The Changing Economics of Knowledge Production

Simona Abis and Laura Veldkamp
Columbia University, NBER and CEPR*

March 20, 2021

Abstract

Big data technologies change the way in which data and human labor combine to create knowledge. Is this a modest technological advance or a data revolution? Using hiring and wage data from the financial sector, we estimate firms' data stocks and the shape of their knowledge production functions. Knowing how much production functions have changed informs us about the likely long-run changes in output, in factor shares, and in the distribution of income, due to the new, big data technologies. Using data from the investment management industry, our results suggest that the labor share of income in knowledge work may fall from 29% to 21%. The change associated with big data technologies is two-thirds of the magnitude of the change brought on by the industrial revolution.

Machine learning, artificial intelligence (AI), or big data all refer to new technologies that reduce the role of human judgment in producing usable knowledge. Is this an incremental improvement in existing statistical techniques or a transformative innovation? The nature of this technological shift is similar to industrialization: In the 19th and 20th centuries, industrialization changed the capital-labor ratio, allowing humans to use more machines, factories and sophisticated tools to be more efficient producers of goods and services. Today, machine learning is changing the data-labor ratio, allowing each knowledge worker to leverage

*Preliminary work. Numerical estimates will surely change. Comments welcome. Email: lv2405@columbia.edu or sa3518@columbia.edu. Columbia Business School, 3022 Broadway, New York, NY 10025. We are grateful to PayScale for use of their wage data. Thanks to Ellen McGrattan and the participants of the NBER economic growth group, women in finance conference and Columbia macro and finance lunches and seminars at the IMF and Princeton for comments. Thanks also to many graduate students, including Sahil Arora, Sagar Agarwal, Samrat Halder, Anshuman Ramachandran, Ran Liu and Saad Naqvi, who substantially advanced the project with their research assistance. We acknowledge the generous financial support and helpful feedback from IBM and the Columbia University Data Science Institute. Keywords: FinTech, machine learning, artificial intelligence, big data, labor share.

more data, to be a more efficient producer of knowledge. Given the myriad of differences between the industrialization era and today's knowledge economy, and the early stage of data technology adoption, how might one compare the magnitude of today's change with its historical counterpart? Economists model industrialization as a change in production technology: a move from a technology with starkly diminishing returns to capital, to one with less diminishing returns. The size of the industrial revolution can therefore be summarized by the magnitude of the change in the production parameter that governs diminishing returns. That same statistic can be estimated for knowledge production, using old and new data technologies. Measuring how much big data technology adoption changes the diminishing returns to data and comparing this to the change that took place during the industrial revolution informs us about whether this is a useful, but common innovation, or the next economic revolution.

Using labor market data from the financial sector, we estimate two production functions – one for classical data analysis and one for machine learning. The decline in diminishing returns to data shows up as an exponent on data in the production function that is closer to one: We estimate that the data exponent rose from 0.711 to 0.791. The 0.08 increase in the parameter governing diminishing returns implies that knowledge-producing firms should optimally have more data per worker, or equivalently, fewer workers for a given size data set. This change also affects wages. It predicts an 8% decline in the share of firm profits paid to labor. Such a change in the profit share could matter for income inequality. The flip side of the declining labor share is an 8% increase in the share of knowledge profits paid to data owners. In other words, new data technologies structurally increase the value of data as an asset and enrich those who own the data. Finally, the magnitude of these shifts represent a change in production that is about two-thirds of the size of the change experienced during the industrial revolution.

Estimating old and new knowledge production functions is challenging, because for most firms, we do not know how much data they have, nor how much knowledge they create, nor do they announce which technology or what mix of technologies they employ. What we can observe is hiring, skill requirements and wages. A simple model of a two-layer production economy teaches us how to infer the rest. The two layers of production are as follows: Raw data is turned into usable, processed data (sometimes called information) by data managers; processed data and data analyst labor combine to produce knowledge. Thus, we use hiring of data managers to estimate the size of the firm's data stock, the skills mix of analysts to estimate the mix of data technologies at work, and we bypass the need to measure knowledge by using wage data to construct income shares, which inform us about the returns, and the rate of diminishing returns, to each factor.

To estimate production functions, it is imperative that we precisely categorize job postings

and match postings by employer. Unlike other work that measures machine-learning-related employment (e.g., Acemoglu and Restrepo (2018)), our work demands a finer partition of jobs. We need to distinguish between workers that prepare data to be machine-analyzed, workers that primarily use machine learning, and workers that use statistical skills that are of a previous vintage. We also need to know whether data managers are being hired by the same firm that is also hiring machine-learning analysts.

Because different industries have different job vocabularies, we can categorize jobs more accurately by focusing on one industry: finance, more specifically we focus on investment management. Since investment management is primarily a knowledge industry, with no physical output, it is a useful setting in which to tease apart these various types of knowledge jobs. According to Webb (2019) and Brynjolfsson et al. (2018a), finance is also the industry with the greatest potential for artificial intelligence labor substitution. We use Burning Glass hiring data, including the textual descriptions of each job, to isolate financial analysis jobs that do and do not predominantly use machine learning, as well as data management jobs, for each company that hires financial analysts. We adjust the number of job postings by a probability of job filling. That product is our measure of a company's desired addition to their labor force. This series of worker additions, along with job separations by job category, enables us to build up a measure of each firm's labor stock.

The next challenge is to estimate the amount of data each firm has. We consider data management work to be a form of costly investment in a depreciating data asset. Therefore, we use the job postings for data managers, the job filling and separation rates for such jobs, and an estimate of the initial data stock to construct data inflows (investments), per firm, each year. To estimate the 2010 initial stock of data of each financial firm, we estimate which stock best rationalizes the firm's subsequent hiring choices. Specifically, we choose an initial stock of data that minimizes the distance between each firm's actual hiring and the optimal amount of hiring in each category, dictated by the firm's first order conditions. Combining this initial stock, with a data depreciation rate and a data inflows series gives us an estimate of the size of the data stock that every financial firm has in its data warehouse.

Armed with data stocks, labor forces in each category, and wages from PayScale, we estimate the data and labor income shares. These income shares correspond to the exponents in a Cobb-Douglas production function. We estimate a constant-returns Cobb-Douglas specification because we are exploring the analogy that AI is like industrialization and this is the type of production function most often used to describe industrial output. Therefore, we model knowledge production in a parallel way to industrialization, to facilitate comparison, while recognizing the non-rival nature of data. By comparing the estimated exponent for classical data analysis and machine-learning data analysis, we can assess the magnitude of the technological

change.

This approach bypasses two forces: The role of capital and the potential for increasing returns. Typically, knowledge is combined with capital, real or financial, to generate profits, in a production function that might exhibit increasing returns. The start of Section 1 shows how we could incorporate either feature in our model of firm profits, without changing how we estimate the production of knowledge. As long as there exists some amount of knowledge that produces $1 in profit, at each point in time, we can estimate how data and labor combine to create that amount of knowledge, without taking a stand on how that knowledge will be used to create profit.

Our data reveals a steady shift underway in the employment of knowledge workers in the investment management sector. We see a steady increase in the fraction of the workforce skilled in new big data technologies. However, while the declining labor share might lead one to expect fewer knowledge workers, we find an increase in the size of the sector large enough so that even though the share shrinks, the number of workers and their pay rises. Even for workers with the old skills, jobs are still abundant. The number of old technology jobs in the sector has not fallen; it simply represents a smaller share of employment. While AI job postings were a tiny fraction of all analysis jobs through 2015, by the end of 2018, about $1/7^{th}$ of all financial analysts in investment management firms had big data or AI-related skills. This shifts we measure are just the first few years of adoption of this new technology. But they indicate the direction of a transformation that we expect to continue for years to come.

Related Literature Our paper is most closely related to the literature exploring the shape of production functions Jones (2005) and the nature of structural economic transformations (Acemoglu and Guerrieri, 2008; Lagakos and Waugh, 2013; Buera and Kaboski, 2009; Cheremukhin et al., 2017). Such shifts in production are also related to the changes in the labor share of income (Karabarbounis and Neiman, 2014). Just like these existing papers, estimating how much the production function has changed allows us a more holistic understanding of the nature of the transformation. What differs is the scope of the analysis and the historial or future orietation. While these papers look backwards at what trends have been, this paper projects forward, by considering how a new technology in its infancy is contributing to the labor share decline. A structural model allows us to forecast and to make inferences about future income redistribution. By focusing on knolwedge production, the scope of our project is surely more narrow. But this allows us to speak more specifically to policy-relevant questions about how the data economy is changing.

Models of the role of data in the process of economic growth (Jones and Tonetti, 2020; Agrawal et al., 2018a; Aghion et al., 2017; Farboodi and Veldkamp, 2019) share our model-

based approach but equate data and knowledge. In these theories, firms accumulate a stock of useable knowledge that enhances productivity or facilitates prediction. In contrast, this study unpacks how raw data is transformed into that valuable output-enhancing knowledge.

On the topic of big data technologies, many recent working papers use labor market data to investigate how machine learning and artificial intelligence are affecting labor demand. They primarily use a difference-in-difference approach. Acemoglu and Restrepo (2018), Babina et al. (2020) and Deming and Noray (2018) identify industries and/or regions that are more exposed to machine learning-related technology. Then, controlling for other labor-related variables, they report how many jobs have been lost or gained, relative to unexposed regions or industries. Others offer useful inputs in this exercise by reporting the number of AI jobs postings or patents by industry and occupation (Cockburn et al., 2018; Alekseeva et al., 2020). Agrawal et al. (2017) and Agrawal et al. (2018b) argue that machine learning is likely to be a general purpose technology, because of the breadth of industries in which it is being adopted. In our approach, the number of jobs gained or lost due to machine learning to date is an important piece of evidence; it informs our work. But labor demand is not our main question. It is just one piece of our puzzle. Because our focus is on how the technology affects knowledge production, we need to use a different, structural approach.[1]

Measuring data and its value is a complement to work that estimates the value of intangible assets (Crouzet and Eberly, 2020; McGrattan, 2020). However, the difference in our main question necesscitates a different approach. The objective of previous work was decomposing the sources of value in a firm. Using Q theory, they backed out features of a production function from asset prices and book values. We are interested in how much two technologies, often both used within the same firm, differ. A hiring-based approach is more suitable for our question because we can identify workers using one technology or another. We cannot tell what firm value is attached to each mode of production, within the same firm. It is the skills required in posted jobs that reveals what technologies the firm is using.

In what follows, Section 1 sets up a three-equation model that is the basis of our estimation and derives optimality conditions that we use to infer parameters from our data. Section 2 describes the data and how we use it to assemble variables that correspond to objects in the model. Section 3 presents the estimation results, explores changes in employment, wages, and the cross-firm heterogeneity which informs our estimated parameters. We also estimate the

[1] The literature on automation and robotics asks similar questions about production of physical goods (Berg et al., 2018). Our focus is on knowledge production, rather than manual task automation. The scope for computers to replace human thought and judgment may be quite different from their ability to replicate repetitive physical movements. Others examine the productivity gains or potential discrimination costs that follow the adoption of AI techniques in providing credit (Fuster et al., 2018), in equity analysis (Grennan and Michaely, 2018), or in deep learning more generally (Brynjolfsson et al., 2017). These insights are also distinct from the question of how knowledge production is changing.

value of firms' data stocks. Section 4 concludes.

1 A Model for Measurement

The objective in writing down this model is not to provide insight into new economic mechanisms, nor it is to provide the most realistic, detailed description of financial knowledge production. Rather, the goal is to write down a simple framework that maps objects we observe into those that we want to measure. It needs to relate hiring to labor as well as quantities and prices of labor to data stocks and knowledge production. There are three types of workers: AI (artificial intelligence) analysts, old technology (OT) analysts, and data managers. We use AI as a shorthand to denote a diverse array of big data technologies. The data managers create structured data sets, which, along with labor, are the inputs into knowledge production. Among data managers we also include workers who select, purchase and integrate externally produced data sets into the firm's databases. We define as data (D) only information that is readily available for analysis. This production process is illustrated in Figure 1.

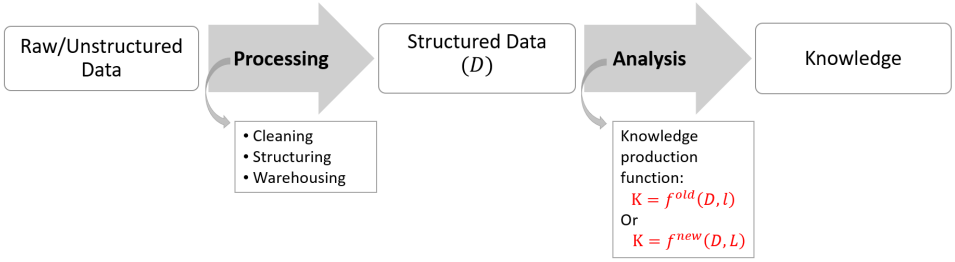

Figure 1: Production process for knowledge

The new technology knowledge production function is:

$$K_{it}^{AI} = A_t^{AI} D_{it}^{\alpha} L_{it}^{1-\alpha}, \tag{1}$$

where D_{it} is structured data, L_{it} is labor input for data analysts with machine-learning skills, and K_{it}^{AI} is the knowledge generated using the new technology. The old technology knowledge production function is:

$$K_{it}^{OT} = A_t^{OT} D_{it}^{\gamma} l_{it}^{1-\gamma}, \tag{2}$$

where l_{it} is labor input for data analysts with traditional analysis skills, K_{it}^{OT} is the knowledge generated using the old technology. A_t^{AI} and A_t^{OT} are time-varying productivity parameters.

We use a Cobb-Douglas production function for knowledge because it offers a clear mapping

between incomes shares and the production function parameters and it facilitates our comparison between new data technologies and the changes induced by industrialization. A Cobb-Douglas approach is also supported by Jones (2005). Our specification does embody the non-rival nature of data: both technologies make use of the same data set, at the same time.

Of course, one might object to assuming constant returns to scale, within each type of knowledge production. However, keep in mind that this is not different from what the growth literature does with idea production. Idea or technology production is typically produced using constant, or even diminishing returns. Then the ideas or technologies themselves enter into goods production in a way that creates increasing returns. In our setting, the analog to the increasing returns in growth models would be a final goods sector that produced with increasing returns to scale in knowledge, capital and labor: $(\text{final output}_{it}) = (K_{it}^{OT} + K_{it}^{ML})\,\text{capital}^{\zeta}\ \text{labor}^{1-\zeta}$. For our measurement exercise, we do not need to take a stand on this form of final goods production. But our exercise does not rule out increasing returns to knowledge.

Similarly, one could include capital in the knowledge production function. We exclude it for simplicity, beacuse it is small and fairly constant. For the types of financial analysis firms we examine, physical capital is a small, stable fraction of their firm value. In our measurement, the value of capital is simply reflected in the residual productivity term A_t.

Finally, this structure also implies that the nature of the data inputs is the same for both types of analysis. This simplifies measurement, but the obvious counterfactual would be: Machine learning can make use of a broader array of data types than traditional analysis. One way to interpret this is that it is the source of greater decreasing returns to data from the old technology. Suppose that data is ordered, from easily usable to difficult to use. Once the easiest data is incorporated, the next additional piece of data for traditional analysis has very low marginal value. For machine learning, that next piece of data has higher marginal value. Thus, the difference in the usability of data could be the primary reason for the difference in returns to data.

Data management and Data Stocks. Data inputs for analysis are not raw data. They need to be structured, cleaned and machine-readable. This requires labor. Suppose that structured data, sometimes referred to as "information," is produced according to $\lambda_{it}^{1-\phi}$, where λ_{it} is labor input for data managers.[2] Labor with diminishing marginal returns can turn raw or purchased

[2] One might be tempted to add a producutivity term A^{DM} to the data production function. However, such a term would not be identified. The reason is that data does not have natural units. Multiplying production and initial data by a constant is just a change of units of data. Multiplying D_{it} by a constant simply creates a constant that can be included in A^{AI} and A^{OT}. So if we re-interpret those parameters as productivity, relative to the productivity of data production, the rest of the estimates are unchanged.

data into an integrated, searchable data source that the firm can use. New processed data is added to the existing stock of processed data. But data also depreciates at rate δ. Overall, processed data follows the dynamics below:

$$D_{i(t+1)} = (1-\delta)D_{it} + \lambda_{it}^{1-\phi} = D_{i0}(1-\delta)^t + \sum_{s=0}^{t}(1-\delta)^{t-s}\lambda_{is}^{1-\phi}. \tag{3}$$

If we estimate the rate of diminishing returns to data management labor λ_{it}, initial data D_{i0} and the depreciation rate δ, we can recover D_{it} from data management labor λ_{it}.

Equilibrium We are interested in a competitive market equilibrium where all firms choose the three types of labor to maximize firm value. We can express this problem recursively, with the firm's data stock as the state variable. In this equilibrium, each firm i solves the following optimization problem:

$$v(D_{it}) = \max_{\lambda_{it}, L_{it}, l_{it}} A_t^{AI} D_{it}^{\alpha} L_{it}^{1-\alpha} + A_t^{OT} D_{it}^{\gamma} l_{it}^{1-\gamma} - w_{L,t}L_{it} - w_{l,t}l_{it} - w_{\lambda,t}\lambda_{it} + \frac{1}{r}v(D_{i(t+1)}) \tag{4}$$

$$\text{where } \; D_{i(t+1)} = (1-\delta)D_{it} + \lambda_{it}^{1-\phi}, \tag{5}$$

and $v(D_{it})$ is the present discounted value of firm i's data stock at time t. Note that we have implicitly normalized the price of knowledge to 1. This is not restrictive because knowledge does not have any natural units. In a way, we are saying that one unit of knowledge is however much knowledge is worth \$1. Seen differently, our A parameters measure a combination of productivity and price. We cannot disentangle the two and do not need to for our purposes.

Optimal firm hiring and wages. The first order condition with respect to new technology (AI) analyst labor L_{it} is

$$(1-\alpha)K_{it}^{AI} - w_{L,t}L_{it} = 0, \tag{6}$$

which says that total payments to new technology analysis labor $w_{L,t}L_{it}$ are a fraction $(1-\alpha)$ of the value of knolwedge output from AI analysis, K_{it}^{AI}. The first order condition with respect to old tech analyst labor l_{it} is

$$(1-\gamma)K_{it}^{OT} - w_{l,t}l_{it} = 0. \tag{7}$$

This says that the total payments to old technology analysis labor $w_{l,t}l_{it}$ are a fraction $(1-\gamma)$ of the value of total output K_{it}^{OT}. Taking the ratio of the two first order conditions implies that

$$\frac{(1-\alpha)K_{it}^{AI}}{(1-\gamma)K_{it}^{OT}} = \frac{w_{L,t}L_{i,t}}{w_{l,t}l_{i,t}} \tag{8}$$

This ratio varies by time t and it measures how much knowledge production technology has changed. The first order condition with respect to data management labor λ_{it} is

$$\frac{1}{r}v'(D_{i(t+1)})(1-\phi)\lambda_{it}^{-\phi} = w_{\lambda,t}. \tag{9}$$

If the marginal value of data today and tomorrow are similar, we can solve for $v'(D)$ and replace $\lambda^{1-\phi}$ by the change in the data stock, to get[3]

$$\frac{(\alpha K_{it}^{AI} + \gamma K_{it}^{OT})(1-\phi)}{r-(1-\delta)} \frac{D_{i(t+1)} - (1-\delta)D_{it}}{D_{it}} - w_{\lambda,t}\lambda_{it} = 0. \tag{10}$$

Intuitively, total payments to data management $w_{\lambda,t}\lambda_{it}$ are a portion of $(\alpha K_{it}^{AI} + \gamma K_{it}^{OT})(1-\phi)$, pdv (Gordon growth), or total output times the percentage increase in the data stock.

Using these expressions for optimal labor choices, we can derive an expression for the optimal stock of data for a firm. This is an expression we will use to impute the initial data stock of each firm. We start with (10) and substitute in $\lambda_{it}^{1-\phi}$, in place of $D_{i(t+1)} - (1-\delta)D_{it}$. Next, we need to replace K_{it}^{AI} and K_{it}^{OT} which are the unobserved knowledge produced with each technology. To do this, we use the first order conditions for AI and OT labor, (6) and (7), to substitute wage per worker expressions; $K_{it}^{AI} = w_{L,t}L_{i,t}/(1-\alpha)$ and $K_{it}^{OT} = w_{l,t}l_{i,t}/(1-\gamma)$. This yields an expression that relates firm i stock of data to production function exponents and observable hiring and wages:

$$D_{it} - \frac{\left(\frac{\alpha}{1-\alpha}w_{L,t}L_{i,t} + \frac{\gamma}{(1-\gamma)}w_{l,t}l_{i,t}\right)(1-\phi)}{r-(1-\delta)} \frac{\lambda_{it}^{-\phi}}{w_{\lambda,t}} = 0. \tag{11}$$

2 Data and Estimation

Why look at the investment management industry? Our model is about knowledge production generally, in any industry. But as we turn to estimating this model, we use asset management industry labor and data estimates. One reason we do this is that the investment management industry is primarily a knowledge industry, where information is processed to form forecasts about asset returns and profitable portfolios. But the main reason is that finance is an early adopter of AI and big data technology. If we want to study the nascent adoption of this new technology, it is helpful to look in corners of the economy where adoption is most substantial. In independent studies with different methodologies, Felten et al. (2018) and Brynjolfsson et al. (2018b) both came to the conclusion that the finance/insurance industry was the one with the greatest potential for labor substitution with AI. Acemoglu et al. (2019) document that finance has the third most number of AI job postings, behind information and business services.

[3]See appendix for step-by-step derivation.

Finally, the financial industry is a useful laboratory because finance jobs are typically filled. JOLTS data tell us that finance is an industry with one of the highest vacancy conversion rates into new employment, presumably because the finance sector pays more than others. Thus, when they want a worker with a specific set of skills, they can buy them. Since our work relies on job postings, it is helpful if many of these postings are, in fact, filled.

Of course, one could argue that we could include the investment management industry, as well as all other industries, to broaden our sample and sharpen our estimates. The problem with this approach is that distinguishing which workers combine data and labor to produce knowledge is tricky. Determining which workers use which technology is even more delicate. Different industries use different vocabularies to describe this type of work. The type of work that the investment management industry calls an analyst, the retail industry might call an online marketing expert. Both are using data and labor to make predictions that will enhance their company's profit. But because the language used to describe jobs differs, one needs a separate dictionary/model to identify relevant jobs in each context. Therefore, restricting our analysis to the asset management sector allows us to obtain a cleaner sample of job postings and improve the accuracy of our estimates.

Labor demand Our data is the job postings data set collected by Burning Glass, from January 2010 through December 2018. These postings are scraped from more than 40,000 sources (e.g. job boards, employer sites, newspapers, public agencies, etc.), with a careful focus on avoiding job duplication. Acemoglu et al. (2019) show that Burning Glass data covers 60-80% of all U.S. job vacancies. The finance and technology industries have especially good coverage. It includes jobs posted in non-digital forms as well. Importantly, for a large portion of job postings, the data reports employer names, as well as the sector, job title, skill requirements, and sometimes the offered salary range. In addition to the structured data fields, we also make use of the full text of the job posting, as written by employers.

The total number of relevant job postings for the employers in our sample is 308,600, categorized as searching for old-tech financial analysts, AI financial analysts, or data managers. The unique number of employers goes from 442 in January 2015 to 739 in December 2018. The total number of unique employers is 812.

In order to construct this data set of interest we proceed in three steps. (1) We subset the data to candidate jobs of interest in the financial industry. (2) Among the candidate finance jobs, we identify the ones belonging to one of the following categories: data managemers, AI analysis or old tech analysis. (3) We compile a list of employers of interest (who operate in the investment management industry) and identify their job postings among the categorized ones. This procedure leads to the identification of 308,600 job postings categorized as AI, old tech or

data management for 812 unique employers.

In our initial step (1), we use the jobs' NAICS, O*NET and proprietary Burning Glass codes to restrict the Burning Glass data set to candidate jobs in the financial industry. More specifically, we first drop all job postings that do not belong to one of the following 2-digit NAICS codes: 'Professional, Scientific, and Technical Services', 'Finance and Insurance', 'Information' and 'Management of Companies and Enterprises'. We also keep all jobs for which the NAICS code is not available. Next we compile lists of O*NET codes and Burning Glass proprietary codes (BGT Occupation Group, BGT Career Area) of job categories that should clearly not be contained in our sample[4]. After eliminating all jobs belonging to those categories, we are left with a sample of candidate finance jobs.

In our second step (2), for all jobs in the candidate Finance sample, we then use the full text of the selected job postings in order to identify analysis jobs and data management jobs. We define 'data management' jobs as those requiring skills related to the cleaning, purchasing, structuring, storage and retrieval of data. What define as "analysis jobs" those jobs that combine structured data with skilled labor. We call these analysts because they analyze data in different ways. They are not necessarily what the financial industry calls analysts. Within the analysis jobs we further distinguish between those that mostly require old (Old Technology - OT) or new (Artificial Intelligence - AI) skills.

This classification is obtained by developing a dictionary of words and short phrases that indicate 'data management' or 'data analysis', and then counting the relative frequency of these words or expressions in each pre-processed job text.[5] Among the 'data analysis' keywords we further identify those clearly indicative of the old and new technologies and we assign jobs to 'Old Tech - OT' or 'Artificial Intelligence - AI' depending on the relative frequency of words of the two types present in the posting. The full dictionaries used are available in Appendix A.2.

While this last step is similar in nature to the decompositions by Acemoglu and Restrepo (2018) or Babina et al. (2020), working with one type of job in a single industry allows us to partition the data more precisely. The approach of these authors is to define a dictionary of big data related words in all industries. They then identify job postings that contain those words in the standardized skills list provided by Burning Glass. Those are categorized as AI jobs; everything else is non-AI. This approach does not work for our exercise: Burning Glass' skills list is not detailed enough to distinguish between different types of data analysis in finance. Misclassification that might wash out in a job counting exercise is more serious for us. We need

[4]Examples of excluded 6-digit O*NET codes that were still present in the sample: 'Bookkeeping', 'Accounting, and Auditing Clerks', 'Customer Service Representatives', 'Cashiers', 'Retail Salespersons' ...

[5]We pre-process the text of each job posting by first removing symbols, numbers and stop-words (e.g. is, the, and, etc.) and then stemming each word to its root using the Porter stemmer algorithm (thus, e.g. 'mathematic', 'mathematics', ... = 'mathemat').

to match data and labor stocks firm-by-firm. This is why we analyze the full text of the job posting. Analyzing the full text, rather than using the Burning Glass skills list, greatly improves our classification by allowing us to account for the frequency of mentions of each type of skill.[6]

In our third step (3), to match the categorized job postings to the right employers, we use a "master" list of investment management firms and identify among the categorized job posting those that most likely belong to the employers of interest, through fuzzy matching of employer names. Appendix A.3 provides a detailed description of that process. This procedure allows to map all relevant job postings to the employers of interest, so identifying their labor demand. We further restrict the sample to employers that posted at least 5 'Old Technology' or 'Machine Learning' jobs throughout the entire period of interest (2010-2018).

There is lots of entry in our data set. 58% of firms are in our data set in 2015. The remaining 42% appear for the first time in 2016-2018. That does not mean these 42% are all new firms. Instead, many of them are existing firms that enter our data set when they hire data workers for the first time.

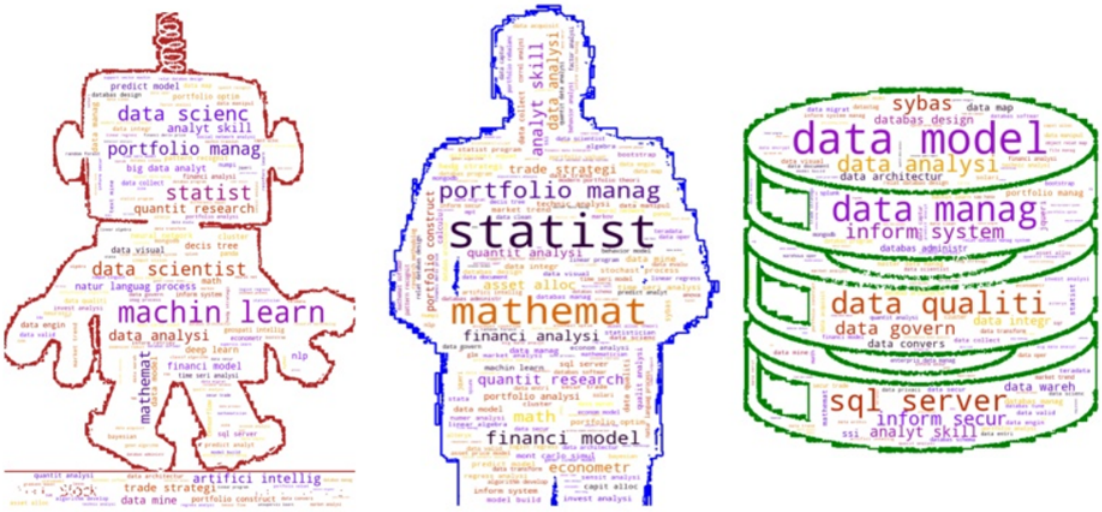

Figure 2: Keywords in the full text of the categorized machine learning, old technology and data management jobs. Larger fonts indicate a higher word frequency. Burning glass job postings, 2010-2018.

Figure 2 illustrates the frequency of all keywords in the job postings categorized as belonging to each type. Note that even if all 'data analysis' and 'data management' keywords are included in all three word clouds, the keywords specific to the assigned category have a significantly higher relevance. The word overlap illustrates why counting word frequency is important. At

[6]For instance a job that mentions 'Machine Learning' 10 times withing the job text and then also states "Masters in Statistics also accepted", in our approach would be clearly classified in the 'AI' category. Looking at the skills lists, instead, the categorization of the job would be ambiguous as it would appear to require both old and new technology skills in the same proportion: 'Statistics' and 'Machine Learning'.

the same time, the significat differences between the word clouds validates our approach. If a clear distinction between the three types of job postings did not exist we would observe that the most frequently mentioned words in each category would be less distinct.

Sample job postings To provide a clear idea of how this methdology classifies jobs, we list three sample job postings here, one each of old technology, AI or big data-related skills, and data warehousing. In this example, all three jobs are posted by the firm Two Sigma. The text of the first job reads:

> "We are looking for world-class quantitative modelers to join our highly motivated team. Quant candidates will have exceptional quantitative skills as well as programming skills, and will write production quality, high reliability, highly-tuned numerical code. Candidates should have: a bachelor's degree in mathematics and/or computer science from a top university; an advanced degree in hard science, computer science, or the equivalent (a field where strong math and statistics skills are necessary); 2 or more years of professional programming experience in Java and C, preferably in the financial sector; strong numerical programming skills; strong knowledge of computational numerical algorithms, linear algebra and statistical methods; and experience working with large data sets. (...) "

This job is classified as old tech because it uses words such as "mathemat" (x1), "math" (x1), "statist" (x2), "algebra" (x1), and does not contain words related to AI or data management skills.

This first posting contrasts with the text of the second job, which reads:

> "As machine learning and data-driven business intelligence have permeated industries, an abundance of new datasets and techniques have created opportunities for granular measurement of increasingly varied aspects of our economy. Two Sigma is looking to hire a highly creative & motivated Lead Data Scientist to further scale our long-standing efforts to leverage these advancements to measure and predict the world's financial outcomes.
>
> Two Sigma's data engineering platform enables us to harness some of the world's most complex & challenging content, as we structure and integrate new datasets into a diverse ecosystem of syndicated financial and industry-specific data products. *Two Sigma's data scientists are focused on joining, enriching, and transforming datasets into novel creative measures of economic activity.* (...) "

This job is harder to classify. It contains the word "statist" (x2), indicative of old tech. It also contains data-management-related vocabulary, "data engin" (x1), "data sourc" (x1), and

"support data" (x1). But what ultimately gets this job classified as AI is the higher frequency of AI-related words: "data scienc" (x4), "data scientist" (x5), "machin learn" (x1). An algorithm that just looked for the presence of skills or words, without measuring their frequency, would likely misclassify this job, and many others like it.

Finally, the text of the third job reads:

> " (...) Technology drives our business it's our main competitive advantage and as a result, software engineers play a pivotal role. They tackle the hardest problems through analysis, experimentation, design, and elegant implementation. Software engineers at Two Sigma build what the organization needs to explore data's possibilities and act on our findings to mine the past and attempt to predict the future. We create the tools at scale to enable vast data analysis; the technology we build enables us to engage in conversation with the data, and search for knowledge and insight. (...) You will be responsible for the following: *Capturing and processing massive amounts of data for thousands of different tradable instruments*, including stocks, bonds, futures, contracts, commodities, and more; (...) "

This job is classified as data management because of the words, "explor data possibl," "enabl vast data analysis," "data specialist," and "data team."

Wages Many, but not all jobs in Burning Glass list a salary range. Because listed salaries are not representative, we obtained salary data from PayScale.[7]

Figure 3 shows the average wage for data managers, old technology analysts and machine learning analysts, in each year. The key insight is that AI jobs consistently pay more – around $20,000 more per year – than traditional analyst jobs. This suggests that AI workers make more productive use of their data. This difference in wages is a key input that determines the difference in production function estimates.

Cumulating hiring to get labor. The data series we need in order to estimate production is the labor force working in a given month, for both knowledge and data processing workers. We do not observe the stock of labor. Therefore, we use the following procedure to estimate labor from observed job postings by firm. The number of observed job postings for the three categories of interest is displayed in Figure 4, together with the number of employers hiring in each category.

Job postings are not the same as net hiring. One might be concerned that AI workers, in particular, are so scarce that many postings go unfilled and/or that workers jump from job-to-job. There are two key differences between postings and net hiring: the probability that

[7] www.payscale.com. Data last updated December 2020.

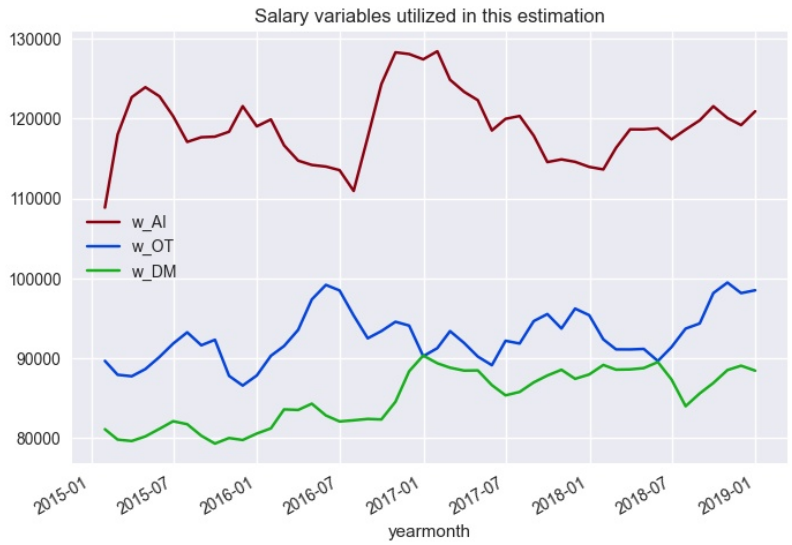

Figure 3: Distribution of wages for data managers, old technology analysts and machine learning analysts. Job postings from Burning Glass matched to wage data from PayScale.

a vacancy is filled and the probability that an employed worker separates from their job. We adjust for both of these using data on vacancy fill rates and job separation rates from the Bureau of Labor Statistics (BLS).

Each month, the BLS reports the job posting, job filling and separation rate for each occupation. The three occupation brackets present in the final sample are: 'Finance and Insurance', 'Professional, Scientific and Technical Services' and 'Information'. Since we want to map our job postings into expected hires, we multiply each job posting number by the fraction of job postings that results in a new hire (h).

Of course, machine learning jobs are not an occupation. We need a way to map our technology-based job classification into the BLS occupation classification. Fortunately, each Burning Glass job posting has a listed occupation. Of course, different postings have different classifications, even within machine learning, old technology or data management jobs. Thus, we measure the proportion of jobs in each of our samples that belongs to each occupation. Each month we compute a vector of occupation weights for machine learning jobs, one for old tech jobs and one for data management jobs that is the fraction of jobs in each category that belongs to each occupation. We multiply this weight vector by each of the fill and separation rates that month, to get the imputed fill and separation rates for machine-learning financial analysis jobs (h_t^{AI} and s_t^{AI}), the imputed fill and separation rates for old technology financial analysis jobs (h_t^{OT} and s_t^{OT}) and those for data management jobs (h_t^{DM} and s_t^{DM}). See Appendix A.4 for more detail on how BLS data is mapped into our job categories and how h and s are derived from BLS reported rates.

Figure 4: Job postings and Labor Stocks: Panel 1 shows the fraction of employers hiring in each category. Panel 2 shows the stock of labor in each category, measured as a cumulated number of job postings, adjusting for filling and separation rates as in (12).

For $type = [AI, OT, DM]$, if s_t^{type} are separation rates by type-month, and h_t^{type} are the fraction of posted vacancies filled by type-month and j_t^{type} are Burning Glass job postings rates by type-month, we cumulate labor flows into stocks as follows:

$$L_{it} = (1 - s_t^{AI})L_{i(t-1)} + j_{it}^{AI} h_t^{AI}, \tag{12}$$

$$l_{it} = (1 - s_t^{OT})l_{i(t-1)} + j_{it}^{OT} h_t^{OT}, \tag{13}$$

$$\lambda_{it} = (1 - s_t^{DM})\lambda_{i(t-1)} + j_{it}^{DM} h_t^{DM}. \tag{14}$$

To use this cumulative approach, we need to know the initial number of workers of each type (L_{i0}, l_{i0} and λ_{i0}). Unfortunately, that information is not available, but we know that the initial number of workers becomes less relevant the further we are from initialization. For this reason we start the initialization from zero for all job types and we use the first 5 years of data $[2010 - 2014]$ as a burn-in period. We then use the last 4 years $[2015 - 2018]$ for the structural estimation of the model's parameters.[8]

The right panel of Figure 4 shows the imputed labor stocks for each job type. AI workers in finance are still a small fraction of the overall labor supply, suggesting that the transition to a new model of knowledge production is just in its beginnings. However, what looks like a small uptick on this axis looks like an explosion when we zoom in. Prior to 2015, hiring in AI is mostly flat. From 2015 to 2019, the stock of AI labor increases about 8-fold.

[8] Incorporation of 2019-2020 Burning Glass data is in process.

	Data Management λ_{it}	AI analysts L_{it}^{AI}	Traditional analysts L_{it}^{OT}
mean	53.73	2.33	32.37
stdev	441.17	29.78	204.84
minimum	0	0	0
median	5.72	0	3.63
maximum	11409.26	1765.88	4420.53
Observations	33,392	33,392	33,392

Table 1: **Labor Stock Summary Statistics**.

Table 1 reports the summary statistics for the stock of each type of labor. What is salient in all three categories is the large disperion. This is helpful because the cross-firm heterogeneity will allow us to estimate the technology parameters. Our final data set contains $33,392$ employer-month observatiobs. These will be used in the structural estimation of the model's parameters.

Cumulating data management to get structured data stocks We measure each firm's stock of data in each period by adding the data management inputs to the depreciated stock of yesterday's data:

$$D_{it} = (1-\delta)^t D_{i0} + \sum_{s=0}^{t} (1-\delta)^{t-s} \lambda_{is}^{1-\phi}. \tag{15}$$

We fix the depreciation rate of data at $\delta = 0.0.25$, which is a 2.5% depreciation rate per month. We also report results for 1% and 10% deprecation. This represents some high-frequency data, whose value lasts for fractions of a second, as well as longer term data used to value companies. In future iterations, we will experiment with other values for depreciation.

To use this approach, we need information about firms' initial data stocks. We estimate this initial stock, by finding the initial stock that makes all subsequent data levels closest to the firm's optimal level. Specifically, the initial data stock of each firm is the D_{i0} that best fits the sequence of the firm's data optimality condition (11).

If we estimate this recursive system of data stocks, production parameters and data inputs for every firm in our sample, the problem quickly becomes unmanageable. At the same time, we do not want to lose the interesting cross-firm heterogeneity. Therefore, instead of estimating D_{it} for each firm in our sample, we compute it for the average firm and use a rule to map the average into a firm's initial data. We use the initial data stocks to estimate the production function parameters. Then, given the parameters, we can recover the best-fit initial data and cumulate up a data stock for each firm easily.

Specifically we express the D_{i0} of each firm as a function of a unique average data stock by

setting each firm's initial data proportional to the average data stock and to their cumulative hiring in data management from 2010-2015. In other words, we take the estimated data management labor stock in 2015, $\lambda_{2015,i}$ and raise it to the production function exponent to turn it into an amount of data produced: $\lambda_{2015,i}^{1-\phi}$, and then choose a constant ι so that the average initial data stock is the estimated average stock: $(1/N)\sum_i \iota\lambda_{2015,i}^{1-\phi} = \bar{D}_0$.

Then we can express equation 15 as follows:

$$D_{it} = (1-\delta)^t \iota \lambda_{2015,i}^{1-\phi} + \sum_{s=0}^{t} (1-\delta)^{t-s} \lambda_{is}^{1-\phi}. \tag{16}$$

where ι is a function of $\bar{D}_0$. For each firm we then cumulate up the data management flows to construct a stock of data.

The initial data stock that best explains the sequence of data management hiring is the $\bar{D}_0$ that minimizes the sum of squared errors or the right hand side of (11), for each firm i.

Data depreciation The rate of data depreciation depends on the time-series properties of the variable being forecasted, as well as on the nature of data management. If data is being used to foreacst firm earnings, for example, and firm earnings are quite stable over time, with high persistence and small innovations, then data from a few months ago is still quite useful for predicting today's earnings. Because interest rates are even more persistent and less volatil, data used for forecasting interest rates would retain its value even more. However, if data is being used to forecast order flow, which has a persistence of only a few days, then order flow data from a month ago is nearly worthless. Customer data, with fixed customer characteristics, might not depreciate at all. Firms' data sets are a mixture of these different types of data. To get some sense of a reasonable depreciation rate, we base our first depreciation estimate on the properties of earnings data, because earnings lies in between the extremes of highly transitory and highly persistent data.

From Farboodi and Veldkamp (2019), we know that the data depreciation rate is $1-\rho^2(\rho^2+\sigma_\theta^2 D_{it})^{-2}$, where ρ is the persistence of the AR(1) process for earnings and σ_θ^2 is the variance of its innovations. Farboodi et al. (2020) report these coefficients for averge small value, large value, small growth and large growth equities. For amounts of data that increase earnings forecast precision between 0 and 10 times their initial precision, we find depreciation rates that range from 58% to 91% annually, for all four types of assets. In our monthly model, that is the equivalent of 5-7.5% per month.

However, what matters for the structural estimate of data is not really how much data depreciates, but how much the output of data management labor depreciates. If what data workers do is to collect each data point, one at a time, and add them to the data set, then

depreciation of data is the relevant depreciation rate. But data workers would never hand-collect a stream of data like this. They automate the collection of a particular type of data. Each month, each day or each microsecond, their system automatically pulls the next piece of data. That matters because a unit of data management labor now doesn't depreciate just because the data series is not persistent. In this view of data management, depreciation is hardware breaking, data links changing, or software needing updates. That type of depreciation sounds very much like the standard capital depreciation of macro theory. Typical estimates of 12% per year (1% per month) might then seem to be more appropriate. Standard accounting practice is to amortize data warehouses like software, over 36 months. That translates to a depreciation rate of 3% per month.

Given this range of estimates, we explore depreciation rates of 1%, 3% and 10% monthly, with the understanding that rates around 1-3% more accurately reflect the automated way in which data is collected.

Estimating production functions The key variables of interest are the two production function exponents, α and γ from (1) and (2). There are four variables we need to estimate: α, γ, the exponent ϕ on data management in the structured data production function (3), and finally, we need the initial average data stock $\bar{D}_0$. For three of our moment conditions, we use the first order conditions for each of the three types of labor (6), (7) and (10), for the fourth, we use the optimal data stock condition, (11).

When we estimate the machine learning labor first order condition, we use only firms that employ some machine learning workers and some data management workers. Requiring that the firm currently employs a type of worker does not imply they hired someone that month. Rather, it means that some worker was hired at some time in the past. If we do not exclude these firms, our production exponent estimate would be heavily influenced by the many observations with zero labor and abundant data, or vice-versa. Similarly, when we estimate the traditional labor first order condition, we use only observations from firms that have, at some point, hired a data manager and a traditional analyst.

We also need to solve for the productivity parameters A_t^{AI} and A_t^{OT}. Given a set of guessed parameters (α, γ, ϕ and $\bar{D}_0$), we solve for A_t^{AI}, A_t^{OT} using the first order conditions 6 and 7 computed on cross-sectional averages. In other words, the A parameters reconcile the average magnitudes of knowledge with average wages, while the production exponents are identified off of the cross-firm heterogeneity.

We then substitute the computed productivity parameters into the four conditions and compute a vector of residual using the full time-series and cross-sectional variation. The residual vector contains $(33,392 \times 4)$ observations.

Finally we use non-linear least squares to iterate over different combinations of α, γ, ϕ and $\bar{D}_0$. The algorithm converges when it finds the combination of parameters that yields the smallest sum of squared errors.

As a check on convergence, we also re-estimate the parameters using a grid search method. This is viable because many of our parameters, like the production exponents are bounded between zero and one. While it takes longer to run, our grid search does identify the same solution.

3 Results

The results are broken into four parts. The first part is the main results, with our estimates of the production function parameters. Our baseline results reveal that the size of the change in knowledge production is about two-thirds the size of the industrial revolution in goods production. The second part explores why we come to this conclusion. It explains why the data-labor ratios for firms that do lots of AI and those that do not are key statistics for identifying production function exponents. Third, we relate our findings to a literature on labor-replacing technological change. We document that labor demand in this sector is rising with technology adoption. That may not be causal. But there is no evidence of technology crowding workers out. Finally, we use our structural model to value the data that financial analysis firms are accumulating.

3.1 Main Result: Comparing Changes in Knowledge Production to Industrialization

Our main question is: What are the production function exponents from each technology? Table 2 reports these main results. The exponents α and γ represent the diminishing returns to data in the old and new technologies. The fact that $\alpha > \gamma$ means that the rate of diminishing returns to data is less with the new AI technology. In other words, new data technology has significantly raised the productivity of analyzing larger data sets. That is not surprising. The fact that the exponent rose by 0.08, which is 11% of its previous value, suggests that the rise is substantial. With standard errors one hundredth of that size, the change is statistically significant, at any reasonable threshold.

The other parameter we estimate is the average initial data stock, which is $(1425, 808, 226)$ for $\delta = (0.01, 0.03, 0.1)$. From here on, we present results for the medium depreciation case of $\delta = 3\%$ and report results for the other two cases in the appendix.

The labor first order conditions (6) and (7) tell us that these exponents also govern the distribution of income to factor owners. Our results imply that owners of data have gained

		$\delta = 1\%$	$\delta = 3\%$	$\delta = 10\%$
AI Analysis	α	0.894	0.791	0.702
		(0.0005)	(0.0009)	(0.0013)
Old Technology Analysis	γ	0.634	0.711	0.678
		(0.0017)	(0.0007)	(0.0004)
Data Management	ϕ	0.152	0.147	0.142
		(0.0012)	(0.0008)	(0.0008)

Table 2: **Main Result: AI Reduced the Share of Knowledge Income Paid to Labor ($\alpha > \gamma$).** The estimates are for the exponents on data in the knowledge production functions in (1) and (2) and the production of structured data in (3). Data covers 2015-19 from PayScale and Burning Glass. Standard errors in parentheses.

enormously from this technological change. While they used to be paid 71% of the value of the knowledge output, they can now extract 79% of that value. In addition, since more knowledge is being produced, this is 79% of a larger revenue number. This finding is consistent with the overall economic trend of a decrease in the labor share of income (Karabarbounis and Neiman, 2017).

Of course, owners of data had to pay data managers to build these data sets, just like owners of capital had to pay for the investment in their capital stocks. But once they own these data stocks, they get the income associated with their factor.

How can we gauge the size of this change in knowledge production? Since this paper is pursuing an analogy between knowledge production with big data technologies and the change in physical production in the industrial revolution, a historical comparison seems most relevant. Klein and Kosobud (1961) estimate that between 1900 and 1920, the labor share of income fell from 0.909 to 0.787. Since the labor share of income corresponds to one minus the exponent on capital in the production function, this estimate suggests that the capital exponent in the production function rose by 0.122. Our rise of 0.080 is about two-thirds of the industrial revolution value. That simple comparisons suggests that the magnitude of the technological change in the big data revolution is somewhat smaller, but still comparable to that of the industrial revolution.

The data depreciation rate matters for this conclusion. If data management is mostly maintenance of physical infrastructure and thus depreciates like physical capital, at a rate around 12% per year or 1% per month, then the effect of AI is twice as large as the industrial revolution. When assuming a very high depreciation rate of data management (10% monthly) we still obtain a decrease of 0.024, which represents a fifth of the industrial revolution value.

What data features identify production parameters? Our exponents are estimated, not calibrated to a particular features of the data. So all data features matter. However, some are particularly informative. One feature of the data that is particularly helpful to identify production function exponents is the data-labor ratio and how it covaries with AI analysis.

Just like the shift to industrialization has been characterized as a shift to a more capital-intensive form of production, our estimates suggest that AI is analogous – a more data-intensive form of knowledge production. Instead of more machines per person, this shows up as more data per analysis worker.

We can use our structural model to explain the relationship between production exponents and data-labor ratios. Consider a firm that produces using only old technology analysis. The analysis labor first order condition for this firm is $(1-\gamma)K_{it} = w_{lt}l_{it}$. Now imagine an economy where a firm would just rent data for one period, at rate r_D, and have it automatically cleaned and integrated in its data repository. The data first order condition would then be $\gamma K_{it} = r_D D_{it}$. One could then divide the optimal data condition by the optimal labor condition to get $D_{it}/l_{it} = \gamma/(1-\gamma)(w_l/w_D)$. Similarly, for a firm that rented data and produced only with the AI technology, the optimal data-labor ratio would be $D_{it}/L_{it} = \alpha/(1-\alpha)(w_L/w_D)$. In this simplified model, it is clear that a higher data-labor ratio in AI firms, after adjusting for wage differences, would reveal how much larger the production exponent α is than γ.

The model we wrote down differs because data is produced with data management labor, is a long-lived asset and can be used for both AI and OT analysis. The data first-order condition in our richer model is (11). If a firm did only old tech analysis ($A^{AI} = w_l = 0$), then this condition would reveal and optimal data-labor ratio for a pure OT firm:

$$D_{it}/l_{it} = \gamma/(1-\gamma) \cdot (w_l/w_\lambda) \cdot (1-\phi)\lambda^{-\phi}/(r+\delta-1).$$

For a pure AI firm, the optimal data-labor ratio is

$$D_{it}/L_{it} = \alpha/(1-\alpha) \cdot (w_L/w_\lambda) \cdot (1-\phi)\lambda^{-\phi}/(r+\delta-1).$$

Just as before, if the AI firm has a higher data-labor ratio than the OT firm, after correcting for wages and λ, it tells us that the exponent $\alpha > \gamma$.

Our estimation is more complex because it considers firms operating both technologies at the same time, with different intensities. But this math illustrates why data-labor ratios are particularly informative about the exponent that governs diminishing returns and income shares. Thus, it is the heterogeneity in the firms' cross-section of data-labor ratios, ratio of AI to OT analysts, existing data management labor (λ_{it}) and wage rates that most clearly identify the production parameters.

Figure 5 plots the distribution of data and the distribution of data-labor ratios for firms that do AI analysis and firms that do not. The threshold for doing AI analysis is hiring more than the median number of AI workers.

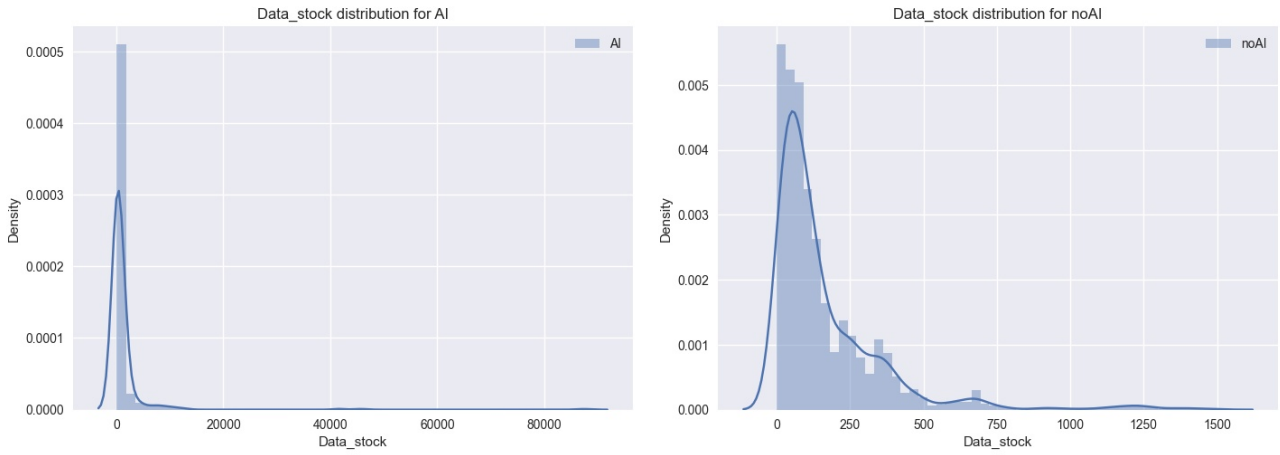

Figure 5: AI firms have larger data sets. AI firms are defined as those that hire more than the median number of AI workers, in total over all months of observations. The left panel excludes Goldman Sachs, JP Morgan and BoA simply because their data is an order of magnitude larger than others. Excluding them makes the rest of the data set more visible. Source: Burning Glass, 2018 data.

Figure 5 illustrates the enormous heterogeneity in firms' data stocks. In particular, there is a right tail of firms with troves of data. We also see a systematic difference between the data stocks of firms that use AI and those that do not. This does not imply a direction of causality. In our structural model, abundant data creates an incentive for firms to pay more for AI-skilled analysts. Having AI-skilled analysts also motivates a firm to acquire more data. Both of these forces are embodied in our optimality conditions and both inform our estimation. This bi-directional causality shows why a structural estimation is essential for our purposes.

Of course, data stocks are something we impute from firms' hiring choices. Since this relationship is crucial for our estimates, one might want to see what features of the underlying data inform it. Evidence of the relationship between AI, OT and data-labor ratios shows up in the cumulated hiring decisions of firms. The firms we estimate to have large data sets are firms that hire more data management workers. This is supported by the fact that such firms also hire more analysts to work with their data. This is apparent in Figure 6 where both plots show an upward slope, a positive relationship between analyst and data manager hiring.

Even more importantly, we see that the slope of the relationship between data and analyst labor is steeper for AI analysts than for OT analysts. This is a feature of the underlying hiring data that informs us about how much more data intensive AI-based knowledge production is becoming.

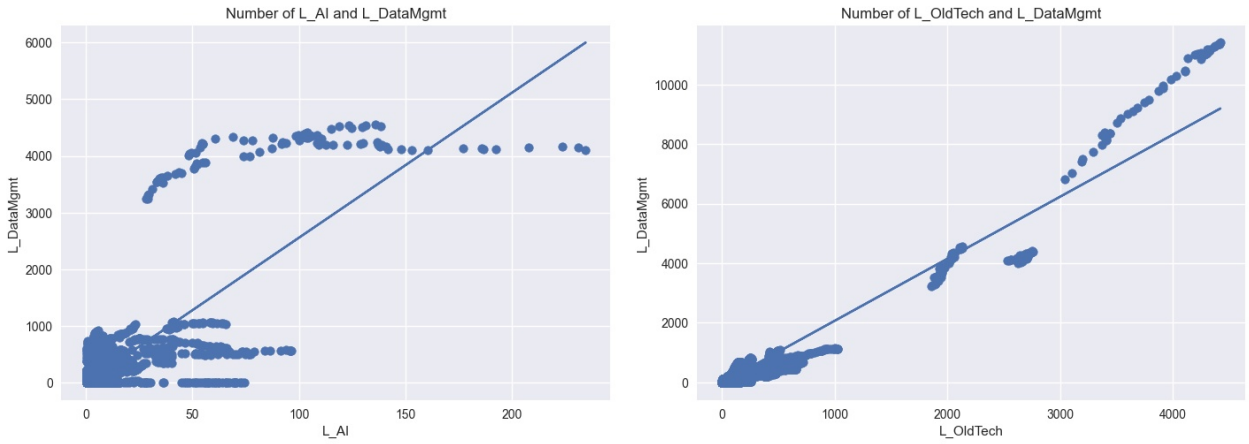

Figure 6: Firms with more structured data hire more AI analysts (left panel) and more traditional analysts (right panel). The left panel excludes Goldman Sachs simply because their hiring is an order of magnitude larger than others. Excluding them makes the rest of the data set more visible. Source: Burning Glass, 2015-2018.

One might think that the time-series of data is more informative, since more and more firms hire AI analysts over time. But the time series of the data distribution quantiles are remarkably stable. Figure 7 illustrates the evolution of the data stock of firms in each percentile of the cross-firm distribution. Surprisingly, the lines are not increasing. What is going on here is two-fold. First, the sample of firms is growing over time. Many firms are starting to hire data workers, and thus entering our sample. As a result, the top decile of firms has a lot more firms in it at the end and the firm at the 90th percentile is much lower in the rankings. Second, much of the data accumulation is happening at the top of the distribution. The top 1% of firms is not reflected. The 90th percentile is not an average of the top 10% of firms. It is the stock of the single firm at the 90th percentile. The take-away is that, while the aggregate stock of data is growing rapidly, the most informative moments for production come from the cross-sectional heterogeneity in firms' data and labor.

3.2 Is Data Replacing Labor?

One of the main concerns people have with new data technologies like AI is that they might be labor replacing. Our results show how even labor-replacing technologies can expand labor demand.

Figure 4 illustrates the aggregate stock of analysis labor. Despite our finding that knowledge production is becoming less labor-intensive, we see that there are more and more workers doing analysis. Production can be less labor-intensive and still have more labor demand because production of knowledge is rising. The expansion is made possible by the improvement in

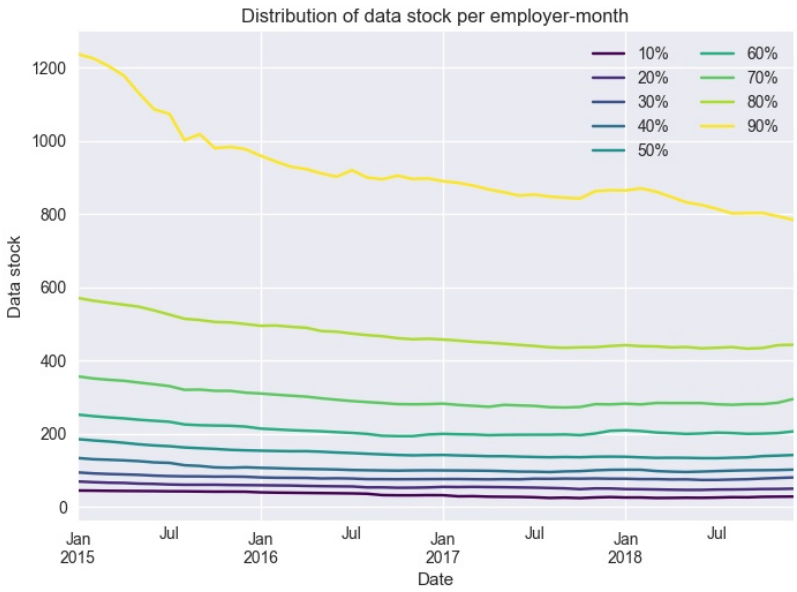

Figure 7: Estimated Stock of Data, Across Firms ($\delta = 2.5\%$), 2015-2018.

analysis productivity. So even though AI is labor-replacing, in the sense that it requires less labor per unit of data, it is also labor-demand-enhancing because it causes the whole sector to grow.

The growing labor force is not an artifact of our parameter estimates. It is also not dependent on most model assumptions. The growing labor result comes from simply counting up the new hires and adjusting for BLS-reported departures. Much of this increase comes from there being more firms in our sample. But the growth of firms working with financial data is hardly a sign of low labor demand.

Both old tech and AI-skilled analysts become more abundant. AI jobs grew at a faster rate (from about 0 to 2000). However, they account for only about half of the increase. The other half comes from more hiring of old technology analysts. While old technology productivity may not have improved much, these workers are made more productive by the abundance of structured data.

3.3 Estimating the Value of Data

One of the big questions in economics and finance today is how to value firms' data stocks. Four of the five largest firms in the U.S. economy, by market capitalization, have valuations that are well beyond the value that their physical assets might plausibly justify. These firms have future expected revenues based on their accumulated stocks of data. Our structural estimation offers

a straightforward way to compute this value.

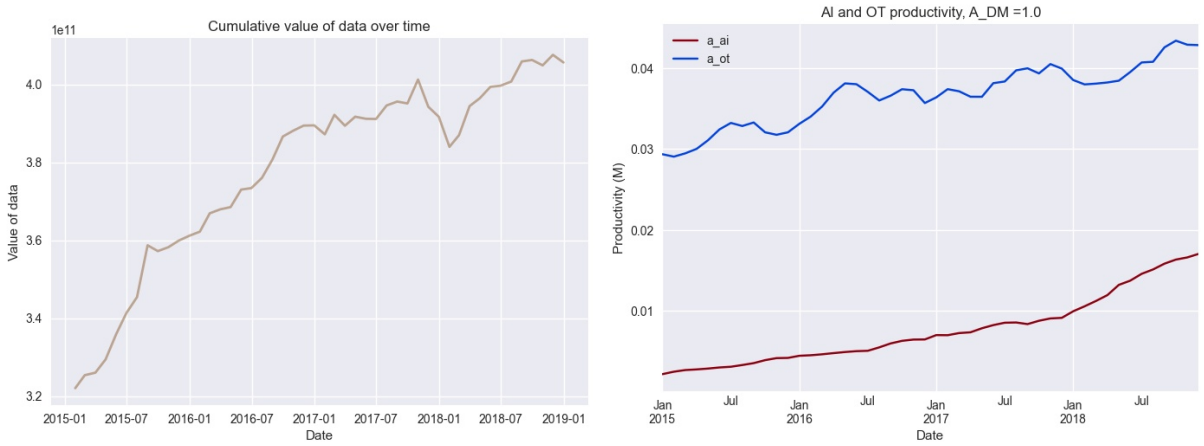

Figure 8: Estimated Value of the Aggregate Stock of Data, in billions of current U.S. dollars, and the Productivity of Financial Analysts, 2015-2018. Productivity is the estimated values of A^{AI} and A^{OT} for AI and old technology analysts, as defined in equations (1) and (2).

Once we have estimated production parameters and data stocks, we can put them back into our value function, and approximate the value of each firm's stock of data in each month. This value is in nominal dollar units, since those are the units of the wages we use. Figure 8 plots this aggregate value. This is our estimate of the value function in (4) for the aggregate stock of data. These results are presented with an important caveat: The wage data we have is sparse. Therefore, it is incredibly volatile. Since the value of data depends very much on the wages of the workers who work with it, results might change once we repeat the estimation using better wages data, which we are in the process of acquiring.

The units of Figure 8 are tens of billions of U.S. dollars. Over the time period, 2015-2018, we see a rise in the value of this data stock from about $ 18 billion to about $ 24 billion, a 33% increase in value.

Where does this increase in value come from? The first source is simply the accumulation of data. The aggregate stock of data rose just over 50%. More than half of the increase in the value of data comes from this rise in the size of the structured data stock. A second contributor to the increase in the value of data is the increase in financial analysts that work with data (Figure 4). The more workers there are, the higher is the marginal value of data and the more valuable the stock of data is.

Finally, firms are becoming more productive at using data. More productivity also contributes to the rise in the value of data. The right panel of Figure 8 reports our estimates of the analysis productivity parameters, A^{AI} and A^{OT}, for each month. While productivity with the old technology show no trend over time, the productivity of working with the new

(AI) data technologies displays a clear jump in 2017. This productivity jump is additional evidence of the transformative power of new big data technologies.

4 Conclusion

Modern discourse describes new big data technologies as the next industrial revolution, or more specifically, as the industrialization of knowledge production. What does that mean? Industrialization was the adoption of new production technologies that involved less human input and less diminishing returns to capital. In other words, the key feature of industrialization is that factor shares changed. Thus if big data technologies are the industrialization of knowledge production, they should offer less diminishing returns to data.

We explored this hypothesis by modeling the production of knowledge, in the same why economists model industrial production. Instead of mixing capital and labor with a Cobb-Douglas production function to produce goods, we described how labor and data can be mixed with a Cobb-Douglas production function to produce knowledge. Then, just as 20th-century economists estimated the exponents of the industrial production function using labor income shares, we similarly measure the exponents of the knowledge production function using wages and labor flows in a particular type of knowledge production, financial analysis. We find a substantial change in the production function, of magnitude larger than the change due to industrialization. Thus, describing this change as a new industrialization seems to be a fair comparison.

Adoption of AI and big data technologies, as well as the accumulation of stocks of data vary widely by firm. The firms with more data are more prone to hire more big-data or AI workers. This supports the idea that this is a technology that is changing the factor mix of production. This finding has important implications for the future of the income distribution: It changes the future labor share of income. In a model that did not have constant returns to scale, such a change would alter the optimal size of a firm: Firms with less diminishing returns to data may well take on a larger optimal size. It also tells us that knowledge will be significantly more abundant going forward.

Two extensions of the model would be useful next steps. One would be to relax the assumption of constant returns to scale in knowledge production. It is possible that doubling data and doubling data workers more than increases the production of knowledge. It is also possible that there is a form of knowledge crowd-out, where it gets harder and harder to produce new knowledge (Bernard and Jones, 1996). We use constant returns because it facilitates a comparison with industrialization, which typically used such production functions. Constant returns also yields a clear mapping from labor shares to production function

exponents. In the absence of constant returns, there is considerable dispute about the best way to determine market wages or factor shares. Getting caught up in that debate would distract from the simple main message of this paper.

Another extension would be to consider market power. Owners of data extract rents because data is not perfectly substitutable. Knowledge producing firms also produce differentiated products that allow them to profit. Market power does interact with equilibrium wages. Correcting for it would complicate the mathematics of the model, but could also sharpen the production function estimates.

Of course, this estimation was for workers doing one type of work in one sector. In other sectors, big data might be more or less of a change to output. It may also be too early to tell since machine learning is not widely adopted in most other sectors. Much work in this area remains to be done to understand the magnitude and consequences of the technological changes in data processing that we are currently experiencing.

References

Acemoglu, Daron and Pascual Restrepo, "Artificial Intelligence, Automation and Work," Working Paper 24196, National Bureau of Economic Research January 2018.

_ **and Veronica Guerrieri**, "Capital Deepening and Nonbalanced Economic Growth," *Journal of Political Economy*, 2008, *116*, 467–498.

_ **, David Autor, and Jonathon Hazell**, "AI and Jobs: Evidence from Online Vacancies," Working Paper, Massachusetts Institute of Technology October 2019.

Aghion, Philippe, Benjamin F. Jones, and Charles I. Jones, "Artificial Intelligence and Economic Growth," 2017. Stanford GSB Working Paper.

Agrawal, Ajay, John McHale, and Alexander Oettl, "Finding Needles in Haystacks: Artificial Intelligence and Recombinant Growth," in "The Economics of Artificial Intelligence: An Agenda," National Bureau of Economic Research, Inc, 2018.

_ **, Joshua Gans, and Avi Goldfarb**, *What to expect from artificial intelligence*, MIT Sloan Management Review, 2017.

_ **,** _ **, and** _ , "The economics of artificial intelligence," *McKinsey quarterly*, 2018.

Alekseeva, Liudmila, José Azar, Mireia Gine, Sampsa Samila, and Bledi Taska, "The Demand for AI Skills in the Labor Market," 2020.

Babina, Tania, Anastassia Fedyk, Alex He, and James Hodson, "How Does Artificial Intelligence Affect Jobs? Evidence from US Firms and Labor Markets," Technical Report, Working Paper 2020.

Berg, Andrew, Edward F Buffie, and Luis-Felipe Zanna, "Should we fear the robot revolution?(The correct answer is yes)," *Journal of Monetary Economics*, 2018, *97*, 117–148.

Bernard, Andrew B and Charles I Jones, "Comparing apples to oranges: productivity convergence and measurement across industries and countries," *The American Economic Review*, 1996, pp. 1216–1238.

Brynjolfsson, Erik, Daniel Rock, and Chad Syverson, "Artificial intelligence and the modern productivity paradox: A clash of expectations and statistics," Technical Report, National Bureau of Economic Research 2017.

_ **, Tom Mitchell, and Daniel Rock**, "What can machines learn, and what does it mean for occupations and the economy?," in "AEA Papers and Proceedings," Vol. 108 2018, pp. 43–47.

_ , _ , **and** _ , "What can machines learn, and what does it mean for occupations and the economy?," in "AEA Papers and Proceedings," Vol. 108 2018, pp. 43–47.

Buera, Francisco and Joseph Kaboski, "Can Traditional Theories of Structural Change Fit the Data?," *Journal of the European Economic Association*, 2009, *7*, 469–77.

Cheremukhin, Anton, Mikhail Golosov, Sergei Guriev, and Aleh Tsyvinski, "The Industrialization and Economic Development of Russia through the Lens of a Neoclassical Growth Model," *Review of Economic Studies*, 2017, *84* (2), 613–49.

Cockburn, Iain M, Rebecca Henderson, and Scott Stern, "The impact of artificial intelligence on innovation," Technical Report, National bureau of economic research 2018.

Crouzet, Nicolas and Janice Eberly, "Rents and Intangible Capital: A Q+ Framework," 2020. Northwestern University Working Paper.

Deming, David J and Kadeem L Noray, "Stem careers and the changing skill requirements of work," Technical Report, National Bureau of Economic Research 2018.

Farboodi, Maryam, Adrien Matray, Laura Veldkamp, and Venky Venkateswaran, "Where Has All the Data Gone?," 2020. Working Paper.

_ **and Laura Veldkamp**, "A Growth Model of the Data Economy," 2019. Working Paper, MIT.

Felten, Edward W, Manav Raj, and Robert Seamans, "Linking Advances in Artificial Intelligence to Skills, Occupations, and Industries," in "AEA Papers and Proceedings" 2018.

Fuster, Andreas, Paul Goldsmith-Pinkham, Tarun Ramadorai, and Ansgar Walther, "Predictably unequal? the effects of machine learning on credit markets," *The Effects of Machine Learning on Credit Markets (November 6, 2018)*, 2018.

Grennan, Jillian and Roni Michaely, "Fintechs and the market for financial analysis," *Michael J. Brennan Irish Finance Working Paper Series Research Paper*, 2018, (18-11), 19–10.

Jones, Charles I., "The Shape of Production Functions and the Direction of Technical Change," *Quarterly Journal of Economics*, 2005, *120* (2), 517–549.

Jones, Charles I. and Chris Tonetti, "Nonrivalry and the Economics of Data," *American Economic Review*, 2020, *110* (9), 2819–2858.

Karabarbounis, Loukas and Brent Neiman, "The Global Decline of the Labor Share," *Quarterly Journal of Economics*, 2014, *129* (1), 61–103.

_ **and** _ , "Trends in factor shares: Facts and implications," *NBER Reporter*, 2017, (4), 19–22.

Klein, Lawrence R and Richard F Kosobud, "Some econometrics of growth: Great ratios of economics," *The Quarterly Journal of Economics*, 1961, *75* (2), 173–198.

Lagakos, David and Michael Waugh, "Selection, Agriculture and Cross-Country Productivity Differences," *American Economic Review*, 2013, *103*, 948–80.

McGrattan, Ellen R., "Intangible capital and measured productivity," *Review of Economic Dynamics*, 2020.

Webb, Michael, "The Impact of Artificial Intelligence on the Labor Market," *Available at SSRN 3482150*, 2019.

Not-for-Publication Appendix: Measurement Details, Model Derivations and Robustness Results

A Measurement

A.1 Identifying Investment Management Jobs

We identify Investment management jobs as those that require at least one skill belonging to the following Burning Glass skill clusters: 'Asset Management Industry Knowledge', 'Electronic Trading Systems', 'Investment Management', 'Financial Trading', 'Financial Trading Industry Knowledge', 'Investment Services Industry Knowledge', 'Financial Advisement'.

This list of skill clusters was compiled by tabulating all skill clusters required by any of the jobs in our sample and selecting those most related to investment management.

Since sometimes skills clusters are missing, we compile a list of all skills ever present in the list of relevant skill clusters and also classify as 'investment management' those jobs that require at least one of those underlying skills.

We finally check the full list of skills required by the selected jobs and exclude those jobs which require the following skills, as we believe these jobs are not likely to be actual investment management jobs: 'Marketing Strategy', 'General Marketing', 'Urban Planning', 'Technical Support', 'Telemarketing', 'Business-to-Business (B2B) Sales', 'Marketing Automation', 'Litigation', 'Retail Sales', 'Billing and Invoicing', 'General Administrative and Clerical Tasks', 'Journalism', 'Claims Processing', 'Merchandising', 'Carpentry', 'Animation and Game Design', 'Basic Customer Service', 'Cash Register Operation', 'Real Estate and Rental', 'Marketing Software', 'Online Marketing', 'Accounts Payable and Receivable', 'Packaging and Labeling', 'Inventory Management', 'Advanced Customer Service', 'Payroll', 'Underwriting', 'Marketing Management', 'Supply Chain Planning'.

A.2 Categorizing Jobs

Jobs are first categorized into 'data management' (DM) and 'data analysis' by looking at the relative frequency of the 'data management' vs. 'data analysis' keywords listed below in the full text of the underlying job postings. Jobs identified as 'data analysis' are further categorized (where possible) as AI or old technology (OT), by looking at the relative frequency of the AI and OT keywords listed below - these are subsets of the 'data analysis' keywords.

All keywords lists are obtained by first tabulating all Burning Glass skills present in the selected sample and identifying skills that best map to the types of jobs described by the model. We then also inspected the text of selected job postings requiring most of the selected skills in order to refine the keywords and phrases to best reflect the format in which they are most frequently present in the text.

Before computing relative frequencies both the keywords lists and the underlying text are pre-processed and stemmed to their root using the Porter stemmer.

Data Management keywords : 'Apache Hive', 'Information Retrieval', 'Data Management Platform (DMP)', 'Data Collection', 'Data Warehousing', 'SQL Server', 'Data Visualization', 'Database Management', 'Data Governance', 'Data Transformation', 'Extensible Markup Language (XML)', 'Data Validation', 'Data

Architecture', 'Data Mapping', 'Oracle PL/SQL', 'Database Design', 'Data Integration', 'Teradata', 'Database Administration', 'BigTable', 'Data Security', 'Database Software', 'Data Integrity', 'File Management', 'Splunk', 'Relational DataBase Management System', 'Teradata DBA', 'Data Migration', 'Information Assurance', 'Enterprise Data Management', 'SSIS', 'Sybase', 'jQuery', 'Data Conversion', 'Data Acquisition', 'Master Data Management', 'Data Capture', 'Data Verification', 'MongoDB', 'Data Warehouse Processing', 'SAP HANA', 'Data Loss Prevention', 'Data Engineering', 'Database Schemas', 'Database Architecture', 'Data Documentation', 'Data Operations', 'Oracle Big Data', 'Domo', 'Data Manipulation', 'Data Management Platform', 'DMP', 'HyperText Markup Language', 'Data Access Object (DAO)', 'Structured Query Reporter', 'SQR', 'Data Dictionary System', 'Data Entry', 'Data Quality', 'Data Collection', 'Information Systems', 'Information Security', 'Change data capture', 'Data Management', 'Data Governance', 'Data Encryption', 'Data Cleaning', 'Semi-Structured Data', 'Data Evaluation', 'Data Privacy', 'Dimensional and Relational Modeling', 'Data Loss Prevention', 'Data Operations', 'Relational Database Design', 'Database Programming', 'Information Systems Management', 'Database Tuning', 'Object Relational Mapping', 'Columnar Databases', 'Datastage', 'Data Taxonomy', 'Informatica Data Quality', 'Data Munging', 'Data Archiving', 'Warehouse Operations', 'Solaris', 'Data Modeling', 'data feed management', 'data discovery', 'exporting large datasets', 'exporting datasets', 'database performance', 'disigning relational databases', 'implementing relational databases', 'designing and implementing relational databases', 'database development', 'data production process', 'normalize large datasets', 'normalize datasets', 'create database', 'Develop database', 'data onboarding', 'Data Sourcing', 'data purchase', 'data inventory', 'cloud Security', 'negotiating data', 'data attorney', 'data and technology attorney', 'reliability engineering', 'reliability engineer', 'data specialist', 'enable vast data analysis', 'enable data analysis', 'Data team', 'capturing data', 'processing data', 'Supporting data', error free data sets', 'error free datasets', 'live streams of data', 'data accumulation', 'Kernel level development', 'large scale systems', 'Hadoop', 'distributed computing', ' multi database web applications', 'connect software packages to internal and external data', 'explore data possibilities', 'architect complex systems', 'build scalable infrastructure for data analysis', 'build infrastructure for data analysis', 'solutions for at scale data exploration', 'solutions for data exploration', 'information technology security', 'security engineer', 'security architect', 'architect solutions to allow modelers to process query and visualize higher dimensional data'

Analysis keywords

- General Analysis: 'Regression Algorithms', 'Regression Analysis', 'Quantitative Analysis', 'Clustering', 'Time Series Analysis', 'Economic Analysis', 'Model Building', 'Quantitative Research', 'pandas', 'numpy', 'Hedging Strategy', 'Quantitative Data Analysis', 'Investment Analysis', 'Economic Models', 'Predictive Analytics', 'Market Trend', 'Portfolio Optimization', 'Portfolio Rebalancing', 'Financial Derivatives Pricing', 'Active Alpha Generation', 'Financial Data Interpretation', 'Alteryx', 'Predictive Models', 'Exploratory Analysis', 'Sensitivity Analysis', 'News Analysis', 'Asset Allocation', 'Research Methodology', 'Mathematical Software', 'Portfolio Construction', 'Portfolio Analysis', 'Portfolio Analyst', 'Market Analysis', 'Data Techniques', 'Capital Allocation', 'Financial Modeling', 'Algorithm Development', 'Securities Trading', 'Trading Strategy', 'Statistical Programming', 'Data Mining', 'Social Network Analysis', 'Dimensionality Reduction', 'Principal Components Analysis (PCA)', 'Statistical Software', 'Portfolio Management', 'Numerical Analysis', 'Time Series Models', "Asset Allocation Theory", 'Analytical Skills', 'Financial Analysis', 'Financial Modeling', 'Modern Portfolio Theory', 'MPT', 'Portfolio Valuation', 'strategic portfolio decisions'

- Old Technology: 'Linear Regression', 'Logistic Regression', 'Statistic', 'STATA', 'Emacs', 'Technical Analysis', 'Qualitative Analysis', 'Qualitative Portfolio Management', 'Data Trending', 'Stochastic Optimization', 'Multivariate Testing', 'Bootstrapping', 'Time Series Models', 'Factor Analysis', 'Durations analysis', 'Markov', 'HMM', 'Econometrics', 'Stochastic Processes', 'Calculus', 'Statsmodels', 'Linear Algebra', 'Mathematics', 'Maths', 'Monte Carlo Simulation', 'Generalized Linear Model', 'GLM', 'Linear Programming', 'Bayesian', 'Analysis Of Variance', 'ANOVA', 'Behavioral Modeling', 'Black-Scholes', 'Behavior Analysis', 'Discounted Cashflow', 'Numerical Analysis', 'Correlation Analysis', 'E-Views', 'Differential Equations', 'Algebra', 'Value at Risk', 'Asset Pricing Models', 'Statistician', 'Mathematician', 'Econometrician'
- AI: 'Artificial Intelligence', 'Machine Learning', 'Natural Language Processing', 'NLP', 'Speech Recognition', 'Gradient boosting', 'DBSCAN', 'Nearest Neighbor', 'Supervised Learning', 'Unsupervised Learning', 'Deep Learning', 'Automatic Speech Recognition', 'Torch', 'scikit-learn', 'Conditional Random Field', 'TensorFlow', 'Tensor Flow', 'Platfora', 'Neural Network', 'CNN', 'RNN', 'Neural nets', 'Decision Trees', 'Random Forest', 'Support Vector Machine', 'SVM', 'Reinforcement Learning', 'Torch', 'Lasso', 'Stochastic Gradient Descent', 'SGD', 'Ridge Regression', 'Elastic-Net', 'Text Mining', 'Classification Algorithms', 'Image Processing', 'Natural Language Toolkit', 'NLTK', 'Pattern Recognition', 'Computer Vision', 'Long Short-Term Memory', 'LSTM', 'K-Means', 'Geospatial Intelligence', 'Big Data Analytics', 'Latent Dirichlet Allocation', 'LDA', 'Backpropagation', 'Machine Translation', 'Caffe Deep Learning Framework', 'Word2Vec', 'Genetic Algorithm', 'Evolutionary Algorithm', 'Data Science', 'Sentiment Analysis / Opinion Mining', 'Maximum Entropy Classifier', 'Neuroscience', 'Computational Linguistics', 'Semi-Supervised Learning', 'Data Scientist'

A.3 Identifying jobs for employers of interest:

To match the categorized job postings to the right employers, we use the following procedure:

1. *Create a master list of employers of interest:* We compile a comprehensive list of investment management companies using firms included in two data sources, Preqin and SEC. From Preqin, we select alternative asset managers. From the SEC database, we compile filers of Form 13F, a quarterly report of top ten equity holdings, filed by institutional investment managers with at least $100 million in assets under management. From the final list of firms we exclude commercial banks, insurance companies and private equity firms. To avoid repetitions, we manually cluster entities that refer to the same underlying company (e.g. "citigroup", "citigroup north america"). We then standardize these employers names to create a canonical form for each cluster that uniquely identifies it.[9]

2. *Extract candidate employers:* We use three techniques to identify strings that could potentially be the correct employer:

   (a) *Employer from BGT* - for a significant number of job postings, BGT identifies the employer using both manual and automated techniques. While it is not always available and can be incorrect, this employer will be added to our set of candidates.

[9] Job descriptions are scraped and therefore dirty. We remove excessive spaces and line breaks, unrecognizable symbols, HTML codes, and other irrelevant artifacts.

(b) *Keyword Search* - for each employer in the master list, we can generate keywords that identify this employer. We remove keywords that are too general and look for exact matches of these keywords in the job description. These matching words or phrases are added to our set candidates.[10]

(c) *Named Entity Recognition – NER* - using part-of-speech tagging and word capitalization, we can identify words or phrases that are likely to be named entities (e.g. organization names, countries, people's names, etc.) from the job descriptions. These named entities, which potentially overlap with candidates from the previous methods, are added to our set of candidates.

3. *Standardize candidates and master list* The first step to matching the candidates to the master list is to standardize employer names on both sides. Our standardization algorithm goes a step further than the basic cleaning applied to the job descriptions. The algorithm removes non-identifying suffixes and prefixes, such that leading the's and corporate designations such as "inc" and "llc". It also intelligently remove generic words (for instance "management" or "capital") only when they are not useful. For example, "The Blackstone Group" will become "blackstone" because "blackstone" is highly identifying. On the other hand, "Capital Group" will remain as "capital group" because stripping out "group" will reduce the employer name to an exceedingly common word "capital".

4. *Map raw candidates to master list:* After collecting a list of raw candidates for each job posting, we first deduplicate the candidate set, then we compute a similarity score for every possible candidate-master employer pair. The computation of the simialrity metric requires as input the frequency of all words appearing in any of the canonical names in the master list of employers.

word	**frequency (F)**
capital	2,799
asset	745
advisors	684
...	...
sachs	2
vanguard	1

The optimal master employer given a candidate is then defined as follows:

$$master^* = \arg \max_{master} sim(candidate, master)$$

The computation of the similarity metric will be demonstrated using as an example a single candidate-master employer pair:

candidate = "Royal Banks of Canada."
master = "royal bank canada"

We begin by standardizing the candidate string, removing the period and uppercase in this case:

"Royal Banks of Canada." → "royal banks of canada"

Next, we obtain an optimal matching of the words such that the total Levenshtein-based similarity (modified to give a slight bonus to exact matches) is maximized. The word "of" is unmatched.

[10] These are also pre-processed, as previously outlined.

sim :	1.0	0.7	1.0	0
matches:	(royal, royal),	(banks, bank),	(canada, canada),	(of,)

Lastly, we take the weighted sum of all the matched words, with the inverse frequency as weight. We set a minimum weight of 0.1 for all words to avoid shrinking the weights of common words to near 0.

sim :	1.0	0.7	1.0	0
matches:	(royal, royal),	(banks, bank),	(canada, canada),	(of,)
$weight$:	$F(royal)^{-1}$	$F(bank)^{-1}$	$F(canada)^{-1}$	$F(of)^{-1}$

sim("Royal Canada Bank.", "royal bank of canada") = 0.867

$sim_{Levenshtein}$("Royal Canada Bank.", "royal bank of canada")=0.55

We only consider matches above a threshold of 0.75. In this example, the final similarity score of our algorithm is 0.867. That is high, given that half the words have non-exact matches. That is because the frequency in the master list of the words "bank" and "of" relative to "royal" and "canada" is high; hence they are downweighted in the similarity score computation. This match, instead, would have been discarted using for instance Levenshtein similarity.

FInally, due the bonus we award to exact word matches, minor typos or spacing issues can cause an otherwise obvious (to the human eye) match to be left out. To salvage these edge cases as much as possible, we use the following heuristic:

Example

candidate = "Royale Bank ofCanada inc."
master = "royal bank of canada"

(a) standardize candidate employer

candidate: "Royale Bank ofCanada inc." → "royale bank ofcanada"

(b) remove spaces from both candidate and master to form a single word

candidate: "royale bank ofcanada" → "royalebankofcanada"
master: "royal bank of canada" → "royalbankofcanada"

(c) compute the Levenshtein distance between these two single words and accept matching if similarity score is greater than 0.9.

$sim_{Levenshtein}$("royalebankofcanada", "royalbankofcanada")=0.945

The initial step of word matching is inspired by how a human would approach the problem. If we encounter "canandian bank" and "bank of canada" for example, it is natural to associate highly similar words and compare locally. This gives us an edge over direct applications of traditional metrics such as Levenshtein distance and Jaro-Winkler distance.

5. *Select the best candidate* At this point, we have a list of candidates for each job posting. All the candidates correspond exactly to a single employer in our master list. The final step is to select the most likely candidate. To evaluate the quality of each candidate, we took into consideration three types of features:

- Consistency - is the same candidate identified using multiple methods (BGT, text search, NER)?
- Similarity to employer in master - how similar is the raw candidate to the employer in the master list?
- Frequency - for the methods that use the full job posting (text search and NER), how frequently was the employer mentioned in the text?

More specifically, when the Burning Glass employer is corroborated by at least one of the other two methods, we assign that job posting to the matched employer. When a Burning Glass employer is not corroborated by any of the other two methods, we accept the match only if it is exact (excluding all fuzzy matches). Finally, when the Burning glass employer is not present but both text-based methods agree, we assign that job to the matched employer only if it is an exact match and the employer name is repeated at least 3 times in the text. This reduces matching noise. We do not consider matches that only appear in one of the text-based methods, as too noisy.

If after following this procedure the candidate set is empty, we decide that the true employer is not found in our master list. If the candidate set contains more than one element, we pick the candidate with the highest similarity score.

A.4 Constructing the Labor Inflows Data

Job openings, filling and separation data Our data comes from

`https://www.bls.gov/news.release/jolts.tn.htm`

Job Openings Rate: Job openings information is collected for the last business day of the reference month. A job opening requires that: 1) a specific position exists and there is work available for that position, 2) work could start within 30 days whether or not the employer found a suitable candidate, and 3) the employer is actively recruiting from outside the establishment to fill the position. The job openings rate is computed by dividing the number of job openings by the sum of employment and job openings and multiplying that quotient by 100.

Hiring Rate: The hires level is the total number of additions to the payroll occurring at any time during the reference month, including both new and rehired employees, full-time and part-time, permanent, short-term and seasonal employees, employees recalled to the location after a layoff lasting more than 7 days, on-call or intermittent employees who returned to work after having been formally separated, and transfers from other locations. The hires rate is computed by dividing the number of hires by employment and multiplying that quotient by 100.

Separations Rate: The separations level is the total number of employment terminations S occurring at any time during the reference month, and is reported by type of separation - quits, layoffs and discharges, and other separations. The separations rate is computed by dividing the number of separations by employment and multiplying that quotient by 100: $s = S/E \cdot 100$.

Deriving the probability of filling an opening. If n_O is the total number of posted job openings, n_E is total employment and n_H is the number of new hires in this sub-occupation and month, then the BLS hiring rate is defined to be $r_h = n_H/n_E$, while the job opening rate is $r_o = n_O/(n_E + n_O)$. What we need to adjust the openings data from our model, is the fraction of openings that result in hires, $h = n_H/n_O$.

To solve for h, note that rearranging the definition of the opening rate yields $r_o = (1 - r_o)n_O/n_E$. Dividing r_h by this expression yields $r_h/r_o = (n_H/n_E)/((1-r_o)n_O/n_E) = (n_H/n_O) \cdot 1/(1-r_o)$. Therefore, we can express the n_H/n_O rate we want as $h = r_h(1 - r_o)/r_o$.

Time to Fill a Job Vacancy In our calculations, we have implicitly equated a job posting with a one-month job vacancy. We do that because most of our job postings remain up and unfilled for approximately one month. Below, we report the distribution of the average time that job postings remain open in our data set. This data is for jobs that have the same occupations and regions as our sample for the years 2015, 2016 and 2017. The average time to fill is available for 86% of all the occupation (SOC) - region (MSA) combinations in our sample. Below is the distribution of the average time a Burning Glass job posting stayed online for all the SOC-MSA combinations in our sample for 2015-2017.

mean	35.6857
std	7.1003
min	14.0000
1%	21.0000
5%	24.0000
10%	27.0000
15%	28.0000
20%	30.0000
25%	31.0000
30%	32.0000
35%	33.0000
40%	34.0000
45%	35.0000
50%	35.0000
55%	36.0000
60%	37.0000
65%	38.0000
70%	39.0000
75%	40.0000
80%	41.0000
85%	43.0000
90%	44.4000
95%	48.0000
99%	54.0000
max	75.0000

Table 3: **Time to Fill Posted Vacancies**.

If we weight each of these fill times by the number of jobs present in our sample for each the SOC-MSA combinations, we get an average fill times of 38.12 days.

B Model derivations

Firm i faces the following optimizing problem:

$$v(D_{it}) = \max_{\lambda_{it}, L_{it}, l_{it}} D_{it}^{\alpha} L_{it}^{1-\alpha} + D_{it}^{\gamma} l_{it}^{1-\gamma} - w_{L,t} L_{it} - w_{l,t} l_{it} - w_{\lambda,t} \lambda_{it} + \frac{1}{r} v(D_{i(t+1)}) \tag{17}$$

$$\text{where } D_{i(t+1)} = (1-\delta) D_{it} + \lambda_{it}^{1-\phi}. \tag{18}$$

Here the state variable is structured data D_{it}, and the control variables are data management labor λ_{it}, the machine learning analyst labor L_{it} and the old technology analysis labor l_{it}. Plugging (18) into (17), we have

$$v(D_{it}) = \max_{\lambda_{it}, L_{it}, l_{it}} D_{it}^{\alpha} L_{it}^{1-\alpha} + D_{it}^{\gamma} l_{it}^{1-\gamma} - w_{L,t} L_{it} - w_{l,t} l_{it} - w_{\lambda,t} \lambda_{it} + \frac{1}{r} v\left((1-\delta) D_{it} + \lambda_{it}^{1-\phi}\right) \tag{19}$$

Taking partial derivative with respect to L_{it}, we have

$$(1-\alpha) D_{it}^{\alpha} L_{it}^{-\alpha} - w_{L,t} = 0 \implies \frac{(1-\alpha) K_{it}^{AI}}{L_{it}} = w_{L,t}. \tag{20}$$

Taking partial derivative with respect to l_{it}, we have

$$(1-\gamma) D_{it}^{\gamma} l_{it}^{-\gamma} - w_{l,t} = 0 \implies \frac{(1-\alpha) K_{it}^{OT}}{L_{it}} = w_{l,t}. \tag{21}$$

Taking partial derivative with respect to λ_{it} and rearranging, we have

$$\frac{1}{r} v'(D_{i(t+1)})(1-\phi) \lambda_{it}^{-\phi} = w_{\lambda,t}. \tag{22}$$

We then total differentiate (19) to get

$$v'(D_{it}) = \frac{\alpha K_{it}^{AI}}{D_{it}} + \frac{\gamma K_{it}^{OT}}{D_{it}} + \frac{1}{r} v'(D_{i(t+1)})(1-\delta). \tag{23}$$

If we further assume that the marginal value of data today and tomorrow are similar, then

$$v'(D_{it}) = \frac{(\alpha K_{it}^{AI} + \gamma K_{it}^{OT})}{D_{it}} \frac{r}{r-(1-\delta)}. \tag{24}$$

Plugging it back to the first order condition (22) and combining it with the structured data dynamics (18), we arrive at

$$\frac{(\alpha K_{it}^{AI} + \gamma K_{it}^{OT})(1-\phi)}{r-(1-\delta)} \frac{D_{i(t+1)} - (1-\delta) D_{it}}{D_{it}} = w_{\lambda} \lambda_{it}. \tag{25}$$

C Robustness

Figures 9 and 10 (left panel) illustrate the evolution of the aggregate data stock of firms for 1% and 10% monthly rates of data depreciation. The total amount of data with 1% depreciation is an order of magnitude higher. This estimate comes from inferring firms' initial data stocks from their employment choices, measuring the data managment workers who build up these stocks of data, and adjusting for data depreciation. The dip in the stock of data in the first year for 10% depreciation reflects a high initial data stock that would not have been possible

to maintain with 10% depreciation, given the level of data managment labor in that year. This suggests that perhpas 10% depreciation is too high.

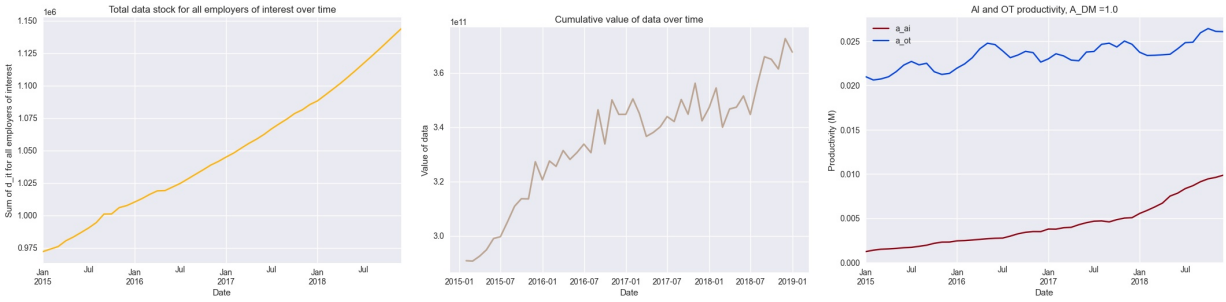

Figure 9: Estimated Data Stock, Data Value and Analysis Productivity with 1% data depreciation. Left panel: The aggregate data stock is $\sum_i D_{it}$ in each month t. Middle panel: The cumulative value of data is $\sum_i v(D_{it})$ in each month t, where the value function $v(\cdot)$ is given by (4). Right panel: Productivity is (A_t^{OT}) and (A_t^{IT}) as defined in (2) and (1). Data source: PayScale and Burning Glass, 2015-2018.

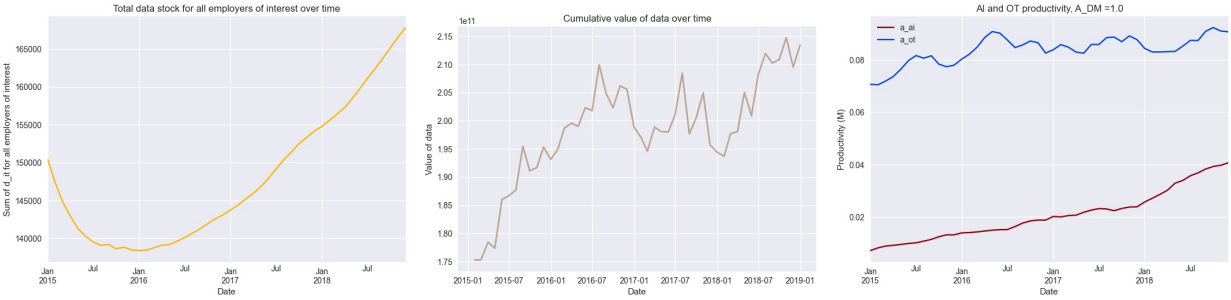

Figure 10: Estimated Data Stock, Data Value and Analysis Productivity with 10% data depreciation. Left panel: The aggregate data stock is $\sum_i D_{it}$ in each month t. Middle panel: The cumulative value of data is $\sum_i v(D_{it})$ in each month t, where the value function $v(\cdot)$ is given by (4). Right panel: Productivity is (A_t^{OT}) and (A_t^{IT}) as defined in (2) and (1). Data source: PayScale and Burning Glass, 2015-2018.

Once we have estimated production parameters and data stocks, we can put them back into our value function, and approximate the value of each firm's stock of data in each month. This value is in nominal dollar units, since those are the units of the wages we use. This is our estimate of the value function in (4) for the aggregate stock of data. The middle panels of Figures 9 and 10 show the value the model assigns to these aggregate stocks of data, for data depreciation of 1% and 10% per month. Although noisy, this value is clearly increasing. This rise reflects both more data and the fact that each unit of data earns a higher share of firm profit.

Finally, firms are becoming more productive at using data. More productivity also contributes to the rise in the value of data. The right panels show the evolution of analysis productivity in the old technology (A_t^{OT}) and the new big-data technology (A_t^{IT}) . Both types of analysts are becoming more productive each month, for data depreciation rates of 1% and 10% per month.