Imputing Mutual Fund Trades *

Dion Bongaerts† Joop Huij‡ Jean-Paul van Brakel§ Mathijs van Dijk‖

This draft: May 2023

ABSTRACT

We propose a novel method to impute daily mutual fund trades in individual stocks from data on quarterly fund holdings, monthly total net assets, and daily fund returns – so that the method can be applied to standard CRSP mutual fund data. We set up an (underidentified) system of linear equations and solve the underidentification issue with an iterative method that applies random and adaptive constraints on trade incidence. The method produces daily, stock-level trade estimates with associated confidence levels. Validation and simulation analyses using proprietary daily fund trading data show good accuracy, especially for larger trades.

JEL classification: G23, C63, C81.

Keywords: Mutual funds; latent variables; daily frequency; Thompson sampling.

*We thank Mathijs Cosemans, Georgi Kyosev, Raoul Martin, Marno Verbeek, and seminar participants at Rotterdam School of Management, Erasmus University and ESSEC Business School for helpful comments and discussion. We are grateful to SURF (www.surf.nl) for support in using the Dutch National Supercomputer Snellius.

†Rotterdam School of Management, Erasmus University, Netherlands; E-mail: dbongaerts@rsm.nl.

‡Rotterdam School of Management, Erasmus University; Robeco Institutional Asset Management, Netherlands; E-mail: jhuij@rsm.nl.

§Rotterdam School of Management, Erasmus University; Robeco Institutional Asset Management, Netherlands; E-mail: brakel@rsm.nl.

‖Rotterdam School of Management, Erasmus University, Netherlands; E-mail: madijk@rsm.nl.

Mutual funds manage a sizeable fraction of overall equity market capitalization (about 27% globally and 30% in the U.S.[1]). There is a large literature on the behavior of mutual funds and their impact on financial markets, covering issues such as mutual fund managerial skill (e.g. Jiang, Yao, and Yu, 2007; Cremers and Petajisto, 2009; Fama and French, 2010; Berk and Van Binsbergen, 2015), herding behavior (Grinblatt, Titman, and Wermers, 1995; Wermers, 1999), agency problems and tournament effects (Brown, Harlow, and Starks, 1996; Sirri and Tufano, 1998; Huang, Sialm, and Zhang, 2011), tax-induced trading and window dressing (Gibson, Safieddine, and Titman, 2000; Agarwal, Gay, and Ling, 2014), fire sales and trading activity during financial crises (Coval and Stafford, 2007; Hau and Lai, 2017), and impact on financial market quality (Campbell, Ramadorai, and Schwartz, 2009; Jiang, Verbeek, and Wang, 2014).

Many studies in this literature rely on quarterly mutual fund holdings data (e.g., CRSP data based on SEC forms N-CSR and N-Q) and use the change in the quarterly holdings of individual stocks as a proxy for fund trades during the quarter. This approach is less than ideal for studying various of the issues listed above. For example Elton, Gruber, Blake, Krasny, and Ozelge (2010) show that even increasing the frequency of holdings data from quarterly (from CRSP) to monthly (from Morningstar; only for a subset of funds) overturns some conclusions from the literature (e.g., on tournament behavior).

Yet, higher frequency holdings or trade data are hard to come by. Some studies combine (anonymous) microstructure data from TAQ with 13F filings (e.g., Campbell et al., 2009), which is hampered by difficulties in linking anonymous trades to fund holding companies. Other studies use daily trading data from Ancerno (formerly Abel Noser; Puckett and Yan, 2011; Chakrabarty, Moulton, and Trzcinka, 2017; Hu, Jo, Wang, and Xie, 2018; Busse, Chordia, Jiang, and Tang, 2021), which have limited cross-sectional and time-series coverage and are no longer available to researchers due to a company policy change.

As an alternative, in this paper, we propose a method to impute daily mutual fund trades in individual securities (i.e., trade incidences and sizes at the fund-stock-day level) using only standard CSRP data on quarterly holdings, monthly total net assets (TNA), and daily net asset values (NAV) as well as daily fund and security returns. This method allows us to estimate daily trades for all

[1]ICI (2022, p. 5 and 30) reports that the worldwide total net assets of equity funds is $33.6 trillion and that 30% of the U.S. stock market capitalization is owned by U.S. domestic equity mutual funds and ETFs as of 2021. WFE (2022, p. 1) reports that the worldwide stock market capitalization is $123 trillion at the end of 2021.

U.S. mutual funds from 2008 onward, but could also be applied to international data, mutual fund trades in other asset classes, and – to the extent that CRSP mutual fund holdings data before 2008 were complete and accurate – also to U.S. mutual fund data before 2008.

The idea is as follows. Changes in a mutual fund's quarterly holdings of individual stocks (from CRSP) indicate the minimum number of shares the fund has traded in each stock during the quarter. Daily fund returns (from CRSP) reflect (a weighted average of) the returns of the underlying holdings, and hence contain important information about the size and timing of fund trades within the quarter. We can thus use daily stock prices and returns (from CRSP) to infer trades in individual stocks that best align with daily fund returns. We further obtain monthly total net assets (TNA; from CSRP), which gives us two extra data points per quarter to pin down trades in individual stocks (and fund in- and outflows).

Despite the information that these four data sources contain about a fund's underlying trades within the quarter, imputing daily trades (trade incidence and trade size) in individual stocks is not straightforward. We can formulate the problem as a system of equations, albeit one that is clearly underidentified. We tackle the underidentification issue in two steps. First, we impose additional constraints that reduce the degree to which the system is underidentified. We are careful to keep these constraints as innocuous as possible. For example, we restrict short sales for a given fund in a given quarter (we call this a "fund-quarter") if a fund does not report any short positions in its holdings at the start or end of the quarter.

Second, and more importantly, we permute the problem in a specific way, solve it, and then iterate till convergence. The permutations we apply consist of no-trade constraints in certain stocks on certain days (on what we call "stock-days"). Since most funds only change their positions on a (very) limited number of days during each quarter, these permutations are in line with market practice. Moreover, since these permutations (massively) reduce the dimensionality of the problem, they are really useful in addressing the underidentification issue. As these permutations can preclude an exact solution, we solve, for each permutation, a linear optimization problem (as opposed to a system of equations) in which me minimize absolute residual fund return – defined as the absolute difference between the actual daily fund return and the daily fund return implied by the imputed daily trades in individual stocks.[2]

[2]We cast our constrained optimization problem into a linear program form as linear programs can be efficiently

How to construct these permutations is a key challenge in our approach, since, as econometricians, we do not know on which days a given fund trades which stock. To this end, we employ a reinforcement learning approach in which the algorithm (Thompson sampling; see Thompson, 1933), in each iteration, randomly sets trades in specific stocks on specific dates for a given fund to zero. The algorithm is adaptive in that it gives a higher probability for a (potentially) non-zero stock-day if that stock-day showed up more frequently as non-zero in earlier iterations. Still, draws are conditionally-random (conditional on previous iterations) to allow the algorithm to escape local optima and converge to a global optimum.

Our permutations are random and some of the imposed constraints in each iteration are likely to be violated. As a result we get noise and the probability that a non-zero trade occurs on an unconstrained stock-day is unlikely to be exactly equal to zero, which hinders quick convergence of the algorithm. For that reason, in each iteration, we shrink very small but non-zero estimated trade sizes to zero. Because of this shrinkage and because very small trades hardly affect fund returns, our method is expected to perform well for identifying the larger trades (which arguably matter the most), but less so for very small trades.

The reinforcement learning approach we apply also has other advantages. The algorithm produces at each iteration the probability that a given stock-day is allowed to be non-zero in that iteration. This provides us, after convergence, with a confidence level for how certain the algorithm is about an identified trade. These confidence levels can be very useful when using the imputed trades in subsequent analysis. One can, for example, screen on a minimum confidence level for identified trades or one can weigh each identified trade with its corresponding confidence level. In other words, the confidence levels of imputed trades allow for a straightforward way to account for the effects of measurement error when using these imputed trades in empirical analyses.

We proceed by assessing the performance of our method to impute mutual fund trades, and by analyzing the sensitivity of the method's results to parameter and implementation choices. To validate our method, we obtain data on the holdings, returns, TNAs, and exact trades conducted by all fundamentally-managed funds of a large anonymous European fund family over a period of 4 quarters of a recent year. Since these funds are active globally, we match these fund data to stock return data from Refinitiv Eikon.

solved. This preserves computational resources in an already computationally intensive iterative solution method.

Because the number of fund-quarters in this proprietary dataset is limited, we use these data to simulate quarters with hypothetical mutual funds. For this analysis, we start by randomly drawing a number of positions in a simulated fund-quarter, say K. Next, we randomly draw a quarter Q from the dataset. Subsequently, we randomly draw K positions and their corresponding trades from all funds in the dataset during quarter Q. Finally, we scale positions to ensure that portfolio weights add up to 100% on each day. This way, we have created a hypothetical mutual fund quarter from historical observations (which is reminiscent of the historical simulation approach for calculating market value-at-risk). For each of such hypothetical fund-quarters, we construct quarterly holdings, monthly TNAs, and daily returns. We then apply our method to impute daily fund trades in individual stocks and compare our imputed trades to the simulated trades.

The results of our simulation study indicate that our method performs well on generally accepted performance metrics, especially if we allow for a margin of error of a day earlier or later in the estimated timing of the trade (which for intercontinentally active funds is reasonable due to differences in time zones). Regarding trade incidence, the probability that our model ranks a given stock-day on which an actual trade took place higher than one on which no trade took place is 66% for the whole sample, which increases to 75% when one allows for a window of one day earlier or later and focuses on trades part of larger net position changes (exceeding 50bps of average fund value over the quarter) and to 85% for very large net position change (exceeding 250 bps) with a window of two days before and after the actual trading day. Regarding the trade size of correctly identified trades, the imputed trade size estimates are biased upwards due to the shrinkage that is applied, but the bias is small and decreasing with the relative size of the net position change. The variance of the size estimates is small.

In a further validation exercise, we calculate the active shares (Cremers and Petajisto, 2009) of the imputed daily mutual fund holdings (using the imputed trades in individual stocks) relative to the actual fund holdings, averaged over all days in the quarter, for both the validation and simulated data. We benchmark our results with the average active shares under the assumption that all trades take place at the start, in the middle, or at the end of the quarter (as is common in the literature). For both datasets, we show that our method outperforms by a large margin.

Our simulation approach also allows us to conduct sensitivity tests of our method. For example, we can vary the number of net position changes to assess the differential ability of our method for identifying trades of more concentrated vs. more diversified funds.

Although the primary purpose of our method is to identify daily fund trades in individual stocks, the algorithm also generates daily net mutual fund flow estimates as a side product (assuming that flows are accommodated the same day).[3] Since mutual fund TNA data are commonly only available at a monthly frequency, obtaining higher-frequency data on net flows is as challenging as it is for fund trades. While some data on daily net flows are available from data vendors like TrimTabs, these data have only limited coverage and are prohibitively expensive.

Our paper contributes to several strands of literature. First and foremost, we contribute to the broader empirical literature that uses mutual fund holdings data to study mutual fund managerial skill, herding behavior, agency problems and tournament effects, tax-induced trading and window-dressing, fire sales and trading activity during financial crises, and the impact of mutual funds on market quality. Since mutual fund holdings data is generally only available at a quarterly frequency, many studies that want to use data on fund trades in individual stocks make assumptions on the timing of trades and flows within the quarter. A common assumption is that all trades and flows materialize at the end of the quarter. In our validation analysis, we show that among the simple assumptions (begin, middle, end of quarter), this is the least accurate choice. We provide an alternative method to impute the timing and size of mutual fund trades in individual stocks (as well as fund fows) and show that it is superior to these simpler approaches. We hereby hope to facilitate future studies on a variety of issues in the mutual fund literature using these more detailed data on fund trades and flows.

We also contribute to a subsection of this literature that uses proprietary Ancerno data (e.g. Puckett and Yan, 2011; Chakrabarty et al., 2017; Hu et al., 2018; Busse et al., 2021). The Ancerno data have limited cross-sectional and time-series coverage and are potentially subject to reporting biases. This may raise concerns about representativeness and external validity. Furthermore, the Ancerno data are no longer updated or distributed, posing a data problem for future studies. Our approach only uses commonly available CRSP data (or SEC filings), which allows many researchers to work with imputed daily mutual fund trade and flow data and is beneficial for replicability in light of recent open science initiatives. Moreover, our approach enables assessing the external

[3]Anecdotal evidence suggests that daily accommodation of flows is standard practice for the majority of flows, as otherwise funds would need to borrow from their custodian or broker, which is costly.

validity of the results reported in studies conducted using Ancerno data.

A few prior studies discuss and analyze the effect of data quality and frequency on findings in the mutual fund literature. Elton et al. (2010) show that monthly holdings data from Morningstar instead of quarterly data from CRSP reverse some of the findings from the literature. However, the Morningstar sample in their study includes only 215 funds with monthly holdings data after applying data filters. Part of the fund attrition is likely due to funds refusing to report at a monthly frequency. By contrast, our method can be applied to all mutual funds reporting holdings at the mandatory quarterly frequency.

To the best of our knowledge, there is only one other study that tries to impute trades from the reported data available (Farrell, 2018). This study uses a greedy algorithm (a genetic algorithm), which is a local optimization approach as opposed to the global optimization algorithm embedded in our method. The combinatorial as well as the underidentified nature of the problem requires Farrell (2018) to make important additional assumptions, such as the split-up of each quarterly net position change into four identically sized trades. Our approach does not require such strong assumptions and even allows funds to buy and sell the same stock (within limits) at different times within the same quarter (i.e., intra-quarter round-trip trades). The latter is important since Chakrabarty et al. (2017) show that 23% of institutional round-trip trades are held for less than 3 months.

Finally, we also contribute to the robust optimization literature by showing that reinforcement learning can be used in conjunction with (linear) optimization problems to improve identification. The typical approach to robust optimization is to include uncertainty in either the objective function or the constraints of the problem, which requires strong assumptions and often results in intractability issues (Ben-Tal, El Ghaoui, and Nemirovski, 2009). Building on the idea of Liu and Ročková (2021), we show that sparse optimization problems can be interpreted as a variable-selection problem (in the class of multi-armed bandit problems), such that Thompson sampling can be used to find robust solutions instead. Even though the theory for this class of problems is still being developed (see the discussions in Liu and Ročková, 2021; Hong, Kveton, Zaheer, and Ghavamzadeh, 2022), our results provide new evidence that the application of reinforcement learning to robust optimization is a promising area for further research.

I. Methodology

In this section, we explain how we use available mutual fund data to impute the trades executed by a given fund in a given quarter. Our goal is to identify the (latent) trades of the fund that are consistent with three publicly available fund characteristics: the daily returns of the fund, the total value of the fund at the end of each month, and the holdings of the fund at the (beginning and) end of each quarter. To impute trades at a daily frequency, we also use the (publicly available) daily prices and returns of individual stocks.

Our approach consists of three steps. First, we show that the problem of imputing trades leads to a large and highly sparse system of linear equations. Any solution to this system of equations corresponds to a set of trades that is consistent with our three publicly available measures. Yet, the system is typically underidentified, and multiple solutions are therefore likely to exist. To improve identification, we impose additional structure on the model by restricting improbable trading patterns. This leads to an extended and tractable linear programming problem. Yet, the additional restrictions do not guarantee unique solutions.

Even in such an underidentified linear program, there are trades that will show up in virtually any solutions because their signal-to-noise ratio is high. We therefore propose an iterative reinforcement-learning approach that aims to identify the trades that show up in a robust fashion over small permutations of the original problem. These permutations, however, lead to noise. The approach we take is based on the product of trade incidence (a binary variable) and trade size (a continuous variable). Due to the noise induced by the permutations, our approach in each iteration (spuriously) identifies many trades with very small trade sizes. If unaddressed, these very small trades would lead to an upward bias in trade incidence estimates. To mitigate this bias, we shrink small trades in each iteration towards zero. This also improves the convergence speed of our proposed algorithm.

Our approach also has another side benefit. We realize that while data quality is good, it is not perfect. Moreover, we make some assumptions that may only approximately hold. For example, most, but not all mutual fund trades may take place at end-of-day prices. Our focus on trades that show up in a robust fashion over small permutations of the original problem also makes our estimates robust to small data errors and small degrees of model misspecification.

A. Model setup

We define trades as the day-to-day differences between the number of shares that a fund holds in a given stock (as of day end). Let $s_{i,t}$ denote the number of shares that a fund holds in stock i on day t. The trade size of stock i on day t is then defined as $s_{i,t} - s_{i,t-1}$. Without loss of generality, we model the cash position of the fund as if they are shares in a stock with a share price equal to unity and daily returns equal to zero. We assign stock $i = 1$ to reflect the cash position denominated in the base currency of the fund.[4] We impute trades for one fund-quarter at a time. The model space is therefore defined over the stock positions, denoted by the index $i \in \{1, ..., N\}$, and the days, denoted by the index $t \in \{1, ..., T\}$, within a given fund-quarter. We use N to denote the total number of unique stocks that the fund reports a position in at the start and/or end of the quarter. The number of trading days T in a given quarter typically ranges between 61 and 64 days. All our subsequent derivations are for a given fund-quarter.

We start the process of identifying the trades by relating them to the publicly available data. The first piece of information that we incorporate are the holdings that U.S. mutual funds are mandated to report on a quarterly basis. These data include the number of shares of each position in the portfolio as of the close of the last business day in the respective quarter. Following Lakonishok, Shleifer, and Vishny (1992), Wermers (1999) and Ke and Ramalingegowda (2005), among others, we infer the net position change (net number of shares traded) in each stock from the differences between the consecutively reported quarterly holdings. For each stock i, the net position change is calculated as $s_{i,T} - s_{i,0}$, with $t = 0$ defined as the last day of the previous quarter, and $t = T$ the last day of the current quarter. As an identity, the sum of the trades in stock i during the quarter must equal the net position change over the quarter:

CONDITION 1: $\sum_{t=1}^{T} (s_{i,t} - s_{i,t-1}) = s_{i,T} - s_{i,0}, \forall i.$

For modeling purposes, we standardize daily trades by the net position change over the quarter to obtain $x_{i,t}$, which denotes the fraction of shares that is traded in stock i on day t relative to the total net position change over the quarter.[5] The daily trading fraction $x_{i,t}$ and the daily position

[4] In principle, funds can have cash positions in multiple currencies, e.g., for collateral, margin, or trading purposes. We convert foreign currency cash positions to the base currency and add them to the base currency cash position.

[5] The two problem formulations are mathematically equivalent and lead to the same results. We prefer the standardized form because this allows for concise notation and intuitive comparisons among trades. For example, a fund that trades 10 shares of stock i on day t and has a net position change of 100 shares of stock i over the whole

$s_{i,t}$ of stock i on day t are related as follows:

$$s_{i,t} = s_{i,0} + (s_{i,T} - s_{i,0}) \sum_{j=1}^{t} x_{i,j}, \forall i, \forall t. \tag{1}$$

To adhere to Condition (1), the sum of all fractions $x_{i,t}$ over the quarter must equal one for each stock:

CONDITION 2: $\sum_{t=1}^{T} x_{i,t} = 1, \forall i.$

The second piece of information that we use comprises of the total net assets (TNA) that the fund reports at the end of each month. The TNA is calculated as the aggregate market value of the fund's underlying assets (stocks and cash), minus its liabilities (accrued management fees, 12b-1 fees, expenses, etc.), calculated against the closing prices of that day. As such, the TNA represents the total dollar value of a fund at a given moment in time. To keep our notation concise, we first express the market value of each position in the fund at the end of each day. We calculate the market value $v_{i,t}$ of stock i at the close of day t by multiplying the number of shares $s_{i,t}$ that the fund holds with the closing price $p_{i,t}$:

$$v_{i,t} = s_{i,t} p_{i,t}, \forall i, \forall t. \tag{2}$$

Since we use daily closing prices, we implicitly assume that all trades have been executed as market-on-close orders. From personal conversations with fund managers, we believe this assumption to be fairly realistic. Mutual funds typically guarantee single-day liquidity, and net asset values (and therefore cashflows to clients) are typically calculated using closing prices by the custodian. Therefore, fund managers aim to transact at the closing price so as to avoid exposing themselves (and thereby other investors in the fund) to unnecessary market risk. Failing to transact on the same day may also require the custodian to advance or absorb cash flows on behalf of the fund, for which the custodian charges the fund fees and/or interest. This gives another incentive to fund managers to trade at end-of-day prices. Another advantage of using end-of-day prices is that it improves our model's robustness to transaction costs. Transaction costs can be decomposed into fixed-cost transaction fees, paid in cash to the brokers, and variable transaction costs, via price

quarter, has $x_{i,t} = \frac{10}{100} = 0.1$.

impact on the stock (see Chan and Lakonishok, 1993, 1995; Chiyachantana, Jain, Jiang, and Wood, 2004). By modeling cash positions, we already incorporate any fee payments. By using end-of-day prices, we can also incorporate variable transaction costs, since end-of-day prices reflect any potential market impact by (large) institutional trades.

Since we know the TNA at the end of each month, we can add this information to the problem by restricting the corresponding market values of the fund. We denote the last day of each month as m_1 and m_2 for the first and second month of the quarter, respectively. The TNA of the third month does not add new information, since it is already incorporated in Condition (1). Denoting the TNA on day t as TNA_t, we therefore require the following condition to be met:

CONDITION 3: $\sum_{i=1}^{N} v_{i,t} = TNA_t, \forall t \in \{m_1, m_2\}$.

The last, and arguably most important, piece of information that helps to identify individual trades are the daily returns of the fund. We calculate daily returns from the net asset values (NAV) and dividend distributions that funds report at the end of each day. The NAV of a fund represents the market value of its assets at the close of the day, minus its liabilities, divided by the fund's number of outstanding shares. As such, the returns of a fund can be calculated as the relative changes in NAV with reinvested dividends. By definition, the daily returns of the fund are the same as the value-weighted average returns of the underlying holdings. Let $r_{i,t}$ represent the return of stock i on day t, including reinvested dividends. We can then express the fund return R_t on day t using the market values and returns of the underlying holdings:

CONDITION 4: $R_t = \frac{\sum_{i=1}^{N} v_{i,t-1} r_{i,t}}{\sum_{i=1}^{N} v_{i,t-1}}, \forall t \in \{2, ..., T\}$.

B. Additional structure

At this point we have developed a system of linear equations that incorporates the information of the quarterly holdings, monthly total net assets, and daily returns of the fund. Yet, the system is likely underidentified. As a first step to solve the underidentification issue, we impose realistic constraints that ensure that any solutions are as realistic as possible by prohibiting improbable trading patterns. Specifically, we restrict short positions in stocks for which the fund does not report short positions, we restrict the maximum allowed round-trip trade during the quarter, and we place an upper bound on the use of cash. Together, these restrictions greatly reduce the total number of possible trade combinations. In turn, this increases the likelihood that the optimization problem we develop later produces unique optimal solutions and converges quickly.

Although mutual funds are legally allowed to use short positions, very few mutual funds do so in practice (Almazan, Brown, Carlson, and Chapman, 2004; Nagel, 2005). Possible explanations for this phenomenon are the regulatory environment, institutional client restrictions, social stigma, high associated costs, risk management concerns, or a lack of short-selling expertise among portfolio managers (see e.g. Chen, Desai, and Krishnamurthy, 2013a; Nagel, 2005). We therefore restrict short positions when there is no clear evidence that the fund has any. That is, we only allow for short selling in a stock if the fund reported a short position in the stock at the beginning and/or end of the quarter. Hence, if the holdings for stock i are nonnegative at the start and end of the quarter, we enforce that the holdings remain nonnegative during the quarter as well:

CONDITION 5: $s_{i,t} \geq 0, \forall i \in \{1, ..., N : (s_{i,0} \geq 0 \wedge s_{i,T} \geq 0)\}, \forall t.$

In view of the low reported incidence rate of short-selling by mutual funds, we expect Condition (5) to apply to the entire portfolios of the vast majority of the fund-quarters we analyze.

Next, we restrict the number of shares that can be allocated to intra-quarter round-trip trades. Intra-quarter round-trip trades cannot be inferred from net position changes, because they are reversed again within the quarter. For example, consider a fund that buys 200 shares of a stock (starting from zero) and then sells 50 shares again before the end of the quarter. Although the fund has traded 250 shares in total, it only reports 150 shares of the stock in the subsequent quarterly filing. These intra-quarter round-trips are an important driver of underidentification in our setup, because the model can buy and sell shares as often as required to meet any restrictions. Putting a ceiling on the use of intra-quarter round-trip trades therefore greatly improves identification. We implement this restriction by assuming that the total number of buys and sells during the quarter is at most δ times the absolute net position change. This can be modelled by introducing auxiliary variables $y_{i,t}$ that capture the absolute values of the trade fractions $x_{i,t}$ for each stock i and day t:

CONDITION 6: $y_{i,t} \geq |x_{i,t}|, \forall i, \forall t.$

To retain linearity, we rewrite equation (6) using two inequality constraints:

$$y_{i,t} \geq x_{i,t}, \forall i, \forall t,$$

$$y_{i,t} \geq -x_{i,t}, \forall i, \forall t.$$

We then limit the sum of the absolute value of the trades to δ, with $\delta \geq 1$, to comply with Condition (2). This yields the following Condition for each stock i (excluding the cash position):

CONDITION 7: $\sum_{t=1}^{T} y_{i,t} \leq \delta, \forall i \in \{2, ..., N\}$.

The parameter δ can be interpreted as the maximum volume to net position change ratio we allow. A δ above one therefore allows for an intra-quarter round-trip trade up to $(\delta - 1)/2$ times the reported absolute net position change. Hence, with $\delta = 2$ and a given quarterly net position change of 100 shares, our model imputes at most 150 shares worth of buy trades and at most 50 shares worth of sell trades for that stock within the quarter ($150 + 50 \leq \delta \cdot 100$, with $\delta = 2$).

Allowing for an unlimited use of cash also contributes to the underidentification problem, since by changing the cash position, the model can scale the value-weighted average returns of the holdings to match the returns of the fund.[6] The use of cash must therefore be penalized or restricted. To this end, we assume that the maximum use of cash is a multiple of the total net use of cash over the quarter, similar to how we restrict round-trip trades for stock positions. We therefore introduce a separate parameter γ that limits the total use of cash as follows:

CONDITION 8: $\sum_{t=1}^{T} y_{1,t} \leq \gamma$.

Jointly, Conditions (7) and (8) greatly reduce the total number of nonzero variables that are allowed in solutions. For example, if we assume trades in stock and cash to sum up to at most twice the corresponding absolute net position change over the quarter, trades are 10% of the absolute net position change on average, and there are 61 trading days, this forces more than two thirds of all stock-days to zero.[7] In reality, we expect this number to be even higher. For example, the average

[6] The model can only use the cash position to make the value-weighted average returns of the holdings equal to the return of the fund if both have the same sign. In that case, increasing the cash position lowers the absolute value-weighted average return of the holdings, and vice versa.

[7] Without restrictions, the model space would be $N \times T = N \times 61$. Restricting round-trip trades with $\delta = \gamma = 2$, where trades are $\mu = 10\%$ of the absolute net position change on average, reduces the model space to $N \times (\delta/\mu) = N \times 20$. This yields a parameter reduction of $(N \times 20)/(N \times 61) - 1 \approx -67\%$.

trade is 57% of the absolute net position in our validation sample (Section II.A). Under similar assumptions, this results in a reduction in the number of parameter to be estimated of more than 94%.

C. The objective function

Despite the equality constraints from the publicly available information and the inequality constraints from the additional structure we impose, our system may still be underidentified. Yet, most funds trade positions only a couple of times per quarter, as demonstrated by the descriptive statistics in Section II.A, so the solution (i.e., the imputed set of trades by the fund) we are looking for is likely highly sparse. Therefore, we aim to identify the trades that show up most strongly and in the most robust way in our our solutions. To operationalize this notion, we develop in Section I.E a recursive learning method that iteratively solves a perturbed version of the problem. Yet, with perturbations, a solution is not guaranteed to exist. For that reason, we transform our system of linear equations into a linear program with an objective function. This objective function then serves as a residual, such that the perturbed version of the problem is very likely to be feasible.[8]

For the objective function, we isolate the two terms in Condition (4) and minimize the difference. This can be interpreted as minimizing the absolute differences between the returns of a mimicking portfolio on the one hand, determined by the imputed trades in our model, and the observed portfolio returns on the other. To operationalize this notion, we introduce auxiliary variables ϵ_t that are bounded by the absolute differences between the value of the portfolio on day $t-1$ multiplied with the portfolio return on day t, and the sum of the values of the positions on day $t-1$ multiplied with the respective stock returns on day t. Denoting the portfolio return as $r_{p,t}$, this resuls in the following condition:

CONDITION 9: $\epsilon_t \geq \left| r_{p,t} \sum_{i=1}^{N} v_{i,t-1} - \sum_{i=1}^{N} v_{i,t-1} r_{i,t} \right|, \forall t \in \{2, ..., T\}$,

where the auxiliary variables ϵ_t can be interpreted as absolute residuals. To retain linearity, we

[8] It would only be infeasible if the linear constraints we impose are mutually contradicting, for example when in a perturbation we do not allow any trading in a given stock with a non-zero position change over the whole quarter. We take care to avoid such situations. For the example given, we re-draw the perturbation when such a scenario arises.

express Condition 9 using the following two inequalities instead:[9]

$$\epsilon_t \geq + \left(r_{p,t} \sum_{i=1}^{N} v_{i,t-1} - \sum_{i=1}^{N} v_{i,t-1} r_{i,t} \right), \forall t \in \{2, ..., T\}, \tag{3}$$

$$\epsilon_t \geq - \left(r_{p,t} \sum_{i=1}^{N} v_{i,t-1} - \sum_{i=1}^{N} v_{i,t-1} r_{i,t} \right), \forall t \in \{2, ..., T\}. \tag{4}$$

Our final objective is then to minimize the sum of the absolute residuals:

CONDITION 10: $z = \sum_{t=2}^{T} \epsilon_t$.

Since the objective function is linear in $x_{i,t}$, we obtain a linear programming problem, which, if feasible, guarantees convergence to the global minimum (Dantzig, 1963).

[9]The reason we employ absolute value penalization of the residuals (ℓ_1 norm) opposed to, for example, least-squares penalization (ℓ_2 norm), is due to its computational simplicity. Whereas the ℓ_2 norm is quadratic, the ℓ_1 norm is piecewise linear. As such, we can separate the positive and negative domain of the absolute value function into separate linear conditions, as done in equations (3) and (4), and retain overall linearity in the problem. Since our problem is underidentified, we do not expect large residuals and we therefore expect similar results between minimizing the ℓ_1 or the ℓ_2 norm.

D. Final model

We now present our final model, defined by the following linear program:

$$
\begin{aligned}
\min_{x,\epsilon,y} \quad & z = \sum_{t=2}^{T} \epsilon_t & \\
\text{such that} \quad & \epsilon_t \geq + \left(r_{p,t} \sum_{i=1}^{N} v_{i,t-1} - \sum_{i=1}^{N} v_{i,t-1} r_{i,t} \right) & \forall t \in \{2, ..., T\}, \\
& \epsilon_t \geq - \left(r_{p,t} \sum_{i=1}^{N} v_{i,t-1} - \sum_{i=1}^{N} v_{i,t-1} r_{i,t} \right) & \forall t \in \{2, ..., T\}, \\
& v_{i,t} = s_{i,t} p_{i,t} & \forall i, t, \\
& s_{i,t} = s_{i,0} + (s_{i,T} - s_{i,0}) \sum_{j=1}^{t} x_{i,j} & \forall i, t, \\
& \sum_{t=1}^{T} x_{i,t} = 1 & \forall i, \\
& \sum_{i=1}^{N} v_{i,t} = TNA_t & \forall t \in \{m_1, m_2\}, \\
& y_{i,t} \geq x_{i,t} & \forall i, t, \\
& y_{i,t} \geq -x_{i,t} & \forall i, t, \\
& \sum_{t=1}^{T} y_{i,t} \leq \gamma & i = 1, \\
& \sum_{t=1}^{T} y_{i,t} \leq \delta & \forall i \in \{2, ..., N\}, \\
& s_{i,t} \geq 0 & \forall i, t : (s_{i,0}, s_{i,T} \geq 0).
\end{aligned}
\tag{5}
$$

The variables ϵ_t and $y_{i,t}$ are auxiliary variables to turn absolute values into linear expressions. The variables $v_{i,t}$ and $s_{i,t}$ are present for notational convenience, but can be substituted away. The decision space is therefore defined by $x \in \mathbb{R}^{NT}$. Despite the multitude of constraints imposed, the system is in general still underidentified. Therefore, in the next section, we adopt an iterative approach that perturbs the linear program by imposing additional (conditionally random) equality constraints.

E. An iterative solution approach

The model we define in formulation (5) is linear and bounded. Solving this problem therefore results in a single global optimum (if a solution exists, which is very likely). However, the global optimum may correspond to multiple solutions.[10] This also implies instability: small changes to the input data can lead to relatively large changes in imputed trades. To address both issues simultaneously, we propose an iterative reinforcement learning approach that in each iteration solves a randomly perturbed version of the linear programming problem. In each iteration, we create the perturbed version by setting a subset of the $x_{i,t}$ variables to zero in a conditionally random way, such that the perturbed problem is not underidentified anymore. Based on the estimated trade size of a given non-restricted stock-day $x_{i,t}$, we update the probability for setting $x_{i,t}$ to zero in the next iteration (and hence, the procedure is adaptive).[11] We call the updated probabilities the "posterior probabilities." The conditionally random nature of the procedure allows to get out of local minima. We conduct 2000 iterations of this procedure after which we typically have convergence (see the convergence graph in the Appendix Figure 5).

Since our procedure considers perturbations of the original problem and is adaptive, the solutions converge over the iterations to a set of trades that are identified in most of the perturbations considered, plus some part of the net position change that is hard to pinpoint and is split among a small number of possible trades. As such, our procedure not only provides trade estimates that are robust to small degrees of data inaccuracies and model-misspecification, but also provides an indication of statistical significance of each estimate.

Without any further adjustments, our approach as described above would encounter problems with respect to convergence. In each perturbation, many of the non-restricted $x_{i,t}$s would get a value close to, but slightly different from zero, while only few of those would equal zero exactly. A large fraction of these non-zero values is likely to reflect noise induced by the random perturbations. This noise makes it hard for the procedure to update the posterior probability for a given variable towards zero and may stand in the way of (rapid) convergence. Therefore, we shrink small trades to zero for the purpose of updating our posterior probabilities. This shrinkage procedure promotes

[10] I.e., different combinations of trades may lead to the same solution.

[11] We do not update these probabilities for day-trades that are set to zero in an iteration. This is one of the reasons why the algorithm takes some time to converge.

sparse solutions to our linear program.

At this stage, we note that looking for a sparse solution of our linear programming problem (5) is equivalent to a multi-armed bandit problem (MAB). In a multi-armed bandit problem, one is trying to maximize expected revenue from playing a selection from a battery of possibly heterogeneous slot-machines (each slot-machine is also called a one-armed bandit). This requires deciding on the selection of slot machines to play. The typical approach is to select a sub-set of all available slot-machines, play them, update the selection of slot machines according to the outcome, and iterate to convergence. In our setting, variables to be selected correspond to slot machines and whether the linear program imputes a trade (larger than the shrinkage threshold) on a given stock-day or not corresponds to a (binary) slot machine reward.

Due to the equivalence with the multi-armed bandit problem, we can draw on the literature on the MAB problem for solution methods. The MAB is a classic reinforcement learning problem that has been studied widely. Starting with the seminal papers by Robbins (1952) and Gittins (1979), research into extensions and improvements of bandit problems is still very active, especially in the booming machine learning literature (see e.g. Lattimore and Szepesvári, 2020). The classic variant of the MAB problem, in which the player selects one slot machine at a time, can be solved quickly and efficiently (Slivkins et al., 2019). We face a more difficult variant of the MAB problem, however, that relaxes several assumptions of the classic MAB problem. First, we wish to select multiple variables each round. This makes our problem equivalent to a multi-play multi-armed bandit problem. Second, the number of variables we select is allowed to be different each round. This makes it equivalent to a combinatorial bandit problem (Chen, Wang, and Yuan, 2013b; Cesa-Bianchi and Lugosi, 2012). Third, the rewards we have in each round are binary (above the regularization threshold or not), local (different per arm), and dependent, since if one influential variable is excluded from the model the closest alternative is more likely to be chosen. This makes it a stochastic combinatorial bandit problem. Therefore, the algorithm we require to solve our problem needs to be able to stochastically explore the variable space such that it can escape local optima.

The only efficient algorithm that meets our requirements is based on (recently rediscovered) Thompson sampling (Thompson, 1933). Thompson sampling (TS) combines elements of random sampling and Bayesian updating to minimize regret, which is defined as the total loss by not taking the right actions. Although sporadically used in machine learning, it was only when Agrawal and Goyal (2012) published their seminal paper that the use of TS really took off. They show that TS provides a very efficient balance between the exploration and exploitation of potential solutions.[12] Liu and Ročková (2021) show that TS has properties that make it perfectly suitable for variable selection in highly dimensional nonparametric problems, exactly meeting our requirements.[13]

Following Liu and Ročková (2021), we define for each variable a prior that models the probability that a variable (a stock-day) is selected (not restricted) in the linear programming problem. Thompson sampling assumes that the (binary) outcome of an experiment (in our case, whether imputed trades are larger than the shrinkage threshold) depends on a conditionally-random Bernoulli trial. We therefore use the Beta distribution, which is the conjugate prior to the Bernoulli distribution, as our prior distribution. Specifically, let $a_{i,t}^{(k)}$ and $b_{i,t}^{(k)}$ be the hyperparameters of the prior distribution $\text{Beta}(a_{i,t}^{(k)}, b_{i,t}^{(k)})$ of stock i on day t in iteration k of our algorithm. We start our algorithm with the flat priors $a_{i,t}^{(0)} = b_{i,t}^{(0)} = 1$, such that every stock-day has the same probability of being included. We use λ to represent the regularization threshold parameter below which we shrink small trades to zero. In each iteration k, we get for each stock-day a random draw from the corresponding prior distribution and include the stock-day if the corresponding random draw exceeds $\frac{1}{2}$.[14] We now employ the following routine:

Reinforcement learning with Thompson sampling

For each iteration $k = 1, ..., 2000$:

Step 1. Sample $\theta_{i,t} \sim \text{Beta}\left(a_{i,t}^{(k)}, b_{i,t}^{(k)}\right) \forall i, \forall t.$

Step 2. Add the following restrictions to the LP problem in (5):

$$x_{i,t} = 0 \quad \forall i, \forall t : \theta_{i,t} < \tfrac{1}{2}. \tag{6}$$

Step 3. Solve the restricted LP problem to optimality. If the problem is infeasible, return to Step

[12] Agrawal and Goyal (2012) were the first to show that TS achieves logarithmic regret with respect to time. This is the reason for the efficient balance between the exploration and exploitation of the TS algorithm.

[13] By extending the work of Wang and Chen (2018), Liu and Ročková (2021) derive regret bounds for combinatorial multi-armed bandits in which outcomes may be related. They find that TS achieves linear regret with respect to the number of variables if the separation between correct and the incorrect variables is large enough (surpassing 50% probability). Additionally, they show that TS is consistent in selecting the right variables in probability.

[14] This implies $C = (\sqrt{5} - 1)/2$ in the computational oracle of Liu and Ročková (2021).

1 and re-sample $\theta_{i,t}$ $\forall i, t$.

Step 4. Inspect the optimal solutions $x^*_{i,t}$ $\forall i, \forall t$, and update the posterior distributions:

$$a_{i,t}^{(k+1)} = a_{i,t}^{(k)} + \mathbb{1}\{|x^*_{i,t}| \geq \lambda\} \qquad \forall i, \forall t : \theta_{i,t} \geq \tfrac{1}{2}, \tag{7}$$

$$b_{i,t}^{(k+1)} = b_{i,t}^{(k)} + \left(1 - \mathbb{1}\{|x^*_{i,t}| \geq \lambda\}\right) \qquad \forall i, \forall t : \theta_{i,t} \geq \tfrac{1}{2}, \tag{8}$$

where $\mathbb{1}\{x \geq \lambda\}$ is an indicator function that equals 1 if $x \geq \lambda$ and 0 otherwise.

Update the maximum allowed use of cash by recalculating the γ parameter:

$$\gamma = \sum_{t=1}^{T} |x^*_{1,t}|. \tag{9}$$

In Step 4 of the Thompson sampling routine, we update the constraint on the cash position. Since the cash position seems to converge more quickly than the estimates for the other positions, we start out with a very loose constraint on cash and tighten it quickly. This is done by restricting the maximum use of cash in future iterations to be less than or equal to the imputed use of cash in previous iterations.[15]

The updated parameters of the Beta distribution in Eqns. (7) and (8) allow us to calculate the posterior inclusion probabilities $\pi_{i,t}^{(k+1)}$ in Step 1 of the subsequent iteration as

$$\pi_{i,t}^{(k+1)} = \frac{a_{i,t}^{(k+1)}}{a_{i,t}^{(k+1)} + b_{i,t}^{(k+1)}}, \quad \forall i, \forall t. \tag{10}$$

After solving the LP problem 2000 times, we obtain the solution from the final iteration. Consistent with the threshold of $\frac{1}{2}$ for sampling $\theta_{i,t}$, we then solve the LP problem one last time including only trades with a posterior inclusion probability above 50%. It is not guaranteed, however, that every position has at least one trade with a posterior inclusion probability above 50%. For example, the model may not be able to clearly identify any trades for a given position if the net position change only represent a tiny fraction of total fund value. In that case, we include for that position only the $x_{i,t}$ with the highest posterior inclusion probability. The optimal solution we get out of this

[15]We would have preferred to use a Bayesian updating scheme to update the maximum allowed use of cash in each iteration. The inequality constraint on γ (Condition 8) makes this very difficult, however, since it constraints the maximum allowed use of cash. This violates Bayes' rule, since we cannot update our posterior distribution if the observed data is truncated.

last pass of solving our problem yields our point estimates for each trade. The posterior inclusion probabilities of the last iteration (i.e., when $k = 2000$) tell us how successfully we were able to identify each trade. We refer to these as "inclusion probabilities." These inclusion probabilities act as proxies for significance. Appendix B provides an illustration of how the algorithm works through the iterations and converges on a simplified portfolio.

II. Validation

In this section, we validate the approach we propose for imputing fund trades from data available at the regulatory reporting frequencies. We validate our approach in two ways. First, we collect proprietary trading and return data from an anonymous, large, European asset management firm on their active funds with fundamental mandates. These data we merge with the daily return data of the underlying holdings. We then apply our approach and compare the imputed trades with the actual ones. Second, we use these data to simulate trades and positions for hypothetical funds. Since we simulate, we can generate an arbitrary number of observations and therefore obtain more power for assessing our estimation method. Moreover, we can study the effects of parameter changes in a perfectly controlled setting. This allows us to investigate the external validity and robustness of our approach. Below, we describe our validation exercise in more detail.

A. Validation data

For our validation study, we obtain proprietary trade, holdings, size, and return data for all active funds with fundamental mandates of an anonymous, large, European asset management firm. The data cover 36 funds over a period of 1 year (Q1 to Q4 of a recent year) with a total of 136 fund-quarters. Table I presents summary statistics of these funds (variable definitions are in Appendix Table V).

[Place Table I about here]

These funds are on average large (about $1 bln in assets under management, AUM), turn over their portfolios about once per year, and hold and trade a limited number of positions. Any quarterly net position change takes on average about 3 days to be traded, which corresponds to about one trade per month. Most net position changes are very small rebalancing trades (0-10 bps of fund size). Yet, there is also a sizable number of larger net position changes (on average about 10 per fund quarter exceeding 50 bps of fund size). It is the trades associated with these larger net position changes that matter most and that our algorithm is likely to pick up. On average 17% of net position changes involve round-trip trades, but this percentage varies widely across fund quarters. One concern could be that funds do round-trip trades in stocks for which they have zero holdings at the beginning and end of the quarter. This issue turns out to be moot (on average only 0.15% of positions falls into this category).

We merge the holdings, trade, and fund data with daily return data of the portfolio constituents. Since these funds are globally active, we merge with daily return data from Refinitiv Eikon.

Our validation data can be directly used to assess the performance of our trade imputations. Moreover, we use these data as a basis for simulations. In the next subsection, we describe how we set up our simulation study.

B. *Simulation study*

To create a wider variety of validation data than our proprietary fund data allows for, we conduct a simulation study. We start by randomly drawing a number of positions K ranging from 10 to 150 according to a uniform distribution. Next, we randomly draw a quarter Q from the four quarters in our fund data. Within this quarter, we then randomly select a fund from which we copy the daily cash position over the quarter. Subsequently, we also randomly draw K unique stocks and the associated trades from all the funds with data in this quarter (without replacement). Note that these trades could (and are likely to) be from different funds. We then scale all positions and net position changes by a constant such that portfolio weights add up to one. We simulate a hypothetical dataset of 5000 of such fund-quarters. This approach is reminiscent of the historical simulation approach for calculating value-at-risk for trading portfolios (see e.g., Hull, 2012).

Table I shows the summary statistics of our simulated dataset. As one would expect, the simulated dataset is close to the dataset that it is derived from. As the average number of positions is slightly higher than in the validation data, the average turnover, net position changes, and number of trades are also slightly higher.

C. Validation results

In this section, we present results of our validation exercises. We do this by assessing the identification accuracy of trade incidence, the bias and variance of trade size estimates, and the average active share of our estimates relative to the actual/simulated trades. In addition, we also benchmark our method to commonly used flow and trade timing assumptions used in the literature.

C.1. Trade incidence results

Our first set of validation results relate the ability of our procedure to identify trade incidence. Note that most funds only trade sporadically (see Table 1), such that this analysis is meaningful and not trivial. We present the main validation results of the discriminatory ability of our model in Panel A of Table II.

[Place Table II about here]

The results in Table II are weighted by the size of the net position change relative to average fund value over the quarter (we call this "turnover-weighted").[16] The results in Table II are derived from our simulated dataset. Results derived from the actual validation sample are presented in Appendix Tables VII and VIII (turnover- and equally-weighted respectively).

We split the results for all of our valuation metrics by size of the net position change (relative to average fund value over the quarter) as we expect our model to be better able to identify trades that are part of relatively large position changes. Moreover, we consider the ability of our procedure to pinpoint a trade at exactly the correct day, but also consider windows of one, two, or three days before and after the actual day a stock was traded. These windows are particularly relevant given the international nature of the funds in our sample as a result of differences in time zones.

The first and most comprehensive evaluation metric is the area under the receiver operator curve (AUC). We present the ROC curves for the simulation in Figure 1. One can interpret the AUC of our model as the probability that an average stock-day with a trade is ranked higher in terms of the posterior inclusion probability than an average stock-day without a trade. Hence, an AUC of 0.5 corresponds to a completely uninformative model, whereas an AUC of 1 to a perfectly informative

[16] Equally weighted results are presented in Appendix Table VI.

model. A benefit of looking at the AUC is that it does not require any decision on a threshold on the posterior inclusion probability to classify a stock-day as having a trade or not (which in the end depends on the end-user's aversion of type I relative to type II errors). As a rule of thumb, an AUC between 0.5 and 0.7 has discriminatory power but is for most uses inadequate, between 0.7 and 0.8 is considered good and above 0.8 considered very good to excellent (see Hosmer Jr., Lemeshow, and Sturdivant, 2013, p. 177).

[Place Figure 1 about here]

As one can see, the model has discriminatory power for the whole simulated sample to pinpoint trades to specific days, but this power for the whole sample would not be adequate for most uses. Yet, when we focus on the larger trades, or when we allow for a window of one day earlier or later (e.g., due to the international nature of the fund data the simulated sample was derived from), our model performs well to very well. Note that for many uses, larger portfolio moves are more important and having a tolerance for a day earlier or later is common (e.g., in event studies).

Another common way in finance to present classification success is by presenting graphs depicting cumulative accuracy profiles (CAP curves). This is done by e.g., credit rating agencies to show the performance of their credit ratings in predicting defaults. A CAP curve plots the cumulative rank of the fraction of identified positives (y-axis) against the cumulative rank of posterior inclusion probability. The higher the CAP curve, the more accurate the classification. A 45 degrees diagonal CAP curve corresponds to an uninformative model. Figure 2 shows CAP curves for the simulated data and Appendix Figure 8 for the actual validation sample. One sees that the curves increase quite steeply in the beginning, indicating that the model captures certain, strongly present trades very accurately. One can also see that the model performs better with net position changes that are relatively larger, consistent with the AUC results in Table II.

[Place Figure 2 about here]

If one wants to use the model results to e.g., analyze fund manager behavior w.r.t. timing of trades, one would need to classify estimated trade incidence based on the posterior inclusion probability, by comparing the posterior inclusion probability to a threshold. The choice of this threshold depends on the relative aversion to type I vs type II errors. We would argue that the typical user would need to classify trades precisely (i.e., be very sure about the stock-days classified as a trade), and would be more tolerant towards missing stock-days with trades. To this end, we present in Table II standard classification measures for trades classified with a threshold on the posterior inclusion probability of 0.99. Obviously, this choice favors a lower false positive rate at the expense of a higher false negative rate. If the false negative rate is deemed too high, one can pick a lower threshold, e.g., 0.95 or 0.9 (but of course the false positive rate will increase).

The first metric considered here is precision. This defined as the rate of true positives over all positive predictions. As one can see, the precision for our model is high especially if one allows for a window of plus or minus one day (in which case 80% to 90% of identified trades indeed correspond to an actual trade). As precision is important for the subsequent use of our trade estimates, we present a heatmap of our model's precision as a function of the number of positions in the portfolio and minimum trade size considered in Figure 3. As expected, the precision is particularly high in more concentrated portfolios and for identifying trades as part of larger position changes.

[Place Figure 3 about here]

The next statistic is the Negative Predictive Value (NPV). It is defined as the true negative rate over all negative predictions. As one can see, our model predicts days without a trade very well.

Given that we chose a very high threshold to classify a stock-day as a trade, we expect to have relatively many false negatives. This is reflected by the rather low sensitivity (defined as the fraction of all trades correctly classified as a trade) of our approach with the chosen threshold level. Since larger trades are easier to classify, we see a higher sensitivity for larger trades.

Finally, we look at specificity, which is defined as the fraction of correctly identified negatives. As our model yields very few false positives, this rate is very high.

C.2. Trade size results

Given a choice of threshold for trade incidence classification, one would also like to know whether trade sizes are correctly identified. Panel B of Table II presents the the bias and variance of trade size estimates for correctly classified trades with a classification threshold of 0.99.[17] Since our

[17]One can only do this analysis for correctly identified trade incidences.

algorithm uses a shrinkage procedure, it biases against finding small trades. Moreover, since the shrinkage is less likely to affect large trades compared to smaller trades, we see that this bias shrinks with trade size. In general, the variance of our estimates is small and shrinks further as trade size increases. Hence, the trade size results indicate a good performance of our method.

C.3. Active shares

In this section, we investigate the extent to which imputed positions deviate compared to the actual ones. We also benchmark against three alternative ways of allocating net position changes used in the literature (assumed to have all taken place at the start, middle, or end of the quarter).[18] Assuming that all net position changes only realized at the end of the quarter is the approach most frequently adopted.

To measure the degree of deviation, we use the active share measure, developed by Cremers and Petajisto (2009) to detect the degree of active portfolio management (a higher active share in our context corresponds to poorer fit). Yet, instead of computing active share as deviation of fund holdings from benchmark index holdings, we calculate it as a deviation of imputed fund holdings from actual fund holdings, averaged over the fund-quarter. Denoting the imputed and actual weights as $w^*_{i,t}$ and $w_{i,t}$, respectively, we calculate the average active share (AS) as follows:

$$\text{AS} = \frac{1}{2T}\sum_{t=1}^{T}\sum_{i=1}^{N}\left|w^*_{i,t} - w_{i,t}\right| \tag{11}$$

The day-by-day positions are constructed from the trade estimates in the last iteration of our Thompson sampling procedure. Hence, there is no further filtering by comparing the posterior inclusion probability to a classification threshold.[19]

Since only a relatively small fraction of the portfolio is turned over during each quarter and since the portfolio holdings at the start and end of each quarter are available, we expect active shares to be small in nominal terms. This is indeed confirmed in Panel A of Table III. All three methods achieve a low active share, but our method tends to generate a much lower active share

[18] Note that assuming that all trades happen in the middle of the quarter requires more data and work to construct than assuming that all trades happen at the start or end of the quarter and is therefore (much) less often used (one needs to match all portfolio positions with daily stock returns).

[19] If one were to do so, trades would not necessarily add up to net position changes anymore.

(and hence better fit) than the other, simpler methods.

Panel B of Table III presents the distribution of relative improvements of our method compared to the other methods. As one can see, the improvement is almost exclusively positive. The only exception is that the middle method in a small number of cases is more accurate than ours.

[**Place Table III about here**]

In Table IV we investigate the relationship of our method's outperformance w.r.t. number of positions by regressing outperformance on the log of number of positions in a portfolio. We expect a negative relationship because it is more difficult for our method to pinpoint trades with a higher number of positions. As one can see, both for the simulated as well as for the actual sample, this relationship is indeed negative and strongly statistically significant.

[**Place Table IV about here**]

Panels A and and B of Figures 4 presents the result of Table IV graphically. These show our active share improvement versus the other methods considered depends on the number of positions in the portfolio (plotted as a fitted logarithmic function of the number of trades per quarter) for our validation data and simulated data, respectively. Consistent with Tables III and IV, our method performs much better than allocating the whole net position change to the start or end of the quarter throughout the range of number of positions in a portfolio that we consider. Our method also performs (on average) better than allocating to the middle of the quarter unless the number of positions in a portfolio is very large.

Finally, comparing Panel A to Panel B of Figure 4 shows the value of doing the simulations. Since the number of fund quarters is much higher, average active shares can be estimated more precisely, leading to tighter confidence bounds. Moreover, our simulation allows us to create much more variation in the number of trades because of which we can estimate our average active share curves on a much more balanced dataset.

III. Conclusion

In this paper, we have shown how mutual fund trades can be imputed from standard data sources at the regulatory frequencies. While our approach leads to an underidentified system of equations, we manage to obtain accurate estimates by imposing additional constraints and using an advanced reinforcement learning technique. Not only does our method provide estimates of daily fund trades at the position level, our method also provides confidence levels of these estimates.

Ours paper provides many new avenues for further research, both with respect to the method itself as well as what can be done with it. The method that is currently presented provides a lower bound on what is possible. We plan to further investigate the method's accuracy drivers and optimize the estimation parameters. As our method involves shrinkage, estimates may contain a bias. By furthering our understanding in when the method performs well, we can try to predict any potential biases and correct for it.

After the disappearance of the Ancerno data, this method re-enables researchers with standard data subscriptions to do detailed and precise studies on fund behavior and the implications of fund decisions on performance and will thereby hopefully give a boost to the academic research efforts on mutual funds.

REFERENCES

Agarwal, Vikas, Gerald D. Gay, and Leng Ling, 2014, Window dressing in mutual funds, *Review of Financial Studies* 27, 3133–3170.

Agrawal, Shipra, and Navin Goyal, 2012, Analysis of thompson sampling for the multi-armed bandit problem, in *Conference on Learning Theory*, 39–1, Journal of Machine Learning Research Workshop and Conference Proceedings.

Almazan, Andres, Keith C Brown, Murray Carlson, and David A. Chapman, 2004, Why constrain your mutual fund manager?, *Journal of Financial Economics* 73, 289–321.

Ben-Tal, Aharon, Laurent El Ghaoui, and Arkadi Nemirovski, 2009, *Robust optimization*, volume 28 (Princeton university press).

Berk, Jonathan B., and Jules H. Van Binsbergen, 2015, Measuring skill in the mutual fund industry, *Journal of Financial Economics* 118, 1–20.

Brown, Keith C., W. Van Harlow, and Laura T. Starks, 1996, Of tournaments and temptations: An analysis of managerial incentives in the mutual fund industry, *Journal of Finance* 51, 85–110.

Busse, Jeffrey A., Tarun Chordia, Lei Jiang, and Yuehua Tang, 2021, Transaction costs, portfolio characteristics, and mutual fund performance, *Management Science* 67, 1227–1248.

Campbell, John Y., Tarun Ramadorai, and Allie Schwartz, 2009, Caught on tape: Institutional trading, stock returns, and earnings announcements, *Journal of Financial Economics* 92, 66–91.

Cesa-Bianchi, Nicolo, and Gábor Lugosi, 2012, Combinatorial bandits, *Journal of Computer and System Sciences* 78, 1404–1422.

Chakrabarty, Bidisha, Pamela C. Moulton, and Charles Trzcinka, 2017, The performance of short-term institutional trades, *Journal of Financial and Quantitative Analysis* 52, 1403–1428.

Chan, Louis K.C., and Josef Lakonishok, 1993, Institutional trades and intraday stock price behavior, *Journal of Financial Economics* 33, 173–199.

Chan, Louis K.C., and Josef Lakonishok, 1995, The behavior of stock prices around institutional trades, *Journal of Finance* 50, 1147–1174.

Chen, Honghui, Hemang Desai, and Srinivasan Krishnamurthy, 2013a, A first look at mutual funds that use short sales, *Journal of Financial and Quantitative Analysis* 48, 761–787.

Chen, Wei, Yajun Wang, and Yang Yuan, 2013b, Combinatorial multi-armed bandit: General framework and applications, in *International conference on machine learning*, 151–159, Proceedings of Machine Learning Research.

Chiyachantana, Chiraphol N., Pankaj K. Jain, Christine Jiang, and Robert A. Wood, 2004, International evidence on institutional trading behavior and price impact, *Journal of Finance* 59, 869–898.

Coval, Joshua, and Erik Stafford, 2007, Asset fire sales (and purchases) in equity markets, *Journal of Financial Economics* 86, 479–512.

Cremers, K.J. Martijn, and Antti Petajisto, 2009, How active is your fund manager? A new measure that predicts performance, *Review of Financial Studies* 22, 3329–3365.

Dantzig, George, 1963, *Linear programming and extensions* (Princeton University Press).

Elton, Edwin J., Martin J. Gruber, Christopher R. Blake, Yoel Krasny, and Sadi O. Ozelge, 2010, The effect of holdings data frequency on conclusions about mutual fund behavior, *Journal of Banking & Finance* 34, 912–922.

Fama, Eugene F., and Kenneth R. French, 2010, Luck versus skill in the cross-section of mutual fund returns, *Journal of Finance* 65, 1915–1947.

Farrell, Michael, 2018, Read between the filings: Daily mutual fund holdings and liquidity provision, *Available at SSRN: https://dx.doi.org/10.2139/ssrn.3241664* .

Gibson, Scott, Assem Safieddine, and Sheridan Titman, 2000, Tax-motivated trading and price pressure: An analysis of mutual fund holdings, *Journal of Financial and Quantitative Analysis* 35, 369–386.

Gittins, John C., 1979, Bandit processes and dynamic allocation indices, *Journal of the Royal Statistical Society: Series B (Methodological)* 41, 148–164.

Grinblatt, Mark, Sheridan Titman, and Russ Wermers, 1995, Momentum investment strategies, portfolio performance, and herding: A study of mutual fund behavior, *American Economic Review* 1088–1105.

Hau, Harald, and Sandy Lai, 2017, The role of equity funds in the financial crisis propagation, *Review of Finance* 21, 77–108.

Hong, Joey, Branislav Kveton, Manzil Zaheer, and Mohammad Ghavamzadeh, 2022, Hierarchical bayesian bandits, in *International Conference on Artificial Intelligence and Statistics*, 7724–7741, Proceedings of Machine Learning Research.

Hosmer Jr., David W., Stanley Lemeshow, and Rodney X. Sturdivant, 2013, *Applied logistic regression*, volume 398 (John Wiley & Sons).

Hu, Gang, Koren M. Jo, Yi Alex Wang, and Jing Xie, 2018, Institutional trading and abel noser data, *Journal of Corporate Finance* 52, 143–167.

Huang, Jennifer, Clemens Sialm, and Hanjiang Zhang, 2011, Risk shifting and mutual fund performance, *Review of Financial Studies* 24, 2575–2616.

Hull, John, 2012, *Risk management and financial institutions* (John Wiley & Sons).

ICI, 2022, *2022 Investment Company Fact Book*, volume 62 (Investment Company Institute), Available at `https://www.icifactbook.org/` (Accessed: April 15, 2023).

Jiang, George, Tong Yao, and Tong Yu, 2007, Do mutual funds time the market? Evidence from portfolio holdings, *Journal of Financial Economics* 86, 724–758.

Jiang, Hao, Marno Verbeek, and Yu Wang, 2014, Information content when mutual funds deviate from benchmarks, *Management Science* 60, 2038–2053.

Ke, Bin, and Santhosh Ramalingegowda, 2005, Do institutional investors exploit the post-earnings announcement drift?, *Journal of Accounting and Economics* 39, 25–53.

Lakonishok, Josef, Andrei Shleifer, and Robert W. Vishny, 1992, The impact of institutional trading on stock prices, *Journal of Financial Economics* 32, 23–43.

Lattimore, Tor, and Csaba Szepesvári, 2020, *Bandit algorithms* (Cambridge University Press).

Liu, Yi, and Veronika Ročková, 2021, Variable selection via thompson sampling, *Journal of the American Statistical Association* 1–18.

Nagel, Stefan, 2005, Short sales, institutional investors and the cross-section of stock returns, *Journal of Financial Economics* 78, 277–309.

Puckett, Andy, and Xuemin Yan, 2011, The interim trading skills of institutional investors, *Journal of Finance* 66, 601–633.

Robbins, Herbert, 1952, Some aspects of the sequential design of experiments, *Bulletin of the American Mathematical Society* 58, 527–535.

Sirri, Erik R., and Peter Tufano, 1998, Costly search and mutual fund flows, *Journal of Finance* 53, 1589–1622.

Slivkins, Aleksandrs, et al., 2019, Introduction to multi-armed bandits, *Foundations and Trends® in Machine Learning* 12, 1–286.

Thompson, William R., 1933, On the likelihood that one unknown probability exceeds another in view of the evidence of two samples, *Biometrika* 25, 285–294.

Wang, Siwei, and Wei Chen, 2018, Thompson sampling for combinatorial semi-bandits, in *International Conference on Machine Learning*, 5114–5122, Proceedings of Machine Learning Research.

Wermers, Russ, 1999, Mutual fund herding and the impact on stock prices, *Journal of Finance* 54, 581–622.

WFE, 2022, *FY 2021 Market Highlights Report* (World Federation of Exchanges), Available at `https://www.world-exchanges.org/our-work/articles/fy-2021-market-highlights-report` (Accessed April 15, 2023).

IV. Tables and figures

Table I. Descriptive statistics of actual and simulated samples. In this table, we show descriptive statistics of our sample of 136 fund-quarters that we sourced from a European asset manager and our simulated data that is constructed by forming 5000 new portfolios by randomly selecting a quarter and randomly selecting between 10 and 150 positions from the actual sample of funds in that quarter. We present the variable definitions in Table V in the appendix. For each variable, we report the average, median, 5th and 95th percentile of each variable for both the actual and simulated samples.

	Actual				Simulated			
	Avg.	Median	P5	P95	Avg.	Median	P5	P95
AuM ($B)	0.99	0.43	0.02	3.87	1.26	1.22	0.28	2.37
Quarterly turnover (%)	26.10	20.20	6.26	72.98	26.29	25.92	16.34	38.14
Positions	59.99	54.50	42.00	90.75	72.33	70.00	15.00	137.00
Traded positions	40.21	41.00	11.75	70.25	49.42	47.00	10.00	96.00
Position turnover (%)	0.58	0.48	0.23	1.19	0.73	0.47	0.23	2.13
between $0-25$ bps	17.79	14.00	1.00	47.25	26.82	22.00	1.00	66.00
between $25-50$ bps	7.70	6.50	1.00	20.25	9.17	9.00	1.00	19.00
between $50-100$ bps	7.40	6.00	1.00	15.25	7.68	7.00	1.00	15.00
between $100-250$ bps	6.41	5.00	0.00	24.00	4.71	4.00	1.00	9.00
above 250 bps	0.90	0.00	0.00	3.00	1.05	0.00	0.00	4.00
Trades	133.81	98.00	20.00	446.75	162.14	153.00	31.00	332.00
Trades per traded position	2.95	2.47	1.12	6.18	3.24	3.17	2.39	4.31
Fraction net volume	0.57	0.50	0.22	1.03	0.48	0.45	0.30	0.75
Sum of absolute fractions	1.44	1.09	1.00	2.61	1.55	1.42	1.08	2.40
HHI (normalized %)	0.52	0.52	0.13	0.92	0.48	0.47	0.34	0.61
With roundtrip (%)	17.17	5.77	0.00	88.39	17.08	16.98	7.14	27.67
Unobserved (%)	0.15	0.00	0.00	1.54	0.15	0.00	0.00	1.09
New positions (%)	6.20	4.55	0.00	17.21	6.37	6.09	0.00	12.70
Liquidated positions (%)	6.17	4.44	0.00	16.33	6.24	5.97	0.00	12.50
Cash position (%)	2.27	2.29	0.40	4.21	2.81	1.75	0.32	9.24
Funds	36				-			
Observations	136				5.000			
Total number of trades	18.198				810.679			

Table II. Trade identification results on simulated data (turnover-weighted). In this table, we summarize the ability of our model to predict trade incidence and size in our simulated dataset. The estimates are generated using model parameters $\delta = 1$ (maximum traded shares per stock is equal to the net position change over the quarter) and $\lambda = 0.10$ (estimated trade fractions below 0.10 are regularized to zero). All observations are weighted by the position turnover (net position change divided by average portfolio value over the quarter) of the corresponding position. The predictive power of trade incidence is summarized using the area under the receiver operating characteristic curve (AUC). The AUC can be interpreted as the probability that our model ranks a random observation with a trade higher than a random observation without a trade. We also show the precision (Prec.), negative predictive value (NPV), sensitivity (Sens.) and specificity (Spec.) for a posterior inclusion probability threshold of 0.99. The last two columns shown with the bias and variance of estimated trade size for correctly identified trade incidences. The results are split out along two dimensions. First, we show results separately for all trades, and for trades where the net position turnover is at least 25, 50, 100 or 250 basis points (bps). Second, we show results on the same day as, and within one, two and three-day intervals of trade estimates.

	Trade incidence					Trade size	
	AUC	Prec.	NPV	Sens.	Spec.	Bias	Variance
All trades							
Same day	0.66	0.66	0.94	0.03	1.00	0.04	0.06
One day interval	0.71	0.80	0.94	0.04	1.00	0.02	0.06
Two day interval	0.74	0.85	0.94	0.04	1.00	0.02	0.05
Three day interval	0.76	0.88	0.94	0.05	1.00	0.01	0.05
25+ bps							
Same day	0.67	0.66	0.94	0.03	1.00	0.04	0.06
One day interval	0.73	0.80	0.94	0.04	1.00	0.02	0.06
Two day interval	0.76	0.85	0.94	0.05	1.00	0.02	0.05
Three day interval	0.78	0.88	0.93	0.05	1.00	0.01	0.05
50+ bps							
Same day	0.69	0.67	0.93	0.03	1.00	0.04	0.06
One day interval	0.75	0.81	0.93	0.05	1.00	0.02	0.05
Two day interval	0.78	0.86	0.93	0.05	1.00	0.02	0.05
Three day interval	0.80	0.89	0.93	0.06	1.00	0.01	0.05
100+ bps							
Same day	0.72	0.70	0.93	0.05	1.00	0.04	0.05
One day interval	0.78	0.83	0.93	0.06	1.00	0.02	0.05
Two day interval	0.81	0.89	0.94	0.08	1.00	0.01	0.04
Three day interval	0.83	0.91	0.94	0.09	1.00	0.01	0.04
250+ bps							
Same day	0.77	0.75	0.94	0.08	1.00	0.03	0.04
One day interval	0.82	0.88	0.94	0.12	1.00	0.02	0.04
Two day interval	0.85	0.92	0.94	0.14	1.00	0.01	0.03
Three day interval	0.86	0.94	0.94	0.16	1.00	0.00	0.03

Figure 1. ROC curves for simulated data. In this figure, we show the receiver operating characteristic curves (ROC curves) when predicting trade incidence on the simulated sample. The ROC curve shows the trade-off between the false positive rate (x-axis) and the true positive rate (y-axis; also known as sensitivity) for different inclusion probability cutoff points (π). The area under the ROC curve (AUC) can be interpreted as the probability that our model ranks a random observation with a trade higher than a random observation without a trade. We show ROC curves when predicting trade incidence on the same day, and within one-, two- and three-day intervals of the estimated trades. The estimates are generated using model parameters $\delta = 1$ (maximum traded shares per stock is equal to the net position change over the quarter) and $\lambda = 0.10$ (estimated trade fractions below 0.10 are regularized to zero). All observations are weighted by the position turnover. We show the ROC curves separately for all trades, and for trades where the position turnover is at least 50, 100 or 250 basis points (bps).

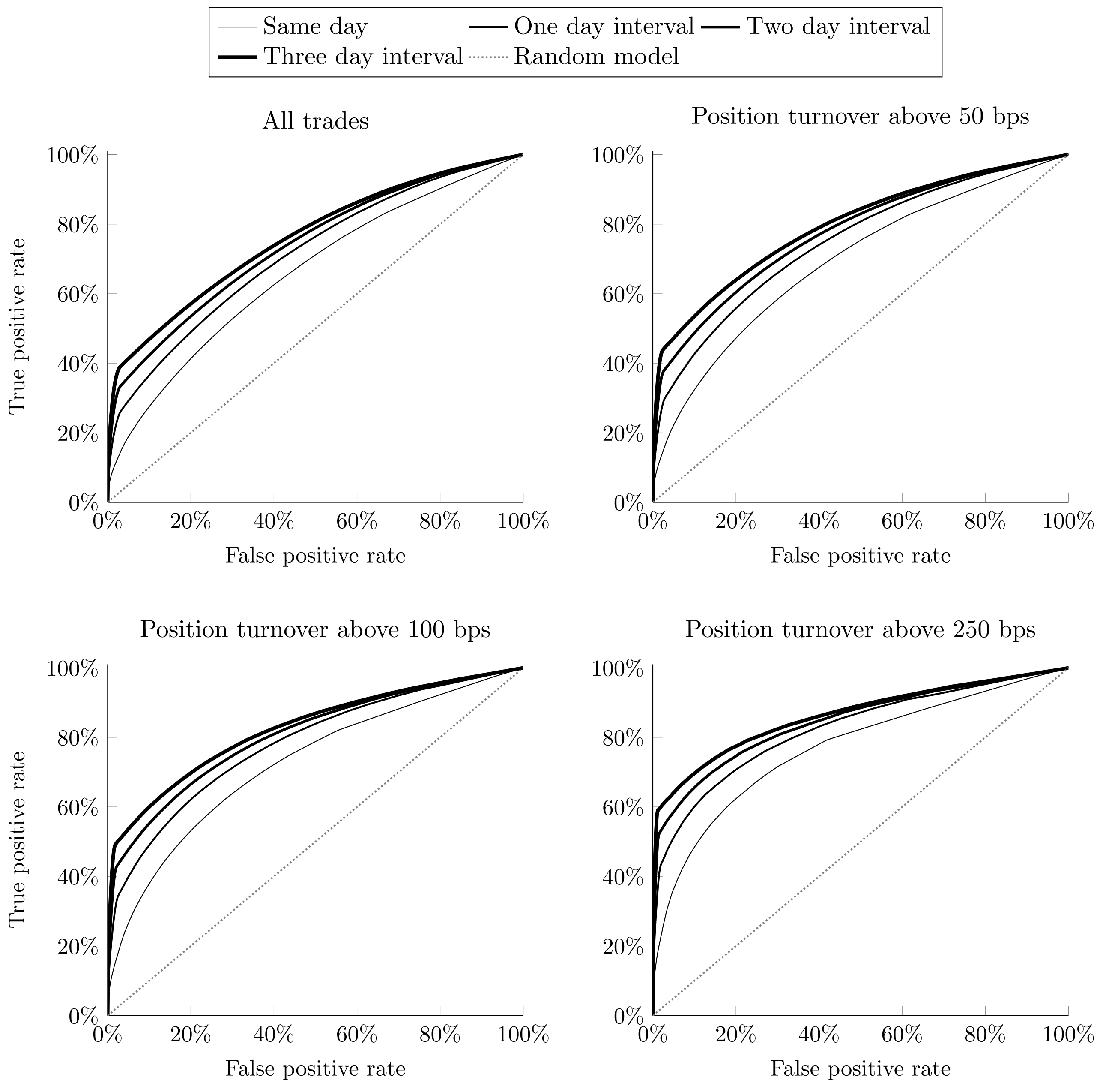


Figure 2. CAP curves for simulated data. In this figure, we show the cumulative accuracy profile curves (CAP curves) when predicting trade incidence on the simulated sample. The CAP curve represents the cumulative fraction of correctly detected trades (y-axis), when ranking the trade estimates from our model from highest to lowest inclusion probability (x-axis). We show CAP curves when predicting trade incidence on the same day, and within one-, two- and three-day intervals of the estimated trades. The estimates are generated using model parameters $\delta = 1$ (maximum traded shares per stock is equal to the net position change over the quarter) and $\lambda = 0.10$ (estimated trade fractions below 0.10 are regularized to zero). All observations are weighted by the position turnover of the position. We show the CAP curves separately for all trades, and for trades where the position turnover is at least 50, 100 or 250 basis points (bps).

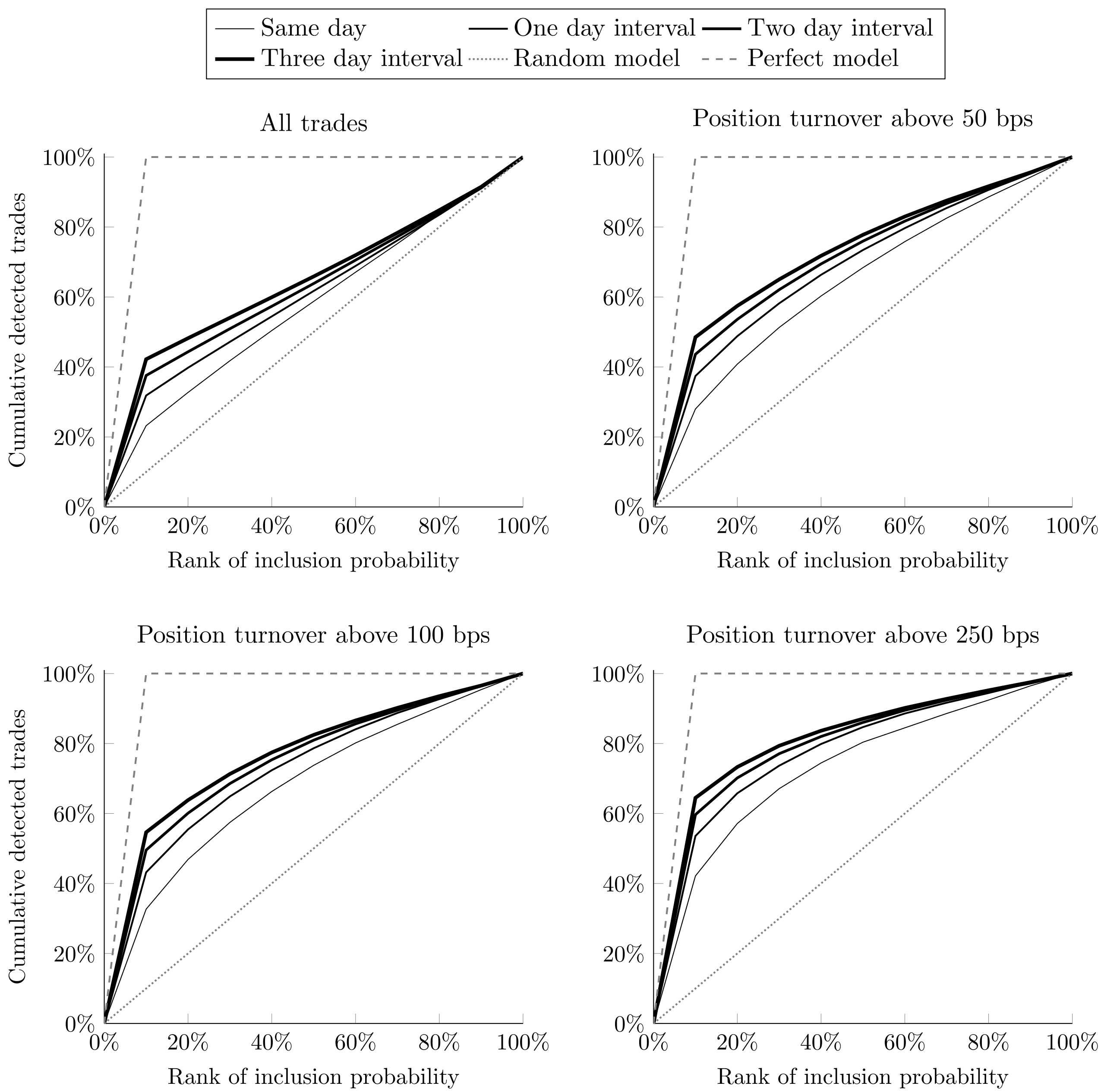


Figure 3. Heatmap of the precision of identifying a trade on the same day as the estimate. In this figure, we show the precision (Prec.) of predicting trade incidence on the same day as trade estimates in our simulated data with an inclusion probability threshold of 0.99. The precision of each cell is calculated over the set of trades for which the position has at least a certain position turnover (in bps, y-axis) and are part of a portfolio that contains at most a given number of positions (x-axis). The estimates are generated using model parameters $\delta = 1$ (maximum traded shares per stock is equal to the net position change over the quarter) and $\lambda = 0.10$ (estimated trade fractions below 0.10 are regularized to zero). All observations are weighted by the position turnover.

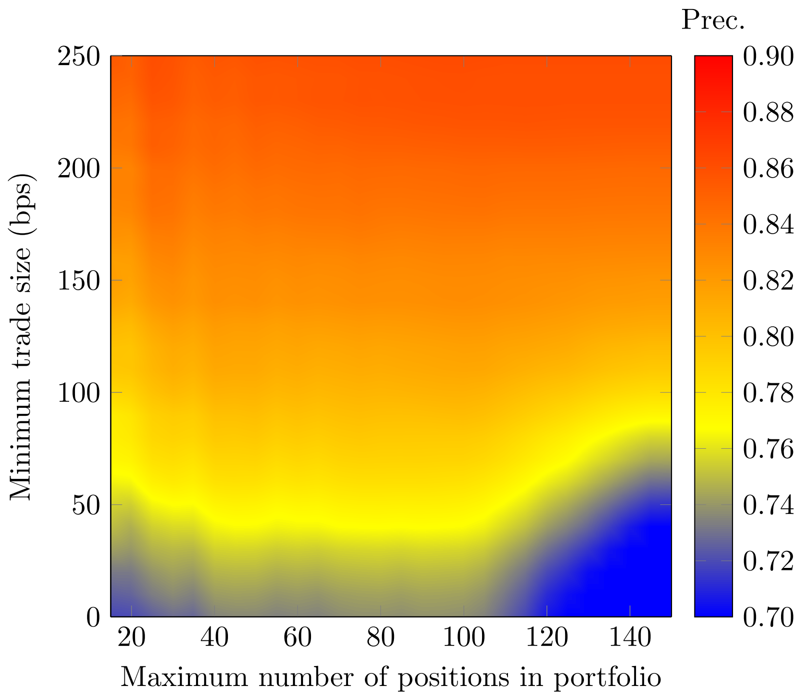


Table III. Active share and active share improvement results. In this table, we show the average, median, 5th and 95th percentile of the active share (Cremers and Petajisto, 2009) of imputed fund holdings to actual fund holdings when using the estimated trades from our model (Ours) and when assuming all trades are executed at either the start (Start), middle (Middle) or end (End) of the quarter. Panel A shows the active shares (in percentages) and Panel B shows the improvement in active share (in percentages) when using our method relative to the start, middle and end assumptions, respectively. We show results both for the actual and simulated samples.

Panel A: Active share (%)

	Actual				Simulated			
	Avg.	Median	P5	P95	Avg.	Median	P5	P95
Ours	2.28	1.66	0.17	6.66	2.81	2.48	0.97	5.18
Start	5.50	4.67	1.88	10.54	6.90	6.29	4.09	10.39
Middle	3.45	2.80	1.13	6.47	4.49	3.90	2.58	7.05
End	6.31	5.00	1.84	12.17	7.93	7.29	4.79	12.05

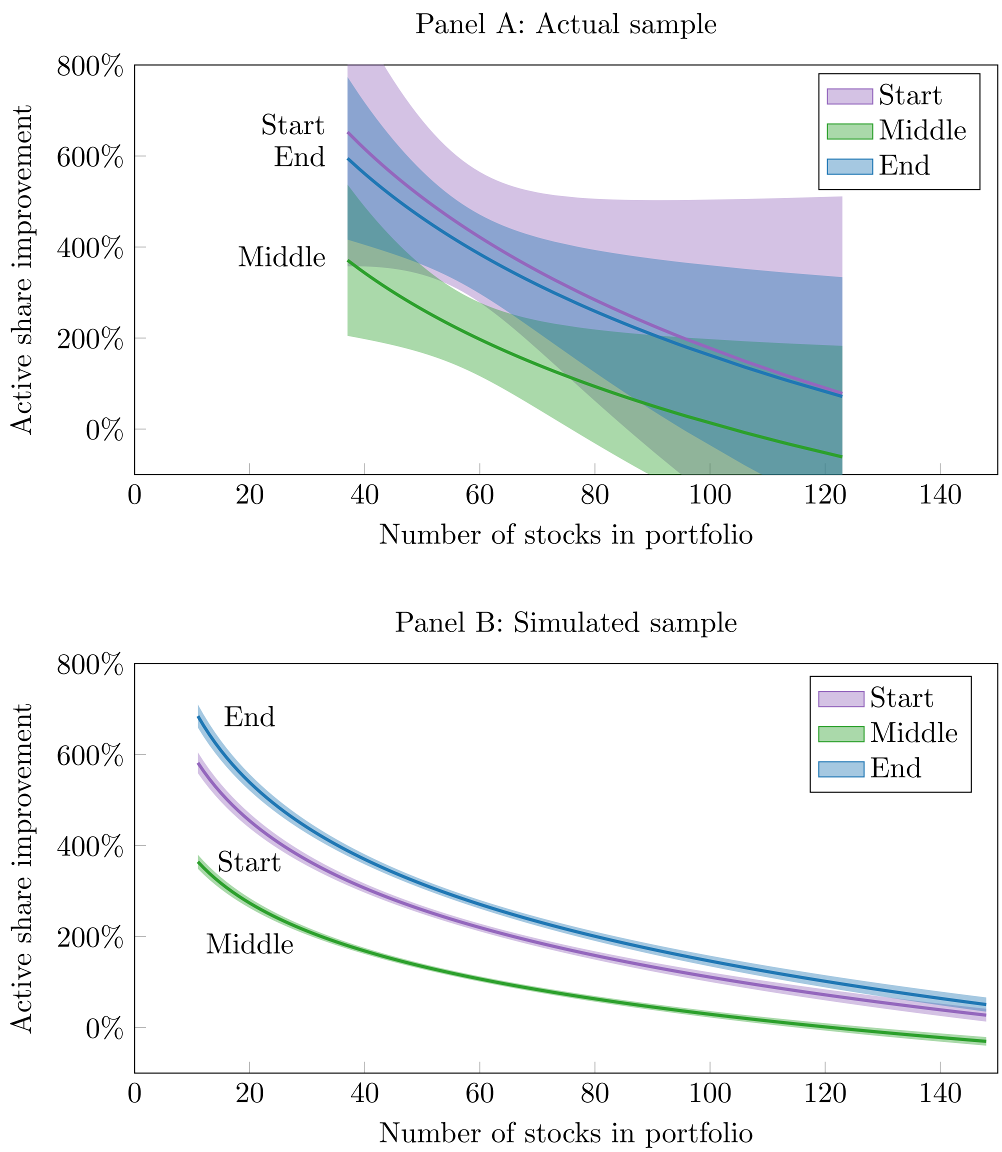
Panel B: Active share improvement (%)

	Actual				Simulated			
	Avg.	Median	P5	P95	Avg.	Median	P5	P95
Ours vs. Start	430.39	187.15	14.04	1533.97	218.85	151.77	28.75	565.70
Ours vs. Middle	203.77	68.71	−30.53	900.76	106.09	57.03	−25.07	359.63
Ours vs. End	392.39	220.61	53.08	1408.53	269.62	192.30	48.93	702.02

Table IV. Active share improvement models. In this table, we show the coefficients and adjusted R^2 when regressing the improvement in average active share of our model, relative to three commonly used assumptions, on the log of the number of stocks in the portfolio. To this end, we first calculate the active share (Cremers and Petajisto, 2009) of imputed fund holdings to actual fund holdings when using the estimated trades from our model and when assuming all trades are executed at the start (Start), middle (Middle) or end (End) of the quarter. We then calculate the relative improvement of our model to each assumption, and regress the improvements on the log of the number of stocks in the portfolio. The estimates of our model are generated using model parameters $\delta = 1$ (maximum traded shares per stock is equal to the net position change over the quarter) and $\lambda = 0.10$ (estimated trade fractions below 0.10 are regularized to zero). t-statistics are denoted in brackets below the corresponding coefficients. *, **, *** denote significance at the 10%, 5% and 1% levels, respectively.

	Actual			Simulated		
	Intercept	Log(stocks)	Adj. R^2	Intercept	Log(stocks)	Adj. R^2
Ours vs. Start	23.80**	−4.78*	0.01	10.93***	−2.13***	0.18
	(2.08)	(−1.70)		(41.21)	(−33.41)	
Ours vs. Middle	16.70**	−3.60**	0.03	7.28***	−1.52***	0.20
	(2.59)	(−2.28)		(40.82)	(−35.35)	
Ours vs. End	21.68***	−4.36**	0.04	12.69***	−2.44***	0.19
	(3.12)	(−2.56)		(43.02)	(−34.35)	
Observations		136			5000	

Figure 4. Average active share improvement over different assumptions. In this graph, we show the predictions and 95% prediction intervals when regressing the improvement in average active share of our model, relative to three commonly used assumptions, on the log of the number of stocks in the portfolio. To this end, we first calculate the active share (Cremers and Petajisto, 2009) of imputed fund holdings to actual fund holdings for each day in the quarter, and then average the active share over the quarter. We impute the fund holdings using the estimated trades from our model and when assuming all trades are executed at either the start (Start), middle (Middle) or end (End) of the quarter. We then calculate the relative improvement of our model to each assumption, and regress the improvements on the log of the number of stocks in the portfolio. The estimates of our model are generated using model parameters $\delta = 1$ (maximum traded shares per stock is equal to the net position change over the quarter) and $\lambda = 0.10$ (estimated trade fractions below 0.10 are regularized to zero). The results for the actual and simulated samples are presented in Panel A and Panel B, respectively.



Appendix A. Supplementary tables and figures

Table V. Variable definitions. In this table we present the variable definitions as used in the text and Table I.

AuM or fund value ($B)	The average assets under management during the quarter, in billions USD.
Quarterly turnover (%)	The sum of the absolute buy and sell trades during the quarter, divided by the average fund value over the quarter. This is also known as double-counted turnover.
Positions	The total number of unique non-cash positions that the fund is invested in during the quarter.
Traded positions	The number of positions for which the fund executed a trade during the quarter.
Position turnover (%)	The dollar value of the net position change over the quarter, divided by the average fund value over the quarter.
Trades	The total number of executed trades during the quarter.
Trades per traded position	The average number of trades that are executed given that a position is traded at any time during the quarter.
Net position change	The difference between the number of shares held at the beginning and end of the quarter.
Fraction net volume	The average fraction of the number of shares traded on a single day divided by the net position change.
Sum of absolute fractions	The average sum of the absolute fraction of the number of shares traded on a single day divided by the net position change.
HHI (normalized)	The normalized Herfindahl–Hirschman Index (HHI) of the fraction of shares traded in a position during the quarter relative to the total number of shares traded in that position during the quarter, averaged over all traded positions in a given fund-quarter.
With roundtrip (%)	The percentage of positions for which the total number of transacted shares is more than the net trading volume. These positions have at least one buy and one sell trade during the quarter.
Unobserved (%)	The percentage of positions in which the fund is not invested at the beginning and end of the quarter, but for which the fund is invested at any time during the quarter.
New positions (%)	The percentage of positions in which the fund is not invested at the beginning of the quarter, but is invested at the end of the quarter.
Liquidated positions (%)	The percentage of positions in which the fund is invested at the beginning of the quarter, but is not invested at the end of the quarter.
Cash position (%)	The average value of the cash position during the quarter, divided by the average fund value over the quarter.

Figure 5. Convergence. This figure shows the average active share of the simulated data after 1, 10, 50, 100, 250, 500, 1000 and 2000 iterations of the reinforcement learning algorithm relative to the average active share after 2000 iterations. The black line depicts the average over all 5000 observations. Depicted in gray is the area between the 5th and 95th percentile of this data. The trade estimates are generated using model parameters $\delta = 1$ (maximum traded shares per stock is equal to the net position change over the quarter) and $\lambda = 0.10$ (estimated trade fractions below 0.10 are regularized to zero).

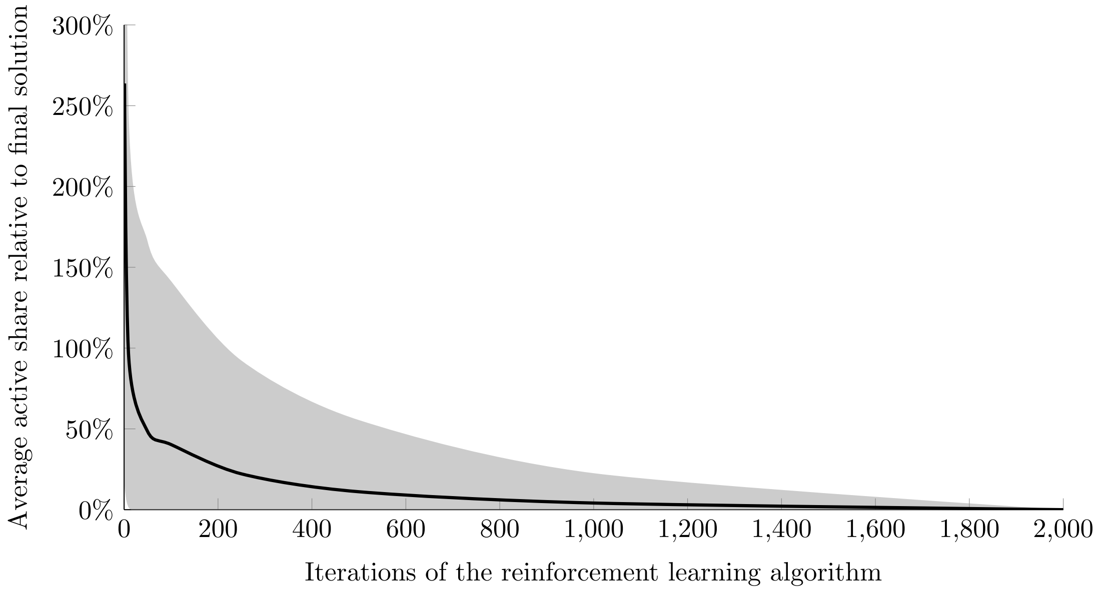
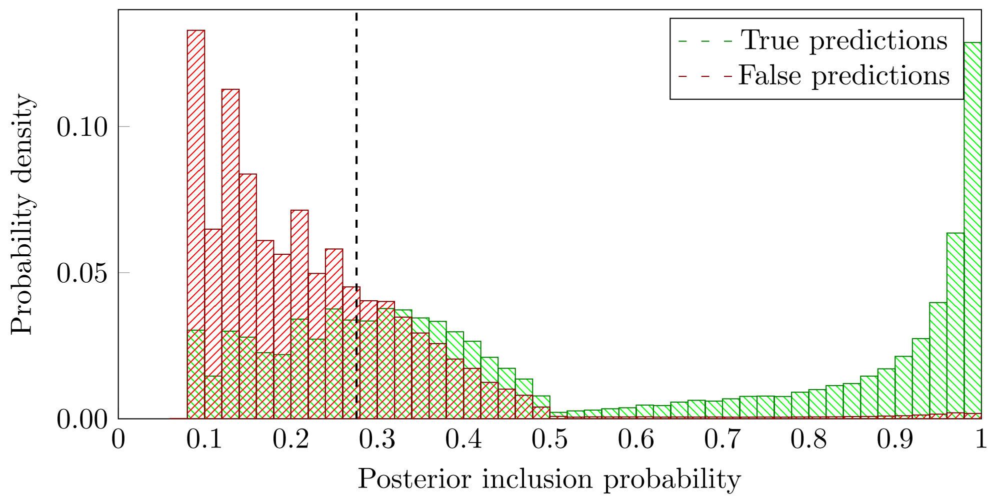


Table VI. Trade identification results on simulated data (equally-weighted). In this table, we summarize the diagnostic ability of our model in predicting trade incidence and trade size on our simulated data. The trade estimates are generated using model parameters $\delta = 1$ (maximum traded shares per stock is equal to the net position change over the quarter) and $\lambda = 0.10$ (estimated trade fractions below 0.10 are regularized to zero). All observations are weighted equally. Incidence predictive power is summarized using the area under the receiver operating characteristic curve (AUC). The AUC can be interpreted as the probability that our model ranks a random observation with a trade higher than a random observation without a trade. We also show the precision (Prec.), negative predictive value (NPV), sensitivity (Sens.) and specificity (Spec.) for the posterior inclusion probability cutoff point 0.99 to demonstrate the model's ability to generate high-precision estimates. Trade size predictive power is shown with the bias and variance of the estimated trade fractions of true positive observations. The results are sliced along two dimensions. First, we show results separately for all trades, and for trades where the position turnover is at least 25, 50, 100 or 250 basis points (bps). Second, we show results on the same day as, and within one, two and three-day intervals of, trade estimates.

	Trade incidence					Trade size	
	AUC	Prec.	NPV	Sens.	Spec.	Bias	Variance
All trades							
Same day	0.57	0.52	0.95	0.01	1.00	0.06	0.10
One day interval	0.60	0.65	0.95	0.01	1.00	0.04	0.09
Two day interval	0.63	0.72	0.95	0.01	1.00	0.03	0.09
Three day interval	0.64	0.76	0.95	0.01	1.00	0.03	0.08
25+ bps							
Same day	0.62	0.54	0.94	0.01	1.00	0.06	0.10
One day interval	0.67	0.68	0.94	0.01	1.00	0.04	0.09
Two day interval	0.70	0.74	0.94	0.02	1.00	0.03	0.08
Three day interval	0.72	0.78	0.93	0.02	1.00	0.03	0.08
50+ bps							
Same day	0.65	0.57	0.93	0.02	1.00	0.06	0.09
One day interval	0.71	0.71	0.93	0.02	1.00	0.04	0.08
Two day interval	0.74	0.78	0.93	0.03	1.00	0.03	0.07
Three day interval	0.76	0.81	0.93	0.03	1.00	0.03	0.07
100+ bps							
Same day	0.70	0.64	0.93	0.03	1.00	0.06	0.07
One day interval	0.76	0.78	0.93	0.04	1.00	0.03	0.07
Two day interval	0.79	0.84	0.93	0.05	1.00	0.02	0.06
Three day interval	0.80	0.87	0.93	0.05	1.00	0.02	0.05
250+ bps							
Same day	0.76	0.72	0.94	0.07	1.00	0.03	0.04
One day interval	0.82	0.86	0.94	0.10	1.00	0.02	0.04
Two day interval	0.84	0.91	0.94	0.12	1.00	0.01	0.03
Three day interval	0.86	0.93	0.94	0.13	1.00	0.01	0.03

Figure 6. Probability separation. In this plot, we show the probability densities of the posterior inclusion probabilities (π) of both true (green) and false (red) predictions of trade incidence within two days of an estimated trade, for positions with a position turnover of 100 bps or more, using our simulated data. The estimates are generated using model parameters $\delta = 1$ (maximum traded shares per stock is equal to the net position change over the quarter) and $\lambda = 0.10$ (estimated trade fractions below 0.10 are regularized to zero). All observations are weighted by the position turnover. The dashed black line indicates the posterior inclusion probability cutoff point where the model's sensitivity is exactly equal to the specificity ($\pi = 0.28$).



Appendix B. Demonstration of algorithm

Table VII. Trade identification results on actual data (turnover-weighted). In this table, we summarize the diagnostic ability of our model in predicting trade incidence and trade size on the actual sample. The trade estimates are generated using model parameters $\delta = 1$ (maximum traded shares per stock is equal to the net position change over the quarter) and $\lambda = 0.10$ (estimated trade fractions below 0.10 are regularized to zero). All observations are weighted by the position turnover. Incidence predictive power is summarized using the area under the receiver operating characteristic curve (AUC). The AUC can be interpreted as the probability that our model ranks a random observation with a trade higher than a random observation without a trade. We also show the precision (Prec.), negative predictive value (NPV), sensitivity (Sens.) and specificity (Spec.) for the posterior inclusion probability cutoff point 0.99 to demonstrate the model's ability to generate high-precision estimates. Trade size predictive power is shown with the bias and variance of the estimated trade fractions of true positive observations. The results are sliced along two dimensions. First, we show results separately for all trades, and for trades where the position turnover is at least 25, 50, 100 or 250 basis points (bps).The results for the trades with position turnover above 250 bps are inaccurate due to the small sample size in this category. Second, we show results on the same day as, and within one, two and three-day intervals of, trade estimates.

	Trade incidence					Trade size	
	AUC	Prec.	NPV	Sens.	Spec.	Bias	Variance
All trades							
Same day	0.68	0.59	0.94	0.01	1.00	0.06	0.04
One day interval	0.73	0.80	0.94	0.02	1.00	0.04	0.04
Two day interval	0.76	0.87	0.94	0.03	1.00	0.05	0.03
Three day interval	0.78	0.89	0.94	0.03	1.00	0.03	0.02
25+ bps							
Same day	0.69	0.59	0.94	0.02	1.00	0.06	0.04
One day interval	0.74	0.80	0.94	0.02	1.00	0.04	0.04
Two day interval	0.76	0.87	0.94	0.03	1.00	0.05	0.03
Three day interval	0.78	0.89	0.93	0.03	1.00	0.03	0.02
50+ bps							
Same day	0.69	0.59	0.93	0.02	1.00	0.06	0.04
One day interval	0.75	0.80	0.93	0.03	1.00	0.04	0.03
Two day interval	0.77	0.88	0.93	0.03	1.00	0.05	0.03
Three day interval	0.79	0.89	0.93	0.03	1.00	0.03	0.02
100+ bps							
Same day	0.70	0.61	0.93	0.02	1.00	0.07	0.04
One day interval	0.76	0.82	0.93	0.03	1.00	0.05	0.03
Two day interval	0.78	0.91	0.93	0.04	1.00	0.06	0.03
Three day interval	0.80	0.92	0.93	0.04	1.00	0.04	0.02
250+ bps							
Same day	0.70	0.60	0.94	0.04	1.00	0.04	0.02
One day interval	0.75	0.83	0.94	0.06	1.00	0.06	0.02
Two day interval	0.78	0.92	0.94	0.07	1.00	0.07	0.02
Three day interval	0.80	0.92	0.94	0.07	1.00	0.05	0.01

Table VIII. Trade identification results on actual data (equally-weighted). In this table, we summarize the diagnostic ability of our model in predicting trade incidence and trade size on the actual sample. The trade estimates are generated using model parameters $\delta = 1$ (maximum traded shares per stock is equal to the net position change over the quarter) and $\lambda = 0.10$ (estimated trade fractions below 0.10 are regularized to zero). All observations are weighted equally. Incidence predictive power is summarized using the area under the receiver operating characteristic curve (AUC). The AUC can be interpreted as the probability that our model ranks a random observation with a trade higher than a random observation without a trade. We also show the precision (Prec.), negative predictive value (NPV), sensitivity (Sens.) and specificity (Spec.) for the posterior inclusion probability cutoff point 0.99 to demonstrate the model's ability to generate high-precision estimates. Trade size predictive power is shown with the bias and variance of the estimated trade fractions of true positive observations. The results are sliced along two dimensions. First, we show results separately for all trades, and for trades where the position turnover is at least 25, 50, 100 or 250 basis points (bps). The results for the trades with position turnover above 250 bps are inaccurate due to the small sample size in this category. Second, we show results on the same day as, and within one, two and three-day intervals of, trade estimates.

	Trade incidence					Trade size	
	AUC	Prec.	NPV	Sens.	Spec.	Bias	Variance
All trades							
Same day	0.62	0.55	0.95	0.01	1.00	0.04	0.06
One day interval	0.67	0.75	0.95	0.01	1.00	0.00	0.05
Two day interval	0.69	0.80	0.95	0.01	1.00	0.01	0.04
Three day interval	0.71	0.81	0.95	0.01	1.00	0.00	0.04
25+ bps							
Same day	0.67	0.57	0.94	0.01	1.00	0.04	0.05
One day interval	0.72	0.76	0.94	0.01	1.00	0.01	0.04
Two day interval	0.75	0.81	0.94	0.02	1.00	0.03	0.03
Three day interval	0.77	0.82	0.94	0.02	1.00	0.02	0.02
50+ bps							
Same day	0.69	0.58	0.93	0.01	1.00	0.05	0.05
One day interval	0.74	0.78	0.93	0.02	1.00	0.02	0.04
Two day interval	0.77	0.83	0.93	0.02	1.00	0.03	0.03
Three day interval	0.79	0.84	0.93	0.02	1.00	0.02	0.02
100+ bps							
Same day	0.70	0.63	0.93	0.02	1.00	0.07	0.04
One day interval	0.76	0.82	0.93	0.03	1.00	0.04	0.03
Two day interval	0.78	0.90	0.93	0.03	1.00	0.05	0.03
Three day interval	0.80	0.91	0.93	0.04	1.00	0.03	0.02
250+ bps							
Same day	0.70	0.64	0.94	0.03	1.00	0.03	0.02
One day interval	0.75	0.84	0.94	0.05	1.00	0.05	0.02
Two day interval	0.77	0.91	0.94	0.06	1.00	0.06	0.02
Three day interval	0.80	0.91	0.94	0.07	1.00	0.04	0.01

Figure 7. ROC curves for actual data. In this figure, we show the receiver operating characteristic curves (ROC curves) when predicting trade incidence on the actual sample. The ROC curve shows the trade-off between the false positive rate (x-axis) and the true positive rate (y-axis; also known as sensitivity) for different inclusion probability cutoff points (π). The area under the ROC curve (AUC) can be interpreted as the probability that our model ranks a random observation with a trade higher than a random observation without a trade. We show ROC curves when predicting trade incidence on the same day, and within one-, two- and three-day intervals of the estimated trades. The estimates are generated using model parameters $\delta = 1$ (maximum traded shares per stock is equal to the net position change over the quarter) and $\lambda = 0.10$ (estimated trade fractions below 0.10 are regularized to zero). All observations are weighted by the position turnover. We show the ROC curves separately for all trades, and for trades where the position turnover is at least 50, 100 or 250 basis points (bps). The CAP curves for the trades with position turnover above 250 bps are inaccurate due to the small sample size in this category.

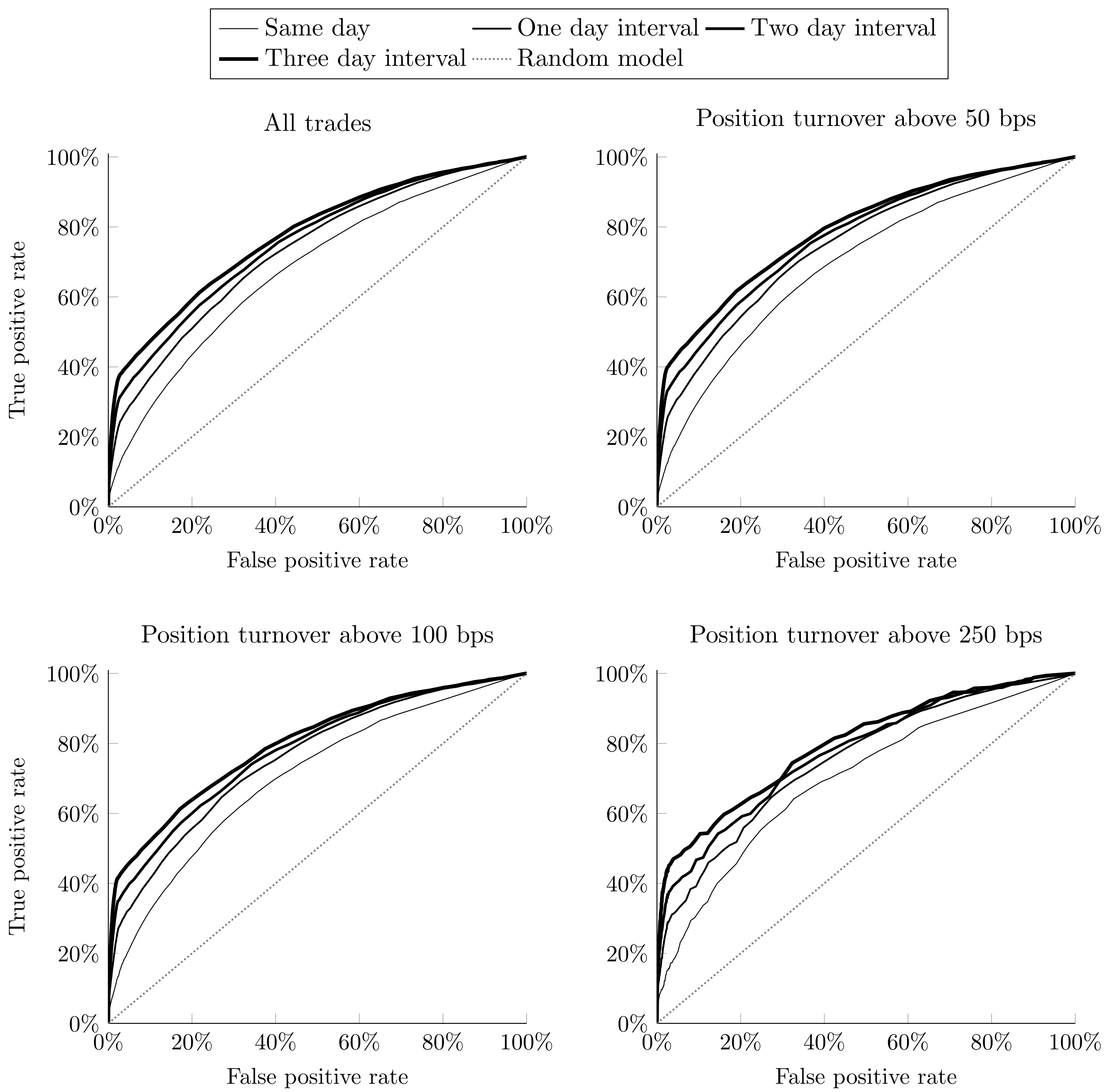


Figure 8. CAP curves for actual data. In this figure, we show the cumulative accuracy profile curves (CAP curves) when predicting trade incidence on the actual sample. The CAP curve represents the cumulative fraction of correctly detected trades (y-axis), when ranking the trade estimates from our model from highest to lowest inclusion probability (x-axis). We show CAP curves when predicting trade incidence on the same day, and within one-, two- and three-day intervals of the estimated trades. The estimates are generated using model parameters $\delta = 1$ (maximum traded shares per stock is equal to the net position change over the quarter) and $\lambda = 0.10$ (estimated trade fractions below 0.10 are regularized to zero). All observations are weighted by the position turnover. We show the CAP curves separately for all trades, and for trades where the position turnover is at least 50, 100 or 250 basis points (bps). The CAP curves for the trades with position turnover above 250 bps are inaccurate due to the small sample size in this category.

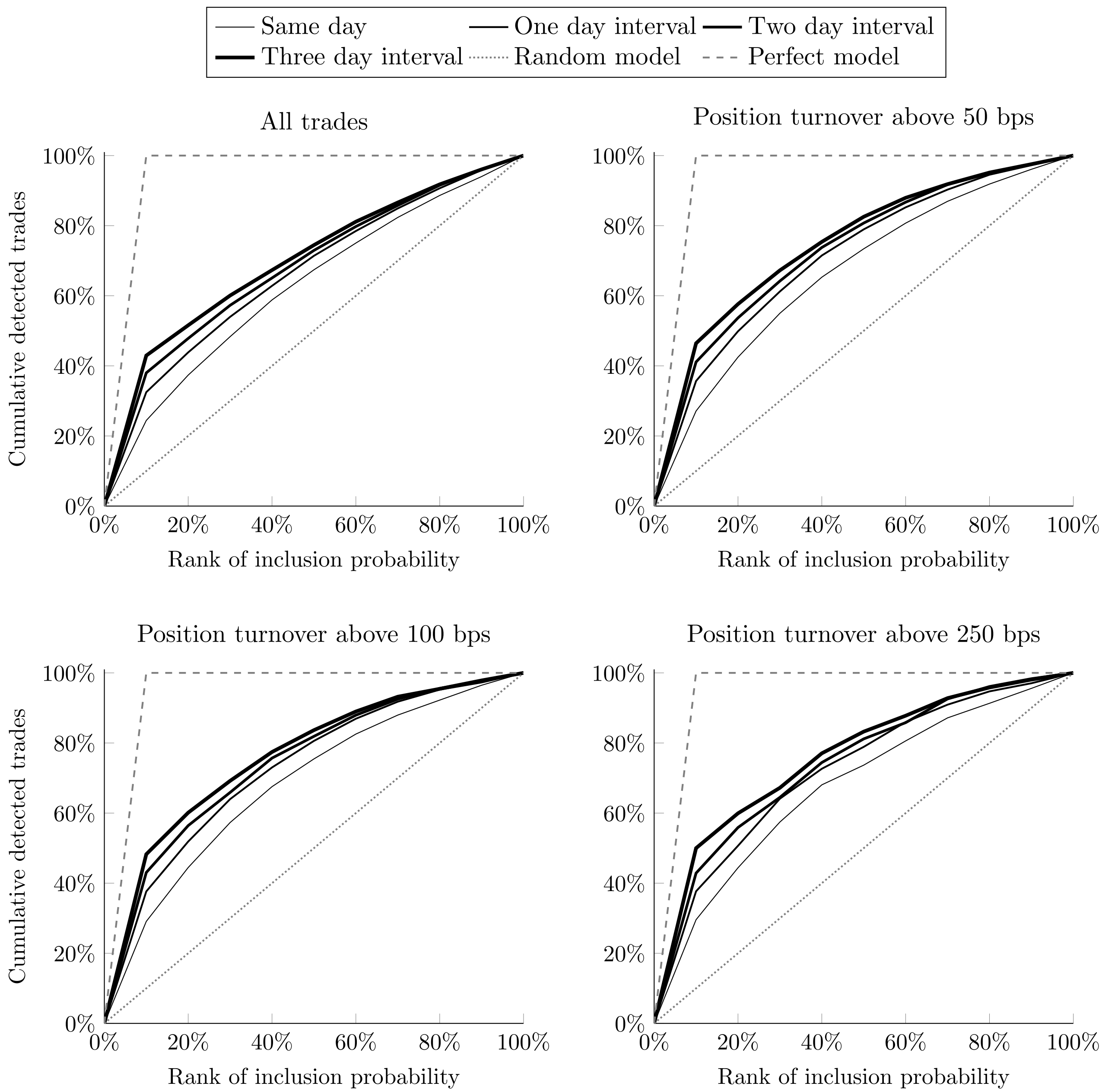
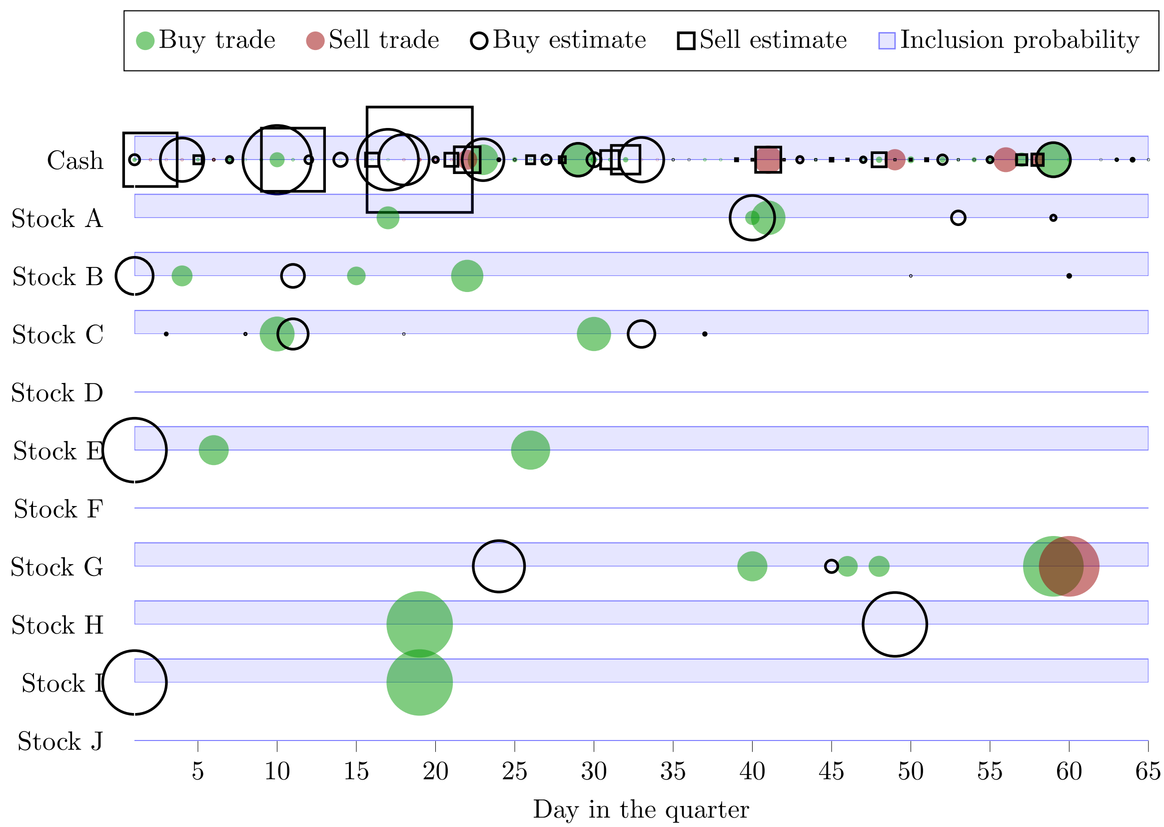
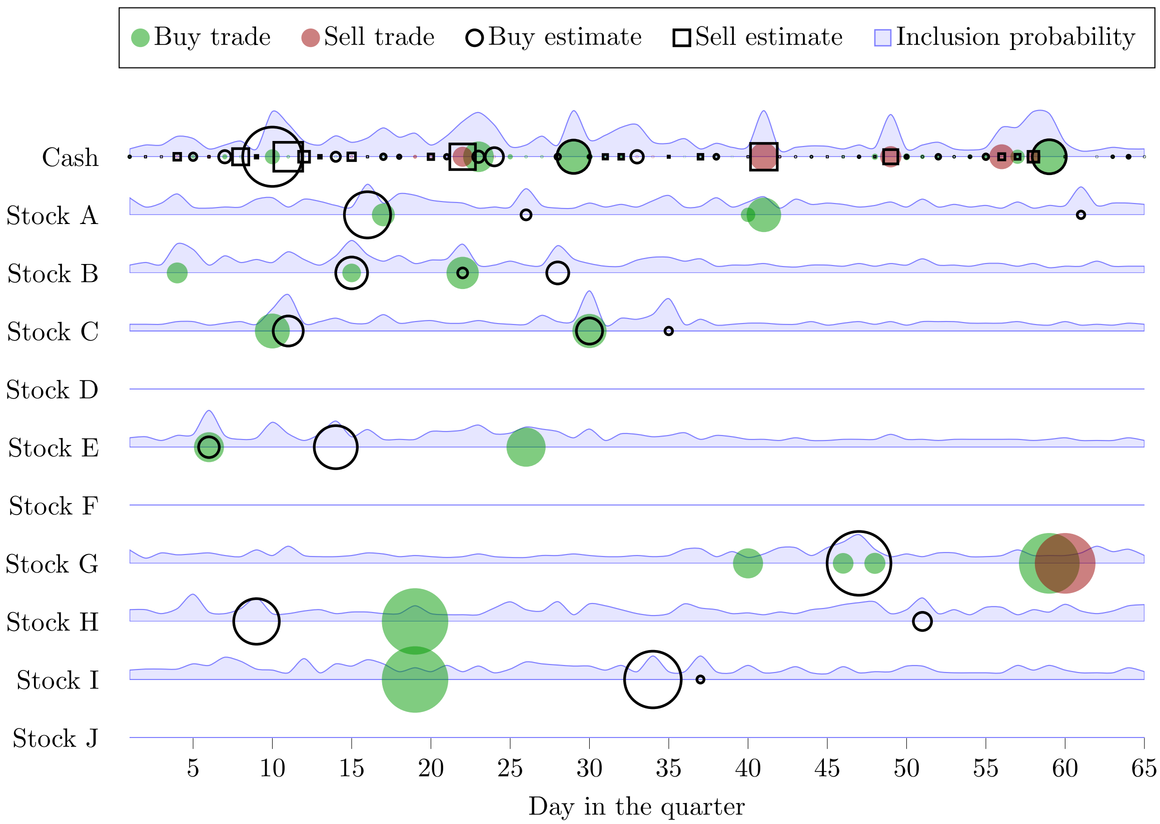


Figure 9. Solution example. In the figures below, we visualize the solutions of the reinforcement learning algorithm after 1 (panel a), 50 (panel b), 250 (panel c) and 2000 (panel d) iterations for a simulated portfolio of 10 stocks. The actual buy and sell trades are shown using green and red circles, respectively. The buy and sell estimates are indicated with black circles and squares, respectively. The size of the markers correspond to the absolute fraction of the net position change over the quarter. The inclusion probabilities are plotted in the background, ranging from 0 to 1. The estimates are generated using model parameters $\delta = 1$ and $\lambda = 0.10$.

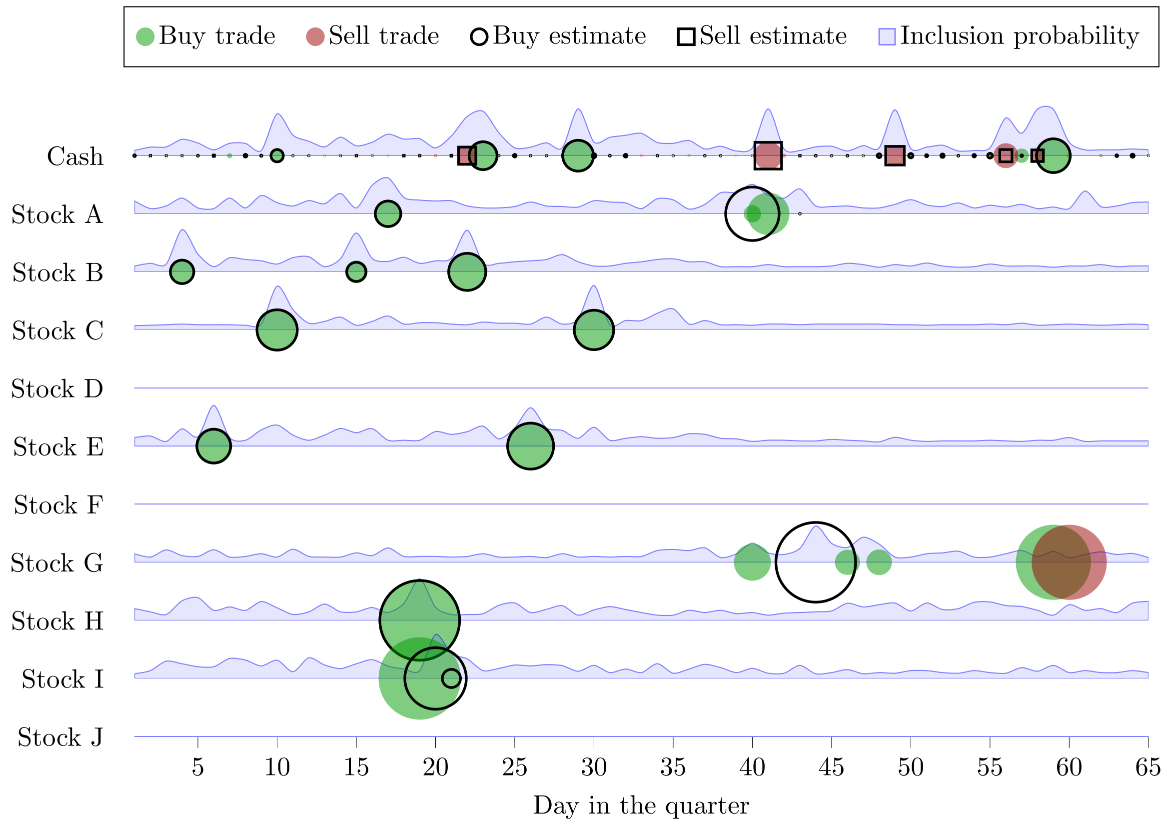
(a) Solution after solving the LP problem once. In the first iteration, the maximum allowed use of cash is still large, which leads to underidentification. As such, the majority of the imputed trades in both the cash position and the stocks are wrongly identified.



(b) Solution after 50 iterations of the reinforcement learning algorithm. After 50 iterations, the maximum allowed use of cash has started to converge, and the algorithm started narrowing down on the position and size of individual trades. Yet, the inclusion probabilities of most stock-days are still high, implying that the model is still exploring alternative solutions.



(c) Solution after 250 iterations of the reinforcement learning algorithm. After 250 iterations, the inclusion probabilities are starting to converge to individual stock-days. Most of the trades have already converged, but for some trades the algorithm is having trouble identifying the number of trades and their exact location. For these stocks, the algorithm is exploring different possibilities. For example, for Stock G, H and I, the inclusion probabilities are still large for many days in the quarter.



(d) Solution after 2000 iterations of the reinforcement learning algorithm. After 2000 iterations, all trades that are identifiable have been found. The algorithm remains uncertain about the trades for Stock G. The reason is that Stock G contains a large single-day round-trip trade between day 59 and 60, which it cannot identify due to the parameter $\delta = 1$ (maximum traded shares per stock is equal to the net position change over the quarter). This also causes uncertainty in the other trades for that stock. For Stock H and I, the algorithm identifies multiple smaller trades, instead of one large trade, which can happen if the individual stock returns on those days are close to zero.