The Stata Journal (2018)
18, Number 2, pp. 461–476

Event study estimations using Stata: The estudy command

Fausto Pacicco
LIUC–Università Carlo Cattaneo
Castellanza, Italy
fpacicco@liuc.it

Luigi Vena
LIUC–Università Carlo Cattaneo
Castellanza, Italy
lvena@liuc.it

Andrea Venegoni
LIUC–Università Carlo Cattaneo
Castellanza, Italy
avenegoni@liuc.it

Abstract. In this article, we introduce the community-contributed command `estudy` and illustrate how it can be used to perform an event study customizing the statistical framework, from the estimates of abnormal returns to the tests for their statistical significance. Our command significantly improves the existing commands in terms of both completeness and user comprehension.

Keywords: st0532, estudy, event study, financial econometrics

1 Introduction

If and how a given event affects financial markets is a relevant question that researchers and practitioners aim to answer. This is why the event study framework has become a statistical technique used in many areas, including economics, accounting, finance, and law. According to Kothari and Warner (2007), between 1974 and 2000, almost 600 studies conducted in various fields used such a technique. If we consider that the mentioned analysis accounts for only four main academic journals, it is easy to understand that the numbers describing the popularity of such a technique exponentially grow when we extend the focus to other academic journals as well as private and public institutions.

In this article, we introduce and comment on the community-contributed command `estudy`, which performs an event study permitting the user to i) work with multiple *varlists*, computing the abnormal returns (ARs), average abnormal returns (AARs), cumulative abnormal returns (CARs), and cumulative average abnormal returns (CAARs);[1] ii) specify up to six event windows; iii) customize the length of the estimation window; iv) select the model for the calculation of normal or abnormal returns; v) specify the diagnostic test, among the parametric and nonparametric ones that we propose and that are the most commonly used in the literature; vi) customize the output table; vii) and store the results in an Excel file and in a Stata data file.

1. Throughout the entire article, we refer to ARs only for exposition's sake.

 st0532

`estudy` improves the existing commands (see, for example, the `eventstudy` command written by Zhang et al. [2013] and Kaspereit's [2015] `eventstudy2` command, available on the Statistical Software Components archive) that allow one to perform an event study in Stata under several perspectives. `estudy`

- simultaneously computes ARs (CARs) of more than one group of variables (securities' returns) without repeating the command;
- offers the possibility to work with different levels of aggregation, running the event study on single firms (groups of them) computing ARs and CARs (AARs and CAARs) and testing their statistical significance;
- provides a customizable output according to the needs of the user; and
- simplifies the approach for the user, resulting in an easier and faster setup.

2 The event study framework

If an event is unexpected and value relevant for some firms, it is bound to cause an AR as measured by the actual ex post return net to the normal (or expected) one over the same period (see, for example, MacKinlay [1997], Kolari and Pynnönen [2011], or Brown and Warner [1985]). The event study technique allows one to measure such an AR and thus assess whether a given fact has influenced firms' securities' market value. We refer generally to firms' securities because event studies apply most frequently to common stocks, even if they are conducted on others securities like bonds (Bessembinder et al. 2009) or credit default swaps (Andres, Betzer, and Doumet 2016).

Equation (1) defines the AR of a generic firm i in the period t,

$$\mathrm{AR}_{i,t} = R_{i,t} - E(R_{i,t}|X_t) \tag{1}$$

where $R_{i,t}$ is the actual ex post return and $E(R_{i,t}|X_t)$ is the expected return conditioned to the information X of period t, unrelated to the event.

2.1 Measuring ARs

As pointed out by MacKinlay (1997), the conduction of the event study customarily follows an established flow divided in these steps:

1. Definition of the event window
2. Computation of the normal returns
   a. Definition of the estimation window
   b. Choice of the estimation model
3. Estimation of the ARs
4. Statistical testing for the significance of the ARs

The procedure begins with the definition of the period or periods over which the event is supposed to influence the market return of firms' securities, that is, the event window or windows. Usually, each event window spans one or more days, including the event date itself. It is common to include the days before and after the event, allowing for the possibility of news leakages preceding the event itself or delayed reactions of the markets.

To define the AR, one must proceed to the second step of the analysis and compute the normal or expected performance. This task requires the definition of an estimation window, that is, a sample period prior to the event window, usually leaving a cushion of at least one month to exclude market returns influenced by the event, avoiding the estimation window to include anticipation effects (or news leakages).

The estimation of ARs is carried out using different models, the most common of which is the single index model (SIM) (MacKinlay 1997; Sorokina, Booth, and Thornton 2013) as represented in (2):

$$\mathrm{AR}_{i,t} = R_{i,t} - (\alpha_i + \beta_i R_{m,t}) \tag{2}$$

According to the SIM,[2] the normal return depends on the parameters α_i and β_i (estimated over the estimation windows) and the market return $R_{m,t}$. A special case of this model is the market adjusted model (MAM), where a constraint on the parameters α_i and β_i wants them to be set equal to 0 and 1, respectively.

Another possible specification that, as Brown and Warner (1985) claim, yields results comparable with the SIM is the historical mean model (HMM). According to the HMM, the security's historical mean return over the estimation window (μ_i) represents the expected normal performance unconditioned to the event [as shown in (3)]:

$$E(R_{i,t}|X_t) = \mu_i \tag{3}$$

In an attempt to improve the variance explained by the SIM (hence facilitating AR detection), sometimes the expected return is estimated using more than one factor, that is, modeling a multifactor model (MFM) such as the three-factors model introduced by Fama and French (1993).

Once normal returns are computed, it is possible to obtain ARs. When the aim is to compute the event impact for each single security on a single day event, it is possible to obtain ARs by applying (1). Sometimes, the user will also be interested in investigating the effect of the event on a multiday period. Hence, it becomes necessary to operate a time-series aggregation of the ARs, obtaining the CARs as described by (4).

2. Here the expected returns conditioned to the information X of period t are computed as $E(R_{i,t}|X_t) = \alpha_i + \beta_i R_{m,t}$.

$$\text{CAR}_i(t_1, t_2) = \sum_{t=t_1}^{t_2} \text{AR}_{i,t} \tag{4}$$

with $t_1 < t_2$ and $t_1, t_2 \in$ (event window).

If instead the object of interest is the impact on a pool of firms, a cross-section aggregation becomes necessary, and AAR calculation can be performed using (5).

$$\text{AAR}_t = \frac{1}{N} \sum_{i=1}^{N} \text{AR}_{i,t} \tag{5}$$

where $\text{AR}_{i,t}$ represents the AR estimated on the ith security and N the securities' population. In the words of Kothari and Warner (2007), the cross-sectional aggregation of ARs makes sense if one aims either at studying whether the event alters, on average, the security holders' wealth or at testing economic models and alternative hypotheses suggesting the sign of the mean impact.

Finally, when the focus is on the average effect over multiple days, it is necessary to perform both of the aggregations just described and compute the CAARs by summing over time the AARs, as shown in (6).

$$\text{CAAR}(t_1, t_2) = \sum_{t=t_1}^{t_2} \text{AAR}_t \tag{6}$$

The literature suggests the portfolio approach, which is an alternative method to computing both AARs and CAARs (Kothari and Warner 2007). Levering on an equally weighted portfolio that groups all securities under scrutiny (before computing the abnormal components), one can compute the portfolio ARs (a substitute for the AARs) and CARs (instead of the CAARs), considering the portfolio as a single security. By default, the `estudy` command performs both techniques.

2.2 Statistical properties of ARs

Once ARs are computed in any form that suits the analysis, it is necessary to study their statistical significance. To assume economic relevance, one must consider ARs[3] statistically significant; that is, their difference from zero must be verified using an ad hoc test.

To this end, the literature offers two types of tests, parametric and nonparametric. While the former assumes a certain distribution of returns, the latter is not anchored to any a priori assumption (Kolari and Pynnönen 2010, 2011; Kothari and Warner 2007).

3. The same holds for AARs, CARs, and CAARs.

With respect to the former family, under the assumption of normally distributed securities' returns, ARs follow a normal distribution centered on 0, with variance σ^2_{AR}. Accordingly, AARs, CARs, and CAARs also are normally distributed with mean 0 and variance σ^2_{AAR}, σ^2_{CAR}, and σ^2_{CAAR}.[4] We dub this test "normal".

Alternatively, Patell (1976) suggests a parametric test that, based on scaled ARs, brings a twofold benefit. On one hand, the test accounts for the diverse standard deviations between event-period and estimation-period residuals. On the other hand, it prevents securities with large variance to heavily influence the outcome; we refer to this test as "Patell".

Boehmer, Masumeci, and Poulsen (1991) (BMP test) improve Patell's test by accounting for the possible cross-sectional increase in the variance of the returns that may occur within the event window.

However, these three tests (normal, Patell, and BMP) suffer from the cross-sectional correlation of ARs that heavily affects their outcome in the case of event-day clustering that verifies when a single event simultaneously affects all securities included in the analysis. To overcome this problem, Kolari and Pynnönen (2010) modify both Patell and BMP tests, introducing a correction for the cross-correlation and hence proposing the adjusted Patell (AdjPatell) and Kolari and Pynnönen (2010) tests.

Being linked to the normality assumption of the securities' return distributions, the aforementioned tests may underperform when returns are not normal. Thus, without relying on any distribution, the test presented by Wilcoxon (1945) checks for the statistical significance of AARs, considering both the signs and the magnitude of ARs, while the Kolari and Pynnönen (2011) generalized rank test outperforms both the previous rank tests and the parametric ones without suffering either from the serial correlation of ARs or from the event-induced volatility.

All the listed tests are included in the `estudy` package.

3 The estudy command

The `estudy` command performs the event study, computing the ARs and running the proper diagnostic, as described in the previous section.

4. For a complete description of these variances, see MacKinlay (1997) and Binder (1998), among others.

3.1 Syntax

The syntax for `estudy` is

`estudy` *varlist1* [`(`*varlist2*`)` ... `(`*varlistN*`)`] `,` `datevar(`*varname*`)` `evdate(`*date*`)` `dateformat(`*string*`)` `lb1(`#`)` `ub1(`#`)` [`lb2(`#`)` `ub2(`#`)` ... `lb6(`#`)` `ub6(`#`)` `eswlbound(`#`)` `eswubound(`#`)` `modtype(`*string*`)` `indexlist(`*varlist*`)` `diagnosticstat(`*string*`)` `suppress(`*string*`)` `decimal(`#`)` `showpvalues` `nostar` `outputfile(`*filename*`)` `mydataset(`*datasetname*`)`]

The (maximum) N *varlists* must be the securities' logarithmic return of the financial instruments subject to the event study. Each *varlist* except the first one must be specified in parentheses.

3.2 Options

`datevar(`*varname*`)` specifies the date variable in the dataset. The command cannot perform the event studies if the time series of securities' returns is not linked to a date variable. When the variable indicated here is not date formatted, the command stops. `datevar()` is required.

`evdate(`*date*`)` specifies the date of the event. No matter the format of `datevar()`, `evdate()` may be expressed as *mmddyyyy*, *ddmmyyyy*, or *yyyymmdd*. `evdate()` is required.

`dateformat(`*string*`)` specifies the date format of the event date (`evdate()`). There are three cases—`MDY`, `DMY`, or `YMD`—to indicate that the `evdate()` option has been specified, respectively, as *mmddyyyy*, *ddmmyyyy*, or *yyyymmdd*. `dateformat()` is required.

`lb1(`#`)`, `ub1(`#`)`, `lb2(`#`)`, `ub2(`#`)`, ..., `lb6(`#`)`, and `ub6(`#`)` specify up to six event windows around the event date (`lb1()` and `ub1()` are required). For each event window, both the lower and upper bounds must be specified and have an integer format.

`eswlbound(`#`)` specifies the lower bound of the estimation window. By default, the command uses the first trading day in the database.

`eswubound(`#`)` specifies the upper bound of the estimation window. By default, it corresponds to the 30th trading day prior to the event, thus avoiding an overlap between the estimation and the event windows.

`modtype(`*string*`)` specifies which model must be used to compute the normal or abnormal returns. *string* may be one of the following:

`SIM` (SIM), the default, requires to specify only one variable (factor) in `indexlist()`.

`MAM` (MAM) requires to specify only one variable (factor) in `indexlist()`.

`MFM` (multifactor model) requires to specify more than one variable (factors) in `indexlist()`.

`HMM` (HMM) ignores `indexlist()`.

`indexlist(`*varlist*`)` specifies the *varlist* useful to compute a normal or an abnormal component of securities' returns and is conditional to the `modtype()` option. When either the SIM (`modtype(SIM)`) or the MAM (`modtype(MAM)`) has been specified, this option must indicate only one variable. With the MFM (`modtype(MFM)`), more than one variable must be specified. When the HMM (`modtype(HMM)`) has been set, the command ignores this option.

`diagnosticstat(`*string*`)` specifies which test must be used to analyze whether ARs statistically differ from zero (parametric and nonparametric tests are available).

string for parametric tests may be one of the following:

`Norm` is the default test, assuming that securities' returns, and hence ARs, are normally distributed and homoskedastic across the estimation and the event windows. Despite that these assumptions are often violated, this test is commonly used to evaluate the statistical significance of ARs and CARs on single securities.

`Patell` performs the test proposed by Patell (1976).

`ADJPatell` performs the test proposed by Patell (1976) with the Kolari and Pynnönen (2010) adjustment for cross-correlation of ARs.

`BMP` performs the test proposed by Boehmer, Masumeci, and Poulsen (1991), which improves Patell's test while accounting for the event-induced volatility.

`KP` performs the BMP test corrected for the cross-sectional correlation of ARs (see Kolari and Pynnönen [2010] for further details).

string for nonparametric tests may be one of the following:

`Wilcoxon` performs the signed-rank test proposed by Wilcoxon (1945).

`GRANK` performs the generalized rank test proposed by Kolari and Pynnönen (2011).

`suppress(`*string*`)` specifies the format of the output table. *string* may be either `group` or `ind`. With the former, the output table shows only the ARs on each input variable, hiding those on the group as a whole and on the portfolio. With the latter, only group ARs, and the results for the portfolio approach are shown, with individual ARs excluded from the output table. By default, the output table prints the single ARs for each security of each *varlist*, the portfolio ARs, and the group AR of each *varlist*. A horizontal line separates the *varlists*.

`decimal(#)` specifies the number of decimals that must be used in the output table. The maximum value is `decimal(7)`. By default, the number of decimals is set equal to two.

`showpvalues` specifies that the output table must show the *p*-value of each AR. When this option is specified, *p*-values are shown in parentheses below the corresponding AR.

`nostar` specifies that the output file must not contain the stars indicating the significance level. By default, `***`, `**`, and `*` denote that ARs are statistically significant at the 1%, 5%, and 10% levels, respectively.

`outputfile(`*filename*`)` specifies the name of the `.xlsx` file in which both the ARs (always without significance stars) and the *p*-values are stored in two separate sheets. The format imposed with `suppress()` is maintained. The command automatically replaces the file, should it already exist.

`mydataset(`*datasetname*`)` specifies the name of the `.dta` file in which ARs (always without significance stars) are stored. The format imposed with `suppress()` is maintained. The command automatically replaces the file, should it already exist. The work file, stored in the directory in use, contains a first variable with the securities' labels and the ARs on each event window in just as many variables.

4 Examples

We illustrate how `estudy` works, using the dataset `data_estudy` provided by us. This dataset contains the time series of market returns for 10 companies' shares, as well as the Fama and French (1993) three factors and the risk-free rate. Through our example, we show the command's syntax, clarifying how each option can be used to customize the command and thus meet the users' needs.

As previously pointed out, the options for `estudy` can be divided in two groups, required and additional. While the former is necessary, the latter can be used to customize the analysis but does not impair the functioning of the command. We first show the basic model, encompassing only required options, leaving to subsequent examples the demonstration and use of the additional ones.

Thus, we start with a simple setup, performing an event study on two (separate) *varlists*, with only one event window of seven days (from -3 to $+3$) around 09 July 2015, that is, the event date. In addition to the `evdate()` and `lb1()/lb2()` options (useful to specify, respectively, the event date and window), two other options are mandatory: 1) `dateformat()`, the format of the event date (in this case, `MDY` as the event date has the format *mmddyyyy*); and 2) `datevar()`, the date variable present in the dataset.

Because we are not specifying any model type, normal and abnormal returns are computed according to a SIM (which is set to the default). As such, the market (or index) return must be indicated with the option `indexlist()`.

```
. use data_estudy.dta

. estudy boa ford boeing (apple netflix amazon facebook google), datevar(date)
> evdate(07092015) dateformat(MDY) indexlist(mkt) lb1(-3) ub1(3)
By default the upper bound of the estimation window has been set to (-30)
Event date: 09jul2015, with 1 event windows specified, under the Normality
> assumption
SECURITY                      CAAR[-3,3]
Bank of America Corporation    -1.15%
Ford Motor Company             -1.85%
The Boeing Company              3.48%
Ptf CARs n 1 (3 securities)     0.16%
CAAR group 1  (3 securities)    0.16%
-----------------------------------------------------------
Apple Inc                      -2.12%
Netflix Inc                     3.00%
Amazon com Inc                  4.17%
Facebook Inc                    0.00%
Alphabet Inc                    5.33%*
Ptf CARs n 2 (5 securities)     2.08%
CAAR group 2  (5 securities)    2.08%
-----------------------------------------------------------

*** p-value < .01, ** p-value <.05, * p-value <.1
```

When the estimation window is not specified, as in this case, then by default the command considers it to be from the first available to the 30th trading day prior to the event (-30). A warning message reminds the user of this. Moreover, the header of the output briefly recaps the setup of the event study performed, reminding the user of the event date, the number of event windows specified, and the diagnostic test implemented.

The first column reports the labels of the variables on which the event study has been performed. By default, the command adds two rows per *varlist*, showing the results for the portfolio approach and the group ARs; both are useful to evaluate the average impact of the event. The remaining column reports the ARs (in this case, CARs and CAARs over the $[-3, 3]$ window). Statistically significant ARs are identified by asterisks as explained by the legend at the bottom of the table. Horizontal lines separate the table in panels, each showing the specified *varlists*.

The results from this application tell us that in the specified event window, the event that occurred in the selected date has not exercised a significant impact on all the firms included in the analysis except for on Alphabet Inc., which reports a positive and significant AR. From this, we can infer that some kind of news concerning this company may have been released on that day. Looking at the aggregate CARs, we see that neither the ones computed using the portfolio technique nor the ones estimated averaging the single firms' CARs have significant results, meaning that the event registered on that given date has no influence on both groups stock returns.

In the next example, we customize our analysis by i) changing the statistical test implemented, with the option `diagnosticsstat`; ii) setting a precise estimation window, using the options `eswlbound()` and `eswubound()` to specify the lower and the upper bound, respectively; and iii) adding two event windows (the options `lb2()`, `ub2()`, `lb3()`, and `ub3()`).

```
. estudy boa ford boeing (apple netflix amazon facebook google), datevar(date)
> evdate(07092015) dateformat(MDY) indexlist(mkt) lb1(-3) ub1(3) lb2(-3) ub2(-1)
> lb3(0) ub3(3) diagnosticsstat(BMP) eswlb(-250) eswub(-20)
Event date: 09jul2015, with 3 event windows specified, using the Boehmer,
> Musumeci, Poulsen test
SECURITY                      CAAR[-3,3]       CAAR[-3,-1]      CAAR[0,3]
Bank of America Corporation    -1.29%           -2.99%*           1.70%
Ford Motor Company             -1.42%           -1.43%            0.01%
The Boeing Company              3.71%            2.57%            1.14%
Ptf CARs n 1 (3 securities)     0.33%           -0.62%            0.95%
CAAR group 1  (3 securities)    0.33%           -0.62%            0.95%*
--------------------------------------------------------------------------------
Apple Inc                      -3.00%           -1.72%           -1.27%
Netflix Inc                     3.77%            0.70%            3.07%
Amazon com Inc                  4.02%           -0.54%            4.56%
Facebook Inc                    0.54%           -0.39%            0.94%
Alphabet Inc                    5.86%**          0.42%            5.44%***
Ptf CARs n 2 (5 securities)     2.24%           -0.31%            2.55%
CAAR group 2  (5 securities)    2.24%           -0.31%            2.55%
--------------------------------------------------------------------------------
*** p-value < .01, ** p-value <.05, * p-value <.1
```

Because we have set the estimation window, there is no warning message. The output table now has two more columns showing the new event windows. The table header recalls that the Boehmer, Masumeci, and Poulsen (1991) test has been implemented to test the significance of ARs over the three event windows specified.

Here the results tell us that if we change the length of the event windows surrounding the selected event date, we find that Bank of America registers a significant yet negative AR, so the information gathered by agents was bad news for the company. Considering the aggregate, if we look at the last event window (CA(A)R $[0, 3]$), the first group of firms, according to the results yielded by the cross-sectional averaging aggregation technique, appears to be positively affected by the event that occurred.

Should one be interested in the mean effect of an event, ARs on the single variables can be suppressed. The option `suppress(ind)` meets this need.

```
. estudy boa ford boeing(ibm facebook apple) (netflix cocacola amazon) (facebook
> boa ford boeing google), datevar(date) evdate(07092015) dateformat(MDY)
> modtype(HMM) lb1(-3) ub1(3) lb2(-3) ub2(-1) lb3(0) ub3(3) diagnosticsstat(KP)
> eswlb(-250) eswub(-20) suppress(ind)
Event date: 09jul2015, with 3 event windows specified, using the Boehmer,
> Musumeci, Poulsen test, with the Kolari and Pynnonen adjustment
SECURITY                          CAAR[-3,3]       CAAR[-3,-1]      CAAR[0,3]
Ptf CARs n 1 (3 securities)         1.71%            -2.34%           4.05%**
CAAR group 1  (3 securities)        1.71%            -2.34%           4.05%***
-------------------------------------------------------------------------------
Ptf CARs n 2 (3 securities)         0.94%            -2.25%           3.19%
CAAR group 2  (3 securities)        0.94%            -2.25%**         3.19%***
-------------------------------------------------------------------------------
Ptf CARs n 3 (3 securities)         4.96%            -0.78%           5.74%**
CAAR group 3  (3 securities)        4.96%***         -0.78%           5.74%***
-------------------------------------------------------------------------------
Ptf CARs n 4 (5 securities)         2.85%            -2.08%           4.93%***
CAAR group 4  (5 securities)        2.85%            -2.08%           4.93%**
-------------------------------------------------------------------------------
*** p-value < .01, ** p-value <.05, * p-value <.1
```

As required, the table reports only the portfolio and group ARs. Furthermore, in this case, ARs are computed according to the HMM specified through the option `modtype()`. Because normal returns are supposed to be equal to the historical averages, the option `indexlist()` (specifying the market returns) is no longer required. The output table now is divided into four panels, reporting the variables specified in each of the four *varlists*.

The results here show that, after we aggregate the firms into four groups, all of those in the $[0, 3]$ window show a positive and significant AR. This means that the market had not anticipated the event and that, once it occurred, the event was judged as positive for all the groups of firms analyzed.

In contrast, should one be interested only in the ARs on each variable, the option `suppress(group)` may be specified.

```
. estudy boa ford boeing(ibm facebook apple) (netflix cocacola amazon)
> (facebook boa ford boeing google), datevar(date) evdate(07092015)
> dateformat(MDY) modtype(MFM) indexlist(mkt smb hml) lb1(-3) ub1(3) lb2(-3)
> ub2(-1) lb3(0) ub3(3) diagnosticsstat(KP) eswlb(-250) eswub(-20)
> suppress(group) showpvalues nostar
Event date: 09jul2015, with 3 event windows specified, using the Boehmer,
> Musumeci, Poulsen test, with the Kolari and Pynnonen adjustment
SECURITY                        CAAR[-3,3]       CAAR[-3,-1]      CAAR[0,3]
Bank of America Corporation      0.25%           -2.03%            2.28%
                                 (0.9181)         (0.1942)         (0.2076)
Ford Motor Company              -1.07%           -1.21%            0.14%
                                 (0.7036)         (0.5100)         (0.9464)
The Boeing Company               3.30%            2.32%            0.98%
                                 (0.2198)         (0.1878)         (0.6293)
-------------------------------------------------------------------------------
IBM Corp                         1.48%            0.48%            1.00%
                                 (0.5680)         (0.7778)         (0.6093)
Facebook Inc                    -0.77%           -1.13%            0.36%
                                 (0.8076)         (0.5830)         (0.8781)
Apple Inc                       -3.93%           -2.31%           -1.63%
                                 (0.1743)         (0.2235)         (0.4575)
-------------------------------------------------------------------------------
Netflix Inc                      3.55%            0.57%            2.98%
                                 (0.6047)         (0.8989)         (0.5654)
The Coca-Cola Company            3.72%            1.64%            2.08%
                                 (0.1030)         (0.2717)         (0.2283)
Amazon com Inc                   3.05%           -1.13%            4.18%
                                 (0.5393)         (0.7284)         (0.2658)
-------------------------------------------------------------------------------
Facebook Inc                    -0.77%           -1.13%            0.36%
                                 (0.8076)         (0.5830)         (0.8781)
Bank of America Corporation      0.25%           -2.03%            2.28%
                                 (0.9181)         (0.1942)         (0.2076)
Ford Motor Company              -1.07%           -1.21%            0.14%
                                 (0.7036)         (0.5100)         (0.9464)
The Boeing Company               3.30%            2.32%            0.98%
                                 (0.2198)         (0.1878)         (0.6293)
Alphabet Inc                     5.30%            0.09%            5.22%
                                 (0.0397)         (0.9593)         (0.0074)
-------------------------------------------------------------------------------

p-values in parentheses
```

Contrary to the previous example, the table now reports only ARs on single securities, hiding those on the portfolio and of the group as a whole. Moreover, the option `showpvalues` prints in parentheses the p-value of each significance test below the ARs that are referred to. Although some ARs are statistically different from zero, asterisks do not appear in the table because of the option `nostar`. ARs are computed according to the Fama and French (1993) three-factors model through the options `modtype(MFM)` and `indexlist(mkt smb hml)`.

4.1 Export

Tables 1 and 2 show what the command exports when the option `outputfile()` is specified, using the same setup of the last example.[5] This option creates, or replaces should

5. We omit the option `suppress()` to obtain the most complete tables.

it already exist, an `.xlsx` file containing two sheets reporting ARs and *p*-values respectively. In both cases, the command stores in the first column the labels of securities on which the event study is performed (that is, those included in each *varlist*), whereas the specified event windows are reported in the first row. In tables 1 and table 2, each cell is filled with the corresponding ARs or *p*-values, respectively.

Table 1. Output results: ARs

	CAAR(-3,3)	CAAR(-3,-1)	CAAR(0,3)
Bank of America Corporation	0.002461072	-0.020349431	0.022810503
Ford Motor Company	-0.010653524	-0.012076545	0.001423022
The Boeing Company	0.033006828	0.023191459	0.00981537
Ptf CARs n 1 (3 securities)	0.008271458	-0.003078174	0.011349632
CAAR group 1 (3 securities)	0.008271459	-0.003078173	0.011349632
IBM Corp	0.014830532	0.004797428	0.010033104
Facebook Inc	-0.007661726	-0.011310653	0.003648928
Apple Inc	-0.039335353	-0.02307519	-0.016260163
Ptf CARs n 2 (3 securities)	-0.010722181	-0.009862804	-0.000859377
CAAR group 2 (3 securities)	-0.010722181	-0.009862805	-0.000859377
Netflix Inc	0.035488194	0.005702812	0.029785382
The Coca-Cola Company	0.037154107	0.016399572	0.020754535
Amazon com Inc	0.030487834	-0.011293763	0.041781597
Ptf CARs n 3 (3 securities)	0.034376711	0.003602873	0.030773838
CAAR group 3 (3 securities)	0.034376712	0.003602874	0.030773839
Facebook Inc	-0.007661726	-0.011310653	0.003648928
Bank of America Corporation	0.002461072	-0.020349431	0.022810503
Ford Motor Company	-0.010653524	-0.012076545	0.001423022
The Boeing Company	0.033006828	0.023191459	0.00981537
Alphabet Inc	0.053035371	0.000861924	0.052173446
Ptf CARs n 4 (5 securities)	0.014037606	-0.003936649	0.017974255
CAAR group 4 (5 securities)	0.014037605	-0.003936649	0.017974254

Table 2. Output results: p-values

	CAAR(-3,3)	CAAR(-3,-1)	CAAR(0,3)
Bank of America Corporation	0.918138037	0.194238139	0.207601057
Ford Motor Company	0.703600253	0.510025195	0.946399969
The Boeing Company	0.219788137	0.187834664	0.62929171
Ptf CARs n 1 (3 securities)	0.57873565	0.752285556	0.313529052
CAAR group 1 (3 securities)	0.494485035	0.780551056	0.076034267
IBM Corp	0.567972685	0.777815324	0.609315581
Facebook Inc	0.807642134	0.582987469	0.878089558
Apple Inc	0.174282537	0.223452377	0.457528887
Ptf CARs n 2 (3 securities)	0.544844156	0.394885695	0.948810763
CAAR group 2 (3 securities)	0.57381597	0.296453634	0.948809709
Netflix Inc	0.604685451	0.898883328	0.56544604
The Coca-Cola Company	0.103024088	0.271655934	0.228285386
Amazon com Inc	0.539348007	0.728352713	0.265808967
Ptf CARs n 3 (3 securities)	0.260282196	0.856979172	0.182493235
CAAR group 3 (3 securities)	0.013105524	0.509152872	2.51249E-06
Facebook Inc	0.807642134	0.582987469	0.878089558
Bank of America Corporation	0.918138037	0.194238139	0.207601057
Ford Motor Company	0.703600253	0.510025195	0.946399969
The Boeing Company	0.219788137	0.187834664	0.62929171
Alphabet Inc	0.039683063	0.959272814	0.007429037
Ptf CARs n 4 (5 securities)	0.257950124	0.627962505	0.055342784
CAAR group 4 (5 securities)	0.247817905	0.612124739	0.059886057

5 Acknowledgments

We would like to thank Joe Newton (the editor) and an anonymous referee for their valuable contributions. We are also grateful to Elisa Borghi, Giuseppe Risalvato, Bojan Srbinoski, Luca Corazzini, and Massimiliano Serati for their helpful comments. We finally thank all the users whose feedback has substantially improved the quality of our research.

6 References

Andres, C., A. Betzer, and M. Doumet. 2016. Measuring abnormal credit default swap spreads. Working Paper. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2194320.

Bessembinder, H., K. M. Kahle, W. F. Maxwell, and D. Xu. 2009. Measuring abnormal bond performance. *Review of Financial Studies* 22: 4219–4258.

Binder, J. 1998. The event study methodology since 1969. *Review of Quantitative Finance and Accounting* 11: 111–137.

Boehmer, E., J. Masumeci, and A. B. Poulsen. 1991. Event-study methodology under conditions of event-induced variance. *Journal of Financial Economics* 30: 253–272.

Brown, S. J., and J. B. Warner. 1985. Using daily stock returns: The case of event studies. *Journal of Financial Economics* 14: 3–31.

Fama, E. F., and K. R. French. 1993. Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics* 33: 3–56.

Kaspereit, T. 2015. eventstudy2: Stata module to perform event studies with complex test statistics. Statistical Software Components S458086, Department of Economics, Boston College. https://ideas.repec.org/c/boc/bocode/s458086.html.

Kolari, J. W., and S. Pynnönen. 2010. Event study testing with cross-sectional correlation of abnormal returns. *Review of Financial Studies* 23: 3996–4025.

———. 2011. Nonparametric rank tests for event studies. *Journal of Empirical Finance* 18: 953–971.

Kothari, S. P., and J. B. Warner. 2007. Econometrics of event studies. In *Handbook of Empirical Corporate Finance, vol. 1*, ed. B. E. Eckbo, 3–36. Amsterdam: Elsevier.

MacKinlay, A. C. 1997. Event studies in economics and finance. *Journal of Economic Literature* 35: 13–39.

Patell, J. M. 1976. Corporate forecasts of earnings per share and stock price behavior: Empirical test. *Journal of Accounting Research* 14: 246–276.

Sorokina, N., D. E. Booth, and J. H. Thornton, Jr. 2013. Robust methods in event studies: Empirical evidence and theoretical implications. *Journal of Data Science* 11: 575–606.

Wilcoxon, F. 1945. Individual comparisons by ranking methods. *Biometrics Bulletin* 1: 80–83.

Zhang, X., C. Li, and X. Xu. 2013. eventstudy: Stata module to perform event studies in finance. Statistical Software Components S457615, Department of Economics, Boston College. https://econpapers.repec.org/software/bocbocode/s457615.htm.

About the authors

Fausto Pacicco has a PhD in management, finance, and legal disciplines for integrated company management from LIUC–Università Carlo Cattaneo. He is a researcher for the Centre on the Development of Territories and Sectors, LIUC–Università Carlo Cattaneo; his research interests are econometrics and economic indicators. He teaches public economics, economic policy, and time-series analysis at LIUC–Università Carlo Cattaneo.

Luigi Vena has a PhD in management, finance, and legal disciplines for integrated company management from LIUC—Università Carlo Cattaneo. He is a researcher for LIUC–Università Carlo Cattaneo; his research interest is financial intermediation and corporate governance. He teaches financial intermediation, corporate governance, and international financial markets at LIUC–Università Carlo Cattaneo.

Andrea Venegoni has a PhD in management, finance, and legal disciplines for integrated company management from LIUC–Università Carlo Cattaneo. He is a researcher for the Centre on the Development of Territories and Sectors, LIUC–Università Carlo Cattaneo; his research interest is the analysis of business sectors and macroeconomics. He teaches macroeconomics, economics, and cross-section analysis at LIUC–Università Carlo Cattaneo.