University of California
Los Angeles

Volatility at High Frequency

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Economics

by

Duke Whang

2012

Abstract of the Dissertation

Volatility at High Frequency

by

Duke Whang

Doctor of Philosophy in Economics

University of California, Los Angeles, 2012

Professor Bryan Ellickson, Chair

The availability of software tools, high frequency data, and recent advances in statistical inference all allow a greater study of continuous-time models of price and volatility processes.

This research studies the structure of intraday stock volatility over a selected group of stocks from 2007 to 2011. We use nearly every valid transaction in the Trades and Quotes database to obtain a price series which is sampled every second. We calculate realized variation (RV), the sum of squared log returns, to estimate squared volatility.

We partition the trading day at the level of 100-second time intervals, and we observe mean reversion in RV even at this time scale. We estimate a modified Heston model for RV in which statistical criteria are used to detect volatility jumps.

The dissertation of Duke Whang is approved.

Walter Torous

Hanno Lustig

Joseph Ostroy

Bryan Ellickson, Committee Chair

University of California, Los Angeles

2012

Dedicated to my parents, my brother, and sister.

Dedicated to **Jesus Christ**, Who Redeems and shows Grace and Mercy.

Table of Contents

List of Figures

List of Tables

Acknowledgments

First and foremost, I express my deepest gratitude to my advisor Bryan Ellickson. His dedication, commitment, and encouragement made this thesis possible.

I thank my fellow research group members Benjamin Thomas Hood and Tin Shing Liu. The collaboration between all of us contributed greatly to this research project.

I thank professors John Mamer and Stephen Lippman. I have had many conversations with them regarding many topics, and they have provided support.

I am grateful to my many friends at the Graduate Christian Fellowship here at UCLA. Among them are Stephen Hurley, Sathish Manickam, Amy Pojar, and Joy Trujillo. (There are also many others). They prayed, counseled, and encouraged me for many years.

My parents, my brother Andrew, and my sister Cherri have indirectly contributed immeasurably. I am grateful.

Finally, I thank God for ultimately providing every blessing in my life.

Vita

1992	B.S. (Mathematics) and B.S. (Economics), California Institute of Technology.
1994	M.S. (Computer Science), University of Chicago
2005	M.S. (Business Administration), University of California, Los Angeles
2007	M.A. (Economics), University of California, Los Angeles

CHAPTER 1

Introduction

Over the past fifty years finance has been revolutionized by the application of the tools of continuous-time stochastic processes to financial markets. Starting with the work of Fischer Black, Myron Scholes and Robert Merton in the 1970s, asset pricing models within a continuous-time framework have been applied to a wide variety of assets including stocks, bonds, derivatives, forward contracts, futures contracts and much more. During the 1980s the theoretical basis of arbitrage theory in continuous time was extended by David Kreps, Michael Harrison, Stanley Pliska and others to the very general class of semimartingales, a class that allows for price processes with jumps as well as the continuous-path processes (Itô processes) that were the focus of the earlier work by Black, Scholes and Merton.

Unfortunately, empirical testing of the continuous-time theory has not progressed as rapidly as the theory. In the early 1980s, Robert Merton ([Mer80]) advocated the use of high-frequency data as a way to estimate the instantaneous variances and covariances that are crucial to the asset-pricing models he and others had developed. The key insight is that, with high-frequency data, it is "in principle" possible to estimate second-order statistics (quadratic variation and quadratic covariation) with great accuracy by subdividing time periods such as a day into finer and finer subperiods (e.g., one-second intervals).

Starting in the 1990s high-frequency data for stocks became available to academic researchers. Financial econometricians followed Merton's lead, estimating quadratic variation and quadratic covariation using realized variation and real-

ized covariation. Unfortunately, these estimators turned out to work poorly in practice. Many financial econometricians concluded that asset prices must not be semimartingales. Asset prices, according to this view, can be handled using semimartingale methodology only if the semimartingale is treated as latent. Observed transaction processes are then the sum of the latent semimartingale and a noise term, a conclusion with potentially disturbing conclusions for the theoretic edifice built on those foundations.

Ellickson, Hood, Liu, Whang and Zhou ([EHL12]) [henceforth EHLWZ] challenges this commonly-held view, demonstrating that key parameters of the Heston ([Hes93]) model of stochastic volatility can be estimated quite accurately for the 30 stocks of the Dow Jones Industrial Average over the period 2001–2009 using stock prices sampled once a second and estimating realized variation for every 5-minute interval during the trading day over the entire 9-year period for each of these stocks. The statistical performance of the estimation is spectacular.

The main focus in this thesis is on improvements to the model presented in EHLWZ (2012). This chapter summarizes the conclusions of the earlier paper. The improvements center on the detection of data errors and a three-fold increase in resolution. In Chapter 2 I describe the programming environment that allows us to improve upon the earlier paper. Chapter 3 compares and contrasts the new results with the old.

1.1 Pricing by arbitrage

The theory of pricing by arbitrage takes its most impressive and powerful form in continuous time. The theory has its roots in Samuelson ([Sam65]) and in Merton's ([Mer73]) reformulation of the Black-Scholes ([BS73]) model of derivative pricing. Ross ([Ros76]) was the first to realize the potential of pricing by arbitrage as a general theory of the pricing of financial assets. Harrison and Kreps ([HK79])

developed no-arbitrage theory in discrete time, and Harrison and Pliska ([HP81]) were the first to recognize the relevance of semimartingale methodology for the continuous-time case. Kreps ([Kre81]) pioneered the extension of the theory to continuous time, and Delbaen and Schachermayer ([DS94],[DS98]) brought the theory to a state which many experts regard as perhaps its final form.

Semimartingales are now regarded as the central concept in the modern theory of stochastic integration. In discrete time, every stochastic process is a semimartingale. In continuous time, fractal Brownian motion is an example of a stochastic process that is not a semimartingale. Fortunately, the class of semimartingales is quite broad, encompassing not only geometric Brownian motion but Itô processes with stochastic volatility (such as the Heston ([Hes93]) model) and processes in which price and/or volatility processes jump. Protter ([Pro04]) gives a accessible and quite general treatment of stochastic integration for semimartingales. Delbaen and Schachermayer ([DS98]) is widely regarded as the definitive treatment of the general theory of arbitrage pricing for semimartingales.

Delbaen and Schachermayer ([DS94]) proved, for the class of semimartingales with bounded jumps, the following version of the *Fundamental Theorem of Asset-Pricing Theory*:

Theorem. *If asset prices are semimartingales with bounded jumps and the NFLVR condition holds, then there exists an equivalent martingale measure.*

Their *no-free-lunch-with-vanishing-risk (NFLVR) condition* is a strengthening of the *no-arbitrage (NA) condition* that suffices in discrete time. *Equivalent martingale measures* allow modern financial theory to price assets.

A version of the fundamental theorem also holds for semimartingales with unbounded jumps, provided that the price processes are sigma-martingales ([DS94]).

Theorem. *If asset prices are sigma-martingales and the NFLVR condition holds, then there exists an equivalent martingale measure.*

Delbaen and Schachermayer ([DS05]) reprints the two key articles as well as five other articles. They also provide a very useful 140-page "Guided Tour to Arbitrage Theory."

Itô processes are semimartingales with continuous paths and hence no jumps. Brownian motion is the simplest form of an Itô process. In their pioneering work of the 1970s, Black, Merton and Scholes assumed that stock price processes follow a *geometric Brownian motion*, which means that the log price process is a Brownian motion. If the stock-price process $S = (S_t)_{t\geq 0}$ is a geometric Brownian motion and $X = (X_t)_{t\geq 0}$ where $X_t = \log(S_t)$, then the stochastic process X is the solution a *stochastic differential equation* (SDE) of the form

$$dX_t = \mu dt + \sigma dW_t$$

where W is a Wiener process with drift 0 and volatility 1 (i.e, a *standard Wiener process*). The parameter μ is called the *drift* and the parameter σ is called the *volatility* of the stochastic process X.[1]

1.2 Quadratic variation

Associated with any semimartingale $X = (X_t)_{t\geq 0}$ is another stochastic process $[X, X] = ([X, X]_t)_{t\geq 0}$, called the *quadratic variation* of the process X. Quadratic variation plays a key role in the development of stochastic integrals for semimartingales.[2] In the case of geometric Brownian motion

$$[X, X]_t = \int_0^t \sigma^2 dt = \sigma^2 t$$

Consequently, if X has drift 0, $[X, X]_t$ is the *variance* of X_t at time t.

[1] Shreve ([Shr04]) provides an excellent textbook exposition of stochastic calculus from a finance perspective.

[2] See Protter ([Pro04]) for a general development of stochastic integration for semimartingales.

For a general Itô process with stochastic differential

$$dX_t = \mu_t dt + \sigma_t dW_t \qquad (t \geq 0)$$

where μ and σ are stochastic processes defined on the same filtered probability space as the Wiener process W, the quadratic variation process is given by

$$[X, X]_t = \int_0^t \sigma_s^2 ds \qquad (t \geq 0)$$

More generally, suppose X_t is a jump diffusion with stochastic differential

$$dX_t = \mu_t dt + \sigma_t dW_t + dJ_t \qquad (t \geq 0)$$

where $J = (J_t)_{t \geq 0}$ is a jump process driven by a counting process with finite intensity. Then the quadratic variation of X is given by

$$[X, X]_t = \int_0^t \sigma_s^2 ds + \sum_{s \leq t} (\Delta J_s)^2$$

where $\Delta J_t := J_t - J_{t-}$ is the jump at time t.

1.3 Realized variation

In the mathematical literature on semimartingales, *realized variation* is the natural way to estimate quadratic variation path by path. Let $[0, 1]$ represent the time interval of a trading day. Time begins at the market open $t = 0$ (9:30 AM for the NYSE) and time ends at the market close at $t = 1$ (4:00 PM for the NYSE). A remarkable fact about estimating the increment

$$[X, X]_1 - [X, X]_0$$

to the quadratic variation process over the trading day is that we can increase the precision of the estimate by sampling the process over finer and finer grids. In particular, suppose we specify a grid $\mathcal{G}^N$ that subdivides the interval $[0,1]$ into N subintervals of equal length:[3]

$$\mathcal{G}^N = \{t_0, t_1, \ldots, t_N\}$$

where

$$0 = t_0 < t_1 < \ldots < t_N = 1$$

and $\Delta t_j = 1/N$ for all $j = 1, 2, \ldots, N$. By definition, the *realized variation* over the interval $[0,1]$ is

$$\mathrm{RV}_{[0,1]}^{\mathcal{G}^N} = \sum_{j=1}^{N} (\Delta X_{t_j})^2 \tag{1.1}$$

The *mesh* of the grid $\mathcal{G}^n$ is the maximum spacing between adjacent times in the grid,

$$\mathrm{mesh}(\mathcal{G}^N) := \max\{\Delta t_j : j = 1, 2, \ldots, N\}$$

Because we have specified uniform spacing, in this case $\mathrm{mesh}(\mathcal{G}^n) = 1/N$. As $N \to \infty$ (or, equivalently, the mesh of the grid goes to 0) $\mathrm{RV}_{[0,1]}^{\mathcal{G}^N}$ converges in probability to the increment to quadratic variation over the interval $[0,1]$: i.e.,

$$\mathrm{RV}_{[0,1]}^{\mathcal{G}^N} \xrightarrow{p} [X,X]_1 - [X,X]_0$$

Because we will always assume that our semimartingales have no jump at $t = 0$, it follows that $[X,X]_0 = 0$ and hence

$$\mathrm{RV}_{[0,1]}^{\mathcal{G}^N} \xrightarrow{p} [X,X]_1$$

[3]It is not necessary that the points of the grid be equally spaced. We do so only to simply the exposition.

1.4 Estimating QV within the trading day

High frequency TaQ data on U.S. equity securities are available starting in 1993. As TaQ data became more widely available in the late 1990's, financial econometricians began to explore the potential of using realized variation to estimate quadratic variation over the trading day. However, realized variation performed badly, yielding estimates of volatility inconsistent with estimates of stock price volatility using daily stock returns or from pricing formulas for derivative assets (*implicit volatility*). To explain this failure, Anderson, Bollerslev, Diebold, and Labys ([ABD99]) introduced a graphical device called a *(volatility) signature plot* that suggests realized volatility does not converge uniformly in probability to quadratic variation on compact intervals, as semimartingale theory suggests that it should.[4]

For a given trading day, the signature plot plots an average of realized variation estimates on the vertical axis where the average is taken over a collection of grids with increasing mesh. The horizontal axis is the number K of subgrids with $K = 1$ representing the finest subgrid. For small values of K, the averages should be close, but they are not. From this evidence, Anderson et. al. infer that asset prices are not semimartingales.

Their paper has been very influential, leading most financial econometricians working with high frequency data to abandon the hypothesis that log price processes are semimartingales. Since the semimartingale hypothesis is of crucial importance in dealing with continuous-time stochastic processes, econometricians assume there is a latent process that is a *semimartingale* and that observed prices equal this latent process plus an error:

$$X_t = X_t^* + \varepsilon_t \qquad t \in [0, 1]$$

[4]See Mykland and Zhang ([MZ12]) for a clear and concise description of signature plots.

where X_t^* is the (unobserved) *latent* price and ε_t is noise. The noise is attributed to "market microstructure."

We admit to a distaste for the hypothesis that stock prices are latent. Investors and stock exchanges have a lot invested in reporting transactions prices accurately. As we will see, the TaQ data goes to considerable trouble to identify prices corresponding to trades that were reversed because of errors of various sorts, trades identified by a code that allows researchers to exclude such trades (which we do). As we will also see, when we use realized volatility within the trading day, we obtain very sensible estimates. Signature plots do behave as Anderson et al. say they do. However, we suspect the reason for that, and the reason their realized volatility estimates made little sense, is that they asked the wrong question. asking

- How much realized variation is accumulated over an entire day?

rather than

- How does realized variation behave over the course of a trading day?

1.5 Estimating QV over short intervals

EHLWZ (2012) and this thesis represent a return to the realized variation methodology, applied to intervals within each trading day. The current literature argues that prices should be sampled very infrequently (once every 5 minutes) in order to obtain accurate estimates of $[X, X]_1$ (the quadratic variation accumulated over a trading day) because prices are contaminated by noise. We propose instead to sample prices as frequently as possible (once a second), to estimate quadratic variation for much shorter intervals, and to compute realized variation directly from observed prices.

Because we sample prices once a second, the grid we use to compute realized variation is

$$\mathcal{G}^N = \{t_0, t_1, t_2, \ldots, t_N\}$$

where $N = 23400$ is the number of seconds within a 6.5-hour trading day.[5] The gap between adjacent points of the grid is $\Delta t_j = 1/23400$, the length of a 1-second interval expressed as a fraction of the interval $[0, 1]$.

Thus far this is no different from what financial econometricians did in the late 1990s before realized volatility became discredited as an estimator. However, we now group the 1-second intervals into *blocks*. Let $\mathcal{H} \subset \mathcal{G}$ be the subgrid

$$\mathcal{H} = \{\tau_0, \tau_1, \ldots, \tau_M\}$$

where M is the number of blocks in a trading day. We assume that $\tau_0 = t_0 = 0$, $\tau_M = t_N = 1$, and that the $\tau_i \in \mathcal{H}$ are equally spaced (i.e., $\Delta\tau_i = 1/M$). The times $\tau_j \in \mathcal{H}$ are the boundaries the intervals we call blocks, the terminology used by Mykland and Zhang ([MZ09]).

We define the *quadratic variation accumulated over block i*,

$$\Delta[X, X]_{\tau_i} := [X, X]_{\tau_i} - [X, X]_{\tau_{i-1}}$$

Because $[X, X]_0 = 0$

$$[X, X]_{[0,1]} = \sum_{i=1}^{M} \Delta[X, X]_{\tau_i}$$

The *realized variation over block i* is

$$\mathbf{RV}^{\mathcal{G}^N}_{[\tau_{i-1},\tau_i]} = \sum_{t_j \in \mathcal{G} \cap [\tau_{i-1},\tau_i]} (\Delta X_{t_j})^2 \tag{1.2}$$

Each of the terms $(\Delta X_{t_j})^2$ is a squared one-second log return, and there are N/M

[5] The trading day starts at 9:30:00 AM Eastern time and ends at 4:00:00 PM Eastern time.

terms in the sum.

- In EHLWZ (2012) we set $M = 78$ (the number of 5-minute blocks in a trading day) and $N = 23400$ (the number of seconds in a trading day), so $N/M = 300$ (the length of a 5-minute block expressed in seconds).
- In Chapter 3 of this thesis, we set $M = 234$ (the number of 100-second blocks in a trading day) which implies $N/M = 100$ (the blocks are 100 seconds long).

1.6 The Heston model

In EHLWZ (2012) we estimate a Heston ([Hes93]) model of stochastic volatility for the 30 stocks in the DJIA and for SPY (an exchange-traded fund that tracks the S&P 500) over the period 2001–2009. In this section, I will briefly describe the Heston model and the regression equation we derive. This regression equation allows us to use our block estimates of realized variation to estimate the starting value and speed of mean reversion of the Heston model.

Let $\zeta_t := \sigma_t^2$ denote the squared volatility. The Heston model assumes that

$$\begin{aligned} dX_t &= \mu_t dt + \sqrt{\zeta_t} dW_t \\ d\zeta_t &= \kappa(\bar{\zeta} - \zeta_t) + \gamma\sqrt{\zeta_t} dB_t \end{aligned}$$

where X_t is the log stock price, W_t and B_t are Wiener processes (possibly correlated), $\kappa > 0$ is the speed of mean reversion to the asymptotic mean $\bar{\zeta}$, and $\gamma > 0$ is the volatility of volatility.[6] In EHLWZ (2012) we let $M = 78$ and estimate RV for each 5-minute interval for each of our assets over the nine year period from 2001

[6]We require $\gamma^2 < 2\kappa\bar{\zeta}$, which guarantees that (with probability 1) the ζ process remains positive for all t. The process ζ characterized by the Heston model is also called a CIR process because of its use in the Cox, Ingersoll and Ross ([CIR85]) model of the short rate. The stochastic process was first introduced into the mathematics literature by Feller ([Fel51]).

to 2009.

For each trading day in our sample, the price process X and the volatility process ζ are assumed adapted to a filtered probability space $(\Omega, \mathcal{F}, \mathbb{F}, \mathbb{P})$ where $\mathbb{F} = (\mathcal{F}_t)_{t\in[0,1]}$ denotes the filtration, a collection of increasing σ-algebras of the probability space $(\Omega, \mathcal{F}, \mathbb{P})$. Because we assume that ζ_0, κ and γ are measurable with respect to the σ-algebra $\mathcal{F}_0$ at the beginning of the trading day, they act as parameters (i.e., constants) during the course of the trading day. However, we do not assume that $\mathcal{F}_0 = \{\emptyset, \Omega\}$, the trivial σ-algebra. Consequently, these "parameters" are free to vary randomly from one day to the next.

Using the framework of Mykland and Zhang ([MZ09]) for justification, we approximate the stochastic process $[X, X] = [X, X]_{t\in[0,1]}$ as a discrete-time process on $\mathcal{H}$ (the partition points for the blocks). We derive from the Heston model a difference equation on the partition $\mathcal{H}$. Let $h = 1/M$ denote the length of a block. The SDE for ζ_t implies[7]

$$\zeta_{\tau_i} = \bar{\zeta} + e^{-\kappa h}(\zeta_{\tau_{i-1}} - \bar{\zeta}) + \gamma e^{-\kappa h} \int_{\tau_{i-1}}^{\tau_i} e^{\kappa u} \sqrt{\zeta_u}\, dB_u$$

and hence

$$\zeta_{\tau_i} - \zeta_{\tau_{i-1}} = (1 - e^{-\kappa h})(\bar{\zeta} - \zeta_{\tau_{i-1}}) + \gamma e^{-\kappa h} \int_{\tau_{i-1}}^{\tau_i} e^{\kappa u} \sqrt{\zeta_u}\, dB_u$$

Since the Wiener process B is a martingale, the expectation conditioned on $\mathcal{F}_{\tau_{i-1}}$ of the stochastic integral appearing on the right-hand side is equal to zero, and hence

$$\mathbb{E}[\zeta_{\tau_i} - \zeta_{\tau_{i-1}} \mid \mathcal{F}_{\tau_{i-1}}] = (1 - e^{-\kappa h})(\bar{\zeta} - \zeta_{\tau_{i-1}})$$

[7] Shreve [Shr04] presents a derivation on page 152.

This leads us to consider the linear regression

$$\zeta_{\tau_i} - \zeta_{\tau_{i-1}} = \beta(\bar{\zeta} - \zeta_{\tau_{i-1}}) + \varepsilon_{\tau_{i-1}} \tag{1.3}$$

where

$$\beta = 1 - e^{-\kappa h} \in (0,1)$$
$$\varepsilon_{\tau_{i-1}} = \gamma e^{-\kappa h} \int_{\tau_{i-1}}^{\tau_i} e^{\kappa u} \sqrt{\zeta_u}\, dB_u$$

We use β rather than κ as our measure of the the speed of reversion of the process ζ toward its asymptotic mean $\bar{\zeta}$.

We estimate this difference equation by replacing the increments to quadratic variation over blocks i and $i-1$ by their corresponding realized variation estimates. We estimate $\bar{\zeta}$ as the median of the ζ_i's in the *settled region* from 10:40:00 AM to 3:15:00 PM. The asymptotic mean $\bar{\zeta}$ is estimated separately for each trading day. As in Mykland and Zhang ([MZ09]), we normalize the estimates to a rate per trading day by dividing the increment to realized variation for each block by the length of the block. The realized variation estimate for block i is then

$$\hat{\zeta}_{\tau_i} = \frac{\mathbf{RV}^{\mathcal{G}^N}_{[\tau_{i-1},\tau_i]}}{\tau_i - \tau_{i-1}} = M\left(\mathbf{RV}^{\mathcal{G}^N}_{[\tau_{i-1},\tau_i]}\right)$$

In EHLWZ (2012) we demonstrate that for most trading days the volatility estimate $\hat{\zeta}_0$ for the first block is much higher than the median volatility estimate in the *settled region*, which we define to be the period from 10:40:00 AM to 3:15:00 PM. We interpret the high initial value to the build-up of uncertainty about the price of the asset when markets are closed. Although the Heston process is stationary and ergodic, most days it starts out of equilibrium. For that reason, the mean of the estimates $\hat{\zeta}_i$ over the entire trading day typically overstates the asymptotic mean. Consequently, we use the *median* of the realized variation estimates for the

55 five-minute blocks in the settled region as the estimate of the asymptotic mean $\bar{\zeta}$ of the process ζ. *We emphasize that, contrary to the existing literature, we allow this asymptotic mean to vary randomly from day to day.*

1.7 Estimating the Heston model using 300-second blocks

In ELHWZ (2012) we estimate mean reversion for the 30 stocks in the Dow Jones Industrial Average and for SPY, an exchange-traded fund that tracks the S&P 500. We estimate equation (1.3) separately for each of our 31 assets for each year in our sample period 2001–2009. Table 1.1 lists the ticker symbols for each of the stocks in our sample, which consist of the components of the DJIA in 2007.

Because we have 31 assets and 9 years of data, we ran 276 separate regressions. Since there are approximately 250 trading days in a year and 77 pairs of adjacent 5-minute blocks in a day, a typical regression has 19250 ($= 250 * 77$) observations. Those regressions perform quite well. However, as demonstrated in EHLWZ (2012), the performance of the regressions improves substantially by using the Heston model to infer the presence of a jump component driving the volatility process in addition to the Wiener component. On average our *volatility jump filter* detects around 10 blocks per trading day that contain a volatility jump. When those

Table 1.1: Ticker symbols for the DJIA stocks used in EHLWZ (2012)

AA	Alcoa	AIG	AIG	AXP	AmExpress
BA	Boeing	BAC	BankAm	C	Citigroup
CAT	Caterpillar	CVX	Chevron	DD	DuPont
DIS	Disney	GE	Gen Elec	GM	GenMotors
HD	Home Depot	HPQ	Hewlett Pk	IBM	IBM
INTC	Intel	JNJ	JohnsJohns	JPM	JPMorgChase
KO	CocaCola	MCD	McDonalds	MMM	3M
MRK	Merck	MSFT	Microsoft	PFE	Pfizer
PG	ProctGamb	T	AT&T	UTX	UnitedTech
VZ	Verizon	WMT	Walmart	XOM	ExxonMobil

blocks are excluded from the regression, as our specification dictates they should be, the precision of the estimates of mean reversion improves substantially and the coefficients vary much less over time and across assets. When the blocks containing volatility jumps are excluded, the typical annual regression for an asset has around 16250 observations. The filtering procedure to detect volatility jumps is discussed in detail in EHLWZ (2012) and more briefly in Section 2.5 of the next chapter.

Table 1.2 reports the OLS regression estimate of the mean-reversion parameter β of the Heston ([Hes93]) model of the quadratic variation process for each of our assets, pooling the five-minute realized variation estimates for all of the trading days in a year. Table 1.3 reports the corresponding t-statistic for each mean-reversion parameter estimate. In the regression, the change in slope from one interval to the next is the dependent variable and the gap between $\bar{\zeta}$ and the slope estimate in the first of the pair of intervals is the independent variable. For example, for AT&T in 2001, $\beta = 0.77$: 77% of the gap is eliminated in 5 minutes. The regression is run without a constant term, and the median slope in the *settled region* (blocks 15 though 69) is used to estimate the asymptotic mean of the Heston process. The final row gives the column medians.

When we pool the observations for trading days into an annual sample, we do **not** treat the final block of one trading day as "adjacent" to the initial block of the following trading day. The parameters of the Heston model within a trading day are treated as measurable with respect to the initial information set $\mathcal{F}_0$ for that trading day, but the initial information set is **not** the trivial σ-algebra $\{\varnothing, \Omega\}$. Consequently, parameters such as the mean reversion parameter β, the asymptotic mean $\bar{\zeta}$ or the value ζ_0 can vary randomly from one day to the next. Our results suggest that β may be nearly constant across days, but $\bar{\zeta}$ and ζ_0 clearly do vary across assets and over time.

The results almost speak for themselves. There are 279 separate estimates of

Table 1.2: Estimates of mean reversion: 300-second blocks

	2001	2002	2003	2004	2005	2006	2007	2008	2009
AA	0.87	0.87	0.82	0.92	0.66	0.69	0.70	0.90	0.88
AIG	0.73	0.83	0.83	0.37	0.64	0.72	0.93	0.99	0.60
AXP	0.82	0.80	0.85	0.88	0.76	0.88	0.63	0.98	1.00
BA	0.89	0.79	0.77	0.72	0.71	0.67	0.97	0.84	0.75
BAC	0.88	0.80	0.82	0.87	0.82	0.87	0.83	0.99	1.00
C	0.73	0.84	0.84	0.88	0.80	0.87	0.91	0.99	0.94
CAT	0.85	0.88	0.78	0.65	0.83	0.79	0.91	0.95	0.74
CVX	0.84	0.79	0.91	0.81	0.97	0.80	0.71	0.65	0.79
DD	0.84	0.83	0.99	0.82	0.73	0.71	0.76	0.80	0.84
DIS	0.77	0.81	0.78	0.78	0.81	0.77	0.65	0.80	0.81
GE	0.62	0.71	0.96	0.97	0.86	0.98	0.90	0.99	0.98
GM	0.86	0.91	0.87	0.88	0.74	0.73	0.95	0.86	0.75
HD	0.75	0.82	0.73	0.71	0.75	0.72	0.93	0.86	0.92
HPQ	0.71	0.81	0.78	0.75	0.77	0.79	0.73	0.86	0.78
IBM	0.74	0.78	0.76	0.75	0.69	0.77	0.70	0.82	0.96
INTC	0.80	0.87	0.91	0.93	0.83	0.83	0.80	0.91	0.84
JNJ	0.83	0.63	0.79	0.93	0.81	0.82	0.73	0.84	0.85
JPM	0.77	0.78	0.76	0.79	0.78	0.89	0.83	0.65	0.95
KO	0.86	0.96	0.91	0.79	0.78	0.89	0.87	0.75	0.85
MCD	0.80	0.77	0.75	0.89	0.83	0.79	0.70	0.84	0.99
MMM	0.90	0.82	0.77	0.88	0.72	0.79	0.76	0.66	0.77
MRK	0.77	0.85	0.84	0.83	0.70	0.88	0.77	0.79	0.90
MSFT	0.95	0.90	0.92	0.94	0.91	0.88	0.84	0.92	0.77
PFE	0.74	0.87	0.94	0.78	0.72	0.86	0.53	0.93	0.94
PG	0.77	0.82	0.82	0.85	0.89	0.83	0.62	0.73	0.98
T	0.77	0.91	0.82	0.92	0.85	0.81	0.81	0.92	0.88
UTX	0.80	0.82	0.80	0.75	0.72	0.66	0.75	0.84	0.82
VZ	0.80	0.83	0.91	0.81	0.75	0.78	0.73	0.85	0.95
WMT	0.97	0.84	0.87	0.77	0.89	0.80	0.81	0.73	0.89
XOM	0.81	0.79	0.76	0.91	0.83	0.89	0.72	0.87	0.98
SPY	0.76	0.60	0.68	0.81	0.71	0.90	0.88	0.68	0.98
Median	0.80	0.82	0.82	0.82	0.78	0.80	0.77	0.85	0.88

Table 1.3: t-statistics for the mean reversion estimates: 300-second blocks

	2001	2002	2003	2004	2005	2006	2007	2008	2009
AA	316.0	317.8	388.9	793.8	341.7	303.5	228.4	863.1	573.5
AIG	217.8	300.8	370.6	102.2	256.5	322.1	623.0	2573.3	117.0
AXP	266.2	244.5	365.7	382.4	295.0	514.5	205.9	1553.7	3324.5
BA	365.0	237.3	313.7	334.0	314.5	300.4	1118.7	507.0	440.3
BAC	291.9	266.5	393.4	461.4	412.0	498.9	491.8	1278.2	2891.1
C	181.1	306.5	364.6	452.1	346.6	454.4	781.6	1829.4	451.4
CAT	275.5	373.7	309.1	257.5	398.5	389.6	777.9	2013.6	436.0
CVX	286.7	200.1	516.1	383.3	920.7	347.6	231.6	265.3	360.2
DD	270.2	258.8	1558.0	351.4	286.3	293.7	337.0	497.7	541.3
DIS	316.9	301.3	328.7	336.3	428.2	373.3	360.5	422.8	476.2
GE	222.4	222.9	794.8	941.1	416.7	1089.9	686.0	2352.3	1227.0
GM	311.8	413.4	470.4	630.2	379.8	345.3	680.4	517.3	196.4
HD	277.9	307.9	347.2	299.5	339.0	342.7	694.2	544.2	665.5
HPQ	358.3	341.9	361.4	521.5	356.9	357.3	219.3	529.5	464.0
IBM	187.5	260.9	298.8	305.1	348.6	370.8	216.0	439.8	880.6
INTC	267.7	436.6	637.7	770.4	380.2	350.2	345.8	600.9	397.9
JNJ	258.4	211.2	327.5	604.1	357.7	427.7	331.4	471.4	575.5
JPM	225.4	283.5	316.7	353.5	303.8	501.5	368.1	319.8	776.2
KO	266.5	613.1	585.2	586.5	329.7	507.6	609.3	627.8	550.3
MCD	252.8	324.8	273.6	586.9	473.6	385.2	351.4	510.6	1568.8
MMM	334.9	231.4	265.5	533.3	416.1	343.8	384.5	408.0	436.4
MRK	204.9	293.8	419.7	489.1	218.2	563.7	656.7	534.2	728.8
MSFT	853.2	597.3	710.1	696.6	474.8	398.0	456.5	580.7	345.8
PFE	233.6	353.1	662.1	507.6	408.3	447.3	218.9	715.9	631.6
PG	199.0	239.4	283.1	636.0	681.5	481.9	315.0	379.0	1361.6
T	278.4	471.0	415.4	663.2	561.0	372.5	647.2	748.5	601.3
UTX	368.5	252.4	281.1	288.3	422.6	312.8	441.6	838.8	466.3
VZ	239.5	282.2	465.2	414.9	310.1	480.9	336.2	527.2	887.7
WMT	752.2	284.2	438.2	309.0	494.9	358.4	628.2	421.6	593.5
XOM	242.3	237.4	263.2	479.5	418.6	549.8	368.4	465.4	1175.5
SPY	575.1	154.3	211.4	240.4	178.8	351.5	1322.7	238.9	986.3
Median	270.2	284.2	365.7	461.4	379.8	373.3	384.5	529.5	575.5

the mean-reversion parameter, one estimate for each combination of an asset and a year. All of the estimates are highly significant. All t statistics are in 3 digits, most well above 200. A few even reach four digits.

These results provide the baseline for the remaining two chapters of this dissertation. Impressive as these results are, they can be improved. I improve upon our results in EHLWZ (2012) in several ways. The most important ways are (1) to decrease the size of the blocks from 300 seconds (5 minutes) to 100 seconds, (2) to handle more appropriately the issue of multiple transactions arriving in a single second, and (3) to deal more effectively with the presence of anomalous prices.

I also change the sample period from 2001–2009 to 2007–2011. Eliminating the first few years is dictated by the increase in resolution. Decreasing the size of the blocks improves resolution from 78 blocks per day to 234 blocks to day, but the number of sampled prices within each block falls. This decrease in the sample size for the block estimates of realized variation has a potentially damaging effect on the precision of our volatility estimates for each block. However, over the period 2001-2011, trading volume has increased dramatically. As we will see, for the period 2007 to 2011 the benefits of increased resolution much more than compensate for the potential loss of precision of the block realized-variation estimator.

1.8 Building a computational environment

Using high-frequency data on stock prices is a major conceptual challenge, similar to the challenges faced in many disciplines as they become engulfed with massive amounts of digitalized data. The challenge is not simply a matter of the cost of data storage and appropriate software, but also the development of new tools and modeling strategies appropriate to these new sources of data. Chapter 2 describes the computational environment we have built, a process that took full advantage

of my extensive background and experience as a computer scientist.

1.9 Improving the estimate of the Heston model

Chapter 3 presents the fruit of this effort to improve resolution, deal more effectively with multiple transactions arriving per second and remove anomalous prices. Taking as its baseline the results in EHLWZ (2012) that I have just reprised, Chapter 3 assesses the impact of the improvements made in this dissertation.

CHAPTER 2

The Computational Environment

Development of appropriate computational environments is rapidly entering the forefront of academic research. The world is awash with a flood of digitalized information. Even in the humanities, access to digitalized manuscripts, articles, books and images allows for an approach to literary, historical and other sorts of humanistic research and discourse that seems qualitatively as well as quantitatively new. Specialists in a narrowly defined fields living in dispersed areas of the globe can set up specialized web sites that function both as repositories of digitalized research material and analysis and as a platform for communication without the need for physical proximity.

Of course, it is the physical and biological sciences where this process has progressed most rapidly. Output from the recently opened CERN hadron collider generates streams of digitalized images of particle collisions that in a few minutes produces much more information than we have available in the TaQ database from 1992 to the present. Molecular biologists must now cope with vast amounts of information generated by the sequencing of the human genome as well as the genomes of many other species. As is beginning to happen in the humanities, much of this new style of academic discourse involves the joint efforts of teams of researchers, and these teams interact cooperately as well as competitively in creating joint systems of shared data and shared tools of analysis.

In our own small way my coauthors and I have been engaged in a similar venture. In this chapter I will provide an overview of the computational environment we have

built, illustrated by specific challenges we faced and how those challenges were met. Fortunately, I brought to this endeavor an extensive academic background and professional experience in computer science. A much more detailed description of the environment is available in my manuscript entitled "**R** programming techniques" ([Wha12]).

Our motivation in constructing this computational environment has been pragmatic, the only way we could cope with what, by the standards of economics or finance, is an enormous amount of data using hardware that is powerful yet affordable and software that allows us to build upon a platform of analytical tools ("software") that has been developing for many years. The tools we have built will, in turn, be shared with others in that research community.

The major software tools we use are the Python *scripting language*, the Linux *operating system*, the **R** *statistical environment* and the TeX *typesetting environment*. All are available as *open source* software, produced and maintained by a community of users and available at no cost. As economists we know, of course, that nothing is really free, and the open source movement recognizes that. Users are required to acknowledge their understanding of the rights and obligations expected of them as users and expected to acknowledge use of the platforms.

In broad outline, here are the major challenges that we have faced and how we have met them:

- *Transferring the TaQ data from the WRDS website to our Linux server and restructuring it in a form suitable for analysis using the **R** statistical package.* The WRDS website provides the TaQ data in a SAS format. SAS is a venerable statistical package that has been around for many decades. It excels in managing databases of many different kinds, but it shows its age when it comes to modern statistical analysis. Until a few years ago, WRDS organized the TaQ data by month: data for all of the transactions

for a given stock for a given month could be downloaded as a single file. Downloading all of the data for say IBM over the period 2001–2009 involved separate requests for 108 files (9 years, 12 files per year). However, the volume of trades became so heavy in recent years that monthly files were too large for SAS to handle. WRDS switched to daily files, which require 250 or so separate "requests" to download the data for a single stock for a single year. They also have been working backward in time to convert all the earlier monthly files to the daily format. Our environment automates the downloading process, using the Python scripting language to send the request for downloads to WRDS and to manage receipt of the data files as they arrive to our server.

Accessing and transforming the TaQ data

The WRDS website provides the TaQ data in a SAS format. SAS is a venerable statistical package that has been around for many decades. It excels in managing databases of many different kinds, but it shows its age when it comes to modern statistical analysis. Until a few years ago, WRDS organized the TaQ data by month: data for all of the transactions for a given stock for a given month could be downloaded as a single file. Downloading all of the data for say IBM over the period 2001–2009 involved separate requests for 108 files (9 years, 12 files per year). However, the volume of trades became so heavy in recent years that monthly files were too large for SAS to handle. WRDS switched to daily files, which require 250 or so separate "requests" to download the data for a single stock for a single year. Eventually WRDS will convert all of the old monthly files to the daily format. Our environment automates the downloading process, using the Python scripting language to send the request for downloads to WRDS and to manage receipt and processing of the data files as they arrive at our server.

Once at our server, we convert the SAS files for each stock for each trading day in our sample to a **R** *data frame*, a flexible data *object* that can then be manipulated in **R**'s *object-oriented* programming environment. These conversion programs recognize the condition codes that SAS assigns to individual transactions, codes that determine whether a transaction should be analyzed or instead has been flagged as suffering from a "condition" that caused the trade to be reversed or canceled. The **R** program selects the transactions suitable for analysis, preserves the data relevant for further analysis (the time of the transaction within the trading day, price, number of shares traded, and the exchange on which the transaction occurred) and creates a data frame containing that data.

Transferring the data from WRDS to UCLA and creating the **R** data frame is by far the most time-consuming step in the analysis, requiring the better part of a day to transfer the daily transactions data for a single stock for the period 2007–2011 and create 1250 or so data frames (one for each of the roughly 250 or so trading days in a year). Recently we upgraded to a 12-core server, which greatly speeds up this process. In Section 2.1 I describe in more detail the procedures uses to transfer the TaQ data from WRDS and restructure it along the lines discussed above.

Reducing the data

The sample period for EHLWZ (2012) is 2001–2009. In the early years of this period, trade volume was relatively low. This thesis focuses on the period 2007–2011, where trade volume is much higher. As a consequence of increased volume, there are many instances when several transactions are recorded at the same second of the trading day. In EHLWZ (2012) we ignored this problem, simply choosing the "last" transaction associated with a given second. However, that means that the data we used in EHLWZ (2012) cannot

be reproduced exactly by other researchers using the TaQ data because the "last" transaction will depend on how they sort the data. In Section 2.2 we address this problem, describing the program we use to *reduce* our daily data frames to data frames that aggregate all transactions for a particular second that trade at the same price. This innocuous step reduces the size of our data frames by an order of magnitude, greatly reducing storage requirements and simplifying data analysis. The data frame of reduced transactions can be replicated exactly by other researchers using the TaQ data. Using the reduced transactions data frame, we then construct a new data frame by using all of the reduced transactions for a given second to construct a data frame with one observation per *active second*.[1] The price associated with each active second is the price of the *median share* in the reduced transaction date frame.[2]

Remarkably, despite the heavy transaction volume in our sample period 2007–2011, the proportion of seconds during a trading day for which there is no transaction is quite high. This surprised us. The impact of high-frequency traders has been the subject of much debate in the press in the past few years. These high-speed trades are actually executed by computer algorithms (not people) co-located within a few meters of an exchange in order to minimize the distance electrons or photons have to travel to send or receive a message from another computer algorithm. News reports claim that 2/3 of all trades on US exchanges are now conducted this way. These algorithms routinely buy and then sell the same shares within 150 *milliseconds*. We suspect this is directly related to what we find in our data: trading volume has become quite concentrated in less than one-half of the 23,400 one-second intervals that make up the trading day, with the remaining intervals having

[1] We refer to seconds for which there is at least one reduced transaction as *active seconds*.

[2] More precisely, the *median-share price* for an active second is the median of the distribution of price per share for that second with each share of stock traded in that second treated as a separate observation.

no transactions. The reduced data sets we construct potentially provide a way to assess the impact of high-frequency trading by humans with reaction times measured on the human scale of seconds rather than the scale of milliseconds or microseconds that is the province of the computer algorithm.

Trimming the data

EHLWZ (2012) does almost no filtering of the price data, eliminating at most 20 or so prices for a stock for a year's worth of transactions data. Nevertheless, the estimates of the Heston model performed quite well. However, this thesis takes a more systematic approach to trimming the data, creating a statistic (which I call the *influence statistic*) that measures the influence (*marginal product*) of an transaction on our estimate of realized variation for the block in which that transaction appears. I use this influence statistic to remove observations in a systematic way, improving the quality of the data. Section 2.3 discusses this procedure.

Estimating intraday volatility

Section 2.4 describes the algorithms used to (1) construct estimates of realized variation for every 300-second or 100-second interval of the trading day for each of our stocks over the period 2007–2011 and (2) assemble these estimates into data frames that can be accessed by our OLS regression routines.

Identifying volatility jumps

In Section 2.5 I describe how we implement the filters we use to distinguish blocks that contain a volatility jump from those that do not. These filters play a crucial role in EHLWZ (2012) and in Chapter 3 of this dissertation.

2.1 Accessing and transforming the TaQ data

In EHLWZ (2012) our analysis focused on the 30 stocks that were components of the Dow Jones Industrial Average in 2007 and an exchange-traded fund (SPY) that tracks the S&P 500. A primary focus of this dissertation is increasing the resolution of our quadratic variation estimates (i.e., decreasing the block size from 300 to 100 seconds). Trading volume increased rapidly from 2001 to 2009, the period analyzed in EHLWZ (2012). Trading volume in the early years is probably too low to yield realized estimates of realized variation for 100-second blocks. While trading volumes vary from one Dow stock to another, for a relatively low-volume stock such as Alcoa 600 transactions a day would be typical in 2001, yielding perhaps 3 transactions on average in each of the 234 100-second blocks of the trading day. For that reason, my thesis starts in 2007.

The improved computational environment discussed in this chapter allows me to cope more effectively with the high trading volumes in later years. For that reason, I extend the analysis two years, yielding an analysis period 2007–2011. Some of the stocks that were in the DJIA in 2007 are no longer there, and other stocks have been added in their place. Table 2.1 lists the stocks in the Dow Industrials that are used in my analysis.

Table 2.1: Ticker Symbols for the Dow Industrials

AA	Alcoa	AXP	AmExpress	BA	Boeing
BAC	BankAm	CAT	Caterpillar	CSCO	Cisco
CVX	Chevron	DD	DuPont	DIS	Disney
GE	Gen Elec	HD	Home Depot	HPQ	Hewlett Packard
IBM	IBM	INTC	Intel	JNJ	JohnsJohns
JPM	JPMorgChase	KFT	Kraft	KO	Coca-Cola
MCD	McDonalds	MMM	3M	MRK	Merck
MSFT	Microsoft	PFE	Pfizer	PG	ProctGamb
T	AT&T	TRV	Traveler's Companies	UTX	UnitedTech
VZ	Verizon	WMT	WalMart	XOM	ExxonMobil

Although `C` (Citigroup) is no longer part of the Dow-30, it is still actively and publicly traded as of the time of this writing. Citigroup was never delisted from any major exchange. It has been publicly traded throughout the period 2007–2011. We will also include it in our analysis. On the other hand, `AIG` and `GM`, included in our earlier analysis, were not always publicly traded in this period, and they are excluded from the analysis.

The TaQ (Trade and Quote) database contains a record of all officially and publicly recorded stock transactions in the United States. More specifically, TaQ contains intraday trades and quotes for securities listed on the New York Stock Exchange, the American Stock Exchange, the Nasdaq national market system, regional exchanges, as well as over-the-counter trades. We use a version of the TaQ database provided by Wharton Research Data Services (WRDS). In the version we use, the time of each transaction (called the time stamp) is recorded to the nearest second. Very recently, WRDS has begun providing a version of the data with times recorded to the nearest millisecond, but that is not yet available to researchers at UCLA, At WRDS, the TaQ data sets are stored as SAS data sets.

The trading day is the central focus of our research. In the **R** statistical environment, the transactions data becomes the fundamental data object, represented in **R** as a *data frame*. Here are some sample lines from the TaQ data for `SPY` on `2011.05.18` in its native SAS format, ordered by the time stamp (the column labeled `TIME`).

	SYMBOL	DATE	TIME	PRICE	SIZE	G127	CORR	COND	EX
1	SPY	18765	14414	133.30	100	0	0		P
2	SPY	18765	14414	133.35	200	0	0		P
3	SPY	18765	14544	133.47	100	0	0	F	P
4	SPY	18765	14544	133.47	100	0	0	F	P
5	SPY	18765	14544	133.62	300	0	0	F	P
5100	SPY	18765	34198	133.26	100	0	0		T
5101	SPY	18765	34198	133.26	100	0	0	F	Z
5102	SPY	18765	34199	133.25	100	0	0		T
5103	SPY	18765	34199	133.25	100	0	0	F	T
5104	SPY	18765	34199	133.24	100	0	0	F	T
5105	SPY	18765	34200	133.24	400	0	0		T
5106	SPY	18765	34200	133.24	200	0	0		T
5107	SPY	18765	34200	133.24	100	0	0		T
5108	SPY	18765	34200	133.24	600	0	0		T
5109	SPY	18765	34200	133.24	400	0	0		T
153433	SPY	18765	45624	133.8636	55000	0	8	@	M
183215	SPY	18765	49137	133.7727	110000	0	12		D
189280	SPY	18765	50003	133.8636	55000	0	10	@	M
318452	SPY	18765	61936	134.0025	1000000	0	12		D

The numbers on the left, the row numbers in the original TaQ data set, have no relevance for us. The first variable, `SYMBOL`, is the ticker symbol of the asset, in this case the exchange traded fund `SPY`. The next four variables describe the date, time, price, and the number of shares of a transaction. They are followed by condition codes and a variable identifying the exchange on which the transaction occurred. More specifically,

- `DATE` is given in the SAS date format which is the number of days since the beginning of 1960. In particular, May 18, 2011 is 18765 days after December 31, 1959.

- `TIME` is the number of seconds after midnight. For example, 9:30:00 AM (the start of the trading day) is 34200 seconds past midnight. From the listing, we can see that on May 18, 2011, 5104 transactions in `SPY` occurred before the market opened at 9:30:00 AM. We consider only trades during the trading day, between 9:30:00 AM (`TIME = 34200`) and 4:00:00 PM (`TIME = 57600`).

- `PRICE` and `SIZE` are the transaction price and the number of shares traded, respectively. If either `PRICE` or `SIZE` is non-positive (which rarely happens), we delete that transaction. Prices are quoted to within a subpenny, one hundredth of a penny.

The next three variables are "condition" codes, flags indicating problems with transactions.

- `G127` is a trade attribute, indicating the reason for a stopped trade. (It is the combined "G", rule 127, and stopped trade indicator). We delete trades stopped for any reason, flagging trades that are replaced in the TaQ data set by a "corrected" trade. (Specifically we delete all trades with a `G127` value not equal to 0). No rows in the listing are affected.

- `CORR` is the correction indicator. We delete trades which have a `CORR` indicator which is strictly greater than 2. In the listing, we filter out the last 4 rows.

- `COND` is the sales condition code. We delete trades which have a `COND` code which is not blank and which is not a member of the set {@, *, E, F}. No rows in the listing are affected.

The final variable describes the exchange code :

- `EX` codes the exchange that executed the trade. In the listing, 5 rows correspond to transactions on `P` (NYSE Arca), 9 on `T` (NASDAQ), 2 on `M`

(Chicago), and 1 on `Z` (BATS). Note that none are on the NYSE (`N`) or on AMEX (`A`)! Over the period 2001-2011 the fraction of trades for DJIA stocks listed on the NYSE that were traded on the NYSE fell from over 90% in 2001 to less than 10% by 2009.

Although many statistical software packages (SAS, EViews, Stata, SPSS) could deal with this data, several features of the data set pose difficulties for most of these environments. Our sampled prices are irregularly spaced: some of the 23401 time stamps that could appear during the trading day do not appear, while other time stamps may have several transactions. *Inactive* time stamps (seconds without transactions) pose problems for techniques such as linear regressions. Linear interpolation can be used to patch this difficulty, but we avoid this problem entirely.

R is an open source statistical programming environment which offers the flexibility to conveniently handle these problems. Many features of **R** facilitate processing the TaQ database. In particular, the *data frame* is a natural representation of TaQ data. It is a two-dimensional matrix with named columns.[3] Each row of the data frame for a trading day represents a transaction and each column represents a transaction attribute such as the execution time, price, volume, or exchange.

The following sample **R** code imposes our initial filtering of the raw TaQ data set. In this code `ct.data` is the name of the data frame that holds the raw TaQ data, the input to the program. The name of the data frame produced by this code is `price.data`. This data frame object contains the transactions not excluded by the condition codes with variables such as time converted from SAS format to a

[3]Rows also have names, which are usually ignored. By default, the row names of a **R** data frame are the strings associated with the positive integers between 1 and the number of rows. ("1","2", ... ,"10" is a typical example of a set of row names). **R** data frames are quite similar to SQL database tables.

more convenient format for our analysis.[4].

```
library(foreign)
ct.data = read.xport("ct.SPY.20110518.xpt")

## 34200 corresponds to  9:30:00
## 57600 corresponds to 16:00:00
min.valid.second = 34200
max.valid.second = 57600
valid.COND.codes = c("","@","*","E","F")

price.data = subset(ct.data,
  (CORR <= 2) & (COND %in% valid.COND.codes) &
  (TIME >= min.valid.second) & (TIME <= max.valid.second) &
  (PRICE > 0) & (SIZE > 0))
```

For the sample of rows listed above, only the rows which have row numbers between 5105 and 5109 remain after this initial filtering.

The filtering program not only implements the filter but also generates aggregate statistics about the processed data. Figure 2.1 plots the median number of shares traded for each year from 2007 to 2011 for our set of stocks and the `SPY` ETF. The entries report trade volume (in millions of shares) for a specific stock in a given year. For example, the median number of shares traded per day in `IBM` during the year 2009 is approximately 6.6 million. Note that the vertical axis is shown on a *logarithmic* scale.

We emphasize three of the financial companies and `SPY` in the graph. Note that

- The median daily volumes in 2011 for `BAC` (148.9 million) and for `SPY` (147.4 million) were almost equal.

[4]The `subset` operator selects all rows of a data frame which satisfy a boolean condition and returns a data frame. This is quite similar to the `select` statement in SQL.

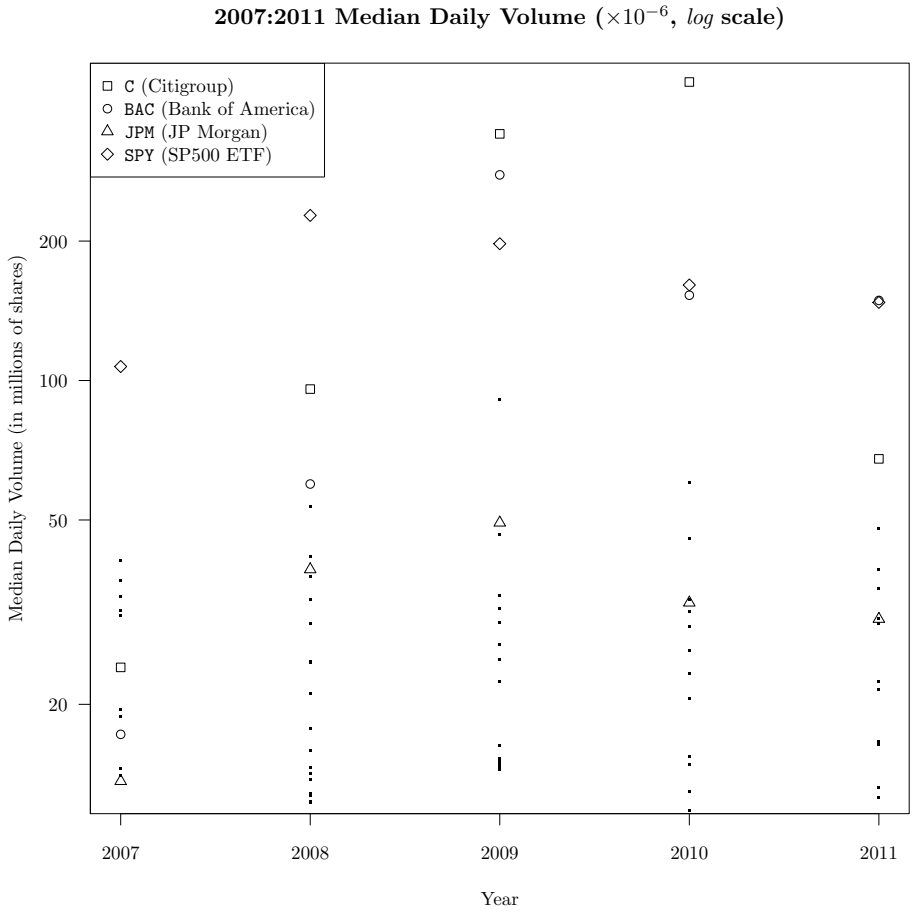

Figure 2.1: *Median Daily Volume in millions of shares.* The vertical axis is plotted on a **log** scale.

- `C` (Citigroup) achieved median volumes of 340.7 million shares in 2009 and 440.9 million shares in 2010.

For most of our companies, median daily volume peaked in 2009 or 2010, most notably for the companies which had the greatest exposure to the financial sector.[5] We have emphasized the median daily trading volume for `BAC`, `C`, `JPM`, and `SPY`. It is remarkable that the median volumes for `BAC` (Bank of America) and `C` were comparable or even exceeded the median volumes for `SPY` in 2009 and 2010. By

[5]The companies most exposed to financial sector risk were `AXP` (American Express), `BAC` (Bank of America), `C` (Citigroup), `GE` (General Electric), `JPM` (JP Morgan), and `TRV` (Traveler's).

2011 SPY was back near the top in median daily volume among our sample of stocks.

2.2 Reducing the Data

In the early years of the sample period 2001–2009 examined in EHLWZ (2012), few time stamps were associated with more than one trade. In the sample period 2007–2009 that is the focus of this dissertation, it is commonplace to find time stamps with more than one trade. In EHLWZ (2012) we reduced our data set to at most one transaction per time stamp by sorting the file by time stamp and selecting the "last" trade for each time stamp.

While this procedure is reasonable, it is not ideal: other researchers will not be able to replicate our results exactly. If they download the WRDS data, do their own sorting by time stamp and select the last trade for each time stamp, they will almost certainly end up with a data set that differs from ours. Our ideal, which we attain in the analysis presented here, is to create derived data sets that can be replicated exactly from the WRDS website (or any other site that provides the TaQ data) using our **R** programs.

This could be accomplished in many ways. The procedure we adopt here is to *reduce* the data frames created by the programs described in Section 2.1 by constructing a data frame with one *reduced* transaction for each active time stamp. We order transactions lexicographically, first by (increasing) TIME of transaction and then by (increasing) PRICE. Then we aggregate all transactions with the same time and price by summing the shares traded. Finally, if a particular time stamp has reduced trades with different prices, we *construct* a single transaction with price equal to the price of the median share for that time stamp. We retain two measures of share volume for that transaction, the total shares traded and the shares traded at the price for the median share.

This is the **R** code that orders transactions lexicographically, first by increasing TIME and then by increasing PRICE:

```
price.data[["NUMBER.TRANSACTIONS"]] = 1

unsorted.reduced.data =
  with(price.data,
       aggregate(cbind(SIZE, NUMBER.TRANSACTIONS)
                         ~ TIME + PRICE, FUN = sum))

sorted.reduced.data = with(unsorted.reduced.data,
    unsorted.reduced.data[order(TIME, PRICE),])
```

The first statement creates a new column NUMBER.TRANSACTIONS, initialized to 1 for all rows. The second statement applies the sum function to aggregate SIZE (the number of shares traded) and NUMBER.TRANSACTIONS for all sets of rows with the same TIME and PRICE.[6]. Finally, the third statement orders the data frame first by increasing TIME and then by increasing PRICE.

We display the first few lines for the reduced data set for SPY on 2011.05.18 when the program is applied to the snippet of WRDS data we used for illustration in Section 2.1 (we show only the most relevant columns):

TIME	PRICE	SIZE	NUMBER.TRANSACTIONS
34200	133.2400	14629	76
34200	133.2450	1200	7
34200	133.2490	200	1
34200	133.2500	48971	117
34200	133.2532	200	1
34200	133.2600	305519	43
34201	133.2400	2195	10
34201	133.2450	300	3

[6]The with operator allows us to refer to the column names of price.data simply by their names. Absent the with operator, we would need to refer to price.data[["SIZE"]] rather than the simpler expression SIZE. The use of cbind ("column bind") allows us to aggregate both SIZE and NUMBER.TRANSACTIONS "simultaneously" in a single statement

In this example, there are 6 distinct prices with the 34200 time stamp. There were 43 transactions at the share price of $133.26 with a total share volume of 305,519.

We next associate a single price with each active time stamp by computing the price of the median share. In the data just exhibited, the median-share price associated with time stamp 34200 is $133.26.[7] Hansen and Lunde ([HL06a]) use the volume-weighted average. They write

> Multiple transaction prices often have the same time stamp. The various transaction prices are all proxies for the (same) latent efficient price at that particular point in time. Although it is unclear how to best handle such observations, a simple and natural estimate of the efficient price (at time t) is the average transaction price (at time t). This was the approach that we took.
>
> Applying an estimator to the raw data, as AMZ [Ait-Sahalia, Mykland, Zhang [AMZ05]] did, may give rise to the problem that the resulting estimate depends on the ordering of the observations with the same time stamp. As AMZ acknowledge, there is no reason to trust the ordering of such observations, and it is not immediately clear to us how sensitive various estimators, including the subsample estimator, are to distorted ordering of the observations. In any case, we argue that the aggregation of data by taking averages of prices with the same time stamp improves the precision of many estimators of IV. ([HL06b])

One advantage to using the median-share price rather than the volume-weighted price is that there is almost always a transaction with the median-share price. An exception will occur only for a time stamp for which there is an equal number

[7] Note that the *median-share price* is **not** the median of the prices for the six reduced transactions.

of shares at or below one price and at or above an adjacent price in the lexicographically ordered data set with the same time stamp, which happens very rarely. On the other hand, if a time stamp in the reduced data set has multiple prices, the volume-weighted average will rarely correspond to the price of an actual transaction (in particular, the price will not be expressed exactly in pennies or subpennies). The median-share price is also a more robust measure of the typical price associated with a time stamp if some prices are extreme outliers.[8]

The following **R** code computes the share-weighted median price for each value of `TIME` with at least one transaction.

```
library(plyr)
library(Hmisc)

weighted.median.price.df =
   ddply(sorted.reduced.data, ~TIME,
         function(df)
           as.double(wtd.quantile(df$PRICE, df$SIZE,
                                  probs = c(0.5))))

names(weighted.median.price.df)[
   names(weighted.median.price.df) == "V1"] =
    "VW_MEDIAN"
```

The data frame containing reduced transactions is called `sorted.reduced.data`. The first two lines of this code load **R** libraries: the `plyr` library provides the `ddply` function, and the `Hmisc` library provides the `wtd.quantile` function. This

[8]Hansen and Lunde ([HL06a]) use transactions only from a single exchange, whereas we use transactions from all the exchanges. For our initial processing, we use all of the TaQ data which passes our filters, regardless of the exchange. Another difference is that Hansen and Lunde later use a bid-ask quote filter for prices. Specifically, they discard any prices which are strictly greater than 1.5 times the bid-ask spread away from the midpoint (the average of the bid and the ask quotes). We use a filter which is based on increments to quadratic variation, which we will discuss in Section 2.3.

is a good illustration of how open source facilitates sharing of tools created by many different researchers.

In the third statement, the `ddply` operator splits the `sorted.reduced.data` data frame into separate data frames by distinct values of `TIME`.[9] The `ddply` operator then applies the `wtd.quantile` function to every such data frame (listed as `df` above). (The values are given by `PRICE`, and the weights used for the median computation is given by `SIZE`). `ddply` stores the value of the weighted median into a column called `V1`. The fourth statement renames this column (`V1`) to `VW_MEDIAN`. Here are some sample lines from our usual example, `SPY` on `2011.05.18`. (We show only the 2 most relevant columns)

TIME	VW_MEDIAN
34200	133.26
34201	133.25
34202	133.25
34203	133.26
34204	133.26

How have the increased trading volumes of recent years affected the number of prices within a second? There are differences between the financial sector stocks (and we include `SPY`) and the rest of our sample of stocks. Figure 2.2 plots the relative frequencies of prices within seconds.

Once again we can add features to our program that report the effect of the code, allowing us to access (for example) how increasing trading volume in recent years has affected the incidence of multiple prices associated with a time stamp. There are differences between the financial sector stocks (and `SPY`) and the rest of our sample of stocks. Figure 2.2 plots the relative frequencies of prices within seconds for all of our stocks except for the three financial stocks and SPY. Remarkably,

[9]There are 23401 time stamps (in our case, seconds) during a trading day. In our example day with `SPY` on 2011.05.18, 18228 time stamps were *active seconds*, time stamps associated with at least one transaction. The `ddply` operator splits the original data frame into 18228 separate data frames, one for each active time stamp.

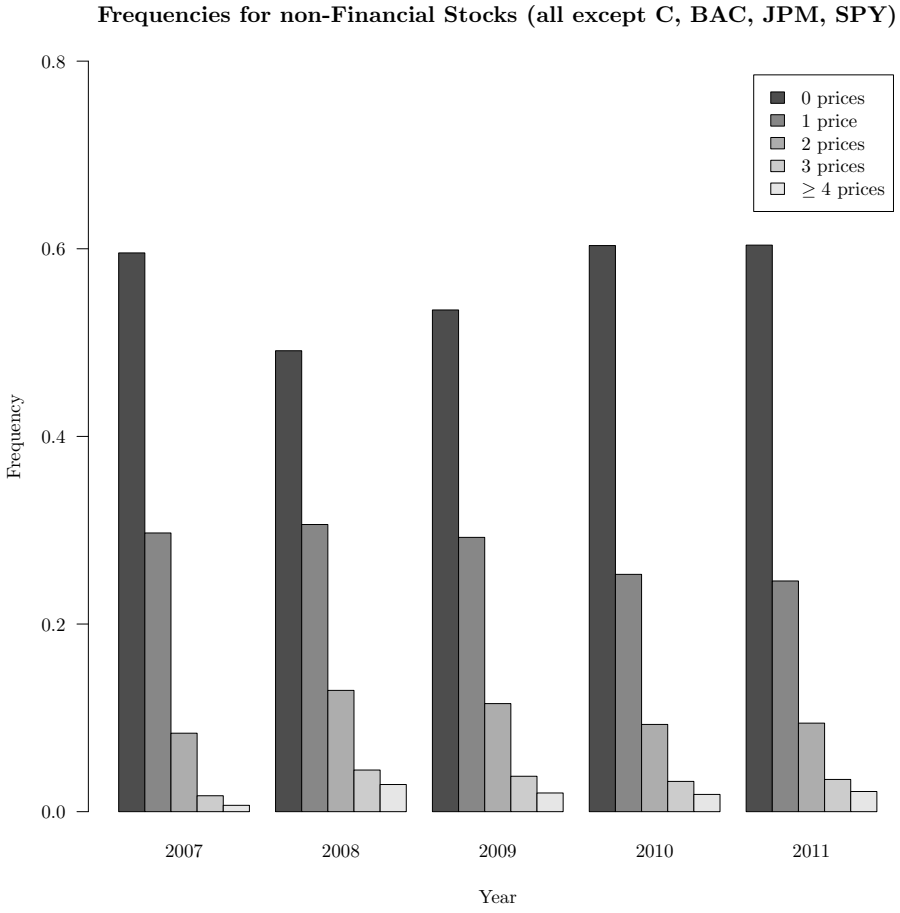

Figure 2.2: Relative Frequency of distinct prices per second (non-financials)

despite trading volumes that typically exceed (by a large margin) one transaction per second on average, approximately 60% of seconds (in 2007, 2010, and 2011) had no transaction at any price: i.e, they were *inactive*. We contrast this to the frequency tables for the financial stocks and `SPY` shown in Figure 2.3 where for most years less than one-third of seconds were inactive. (The fraction of inactive seconds dropped to 0.23 in 2008 but has since increased to nearly 0.30.)

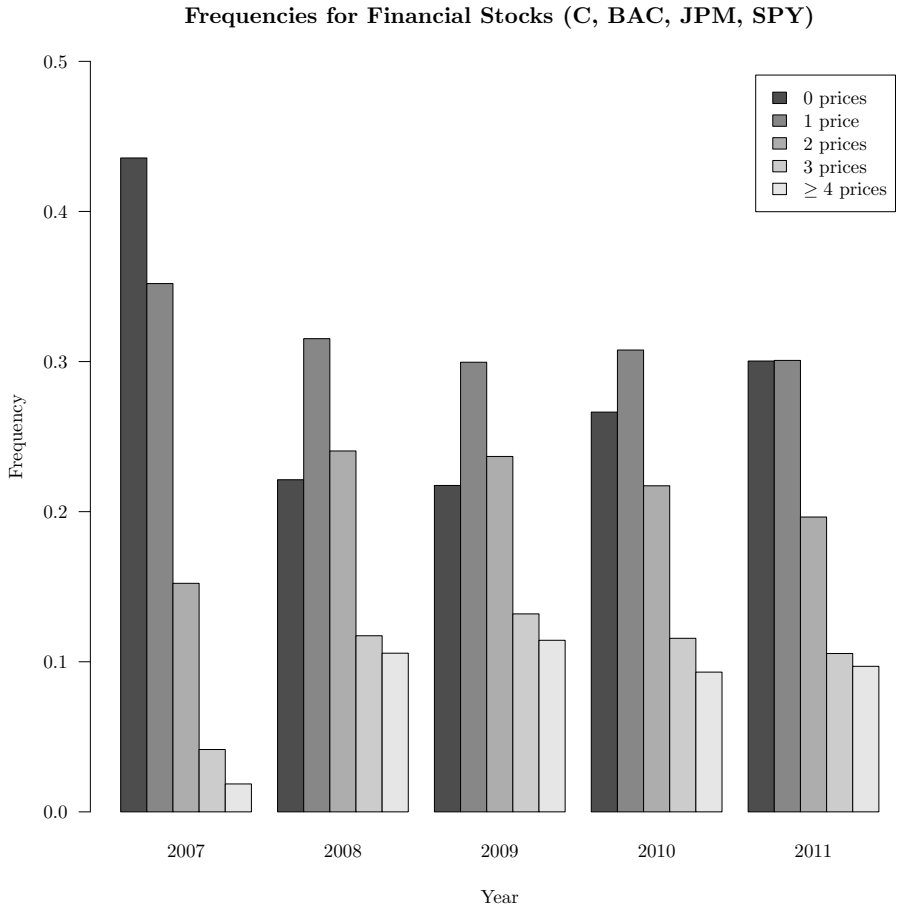

Figure 2.3: Relative Frequency of distinct prices per second for the financial stocks (BAC, C, JPM, and SPY)

Table 2.2: Median Daily Reduced Prices ($\times 10^{-3}$), 2007-2011

	2007	2008	2009	2010	2011
AA	11.80	17.78	16.62	13.63	13.76
AXP	10.34	19.77	17.09	11.92	10.68
BA	10.97	15.69	12.47	10.42	10.21
BAC	14.79	32.69	44.32	34.84	32.10
C	17.99	30.38	33.67	39.33	32.31
CAT	11.05	16.19	18.44	13.74	18.52
CSCO	14.10	15.16	15.10	16.71	21.95
CVX	15.80	25.12	19.12	14.55	15.97
DD	8.00	13.05	11.63	9.70	10.55
DIS	9.22	12.95	11.93	10.83	10.77
GE	16.38	25.27	31.19	24.12	22.54
HD	12.68	16.68	13.78	12.07	10.83
HPQ	12.32	18.41	17.20	14.96	15.26
IBM	12.97	19.58	17.13	13.20	12.97
INTC	13.14	14.25	16.12	18.50	18.70
JNJ	12.38	15.50	16.31	13.34	14.15
JPM	13.65	32.15	30.63	22.40	21.75
KFT	8.77	11.16	10.99	9.73	9.25
KO	9.91	15.05	13.19	11.24	10.92
MCD	8.84	15.83	14.99	10.27	10.98
MMM	7.85	11.66	10.59	8.78	8.63
MRK	11.68	16.71	14.80	12.53	12.70
MSFT	13.60	17.96	17.81	18.66	19.89
PFE	14.37	17.53	19.85	19.31	18.34
PG	11.66	17.28	17.59	12.26	11.75
T	14.32	19.83	18.65	16.89	17.52
TRV	5.41	10.26	11.44	7.60	7.16
UTX	9.13	13.01	11.88	9.02	9.28
WMT	13.19	21.35	18.52	13.07	13.14
VZ	10.75	16.33	16.06	15.19	14.45
XOM	22.49	31.46	24.43	19.07	20.29
SPY	23.46	41.38	40.82	39.10	40.15

Table 2.3: Median Active Seconds ($\times 10^{-3}$), 2007-2011

	2007	2008	2009	2010	2011
AA	9.03	11.94	11.35	9.24	9.10
AXP	7.93	12.08	11.02	8.16	7.31
BA	7.89	9.86	8.39	6.91	6.74
BAC	11.38	17.97	19.86	16.87	16.02
C	13.09	17.78	18.19	19.55	16.66
CAT	8.06	10.42	11.26	8.61	9.90
CSCO	10.98	11.14	10.61	10.57	13.06
CVX	10.92	13.56	11.39	9.28	9.47
DD	6.38	9.33	8.29	6.97	7.36
DIS	7.87	10.00	8.85	7.83	7.55
GE	12.56	16.13	16.85	13.96	13.25
HD	10.04	11.73	9.79	8.60	7.60
HPQ	9.65	12.40	11.44	10.07	10.16
IBM	9.00	11.03	9.90	8.19	7.52
INTC	10.48	10.66	11.07	11.52	11.72
JNJ	9.73	11.11	11.08	9.04	9.40
JPM	10.54	17.01	16.33	13.04	12.71
KFT	7.29	8.82	8.16	7.01	6.64
KO	8.19	10.91	9.42	7.91	7.59
MCD	7.40	10.71	10.08	7.26	7.62
MMM	6.08	8.04	7.18	5.98	5.70
MRK	9.15	11.64	10.32	8.60	8.57
MSFT	10.74	12.52	11.99	11.60	12.03
PFE	11.54	12.76	13.31	12.09	11.66
PG	9.05	11.70	11.41	8.59	8.09
T	11.11	13.59	12.49	10.96	11.14
TRV	4.63	7.31	8.03	5.70	5.26
UTX	7.01	9.01	8.12	6.30	6.30
WMT	10.24	13.69	12.04	9.18	8.86
VZ	8.86	11.51	11.10	9.94	9.53
XOM	14.18	16.25	14.16	11.78	11.89
SPY	15.24	19.36	19.31	18.72	18.89

2.3 Trimming the data

Data reduction yields a collection of median-share prices, one for each *active* time stamp, which we use to compute realized variation. Recall equation 1.2 from Chapter 1:

$$\mathbf{RV}^{\mathcal{G}^N}_{[\tau_{i-1},\tau_i]} = \sum_{t_j \in \mathcal{G} \cap (\tau_{i-1},\tau_i]} (\Delta X_{t_j})^2 \tag{2.1}$$

This is the classic realized variation estimator over block i if prices are sampled uniformly over time. If prices are sampled once a second, it is the sum of squared log returns contained in the interval $[\tau_{i-1}, \tau_i]$. In Chapter 1 we assumed that prices could be observed every second, but noted that realized variation does not require the sampling times to be uniformly spaced. Here we take advantage of that fact, taking the sum over the median-share prices for the active seconds in the block.

Our focus in this section is on prices that have a large effect on our estimate of realized variation for a block. To illustrate, we examine the price and realized-variation processes for SPY on `2011.05.18`. Figure 2.4 plots the median-share prices and the associated realized-variation process constructed from those prices. The value of the realized-variation process at time t is the sum of squared log returns for the median-share price process up to time t.

We can see several instances in which the price process seems erratic, and the realized variation process increases by an unusual amount at the same time. The largest increase to realized variation results from a single median-share price at 10:41:37 AM associated with the time stamp 39487. Figure 2.5 provides a closer look at these processes with a 5-minute local window including that time stamp.

Notice that the realized variation experiences two large jumps in a row. This is the characteristic appearance of realized variation when an outlier price is present. We list a portion of the data frame containing the median-share prices for the active seconds and the reduced transaction date frame in the vicinity of time stamp

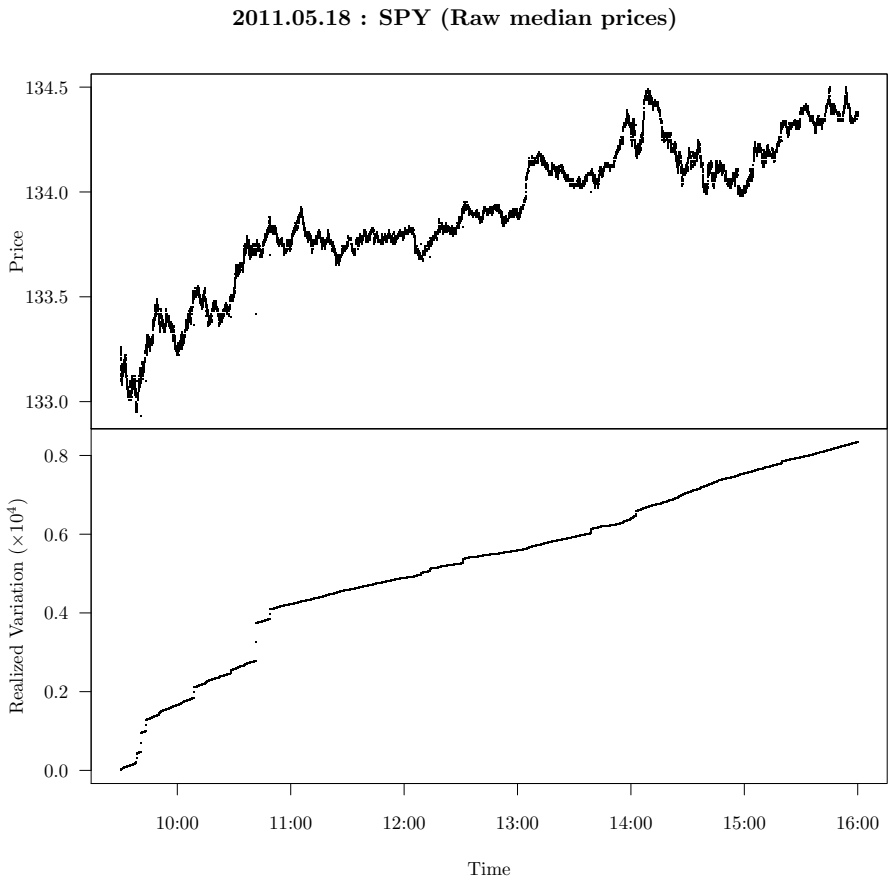

Figure 2.4: Reduced price and RV processes for SPY on 2011.05.18

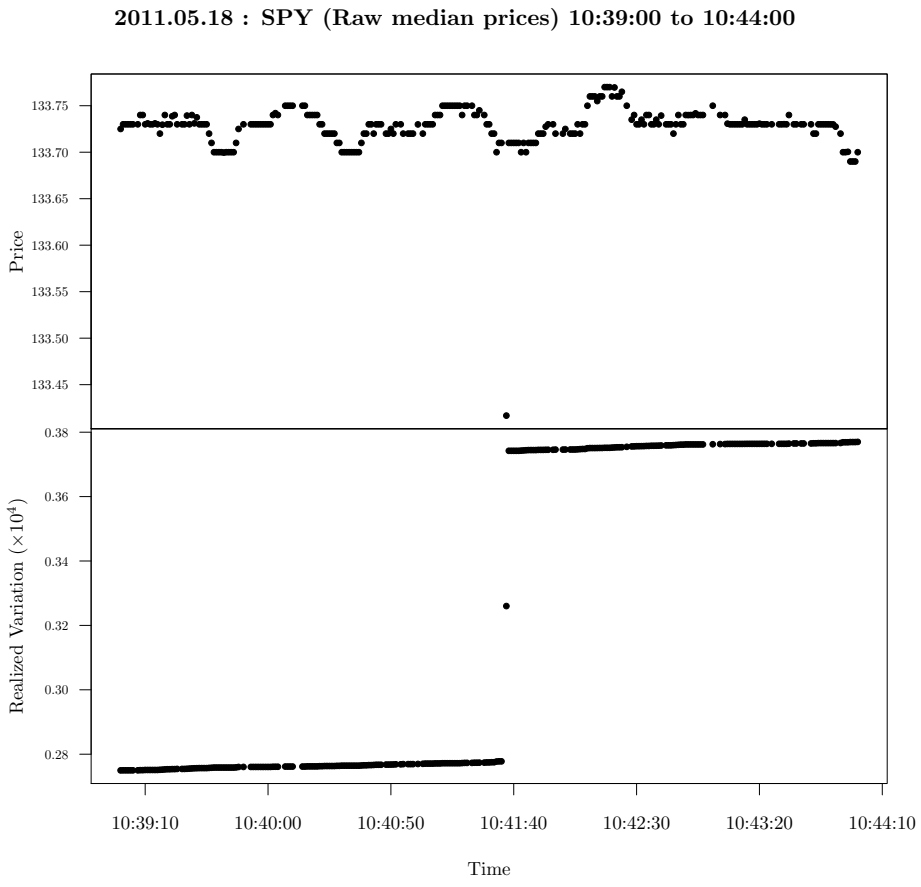

Figure 2.5: A local window for `SPY` on 2011.05.18, 10:39:00–10:44:00

39487:

TIME	Price	RV
38490	133.7300	0.2774447
38491	133.7200	0.2775007
38492	133.7200	0.2775007
38493	133.7000	0.2777244
38494	133.7100	0.2777803
38495	133.7100	0.2777803
38497	133.4167	0.3260029
38498	133.7100	0.3742254
38499	133.7100	0.3742254
38500	133.7100	0.3742254
38501	133.7100	0.3742254

TIME	PRICE	SIZE	NUMBER.TRANSACTIONS
38494	133.7000	500	5
38494	133.7100	9053	40
38494	133.7150	200	2
38494	133.7200	3000	16
38495	133.7100	100	1
38497	133.4167	60000	1
38498	133.7000	19002	23
38498	133.7100	25393	102
38498	133.7150	100	1
38498	133.7200	18400	79
38498	133.7300	2300	16

A single trade of 60000 shares was executed at the price of \$133.4167, while the other prices in the vicinity had prices between \$133.70 and \$133.73. The trade was quite large (60000 shares), so a large discount relative to neighboring prices is not surprising.

We wish to filter out prices such as this which have a very large impact on realized variation.[10] We will create an *influence statistic* for every price observation

[10]There are two possibilities for such transactions. Some are data errors that were not caught

that measures the extra increment to the estimate of realized variation caused by that price. If the influence of a reduced price is too large, we will delete the offending price and compute a new one for that time stamp.

To simplify notation, we temporarily denote X_{t_j} as X_j and the realized volatility estimator for the block $[\tau_{i-1}, \tau_i]$ that contains X_j as $\mathbf{RV}_i^{\mathcal{G}^N}$. We define the *influence statistic* of X_j as

$$\begin{aligned}\text{Influence}_j &:= \mathbf{RV}_i^{\mathcal{G}^N} - \mathbf{RV}_i^{\mathcal{G}^N \setminus \{X_j\}} \\ &= \begin{cases} (X_{j+1} - X_j)^2 + (X_j - X_{j-1})^2 - (X_{j+1} - X_{j-1})^2 & \text{if } t_0 < t_j < t_N; \\ (X_{\tau_j} - X_0)^2, & \text{if } t_j = t_0; \\ (X_N - X_{N-1})^2, & \text{if } t_j = t_N. \end{cases}\end{aligned}$$

This is the marginal contribution of X_j to the realized variation of the block to which X_j belongs.

Consider an *interior* index t_j, $t_0 < t_j < t_N$, and examine Influence_j. To simplify notation still further, let $a := X_{j+1} - X_j$ and $b := X_j - X_{j-1}$. As a consequence, $X_{j+1} - X_{j-1} = (X_{j+1} - X_j) + (X_j - X_{j-1}) = a + b$).

$$\begin{aligned}\text{Influence}_j &= (\underbrace{X_{j+1} - X_j}_{=:a})^2 + (\underbrace{X_j - X_{j-1}}_{=:b})^2 - (\underbrace{X_{j+1} - X_{j-1}}_{=a+b})^2 \\ &= a^2 + b^2 - (a+b)^2 \\ &= -2ab \\ &= -2(X_{j+1} - X_j)(X_j - X_{j-1}) \\ &= -2(\Delta X_{j+1} \Delta X_j) \\ &= -2(X_{t_{j+1}} - X_{t_j})(X_{t_j} - X_{t_{j-1}})\end{aligned}$$

by the TaQ correction codes. Others are *extremely* large trades, which have special prices quite different from the adjacent prices. We will see circumstantial evidence later in this section that our example day of `SPY` on `2011.05.18` has several instances of this phenomenon.

Observe

- This last expression is exactly -2 times the (signed) bipower variation of X_{t_j}.
- The influence is 0 if $X_{t_j} = X_{t_{j-1}}$ or $X_{t_j} = X_{t_{j+1}}$.
- The influence is negative if $X_{t_{j-1}} < X_{t_j} < X_{t_{j+1}}$ or $X_{t_{j+1}} < X_{t_j} < X_{t_{j-1}}$.
- The influence is positive if $X_i < \min\{X_{i-1}, X_{i+1}\}$ or $X_i > \max\{X_{i-1}, X_{i+1}\}$.

Our filter is based on this influence statistic. An outlier price has a large, positive influence. A median-share price is flagged if its influence exceeds 0.20 of the RV of a window of 201 observations centered on X_{t_j} or if its influence exceeds 0.05 of the total RV for the day. If the price is greater than both the immediately preceding median-share price and the median-share price that immediately follows, we delete the maximum reduced-price of that second (and recompute the median share price). We also delete the minimum reduced-price of a time stamp if the median-price is less than the two immediately adjacent prices. (We strive to delete the fewest number of reduced-prices). We iterate this process until we obtain a series of median-share prices that do not fail this test.

Table 2.4 reports the median daily number of prices in the reduced transaction data frame. Even on the most extreme days, the filter removes very few reduced prices. Table 2.5 reports the maximum daily number of removed reduced prices.[11]

For example, on `2011.05.18` (the day we have used for illustrations) 16 prices (corresponding to 18 transactions in the original data frame before reduction) were removed. The average volume of the deleted reduced prices is 182137.5 which is

[11] To place these numbers in context, our stocks typically have several thousand active time stamps each day, and the number of reduced prices exceeds that number (since each active time stamp has at least one reduced price).

much greater than the average volume for every reduced price (including these 16) of 3132.779.

	TIME	PRICE	SIZE	number.original.transactions
1	38497	133.4167	60000	1
2	34842	132.9333	24000	1
3	35012	133.1000	143750	1
4	36538	133.3667	90000	1
5	38955	133.7000	175000	1
6	34705	133.1000	45000	1
7	49146	134.0000	1000000	1
8	50558	134.3200	50000	1
9	45082	133.8327	110000	1
10	37713	133.4048	105000	1
11	44026	133.6900	1200	3
12	43725	133.7500	504000	1
13	35431	133.4400	50000	1
14	55195	134.2441	510000	1
15	36903	133.4100	21250	1
16	37356	133.3700	25000	1

The impact of this filtering process on the RV process can be seen in figure 2.6. The realized variation process appears nearly linear for most of the day, especially during the middle of the day, which we will later designate as the *settled region*.

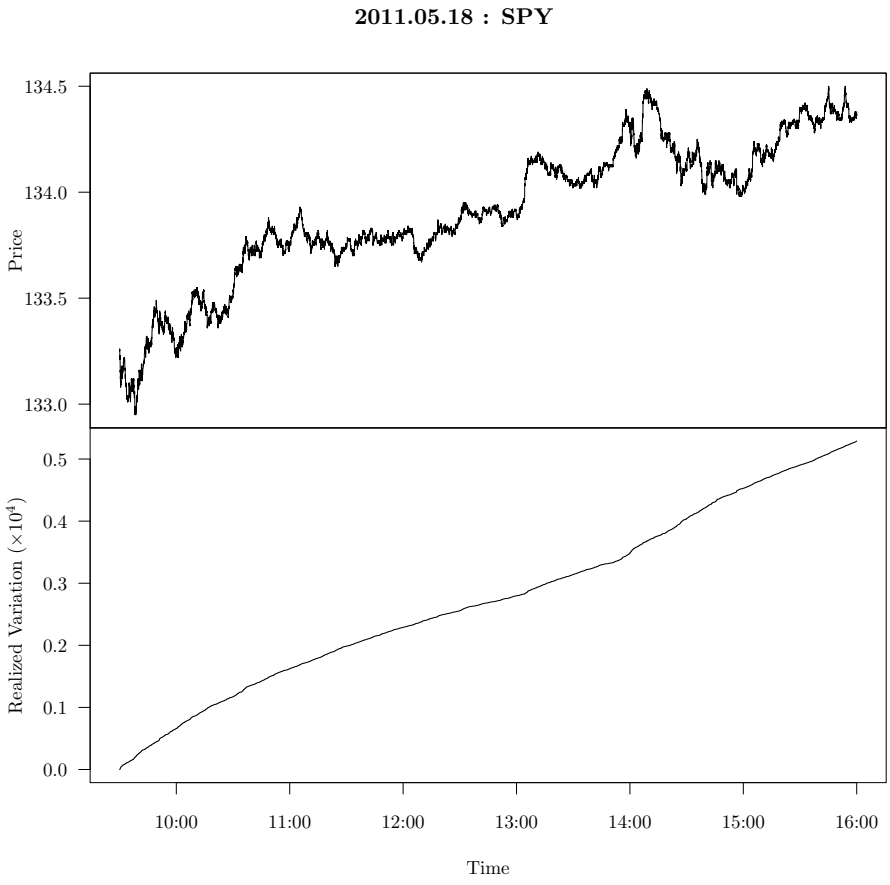

Figure 2.6: Price and Realized Variation for `SPY` on 2011.05.18 (*after* filtering)

Table 2.4: **Median** number of reduced prices removed per day

	2007	2008	2009	2010	2011
AA	4	5	2	1	1
AXP	3	5	3	1	1
BA	4	5	2	1	1
BAC	5	13	4	3	2
C	7	11	3	1	2
CAT	3.5	4	3	2	2
CSCO	6	4	2	2	2
CVX	6	6	4	2	2
DD	3	3	1	1	1
DIS	3	4	2	1	1
GE	4	9	3	2	2
HD	4	5	2	1	1
HPQ	5	7	4	2	2
IBM	6	6	3	2	1
INTC	5	3	2	2	2
JNJ	4	5	4	2	2
JPM	6	9	6	3	3
KFT	3	3	1	1	1
KO	3	5	2	1	2
MCD	3	6	3	1	1
MMM	3	3	2	1	1
MRK	4	6	6	2	2
MSFT	5.5	4	3	3	2
PFE	3	5	4	1	1.5
PG	4	7	4	2	2
T	5	8	2	1	2
TRV	2	3	2	1	0
UTX	2	3	2	1	1
WMT	5	8	4	2	2
VZ	3	7	3	1	2
XOM	9	8	6	4	2.5
SPY	17	16	11	13	16

Table 2.5: **Maximum** number of reduced prices removed per day

stock	2007	2008	2009	2010	2011
AA	61	89	125	42	19
AXP	44	84	79	28	17
BA	50	78	37	20	28
BAC	47	140	103	36	32
C	60	96	56	53	50
CAT	53	69	73	28	18
CSCO	47	53	19	20	23
CVX	58	121	50	30	45
DD	26	50	53	40	20
DIS	53	71	88	76	17
GE	43	185	141	52	26
HD	49	102	50	80	23
HPQ	47	122	95	149	17
IBM	59	79	47	18	22
INTC	29	42	15	10	16
JNJ	33	108	55	41	41
JPM	52	99	117	59	34
KFT	29	70	39	30	18
KO	23	129	68	36	36
MCD	26	127	59	39	27
MMM	39	80	46	20	12
MRK	46	127	75	44	17
MSFT	39	66	14	19	14
PFE	47	112	76	37	30
PG	32	106	88	33	27
T	44	85	67	43	21
TRV	28	44	54	13	7
UTX	28	50	77	22	11
WMT	34	88	76	20	30
VZ	28	61	117	40	19
XOM	85	94	88	39	38
SPY	134	107	136	96	47

2.4 Estimating intraday volatility

Recall once again equation 1.2 from Chapter 1

$$\mathbf{RV}^{\mathcal{G}^N}_{[\tau_{i-1},\tau_i]} = \sum_{t_j \in \mathcal{G} \cap [\tau_{i-1},\tau_i]} (\Delta X_{t_j})^2$$

which gives the definition of realized variation accumulated over the interval $[\tau_{i-1}, \tau_i]$. We now discuss our computational procedure for computing this realized variation using median-share prices defined on the grid of *active* time stamps. We use the `diff` and `cumsum` operators in **R** to compute realized variation for `price.data[["VW_PRICE"]]`.

```
## The log return on the first price is 0.
log.returns = c(0, diff(log(price.data[["VW_PRICE"]])))
delta.rv    = (log.returns)^2
price.data[["RV"]] = cumsum(delta.rv)
```

Let n denote the sample size, the number of median-share prices within a trading day. The `diff` operator takes a vector of size n, and returns a vector of consecutive differences of size $n - 1$. (Log returns are simply differences of log prices.) We prepend 0 to the log return vector `log.returns` for two reasons: (1) the `log.return` of the very first price is 0 (by convention), and (2) we wish to preserve the length of the vector. `delta.qv` is the vector (of size n) of increments to realized variation produced by that corresponding price. The `cumsum` operator takes a vector of size n, and returns a vector of size n of the partial cumulative sums. Hence `price.data[["RV"]]` is a vector of size n which contains the value of total realized variation at the corresponding time.

We illustrate with some some calculations for `SPY` on `2011.05.18` (We show only the 3 most relevant columns):

```
TIME  VW_MEDIAN  RV
34200 133.26     0.000000e+00
34201 133.25     5.631615e-09
34202 133.25     5.631615e-09
34203 133.26     1.126323e-08
34204 133.26     1.126323e-08
```

To compute the RV over an interval $[\tau_{i-1}, \tau_i]$, we simply subtract the RV for the last active time stamp before or equal to τ_{i-1} from the RV for the last active time stamp before or equal to τ_i. Fortunately, the `findInterval` function[12] in **R** allows us to easily compute the very last price before a given set of interval times. The following **R** code computes the log return and the realized variation for each interval.

[12] `findInterval(t,v)` is a built-in function in R-base which takes two arguments, a vector of "targets" **t** and a vector of values **v**. This vector of values **v** must be (weakly) increasing. The result is a vector of indices (into the vector of values **v**) which are the last indices which are equal to or less than the values in **t**. We will apply `findInterval` by setting **t** to be our interval boundary times and **v** as the observation times. The result vector will contain the indices to the last times (and hence the last prices) before or at the interval boundary times.

```
min.valid.second = 34200
max.valid.second = 57600
## Set interval.size to 100 for 100-second intervals
interval.size = 300

## This INCLUDES 34200
interval.boundary.times = seq(from = min.valid.second,
  to = max.valid.second, by = interval.size)

index.last.trades =
  findInterval(interval.boundary.times,
               price.data[["TIME"]])

## This is necessary!  (It handles the case when
## the first trade of the day occurs strictly after
## 34200).
index.last.trades[index.last.trades == 0] = 1

last.prices = with(price.data, PRICE[index.last.trades])
log.returns.intervals = diff(log(last.prices))
rv.intervals = with(price.data, diff(RV[index.last.trades]))
```

2.5 Filtering volatility jumps

We now give the definitions of filters 1 and 2 as implemented in EHLWZ (2012).

- $\bar{\zeta}$ is the median $\hat{\zeta}_i$ for intervals i within the *settled period* from 10:40 AM and 3:15 PM. This is our estimate of the equilibrium level of squared volatility.
- $\zeta^T = \bar{\zeta} + 3 \times (q_{0.75} - q_{0.25})$ is a threshold level, where $q_{0.75}$ and $q_{0.25}$ are the 75th and 25th percentiles of the realized variations during the settled period.
- τ^* is the end of the first interval i such that $\zeta_i \leq \zeta^T$ or the end of the first interval i within the settled period, whichever interval comes first.

Filter 1 identifies an interval i as a volatility jump if $\tau_i > \zeta^T$ and $\tau_i > \tau^*$. Filter 2 identifies an interval i as a volatility jump if $\zeta_{i-1} > \zeta^T$ and $\zeta_i > \zeta_{i-1}$. Below we present **R** code which implements these filters. First, we start with some rows of the `aggregate.analysis.df` data frame for our example day of `SPY` on `2011.05.18`.

INTERVAL	INTERVAL_START	INTERVAL_END	LAST_VW_PRICE	cum.RV
1	34200	34300	133.17	6.855112e-07
2	34300	34400	133.09	9.904259e-07
3	34400	34500	133.10	1.251076e-06
4	34500	34600	133.06	1.542313e-06
5	34600	34700	132.97	2.044087e-06

The column `cum.RV` is the accumulated realized variation at the end of the interval.[13]. Now we proceed :

[13]Basically, `cum.RV` is $\mathrm{RV}^{\mathcal{G}^N}_{[0,\tau_i]}$, the cumulative realized variation up to the end of the interval. In simpler terms, it is the sum of squared log returns by time stamps (namely, second by second) from the beginning of the day until the time stamp listed by `INTERVAL_END`.

```
## This corresponds to 10:40:00 and 15:15:00
begin.settled.region.time = 38400
end.settled.region.time   = 54900

aggregate.analysis.df[["RV"]] =
  diff(c(0, aggregate.analysis.df[["cum.RV"]]))

aggregate.analysis.df[["delta.RV"]] =
  diff(c(0, aggregate.analysis.df[["RV"]]))

settled.region = subset(aggregate.analysis.df,
  ((INTERVAL_START >= begin.settled.region.time) &
   (INTERVAL_END <= end.settled.region.time)))

zeta.bar = median(settled.region[["RV"]])

quartile.vector = as.double(quantile(settled.region[["RV"]],
  probs = c(0.25, 0.50, 0.75), na.rm = TRUE))

## The difference between the 75th percentile and the median ...
interquartile.value = quartile.vector[3] - quartile.vector[2]

zeta.threshold = zeta.bar + 3 * interquartile.value
```

Now that $\bar{\zeta}$ (`zeta.bar`) and ζ^T (`zeta.threshold`) have been identified, we compute the hitting time τ^* (`hitting.time`), and then create a column called `filter.status` which equals 0 if the interval is not a volatility jump, 1 or 2 if it fails filter 1 or filter 2, and equals 3 if it fails both filters. (Below, `lag.RV` for interval i is simply ζ_{i-1}, the realized variation of the previous interval.[14]

[14]Note that there were intervals without trades. For several stocks and several days in our data set, the first trade did not occur until after the first 100-second interval. However, there are examples of trade stoppages within the trading day. One curious example is `INTC` on January 31, 2011. In particular, after a median price of \$21.455 at 9:54:29, the next active second was 10:20:00 with a price of \$21.10. (Trading was halted, pending the announcement of a \$300 million reduction to revenue, after a design flaw was found in a product).

```
equilibrium.set.df = subset(aggregate.analysis.df,
  (RV <= zeta.threshold))

## This will work ... see below.
if (nrow(equilibrium.set.df) > 0){
  first.time.below.threshold =
     min(equilibrium.set.df[["INTERVAL_END"]])
} else {
  first.time.below.threshold = Inf
}

## R does the sensible thing ... min(v, Inf) = v for any finite
## value v.
hitting.time =
  min(begin.settled.region.time, first.time.below.threshold)

RV = aggregate.analysis.df[["RV"]]
RV.without.last = RV[-length(RV)]
aggregate.analysis.df[["lag.RV"]] = c(0, RV.without.last)

failed.filter.1 = with(aggregate.analysis.df,
  ((INTERVAL_START >= hitting.time) & (RV > zeta.threshold)))

failed.filter.2 = with(aggregate.analysis.df,
  ((lag.RV > zeta.threshold) & (delta.RV > 0)))

aggregate.analysis.df[["filter.status"]] =
  as.integer(failed.filter.1) + 2 * as.integer(failed.filter.2)
```

Here are some results of the computation. (We only show 2 columns)

```
> aggregate.analysis.df[1:10, c("INTERVAL", "filter.status")]
   INTERVAL filter.status
1         1             0
2         2             0
3         3             0
4         4             0
5         5             1
6         6             3
7         7             1
8         8             0
9         9             0
10       10             0
```

Intervals 5, 6, and 7 all failed filter 1, and interval 6 also failed filter 2.

2.6 Illustration

Figure 2.7 plots the 234 100-second realized variation estimates for `SPY` on May 18, 2007. This is the basic data used to estimate the Heston model in Chapter 3. For each of the 250 trading days in 2007 for each of the years from 2007 through 2009 we could display a plot such as this for `SPY`, and the same is true for each of the 32 assets in our sample.

Figure 2.8 plots the OLS regression line and the data points for SPY on the same day, May 18, 2007. In Chapter 3 we will examine regression results like this, except instead of a separate regression for each day in 2007, we look at a pooled regression for all of 2007.

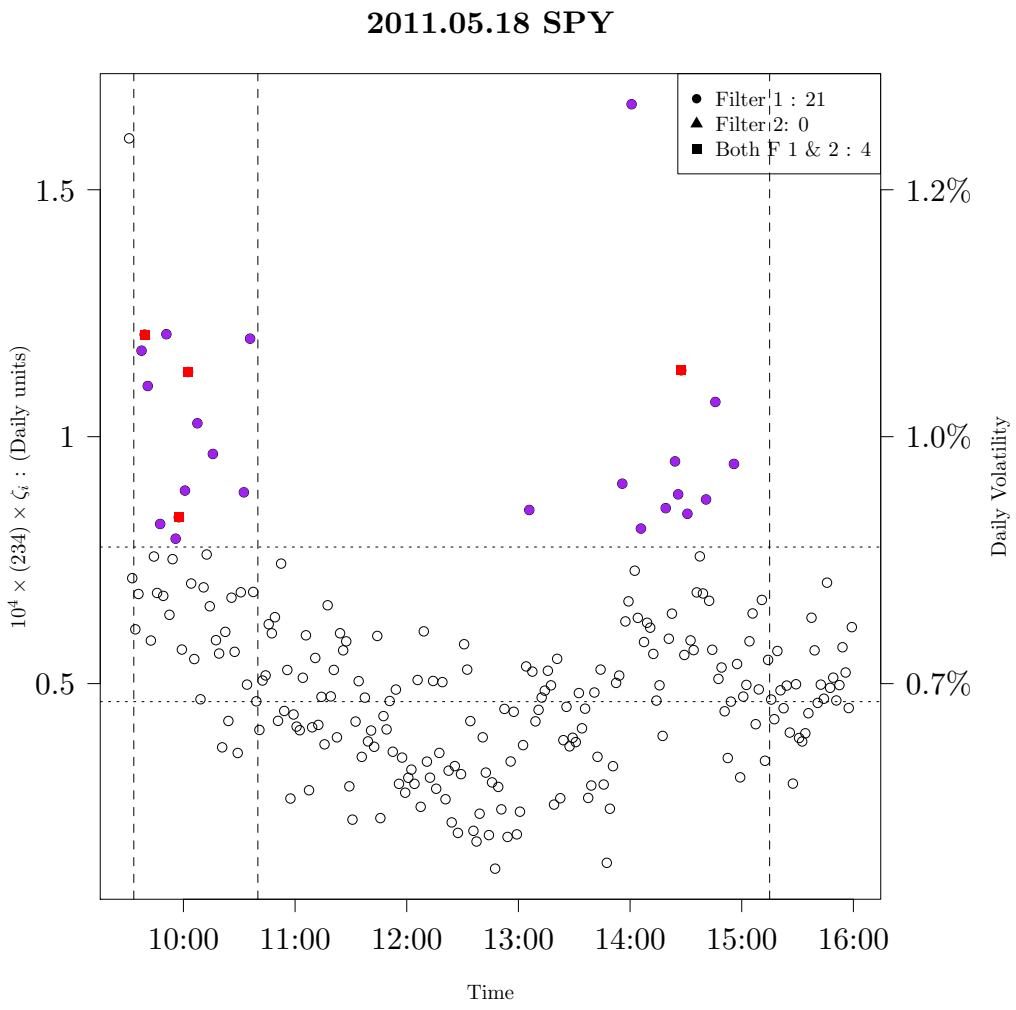

Figure 2.7: Realized Variation estimates: 100-second blocks

There were 25 volatility jumps (out of 234 intervals total), as denoted by the solid points on the graph. The left-side axis plots units of daily squared volatility times 10^4. (Namely, the interval RV is multiplied by 234 to compute the equivalent daily variation). The right-side axis show the equivalent daily volatility numbers, in daily % units. The settled level of squared volatility $\bar{\zeta} = 0.463$ ($\times 10^{-4}$) equals a settled value of daily volatility $\bar{\sigma} = \sqrt{\bar{\zeta}} = 0.680\%$. The threshold value $\zeta^T = 0.777 \times 10^{-4}$ equals a threshold daily volatility $\sigma^T = \sqrt{\zeta^T} = 0.881\%$.

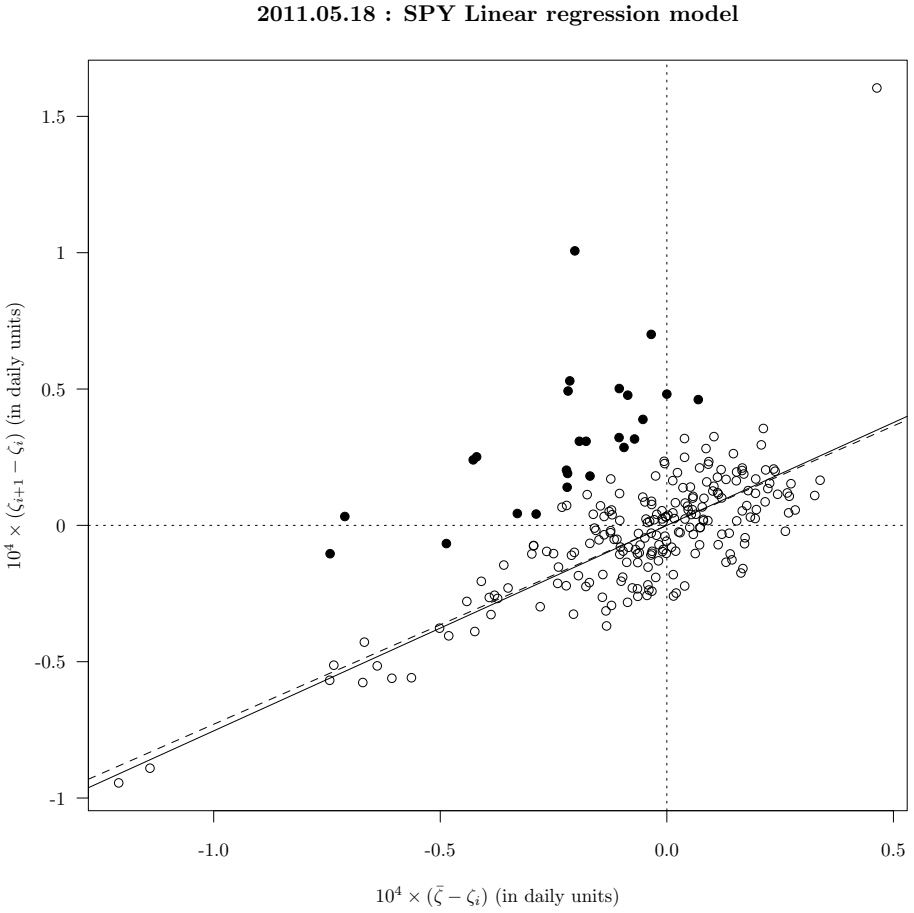

Figure 2.8: A daily regression estimate for SPY

The solid points represent the 25 points (out of 233) which were identified by filters 1 and 2. The solid line is the regression line for this day (without the solid points), and the dashed line is the annual regression line (computed after pooling all the intervals within the calendar year). The slopes for the daily and the annual regressions are 0.7538 and 0.7294 respectively. The R^2 values for the daily and annual regression lines are 0.5803 and 0.6417 respectively. (Note that for linear regressions without an intercept term, we do not have the usual interpretation of R^2 as the fraction of explained variance).

CHAPTER 3

Estimating the Heston model: 2007–2011

In Chapter 1 I described the results of estimating the Heston model of stochastic volatility using realized variation estimates for each 5-minute interval from the beginning of 2001 to the end of 2009 for every stock in the Dow Jones Industrial Average and for an exchange-traded fund (SPY) that tracks the S&P 500. Contrary to what the existing literature suggests should happen, our estimation procedure was very successful: the estimates of the mean-reversion parameter of the Heston model were high-significant for every asset for every year of our sample.

Nevertheless, the results of EHLWZ (2012) can be improved. In Chapter 2 I described an improved computational environment that makes this possible. In this chapter, I assess the consequences.

In Section 3.1 I make a direct comparison of the estimates of the Heston model using 100-second blocks rather than 300-second blocks for the realized-variation estimates within the trading days. Section 3.2 demonstrates that these new estimates can themselves be improved by eliminating a tiny fraction of 5-minute blocks from the yearly regressions. We find that our estimates of mean reversion are on average considerably higher with 100-second blocks than with 300-second blocks. In Section 3.3 we make this comparison more explicit.

3.1 The effects of improved resolution

Chapter 1 presented the estimates of the mean-reversion parameter β and its

t-statistic for each of the nine years of our sample for each of the 31 assets (the 30 Dow stocks and SPY), a total of 279 parameter estimates and 279 t-statistics. Because I have improved upon the identification of price anomalies, replaced the "last" price in each block with the median-share price and changed the sample period from 2001-2009, to make a direct assessment of the improvement we need to estimate the Heston model using 300-second blocks as well as estimating it for the higher-resolution 100-second blocks.

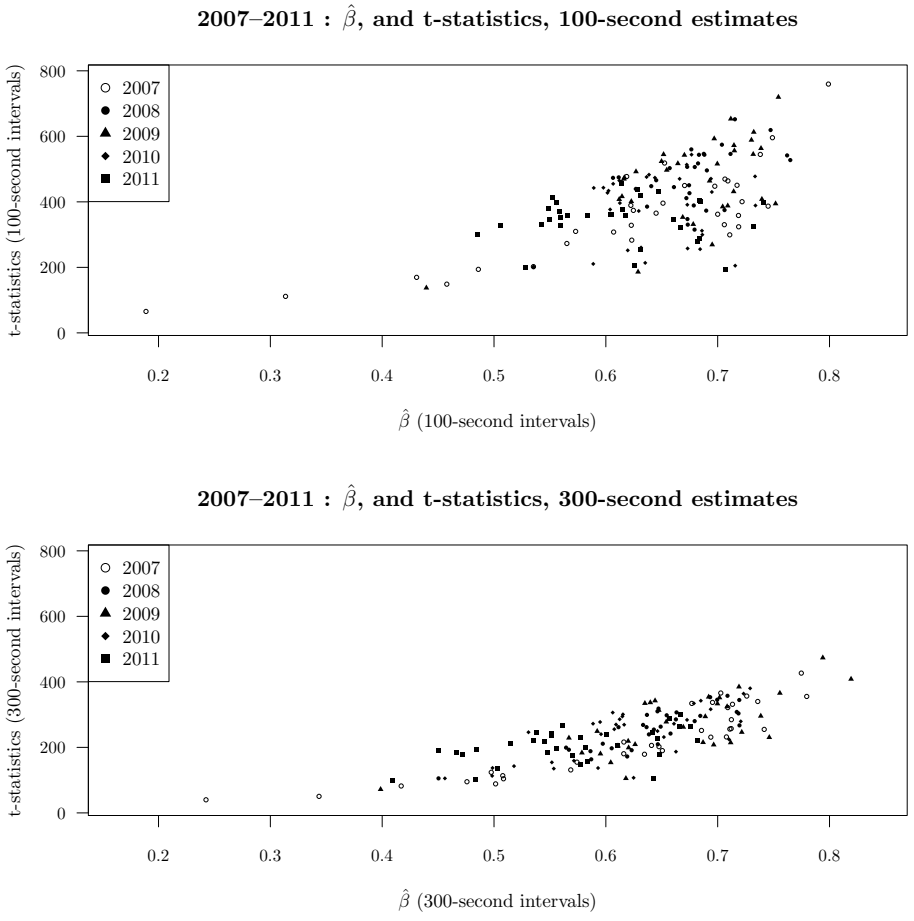

Figure 3.1: Comparing mean-reversion estimates and t statistics.

Rather than use tables to present the results (as we did in EHLWZ (2012) and in Chapter 1), here I plot the estimates and t-statistics in a single graph. The two panels of Figure 3.1 adopt the same format, plotting the estimated coefficient

β on the horizontal axis and the corresponding t-statistic on the vertical axis. The top panel plots the estimates and t-statistics for the regressions that use 100-second estimates of realized variation. The bottom panel plots the estimates and t-statistics for the regression that use 300-second estimates.[1] Data points correspond to a calculation for each asset-year, and hence there are 160 ($= 32 * 5$) points in each panel. Different plotting symbols are used to distinguish the 5 years of the sample. It is apparent from the graph that the distribution of points has shifted up and to the right: t-statistics tend to be even higher when the resolution is increased, and the degree of mean reversion is higher. Since in the top panel, β measures mean reversion over an interval one-third the length of mean reversion in the bottom panel, this is truly remarkable. Mean reversion is even more rapid than the very high value of mean reversion reported in EHLWZ (2012).

Figure 3.2 has a similar format except that we plot the standard error of the estimate of β rather than the t-statistic. Once again the top panel reports the results using 100-second blocks and the bottom panel the results for 300-second blocks. estimates. When resolution is increased, the distribution of points shifts down and to the right: the mean-reversion estimates tend to be higher and more precisely estimated.

In Figure 3.2, the five different years are marked with different symbols. Because it is difficult to distinguish the different years, in Figure 3.3 the plots for each of the years are plotted in separate panels. The scale of each axis in these panels is identical to the scale of the corresponding axis in Figure 3.2.

Each of the panels contains 32 data points, corresponding to each of our assets for a given year. Most of the data points appear as a small dot. However, 4 assets — `SPY` and the three financial assets `BAC` (Bank of America), `C` (Citigroup) and `JPM` (JP Morgan) — are each marked with a different symbol. `C` exhibits extremely large mean reversion, while the estimated mean reversion of `SPY` is relatively low

[1] We show the calculations after removing the 20 farthest outliers. See Section 3.2

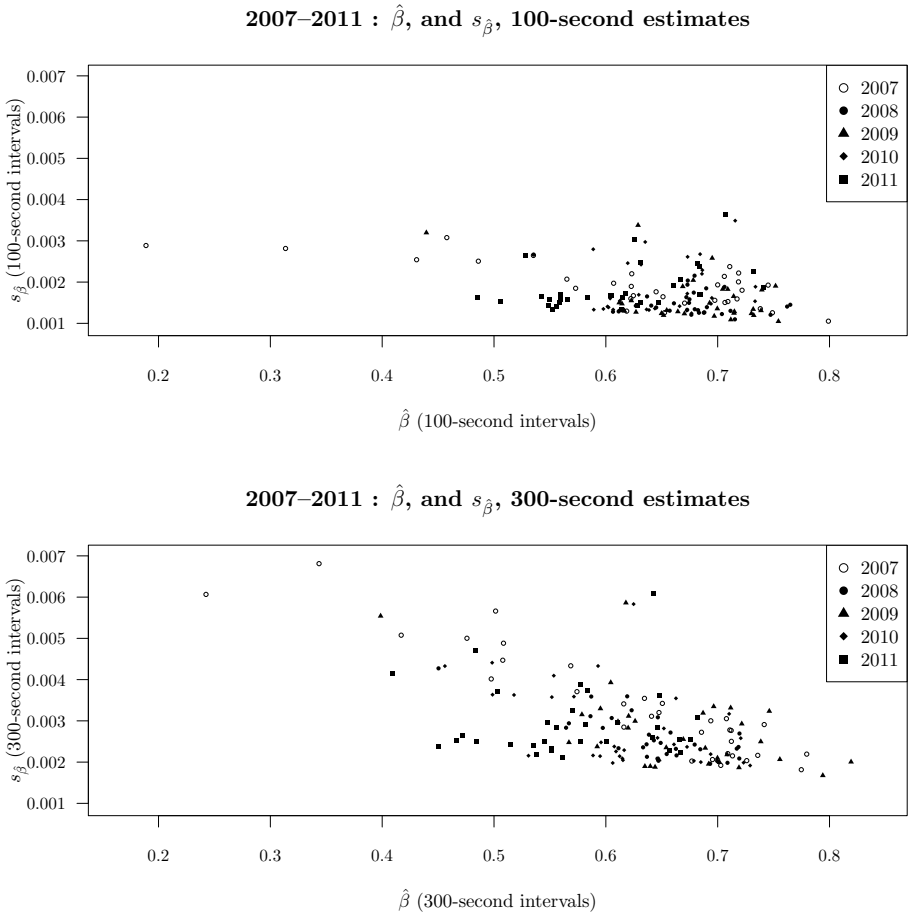

Figure 3.2: Comparing mean-reversion estimates and standard errors.

value.

In Figure 3.4 we compare the standard errors of $\hat{\beta}$ of β using 100-second blocks with the estimates using 300-second blocks. The solid line is the identity line, where $s_{\beta_{100}} = s_{\beta_{300}}$. As we can observe, nearly all points (except for one) lie above the line, indicating that we obtain **more** accurate estimates using **shorter** blocks! The dashed line is the line $y = x\sqrt{3}$, the relationship we would expect from a regression in which the number of observations has tripled with observations of the same quality. This is what we meant earlier when we said that the benefits of increasing resolution appear to more than compensate for the smaller "effective sample size" of the price observations within each block.

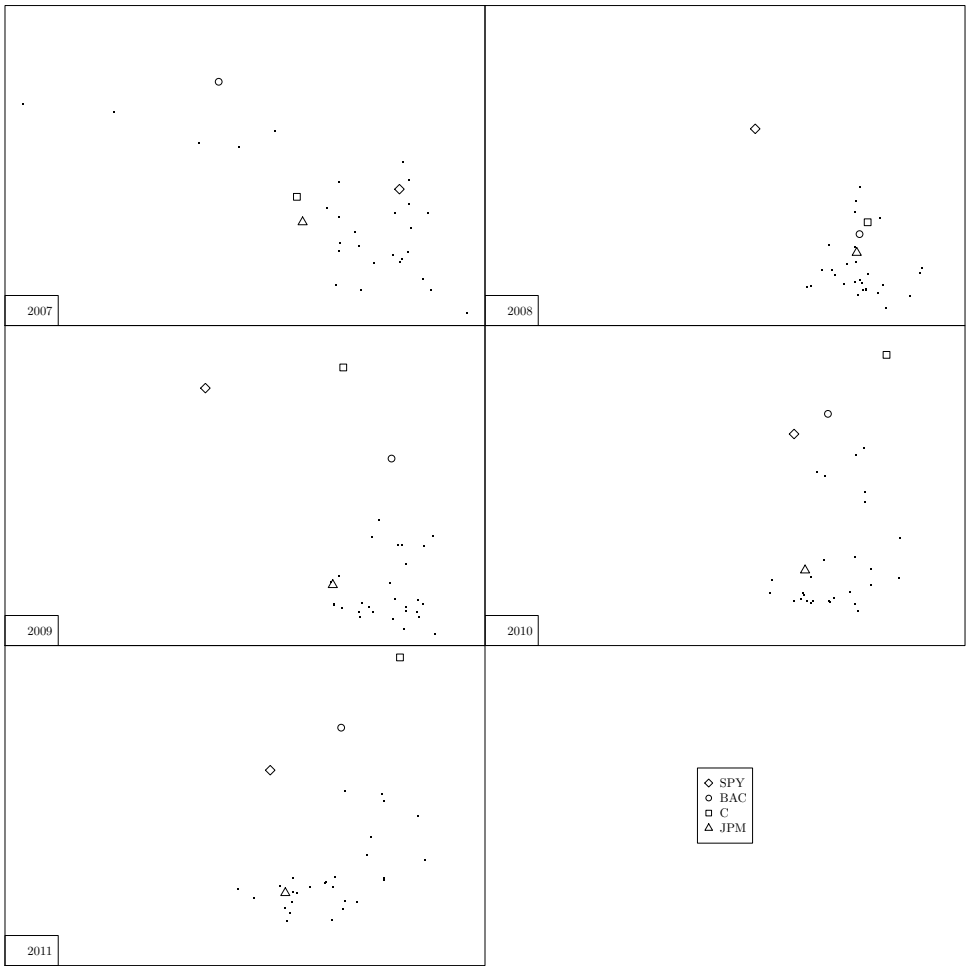

Figure 3.3: Comparing 100-second mean-reversion estimates and standard errors over time.

The horizontal axis is the value of $\hat{\beta}$, and the vertical axis is the standard error. All of the axes have the same scales as the aggregate version, Figure 3.2.

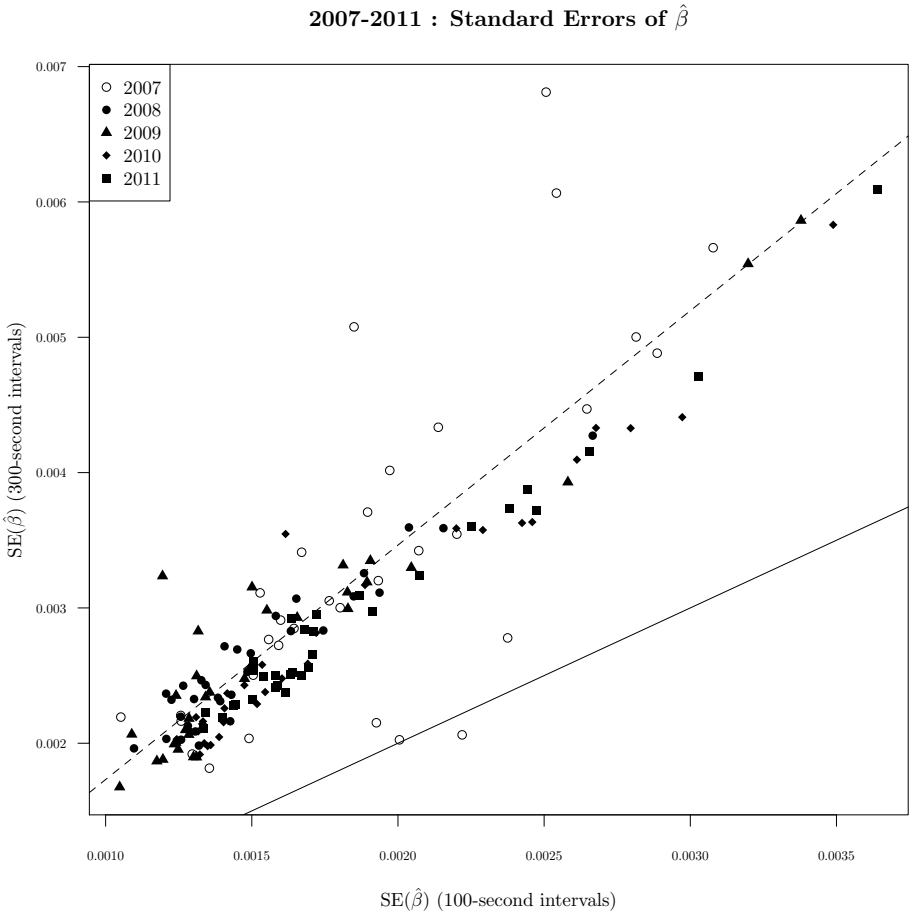

Figure 3.4: Standard Errors of $\hat{\beta}$ using 100-second intervals and using 300-second intervals

3.2 Deleting a few observations.

As discussed in Chapter 2, I have altered the procedure we used in EHLWZ (2012) to remove anomalous prices. The procedure I use eliminates more anomalous prices than EHLWZ (2012), but like EHLWZ (2012) the new procedure is still very conservative (i.e., it removes very few prices).

In this section, we explore the effects of removing some anomalous observations from our regressions. Each of our annual regressions for a specific stock contains approximately 58,280 observations ($= 233 * 250$, one observation for each of the 233 adjacent pairs of 100-second blocks in a trading day times 250 trading days in a year). For each of our assets for each of the five years of our sample, we identify the 20 observations that have the most influence on the estimate of the mean-reversion parameter β and delete them from the regression.

Figure 3.5 compares the estimated mean-reversion parameter $\hat{\beta}$ calculated with and without the 20 most influential observations. The graph plots 160 points, one for each of our regressions ($= 32 * 5$, one point for each asset for each year of the sample). On the horizontal axis is the estimate of β before excluding the 20 outliers from the regression. The vertical axis plots the estimate after the 20 most influential observations have been deleted. The solid line is the identity function, namely, the graph of $y = x$. If a point lies below the line, that means that deleting the 20 outliers from the regression lowered our estimate of mean reversion. Most of the points lie below the line.

For example, for `SPY` in 2010 twelve of the twenty outliers occurred on three days: May 6, 2010 (the day of the notorious Flash Crash), the day preceding the Flash Crash or the day after. These three days in May had a large impact on the estimate of β for SPY in 2010 and for many of the other assets in our sample, resulting in some cases in estimates of β close to one (its theoretical upper limit) for that year. This strongly suggests that the Flash Crash was truly an anomaly.

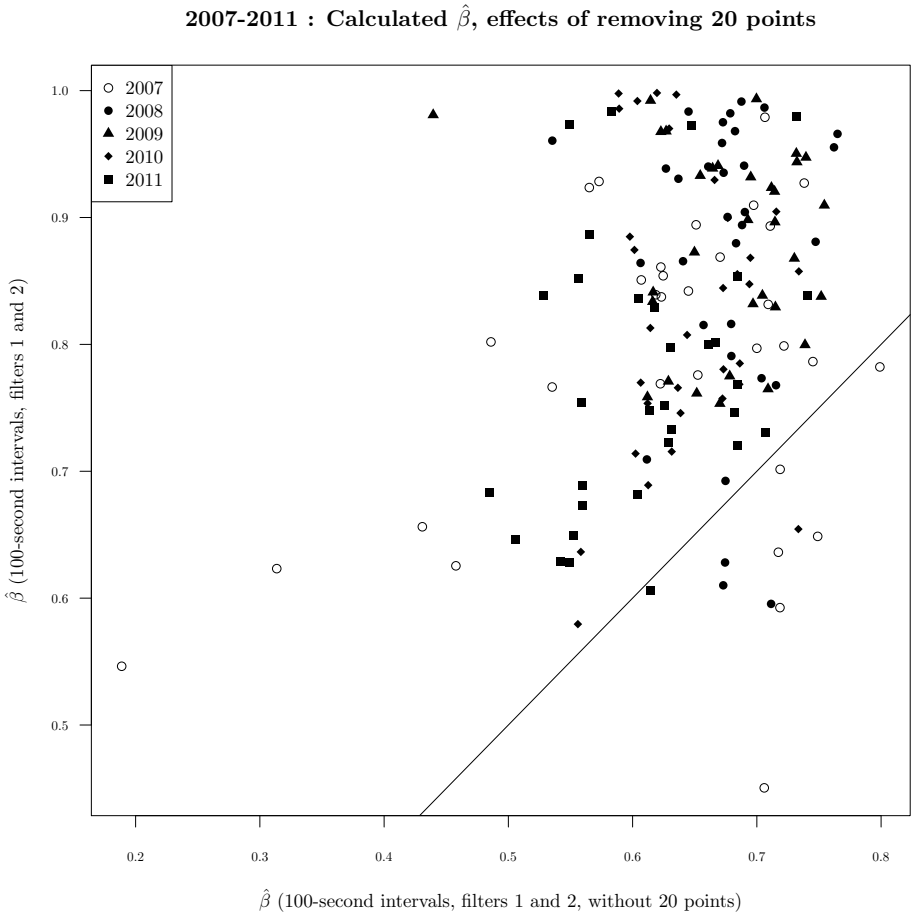

Figure 3.5: $\hat{\beta}$ estimates with and without 20 outliers

The solid line is the identity line $y = x$. Points above the line are stocks and years in which the 20 most distant points increased our estimated slope $\hat{\beta}$.

Figure 3.6 plots the annual regression for `SPY` (with 100-second blocks) in another year (2009), **before** eliminating the 20 outliers. The variables on both axes have been multiplied by 10^4 to improve readability.[2] In addition to the regression line, I have plotted the 95%-confidence bounds, and I have boxed the points which are outside the 95%-confidence band.

[2]In the regressions, realized variations are **not** scaled to a rate per trading day (i.e., they are not multiplied by $M = 234$.

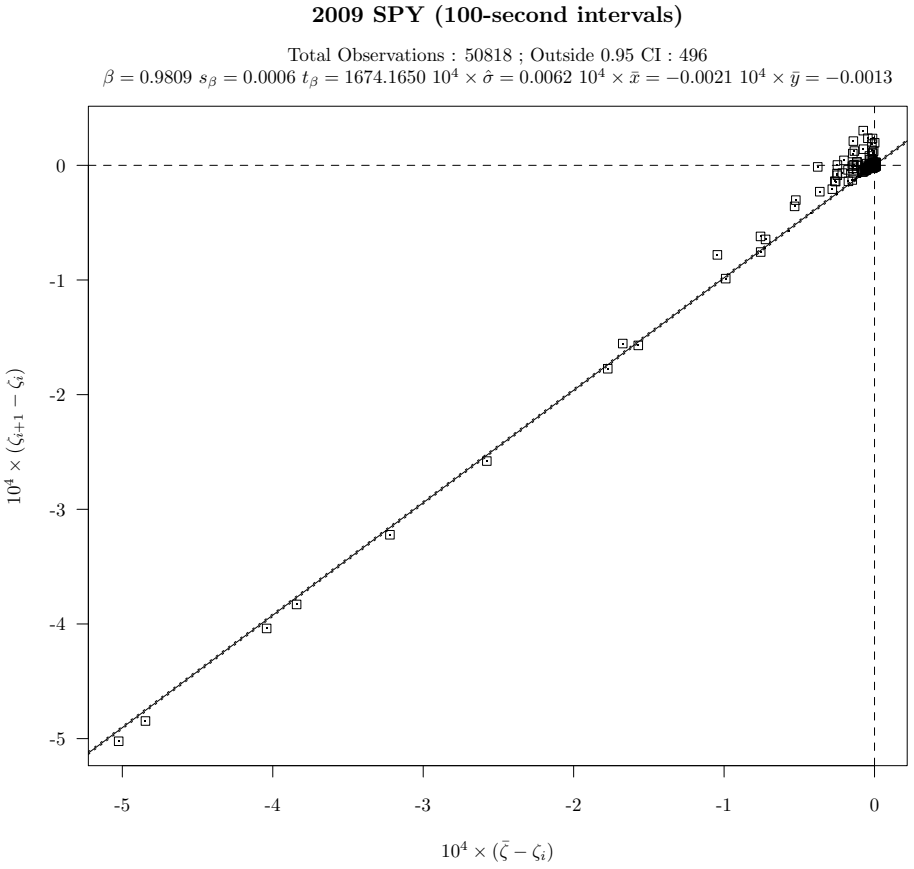

Figure 3.6: The annual regression for `SPY` in 2009 (100-second blocks).

After deleting volatility jumps, there were 50,818 observations in this regression. The solid line is the estimated linear regression. The estimated slope is $\hat{\beta} = 0.9809$, with a standard error of 0.0006 and a t-statistic of 1674.1650. Of the 50818 data points, 496 were outside of the $\pm 1.96\hat{\sigma}$ bounds, shown with dotted lines. Note that both $\hat{\beta}$ and the t-statistic are unreasonably high. Several extreme outliers far away from the origin exert an undue effect on the estimated slope coefficient and its t-statistic.[3]

We now eliminate the 20 points farthest from the origin, and show the result

[3]For example, the average horizontal and vertical values (for all 50818 points) are -0.0021 and -0.0013 respectively (after multiplication by 10^4 to have the same scaling as the plot). Points with horizontal or vertical values less than -1 are outliers.

in Figure 3.7. The estimated mean-reversion coefficient $\hat{\beta}$ decreased from 0.9809 to a much more reasonable value of 0.4395.

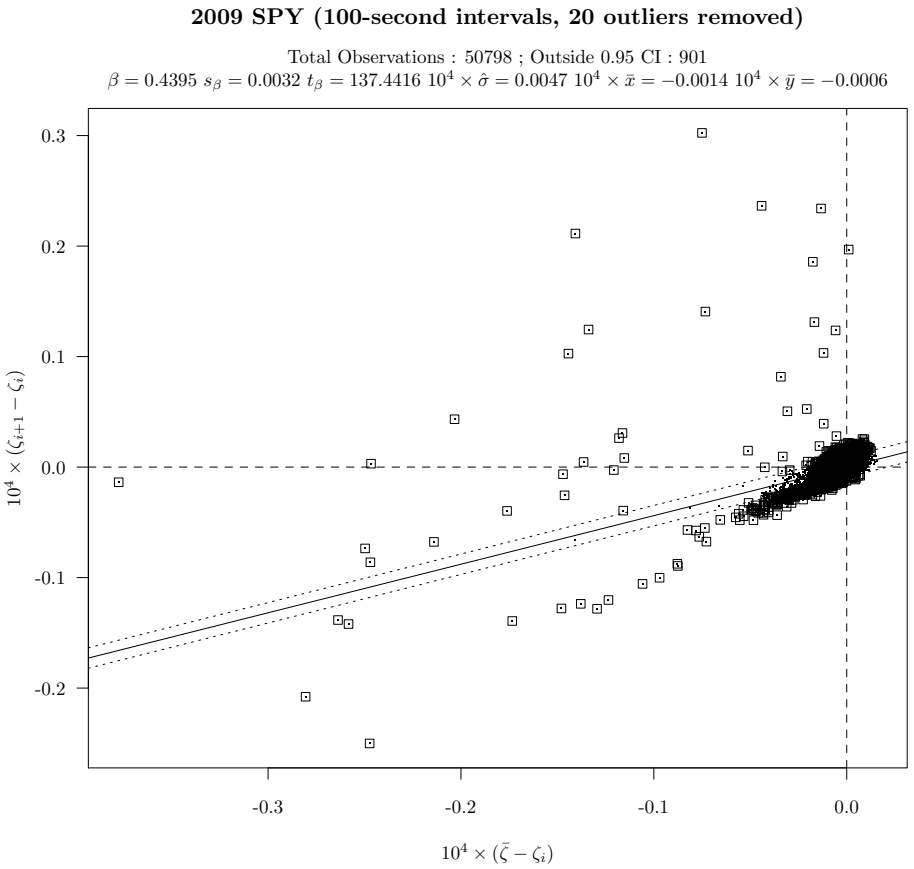

Figure 3.7: The annual regression for `SPY` in 2009 (100-second blocks), *without* 20 outliers.

3.3 How strong is mean reversion?

In Section 3.1 we noted that our estimates of mean reversion within a 100-second block are almost as large as the estimates we obtained in EHLWZ (2012) over 300-second blocks. This indicates a much faster speed of mean reversion. In this section, we make the comparison more explicit.

Recall the original Heston model from section 1.6 :

$$dX_t = \mu_t dt + \sqrt{\zeta_t} dW_t$$
$$d\zeta_t = \kappa(\bar{\zeta} - \zeta_t) + \gamma\sqrt{\zeta_t} dB_t$$

and the derived difference equation which we will use for linear regression, equation (1.3)

$$\zeta_{\tau_i} - \zeta_{\tau_{i-1}} = \beta(\bar{\zeta} - \zeta_{\tau_{i-1}}) + \varepsilon_{\tau_{i-1}} \tag{3.1}$$

where

$$\beta = 1 - e^{-\kappa h} \in (0, 1)$$
$$\varepsilon_{\tau_{i-1}} = \gamma e^{-\kappa h} \int_{\tau_{i-1}}^{\tau_i} e^{\kappa u} \sqrt{\zeta_u}\, dB_u$$

Here, $h = \Delta t$, the time in one of our equally spaced intervals. Our estimate of β is $\hat{\beta}$, the ordinary least squares (OLS) estimate. Note that κ is scale-invariant (it does not depend on our sampling frequency), but β does depend on the interval $h = \Delta t$.

Let Δ denote 100 seconds[4]. Let β_Δ denote the value of the mean-reversion coefficient when the block length is 100 seconds (corresponding to $M = 234$) and let $\beta_{3\Delta}$ denote the value when we use 300 second intervals (corresponding to $M = 78$). By definition,

$$\beta_\Delta = 1 - e^{-\kappa\Delta}$$
$$\beta_{3\Delta} = 1 - e^{-\kappa(3\Delta)}$$

[4]If we let $[0, 1]$ represent time within the trading day, then $\Delta = 100/23400 = 1/234$

Consequently,

$$\begin{aligned} e^{-3\kappa\Delta} &= 1 - \beta_{3\Delta} \\ e^{-\kappa\Delta} &= (1 - \beta_{3\Delta})^{1/3} \\ \beta_{\Delta} &= 1 - (1 - \beta_{3\Delta})^{1/3} \end{aligned}$$

Figure 3.8 plots $\hat{\beta}_{3\Delta}$ (the 300-second estimates) on the horizontal axis, $\hat{\beta}_{\Delta}$ (the 100-second estimates) on the vertical axis, and the corresponding comparison function $y = 1 - (1 - x)^{1/3}$. All calculations were done after removing the 20 outliers from each regression. We see that the 100-second estimates are almost always above the comparison curve. The estimates using 100-seconds suggest as much mean reversion occurs in 100 seconds as occurs over 300 seconds, which is astonishing. However, it must be noted that these estimates of mean-reversion apply to consecutive intervals in which the second interval is not a volatility jump. We will analyze the frequency of volatility jumps (as identified by filters 1 and 2) for both 100-second intervals and 300-second intervals in the next section.

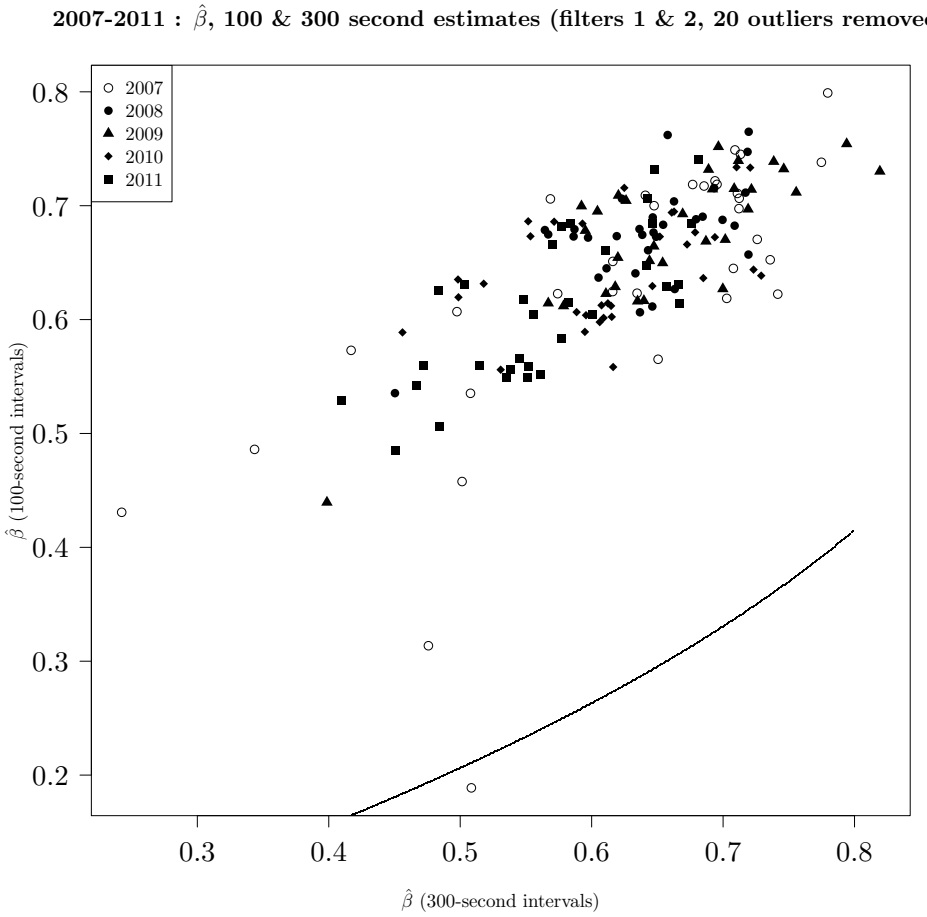

Figure 3.8: Comparing mean-reversion estimates with differing resolutions. The curve drawn is $y = 1 - (1 - x)^{1/3}$, which is the comparison between $\beta_\Delta = \beta_{100}$ (on the vertical axis) and $\beta_{3\Delta} = \beta_{300}$ (on the horizontal axis). (Recall that $\beta_\Delta = 1 - (1 - \beta_{3\Delta})^{1/3}$). Note that for all but 1 of the 160 points shown, the appropriately scaled speed of mean reversion for 100-second intervals is greater than the speed of mean reversion for 300-second intervals.

3.4 Volatility jumps

In our analysis we have applied the same volatility jump filters to our regressions using 100-second blocks that EHLWZ (2012) applied to the regressions using 300-second blocks. Figure 3.9 compares the fraction of intervals removed by filters 1 and 2 with 300 second blocks with the fraction removed with 100-second blocks.

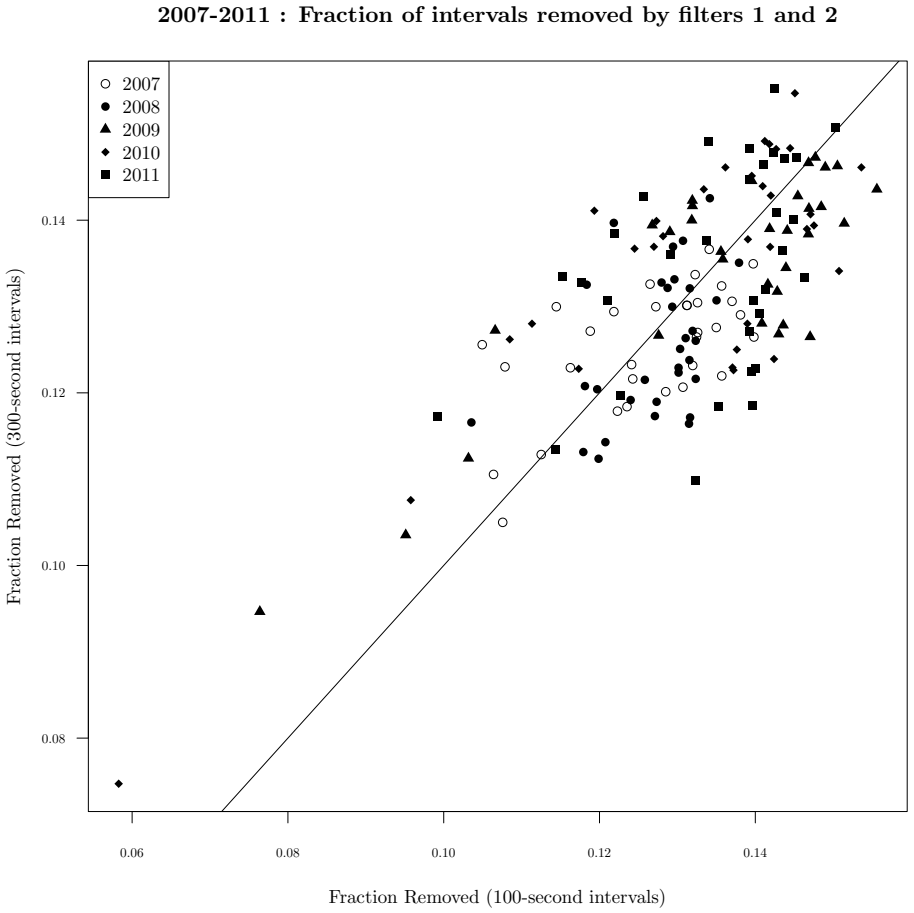

Figure 3.9: Comparing the fraction of intervals removed by filters 1 and 2. The solid line is the identity function $y = x$. Points above the line are stock-years in which filters 1 and 2 removed a greater fraction of the 300-second intervals than of the 100-second intervals.

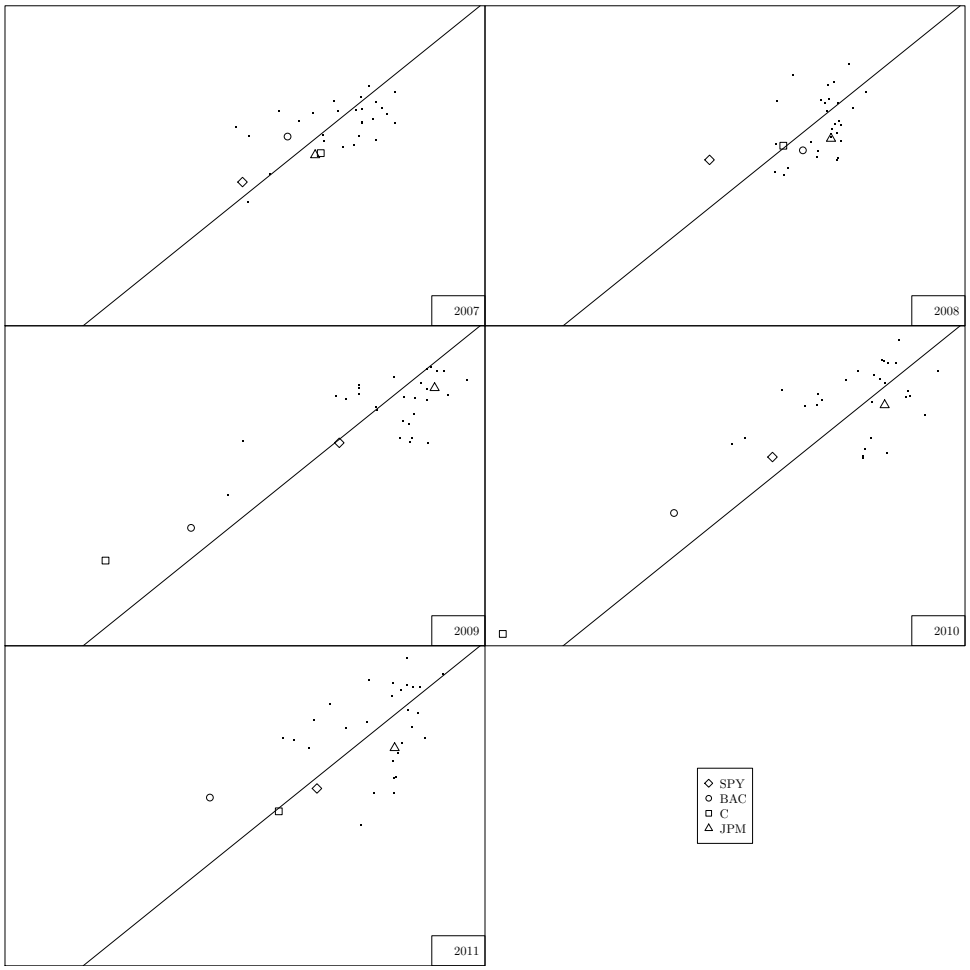

Figure 3.10: Comparing the fraction of intervals removed by filters 1 and 2, by year.

The plots contain 160 points (32 stocks times 5 years). The axes indicate the fraction of blocks that were removed. The fraction with 100-second blocks is on the horizontal axis, and the fraction with 300-second intervals on the vertical axis. The solid line is the graph of the identity, $y = x$, corresponding to equality of the fractions of observations removed from the regression (or, equivalently, the number of blocks flagged as containing a volatility jump). The numbers scale roughly in proportion to the number of blocks, and hence the fraction of blocks flagged as containing volatility jumps is about the same. Figure 3.10 provides a panel display of the same information by year. The axes have the same scale as Figure 3.9.

Section 3.4 shows the empirical distribution of the intervals flagged by our filters 1 and 2. The horizontal axis is expressed in units of extra daily volatility. The vertical axis shows the relative fraction of flagged intervals within that level of extra volatility.

Section 3.4 shows the time series of the empirical distribution of the intervals flagged by our filters 1 and 2. Both the horizontal and the vertical axes are the same as the previous figure.

We can observe that for every year, our filters 1 and 2 flag more 300-second intervals than 100-second intervals (as a relative fraction of the total number of intervals) at the lowest level (volatility jumps of less than 0.5%). At higher levels of excess volatility, our filters identify relatively more 100-second intervals than 300-second intervals as volatility jumps.

In general, the fractions are decreasing as jump levels increase, with the notable exception for 100-second intervals in 2008 and 2009. In those years, there are *more* intervals between 0.5% and 1% (daily volatility) than between 0 and 0.5%.

We will place these levels in perspective by looking at the time series of daily ζ_1 (the sum of realized variation over the day) as well as the daily $\bar{\zeta}$.

Figure 3.13 shows the values for ζ_1 (total realized variation times 10^4) and $\bar{\zeta}$ (our estimate of the settled variation for each trading day over the period 2007–2011. The axis on the right shows the daily volatility numbers in the units of percentage volatility per day. (For example, a value of 4 on the left axis represents a value of $\zeta = 4 \times 10^{-4}$, which corresponds to a daily volatility of $\sigma = \sqrt{\zeta} = 2 \times 10^{-2}$ or a 2% daily volatility). The dotted vertical lines on the lower panel correspond to the FOMC (Federal Reserve Open Market Committee) announcement dates during our analysis period from 2007 to the end of 2011. We can note the substantial time variation both in realized variation (squared volatility) and in our estimates of $\bar{\zeta}$ (squared settled volatility).

Hence, volatility jumps in excess of 1% daily volatility would represent a very large fraction of the total daily realized variation for most of the trading days in our sample.

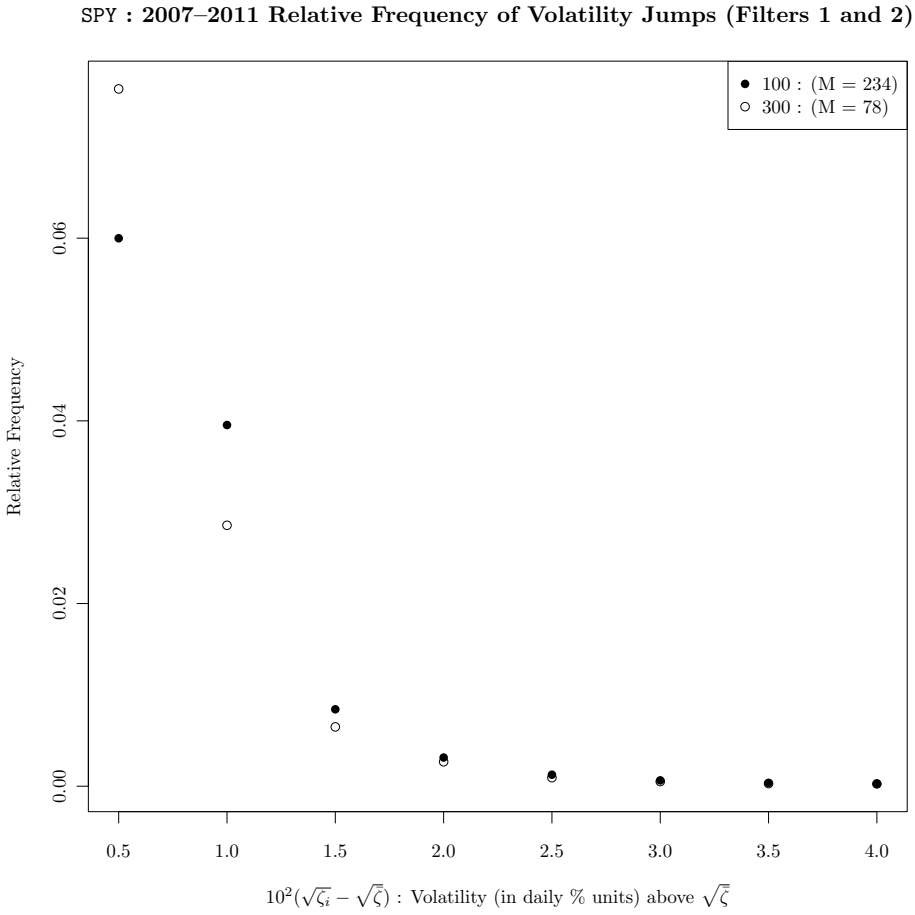

Figure 3.11: Relative Frequency of Volatility Jumps (identified by filters 1 and 2) The horizontal axis represents the extra daily volatility (in percentage units) above the equilibrium level of volatility. The vertical axis is the fraction of intervals which have that extra level of volatility (between the value and the previous value). For example, about 6% (respectively 7.6%) of all 100-second (respectively 300-second) intervals were flagged by filters 1 and 2 and had a level of ζ_i which represented between 0 and 0.5 percent level of daily volatility. The corresponding frequencies for intervals with between 0.5 and 1.0 percent extra daily volatility were 3.9% (for 100-second intervals) and 2.8% (for 300-second intervals).

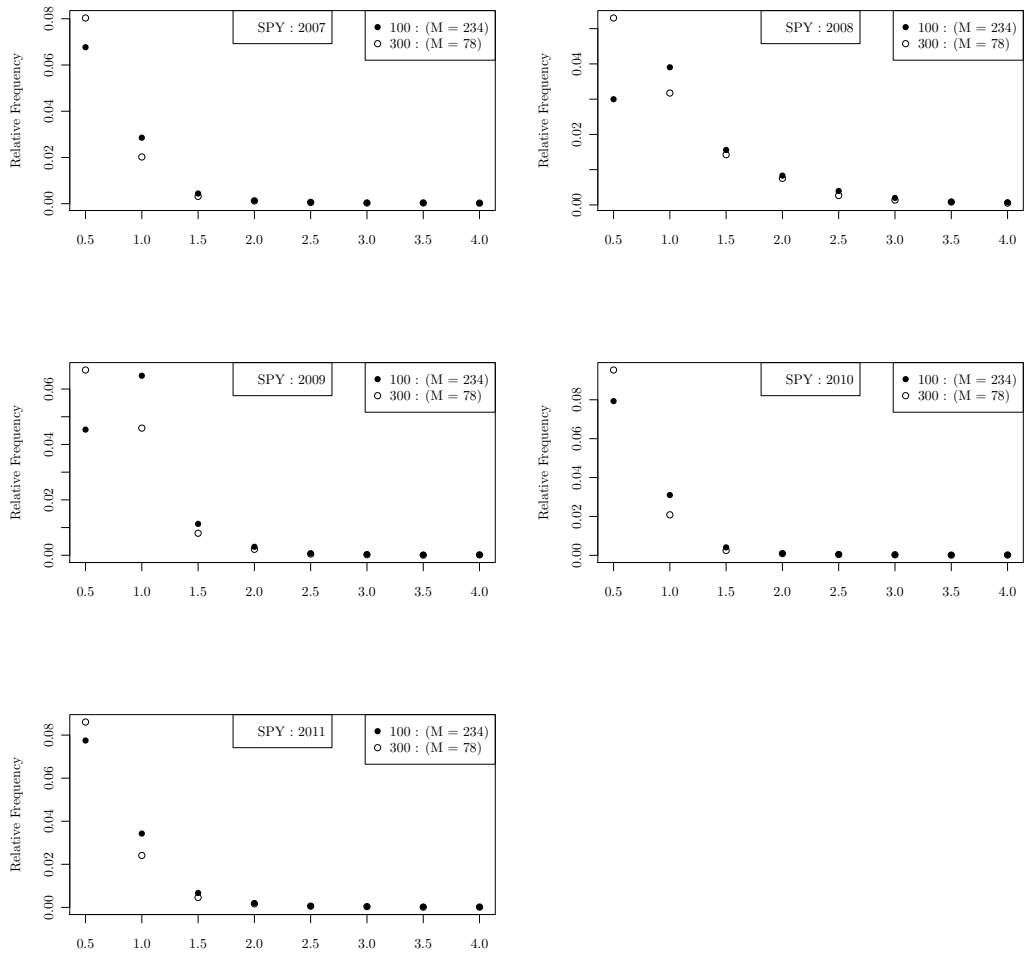

Figure 3.12: Relative Frequency of intervals identified by filters 1 and 2, by year See Section 3.4 for the definitions of the axes. Note that the scales of the vertical axis change from year to year.

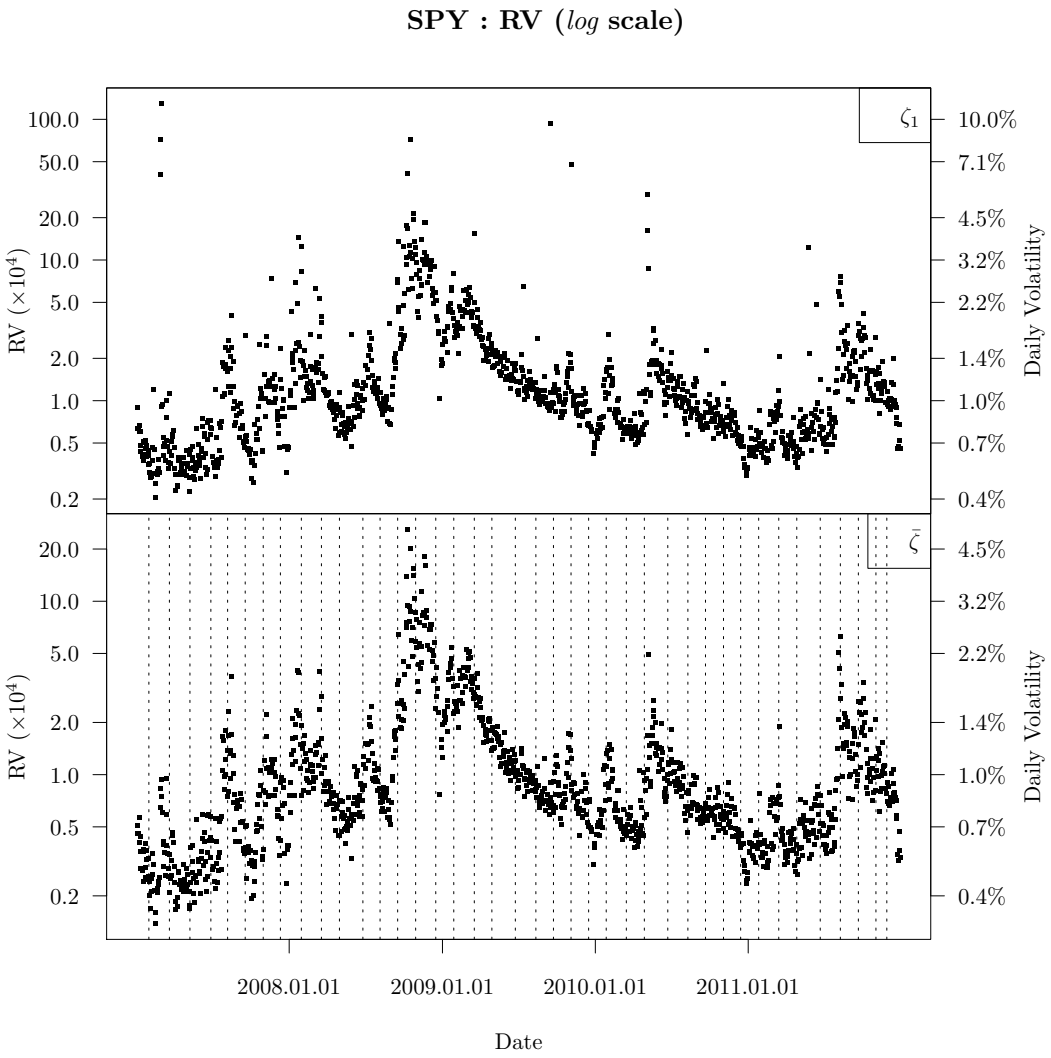

Figure 3.13: ζ_1 and $\bar{\zeta}$ for `SPY`, 2007–2011

ζ_1 is the total daily realized variation, and $\bar{\zeta}$ is the total daily settled variation. Equivalent daily volatility (in percentage units per day) is shown on the axis on the right. The vertical lines correspond to FOMC announcement dates.

3.5 Conclusion

In this work, we have refined the analysis of EHLWZ (2012) in several ways. We used the tools and the techniques presented in Chapter 2 to handle the challenge of an increasing number of transactions. We compute median prices for each *active* time stamp, and then apply a price filter based on the marginal contribution to realized variation, the *influence* of a price. We filter out outlier prices which have a very influence on the realized variation within a local window. Both of these techniques allow us to construct a more economically meaningful series of price observations.

More importantly, we switch from the 300-second block resolution to the 100-second block resolution in order to analyze the last 5 years through 2011 for our set of 32 stocks. Several features of our analysis are noteworthy :

- After we pool our data for all the interval in a year, apply our filters 1 and 2, and remove the 20 furthest outliers, we obtain estimates of mean reversion ($\hat{\beta}$) with extremely large t-statistics, which improve when we switch from 300-second intervals to 100-second intervals. All but 9 of the 160 t-statistics exceed 200. (`AXP` (American Express) in 2007 produced the smallest t-statistic (65.4) for the estimated $\hat{\beta}$, and $\hat{\beta}$ has an asymptotic 95% confidence interval of $0.189 \pm 1.96(0.003) = [0.183, 0.195]$). A surprising finding is that the standard errors of our estimated slope coefficients are comparable after we divide by $\sqrt{3}$, which would result if we triple the number of observations, but kept the observations of the same quality.

- Our estimates $\hat{\beta}$ of mean reversion increase at the finer resolution of 100-seconds, after we scale both $\hat{\beta}_{\Delta}$ (the 100-second estimate) and $\hat{\beta}_{3\Delta}$ to make the estimates comparable.

- The relative frequency of volatility jumps, as identified by our filters 1 and 2,

remains comparable. As we would expect, after switching to the more local time scale of 100-second intervals, we detect a higher fraction (of the total number of intervals) of extreme volatility jumps and a smaller fraction of small jumps (after the appropriate scaling to daily levels).

- The performance of our Heston model linear regressions are substantially improved after we removed a relatively small number of points from out data set (namely 20 outliers from among annual regression data sets of over 50,000 observations).

The availability of software tools, high frequency data, and recent advances in statistical inference all allow a greater study of continuous-time models of price and volatility processes. This research validates Merton's conjecture that high-frequency data can contribute significantly to our understanding of financial markets.

References

[ABD99] Torben G. Andersen, Tim Bollerslev, Francis X. Diebold, and Paul Labys. "Realized Volatility and Correlations." (Published in revised form as "Great Realizations" in Risk in March 2000), 1999.

[AMZ05] Yacine Aït-Sahalia, Per Mykland, and Lan Zhang. "How Often to Sample a Continuous-Time Process in the Presence of Market Microstructure Noise." *Review of Financial Studies*, **18**:351–416, 2005.

[BS73] Fischer Black and Myron Scholes. "The Pricing of Options and Corporate Liabilities." *The Journal of Political Economy*, pp. 637–654, 1973.

[BS01] Ole E. Barndorff-Nielsen and Neil Shephard. "Non-Gaussian Ornstein-Uhlenbeck-based models and some of their uses in Financial Economics." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **63**(2):167–241, 2001.

[CIR85] John C. Cox, Jonathan E. Ingersoll, and Stephen A. Ross. "A Theory of the Term Structure of Interest Rates." *Econometrica*, **53**:385–407, 1985.

[DS94] Freddy Delbaen and Walter Schachermayer. "A General Version of the Fundamental Theorem of Asset Pricing." *Mathematische Annalen*, **300**(1):463–520, 1994.

[DS98] Freddy Delbaen and Walter Schachermayer. "The Fundamental Theorem of Asset Pricing for Unbounded Stochastic Processes." *Mathematische Annalen*, **312**(2):215–250, 1998.

[DS05] Freddy Delbaen and Walter Schachermayer. *The Mathematics of Arbitrage*. Springer-Verlag, 2005.

[EHL12] Bryan Ellickson, Benjamin Hood, Tin Shing Liu, Duke Whang, and Peilan Zhou. "Stocks in the Short Run." 2012.

[Fel51] William Feller. "Two Singular Diffusion Problems." *The Annals of Mathematics*, **54**(1):173–182, 1951.

[Hes93] Steven Heston. "A Closed-Form Solution for Options with Stochastic Volatility with Aplications to Bond and Currency Options." *The Review of Financial Studies*, **6**(2):327–343, 1993.

[HK79] J. Michael Harrison and David M. Kreps. "Martingales and Arbitrage in Multiperiod Securities Markets." *Journal of economic theory*, **20**(3):381–408, 1979.

[HL06a] Peter R. Hansen and Asger Lunde. "Realized Variance and Market Microstructure Noise." *Journal of Business and Economic Statistics*, **24**(2):127–161, 2006.

[HL06b] Peter R. Hansen and Asger Lunde. "Rejoinder : Realized Variance and Market Microstructure Noise." *Journal of Business and Economic Statistics*, **24**(2):208–218, 2006.

[Hoo11] Benjamin Hood. *Essays on Intraday Stock Price Volatility*. PhD thesis, UCLA, 2011.

[HP81] J. Michael Harrison and Stanley R. Pliska. "Martingales and Stochastic Integrals in the Theory of Continuous Trading." *Stochastic Processes and their Applications*, **11**(3):215–260, 1981.

[JS03] Jean Jacod and Albert N. Shiryaev. *Limit Theorems for Stochastic Processes*. Springer-Verlag, Second, second edition, 2003.

[Kre81] D.M. Kreps. "Arbitrage and Equilibrium in Economies with Infinitely Many Commodities." *Journal of Mathematical Economics*, **8**(1):15–35, 1981.

[Liu11] Tin Shing Liu. *Stock Price Volatility within a Trading Day*. PhD thesis, UCLA, 2011.

[Mer73] Robert C. Merton. "Theory of Rational Option Pricing." *The Bell Journal of Economics and Management Science*, pp. 141–183, 1973.

[Mer80] Robert C. Merton. "On Estimating the Expected Return on the Market: An exploratory investigation." *Journal of Financial Economics*, **8**:323–361, 1980.

[MZ09] Per Mykland and Lan Zhang. "Inference for Continuous Semimartingales observed at High Frequency." *Econometrica*, **77**(5):1403–1445, 2009.

[MZ12] Per Mykland and Lan Zhang. "The Econometrics of High Frequency Data." In Mathieu Kessler, Alexander Lindner, and Michael Sorensen, editors, *Statistical Methods for Stochastic Differential Equations*, chapter 2. Chapman & Hall, CRC Press, 2012.

[Oom06] Roel C. A. Oomen. "Properties of Realized Variance under Alternative Sampling Schemes." *Journal of Business and Economic Statistics*, **24**(2):219–237, 2006.

[Pro04] Philip Protter. *Stochastic Integration and Differential Equations*. Springer-Verlag, 2004.

[Ros76] Stephen A. Ross. "The Arbitrage Theory of Capital Asset Pricing." *Journal of Economic Theory*, **13**(3):341 – 360, 1976.

[Sam65] Paul A. Samuelson. "Proof that Properly Anticipated Prices Fluctuate Randomly." *Industrial management review*, **6**(2):41–49, 1965.

[Shr04] Steven E. Shreve. *Stochastic Calculus for Finance : II.* Springer-Verlag, 2004.

[Tay05] Stephen J. Taylor. *Asset Price Dynamics, Volatility and Prediction.* Princeton University Press, 2005.

[Wha12] Duke Whang. "**R** programming techniques." 2012.

[Zho07] Peilan Zhou. *Essays on Financial Asset Return Volatility.* PhD thesis, UCLA, 2007.