Code for:
Cross-sectional identification of private information

Dion Bongaerts
Rotterdam School of Management, Erasmus University

Dominik Rösch
University at Buffalo, State University of New York

Mathijs van Dijk
Rotterdam School of Management, Erasmus University

ABSTRACT

This document provides code to replicate the results of "Cross-Sectional Identification of Private Information" (Bongaerts, Rösch, and van Dijk, 2025) using data available through Wharton Research Data Services (WRDS). Specifically, it reproduces parts of Tables 1, 2, 3, 5, 6, and 8. The document is written as a knitr (R Markdown) file. To execute the code and generate a PDF, the recommended approach is to use a local installation of RStudio. Upon execution, the code automatically downloads and stores all necessary data from WRDS locally. To enable this, you must first configure WRDS access as described below. Note: While the published article uses data from Refinitiv/TRTH, this replication uses WRDS and TAQ data. The results are qualitatively robust to these differences in data sources and partially different methodologies, as discussed in the relevant sections below.

1 Discussion

This code replicates the main results of "Cross-Sectional Identification of Private Information", following a setup similar to that proposed in Aghbabali, Choi, Im, Rösch, and Roy (2025). The document includes all necessary code to reproduce this PDF: it downloads the required data from WRDS, performs the analysis, and presents the results. The actual R code used to generate this document is included in the Appendix at the end of the file.

Ideally, this code would fully reproduce the results of "Cross-Sectional Identification of Private Information" using the original data and methodology. However, the reference study relies on Thomson Reuters Tick History (TRTH) data, which is prohibitively large (multiple terabytes), costly, and infrequently used in academic research. Instead, this replication focuses on making the analysis as accessible as possible by using WRDS Intraday Indicators, which are available to anyone with access to WRDS and TAQ. These indicators, derived from TAQ data, provide stock-day level estimates and therefore do not require analyzing massive tick-by-tick data. They offer a practical alternative to working directly with the full TRTH (or TAQ) dataset. For more details on WRDS Intraday Indicators, see WRDS (2023).

Using WRDS Intraday Indicators differs from Bongaerts et al. (2025) in two main ways:

- Quoted spreads (PQSPR) in Bongaerts et al. (2025) is estimated as the average "spread associated with all trades" but in WRDS Intraday Indicators it is the time-weighted average.

- Price impact in Bongaerts et al. (2025) is estimated as λ from stock-day regressions, explaining 5-minute log-returns by the signed, trading volume within the interval. WRDS Intraday Indicators reports price impact from the same regressions but using the square-root of trading volume as the explanatory variable.

We discuss both differences in more details in later Sections.

The following sections, Sections 2 to 2.2, are more or less copied from Aghbabali et al. (2025), they contain general setup steps, required for any project.

2 Setup

Before we run any code, we setup an error handler. This ensures that if the code breaks at any point, e.g., because the working directory is not yet set correctly, the output will stop and display the error message at the end of the document.

```
# see https://stackoverflow.com/q/74097101
knitr::knit_hooks$set(error = function(x, options) {
  ERROR_GLOBAL <<- x
  knitr::knit_exit()
})

knit_chunk <- knitr::knit_hooks$get("chunk")

knitr::knit_hooks$set(chunk = function(x, options){

  out <- x
  if (exists('ERROR_GLOBAL', inherits = TRUE)) {
    out <- paste0(out,'\n**stopped with error**:',ERROR_GLOBAL,'\n\n')
    options$size       <- "large"
    options$background <- '#FFDDDD'
    options$tidy       <- 'T'
    options$tidy.opts  <- list(width.cutoff=50)
    options$results    <-'markup'
  }

  knit_chunk(out, options)

})
```

2.1 Prerequisites

In this section we check whether the system can execute the analysis.

First we ensure that the system has enough free memory to successfully run the analysis. Unfortunately, retrieving available memory is not straight forward and depends on, e.g., the operating system and whether the code is running on a virtual machine. Because of that we do not provide any actual code, but ask users to manually verify that enough RAM is available. A lack of RAM often explain sudden termination, without generating any output files.

```
#to do - we need to have a few GB's
```

Next, we load all required R libraries. While this must be done at the beginning of the script, to not disrupt the reading flow we only evaluate the code here and show it at the end of the file, Section 4.

```
# The actual code is executed here, but shown only at the end of the file.
```

```
## [1] TRUE
```

Similarly, we ensure that we compile the document using tinytex and that all required Latex packages are available. If any package is missing we try to install it:

```
if(! tinytex::is_tinytex()) {
  stop("We use tinytex to install missing Latex packages, if you
  are using a different Latex version this might not be compatible.
  It might be easiest to run tinytex::install_tinytex()")
}

# if(! grepl("TinyTeX", Sys.which("pdflatex"))) {
# stop("It looks like you are using a different Latex version
# which might not be compatible.
# It might be easiest to switch to TinyTex, for example,
# by adding the PATH to pdflatex in TinyTex directory
# to PATH environmental variable. See the code in the Rnw file below.")
```

```
#
#   cat('Sys.setenv(PATH = paste0(tinytex::tinytex_root(),
#"\\bin\\windows;", Sys.getenv("PATH")))\n',
#     file = file.path(Sys.getenv("HOME"), ".Rprofile"),
#     append = TRUE)
# }

latex_packages <- c('listings', 'caption', 'mathtools', 'floatrow',
                    'setspace', 'cmap', 'filecontents')

for (pkg in latex_packages) {
  tinytex::check_installed(pkg) || tinytex::tlmgr_install(pkg)
}
```

2.2 Setup variable names

This section contains various code blocks to setup the analysis such as variables determining how the compiled PDF will look (e.g., whether it will contain the underlying R code.); environment variables that need to be adjusted (like where to store the data); and variables related to the analysis (such as start and end dates).

- First, the code contains variables that determine how the PDF will look, e.g., with ("echo=T") or without the underlying R code ("echo=F".)

```
knitr::opts_knit$set(progress = TRUE, verbose = TRUE)
knitr::opts_knit$set(self.contained=T)

# cache.lazy = F (helps error when caching large data sets)
knitr::opts_chunk$set(echo=T,warning=F,message=F, error=T, tidy=T,
tidy.opts=list(width.cutoff=60), results='markup',cache.lazy = FALSE,
cache.rebuild=F)
```

- Second, the code sets important environment variables (which need to be adjusted by the user):

```
setwd("~/repositories/xpin")
data_dir <- "/vscratch/grp-drosch/xpin/"


if (!file.exists(data_dir)) {
    stop(sprintf("Could not find data_dir '%s' to store data,
                 first create this directory.",
        data_dir))
}
```

- Third, we define variables specific to the analysis (which can be adjusted by the user). As in Bongaerts et al. (2025) the "sample starts on February 1, 2001 (to prevent issues stemming from the tick size change on January 29, 2001) and runs until the end of 2014."

```
date_start <- "2001-02-01"
date_end <- "2014-12-31"
```

- Fourth, we setup the username and password to download all required data from WRDS. We use the R-package "keyring" to safely store the username and password. To do either, you need to execute the code directly in the console to set the key ring (see code below, within tryCatch).

```
tryCatch({
    keyring_create("credentials_wrds", password = "letmein")
    key_set("wrds_username", keyring = "credentials_wrds", prompt = "Your WRDS us
    key_set("wrds_password", keyring = "credentials_wrds", prompt = "Your WRDS pa
}, error = function(e) {
})

## are you sure you want to overwrite ~/.config/r-keyring/credentials_wrds.keyrin
## NULL

keyring_unlock("credentials_wrds", password = "letmein")
```

- Last, we establish a connection to WRDS. If required data cannot be found, we can automatically download the data from WRDS.

```
postregs <- ifelse(exists("Postgres"), Postgres, PostgreSQL)

tryCatch({
    wrds <- dbConnect(postregs(), host = "wrds-pgdata.wharton.upenn.edu",
        port = 9737, dbname = "wrds", user = key_get("wrds_username",
            keyring = "credentials_wrds"), password = key_get("wrds_password",
            keyring = "credentials_wrds"))
}, error = function(e) {
    stop("It is likely that wrds_username and
    wrds_password are not stored in keyring.
    For that, run code in setup_wrds_password directly
    in R console. An error occurred: ",
        e$message)
})
```

2.3 Setup Data

If we can find the data in the previously defined data directory, read it. Otherwise download it. For that we first define a function to download the relevant stock data from WRDS, i.e., as in the reference paper we only download common-stocks ('shrcd' is 10 or 11) listed on the NYSE ('primexch' is 'N'). We use data from WRDS to determine the sample, adjust prices by corporate actions, and estimate the size (marketcap) of companies.

```
get_stock_returns <- function(wrds, date_start, date_end) {
    sql_query <- sprintf("SELECT a.cusip, a.permno, a.date, a.prc, a.vol,
           a.ret, a.openprc, a.askhi, a.bidlo,
           b.ticker, b.primexch, a.shrout, b.shrcd
      FROM crsp.dsf AS a, crsp.dsenames AS b
      WHERE a.permno = b.permno
```

```
      AND a.date BETWEEN b.namedt AND b.nameendt
      AND a.date >= '%s' AND a.date <= '%s'
      AND b.shrcd IN ('10', '11')
      AND b. IN ('N')",
        date_start, date_end)
    crsp_data <- dbGetQuery(wrds, sql_query) %>%
        as.data.table()
    return(crsp_data)
}
```

Then we call above function, in case we cannot find the relevant data file:

```
data_file_name <- paste(data_dir, "data_wrds_crsp_complete.csv",
    sep = "/")

if (!file.exists(data_file_name)) {
    data_sample <- get_stock_returns(wrds, date_start, date_end)
    fwrite(data_sample, file = data_file_name)
}

data_sample <- fread(data_file_name)
data_sample[, `:=`(prc, abs(prc))]
data_sample[, `:=`(date, as.Date(date))]
```

In the replication study, we use WRDS Intraday Indicators, which provides stock-day variables estimated from Trade-And-Quote (TAQ) data. In particular, we get an estimate of the stock-day price impact and order imbalance. Given that in the reference paper (Bongaerts et al., 2025) we used TRTH data, the high-frequency data will be filtered differently compared to the reference paper. But both differences the difference in data source and the difference in how data is filtered, will likely have a minor impact on the results, see., e.g., Footnote 7 of Rösch, Subrahmanyam, and van Dijk (2017). More impactful is the way price impact is estimated, as we will elaborate in more details when discussing lambda and price impact.

WRDS Intraday Indicators (based on MTAQ, relevant for our sample starting in 2001) are not available for direct download, as when using DTAQ, see below. We need to download the relevant data from the WRDS Website. The code below indicates the relevant variables and how these map to WRDS Intraday Indicators based on DTAQ.

```
data_file_name <- paste(data_dir, "data_wrds_intraday_mtaq.csv",
    sep = "/")

if (!file.exists(data_file_name)) {

    .select <- "TSignSqrtDVol2, QSpreadPct_TW_m, BuyDollar_LR0,
               SellDollar_LR0, BidLo, NumTrades_m, Mid_4pm"

    stop("WRDS does not offer intraday indicators calculated
         from MTAQ via Postgres. You need to go to their website
        and download the data.")
}

data_wrds_mintraday <- fread(data_file_name)
data_wrds_mintraday[, `:=`(date, as.Date(date))]

setnames(data_wrds_mintraday, "symbol", "ticker")

# map from MTAQ to DTAQ WRDS intraday indicators:
setnames(data_wrds_mintraday, "TSignSqrtDVol2", "tsignsqrtdvol2")
setnames(data_wrds_mintraday, "QSpreadPct_TW_m", "quotedspread_percent_tw")
setnames(data_wrds_mintraday, "BuyDollar_LR0", "buy_dv_lr")
setnames(data_wrds_mintraday, "SellDollar_LR0", "sell_dv_lr")
setnames(data_wrds_mintraday, "BidLo", "price_low_m")
setnames(data_wrds_mintraday, "NumTrades_m", "total_n_trades_m")
setnames(data_wrds_mintraday, "Mid_4pm", "mid_4pm")
```

We can get more recent Intraday Indicators (based on DTAQ) directly. The University at Buffalo only has access to DTAQ data from 2010 onwards. Therefore we

hardcode the ".year_start" to 2010.

```
in_stock_ticker_list <- paste0("('", paste(unique(data_sample[,
    "ticker"]), collapse = "', '"), "')")

data_file_name <- paste(data_dir, "data_wrds_intraday_taq.csv",
    sep = "/")

if (!file.exists(data_file_name)) {

    .select <- "date, sym_root, sym_suffix, TSignSqrtDVol2, quotedspread_percent_tw ,
    .where <- sprintf("(sym_root IN %s)", in_stock_ticker_list)
    .where <- "(1 = 1)"

    # .year_start <- year(date_start) TODO lack of
    # permission to earlier DTAQ and cannot find data for
    # MTAQ
    .year_start <- 2010
    .year_end <- year(date_end)

    query_parts <- c()
    for (year in .year_start:.year_end) {
        part <- sprintf("SELECT %s FROM taqmsec.wrds_iid_%d WHERE %s",
            .select, year, .where)
        query_parts <- c(query_parts, part)
    }

    sql_query <- paste(query_parts, collapse = " UNION ALL\n")

    data_wrds_intraday <- dbGetQuery(wrds, sql_query) %>%
        as.data.table()

    fwrite(data_wrds_intraday, file = data_file_name)
```

```
}

data_wrds_intraday <- fread(data_file_name)
data_wrds_intraday[, `:=`(date, as.Date(date))]

setnames(data_wrds_intraday, "sym_root", "ticker")
```

We now merge CRSP with relevant Intraday Indicators based on MTAQ. CRSP allows us to identify stocks by their unique PERMNO and to adjust prices by corporate actions. All the results in this document are based on MTAQ using the same sample period as in the reference paper.

```
# data_complete <- merge(data_wrds_intraday, data_sample,
# by = c('date', 'ticker'))
data_complete <- merge(data_wrds_mintraday, data_sample, by = c("date",
    "ticker"))
```

As described in the paper, we use five other data sources:

- stock-days on which Form-4 trades occured
- stock-days on which 13D trades occured
- estimates of Good and Bad PIN
- M&A transactions
- loss of analyst coverage

Unfortunately, these data are not commonly available and need to be computed. For details we refer to the reference paper.

2.4 Variable construction according to the reference paper

As in the reference paper we

- "discard stocks that trade for less than $5 at any time during our sample period" (p. 23) and "stock-days with fewer than 50 trades" (p. 24). We drop four days from the analysis on which the cross-sectional average PQSPR differs significantly between the reference paper and the replication study. Difference in PQSPR are because in Bongaerts et al. (2025) we average spreads only associated with trades, in WRDS Intraday Indicators PQSPR is the time-weighted average spread.

```
data_complete[, `:=`(sample_price_low, min(price_low_m)), by = .(permno)]

data_complete <- data_complete[sample_price_low > 5]
data_complete <- data_complete[total_n_trades_m > 50]

dates_bad <- c("2012-11-23", "2011-11-25", "2014-11-28", "2013-11-29")
data_complete <- data_complete[!date %in% dates_bad]
```

- we estimate stock returns "from corporate action adjusted end-of-day mid-quotes (*Return*; in %; cross-sectionally winsorized each day at the 0.1% and 99.9% level)"

```
data_complete[order(date), `:=`(lag_mid_4pm, shift(mid_4pm, 1,
    fill = NA)), by = .(permno)]
data_complete[order(date), `:=`(lag_prc, shift(prc, 1, fill = NA)),
    by = .(permno)]

data_complete[, `:=`(corpact_adj, round((1 + ret)/(prc/lag_prc),
    digits = 6))]

data_complete[, `:=`(ret, corpact_adj * mid_4pm/lag_mid_4pm -
    1)]
data_complete[, `:=`(ret, pmin(ret, quantile(ret, 0.999, na.rm = TRUE))),
    by = .(date)]
data_complete[, `:=`(ret, pmax(ret, quantile(ret, 0.001, na.rm = TRUE))),
    by = .(date)]
```

- we estimate returns and quoted spread in percent and buyer-initiated, seller-initiated, and total trading volumes in USD millions.

```
data_complete[, `:=`(ret, 100 * ret)]
data_complete[, `:=`(quotedspread_percent_tw, quotedspread_percent_tw *
    100)]

data_complete[, `:=`(vol, vol/1e+06)]
data_complete[, `:=`(buy_dv_lr, buy_dv_lr/1e+06)]
data_complete[, `:=`(sell_dv_lr, sell_dv_lr/1e+06)]

data_complete[, `:=`(marketcap, prc * shrout/1e+06)]
```

- we estimate order imbalance (oib) as the "dollar volume of buyer- minus seller-initiated trades" (p. 23). To estimate the quoted spread (PQSPR), in the reference paper we "averag[e] the spread associated with all trades" in the replicating code we use the time-weighted average across all spreads from WRDS, as mentioned before.

```
data_complete[, `:=`(oib, buy_dv_lr - sell_dv_lr)]
data_complete[, `:=`(PQSPR, quotedspread_percent_tw)]
```

- we estimate volatility as the squared-return, trading volume in USD, the day return as the proportional price change from open to close, and " market capitalization ... at the beginning of each calendar year" (p. 26):

```
data_complete[, `:=`(RETSQ, ret^2)]

data_complete[, `:=`(vol_usd, vol * openprc)]

data_complete[, `:=`(year, year(date))]
data_complete[, `:=`(month, month(date))]
```

```
data_complete[, `:=`(return_day, 100 * (prc/openprc - 1))]

data_complete[order(date), `:=`(marketcap, .SD[1, marketcap]),
    by = .(permno, year)]
```

In the reference paper we estimate price impact following our theoretical mode, in particular, "[i]n contrast to Goyenko et al. (2009, their Eq. (5)), we do not take the square-root of the dollar trading volume in this regression, such that our price impact measure has a straightforward interpretation: the percentage price change per unit of dollar trading volume." (p. 24). WRDS Intraday Indicators provide price impact estimates only when using the square-root of dollar trading volume, see Formula 31 of WRDS (WRDS) (MTAQ) and WRDS (2023) (DTAQ). To approximate the former from the later, we divide the later by the daily square-root of dollar trading volume.

```
data_complete[, `:=`(tsignsqrtdvol2, tsignsqrtdvol2 * 1e+06)]
data_complete[, `:=`(lambda, tsignsqrtdvol2/sqrt(vol_usd))]
```

We then estimate price impact "as the moving average of the estimated daily price impact parameters for that stock over the past 20 trading days, where we set negative price impact estimates to zero. To further reduce the influence of outliers, we cross-sectionally winsorize the resulting expected price impact estimates each day at the 95% level." (p. 24)

```
data_complete[, `:=`(pi_z20, lambda)]
data_complete[pi_z20 < 0, `:=`(pi_z20, 0)]

data_complete[order(date), `:=`(pi_z20, frollmean(pi_z20, n = 20,
    align = "right")), by = .(permno)]

data_complete[, `:=`(pi_z20_w, pmin(pi_z20, quantile(pi_z20,
    0.95, na.rm = TRUE))), by = .(date)]
data_complete[, `:=`(lambda_w, pmin(lambda, quantile(lambda,
    0.95, na.rm = TRUE))), by = .(date)]
```

Our main variable of interest are the signed and unsigned (absolute) product of price impact and order imbalance:

```
data_complete[, `:=`(lambda_oib, pi_z20_w * oib)]

data_complete[, `:=`(abs_lambda_oib, abs(lambda_oib))]
```

```
no_sample_stocks <- length(row.names(unique(data_complete[, "permno"])))
no_sample_days <- length(row.names(unique(data_complete[, "date"])))
no_sample_trades <- sum(data_complete[, "total_n_trades_m"])
```

In the reference paper the "final sample consists of all 1,388 NYSE stocks that survive these data screens over 2001-2014 and is based on a total of 18,626,168,999 signed trades." In the replication we have 1,559 stocks and a total of 18,261,750,964 trades.

2.5 Summary statistics

We report cross-sectional summary statistics in Table 1.

```
data_summary <- data_complete[, .(days = .N, ret = mean(ret,
    na.rm = T), PQSPR = mean(PQSPR, na.rm = T), lambda_w = mean(lambda_w,
    na.rm = T), oib = mean(oib, na.rm = T), lambda_oib = mean(lambda_oib,
    na.rm = T)), by = .(permno)]

data_means <- colMeans(data_summary, na.rm = T)
obs_fct <- function(x) sum(!is.na(x))
data_obs <- colwise(obs_fct)(data_summary)
data_sd <- sqrt(colwise(var)(data_summary, na.rm = T))
data_min <- colwise(min)(data_summary, na.rm = T)
data_median <- colwise(median)(data_summary, na.rm = T)
data_max <- colwise(max)(data_summary, na.rm = T)
```

```
probs <- c(0.25, 0.75)
quantiles <- colwise(quantile)(data_summary, probs = probs, na.rm = T)
```

Table 1 – Cross-sectional summary statistics of time-series averages

This table reports the cross-sectional (across the 1,559 NYSE stocks in the sample) mean, standard deviation, first quartile, median, and third quartile of the time-series average by stock of the daily percentage return from corporate action adjusted end-of-day mid-quotes (*Return*; in %; cross-sectionally winsorized each day at the 0.1% and 99.9% level), the daily percentage average proportional quoted spread (*PQSPR*; in %), the price impact defined as the percentage price change associated with a million US dollars of trading volume (λ; expressed as %Return/US$m.; cross-sectionally winsorized each day at the 95% level), the daily order imbalance or difference between the total US dollar volume of trades initiated by buyers and sellers expressed in millions of dollars (OIB; in US$m.), and the daily private information measure $\lambda \times OIB$. The first column indicates the number of stocks over which the summary statistics are computed. The second column indicates the average number of days a stock is in the sample. The sample includes 1,559 NYSE stocks during our sample period 2001-2014. The factor to adjust daily closing mid-quote data for corporate actions is from CRSP.

	#stocks	days	mean	stddev	25%	median	75%
Return	1,559	1,884	0.07	0.17	0.04	0.07	0.10
PQSPR	1,559	1,884	0.19	0.16	0.09	0.14	0.24
λ	1,559	1,884	0.27	2.10	0.01	0.16	0.62
OIB	1,559	1,884	1.27	3.03	0.00	0.32	1.34
$\lambda \times OIB$	1,559	1,884	0.27	0.68	0.03	0.22	0.42

3 Empirical results

We now replicate results of Bongaerts et al. (2025) using data from 2001 to 2014 from WRDS Intraday Indicators. Table 2 reports pooled, pairwise correlations.

```
table2_corr <- cor(data_complete[, .(lambda_oib, abs_lambda_oib,
    pi_z20_w, oib, PQSPR, vol_usd, ret)], use = "pairwise.complete.obs")

# table2_corr_t <- cor.test( data_complete[, .(lambda_oib,
# abs_lambda_oib, pi_z20_w, oib, PQSPR, vol_usd, ret)], use
# = 'pairwise.complete.obs')
```

Table 2 – Pooled correlations of daily private information, liquidity, order imbalance, and returns

This table reports pooled Pearson correlation coefficients between nine daily stock-specific variables: the signed and the absolute private information measure ($\lambda \times OIB$ and $\lambda \times |OIB|$), the price impact (λ), the dollar order imbalance (OIB), the proportional quoted spread ($PQSPR$), the dollar trading volume ($Volume$), the stock and returns ($Return$). We refer to Table 1 for a description of these variables. The sample includes 1,559 NYSE stocks during our sample period 2001-2014. The table reports p-values in parentheses below the correlations.

	$\lambda \times OIB$	$\lambda \times \lvert OIB \rvert$	λ	OIB	$PQSPR$	$Volume$	$Return$	$G.\ PIN$	$B.\ PIN$
$\lambda \times \lvert OIB \rvert$	0.27	1.000							
	(0.00)								
λ	0.21	0.04	1.000						
	(0.00)	(0.00)							
OIB	0.08	0.20	-0.02	1.000					
	(0.00)	(0.00)	(0.00)						
$PQSPR$	0.15	0.02	0.65	-0.02	1.000				
	(0.00)	(0.00)	(0.00)	(0.00)					
$Volume$	-0.00	-0.00	-0.17	0.01	-0.26	1.000			
	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)				
$Return$	0.05	0.10	0.01	0.03	-0.00	0-0.01	1.000		
	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)			

As in the reference paper, Table 3 shows summary statistics of portfolios sorted by private information. "Portfolios are formed every month and all variables are estimated using data from the previous month." (caption of Table 3 Bongaerts et al., 2025)

```
summary <- data_complete[, .(abs_lambda_oib = mean(abs_lambda_oib,
    na.rm = T), marketcap = mean(marketcap, na.rm = T), RETSQ = mean(RETSQ,
    na.rm = T), PQSPR = mean(PQSPR, na.rm = T), pi_z20_w = mean(pi_z20_w,
    na.rm = T)), by = .(permno, year, month)]

summary <- summary[!is.na(pi_z20_w)]

summary[, `:=`(time, as.yearmon(paste(year, month, sep = "-"),
    "%Y-%m"))]
setkey(summary, permno, time)

summary[order(time), `:=`(f_abs_lambda_oib, shift(abs_lambda_oib,
```

```
    -1, fill = NA)), by = .(permno)]

# to address missing lagged values, drop these:
summary[, `:=`(l_time, shift(time, 1)), by = .(permno)]
summary <- summary[l_time + 1/12 == time, ]


summary[, `:=`(quantile_v, dplyr::ntile(f_abs_lambda_oib, n = 5)),
    by = .(year, month)]

test_abs_lambda_oib <- wilcox.test(abs_lambda_oib ~ quantile_v,
    data = summary[quantile_v == 1 | quantile_v == 5])
test_marketcap <- wilcox.test(marketcap ~ quantile_v, data = summary[quantile_v ==
    1 | quantile_v == 5])
test_RETSQ <- wilcox.test(RETSQ ~ quantile_v, data = summary[quantile_v ==
    1 | quantile_v == 5])
test_PQSPR <- wilcox.test(PQSPR ~ quantile_v, data = summary[quantile_v ==
    1 | quantile_v == 5])
test_pi_z20_w <- wilcox.test(pi_z20_w ~ quantile_v, data = summary[quantile_v ==
    1 | quantile_v == 5])

summary <- summary[, .(stocks = uniqueN(permno), abs_lambda_oib = median(abs_lambda_oi
    marketcap = median(marketcap, na.rm = T), RETSQ = median(RETSQ),
    PQSPR = median(PQSPR), pi_z20_w = median(pi_z20_w)), by = quantile_v]


diff_summary <- summary[quantile_v == 5] - summary[quantile_v ==
    1]
```

Table 3 – Characteristics of stocks with large vs. small private information shocks

This table reports the following median characteristics of quintile portfolios sorted on the absolute private information measure ($\lambda \times |OIB|$) each month: the market capitalization in billions of dollars ($Mktcap$), the squared stock return ($Volatility$), the proportional quoted spread ($PQSPR$), the price impact defined as the percentage price change associated with a million dollars of trading volume (λ), the number of analysts covering the firm (*Analyst coverage*), and the dispersion in analysts' earnings forecasts, defined as the standard deviation of annual analysts' earnings per share forecasts scaled by the mean forecast (*Analyst dispersion*). Portfolios are formed every month and all variables are estimated using data from the previous month. The sample includes 1,559 NYSE stocks during our sample period 2001-2014. *, **, and *** indicate significance at the 10%, 5%, and 1% levels based on a **Wilcoxon rank sum** test, respectively.

$\lambda \times \|OIB\|$ Quintiles	$Mktcap$	$Volatility$	$PQSPR$	λ
1	4.93	2.05	0.06	0.05
2	3.74	2.34	0.07	0.09
3	2.74	2.62	0.09	0.14
4	1.96	3.18	0.12	0.25
5	1.33	3.97	0.16	0.53
Difference [5] - [1]	-3.60***	1.92***	0.10***	0.48***

To estimate Fama-MacBeth regression with Newey-West standard errors, we define the following function:

```
lags <- 5
nwt_stats <- function(x) t(coeftest(lm(x ~ 1), vcov = NeweyWest(lm(x ~
    1), lag = lags, prewhite = FALSE))[, c("t value", "Pr(>|t|)")])

fmb_reg <- function(data, formula) {

    vars <- all.vars(formula)
    data_no_na <- data[complete.cases(data[, ..vars])]

    fmb_results <- data_no_na[, {
        reg <- lm(formula, data = .SD)
        c(reg = as.list(coef(reg)), r_squared = summary(reg)$r.squared)
    }, by = .(date)]
```

```
    fmb_results[, `:=`(date, NULL)]

    fmb_coef <- colwise(mean, na.rm = T)(fmb_results)
    fmb_t_stat <- colwise(nwt_stats)(fmb_results)

    days <- length(unique(data_no_na[, date]))

    return(list(coef = fmb_coef, t_stats = fmb_t_stat, days = days,
        r_squared = fmb_coef[["r_squared"]]))
}
```

As in the reference paper, in regressions returns are not scaled by 100, i.e., they are not in per cent:

```
data_complete[, `:=`(ret, ret/100)]
data_complete[, `:=`(RETSQ, ret^2)]

data_complete[, `:=`(lambda_oib, lambda_oib/1000)]
data_complete[, `:=`(abs_lambda_oib, abs_lambda_oib/1000)]
```

We now replicate three tables from the reference paper showing results of daily Fama-MacBeth regressions explaining returns by contemporaneous private information (Table 5 of Bongaerts et al., 2025); explaining returns by previous private information (Table 6); and explaining volatility by previous private information (Table 8).

```
data_complete[order(date), `:=`(lag_Return, shift(ret, 1, fill = NA)),
    by = .(permno)]

data_complete[, `:=`(marketcap_inv, 1/marketcap)]

formula <- "ret  ~ lag_Return + lambda_oib"
fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
reg1 <- fmb_stock_reg$coef
t_stats1 <- fmb_stock_reg$t_stats
```

```
days1 <- fmb_stock_reg$days
reg1[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

formula <- "ret  ~ lag_Return + lambda_oib + oib + pi_z20_w"
fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
reg2 <- fmb_stock_reg$coef
t_stats2 <- fmb_stock_reg$t_stats
days2 <- fmb_stock_reg$days
reg2[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

formula <- "ret  ~ lag_Return + lambda_oib + oib + pi_z20_w + oib* marketcap_inv"
fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
reg3 <- fmb_stock_reg$coef
t_stats3 <- fmb_stock_reg$t_stats
days3 <- fmb_stock_reg$days
reg3[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

formula <- "ret  ~ lag_Return + lambda_oib + oib + pi_z20_w + oib* PQSPR"
fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
reg4 <- fmb_stock_reg$coef
t_stats4 <- fmb_stock_reg$t_stats
days4 <- fmb_stock_reg$days
reg4[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared
```

Table 4 – Daily Fama-MacBeth regressions of returns on contemporaneous private information

This table reports the time-series averages of the estimated slope coefficients from daily Fama-MacBeth regressions to explain cross-sectional variation in daily stock returns. The dependent variable is the end-of-day mid-quote return of stock i on day d ($Return_{i,d}$). The independent variables are: the return of stock i on day $d-1$ ($Return_{i,d-1}$), the order imbalance of stock i on day d ($OIB_{i,d}$), the price impact of stock i on day d – calculated as the stock's average price impact estimate over the past 20 days with setting non-positive price impact estimates to zero ($\lambda_{i,d}$), the inverse of the market capitalization of stock i at the beginning of each year ($1/Mktcap_{i,y-}$), the proportional quoted spread of stock i on day $d-1$ ($PQSPR_{i,d-1}$), as well as various interaction terms. The table reports Fama-MacBeth t-statistics with Newey-West corrections in parentheses below the average coefficients. Some coefficients have been scaled for ease of presentation. Intercepts are suppressed to conserve space. The final two rows report the R^2 and the number of regressions. The sample includes 1,559 NYSE stocks during our sample period 2001-2014. *, **, and *** indicate significance at the 10%, 5%, and 1% levels, respectively.

	Dependent variable: $Return_{i,d}$			
	(1)	(2)	(3)	(4)
$Return_{i,d-1}$	-0.01***	-0.01***	-0.01***	-0.01***
	(-5.15)	(-5.60)	(-6.04)	(-6.05)
$\Lambda_{i,d} \times OIB_{i,d}$	1.13***	1.18***	1.48***	1.05***
	(25.63)	(27.13)	(27.18)	(22.07)
$\Lambda_{i,d}$		0.00***	0.00**	0.00***
		(4.68)	(2.06)	(7.67)
$OIB_{i,d}$		0.00**	0.00***	-0.00***
		(2.41)	(3.76)	(-17.26)
$1/Marketcap_{i,d}$			0.00***	
			(5.37)	
$OIB_{i,d} \times 1/Marketcap_{i,d}$			-0.00***	
			(-3.04)	
$PQSPR_{i,d}$				-0.00***
				(-3.26)
$OIB_{i,d} \times PQSPR_{i,d}$				0.00***
				(13.44)
R^2 [%]	3.89	5.55	6.64	6.82
#Regressions	3,479	3,479	3,479	3,479

```
data_complete[order(date), `:=`(lag_Return, shift(ret, 1, fill = NA)),
    by = .(permno)]
data_complete[order(date), `:=`(l_abs_return_day, shift(abs(return_day),
    1, fill = NA)), by = .(permno)]
data_complete[order(date), `:=`(l_pi_abs_oib, shift(abs_lambda_oib,
    1)), by = .(permno)]
data_complete[order(date), `:=`(l_piz, shift(pi_z20_w, 1)), by = .(permno)]
data_complete[order(date), `:=`(l_abs_oib, shift(abs(oib), 1)),
    by = .(permno)]
data_complete[order(date), `:=`(l_piz_marketcap, shift(pi_z20_w *
    marketcap, 1)), by = .(permno)]

formula <- "ret  ~ lag_Return"
fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
reg1 <- fmb_stock_reg$coef
t_stats1 <- fmb_stock_reg$t_stats
days1 <- fmb_stock_reg$days
reg1[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

formula <- "ret ~ lag_Return * l_pi_abs_oib"
fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
reg2 <- fmb_stock_reg$coef
t_stats2 <- fmb_stock_reg$t_stats
days2 <- fmb_stock_reg$days
reg2[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

formula <- "ret  ~ lag_Return + lag_Return * l_piz_marketcap"
fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
reg3 <- fmb_stock_reg$coef
t_stats3 <- fmb_stock_reg$t_stats
days3 <- fmb_stock_reg$days
reg3[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared
```

```
formula <- "ret ~ lag_Return * l_piz + lag_Return * l_abs_oib"
fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
reg4 <- fmb_stock_reg$coef
t_stats4 <- fmb_stock_reg$t_stats
days4 <- fmb_stock_reg$days
reg4[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

formula <- "ret ~ lag_Return * l_pi_abs_oib + lag_Return * l_piz + lag_Return * l_abs_
fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
reg5 <- fmb_stock_reg$coef
t_stats5 <- fmb_stock_reg$t_stats
days5 <- fmb_stock_reg$days
reg5[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

formula <- "ret ~ lag_Return * l_pi_abs_oib + lag_Return * l_piz + lag_Return * l_abs_
fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
reg6 <- fmb_stock_reg$coef
t_stats6 <- fmb_stock_reg$t_stats
days6 <- fmb_stock_reg$days
reg6[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared
```

Table 5 – Daily Fama-MacBeth regressions of returns on previous day private information

This table reports the time-series averages of the estimated slope coefficients from daily Fama-MacBeth regressions to explain cross-sectional variation in daily stock returns. The dependent variable is the end-of-day mid-quote return of stock i on day d ($Return_{i,d}$). The independent variables are: the return of stock i on day $d-1$ ($Return_{i,d-1}$), the market capitalization of stock i on day $d-1$, the previous day absolute private information measure $\lambda \times OIB$ computed as the product of the price impact of stock i on day $d-1$ – which is calculated as the stock's average price impact estimate over the past 20 days with setting non-positive price impact estimates to zero ($\lambda_{i,d-1}$) and the absolute order imbalance of stock i on day $d-1$ ($|OIB_{i,d-1}|$), $\lambda_{i,d-1}$ and $|OIB_{i,d-1}|$ separately, as well as various interaction terms. In column 5 we also control for the absolute day-return, i.e., the return from open to close. Some coefficients have been scaled for ease of presentation. Intercepts are suppressed to conserve space. The final three rows report the R^2, the adjusted R^2, and the number of regressions. The sample includes 1,559 NYSE stocks during our sample period 2001-2014. *, **, and *** indicate significance at the 10%, 5%, and 1% levels, respectively.

	Dependent variable: $Return_{i,d}$					
	(1)	(2)	(3)	(4)	(5)	(6)
$Return_{i,d-1}$	-0.01***	-0.01***	-0.00*	-0.01**	-0.01**	-0.00
	(-2.84)	(-3.90)	(-1.72)	(-1.97)	(-2.22)	(-0.60)
$\Lambda_{i,d-1} \times \lvert OIB_{i,d-1}\rvert$		0.20***			0.19***	0.19***
		(5.14)			(4.83)	(5.18)
$Return_{i,d-1} \times \Lambda_{i,d-1} \times \lvert OIB_{i,d-1}\rvert$		3.01**			2.82**	3.44**
		(2.19)			(1.97)	(2.36)
$\Lambda_{i,d-1} \times Marketcap_{i,d-1}$			0.00			
			(0.25)			
$Return_{i,d-1} \times \Lambda_{i,d-1} \times Marketcap_{i,d-1}$			0.00			
			(0.11)			
$\Lambda_{i,d-1}$				0.00***	0.00**	0.00**
				(3.12)	(2.18)	(2.01)
$Return_{i,d-1} \times \Lambda_{i,d-1}$				-0.00**	-0.01***	-0.01***
				(-2.17)	(-3.26)	(-2.90)
$\lvert OIB_{i,d-1}\rvert$				-0.00	-0.00***	-0.00***
				(-1.31)	(-2.96)	(-2.82)
$Return_{i,d-1} \times \lvert OIB_{i,d-1}\rvert$				0.00*	0.00	0.00
				(1.92)	(0.81)	(0.67)
$\lvert Day - Return_{i,d-1}\rvert$						0.00
						(0.82)
$Return_{i,d-1} \times \lvert Day - Return_{i,d-1}\rvert$						-0.00*
				(1.92)	(0.81)	(0.67)
R^2 [%]	1.57	2.40	2.47	3.68	4.41	5.55
#Regressions	3,496	3,478	3,478	3,478	3,478	3,478

```
data_complete[order(date), `:=`(l_RETSQ, shift(RETSQ, 1)), by = .(permno)]
data_complete[order(date), `:=`(l2_RETSQ, shift(RETSQ, 2)), by = .(permno)]
data_complete[order(date), `:=`(l3_RETSQ, shift(RETSQ, 3)), by = .(permno)]
data_complete[order(date), `:=`(l4_RETSQ, shift(RETSQ, 4)), by = .(permno)]
data_complete[order(date), `:=`(l5_RETSQ, shift(RETSQ, 5)), by = .(permno)]
data_complete[order(date), `:=`(l_pi_abs_oib, shift(abs_lambda_oib,
    1)), by = .(permno)]
data_complete[order(date), `:=`(l_piz, shift(pi_z20_w, 1)), by = .(permno)]
data_complete[order(date), `:=`(l_abs_oib, shift(abs(oib), 1)),
    by = .(permno)]
data_complete[order(date), `:=`(l_PQSPR, shift(PQSPR, 1)), by = .(permno)]
data_complete[order(date), `:=`(l_abs_return_day, shift(abs(return_day),
    1, fill = NA)), by = .(permno)]

data_complete[, `:=`(marketcap_inv, 1/marketcap)]
data_complete[order(date), `:=`(l_marketcap_inv, shift(marketcap_inv,
    1)), by = .(permno)]

formula <- "RETSQ ~ lag_Return + l_RETSQ"
fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
reg1 <- fmb_stock_reg$coef
t_stats1 <- fmb_stock_reg$t_stats
days1 <- fmb_stock_reg$days
reg1[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

formula_2 <- "RETSQ ~ lag_Return + l_RETSQ + l2_RETSQ + l3_RETSQ + l4_RETSQ + l5_RETSQ
fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula_2))
reg2 <- fmb_stock_reg$coef
t_stats2 <- fmb_stock_reg$t_stats
days2 <- fmb_stock_reg$days
reg2[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared
```

```r
formula_3 <- paste0(formula_2, "+ l_pi_abs_oib +  l_piz +  l_abs_oib")
fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula_3))
reg3 <- fmb_stock_reg$coef
t_stats3 <- fmb_stock_reg$t_stats
days3 <- fmb_stock_reg$days
reg3[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

formula <- paste0(formula_3, "+ l_abs_oib* marketcap_inv")
fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
reg4 <- fmb_stock_reg$coef
t_stats4 <- fmb_stock_reg$t_stats
days4 <- fmb_stock_reg$days
reg4[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

formula <- paste0(formula_3, "+ l_abs_oib*l_PQSPR")
fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
reg5 <- fmb_stock_reg$coef
t_stats5 <- fmb_stock_reg$t_stats
days5 <- fmb_stock_reg$days
reg5[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

formula <- paste0(formula_3, "+ l_abs_return_day")
fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
reg6 <- fmb_stock_reg$coef
t_stats6 <- fmb_stock_reg$t_stats
days6 <- fmb_stock_reg$days
reg6[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared
```

Table 6 – Daily Fama-MacBeth regressions of volatility on previous day private information

This table reports the time-series averages of the estimated slope coefficients from daily Fama-MacBeth regressions to explain cross-sectional variation in daily stock return volatility. The dependent variable is the squared end-of-day mid-quote return of stock i on day d ($Return^2_{i,d}$). The independent variables are: the return on day $d-1$ ($Return_{i,d-1}$), the squared return on days $d-1$ through day $d-5$ ($Return^2_{i,d-x}$), the absolute order imbalance on day $d-1$ ($|OIB_{i,d-1}|$), the price impact on day $d-1$ – calculated as the stock's average price impact estimate over the past 20 days with setting non-positive price impact estimates to zero ($\lambda_{i,d-1}$), the inverse of the market capitalization at the beginning of each year ($1/Mktcap_{i,y-}$), the proportional quoted spread on day $d-1$ ($PQSPR_{i,d-1}$), and various interaction terms. In column 6 we also control for the absolute day-return, i.e., the return from open to close. The table reports Fama-MacBeth t-statistics with Newey-West corrections in parentheses below the average coefficients. Intercepts are suppressed to conserve space. The final two rows report the R^2 and the number of regressions. The sample includes 1,559 NYSE stocks during our sample period 2001-2014. *, **, and *** indicate significance at the 10%, 5%, and 1% levels, respectively.

	Dependent variable: $Return^2_{i,d}$					
	(1)	(2)	(3)	(4)	(5)	(6)
$Return_{i,d-1}$	-0.00***	-0.00***	-0.00***	-0.00***	-0.00***	-0.00***
	(-7.21)	(-7.20)	(-7.10)	(-7.10)	(-6.86)	(-7.20)
$Return^2_{i,d-1}$	0.16***	0.12***	0.11***	0.11***	0.10***	0.06***
	(54.00)	(51.76)	(46.17)	(44.26)	(43.49)	(18.69)
$Return^2_{i,d-2}$		0.08***	0.07***	0.07***	0.07***	0.06***
		(36.36)	(32.56)	(31.21)	(31.77)	(30.08)
$Return^2_{i,d-3}$		0.06***	0.06***	0.05***	0.05***	0.05***
		(34.60)	(31.98)	(30.76)	(30.68)	(30.31)
$Return^2_{i,d-4}$		0.06***	0.06***	0.05***	0.05***	0.05***
		(28.02)	(26.22)	(25.25)	(25.02)	(25.11)
$Return^2_{i,d-5}$		0.06***	0.06***	0.06***	0.05***	0.06***
		(30.60)	(27.89)	(26.82)	(26.73)	(26.79)
$\Lambda_{i,d-1} \times \lvert OIB_{i,d-1}\rvert$			0.04***	0.02***	0.02***	0.04***
			(11.82)	(5.52)	(5.03)	(11.60)
$\Lambda_{i,d-1}$			0.00***	0.00***	0.00*	0.00***
			(18.60)	(10.12)	(1.80)	(17.64)
$\lvert OIB_{i,d-1}\rvert$			-0.00***	-0.00***	-0.00***	-0.00***
			(-7.00)	(-8.09)	(-19.89)	(-6.35)
$1/Marketcap_{i,d-1}$				-0.00***		
				(14.83)		
$\lvert OIB_{i,d-1}\rvert \times 1/Marketcap_{i,d-1}$				0.00***		
				(6.18)		
$PQSPR_{i,d-1}$					0.00***	
					(21.23)	
$\lvert OIB_{i,d-1}\rvert \times PQSPR_{i,d-1}$					0.00***	
					(19.00)	
$\lvert Day - Return_{i,d-1}\rvert$						0.00***
						(30.12)
R^2 [%]	3.38	7.16	8.50	9.29	9.32	9.09
#Regressions	3,496	3,492	3,478	3,478	3,478	3,478

4 Functions and Setup Code

```
library(zoo)
library(plyr)
library(dplyr)
library(sqldf)
options(sqldf.driver = "SQLite")

require(RPostgres) | require(RPostgreSQL)

library(reshape2)
library(ggplot2)
library(xts)

# for R tidy in knitr
library(formatR)

library(MatchIt)
library(plm)

# for fwrite, fread
library(data.table)
library(anytime)
library(lubridate)

# For panel regression
library(fixest)
library(lmtest)
library(roll)

# For Newey-West std.error
library(sandwich)
```

```r
# for storing passwords
library(keyring)

# Setup default number layouts
format_numeric <- "%.2f"
format_integer <- "%.0f"
format_percent <- "%.2f"
format_significance <- "(%.2f)"

pretty_numbers <- function(format, number, p_value = -1, t_value = -1,
    ...) {
    result <- sprintf(format, number, ...)

    if (format == format_integer || format == format_numeric) {
        result <- prettyNum(result, big.mark = ",", scietific = F)
    }

    if (p_value >= 0) {

        stars <- ""
        if (p_value < 0.1) {
            stars <- "*"
        }
        if (p_value < 0.05) {
            stars <- "**"
        }
        if (p_value < 0.01) {
            stars <- "***"
        }
        result <- sprintf("%1.2f%2$-3s", number, stars)
    }
    if (t_value > 0) {
```

```
        result <- prettyNum(result, big.mark = ",", scietific = F)
        result <- paste0("(", result, ")")
    }
    return(result)
}

sanitize <- function(str) {
    result <- str
    result <- gsub("&", "\\&", result, fixed = TRUE)
    result <- gsub("_", "\\_", result, fixed = TRUE)

    return(result)
}

# Setup default charting layout

theme_report <- function(base_size = 30, base_family = "", ...) {
    theme_bw(base_size = base_size, base_family = base_family) +
        theme(line = element_line(colour = "black", size = 4,
            linetype = "solid"), legend.position = "top", legend.justification = c("ri
            "top"), legend.key.height = unit(3, "line"), legend.key.width = unit(3,
            "cm"), legend.title = element_text(size = base_size *
            0.4), legend.text = element_text(size = base_size *
            0.4), legend.background = element_blank(), legend.key = element_blank(),
            strip.text.x = element_text(size = base_size * 1,
                colour = "black"), strip.background = element_rect(color = "black",
                size = 4), text = element_text(colour = "black",
                size = base_size * 1), axis.title = element_text(size = base_size *
                0.5), axis.text = element_text(size = base_size *
                0.4), axis.ticks = element_line(colour = "black",
                size = 2), panel.grid = element_blank(), panel.grid.minor = element_bl
            panel.background = element_rect(fill = NA, colour = "black",
```

```
                size = 4))
}
```

```
sessionInfo()

## R version 4.2.0 (2022-04-22)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 24.04.2 LTS
##
## Matrix products: default
## BLAS/LAPACK: /cvmfs/soft.ccr.buffalo.edu/versions/2023.01/easybuild/software/avx512
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8 LC_NUMERIC=C         LC_TIME=C
##  [4] LC_COLLATE=C         LC_MONETARY=C        LC_MESSAGES=C
##  [7] LC_PAPER=C           LC_NAME=C            LC_ADDRESS=C
## [10] LC_TELEPHONE=C       LC_MEASUREMENT=C     LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] keyring_1.3.2     sandwich_3.0-1    roll_1.1.7        lmtest_0.9-40
##  [5] fixest_0.12.1     lubridate_1.8.0   anytime_0.3.10    data.table_1.14.2
##  [9] plm_2.6-4         MatchIt_4.3.4     formatR_1.12      xts_0.12.1
## [13] ggplot2_3.5.0     reshape2_1.4.4    RPostgreSQL_0.7-5 DBI_1.2.3
## [17] sqldf_0.4-11      RSQLite_2.2.12    gsubfn_0.7        proto_1.0.0
## [21] dplyr_1.0.8       plyr_1.8.7        zoo_1.8-10        knitr_1.38
##
## loaded via a namespace (and not attached):
##  [1] bit64_4.0.5         RcppParallel_5.1.5  Formula_1.2-4
##  [4] Rdpack_2.3          assertthat_0.2.1    highr_0.9
##  [7] blob_1.2.3          yaml_2.3.5          numDeriv_2016.8-1.1
```

```
## [10] pillar_1.7.0        backports_1.4.1     lattice_0.20-45
## [13] glue_1.6.2          chron_2.3-56        digest_0.6.29
## [16] rbibutils_2.2.8     colorspace_2.0-3    pkgconfig_2.0.3
## [19] purrr_0.3.4         scales_1.3.0        collapse_2.0.18
## [22] tibble_3.1.6        generics_0.1.3      ellipsis_0.3.2
## [25] cachem_1.0.6        withr_2.5.0         maxLik_1.5-2
## [28] cli_3.6.2           magrittr_2.0.3      crayon_1.5.1
## [31] memoise_2.0.1       evaluate_0.15       fansi_1.0.3
## [34] nlme_3.1-157        MASS_7.3-57         dreamerr_1.4.0
## [37] tools_4.2.0         lifecycle_1.0.4     stringr_1.5.1
## [40] munsell_0.5.0       sodium_1.3.2        compiler_4.2.0
## [43] tinytex_0.38        rlang_1.1.3         grid_4.2.0
## [46] rstudioapi_0.13     miscTools_0.6-26    rappdirs_0.3.3
## [49] tcltk_4.2.0         gtable_0.3.0        R6_2.5.1
## [52] bdsmatrix_1.3-4     fastmap_1.1.0       bit_4.0.4
## [55] utf8_1.2.2          filelock_1.0.3      stringmagic_1.1.2
## [58] stringi_1.7.6       parallel_4.2.0      Rcpp_1.0.8.3
## [61] vctrs_0.6.5         tidyselect_1.1.2    xfun_0.30
```

```
lapply(dbListConnections(PostgreSQL()), dbDisconnect)

## [[1]]
## [1] TRUE
```

References

Aghbabali, T., W. S. Choi, T. Im, D. Rösch, and P. Roy (2025). How to: Dynamic Reports and Reproducible Research. *working paper, University at Buffalo*.

Bongaerts, D., D. Rösch, and M. van Dijk (2025). Cross-sectional identification of private information. *Reiview of Asset Pricing Studies, forthcoming*.

Rösch, D. M., A. Subrahmanyam, and M. A. van Dijk (2017). The Dynamics of Market Efficiency. *Review of Financial Studies 30*, 1151–1187.

WRDS (????). IID Formula Note .

WRDS (2023). WRDS INTRADAY INDICATOR DATA: Millisecond IID V2.0 User Manual.

5 Appendix: Source Code

Listing 1 – Source Code

```
%
% This is an knitR file. To build it, it is easiest to use a local version of RStudio,
% alternatively, one could try RStudio as provided by WRDS: https://wrds-rstudio.wharton.upenn.edu/
% alternatively, one can build the file from the command line: R CMD Sweave --pdf floor_trading_tables_jf.Rnw
%
% By default, RStudio compiles PDF using Sweave. However, it is required to use `knitr` % instead. To do this, update the setting in:
% "Tools" -> "Project Options" -> "Weave Rnw files using" -> "knitr", or include the following directive:
% %% !Rnw weave = knitr
%


\documentclass[authoryear,longnamesfirst,12pt]{article}

\usepackage{filecontents}
\begin{filecontents}{\jobname.bib}
@article{WRDS2025,
title = {{WRDS INTRADAY INDICATOR DATA: Millisecond IID V2.0 User Manual}},
year = "2023",
author = "WRDS",
}

@article{WRDS2020,
title = {{IID Formula Note }},
year = "????",
author = "WRDS",
}

@article{Roesch2017,
author = {R\"{o}sch, Dominik M. and Subrahmanyam, Avanidhar and van Dijk, Mathijs A.},
title = {{The Dynamics of Market Efficiency}},
journal = {Review of Financial Studies},
volume = {30},
pages = {1151-1187},
year = {2017},
}

@article{Aghbabali2025,
author = {Toghrul Aghbabali and Wan Soo Choi and Taihun Im and Dominik R\"{o}sch and Prince Roy},
title = {{How to: Dynamic Reports and Reproducible Research}},
journal = {working paper, University at Buffalo},
year = {2025},
}


@article{Bongaerts2025,
author = {Dion Bongaerts and Dominik R\"{o}sch and Mathijs van Dijk},
title = {{Cross-sectional identification of private information}},
journal = {Reiview of Asset Pricing Studies, forthcoming},
year = {2025},
}



\end{filecontents}


\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}


\usepackage{natbib}
```

```
\usepackage{listings}  % Allows reading and displaying source files
\usepackage{xcolor}  % For color formatting (optional)

% To handle both R and Latex comments after line breaks in listing package:
% we add 1%%1# in front of each new line, which works both in R and Latex
% see https://tex.stackexchange.com/questions/376734/listings-make-linebroken-line-comment-a-comment
\makeatletter
\lst@AddToHook{AfterBeginComment}{%
    \CommentLinetrue%
}
\makeatother

%%%%%%%%%%%%%%%%%%%%%%


% to get URL formating
\usepackage[hidelinks]{hyperref}
\hypersetup{
    colorlinks=false,
}



% for table setup
\usepackage{tabularx}
\usepackage[margin=10pt,font=footnotesize,labelfont=bf,labelsep=endash]{caption}
\usepackage{booktabs}
\usepackage{subcaption}

\usepackage{mathtools}
\DeclarePairedDelimiter\abs{\lvert}{\rvert}

\usepackage{floatrow}
\floatsetup[table]{font={singlespacing,footnotesize}}
\floatsetup[table]{capposition=top}
\floatsetup[figure]{capposition=top}

% Set spacings
\usepackage[margin=3cm,includefoot]{geometry}
\usepackage{setspace}
% \doublespacing
\onehalfspacing

\begin{document}

% so that we can import into other file using standalone, these lines need to be after begin{document}
\newif\ifCommentLine%
\newcommand*{\CommentLineContinued}{\ifCommentLine 1\%\%1\#\space\fi}


\ifdefined\includemain

% drop the following if included into other file

\else

  \begin{center}
  \renewcommand{\thefootnote}{\fnsymbol{footnote}}
  \thispagestyle{empty}
  \begin{Large}
  \textbf{Code for:\\ Cross-sectional identification of private information}
  \end{Large}
  \normalsize

  \vspace{7mm}
  \textbf{Dion Bongaerts}\\
  Rotterdam School of Management, Erasmus University
```

```
  \vspace{5mm}
  \textbf{Dominik R\"{o}sch}\\
  University at Buffalo, State University of New York

  \vspace{5mm}
  \textbf{Mathijs van Dijk}\\
  Rotterdam School of Management, Erasmus University


  \vspace{3mm}

  \vspace{10mm}
  \textbf{ABSTRACT}\\
  \vspace{-4mm}
  \end{center}
  \begingroup
  \leftskip2em
  \rightskip\leftskip
This document provides code to replicate the results of ``Cross-Sectional Identification of Private Information'' \citep{Bongaerts2025}
      using data available through Wharton Research Data Services (WRDS).
  Specifically, it reproduces parts of Tables 1, 2, 3, 5, 6, and 8.
  The document is written as a knitr (R Markdown) file. To execute the code and generate a PDF, the recommended approach is to use a
      local installation of RStudio.
  Upon execution, the code automatically downloads and stores all necessary data from WRDS locally. To enable this, you must first
      configure WRDS access as described below.
  Note: While the published article uses data from Refinitiv/TRTH, this replication uses WRDS and TAQ data. The results are
      qualitatively robust to these differences in data sources and partially different methodologies, as discussed in the relevant
      sections below.
  \\
  \par
  \endgroup

  \pagebreak

\fi

\section{Discussion}

This code replicates the main results of Cross-Sectional Identification of Private Information, following a setup similar to that
      proposed in \cite{Aghbabali2025}. The document includes all necessary code to reproduce this PDF: it downloads the required data
      from WRDS, performs the analysis, and presents the results.
The actual R code used to generate this document is included in the Appendix at the end of the file.

Ideally, this code would fully reproduce the results of Cross-Sectional Identification of Private Information using the original data
      and methodology.
However, the reference study relies on Thomson Reuters Tick History (TRTH) data, which is prohibitively large (multiple terabytes),
      costly, and infrequently used in academic research.
Instead, this replication focuses on making the analysis as accessible as possible by using WRDS Intraday Indicators, which are
      available to anyone with access to WRDS and TAQ.
These indicators, derived from TAQ data, provide stock-day level estimates and therefore do not require analyzing massive tick-by-tick
      data.
They offer a practical alternative to working directly with the full TRTH (or TAQ) dataset.
For more details on WRDS Intraday Indicators, see \cite{WRDS2025}.

Using WRDS Intraday Indicators differs from \cite{Bongaerts2025} in two main ways:
\begin{itemize}
\item Quoted spreads (PQSPR) in \cite{Bongaerts2025}  is estimated as the average ``spread associated with all trades'' but in WRDS
      Intraday Indicators it is the time-weighted average.
\item Price impact in \cite{Bongaerts2025} is estimated as $\lambda$ from stock-day regressions, explaining 5-minute log-returns by the
      signed, trading volume within the interval. WRDS Intraday Indicators reports price impact from the same regressions but using
      the square-root of trading volume as the explanatory variable.
\end{itemize}
We discuss both differences in more details in later Sections.

The following sections, Sections~\ref{sec:3r_start} to \ref{sec:3r_end}, are more or less copied from \cite{Aghbabali2025}, they
      contain general setup steps, required for any project.

\section{Setup}
```

```
\label{sec:3r_start}

Before we run any code, we setup an error handler.
This ensures that if the code breaks at any point, e.g., because the working directory is not yet set correctly, the output will stop
      and display the error message at the end of the document.
<<setup_error_handler,echo=TRUE>>=

# see https://stackoverflow.com/q/74097101
knitr::knit_hooks$set(error = function(x, options) {
  ERROR_GLOBAL <<- x
  knitr::knit_exit()
})

knit_chunk <- knitr::knit_hooks$get("chunk")

knitr::knit_hooks$set(chunk = function(x, options){

  out <- x
  if (exists('ERROR_GLOBAL', inherits = TRUE)) {
    out <- paste0(out,'\n**stopped with error**:',ERROR_GLOBAL,'\n\n')
    options$size       <- "large"
    options$background <- '#FFDDDD'
    options$tidy       <- 'T'
    options$tidy.opts  <- list(width.cutoff=50)
    options$results    <-'markup'
  }

  knit_chunk(out, options)

})

@

\subsection{Prerequisites}
In this section we check whether the system can execute the analysis.

First we ensure that the system has enough free memory to successfully run the analysis.
Unfortunately, retrieving available memory is not straight forward and depends on, e.g., the operating system and whether the code is
      running on a virtual machine.
Because of that we do not provide any actual code, but ask users to manually verify that enough RAM is available.
A lack of RAM often explain sudden termination, without  generating any output files.
<<setup_check_memory>>=

#to do - we need to have a few GB's

@


Next, we load all required R libraries.
While this must be done at the beginning of the script, to not disrupt the reading flow we only evaluate the code here and show it at
      the end of the file, Section~\ref{R:setup}.
<<setup_dummy>>=

# The actual code is executed here, but shown only at the end of the file.

@
<<setup, echo=FALSE,warning=F,message=F>>=

library(zoo)
library(plyr)
library(dplyr)
library(sqldf)
options(sqldf.driver = "SQLite")

require(RPostgres) | require(RPostgreSQL)

library(reshape2)
library(ggplot2)
```

```
library(xts)

# for R tidy in knitr
library(formatR)

library(MatchIt)
library(plm)

# for fwrite, fread
library( data.table)
library(anytime)
library(lubridate)

# For panel regression
library(fixest)
library(lmtest)
library(roll)

# For Newey-West std.error
library(sandwich)
# for storing passwords
library(keyring)

# Setup default number layouts
format_numeric    <- "%.2f";
format_integer    <- "%.0f";
format_percent    <- "%.2f";
format_significance <- "(%.2f)";

pretty_numbers <- function(format, number, p_value = -1, t_value = -1, ...)
{
    result <- sprintf(format, number, ...)

    if (format == format_integer || format == format_numeric)
    {
        result <- prettyNum(result, big.mark=",",  scietific=F)
    }

    if (p_value >= 0) {

        stars <- ""
        if (p_value < 0.10) {  stars <- "*" }
        if (p_value < 0.05) {  stars <- "**" }
        if (p_value < 0.01) {  stars <- "***" }
        result <- sprintf("%1.2f%2$-3s", number, stars)
    }
    if (t_value >0) {
      result <- prettyNum(result, big.mark=",",  scietific=F)
      result <- paste0("(", result, ")")
    }
    return(result)
}

sanitize <- function(str) {
  result <- str
  result <- gsub("&", "\\&", result, fixed = TRUE)
  result <- gsub("_", "\\_", result, fixed = TRUE)

  return(result)
}

# Setup default charting layout

theme_report <- function(base_size = 30, base_family = "", ...) {
  theme_bw(base_size = base_size, base_family = base_family) +
    theme(
      line = element_line(colour = "black", size = 4, linetype = "solid"),
      legend.position = "top",
```

```
      legend.justification = c("right", "top"),
      legend.key.height = unit(3, "line"),
      legend.key.width = unit(3, "cm"),
      legend.title = element_text(size = base_size * 0.4),
      legend.text = element_text(size = base_size * 0.4),
      legend.background = element_blank(),
      legend.key = element_blank(),
      strip.text.x = element_text(size = base_size * 1.0, colour = "black"),
      strip.background = element_rect(color = "black", size = 4),
      text = element_text(colour = "black", size = base_size * 1.0),
      axis.title = element_text(size = base_size * 0.5),
      axis.text = element_text(size = base_size * 0.4),
      axis.ticks = element_line(colour = "black", size = 2),
      panel.grid = element_blank(),
      panel.grid.minor = element_blank(),
      panel.background = element_rect(fill = NA, colour = "black", size = 4)
    )
}

@

Similarly, we ensure that we compile the document using tinytex and that all required Latex packages are available.
If any package is missing we try to install it:
<<setup_latex_packages>>=

if(! tinytex::is_tinytex()) {
  stop("We use tinytex to install missing Latex packages, if you
  are using a different Latex version this might not be compatible.
  It might be easiest to run tinytex::install_tinytex()")
}

# if(! grepl("TinyTeX", Sys.which("pdflatex"))) {
# stop("It looks like you are using a different Latex version
# which might not be compatible.
# It might be easiest to switch to TinyTex, for example,
# by adding the PATH to pdflatex in TinyTex directory
# to PATH environmental variable. See the code in the Rnw file below.")
#
#   cat('Sys.setenv(PATH = paste0(tinytex::tinytex_root(),
#"\\bin\\windows;", Sys.getenv("PATH")))\n',
#     file = file.path(Sys.getenv("HOME"), ".Rprofile"),
#     append = TRUE)
# }

latex_packages <- c('listings', 'caption', 'mathtools', 'floatrow',
                    'setspace', 'cmap', 'filecontents')

for (pkg in latex_packages) {
  tinytex::check_installed(pkg) || tinytex::tlmgr_install(pkg)
}

@

\subsection{Setup variable names}
This section contains various code blocks to setup the analysis such as variables determining how the compiled PDF will look (e.g.,
      whether it will contain the underlying R code.);
environment variables that need to be adjusted (like where to store the data); and variables related to the analysis (such as start and
      end dates).
\begin{itemize}

  \item  First, the code contains variables that determine how the PDF will look, e.g., with (``echo=T'') or without the underlying R
      code (``echo=F''.)
<<setup_variables_knitr,echo=TRUE>>=

knitr::opts_knit$set(progress = TRUE, verbose = TRUE)
knitr::opts_knit$set(self.contained=T)

# cache.lazy = F (helps error when caching large data sets)
```

```
knitr::opts_chunk$set(echo=T,warning=F,message=F, error=T, tidy=T,
tidy.opts=list(width.cutoff=60), results='markup',cache.lazy = FALSE,
cache.rebuild=F)

@

    \item  Second, the code sets important environment variables (which need to be adjusted by the user):
<<setup_variables_user,echo=T>>=

setwd("~/repositories/xpin")
data_dir <- "/vscratch/grp-drosch/xpin/"


 if(! file.exists(data_dir)) {
   stop(sprintf("Could not find data_dir '%s' to store data,
                first create this directory.", data_dir))
 }

@

 \item  Third, we define variables specific to the analysis (which can be adjusted by the user). As in \cite{Bongaerts2025} the
      ``sample starts on February 1, 2001 (to prevent issues stemming
 from the tick size change on January 29, 2001) and runs until the end of 2014.''
<<setup_variables_analysis>>=

date_start <- "2001-02-01"
date_end   <- "2014-12-31"

@

    \item  Fourth, we setup the username and password to download all required data from WRDS.
We use the R-package ``keyring'' to safely store the username and password. To do either, you need to execute the code directly in the
      console to set the key ring (see code below, within tryCatch).
<<setup_wrds_password_2>>=
tryCatch({
  keyring_create("credentials_wrds", password = 'letmein')
  key_set("wrds_username", keyring ="credentials_wrds"
      ,prompt = "Your WRDS username:")
  key_set("wrds_password", keyring ="credentials_wrds"
      ,prompt = "Your WRDS password:")
},  error = function(e) {}
)

keyring_unlock("credentials_wrds", password = 'letmein')

@


\item  Last, we establish a connection to WRDS. If required data cannot be found, we can automatically download the data from WRDS.
<<setup_db>>=

postregs <- ifelse(exists("Postgres"), Postgres, PostgreSQL)

tryCatch({
wrds <- dbConnect(postregs(),
        host='wrds-pgdata.wharton.upenn.edu',
        port=9737,
        dbname='wrds',
        user=key_get("wrds_username",
                     keyring ="credentials_wrds"),
        password =  key_get("wrds_password",
                     keyring ="credentials_wrds"))
},
  error = function(e) {
    stop("It is likely that wrds_username and
    wrds_password are not stored in keyring.
    For that, run code in setup_wrds_password directly
    in R console. An error occurred: ", e$message)
```

```
  }
)

@

\end{itemize}
\label{sec:3r_end}


\subsection{Setup Data}
If we can find the data in the previously defined data directory, read it. Otherwise download it.
For that we first define a function to download the relevant stock data from WRDS, i.e., as in the reference paper we only download
      common-stocks ('shrcd' is 10 or 11) listed on the NYSE ('primexch' is 'N').
We use data from WRDS to determine the sample, adjust prices by corporate actions, and estimate the size (marketcap) of companies.
<<function_get_stock_returns>>=

get_stock_returns <- function(wrds, date_start, date_end) {
    sql_query <- sprintf(
    "SELECT a.cusip, a.permno, a.date, a.prc, a.vol,
           a.ret, a.openprc, a.askhi, a.bidlo,
           b.ticker, b.primexch, a.shrout, b.shrcd
      FROM crsp.dsf AS a, crsp.dsenames AS b
      WHERE a.permno = b.permno
      AND a.date BETWEEN b.namedt AND b.nameendt
      AND a.date >= '%s' AND a.date <= '%s'
      AND b.shrcd IN ('10', '11')
      AND b. IN ('N')",
      date_start, date_end
      )
  crsp_data <- dbGetQuery(wrds,sql_query) %>% as.data.table()
  return(crsp_data)
}

@

Then we call above function, in case we cannot find the relevant data file:
<<get_stock_returns>>=
data_file_name <- paste(data_dir, "data_wrds_crsp_complete.csv", sep = "/")

if(! file.exists(data_file_name)){
  data_sample <- get_stock_returns(wrds, date_start, date_end)
  fwrite(data_sample, file=data_file_name)
}

data_sample <- fread(data_file_name)
data_sample[ ,prc := abs(prc)]
data_sample[ ,date:=as.Date(date)]

@

In the replication study, we use WRDS Intraday Indicators, which provides stock-day variables estimated from Trade-And-Quote (TAQ) data.
In particular, we get an estimate of the stock-day price impact and order imbalance.
Given that in the reference paper \citep{Bongaerts2025} we used TRTH data,  the high-frequency data will be filtered differently
      compared to the reference paper.
But both differences the difference in data source and the difference in how data is filtered, will likely have a minor impact on the
      results, see., e.g., Footnote 7 of \cite{Roesch2017}.
More impactful is the way price impact is estimated, as we will elaborate in more details when discussing lambda and price impact.

WRDS Intraday Indicators (based on MTAQ, relevant for our sample starting in 2001) are not available for direct download, as when using
      DTAQ, see below.
We need to download the relevant data from the WRDS Website.
The code below indicates the relevant variables and how these map to WRDS Intraday Indicators based on DTAQ.
 <<read_data_wrds_intraday_mtaq>>=

 data_file_name <-  paste(data_dir, "data_wrds_intraday_mtaq.csv", sep="/")

 if(! file.exists(data_file_name)) {
```

```
   .select <- "TSignSqrtDVol2, QSpreadPct_TW_m, BuyDollar_LR0,
               SellDollar_LR0, BidLo, NumTrades_m, Mid_4pm"

    stop("WRDS does not offer intraday indicators calculated
          from MTAQ via Postgres. You need to go to their website
         and download the data.")
 }

 data_wrds_mintraday <-   fread(data_file_name)
 data_wrds_mintraday[, date:=as.Date(date)]

 setnames(data_wrds_mintraday, "symbol", "ticker")

 # map from MTAQ to DTAQ WRDS intraday indicators:
 setnames(data_wrds_mintraday, "TSignSqrtDVol2", "tsignsqrtdvol2")
 setnames(data_wrds_mintraday, "QSpreadPct_TW_m", "quotedspread_percent_tw")
 setnames(data_wrds_mintraday, "BuyDollar_LR0", "buy_dv_lr")
 setnames(data_wrds_mintraday, "SellDollar_LR0", "sell_dv_lr")
 setnames(data_wrds_mintraday, "BidLo", "price_low_m")
 setnames(data_wrds_mintraday, "NumTrades_m", "total_n_trades_m")
 setnames(data_wrds_mintraday, "Mid_4pm", "mid_4pm")

 @

We can get more recent Intraday Indicators (based on DTAQ) directly.
The University at Buffalo only has access to DTAQ data from 2010 onwards.
Therefore we hardcode the ``.year\_start'' to 2010.
<<read_data_wrds_intraday_dtaq>>=

 in_stock_ticker_list <- paste0("('", paste(unique(data_sample[, "ticker"]), collapse = "', '"), "')")

 data_file_name <-  paste(data_dir, "data_wrds_intraday_taq.csv", sep="/")

 if(! file.exists(data_file_name)) {

    .select <- "date, sym_root, sym_suffix, TSignSqrtDVol2, quotedspread_percent_tw , buy_dv_LR, sell_dv_LR, price_low_m,
      total_n_trades_m, mid_4pm"
    .where <-  sprintf("(sym_root IN %s)", in_stock_ticker_list)
    .where <-  "(1 = 1)"

    # .year_start <- year(date_start)
    # TODO lack of permission to earlier DTAQ and cannot find data for MTAQ
    .year_start <- 2010
    .year_end  <- year(date_end)

    query_parts <- c()
    for (year in .year_start:.year_end) {
      part <- sprintf("SELECT %s FROM taqmsec.wrds_iid_%d WHERE %s", .select, year, .where)
      query_parts <- c(query_parts, part)
    }

    sql_query <- paste(query_parts, collapse = " UNION ALL\n")

  data_wrds_intraday <- dbGetQuery(wrds,sql_query) %>% as.data.table()

  fwrite(data_wrds_intraday, file=data_file_name)

 }

 data_wrds_intraday <-   fread(data_file_name)
 data_wrds_intraday[, date:=as.Date(date)]

 setnames(data_wrds_intraday, "sym_root", "ticker")

 @

We now merge CRSP with relevant Intraday Indicators based on MTAQ.
CRSP allows us to identify stocks by their unique PERMNO and to adjust prices by corporate actions.
```

```
All the results in this document are based on MTAQ using the same sample period as in the reference paper.
<<data_merge>>=


 # data_complete <- merge(data_wrds_intraday, data_sample, by = c("date", "ticker"))
 data_complete <- merge(data_wrds_mintraday, data_sample, by = c("date", "ticker"))

@


As described in the paper, we use five other data sources:
\begin{itemize}
\item stock-days on which Form-4 trades occured
\item stock-days on which 13D trades occured
\item estimates of Good and Bad PIN
\item M\&A transactions
\item loss of analyst coverage
\end{itemize}
Unfortunately, these data are not commonly available and need to be computed.
For details we refer to the reference paper.

\subsection{Variable construction according to the reference paper}

As in the reference paper we
\begin{itemize}
\item ``discard stocks that trade for less than \$5 at any
time during our sample period'' (p. 23) and ``stock-days with fewer than 50 trades`` (p. 24).
We drop four days from the analysis on which the cross-sectional average PQSPR differs significantly between the reference paper and
      the replication study.
Difference in PQSPR are because in \cite{Bongaerts2025} we average spreads only associated with trades, in WRDS Intraday Indicators
      PQSPR is the time-weighted average spread.
<<variables_filter>>=
data_complete[,sample_price_low := min(price_low_m), by = .(permno)]

data_complete <- data_complete[sample_price_low > 5]
data_complete <- data_complete[total_n_trades_m > 50]

dates_bad <- c("2012-11-23", "2011-11-25", "2014-11-28", "2013-11-29")
data_complete <- data_complete[! date %in% dates_bad]

@

\item we estimate stock returns ``from corporate action adjusted end-of-day mid-quotes ($Return$; in \%; cross-sectionally winsorized
      1%%1# each day at the 0.1\% and 99.9\% level)''
<<variables_main_lhs>>=

data_complete[order(date), lag_mid_4pm := shift(mid_4pm, 1, fill = NA), by = .(permno)]
data_complete[order(date), lag_prc := shift(prc, 1, fill = NA), by = .(permno)]

data_complete[, corpact_adj := round((1+ret)/(prc/lag_prc), digits=6)]

data_complete[, ret := corpact_adj * mid_4pm/lag_mid_4pm - 1]
data_complete[, ret := pmin(ret, quantile(ret, 0.999, na.rm = TRUE)), by = .(date)]
data_complete[, ret := pmax(ret, quantile(ret, 0.001, na.rm = TRUE)), by = .(date)]

@

\item we estimate returns and quoted spread in percent and buyer-initiated, seller-initiated, and total trading volumes in USD millions.
<<variables_scaling>>=

data_complete[, ret := 100 * ret ]
data_complete[, quotedspread_percent_tw := quotedspread_percent_tw    * 100]

data_complete[, vol := vol/1000000]
data_complete[, buy_dv_lr  := buy_dv_lr/1000000]
data_complete[, sell_dv_lr := sell_dv_lr/1000000]

data_complete[, marketcap := prc*shrout/1000000]
```

```
@

\item we estimate order imbalance (oib) as the ``dollar volume of buyer- minus seller-initiated trades'' (p. 23).
To estimate the quoted spread (PQSPR), in the reference paper we ``averag[e] the spread associated with all trades'' in the replicating
      code we use the time-weighted average across all spreads from WRDS, as mentioned before.
<<variables_map>>=
data_complete[,  oib    := buy_dv_lr - sell_dv_lr ]
data_complete[,  PQSPR  := quotedspread_percent_tw]
@

\item we estimate volatility as the squared-return, trading volume in USD, the day return as the proportional price change from open to
      close, and `` market capitalization ... at the beginning of each calendar year'' (p. 26):
<<variables_other>>=

data_complete[,  RETSQ  := ret^2]

data_complete[,  vol_usd   := vol * openprc]

data_complete[,year  := year(date)]
data_complete[,month := month(date)]

data_complete[, return_day   := 100 * (prc/openprc -1)]

data_complete[order(date), marketcap := .SD[1, marketcap], by = .(permno, year)]

@
\end{itemize}


In the reference paper we estimate price impact following our theoretical mode, in particular, ``[i]n contrast to Goyenko et al. (2009,
      their Eq. (5)), we do not take the square-root of the dollar trading volume in this regression, such that our  price impact
      measure has a straightforward interpretation: the percentage price change  per unit of dollar trading volume.'' (p. 24).
WRDS Intraday Indicators provide price impact estimates only when using the square-root of dollar trading volume, see Formula 31 of
      \cite{WRDS2020} (MTAQ) and \cite{WRDS2025} (DTAQ).
To approximate the former from the later, we divide the later by the daily square-root of dollar trading volume.
<<variables_lambda_1>>=

data_complete[, tsignsqrtdvol2 := tsignsqrtdvol2 * 1000000  ]
data_complete[,  lambda := tsignsqrtdvol2 / sqrt(vol_usd)]

@

We then estimate price impact ``as the moving average of the estimated daily price impact parameters for that stock over the past 20
      trading days, where we set negative price impact estimates to zero. To further reduce the influence of outliers, we
      cross-sectionally winsorize the resulting expected price impact estimates each day at the 95\% level.'' (p. 24)
<<variables_lambda_2>>=
data_complete[         , pi_z20 := lambda]
data_complete[pi_z20< 0, pi_z20 := 0]

data_complete[order(date), pi_z20 := frollmean(pi_z20, n = 20, align = "right"), by = .(permno)]

data_complete[, pi_z20_w := pmin(pi_z20, quantile(pi_z20, 0.95, na.rm = TRUE)), by = .(date)]
data_complete[, lambda_w := pmin(lambda, quantile(lambda, 0.95, na.rm = TRUE)), by = .(date)]

@

Our main variable of interest are the signed and unsigned (absolute) product of price impact and order imbalance:
<<variables_main_rhs>>=

data_complete[, lambda_oib := pi_z20_w * oib]

data_complete[, abs_lambda_oib := abs(lambda_oib)]
@



<<variables_sample>>=
```

```
no_sample_stocks <- length( row.names(unique(data_complete[, "permno"])))
no_sample_days   <- length( row.names(unique(data_complete[, "date"])))
no_sample_trades <- sum(data_complete[,"total_n_trades_m"])

@
In the reference paper the ``final sample consists of all 1,388 NYSE stocks that survive these data screens over 2001-2014 and is based
      on a total of 18,626,168,999 signed trades.''
In the replication we have \Sexpr{pretty_numbers(format_integer, no_sample_stocks)} stocks and a total of
      \Sexpr{pretty_numbers(format_integer, no_sample_trades)} trades.
\subsection{Summary statistics}

We report cross-sectional summary statistics in Table 1.
<<table_summary_stats>>=
data_summary <- data_complete[, .(
                          days       = .N,
                          ret        = mean(ret, na.rm=T),
                          PQSPR      = mean(PQSPR, na.rm=T),
                          lambda_w   = mean(lambda_w, na.rm=T),
                          oib        = mean(oib, na.rm=T),
                          lambda_oib = mean(lambda_oib, na.rm=T)
                        ), by = .(permno)]

data_means <- colMeans(data_summary, na.rm=T)
obs_fct <- function(x) sum(! is.na(x))
data_obs     <- colwise(obs_fct)(data_summary )
data_sd      <- sqrt(colwise(var)(data_summary, na.rm=T))
data_min     <- colwise(min)(data_summary, na.rm=T)
data_median  <- colwise(median)(data_summary, na.rm=T)
data_max     <- colwise(max)(data_summary, na.rm=T)

probs <- c(0.25, 0.75)
quantiles <- colwise(quantile)(data_summary, probs = probs, na.rm=T)

@


\begin{table}[H]\renewcommand{\arraystretch}{1.5}\addtolength{\tabcolsep}{+2pt}
  \centering
 \caption[Summary statistics]{\textbf{Cross-sectional summary statistics of time-series averages }\\
\newline
This table reports the cross-sectional (across the \Sexpr{pretty_numbers(format_integer, no_sample_stocks)} NYSE stocks in the sample)
      mean, standard  deviation, first quartile, median, and third quartile of the time-series average by stock of
the daily percentage return from corporate action adjusted end-of-day mid-quotes ($Return$; in \%; cross-sectionally winsorized each
      1%%1# day at the 0.1\% and 99.9\% level),
the daily percentage average proportional quoted spread ($PQSPR$; in \%),
the price impact defined as the percentage price change associated with a million US dollars of trading volume ($\lambda$; expressed as
      \%Return/US\$m.; cross-sectionally winsorized each day at the 95\% level),
the daily order imbalance or difference between the total US dollar volume of trades initiated by buyers and sellers expressed in
      millions of dollars ($OIB$; in US\$m.),
and the daily private information measure $\lambda  \times  OIB$.
 The first column indicates the number of stocks over which the summary statistics are computed.
 The second column indicates the average number of days a stock is in the sample.
The sample includes \Sexpr{pretty_numbers(format_integer, no_sample_stocks)} NYSE stocks during our sample period 2001-2014.
 The factor to adjust daily closing mid-quote data for corporate actions is from CRSP.
    }
        \begin{tabularx}{1.0\textwidth}{ p{4.5cm}  >{\centering\arraybackslash}X >{\centering\arraybackslash}X
      >{\centering\arraybackslash}X >{\centering\arraybackslash}X >{\centering\arraybackslash}X >{\centering\arraybackslash}X
      >{\centering\arraybackslash}X >{\centering\arraybackslash}X}
    \toprule
      \
     &
     \#stocks &
     days &
     mean   &
     stddev &
     25\%   &
     median &
     75\%
```

```
      \\
    \midrule
    \emph{Return} &
       \Sexpr{pretty_numbers(format_integer, no_sample_stocks)} &
       \Sexpr{pretty_numbers(format_integer, data_means[["days"]])} &
       \Sexpr{pretty_numbers(format_numeric, data_means[["ret"]])} &
       \Sexpr{pretty_numbers(format_numeric, data_sd[["ret"]])} &
       \Sexpr{pretty_numbers(format_numeric, quantiles[["ret"]][["25%"]])} &
       \Sexpr{pretty_numbers(format_numeric, data_median[["ret"]])} &
       \Sexpr{pretty_numbers(format_numeric, quantiles[["ret"]][["75%"]])}
      \\
    \emph{PQSPR}  &
       \Sexpr{pretty_numbers(format_integer, no_sample_stocks)} &
       \Sexpr{pretty_numbers(format_integer, data_means[["days"]])} &
       \Sexpr{pretty_numbers(format_numeric, data_means[["PQSPR"]])} &
       \Sexpr{pretty_numbers(format_numeric, data_sd[["PQSPR"]])} &
       \Sexpr{pretty_numbers(format_numeric, quantiles[["PQSPR"]][["25%"]])} &
       \Sexpr{pretty_numbers(format_numeric, data_median[["PQSPR"]])} &
       \Sexpr{pretty_numbers(format_numeric, quantiles[["PQSPR"]][["75%"]])}
      \\
    $\lambda$ &
       \Sexpr{pretty_numbers(format_integer, no_sample_stocks)} &
       \Sexpr{pretty_numbers(format_integer, data_means[["days"]])} &
       \Sexpr{pretty_numbers(format_numeric, data_means[["lambda_w"]])} &
       \Sexpr{pretty_numbers(format_numeric, data_sd[["lambda_w"]])} &
       \Sexpr{pretty_numbers(format_numeric, quantiles[["lambda_w"]][["25%"]])} &
       \Sexpr{pretty_numbers(format_numeric, data_median[["lambda_w"]])} &
       \Sexpr{pretty_numbers(format_numeric, quantiles[["lambda_w"]][["75%"]])}
      \\
    \emph{OIB} &
       \Sexpr{pretty_numbers(format_integer, no_sample_stocks)} &
       \Sexpr{pretty_numbers(format_integer, data_means[["days"]])} &
       \Sexpr{pretty_numbers(format_numeric, data_means[["oib"]])} &
       \Sexpr{pretty_numbers(format_numeric, data_sd[["oib"]])} &
       \Sexpr{pretty_numbers(format_numeric, quantiles[["oib"]][["25%"]])} &
       \Sexpr{pretty_numbers(format_numeric, data_median[["oib"]])} &
       \Sexpr{pretty_numbers(format_numeric, quantiles[["oib"]][["75%"]])}
      \\
     $\lambda \times  OIB$ &
       \Sexpr{pretty_numbers(format_integer, no_sample_stocks)} &
       \Sexpr{pretty_numbers(format_integer, data_means[["days"]])} &
       \Sexpr{pretty_numbers(format_numeric, data_means[["lambda_oib"]])} &
       \Sexpr{pretty_numbers(format_numeric, data_sd[["lambda_oib"]])} &
       \Sexpr{pretty_numbers(format_numeric, quantiles[["lambda_oib"]][["25%"]])} &
       \Sexpr{pretty_numbers(format_numeric, data_median[["lambda_oib"]])} &
       \Sexpr{pretty_numbers(format_numeric, quantiles[["lambda_oib"]][["75%"]])}
      \\
    \bottomrule
    \end{tabularx}%
  \label{tab:Summary_statistics}%
\end{table}%

\section{Empirical results}
We now replicate results of \cite{Bongaerts2025} using data from 2001 to 2014 from WRDS Intraday Indicators.
Table 2 reports pooled, pairwise correlations.
<<table2_correlations>>=

       table2_corr <- cor( data_complete[, .(lambda_oib, abs_lambda_oib, pi_z20_w, oib, PQSPR, vol_usd, ret)], use =
      "pairwise.complete.obs")

       # table2_corr_t <- cor.test( data_complete[, .(lambda_oib, abs_lambda_oib, pi_z20_w, oib, PQSPR, vol_usd, ret)], use =
      1%%1# "pairwise.complete.obs")
@
\begin{table}[H]\renewcommand{\arraystretch}{1.5}\addtolength{\tabcolsep}{+2pt}
  \centering
  \caption{\textbf{Pooled correlations of daily private information, liquidity, order imbalance, and returns } \\
\newline
```

```
This table reports pooled Pearson correlation coefficients between nine daily stock-specific variables: the signed and the absolute
      private information measure ($\lambda\times OIB$ and $\lambda\times |OIB|$), the price impact ($\lambda$), the dollar order
      imbalance ($OIB$), the proportional quoted spread ($PQSPR$), the dollar trading volume ($Volume$), the stock and returns
      ($Return$).
We refer to Table~\ref{tab:Summary_statistics} for a description of these variables.
The sample includes \Sexpr{pretty_numbers(format_integer, no_sample_stocks)} NYSE stocks during our sample period 2001-2014.
The table reports $p$-values in parentheses below the correlations.
    }\scalebox{0.8}{
    \begin{tabularx}{1.0\textwidth}{ p{2.0cm} >{\centering\arraybackslash}X@{}  >{\centering\arraybackslash}X@{}
      >{\centering\arraybackslash}X@{} >{\centering\arraybackslash}X@{} >{\centering\arraybackslash}X@{}
      >{\centering\arraybackslash}X@{} >{\centering\arraybackslash}X@{} >{\centering\arraybackslash}X@{}
    >{\centering\arraybackslash}X@{} >{\centering\arraybackslash}X@{}
    >{\centering\arraybackslash}X@{} }
    \toprule
    &
      $\lambda\times\emph{OIB}$ &
      $\lambda\times |OIB|$ &
      $\lambda$ &
      $OIB$ &
      $PQSPR$ &
      $Volume$ &
      $Return$ &
      $G.\ PIN$ &
      $B.\ PIN$ \\
    \hline
$\lambda\times |OIB|$ & \Sexpr{pretty_numbers(format_numeric, table2_corr["lambda_oib", "abs_lambda_oib"]) } & 1.000 &  &  &  \\[-1.5ex]
 & (0.00) &  &  &  &  &  \\
$\lambda$ & \Sexpr{pretty_numbers(format_numeric, table2_corr["pi_z20_w", "abs_lambda_oib"]) }  & \Sexpr{pretty_numbers(format_numeric,
      table2_corr["pi_z20_w", "lambda_oib"]) }  & 1.000 &  &  \\[-1.5ex]
& (0.00) & (0.00) &  &  &  &  \\
$OIB$  & \Sexpr{pretty_numbers(format_numeric, table2_corr["oib", "abs_lambda_oib"]) } & \Sexpr{pretty_numbers(format_numeric,
      table2_corr["oib", "lambda_oib"]) } & \Sexpr{pretty_numbers(format_numeric, table2_corr["oib", "pi_z20_w"]) } & 1.000 &
      \\[-1.5ex]
 & (0.00) & (0.00) & (0.00) &  &  &  \\
$PQSPR$ & \Sexpr{pretty_numbers(format_numeric, table2_corr["PQSPR", "abs_lambda_oib"]) } & \Sexpr{pretty_numbers(format_numeric,
      table2_corr["PQSPR", "lambda_oib"]) } & \Sexpr{pretty_numbers(format_numeric, table2_corr["PQSPR", "pi_z20_w"]) } &
      \Sexpr{pretty_numbers(format_numeric, table2_corr["PQSPR", "oib"]) } & 1.000 \\[-1.5ex]
 & (0.00) & (0.00) & (0.00) & (0.00) &  &  \\
$Volume$  & \Sexpr{pretty_numbers(format_numeric, table2_corr["vol_usd", "abs_lambda_oib"]) } & \Sexpr{pretty_numbers(format_numeric,
      table2_corr["vol_usd", "lambda_oib"]) } & \Sexpr{pretty_numbers(format_numeric, table2_corr["vol_usd", "pi_z20_w"]) } &
      \Sexpr{pretty_numbers(format_numeric, table2_corr["vol_usd", "oib"]) } & \Sexpr{pretty_numbers(format_numeric,
      table2_corr["vol_usd", "PQSPR"]) }  & 1.000 \\[-1.5ex]
 & (0.00) & (0.00) & (0.00) & (0.00) & (0.00) &  \\
$Return$  & \Sexpr{pretty_numbers(format_numeric, table2_corr["ret", "abs_lambda_oib"]) } & \Sexpr{pretty_numbers(format_numeric,
      table2_corr["ret", "lambda_oib"]) } & \Sexpr{pretty_numbers(format_numeric, table2_corr["ret", "pi_z20_w"]) } &
      \Sexpr{pretty_numbers(format_numeric, table2_corr["ret", "oib"]) } & \Sexpr{pretty_numbers(format_numeric, table2_corr["ret",
      "PQSPR"]) } & 0\Sexpr{pretty_numbers(format_numeric, table2_corr["ret", "vol_usd"]) } & 1.000 \\[-1.5ex]
& (0.00) & (0.00) & (0.00) & (0.00) & (0.00) & (0.00)  \\
    \bottomrule
    \end{tabularx}}%
  \label{tab:correlations}%
\end{table}%


As in the reference paper, Table 3 shows summary statistics of portfolios sorted by private information. ``Portfolios are formed every
      month and all variables are estimated using data from the previous month.'' \citep[caption of Table 3][]{Bongaerts2025}
<<robustness0_table3>>=

summary <- data_complete[, .( abs_lambda_oib = mean(abs_lambda_oib, na.rm=T), marketcap = mean(marketcap, na.rm=T), RETSQ = mean(RETSQ,
      na.rm=T),PQSPR = mean(PQSPR, na.rm=T), pi_z20_w = mean(pi_z20_w, na.rm=T) ), by= .(permno, year, month)]

summary <- summary[!is.na(pi_z20_w)]

summary[, time := as.yearmon(paste(year, month, sep = "-"), "%Y-%m")]
setkey(summary, permno, time)

summary[order(time), f_abs_lambda_oib := shift(abs_lambda_oib, -1, fill = NA), by = .(permno)]
```

```
# to address missing lagged values, drop these:
summary[, l_time := shift(time, 1), by = .(permno)]
summary <- summary[l_time + 1/12 == time,]


summary[, quantile_v := dplyr::ntile(f_abs_lambda_oib, n=5), by = .(year, month)]

test_abs_lambda_oib <- wilcox.test(abs_lambda_oib ~ quantile_v, data = summary[quantile_v == 1 | quantile_v == 5])
test_marketcap      <- wilcox.test(marketcap ~ quantile_v, data = summary[quantile_v == 1 | quantile_v == 5])
test_RETSQ          <- wilcox.test(RETSQ ~ quantile_v, data = summary[quantile_v == 1 | quantile_v == 5])
test_PQSPR          <- wilcox.test(PQSPR ~ quantile_v, data = summary[quantile_v == 1 | quantile_v == 5])
test_pi_z20_w       <- wilcox.test(pi_z20_w ~ quantile_v, data = summary[quantile_v == 1 | quantile_v == 5])

summary <- summary[, .( stocks = uniqueN(permno), abs_lambda_oib = median(abs_lambda_oib), marketcap = median(marketcap, na.rm=T),
       RETSQ = median(RETSQ), PQSPR = median(PQSPR), pi_z20_w = median(pi_z20_w) ), by=quantile_v]


diff_summary <- summary[quantile_v == 5] - summary[quantile_v == 1]

@
\begin{table}[H]\renewcommand{\arraystretch}{1.5}\addtolength{\tabcolsep}{+2pt}
  \centering
 \caption[Summary statistics]{\textbf{Characteristics of stocks with large vs. small private information shocks}\\
 \newline
This table reports the following median characteristics of quintile portfolios sorted on the absolute private information measure
       ($\lambda\times |OIB|$) each month:
 the market capitalization in billions of dollars ($Mktcap$),
 the squared stock return ($Volatility$),
 the proportional quoted spread ($PQSPR$),
the price impact defined as the percentage price change associated with a million dollars of trading volume ($\lambda$),
 the number of analysts covering the firm  ($Analyst$ $coverage$),
and the dispersion in analysts earnings forecasts, defined as the standard deviation of annual analysts earnings per share forecasts
       scaled by the mean forecast ($Analyst$ $dispersion$).
 Portfolios are formed every month and all variables are estimated using data from the previous month.
 The sample includes \Sexpr{pretty_numbers(format_integer, no_sample_stocks)} NYSE stocks during our sample period 2001-2014.
 *, **, and *** indicate significance at the 10\%, 5\%, and 1\% levels based on a **Wilcoxon rank sum** test, respectively.
    }
        \scalebox{0.8}{\begin{tabularx}{1.0\textwidth}{ p{3.5cm}  >{\centering\arraybackslash}X >{\centering\arraybackslash}X
       >{\centering\arraybackslash}X >{\centering\arraybackslash}X >{\centering\arraybackslash}X  >{\centering\arraybackslash}X}
    \toprule
    $\lambda\times |OIB|$ Quintiles     & $Mktcap$    & $Volatility$  & $PQSPR$  & $\lambda$ &    \\
  \hline
  1                           & \Sexpr{pretty_numbers(format_numeric, summary[quantile_v  == 1,marketcap])} &
       \Sexpr{pretty_numbers(format_numeric, summary[quantile_v  == 1,RETSQ])} & \Sexpr{pretty_numbers(format_numeric,
       summary[quantile_v  == 1,PQSPR])} & \Sexpr{pretty_numbers(format_numeric, summary[quantile_v  == 1,pi_z20_w])} & \\
  2                           & \Sexpr{pretty_numbers(format_numeric, summary[quantile_v  == 2,marketcap])} &
       \Sexpr{pretty_numbers(format_numeric, summary[quantile_v  == 2,RETSQ])} & \Sexpr{pretty_numbers(format_numeric,
       summary[quantile_v  == 2,PQSPR])} & \Sexpr{pretty_numbers(format_numeric, summary[quantile_v  == 2,pi_z20_w])} &  \\
  3                           & \Sexpr{pretty_numbers(format_numeric, summary[quantile_v  == 3,marketcap])} &
       \Sexpr{pretty_numbers(format_numeric, summary[quantile_v  == 3,RETSQ])} & \Sexpr{pretty_numbers(format_numeric,
       summary[quantile_v  == 3,PQSPR])} & \Sexpr{pretty_numbers(format_numeric, summary[quantile_v  == 3,pi_z20_w])} &  \\
  4                           & \Sexpr{pretty_numbers(format_numeric, summary[quantile_v  == 4,marketcap])} &
       \Sexpr{pretty_numbers(format_numeric, summary[quantile_v  == 4,RETSQ])} & \Sexpr{pretty_numbers(format_numeric,
       summary[quantile_v  == 4,PQSPR])} & \Sexpr{pretty_numbers(format_numeric, summary[quantile_v  == 4,pi_z20_w])} &  \\
  5                           & \Sexpr{pretty_numbers(format_numeric, summary[quantile_v  == 5,marketcap])} &
       \Sexpr{pretty_numbers(format_numeric, summary[quantile_v  == 5,RETSQ])} & \Sexpr{pretty_numbers(format_numeric,
       summary[quantile_v  == 5,PQSPR])} & \Sexpr{pretty_numbers(format_numeric, summary[quantile_v  == 5,pi_z20_w])} &  \\
  \hline
  \textbf{Difference {[}5{]} - {[}1{]}} & \Sexpr{pretty_numbers(format_numeric, diff_summary[["marketcap"]], p_value =
       test_abs_lambda_oib[["p.value"]])} &
                                        \Sexpr{pretty_numbers(format_numeric, diff_summary[["RETSQ"]],     p_value =
       test_RETSQ[["p.value"]])} &
                                        \Sexpr{pretty_numbers(format_numeric, diff_summary[["PQSPR"]],     p_value =
       test_PQSPR[["p.value"]])} &
                                        \Sexpr{pretty_numbers(format_numeric, diff_summary[["pi_z20_w"]],  p_value =
       test_pi_z20_w[["p.value"]])} &
                                         &    \\
  \bottomrule
```

```
  \end{tabularx}}%
\label{tab:summary_by_symbol}%
\end{table}%


To estimate Fama-MacBeth regression with Newey-West standard errors, we define the following function:
<<setup_fmb_analysis>>=

       lags <- 5
       nwt_stats <- function(x) t(coeftest(lm(x ~ 1), vcov = NeweyWest(lm(x ~ 1), lag = lags, prewhite = FALSE))[,c("t value",
      "Pr(>|t|)")])

       fmb_reg <- function(data, formula) {

          vars <- all.vars(formula)
          data_no_na <- data[complete.cases(data[, ..vars])]

          fmb_results <- data_no_na[, {
            reg <- lm(formula, data = .SD)
            c(reg = as.list(coef(reg)), r_squared = summary(reg)$r.squared)
          }, by = .(date)]

          fmb_results[, date := NULL]

          fmb_coef   <- colwise(mean, na.rm=T)(fmb_results)
          fmb_t_stat <- colwise(nwt_stats)(fmb_results)

          days <- length(unique(data_no_na[, date]))

          return(list(coef=fmb_coef, t_stats = fmb_t_stat, days = days, r_squared = fmb_coef[["r_squared"]]))
      }
@

As in the reference paper, in regressions returns are not scaled by 100, i.e., they are not in per cent:
<<variables_rescale_for_regressions>>=

data_complete[, ret := ret/100 ]
data_complete[, RETSQ  := ret^2]

data_complete[, lambda_oib := lambda_oib/1000 ]
data_complete[, abs_lambda_oib := abs_lambda_oib/1000 ]

@

We now replicate three tables from the reference paper showing results of daily Fama-MacBeth regressions explaining returns by
      contemporaneous private information  \citep[Table 5 of][]{Bongaerts2025}; explaining returns by previous private information
      (Table 6); and explaining volatility by previous private information (Table 8).
<<table5>>=

      data_complete[order(date), lag_Return := shift(ret, 1, fill = NA), by = .(permno)]

      data_complete[, marketcap_inv  := 1/marketcap]

      formula <- "ret  ~ lag_Return + lambda_oib"
      fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
      reg1     <-  fmb_stock_reg$coef
      t_stats1 <- fmb_stock_reg$t_stats
      days1    <- fmb_stock_reg$days
      reg1[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

      formula <- "ret  ~ lag_Return + lambda_oib + oib + pi_z20_w"
      fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
      reg2     <-  fmb_stock_reg$coef
      t_stats2 <- fmb_stock_reg$t_stats
      days2    <- fmb_stock_reg$days
      reg2[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

      formula <- "ret  ~ lag_Return + lambda_oib + oib + pi_z20_w + oib* marketcap_inv"
```

```
      fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
      reg3     <-  fmb_stock_reg$coef
      t_stats3 <- fmb_stock_reg$t_stats
      days3    <- fmb_stock_reg$days
      reg3[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

      formula <- "ret  ~ lag_Return + lambda_oib + oib + pi_z20_w + oib* PQSPR"
      fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
      reg4     <-  fmb_stock_reg$coef
      t_stats4 <- fmb_stock_reg$t_stats
      days4    <- fmb_stock_reg$days
      reg4[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared
@
\begin{table}[H]\renewcommand{\arraystretch}{1.5}\addtolength{\tabcolsep}{+2pt}
\centering
  \caption{\textbf{Daily Fama-MacBeth regressions of returns on contemporaneous private information}\\
\newline
This table reports the time-series averages of the estimated slope coefficients from daily Fama-MacBeth regressions to explain
      cross-sectional variation in daily stock returns.
The dependent variable is the end-of-day mid-quote return of stock $i$ on day $d$ ($Return_{i,d}$).
The independent variables are:
the return of stock $i$ on day $d-1$ ($Return_{i,d-1}$),
the order imbalance of stock $i$ on day $d$ ($OIB_{i,d}$),
the price impact of stock $i$ on day $d$ -- calculated as the stocks average price impact estimate over the past 20 days with setting
      non-positive price impact estimates to zero ($\lambda_{i,d}$),
the inverse of the market capitalization of stock $i$ at the beginning of each year ($1/Mktcap_{i,y-}$),
the proportional quoted spread of stock $i$ on day $d-1$ ($PQSPR_{i,d-1}$), as well as various interaction terms.
The table reports Fama-MacBeth $t$-statistics with Newey-West corrections in parentheses below the average coefficients.
Some coefficients have been scaled for ease of presentation.
Intercepts are suppressed to conserve space. The final two rows report the $R^2$ and the number of regressions. The sample includes
      \Sexpr{pretty_numbers(format_integer, no_sample_stocks)} NYSE stocks during our sample period 2001-2014.
*, **, and *** indicate significance at the 10\%, 5\%, and 1\% levels, respectively.
}
    \begin{tabularx}{1.0\textwidth}{ p{4cm}  >{\centering\arraybackslash}X@{} >{\centering\arraybackslash}X@{}
      >{\centering\arraybackslash}X@{} >{\centering\arraybackslash}X@{} >{\centering\arraybackslash}X@{} }
    \toprule
        & \multicolumn{5}{l}{ Dependent variable: $Return_{i,d}$}\\
    & (1) & (2) & (3) & (4) &  \\
      \hline
    $Return_{i,d-1}$  &
       \Sexpr{pretty_numbers(format_numeric, reg1[["reg.lag_Return"]], p_value = t_stats1[["reg.lag_Return"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg2[["reg.lag_Return"]], p_value = t_stats2[["reg.lag_Return"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg3[["reg.lag_Return"]], p_value = t_stats3[["reg.lag_Return"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg4[["reg.lag_Return"]], p_value = t_stats4[["reg.lag_Return"]][1,"Pr(>|t|)"])} &
      \\
     &
       \Sexpr{pretty_numbers(format_significance, t_stats1[["reg.lag_Return"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats2[["reg.lag_Return"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats3[["reg.lag_Return"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.lag_Return"]][1,"t value"])} &
      \\
    $\Lambda_{i,d}\times OIB_{i,d}$  &
       \Sexpr{pretty_numbers(format_numeric, reg1[["reg.lambda_oib"]], p_value = t_stats1[["reg.lambda_oib"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg2[["reg.lambda_oib"]], p_value = t_stats2[["reg.lambda_oib"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg3[["reg.lambda_oib"]], p_value = t_stats3[["reg.lambda_oib"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg4[["reg.lambda_oib"]], p_value = t_stats4[["reg.lambda_oib"]][1,"Pr(>|t|)"])} &
      \\
     &
       \Sexpr{pretty_numbers(format_significance, t_stats1[["reg.lambda_oib"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats2[["reg.lambda_oib"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats3[["reg.lambda_oib"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.lambda_oib"]][1,"t value"])} &
      \\
        $\Lambda_{i,d}$  &
        &
       \Sexpr{pretty_numbers(format_numeric, reg2[["reg.pi_z20_w"]], p_value = t_stats2[["reg.pi_z20_w"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg3[["reg.pi_z20_w"]], p_value = t_stats3[["reg.pi_z20_w"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg4[["reg.pi_z20_w"]], p_value = t_stats4[["reg.pi_z20_w"]][1,"Pr(>|t|)"])} &
```

```
    \\
   &
   &
     \Sexpr{pretty_numbers(format_significance, t_stats2[["reg.pi_z20_w"]][1,"t value"])} &
     \Sexpr{pretty_numbers(format_significance, t_stats3[["reg.pi_z20_w"]][1,"t value"])} &
     \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.pi_z20_w"]][1,"t value"])} &
    \\
      $OIB_{i,d}$  &
      &
     \Sexpr{pretty_numbers(format_numeric, reg2[["reg.oib"]], p_value = t_stats2[["reg.oib"]][1,"Pr(>|t|)"])} &
     \Sexpr{pretty_numbers(format_numeric, reg3[["reg.oib"]], p_value = t_stats3[["reg.oib"]][1,"Pr(>|t|)"])} &
     \Sexpr{pretty_numbers(format_numeric, reg4[["reg.oib"]], p_value = t_stats4[["reg.oib"]][1,"Pr(>|t|)"])} &
    \\
   &
   &
     \Sexpr{pretty_numbers(format_significance, t_stats2[["reg.oib"]][1,"t value"])} &
     \Sexpr{pretty_numbers(format_significance, t_stats3[["reg.oib"]][1,"t value"])} &
     \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.oib"]][1,"t value"])} &
    \\
    $1/Marketcap_{i,d}$  &
      &
    &
     \Sexpr{pretty_numbers(format_numeric, reg3[["reg.marketcap_inv"]], p_value = t_stats3[["reg.marketcap_inv"]][1,"Pr(>|t|)"])} &
    \\
   &
   &
    &
     \Sexpr{pretty_numbers(format_significance, t_stats3[["reg.marketcap_inv"]][1,"t value"])} &
    \\
    $OIB_{i,d} \times 1/Marketcap_{i,d}$  &
      &
    &
     \Sexpr{pretty_numbers(format_numeric, reg3[["reg.oib:marketcap_inv"]], p_value =
    t_stats3[["reg.oib:marketcap_inv"]][1,"Pr(>|t|)"])} &
    \\
   &
   &
    &
     \Sexpr{pretty_numbers(format_significance, t_stats3[["reg.oib:marketcap_inv"]][1,"t value"])} &
    \\
    $PQSPR_{i,d}$  &
      &
    &
    &
     \Sexpr{pretty_numbers(format_numeric, reg4[["reg.PQSPR"]], p_value = t_stats4[["reg.PQSPR"]][1,"Pr(>|t|)"])} &
    \\
   &
   &
    &
    &
     \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.PQSPR"]][1,"t value"])} &
    \\
    $OIB_{i,d} \times PQSPR_{i,d}$  &
      &
    &
    &
     \Sexpr{pretty_numbers(format_numeric, reg4[["reg.oib:PQSPR"]], p_value = t_stats4[["reg.oib:PQSPR"]][1,"Pr(>|t|)"])} &
    \\
   &
   &
    &
    &
     \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.oib:PQSPR"]][1,"t value"])} &
    \\
   $R^2$ [\%] &
     \Sexpr{pretty_numbers(format_percent, 100 * reg1[["r.squared"]][["rsq"]])} &
     \Sexpr{pretty_numbers(format_percent, 100 * reg2[["r.squared"]][["rsq"]])} &
     \Sexpr{pretty_numbers(format_percent, 100 * reg3[["r.squared"]][["rsq"]])} &
```

```
      \Sexpr{pretty_numbers(format_percent, 100 * reg4[["r.squared"]][["rsq"]])} &
    \\
    \#Regressions &
       \Sexpr{pretty_numbers(format_integer, days1)} &
       \Sexpr{pretty_numbers(format_integer, days2)} &
       \Sexpr{pretty_numbers(format_integer, days3)} &
       \Sexpr{pretty_numbers(format_integer, days4)} &
    \\
    \bottomrule
    \end{tabularx}%
  \label{tab:5}%
\end{table}%


<<robustness0_table6>>=
    data_complete[order(date), lag_Return       := shift(ret, 1, fill = NA), by = .(permno)]
    data_complete[order(date), l_abs_return_day := shift(abs(return_day), 1, fill = NA), by = .(permno)]
    data_complete[order(date), l_pi_abs_oib     := shift(abs_lambda_oib, 1), by = .(permno)]
    data_complete[order(date), l_piz            := shift(pi_z20_w, 1), by = .(permno)]
    data_complete[order(date), l_abs_oib        := shift(abs(oib), 1), by = .(permno)]
    data_complete[order(date), l_piz_marketcap  := shift(pi_z20_w * marketcap, 1), by = .(permno)]

      formula <- "ret  ~ lag_Return"
      fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
      reg1     <-  fmb_stock_reg$coef
      t_stats1 <- fmb_stock_reg$t_stats
      days1    <- fmb_stock_reg$days
      reg1[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

      formula <- "ret ~ lag_Return * l_pi_abs_oib"
      fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
      reg2     <-  fmb_stock_reg$coef
      t_stats2 <- fmb_stock_reg$t_stats
      days2    <- fmb_stock_reg$days
      reg2[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

      formula <- "ret  ~ lag_Return + lag_Return * l_piz_marketcap"
      fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
      reg3     <-  fmb_stock_reg$coef
      t_stats3 <- fmb_stock_reg$t_stats
      days3    <- fmb_stock_reg$days
      reg3[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

      formula <- "ret ~ lag_Return * l_piz + lag_Return * l_abs_oib"
      fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
      reg4     <-  fmb_stock_reg$coef
      t_stats4 <- fmb_stock_reg$t_stats
      days4    <- fmb_stock_reg$days
      reg4[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

      formula <- "ret ~ lag_Return * l_pi_abs_oib + lag_Return * l_piz + lag_Return * l_abs_oib"
      fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
      reg5     <-  fmb_stock_reg$coef
      t_stats5 <- fmb_stock_reg$t_stats
      days5    <- fmb_stock_reg$days
      reg5[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

      formula <- "ret ~ lag_Return * l_pi_abs_oib + lag_Return * l_piz + lag_Return * l_abs_oib + lag_Return * l_abs_return_day"
      fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
      reg6     <-  fmb_stock_reg$coef
      t_stats6 <- fmb_stock_reg$t_stats
      days6    <- fmb_stock_reg$days
      reg6[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared



@
```

```
\begin{table}[H]\renewcommand{\arraystretch}{1.2}\addtolength{\tabcolsep}{+2pt}
\centering
  \caption{\textbf{Daily Fama-MacBeth regressions of returns on previous day private information}\\
\newline
This table reports the time-series averages of the estimated slope coefficients from daily Fama-MacBeth regressions to explain
      cross-sectional variation in daily stock returns.
The dependent variable is the end-of-day mid-quote return of stock $i$ on day $d$ ($Return_{i,d}$).
The independent variables are:
the return of stock $i$ on day $d-1$ ($Return_{i,d-1}$),
the market capitalization of stock $i$ on day $d-1$,
the previous day absolute private information measure $\lambda\times\emph{OIB}$ computed as the product of the price impact of stock
      $i$ on day $d-1$ -- which is calculated as the stocks average price impact estimate over the past 20 days with setting
      non-positive price impact estimates to zero ($\lambda_{i,d-1}$) and the absolute order imbalance of stock $i$ on day $d-1$
      ($|OIB_{i,d-1}|$), $\lambda_{i,d-1}$ and $|OIB_{i,d-1}|$ separately, as well as various interaction terms.
In column 5 we also control for the absolute day-return, i.e., the return from open to close.
Some coefficients have been scaled for ease of presentation.
Intercepts are suppressed to conserve space. The final three rows report the $R^2$, the adjusted $R^2$, and the number of regressions.
      The sample includes \Sexpr{pretty_numbers(format_integer, no_sample_stocks)} NYSE stocks during our sample period 2001-2014.
*, **, and *** indicate significance at the 10\%, 5\%, and 1\% levels, respectively.
    }
    \begin{tabularx}{1.0\textwidth}{ p{5.0cm}  >{\centering\arraybackslash}X@{}  >{\centering\arraybackslash}X@{}
      >{\centering\arraybackslash}X@{} >{\centering\arraybackslash}X@{} >{\centering\arraybackslash}X@{}
      >{\centering\arraybackslash}X@{}
    >{\centering\arraybackslash}X@{} }
    \toprule
    & \multicolumn{6}{l}{ Dependent variable: $Return_{i,d}$} \\
        & (1) & (2) & (3) & (4) & (5) & (6) &   \\
      \hline
    $Return_{i,d-1}$  &
       \Sexpr{pretty_numbers(format_numeric, reg1[["reg.lag_Return"]], p_value = t_stats1[["reg.lag_Return"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg2[["reg.lag_Return"]], p_value = t_stats2[["reg.lag_Return"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg3[["reg.lag_Return"]], p_value = t_stats3[["reg.lag_Return"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg4[["reg.lag_Return"]], p_value = t_stats4[["reg.lag_Return"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg5[["reg.lag_Return"]], p_value = t_stats5[["reg.lag_Return"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg6[["reg.lag_Return"]], p_value = t_stats6[["reg.lag_Return"]][1,"Pr(>|t|)"])} &
      \\[-0.8ex]
     &
       \Sexpr{pretty_numbers(format_significance, t_stats1[["reg.lag_Return"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats2[["reg.lag_Return"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats3[["reg.lag_Return"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.lag_Return"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.lag_Return"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.lag_Return"]][1,"t value"])} &
      \\
    $\Lambda_{i,d-1} \times |OIB_{i,d-1}|$  &
       &
       \Sexpr{pretty_numbers(format_numeric, reg2[["reg.l_pi_abs_oib"]], p_value = t_stats2[["reg.l_pi_abs_oib"]][1,"Pr(>|t|)"])} &
        &
        &
       \Sexpr{pretty_numbers(format_numeric, reg5[["reg.l_pi_abs_oib"]], p_value = t_stats5[["reg.l_pi_abs_oib"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg6[["reg.l_pi_abs_oib"]], p_value = t_stats6[["reg.l_pi_abs_oib"]][1,"Pr(>|t|)"])} &
      \\[-0.8ex]
     &
      &
       \Sexpr{pretty_numbers(format_significance, t_stats2[["reg.l_pi_abs_oib"]][1,"t value"])} &
        &
        &
       \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.l_pi_abs_oib"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.l_pi_abs_oib"]][1,"t value"])} &
    \\
    $Return_{i,d-1} \times \Lambda_{i,d-1} \times |OIB_{i,d-1}|$  &
    &
       \Sexpr{pretty_numbers(format_numeric, reg2[["reg.lag_Return:l_pi_abs_oib"]], p_value =
      t_stats2[["reg.lag_Return:l_pi_abs_oib"]][1,"Pr(>|t|)"])} &
       &
       &
```

```
    \Sexpr{pretty_numbers(format_numeric, reg5[["reg.lag_Return:l_pi_abs_oib"]], p_value =
  t_stats5[["reg.lag_Return:l_pi_abs_oib"]][1,"Pr(>|t|)"])} &
    \Sexpr{pretty_numbers(format_numeric, reg6[["reg.lag_Return:l_pi_abs_oib"]], p_value =
  t_stats6[["reg.lag_Return:l_pi_abs_oib"]][1,"Pr(>|t|)"])} &
  \\[-0.8ex]
  &
 &
    \Sexpr{pretty_numbers(format_significance, t_stats2[["reg.lag_Return:l_pi_abs_oib"]][1,"t value"])} &
    &
    &
    \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.lag_Return:l_pi_abs_oib"]][1,"t value"])} &
    \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.lag_Return:l_pi_abs_oib"]][1,"t value"])} &
    \\

      $\Lambda_{i,d-1} \times Marketcap_{i,d-1}$  &
&
    &
   \Sexpr{pretty_numbers(format_numeric, reg3[["reg.l_piz_marketcap"]], p_value = t_stats3[["reg.l_piz_marketcap"]][1,"Pr(>|t|)"])}
  &
    &
  \\[-0.8ex]
  &
 &
    &
    \Sexpr{pretty_numbers(format_significance, t_stats3[["reg.l_piz_marketcap"]][1,"t value"])} &
    &
    \\
       $Return_{i,d-1} \times \Lambda_{i,d-1} \times Marketcap_{i,d-1}$  &
&
    &
    \Sexpr{pretty_numbers(format_numeric, reg3[["reg.lag_Return:l_piz_marketcap"]], p_value =
  t_stats3[["reg.lag_Return:l_piz_marketcap"]][1,"Pr(>|t|)"])} &
    &
  \\[-0.8ex]
  &
 &
    &
    \Sexpr{pretty_numbers(format_significance, t_stats3[["reg.lag_Return:l_piz_marketcap"]][1,"t value"])} &
    &
    \\
 $\Lambda_{i,d-1}$  &
 &
 &
 &
   \Sexpr{pretty_numbers(format_numeric, reg4[["reg.l_piz"]], p_value = t_stats4[["reg.l_piz"]][1,"Pr(>|t|)"])} &
   \Sexpr{pretty_numbers(format_numeric, reg5[["reg.l_piz"]], p_value = t_stats5[["reg.l_piz"]][1,"Pr(>|t|)"])} &
   \Sexpr{pretty_numbers(format_numeric, reg6[["reg.l_piz"]], p_value = t_stats6[["reg.l_piz"]][1,"Pr(>|t|)"])} &
  \\[-0.8ex]
 &
 &
 &
 &
   \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.l_piz"]][1,"t value"])} &
   \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.l_piz"]][1,"t value"])} &
   \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.l_piz"]][1,"t value"])} &
  \\
 $Return_{i,d-1} \times \Lambda_{i,d-1}$  &
 &
 &
 &
   \Sexpr{pretty_numbers(format_numeric, reg4[["reg.lag_Return:l_piz"]], p_value =
  t_stats4[["reg.lag_Return:l_piz"]][1,"Pr(>|t|)"])} &
   \Sexpr{pretty_numbers(format_numeric, reg5[["reg.lag_Return:l_piz"]], p_value =
  t_stats5[["reg.lag_Return:l_piz"]][1,"Pr(>|t|)"])} &
   \Sexpr{pretty_numbers(format_numeric, reg6[["reg.lag_Return:l_piz"]], p_value =
  t_stats6[["reg.lag_Return:l_piz"]][1,"Pr(>|t|)"])} &
  \\[-0.8ex]
 &
```

```
 &
 &
 &
   \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.lag_Return:l_piz"]][1,"t value"])} &
   \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.lag_Return:l_piz"]][1,"t value"])} &
   \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.lag_Return:l_piz"]][1,"t value"])} &
  \\
$|OIB_{i,d-1}|$  &
&
&
&
   \Sexpr{pretty_numbers(format_numeric, reg4[["reg.l_abs_oib"]], p_value = t_stats4[["reg.l_abs_oib"]][1,"Pr(>|t|)"])} &
   \Sexpr{pretty_numbers(format_numeric, reg5[["reg.l_abs_oib"]], p_value = t_stats5[["reg.l_abs_oib"]][1,"Pr(>|t|)"])} &
   \Sexpr{pretty_numbers(format_numeric, reg6[["reg.l_abs_oib"]], p_value = t_stats6[["reg.l_abs_oib"]][1,"Pr(>|t|)"])} &
  \\[-0.8ex]
 &
 &
 &
 &
   \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.l_abs_oib"]][1,"t value"])} &
   \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.l_abs_oib"]][1,"t value"])} &
   \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.l_abs_oib"]][1,"t value"])} &
   \\
$Return_{i,d-1} \times |OIB_{i,d-1}|$  &
&
&
&
   \Sexpr{pretty_numbers(format_numeric, reg4[["reg.lag_Return:l_abs_oib"]], p_value =
  t_stats4[["reg.lag_Return:l_abs_oib"]][1,"Pr(>|t|)"])} &
   \Sexpr{pretty_numbers(format_numeric, reg5[["reg.lag_Return:l_abs_oib"]], p_value =
  t_stats5[["reg.lag_Return:l_abs_oib"]][1,"Pr(>|t|)"])} &
   \Sexpr{pretty_numbers(format_numeric, reg6[["reg.lag_Return:l_abs_oib"]], p_value =
  t_stats6[["reg.lag_Return:l_abs_oib"]][1,"Pr(>|t|)"])} &
  \\[-0.8ex]
 &
 &
 &
 &
   \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.lag_Return:l_abs_oib"]][1,"t value"])} &
   \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.lag_Return:l_abs_oib"]][1,"t value"])} &
   \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.lag_Return:l_abs_oib"]][1,"t value"])} &
   \\
   $|Day-Return_{i,d-1}|$  &
&
&
&
    &
    &
   \Sexpr{pretty_numbers(format_numeric, reg6[["reg.l_abs_return_day"]], p_value =
  t_stats6[["reg.l_abs_return_day"]][1,"Pr(>|t|)"])} &
  \\[-0.8ex]
 &
 &
 &
 &
   &
   &
   \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.l_abs_return_day"]][1,"t value"])} &
   \\
   $Return_{i,d-1} \times |Day-Return_{i,d-1}|$  &
&
&
&
   &
   &
   \Sexpr{pretty_numbers(format_numeric, reg6[["reg.lag_Return:l_abs_return_day"]], p_value =
  t_stats6[["reg.lag_Return:l_abs_return_day"]][1,"Pr(>|t|)"])} &
  \\[-0.8ex]
```

```
     &
     &
     &
     &
       \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.lag_Return:l_abs_oib"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.lag_Return:l_abs_oib"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.lag_Return:l_abs_oib"]][1,"t value"])} &
       \\
    $R^2$ [\%] &
      \Sexpr{pretty_numbers(format_percent, 100 * reg1[["r.squared"]][["rsq"]])} &
      \Sexpr{pretty_numbers(format_percent, 100 * reg2[["r.squared"]][["rsq"]])} &
      \Sexpr{pretty_numbers(format_percent, 100 * reg3[["r.squared"]][["rsq"]])} &
      \Sexpr{pretty_numbers(format_percent, 100 * reg4[["r.squared"]][["rsq"]])} &
      \Sexpr{pretty_numbers(format_percent, 100 * reg5[["r.squared"]][["rsq"]])} &
      \Sexpr{pretty_numbers(format_percent, 100 * reg6[["r.squared"]][["rsq"]])} &
    \\
    \#Regressions &
       \Sexpr{pretty_numbers(format_integer, days1)} &
       \Sexpr{pretty_numbers(format_integer, days2)} &
       \Sexpr{pretty_numbers(format_integer, days3)} &
       \Sexpr{pretty_numbers(format_integer, days4)} &
       \Sexpr{pretty_numbers(format_integer, days5)} &
       \Sexpr{pretty_numbers(format_integer, days6)} &
    \\
    \bottomrule
    \end{tabularx}%
  \label{tab:6}%
\end{table}%

<<table7>>=
    data_complete[order(date), l_RETSQ  := shift(RETSQ , 1), by = .(permno)]
    data_complete[order(date), l2_RETSQ  := shift(RETSQ , 2), by = .(permno)]
    data_complete[order(date), l3_RETSQ  := shift(RETSQ , 3), by = .(permno)]
    data_complete[order(date), l4_RETSQ  := shift(RETSQ , 4), by = .(permno)]
    data_complete[order(date), l5_RETSQ  := shift(RETSQ , 5), by = .(permno)]
    data_complete[order(date), l_pi_abs_oib := shift(abs_lambda_oib, 1), by = .(permno)]
    data_complete[order(date), l_piz := shift(pi_z20_w, 1), by = .(permno)]
    data_complete[order(date), l_abs_oib := shift(abs(oib), 1), by = .(permno)]
    data_complete[order(date), l_PQSPR := shift(PQSPR, 1), by = .(permno)]
    data_complete[order(date), l_abs_return_day := shift(abs(return_day), 1, fill = NA), by = .(permno)]

      data_complete[, marketcap_inv  := 1/marketcap]
    data_complete[order(date), l_marketcap_inv := shift(marketcap_inv, 1), by = .(permno)]

      formula <- "RETSQ ~ lag_Return + l_RETSQ"
      fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
      reg1     <-  fmb_stock_reg$coef
      t_stats1 <- fmb_stock_reg$t_stats
      days1    <- fmb_stock_reg$days
      reg1[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

      formula_2 <- "RETSQ ~ lag_Return + l_RETSQ + l2_RETSQ + l3_RETSQ + l4_RETSQ + l5_RETSQ"
      fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula_2))
      reg2     <-  fmb_stock_reg$coef
      t_stats2 <- fmb_stock_reg$t_stats
      days2    <- fmb_stock_reg$days
      reg2[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

      formula_3 <- paste0(formula_2, "+ l_pi_abs_oib +  l_piz +  l_abs_oib")
      fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula_3))
      reg3     <-  fmb_stock_reg$coef
      t_stats3 <- fmb_stock_reg$t_stats
      days3    <- fmb_stock_reg$days
      reg3[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

      formula <- paste0(formula_3, "+ l_abs_oib* marketcap_inv")
      fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
      reg4     <-  fmb_stock_reg$coef
```

```
      t_stats4 <- fmb_stock_reg$t_stats
      days4    <- fmb_stock_reg$days
      reg4[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

      formula <- paste0(formula_3, "+ l_abs_oib*l_PQSPR")
      fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
      reg5     <-  fmb_stock_reg$coef
      t_stats5 <- fmb_stock_reg$t_stats
      days5    <- fmb_stock_reg$days
      reg5[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared

      formula <- paste0(formula_3, "+ l_abs_return_day")
      fmb_stock_reg <- fmb_reg(data_complete, as.formula(formula))
      reg6     <-  fmb_stock_reg$coef
      t_stats6 <- fmb_stock_reg$t_stats
      days6    <- fmb_stock_reg$days
      reg6[["r.squared"]][["rsq"]] <- fmb_stock_reg$r_squared
@
\renewcommand{\arraystretch}{1.4}\addtolength{\tabcolsep}{-1pt}
\begin{table}[H]
\caption[Predicting Volatility]{\textbf{Daily Fama-MacBeth regressions of volatility on previous day private information}
 \newline\newline
This table reports the time-series averages of the estimated slope coefficients from daily Fama-MacBeth regressions to explain
      cross-sectional variation in daily stock return volatility.
The dependent variable is the squared end-of-day mid-quote return of stock $i$ on day $d$ ($Return^2_{i,d}$).
The independent variables are:
the return on day $d-1$ ( $Return_{i,d-1}$),
the squared return on days $d-1$ through day $d-5$ ( $Return^2_{i,d-x}$),
the absolute order imbalance on day $d-1$ ($|OIB_{i,d-1}|$),
the price impact on day $d-1$ -- calculated as the stocks average price impact estimate over the past 20 days with setting non-positive
      price impact estimates to zero ($\lambda_{i,d-1}$),
the inverse of the market capitalization  at the beginning of each year ($1/Mktcap_{i,y-}$),
the proportional quoted spread on day $d-1$ ($PQSPR_{i,d-1}$),
and various interaction terms.
In column 6 we also control for the absolute day-return, i.e., the return from open to close.
The table reports Fama-MacBeth $t$-statistics with Newey-West corrections in parentheses below the average coefficients.
Intercepts are suppressed to conserve space. The final two rows report the $R^2$ and the number of regressions. The sample includes
      \Sexpr{pretty_numbers(format_integer, no_sample_stocks)} NYSE stocks during our sample period 2001-2014.
*, **, and *** indicate significance at the 10\%, 5\%, and 1\% levels, respectively.}
   \begin{tabular}{@{\extracolsep{\fill}}l cccccc}
   \toprule
         &   \multicolumn{5}{1}{ Dependent variable: $Return^2_{i,d}$} \\
             & (1) & (2) & (3) & (4) & (5) & (6)  \\
    \midrule
    $Return_{i,d-1}$  &
       \Sexpr{pretty_numbers(format_numeric, reg1[["reg.lag_Return"]], p_value = t_stats1[["reg.lag_Return"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg2[["reg.lag_Return"]], p_value = t_stats2[["reg.lag_Return"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg3[["reg.lag_Return"]], p_value = t_stats3[["reg.lag_Return"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg4[["reg.lag_Return"]], p_value = t_stats4[["reg.lag_Return"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg5[["reg.lag_Return"]], p_value = t_stats5[["reg.lag_Return"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg6[["reg.lag_Return"]], p_value = t_stats6[["reg.lag_Return"]][1,"Pr(>|t|)"])}
      \\[-0.8ex]
     &
       \Sexpr{pretty_numbers(format_significance, t_stats1[["reg.lag_Return"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats2[["reg.lag_Return"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats3[["reg.lag_Return"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.lag_Return"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.lag_Return"]][1,"t value"])} &
       \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.lag_Return"]][1,"t value"])}
      \\
        $Return^2_{i,d-1}$  &
       \Sexpr{pretty_numbers(format_numeric, reg1[["reg.l_RETSQ"]], p_value = t_stats1[["reg.l_RETSQ"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg2[["reg.l_RETSQ"]], p_value = t_stats2[["reg.l_RETSQ"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg3[["reg.l_RETSQ"]], p_value = t_stats3[["reg.l_RETSQ"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg4[["reg.l_RETSQ"]], p_value = t_stats4[["reg.l_RETSQ"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg5[["reg.l_RETSQ"]], p_value = t_stats5[["reg.l_RETSQ"]][1,"Pr(>|t|)"])} &
       \Sexpr{pretty_numbers(format_numeric, reg6[["reg.l_RETSQ"]], p_value = t_stats6[["reg.l_RETSQ"]][1,"Pr(>|t|)"])}
      \\[-0.8ex]
```

```
&
  \Sexpr{pretty_numbers(format_significance, t_stats1[["reg.l_RETSQ"]][1,"t value"])} &
  \Sexpr{pretty_numbers(format_significance, t_stats2[["reg.l_RETSQ"]][1,"t value"])} &
  \Sexpr{pretty_numbers(format_significance, t_stats3[["reg.l_RETSQ"]][1,"t value"])} &
  \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.l_RETSQ"]][1,"t value"])} &
  \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.l_RETSQ"]][1,"t value"])} &
  \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.l_RETSQ"]][1,"t value"])}
 \\
   $Return^2_{i,d-2}$  &
&
  \Sexpr{pretty_numbers(format_numeric, reg2[["reg.l2_RETSQ"]], p_value = t_stats2[["reg.l2_RETSQ"]][1,"Pr(>|t|)"])} &
  \Sexpr{pretty_numbers(format_numeric, reg3[["reg.l2_RETSQ"]], p_value = t_stats3[["reg.l2_RETSQ"]][1,"Pr(>|t|)"])} &
  \Sexpr{pretty_numbers(format_numeric, reg4[["reg.l2_RETSQ"]], p_value = t_stats4[["reg.l2_RETSQ"]][1,"Pr(>|t|)"])} &
  \Sexpr{pretty_numbers(format_numeric, reg5[["reg.l2_RETSQ"]], p_value = t_stats5[["reg.l2_RETSQ"]][1,"Pr(>|t|)"])} &
  \Sexpr{pretty_numbers(format_numeric, reg6[["reg.l2_RETSQ"]], p_value = t_stats6[["reg.l2_RETSQ"]][1,"Pr(>|t|)"])}
 \\[-0.8ex]
&
&
  \Sexpr{pretty_numbers(format_significance, t_stats2[["reg.l2_RETSQ"]][1,"t value"])} &
  \Sexpr{pretty_numbers(format_significance, t_stats3[["reg.l2_RETSQ"]][1,"t value"])} &
  \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.l2_RETSQ"]][1,"t value"])} &
  \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.l2_RETSQ"]][1,"t value"])} &
  \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.l2_RETSQ"]][1,"t value"])}
 \\
   $Return^2_{i,d-3}$  &
&
  \Sexpr{pretty_numbers(format_numeric, reg2[["reg.l3_RETSQ"]], p_value = t_stats2[["reg.l3_RETSQ"]][1,"Pr(>|t|)"])} &
  \Sexpr{pretty_numbers(format_numeric, reg3[["reg.l3_RETSQ"]], p_value = t_stats3[["reg.l3_RETSQ"]][1,"Pr(>|t|)"])} &
  \Sexpr{pretty_numbers(format_numeric, reg4[["reg.l3_RETSQ"]], p_value = t_stats4[["reg.l3_RETSQ"]][1,"Pr(>|t|)"])} &
  \Sexpr{pretty_numbers(format_numeric, reg5[["reg.l3_RETSQ"]], p_value = t_stats5[["reg.l3_RETSQ"]][1,"Pr(>|t|)"])} &
  \Sexpr{pretty_numbers(format_numeric, reg6[["reg.l3_RETSQ"]], p_value = t_stats6[["reg.l3_RETSQ"]][1,"Pr(>|t|)"])}
 \\[-0.8ex]
&
&
  \Sexpr{pretty_numbers(format_significance, t_stats2[["reg.l3_RETSQ"]][1,"t value"])} &
  \Sexpr{pretty_numbers(format_significance, t_stats3[["reg.l3_RETSQ"]][1,"t value"])} &
  \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.l3_RETSQ"]][1,"t value"])} &
  \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.l3_RETSQ"]][1,"t value"])} &
  \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.l3_RETSQ"]][1,"t value"])}
 \\
   $Return^2_{i,d-4}$  &
&
  \Sexpr{pretty_numbers(format_numeric, reg2[["reg.l4_RETSQ"]], p_value = t_stats2[["reg.l4_RETSQ"]][1,"Pr(>|t|)"])} &
  \Sexpr{pretty_numbers(format_numeric, reg3[["reg.l4_RETSQ"]], p_value = t_stats3[["reg.l4_RETSQ"]][1,"Pr(>|t|)"])} &
  \Sexpr{pretty_numbers(format_numeric, reg4[["reg.l4_RETSQ"]], p_value = t_stats4[["reg.l4_RETSQ"]][1,"Pr(>|t|)"])} &
  \Sexpr{pretty_numbers(format_numeric, reg5[["reg.l4_RETSQ"]], p_value = t_stats5[["reg.l4_RETSQ"]][1,"Pr(>|t|)"])} &
  \Sexpr{pretty_numbers(format_numeric, reg6[["reg.l4_RETSQ"]], p_value = t_stats6[["reg.l4_RETSQ"]][1,"Pr(>|t|)"])}
 \\[-0.8ex]
&
&
  \Sexpr{pretty_numbers(format_significance, t_stats2[["reg.l4_RETSQ"]][1,"t value"])} &
  \Sexpr{pretty_numbers(format_significance, t_stats3[["reg.l4_RETSQ"]][1,"t value"])} &
  \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.l4_RETSQ"]][1,"t value"])} &
  \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.l4_RETSQ"]][1,"t value"])} &
  \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.l4_RETSQ"]][1,"t value"])}
 \\
   $Return^2_{i,d-5}$  &
&
  \Sexpr{pretty_numbers(format_numeric, reg2[["reg.l5_RETSQ"]], p_value = t_stats2[["reg.l5_RETSQ"]][1,"Pr(>|t|)"])} &
  \Sexpr{pretty_numbers(format_numeric, reg3[["reg.l5_RETSQ"]], p_value = t_stats3[["reg.l5_RETSQ"]][1,"Pr(>|t|)"])} &
  \Sexpr{pretty_numbers(format_numeric, reg4[["reg.l5_RETSQ"]], p_value = t_stats4[["reg.l5_RETSQ"]][1,"Pr(>|t|)"])} &
  \Sexpr{pretty_numbers(format_numeric, reg5[["reg.l5_RETSQ"]], p_value = t_stats5[["reg.l5_RETSQ"]][1,"Pr(>|t|)"])} &
  \Sexpr{pretty_numbers(format_numeric, reg6[["reg.l5_RETSQ"]], p_value = t_stats6[["reg.l5_RETSQ"]][1,"Pr(>|t|)"])}
 \\[-0.8ex]
&
&
  \Sexpr{pretty_numbers(format_significance, t_stats2[["reg.l5_RETSQ"]][1,"t value"])} &
  \Sexpr{pretty_numbers(format_significance, t_stats3[["reg.l5_RETSQ"]][1,"t value"])} &
```

```
    \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.l5_RETSQ"]][1,"t value"])} &
    \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.l5_RETSQ"]][1,"t value"])} &
    \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.l5_RETSQ"]][1,"t value"])}
  \\
 $\Lambda_{i,d-1} \times |OIB_{i,d-1}|$  &
 &
 &
    \Sexpr{pretty_numbers(format_numeric, reg3[["reg.l_pi_abs_oib"]], p_value = t_stats3[["reg.l_pi_abs_oib"]][1,"Pr(>|t|)"])} &
    \Sexpr{pretty_numbers(format_numeric, reg4[["reg.l_pi_abs_oib"]], p_value = t_stats4[["reg.l_pi_abs_oib"]][1,"Pr(>|t|)"])} &
    \Sexpr{pretty_numbers(format_numeric, reg5[["reg.l_pi_abs_oib"]], p_value = t_stats5[["reg.l_pi_abs_oib"]][1,"Pr(>|t|)"])} &
    \Sexpr{pretty_numbers(format_numeric, reg6[["reg.l_pi_abs_oib"]], p_value = t_stats6[["reg.l_pi_abs_oib"]][1,"Pr(>|t|)"])}
  \\[-0.8ex]
 &
 &
 &
    \Sexpr{pretty_numbers(format_significance, t_stats3[["reg.l_pi_abs_oib"]][1,"t value"])} &
    \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.l_pi_abs_oib"]][1,"t value"])} &
    \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.l_pi_abs_oib"]][1,"t value"])} &
    \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.l_pi_abs_oib"]][1,"t value"])}
  \\
 $\Lambda_{i,d-1}$  &
 &
 &
    \Sexpr{pretty_numbers(format_numeric, reg3[["reg.l_piz"]], p_value = t_stats3[["reg.l_piz"]][1,"Pr(>|t|)"])} &
    \Sexpr{pretty_numbers(format_numeric, reg4[["reg.l_piz"]], p_value = t_stats4[["reg.l_piz"]][1,"Pr(>|t|)"])} &
    \Sexpr{pretty_numbers(format_numeric, reg5[["reg.l_piz"]], p_value = t_stats5[["reg.l_piz"]][1,"Pr(>|t|)"])} &
    \Sexpr{pretty_numbers(format_numeric, reg6[["reg.l_piz"]], p_value = t_stats6[["reg.l_piz"]][1,"Pr(>|t|)"])}
  \\[-0.8ex]
 &
 &
 &
    \Sexpr{pretty_numbers(format_significance, t_stats3[["reg.l_piz"]][1,"t value"])} &
    \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.l_piz"]][1,"t value"])} &
    \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.l_piz"]][1,"t value"])} &
    \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.l_piz"]][1,"t value"])}
  \\
 $|OIB_{i,d-1}|$  &
 &
 &
    \Sexpr{pretty_numbers(format_numeric, reg3[["reg.l_abs_oib"]], p_value = t_stats3[["reg.l_pi_abs_oib"]][1,"Pr(>|t|)"])} &
    \Sexpr{pretty_numbers(format_numeric, reg4[["reg.l_abs_oib"]], p_value = t_stats4[["reg.l_pi_abs_oib"]][1,"Pr(>|t|)"])} &
    \Sexpr{pretty_numbers(format_numeric, reg5[["reg.l_abs_oib"]], p_value = t_stats5[["reg.l_pi_abs_oib"]][1,"Pr(>|t|)"])} &
    \Sexpr{pretty_numbers(format_numeric, reg6[["reg.l_abs_oib"]], p_value = t_stats6[["reg.l_pi_abs_oib"]][1,"Pr(>|t|)"])}
  \\[-0.8ex]
 &
 &
 &
    \Sexpr{pretty_numbers(format_significance, t_stats3[["reg.l_abs_oib"]][1,"t value"])} &
    \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.l_abs_oib"]][1,"t value"])} &
    \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.l_abs_oib"]][1,"t value"])} &
    \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.l_abs_oib"]][1,"t value"])}
    \\
 $1/Marketcap_{i,d-1}$  &
 &
 &
     &
    \Sexpr{pretty_numbers(format_numeric, reg4[["reg.l_abs_oib"]], p_value = t_stats4[["reg.marketcap_inv"]][1,"Pr(>|t|)"])} &
     &
  \\[-0.8ex]
 &
 &
 &
   &
    \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.marketcap_inv"]][1,"t value"])} &
     &
 \\
 $|OIB_{i,d-1}|\times 1/Marketcap_{i,d-1}$  &
 &
```

```
&
   &
   \Sexpr{pretty_numbers(format_numeric, reg4[["reg.l_abs_oib:marketcap_inv"]], p_value =
  t_stats4[["reg.l_abs_oib:marketcap_inv"]][1,"Pr(>|t|)"])} &
   &
  \\[-0.8ex]
&
&
&
   &
   \Sexpr{pretty_numbers(format_significance, t_stats4[["reg.l_abs_oib:marketcap_inv"]][1,"t value"])} &
   &
\\
$PQSPR_{i,d-1}$  &
&
&
   &
   &
   \Sexpr{pretty_numbers(format_numeric, reg5[["reg.l_PQSPR"]], p_value = t_stats5[["reg.l_PQSPR"]][1,"Pr(>|t|)"])} &
  \\[-0.8ex]
&
&
&
   &
   &
   \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.l_PQSPR"]][1,"t value"])} &
\\
$|OIB_{i,d-1}|\times PQSPR_{i,d-1}$  &
&
&
   &
   &
   \Sexpr{pretty_numbers(format_numeric, reg5[["reg.l_abs_oib:l_PQSPR"]], p_value =
  t_stats5[["reg.l_abs_oib:l_PQSPR"]][1,"Pr(>|t|)"])} &
  \\[-0.8ex]
&
&
&
   &
   &
   \Sexpr{pretty_numbers(format_significance, t_stats5[["reg.l_abs_oib:l_PQSPR"]][1,"t value"])} &
\\
   $|Day-Return_{i,d-1}|$  &
&
&
   &
   &
   &
   \Sexpr{pretty_numbers(format_numeric, reg6[["reg.l_abs_return_day"]], p_value =
  t_stats6[["reg.l_abs_return_day"]][1,"Pr(>|t|)"])}
  \\[-0.8ex]
&
&
&
   &
   &
   &
   \Sexpr{pretty_numbers(format_significance, t_stats6[["reg.l_abs_return_day"]][1,"t value"])}
 \\
$R^2$ [\%] &
  \Sexpr{pretty_numbers(format_percent, 100 * reg1[["r.squared"]][["rsq"]])} &
  \Sexpr{pretty_numbers(format_percent, 100 * reg2[["r.squared"]][["rsq"]])} &
  \Sexpr{pretty_numbers(format_percent, 100 * reg3[["r.squared"]][["rsq"]])} &
  \Sexpr{pretty_numbers(format_percent, 100 * reg4[["r.squared"]][["rsq"]])} &
  \Sexpr{pretty_numbers(format_percent, 100 * reg5[["r.squared"]][["rsq"]])} &
  \Sexpr{pretty_numbers(format_percent, 100 * reg6[["r.squared"]][["rsq"]])}
\\
\#Regressions &
```

```
      \Sexpr{pretty_numbers(format_integer, days1)} &
      \Sexpr{pretty_numbers(format_integer, days2)} &
      \Sexpr{pretty_numbers(format_integer, days3)} &
      \Sexpr{pretty_numbers(format_integer, days4)} &
      \Sexpr{pretty_numbers(format_integer, days5)} &
      \Sexpr{pretty_numbers(format_integer, days6)}
    \\
    \bottomrule
    \end{tabular}%
  \label{tab:7}%
\end{table}%


\clearpage

\section{Functions and Setup Code}

\label{R:setup}
<<setup, eval=FALSE>>=
@


<<sessionInfo>>=
sessionInfo()
@

<<cleanup>>=
lapply(dbListConnections(PostgreSQL()), dbDisconnect)
 @

\ifdefined\includemain

% drop the following if included into other file

\else

  \addcontentsline{toc}{chapter}{\numberline{}References}
  \renewcommand{\bibname}{References}
  \bibliographystyle{chicago}
  \bibliography{\jobname}


  \clearpage
  \pagenumbering{gobble}

  \section{Appendix: Source Code}

  \lstset{
    basicstyle=\ttfamily\tiny,
    breaklines=true,
    breakatwhitespace=true,
    columns=fullflexible,
    frame=single,
    backgroundcolor=\color{gray!10},
    xleftmargin=10pt,
    language=R,
    keepspaces=true,
    showstringspaces=false,
    morecomment=[l][\color{gray}]{\%},
    morecomment=[l][\color{gray}]{\#},
    postbreak={\CommentLineContinued},
    upquote=true
  }



  \lstinputlisting[caption=Source Code]{\jobname.Rnw}
```

```
\fi

\end{document}
```