta(z) = \bar{r}(z;\theta) + \left(1-\tilde{\beta}\right)\left\{h_\theta\left(z-c\right)\bar{\pi}(1|z;\theta) + h_\theta(z)\bar{\pi}(0|z;\theta)\right\},$$

together with the terminal condition $h_\theta(0) = 0$.

The social planner's decision problem is to choose the optimal θ^* that maximizes

the ex-ante expected welfare $h_\theta(\bar{z})$:

$$\theta^* = arg\max_\theta h_\theta(\bar{z}).$$

Estimated problem Our agent does not solve this problem, however, as we cannot model the true distribution of individual, rewards, or arrival rates. We therefore employ a two step procedure: In the first step, we build the agent's environment. We proxy the true distribution F with an empirical distribution F_n using an RCT dataset. Based on this full dataset, we estimate rewards $\hat{r}$ and arrival rates $\hat{\lambda}(t)$ (we can estimate $\hat{\lambda}(t)$ as our dataset stores information about the date individuals became unemployed). Only in a second step, our agent then solves for the optimal policy given this estimated world:

$$\max_\theta \quad \mathbb{E}_n\left[\sum_{i=1}^{\infty} e^{-\beta \hat{T}_i}\, \pi(a=1|x_i,z_i;\theta)\, \hat{r}(x_i,a=1)\, \mathbb{I}\{z_i \geq c\}\right] \tag{2.4}$$

$$s.t. \quad z_{i+1} = z_i - ac \tag{2.5}$$

$$z_1 = \bar{z} \tag{2.6}$$

where $\mathbb{E}_n[\cdot]$ is the joint expectation over the times of arrival T_i of each individual, covariates $x \sim F_n$ and z_i which evolves according to the distribution of x and the randomization of the policy $\pi(.)$. The solution to this problem now is an optimal policy parameter vector $\hat{\theta}$.

As before, we can rewrite the computation of $\hat{\theta}$ in a recursive form. Define $\hat{\pi}(a|z;\theta) = E_{x\sim F_n}[\pi(a|x,z;\theta)]$ and $\hat{r}(z;\theta) = E_{x\sim F_n}\left[r(x,1)\pi(1|x,z;\theta)\right]$. Based on the knowledge of $\hat{r}(.)$ and F_n, we can calculate a sample estimate of the integrated value function as the solution to

$$\hat{h}_\theta(z) = \hat{r}(z;\theta) + \left(1-\tilde{\beta}\right)\left\{\hat{h}_\theta\left(z-c\right)\hat{\pi}(1|z;\theta) + \hat{h}_\theta(z)\hat{\pi}(0|z;\theta)\right\},$$

together with the terminal condition $\hat{h}_\theta(0) = 0$. Using $\hat{h}_\theta(.)$ we can solve a sample version of the social planner's problem:

$$\hat{\theta} = \arg\max_\theta \hat{h}_\theta(\bar{z}).$$

As $\hat{\theta}$ is a function of estimated rewards, arrival rates, etc., it contains the original uncertainty of our sample and we refer refer to it as an estimate in itself.

In Adusumilli et al. (2019) we show that the difference in welfare from using $\hat{\theta}$

instead of θ^*, i.e the difference $h_\theta(\bar{z}) - h_{\hat{\theta}}(\bar{z})$, decays to 0 at the rate of $n^{-1/2}$, where n is the number of observations in the RCT. Thus we get arbitrarily close to the optimal welfare as long as the dataset is sufficiently large. Note that this doesn't by itself imply $\hat{\theta}$ is consistent for θ^*, only that the welfare difference between the two is small.

Given θ, one could solve for $\hat{h}_\theta$ by backward induction starting from $z = 0$ using the recursive formulation. In this simple example this is feasible as long as the number of decision points is not too large, but note that one would still need to calculate the summations $E_{x\sim F_n}[\pi_\theta(a|x,z)]$ and $E_{x\sim F_n}[r(x,1)\pi_\theta(1|x,z)]$, i.e averages of the order n, for all possible values of z. And even where solving for $\hat{h}_\theta(\bar{z})$ is feasible, we yet have to maximize this over θ. Such a strategy is computationally too demanding. Therefore in this paper we advocate a reinforcement learning algorithm that directly ascends along the gradient of $\hat{h}_\theta(\bar{z})$ and simultaneously calculates $\hat{h}_\theta(\bar{z})$ in the same series of steps. This makes the algorithm quite efficient.

Our agent now solves the problem for a certain policy function class, e.g. a logistic function. To describe the policy function we use, first define a vector of basis functions $f(x_i, z_i, t_i)$ of dimension k over the space of (x, z, t). These can be transformations e.g. interactions of the states in x, z, and t. Using this vector of transformed states, we define the policy function as a logistic function or a softmax with binary actions:

$$\pi(a = 1|x_i, z_i, t_i; \theta) = \frac{\exp(\theta' f(x_i, z_i, t_i))}{1 + \exp(\theta' f(x_i, z_i, t_i))}$$

The reinforcement learning agent will learn these parameters $\theta \in R^k$ while exploring the problem and trying out (decreasingly) random actions. Her process of learning the policy function is faster if an estimate of the value function is used as a baseline or "critic" (see Section 2.4 for details). This value function approximator uses a different vector of basis functions $\phi(z_i, t_i)$, containing only transformations of budget and time, and has a parameter vector $\nu \in R^m$ which the agent learns as well while exploring the environment. Note hereby that we define a value function $h(\cdot)$ solely depending on budget and time with individual covariates integrated out.[3] We approximate the integrated value function with

[3]In computer game applications with such methods, rewards e.g. are the *true* score of the game. In our case we also have to estimate rewards coming from an RCT which introduced additional noise. We found this formulation of the value function without the individual covariates useful because it reduced the dimension of the problem. The advantage actor critic algorithm uses this value function in the parameter updates when learning the optimal policy function, see Section 2.4 for full details.

its basis functions and an additively linear form (the basis function can contain squares, interactions, etc.):

$$\mathbb{E}^x \left[v(x_i, z_i, t_i) \right] = h(z_i, t_i) \approx \nu' \phi(z_i, t_i)$$

For an in depth discussion of the theoretical setup written in continuous time and Hamilton-Jacobi-Bellman type PDEs, and with extensions not considered in the application, see Adusumilli et al. (2019).

2.3 Building the environment

Recall that our environment describes individuals who became unemployed and arrive sequentially at a policy maker's office to request training. After providing training, she observes an estimates of the treatment effect. To begin with, what exactly does a reinforcement learning "environment" mean? In a nutshell, it is given by possible actions, rewards, and transition rules between states. Living in the environment, a reinforcement learning agent learns optimal behaviour by repeatedly observing the current state s, picking an action a, and observing the resulting reward r and the next state s'. In the Atari games in Mnih et al. (2015), the state is the screen image, the actions are buttons, and the rewards are changes in the score. For us the state is given by $s = (x, z, t)$, i.e. the covariates x of the currently arriving individual which we sample from RCT data, the policy makers remaining budget z, and the time of arrival of the individual at the policy makers office t. The action a the policy maker can take is to either treat ($a = 1$) or not treat ($a = 0$) the individual. The observed reward r is an individual treatment effect estimate based on the RCT data. The next state s' is then given by (x', z', t'): The covariates of the individual arriving next being sampled (with replacement) from the RCT data, the next remaining budget, and the next time. And so on.

This section describes two parts that we need to complete modelling our environment: Rewards and arrivals of individuals at the policy maker's office. Section 2.3.2 is about how we estimate rewards/individual treatment effects. Section 2.3.3 then describes how we estimate arrival rates of different types of individuals over the year. This allows us to get transitions between states after the policy maker decided whether to provide treatment to an individual or not. Before this, section 2.3.1 briefly introduces the dataset.

2.3.1 Dataset

We use the popular dataset on randomized training provided under the JTPA, akin to e.g Kitagawa and Tetenov (2018) or Abadie et al. (2002). During 18 months, applicants who contacted job centers after becoming unemployed were randomized to either obtain support or not. Local centers could choose to supply one of the following forms of support: training, job-search assistance, or other support. Again akin to Kitagawa and Tetenov (2018), we consolidate all forms of support. Baseline information about the 20601 applicants was collected as well as their subsequent earnings for 30 months. We follow the sample selection procedure of Kitagawa and Tetenov (2018) and delete entries with missing earnings or education variables as well as those that are not in the analysis of the adult sample of Abadie et al. (2002). This results in 9223 observations for which we have data on their earnings 30 months later for both treatment and control group. We also store information on the individuals' education, previous earnings, age, and the date at which they took part in the RCT.

2.3.2 Estimating rewards from treatment effects

First we need to construct rewards for our reinforcement learning agent. After she decides to provide job training to an individual, she observes the estimated treatment effect of that individual as the reward. We estimate these treatment effects from the RCT data. In detail, the study recorded earnings Y measured 30 months after random treatment. Let us assume that the RCT consists of an iid draw of size n from the distribution F. The empirical distribution F_n of these observations is thus a good proxy for F. Let $w \in \{0, 1\}$ denote the treatment assignment in the RCT data with $w = 0$ indicating no treatment and $w = 1$ indicating treatment.

Standard OLS rewards We run two distinct standard OLS regressions: One on the control and one on the treatment observations. Each of the models predicts future earnings Y from the covariates X, i.e. age, previous earnings, and education: OLS model 1 predicts future earnings of the treated, OLS model 2 predicts future earnings of the untreated. After obtaining fitted values, we swap the models and predict counterfactual earnings for the respective other group. Subtracting treated vs. untreated predicted earnings for each individual gives the estimate treatment effect of providing job training to that individual. More formally, our regressions fit the conditional expectation function of

$\mu(x, w) = E[Y(w)|X = x, W = w]$. Hence model 1 yields the estimated linear function $\hat{\mu}(x, 1)$ and model 2 yields $\hat{\mu}(x, 0)$. The estimated individual treatment effect on future earnings for a person with covariates x are given by:

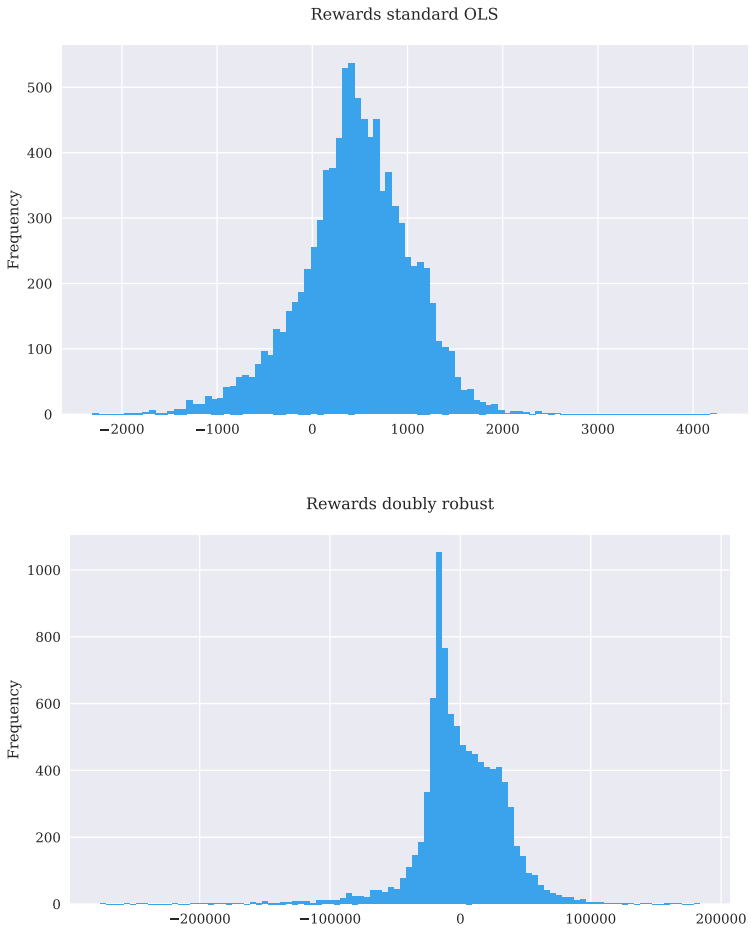
$$\hat{r}^{\text{standard}}_{\text{OLS}}(x, 1) = \hat{\mu}(x, 1) - \hat{\mu}(x, 0)$$

Doubly robust OLS rewards Due to its linear functional form and the limited amount of covariates available in this RCT, the benchmark treatment effects estimated with standard OLS are very smooth. This makes learning for our reinforcement learning agent relatively easy, however, it likely underestimates the true heterogeneity in treatment effects. To alleviate this problem, we recommend a doubly robust method to estimate $r(x, 1)$ (see Athey and Wager, 2018), and thus estimate a second set of rewards with the following equation:

$$\hat{r}^{\text{doubly robust}}_{\text{OLS}}(x, 1) = \hat{\mu}(x, 1) - \hat{\mu}(x, 0) + (2W - 1)\frac{Y - \hat{\mu}(x, W)}{W\hat{p}(x) + (1 - W)(1 - \hat{p}(x))}$$

The propensity score $\hat{p}(x)$ describes the estimated treatment probability of an individual conditional on his covariates. As we are using RCT data where treatment has been allocated at random to two thirds of participants, we know that $\hat{p}(x) = 2/3$ in our case. $\mu(x, w)$ remain the linear OLS function approximators we used before, however, the doubly robust reward procedure adds a third term to the equation. This term is positive if the individual has been treated and negative if not. It adds or subtracts a scaled version of the OLS residual to our previously estimated treatment effect. This allows for significant heterogeneity in the estimated treatment effects even if $\mu(\cdot)$ is linear and/or the information contained in the available covariates x is limited. In other words, in applications some people likely have very high or very low treatment effects and our reinforcement learning agent could not adjust to this likely variability if we used rewards as smooth as the standard OLS rewards. Indeed, while the doubly robust procedure yields consistent estimates, the benchmark standard OLS only does so if the heterogeneity structure of the true specifications is linear. We therefore expect different parameters in the policy functions and treatment decisions. Lastly, we apply our doubly robust procedure cross-fitted to not underestimate the uncertainty: We split the data into N folds, estimate the model $\hat{\mu}(\cdot)$ on N-1 folds, and

Figure 2.1: Reward histograms



then employ this to obtain the doubly robust rewards for the remaining fold (see Athey and Wager, 2018).

Lastly, with an empirical cost of treatment of \$774 subtracted, both the doubly robust and the standard OLS rewards are shown in the histograms in Figure 2.1 with 100 bins ($\mu_{\text{standard ols}} = 451.78$, $\sigma_{\text{standard ols}} = 577.60$, $\mu_{\text{doubly robust}} = 450.35$, $\sigma_{\text{doubly robust}} = 33019.26$). Doubly robust estimates therefore have much higher standard deviations and larger tails. In our algorithm we standardise rewards equalising variances, however, the very different higher order moments still prevail which makes training harder for doubly robust rewards (see Appendix 2.8.1 for the standardised rewards used in the algorithm).

2.3.3 Estimating arrival rates

The frequency at which people with given characteristics apply for training at the policy maker is not constant throughout the year. Individuals with different

occupations and characteristics have different seasonal patterns of unemployment. As the JTPA data contains information regarding when participants arrived, we can estimate Poisson processes that are changing over the course of the year. For this we first partition the data into clusters using k-median clustering on education, previous earnings, and age. Given the limited amount of data, the number of clusters we can reliably estimate is limited too. We chose to use four clusters. With more data, more clusters and hence a more detailed picture of differential arrival of applications becomes possible. Prior to the clustering which relies on comparable distances between observations, we standardize the variables.

Table 2.1 describes the clusters resulting from the JTPA example. Cluster 1 appears to contain predominantly candidates with high previous earnings. Cluster 2's distinguishing factor is the high age, and for cluster 3 it is few years of education. Cluster 4 contains young educated candidates with low previous earnings.

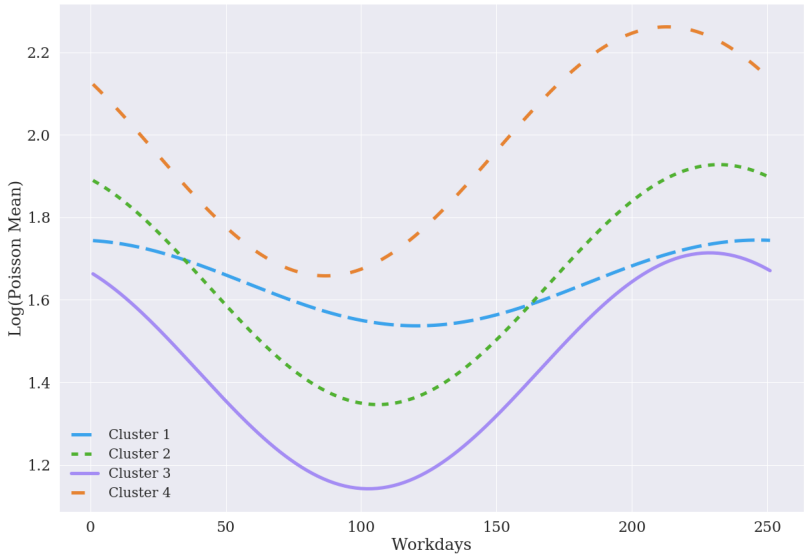
Table 2.1: Cluster summary statistics

	Cluster 1	Cluster 2	Cluster 3	Cluster 4
Age: Mean	31.8	44.9	31.3	26.9
Age: Min.	22	34	22	22
Age: Max.	63	78	57	34
Prev. Earnings: Mean	8999	1439	1413	1231
Prev. Earnings: Min.	3600	0	0	0
Prev. Earnings: Max.	63000	12000	9076	5130
Education: Mean	12.1	12.1	9.0	12.3
Education: Min.	7	8	7	11
Education: Max.	18	18	10	18
Observations	2278	2198	1698	3049

For each cluster, we then estimate the arrival probabilities. While we assume that seasonal patterns are constant across years, we allow for variation within a year. In particular, we specify the following functional form for the cluster-specific Poisson parameter: $\lambda_c(t) = exp\left(\beta_{0,c} + \beta_{1,c} sin(2\pi t) + \beta_{2,c} cos(2\pi t)\right)$, where t is normalized so that $t = 1$ corresponds to a year (note that time could still run for e.g. $t = 3.14$ years until the budget runs out). For each cluster, we obtain the estimates β_c (and hence $\lambda_c(t)$) using maximum likelihood. Figure 2.2 shows the estimated dynamic behavior of each cluster. People from cluster 1, for example, display a less pronounced seasonal pattern regarding their arrival rates than people from cluster 2. Our parametrisation of the lambda parameter with *sin* and *cos*, forces smooth arrival rates at the beginning and end of the year, yet, still allows for fitting quite flexible seasonal shapes within the year. As we assume no

trend in seasonal patterns, expected arrivals should be the same on 31 December at end of day as at the beginning 1 January. The functional form furthermore ensures that multiple years can pass until the agent depleted her budget, as e.g. $cos(0.5) = cos(4.5)$ will still both refer to the seasonal pattern in the middle of the year. Without smoothed start and end points, arrival rates could also exhibit strong jumps which would introduce discontinuities and further noise into the environment.

Figure 2.2: Clusters-Specific Arrival Rates over Time



2.3.4 An exemplary period

Now that we have built the individual parts, we can put them together and describe the full environment in which the reinforcement learning agent lives. The easiest way to understand it, is to go through an exemplary period that she faces. Recall that for us one period is the same as one arrival of a person, where the exact arrival time of that person is continuous.

– Say that two individuals have already arrived at the policy maker's office in this episode (one episode runs until the budget finishes). The time at which the third individual arrives is e.g. $t_3 = 0.003$ and the budget available $z_3 = 1 - 1c$ (i.e. one of the first two individuals has been treated). Next, we have to determine which individual is arriving third.

– Given the current time $t_3 = 0.003$, first compute the expected amount of arriving individuals. Using our arrival rate estimates for each cluster from Section 2.3.3, this is given by their sum $\hat{\lambda}_{sum}(t = 0.003) = \sum_{c=1}^{4} \hat{\lambda}_c(t = 0.003)$.

– Now determine from which cluster c the precise individual arrives. For this, draw the cluster from a multinominal distribution $c \sim \text{multinomial}(p_1, \ldots, p_C)$ where $p_c := \hat{\lambda}_c(t)/\hat{\lambda}_{sum}(t)$. Say we draw $c = 2$, i.e. the individual arrives from cluster two.

– From those observations contained in the RCT dataset which belong to cluster two, sample (with replacement) one individual. Store its covariates x_3 about age, previous earnings, and education as well as it's treatment effect $\hat{r}(x_3, a = 1)$ estimated in Section 2.3.2.

– Using the current estimate of the policy function, let it depict its treatment vs. non-treatment probabilities for this individual's characteristics x_3 at the current time t_3 and budget z_3. Say the policy function π outputs treatment $(a = 1)$ and non-treatment $(a = 0)$ probabilities of $[0.64, 0.36]$.

– Sample from these action probabilities (sampling ensures exploration and avoids quickly converging to a bad policy). Say we draw an $a = 1$, so the planner decides to treat and she observes the treatment effect $\hat{r}(x_3, a = 1)$ (with $a = 0$ the observed reward would have been zero).

– Now note that for the update of function approximators, we actually have to know the next arrival's time and budget (see Section 2.4 for details of the TD error). We already know that budget will be $z_4 = 1 - 2c$, however, we are missing t_4. For this, sample the time increment until the next individual arrives from an exponential function with this mean: $\Delta t \sim \text{Exponential}(\hat{\lambda}_{sum}(t = 0.003))$. Say the drawn value is $\Delta t = 0.0012$, so $t_4 = 0.003 + 0.0012 = 0.0042$.

– Using the values for t_3, t_4, z_3, z_4, x_3, $\hat{r}(x_3, a = 1)$, discount rate, and value function, then update the function approximators for policy and value functions. This update's magnitude is influenced by how much the observed reward for the individual deviated from the value function's prediction. On average, an update makes the action more likely if the reward/observed treatment effect was higher than expected by the value function and less likely if it was lower.

– Lastly, update cumulative discounting, and restart from above with period/arrival 4. The episode ends when the policy maker used up her budget,

i.e. $z = 0$. Keeping all function approximator parameter estimates, we then restart with a new episode with fresh budget $z_1 = 1$ and $t_1 = 0$ until the budget is depleted again. And so on. After many episodes of training in the environment, her policy and value function estimators should converge.

Equipped with this description of our environment and the timing, the next section discusses the algorithm's pseudo-code.

2.4 Algorithm

The reinforcement learning algorithm we use is based on Chapter 13 in Sutton and Barto (2018) and comes from the class of policy gradient methods in which function approximators for policy functions are learned directly through training (for an excellent description see Sutton and Barto, 2018). It is called an actor-critic method, where, in very short, actor is a metaphor for the policy function and critic for the value function which serves as the baseline in such algorithms. These two functions interact in a particular way: If the reward observed after sampling an action from the policy function (i.e. the actor) is larger than the current estimate of the state's value (obtained from the value function, i.e. the critic), the gradient update makes this action on average more likely in the future. This mechanism is represented in the temporal difference (TD) error δ in Algorithm 1. Recall that our value function approximator is $\nu^\intercal \phi_{z,t}$ with ν being a parameter vector and ϕ being the basis function with transformations of states. The predicted reward of today's state is given by the current value function estimate $\nu^\intercal \phi_{z,t}$, also named the baseline. After observing the reward R, however, we have a potentially better estimate of the value of today's state: The observed reward plus the discounted value of tomorrows state. The TD error depicts the difference of these two competing estimates: $\delta = R + e^{-\beta(t'-t)} \nu^\intercal \phi_{z',t'} - \nu^\intercal \phi_{z,t}$. Both parameter updates for policy and value function are proportional to this error. If the policy function has chosen an action which achieved a reward that indicates a lower state value than that of the critic, this action becomes less likely.

After this discussion of its main mechanism, let us now delve into the algorithm's pseudo code a bit more deeply. For a detailed derivation and discussion see (Sutton and Barto, 2018, pages 321 - 332) and the paper Adusumilli et al. (2019). We use the book's actor critic version (Sutton and Barto, 2018, pages 331 - 332) and amend it by several features commonly employed in the literature. First, we add batch updates instead of plain stochastic gradient descent. Particularly

for the case of doubly robust rewards which are very volatile, this averages out some erratic updates: Batches sum up all function approximator updates for B (e.g. 128 or 1024) periods before updating the function approximator only once - some extreme positive and negative updates will average out before influencing the update policy. Next, we parallelise the training following Mnih et al. (2016). In detail, we spawn P parallel processes each with a reinforcement learning agent living and training in her own environment. Yet, while experiencing different episodes of training, they all access and update the same global function approximators for policy and value functions. Despite not using deep neural networks (we use a logistic function and additively linear function approximation with squares, interactions, etc.), this is very useful. The parallelisation and decorrelation of function approximator updates does make training and convergence faster.

In the full Algorithm 1, each process therefore runs its own world with different people arriving etc., but accesses and updates the the same function approximators for policy and value functions. Training runs for a total of E (e.g. $20,000$) episodes with a new training episode starting whenever the budget has been depleted. Each episode consists of periods with one period being one arrival. Each arrival is associated with an arrival time t (at the end of an episode, t is reset to 0). Hence, while arrivals are discrete, time, at least theoretically, is not.

Algorithm 1's pseudo code is now in essence a more formal version of the exemplary period described in Section 2.3.4: While there is still budget within an episode, first draw the amount of individuals expected to arrive at the current time of the year and then sample a precise cluster and individual from that cluster. Next sample an action using the current policy estimate, store the resulting estimated treatment effect, draw the arrival time of the next individual and the budget available then, and with their help compute the TD error. Now one can compute the function aproximator updates $\alpha_\theta I \delta \frac{\nabla_\theta \pi(a|s;\theta)}{\pi(a|s;\theta)} = \alpha_\theta I \delta \nabla_\theta \ln \pi(a|s;\theta)$ and $\alpha_\nu \delta \nabla_\nu (\nu^\intercal \phi_{z,t}) = \alpha_\nu \delta \phi_{z,t}$ for policy and value functions and store them in the batch updates (I captures cumulative discounting). After B periods/arrivals, update the function approximators. Once budget is not enough to treat one further individual, i.e. $z < c$, terminate the training episode and start a new one.

In the pseudo-code, the term b_n represents the approximate amount of individuals arriving in a year, given our previously estimated arrival rates. Note that we should normalise the cost of treatment c and the reward r to be of the same order of magnitude as Δt. Because we set a value of $t = 1$ corresponding to one year, and there are approximately b_n people arriving per year, Δ_t is of the order of $1/b_n$. Hence we should also divide c an r by b_n to make them of a comparable

order of magnitude. Finally, in our implementation, since we estimated arrivals from daily data, we also have to adjust the exponential draws to be consistent with our definition of time in years. Appendix 2.8.3 describes these adjustments in greater detail.

2.5 Parametrisation

2.5.1 Environment

We set the budget such that 1600 people can be treated, which is about a quarter of the expected number of arrivals in a year (the number of 6400 individuals is a rough approximation of the yearly arrivals given our estimated Poisson rates which themselves are approximations of arrival in the data). In our environment this is achieved by setting z as $z_1 = 0.25$ and the cost of treatment to $c = 1/6400$. With approximately 6400 arrivals per year, a budget of 1 would allow to treat everyone with at this cost, so a budget of 0.25 means that the planner can treat around 25% of arrivals which equals 1600 individuals. A year is set to have 252 working days and each day is discretised to have 100 time increments. Time is used such that $t = 1$ is one year, $t = 2.5$ are two and a half years, and so on. Budget could last for many years if the policy maker decided to treat few arrivals. Besides normalising the cost by 6400 as discussed above, we use normalisations for expected arrivals and time increments (see Appendix 2.8.3 for an in detail explanation of these normalisations). We also use a discount factor of $\beta = -\log(0.9)$, which implies an annualised discount rate of around 10% (since t=1 corresponds to an year). The episode terminates when all budget is used up. Individual characteristics that our policy maker observes for each arriving individual are $\mathbf{x}$ = (age, education, previous earnings)'. Instead of t we use $\cos(2\pi t)$ in the function approximation. As it can take multiple years until the budget is used up, this periodic form ensures that the policy maker always knows at which time of the year she is (and hence can implicitly take into account the arrival rates at that time). We use these covariates as well as interactions to construct the following basis functions for the value and policy function approximators:

$$\text{Policy basis function: } f(\mathbf{x}, z, t) = \begin{pmatrix} 1, & \mathbf{x}', & z, & \cos(2\pi t), & z\mathbf{x}', & \cos(2\pi t)\mathbf{x}' \end{pmatrix}'$$

$$\text{Value basis function: } \phi(z, t) = \begin{pmatrix} z, & z\sin(2\pi t), & z\cos(2\pi t), & z^2\cos(2\pi t), & z^2, & z^3, & z^4 \end{pmatrix}'$$

Algorithm 1: Parallel Actor-Critic with clusters

Initialise policy parameter weights $\theta \leftarrow 0$
Initialise value function weights $\nu \leftarrow 0$
Batch size B
Clusters $c = 1, 2, \ldots, C$
Cluster specific arrival rates $\lambda_c(t)$

For $p = 1, 2, \ldots$ processes, launched in parallel, each using and updating the same global parameters θ and ν:

Repeat for E **training episodes:**

Reset budget: $z \leftarrow z_0$
Reset time: $t \leftarrow t_0$
$I \leftarrow 1$

While $z > c$**:**

batch_policy_upates $\leftarrow 0$

batch_value_upates $\leftarrow 0$

For $b = 1, 2, ..., B$**:**

$\lambda(t) \leftarrow \sum_c \lambda_c(t)$ (Calculate arrival rate for next individual)

$c \sim$ multinomial$(p_1, \ldots, p_C)$ (where $p_c := \hat{\lambda}_c(t)/\hat{\lambda}(t)$)

$x \sim F_{n,c}$ (Draw new covariate at random from data cluster c)

$a \sim \pi(a|s;\theta)$ (Draw action, note: $s = f(x, z, t)$)

$R \leftarrow \hat{r}(x, a)/b_n$ (with $R = 0$ if $a = 0$)

$\Delta t \sim$ Exponential$(\lambda(t))$ (Sample time increment until next arrival)

$t' \leftarrow t + \Delta t$

$z' \leftarrow z - \mathbb{I}\{a = 1\}c/b_n$

$\delta \leftarrow R + \mathbb{I}\{z' > c\}e^{-\beta(t'-t)}\nu^\intercal\phi_{z',t'} - \nu^\intercal\phi_{z,t}$ (TD error)

batch_policy_updates $\leftarrow$
batch_policy_updates $+ \alpha_\theta I\delta\nabla_\theta \ln\pi(a|s;\theta)$

batch_value_updates $\leftarrow$ batch_value_updates $+ \alpha_\nu\delta\phi_{z,t}$

$z \leftarrow z'$

$t \leftarrow t'$

$I \leftarrow e^{-\beta(t'-t)}I$

If $z < c$, break the batch **For**

Globally update: $\nu \leftarrow \nu +$ batch_value_updates
Globally update: $\theta \leftarrow \theta +$ batch_policy_updates

2.5.2 Algorithm

Our implementation has 23 reinforcement learning agents training in parallel. For the standard OLS rewards, we set the policy function learning rate to $\alpha_\theta = 0.3$, the value function learning rate to $\alpha_\nu = 0.6$, the batch size to $B = 1024$, and training depicted here ran for a bit over $E = 10,000$ episodes. Due to the much larger tails in the doubly robust rewards, the agent has to train longer. Here we again set the policy function learning rate to $\alpha_\theta = 0.3$, the value function learning rate to $\alpha_\nu = 0.6$, the batch size to $B = 1024$, and training depicted here ran for a bit over $E = 40,000$ episodes. When evaluating the current policy every approximately 500 episodes, we fix the parameters and compute the average return achieved over 500 evaluation episodes. This means that each point in Figure 2.9 is an average over 500 evaluation episodes. For standard OLS rewards, the figure would already look as steady if we took an average over many fewer evaluation episodes for each point. Yet, the large tails of the doubly robust rewards require an average over 500 episodes for each evaluation step / point in the figure. Taking this average over evaluation periods smooths the reward trajectory and makes the learning visible. In both cases we normalise average cumulative welfare per episode so that following a random policy would yield an average episode welfare of one.[4]

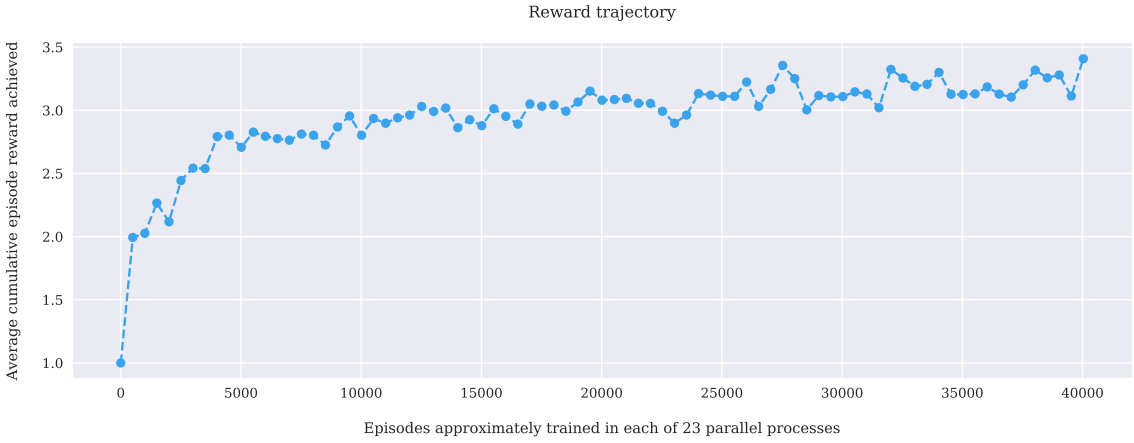
2.6 Results

Equipped with the descriptions in Section 2.5, we can now proceed directly to the main results. After experiencing the environment for a few thousand episodes and understanding its dynamics, the agent relatively quickly figures out a policy that yields a reward of around 2.75 to 3.25 times higher than the random benchmark reward. Figure 2.9 shows these reward trajectories over $40,000$ (doubly robust rewards) and $10,000$ (standard OLS rewards) episodes of training.

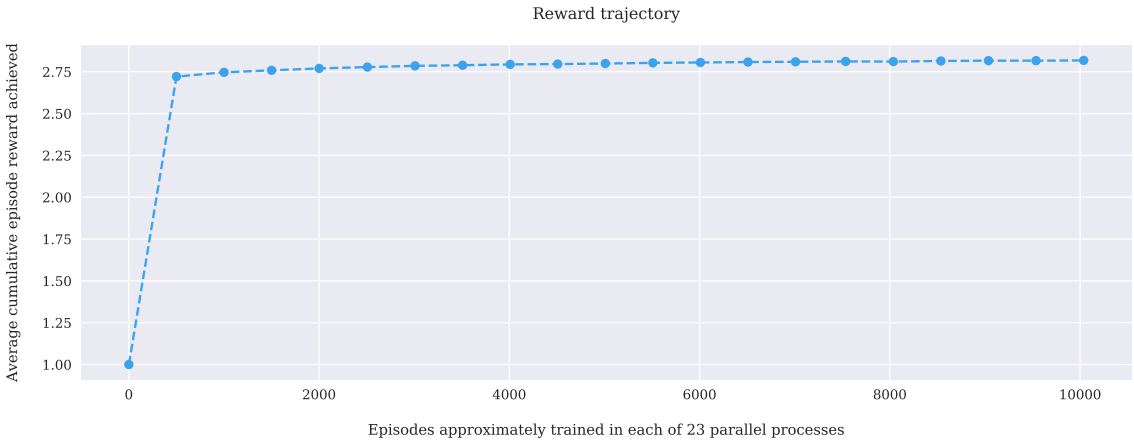
Welfare (average cumulative reward achieved per episode) begins to stabilise after some initial training episodes, particularly in the OLS case. For coefficients on the other hand, there might be multiple combinations leading to the same welfare. Yet, judging from Figure 2.4, also coefficient paths depict generally decreasing slopes over the case of learning.

[4]We estimated the reward for this benchmark random welfare by running the algorithm for 100000 episodes with a 50/50 policy and take a mean over episode welfare achieved.

Figure 2.3: Reward trajectories obtained with doubly robust and standard OLS reward over the course of training

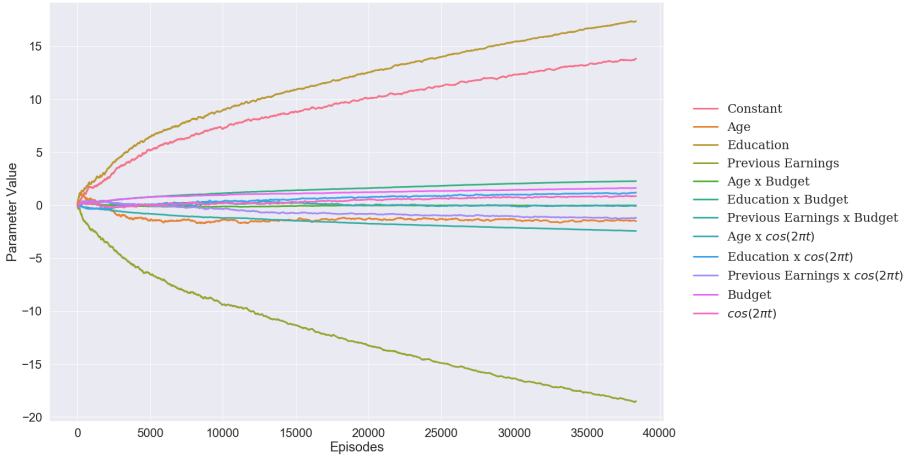


(a) Doubly robust rewards

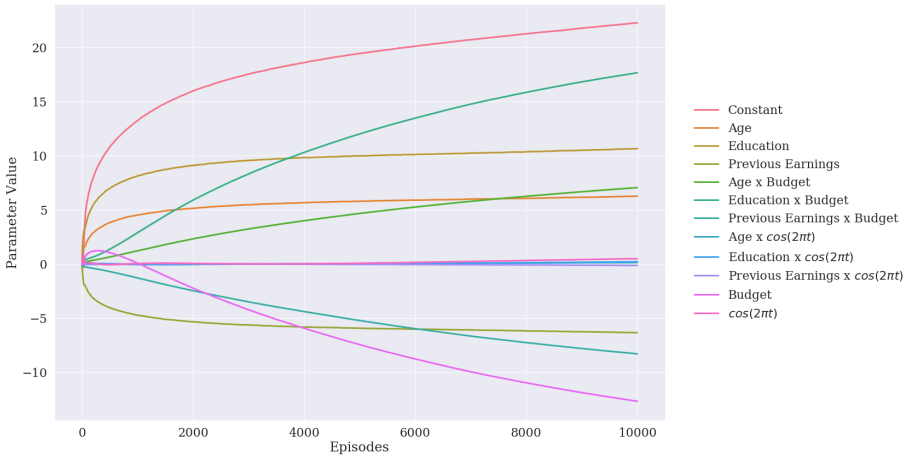


(b) Standard OLS rewards

Figure 2.4: Policy function parameters over the course of training



(a) Doubly robust rewards



(b) Standard OLS rewards

The key result of running this algorithm is a policy function estimate. This policy function represents the optimal treatment probabilities for candidates conditional on characteristics. Since it is a function for actions in a dynamic context, time and budget affect how the characteristics affect the treatment decision. Specifically, if we slightly change the notation of the θ subscripts such that $log(\frac{\pi}{1-\pi}) = \theta_0 + \theta_{A1}age + \theta_{A2}age * z + \theta_{A3}age * cos(2\pi t) + ...$, then age affects the treatment decision with the parameter $\theta_{Age}(z,t) = \theta_{A1} + \theta_{A2}z + \theta_{A3}cos(2\pi t)$. Figure 2.5 illustrate how $\theta_{Age}(z,t)$ and the analogously $\theta_{Education}(z,t)$ and $\theta_{Prev.Education}(z,t)$ vary with budget and time.

Note that time is periodic, so a low budget indicates that the end of an episode is near, while time reflects differences in seasonal patterns related to the non-constant arrival rates. An episode lasts for approximately three years in the case with standard OLS reward estimates, and around one year with doubly robust rewards. Consequently, the typical path in the left panel of Figure 2.5 is from the top-left to bottom-right. In the right panel, it also starts top-left, but horizontally crosses the figure three times until terminating bottom-right. The heat structure of the plots indicates how large the coefficient value is, i.e. how strongly this variable influences the decision of treatment. For example, in the case of doubly robust rewards older individuals are more likely to be treated in summer (the middle of the figure), although the difference in coefficients (-1.44 vs -1.60) is small. Looking at plot (d) as another example (now with standard OLS rewards), it indicates that people with higher education are more likely to be treated when there is still relatively much budget available.

In order to further interpret the resulting (final) policy function, we use that function in 100 evaluation episodes and record the treated candidates to create additional figures below. As a measure of selectivity, we record how many candidates were declined before one was treated. Figure 2.7 illustrates for each treated person in the 100 episodes how many candidates were declined since the last treatment, plotted against the remaining budget. In the case of standard OLS rewards, the algorithm becomes more selective when the budget is scarce. This is in line with economic considerations due to discounting. The inter-temporal trade-off is between treating a person now versus treating a person at the end of the budget. If the remaining budget is large, the reward from treating a person at the end of the budget is discounted more heavily compared to the case where the remaining budget is small. In the case of doubly robust rewards, this trade-off does not seem to be important and the selectivity appears to be quite constant. There are 160,000 points depicted in Figure 2.7. Figure 2.8 simplifies this illustration by imposing a linear/quadratic structure.

Table 2.2 shows again the correlation between the number of declined candidates with the remaining budget, but also with $cos(2\pi t)$. For the standard OLS rewards, the correlation with $cos(2\pi t)$ is low: The depletion of budget appears to be the main driver of increased selectivity. For the doubly robust rewards, the correlations are almost zero - again selectivity appears to be constant.

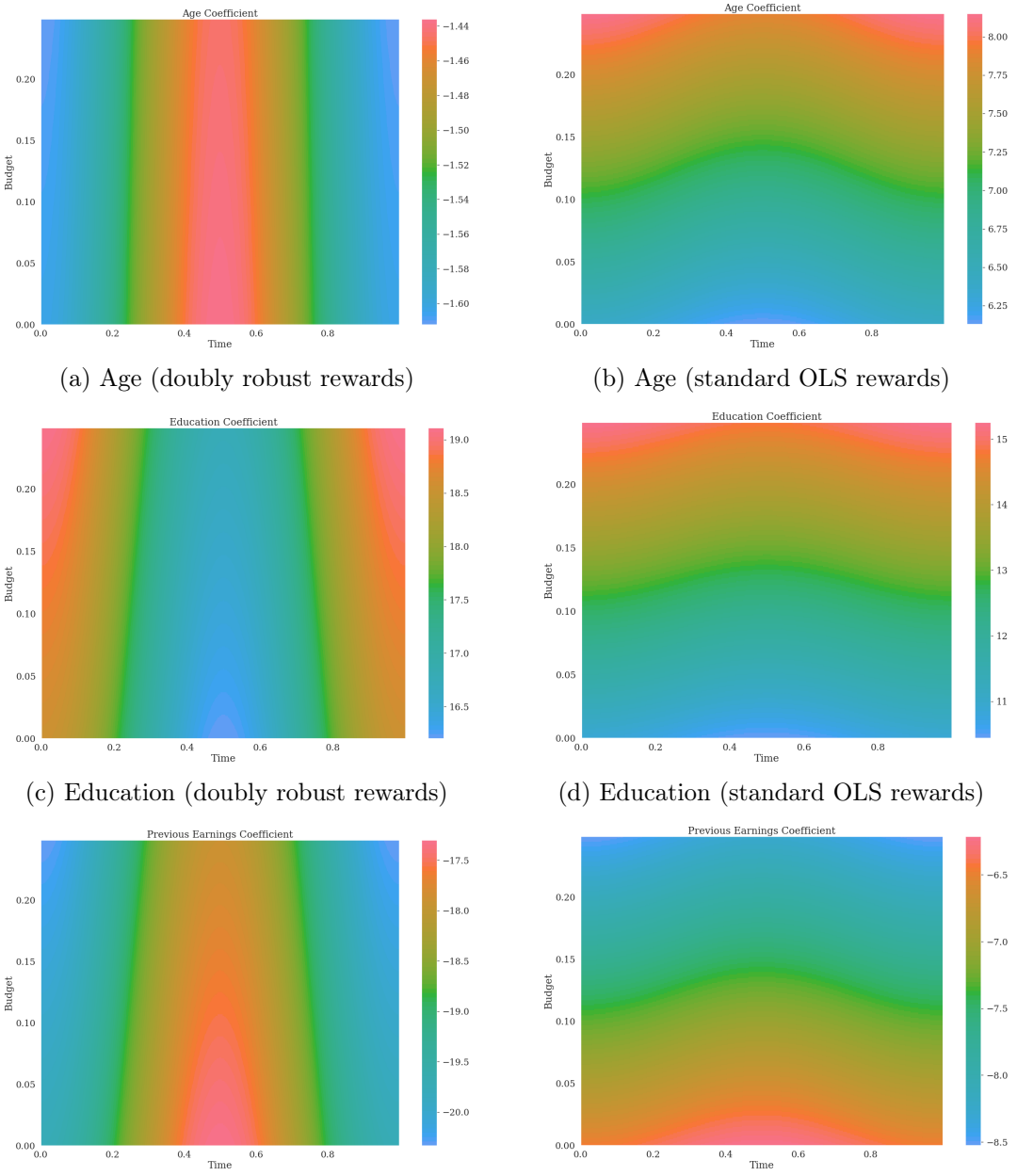
Figure 2.6 provides further illustration of the policy function's behaviour throughout the year. For the standard OLS rewards, an increasing number of rejections over the year is observable. As seasonality has mechanically been smoothed across seasons using the cosine function, the sharp difference between December and January in Figure 2.6 is further evidence against the effect of seasonality. It does support the notion of increased selectivity with depleted budget as every episode starts on January 1st with a complete budget - i.e. months later in the year are generally months with less budget.

For the doubly robust rewards, the picture resembles Figure 2.7. It depicts hardly any changes in selectivity - except possibly a slight seasonality-effect.

The differences between the optimal policy for doubly robust and standard OLS rewards appear to be large – after all, they are entirely different distributions. We offer the following interpretation. There are considerably more outliers for doubly robust rewards. Consequently, it appears optimal for these rewards to focus entirely on avoiding negative outliers - which can imply mild seasonality as the relative cluster-arrival rates change slightly over the year (e.g. how many cluster-1-candidates arrive per cluster-2-candidate). Considerations regarding discounting (within the 11-12 months until budget runs out) appear to be overpowered by this focus. Hunting positive outliers, most pronounced with the DR rewards, might be part of the explanation too, but given that the algorithm is not very selective (treating every fourth person), at this stage of training it does not seem to be the main task. Yet, training the algorithm for more periods or with different starting values, can imply, due to the tails in the DR rewards, that the agent would reach higher welfare when only treating outlier individuals in the right tail and waiting for many years.[5] For OLS rewards, outliers play a much smaller role, so increased selectivity with depleted budget (see above) is more important.

[5]In a future version of Adusumilli et al. (2019), we alleviate this concern by adding a time constraint forcing the agent to spend all budget within e.g. a year rather than potentially making it optimal to wait for outliers. Another option could be to set a larger discount rate.

Figure 2.5: Coefficient interactions in the resulting policy function



(a) Age (doubly robust rewards)

(b) Age (standard OLS rewards)

(c) Education (doubly robust rewards)

(d) Education (standard OLS rewards)

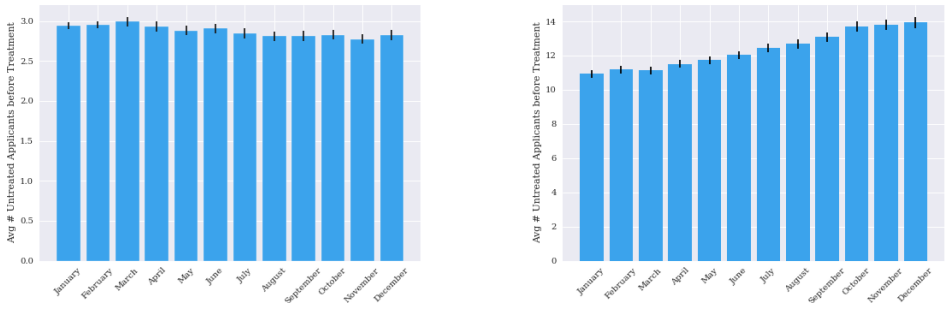
(e) Previous earnings (doubly robust rewards)

(f) Previous earnings (standard OLS rewards)

Table 2.2: Correlation of the average number of rejected individuals prior to a treatment with time and budget of the policy functions

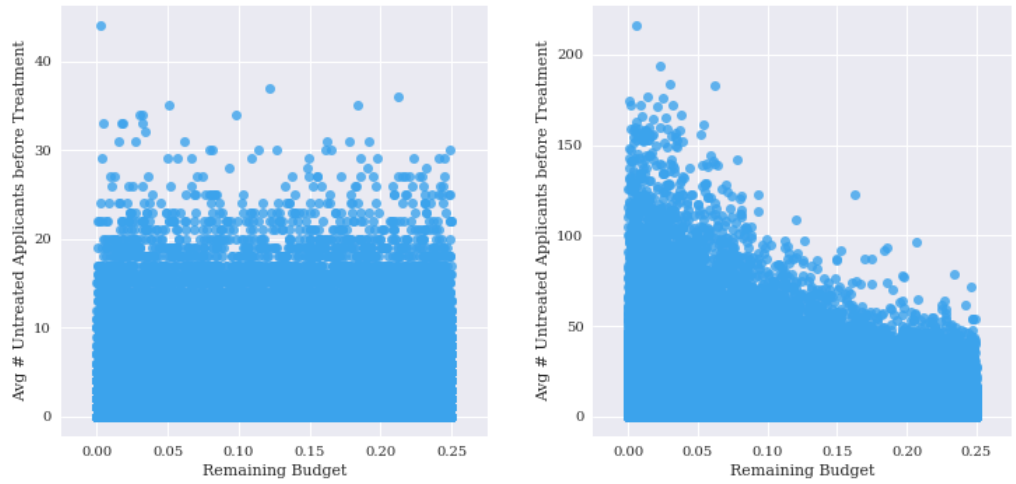
	Doubly Robust Rewards	Standard OLS Rewards
Remaining Budget	0.008	-0.353
$cos(2\pi t)$	0.005	-0.003

Figure 2.6: Seasonal differences in the average number of rejected individuals prior to a treatment



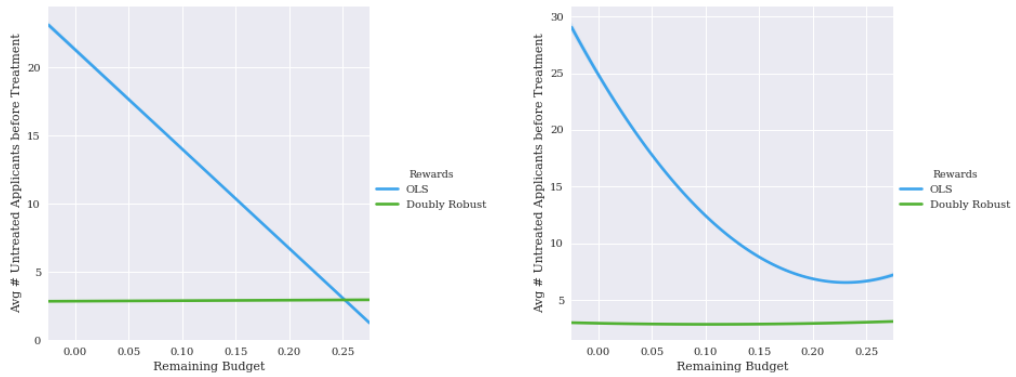
(a) Doubly robust reward estimates (b) Standard OLS reward estimates

Figure 2.7: Remaining Budget and Average Number of Rejected Individuals Prior to a Treatment across 100 Simulations



(a) Doubly robust reward estimates (b) Standard OLS reward estimates

Figure 2.8: Effect of Remaining Budget on Average Number of Rejected Individuals Prior to a Treatment (first and second order)



(a) Doubly robust reward estimates (b) Standard OLS reward estimates

2.7 Conclusion

This paper gives an example of how methods from artificial intelligence and reinforcement learning can try to improve the allocation of scarce resources when economic policy makers face complicated dynamic problems. Through the use of RCT data on job trainings, we model an environment in which unemployed individuals arrive sequentially at a social planner's office. With a limited budget and some knowledge about the individuals, she picks an allocation to maximise average welfare. For this the planner also has to understand the dynamics of the environment. As the rewards she observes are estimates of individual level treatment effects from the RCT, we also provide an example of how to link reinforcement learning with causal inference in an economic context.

An assumption we make is that individuals do not strategically influence their unemployment. Strategic behaviour on the side of the treated would have to be modelled additionally. In this case, individuals would have to form beliefs over the actions of the reinforcement learning agent / policy maker. Another limitation is that these algorithms are able to find very good solutions but typically not globally optimal ones as they can get stuck on saddle paths or in local optima. Yet, arguably their main appeal is that they can find these very good solutions to problems otherwise out of reach due to vast state spaces. A principal reason why we chose these methods is this scalability which would allow to apply them to real world settings and RCT datasets with very large amounts of covariates/states in the future. Lastly, in Section 2.6, we have seen that cumulative welfare achieved unambiguously rises through training, however, that the reasons for the agent's decisions can be difficult to understand. This points to a limitation of our approach that many methods from reinforcement learning or deep learning share. It can be ambiguous *why* an algorithm makes decisions. Any implementation of such algorithms in practise would therefore require significant governance and raise difficult questions: Do we understand its decisions well enough and could e.g. biased training data lead to algorithm bias? Who supervises the algorithm and who is eventually responsible for its actions? Once trained, the method is also much more scalable than the decisions of one human being. It therefore requires substantial scrutiny before and during implementation. If applied carefully in very well controlled environments, comparable methods might some day be able to assist policy makers to achieve higher welfare in very complicated dynamic problems and to make more consistent decisions across individuals.

References

Abadie, Alberto, Joshua Angrist, and Guido Imbens, "Instrumental Variables Estimates of the Effect of Subsidized Training on the Quantiles of Trainee Earnings," *Econometrica*, jan 2002, *70* (1), 91–117.

Abbeel, Pieter, Adam Coates, Morgan Quigley, and Andrew Y. Ng, "An Application of Reinforcement Learning to Aerobatic Helicopter Flight," in B. Schölkopf, J. C. Platt, and T. Hoffman, eds., *Advances in Neural Information Processing Systems 19*, MIT Press, 2007, pp. 1–8.

Adusumilli, Karun, Friedrich Geiecke, and Claudio Schilter, "Dynamically optimal treatment allocation using Reinforcement Learning," *arXiv e-prints*, Apr 2019, p. arXiv:1904.01047.

Agarwal, Alekh, Daniel Hsu, Satyen Kale, John Langford, Lihong Li, and Robert Schapire, "Taming the monster: A fast and simple algorithm for contextual bandits," in "International Conference on Machine Learning" 2014, pp. 1638–1646.

Athey, Susan and Stefan Wager, "Efficient Policy Learning," *arXiv:1702.02896*, feb 2018.

Bahdanau, Dzmitry, Philemon Brakel, Kelvin Xu, Anirudh Goyal, Ryan Lowe, Joelle Pineau, Aaron Courville, and Yoshua Bengio, "An actor-critic algorithm for sequence prediction," *arXiv preprint arXiv:1607.07086*, 2016.

Benitez-Silva, Hugo, George Hall, Günter J Hitsch, Giorgio Pauletto, and John Rust, "A comparison of discrete and parametric approximation methods for continuous-state dynamic programming problems," *manuscript, Yale University*, 2000.

Bhatnagar, Shalabh, Richard S. Sutton, Mohammad Ghavamzadeh, and Mark Lee, "Natural Actor-Critic Algorithms," *Automatica*, nov 2009, *45* (11), 2471–2482.

Bhattacharya, Debopam and Pascaline Dupas, "Inferring Welfare Maximizing Treatment Assignment under Budget Constraints," *Journal of Econometrics*, 2012, *167*, 168–196.

Chakraborty, Bibhas and Susan A. Murphy, "Dynamic Treatment Regimes," *Annual Review of Statistics and Its Application*, jan 2014, *1* (1), 447–464.

Chamberlain, Gary, "Bayesian Aspects of Treatment Choice," in John Geweke, Gary Koop, and Herman Van Dijk, eds., *The Oxford Handbook of Bayesian Econometrics*, Oxford University Press, sep 2011, pp. 11–39.

Dimakopoulou, Maria, Susan Athey, and Guido Imbens, "Estimation considerations in contextual bandits," *arXiv preprint arXiv:1711.07077*, 2017.

Dudik, Miroslav, Daniel Hsu, Satyen Kale, Nikos Karampatziakis, John Langford, Lev Reyzin, and Tong Zhang, "Efficient optimal learning for contextual bandits," *arXiv preprint arXiv:1106.2369*, 2011.

Hirano, Keisuke and Jack R. Porter, "Asymptotics for Statistical Treatment Rules," *Econometrica*, sep 2009, *77* (5), 1683–1701.

Kitagawa, Toru and Aleksey Tetenov, "Who Should Be Treated? Empirical Welfare Maximization Methods for Treatment Choice," *Econometrica*, mar 2018, *86* (2), 591–616.

Laber, Eric B, Daniel J Lizotte, Min Qian, William E Pelham, and Susan A Murphy, "Dynamic treatment regimes: Technical challenges and applications," *Electronic journal of statistics*, 2014, *8* (1), 1225.

Lei, H., I. Nahum-Shani, K. Lynch, D. Oslin, and S.A. Murphy, "A "SMART" Design for Building Individualized Treatment Sequences," *Annual Review of Clinical Psychology*, apr 2012, *8* (1), 21–48.

Manski, Charles F., "Statistical Treatment Rules for Heterogeneous Populations," *Econometrica*, 2004, *72* (4), 1221–1246.

Mnih, Volodymyr, Adrià Puigdomènech Badia, Mehdi Mirza, Alex Graves, Timothy P. Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu, "Asynchronous Methods for Deep Reinforcement Learning," *arXiv e-prints*, Feb 2016, p. arXiv:1602.01783.

Mnih, Volodymyr, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis, "Human-

Level Control through Deep Reinforcement Learning," *Nature*, feb 2015, *518* (7540), 529–533.

_ , **Nicolas Heess, Alex Graves et al.**, "Recurrent models of visual attention," in "Advances in neural information processing systems" 2014, pp. 2204–2212.

Murphy, S. A., "An Experimental Design for the Development of Adaptive Treatment Strategies," *Statistics in Medicine*, may 2005, *24* (10), 1455–1481.

Russo, Daniel and Benjamin Van Roy, "An information-theoretic analysis of thompson sampling," *The Journal of Machine Learning Research*, 2016, *17* (1), 2442–2471.

Silver, David, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton et al., "Mastering the game of go without human knowledge," *Nature*, 2017, *550* (7676), 354.

_ , **Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel et al.**, "A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play," *Science*, 2018, *362* (6419), 1140–1144.

Stoye, Jörg, "Minimax Regret Treatment Choice with Finite Samples," *Journal of Econometrics*, jul 2009, *151* (1), 70–81.

_ , "New Perspectives on Statistical Decisions Under Ambiguity," *Annual Review of Economics*, sep 2012, *4* (1), 257–282.

Sutton, Richard S and Andrew G Barto, *Reinforcement learning: An introduction*, MIT press, 2018.

_ , **David A McAllester, Satinder P Singh, and Yishay Mansour**, "Policy gradient methods for reinforcement learning with function approximation," in "Advances in neural information processing systems" 2000, pp. 1057–1063.

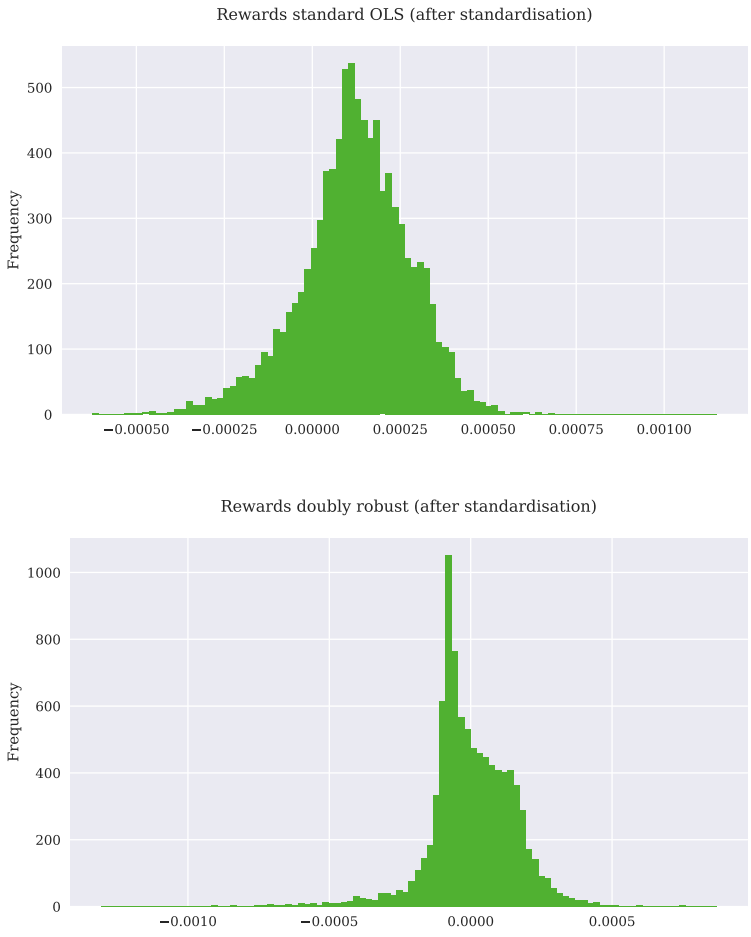
Tetenov, Aleksey, "Statistical treatment choice based on asymmetric minimax regret criteria," *Journal of Econometrics*, 2012, *166* (1), 157–165.

2.8 Appendix

2.8.1 Standardised rewards

Both rewards are now normalised by their standard deviation and an additional normalising factor of 1/6400 with 6400 being a rough approximation of individuals arriving each year given our lambda estimates. When normalising the second moment, the rewards still show very different patterns in higher moments.

Figure 2.9: Reward histograms standardised



2.8.2 Basic algorithm

This pseudo-code describes a basic version of our algorithm. It does not include parallel processes, arrivals from different clusters, or batch updates.

Algorithm 2: Actor-critic

Initialise policy parameter weights $\theta \leftarrow 0$

Initialise value function weights $\nu \leftarrow 0$

Repeat for E training episodes:

Reset budget: $z \leftarrow z_0$

Reset time: $t \leftarrow t_0$

$I \leftarrow 1$

While $z > c$:

$x \sim F_n$ (Draw new covariate at random from data)

$a \sim \pi(a|s;\theta)$ (Draw action, note: $s = f(x, z, t)$)

$R \leftarrow \hat{r}(x, a)/b_n$ (with $R = 0$ if $a = 0$)

$\Delta t \sim \text{Exponential}(\lambda(t))$ (Draw time increment)

$t' \leftarrow t + \Delta t$

$z' \leftarrow z - \mathbb{I}\{a = 1\}c/b_n$

$\delta \leftarrow R + \mathbb{I}\{z' > c\}e^{-\beta(t'-t)}\nu^\intercal\phi_{z',t'} - \nu^\intercal\phi_{z,t}$ (Temporal-Difference error)

$\theta \leftarrow \theta + \alpha_\theta I\delta\nabla_\theta \ln\pi(a|s;\theta)$ (Update policy parameter)

$\nu \leftarrow \nu + \alpha_\nu\delta\phi_{z,t}$ (Update value parameter)

$z \leftarrow z'$

$t \leftarrow t'$

$I \leftarrow e^{-\beta(t'-t)}I$

2.8.3 Normalisations used in the implementation

Both rewards and costs are normalised by $b_n = 6400$, a rough approximation of people arriving in a year given our estimated arrival rates. Time increments require a few additional adjustments. Arrival rates are estimated from daily data, however our unit of measure of time is in years. So to make them comparable, we first divide the day into 100 time parts. Thus the exponential draws are now measured in one-hundreds of days. To give an example, we might for example draw a value of 50 for an exponential draw: Fifty hundreds of a day until the next

person arrives, or half a day. As our time t however runs in years, we need to divide these increments by (100×252) (i.e. dividing the draw by 100 increments per day to first bring it down to daily time and then by 252 to eventually match yearly time). It would then take $50/25200$ years until the next individual arrives. With $t = 1$ being one year, the correct division for the exponential draws is 25200.

Chapter 3

Emotional Dynamics

3.1 Introduction

"Even apart from the instability due to speculation, there is the instability due to the characteristic of human nature that a large proportion of our positive activities depend on spontaneous optimism rather than mathematical expectations, whether moral or hedonistic or economic. Most, probably, of our decisions to do something positive, the full consequences of which will be drawn out over many days to come, can only be taken as the result of animal spirits - a spontaneous urge to action rather than inaction, and not as the outcome of a weighted average of quantitative benefits multiplied by quantitative probabilities." (Keynes, 1936, pp.161-162)

This essay introduces a mechanism through which perpetual business cycles can result and be maintained endogenously. Individuals share economic sentiments similarly to diseases in a stylised model of disease transmission. Two opposing sentiments, optimism and pessimism, alternately expand and contract through a population in waves. Both sentiments change aggregate demand as optimistic consumers have a bias to increase current consumption and pessimistic consumers have a bias to decrease current consumption relative to optimal levels. While the mechanism is more general, I illustrate it in a simple New Keynesian model with boundedly rational agents which are unaware of the consumer's sentiment bias. As the model has a representative consumer, I use the network structure as a computational tool to approximate an average sentiment bias for the economy. The sentiment sharing on the network can endogenously generate cyclical motion in short term aggregate output as an inherent feature of such a economy. To be put into motion, the model only requires some initial distribution of optimistic or pessimistic nodes. In addition, the pattern of macroeconomic sentiment fluctua-

tions can also be maintained and made more realistic with additional, but small, exogenous real output variations every period.

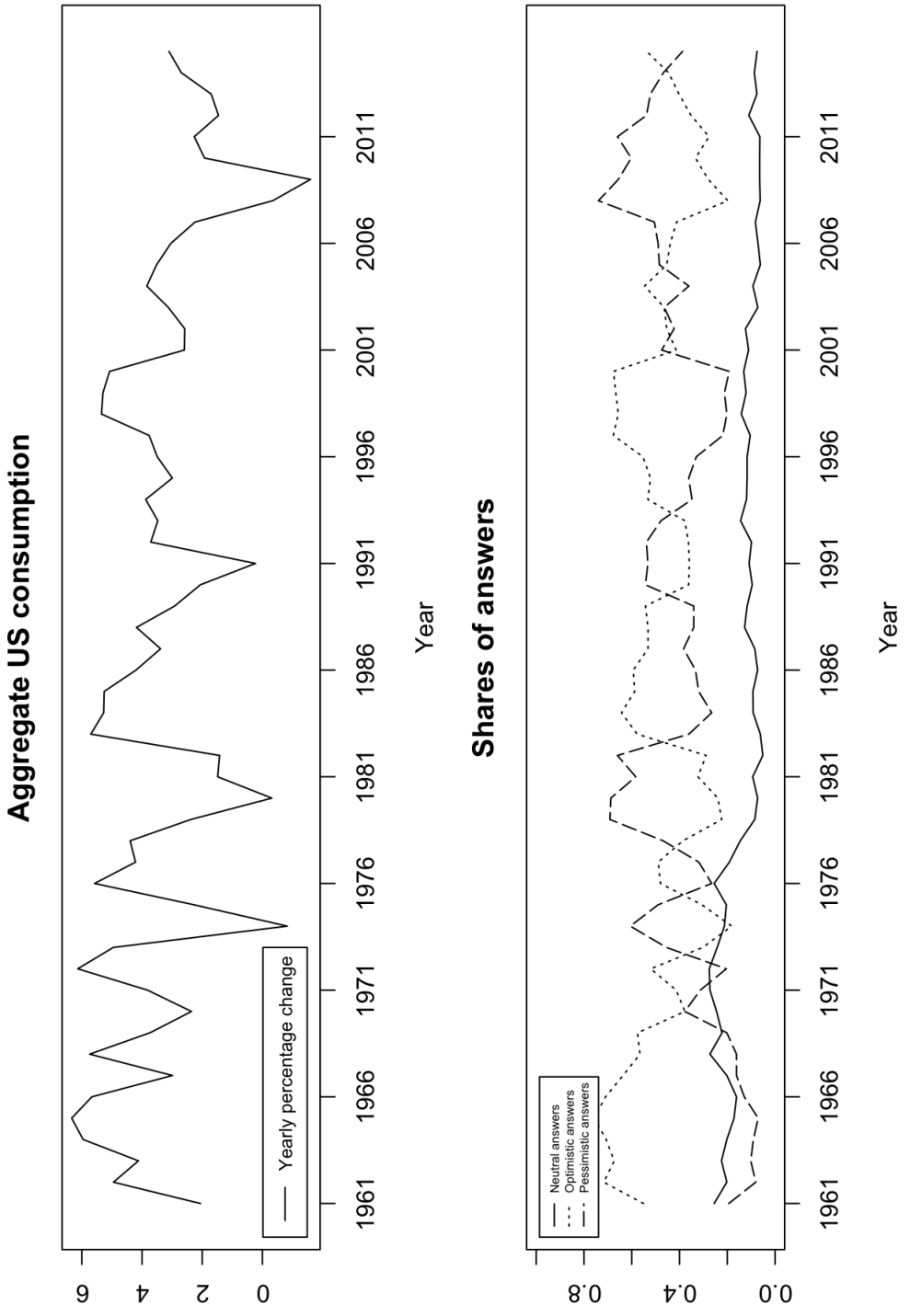
Figure 3.1 motivates this essay and emphasises the heterogeneity present in consumer sentiments. Rather than one aggregate index of sentiments, I depict the distinct shares of optimistic, pessimistic, and neutral answers to one question from the Michigan Index of Consumer sentiment and match them with percentage changes of aggregate US consumption. The movement in shares fits the conjectured mechanism outlined before well. The correlation between the optimistic share and the changes in aggregate consumption is 0.765, the correlation between the pessimistic share and changes in aggregate consumption is -0.783. Replacing aggregate consumption with aggregate output yields a very similar picture (provided in Appendix 3.5.1). Co-movements, however, are more pronounced here because the figure depicts yearly values.[1] Yet, arguably more interesting is the significant heterogeneity in sentiments shares. Also in dire economic times, a considerable share of positive sentiments seems to remain and the reverse. The model introduced in this essay in essence tries to develop a mechanism behind such shares.

I try to stick closely to Keynes's original notion of animal spirits as psychological phenomena which make people deviate from probabilistic decision making. In the model used here, a sentiment induced bias leads to temporarily upward or downward distorted consumption choices. Resulting changes in aggregate demand also change aggregate output as not all producers can adjust their price instantaneously. If aggregate output is currently on an upward trajectory, optimism is shared with relatively higher probability, if aggregate output is on a downward trajectory, pessimism is shared with relatively higher probability. This creates a rich feedback mechanisms between the sharing of sentiments and the resulting macroeconomic fluctuations. Consumer sentiment impacts aggregate demand and aggregate demand then changes aggregate output. Yet, in reverse aggregate output also impacts the sharing of consumer sentiments: Changes in the aggregate output trajectory determine which sentiment is shared more easily.

With its features of epidemiological sentiment sharing, the framework is connected to recent work by Shiller (2017) on the spreading of narratives. Contrary to other models of consumer sentiments, such as e.g. Angeletos and La'O (2013), its mechanism is simpler without the need for concepts such as higher order beliefs or sunspot equilibria. Yet, it also requires considerably stronger assumptions.

[1] These correlations are lower if quarterly values for both the sentiment questions and consumption/output are used. To be precise, note that the model in this essay is calibrated at the quarterly level.

Figure 3.1: Yearly movements in aggregate consumption and the shares of optimistic, pessimistic, and neutral answers



The shares represent answers to the questions "Now turning to business conditions in the country as a whole - do you think that during the next 12 months we'll have good times financially or bad times or what?". I counted people who replied "Good times" as optimistic, people who replied "Bad times" as pessimistic, and people who answered "Don't know" or "Uncertain" as having a neutral sentiment. The US aggregate consumption and GDP changes time series are from St. Louis FRED. Data depicted runs from 1961 to 2015.

Agents are forward looking, however, are assumed not to form expectations or correctly optimise over one key variable: the sentiment bias of consumers. Unlike research such as e.g. Benhabib et al. (2015) the model therefore does not satisfy rational expectations. Yet, the agents depicted here also do not use adaptive rules as e.g. in De Grauwe and Ji (2019). As individual sentiment states and actions depend on the infection levels of neighbours, a related strand of literature is also that of threshold models, see e.g. Granovetter (1978). Lastly, given that perpetual cyclical motion in output is a fundamental property of the economy developed here, it also relates to recent research such as e.g. Beaudry et al. (2019) who discuss an endogenous propagation mechanism based on other features (strategic complementary and financial frictions), or to other models of endogenous fluctuations such as e.g Aymanns et al. (2016).

The paper is structured as follows: Section 3.2 first introduces the broad framework of economic sentiment sharining. It then continues with providing an example of this framework by applying it to a boundedly rational version of the New Keynesian Model. Section 3.3 discusses simulation outcomes and Section 3.4 concludes.

3.2 Model

3.2.1 A framework of population sentiment dynamics in a macroeconomy

In this paper, an Erdős–Rényi random graph of the form $G(n, p)$ is used to model how economic sentiments move through a society. The parameter n describes the size of the population, i.e. the number of nodes. A link between any two nodes is formed independently with probability $0 < p < 1$ in this class of networks. The expected degree, i.e. the mean number of connections that each node has, has the convenient form $d = np$ (see e.g. Jackson, 2008, for discussion of these models). Once formed, all links between individuals are characterised in the so called *adjacency matrix* $\mathbf{A}$, which is of dimension $n \times n$. An element a_{ij} characterises the link of individual i with individual j. Links in the $G(n, p)$ network are *unweighted*, i.e. a link can only exist or not ($a_{ij} \in \{0, 1\}$) and they are *undirected*, i.e. $a_{ij} = a_{ji} \forall i, j$ (the adjacency matrix $\mathbf{A}$ is symmetric). To

illustrate this with a simple example, consider:

$$\underset{n\times n}{\mathbf{A}} = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

This adjacency matrix would represent a network with $n = 3$ individuals, in which links exist between individuals 1 and 2, and between individuals 1 and 3.

All consumers living in this world have one of three possible states of mind regarding the economy: They are *optimistic*, *pessimistic*, or *neutral*. Somewhat similarly to diseases in a simple model of disease transmission they can share their sentiments with their neighbours in the network. Continuing the example above, assume that consumer one is neutral, consumers two and three are optimistic, and none of them is pessimistic. For a given point in time, I store sentiment states in a second matrix:

$$\underset{n\times 2}{\mathbf{E_t}} = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 1 & 0 \end{pmatrix}$$

The first column refers to optimism, the second column to pessimism. Thus $\mathbf{E_t} \in \{0,1\}^{n\times 2}$ with a value of 1 meaning that an individual is infected with the column's associated sentiment. At any point in time t it holds that $\nexists i$ for which $e_{i,1,t} = e_{i,2,t} = 1$. In other words, no individual can have both sentiments simultaneously. Cases in which an individual has coincidentally been infected by both optimism and pessimism in one period are decided by a coin flip.

Optimistic individuals consume more, pessimistic individuals less than neutral individuals. Depending on which of the two sentiments dominates a population, *aggregate* demand is above or below its hypothetical neutral level (in which all individuals were to have a neutral sentiment). In combination with frictions such as price rigidities, these sentiment driven changes in aggregate demand move aggregate output y_t (temporarily) above or below the hypothetical fundamental output level $y_t^{neutral}$. A second possible sources of fluctuations are (very small) changes in aggregate output due to real disturbances $a_t = \rho_a a_{t-t} + \varepsilon_t, \varepsilon_t \overset{i.i.d.}{\sim} N(0, \sigma_\varepsilon^2)$. Sentiment sharing in itself creates an underlying perpetual motion in the economy. Yet, the feedback mechanism between sentiment cycles and other disturbances makes output shapes more realistic.

The sentiment state of individuals can change through two distinct ways:

1. **Sentiment sharing:** Each period, optimistic or pessimistic individuals share their sentiment with the nodes directly linked to them with a certain probability.

2. **Healing:** At the beginning of each period, sentiments of infected individuals "heal" back to neutral with a probability h.

In the following, the first point is explained in more detail. Sentiments are shared with certain probabilities, and they are shared more likely if they are in line with the current *aggregate* economic development: If aggregate output is rising, then optimism is the *supporting sentiment* and shared with a higher probability than pessimism. If output is declining, then pessimism is the *supporting sentiment* and transmitted relatively more easily. For this, first define an *aggregate momentum* variable $momentum_t \equiv y_{t-1} - y_{t-2}$. If $momentum_t \geq 0$, optimism is the supporting sentiment and pessimism is the opposing sentiment, and reverse. Each optimistic or pessimistic individual can share its sentiment once per period with its immediate neighbours (i.e. nodes directly linked to them). A supporting sentiment is shared with probability $q_{supporting}$ and an opposing sentiment is shared with probability $q_{opposing}$. It seems natural that $q_{supporting} > q_{opposing}$, i.e. that the sentiment which fits the most recent development in output well (optimism if output has been rising and pessimism if output has been falling) is shared relatively more easily.

Next define a third matrix $\mathbf{B_t}$ which stores the amount of neighbours of each node which are currently infected with one the two sentiments: $\mathbf{B_t} \in \mathbb{N}_0^{n \times 2}$ where $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$. This matrix can be computed as:

$$\underset{n \times 2}{\mathbf{B_t}} = \mathbf{A} \cdot \mathbf{E_t} \tag{3.1}$$

Each row refers to an individual. The first column refers to optimistic neighbours that this individual has, the second to pessimistic neighbours. A cell in the matrix is denoted by $b_{i,j,t}$. In the example used here, $\mathbf{B_t}$ would have the values:

$$\underset{n \times 2}{\mathbf{B_t}} = \begin{pmatrix} 2 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

Individual one has two neighbours who are optimistic: $b_{1,1,t} = 2$. No other individual has any optimistic or pessimistic neighbours. With this information about neighbours, it is now possible to compute the probability for each individual to be infected with a sentiment in a given period. Assume that in this example the economy is in a boom phase of a cycle, i.e. optimism is the supporting sentiment and pessimism is the opposing sentiment. The probabilities for individual i to be infected with optimism/pessimism in period t are then:

$$Pr(infection_{i,t} = optimism) = 1 - (1 - q_{supporting})^{b_{i,1,t}} \tag{3.2}$$

$$Pr(infection_{i,t} = pessimism) = 1 - (1 - q_{opposing})^{b_{i,2,t}} \tag{3.3}$$

In the simple example above, this would imply that individual 1 is infected with optimism with probability $1 - (1 - q_{supporting})^2$.

This concludes the summary of how sentiments move through society. The persistence of sentiment infections is balanced by healing.

Exemplary period Putting these pieces together allows to describe the timing within one period. This also provides a good summary of the interplay of forces.

1. All infected nodes from period $t - 1$ heal with probability h.
2. Momentum is computed with the two most recent output levels $momentum_t = y_{t-1} - y_{t-2}$. Say output was on an upward trend, then $momentum_t > 0$ and optimism is the supporting sentiment which transmits more easily than the opposing sentiment pessimism, $q_{supporting} > q_{opposing}$.
3. Next, sentiment sharing determines the sentiment bias currently present in the economy. All infected nodes share their sentiments with neighbouring nodes. Each node is infected with optimism with probability $Pr(infection_{i,t} = optimism) = 1 - (1 - q_{supporting})^{b_{i,1,t}}$ and with pessimism with probability $Pr(infection_{i,t} = pessimism) = 1 - (1 - q_{opposing})^{b_{i,2,t}}$. If a node has been infected with both sentiments in the same period, a coin is flipped to decide between them.
4. Ceteris paribus, optimistic individuals consume more than neutral individuals who consume more than pessimistic individuals. Depending on which sentiment currently gained traction relative to the last period, aggregate demand rises or falls.

5. Output y_t rises or falls, e.g. through a New Keynesian type of demand effect (abstracting from additional real disturbances in this example). This concludes the current period and the next one begins.

Why do cycles turn? It might not be apparent from the exemplary period, how aggregate output turns again. Say that we are currently in an upward movement and momentum is positive. The majority of individuals is optimistic, has an upward biased consumption, and pushes up output through their demand effect. Yet, at some point the supporting sentiment (currently optimism) will approach its maximum infection potential in the population. The size of that level is shaped by the relative sizes of transmission probabilities $q_{supporting}$ and $q_{opposing}$, by the healing probability h, and other specifications of the model. Given an exemplary parametrisation, say the maximum potential for a sentiment is to infect 80% of nodes. Once around this maximum, each period a fraction of consumers h heals, both sentiments are shared, and the model ends up at almost exactly 80% of optimistic consumers again. This will go on for a while, but at some point, by chance a few less people will become infected and the share of optimistic people might now be say 79.9%. This drops aggregate demand slightly, and therefore makes aggregate output decline. As a consequence, however, next period's momentum $momentum_{t+1} = y_t - y_{t-1}$ will now be negative, and hence sharing probabilities are switched between the sentiments: Pessimism becomes the supporting sentiment and optimism becomes the opposing sentiment. Thus, pessimism is now shared more easily and the cycle starts to move downwards and only stops again when pessimism reaches its maximum infection potential. Again, at some point a few more people than before became optimists by chance after sentiment sharing, demand rises a little, output rises a little, momentum and transmission probabilities revert, and the cycles turns upwards again. Note that to make turns less predictable, one could e.g. easily switch momentum (and thereby sharing probabilities) only with a certain probability once output changes. Yet, once a certain waves gains traction, it is very hard to stop. This mechanism already gives perpetual motion in output without additional exogenous real variations in output, however, cycles are too regular to be realistic. The cyclical shape becomes much more realistic and irregular if the mechanism is included in a world with small real disturbances of other sources, e.g. financial, productivity, etc. These disturbances can now unexpectedly switch momentum and thereby make cycles revert.

The next section discusses a stylised example of this mechanism in a modified New Keynesian model.

3.2.2 Perpetual motion in a behavioural New Keynesian model

Consider the canonical New Keynesian model presented in Galí (2015) Chapters 2 and 3. The households solves the problem:

$$\underset{\{C_t,N_t,B_t\}_{t=0}^{\infty}}{max} E_0\left[\sum_{t=0}^{\infty}\beta^t\left(\frac{C_t^{1-\sigma}}{1-\sigma}-\frac{N_t^{1+\varphi}}{1+\varphi}\right)Z_t\right] \tag{3.4}$$

$$s.t. \quad P_tC_t + Q_tB_t \leq B_{t-1} + W_tN_t + D_t \tag{3.5}$$

$$B_{-1} \text{ given} \tag{3.6}$$

where C_t denotes consumption, Z_t a preference shifter, N_t hours worked, P_t the price level, Q_t the price per unit bonds, B_t bond holdings, W_t wages, and D_t dividends from firm ownership. The Euler euqation for this problem is given by:

$$\frac{Z_tC_t^{-\sigma}}{P_t}Q_t = \beta E_t\left[\frac{Z_{t+1}C_{t+1}^{-\sigma}}{P_{t+1}}\right]\forall t \tag{3.7}$$

Next define $i_t \equiv -log(Q_t)$, $\rho \equiv -log(\beta)$, $z_t = log(Z_t)$, and $\pi_{t+1} \equiv p_{t+1} - p_t$. Log linearisation around the steady state yields the log linearised Euler equation (Galí, 2015, page 44):

$$c_t = E_t\left[c_{t+1}\right] - \frac{1}{\sigma}\left(i_t - E_t\left[\pi_{t+1}\right] - \rho\right) + \frac{1}{\sigma}(z_t - E_t[z_t]) \tag{3.8}$$

Galí (2015) continues with assuming this preference shifter follows an exogenous AR1 process. In contrast, I assume that it originates from endogenously evolving consumer sentiments. If optimism dominates the population, the aggregate consumer has $z_t > 0$. If pessimism dominates the population, it leads to $z_t < 0$. A key assumption that I make is that the boundeldy rational agents in this economy are unaware of the consumer's sentiment bias and form no expectations over it. This assumption yields $E_t[z_{t+1}] = 0$. In essence, the term becomes an endogenous recurrent, however repeatedly unexpected, preference shock. The consumer's and firm's inability to notice the sentiment bias and form expectations over it makes their sentiment influenced actions sub-optimal.

$$c_t = E_t\left[c_{t+1}\right] - \frac{1}{\sigma}\left(i_t - E_t\left[\pi_{t+1}\right] - \rho\right) + \frac{1}{\sigma} z_t \tag{3.9}$$

Otherwise following Galí (2015) Chapter 3, this yields the following dynamic IS equation after market clearing $c_t = y_t$:

$$\tilde{y}_t = E_t[\tilde{y}_y + 1] - \frac{1}{\sigma}(i_t - E_t[\pi_{t+1}] - r_t^n) \tag{3.10}$$

with the natural rate $r_t^n = \rho - \sigma(1-\rho_a)\psi_{ya} a_t + z_t$ and $\psi_{ya} = \frac{1+\varphi}{\sigma(1-\alpha)+\varphi+\alpha}$ where $1-\alpha$ is the exponent of labour in the production function.[2] Given the assumptions made, the New Keynesian Phillips curve stays the same as in the textbook.[3] The model is completed with an interest rate rule (abstracting from additional monetary policy shocks):

$$\pi_t = \delta E_t\left[\pi_{t+1}\right] + \kappa \tilde{y}_t \tag{3.11}$$

$$i_t = \rho + \phi_\pi \pi_t + \phi_y (y_t - y) \tag{3.12}$$

Here $\kappa \equiv \lambda(\sigma + \frac{\varphi+\alpha}{1-\alpha})$ is computed with $\lambda \equiv \frac{(1-\theta)(1-\beta\theta)}{\theta}\Theta$ and $\Theta \equiv \frac{1-\alpha}{1-\alpha+\alpha\epsilon}$, where θ denotes price stickiness and ϵ the demand elasticity. These three equations can be condensed into a systems of two equations

$$\begin{pmatrix} \tilde{y}_t \\ \pi_t \end{pmatrix} = \mathbf{A_T} \begin{pmatrix} E_t[\tilde{y}_{t+1}] \\ E_t[\pi_{t+1}] \end{pmatrix} + \mathbf{B_T} u_t \tag{3.13}$$

where

$$\mathbf{A_T} = \frac{1}{\sigma + \phi_y + \kappa\phi_\pi} \begin{pmatrix} \sigma & 1 - \delta\phi_\pi \\ \sigma\kappa & \kappa + \delta(\sigma + \phi_y) \end{pmatrix} \text{ and } \mathbf{B_T} = \frac{1}{\sigma + \phi_y + \kappa\phi_\pi} \begin{pmatrix} 1 \\ \kappa \end{pmatrix}$$

and

$$u_t = -\psi_{ya}(\phi_y + \sigma(1-\rho_a))a_t + z_t$$

For full explanations of the original model's equations see Galí (2015) Chapters 2

[2] In the book this equation is $r_t^n = \rho - \sigma(1-\rho_a)\psi_{ya} a_t + (1-\rho_z) z_t$ as the AR1 implies an expectation $E_t[z_{t+1}] = \rho_z z_t$

[3] Firms are also assumed not to build expectations over the bias factor in their optimisation and discounting.

and 3 which are the basis of this section[4], or the very helpful Bergholt (2012) for more details. As I assume here that agents are behavioural and unaware of the introduced sentiment bias, technology a_t remains the only state variable on which they optimise. The system is solved with the method of undetermined coefficients where the stationary solution is conjectured to have the form $\tilde{y}_t = \psi_y u_t$ and $\pi_t = \psi_\pi u_t$ with $E_t[\tilde{y}_{t+1}] = \rho_a \psi_y u_t$ and $E_t[\pi_{t+1}] = \rho_a \psi_\pi u_t$. Note that these expectations would only be correct if u_t contained only the term with a_t, but instead reflect that agents form wrong expectations with the behavioural modifications made in this paper, because the compound state u_t in fact also contains the endogenous sentiment bias z_t and therefore does not follow an AR1 process. With this, rewrite the system as:

$$\begin{pmatrix} \tilde{y}_t \\ \pi_t \end{pmatrix} = \begin{pmatrix} \psi_y \\ \psi_\pi \end{pmatrix} u_t = \mathbf{A_T} \begin{pmatrix} E_t[\tilde{y}_{t+1}] \\ E_t[\pi_{t+1}] \end{pmatrix} + \mathbf{B_T} u_t = \mathbf{A_T} \begin{pmatrix} \psi_y \\ \psi_\pi \end{pmatrix} \rho_a u_t + \mathbf{B_T} u_t \tag{3.14}$$

and solve for the coefficients:

$$\begin{pmatrix} \psi_y \\ \psi_\pi \end{pmatrix} = [\mathbf{I} - \rho_a \mathbf{A_T}]^{-1} \mathbf{B_T} \tag{3.15}$$

Which yields:

$$\begin{pmatrix} \tilde{y}_t \\ \pi_t \end{pmatrix} = \begin{pmatrix} \psi_y \\ \psi_\pi \end{pmatrix} u_t \tag{3.16}$$

In the simulations in Section 3.3, I depict output rather than output gab and therefore add natural (flexible price) output back to output gap.[5] The key equation is therefore given by:

$$\begin{pmatrix} y_t \\ \pi_t \end{pmatrix} = \begin{pmatrix} \tilde{y}_t \\ \pi_t \end{pmatrix} + \begin{pmatrix} y_t^n \\ 0 \end{pmatrix} = \begin{pmatrix} \psi_y \\ \psi_\pi \end{pmatrix} (-\psi_{ya}(\phi_y + \sigma(1-\rho_a))a_t + z_t) + \begin{pmatrix} \psi_{ya} a_t \\ 0 \end{pmatrix} \tag{3.17}$$

[4] See particularly Galí (2015), pages 65-66 and 71-74, for the following derivations.

[5] This drops the constant depicted in Equation 20 on page 62 of Galí (2015): $y_t^n = \psi_{ya} a_t - \frac{(1-\alpha)(\mu - log(1-\alpha))}{\sigma(1-\alpha)+\varphi+\alpha}$ and sets natural output to only $y_t^n = \psi_{ya} a_t$. It centers the later plots around zero and furthermore tries to replicate the second equation on page 72 of Galí (2015) where actual output is also zero in absence of technology shocks.

As a reference point, the neutral sentiment output and inflation without the behavioural bias is given by:

$$\begin{pmatrix} y_t \\ \pi_t \end{pmatrix} = \begin{pmatrix} \tilde{y}_t \\ \pi_t \end{pmatrix} + \begin{pmatrix} y_t^n \\ 0 \end{pmatrix} = \begin{pmatrix} \psi_y \\ \psi_\pi \end{pmatrix} (-\psi_{ya}(\phi_y + \sigma(1-\rho_a))a_t) + \begin{pmatrix} \psi_{ya} a_t \\ 0 \end{pmatrix} \tag{3.18}$$

Dynamics of $\mathbf{z_t}$ A key question remains how the sentiment bias of the representative consumer is computed. The framework introduced in 3.2.1 naturally fits heterogenous agent models, but in this application I use the network as a computational tool to approximate the nontrivial time series of an average bias factor z_t. Say that depending on their infection state, each node on the network has one of three bias terms: $z^{optimistic}$, $z^{pessimistic}$, and $z^{neutral} = 0$ with $z^{optimistic} > 0 > z^{pessimistic}$. At each point in time, the share of each of the three groups is given by the number of nodes with a respective state divided by the total number of nodes in the network: $\alpha_t^{optimistic} = \frac{n_t^{optimistic}}{n^{total}}$, $\alpha_t^{pessimistic} = \frac{n_t^{pessimistic}}{n^{total}}$, and $\alpha_t^{neutral} = 1 - \frac{n_t^{optimistic} + n_t^{pessimistic}}{n^{total}}$. The sentiment bias of the aggregate households is assumed to be approximated by $z_t = \alpha_t^{optimistic} \times z^{optimistic} + \alpha_t^{neutral} \times 0 + \alpha_t^{pessimistic} \times z^{pessimistic}$. In other words, the network works as a numerical tool to compute the latent non-linear two functions $\alpha_t^{optimistic}(\cdot)$ and $\alpha_t^{pessimistic}(\cdot)$.[6]

Exemplary period To illustrate the feedback mechanisms between network and New Keynesian model as well as the intra period timing, consider again an exemplary period.

1. All infected nodes from period $t-1$ heal with probability h.
2. Momentum is computed with the two most recent output levels $momentum_t = y_{t-1} - y_{t-2}$. Say output was now on a downward trend, then $momentum_t < 0$ and pessimism is the supporting sentiment which transmits more easily than the opposing sentiment optimising, $q_{supporting} > q_{opposing}$.
3. Next, sentiment sharing determines the sentiment bias currently present in the economy. All infected nodes share their sentiments with neighbouring nodes. Each node is infected with pessimism with probability

[6] Another option could e.g. be a modified and discretised compartmental SEIS model. Yet, to be similar it would have to be built containing two mutually exclusive diseases and such that also infected shares are susceptible to the other disease.

$Pr(infection_{i,t} = pessimism) = 1 - (1 - q_{supporting})^{b_{i,2,t}}$ and with optimism with probability $Pr(infection_{i,t} = optimism) = 1 - (1 - q_{opposing})^{b_{i,1,t}}$. If a node has been infected with both sentiments in the same period, a coin is flipped to decide between them.

4. Given the network's state after sentiment sharing, the aggregate bias is approximated with $z_t = \alpha_t^{optimistic} \times z^{optimistic} + \alpha_t^{neutral} \times 0 + \alpha_t^{pessimistic} \times z^{pessimistic}$.

5. With this z_t and an additional exogenous real shock a_t (if any), compute actual output y_t and inflation π_t with Equation 3.17. A hypothetical neutral sentiment output can be computed as a benchmark with Equation 3.18 and only depends on a_t. Proceed with the next period.

Assumptions made To achieve the simple form introduced above, I make several assumptions in addition to those of the standard model. First, I need to assume that the average bias factor of nodes is a good approximation of the representative agent's aggregate bias. Questions of aggregation are non-trivial and this omits several effects present in the complex system of interactions in a heterogeneous agent model and can lead to considerable approximation error. Second, I assume that consumers make mistakes relatively to fully dynamically consistent decisions and that these mistakes depend on their sentiment state of mind. Third, no agent in the economy forms expectations over these sentiment biases or considers z_t correctly in their optimisation. This assumption is partially motivated by research such as Tversky and Kahneman (1974) and the large literature that studies biases of which individuals are often unaware. Here, however, I require a stronger assumption that agents do not form expectations or optimise over sentiment preference shifts despite their recurring nature.

Section 3.3 proceeds with simulations of the model.

3.3 Simulations

For the full parametrisation of the model used in the simulations, see Appendix 3.5.3. Figure 3.2 shows the perpetual motion created by the mechanism without exogenous shocks. As the model depicts a world in which sentiment states are always present when there is output, the only requirement is to set an initial allocation of sentiments. For this, I exemplarily set approximately 30% of nodes to

be optimistic and approximately 5% of nodes to be pessimistic.[7] As the change in output was constant, initial momentum is assumed to be positive. Motion would also start, however, if momentum was set to negative and a relatively larger share of pessimistic individuals was created. Due to the high relative infectiousness of the supporting sentiment and the high degree on the network, the model picks up movements for a wide range of initial allocations.

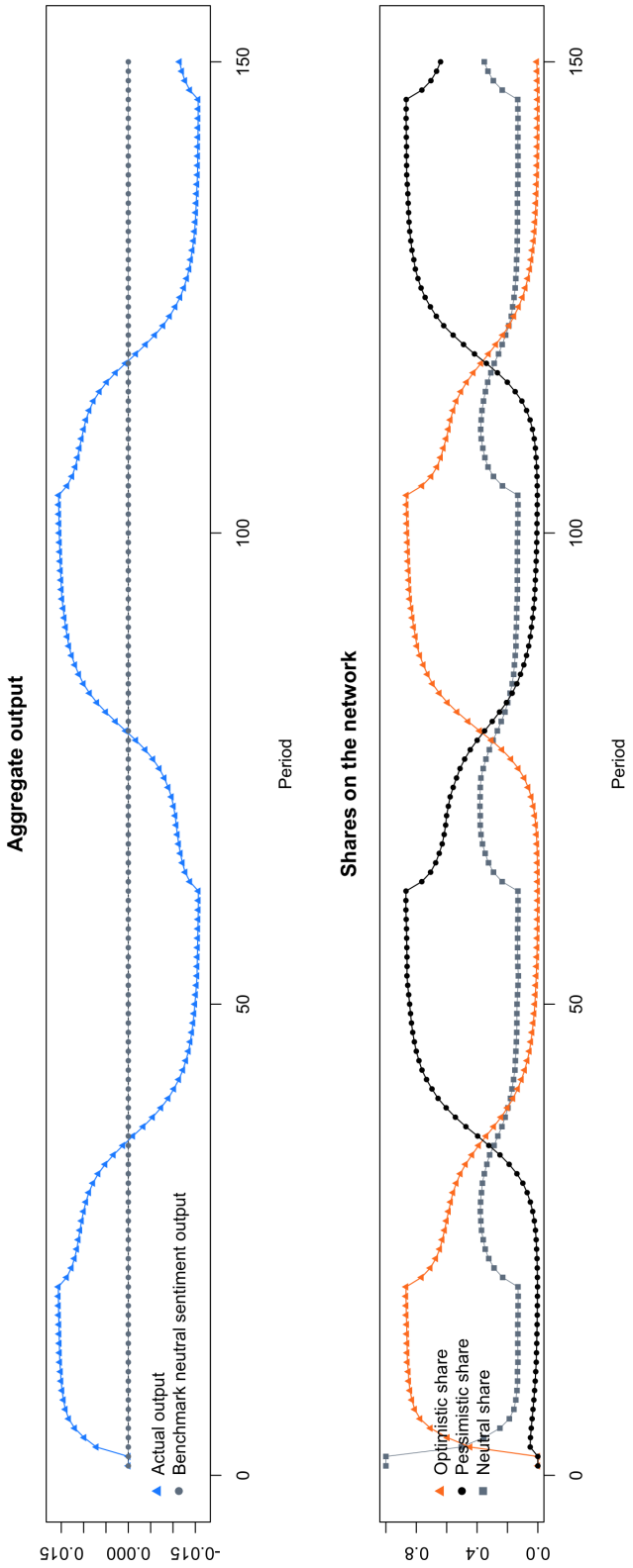
Due to the lack of exogenous shocks, the neutral sentiment benchmark output (in grey) is flat. Note also that despite its strong regularity, the motion depicted in Figure 3.2 is not deterministic. The reason is that sentiment sharing on the network is probabilistic. Yet, due to the large size of the network, infection dynamics in each cycles are highly similar. Whereas the underlying mechanism of sentiment sharing creates an economy which is always in motion, it therefore by itself cannot generate the irregular shapes we see in data.

Figure 3.2 shows, however, that it only requires relatively small real disturbances (without own persistence) to make output look more realistic. The reason is that already small shocks can revert momentum if the current trajectory in output is slowing down. This reversion then switches which sentiment is more infectious and subsequently leads to sentiment dynamics which make the cycle turn. In this example, the disturbances depict technology shocks, but they could also be other real disturbances such as supply shocks, financial shocks etc. The sharing of consumer sentiments is just a layer on top of other mechanisms, but leads to the cyclical shape of output. For the dynamics of the aggregate bias factor z_t and inflation π_t see Figures 3.5 and 3.6 in the Appendix.

Discussion Both output and sentiments influence each other in a feedback mechanism. Sentiments induce demand changes which then change output. Changes in output, however, set momentum and thereby influence which of the two sentiment is more infectious. Exogenous real disturbances make these reversions in momentum and sentiment cycles irregular in contrast to the stylised example with no exogenous shocks. The model's inherent motion thereby relates to the literature of nonlinear dynamical systems where state variables can perpetually move on sets like limit cycles or attractors rather than sitting at a fixed point or steady state as default. This notion can be helpful to structure thinking about economies in motion and is used in work such as e.g. Beaudry et al. (2019) or Aymanns et al. (2016).

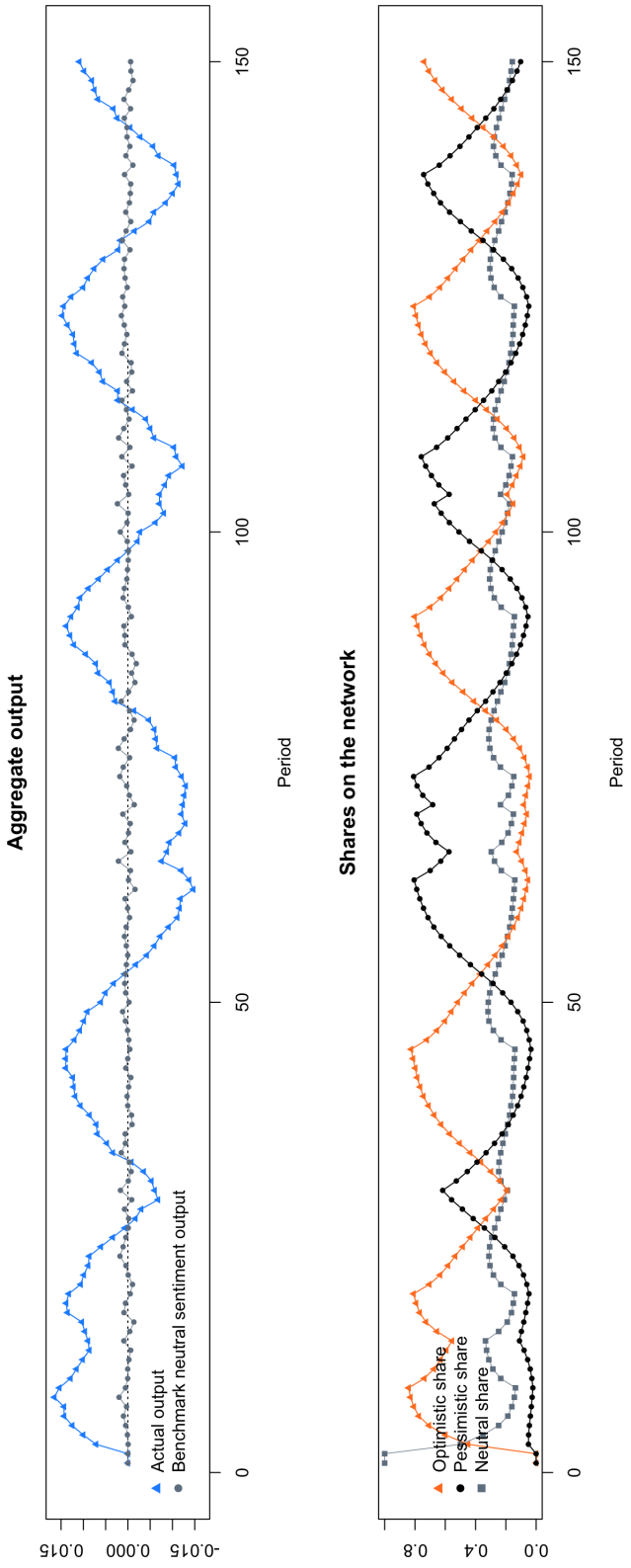
[7] In detail, each node is set to optimistic with probability 0.3 and to pessimistic with probability 0.05 in the initialisation. Note that there can thereby be double infections in the initialisation period.

Figure 3.2: Output with sentiment initialisation only



The first model output is computed in period 3. In period 2, a random share of approximately 30% of nodes is set to optimistic and a random share of approximately 5% to pessimistic. These sentiment states are carried over to period 3, partially "healed" with probability h at the beginning of the new period, and then begin to transmit.

Figure 3.3: Output with sentiment initialisation and real disturbances



While this paper develops a general mechanism of an always present movement in an economy through sentiment dynamics, it would be interesting in further research to relax some of its partly strong assumptions. First, the problems of aggregation could be addressed by embedding the mechanism into a heterogenous agent model. Each node on a network could have an own sentiment bias, optimisation problem, income, momentum, etc. Second, it would be very interesting to develop a formulation of bounded rationality with which agents are at least partially aware of the sentiment bias and form expectations over it. In both simulations here, agents assume no persistence through the parameter $\rho_a = 0$, although the sentiment term z_t is very persistent. A key challenge for both extensions, heterogeneity and rigorous bounded rationality, would be to keep track of the states which would become very large as soon as agents took (parts of) the rich network structure into account.

The following Section 3.4 concludes the paper.

3.4 Conclusion

This paper describes a mechanism of sentiment dynamics which generates a perpetual cyclical pattern in output. Two opposing sentiments, optimism and pessimism, are shared between individuals on a network similarly to diseases in a stylised model of disease transmission. A behavioural bias shifts preferences of boundedly rational consumers who deviate from original consumption levels. Optimistic individuals consume too much and pessimistic consumers too little relative to optimal levels. Depending on which sentiment currently dominates in the economy, aggregate demand is upward or downward biased. I incorporate this general mechanism into a New Keynesian model in which I approximate the aggregate consumer's bias by a weighted average on the network. In this model not all producers can immediately adjust their prices, so changes in aggregate demand also change short term real aggregate output. In reverse, output also influences sentiments. Depending on whether output has been rising or falling, optimism or pessimism transmit with higher probability. The interplay of these forces creates perpetual waves of optimism and pessimism and thereby perpetual fluctuations in output. Areas for future research would be to embed such a mechanism into a full heterogenous agent model and to make the expectation formation and optimisation of boundedly rational agents more sophisticated.

References

Angeletos, George-Marios and Jennifer La'O, "Sentiments," *Econometrica*, 2013, *81* (2), 739–779.

Aymanns, Christoph, Fabio Caccioli, J. Doyne Farmer, and Vincent W.C. Tan, "Taming the Basel leverage cycle," *Journal of Financial Stability*, 2016, *27*, 263 – 277.

Beaudry, Paul, Dana Galizia, and Franck Portier, "Putting the Cycle Back into Business Cycle Analysis," forthcoming: American Economic Review 2019.

Benhabib, Jess, Pengfei Wang, and Yi Wen, "Sentiments and Aggregate Demand Fluctuations," *Econometrica*, 2015, *83* (2), 549–585.

Bergholt, Drago, "The Basic New Keynesian Model," Lecture Notes 2012.

Galí, Jordi, *Monetary Policy, Inflation, and the Business Cycle: An Introduction to the New Keynesian Framework and Its Applications*, second edition ed., Princeton University Press, 2015.

Granovetter, Mark, "Threshold Models of Collective Behavior," *American Journal of Sociology*, 1978, *83* (6), 1420–1443.

Grauwe, Paul De and Yuemei Ji, "Inflation Targets and the Zero Lower Bound in a Behavioural Macroeconomic Model," *Economica*, 2019, *86* (342), 262–299.

Jackson, Matthew O., *Social and Economic Networks*, Princeton University Press, 2008.

Keynes, John M., *The General Theory of Employment, Interest and Money* 1936.

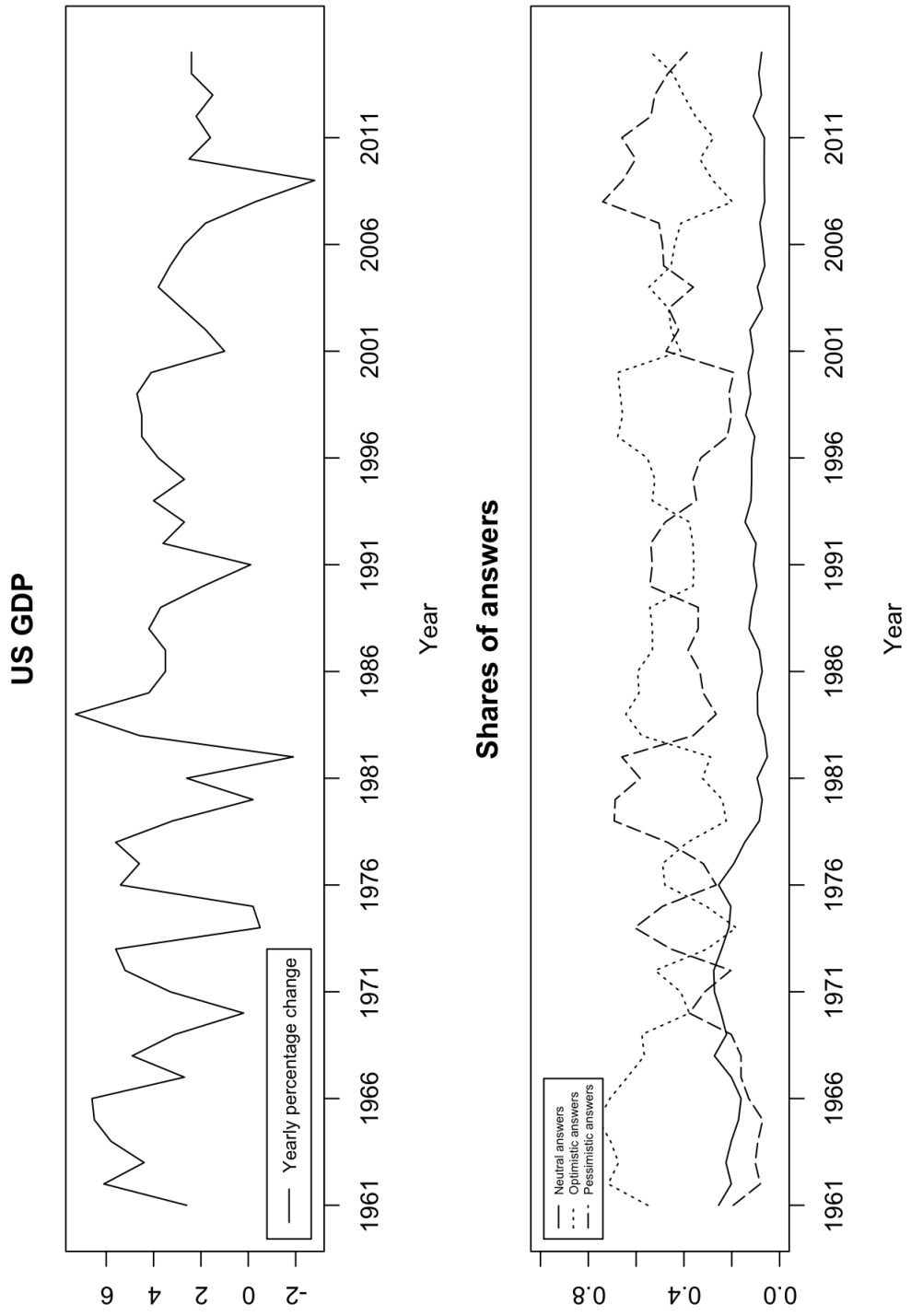
Shiller, Robert J, "Narrative Economics," Working Paper 23075, National Bureau of Economic Research January 2017.

Tversky, Amos and Daniel Kahneman, "Judgment under uncertainty: Heuristics and biases," *Science*, 1974, *185* (4157), 1124–1131.

3.5 Appendix

3.5.1 Aggregate output and sentiments

Figure 3.4: Yearly movements in aggregate output and the shares of optimistic, pessimistic, and neutral answers



3.5.2 Calibration of the New Keynesian model and network

Calibration of the New Keynesian model (following Galí (2015), pages 67-68, except for ρ_a and σ_ε):

1. $\beta = 0.99$ (geometric discount rate)
2. $\sigma = 1$ (implies log utility)
3. $\varphi = 5$ (implies a Frisch elasticity of 0.2)
4. $\alpha = 0.25$ (the exponent of labour in the labour only production function is $1 - \alpha$)
5. $\epsilon = 9$ (implying $\mathcal{M} = 1.125$, a steady state markup of 12.5%)
6. $\mu \equiv log(\mathcal{M})$ (log of desired gross markup)
7. $\eta = 4$ (semi-elasticity of money demand)
8. $\theta = 3/4$ (implying an average price duration of 4 quarters)
9. $\phi_\pi = 1.5$ and $\phi_y = 0.5/4$ (Taylor rule coefficients)
10. $\rho_a = 0$
11. In Figures 3.3 and 3.6 with exogenous shocks: $\sigma_\varepsilon = 0.005$

Calibration of the network:

1. There are $n = 100,000$ nodes/consumers in the population. This large number of nodes is chosen to smooth the random factors in the model
2. Average expected degree $np = 50$, i.e. everyone has on average 50 neighbours
3. Supporting sentiments are shared with probability $q_{supporting} = 0.04$
4. Opposing sentiments are shared with probability $q_{opposing} = 0.02$
5. sentiments heal with probability $h = 0.33$ each period
6. The bias of optimistic individuals is depicted by $z^{optimistic} = 0.025$
7. The bias of pessimistic individuals is depicted by $z^{pessimistic} = -0.025$

3.5.3 Inflation and bias

Figure 3.5: Inflation and bias with sentiment initialisation only

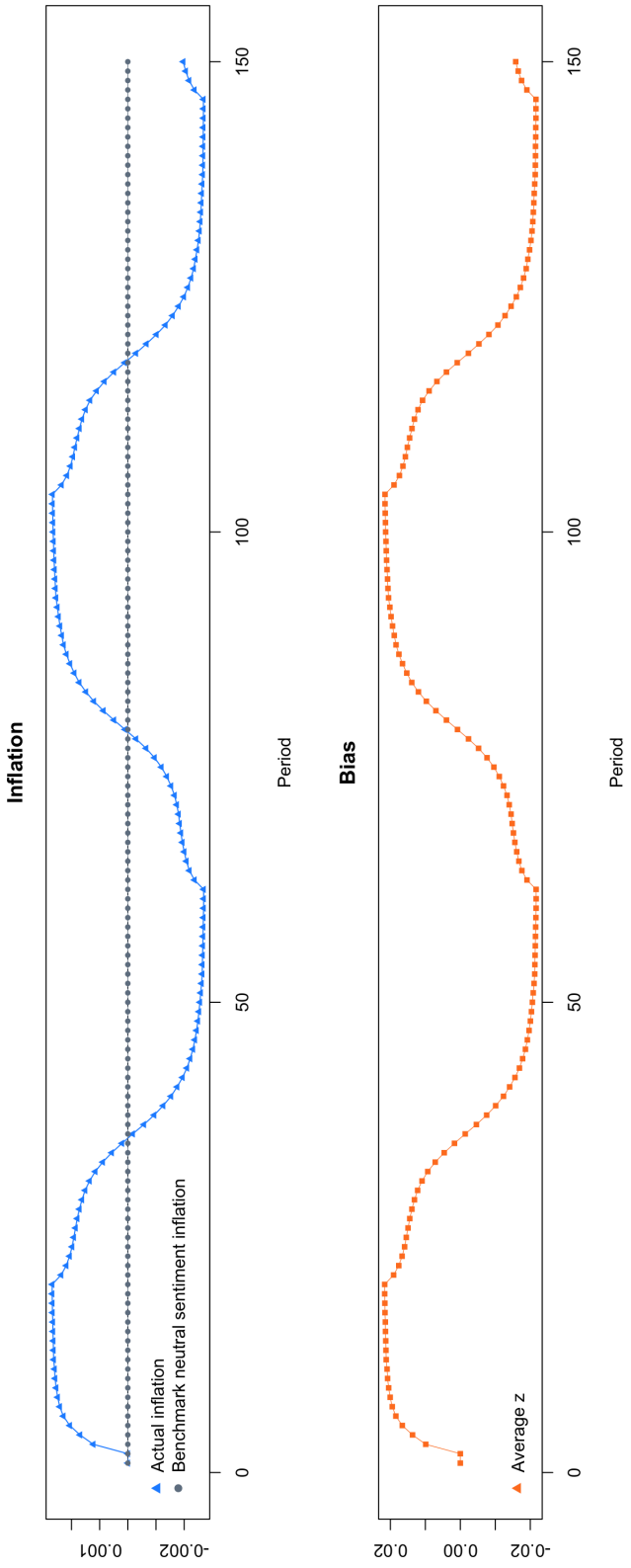
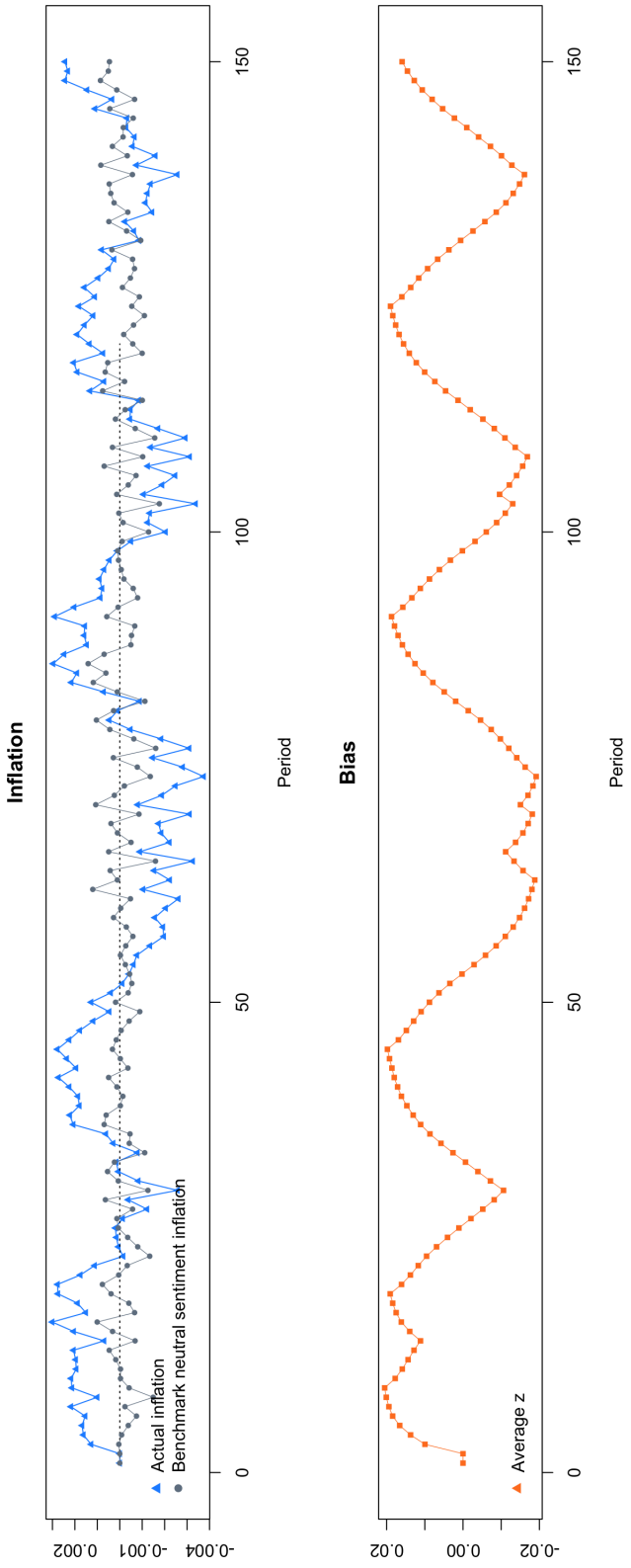


Figure 3.6: Inflation and bias with sentiment initialisation and real disturbances



Chapter 4

Textual Business Indicators

4.1 Introduction

This short paper discusses an approach to estimate business and economics sentiment indicators from historical newspaper data for times where commonly used indicators were not available. While current indicators largely rely on polling a limited set of firms, similar textual methods also allow to compute business sentiment indicators potentially in real time from large amounts of textual data available today.

Using the New York Times (NYT) archive of article lead paragraphs and snippets since 1851, I compute a long run indicator of business and economic sentiments. This is done with a two-step approach. First a classifier is trained on a subset of articles which are known to be about business topics. This classifier is then applied backwards to predict which articles prior to 1980 have been about business topics when this information is missing. In a second step, a dictionary/word list is constructed to detect negative business conditions and it is applied to the predicted business articles. The resulting index is then compared to aggregate economic variables and to existing business sentiment indicators. Lastly, articles identified as being about negative business topics are analysed for trending terms in a stylised way.

This paper relates to a strand of literature that constructs indicators based on newspaper data. In the prominent paper Baker et al. (2016), the authors use newspaper data to quantify economic policy uncertainty. Several other papers have recently used newspaper texts to detect macroeconomic and financial sentiments. For example, Püttmann (2018) builds an indicator of financial stress

ranging back to 1889. He analyses a larger dataset (based on several newspapers) than the one here also with a sentiment word list, but uses words instead of a classifier model to detect business articles and looks at headlines. Tuckett and Nyman (2017) employ a sentiment dictionary based on a theory of decision making and apply it to news data ranging back to 1996. A recent paper which uses more advanced methods to analyses a wide range of newspaper topics and their predictive ability for and comovement with economic variables is Bybee et al. (2019), however, the focus is on more recent data since the 1980s. Similarly, Larsen and Thorsrud (2019) detect narratives with topic modelling since 1990. In other recent work, Hassan et al. (2019), while discussing political risk, also detect business sentiments in transcripts of company calls using word lists by Loughran and McDonald (2011). This emphasises that the possible scope of texts for the detection of business and economic sentiments today is much wider than exclusively news.

The paper is structured as follows: Section 4.2 describes the data, Section 4.3 discusses how the index is constructed, Section 4.4 presents the main results, Section 4.5 discusses extensions with trending words within negative business articles, and Section 4.6 concludes.

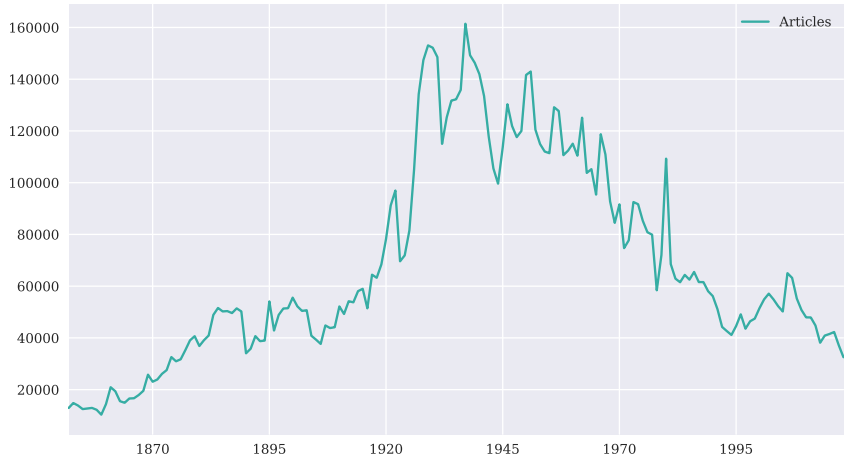
4.2 Data

The analysis of this paper is based on the archive of NYT lead paragraphs and snippets since 1851 which is publicly available via their Application Programming Interface (API).[1] This API allows to download and process large amounts of newspaper information in digitalised form. Data comes in JavaScript Object Notation (JSON) format. The public version of the API, however, only contains shortened texts at the lead paragraphs or snippet lengths. Furthermore, the data downloaded includes a large range of materials in addition to standard articles, such as e.g. birth announcements, obituaries, or in newer years multi media content. Table 4.5 in the Appendix gives an overview of available material in some of the more common categories over the decades. I use article texts only from three classes:[2] "Article", "News", and "Front Page". Out of these, the first two store very similar information in different years and the last one stores (until 1980) those articles that were on the front pages. In particular, the available

[1] `https://developer.nytimes.com/apis`; accessed 30.09.2019; I thank Hubert Mandeville from the NYT for his kind help with the access.

[2] The key in the JSON files under which these classes, among other examples shown in Table 4.5, are stored in the archive is "types_of_material".

Figure 4.1: Yearly articles after cleaning

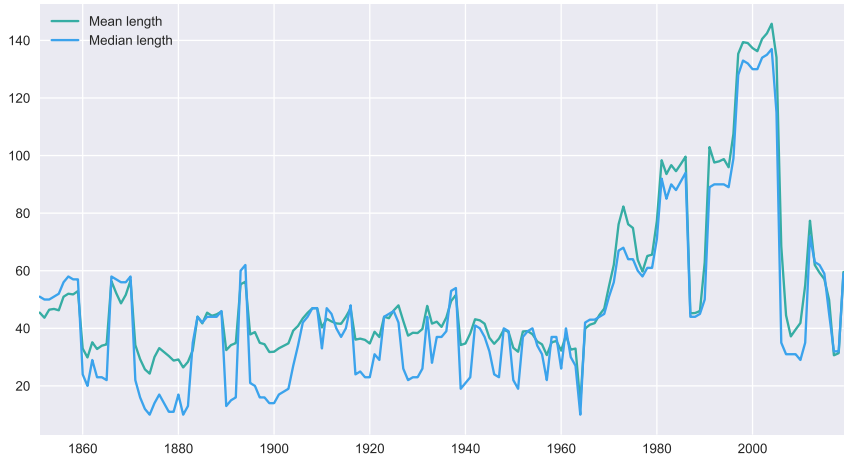


information changes substantially from 1980 onwards, when e.g. data on sections of articles becomes available. The main articles of the newspaper until 1980 are stored as "Article" (with the exception of 1964; see Figure 4.9 in the Appendix), and from 1981 onwards stored as "News". Yet, for the year 1980, there seem to exist materials for both classes "Article" and "News" which is also visible in Figure 4.9. I use the material from both these classes in 1980, however, this leads to a spike in this year e.g. visible in counts or articles visible in Table 4.6 in Appendix 4.7.3.[3] The effect on the index is probably relatively small as this spike in articles only affects one year and in the index the counts of sentiment signal words are normalised by the total amounts of words. For these and some other particularities of the archive that might be useful to other users see Appendix 4.7.2. In general it has to be kept in mind that working with such historical archive data always bears the possibility that some outcomes are driven by data structure particularities.

After cleaning relevant texts to remove punctuation, numbers, etc., I combine available information in one text. In detail, I concatenate (if available) headline, abstract, and (depending on what exists and is longer) lead paragraph or snippet. Abstract and lead paragraph/snippet can be identical resulting in duplicated words, however, the effect should be alleviated as sentiment signal words are divided by total words later. All combined texts which still have less than three words are deleted. In the following I refer to these concatenated texts in short simply as *articles*. Figure 4.1 shows the counts of cleaned articles over the years since around 1850. There is significant decline in articles, potentially driven by

[3]Judging from the time series in each of these two categories, the correct one for 1980 might be "Article" which would lead to numbers of articles in that year more consistent with other years.

Figure 4.2: Words contained in concatenated cleaned articles



the decline in printed content and by particularities in the archive. Figure 4.2 depicts mean and median length of article texts. It emphasises the importance of dividing detected sentiment words by the total amount of words rather than the total amount of articles. In later years, articles are longer and hence have a higher chance to contain keywords (in earlier years articles e.g. more often include only headlines).

Equipped with these cleaned and concatenated texts, the next section continues with how their content is used to build the sentiment index.

4.3 Building the index

The index is constructed with a two-step approach. First, I use a classifier model to estimate which historical articles are likely about business content. Using these predicted business articles for earlier years, I then approximate negative sentiments with a word list.

4.3.1 Detecting business articles

A particularity of the NYT archive is that from 1980 onwards it contains information on the News Desk or Section of an article. I label all articles whose news desk or section names contain one of the strings "financ", "econ", "business", or "money" as being about business and economics content. Using the data from 1981, this yields $355,954$ articles. Next, I randomly sample an identical amount of non business articles. In combination this leads to a dataset of $711,908$ observations.

Using these observations, I build a document term matrix (dtm) X. Such dtms have as many rows as there are documents, and as many columns as there are terms contained in the documents after some adjustments. Each cell in X is the count of a word in the given document. In the application here, I ignore all words which are not at least in two documents. This reduces the amount of columns in X substantially. The resulting dtm has has $711,908$ rows (one for each article) and $142,112$ columns (one for each remaining term). Due to the differences in available article lengths in the public API data visible in Figure 4.2, I divide all row vectors in the dtm by the total count of words in the document.

As it is known for each of these articles whether they discuss as business topic or not, this information is stored in an associated vector y (containing 1's for business articles and 0's for others). This vector is of dimension $711,908 \times 1$. X and y could now e.g. be used in a (regularised) logistic regression. Such a model maps rows of the dtm (X) into article classes (y): $X \mapsto y$. Applied to new article data, the model can then predict whether articles are likely about business topics. As the variable y is not metric but categorical, these models are referred to as *classifiers* in machine learning. Instead of a logistic regression, I use a random forest which usually perform well in classifications tasks with high dimensional data (the dtm matrix X has $142,112$ columns each of which is conceptually a variable). Like in several other parts of this thesis, this uses the Python package scikit-learn, see Pedregosa et al. (2011). For an in depth discussion of tree based methods see for example James et al. (2013) or Hastie et al. (2005).

To obtain an approximate idea of how well such a classifier works, I first split the data into training and test samples. The training dataset is a random sample of 70% of the labelled articles after 1985. I also construct two test samples: One test dataset being the remaining 30% of labelled observations from years after 1985, and the other test dataset being all labelled observations from 1981 to 1985. The second test dataset tries to get a very rough idea of whether the model becomes worse if it is applied to earlier articles than the ones on which it is trained. Tables 4.1 and 4.2 show the results of these evaluations and Table 4.7 in the Appendix depicts the same for the training data itself. The predictions on the training data depicted in Appendix 4.7 are almost 100% accurate and emphasise the importance of training and test splits to evaluate these very flexible models. Yet, the classifier generally seems to work well, also on the test data.[4] It achieves

[4]To be very precise, note hereby that the dtm was built on the full sample. While words contained in the training but not in the test set should still have zero column values, this creation of the dtm in one go might influence which infrequent words are or are not deleted. It might thereby make the predictions better than when dtms were created for the samples separately as it would be truly out of sample.

Table 4.1: Confusion matrix for articles classifier (test data from same years as training data)

	Predicted: Business and finance	Predicted: Other
True: Business and finance	82037	6196
True: Other	8046	81797

Test accuracy of approximately 0.92. $N = 178,076$ *test observations.*

Table 4.2: Confusion matrix for article classifier (early test data 1981-1985)

	Predicted: Business and finance	Predicted: Other
True: Business and finance	59215	2553
True: Other	4971	51582

Training accuracy of approximately 0.9364. $N = 118,321$ *test observations.*

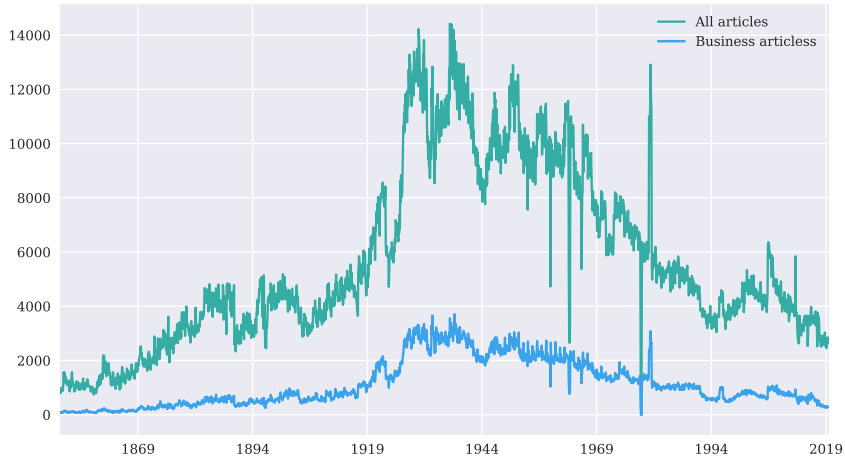
high accuracy both when applied to test data from the same years as the training data and when applied to test data from some earlier years. Of course this has to be taken with a grain of salt. Language e.g. changes over the decades and the classifier will perform less well in very early years of the sample which contain words unknown to it. In summary, although likely worse on the full corpus due to this and other particularities, the model's performance on the test data and some manual readings of its classifications for early years in the sample are generally encouraging.

After the evaluation, I then train a new random forest model now on the full 711, 908 observations. With this model I predict which articles prior to 1981 were about business and economics topics. Figure 4.3 depicts the time series of all cleaned articles and newly predicted business articles. Except for the irregularities around 1980 also mentioned in Section 4.2 and some other outliers in the cleaned dataset, the predicted business article count series progresses relatively smoothly also during decades prior to 1981 (for data from 1981 onwards, the figure depicts when available the true business articles).

4.3.2 Detecting sentiment

The second step is to detect negative sentiments in these business articles. This detection of sentiments within predicted business article classes is done with dictionary approaches. In detail, I use three different word lists of only negative sentiment keywords which generally seem more correlated with economic variables than their positive counterparts. First, I use the negative part of the senti-

Figure 4.3: Cleaned articles and business articles



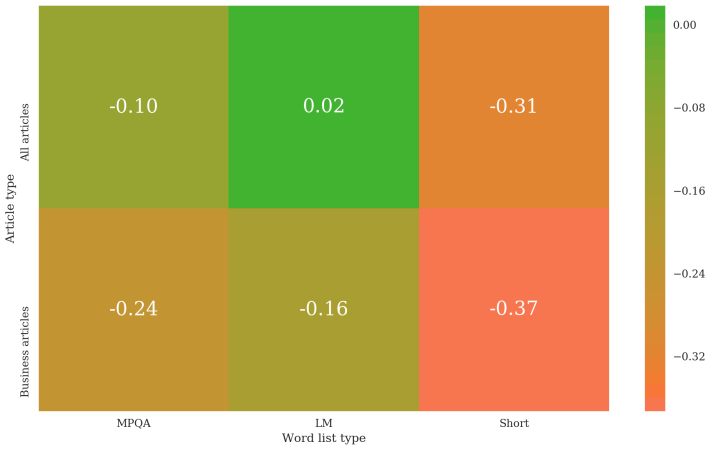
Business articles are predicted prior to 1981.

ment word lists by Wilson et al. (2005) (hereafter: MPQA) which counts 4,154 unique terms. In addition, I use the negative sentiment terms by Loughran and McDonald (2011) (hereafter: LM) which was created to fit financial texts and contains 2,355 terms. Lastly, I use an own list of negative sentiment terms which is substantially shorter (it contains 64 terms) and is tailored towards content in newspaper business articles. This list is the result of hand reading some articles which word lists had classified and of some trial and error. It has to be noted that any such word list creation is prone to over-fitting and has to stand the test truly out of sample. For the sample considered here, the reduced word list shows that already a relatively small amount of reasonable words might be carrying a substantial fractions of the signal. A full lists of the words contained in this reduced word list is provided in Appendix 4.7.5. In future research, it would be interesting to, instead of using word lists, also train a second classifier model to detect negative articles. Yet, one option would require to first manually label many business articles into "negative" or "not negative" to then train a model on them.

4.4 Results

For each word list, I compute an index by dividing the number of negative words in a given month by the total amount of words contained in all articles. This allows to create 6 indices: For each word list, an index based on all articles and an index based on only (predicted) business articles.

Figure 4.4: Correlations with GDP



$N = 289$ *observations from Q2 1947 to Q2 2019*. The indices are used in levels (i.e. the share of their words). Monthly values are then averaged to obtain quarterly values. GDP in quarterly percentage differences.

A first question is whether the classification into business articles can help to increase correlation of such indicators with aggregate variables.[5] The table shown in Figure 4.4 seems to indicate this is the case. All three word lists in fact have higher correlations (in absolute value) with GDP since 1947 if only the business articles are considered. In the correlations presented in this paper, indices are in levels (monthly values averaged to obtain quarterly ones), GDP and investment are in quarterly percentage differences.

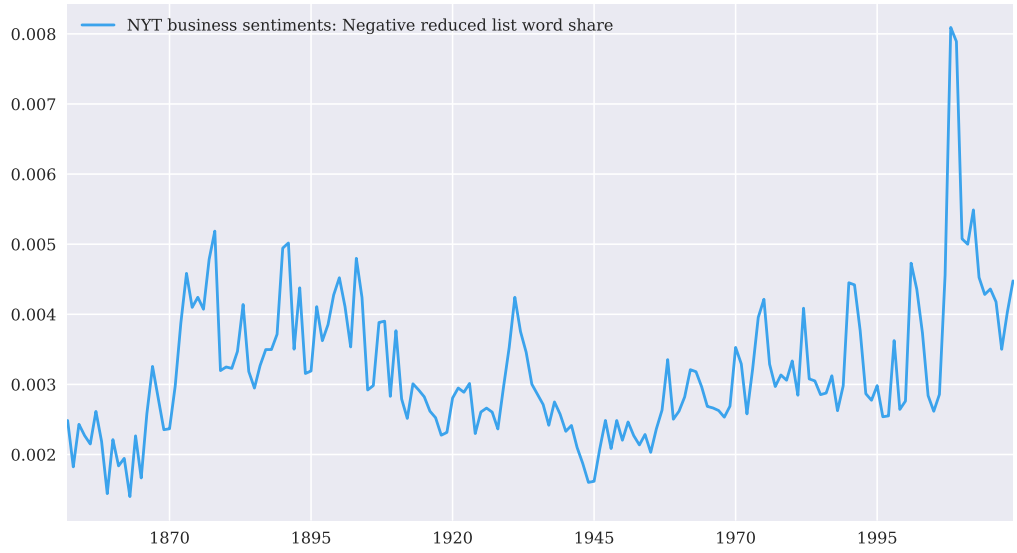
As particularly the MPQA based index seems to trend in the last years (depicted in Appendix 4.7.7), I also compute the same table for index values which have been divided by their mean level over a rolling window of 20 years in Appendix 4.7.6. The order of correlations still looks similar. I therefore choose the reduced word list on business articles as the primary index in this paper. Its resulting index of business and economic sentiments since 1852 is depicted in Figure 4.5.

Several crises over the 20th century are well visible, for example the Great Depression at the end of the 1920s or the Great Recession 2007-2009. In addition to this index, I also provide yearly Figures of the MPQA and LM indices in Appendix 4.7.7.

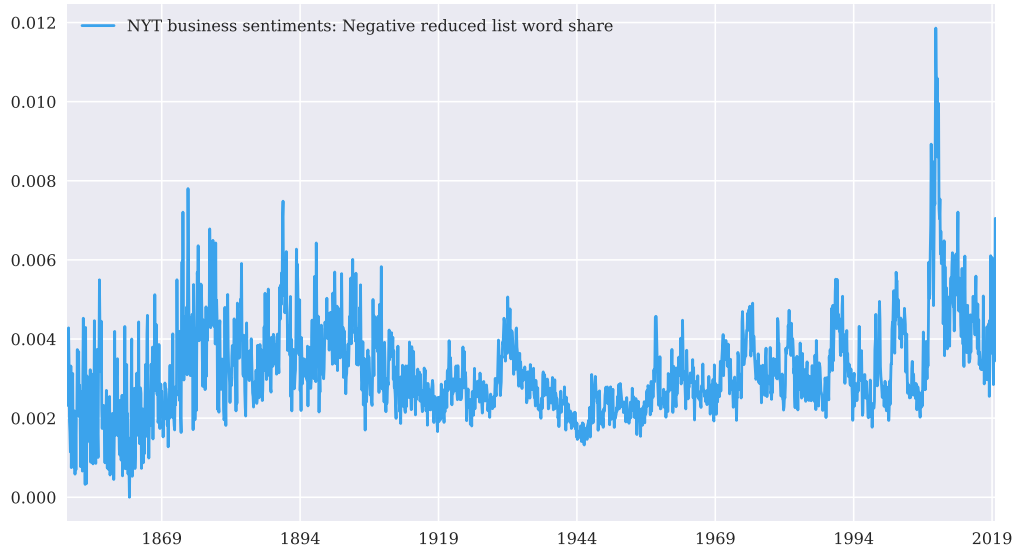
Another key question is how such an index compares to existing business sentiment indicators. For this I compute correlations of the index with GDP and investment and compare it to historical data of the "Manufacturing Business Out-

[5]GDP and investment figures have been obtained from Federal Reserve Economic Data by the St. Louis Fed.

Figure 4.5: NYT sentiment index



(a) Business index with reduced word list yearly



(b) Business index with reduced word list monthly

look Survey" from the Philadelphia Fed over the same time period.[6] Outcomes are shown in Table 4.3, for the full correlations between indices see Appendix 4.7.8.

Table 4.3: Correlations with Philadelphia Fed index since Q3 1968

	GDP	Investment
Philadelphia Fed index	0.669646	0.641193
NYT business index	-0.446995	-0.339248

"Philadelphia Fed index" refers to "Manufacturing Business Outlook Survey". $N = 204$ quarterly observations from Q3 1968 to Q2 2019

A similar table depicts the newspaper index against a frequently used current indicator, the Purchasing Manager Index (PMI): In detail, I use the "US Manufacturing Output Index". Correlations are from 2007 onwards when the PMI data was available.[7] Sample sizes are too short to draw strong conclusions and outcomes also depend on the word list in use. Yet, it seems that comparable approaches applied to richer business news data could yield interesting results.

Table 4.4: Correlations with PMI and Philadelphia Fed index since Q2 2007

	GDP	Investment
PMI Manufacturing index	0.759394	0.789468
Philadelphia Fed index	0.698511	0.613373
NYT business index	-0.720211	-0.585184

"PMI" refers to the "PMI US Manufacturing Output Index". $N = 49$ quarterly observations from Q2 2007 to Q2 2019

4.5 Extensions

A key feature of sentiment indices obtained from textual data is that they contain much wider information then just the index value itself. In the following, I provide a stylised example of this by looking into words which trended in only

[6] For this data see `https://www.philadelphiafed.org/research-and-data/regional-economy/business-outlook-survey/historical-data`; I use 'gacdfsa', the seasonally adjusted monthly general activity diffusion index; accessed 30.09.2019

[7] I am grateful to IHS Markit for sharing the US PMI data with me for this research project.

negative business articles during recessions.[8] This is done with the following simple approach: In the subset of business articles detected as having negative sentiment in a given recession, I compute the frequencies of words. Then I do the same for all negative sentiment business articles 10 years prior to the crisis of interest. The difference in the two word frequency distributions during crisis and reference period allows to detect terms which trended in the crisis. I then delete all words except for nouns and names.[9] Furthermore, I delete all words from the sentiment list (these are contained in the articles by construction) and I delete a further list of frequent terms with limited information. See Appendix 4.7.9 for these terms and a discussion. Finally, the words corresponding to the remaining 100 highest frequency differences are depicted in a word cloud.

The shape of these clouds changes with the assumptions made and the approach can only give a rough overview. Yet, outcomes look reasonable across several recessions. Figure 4.6 shows the set of trending words in negative business articles in the Great Depression. The cloud emphasises a world wide crisis with significant banking component. Further illustrations of the approach are Figure 4.7 which indicates that the 1973-1975 recession must have been an energy crisis, or Figure 4.8 which is in line with expected terms for the Great Recession such as "credit", "bank", or "mortgage". For completeness, I add such trend clouds for all NBER recessions starting from 1865 in Appendix 4.7.9. For the crisis in 1907-1908 for example, the plot suggest that it was a banking crisis (Figure 4.16 (a)). In fact this particular recession is referred to as the "Bankers' Panic" or "Knickerbocker Crisis". The trend clouds for earlier years, however, are less clear. Besides worse data quality for earlier years, one key reason is likely that word clouds only depict content from negative business articles and only few of the shorter articles in earlier years are both predicted to be about business topics and also contain at least one of the negative sentiment keywords from the short list. The early word clouds are therefore based on particularly little data. Related methodologies could also be used today rolling forward to quickly visualise newly trending terms in negative business content at high frequency.

[8]Negative business articles here are defined as those articles which have a (predicted) business label and contain at least one negative sentiment word from the short reduced word list.

[9]To be precise, I delete all words which are not nouns or "unknown" after checking them with NLTK's wordnet in Python, so e.g. names that are the same as verbs or adjectives will be deleted and some other errors will likely result through this automation.

Figure 4.6: Trending words in negative articles - Great Depression

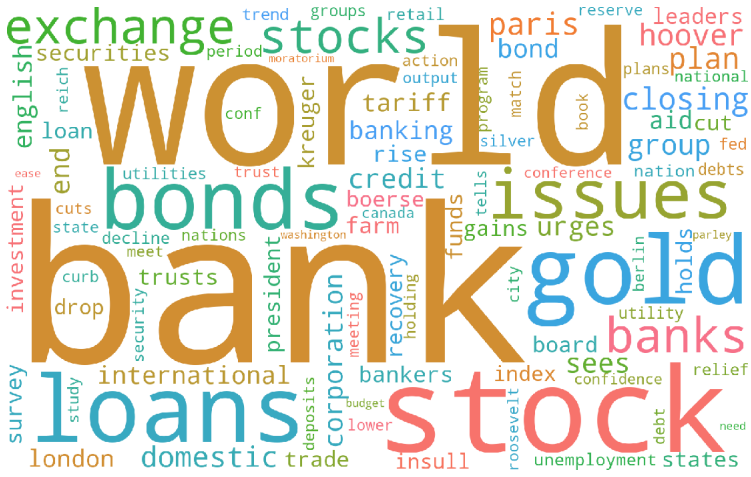


Figure 4.7: Trending words in negative articles - NBER recession 1973 to 1975

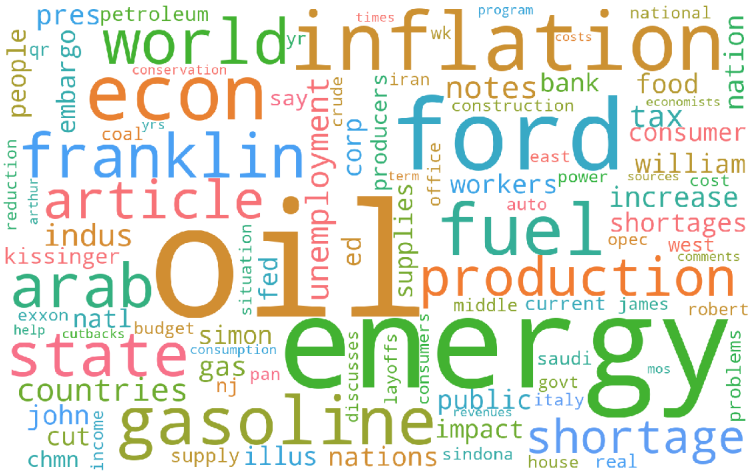
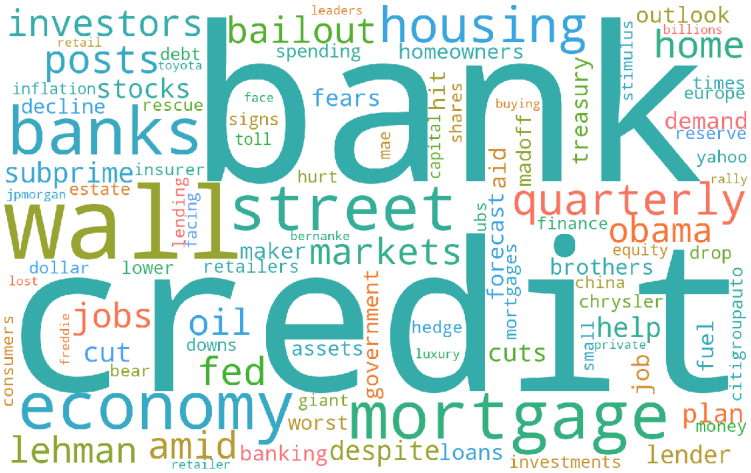


Figure 4.8: Trending words in negative articles - Great Recession



4.6 Conclusion

This paper discussed how data from historical newspapers can be used to approximate business sentiment indicators for times where such indicators were not available. Using a textual classifier, I first estimated which articles where likely about business topics. A word list with negative sentiment terms then approximated the sentiment in these articles. Correlations of the resulting index with aggregate variables and other indicators were discussed. In addition, stylised evidence of trending content present during crises in business articles with negative sentiments was presented. Similar methods can also be used to detect sentiments in real time when applied to data today, e.g. when it is unknown which texts cover business content. Approaches such as the short negative sentiment word list used here, are thereby more vulnerable to over-fitting than other approaches. In future research it could be particularly interesting to extend the methodology of this short paper to study sector specific long run historical sentiments. It could also be helpful to train a second classifier to learn predicting sentiments instead of using dictionaries (e.g. trained on a large database of business articles hand assigned to sentiments). All methods discussed here would also likely benefit from larger amounts of newspaper data than the short article snippets available in the public archive of only the NYT. Yet, as of now, other major newspapers have not provided comparable free databases. With large amounts of data and potentially based on sentiment classifiers, related approaches could be interesting to further study long run historical economic sentiments since the nineteenth century.

References

Baker, Scott R., Nicholas Bloom, and Steven J. Davis, "Measuring Economic Policy Uncertainty*," *The Quarterly Journal of Economics*, 07 2016, *131* (4), 1593–1636.

Bybee, Leland, Brian T. Kelly, Asaf Manela, and Dacheng Xiu, "The Structure of Economic News," Available at SSRN: https://ssrn.com/abstract=3446225 or http://dx.doi.org/10.2139/ssrn.3446225 2019.

Hassan, Tarek A, Stephan Hollander, Laurence van Lent, and Ahmed Tahoun, "Firm-Level Political Risk: Measurement and Effects*," *The Quarterly Journal of Economics*, 08 2019, *134* (4), 2135–2202.

Hastie, Trevor, Robert Tibshirani, Jerome Friedman, and James Franklin, "The elements of statistical learning: data mining, inference and prediction," *The Mathematical Intelligencer*, 2005, *27* (2), 83–85.

James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani, *An introduction to statistical learning*, Vol. 112, Springer, 2013.

Larsen, Vegard and Leif Anders Thorsrud, "Business Cycle Narratives," https://ssrn.com/abstract=3338822 2019.

Loughran, Tim and Bill McDonald, "When Is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks," *The Journal of Finance*, 2011, *66* (1), 35–65.

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

Püttmann, Lukas, "Patterns of Panic: Financial Crisis Language in Historical Newspapers," Available at SSRN: https://ssrn.com/abstract=3156287 or http://dx.doi.org/10.2139/ssrn.3156287 2018.

Tuckett, David and Rickard Nyman, "The Relative Sentiment Shift Series for Tracking the Economy," 05 2017.

Wilson, Theresa, Janyce Wiebe, and Paul Hoffmann, "Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis," Proc. of HLTEMNLP 2005.

4.7 Appendix

4.7.1 NYT archive common materials

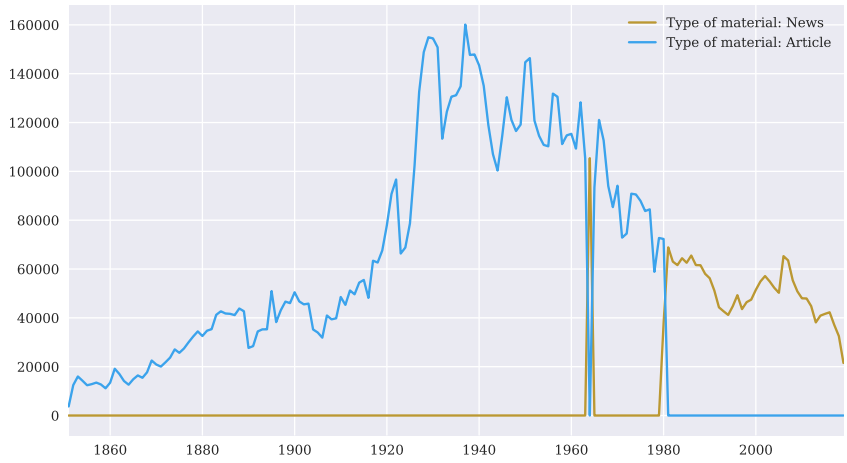
Table 4.5: Common material in the NYT archive over decades

Decade	Year	Obituary,bio	Correction	Article	Front Page	Death Notice	Letter	Op-Ed	Review	Blog	Video	Summary	Caption	News	Editorial Cartoon	Birth Notice	Editorial	Marriage Ann.
1850s	16695	2735	0	109016	34129	0	4310	0	223	0	0	0	0	0	0	44	15	2224
1860s	18645	4091	0	163320	42842	0	4075	0	197	0	0	0	0	0	0	55	14638	3101
1870s	18745	4989	0	263451	74064	0	4491	0	71	0	0	0	0	0	0	44	31108	3056
1880s	18845	7202	0	397687	96886	0	3977	0	0	0	0	0	0	0	1	37	37752	3113
1890s	18945	9707	0	386154	90314	0	10149	0	5475	0	0	0	0	0	7	33	50143	4407
1900s	19045	14781	0	409859	97063	0	29967	0	14793	0	0	0	0	0	4	433	46922	7840
1910s	19145	15387	0	546521	68434	0	28559	0	4863	0	0	0	0	0	42	2553	32782	10929
1920s	19245	26355	0	1017789	54662	0	21957	0	8072	0	0	0	0	0	99	3619	32992	14070
1930s	19345	57725	0	1395314	48289	0	25233	0	18485	0	0	0	0	0	1309	4104	34653	31307
1940s	19445	90514	0	1206235	48638	0	19967	0	18141	0	0	0	0	0	5688	6015	35345	50849
1950s	19545	81016	0	1235827	47467	0	19168	0	14079	0	0	0	0	0	4879	7438	32469	53701
1960s	19645	40390	0	964790	48284	0	21079	0	14406	0	0	0	0	105408	2008	4116	26813	43129
1970s	19745	17374	0	810553	39374	0	15289	0	5882	0	0	0	0	3	441	963	15146	6885
1980s	19845	21240	7781	72300	3060	0	45989	9497	48102	0	0	2448	9264	605065	0	0	14824	1092
1990s	19945	32868	18856	0	0	42941	62817	1990	52597	0	0	17903	16291	467124	0	0	13290	0
2000s	20045	6152	36905	0	0	114075	70677	13561	54042	135629	5013	8437	7837	556025	0	0	10492	0
2010s	20145	2655	9300	0	0	57506	18923	27314	41469	234353	24798	0	0	394858	0	0	1801	0

Common types contained in the archive under "type_of_material". Note, however, that front page indicators from 1980 onwards are, among others, also stored in the section information available the archive from then.

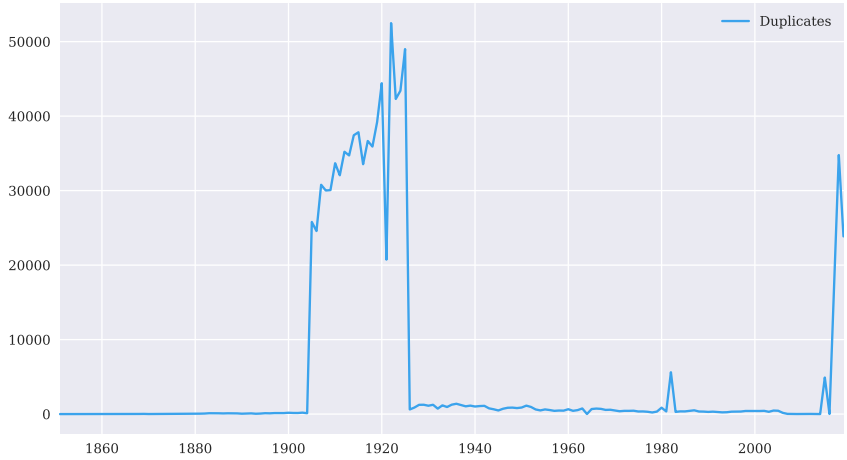
4.7.2 Archive caveats

Figure 4.9: Counts in categories that store main articles



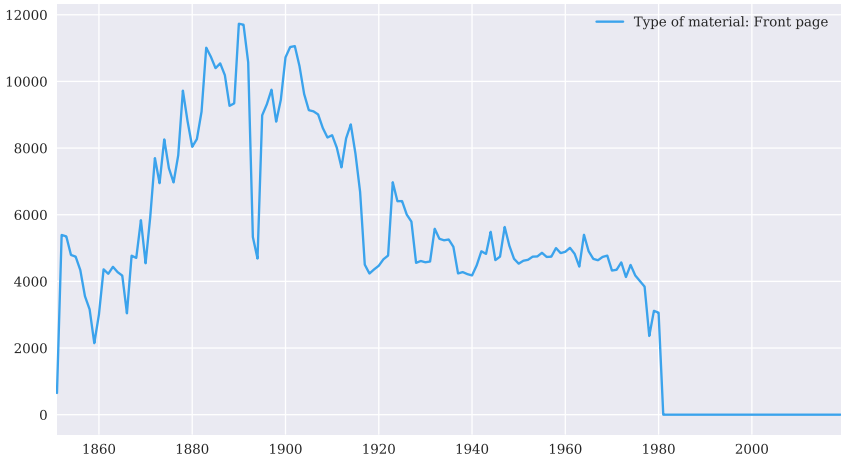
The main class which contains articles switches from the 1980s onwards, however, also for one year before.

Figure 4.10: Duplicates



In the data downloaded from the API, there seem to be substantial duplicates in irregular intervals.

Figure 4.11: Front page



Until 1980 there is a category in type of material which indicates front pages, however, it stops to exist. In subsequent years this information can be found in the section information. Yet, for the most recent years it also seems to disappear from there.

4.7.3 Cleaned article counts

Table 4.6: Cleaned overall articles and (predicted) business articles

	All articles	Business articles
1852	12902	1096
1853	14817	1346
1854	13855	1270
1855	12472	982
1856	12718	1164
1857	12931	1505
1858	12202	1178
1859	10312	901
1860	14405	1721
1861	20927	2214
1862	19368	1729
1863	15527	1330
1864	14916	1334
1865	16573	1794
1866	16663	1588
1867	17907	1912
1868	19529	1716
1869	25785	3100
1870	23049	2615
1871	23915	2798
1872	26101	2822
1873	27552	3686
1874	32595	3790
1875	30956	3664
1876	31715	3568
1877	35262	4588
1878	39041	5033
1879	40660	4787
1880	36852	3823
1881	39117	4401
1882	40918	5144
1883	48892	6209
1884	51565	6918
1885	50233	6553
1886	50356	6533
1887	49589	6941
1888	51401	7374
1889	50228	6975
1890	34045	4130
1891	35822	4468
1892	40689	4650
1893	38734	6109
1894	38959	5698
1895	54104	5571
1896	42841	5100
1897	48874	6447
1898	51358	6297
1899	51474	6783
1900	55560	8077
1901	52201	9312
1902	50430	9586
1903	50681	9619
1904	40824	6241
1905	39314	6143
1906	37663	6834
1907	44814	7616
1908	43806	6832
1909	44122	6733
1910	52157	9230
1911	49239	7434
1912	54169	8936
1913	53749	9027
1914	58067	9752
1915	58977	9813
1916	51423	8803
1917	64434	12969
1918	63237	12841
1919	68403	13517
1920	78259	16741
1921	91188	22203
1922	96967	23492
1923	69616	15797
1924	71891	16207
1925	81508	18391
1926	104911	23677
1927	134244	32067
1928	147395	33702
1929	153072	36897
1930	152170	34934
1931	148543	36319
1932	115005	28609
1933	125223	34190
1934	131748	34410
1935	132219	34081
1936	135854	33237
1937	161444	38033
1938	149268	37036
1939	146343	35298
1940	141941	33556
1941	133493	32095
1942	117799	27126
1943	105512	25335
1944	99616	23918
1945	113708	27376
1946	130312	31605
1947	121921	30823
1948	117610	28138
1949	119986	29557
1950	141642	32217
1951	142973	32191
1952	120500	27219
1953	114932	25594
1954	112024	25678
1955	111401	26145
1956	129162	27464
1957	127703	26235
1958	110631	24374
1959	112390	25211
1960	115081	25222
1961	110434	24848
1962	125122	25151
1963	103775	21767
1964	105231	22142
1965	95388	20888
1966	118732	23531
1967	110892	21618
1968	92761	18800
1969	84469	17266
1970	91625	17269
1971	74701	16425
1972	77734	15521
1973	92523	17146
1974	91679	19342
1975	85305	17761
1976	80835	15216
1977	79910	15494
1978	58403	11945
1979	72031	16534
1980	109289	24971
1981	68518	13435
1982	62899	11953
1983	61514	11739
1984	64362	12426
1985	62508	12215
1986	65498	12630
1987	61582	11718
1988	61568	12936
1989	58044	12539
1990	56157	12179
1991	51136	10976
1992	44235	7888
1993	42644	6954
1994	41132	6838
1995	44588	7840
1996	49102	9775
1997	43551	5998
1998	46417	6718
1999	47457	8111
2000	51459	9143
2001	54898	9869
2002	57107	9956
2003	54911	9330
2004	52270	8191
2005	50209	7856
2006	65015	10086
2007	63190	11710
2008	55218	10654
2009	50795	9492
2010	47933	9001
2011	47903	8799
2012	44783	7611
2013	38131	6674
2014	40896	6958
2015	41507	7648
2016	42273	7022
2017	37123	5095
2018	32528	3715
2019	21445	2275

4.7.4 Confusion matrix training data

Table 4.7: Training data confusion matrix for random forest business article classifier

	Predicted: Business and finance	Predicted: Other
True: Business and finance	205951	2
True: Other	9	209549

Training accuracy of approximately 1. $N = 415,511$ *training observations.*

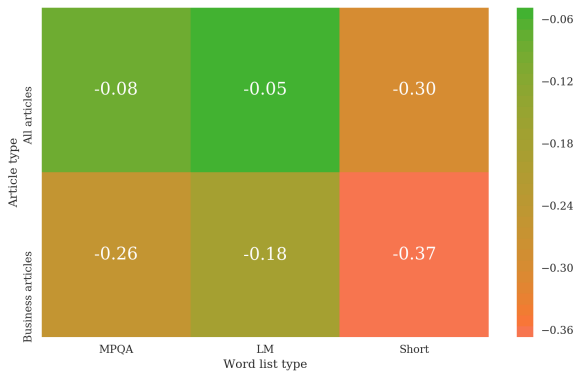
4.7.5 Reduced word list

Short business sentiment word list consisting of 64 terms.

'alarming', 'bad', 'bankrupt', 'bankruptcy', 'concerned', 'concerning', 'concerns', 'crises', 'crisis', 'depressed', 'depression', 'deteriorate', 'deteriorated', 'deteriorates', 'deterioration', 'dire', 'disappoint', 'disappointed', 'disappointing', 'disappoints', 'downturn', 'fail', 'failed', 'failing', 'failings', 'fails', 'failure', 'failures', 'insolvency', 'insolvent', 'layoff', 'loss', 'losses', 'pessimistic', 'recession', 'recessionary', 'recessions', 'slowdown', 'slowdowns', 'sluggish', 'slump', 'stagnate', 'stagnated', 'stagnates', 'stagnating', 'stagnation', 'struggle', 'struggled', 'struggles', 'struggling', 'troubled', 'weak', 'weaken', 'weakened', 'weakening', 'weaker', 'weakest', 'worried', 'worries', 'worry', 'worrying', 'worse', 'worsen', 'worsening'

4.7.6 Heatmap normalised values

Figure 4.12: Correlations with GDP



$N = 289$ *observations from Q2 1947 to Q2 2019. Index values have been normalised by the mean of the last 20 years.*

4.7.7 Indices of different word lists

Figure 4.13: Yearly NYT index based on the MPQA negative word list

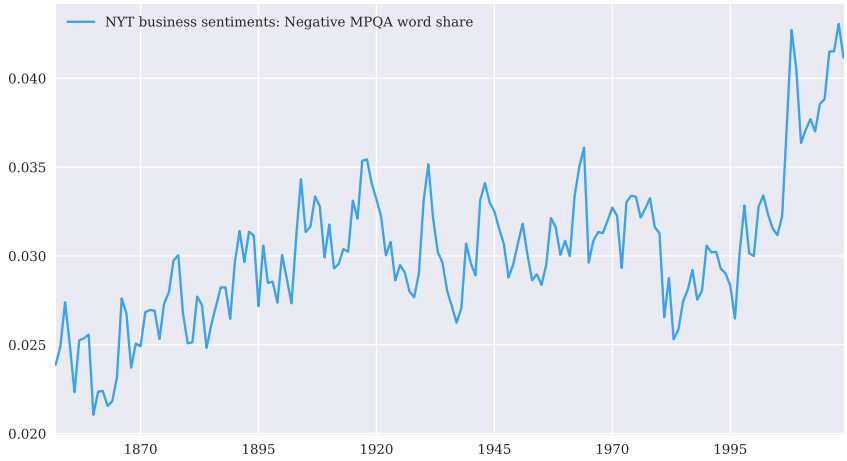
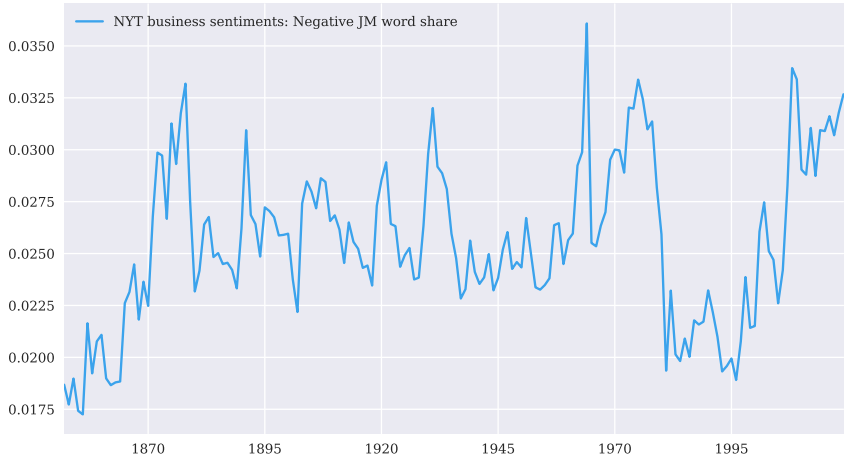


Figure 4.14: Yearly NYT index based on the LM negative word list



4.7.8 Correlations with business indicators

Table 4.8: Correlations with Philadelphia Fed index since Q3 1968

	Phil. Fed Index	GDP	Investment	NYT business index
Phil. Fed Index	1.000000	0.669646	0.641193	-0.486263
GDP	0.669646	1.000000	0.817478	-0.446995
Investment	0.641193	0.817478	1.000000	-0.339248
NYT business index	-0.486263	-0.446995	-0.339248	1.000000

"Philadelphia Fed index" refers to "Manufacturing Business Outlook Survey". $N = 204$ quarterly observations from Q3 1968 to Q2 2019.

Table 4.9: Correlations with PMI and Philadelphia Fed index since Q2 2007

	PMI	GDP	Investment	NYT index	Phil. Fed index
PMI	1.000000	0.759394	0.789468	-0.772415	0.746871
GDP	0.759394	1.000000	0.818107	-0.720211	0.698511
Investment	0.789468	0.818107	1.000000	-0.585184	0.613373
NYT index	-0.772415	-0.720211	-0.585184	1.000000	-0.833448
Phil. Fed index	0.746871	0.698511	0.613373	-0.833448	1.000000

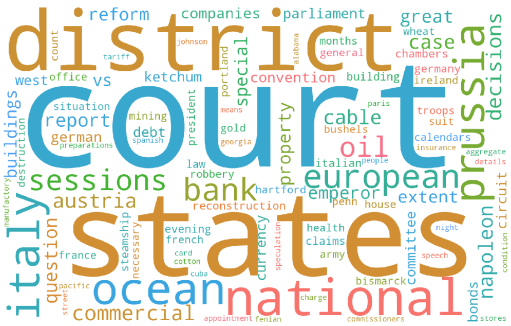
"PMI" refers to the "PMI US Manufacturing Output Index". $N = 49$ quarterly observations from Q2 2007 to Q2 2019.

4.7.9 Trending words in NBER recessions

All words from the sentiment list are omitted from the following word clouds as by definition all articles contain them. Furthermore, I mute a list of words which have arguably relatively low information content, however are very frequently trending, e.g. because of referring to dates or quantities. I also exclude some other frequently trending words such as e.g. those related to firms such as "company". Unlike "bank" (which I do not exclude) "company" has arguably much lower information as in almost all recessions firms make lower profits, however, only some recessions are banking crises. The full list of additionally excluded words is (with the only exception being 2001 where some additional terms related to the 9/11 attacks are excluded to make the business content of the recession visible):

['january', 'februray', 'march', 'april', 'may', 'june', 'july', 'august', 'september', 'october', 'november', 'december', 'jan', 'feb', 'ap', 'may', 'jun', 'jul', 'aug', 'sep', 'sept', 'oct', 'nov', 'dec', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight', 'nine', 'ten', 'first', 'second', 'third', 'fourth', 'fith', 'sixth', 'seventh', 'eight', 'ninth', 'tenth', 'percent', 'percentage', 'million', 'billion', 'today', 'yesterday', 'day', 'week', 'month', 'quarter', 'quarters', 'year', 'monday', 'tuesday', 'wednesday', 'thursday', 'friday', 'saturday', 'sunday', 'fell', 'rose', 'raise', 'reports', 'rept', 'briefing', 'point', 'points', 'says', 'cent', 'cents', 'mr', 'mrs', 'ms', 'profit', 'sales', 'revenue', 'prices', 'price', 'business', 'company', 'firm']

Figure 4.15: Trending words in negative business articles - NBER recessions (I/III)



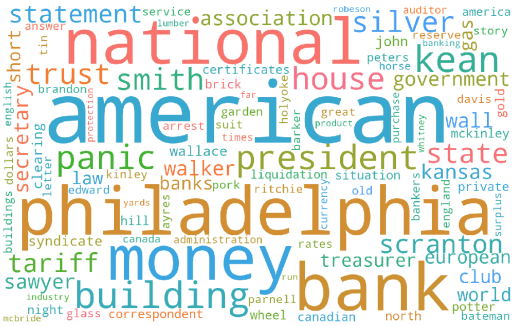
(a) 1865 to 1867

(b) 1869 to 1870

(c) 1873 to 1879

(d) 1882 to 1885

(e) 1887 to 1888



(f) 1890 to 1891

(g) 1893 to 1894

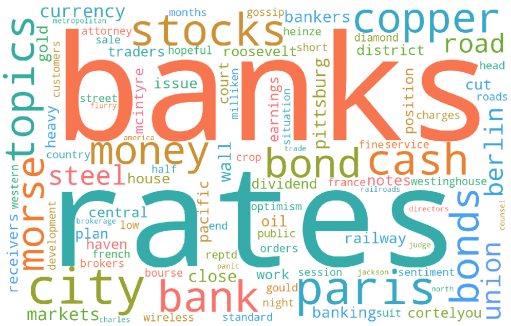
(h) 1895 to 1897

(i) 1899 to 1900

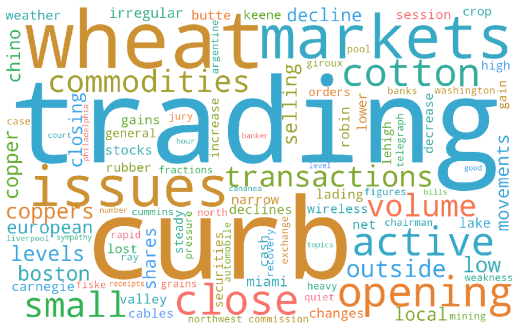
(j) 1902 to 1904

[a]For start and end months of the word clouds, see the NBER US Business Cycle Expansions and Contractions list `https://www.nber.org/cycles.html`.

Figure 4.16: Trending words in negative business articles - NBER recessions (II/III)



(a) 1907 to 1908

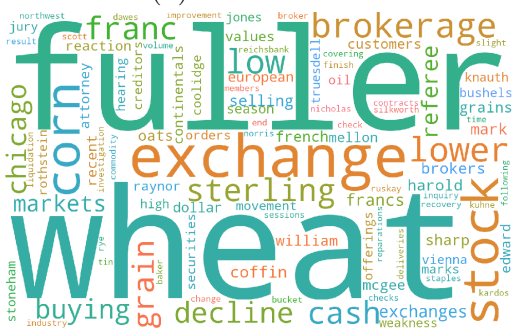


(b) 1910 to 1912

(c) 1913 to 1914

(d) 1918 to 1919

(e) 1920 to 1921

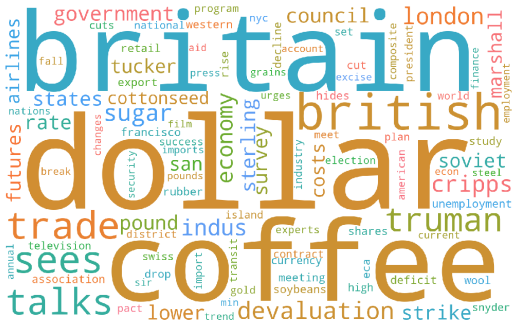


(f) 1923 to 1924

(g) 1926 to 1927

(h) 1937 to 1938

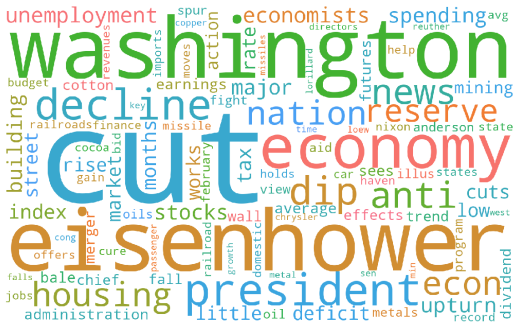
(i) 1945 to 1945



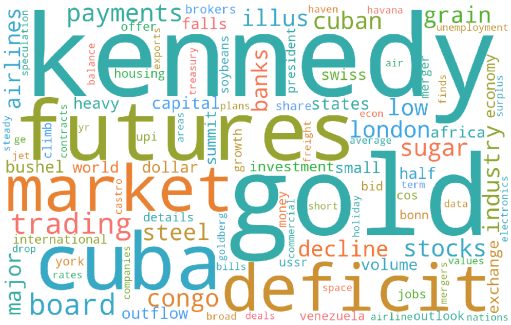
(j) 1948 to 1949

Figure 4.17: Trending words in negative business articles - NBER recessions (III/III)

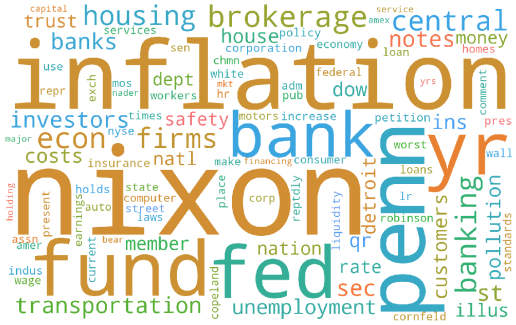
(a) 1953 to 1954



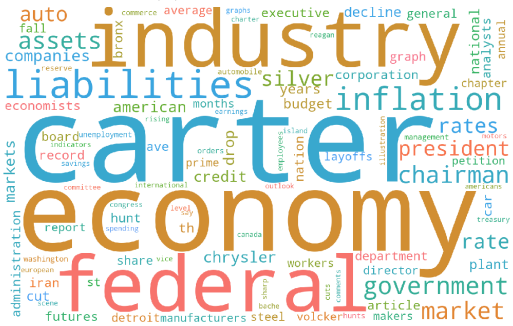
(b) 1957 to 1958



(c) 1960 to 1961

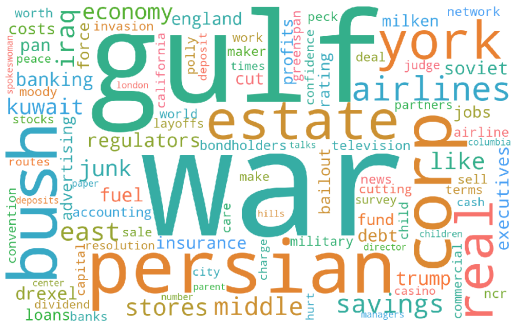


(d) 1969 to 1970

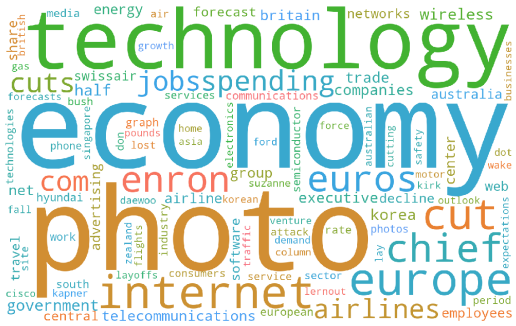


(e) 1980 to 1980

(f) 1981 to 1982



(g) 1990 to 1991



(h) 2001 to 2001